Intel® IXF440 Multiport 10/100 Mbps Ethernet Controller

Datasheet

Product Features

The Intel® IXF440 Multiport 10/100 Mbps Ethernet Controller provides eight 10/100 Mbps intelligent, high-performance MAC ports. It includes network management support and is optimized for switch applications.

- Integration
  - Eight Ethernet 10/100 Mbps MAC ports
  - Onchip scrambler, descrambler, and PCS functions for 100BASE-X connection
  - Handles SNMP and RMON counters
- IX Bus
  - Supports a 4 Gbps high bus bandwidth
  - Variable bus speed of 25 to 66 MHz operational, and from 16 MHz for testing
  - Interfaces a 64-bit bus with a 32-bit optional mode
  - Supports concurrent unidirectional 32-bit buses for transmit and receive in split IX Bus mode
  - Provides transmit- and receive-independent 256-byte FIFOs for each of the eight ports
  - Offers a generic slave FIFO interface
  - Supports little or big endian byte ordering
  - Supports transmit and receive byte alignment
  - Supports receive packet fragmentation on byte boundaries (replay feature)
  - Provides programmable transmit and receive bus thresholds
  - Appends packet status to received packet
- Performance
  - Allows status register access without interrupting packet transfer
  - Enables early address filtering ability, with packet header preprocessing and VLAN detection ability
  - Offers retry or ignore options following packet transmission errors
  - Supports automatic retransmission following excessive collisions
  - Provides programmable automatic discard of badly received packets such as runts, CRC errors, and too long packets
- Operation
  - Enables independent 10 or 100 Mbps port operation
  - Provides full-duplex support
  - Enables standard flow-control functionality in full-duplex mode
  - Offers backpressure logic capability in half-duplex mode
  - Interfaces to standard MII connections
  - Supports 10BASE-T, 100BASE-TX, 100BASE-T4, and 100BASE-FX connections
  - Provides programmable CRC generation and removal
  - Allows backoff limit programming
  - Provides full collision support, including jamming, backoff, and automatic retransmission
  - Complies with IEEE 802.3 Standard
- CPU Interface
  - Supports fully programmable independent ports through a dedicated generic CPU port
  - Enables interrupt programming
- Device
  - Optimized for switch, bridge, and router applications
  - Includes internal and external loopback capabilities
  - Provides software reset support
  - Supports JTAG boundary scan
  - Low-power 3.3 V and 5 V tolerant CMOS device
  - 352-BGA package

Notice: This document contains preliminary information on new products in production. The specifications are subject to change without notice. Verify with your local Intel sales office that you have the latest datasheet before finalizing a design.

Order Number: 278160-008
June, 2002

Revision History

Date	Revision	Description
11/98	001	First Intel version.
3/3/00	002	First update.
4/24/00	003	Device name change to IXF440. Describe operation in various IX Bus modes.
5/25/00	004	Changed to Intel branding.
9/26/00	005	Updates for v1.1 release of IXP1200 software.
5/18/01	006	Updates for the v1.3 SDK.
10/02/01	007	Changes to Table 22, Table 24, Section 9.0, Figure 34.
05/23/02	008	Removed reference to GCIXF440ACT product.

Information in this document is provided in connection with Intel® products. No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted by this document. Except as provided in Intel's Terms and Conditions of Sale for such products, Intel assumes no liability whatsoever, and Intel disclaims any express or implied warranty, relating to sale and/or use of Intel products including liability or warranties relating to fitness for a particular purpose, merchantability, or infringement of any patent, copyright or other intellectual property right. Intel products are not intended for use in medical, life saving, or life sustaining applications.

Intel may make changes to specifications and product descriptions at any time, without notice.

Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them.

The IXF440 may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

Copies of documents which have an ordering number and are referenced in this document, or other Intel literature may be obtained by calling 1-800-548-4725 or by visiting Intel's website at http://www.intel.com.

Copyright © Intel Corporation, 2002

Intel is a registered trademark of Intel Corporation or its subsidiaries in the United States and other countries

*Other names and brands may be claimed as the property of others.

