Click here for production status of specific part numbers.

MAX34451

PMBus 16-Channel V/I Monitor and 12-Channel Sequencer/Marginer

General Description

The MAX34451 is a power-supply system manager that is capable of monitoring up to 16 different voltage rails or currents and is also capable of sequencing and margining up to 12 power supplies. The system manager monitors the power-supply output voltages and currents and constantly checks them for user programmable over and under threshold limits. If a fault is detected, the device automatically shuts down the system in an orderly fashion. The device can sequence the supplies in any order at both power-up and power-down. The device has the ability to close-loop margin the power-supply output voltages up or down to a user-programmable level. The device contains an internal temperature sensor and can support up to four external remote temperature sensors. Once configured, the device can operate autonomously without any host intervention.

Applications

- Network Switches/Routers
- Base Stations
- Servers
- Smart Grid Network Systems

PMBus is a trademark of SMIF, Inc.

***Ordering Information* and *Typical Operating Circuit* appear at end of data sheet.**

Benefits and Features

- Integration Enables Management of Multiple Power Supplies to Maximize System Performance
  - 16 Channels of Voltage or Current Monitoring
  - 12 Channels of Sequencing and Margining (8 PWM, 4 External Current DACs (1 x DS4424), and Sequencing
  - Expandable Channel Operation with Parallel Devices
  - Remote Ground Sensing Improves Measurement Accuracy
  - Programmable Up and Down Time-Based or Event-Based Sequencing
  - Dual Sequencing Loops
  - Configurable Combinatorial Logic Supporting Up to 16 GPIs and 20 GPOs
  - Automatic Closed-Loop Margining
  - No External Clocking Required
  - PMBus™-Compliant Command Interface
- Fast, Reliable Control and Fault Detection Improves System Reliability
  - Fast Minimum/Maximum Threshold Excursion Detection
  - Supports Up to 5 Temperature Sensors (1 Internal and 4 Remote)
  - Fault Detection on All Temperature Sensors
  - Reports Peak, Minimum, and Average Levels for a Number of Parameters
  - Programmable Alarm Outputs
  - On-Board Nonvolatile Black Box Fault Logging and Default Configuration Setting
- I^2C-/SMBus-Compatible Serial Bus with Bus Time-Out Function Simplifies Additional Temperature Sensors and DACs to the MAX34451
- +3.0V to +3.6V Supply Voltage

19-6642; Rev 9; 10/19

maxim integrated™

TABLE OF CONTENTS

TABLE OF CONTENTS (continued)

TABLE OF CONTENTS (continued)

TABLE OF CONTENTS (continued)

LIST OF FIGURES

LIST OF TABLES

LIST OF TABLES (continued)

Absolute Maximum Ratings

V_{DD} and V_{DDA} to V_{SS} .....-0.3V to +4.0V
RSG0 and RSG1 to V_{SS}.....-0.3V to +0.3V
All Other Pins Except REG18
Relative to V_{SS}.....-0.3V to (V_{DD} + 0.3V)*
REG18 to V_{SS}.....-0.3V to +2.0V

**Subject to not exceeding +4.0V.*

Continuous Power Dissipation (T_A = +70°C)
TQFN (derate 27.8mW/°C above +70°C).....2222.2mW
Operating Temperature Range..... -40°C to +85°C
Storage Temperature Range..... -55°C to +125°C
Lead Temperature (soldering, 10s) .....+260°C
Soldering Temperature (reflow).....+260°C

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Recommended Operating Conditions

(T_A = -40°C to +85°C, unless otherwise noted.)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
V_{DD} Operating Voltage Range	V_{DD}	(Note 1)	3.0		3.6	V
Input Logic 1 (Except I²C and GPIn Pins)	V_{IH1}		0.7 x V_{DD}		V_{DD} + 0.3	V
Input Logic 0 (Except I²C and GPIn Pins)	V_{IL1}		-0.3		+0.3 x V_{DD}	V
Input Logic 1: SCL, SDA, MSCL, MSDA	V_{IH2}		2.1		V_{DD} + 0.3	V
Input Logic 0: SCL, SDA, MSCL, MSDA	V_{IL2}		-0.3		+0.8	V
Input Logic 1 (GPIn Pins)	V_{IH3}	Minimum pulse width 5ms	1.5		V_{DD} + 0.3	V
Input Logic 0 (GPIn Pins)	V_{IL3}	Minimum pulse width 5ms	-0.3		+1.0	V
Source Impedance to RSn		ADC_TIME[1:0] = 00			1	kΩ
		ADC_TIME[1:0] = 01			5	
		ADC_TIME[1:0] = 10			10	
		ADC_TIME[1:0] = 11			20	
V_{DD} Rise Time		From 0V to 3.0V			4	ms
V_{DD} Source Impedance					10	Ω

Package Information

PACKAGE TYPE: 56 TQFN	
Package Code	T5677+2
Outline Number	21-0144
Land Pattern Number	90-0043

For the latest package outline information and land patterns (footprints), go to www.maximintegrated.com/packages. Note that a "+", "#", or "-" in the package code indicates RoHS status only. Package drawings may show a different suffix character, but the drawing pertains to the package regardless of RoHS status.

Electrical Characteristics

(V_{DD} and V_{DDA} = 3.0V to 3.6V, T_A = -40°C to +85°C, unless otherwise noted. Typical values are at V_{DD}/V_{DDA} = 3.3V, T_A = +25°C.) (Note 2)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
GENERAL						
Supply Current	I_{CPU}	(Note 3)		12		mA
	$I_{PROGRAM}$			18		
System Clock Error	$f_{ERR:MOSC}$	+25°C < T_A < +85°C	-3		+3	%
		-40°C < T_A < +25°C	-4		+4	
Output Logic-Low (Except I²C Pins)	V_{OL1}	I_{OL} = 4mA (Note 1)			0.4	V
Output Logic-High (Except I²C Pins)	V_{OH1}	I_{OH} = -2mA (Note 1)	V_{DD} - 0.5			V
Output Logic-Low: SCL, SDA, MSCL, MSDA	V_{OL2}	I_{OL} = 4mA (Note 1)			0.4	V
SCL, SDA, MSCL, MSDA Leakage	IL_{I2C}	V_{DD} = 0V or unconnected			±5	µA
CONTROL0 Threshold				2.048		V
CONTROL0 Hysteresis				50		mV
ADC						
ADC Bit Resolution				12		Bits
ADC Conversion Time		ADC_TIME[1:0] = 00		1000		ns
ADC Full Scale	V_{FS}	T_A = 0°C to +85°C	2.032	2.048	2.064	V
ADC Measurement Resolution	V_{LSB}			500		µV
RSn Input Capacitance	C_{RS}			15		pF
RSn Input Leakage	IL_{RS}	0V < V_{RSn} < 2.1V		±0.25		µA
ADC Integral Nonlinearity	INL			±1		LSB
ADC Differential Nonlinearity	DNL			±1		LSB
TEMPERATURE SENSOR						
Internal Temperature-Measurement Error		T_A = -40°C to +85°C		±2		°C

Electrical Characteristics (continued)

(V_{DD} and V_{DDA} = 3.0V to 3.6V, T_A = -40°C to +85°C, unless otherwise noted. Typical values are at V_{DD}/V_{DDA} = 3.3V, T_A = +25°C.) (Note 2)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
FLASH						
Flash Endurance	N_{FLASH}	Note 3	20,000			Write Cycles
Data Retention		T_A = +50°C (Note 4)	100			Years
STORE_DEFAULT_ALL, MFR_STORE_ALL Write Time				80		ms
RESTORE_DEFAULT_ALL		With MFR_STORE_SINGLE data		105		ms
RESTORE_DEFAULT_ALL or MFR_RESTORE_ALL		Without MFR_STORE_SINGLE data		500		µs
MFR_STORE_SINGLE Write Time				310		µs
MFR_NV_FAULT_LOG Write Time		Writing 1 fault log		11		ms
MFR_NV_FAULT_LOG Delete Time		Deleting all fault logs		200		ms
MFR_NV_FAULT_LOG Overwrite Time				40		ms
TIMING OPERATING CHARACTERISTICS						
Round-Robin Voltage and Current Sample Rate		Threshold excursion (Note 5)		64		µs
		Data collection		5		ms
Temperature Sample Rate				1000		ms
Device Startup Time		With MFR_STORE_SINGLE data		170		ms
		Without MFR_STORE_SINGLE data		90		
PWM Frequency		PWM power-supply margining		312.5		kHz
PWM Resolution		PWM power-supply margining		8		Bits

Note 1: All voltages are referenced to ground. Current entering the device are specified as positive and currents exiting the device are negative.

Note 2: Limits are 100% production tested at T_A = +25°C. Limits over the operating temperature range and relevant supply voltage range are guaranteed by design and chacterization.

Note 3: This does not include pin input/output currents.

Note 4: Guaranteed by design.

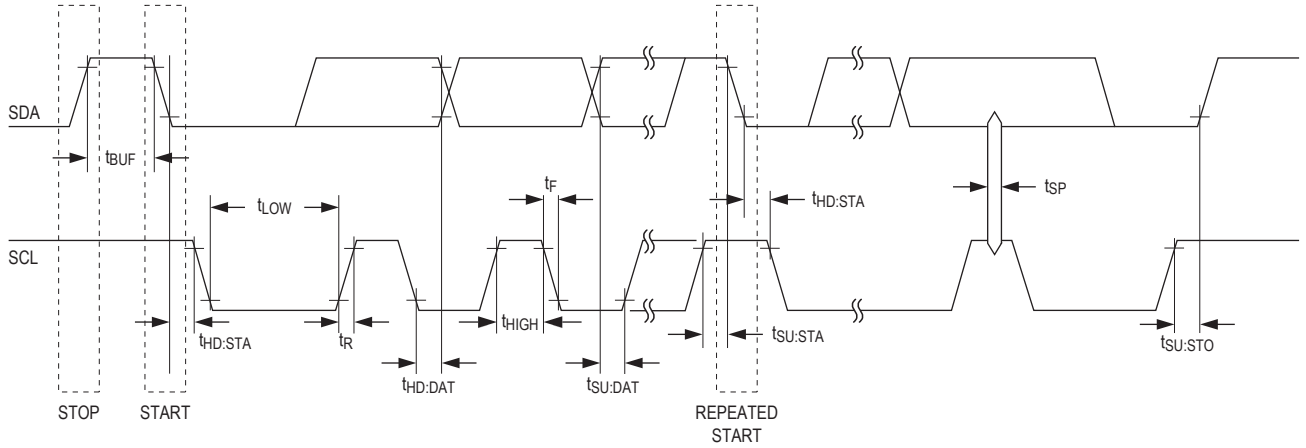
Note 5: The round-robin threshold excursion rate can be changed with the ADC_AVERAGE and ADC_TIME bits in MFR_MODE from 16µs (no averaging and 1µs conversion) to 1024µs (8x averaging and 8µs conversion).

I²C/SMBus Interface Electrical Specifications

(V_{DD} and V_{DDA} = 3.0V to 3.6V, T_A = -40°C to +85°C, unless otherwise noted. Typical values are at V_{DD}/V_{DDA} = 3.3V, T_A = +25°C.)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
SCL Clock Frequency	f_{SCL}		10		400	kHz
MSCL Clock Frequency	f_{MSCL}			100		kHz
Bus Free Time Between STOP and START Conditions	t_{BUF}		1.3			µs
Hold Time (Repeated) START Condition	$t_{HD:STA}$		0.6			µs
Low Period of SCL	t_{LOW}		1.3			µs
High Period of SCL	t_{HIGH}		0.6			µs
Data Hold Time	$t_{HD:DAT}$	Receive	0			ns
		Transmit	300			
Data Setup Time	$t_{SU:DAT}$		100			ns
Start Setup Time	$t_{SU:STA}$		0.6			µs
SDA and SCL Rise Time	t_R				300	ns
SDA and SCL Fall Time	t_F				300	ns
Stop Setup Time	$t_{SU:STO}$		0.6			µs
Clock Low Timeout	t_{TO}		25	27	35	ms

I²C/SMBus Timing

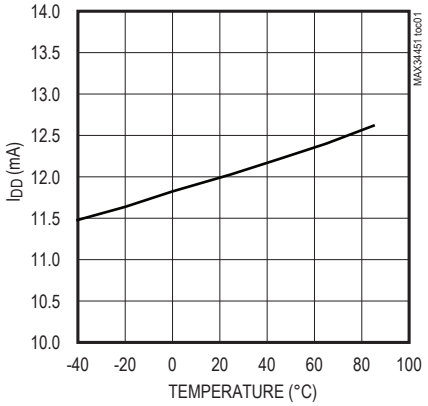


NOTE: TIMING IS REFERENCED TO $V_{IL(MAX)}$ AND $V_{IH(MIN)}$.

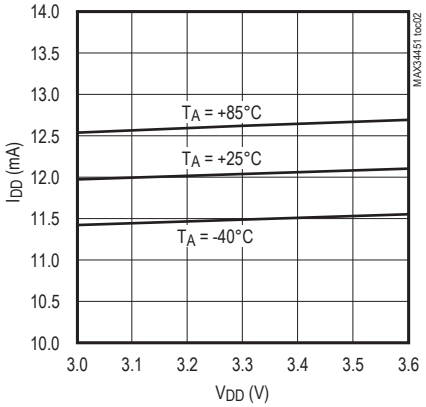
Typical Operating Characteristics

(V_{DD} = 3.3V and T_A = +25°C, without MFR_STORE_SINGLE data, unless otherwise noted.)

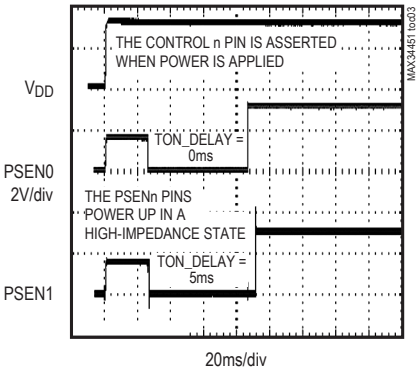
SUPPLY CURRENT vs. TEMPERATURE



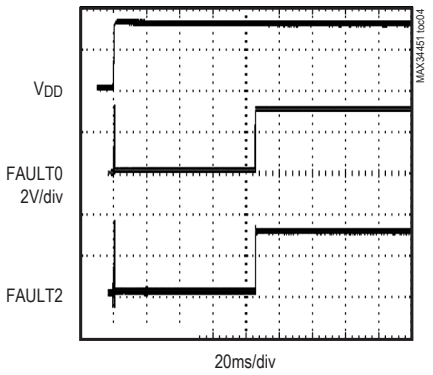
SUPPLY CURRENT vs. SUPPLY VOLTAGE



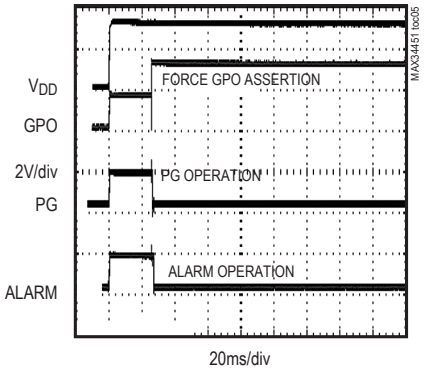
PSENn OUTPUTS DURING POWER-UP



$\overline{\text{FAULT}}$ PINS DURING POWER-UP



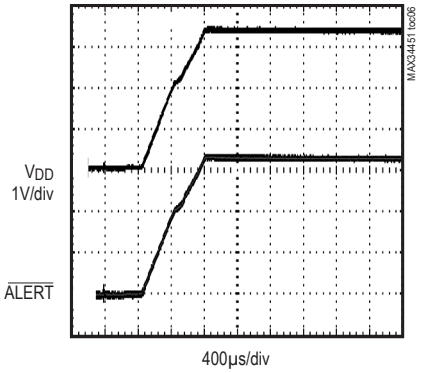
GPOn OUTPUT PINS CONFIGURED DURING POWER-UP (ALL PINS CONFIGURED TO BE PUSH-PULL ACTIVE-HIGH)



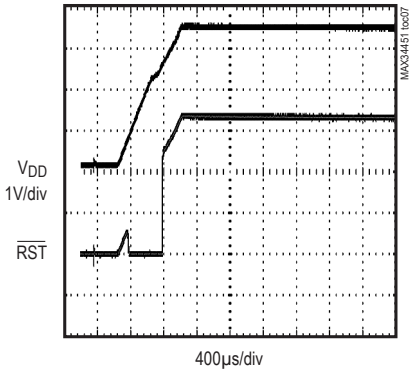
Typical Operating Characteristics (continued)

(V_{DD} = 3.3V and T_A = +25°C, without MFR_STORE_SINGLE data, unless otherwise noted.)

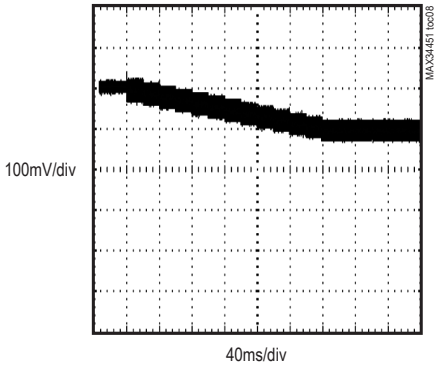
$\overline{ALERT}$ PIN DURING POWER-UP



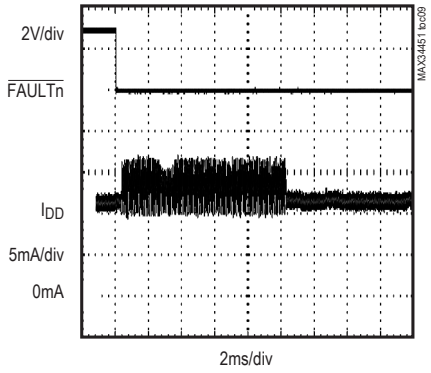
$\overline{RST}$ PIN DURING POWER-UP



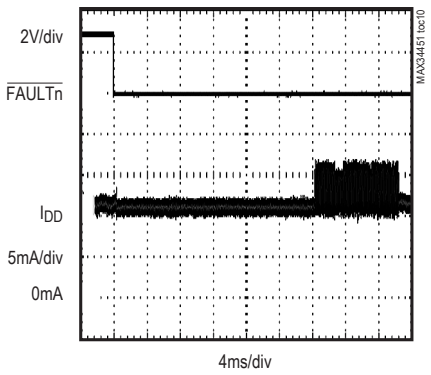
FILTERED MARGINING VOLTAGE vs. TIME



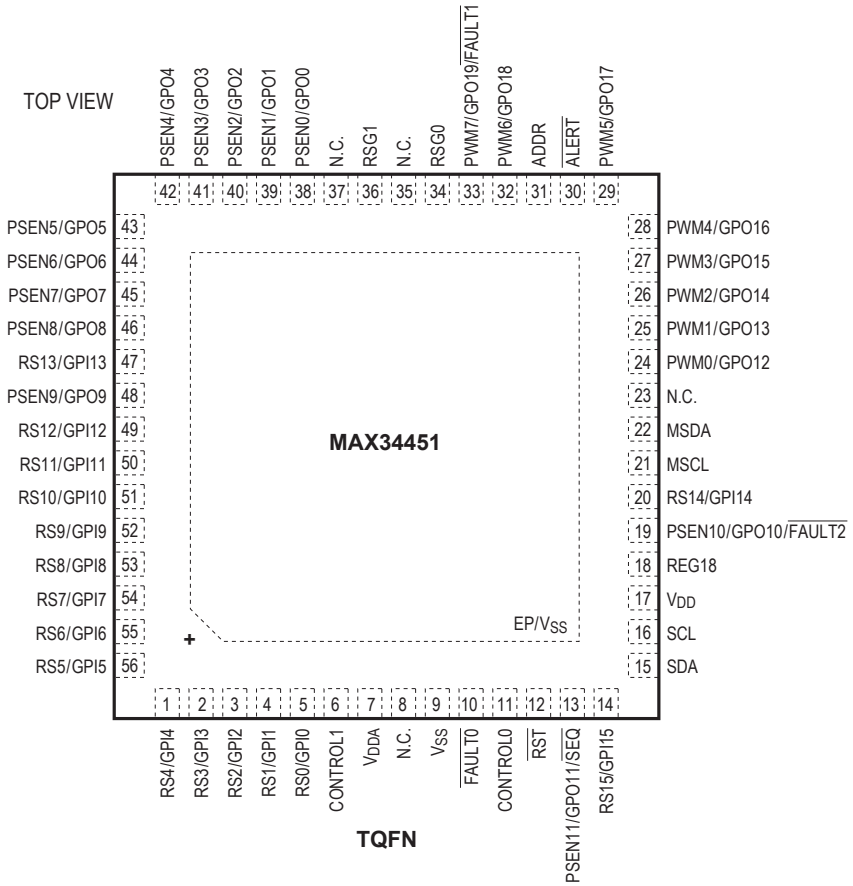
I_{DD} vs. TIME DURING A NONVOLATILE LOG WRITE



I_{DD} vs. TIME DURING A NONVOLATILE LOG WRITE WITH OVERWRITE ENABLED



Pin Configuration



Pin Description

PIN*	NAME	TYPE**	FUNCTION
1	RS4	AI	ADC Voltage-Sense Input 4. Connect to V_{SS} if unused.
	GPI4	AI	General-Purpose Input 4. Connect to V_{SS} if unused.
2	RS3	AI	ADC Voltage-Sense Input 3. Connect to V_{SS} if unused.
	GPI3	AI	General-Purpose Input 3. Connect to V_{SS} if unused.
3	RS2	AI	ADC Voltage-Sense Input 2. Connect to V_{SS} if unused.
	GPI2	AI	General-Purpose Input 2. Connect to V_{SS} if unused.
4	RS1	AI	ADC Voltage-Sense Input 1. Connect to V_{SS} if unused.
	GPI1	AI	General-Purpose Input 1. Connect to V_{SS} if unused.
5	RS0	AI	ADC Voltage-Sense Input 0. Connect to V_{SS} if unused.
	GPI0	AI	General-Purpose Input 0. Connect to V_{SS} if unused.
6	CONTROL1	DI	Power-Supply Master On/Off Control Input 1. Active low or active high based on ON_OFF_CONFIG command. Connect to V_{SS} if unused.

Pin Description (continued)

PIN*	NAME	TYPE**	FUNCTION
7	V_{DDA}	Power	Analog Supply Voltage. Bypass V_{DDA} to V_{SS} with 0.1µF. Connect to V_{DD}.
8	N.C.	—	No Connection. Do not connect any signal to this pin.
9	V_{SS}	Power	Ground Reference. Must be connected to EP (exposed pad).
10	$\overline{FAULT0}$	DIO	Fault Input/Output 0. Open-drain, active-low I/O. See the *Expanded Pin Description* section for more details.
11	CONTROL0	AI	Power-Supply Master On/Off Control Input 0. Active low or active high based on ON_OFF_CONFIG command. Connect to V_{SS} if unused.
12	$\overline{RST}$	DIO	Active-Low Reset Input/Output. Contains an internal pullup.
13	PSEN11	DO	Power-Supply Enable 11. See the *Expanded Pin Description* section for more details.
	GPO11	DO	General-Purpose Output 11
	$\overline{SEQ}$	DIO	Sequencing Input/Output. Open-drain, active-low I/O. This pin is used as a handshake signal to coordinate sequencing in systems using multiple devices.
14	RS15	AI	ADC Voltage-Sense Input 15. Connect to V_{SS} if unused.
	GPI15	AI	General-Purpose Input 15. Connect to V_{SS} if unused.
15	SDA	DIO	I^2C/SMBus-Compatible Input/Output. Open-drain output.
16	SCL	DIO	I^2C/SMBus-Compatible Clock Input/Output. Open-drain output.
17	V_{DD}	Power	Digital Supply Voltage. Bypass V_{DD} to V_{SS} with 0.1µF. Connect to V_{DDA}.
18	REG18	Power	Regulator for Digital Circuitry. Bypass to V_{SS} with 1µF and 10nF (500mΩ maximum ESR). Do not connect other circuitry to this pin.
19	PSEN10	DO	Power-Supply Enable 10. See the *Expanded Pin Description* section for more details.
	GPO10	DO	General-Purpose Output 10
	$\overline{FAULT2}$	DIO	Fault Input/Output 2. Open-drain, active-low I/O. See the *Expanded Pin Description* section for more details.
20	RS14	AI	ADC Voltage-Sense Input 14. Connect to V_{SS} if unused.
	GPI14	AI	General-Purpose Input 14. Connect to V_{SS} if unused.
21	MSCL	DIO	Master I^2C Clock Input/Output. Open-drain output.
22	MSDA	DIO	Master I^2C Data Input/Output. Open-drain output.
23	N.C.	—	No Internal Connection
24	PWM0	DO	PWM Margin Output 0. See the *Expanded Pin Description* section for more details.
	GPO12	DO	General-Purpose Output 12
25	PWM1	DO	PWM Margin Output 1. See the *Expanded Pin Description* section for more details.
	GPO13	DO	General-Purpose Output 13
26	PWM2	DO	PWM Margin Output 2. See the *Expanded Pin Description* section for more details.
	GPO14	DO	General-Purpose Output 14
27	PWM3	DO	PWM Margin Output 3. See the *Expanded Pin Description* section for more details.
	GPO15	DO	General-Purpose Output 15
28	PWM4	DO	PWM Margin Output 4. See the *Expanded Pin Description* section for more details.
	GPO16	DO	General-Purpose Output 16

Pin Description (continued)

PIN*	NAME	TYPE**	FUNCTION
29	PWM5	DO	PWM Margin Output 5. See the *Expanded Pin Description* section for more details.
	GPO17	DO	General-Purpose Output 17
30	$\overline{\text{ALERT}}$	DO	Alert Output. Open-drain, active-low output.
31	ADDR	DI	SMBus Slave Address Select. This pin is sampled on device power-up to determine the SMBus address. See the *PMBus/SMBus Address Select* section for details on how to strap this pin to select the proper slave address.
32	PWM6	DO	PWM Margin Output 6. See the *Expanded Pin Description* section for more details.
	GPO18	DO	General-Purpose Output 18
33	PWM7	DO	PWM Margin Output 7. See the *Expanded Pin Description* section for more details.
	GPO19	DO	General-Purpose Output 19
	$\overline{\text{FAULT1}}$	DIO	Fault Input/Output 1. Open-drain, active-low I/O. See the *Expanded Pin Description* section for more details.
34	RSG0	AI	Remote-Sense Ground for RS0/GPI0 to RS3/GPI3 and RS12/GPI12 to RS15/GPI15.
35	N.C.	—	No Internal Connection
36	RSG1	AI	Remote-Sense Ground for RS4/GPI4 to RS11/GPI11.
37	N.C.	—	No Internal Connection
38	PSEN0	DO	Power-Supply Enable 0. See the *Expanded Pin Description* section for more details.
	GPO0	DO	General-Purpose Output 0
39	PSEN1	DO	Power-Supply Enable 1. See the *Expanded Pin Description* section for more details.
	GPO1	DO	General-Purpose Output 1
40	PSEN2	DO	Power-Supply Enable 2. See the *Expanded Pin Description* section for more details.
	GPO2	DO	General-Purpose Output 2
41	PSEN3	DO	Power-Supply Enable 3. See the *Expanded Pin Description* section for more details.
	GPO3	DO	General-Purpose Output 3
42	PSEN4	DO	Power-Supply Enable 4. See the *Expanded Pin Description* section for more details.
	GPO4	DO	General-Purpose Output 4
43	PSEN5	DO	Power-Supply Enable 5. See the *Expanded Pin Description* section for more details.
	GPO5	DO	General-Purpose Output 5
44	PSEN6	DO	Power-Supply Enable 6. See the *Expanded Pin Description* section for more details.
	GPO6	DO	General-Purpose Output 6
45	PSEN7	DO	Power-Supply Enable 7. See the *Expanded Pin Description* section for more details.
	GPO7	DO	General-Purpose Output 7
46	PSEN8	DO	Power-Supply Enable 8. See the *Expanded Pin Description* section for more details.
	GPO8	DO	General-Purpose Output 8
47	RS13	AI	ADC Voltage-Sense Input 13. Connect to V_{SS} if unused.
	GPI13	AI	General-Purpose Input 13. Connect to V_{SS} if unused.
48	PSEN9	DO	Power-Supply Enable 9. See the *Expanded Pin Description* section for more details.
	GPO9	DO	General-Purpose Output 9

Pin Description (continued)

PIN*	NAME	TYPE**	FUNCTION
49	RS12	AI	ADC Voltage-Sense Input 12. Connect to V_{SS} if unused.
	GPI12	AI	General-Purpose Input 12. Connect to V_{SS} if unused.
50	RS11	AI	ADC Voltage-Sense Input 11. Connect to V_{SS} if unused.
	GPI11	AI	General-Purpose Input 11. Connect to V_{SS} if unused.
51	RS10	AI	ADC Voltage-Sense Input 10. Connect to V_{SS} if unused.
	GPI10	AI	General-Purpose Input 10. Connect to V_{SS} if unused.
52	RS9	AI	ADC Voltage-Sense Input 9. Connect to V_{SS} if unused.
	GPI9	AI	General-Purpose Input 9. Connect to V_{SS} if unused.
53	RS8	AI	ADC Voltage-Sense Input 8. Connect to V_{SS} if unused.
	GPI8	AI	General-Purpose Input 8. Connect to V_{SS} if unused.
54	RS7	AI	ADC Voltage-Sense Input 7. Connect to V_{SS} if unused.
	GPI7	AI	General-Purpose Input 7. Connect to V_{SS} if unused.
55	RS6	AI	ADC Voltage-Sense Input 6. Connect to V_{SS} if unused.
	GPI6	AI	General-Purpose Input 6. Connect to V_{SS} if unused.
56	RS5	AI	ADC Voltage-Sense Input 5. Connect to V_{SS} if unused.
	GPI5	AI	General-Purpose Input 5. Connect to V_{SS} if unused.
—	EP/V_{SS}	Power	Exposed Pad (Bottom Side of Package). Must be connected to local ground. The exposed pad is the ground reference (V_{SS}) for the device.