Contents

1.0 Introduction 9
- 1.1 General Description 9
- 1.2 Block Diagram 9

2.0 Pinout 11
- 2.1 Signal Descriptions 11
- 2.2 Pin Count 16
- 2.3 Connection Rules 16
- 2.4 Pin List 17

3.0 Register Descriptions 21
- 3.1 Register Conventions 21
  - 3.1.1 Access Rules 21
- 3.2 CSR Register 22
  - 3.2.1 Register Mapping 22
  - 3.2.2 Base Registers 23
    - 3.2.2.1 Chip Interrupt Summary Register 23
    - 3.2.2.2 Interrupt Status Register 24
    - 3.2.2.3 Interrupt Enable Register 25
    - 3.2.2.4 Transmit Status Register 26
    - 3.2.2.5 Receive Status Register 27
    - 3.2.2.6 Port Control Register 28
    - 3.2.2.7 Device ID Register 29
    - 3.2.2.8 Revision ID Register 29
    - 3.2.2.9 Serial Command Register 30
  - 3.2.3 Configuration Registers 31
    - 3.2.3.1 FIFO Threshold Register 31
    - 3.2.3.2 IX Bus Mode Register 32
    - 3.2.3.3 Transmit Parameters Register 33
    - 3.2.3.4 Transmit Error Mode Register 34
    - 3.2.3.5 Transmit Threshold and Backoff Register 35
    - 3.2.3.6 Receive Parameters Register 36
    - 3.2.3.7 Receive Filtering Mode Register 37
    - 3.2.3.8 Transmit Pause Time Register 38
    - 3.2.3.9 Maximum Packet Size Register 38
    - 3.2.3.10 InterPacket Gap Register 39
  - 3.2.4 Serial Registers 40
    - 3.2.4.1 Serial Mode Register 40
    - 3.2.4.2 Link Status Register 41
    - 3.2.4.3 Physical Address Register 41
- 3.3 Network Statistic Counter Mapping 42
  - 3.3.1 Register Mapping 42
  - 3.3.2 Network Statistic Counters Access Rules 44
- 3.4 Access Sequences 44
  - 3.4.1 Initialization Sequence 44
  - 3.4.2 Mode Change Sequence 44
  - 3.4.3 Interrupt Handling Sequence 45

4.0 FIFO Interface Operation 47
4.1 FIFO Interface 47
4.1.1 Byte Ordering on IX Bus 47
4.1.2 FIFO Status Signaling 48
4.2 Packet Transmission 48
4.2.1 Packet Loading 48
4.2.2 Network Transmission 49
4.2.3 Excessive Collisions 49
4.2.4 Late Collision 49
4.2.5 FIFO Underflow 49
4.2.6 Stopping Mode on Transmission Errors 50
4.2.7 Transmit Flow Diagram 51
4.3 Packet Reception 51
4.3.1 Packet Storing 52
4.3.2 Header Preprocessing 53
4.3.3 Packet Segmentation 53
4.3.4 Packet Abortion 53
4.3.5 Network Reception 54
4.3.6 Rejecting Mode on Reception Errors 54
4.3.7 Accepting Mode on Reception Errors 54
4.3.8 Receive Flow Diagram 55

5.0 CPU Interface Operation 57
5.1 CPU Interface 57
5.2 Network Management 57
5.2.1 SNMP MIB Support 58
5.2.2 RMON Statistic Group Support 59
5.2.3 RMON Host Group Support 60

6.0 Network Interface Operation 61
6.1 Operating Modes 61
6.2 MII Port Interface 62
6.3 MAC Frame Format 63
6.4 MAC Transmit Operation 64
6.4.1 Transmit Initiation 64
6.4.2 Initial Deferral 64
6.4.3 Frame Encapsulation 64
6.4.4 Collision 65
6.4.5 Terminating Transmission 65
6.4.6 Backpressure 66
6.4.7 Flow Control 66
6.5 MAC Receive Operation 66
6.5.1 Receive Initiation 66
6.5.2 Preamble Processing 67
6.5.3 Frame Decapsulation 67
6.5.4 Terminating Reception 67
6.5.5 Flow Control 68
6.6 MAC Full-Duplex Operation 68
6.7 MAC Loopback Operations 68
6.7.1 Internal Loopback Mode 68

6.7.2 External Loopback Mode 69
6.8 SYM Mode 69

7.0 Timing Diagrams 71

7.1 IX Bus Port Timing Diagrams 71
7.1.1 Transmit Start-of-Packet Timing 71
7.1.2 Transmit End-of-Packet Timing 72
7.1.3 Transmit FIFO Control Timing 72
7.1.4 Transmit Ready (txrdy) Timing 73
7.1.5 Receive Start-of-Packet Timing 73
7.1.6 Receive End-of-Packet Timing 74
7.1.7 Fastest Receive Reaccess after EOP 74
7.1.8 Receive rxfail Timing 75
7.1.9 Receive rxabt Timing 75
7.1.10 Receive rxkep Timing 76
7.1.11 Receive Header Replay Timing 76
7.1.12 Receive FIFO Control Timing 76
7.1.13 Receive Ready (rxrdy) Control Timing 77
7.1.14 Consecutive Transmit-Transmit Timing 77
7.1.15 Consecutive Transmit-Receive Timing 78
7.1.16 Consecutive Receive-Transmit Timing 78
7.1.17 Consecutive Receive-Receive Timing 78
7.2 MII/SYM Port Timing Diagrams 78
7.2.1 Packet Transmission Timing 79
7.2.2 Packet Reception Timing 79
7.2.3 Transmission with Collision Timing 79
7.2.4 False Carrier Timing 79

8.0 Electrical and Environmental Specifications 81

8.1 Functional Operating Range 81
8.2 Absolute Maximum Rating 81
8.3 Supply Current and Power Dissipation 82
8.4 Temperature Limit Ratings 82
8.5 Reset Specification 82
8.6 FIFO Port Specifications 83
8.6.1 Clock Specification 83
8.6.2 3.3 Volt DC Specifications 83
8.6.3 5 Volt DC Specifications 84
8.6.4 IX Bus Signals Timing 84
8.7 CPU Port Specifications 85
8.7.1 DC Specifications 85
8.7.2 Signals Timing 85
8.7.2.1 Read Timing 85
8.7.2.2 Write Timing 86
8.7.2.3 Timing Parameters 87
8.8 MII/SYM Port Specifications 87
8.8.1 DC Specifications 87
8.8.2 Signals Timing 87
8.8.2.1 Clocks Specifications 88
8.8.2.2 Signals Timing Diagrams 88
8.8.2.3 Data Timing Parameters 89

8.9 JTAG Port Specifications 89
8.9.1 DC Specifications 89
8.9.2 Signals Timing 90

9.0 Mechanical Specifications 93

A Joint Test Action Group – Test Logic 97

A.1 General Description 97
A.1.1 Test Access Port Controller 97
A.2 Registers 97
A.2.1 Instruction Register 98
A.2.2 Bypass Register 98
A.2.3 Boundary-Scan Register 98

Figures

1 Block Diagram 9
2 Transmit Flow Diagram 51
3 Receive Flow Diagram 55
4 Transmit Start-of-Packet Timing 71
5 Transmit End-of-Packet Timing 72
6 Transmit FIFO Control Timing 72
7 Transmit txrdy Timing 73
8 Receive Start-of-Packet Timing 73
9 Receive End-of-Packet Timing 74
10 Fastest Receive Reaccess After EOP 74
11 Receive rxfail Timing 75
12 Receive rxabt Timing 75
13 Receive rxkep Timing 76
14 Receive Header Replay Timing 76
15 Receive FIFO Control Timing 76
16 Receive rxrdy Timing 77
17 Consecutive Transmit-Transmit Timing 77
18 Consecutive Transmit-Receive Timing 78
19 Consecutive Receive-Transmit Timing 78
20 Consecutive Receive-Receive Timing 78
21 Packet Transmission Timing 79
22 Packet Reception Timing 79
23 Transmission with Collision Timing 79
24 False Carrier Timing 79
25 IX Bus Clock Timing Diagram 83
26 IX Bus Signals Timing Diagram 84
27 CPU Port Read Timing Diagram 86
28 CPU Port Write Timing Diagram 86
29 MII/SYM Clock Timing Diagram 88
30 MII/SYM Port Transmit Timing Diagram 88
31 MII/SYM Port Receive Timing Diagram 89
32 MII/SYM Port Carrier Sense and Collision Timing Diagram 89
33 JTAG Port Timing Diagram 90
34 Part Marking 93
35 352-BGA Package - Bottom View 94
36 Side View 94
37 **A–A Section View 94**