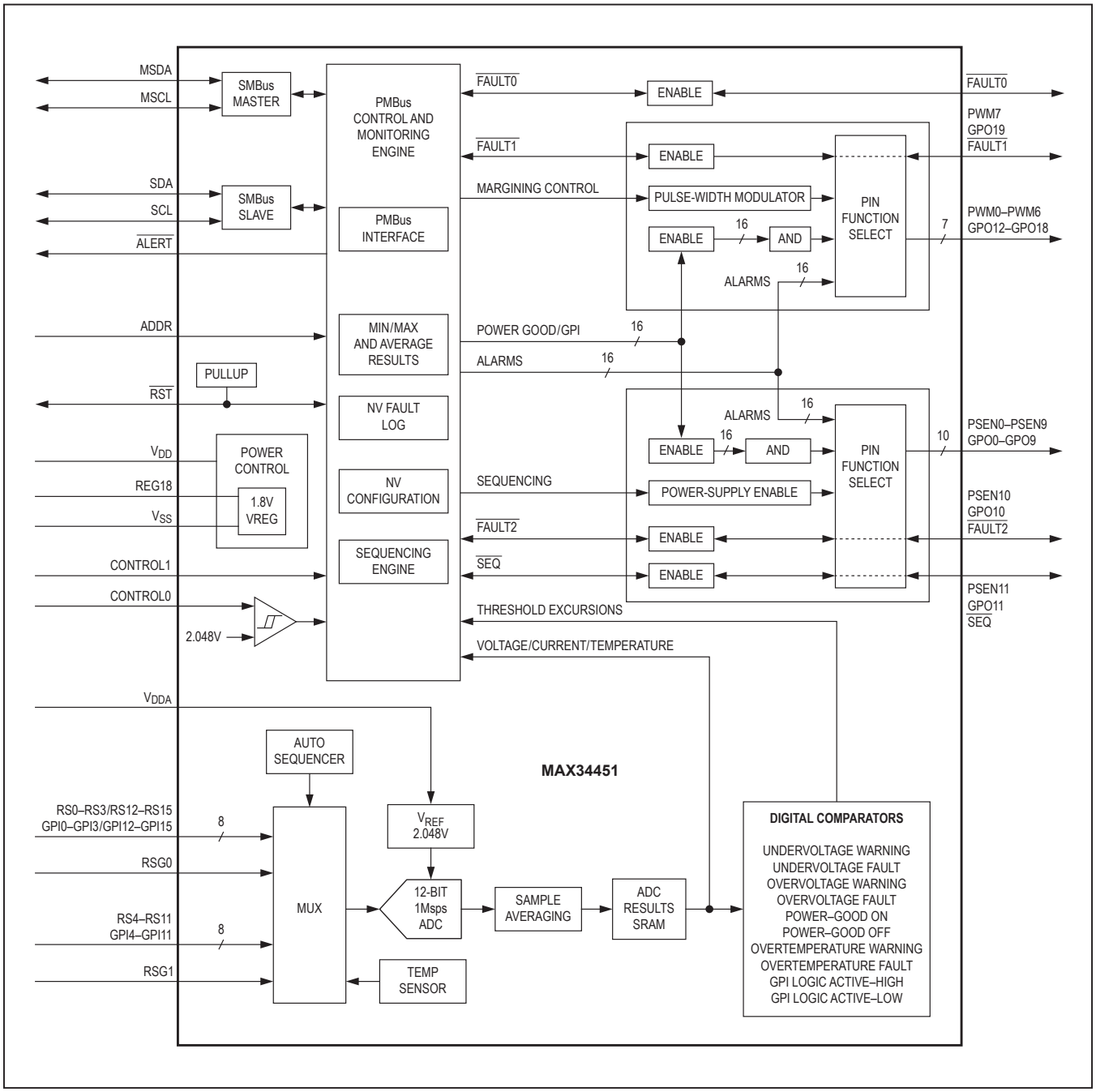
**All pins except the power pins, $\overline{ALERT}$, and ADDR are high impedance during device power-up and reset.*

***AI = Analog input, AO = Analog output, DI = Digital input, DIO = Digital input/output, and DO = Digital output.*

Expanded Pin Description

PIN	FUNCTION
PSEN0–PSEN11	The PSEN0–PSEN11 outputs are programmable with the MFR_PSEN_CONFIG command for either active-high or active-low operation and can be either open drain or push-pull. If not used for power-supply enables, these outputs can be repurposed as general-purpose outputs using the MFR_PSEN_CONFIG command. If these pins are used to enable power supplies, it is **highly recommended** that they have external pullups or pulldowns to force the supplies into an off state when the device is not active.
PWM0–PWM7	The PWM0–PWM7 outputs are high impedance when the margining is disabled. A 100% duty cycle implies the pins are continuously high. If not used for margining, these pins can be repurposed as general-purpose outputs with the MFR_PWM_CONFIG command.
$\overline{FAULT0}$–$\overline{FAULT2}$	The $\overline{FAULT0}$–$\overline{FAULT2}$ pins operate independently. Any global channel can be enabled with the MFR_FAULT_RESPONSE command to assert one or more of the $\overline{FAULTn}$ signals. Also, each global channel can be enabled to shut down when one or more of the $\overline{FAULTn}$ signals asserts. These pins are used to provide hardware control for power supplies across multiple devices. These outputs are unconditionally deasserted while $\overline{RST}$ is asserted or the device is power cycled. After device reset and upon device power-up, these outputs are pulled low immediately after program recall and held low until monitoring starts. Once monitoring starts, the $\overline{FAULTn}$ signals are released if no enabled faults are present.

Block Diagram



Detailed Description

The MAX34451 is a highly integrated system monitor with functionality to monitor up to 16 different voltages or currents and to sequence and close-loop margin up to 12 power supplies. It also supports local and remote thermal sensing.

The power-supply manager monitors the power-supply output voltage and current and constantly checks for user-programmable overvoltage, undervoltage, and overcurrent thresholds. It also has the ability to margin the power-supply output voltage up or down by a user-programmable level. The margining is performed in a closed-loop arrangement, whereby the device automatically adjusts a PWM signal or an external current DAC output and then measures the resultant output voltage. The power-supply manager can also sequence the supplies in any order at both power-up and power-down.

Thermal monitoring can be accomplished using up to five temperature sensors including an on-chip temperature sensor and up to four external remote DS75LV digital temperature sensors. Communications with the DS75LV temperature sensors is conducted through a dedicated I^2C/SMBus interface.

The device provides $\overline{\text{ALERT}}$ and $\overline{\text{FAULTn}}$ output signals. Host communications are conducted through a PMBus-compatible communications port.

See Table 1 and Table 2 for more details on specific device operation.

Table 1. PMBus PAGE to Pin/Resource Mapping

PMBus PAGE	PIN NAME								
	RSn/GPIn (16 AVAILABLE)			PSENn/GPOn (12 AVAILABLE)			PWMn/GPOn (8 AVAILABLE)		
	VOLTAGE OR CURRENT MONITOR	GENERAL-PURPOSE INPUT (GPI)	PIN	POWER-SUPPLY ENABLE (PSEN)	GENERAL-PURPOSE OUTPUT (GPO)	PIN	PWM MARGIN OUTPUT (PWM)	GENERAL-PURPOSE OUTPUT (GPO)	PIN
0	RS0	GPI0	5	PSEN0	GPO0	38	PWM0	GPO12	24
1	RS1	GPI1	4	PSEN1	GPO1	39	PWM1	GPO13	25
2	RS2	GPI2	3	PSEN2	GPO2	40	PWM2	GPO14	26
3	RS3	GPI3	2	PSEN3	GPO3	41	PWM3	GPO15	27
4	RS4	GPI4	1	PSEN4	GPO4	42	PWM4	GPO16	28
5	RS5	GPI5	56	PSEN5	GPO5	43	PWM5	GPO17	29
6	RS6	GPI6	55	PSEN6	GPO6	44	PWM6	GPO18	32
7	RS7	GPI7	54	PSEN7	GPO7	45	PWM7	GPO19	33
8	RS8	GPI8	53	PSEN8	GPO8	46	Margin capability provided through the external DS4424		
9	RS9	GPI9	52	PSEN9	GPO9	48			
10	RS10	GPI10	51	PSEN10	GPO10	19			
11	RS11	GPI11	50	PSEN11	GPO11	13			
12	RS12	GPI12	49	Can monitor voltage or current or be assigned as GPI					
13	RS13	GPI13	47						
14	RS14	GPI14	20						
15	RS15	GPI15	14						

Table 2. Device Channel Capabilities and Options

MAX34451 CHANNEL	PMBus COMMAND PAGE	CHANNEL CAPABILITIES
0–7	0–7	**Voltage Monitor/Sequence/Margin/GPO Option:** Pins RSn/GPIn, PSENn, and PWMn (where n = 0–7) have a one-to-one association for each channel that monitors for voltage (RSn) and can be used to sequence (PSENn) and margin (PWMn) the power supply. The voltage monitored on this channel can also be configured to determine a power-good state. If not required for either sequencing or margining, the associated PSENn and PWMn outputs can be repurposed as GPOn outputs that can either indicate a logic combination of power-good (PG) and GPI states or report alarms. **Current Monitor/GPO Option:** If the RSn/GPIn input is used to monitor current, then the channel is not used to sequence or margin. The associated PSENn and PWMn outputs can be repurposed as GPOn outputs that can either indicate a logic combination of power-good (PG) and GPI states or report alarms. **GPI/GPO Option:** If the RSn/GPIn input is configured as a general-purpose input (GPI), it can be used as a term in a logic combination to determine a power-good (PG) state and assert a GPOn output or act as a condition to allow a power supply to be enabled. The associated PSENn and PWMn outputs can be repurposed as GPOn outputs that can indicate power-good (PG) states or report alarms.
8–11	8–11	**Same as Channels 0–7 Except No PWM Outputs:** Pins RSn/GPIn, and PSENn (where n = 8–11) are the same as channels 0–7, except the PWMn outputs for these channels do not exist and instead the device uses an external DS4424 current DAC (connected to the master I^2C local bus) to margin the power supplies. These channels can also be used to monitor current or be used as GPIn inputs just like channels 0–7.
12–15	12–15	Pins RSn/GPIn (where n = 12–15) cannot be used to control sequencing or for margining. **Voltage Monitor Option:** Monitor voltage including channel power-good (PG) and can also be configured to shut down one or more power supplies if a fault occurs. **Current Monitor Option:** Monitor current and can be configured to shut down one or more power supplies if a fault occurs. **GPI Option:** As a general-purpose input (GPI), can be used as a term in a logic combination to determine a power-good (PG) state and assert a GPOn output or act as a condition to allow a power supply to be enabled.

Table 3. PMBus Command Codes

CODE	COMMAND NAME	TYPE	PAGE				NO. OF BYTES	FLASH STORED/ LOCKED (NOTE 2)	DEFAULT VALUE (NOTE 2)
			0–11	12–15	16–20	255			
			(NOTE 1)						
00h	PAGE	R/W byte	R/W	R/W	R/W	R/W	1	N/N	00h
01h	OPERATION	R/W byte	R/W			W	1	N/N	00h
02h	ON_OFF_CONFIG	R/W byte	R/W	R/W	R/W	R/W	1	Y/Y	1Ah
03h	CLEAR_FAULTS	Send byte	W	W	W	W	0	N/N	—
10h	WRITE_PROTECT	R/W byte	R/W	R/W	R/W	R/W	1	N/Y	00h
11h	STORE_DEFAULT_ALL	Send byte	W	W	W	W	0	N/Y	—
12h	RESTORE_DEFAULT_ALL	Send byte	W	W	W	W	0	N/Y	—
19h	CAPABILITY	Read byte	R	R	R	R	1	N/N	20h/30h
20h	VOUT_MODE	Read byte	R	R	R	R	1	FIXED/N	40h
25h	VOUT_MARGIN_HIGH	R/W word	R/W	—	—	—	2	Y/Y	0000h
26h	VOUT_MARGIN_LOW	R/W word	R/W	—	—	—	2	Y/Y	0000h
2Ah	VOUT_SCALE_MONITOR	R/W word	R/W	R/W	—	—	2	Y/Y	7FFFh
38h	IOUT_CAL_GAIN	R/W word	R/W	R/W	—	—	2	Y/Y	0000h
40h	VOUT_OV_FAULT_LIMIT	R/W word	R/W	R/W	—	—	2	Y/Y	7FFFh
42h	VOUT_OV_WARN_LIMIT	R/W word	R/W	R/W	—	—	2	Y/Y	7FFFh
43h	VOUT_UV_WARN_LIMIT	R/W word	R/W	R/W	—	—	2	Y/Y	0000h
44h	VOUT_UV_FAULT_LIMIT	R/W word	R/W	R/W	—	—	2	Y/Y	0000h
46h	IOUT_OC_WARN_LIMIT	R/W word	R/W	R/W	—	—	2	Y/Y	7FFFh
4Ah	IOUT_OC_FAULT_LIMIT	R/W word	R/W	R/W	—	—	2	Y/Y	7FFFh
4Fh	OT_FAULT_LIMIT	R/W word	—	—	R/W	—	2	Y/Y	7FFFh
51h	OT_WARN_LIMIT	R/W word	—	—	R/W	—	2	Y/Y	7FFFh
5Eh	POWER_GOOD_ON	R/W word	R/W	R/W	—	—	2	Y/Y	0000h
5Fh	POWER_GOOD_OFF	R/W word	R/W	R/W	—	—	2	Y/Y	0000h
60h	TON_DELAY	R/W word	R/W	—	—	—	2	Y/Y	0000h
62h	TON_MAX_FAULT_LIMIT	R/W word	R/W	—	—	—	2	Y/Y	FFFFh
64h	TOFF_DELAY	R/W word	R/W	—	—	—	2	Y/Y	0000h
79h	STATUS_WORD	Read word	R	R	R	R	2	N/N	0000h
7Ah	STATUS_VOUT	Read byte	R	R	—	—	1	N/N	00h
7Bh	STATUS_IOUT	Read byte	R	R	—	—	1	N/N	00h
7Dh	STATUS_TEMPERATURE	Read byte	—	—	R	—	1	N/N	00h
7Eh	STATUS_CML	Read byte	R	R	R	R	1	N/N	00h

Table 3. PMBus Command Codes (continued)

CODE	COMMAND NAME	TYPE	PAGE				NO. OF BYTES	FLASH STORED/ LOCKED (NOTE 2)	DEFAULT VALUE (NOTE 2)
			0–11	12–15	16–20	255			
			(NOTE 1)						
80h	STATUS_MFR_SPECIFIC	Read byte	R	R	—	R	1	N/N	00h
8Bh	READ_VOUT	Read word	R	R	—	—	2	N/N	0000h
8Ch	READ_IOUT	Read word	R	R	—	—	2	N/N	0000h
8Dh	READ_TEMPERATURE_1	Read word	—	—	R	—	2	N/N	0000h
98h	PMBUS_REVISION	Read byte	R	R	R	R	1	FIXED/N	11h
99h	MFR_ID	Read byte	R	R	R	R	1	FIXED/N	4Dh
9Ah	MFR_MODEL	Read byte	R	R	R	R	1	FIXED/N	59h
9Bh	MFR_REVISION	Read word	R	R	R	R	2	FIXED/N	(Note 3)
9Ch	MFR_LOCATION	R/W 64	R/W	R/W	R/W	R/W	8	Y/Y	(Note 4)
9Dh	MFR_DATE	R/W 64	R/W	R/W	R/W	R/W	8	Y/Y	(Note 4)
9Eh	MFR_SERIAL	R/W 64	R/W	R/W	R/W	R/W	8	Y/Y	(Note 4)
D1h	MFR_MODE	R/W word	R/W	R/W	R/W	R/W	2	Y/Y	0020h
D2h	MFR_PSEN_CONFIG	R/W 32	R/W	—	—	—	4	Y/Y	(Note 5)
D4h	MFR_VOUT_PEAK	R/W word	R/W	R/W	—	—	2	N/Y	0000h
D5h	MFR_IOUT_PEAK	R/W word	R/W	R/W	—	—	2	N/Y	0000h
D6h	MFR_TEMPERATURE_PEAK	R/W word	—	—	R/W	—	2	N/Y	8000h
D7h	MFR_VOUT_MIN	R/W word	R/W	R/W	—	—	2	N/Y	7FFFh
D8h	MFR_NV_LOG_CONFIG	R/W word	R/W	R/W	R/W	R/W	2	Y/Y	0000h
D9h	MFR_FAULT_RESPONSE	R/W 32	R/W	R/W			4	Y/Y	(Note 5)
DAh	MFR_FAULT_RETRY	R/W word	R/W	R/W	R/W	R/W	2	Y/Y	0000h
DCh	MFR_NV_FAULT_LOG	Read 32	R	R	R	R	255	Y/Y	(Note 6)
DDh	MFR_TIME_COUNT	R/W 32	R/W	R/W	R/W	R/W	4	N/Y	(Note 5)
DFh	MFR_MARGIN_CONFIG	R/W word	R/W	—	—	—	2	Y/Y	0000h
E0h	MFR_FW_SERIAL	R word	—	—	—	R	2	N/N	<firmware revision>
E2h	MFR_IOUT_AVG	R/W word	R	R	—	—	2	N/Y	0000h
E4h	MFR_CHANNEL_CONFIG	R/W word	R/W	R/W	—	—	2	Y/Y	0000h
E6h	MFR_TON_SEQ_MAX	R/W word	R/W	—	—	—	2	Y/Y	0000h
E7h	MFR_PWM_CONFIG (Note 7)	R/W 32	R/W	—	—	—	4	Y/Y	(Note 5)
E8h	MFR_SEQ_CONFIG	R/W 32	R/W	—	—	—	4	Y/Y	(Note 5)

Table 3. PMBus Command Codes (continued)

CODE	COMMAND NAME	TYPE	PAGE				NO. OF BYTES	FLASH STORED/ LOCKED (NOTE 2)	DEFAULT VALUE (NOTE 2)
			0–11	12–15	16–20	255			
			(NOTE 1)						
EEh	MFR_STORE_ALL	Write byte	W	W	W	W	1	N/Y	—
EFh	MFR_RESTORE_ALL	Write byte	W	W	W	W	1	N/Y	—
F0h	MFR_TEMP_SENSOR_CONFIG	R/W word	—	—	R/W	—	2	Y/Y	0000h
FCh	MFR_STORE_SINGLE	R/W word	R/W	R/W	R/W	R/W	2	N/Y	0000h
FEh	MFR_CRC	R/W word	R/W	R/W	R/W	R/W	2	N/Y	FFFFh

Note 1: Common commands are shaded; access through any page results in the same device response.

Note 2: In the **Flash Stored/Locked** column, the "N" on the left indicates that this parameter is not stored in flash memory when the STORE_DEFAULT_ALL or MFR_STORE_ALL command is executed; the value shown in the **Default Value** column is automatically loaded upon power-on reset or when the $\overline{RST}$ pin is asserted. The "Y" on the left in the **Flash Stored/Locked** column indicates that the currently loaded value in this parameter is stored in flash memory when the STORE_DEFAULT_ALL or MFR_STORE_ALL command is executed and is automatically loaded upon power-on reset or when the RST pin is asserted; the value shown in the **Default Value** column is the value when shipped from the factory. "FIXED" in the **Flash Stored** column means that the value is fixed at the factory and cannot be changed. The value shown in the **Default Value** column is automatically loaded upon power-on reset or when the $\overline{RST}$ pin is asserted. The right-side Y/N indicates that when the device is locked, only the commands listed with "N" can be accessed. All other commands are ignored if written and return FFh if read. Only the PAGE, CLEAR_FAULTS, OPERATION, and MFR_SERIAL commands can be written to. The device unlocks if the upper 4 bytes of MFR_SERIAL match the data written to the device.

Note 3: The factory-set value is dependent on the device hardware and firmware revision.

Note 4: The factory-set default value for this 8-byte block is 3130313031303130h.

Note 5: The factory-set default value for this 4-byte block is 00000040h.

Note 6: The factory-set default value for the complete block of the MFR_NV_FAULT_LOG is FFh.

Note 7: MFR_PWM_CONFIG is only available for PAGES 0–7.

PMBus/SMBus Address Select

On device power-up, the device samples the ADDR pin to determine the PMBus/SMBus serial-port address. The combination of the components shown in Figure 1 determines the serial-port address (also see Table 4).

SMBus/PMBus Operation

The device implements the PMBus command structure using the SMBus format. The structure of the data flow between the host and the slave is shown below for several different types of transactions. All transactions begin with a host sending a command code that is immediately preceded with a 7-bit slave address (R/$\overline{W}$ = 0). Data is sent MSB first.

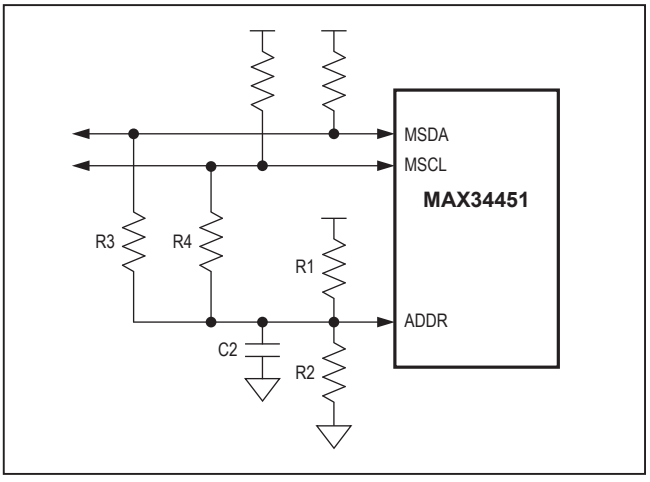


Figure 1. PMBus/SMBus Address Select

Table 4. PMBus/SMBus Serial-Port Address

R1	R2	R3	R4	C2	SLAVE ADDRESS
—	220kΩ	—	—	—	1110 100x (E8h)
220kΩ	—	—	—	—	1110 101x (EAh)
220kΩ	—	—	—	100nF	0010 010x (24h)
22kΩ	—	—	—	100nF	0010 011x (26h)
—	—	0kΩ	—	—	1001 100x (98h)
—	—	220kΩ	—	—	1001 101x (9Ah)
—	—	—	0kΩ	—	1011 000x (B0h)
—	—	—	220kΩ	—	1011 001x (B2h)
—	0kΩ	—	—	—	1001 110x (9Ch)

Note: *The device also responds to a slave address of 34h (this is the factory programming address); the device should not share the same I^2C bus with other devices that use this slave address. The letter "x" in the Slave Address column indicates the R/W bit location.*

SMBus/PMBus Operation Examples

READ WORD FORMAT

1	7	1	1	8	1	1	7	1	1	8	1	8	1	1
S	SLAVE ADDRESS	W	A	COMMAND CODE	A	Sr	SLAVE ADDRESS	R	A	DATA BYTE LOW	A	DATA BYTE HIGH	NA	P

READ BYTE FORMAT

1	7	1	1	8	1	1	7	1	1	8	1	1
S	SLAVE ADDRESS	W	A	COMMAND CODE	A	Sr	SLAVE ADDRESS	R	A	DATA BYTE	NA	P

WRITE WORD FORMAT

1	7	1	1	8	1	8	1	8	1	1
S	SLAVE ADDRESS	W	A	COMMAND CODE	A	DATA BYTE LOW	A	DATA BYTE HIGH	A	P

WRITE BYTE FORMAT

1	7	1	1	8	1	8	1	1
S	SLAVE ADDRESS	W	A	COMMAND CODE	A	DATA BYTE	A	P

SEND BYTE FORMAT

1	7	1	1	8	1	1
S	SLAVE ADDRESS	W	A	COMMAND CODE	A	P

KEY:

S = START
Sr = REPEATED START
P = STOP
W = WRITE BIT (0)
R = READ BIT (1)
A = ACKNOWLEDGE (0)
NA = NOT ACKNOWLEDGE (1)
SHADED BLOCK = SLAVE TRANSACTION

Group Command

The device supports the group command. With the group command, a host can write different data to multiple devices on the same serial bus with one long continuous data stream. All the devices addressed during this transaction wait for the host to issue a STOP before beginning to respond to the command.

Group Command Write Format

SLAVE ADDRESS, COMMAND BYTE, AND DATA WORD FOR DEVICE 1

1	7	1	1	8	1	8	1	8	1
S	SLAVE ADDRESS	W	A	COMMAND CODE	A	DATA BYTE LOW	A	DATA BYTE HIGH	A

• • •

SLAVE ADDRESS, COMMAND BYTE, AND DATA BYTE FOR DEVICE 2

1	7	1	1	8	1	8	1
Sr	SLAVE ADDRESS	W	A	COMMAND CODE	A	DATA BYTE	A

• • •

SLAVE ADDRESS AND SEND BYTE FOR DEVICE 3

1	7	1	1	8	1
Sr	SLAVE ADDRESS	W	A	COMMAND CODE	A

• • •

SLAVE ADDRESS, COMMAND BYTE, AND DATA WORD FOR DEVICE N

1	7	1	1	8	1	8	1	8	1	1
Sr	SLAVE ADDRESS	W	A	COMMAND CODE	A	DATA BYTE LOW	A	DATA BYTE HIGH	A	P

KEY:

S = START

Sr = REPEATED START

P = STOP

W = WRITE BIT (0)

R = READ BIT (1)

A = ACKNOWLEDGE (0)

NA = NOT ACKNOWLEDGE (1)

SHADED BLOCK = SLAVE TRANSACTION

Addressing

The device responds to receiving its fixed slave address by asserting an acknowledge (ACK) on the bus. The device does not respond to a general call address; it only responds when it receives its fixed slave address or the alert response address. See the *ALERT and Alert Response Address (ARA)* section for more details.

ALERT and Alert Response Address (ARA)

If the $\overline{\text{ALERT}}$ output is enabled (ALERT bit = 1 in MFR_MODE) when a fault occurs, the device asserts the $\overline{\text{ALERT}}$ signal and then waits for the host to send an ARA, as shown in the *Alert Response Address (ARA) Byte Format* section.

Alert Response Address (ARA) Byte Format

1	7	1	1	8	1	1
S	ARA 0001100	R	A	DEVICE SLAVE ADDRESS WITH LSB = 0	NA	P

When the ARA is received and the device is asserting $\overline{\text{ALERT}}$, the device ACKs it and then attempts to place its fixed slave address on the bus by arbitrating the bus, since another device could also try to respond to the ARA. The rules of arbitration state that the lowest address device wins. If the device wins the arbitration, it deasserts $\overline{\text{ALERT}}$. If the device loses arbitration, it keeps $\overline{\text{ALERT}}$ asserted and waits for the host to once again send the ARA.

Host Sends or Reads Too Few Bits

If, for any reason, the host does not complete writing a full byte or reading a full byte from the device before a START or STOP is received, the device does the following:

1) Ignores the command.
2) Sets the CML bit in STATUS_WORD.
3) Sets the DATA_FAULT bit in STATUS_CML.
4) Notifies the host through $\overline{\text{ALERT}}$ assertion (if enabled).