Tables

1 Component Description 10
2 Signal Descriptions 11
3 IXF440 Pin Count 16
4 Pin List 17
5 CSR Register Mapping 22
6 Network Statistic Register Mapping 42
7 Little Endian, 64-Bit Bus (BEND=0, BWID=1) 47
8 Little Endian, 32-Bit Bus (BEND=0, BWID=0) 47
9 Big Endian, 64-Bit Bus (BEND=1, BWID=1) 47
10 Big Endian, 32-Bit Bus (BEND=1, BWID=0) 47
11 IX Bus Receive Packet Status 52
12 SNMP MIB to IXF440 Counters Mapping 58
13 RMON Statistics to IXF440 Counters Mapping 59
14 RMON Host to IXF440 Counters Mapping 60
15 MII Port Signals versus Standard Signals 62
16 Ethernet Frame Description 63
17 Flow control Field Matching 68
18 SYM Port Signal Description 69
19 Functional Operating Range 81
20 3.3 V AC Signaling Specifications 81
21 5 V AC Signaling Specifications 81
22 Absolute Maximum Rating 82
23 Supply Current and Power Dissipation 82
24 Temperature Limit Ratings 82
25 IX Bus Clock Timing Specifications 83
26 IX Bus 3.3 V Signaling Specifications 83
27 IX Bus 5-V Signaling Specifications 84
28 IX Bus Signals Timing Specifications 84
29 CPU Port DC Specifications 85
30 Timing Parameters 87
31 MII/SYM Port DC Specifications 87
32 MII/SYM Port Signals Timing Specifications 88
33 Data Timing Parameters 89
34 JTAG Port DC Specifications 90
35 JTAG Port Timing Specifications 91
36 352-BGA Dimensional Attributes 94
A-1 Instructions Register 98

1.0 Introduction

1.1 General Description

The Intel® IXF440 Multiport Ethernet Controller includes eight independent 10/100 Mbps Ethernet MACs and interfaces directly to MII standard PHYs or SYM 100BASE-TX physical devices. The IXF440 handles SNMP and RMON management counter sets, accessible through a generic CPU interface, which is also used for mode programming. Each MAC includes two 256-byte independent FIFOs for packet transmit and receive. All the packets are transferred onto a high-performance, common FIFO interface, or IX Bus. The IXF440 is fully compatible with IEEE standards, including Fast Ethernet and flow-control support. It is implemented in a low-power 3.3 V CMOS device within a 352-BGA package.

1.2 Block Diagram

Figure 1 is a simplified block diagram of the IXF440. Table 1 provides brief descriptions of the major components shown in the figure.

Figure 1. Block Diagram

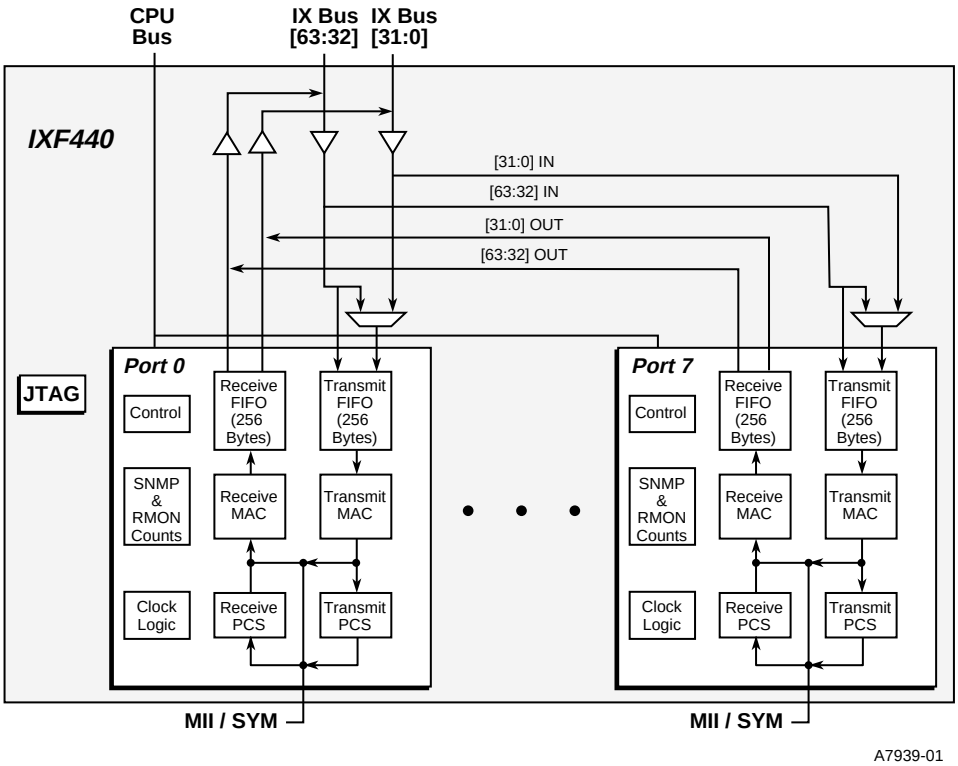


Table 1. Component Description

Component	Description
Transmit FIFO	Handles the transmitted packets while taking care of retransmission in case of collision. The transmit FIFO has direct interface to the IX Bus.
Receive FIFO	Handles the received packets and supports packet deletion in case of errors. The receive FIFO has direct interface to the IX Bus.
Transmit MAC	Implements the IEEE 802.3 transmit MAC functions while interfacing between the transmit FIFO and the front-end ENDEC, providing full MII interface.
Receive MAC	Implements the IEEE 802.3 receive MAC functions while interfacing between the front-end ENDEC through MII interface and the receive FIFO.
PCS	ENDEC implementing the 100BASE-X PCS layer, including the 100BASE-TX scrambler/descrambler function.
SNMP and RMON	Handles network statistic counters for SNMP and RMON.
Control	Handles the chip registers accessible through the CPU bus.
Clock Logic	Generates the clocks required by the chip.
JTAG	Includes the JTAG boundary scan logic.

2.0 Pinout

This section describes the IXF440 pinoust.

2.1 Signal Descriptions

Table 2 describes the signals that the IXF440 uses.

The following conventions are used for the signal names:

_l	Indicates that the pin is active low.
{i}	The i subscript appended to a pin name indicates that each port has its own pin (numbered from 0 to 7).
x/y	Multiple function pin. The pin has name x for one function and name y for another.

The following abbreviations are used for I/O:

I	Input
O	Output
OD	Open Drain
I/O	Input/Output

The following abbreviations are used in the pin descriptions:

Full	64-bit Bidirectional IX Bus Mode
Narrow	32-bit Bidirectional IX Bus Mode
Split	32-bit (Dual) Unidirectional IX Bus Mode

Table 2. Signal Descriptions

Signal Name	I/O	Pin Description
CPU Interface		
cs_l	I	Chip select. This pin must be asserted to enable CPU access to the chip registers.
cps[2:0]	I	CPU port select. Selects one of the 8 internal ports for register accesses.
crd_l	I	Read strobe. Upon assertion, the address signals cadd[9:0], cs_l, and cps[2:0] are latched by the chip. Deassertion occurs after the read data is latched from the cdat[7:0] bus.
cwr_l	I	Write strobe. Upon assertion, the address signals cadd[9:0], cs_l, and cps[2:0] are latched by the chip. Deassertion must occur while the data is valid on the cdat[7:0] bus.
crdy_l	OD	Ready indication. When asserted, indicates that either data is stable on the cdat[7:0] bus during read access or that data was latched by the chip during write access.

Table 2. Signal Descriptions (Continued)