Host Sends or Reads Too Few Bytes

For each supported command, the device expects a fixed number of bytes to be written to or read from the device. If, for any reason, less than the expected number of bytes are written to or read from the device, the device completely ignores the command and takes no action.

Host Sends Too Many Bytes or Bits

For each supported command, the device expects a fixed number of bytes to be written to the device. If for any reason, more than the expected number of bytes or bits is written to the device, the device does the following:

1) Ignores the command.
2) Sets the CML bit in STATUS_WORD.
3) Sets the DATA_FAULT bit in STATUS_CML.
4) Notifies the host through $\overline{\text{ALERT}}$ assertion (if enabled).

Host Reads Too Many Bytes or Bits

For each supported command, the device expects a fixed number of bytes to be read from the device. If, for any reason, more than the expected number of bytes or bits is read from the device, the device does the following:

1) Sends all ones (FFh) as long as the host keeps acknowledging.
2) Sets the CML bit in STATUS_WORD.
3) Sets the DATA_FAULT bit in STATUS_CML.
4) Notifies the host through $\overline{\text{ALERT}}$ assertion (if enabled).

Host Sends Improperly Set Read Bit in the Slave Address Byte

If the device receives the R/$\overline{\text{W}}$ bit in the slave address set to a one immediately preceding the command code, the device does the following (this does not apply to the ARA):

1) ACKs the address byte.
2) Sends all ones (FFh) as long as the host keeps acknowledging.
3) Sets the CML bit in STATUS_WORD.
4) Sets the DATA_FAULT bit in STATUS_CML.
5) Notifies the host through $\overline{\text{ALERT}}$ assertion (if enabled).

Unsupported Command Code Received/Host Writes to a Read-Only Command

If the host sends the device a command code that it does not support, or if the host sends a command code that is not supported by the current PAGE setting, the device does the following:

1) Ignores the command.
2) Sets the CML bit in STATUS_WORD.
3) Sets the COMM_FAULT bit in STATUS_CML.
4) Notifies the host through $\overline{\text{ALERT}}$ assertion (if enabled).

Invalid Data Received

The device checks the PAGE, OPERATION, and WRITE_PROTECT command codes for valid data. If the host writes a data value that is invalid, the device does the following:

1) Ignores the command.
2) Sets the CML bit in STATUS_WORD.
3) Sets the DATA_FAULT bit in STATUS_CML.
4) Notifies the host through $\overline{\text{ALERT}}$ assertion (if enabled).

Host Reads from a Write-Only Command

When a read request is issued to a write-only command (CLEAR_FAULTS, STORE_DEFAULT_ALL, RESTORE_DEFAULT_ALL, MFR_STORE_ALL, MFR_RESTORE_ALL, OPERATION with PAGE = 255), the device does the following:

1) ACKs the address byte.
2) Ignores the command.
3) Sends all ones (FFh) as long as the host keeps acknowledging.
4) Sets the CML bit in STATUS_WORD.
5) Sets the DATA_FAULT bit in STATUS_CML.
6) Notifies the host through $\overline{\text{ALERT}}$ assertion (if enabled).

SMBus Timeout

If, during an active SMBus communication sequence, the SCL signal is held low for greater than the timeout duration (nominally 27ms), the device terminates the sequence and resets the serial bus. It takes no other action. No status bits are set.

PMBus Operation

From a software perspective, the device appears as a PMBus device capable of executing a subset of PMBus commands. In this data sheet, the term SMBus is used to refer to the electrical characteristics of the PMBus communication using the SMBus physical layer. The term PMBus is used to refer to the PMBus command protocol. The device employs a number of standard SMBus protocols (e.g., Write Word, Read Word, Write Byte, Read Byte, Send Byte, etc.) to program output voltage and warning/fault thresholds, read monitored data, and provide access to all manufacturer-specific commands.

The device supports the group command. The group command is used to send commands to more than one PMBus device. It is not required that all the devices receive the same command. However, no more than one command can be sent to any one device in one group command packet. The group command must not be used with commands that require receiving devices to respond with data, such as the STATUS_WORD command. When the device receives a command through this protocol, it immediately begins execution of the received command after detecting the STOP condition.

The device supports the PAGE command and uses it to select which individual channel to access. When a data word is transmitted, the lower order byte is sent first and the higher order byte is sent last. Within any byte, the most-significant bit (MSB) is sent first and the least-significant bit (LSB) is sent last.

PMBus Protocol Support

The device supports a subset of the commands defined in the PMBus Power System Management Protocol Specification Part II - Command Language Revision 1.1. For detailed specifications and the complete list of PMBus commands, refer to Part II of the PMBus specification available at **www.PMBus.org**. The supported PMBus commands and the corresponding device behavior are described in this document. All data values are represented in DIRECT format, unless otherwise stated. Whenever the PMBus specification refers to the PMBus device, it is referring to the device operating in conjunction with a power supply. While the command can call for turning on or off the PMBus device, the device always remains on to continue communicating with the PMBus master and transfers the command to the power supply accordingly.

Data Format

Voltage data for commanding or reading the output voltage or related parameters (such as the overvoltage threshold) are presented in DIRECT format. DIRECT format data is a 2-byte, two's complement binary value. DIRECT format data can be used with any command that sends or reads a parametric value. The DIRECT format uses an equation and defined coefficients to calculate the desired values. Table 5 lists coefficients used by the device.

Interpreting Received DIRECT Format Values

The host system uses the following equation to convert the value received from the PMBus device—in this case the MAX34451—into a reading of volts, degrees Celsius, or other units as appropriate:

$$X = (1/m) \times (Y \times 10^{-R} - b)$$

where X is the calculated real-world value in the appropriate units (V, °C, etc.); m is the slope coefficient; Y is the 2-byte, two's complement integer received from the PMBus device; b is the offset; and R is the exponent.

Table 5. PMBus Command Code Coefficients

PARAMETER	COMMANDS	UNITS	RESOLUTION	MAXIMUM	m	b	R
Voltage	VOUT_MARGIN_HIGH VOUT_MARGIN_LOW VOUT_OV_FAULT_LIMIT VOUT_OV_WARN_LIMIT VOUT_UV_WARN_LIMIT VOUT_UV_FAULT_LIMIT POWER_GOOD_ON POWER_GOOD_OFF READ_VOUT MFR_VOUT_PEAK MFR_VOUT_MIN	mV	1	32767	1	0	0
Voltage Scaling	VOUT_SCALE_MONITOR	—	1/32767	1	32767	0	0
Current	IOUT_OC_FAULT_LIMIT IOUT_OC_WARN_LIMIT READ_IOUT MFR_IOUT_PEAK MFR_IOUT_AVG	A	0.01	327.67	1	0	2
Current Scaling	IOUT_CAL_GAIN	mΩ	0.1	3276.7	1	0	1
Temperature	OT_FAULT_LIMIT OT_WARN_LIMIT READ_TEMPERATURE_1 MFR_TEMPERATURE_PEAK	°C	0.01	327.67	1	0	2
Timing	TON_DELAY TON_MAX_FAULT_LIMIT TOFF_DELAY MFR_FAULT_RETRY MFR_TON_SEQ_MAX	ms	0.2	6553.4	5	0	0

Note: To reliably process fault-retry during simultaneous fault events on multiple channels, it is recommended to set MFR_FAULT_RETRY ≥ 1ms.

Sending a DIRECT Format Value

To send a value, the host must use the following equation to solve for Y:

$$Y = (mX + b) \times 10^{R}$$

where Y is the 2-byte, two's complement integer to be sent to the unit; m is the slope coefficient; X is the real-world value, in units such as volts, to be converted for transmission; b is the offset; and R is the exponent.

The following example demonstrates how the host can send and retrieve values from the device. Table 6 lists the coefficients used in the following parameters.

Table 6. Coefficients for DIRECT Format Value

COMMAND CODE	COMMAND NAME	m	b	R
25h	VOUT_MARGIN_HIGH	1	0	0
8Bh	READ_VOUT	1	0	0

If a host wants to set the device to change the power-supply output voltage to 3.465V (or 3465mV), the corresponding VOUT_MARGIN_HIGH value is:

$$Y = (1 \times 3465 + 0) \times 10^{0} = 3465 \text{ (decimal)}$$
$$= 0D89h \text{ (hex)}$$

Conversely, if the host received a value of 0D89h on a READ_VOUT command, this is equivalent to:

$$X = (1/1) \times (0D89h \times 10^{-(-0)} - 0) = 3465mV = 3.465V$$

Power supplies and power converters generally have no way of knowing how their outputs are connected to ground. Within the power supply, all output voltages are most commonly treated as positive. Accordingly, all output voltages and output-voltage-related parameters of PMBus devices are commanded and reported as positive values. It is up to the system to know that a particular output is negative if that is of interest to the system. All output-voltage-related commands use 2 data bytes.

Fault Management and Reporting

For reporting faults/warnings to the host on a real-time basis, the device asserts the open-drain $\overline{ALERT}$ pin (if enabled in MFR_MODE) and sets the appropriate bit in the various status registers. On recognition of the $\overline{ALERT}$ assertion, the host or system manager is expected to poll the I^2C bus to determine the device asserting $\overline{ALERT}$. The host sends the SMBus ARA (0001 100). The device ACKs the SMBus ARA, transmits its slave address, and deasserts $\overline{ALERT}$. The system controller then communicates with PMBus commands to retrieve the fault/warning status information from the device.

See the individual command sections for more details. Faults and warnings that are latched in the status registers are cleared when any one of the following conditions occur:

- A CLEAR_FAULTS command is received.
- The $\overline{RST}$ pin is toggled or a soft-reset is issued.
- Bias power to the device is removed and then reapplied.

One or more latched-off power supplies are only restarted when one of the following occurs:

- OPERATION commands are received that turn off and turn on the power supplies, or the CONTROLn pins are toggled to turn off and then turn on the power supplies.
- The $\overline{RST}$ pin is toggled or a soft-reset is issued.
- Bias power to the device is removed and then reapplied.

The device responds to fault conditions according to the configuration of the MFR_FAULT_RESPONSE command. This command determines how the device should respond to each particular fault and whether it should assert one or more of the $\overline{FAULTn}$ pins when a fault occurs.

The MFR_FAULT_RESPONSE command also determines whether a channel should power up if a fault is present. With the RESPONSE bits in MFR_FAULT_RESPONSE, each channel can be independently configured to either respond or not respond to each possible fault. Before any power-supply channel is enabled, or the $\overline{FAULTn}$ outputs deasserted, the device checks for overvoltage, overcurrent, and temperature faults (but not for undervoltage) if the channel is configured for a fault response to either latchoff (RESPONSE[1:0] = 01) or retry (RESPONSE[1:0] = 10) in the MFR_FAULT_RESPONSE command. Only after the faults clear is the channel allowed to turn on. See Table 7 for fault-monitoring states.

Table 7. Fault-Monitoring States

FAULT	REQUIRED DEVICE CONFIGURATION FOR ACTIVE MONITORING	WHEN MONITORED
Overvoltage	• Voltage monitoring enabled (SELECT[5:0] = 10h or 20h in MFR_CHANNEL_CONFIG)	Continuous monitoring.
Undervoltage/ Power Good	• Voltage monitoring enabled (SELECT[5:0] = 10h or 20h in MFR_CHANNEL_CONFIG)	• If SELECT[5:0] = 10h (monitor and sequence mode), stops monitoring when PSEN is disabled. Power Good starts monitoring when PSEN is enabled and Undervoltage monitoring starts when voltage exceeds the POWER_GOOD_ON level. • If SELECT[5:0] = 20h (monitor only mode), starts monitoring when the voltage exceeds the POWER_GOOD_ON level.
Overcurrent	• Current monitoring enabled • (SELECT[5:0] = 22h in MFR_CHANNEL_CONFIG)	Continuous monitoring.
Power-Up Time	• Sequencing enabled • (SELECT[5:0] = 10h in MFR_CHANNEL_CONFIG)	Monitored only during power-on sequence.
Overtemperature	• Temperature sensor enabled • (ENABLE = 1 in MFR_TEMP_SENSOR_CONFIG)	Continuous monitoring.

Note: *Device response to faults is determined by the configuration of MFR_FAULT_RESPONSE.*

Password Protection

The device can be password protected by using the LOCK bit in the MFR_MODE command. Once the device is locked, only certain PMBus commands can be accessed with the serial port. See Table 3 for a complete list of PMBus commands. Commands that have password protection return all ones (FFh), with the proper number of data bytes, when read. When the device is locked, only the PAGE, OPERATION, CLEAR_FAULTS, and MFR_SERIAL commands can be written; all other written commands are ignored. When MFR_SERIAL is written and the upper 4 bytes match the internally flash-stored value, the device unlocks and remains unlocked until the LOCK bit in MFR_MODE is activated once again. The LOCK status bit in STATUS_MFR_SPECIFIC is always available to indicate whether the device is locked or unlocked.

Power-Supply Sequencing

Sequencing control for each of the 12 power-supply channels on the device is configured using the MFR_SEQ_CONFIG and ON_OFF_CONFIG commands. See the descriptions of these commands for details on the exact device configuration required. Power supplies can be powered up and down in any order (even across multiple devices). See the command descriptions and Figure 2 for specifics on sequencing control.

Dual-Loop Sequencing

The device contains two independent sequencing groups, SEQUENCE0 and SEQUENCE1. Both groups do not need to be used, but every channel is assigned to one of the two groups with the SEQ_SELECT bit in the MFR_SEQ_CONFIG command. The two sequencing groups operate independently. SEQUENCE0 is always associated with CONTROL0 and SEQUENCE1 is always associated with CONTROL1. The two sequencing groups can also be independently controlled with the OPERATION command. With the ON_OFF_CONFIG command, the device is configured to respond to the CONTROLn pins or the OPERATION command (or both). When the OPERATION command is sent to the device (when the PAGE is set to 255), both sequence groups are controlled, as shown in Table 8.

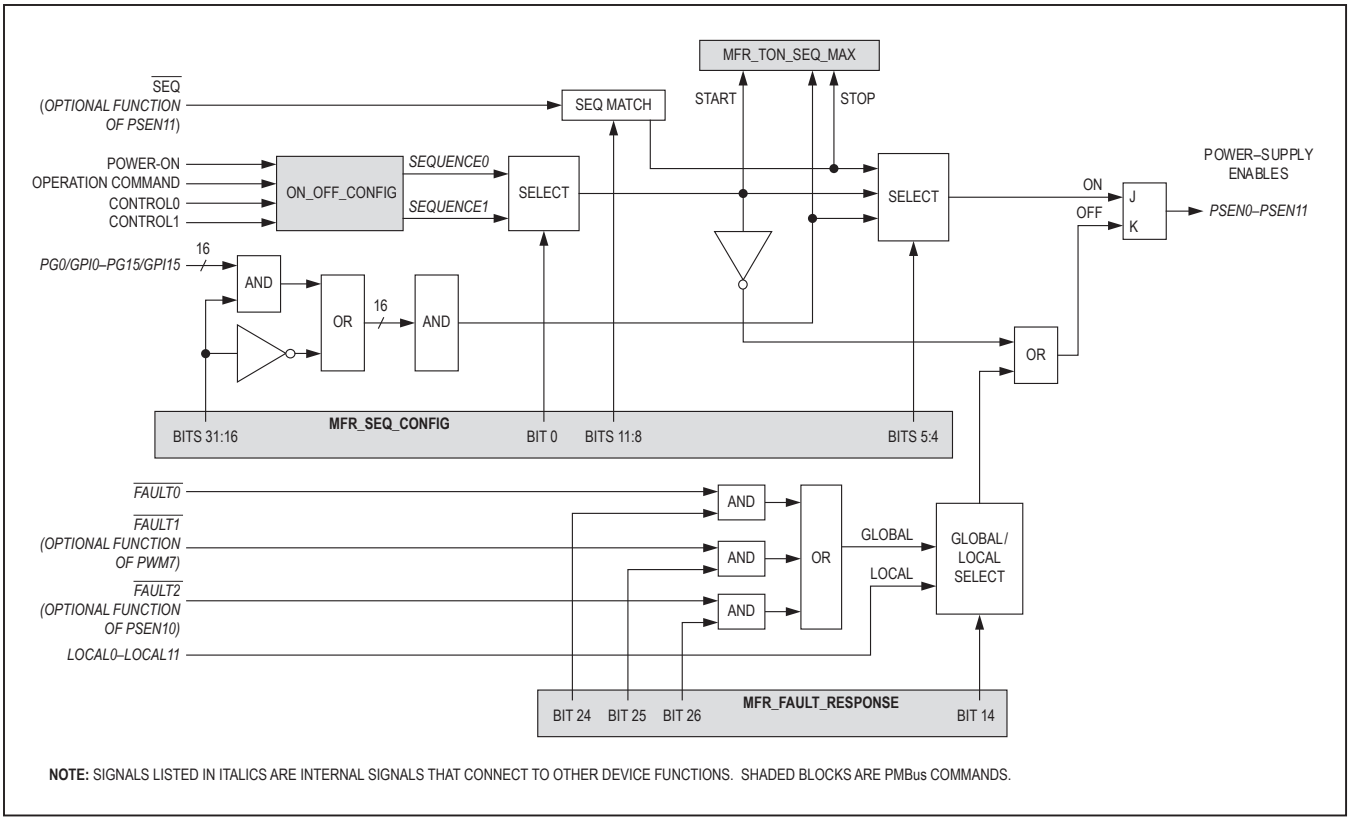


Figure 2. Sequence Control Logic

Table 8. OPERATION Command Sequence Control Options

GROUP	OPERATION COMMAND (PAGE = 255)		
	ON	SOFT-OFF	IMMEDIATE OFF
SEQUENCE0	80h or 81h	40h or 41h	00h or 01h
SEQUENCE1	80h or 82h	40h or 42h	00h or 02h

Power-On Sequencing

The activation of all power-supply channels (even across multiple devices) is initiated from a common START signal that can either be the CONTROL0 or CONTROL1 pin, or the OPERATION command. Each power-supply channel on the device can be sequenced on by one of the following methods:

- Power is applied to the device.
- The CONTROL0 pin goes active.
- The CONTROL1 pin goes active.
- The OPERATION command is received.
- The logic combination of power goods and GPI is valid.
- The $\overline{SEQ}$ pin signal is matched.

Each enabled PSENn output goes active (either active high or active low, as defined in MFR_PSEN_CONFIG) after the associated delay time programmed in TON_DELAY. The power supplies can be sequenced on in any order. Each channel can be sequenced on with either time-based or event-based conditions. The output voltage of each power supply is monitored to ensure that the supply crosses the power-good-on level (as configured in the POWER_GOOD_ON command) within a programmable time limit, as configured in the TON_MAX_FAULT_LIMIT command. This power-up time limit can be disabled by configuring TON_MAX_FAULT_LIMIT to 0000h. For channels using event-based sequencing, the MFR_TON_SEQ_MAX command determines the maximum time limit for the sequence-on event to occur. Like the TON_MAX_FAULT_LIMIT, this limit can be disabled by configuring MFR_TON_SEQ_MAX to 0000h. There is a one-to-one correspondence between the RSn inputs and the PSENn outputs. For example, RS6 monitors the power supply controlled by PSEN6. All power-on sequencing is gated by detected faults. Before any power-supply channel is enabled (or the $\overline{FAULTn}$ output deasserted) the device checks for overvoltage, overcurrent, and temperature faults that are enabled (but not for undervoltage since the supply is off).

Power-Off Sequencing

The order in which the supplies are disabled is determined with the TOFF_DELAY configuration. Alternatively, all the power supplies can be switched off immediately, as configured in the ON_OFF_CONFIG command or with the OPERATION command.

As configured with the ON_OFF_CONFIG command, either the CONTROL0 or CONTROL1 pin or the OPERATION command is the master off switch. When either the CONTROL0 or CONTROL1 pin goes inactive, or the OPERATION off command is received (or one of the enabled $\overline{FAULTn}$ pins asserted), the power supplies are sequenced off. Neither the power-good (PG) or GPI logic combinations, nor the $\overline{SEQ}$ pin, can be used to turn off the power supplies.

Sequencing Example

As an example, Figure 3 details a simple sequencing scheme consisting of four power supplies using a mixture of time-based and event-based sequencing. Channels 0 and 2 use time-based sequencing and channels 1 and 5 use event-based sequencing. When either the CONTROL0 or CONTROL1 pin goes active, or the OPERATION command is received (as defined by the ON_OFF_CONFIG command), PSEN0 is asserted after the delay time configured in TON_DELAY. RS0 is monitored to ensure that the PSEN0 supply crosses the power-good-on level (as configured in POWER_GOOD_ON) within a programmable time limit (as configured in TON_MAX_FAULT_LIMIT). PSEN2 operates in a similar fashion as PSEN0, but with a different TON_DELAY and a different TON_MAX_FAULT_LIMIT. Since the power-up of channels 0 and 2 are based solely on their TON_DELAY values, these channels are time-based.

When RS2 crosses its power-good-on level, PSEN5 is asserted after its configured TON_DELAY and similarly, PSEN1 asserts when RS5 crosses its power-good-on level. Since the power-up of channels 5 and 1 are based on the power-good states of other channels, these channels are event-based. The MFR_TON_SEQ_MAX command can be used to ensure that these events occur and the power-up sequence does not hang waiting for an event to transpire. When RS1 crosses its power-good-on level, it has been configured to generate a $\overline{SEQ}$ pin signal to communicate to another device to turn on one or more of its power supplies.

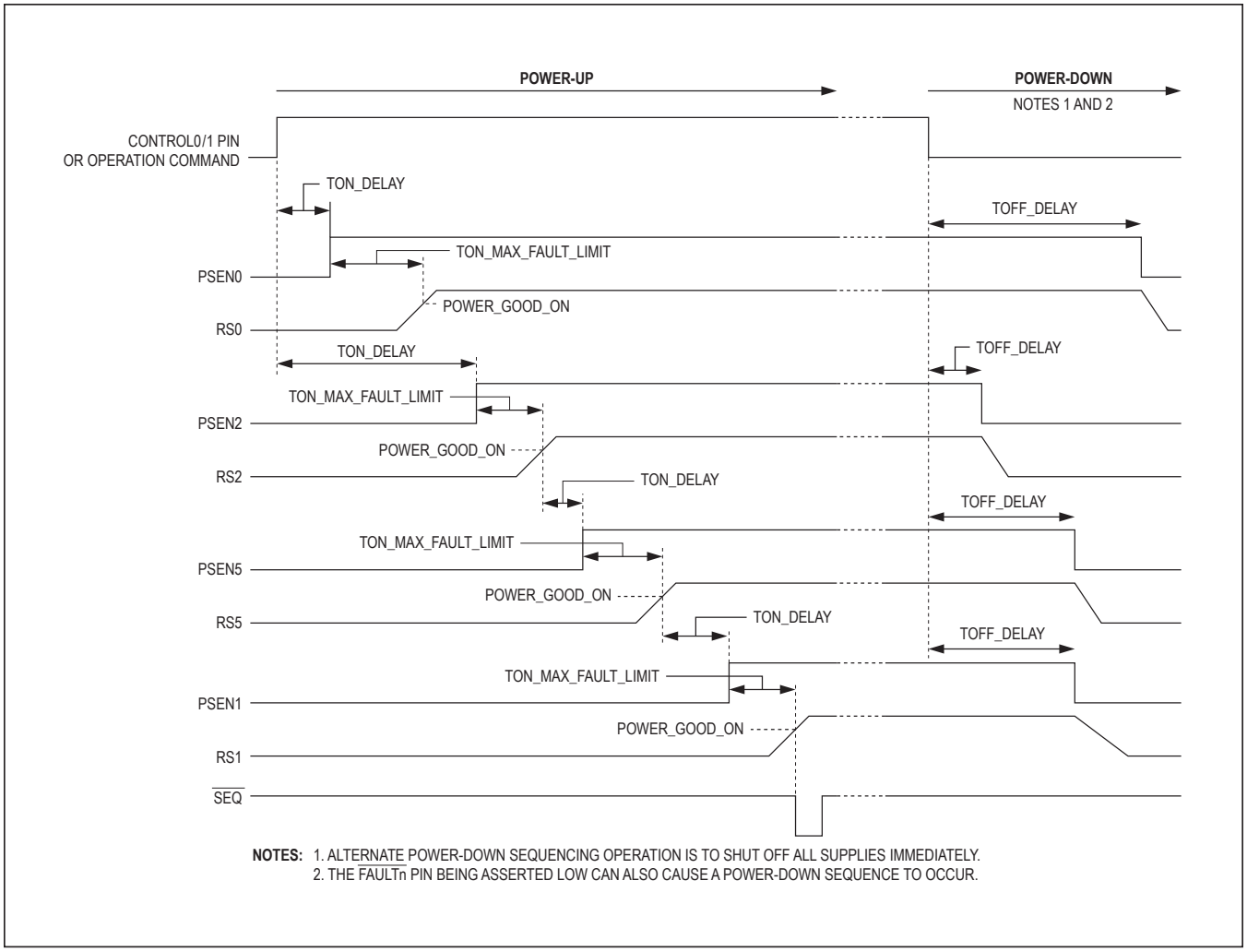


Figure 3. Sequencing Example

Multiple Device Connections

Multiple MAX34451 devices can be connected together to increase the system channel count. Figure 4 details the recommended connection scheme.

All the paralleled devices share the same CONTROLn, $\overline{\text{FAULTn}}$, $\overline{\text{SEQ}}$, and SMBus signals. The devices all use a common signal (either the CONTROL0 or CONTROL1 pin, or the OPERATION command) to enable/disable all the power supplies. Any of the monitored power supplies can be configured with the MFR_FAULT_RESPONSE command to activate one or more of the $\overline{\text{FAULTn}}$ signals and shut down all the other supplies enabled to respond to one or more of the $\overline{\text{FAULTn}}$ signals. The $\overline{\text{FAULT0}}$ signal is always available, whereas $\overline{\text{FAULT1}}$ and $\overline{\text{FAULT2}}$ are optional signals. When they are enabled, the PWM7 and PSEN10 outputs (respectively) are disabled. The use of multiple fault signals allows more flexibility in controlling which power supplies need to shut down during a fault.

USER NOTE:

- All devices must be configured with the same ON_OFF_CONFIG configuration.
- All devices must be powered up and reset at the same time.

$\overline{\text{SEQ}}$ Pin Operation

The $\overline{\text{SEQ}}$ pin is another optional signal. When this function is enabled, it allows multiple devices to coordinate event-based sequencing. With the MFR_CHANNEL_CONFIG command, any channel can be configured to generate one of 15 signatures. When the channel crosses its power-good-on level, it generates the needed $\overline{\text{SEQ}}$ signature if so enabled. With the MFR_SEQ_CONFIG command, any of the sequencing channels (PAGES 0–11) can be configured to wait for a match on the $\overline{\text{SEQ}}$ pin before asserting the PSENn output. To ensure that a valid $\overline{\text{SEQ}}$ signal is received when it should be, the maximum allowable time is configured into the MFR_TON_SEQ_MAX command.

USER NOTE:

- Only one channel should be configured to generate any one particular $\overline{\text{SEQ}}$ signature. If two channels generate the same signature, they might reach their power-good-on levels at different times and corrupt the $\overline{\text{SEQ}}$ signal.
- Allow more than 15ms between consecutive $\overline{\text{SEQ}}$ signatures.

System Watchdog Timer

The device uses an internal watchdog timer. This timer is internally reset every 5ms. In the event the device is locked up, and the watchdog reset does not occur after 210ms, the device is automatically reset. After the reset occurs, the device reloads all configuration values that were stored to flash and begins normal operation. After the reset, the device also does the following:

1) Sets the MFR bit in STATUS_WORD.
2) Sets the WATCHDOG_INT bit in STATUS_MFR_SPECIFIC (for PAGE 255).
3) Notifies the host through $\overline{\text{ALERT}}$ assertion (if enabled in MFR_MODE).

CRC Memory Check

Upon reset, the device runs an internal algorithm to check the integrity of the key internal nonvolatile memory. If the CRC check fails, the device does not power up and remains in a null state with all pins high impedance but asserts the $\overline{\text{FAULT0}}$ output.