Signal Name	I/O	Pin Description
cadd[9:0]	I	Address bus. Selects one of the internal registers to be accessed.
cdat[7:0]	I/O	CPU data bus. Carries data to be written to or read from the registers.
cint7_l/RESERVED	OD/I	Control interrupt 7/Reserved In Full or Narrow, a control interrupt line for MAC 7. Asserted following a variety of programmable conditions. Deassertion occurs after reading the event(s) that caused the interrupt, unless another interrupt is registered meanwhile. In Split, this pin is RESERVED.
cint6_l/tx_fps[2]	OD/I	Control interrupt 6/Transmit FIFO Port Select [2] In Full or Narrow, a control interrupt line for MAC 6. Asserted following a variety of programmable conditions. Deassertion occurs after reading the event(s) that caused the interrupt, unless another interrupt is registered meanwhile. In Split, this pin is used as bit 2 of the transmit FIFO port select address.
cint5_l/tx_fps[1]	OD/I	Control interrupt 5/Transmit FIFO Port Select [1] In Full or Narrow, a control interrupt line for MAC 5. Asserted following a variety of programmable conditions. Deassertion occurs after reading the event(s) that caused the interrupt, unless another interrupt is registered meanwhile. In Split, this pin is used as bit 1 of the transmit FIFO port select address.
cint4_l/tx_fps[0]	OD/I	Control interrupt 4/Transmit FIFO Port Select [0] In Full or Narrow, a control interrupt line for MAC 4. Asserted following a variety of programmable conditions. Deassertion occurs after reading the event(s) that caused the interrupt, unless another interrupt is registered meanwhile. In Split, this pin is used as the bit 0 of the transmit FIFO port select address.
cint3_l/RESERVED	OD/I	Control interrupt 3/Reserved In Full or Narrow, a control interrupt line for MAC 3. Asserted following a variety of programmable conditions. Deassertion occurs after reading the event(s) that caused the interrupt, unless another interrupt is registered meanwhile. In Split, this pin is RESERVED.
cint2_l/tx_sop	OD/I	Control interrupt 2/Transmit Start of Packet (SOP) In Full or Narrow, a control interrupt line for MAC 2. Asserted following a variety of programmable conditions. Deassertion occurs after reading the event(s) that caused the interrupt, unless another interrupt is registered meanwhile. In Split, this pin is the transmit SOP input.
cint1_l/tx_eop	OD/I	Control interrupt 1/Transmit End of Packet (EOP) In Full or Narrow, a control interrupt line for MAC 1. Asserted following a variety of programmable conditions. Deassertion occurs after reading the event(s) that caused the interrupt, unless another interrupt is registered meanwhile. In Split, this pin is the transmit EOP input.
cint0_l/cint_any_l	OD	Control interrupt 0/Control interrupt for any MAC In Full or Narrow, a control interrupt line for MAC 0. Asserted following a variety of programmable conditions. Deassertion occurs after reading the event(s) that caused the interrupt, unless another interrupt is registered meanwhile. In Split, this pin is the logical NOR of all control interrupt lines, MAC7 through MAC0.

Table 2. Signal Descriptions (Continued)

Signal Name	I/O	Pin Description
reset_l	I	General reset. Upon reset, all the registers are reset to their default values and the FIFOs are flushed.
FIFO Interface		
clk	I	System clock. All the FIFO data transfers are synchronized to this clock.
txsel_l	I	Transmit select. This pin must be asserted to enable transmit FIFO write access.
rxsel_l	I	Receive select. This pin must be asserted to enable receive FIFO read access. The following signals are driven upon assertion of rxsel_l: • fdat<63:0>, fbe_l<7:0>, sop, eop and rxfail in full-64 FIFO Bus mode. • fdat<31:0>, fbe_l<3:0>, sop_rxf, eop_rxf and rxfail in split FIFO Bus mode. • fdat<31:0>, fbe_l<3:0>, sop, eop and rxfail in narrow FIFO Bus mode.
fps[2:0]/rx_fps[2:0]	I	FIFO port select/Receive FIFO port select. In Full or Narrow, selects one of eight port FIFOs for data transfer. In Split, selects one of eight port FIFOs for reading data, through fdat[31:0], from the receive FIFO of the selected port.
fdat[63:32]/tx_fdat[31:0]	I/O	FIFO data bus/Transmit FIFO data bus. In Full, carries bits [63:32] of the data to be written to the transmit FIFO or read from the receive FIFO of the selected port. In Narrow, fdat[63:32] should be connected to pull-up resistors. In Split, carries bits [31:0] of the data to be written to the transmit FIFO of the selected port.
fdat[31:0]/rx_fdat[31:0]	I/O	FIFO data bus/Receive FIFO data bus. In Full, carries bits [31:0] of the data to be written to the transmit FIFO or read from the receive FIFO of the selected port. In Split, carries bits [31:0] of the data to be read from the receive FIFO of the selected port.
fbe_l[7:4]/tx_fbe[3:0]	I/O	FIFO byte enable/Transmit FIFO byte enable. In Full: During transmit, indicates which of the bytes driven onto fdat[63:32] contain valid data (valid bytes need to be contiguous and at least one byte must be valid). During receive, indicates which bytes are valid. Each fbe_l signal relates to a different fdat byte (for example, fbe_l[4] relates to fdat[39:32] and fbe_l[5] relates to fdat[47:40]). In Narrow, fbe_l[7:4] should be connected to pull up resistors. In Split, tx_fbe[3:0] are byte enables for tx_fdat[31:0] (input) that carries the data to be written to the transmit FIFO of the selected port.
fbe_l[3:0]/rx_fbe[3:0]	I/O	FIFO byte enable/Receive FIFO byte enable. In Full or Narrow: During transmit, indicates which of the bytes driven onto fdat[31:0] contain valid data (valid bytes need to be contiguous and at least one byte must be valid). During receive, indicates which bytes are valid. Each fbe_l signal relates to a different fdat byte (for example, fbe_l[2] relates to fdat[23:16] and fbe_l[0] relates to fdat[7:0]). In Split, rx_fbe[3:0] are byte enables for rx_fdat[31:0] (output) that carries the data to be read from the receive FIFO of the selected port.