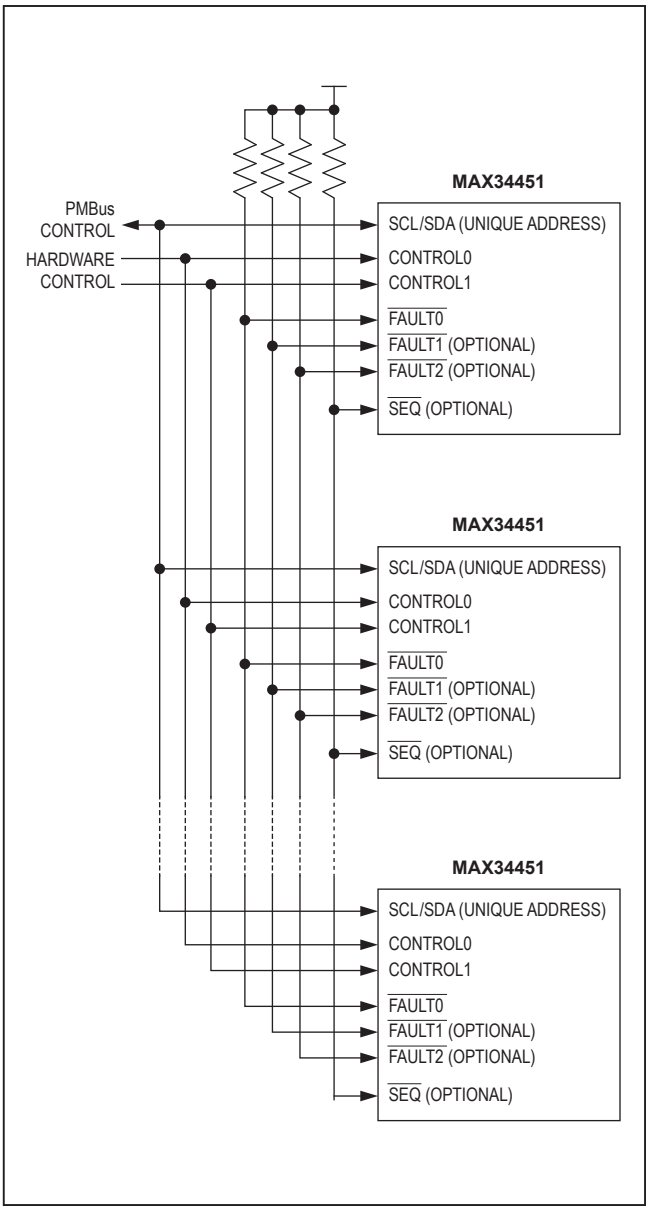


Figure 4. Multiple MAX34451 Hardware Connections

PMBus Commands

A summary of the PMBus commands supported by the device are described in the following sections.

PAGE (00h)

The device can monitor up to 16 voltages or currents, sequence up to 12 power supplies, and margin up to 12 power supplies. The device can monitor up to five temperature sensors, one internal local temperature sensor, plus four external remote temperature sensors (DS75LV). All the monitoring and control is accomplished using one PMBus (I^2C) address. Send the PAGE command with data 0–20 (decimal) to select which power supply or temperature sensor is affected by all the following PMBus commands. Not all commands are supported within each page. If an unsupported command is received, the CML status bit is set. Some commands are common, which means that any selected page has the same effect on and the same response from the device. See Table 9 for PAGE commands.

Set the PAGE to 255 when the following PMBus commands should apply to all pages at the same time. There are only a few commands (OPERATION, CLEAR_FAULTS) where this function has a real application.

Table 9. PAGE (00h) Commands

PAGE*	ASSOCIATED CONTROL
0	Power supply monitored by RS0, controlled by PSEN0, and margined with PWM0.
1	Power supply monitored by RS1, controlled by PSEN1, and margined with PWM1.
2	Power supply monitored by RS2, controlled by PSEN2, and margined with PWM2.
3	Power supply monitored by RS3, controlled by PSEN3, and margined with PWM3.
4	Power supply monitored by RS4, controlled by PSEN4, and margined with PWM4.
5	Power supply monitored by RS5, controlled by PSEN5, and margined with PWM5.
6	Power supply monitored by RS6, controlled by PSEN6, and margined with PWM6.
7	Power supply monitored by RS7, controlled by PSEN7, and margined with PWM7.
8	Power supply monitored by RS8, controlled by PSEN8, and optionally margined by OUT0 of external DS4424 at I^2C address A0h.
9	Power supply monitored by RS9, controlled by PSEN9, and optionally margined by OUT1 of external DS4424 at I^2C address A0h.
10	Power supply monitored by RS10, controlled by PSEN10, and optionally margined by OUT2 of external DS4424 at I^2C address A0h.
11	Power supply monitored by RS11, controlled by PSEN11, and optionally margined by OUT3 of external DS4424 at I^2C address A0h.
12	ADC channel 12 (monitors voltage or current) or GPI.
13	ADC channel 13 (monitors voltage or current) or GPI.
14	ADC channel 14 (monitors voltage or current) or GPI.
15	ADC channel 15 (monitors voltage or current) or GPI.
16	Internal temperature sensor.
17	External DS75LV temperature sensor with I^2C address 90h.
18	External DS75LV temperature sensor with I^2C address 92h.
19	External DS75LV temperature sensor with I^2C address 94h.
20	External DS75LV temperature sensor with I^2C address 96h.
21–254	Reserved.
255	Applies to all pages.

**PAGES 0–11 can also be used to configure GPI and GPO operation.*

OPERATION (01h)

The OPERATION command is used to turn the power supply on and off in conjunction with the CONTROLn input pin. The OPERATION command is also used to cause the power supply to set the output voltage to the upper or lower margin voltages. The power supply stays in the commanded operating mode until a subsequent OPERATION command or a change in the state of the CONTROLn pin (if enabled) instructs the power supply to change to another state. The valid OPERATION command byte values are shown in Table 10. The OPERATION command controls how the device responds when commanded to change the output. When the command byte is 00h, the device immediately turns the power supply off and ignores any programmed turn-off delay. When the command byte is set to 40h, 41h, or 42h the device powers down, according to the programmed turn-off delay. In Table 10, Table 11, and Table 12, "act on any fault" means that if any warning or fault on the selected power supply is detected when the output is margined, the device treats this as a warning or fault and responds as programmed. "Ignore all faults" means that all warnings and faults on the selected power supply are ignored. Any command value not shown in these tables is an invalid command. If the device receives a data byte that is not listed in these tables, then it treats this as invalid data, declares a data fault (sets CML bit and asserts $\overline{\text{ALERT}}$), and responds as described in the *Fault Management and Reporting* section.

In most cases, for power-on and power-off control, the OPERATION command should be sent when the PAGE is set to 255. If the PAGE is set to 0–11, the OPERATION command is only applied to the power supply on that page and the power supply is turned on and off using the associated TON_DELAY and TOFF_DELAY settings without any regard to the other supplies.

For individual channel-margining control, the OPERATION command can be used with the PAGE set to 0–11. When the PAGE is set to 255, the OPERATION margining commands affect all channels.

The OPERATION command for the device contains a few special values that are not part of the PMBus standard to allow the device to offer independent control. See the shaded values in Table 11.

Table 10. OPERATION (01h) Command Byte with PAGE = 0–11 (When Bit 3 of ON_OFF_CONFIG = 1)

COMMAND BYTE	POWER SUPPLY ON/OFF	MARGIN STATE
00h	Immediate off (no sequencing)	—
40h	Soft-off (with sequencing)	—
80h	On	Margin off
94h	On	Margin low (ignore all faults)
98h	On	Margin low (act on any fault)
A4h	On	Margin high (ignore all faults)
A8h	On	Margin high (act on any fault)

Note: *All enabled channels must exceed POWER_GOOD_ON for margining to begin.*

Table 11. OPERATION (01h) Command Byte with PAGE = 255 (When Bit 3 of ON_OFF_CONFIG = 1)

COMMAND BYTE	POWER SUPPLY ON/OFF	SEQUENCE AFFECTED	MARGIN STATE
00h	Immediate off (no sequencing)	SEQUENCE0 and SEQUENCE1	n/a
01h		SEQUENCE0 only	
02h		SEQUENCE1 only	
40h	Soft-off (with sequencing)	SEQUENCE0 and SEQUENCE1	
41h		SEQUENCE0 only	
42h		SEQUENCE1 only	
80h	On	SEQUENCE0 and SEQUENCE1	Margin off
81h		SEQUENCE0 only	
82h		SEQUENCE1 only	
94h	On	SEQUENCE0 and SEQUENCE1	Margin low (ignore all faults)
98h	On		Margin low (act on any fault)
A4h	On		Margin high (ignore all faults)
A8h	On		Margin high (act on any fault)

Note: *Special device OPERATION commands are shaded; when the OPERATION command is read, the device always responds with the standard command; all enabled channels must exceed POWER_GOOD_ON for margining to begin.*

Table 12. OPERATION (01h) Command Byte (When Bit 3 of ON_OFF_CONFIG = 0)

COMMAND BYTE	POWER SUPPLY ON/OFF	MARGIN STATE
00h	Command has no effect	n/a
40h		
80h		Margin off
94h		Margin low (ignore all faults)
98h		Margin low (act on any fault)
A4h		Margin high (ignore all faults)
A8h		Margin high (act on any fault)

Note: *The device only takes action if the supply is enabled; all enabled channels must exceed POWER_GOOD_ON for margining to begin; if PAGE is set to 255, both SEQUENCE0 and SEQUENCE1 are affected.*

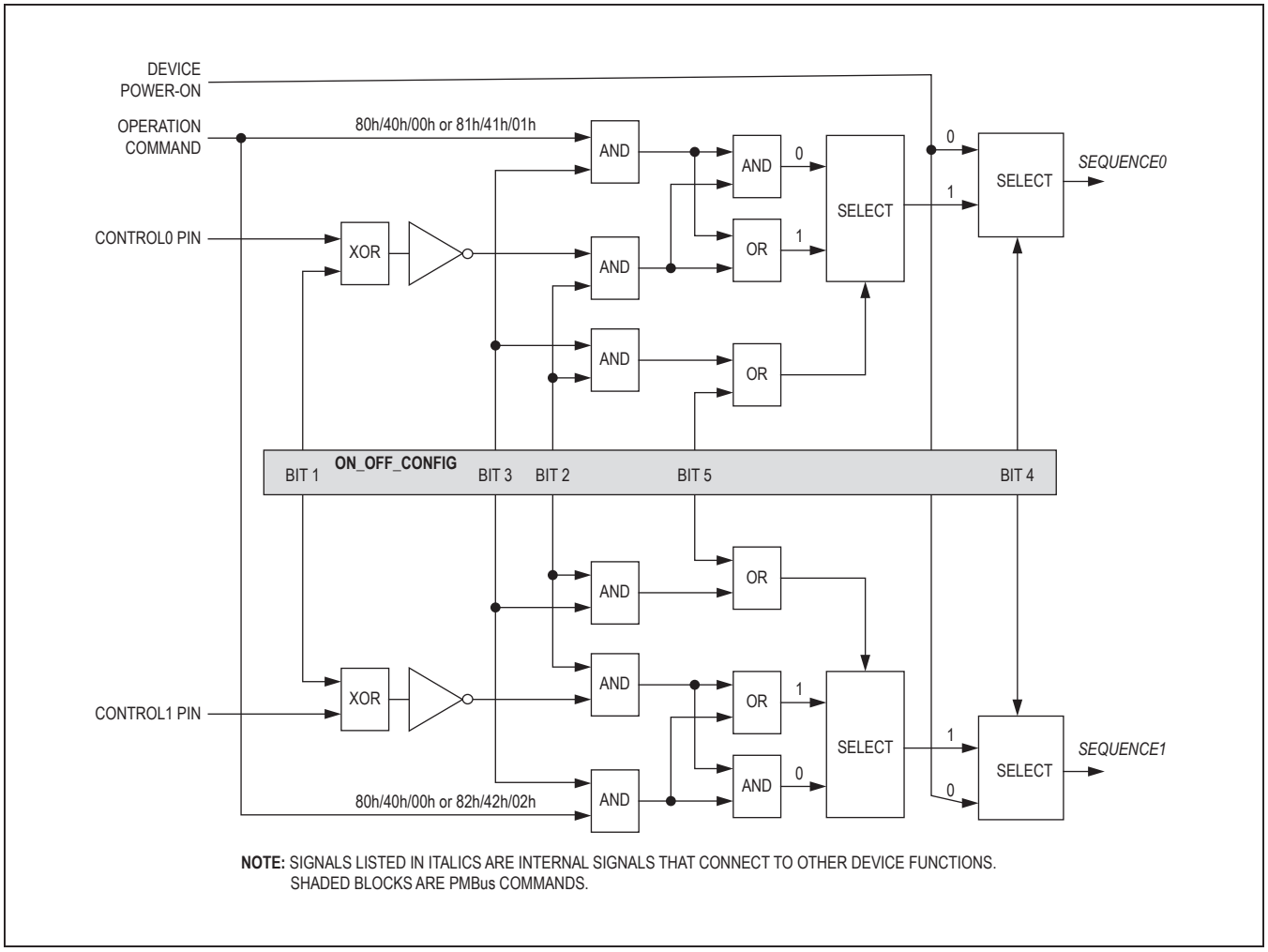
ON_OFF_CONFIG (02h)

The ON_OFF_CONFIG command configures the combination of the CONTROLn input and PMBus OPERATION commands needed to turn the power supply on and off. This indicates how the power supply is commanded when power is applied. The ON_OFF_CONFIG message content is described in Table 13. The host should not modify ON_OFF_CONFIG while the power supplies are active. The configuration of the ON_OFF_CONFIG command applies to both CONTROL0 and CONTROL1. See Figure 5.

Table 13. ON_OFF_CONFIG (02h) Command Byte

BIT	PURPOSE	VALUE	MEANING
7:6	Reserved.	n/a	Always returns 000.
5	OPERATION command and CONTROLn pin and/or select.	0	OPERATION command is ANDed with CONTROLn pin if both are enabled.
		1	OPERATION command is ORed with CONTROLn pin if both are enabled.
4	Turn on supplies when bias is present or use the CONTROLn pin/OPERATION command.	0	Turns on the supplies (with sequencing if so configured) as soon as bias is supplied to the device, regardless of the CONTROLn pin.
		1	Uses CONTROLn pins (if enabled) and/or OPERATION command (if enabled).*
3	OPERATION command enable.	0	On/off portion of the OPERATION command disabled.
		1	OPERATION command enabled.
2	CONTROLn pin enable.	0	CONTROLn pin disabled.
		1	CONTROLn pin enabled.
1	CONTROLn pin polarity.	0	Active low (drive low to turn on the power supplies).
		1	Active high (drive high to turn on the power supplies).
0	CONTROLn pin turn-off action.	0	Uses the programmed turn-off delay (soft-off).
		1	Turns off the power supplies immediately.

**Unless bit 5 is set (if both bits 3:2 are set), both the CONTROL0 or CONTROL1 pin and the OPERATION command are required to turn the supplies on, and either can turn the supplies off.*



NOTE: SIGNALS LISTED IN ITALICS ARE INTERNAL SIGNALS THAT CONNECT TO OTHER DEVICE FUNCTIONS.
SHADED BLOCKS ARE PMBus COMMANDS.

Figure 5. ON_OFF_CONFIG Logical Control

CLEAR_FAULTS (03h)

The CLEAR_FAULTS command is used to clear any latched fault or warning bits in the status registers that have been set and also unconditionally deasserts the $\overline{\text{ALERT}}$ output. This command clears all bits simultaneously. It also clears the POR and WATCHDOG_INT bits in the STATUS_MFR_SPECIFIC register. The CLEAR_FAULTS command does not cause a power supply that has latched off for a fault condition to restart. The state of the PSENn outputs under fault conditions are not affected by this command and changes only if commanded through the OPERATION command or the CONTROLn pins. If a fault is still present after the CLEAR_FAULTS command is executed, the fault status bit is immediately set again, but $\overline{\text{ALERT}}$ is not reasserted. $\overline{\text{ALERT}}$ is only asserted again when a new fault or warning is detected that occurs after the CLEAR_FAULTS command is executed. This command is write-only. There is no data byte for this command.

WRITE_PROTECT (10h)

The WRITE_PROTECT command is used to provide protection against accidental changes to the device's operating memory. All supported commands can have their parameters read, regardless of the WRITE_PROTECT settings. The WRITE_PROTECT message content is described in Table 14.

Device Configuration Data Management

The device stores configuration data in both nonvolatile flash memory and volatile RAM. The PMBus engine manages the device configuration data. See Figure 6.

The flash memory has three separate arrays for configuration parameters, whereas the RAM only has a single array. When a PMBus command is written to the device, it is always written to the RAM. When the device is shipped from the factory, the MAIN and BACKUP flash memory arrays are identical and are configured as shown in Table 3. The SINGLE array is empty.

There is a set of five PMBus commands that can be used to transfer data between the flash and RAM arrays. These commands are described in Table 15.

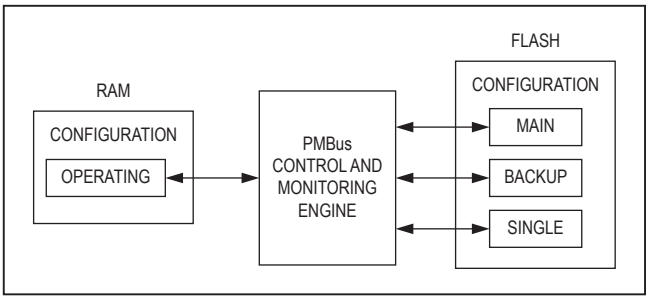


Figure 6. Device Configuration Data Management

Table 14. WRITE_PROTECT (10h) Command Byte

COMMAND BYTE	MEANING
80h	Disables all writes except the WRITE_PROTECT command.
40h	Disables all writes except the WRITE_PROTECT, OPERATION, and PAGE commands.
20h	Disables all writes except the WRITE_PROTECT, OPERATION, PAGE, and ON_OFF_CONFIG commands.
00h	Enables writes for all commands (default).

Note: *No fault or error is generated if the host attempts to write to a protected area.*

Table 15. Memory Transfer PMBus Commands

PMBus COMMAND		RESULTING MEMORY TRANSFER
STORE_DEFAULT_ALL		Copies RAM OPERATING to the flash MAIN.
RESTORE_DEFAULT_ALL		Copies the flash MAIN to RAM OPERATING.
MFR_STORE_ALL	CODE = 00h	Copies RAM OPERATING to the flash MAIN.
	CODE = 01h	Copies RAM OPERATING to the flash BACKUP.
MFR_RESTORE_ALL	CODE = 00h	Copies the flash MAIN array to RAM OPERATING.
	CODE = 01h	Copies the flash BACKUP to RAM OPERATING.
MFR_STORE_SINGLE		Copies RAM OPERATING (single parameter) to the flash SINGLE.

STORE_DEFAULT_ALL (11h)

The STORE_DEFAULT_ALL command instructs the device to copy RAM OPERATING to the flash MAIN memory array. Not all information is stored. Only configuration data is stored, not any status or operational data. If an error occurs during the transfer, $\overline{\text{ALERT}}$ asserts if enabled and the CML bit in STATUS_WORD is set to 1. No bits are set in STATUS_CML. This command is write-only. There is no data byte for this command.

When the STORE_DEFAULT_ALL command is invoked, the device is unresponsive to PMBus commands and does not monitor power supplies while transferring the configuration. The time required to complete this task is listed in the *Electrical Characteristics* section. The MFR_STORE_SINGLE command allows a single command to be stored in much less time.

USER NOTE: V_{DD} must be above 2.9V for the device to perform the STORE_DEFAULT_ALL command.

RESTORE_DEFAULT_ALL (12h)

The RESTORE_DEFAULT_ALL command instructs the device to copy the flash MAIN memory array to RAM OPERATING. The RESTORE_DEFAULT_ALL command should only be executed when the device is not operating the power supplies. This command is write-only. There is no data byte for this command. When RESTORE_DEFAULT_ALL is issued, the data is checked for validity before being transferred. If the MAIN array is corrupt, the device sets bit 1 of STATUS_CML and loads the BACKUP copy. If the BACKUP copy is corrupt, then the device sets bit 2 of STATUS_CML and remains in a null state with all pins (except SCL and SDA) in high impedance. The $\overline{\text{FAULTn}}$ pin(s) are also asserted. To resolve the data corruption, the configuration data must be written to RAM OPERATING and STORE_DEFAULT_ALL must be issued, followed by a device reset.

Upon a device power-on reset, or any device reset, this command is automatically executed by the device without PMBus action required.

MFR_STORE_ALL (EEh)

The MFR_STORE_ALL command instructs the device to copy RAM OPERATING to either the flash MAIN memory array (CODE = 00h) or the flash BACKUP memory array (CODE = 01h). This command is write-only. There is 1 data byte for this command, which is the CODE. The CODE is either 00h to instruct the device to copy into the MAIN array, or 01h to copy into the BACKUP array. All other CODE values are ignored. Not all information is stored. Only configuration data is stored, not any status or operational data. If an error occurs during the transfer, $\overline{\text{ALERT}}$ asserts if enabled and the CML bit in STATUS_WORD is set to 1. No bits are set in STATUS_CML. Note that if the CODE is 00h, then this command operates the same as STORE_DEFAULT_ALL.

CODE = 00h	Copy RAM OPERATING to flash MAIN
CODE = 01h	Copy RAM OPERATING to flash BACKUP

When the MFR_STORE_ALL command is invoked, the device is unresponsive to PMBus commands and does not monitor power supplies while transferring the configuration. The time required to complete this task is listed in the *Electrical Characteristics* section. The MFR_STORE_SINGLE command allows a single command to be stored in much less time.

USER NOTE: V_{DD} must be above 2.9V for the device to perform the MFR_STORE_ALL command.

MFR_RESTORE_ALL (EFh)

The MFR_RESTORE_ALL command instructs the device to copy either the flash MAIN memory array (CODE = 00h) or the flash BACKUP memory array (CODE = 01h) to RAM OPERATING. This command is write-only. There is 1 data byte for this command, which is the CODE. The CODE is either 00h to instruct the device to copy from the MAIN array or 01h to copy from the BACKUP array. All other CODE values are ignored. Note that if the CODE is 00h, then this command operates the same as RESTORE_DEFAULT_ALL.

CODE = 00h	Copy flash MAIN to RAM OPERATING
CODE = 01h	Copy flash BACKUP to RAM OPERATING

The MFR_RESTORE_ALL command should only be executed when the device is not operating the power supplies.

When MFR_RESTORE_ALL is issued, the data is checked for validity before being transferred. If the MAIN array is corrupt, the device sets bit 1 of STATUS_CML. If the BACKUP array is corrupt, then the device sets bit 2 of STATUS_CML. No other action is taken by the device. To resolve the data corruption, the configuration data must be written to RAM OPERATING and STORE_DEFAULT_ALL or MFR_STORE_ALL must be issued.

MFR_STORE_SINGLE (FCh)

MFR_STORE_SINGLE is a read/write word command that instructs the device to transfer a single configuration parameter from RAM OPERATING to the flash SINGLE memory array. The upper byte contains the PAGE and the lower byte contains the PMBus command that should be stored. For example, if the TON_DELAY parameter for the power supply controlled by PAGE 4 needs to be stored to flash, 0460h would be written with this command. When read, this command reports the last single PAGE/ command written to flash. This command can be used while the device is operating the power supplies. If an error occurs during the transfer, $\overline{\text{ALERT}}$ asserts if enabled and the CML bit in STATUS_WORD is set to 1. No bits are set in STATUS_CML. The MFR_STORE_SINGLE command should only be invoked a maximum of 85 times before either a device reset is issued or a device power cycle occurs, or the RESTORE_DEFAULT_ALL command is invoked. Once the MFR_STORE_SINGLE command is invoked, the STORE_DEFAULT_ALL and MFR_STORE_ALL commands should not be used until either a device reset is issued or a device power cycle occurs, or the RESTORE_DEFAULT_ALL command is invoked. Also, MFR_STORE_SINGLE should not be used for commands that are not stored in flash. See Table 3 for a list of commands that are stored in flash.

USER NOTE: V_{DD} must be above 2.9V for the device to perform the MFR_STORE_SINGLE command.

MFR_CRC (FEh)

MFR_CRC is a read/write word command that instructs the device to report the calculated 16-bit CRC value of either the RAM OPERATING or the flash MAIN or BACKUP memory arrays. A CRC value for the flash SINGLE array is not available. Only one 16-bit CRC is reported with each read of MFR_CRC. The CRC value to be reported is determined by the most previous written CODE value, as shown in Table 16. For example, if MFR_CRC is first written with a CODE of 0001h, then the next read of MFR_CRC reports the CRC for the flash BACKUP array. If no CODE value is written, than MFR_CRC returns FFFFh when read. See Table 16.

CAPABILITY (19h)

The CAPABILITY command is used to determine some key capabilities of the device. The CAPABILITY command is read-only. The message content is described in Table 17.

VOUT_MODE (20h)

The VOUT_MODE command is used to report the data format of the device. The device uses the DIRECT format for all the voltage-related commands. The value returned is 40h, indicating DIRECT data format. This command is read-only. If a host attempts to write this command, the CML status bit is asserted. See Table 5 for the m, b, and R values for the various commands.

VOUT_MARGIN_HIGH (25h)

The VOUT_MARGIN_HIGH command loads the device with the voltage to which the power-supply output is to be changed when the OPERATION command is set to margin high. If the power supply is already operating at

Table 16. MFR_CRC (FEh) Command Byte

MFR_CRC CODE VALUE	MEMORY ARRAY CRC VALUE TO BE REPORTED ON NEXT READ OF MFR_CRC
0000h	Flash MAIN
0001h	Flash BACKUP
0002h	RAM OPERATING

Table 17. CAPABILITY (19h) Command Byte

BIT	NAME	MEANING
7	Packet-error checking	0 = PEC not supported.
6:5	PMBus speed	01 = Maximum supported bus speed is 400kHz.
4	ALERT	1 = Device supports an $\overline{\text{ALERT}}$ output ($\overline{\text{ALERT}}$ is enabled in MFR_MODE). 0 = Device does not support $\overline{\text{ALERT}}$ output (ALERT is disabled in MFR_MODE).
3:0	Reserved	Always returns 0000.

margin high, changing VOUT_MARGIN_HIGH has no effect on the output voltage. The device only adjusts the power supply to the new VOUT_MARGIN_HIGH voltage after receiving a new margin-high OPERATION command. The 2 data bytes are in DIRECT format. If the device cannot successfully close-loop margin the power supply, the device keeps attempting to margin the supply and does the following:

1) Sets the MARGIN bit in STATUS_WORD.
2) Sets the MARGIN_FAULT bit in STATUS_MFR_SPECIFIC (PAGES 0–11).
3) Notifies the host through $\overline{\text{ALERT}}$ assertion (if enabled in MFR_MODE).

VOUT_MARGIN_LOW (26h)

The VOUT_MARGIN_LOW command loads the device with the voltage to which the power-supply output changes to when the OPERATION command is set to margin low. If the power supply is already operating at margin low, changing VOUT_MARGIN_LOW has no effect on the output voltage. The device only adjusts the power supply to the new VOUT_MARGIN_LOW voltage after receiving a new margin-low OPERATION command. The 2 data bytes are in DIRECT format. If the device cannot successfully close-loop margin the power supply, the device keeps attempting to margin the supply and does the following:

1) Sets the MARGIN bit in STATUS_WORD
2) Sets the MARGIN_FAULT bit in STATUS_MFR_SPECIFIC (PAGES 0–11)
3) Notifies the host through $\overline{\text{ALERT}}$ assertion (if enabled in MFR_MODE).

VOUT_SCALE_MONITOR (2Ah)

In applications where the measured power-supply voltage is not equal to the voltage at the ADC input, VOUT_SCALE_MONITOR is used. For example, if the ADC input expects a 1.8V input for a 12V output, VOUT_SCALE_MONITOR = 1.8V/12V = 0.15. In applications where the power-supply output voltage is greater than the device input range of 2.048V, the output voltage of the power supply is sensed through a resistive voltage-divider. The resistive voltage-divider reduces or scales the output voltage. The PMBus commands specify the actual power-supply output voltages and not the input voltage to the ADC. To allow the device to map between the high power-supply voltages (such as 12V) and the voltage at the ADC input, the VOUT_SCALE_MONITOR command is used. The 2 data bytes are in DIRECT format. This value is dimensionless. As an example, if the required scaling factor is 0.15, then VOUT_SCALE_MONITOR should be set to 1333h (4915/32,767 = 0.15). See Table 18.

IOUT_CAL_GAIN (38h)

The IOUT_CAL_GAIN command is used to set the ratio of the voltage at the ADC input to the sensed current. The units of the IOUT_CAL_GAIN factor are 0.1mΩ. The 2 data bytes are in DIRECT format. As an example, if a 10mΩ sense resistor is used in conjunction with a 50V/V current-sense amplifier, the IOUT_CAL_GAIN should be set to 500mΩ or 1388h.

USER NOTE: The full-scale ADC voltage on the device is 2.048V. The value of the sense resistor and current-sense amplifier gain must be scaled appropriately. Also, the maximum voltage at the RSn inputs must be less than 4V. The maximum output impedance of the current-sense amplifier is limited by the setting of the ADC_TIME bits in MFR_MODE. See the *Recommended Operating Conditions* section for details.

Table 18. VOUT_SCALE_MONITOR (2Ah) Examples

NOMINAL VOLTAGE LEVEL MONITORED	NOMINAL ADC INPUT VOLTAGE LEVEL*	RESISTIVE DIVIDER RATIO	VOUT_SCALE_MONITOR VALUE
1.8V or less	1.8V	1.0	7FFFh
2.5V	1.8V	0.72	5C28h
3.3V	1.8V	0.545454	45D1h
5V	1.8V	0.36	2E14h
12V	1.8V	0.15	1333h

**The full-scale ADC voltage on the device is 2.048V. A scaling factor where a 1.8V ADC input represents a nominal 100% voltage level is recommended to allow headroom for margining. Resistor-dividers must be used to measure voltage greater than 1.8V. The maximum source impedance of the resistor-divider is limited by the setting of the ADC_TIME bits in MFR_MODE. See the* Recommended Operating Conditions *section for details.*

VOUT_OV_FAULT_LIMIT (40h)

The VOUT_OV_FAULT_LIMIT command sets the value of the output voltage that causes an output overvoltage fault. The monitored voltage must drop by at least 2% below the limit before the fault is allowed to clear. The 2 data bytes are in DIRECT format. In response to the VOUT_OV_FAULT_LIMIT being exceeded, the device does the following:

1) Sets the VOUT_OV bit and the VOUT bit in STATUS_WORD.
2) Sets the VOUT_OV_FAULT bit in STATUS_VOUT.
3) Responds as specified in the MFR_FAULT_RESPONSE.
4) Notifies the host through $\overline{\text{ALERT}}$ assertion (if enabled in MFR_MODE).

VOUT_OV_WARN_LIMIT (42h)

The VOUT_OV_WARN_LIMIT command sets the value of the output voltage that causes an output-voltage high warning. The monitored voltage must drop by at least 2% below the limit before the warning is allowed to clear. This value is typically less than the output overvoltage threshold in VOUT_OV_FAULT_LIMIT. The 2 data bytes are in DIRECT format. In response to the VOUT_OV_WARN_LIMIT being exceeded, the device does the following:

1) Sets the VOUT bit in STATUS_WORD.
2) Sets the VOUT_OV_WARN bit in STATUS_VOUT.
3) Notifies the host using $\overline{\text{ALERT}}$ assertion (if enabled in MFR_MODE).

VOUT_UV_WARN_LIMIT (43h)

The VOUT_UV_WARN_LIMIT command sets the value of the output voltage that causes an output-voltage low warning. The monitored voltage must increase by at least 2% above the limit before the warning is allowed to clear. This value is typically greater than the output undervoltage-fault threshold in VOUT_UV_FAULT_LIMIT. This warning is masked until the output voltage reaches the programmed POWER_GOOD_ON for the first time and also during turn-off when the power supply is disabled. If voltage is being monitored, this should be set to a value greater than 100mV. The 2 data bytes are in DIRECT format. In response to violation of the VOUT_UV_WARN_LIMIT, the device does the following:

1) Sets the VOUT bit in STATUS_WORD.
2) Sets the VOUT_UV_WARN bit in STATUS_VOUT.
3) Notifies the host using $\overline{\text{ALERT}}$ assertion (if enabled in MFR_MODE).

VOUT_UV_FAULT_LIMIT (44h)

The VOUT_UV_FAULT_LIMIT command sets the value of the output voltage that causes an output undervoltage fault. The monitored voltage must increase by at least 2% above the limit before the fault is allowed to clear. This fault is masked until the output voltage reaches the programmed POWER_GOOD_ON for the first time and also during turn-off when the power supply is disabled. If voltage is being monitored, this should be set to a value greater than 100mV. The 2 data bytes are in DIRECT format. In response to violation of the VOUT_UV_FAULT_LIMIT, the device does the following:

1) Sets the VOUT bit in STATUS_WORD.
2) Sets the VOUT_UV_FAULT bit in STATUS_VOUT.
3) Responds as specified in MFR_FAULT_RESPONSE.
4) Notifies the host using $\overline{\text{ALERT}}$ assertion (if enabled in MFR_MODE).

IOUT_OC_WARN_LIMIT (46h)

The IOUT_OC_WARN_LIMIT command sets the value of the current that causes an overcurrent warning. The monitored current must decrease by at least 5% below the limit before the warning is allowed to clear. This value is typically less than the overcurrent-fault threshold in IOUT_OC_FAULT_LIMIT. The 2 data bytes are in DIRECT format. In response to violation of the IOUT_OC_WARN_LIMIT, the device does the following:

1) Sets the IOUT bit in STATUS_WORD.
2) Sets the IOUT_OC_WARN bit in STATUS_IOUT.
3) Notifies the host using $\overline{\text{ALERT}}$ assertion (if enabled in MFR_MODE).

IOUT_OC_FAULT_LIMIT (4Ah)

The IOUT_OC_FAULT_LIMIT command sets the value of the current that causes an overcurrent fault. The monitored current must decrease by at least 5% below the limit before the fault is allowed to clear. This fault is masked until the current is below this limit for the first time. The 2 data bytes are in DIRECT format. In response to violation of the IOUT_OC_FAULT_LIMIT, the device does the following:

1) Sets the IOUT bit in STATUS_WORD.
2) Sets the IOUT_OC_FAULT bit in STATUS_IOUT.
3) Responds as specified in the MFR_FAULT_RESPONSE.
4) Notifies the host using $\overline{\text{ALERT}}$ assertion (if enabled in MFR_MODE).

OT_FAULT_LIMIT (4Fh)

The OT_FAULT_LIMIT command sets the temperature, in degrees Celsius, of the selected temperature sensor at which an overtemperature fault is detected. The monitored temperature must drop by at least 4°C below the limit before the fault is allowed to clear. The 2 data bytes are in DIRECT format. In response to the OT_FAULT_LIMIT being exceeded, the device does the following:

1) Sets the TEMPERATURE bit in STATUS_WORD.
2) Sets the OT_FAULT bit in STATUS_TEMPERATURE register.
3) Responds as specified in the MFR_FAULT_RESPONSE.
4) Notifies the host using $\overline{\text{ALERT}}$ assertion (if enabled in MFR_MODE).

OT_WARN_LIMIT (51h)

The OT_WARN_LIMIT command sets the temperature, in degrees Celsius, of the selected temperature sensor at which an overtemperature warning is detected. The monitored temperature must drop by at least 4°C below the limit before the warning is allowed to clear. The 2 data bytes are in DIRECT format. In response to the OT_WARN_LIMIT being exceeded, the device does the following:

1) Sets the TEMPERATURE bit in STATUS_WORD.
2) Sets the OT_WARN bit in STATUS_TEMPERATURE register.
3) Notifies the host through $\overline{\text{ALERT}}$ assertion (if enabled in MFR_MODE).

POWER_GOOD_ON (5Eh)

The POWER_GOOD_ON command sets the value of the output voltage that the channel must exceed for a power-good state to be declared on this channel. All channels configured as voltage monitoring, with or without sequencing, should have their voltages above POWER_GOOD_ON for power-supply margining to begin. The POWER_GOOD_ON threshold is also used to determine if TON_MAX_FAULT_LIMIT is exceeded. The POWER_GOOD_ON level should always be set higher than the POWER_GOOD_OFF level. The 2 data bytes are in DIRECT format.

POWER_GOOD_ON should be set higher than VOUT_UV_FAULT_LIMIT and VOUT_UV_WARN_LIMIT because their functionality does not become active until the measured output voltage rises above the POWER_GOOD_ON threshold.

POWER_GOOD_OFF (5Fh)

The POWER_GOOD_OFF command sets the value of the output voltage that causes the power-good state on this channel to deassert after it has been asserted. The POWER_GOOD_OFF level should always be set lower than the POWER_GOOD_ON level. The 2 data bytes are in DIRECT format.

When the VOUT level of a power supply falls from greater than POWER_GOOD_ON to less than POWER_GOOD_OFF, the device does the following:

1) Sets the POWER_GOOD# bit in STATUS_WORD.
2) Sets the POWER_GOOD# bit in STATUS_MFR_SPECIFIC register (PAGES 0–11).

Note: If the POWER_GOOD_ON value is configured to be lower than the POWER_GOOD_OFF value, the device sets the POWER_GOOD_OFF to be equal to the POWER_GOOD_ON value. Conversely, if the POWER_GOOD_OFF value is configured to be higher than the POWER_GOOD_ON value, the device sets the POWER_GOOD_ON to be equal to the POWER_GOOD_OFF value.

TON_DELAY (60h)

In the PMBus sequencing configuration, TON_DELAY sets the time, in milliseconds, from when a START condition is received until the PSENn output is asserted. If the PSENn/GPOn output has been configured (with the MFR_PSEN_CONFIG command) as a PG/GPI or alarm, then this command can be used to delay the assertion of the output. The 2 data bytes are in DIRECT format.

TOFF_DELAY (64h)

TOFF_DELAY sets the time, in milliseconds, from when a STOP condition is received (a soft-off OPERATION command, or through the CONTROLn pins when enabled) until the PSENn output is deasserted. When commanded to turn off immediately (either through the OPERATION command or the CONTROLn pins), the TOFF_DELAY value is ignored. If the PSENn/GPOn output has been configured (with the MFR_PSEN_CONFIG command) as a PG/GPI or alarm, then this command can be used to delay the deassertion of the output. The 2 data bytes are in DIRECT format.

Note: For GPOs configured to assert/deassert based on Power Good (PG) condition of monitored channels: If the voltage of a monitored channel repeatedly crosses over the PG threshold during the GPO's TON_DELAY or TOFF_DELAY time, rarely the GPO might get asserted or deasserted incorrectly at the end of the delay time. To avoid this scenario, configure both TON_DELAY and TOFF_DELAY to 0ms. Alternatively, set both TON_DELAY and TOFF_DELAY to nonzero values.

TON_MAX_FAULT_LIMIT (62h)

TON_MAX_FAULT_LIMIT sets an upper time limit, in milliseconds, from when the PSENn output is asserted until the output voltage crosses the POWER_GOOD_ON threshold. The 2 data bytes are in DIRECT format. If the value is zero, then the limit is disabled. In response to the TON_MAX_FAULT_LIMIT being exceeded, the device does the following:

1) Sets the VOUT bit in STATUS_WORD.
2) Sets the TON_MAX_FAULT bit in STATUS_VOUT.
3) Responds as specified in the MFR_FAULT_RESPONSE.
4) Notifies the host using $\overline{\text{ALERT}}$ assertion (if enabled in MFR_MODE).

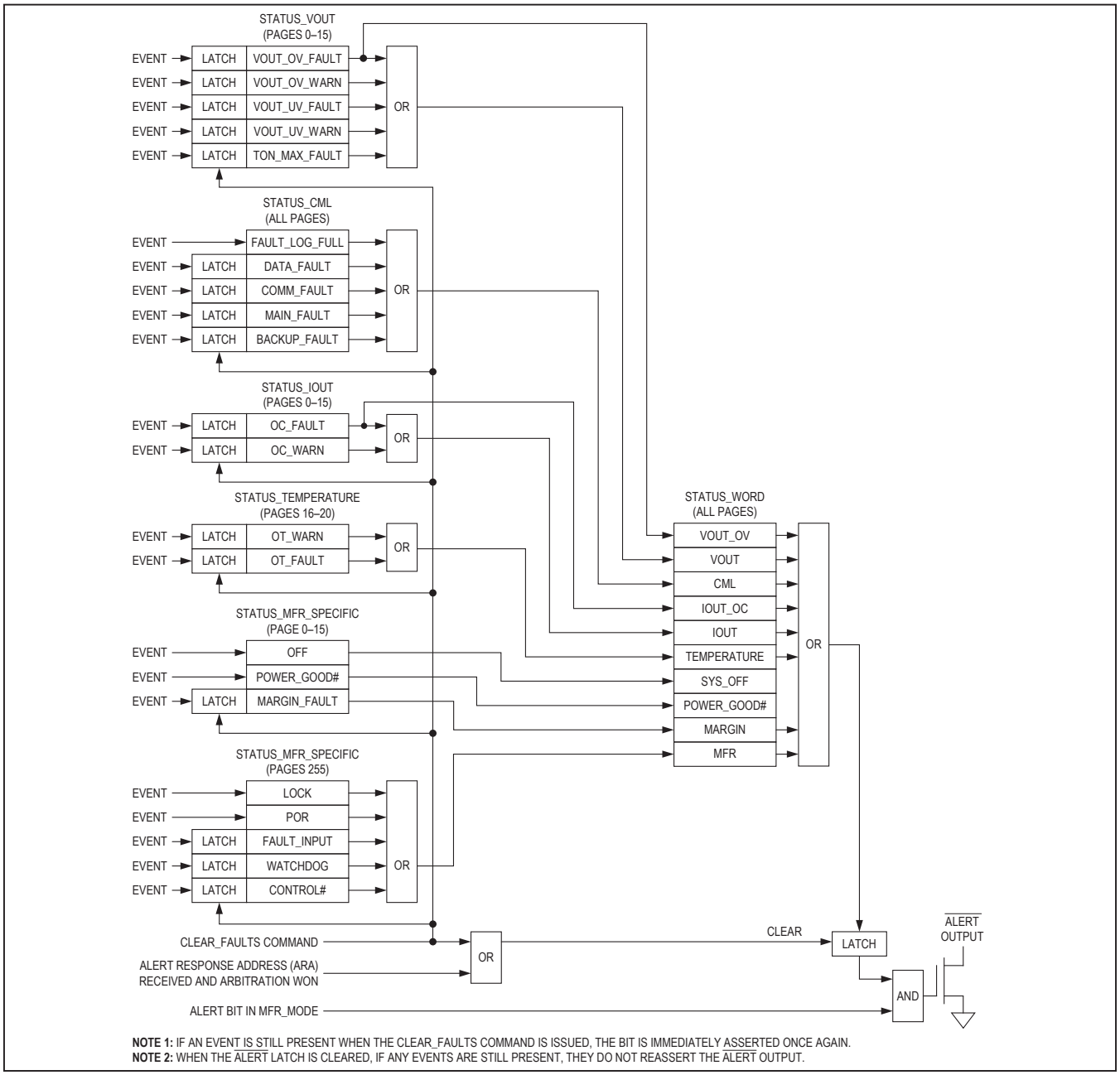


Figure 7. Status Register Organization

STATUS_WORD (79h)

The STATUS_WORD command returns 2 bytes of information with a summary of the reason for a fault. The STATUS_WORD message content is described in Table 19.

STATUS_VOUT (7Ah)

The STATUS_VOUT command returns 1 byte of information with contents, as described in Table 20. All the bits in STATUS_VOUT are latched. When cleared, the bits are set again if the condition persists, or in the case of TON_MAX_FAULT, when the event occurs again.

Table 19. STATUS_WORD (79h)

BIT	NAME	MEANING
15	VOUT	An output voltage fault or warning, or TON_MAX_FAULT_LIMIT or MFR_TON_SEQ_MAX has occurred.
14	IOUT	An overcurrent fault or warning has occurred.
13	0	This bit always returns a 0.
12	MFR	A bit in STATUS_MFR_SPECIFIC (PAGE = 255) has been set.
11	POWER_GOOD#	Any power-supply voltage has fallen from POWER_GOOD_ON to less than POWER_GOOD_OFF (logical OR of all the POWER_GOOD# bits in STATUS_MFR_SPECIFIC).
10	0	This bit always returns a 0.
9	0	This bit always returns a 0.
8	MARGIN	A margining fault has occurred.
7	0	This bit always returns a 0.
6	SYS_OFF	Set when any of the power supplies are sequenced off (logical OR of all the OFF bits in STATUS_MFR_SPECIFC).
5	VOUT_OV	An overvoltage fault has occurred.
4	IOUT_OC	An overcurrent fault has occurred.
3	0	This bit always returns a 0.
2	TEMPERATURE	A temperature fault or warning has occurred.
1	CML	A communication, memory, or logic fault has occurred.
0	0	This bit always returns a 0.

Note: *The setting of the SYS_OFF and POWER_GOOD# bits do not assert the* $\overline{ALERT}$ *signal.*

Table 20. STATUS_VOUT (7Ah)

BIT	NAME	MEANING	LATCHED
7	VOUT_OV_FAULT	VOUT overvoltage fault.	Yes
6	VOUT_OV_WARN	VOUT overvoltage warning.	Yes
5	VOUT_UV_WARN	VOUT undervoltage warning.	Yes
4	VOUT_UV_FAULT	VOUT undervoltage fault.	Yes
3	0	This bit always returns a 0.	—
2	TON_MAX_FAULT	TON_MAX_FAULT_LIMIT or MFR_TON_SEQ_MAX fault.	Yes
1	0	This bit always returns a 0.	—
0	0	This bit always returns a 0.	—

STATUS_IOUT (7Bh)

The STATUS_IOUT command returns 1 byte of information with contents, as described in Table 21. All the bits in STATUS_IOUT are latched. When cleared, the bits are set again if the condition persists.

STATUS_TEMPERATURE (7Dh)

The STATUS_TEMPERATURE command returns 1 byte of information with contents, as described in Table 22. All the bits in STATUS_VOUT are latched. When cleared, the bits are set again if the condition persists.

STATUS_CML (7Eh)

The STATUS_CML command returns 1 byte of information with contents, as described in Table 23. The COMM_FAULT, DATA_FAULT, MAIN_FAULT, and BACKUP_FAULT bits are latched. When cleared, the bits are set again when the event occurs again. The FAULT_LOG_FULL bit reflects the current real-time state of the fault log.

Table 21. STATUS_IOUT (7Bh)

BIT	NAME	MEANING	LATCHED
7	IOUT_OC_FAULT	IOUT overcurrent fault.	Yes
6	0	This bit always returns a 0.	—
5	IOUT_OC_WARN	IOUT overcurrent warning.	Yes
4	0	This bit always returns a 0.	—
3	0	This bit always returns a 0.	—
2	0	This bit always returns a 0.	—
1	0	This bit always returns a 0.	—
0	0	This bit always returns a 0.	—

Table 22. STATUS_TEMPERATURE (7Dh)

BIT	NAME	MEANING	LATCHED
7	OT_FAULT	Overtemperature fault.	Yes
6	OT_WARN	Overtemperature warning.	Yes
5	0	This bit always returns a 0.	—
4	0	This bit always returns a 0.	—
3	0	This bit always returns a 0.	—
2	0	This bit always returns a 0.	—
1	0	This bit always returns a 0.	—
0	0	This bit always returns a 0.	—

Table 23. STATUS_CML (7Eh)

BIT	NAME	MEANING	LATCHED
7	COMM_FAULT	An invalid or unsupported command has been received.	Yes
6	DATA_FAULT	An invalid or unsupported data has been received.	Yes
5	0	This bit always returns a 0.	—
4	0	This bit always returns a 0.	—
3	0	This bit always returns a 0.	—
2	BACKUP_FAULT	Flash BACKUP memory array is corrupt.	Yes
1	MAIN_FAULT	Flash MAIN memory array is corrupt.	Yes
0	FAULT_LOG_FULL	MFR_NV_FAULT_LOG is full and needs to be cleared.	No

Notes: *When the NV fault log overwrite is enabled (NV_LOG_OVERWRITE = 1 in MFR_MODE), FAULT_LOG_FULL is set when the fault log is full, but clears when the fault log is overwritten since two fault logs are cleared before each overwrite; the setting of the BACKUP_FAULT and MAIN_FAULT bits do not assert the $\overline{ALERT}$ signal.*

STATUS_MFR_SPECIFIC (80h)

The STATUS_MFR_SPECIFIC message content varies based on the selected PAGE, and is described in Table 24 and Table 25.

Table 24. STATUS_MFR_SPECIFIC (80h) (for PAGES 0–11)

BIT	NAME	MEANING	LATCHED
7	OFF	For enabled channels, this bit reflects the output state of the sequencer and is set when PSENn is not asserted due to either a sequencing delay or a fault, or the power supply being turned off. This bit is always cleared when the channel is disabled. If PSENn is reconfigured as a GPO, this bit does not reflect the state of the pin.	No
6	0	This bit always returns a 0.	—
5	0	This bit always returns a 0.	—
4	0	This bit always returns a 0.	—
3	MARGIN_FAULT	This bit is set if the device cannot properly close-loop margin the power supply.	Yes
2	POWER_GOOD#	This bit is set when the power-supply voltage has fallen from POWER_GOOD_ON to less than POWER_GOOD_OFF. On device reset, this bit is set until the power supply is greater than POWER_GOOD_ON.	No
1	0	This bit always returns a 0.	—
0	0	This bit always returns a 0.	—

Note: *The setting of the OFF and POWER_GOOD# bits do not assert the* $\overline{ALERT}$ *signal.*

Table 25. STATUS_MFR_SPECIFIC (for PAGE 255)

BIT	NAME	MEANING	LATCHED
7	LOCK	Set when the device is password protected (Note 1).	No
6	FAULT_INPUT	Set each time any of the $\overline{FAULTn}$ inputs are pulled low (Note 2).	Yes
5	POR	Set each time a device Power On Reset (POR) occurs (Note 4).	Yes
4	WATCHDOG_INT	Set upon device reset when the internal watchdog has caused the device reset (Note4).	Yes
3	CONTROL#	Set each time the CONTROLn inputs are deasserted (Note 3).	Yes
2	0	This bit always returns a 0.	—
1	0	This bit always returns a 0.	—
0	0	This bit always returns a 0.	—

Note 1: *Setting the LOCK bit or the POR bit does not assert the ALERT signal.*

Note 2: *Applies to all FAULTn inputs. The fault status bit is set even if the FAULTn pin is configured in MFR_NV_LOG_CONFIG to ignore FAULTn pins. If FAULT1 and FAULT2 are disabled, they do not affect this bit.*

Note 3: *Either the CONTROL0 or CONTROL1 pin can set this bit. ON_OFF_CONFIG must be configured to use the CONTROLn pins for this status bit to function.*

Note 4: *This bit is latched when set and can be cleared by either issuing the CLEAR_FAULTS command or by reading the STATUS_MFR_SPECIFIC register.*

READ_VOUT (8Bh)

The READ_VOUT command returns the actual measured (not commanded) output voltage. READ_VOUT is measured and updated every 5ms. If the RSn/GPIn is configured to be a general-purpose input (GPI), by configuring the SELECT bits in MFR_CHANNEL_CONFIG to either 30h or 34h, then READ_VOUT reports 0000h when the GPIn input is inactive and 0001h when the GPIn input is active. The 2 data bytes are in DIRECT format.

READ_IOUT (8Ch)

The READ_IOUT command returns the latest measured current value. READ_IOUT is measured and updated every 5ms. The 2 data bytes are in DIRECT format.

READ_TEMPERATURE_1 (8Dh)

The READ_TEMPERATURE_1 command returns the temperature returned from the temperature sensor. READ_TEMPERATURE_1 returns 7FFFh if the sensor is faulty and 0000h if the sensor is disabled. READ_TEMPERATURE_1 is measured and updated once per second. The 2 data bytes are in DIRECT format.

PMBUS_REVISION (98h)

The PMBUS_REVISION command returns the revision of the PMBus specification to which the device is compliant. The command has 1 data byte. Bits 7:4 indicate the revision of the PMBus specification Part I to which the device is compliant. Bits 3:0 indicate the revision of the PMBus specification Part II to which the device is compliant. This command is read-only. The PMBUS_REVISION value returned is always 11h, which indicates that the device is compliant with Part I, Rev 1.1 and Part II, Rev 1.1.

MFR_ID (99h)

The MFR_ID command returns the text (ISO/IEC 8859-1) character of the manufacturer's (Maxim) identification. The default MFR_ID value is 4Dh (M). This command is read-only.

MFR_MODEL (9Ah)

The MFR_MODEL command returns the text (ISO/IEC 8859-1) character of the device model number. The default MFR_MODEL value is 59h (Y). This command is read-only.

MFR_REVISION (9Bh)

The MFR_REVISION command returns two text (ISO/IEC 8859-1) characters that contain the device revision numbers for hardware (upper byte) and firmware (lower byte). This command is read-only.

MFR_LOCATION (9Ch)

The MFR_LOCATION command loads the device with text (ISO/IEC 8859-1) characters that identify the facility that manufactures the power supply. The maximum number of characters is 8. This data is written to internal flash using the STORE_DEFAULT_ALL command. The factory-default text string value is 10101010.

MFR_DATE (9Dh)

The MFR_DATE command loads the device with text (ISO/IEC 8859-1) characters that identify the date of manufacture of the power supply. The maximum number of characters is 8. This data is written to internal flash using the STORE_DEFAULT_ALL command. The factory-default text string value is 10101010.

MFR_SERIAL (9Eh)

The MFR_SERIAL command loads the device with text (ISO/IEC 8859-1) characters that uniquely identify the device. The maximum number of characters is 8. This data is written to internal flash using the STORE_DEFAULT_ALL command. The factory default text string value is 10101010. The upper 4 bytes of MFR_SERIAL are used to unlock a device that has been password protected. The lower 4 bytes of MFR_SERIAL are not used to unlock a device and they can be set to any value.

MFR_MODE (D1h)

The MFR_MODE command is used to configure the device to support manufacturer-specific commands. The MFR_MODE command should not be changed while power supplies are operating. The MFR_MODE command is described in Table 26.

Table 26. MFR_MODE (D1h)

BIT	NAME	MEANING
15:14	0	These bits always return a 0.
13	ALERT	0 = $\overline{\text{ALERT}}$ disabled (device does not respond to ARA). 1 = $\overline{\text{ALERT}}$ enabled (device does respond to ARA).
12	0	This bit always returns a 0.
11	SOFT_RESET	This bit must be set, then cleared and set again within 8ms for a soft reset to occur.
10	LOCK	This bit must be set, then cleared and set again within 8ms for the device to become password protected. This bit is cleared when the password is unlocked. The device should only be locked and then unlocked a maximum of 256 times before either a device reset is issued or a device power cycle occurs.
9:8	0	These bits always return a 0.
7:6	ADC_TIME[1:0]	These bits select the ADC conversion time: ADC_TIME[1:0] — ADC CONVERSION TIME 00 — 1µs 01 — 2µs 10 — 4µs 11 — 8µs
5:4	ADC_AVERAGE[1:0]	These bits select the post ADC conversion averaging: ADC_AVERAGE[1:0] — ADC AVERAGING 00 — No Averaging 01 — Average 2 Samples 10 — Average 4 Samples 11 — Average 8 Samples
3:0	IOUT_AVG[3:0]	These bits determine the number of samples to average before reporting the value in MFR_IOUT_AVG: IOUT_AVG[3:0] — AVERAGING — IOUT_AVG[3:0] — AVERAGING 0000 — 1 Sample — 1000 — 256 Samples 0001 — 2 Samples — 1001 — 512 Samples 0010 — 4 Samples — 1010 — 1024 Samples 0011 — 8 Samples — 1011 — 2048 Samples 0100 — 16 Samples — 1100 — 4096 Samples 0101 — 32 Samples — 1101 — 8192 Samples 0110 — 64 Samples — 1110 — 16,384 Samples 0111 — 128 Samples — 1111 — 32,768 Samples

MFR_PSEN_CONFIG (D2h)

The MFR_PSEN_CONFIG command is used to configure the individual PSENn/GPOn (where n = 0–11) outputs. This command should not be changed while the power supplies are operating. The MFR_PSEN_CONFIG command is described in Table 27 and shown in Figure 8.

Each PSENn/GPOn pin can be independently configured using the SELECT[2:0] bits to one of the following:

- Enable and disabled power supplies (SELECT[2:0] = 000)
- Force pin assertion (SELECT[2:0] = 001)
- Force pin deassertion (SELECT[2:0] = 010)
- Assert when all enabled channel power-good (PG) or GPI are asserted (SELECT[2:0] = 011)
- Assert when any enabled alarm goes active (SELECT[2:0] = 100)

If the PSENn/GPOn output is configured to enable and disable power supplies (SELECT[2:0] = 000), then the associated input channel must also be configured to monitor voltage and to sequence by setting the SELECT bits in MFR_CHANNEL_CONFIG to 10h. See the *MFR_CHANNEL_CONFIG (E4h)* for more details.

Also, each PSENn/GPOn pin can be independently configured to be active high or active low and either push-pull or open drain using the HI_LO and PP_OD bits, respectively.

If SELECT[2:0] = 011, the PSENn/GPOn output is configured to assert when some combination of power goods (PGs) and general-purpose inputs (GPIs) from each channel are asserted. The channels that should be used in this combination are selected using the PG_GPI_SELECT bits 31:16. If the PG_GPI_SELECT bit is cleared, then the associated channel is not used in the logical combination to assert the GPOn output. If the PG_GPI_SELECT bit is set, then the PG or GPI from this channel is used in the logical combination to assert or deassert the GPOn output. This function is useful in creating system power-good signals.

If SELECT[2:0] = 100, the PSENn/GPOn output is configured to assert when any of the enabled channel alarms goes active. The channel alarms are enabled with the ALARM_SELECT bits 31:16. If the ALARM_SELECT bit is cleared, then the alarm from this channel is blocked. If the ALARM_SELECT bit is set, the alarm from this channel is routed to an OR function such that any enabled alarm asserts the GPOn output. The alarm function is chosen with the ALARM_CONFIG bits in the MFR_FAULT_RESPONSE command. This function is useful is in system debug or for enabling system status LEDs.

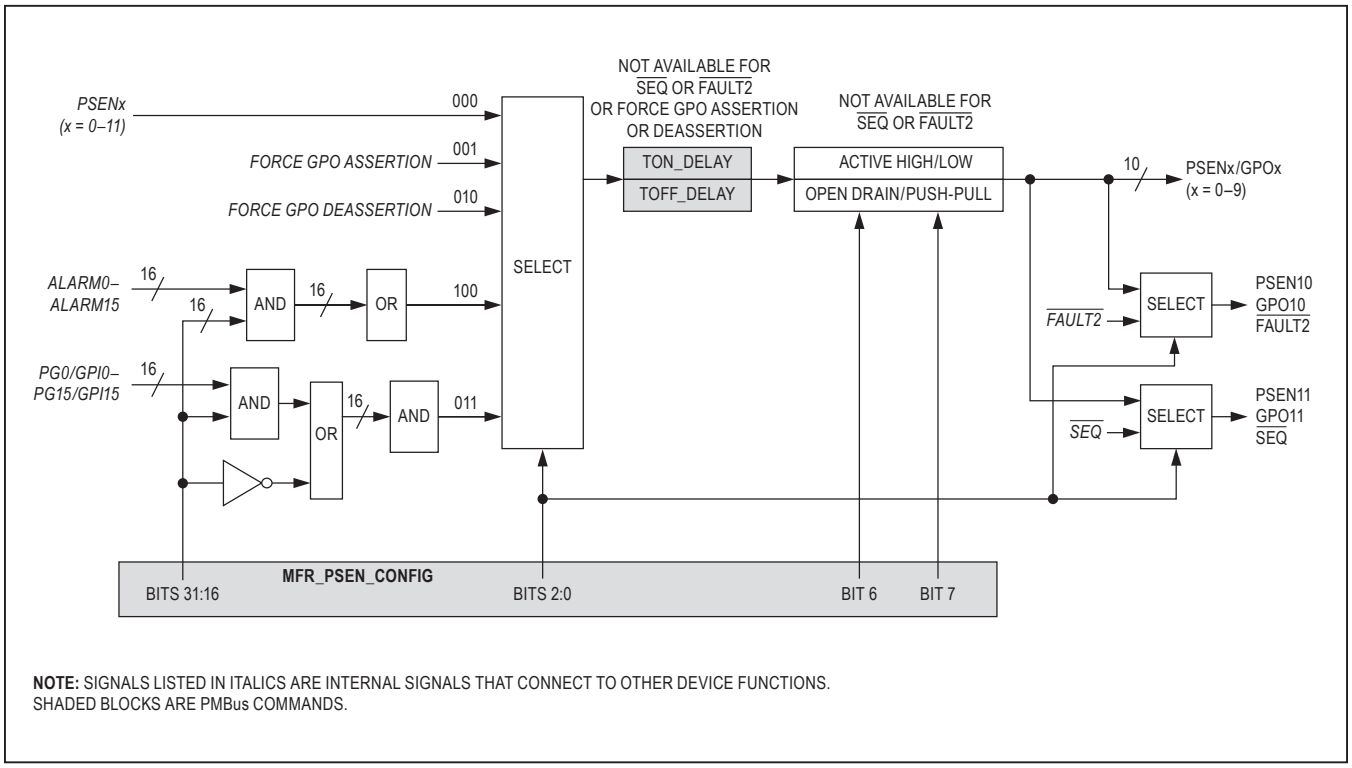
Table 27. MFR_PSEN_CONFIG (D2h)

BIT	NAME	MEANING
31:16	PG_GPI_SELECT ALARM_SELECT	These bits are only used if SELECT[2:0] = 011 or 100. Each bit corresponds to one channel (device channel N + 16 = bit number): SELECT[2:0] — BIT FUNCTION 011 — When this bit is cleared, the power good (PG) or GPI from channel N is not used in the logical AND to assert the GPOn output. When this bit is set, the PG or GPI is used. 100 — When this bit is cleared, the alarm from channel N is blocked from the logical OR to assert the GPOn output. When this bit is set, the alarm signal is routed to the logical OR.
15:8	0	These bits always return a 0.
7	PP_OD	0 = PSEN/GPO push-pull output 1 = PSEN/GPO open-drain output
6	HI_LO	0 = PSEN/GPO active low 1 = PSEN/GPO active high
5:3	0	These bits always return a 0.

Table 27. MFR_PSEN_CONFIG (D2h) (continued)

BIT	NAME	MEANING
2:0	SELECT[2:0]	These bits determine the function selected on the pin: SELECT[2:0] — PSENn/GPOn PIN FUNCTION SELECTED 000 — PSEN operation.* 001 — Force GPO assertion. 010 — Force GPO deassertion. 011 — PG/GPI operation (use bits 31:16). 100 — Alarm operation (use bits 31:16). 101 — $\overline{FAULT2}$ special function (only PAGE 10); $\overline{SEQ}$ special function (only PAGE 11). 11x — Reserved.

**For proper sequencing, the SELECT bits in MFR_CHANNEL_CONFIG must set to 10h.*



NOTE: SIGNALS LISTED IN ITALICS ARE INTERNAL SIGNALS THAT CONNECT TO OTHER DEVICE FUNCTIONS.
SHADED BLOCKS ARE PMBus COMMANDS.

Figure 8. MFR_PSEN_CONFIG Functional Logic

Delay Function

If a delay is configured either on or off, the input must be continuously static through the delay time before the output changes state. See the Figure 9.

MFR_VOUT_PEAK (D4h)

The MFR_VOUT_PEAK command returns the maximum actual measured output voltage. To reset this value to 0, write to this command with a data value of 0. Any values written to this command are used as a comparison for future peak updates. The 2 data bytes are in DIRECT format.

MFR_IOUT_PEAK (D5h)

The MFR_IOUT_PEAK command returns the maximum actual measured current. To reset this value to 0, write to this command with a data value of 0. Any values written to this command are used as a comparison for future peak updates. The 2 data bytes are in DIRECT format.

MFR_TEMPERATURE_PEAK (D6h)

The MFR_TEMPERATURE_PEAK command returns the maximum measured temperature. To reset this value to its lowest value, write to this command with a data value of 8000h. Any other values written by this command are used as a comparison for future peak updates. The 2 data bytes are in DIRECT format.

MFR_VOUT_MIN (D7h)

The MFR_VOUT_MIN command returns the minimum actual measured output voltage. To reset this value, write to this command with a data value of 7FFFh. Any values written to this command are used as a comparison for future minimum updates. The 2 data bytes are in DIRECT format.

MFR_FW_SERIAL (E0h)

The MFR_FW_SERIAL command stores the internal firmware version loaded onto the device. This is a 16-bit unsigned integer. The command is read-only.

MFR_IOUT_AVG (E2h)

The MFR_IOUT_AVG command returns the calculated average current. The number of samples collected in the average before reporting the value in MFR_IOUT_AVG is configured using the IOUT_AVG bits in MFR_MODE. Writes to this command are ignored. The 2 data bytes are in DIRECT format.

MFR_NV_LOG_CONFIG (D8h)

The MFR_NV_LOG_CONFIG command is used to configure the operation of the nonvolatile fault logging in the device. The MFR_NV_LOG_CONFIG command is described in Table 28.

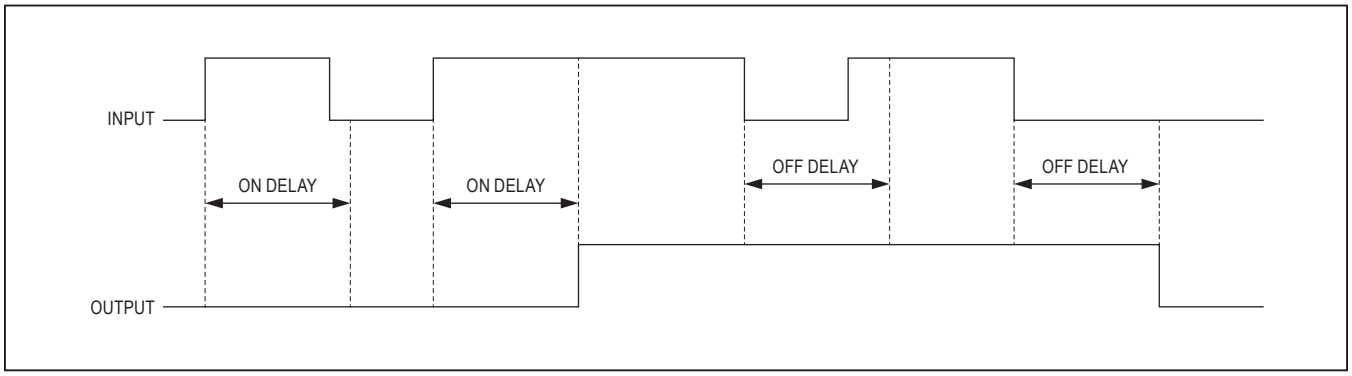


Figure 9. Input-to-Output Delay Action

Table 28. MFR_NV_LOG_CONFIG (D8h)

BIT	NAME	MEANING
15	FORCE_NV_FAULT_LOG	Setting this bit to a 1, forces the device to log data into the nonvolatile fault log. Once set, the device clears this bit when the action is completed. Host must set again for subsequent action. If an error occurs during this action, the device sets the CML bit in STATUS_WORD; no bits are set in STATUS_CML.
14	CLEAR_NV_FAULT_LOG	Setting this bit to a 1, forces the device to clear the nonvolatile fault log by writing FFh to all byte locations. Once set, the device clears this bit when the action is completed. Host must set again for subsequent action. If an error occurs during this action, the device sets the CML bit in STATUS_WORD; no bits are set in STATUS_CML. While clearing the fault log, monitoring is stopped and commands should not be sent to the PMBus port.
13:11	0	These bits always return a 0.
10	NV_LOG_T0_CONFIG	This bit determines the source of the data written into the T0 location of each page when a nonvolatile fault log is written. 0 = Log the last regular collection interval ADC reading 1 = Read the latest ADC value before logging
9	NV_LOG_OVERWRITE	0 = Do not overwrite the NV fault log 1 = Overwrite the NV fault log once it is full*
8:7	NV_LOG_DEPTH[1:0]	These bits determine the depth of the NV fault log: NV_LOG_DEPTH[1:0] / ADC RESULT COLLECTION INTERVAL / NV FAULT LOG DEPTH 00 / 5ms / 15ms 01 / 20ms / 60ms 10 / 80ms / 240ms 11 / 160ms / 480ms
N	NV_LOG_FAULT0	0 = Do not write NV fault log when $\overline{\text{FAULT0}}$ pin is externally pulled low. 1 = Write NV fault log when $\overline{\text{FAULT0}}$ pin is externally pulled low.
5	NV_LOG_FAULT1	0 = Do not write NV fault log when $\overline{\text{FAULT1}}$ pin is externally pulled low. 1 = Write NV fault log when $\overline{\text{FAULT1}}$ pin is externally pulled low and the $\overline{\text{FAULT1}}$ pin is enabled.
4	NV_LOG_FAULT2	0 = Do not write NV fault log when $\overline{\text{FAULT2}}$ pin is externally pulled low. 1 = Write NV fault log when $\overline{\text{FAULT2}}$ pin is externally pulled low and the $\overline{\text{FAULT2}}$ pin is enabled.
3:0	0	These bits always return a 0.

**The device clears two fault logs at a time when overwrite is enabled.*

MFR_FAULT_RESPONSE (D9h)

The MFR_FAULT_RESPONSE command specifies the response to each fault or warning condition supported by the device. In response to a fault/warning, the device always reports the fault/warning in the appropriate status register and asserts the ALERT output (if enabled in MFR_MODE). A CML fault cannot cause any device action other than setting the status bit and asserting the ALERT output. The MFR_FAULT_RESPONSE command is described in Table 30 and shown in Figure 10.

For each fault type (overvoltage or overcurrent, undervoltage, sequencing error, and overtemperature), each channel can be independently configured to respond in the required manner with the RESPONSE bits in MFR_FAULT_RESPONSE. If channels 0–11 are configured to latch off for a particular fault, the channel turns off (either immediately or after the TOFF_DELAY as configured or commanded) and also assert one or more of the FAULTn pins if they are enabled with bits 18:16 in MFR_FAULT_RESPONSE. The channel remains off and the FAULTn outputs remain asserted until either the master power control is toggled using the OPERATION command or CONTROLn pins as configured in the ON_OFF_CONFIG command or the device is reset or power cycled. When the device attempts to sequence the power supplies on, all enabled faults must be cleared before the channel is allowed to power-on or the FAULTn pins deasserted. If channels 12–15 are configured to latch off, they respond like channels 0–11; however, all the power supplies must be turned off before they are allowed to turn back on.

If the channel is configured to retry for a particular fault, the channel turns off (either immediately or after the TOFF_DELAY as configured or commanded) and also assert one or more the FAULTn pins if they are enabled with bits 18:16 in MFR_FAULT_RESPONSE. The channel remains off and the FAULTn outputs remain asserted for the time configured in MFR_FAULT_RETRY. After the time in MFR_FAULT_RETRY expires, the device attempts to sequence the power supplies back on as long as all the enabled faults in the channel are cleared. If all the enabled faults are cleared, then the device deasserts all the FAULTn pins it asserted and as long as no other channels have asserted the FAULTn pins it has been configured to monitor with bits 26:24 in MFR_FAULT_RESPONSE, the power-up sequencing begins.

Global channels must assert a FAULTn pin and respond to that FAULTn pin for the channel to shut down.

LOCAL vs. GLOBAL Channels

With the MFR_FAULT_RESPONSE command (bit 14), each power-supply channel can be tagged as either being LOCAL or GLOBAL. When bit 14 is cleared, the channel is configured as a LOCAL channel, which means that a detected fault only affects this channel (or page). With the RESPONSE bits in the MFR_FAULT_RESPONSE command, the device can be configured to respond differently to each possible fault. When bit 14 is set, the channel is configured as a GLOBAL channel which means that a detected fault on this channel can assert all enabled FAULTn outputs. The FAULTn outputs that are enabled are selected with bits 18:16. Only GLOBAL channels respond to FAULTn pins that are asserted. The FAULTn pins that the channel should respond to are assigned with bits 26:24. LOCAL channels do not respond to the fault pins.

GLOBAL Channels Respond to FAULTn Assertion

Bits 26:24 in the MFR_FAULT_RESPONSE command are used to configure GLOBAL channels to respond or ignore one or more of the FAULTn pins when they are asserted. When one or more of the enabled FAULTn pins is asserted, the channel either deasserts the PSENn output immediately or after the TOFF_DELAY according to the configuration of bit 0 in the ON_OFF_CONFIG command. The channel continues to deassert the PSENn output until all enabled FAULTn pins deassert. When all enabled FAULTn pins deassert, the channel sequences on as configured if no channel faults are present.

Temperature Fault Response

A temperature fault is declared when any of the enabled temperature sensors detect a fault. A temperature fault acts globally and can affect all of the power supplies. For all global supplies, the worst-case fault response of all global channels is applied. If this response is latchoff or retry, all FAULTn pins that are programmed to be asserted by any of the global channels will be asserted. All local channels respond independently, as programmed in that channel's MFR_FAULT_RESPONSE.

Fault Detection Before Power-On Sequencing

Before any power-supply channel is enabled or $\overline{\text{FAULTn}}$ output deasserted, the device checks for overvoltage, overcurrent, and overtemperature faults (but not for undervoltage) if the channel is configured for a fault response to either latch off (RESPONSE[1:0] = 01) or retry (RESPONSE[1:0] = 10) in the MFR_FAULT_REPSONSE command. Undervoltage faults are detected when the power supply turns on and fails to reach the power-good level, and the TON_MAX_FAULT_LIMIT is exceeded and the device takes fault action as configured. See Table 29.

Logging Faults into MFR_NV_FAULT_LOG

If bit 15 of MFR_FAULT_RESPONSE is set, faults are logged into the on-board nonvolatile fault log for this channel unless the response for the associated fault is configured to take no action (RESPONSE[1:0] = 00). To keep from needlessly filling the fault log with excessive data, the following rules are applied when subsequent faults occur. When overvoltage faults occurs, subsequent overvoltage faults on this channel are not written to the fault log until either the CLEAR_FAULTS command is issued or a device reset occurs. The same rule applies to overcurrent, undervoltage, overtemperature, and sequencing faults (see Table 30 and Figure 10).

Table 29. Fault Monitoring States

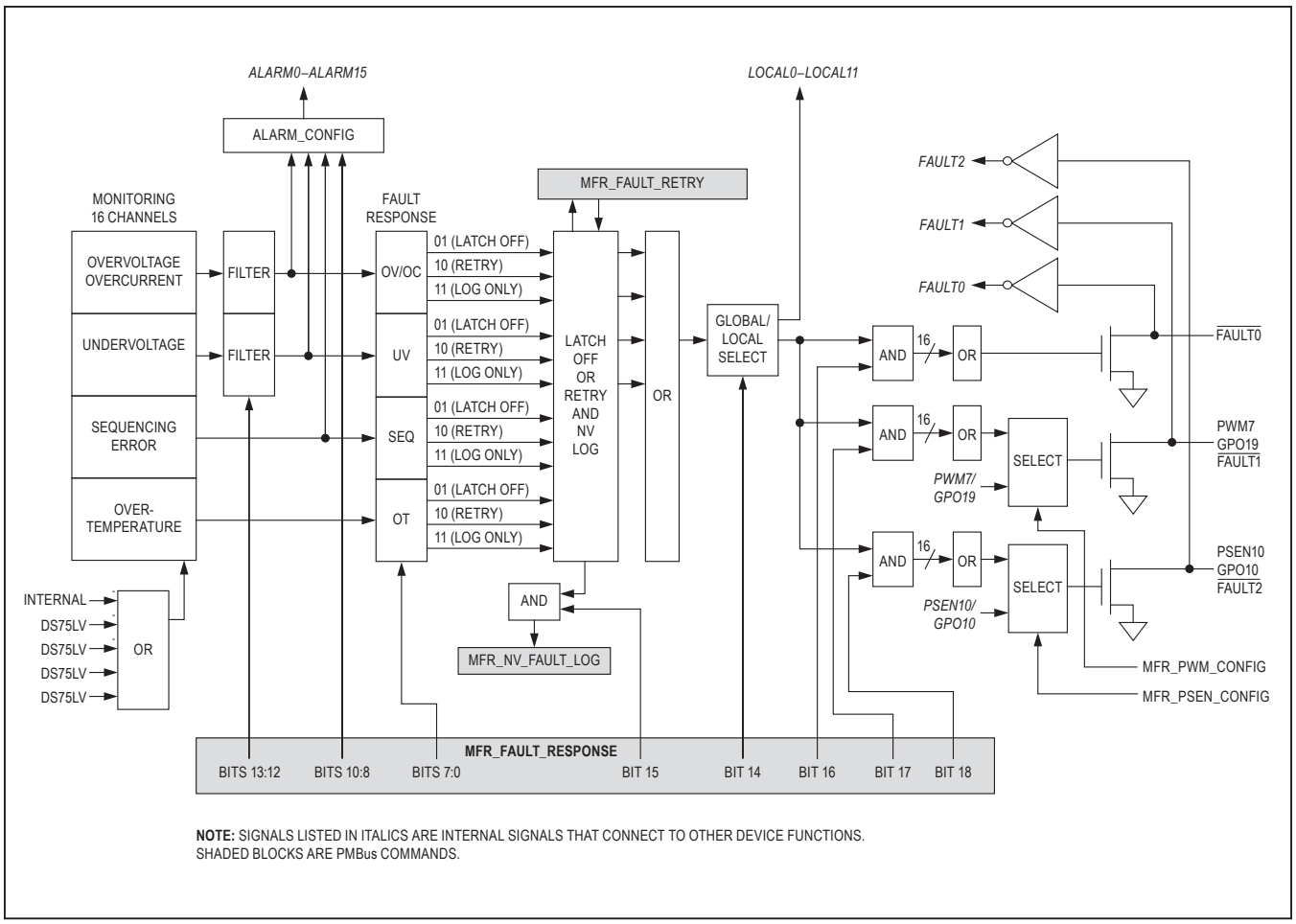
FAULT	REQUIRED DEVICE CONFIGURATION FOR ACTIVE MONITORING	WHEN MONITORED
Overvoltage	• Voltage Monitoring Enabled (SELECT[5:0] = 10h or 20h in MFR_CHANNEL_CONFIG)	Continuous monitoring
Undervoltage	• Voltage Monitoring Enabled (SELECT[5:0] = 10h or 20h in MFR_CHANNEL_CONFIG)	Stop monitoring while the power supply is off; start monitoring when voltage exceeds the POWER_GOOD_ON level
Overcurrent	• Current Monitoring Enabled (SELECT[5:0] = 22h in MFR_CHANNEL_CONFIG)	Continuous monitoring
Power-Up Time	• Sequencing Enabled (SELECT[5:0] = 10h in MFR_CHANNEL_CONFIG)	Monitored only during power on sequence
Overtemperature	• Temperature Sensor Enabled (ENABLE = 1 in MFR_TEMP_SENSOR_CONFIG)	Continuous monitoring

Note: *Device response to faults is determined by the configuration of MFR_FAULT_RESPONSE.*

Table 30. MFR_FAULT_RESPONSE (D9h)

BIT	NAME	MEANING
31:27	0	These bits always return a 0.
26	FAULT2_RESPONSE_ENABLE	0 = $\overline{\text{FAULT2}}$ response disabled 1 = $\overline{\text{FAULT2}}$ response enabled
25	FAULT1_RESPONSE_ENABLE	0 = $\overline{\text{FAULT1}}$ response disabled 1 = $\overline{\text{FAULT1}}$ response enabled
24	FAULT0_RESPONSE_ENABLE	0 = $\overline{\text{FAULT0}}$ response disabled 1 = $\overline{\text{FAULT0}}$ response enabled
23:19	0	These bits always return a 0.
18	FAULT2_ASSERT_ENABLE	0 = $\overline{\text{FAULT2}}$ assertion disabled 1 = $\overline{\text{FAULT2}}$ assertion enabled
17	FAULT1_ASSERT_ENABLE	0 = $\overline{\text{FAULT1}}$ assertion disabled 1 = $\overline{\text{FAULT1}}$ assertion enabled
16	FAULT0_ASSERT_ENABLE	0 = $\overline{\text{FAULT0}}$ assertion disabled 1 = $\overline{\text{FAULT0}}$ assertion enabled
15	NV_LOG	0 = Do not log the fault into MFR_NV_FAULT_LOG 1 = Log the fault into MFR_NV_FAULT_LOG
14	GLOBAL	0 = LOCAL (affect only the selected page) 1 = GLOBAL (Note 1).
13:12	FILTER[1:0]	Continuous excursion time before a fault or warning is declared and action is taken (Note 2). 00 = Immediate 01 = 2ms 10 = 3ms 11 = 4ms
11	0	This bit always returns a 0.
10:8	ALARM_CONFIG[2:0]	See Table 31.
7:6	OT_FAULT_LIMIT_RESPONSE[1:0]	See Tables 32 and 33 (Note 3).
5:4	TON_MAX_FAULT_LIMIT_RESPONSE[1:0] (also applies to MFR_TON_SEQ_MAX)	See Tables 32 and 33 (Notes 4 and 5).
3:2	VOUT_UV_FAULT_LIMIT_RESPONSE[1:0]	See Tables 32 and 33 (Note 4).
1:0	VOUT_OV_FAULT_LIMIT_RESPONSE[1:0] IOUT_OC_FAULT_LIMIT_RESPONSE[1:0]	See Tables 32 and 33 (Note 6).

Note 1: Channels configured to monitor current must be configured as GLOBAL. Also PAGES 12–15 must be configured as GLOBAL.
Note 2: The FILTER selection does not apply to temperature or sequencing faults.
Note 3: All enabled temperature sensor faults are logically ORed together.
Note 4: If the channel is configured to measure current, these bits are ignored.
Note 5: These bits are ignored for PAGES 12–15.
Note 6: Depends on whether the channel is configured to monitor voltage or current.



NOTE: SIGNALS LISTED IN ITALICS ARE INTERNAL SIGNALS THAT CONNECT TO OTHER DEVICE FUNCTIONS.
SHADED BLOCKS ARE PMBus COMMANDS.

Figure 10. MFR_FAULT_RESPONSE Operation

Table 31. ALARM_CONFIG Codes

ALARM_CONFIG[2:0]	ALARM CONDITION	ALARM CRITERIA
000	None	—
001	Sequencing fault	Fault only
010	Undervoltage only	Fault only
011	Undervoltage only	Fault or warning
100	Overvoltage/overcurrent only	Fault only
101	Overvoltage/overcurrent only	Fault or warning
110	Undervoltage or overvoltage/overcurrent	Fault only
111	Undervoltage or overvoltage/overcurrent	Fault or warning

Table 32. MFR_FAULT_RESPONSE Codes for GLOBAL Channels

RESPONSE[1:0]	FAULT RESPONSE
11	• Sets the corresponding fault bit in the appropriate status register (Note 1). • Logs fault into MFR_NV_FAULT_LOG if NV_LOG = 1. • Continues operation.
10 (Retry)	• Asserts all enabled $\overline{\text{FAULTn}}$ outputs. • Sets the corresponding fault bit in the appropriate status register (Note 1). • Logs fault into MFR_NV_FAULT_LOG if NV_LOG = 1. • Waits for the time configured in MFR_FAULT_RETRY and then deassert the $\overline{\text{FAULTn}}$ outputs that were asserted if fault-free (Note 2).
01 (Latch off)	• Asserts all enabled $\overline{\text{FAULTn}}$ outputs. • Sets the corresponding fault bit in the appropriate status register (Note 1). • Logs fault into MFR_NV_FAULT_LOG if NV_LOG = 1.
00	• Sets the corresponding fault bit in the appropriate status register (Note 1). • Continues operation without any action.

Note 1: $\overline{\text{ALERT}}$ is asserted if enabled when a new status bit is set. A status bit is latched when a particular fault occurs that causes a fault response.

Note 2: Fault-free does not include undervoltage.

Table 33. MFR_FAULT_RESPONSE Codes for LOCAL Channels

RESPONSE[1:0]	FAULT RESPONSE
11	• Sets the corresponding fault bit in the appropriate status register (Note 1). • Logs fault into MFR_NV_FAULT_LOG if NV_LOG = 1. • Continues operation.
10 (Retry)	• Shuts down the power supply by deasserting the PSENn output. • Sets the corresponding fault bit in the appropriate status register (Note 1). • Logs fault into MFR_NV_FAULT_LOG if NV_LOG = 1. • Waits for the time configured in MFR_FAULT_RETRY and restarts the supply if fault-free (Note 2).
01 (Latch off)	• Latches off the power supply by deasserting the PSENn output. • Sets the corresponding fault bit in the appropriate status register (Note 1). • Logs fault into MFR_NV_FAULT_LOG if NV_LOG = 1.
00	• Sets the corresponding fault bit in the appropriate status register (Note 1). • Continues operation without any action.

Note 1: $\overline{\text{ALERT}}$ is asserted if enabled when a new status bit is set. A status bit is latched when a particular fault occurs that causes a fault response.

Note 2: Fault-free does not include undervoltage.

Alarm Output Functionality

Any of the GPOn pins can be configured to output the alarm signals. See the MFR_PWM_CONFIG and MFR_PSEN_CONFIG commands for details. When an undervoltage or overvoltage/overcurrent alarm is occurr ing, the output remains asserted as long as the alarm continues. When a sequencing fault occurs, the alarm pin remains asserted until either a CLEAR_FAULTS command is received, or a master power control off input is received with either the OPERATION command or the CONTROLn pins.

MFR_FAULT_RETRY (DAh)

The MFR_FAULT_RETRY command sets the delay time between channel shutdown due to fault event and its restarting if the fault response is configured to retry. This command value is used for all fault responses.

The retry timer starts when the fault occurs. If the faulty channel has been configured to assert one or more FAULT pins, the FAULT pins are asserted. For an undervoltage fault happening on a channel configured for sequencing

(MFR_CHANNEL_CONFIG [5:0]= 10h), the FAULT pin(s) deasserts immediately when the retry timer expires. If the channel is configured for voltage monitoring (MFR_CHANNEL_CONFIG [5:0]= 20h) the FAULT pin deasserts when the retry timer expires if no fault condition is present. For all other fault events, the FAULT pin(s) deasserts when retry timer expires if no fault is present.

MFR_FAULT_RETRY should be configured with a value larger than the largest system TOFF_DELAY. The 2 data bytes are in DIRECT format.

MFR_NV_FAULT_LOG (DCh)

Data from each of the 15 nonvolatile fault logs is constituted (in nonvolatile memory) as a block of 255 bytes. Executing the MFR_NV_FAULT_LOG command using the Read 32 SMBus protocol, each block of 255 bytes can be read back from the device in packets of 4 bytes. Alternatively, the MFR_NV_FAULT_LOG can also be executed to read back a block of 255 bytes by using the Block Read SMBus protocol except that the device does not report the Byte Count during read back. The MFR_NV_FAULT_LOG command must be executed 15 times to dump the complete nonvolatile fault log. If the returned fault log is all FFs (except bytes 0 and 1), this indicates that this fault log has not been written by the device. As the device is operating, it is reading the latest operating conditions for voltage, current, and temperature and updating the status registers. All this information is stored in on-board RAM. When a fault is detected (if so enabled in MFR_FAULT_RESPONSE), the device automatically logs this information to one of the 15 nonvolatile fault logs. After 15 faults have been written, bit 0 of STATUS_CML is set and the device can be configured (with the NV_LOG_OVERWRITE bit in MFR_NV_LOG_CONFIG) to either stop writing additional fault logs or write over the oldest data. The host can clear the fault log by setting the CLEAR_NV_FAULT_LOG bit in MFR_NV_LOG_CONFIG. If a power supply is not enabled to measure voltage, current, or if a temperature sensor is disabled, the associated fault log position returns 0000h (see Figure 11).

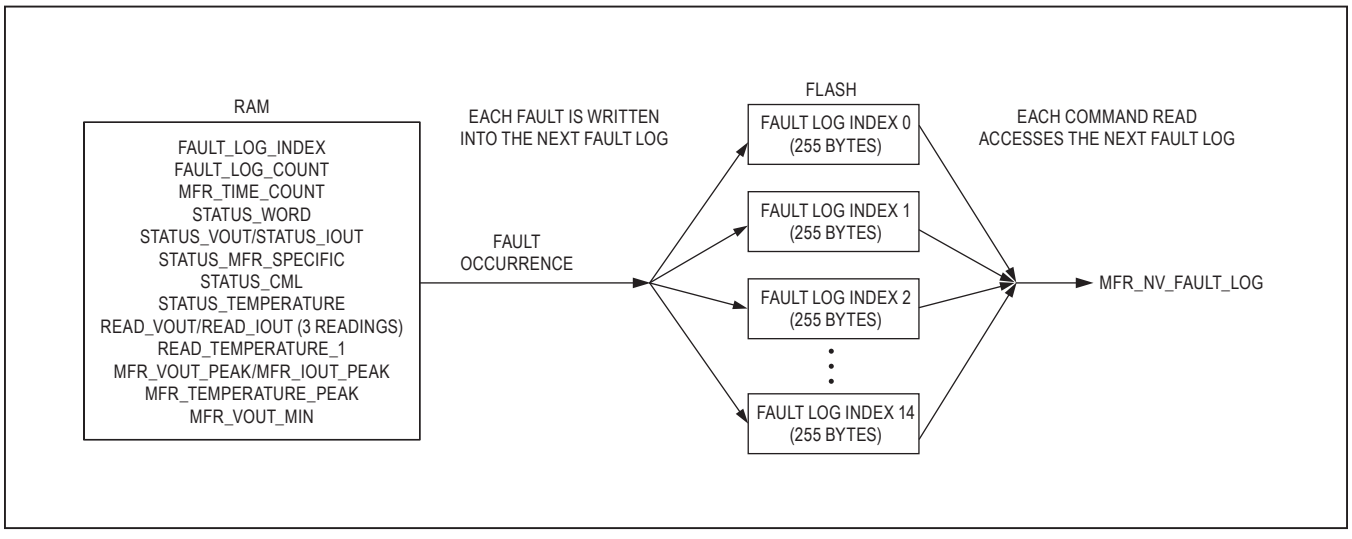


Figure 11. MFR_NV_FAULT_LOG

There is a FAULT_LOG_COUNT (16-bit counter) at the beginning of each fault log that indicates which fault log is the latest. This counter rolls over should more than 65,535 faults be logged. This counter is not cleared when the CLEAR_NV_FAULT_LOG bit in MFR_NV_LOG_CONFIG is toggled. The 255 bytes returned by the MFR_NV_FAULT_LOG command are described in Table 34.

If an error occurs while the device is attempting to write to or clear the MFR_NV_FAULT_LOG, the device sets the CML bit in STATUS_WORD (no bits are set in STATUS_CML) and $\overline{\text{ALERT}}$ is asserted (if enabled in MFR_MODE).

USER NOTE: V_{DD} must be above 2.9V for the device to clear or log data into MFR_NV_FAULT_LOG.

Table 34. MFR_NV_FAULT_LOG (DCh)

BYTE	PARAMETER	BYTE	PARAMETER
0	00h/FAULT_LOG_INDEX	128	READ_VOUT/READ_IOUT T1 PAGE 11
2	FAULT_LOG_COUNT	130	READ_VOUT/READ_IOUT T2 PAGE 11
4	MFR_TIME_COUNT (LSW)	132	READ_VOUT/READ_IOUT T0 PAGE 12
6	MFR_TIME_COUNT (MSW)	134	READ_VOUT/READ_IOUT T1 PAGE 12
8	0000h	136	READ_VOUT/READ_IOUT T2 PAGE 12
10	STATUS_CML/00h	138	READ_VOUT/READ_IOUT T0 PAGE 13
12	STATUS_WORD	140	READ_VOUT/READ_IOUT T1 PAGE 13
14	STATUS_VOUT/STATUS_IOUT PAGES 0/1	142	READ_VOUT/READ_IOUT T2 PAGE 13
16	STATUS_VOUT/STATUS_IOUT PAGES 2/3	144	READ_VOUT/READ_IOUT T0 PAGE 14
18	STATUS_VOUT/STATUS_IOUT PAGES 4/5	146	READ_VOUT/READ_IOUT T1 PAGE 14
20	STATUS_VOUT/STATUS_IOUT PAGES 6/7	148	READ_VOUT/READ_IOUT T2 PAGE 14
22	STATUS_VOUT/STATUS_IOUT PAGES 8/9	150	READ_VOUT/READ_IOUT T0 PAGE 15
24	STATUS_VOUT/STATUS_IOUT PAGES 10/11	152	READ_VOUT/READ_IOUT T1 PAGE 15
26	STATUS_VOUT/STATUS_IOUT PAGES 12/13	154	READ_VOUT/READ_IOUT T2 PAGE 15
28	STATUS_VOUT/STATUS_IOUT PAGES 14/15	156	0000h
30	STATUS_MFR_SPECIFIC PAGES 0/1	158	0000h
32	STATUS_MFR_SPECIFIC PAGES 2/3	160	0000h
34	STATUS_MFR_SPECIFIC PAGES 4/5	162	0000h
36	STATUS_MFR_SPECIFIC PAGES 6/7	164	MFR_VOUT_PEAK/MFR_IOUT_PEAK PAGE 0
38	STATUS_MFR_SPECIFIC PAGES 8/9	166	MFR_VOUT_PEAK/MFR_IOUT_PEAK PAGE 1
40	STATUS_MFR_SPECIFIC PAGES 10/11	168	MFR_VOUT_PEAK/MFR_IOUT_PEAK PAGE 2
42	STATUS_MFR_SPECIFIC PAGES 12/13	170	MFR_VOUT_PEAK/MFR_IOUT_PEAK PAGE 3
44	STATUS_MFR_SPECIFIC PAGES 14/15	172	MFR_VOUT_PEAK/MFR_IOUT_PEAK PAGE 4
46	STATUS_MFR_SPECIFIC PAGE 255/00h	174	MFR_VOUT_PEAK/MFR_IOUT_PEAK PAGE 5
48	STATUS_TEMPERATURE PAGES 16/17	176	MFR_VOUT_PEAK/MFR_IOUT_PEAK PAGE 6
50	STATUS_TEMPERATURE PAGES 18/19	178	MFR_VOUT_PEAK/MFR_IOUT_PEAK PAGE 7
52	STATUS_TEMPERATURE PAGE 20/00h	180	MFR_VOUT_PEAK/MFR_IOUT_PEAK PAGE 8
54	CURRENT_CHANNELS (Note 4)	182	MFR_VOUT_PEAK/MFR_IOUT_PEAK PAGE 9
56	0000h	184	MFR_VOUT_PEAK/MFR_IOUT_PEAK PAGE 10
58	0000h	186	MFR_VOUT_PEAK/MFR_IOUT_PEAK PAGE 11

Table 34. MFR_NV_FAULT_LOG (DCh) (continued)

BYTE	PARAMETER	BYTE	PARAMETER
60	READ_VOUT/READ_IOUT T0 PAGE 0 (Notes 2, 3)	188	MFR_VOUT_PEAK/MFR_IOUT_PEAK PAGE 12
62	READ_VOUT/READ_IOUT T1 PAGE 0 (Notes 2, 3)	190	MFR_VOUT_PEAK/MFR_IOUT_PEAK PAGE 13
64	READ_VOUT/READ_IOUT T2 PAGE 0 (Notes 2, 3)	192	MFR_VOUT_PEAK/MFR_IOUT_PEAK PAGE 14
66	READ_VOUT/READ_IOUT T0 PAGE 1	194	MFR_VOUT_PEAK/MFR_IOUT_PEAK PAGE 15
68	READ_VOUT/READ_IOUT T1 PAGE 1	196	MFR_VOUT_MIN PAGE 0
70	READ_VOUT/READ_IOUT T2 PAGE 1	198	MFR_VOUT_MIN PAGE 1
72	READ_VOUT/READ_IOUT T0 PAGE 2	200	MFR_VOUT_MIN PAGE 2
74	READ_VOUT/READ_IOUT T1 PAGE 2	202	MFR_VOUT_MIN PAGE 3
76	READ_VOUT/READ_IOUT T2 PAGE 2	204	MFR_VOUT_MIN PAGE 4
78	READ_VOUT/READ_IOUT T0 PAGE 3	206	MFR_VOUT_MIN PAGE 5
80	READ_VOUT/READ_IOUT T1 PAGE 3	208	MFR_VOUT_MIN PAGE 6
82	READ_VOUT/READ_IOUT T2 PAGE 3	210	MFR_VOUT_MIN PAGE 7
84	READ_VOUT/READ_IOUT T0 PAGE 4	212	MFR_VOUT_MIN PAGE 8
86	READ_VOUT/READ_IOUT T1 PAGE 4	214	MFR_VOUT_MIN PAGE 9
88	READ_VOUT/READ_IOUT T2 PAGE 4	216	MFR_VOUT_MIN PAGE 10
90	READ_VOUT/READ_IOUT T0 PAGE 5	218	MFR_VOUT_MIN PAGE 11
92	READ_VOUT/READ_IOUT T1 PAGE 5	220	MFR_VOUT_MIN PAGE 12
94	READ_VOUT/READ_IOUT T2 PAGE 5	222	MFR_VOUT_MIN PAGE 13
96	READ_VOUT/READ_IOUT T0 PAGE 6	224	MFR_VOUT_MIN PAGE 14
98	READ_VOUT/READ_IOUT T1 PAGE 6	226	MFR_VOUT_MIN PAGE 15
100	READ_VOUT/READ_IOUT T2 PAGE 6	228	0000h
102	READ_VOUT/READ_IOUT T0 PAGE 7	230	0000h
104	READ_VOUT/READ_IOUT T1 PAGE 7	232	READ_TEMPERATURE_1 PAGE 16
106	READ_VOUT/READ_IOUT T2 PAGE 7	234	READ_TEMPERATURE_1 PAGE 17
108	READ_VOUT/READ_IOUT T0 PAGE 8	236	READ_TEMPERATURE_1 PAGE 18
110	READ_VOUT/READ_IOUT T1 PAGE 8	238	READ_TEMPERATURE_1 PAGE 19
112	READ_VOUT/READ_IOUT T2 PAGE 8	240	READ_TEMPERATURE_1 PAGE 20
114	READ_VOUT/READ_IOUT T0 PAGE 9	242	MFR_TEMPERATURE_PEAK PAGE 16
116	READ_VOUT/READ_IOUT T1 PAGE 9	244	MFR_TEMPERATURE_PEAK PAGE 17
118	READ_VOUT/READ_IOUT T2 PAGE 9	246	MFR_TEMPERATURE_PEAK PAGE 18
120	READ_VOUT/READ_IOUT T0 PAGE 10	248	MFR_TEMPERATURE_PEAK PAGE 19
122	READ_VOUT/READ_IOUT T1 PAGE 10	250	MFR_TEMPERATURE_PEAK PAGE 20
124	READ_VOUT/READ_IOUT T2 PAGE 10	252	0000h
126	READ_VOUT/READ_IOUT T0 PAGE 11	254	LOG_VALID (Note 1)

Note 1: LOG_VALID is set to DDh if the fault log contains valid data.
Note 2: For READ_VOUT, READ_IOUT, T2 is the oldest reading and T0 is the newest reading.
Note 3: STATUS_VOUT/STATUS_IOUT and READ_VOUT/STATUS_IOUT depend on whether the channel is configured to monitor voltage or current.
Note 4: CURRENT_CHANNELS is a bitmask (0 = voltage/1 = current) indicating which channels are enabled for current measurement.

MFR_TIME_COUNT (DDh)

The MFR_TIME_COUNT command returns the current value of a real-time counter that increments every 5ms, 20ms, 80ms, or 160ms depending on the configuration of the NV_LOG_DEPTH bits in MFR_NV_LOG_CONFIG. This counter is useful in determining the time between multiple faults. The counter is a 32-bit value that rolls over. The count is reset to zero upon device power cycle or $\overline{RST}$ action, or a soft-reset. MFR_TIME_COUNT can be preset to any value and starts counting up from the preset value.

MFR_CHANNEL_CONFIG (E4h)

The MFR_CHANNEL_CONFIG command is used to configure the monitoring channels (PAGES 0–15). This command should not be changed while the power supplies are operating. The MFR_CHANNEL_CONFIG command is described in Table 35 and shown in Figure 12.

Each RSn/GPIn pin can be independently configured using the SELECT[5:0] bits to one of the following:

- Monitor voltage; use the monitored voltage for sequencing (SELECT[5:0] = 10h)
- Monitor voltage; do not use for sequencing (SELECT[5:0] = 20h)
- Monitor current (SELECT[5:0] = 22h)
- Read voltage only; do not monitor for voltage faults or warnings (SELECT[5:0] = 21h)
- Read current only; do not monitor for current faults or warnings (SELECT[5:0] = 23h)
- General-purpose input (GPI); active low (SELECT[5:0] = 30h)
- General-purpose input (GPI); active high (SELECT[5:0] = 34h)
- Input is disabled (SELECT[5:0] = 00h)

If the monitoring channel is configured to monitor voltage for sequencing (SELECT[5:0] = 10h), then the associated PSENn output channel must also be configured for controlling power supplies by setting the SELECT bits in MFR_PSEN_CONFIG to 000. See the MFR_PSEN_CONFIG command description for more details.

When the RSn/GPIn pins are configured as general-purpose inputs (GPIs) the READ_VOUT command reports 0000h when the pin is inactive and 0001h when the pin is active.

Also, when the RSn/GPIn pins are configured to monitor voltage (SELECT[5:0] = 10h or 20h) or act as GPI (SELECT[5:0] = 30h or 34h), each channel can be independently configured to generate a signature signal at the $\overline{SEQ}$ output. This would facilitate event-based sequencing (in multiple device systems), by indicating that this power supply has reached its POWER_GOOD_ON level and other channels can now proceed with their power-up.

Table 35. MFR_CHANNEL_CONFIG (E4h)

BIT	NAME	MEANING
15:12	0	These bits always return a 0.
11:8	SEQ_GENERATE	These bits determine which $\overline{SEQ}$ signature the channel should generate after crossing the POWER_GOOD_ON level: 0000 Disabled 0001 Signature 1 0010 Signature 2 0011 Signature 3 0100 Signature 4 0101 Signature 5 0110 Signature 6 0111 Signature 7 1000 Signature 8 1001 Signature 9 1010 Signature 10 1011 Signature 11 1100 Signature 12 1101 Signature 13 1110 Signature 14 1111 Signature 15
7:6	0	These bits always return a 0.
5:0	SELECT[5:0]	These bits select the function of the RSn/GPIn pins: SELECT[5:0] — SELECTED CHANNEL FUNCTION 010000 (10h) Sequencing + voltage monitoring (only valid for PAGES 0–11)* 100000 (20h) Voltage monitoring (no sequencing) 100010 (22h) Current monitoring 100001 (21h) Voltage read only 100011 (23h) Current read only 110000 (30h) General-purpose input active low 110100 (34h) General-purpose input active high 000000 (00h) Disabled

**For proper sequencing, the SELECT bits in MFR_PSEN_CONFIG must be set to 000.*

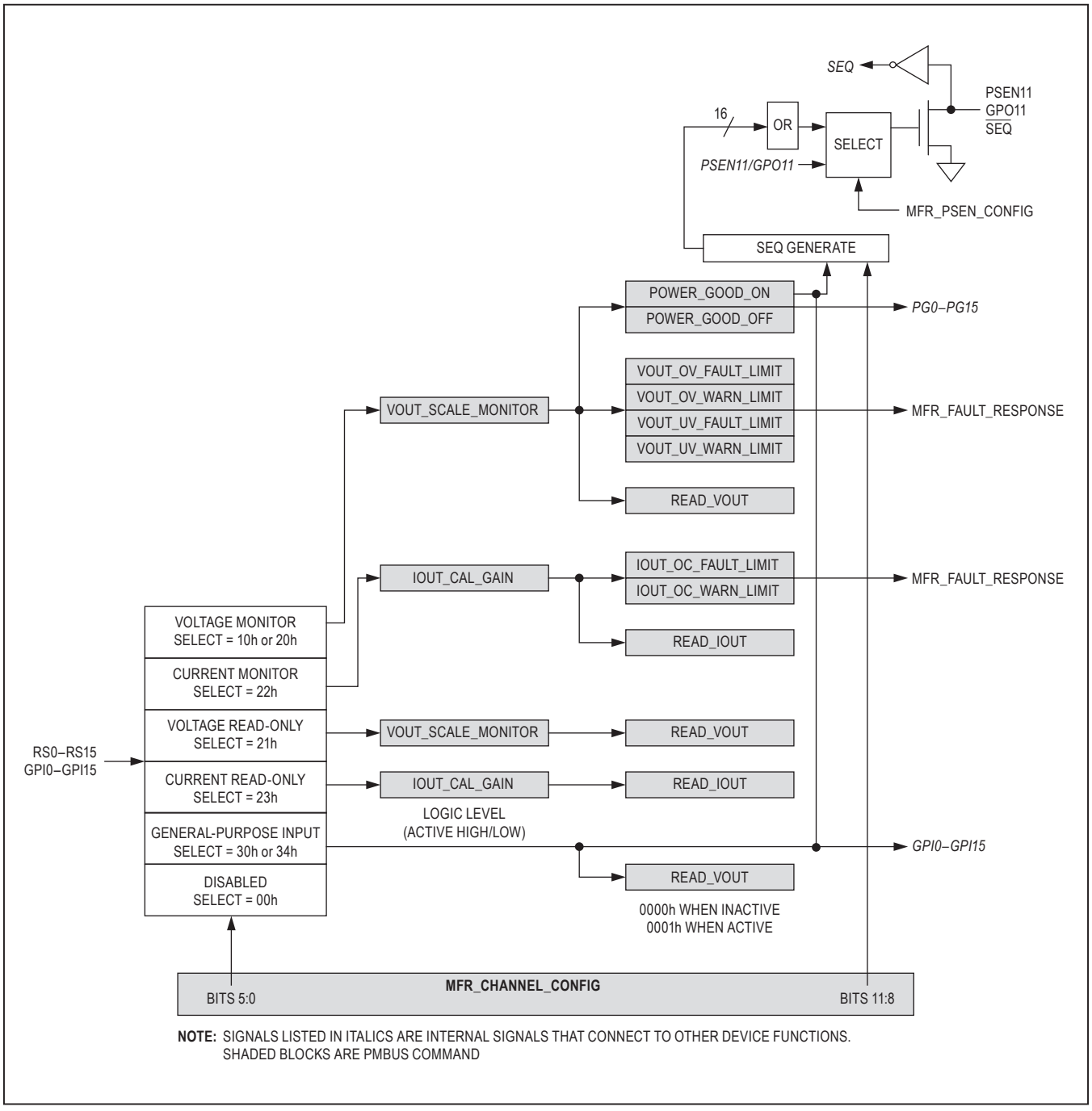


Figure 12. MFR_CHANNEL_CONFIG Command

Table 36. Fault-Monitoring States

FAULT	REQUIRED DEVICE CONFIGURATION FOR ACTIVE MONITORING	WHEN MONITORED
Overvoltage	• Voltage monitoring enabled (SELECT[5:0] = 10h or 20h in MFR_CHANNEL_CONFIG)	Continuous monitoring.
Undervoltage/ Power Good	• Voltage monitoring enabled (SELECT[5:0] = 10h or 20h in MFR_CHANNEL_CONFIG)	• If SELECT[5:0] = 10h (monitor and sequence mode), stops monitoring when PSEN is disabled. Power Good starts monitoring when PSEN is enabled and Undervoltage monitoring starts when voltage exceeds the POWER_GOOD_ON level. • If SELECT[5:0] = 20h (monitor only mode), starts monitoring when the voltage exceeds the POWER_GOOD_ON level.
Overcurrent	• Current monitoring enabled (SELECT[5:0] = 22h in MFR_CHANNEL_CONFIG)	Continuous monitoring.
Power-Up Time	• Sequencing enabled (SELECT[5:0] = 10h in MFR_CHANNEL_CONFIG)	Monitored only during power-on sequence.
Overtemperature	• Temperature sensor enabled (ENABLE = 1 in MFR_TEMP_SENSOR_CONFIG)	Continuous monitoring.

Note: *Device response to faults is determined by the configuration of MFR_FAULT_RESPONSE.*

MFR_TON_SEQ_MAX (E6h)

The MFR_TON_SEQ_MAX command sets an upper limit, in milliseconds, from a sequencing group (either SEQUENCE0 or SEQUENCE1, as chosen by the SEQ_SELECT bit in the MFR_SEQ_CONFIG command), initiating the power-up sequence until the channel expects to begin its power-up based on an event, which could be either a logic combination of power-good (PG) and GPI signals or a match on the $\overline{SEQ}$ pin, as configured with the SELECT bits in MFR_SEQ_CONFIG. The 2 data bytes are in DIRECT format. If this value is zero, then the limit is disabled. In response to the MFR_TON_SEQ_MAX being exceeded, the device does the following:

1) Sets the VOUT bit in STATUS_WORD.
2) Sets the TON_MAX_FAULT bit in STATUS_VOUT.
3) Responds as specified in the MFR_FAULT_RESPONSE.
4) Notifies the host using $\overline{ALERT}$ assertion (if enabled in MFR_MODE).

MFR_PWM_CONFIG (E7h)

The MFR_PWM_CONFIG command is used to configure the individual PWMx/GPOy (x = 0–7/y = 12–19) outputs. This command should not be changed while the power supplies are being PWM margined. The MFR_PWM_CONFIG command is described in Table 37 and shown in Figure 13.

Each PWMn/GPOn pin can be independently configured using the SELECT[2:0] bits to perform one of the following:

- PWM margining operation (SELECT[2:0] = 000)
- Force pin assertion (SELECT[2:0] = 001)
- Force pin deassertion (SELECT[2:0] = 010)
- Assert when all enabled channel power-good (PG) or GPI are asserted (SELECT[2:0] = 011)
- Assert when any enabled alarm goes active (SELECT[2:0] = 100)

Also, each PWMn/GPOn pin can be independently configured to be active high or active low and either push-pull or open drain using the HI_LO and PP_OD bits, respectively.

If SELECT[2:0] = 011, the PWMn/GPOn output is configured to assert when some combination of power-goods (PGs) and GPIs from each channel are asserted. The channels that should be used in this combination are selected using the PG_GPI_SELECT bits 31:16. If the PG_GPI_SELECT bit is cleared, then the associated channel is not used in the logical combination to assert the GPO output. If the PG_GPI_SELECT bit is set, then the PG or GPI from that channel is used in the logical combination to assert or deassert the GPOn output. This function is useful in creating system power-good signals.

If SELECT[2:0] = 100, the PWMn/GPOn output is configured to assert when any of the enabled channel alarms go active. The channel alarms are enabled with the ALARM_SELECT bits 31:16. If the ALARM_SELECT bit is cleared, then the alarm from that channel is blocked. If the ALARM_SELECT bit is set, then the alarm from that channel is routed to an OR function, such that any enabled alarm asserts the GPOn output. The alarm function is chosen with the ALARM_CONFIG bits in the MFR_FAULT_RESPONSE command. This function is useful for system debug or for enabling system status LEDs.

Table 37. MFR_PWM_CONFIG (E7h)

<table>
<tr><th>BIT</th><th>NAME</th><th>MEANING</th></tr>
<tr><td>31:16</td><td>PG_GPI_SELECT
ALARM_SELECT</td><td>These bits are only used if SELECT[2:0] = 011 or 100; each bit corresponds to one channel (device channel N + 16 = bit number):

SELECT[2:0] — BIT FUNCTION
011 — When this bit is cleared, the power good (PG) or GPI from channel N is not used in the logical AND to assert the GPOn output. When this bit is set, the PG or GPI is used.
100 — When this bit is cleared, the alarm from channel N is blocked from the logical OR to assert the GPO output. When this bit is set, the alarm signal is routed to the logical OR.</td></tr>
<tr><td>15:12</td><td>OFF_DELAY</td><td>These bits determine the delay time to pin deassertion; when the pin is operating as a PGn/GPIn or alarm pin (SELECT[2:0] = 011 or 100):

OFF_DELAY[3:0] — DELAY TIME — OFF_DELAY[3:0] — DELAY TIME
0000 — 0ms — 1000 — 200ms
0001 — 5ms — 1001 — 400ms
0010 — 10ms — 1010 — 600ms
0011 — 20ms — 1011 — 800ms
0100 — 40ms — 1100 — 1000ms
0101 — 60ms — 1101 — 1500ms
0110 — 80ms — 1110 — 2000ms
0111 — 100ms — 1111 — 4000ms</td></tr>
<tr><td>11:8</td><td>ON_DELAY</td><td>These bits determine the delay time to pin assertion; when the pin is operating as a PGn/GPIn or alarm pin (SELECT[2:0] = 011 or 100):

ON_DELAY[3:0] — DELAY TIME — ON_DELAY[3:0] — DELAY TIME
0000 — 0ms — 1000 — 200ms
0001 — 5ms — 1001 — 400ms
0010 — 10ms — 1010 — 600ms
0011 — 20ms — 1011 — 800ms
0100 — 40ms — 1100 — 1000ms
0101 — 60ms — 1101 — 1500ms
0110 — 80ms — 1110 — 2000ms
0111 — 100ms — 1111 — 4000ms</td></tr>
<tr><td>7</td><td>PP_OD</td><td>0 = PWMn/GPOn push-pull output
1 = PWMn/GPOn open-drain output</td></tr>
<tr><td>6</td><td>HI_LO</td><td>0 = PWMn/GPOn active low
1 = PWMn/GPOn active high</td></tr>
<tr><td>5:3</td><td>0</td><td>These bits always return a 0.</td></tr>
<tr><td>2:0</td><td>SELECT[2:0]</td><td>These bits determine the function selected on the pin:

SELECT[2:0] — PWMn/GPOn PIN SELECTED FUNCTION
000 — PWM operation
001 — Force GPO assertion
010 — Force GPO deassertion
011 — PG/GPI operation (use bits 31:16)
100 — Alarm operation (use bits 31:16)
101 — $\overline{\text{FAULT1}}$ special function (only PAGE 7)
11x — Reserved</td></tr>
</table>

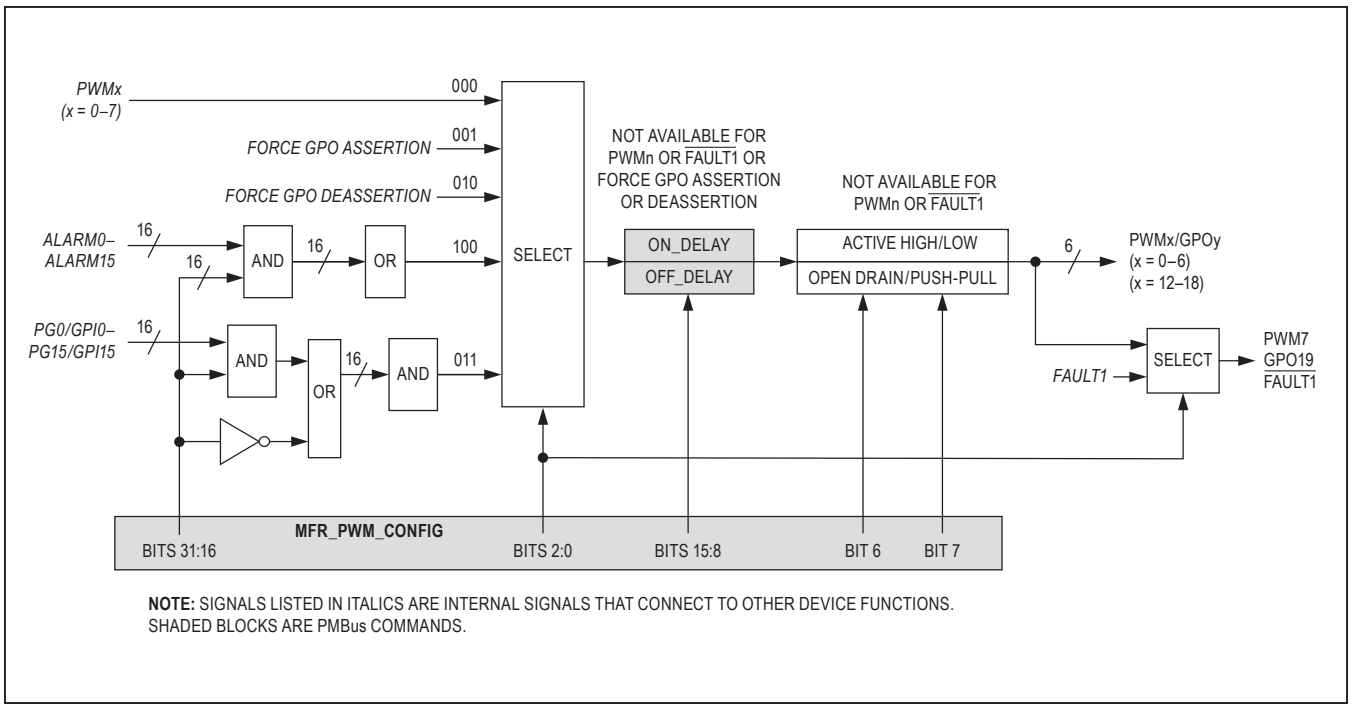


Figure 13. MFR_PWM_CONFIG Functional Logic

Delay Function

If a delay is configured either on or off, the input must be continuously static through the delay time before the output changes state (see Figure 9).

MFR_SEQ_CONFIG (E8h)

The MFR_SEQ_CONFIG command is used to configure the sequencing channels (PAGES 0–11). This command should not be changed while the power supplies are operating. The MFR_SEQ_CONFIG command is described in Table 38 and shown in Figure 2.

Each channel can be independently configured to initiate power-on sequencing, using the SELECT[1:0] bits, to one of the following conditions:

- Wait for either SEQUENCE0 or SEQUENCE1 from ON_OFF_CONFIG decode (SELECT[1:0] = 00)
- Wait for all enabled channel power-good (PG) or GPI to be asserted (SELECT[1:0] = 01)
- Wait for a match on the $\overline{SEQ}$ pin (SELECT[2:0] = 10)

If SELECT[1:0] = 00, then the channel waits for either the SEQUENCE0 or SEQUENCE1 signal to assert before powering on. The sequence signal to use is selected with the SEQ_SELECT bit. The SEQUENCE0 and SEQUENCE1 signals are generated by decoding the OPERATION command and CONTROLn pins using the ON_OFF_CONFIG command. See the ON_OFF_CONFIG command description for details. This selection would be used if the channel is being controlled by time-based sequencing.

If SELECT[1:0] = 01, then sequencing for the channel is initiated when some combination of power-goods (PGs) and general-purpose inputs (GPIs) are asserted. The channels that should be used in this combination are selected using the PG_GPI_SELECT bits 31:16. If the PG_GPI_SELECT bit is cleared, then the associated channel is not used in the logical combination to assert the GPOn output. If the PG_GPI_SELECT bit is set, then the power good or GPI from the channel is used in the logical combination to initiate the power-on sequencing. This selection would be used if the channel is being controlled by event-based sequencing.

If SELECT[1:0] = 10, then sequencing is initiated when the channel matches the selected signature on the $\overline{SEQ}$ pin. The signature to match on is selected with the SEQ_MATCH bits. The $\overline{SEQ}$ signal is used to facilitate event-based sequencing in multiple-device systems. This selection would be used if the channel is being controlled by event-based sequencing.

Table 38. MFR_SEQ_CONFIG (E8h)

BIT	NAME	MEANING
31:16	PG_GPI_SELECT	These bits are only used if SELECT[1:0] = 01; each bit corresponds to one channel (device channel N + 16 = bit number): When these bits are cleared, the power good (PG) or GPI from channel N is not used in the logical AND to initiate power-on sequencing. When these bits are set, the PG or GPI is used.
15:12	0	These bits always return a 0.
11:8	SEQ_MATCH	These bits determine which $\overline{SEQ}$ signature the channel must match before initiating power on sequencing: 0000 Disabled; 0001 Signature 1; 0010 Signature 2; 0011 Signature 3; 0100 Signature 4; 0101 Signature 5; 0110 Signature 6; 0111 Signature 7; 1000 Signature 8; 1001 Signature 9; 1010 Signature 10; 1011 Signature 11; 1100 Signature 12; 1101 Signature 13; 1110 Signature 14; 1111 Signature 15
7:6	0	These bits always return a 0.
5:4	SELECT[1:0]	These bits determine the signal that initiates power-on sequencing: SELECTED POWER-ON SEQUENCING CONTROL SIGNAL / SEQUENCING TYPE 00: SEQUENCE0 or SEQUENCE1 (use bit 0) — Time based 01: PG/GPI logic combination (use bits 31:16) — Event based 10: $\overline{SEQ}$ Match (use bits 11:8) — Event based 11: Reserved
3:1	0	These bits always return a 0.
0	SEQ_SELECT	0 = SEQUENCE0 1 = SEQUENCE1

MFR_MARGIN_CONFIG (DFh)

The MFR_MARGIN_CONFIG command configures both the digital PWMn outputs (PWM0–PWM7) and the external DS4424 current DAC (if present) to margin the associated power supplies. If the PWMn/GPOn pin is configured with MFR_PWM_CONFIG for any function besides PWM operation, this selection overrides the margining functionality. The MFR_MARGIN_CONFIG command is described in Table 39.

Power-Supply Margining Operation

For the power supplies connected to PSEN0–PSEN7 (PAGES 0–7), power-supply margining is implemented using the PWM0–PWM7 outputs, respectively. The PWM frequency is 312.5kHz. For power supplies connected to PSEN8–PSEN11 (PAGES 8–11), power-supply margining is implemented using the external DS4424 DAC outputs according Table 40. The device close-loop controls the PWM duty cycle or DAC output current setting to margin the power supply. When margining is not active, the PWMn and DAC outputs are high impedance.

The device margins the power supplies when OPERATION is set to one of the margin states. Margining of the supplies does not begin until **ALL** channels that have been configured as voltage monitoring (with or without sequencing) have exceeded their programmed POWER_GOOD_ON levels. When this happens, the PWM or DAC output is enabled and margining is initiated. The device then averages four samples of V_{OUT} for a total time of 20ms. If the measured V_{OUT} and the target (set by either VOUT_MARGIN_HIGH or VOUT_MARGIN_LOW) differ, the PWM duty cycle or the DAC setting is adjusted by one step. The direction of the duty-cycle adjustment is determined by the SLOPE bit in MFR_MARGIN_CONFIG. Use a positive slope when the DAC is directly driving the reference voltage of a power supply. Use a negative slope when the DAC is injected into the power supply's feedback pin as shown in Figure 14. All changes to the DAC setting are made after averaging four samples of V_{OUT} over a 20ms period.

Table 39. MFR_MARGIN_CONFIG (DFh)

BIT	NAME	MEANING
15	SLOPE	DAC and PWM setting to resulting voltage relationship: 0 = Negative slope DAC source current results in a lower voltage Increasing PWM duty cycle results in a lower voltage 1 = Positive slope DAC source current results in a higher voltage Decreasing PWM duty cycle results in a higher voltage
14	OPEN_LOOP	0 = Normal closed-loop margining 1 = PWM duty cycle or DAC value set constantly to the DC_DAC value when margining invoked
13:8	0	These bits always return a 0.
7:0	DC_DAC	This 8-bit value has two purposes: 1) With PWM margining, it is used as the initial PWM duty cycle when the device begins to margin a power supply either up or down. 2) When bit 14 is set, this value is used to set the PWM duty cycle or the external current DAC level.

Table 40. Power-Supply DAC Outputs

PAGE	POWER SUPPLY	DS4424 OUTPUT
8	PSEN8	OUT0
9	PSEN9	OUT1
10	PSEN10	OUT2
11	PSEN11	OUT3

When the OPERATION command deactivates margining, and the margining has been running with the “Ignore All Faults” condition, the device does not begin monitoring for faults for 100ms after the “Margin Off” input is received to allow time for the power supplies to return to a normal condition.

Margining Faults

The device detects two possible margining faults. First, if the initial PWM duty cycle or DAC step causes V_{OUT} to exceed the target value (either high or low, depending on whether the device has been instructed to margin high or low, respectively), this creates a fault. Second, if the target value cannot be reached when the PWM duty cycle or DAC reaches zero or full scale, this also creates a fault. If either margining fault occurs, the device continues attempting to margin the power supply and does the following:

1) Sets the MARGIN bit in STATUS_WORD.
2) Sets the MARGIN_FAULT bit in STATUS_MFR_SPECIFIC (PAGES 0–11).
3) Notifies the host through $\overline{ALERT}$ assertion (if enabled in MFR_MODE).

If a communication error occurs between the MAX34451 and the external DS4424, a fault occurs when the MAX34451 attempts to set the DAC to full scale and the target margin value is not reached.

DC_DAC Value

The DC_DAC value for the channels controlled by the PWMn outputs can be determined by the following formula. The DC_DAC value for the channels controlled by the external current DAC is automatically configured by the device and set to 0x00h.

$$\text{PWM DC\_DAC value} = 255 \times (V_{FB}/V_{DD})$$

where V_{FB} is the power-supply feedback node voltage and V_{DD} is the supply voltage.

Example:

$$V_{FB} = 0.8V,\ V_{DD} = 3.3V$$
$$\text{PWM DC\_DAC value} = 255 \times (0.8/3.3) \sim 62d = 0x3Eh$$

PWM/DAC Margining Component Selection

Figure 14 shows how to implement margining by injecting PWM voltage into a power supply’s feedback pin. A low-pass filter is connected to the PWM pin to filter out the pulsation.

For margin high operation, the following equation calculates the maximum allowable resistor:

$$R_{MH} = \frac{(V_{FB} - 0.3V) \times R1}{V_{OUT} \times \left(1 + \frac{MarginRangeHigh}{100}\right) - V_{FB} \times \left(1 + \frac{R1}{R2}\right)}$$

where MarginRangeHigh is the maximum required margin high range in percentage.

For margin low operation, the following equation calculates the maximum allowable resistor:

$$R_{ML} = \frac{\left(V_{FB} - \left(V_{DD} - 0.5\right)\right) \times R1}{V_{OUT} \times \left(1 - \frac{MarginRangeLow}{100}\right) - V_{FB} \times \left(1 + \frac{R1}{R2}\right)}$$

where MarginRangeLow is the maximum required margin low range in percentage and V_{DD} is the supply voltage.

To ensure that both the maximum high and low margin ranges can be achieved, choose a resistor, R_{PWM}, that is lower of R_{MH} and R_{ML}. Further, in order to account for the presence of the ripple filter resistor, choose resistor value for 'R' (see Figure 14) which is 4.7kΩ less than the calculated value of R_{PWM}.

When margining high, the device sinks current from the feedback node, and when margining low, the device sources current into the feedback node. Ensure that 'R' resistor value is chosen such that the PWM sources/sinks not more than 200µA maximum current when operating at full margin range.

Example:

V_{OUT} = 1V, V_{FB} = 0.6V, V_{DD} = 3.3V, R1 = 33.2kΩ, R2 = 49.9kΩ, Margining Range = ±10%

Using the above equations:

R_{MH} = 98.8kΩ; R_{ML} = 736.3kΩ. Choose R_{PWM} = 98.8kΩ

Set R equal to R_{PWM} - 4.7kΩ = 94.1kΩ

DAC "R_{FS}" = (7.75)/(I_{FB} x Margining range x 120%)

where I_{FB} is the feedback node current.

Example:

I_{FB} = 500µA, margining range = ±15%

DAC "RFS" value = (7.75)/(500µA x 15% x 120%) = 86kΩ

Note: 40kΩ < R_{FS} < 160kΩ

Temperature Sensor Operation

The device can monitor up to five different temperature sensors, four external sensors, plus its own internal temperature sensor. The external temperature sensors are all connected in parallel to the master I2C port (MSDA and MSCL pins). The device can support up to four DS75LV devices.

Each of the enabled temperature sensors are measured once per second. The internal temperature sensor is averaged four times to reduce the effect of noise. Each time the device attempts to read a temperature sensor, it checks for faults. For the internal temperature sensor, a fault is defined as reading greater than +130°C or less than -60°C. For the I2C temperature sensors, a fault is

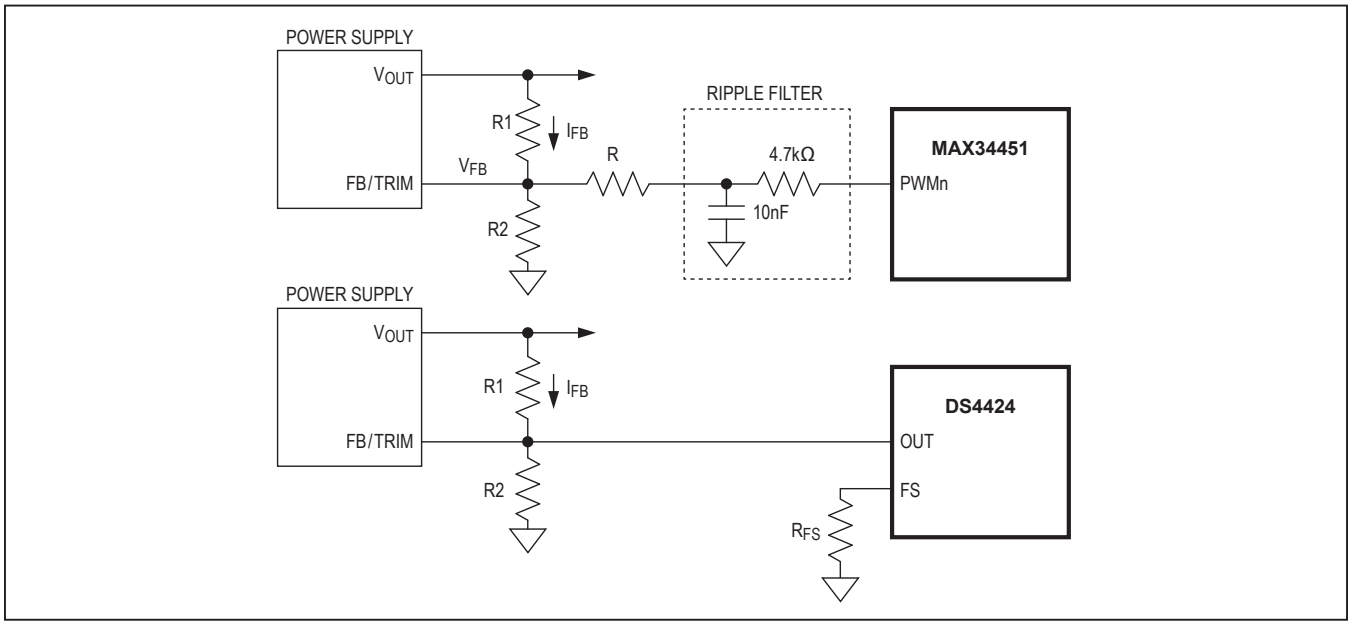


Figure 14. Margining Hardware Configurations

Table 41. DS75LV Address Pin Configuration

PAGE	MAX34451 TEMP SENSOR	DS75LV ADDRESS PIN CONFIGURATION		
		A2	A1	A0
16	MAX34451 internal	—	—	—
17	DS75LV (address 90h)	0	0	0
18	DS75LV (address 92h)	0	0	1
19	DS75LV (address 94h)	0	1	0
20	DS75LV (address 96h)	0	1	1

Table 42. MFR_TEMP_SENSOR_CONFIG (F0h)

BIT	NAME	MEANING
15	ENABLE	0 = Temperature sensor disabled 1 = Temperature sensor enabled
14:0	0	These bits always return a 0.

defined as a communication access failure. Temperature-sensor faults are reported by setting the temperature reading to 7FFFh. A temperature-sensor fault results in the setting of the TEMPERATURE bit in STATUS_WORD and $\overline{\text{ALERT}}$ is asserted (if enabled in MFR_MODE). No bits are set in STATUS_TEMPERATURE. On reset of the device, if the device cannot initialize the external DS75LV device, the TEMPERATURE bit in STATUS_WORD is set and $\overline{\text{ALERT}}$ is asserted (if enabled in MFR_MODE), but the device does not attempt to reinitialize the DS75LV until 8000h is written to MFR_TEMP_SENSOR_CONFIG. Reading disabled temperature sensors returns a fixed value of 0000h.

Up to four DS75LV digital temperature sensors can be controlled by the MAX34451. The A0–A2 pins on the DS75LV should be configured as shown in Table 41. The thermostat function on the DS75LV is not used and hence the O.S. output should be left open circuit.

MFR_TEMP_SENSOR_CONFIG (F0h)

The MFR_TEMP_SENSOR_CONFIG command is used to configure the temperature sensors. The MFR_TEMP_SENSOR_CONFIG command is described in Table 42.

Applications Information

V_{DD}, V_{DDA}, and REG18 Decoupling

To achieve the best results when using the device, decouple V_{DD} and V_{DDA} power inputs each with a 0.1µF capacitor. If possible, use a high-quality, ceramic, surface-mount capacitor. Surface-mount components minimize lead inductance, which improves performance, and ceramic capacitors tend to have adequate high-frequency response for decoupling applications. Decouple the REG18 regulator output using 1µF and 10nF capacitors with a maximum ESR of 500mΩ.

Open-Drain Pins

MSDA, MSCL, SCL, SDA, $\overline{\text{FAULTn}}$, $\overline{\text{SEQ}}$, and $\overline{\text{ALERT}}$ are open-drain pins and require external pullup resistors connected to V_{DD} to realize high logic levels.

PSEN0–PSEN11 can be user-configured as either CMOS push-pull or open-drain outputs. When configured as open drain (see MFR_PSEN_CONFIG), external pullup resistors connected to V_{DD} are required to realize high logic levels.

Keep-Alive Circuit

In systems where the power to the device may not always be present, a keep-alive circuit consisting of a Schottky diode and a bulk capacitor can be added to allow the device time to orderly shut down the power supplies it is controlling and write fault log (if configured) to flash memory before power is lost.

Configuration Port

Some applications require the ability to configure the device when the device has been mounted on a PCB. In such applications, a 3- or 4-wire header can be added to allow access to the slave I^2C pins.

Resistor-Dividers and Source Impedance for RSn Inputs

The maximum full-scale voltage on the ADC inputs is 2.048V (nominal). A resistor-divider must be used to measure voltages greater than 1.8V. The maximum source impedance to the RSn inputs is determined by the ADC_TIME bits in MFR_MODE. See the *Recommended Operating Conditions* section for more details.

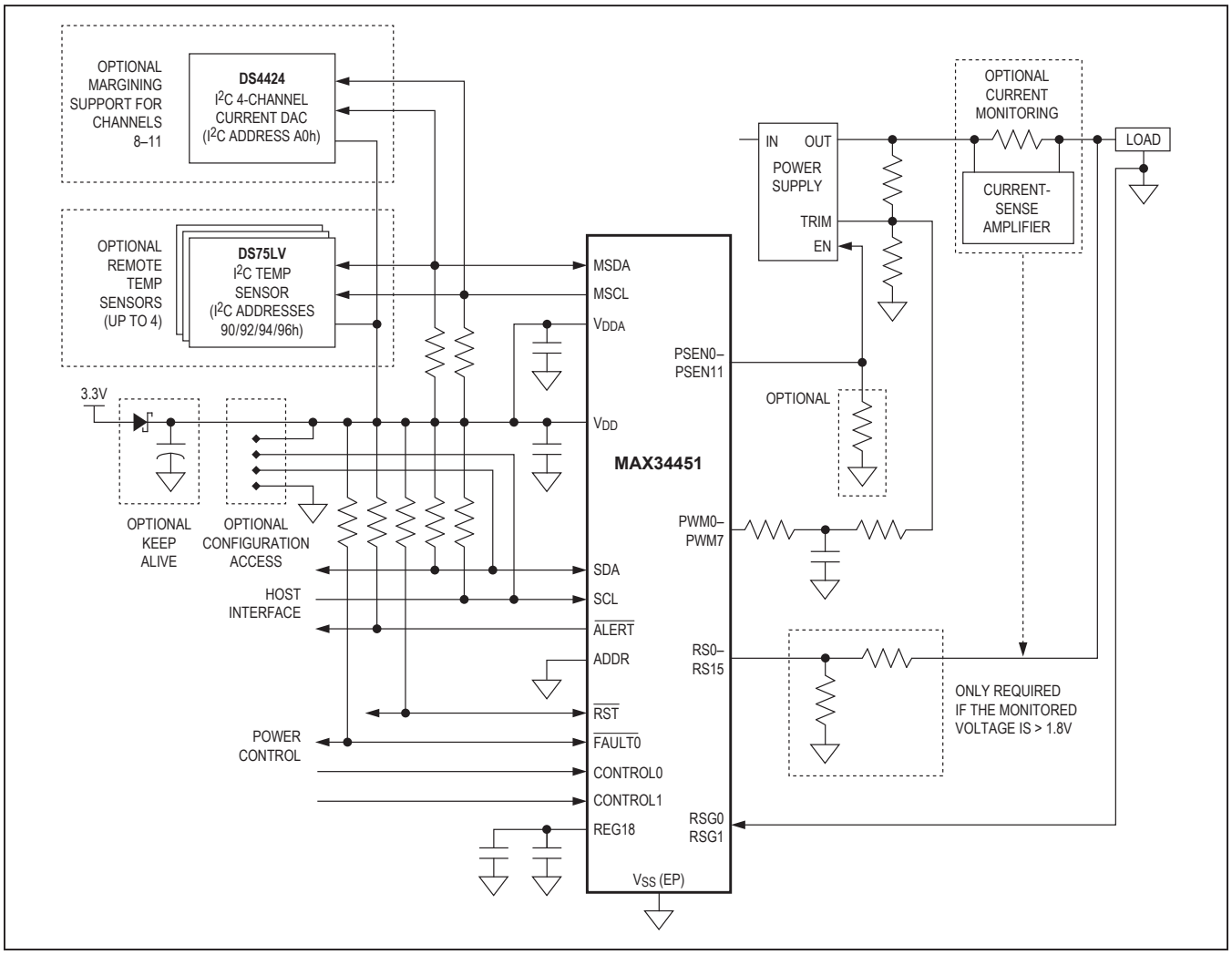
Protecting Input Pins

In applications where voltages can be applied to the RSn or CONTROLn signals, when V_{DD} or V_{DDA} is grounded, a series 100Ω resistor is recommended to protect the device by limiting power dissipation.

Exposed Pad Grounding

The device uses the exposed pad of the TQFN package as the common ground (V_{SS}) for the entire device. The exposed pad must be connected to the local ground plane.

Typical Operating Circuit



Ordering Information

PART NUMBER	PIN-PACKAGE	FIRMWARE	STATUS
MAX34451ETNA2+	56 TQFN-EP*	0005	ACTIVE
MAX34451ETNA2+T	56 TQFN-EP*	0005	ACTIVE
MAX34451ETNA3+	56 TQFN-EP*	0007	ACTIVE
MAX34451ETNA3+T	56 TQFN-EP*	0007	ACTIVE
MAX34451ETNA4+**	56 TQFN-EP*	0009	ACTIVE
MAX34451ETNA4+T**	56 TQFN-EP*	0009	ACTIVE

+Denotes a lead(Pb)-free/RoHS-compliant package.
T = Tape and reel.
**EP = Exposed pad.*
***MAX34451ETNA4+ and MAX34451ETNA4+T are recommended for all future designs.*
***Note:** Part operates over the -40°C to +85°C temperature range.*

Revision History

REVISION NUMBER	REVISION DATE	DESCRIPTION	PAGES CHANGED
0	3/13	Initial release	—
1	8/13	Added V_{DD} rise time and V_{DD} source impedance to *Recommended Operating Conditions* table, updated the 7-bit slave addresses in Table 4	2, 17
2	5/15	Updated *Benefits and Features* section and moved *Ordering Information* to page 1	1, 66
3	4/16	(Firmware update) Updated *Ordering Information*, Table 3, Table 5, Table 7, $\overline{SEQ}$ *Pin Operation* section, *Clear Faults (03h)* section, *POWER_GOOD_ON* section, Figure 7, Table 25, *MFR_VOUT_MIN (D7h)* section, *MFR_FAULT_RETRY (DAh)* section, Table 36, *Power-Supply Margining Operation* section, *DC-DAC Value* section, PWM/DAC *Margining Component Selection* section, Figure 14, *Keep-Alive Circuit* section	1, 15-16, 21-22, 26, 32, 37-38, 41, 46, 53, 57, 63-65
4	3/17	Added future parts to MAX34451ETNA2+ and MAX34451ETNA2+T in the *Ordering Information* section.	75
5	7/17	Updated Table 3 and *TOFF_DELAY (64h), MFR_PSEN_CONFIG (D2h), MFR_NV_FAULT_LOG (DCh)*, and *MFR_PWM_CONFIG (E7h)* sections. Removed future part designations for MAX34451ETNA2+ and MAX34451ETNA2+T, and removed MAX34451ETNA1+ and MAX34451ENTA1+T	22, 45, 52, 61, 68, 75
6	5/18	Added MAX34451ETNA3+ and MAX34451ETNA3+T to the *Ordering Information* table.	75
7	6/18	Updated Table 3, Table 4, *SMBus/PMBus Operation Examples*, and *PMBus Operation, POWER_GOOD_ON (5Eh), POWER_GOOD_OFF (5Fh), MFR_NV_FAULT_LOG (DCh)* and *Power-Supply Margining Operation* sections; Updated the *Ordering Information* table.	22, 24, 27 45, 61, 71, 75
8	9/19	Updated *Ordering Information*	75
9	10/19	Updated *Ordering Information*	75

For pricing, delivery, and ordering information, please visit Maxim Integrated's online storefront at https://www.maximintegrated.com/en/storefront/storefront.html.