Table 2. Signal Descriptions (Continued)

Signal Name	I/O	Pin Description
rxkep	I	Receive keep. When asserted, this signal causes the last read data to be kept in the receive FIFO. May be asserted only with rxsel_l assertion.
sop/rx_sop	I/O	Start of packet/Receive start of packet. In Full or Narrow, when asserted during transmit, indicates that the first data in the packet is written to the transmit FIFO. During receive, sop is asserted when the first data of the packet is read from the receive FIFO. In Split, rx_sop is asserted when the first data of the packet is read from the receive FIFO.
eop/rx_eop	I/O	End of packet/Receive end of packet. In Full or Narrow, when asserted during transmit, indicates that the final data in the packet is written to the transmit FIFO. During receive, eop is asserted when the final data of the packet is read from the receive FIFO. In the following FIFO access, the packet status is driven onto the bus. In Split, rx_eop is asserted when the final data of the packet is read from the receive FIFO. In the following FIFO access, the packet status is driven onto the bus.
txasis/txerr	I	Transmit as is/Transmit error. When asserted during transmit, upon transfer of the packet's first data (together with sop assertion), no padding and/or CRC is appended to the packet even if the port was programmed to do so. When asserted upon transfer of the packet's final data (together with eop assertion), the packet is transmitted with an MII error (if the port is programmed to append CRC) and with a symbol error.
rxfail	O	Receive packet failure. This signal is asserted if a packet was received with errors, had started to appear on the IX Bus, and was discarded from the receive FIFO.
rxabt	I	Receive abort. This signal forces a received packet to be aborted and flushed from the receive FIFO. May be asserted only with rxsel_l assertion.
flct{i}	I	Flow control. When asserted in the half-duplex mode, a collision is generated on each received packet. When asserted in the full-duplex mode, a flow-control packet with the programmed pause time is transmitted. Upon deassertion, a flow-control packet with time equal to 0 is sent if programmed accordingly.
txctl_l	I	Transmit control enable. When asserted, this pin enables txrdy{i} output drivers to report the transmit FIFO status.
rxctl_l	I	Receive control enable. When asserted, this pin enables rxrdy{i} output drivers to report receive FIFO status.
txrdy{i}	O	Transmit FIFO ready. Indicates whether there is enough available space in the transmit FIFO to load data according to the programmable threshold value. Following transmission stop due to an error, the txrdy signal remains deasserted until the transmit error status is read.
rxrdy{i}	O	Receive FIFO ready. Indicates whether there is enough available data in the receive FIFO to be stored according to the programmable threshold value or if the end of the transferred packet is in the FIFO. The rxrdy signal may also be asserted when the packet header is in the FIFO. rxrdy can also be asserted to report packet discard from the receive FIFO due to an error together with the rxfail signal.

Table 2. Signal Descriptions (Continued)

Signal Name	I/O	Pin Description	
Vdd_clmp	I	Vdd clamp. Should be connected to the power of the highest signal level used on the IX Bus.	
MII / SYM Interface			
Note: The pins have various functions according to the port mode: MII or SYM. See a detailed description in Section 6, Network Interface Operation.			
tclk{i}	I	Transmit clock.	
txd{i}[3:0]	O	MII mode:	Nibble transmit data.
		SYM mode:	Four low bits of the encoded transmit data.
ten{i}/txd{i}[4]	O	MII mode:	Transmit enable signal.
		SYM mode:	The fifth bit of the encoded transmit data.
rclk{i}	I	Receive recovered clock.	
rxd{i}[3:0]	I	MII mode:	Nibble receive data.
		SYM mode:	Four low bits of the encoded received data.
dv{i}/rxd{i}[4]	I	MII mode:	Receive enable signal.
		SYM mode:	The fifth bit of the encoded received data.
col{i}/act{i}	I/O	MII mode:	Collision detection signal (input).
		SYM mode:	Activity indication (output).
crs{i}/sd{i}	I	MII mode:	Carrier detection signal.
		SYM mode:	Signal detection indicating link status.
rerr{i}	I	MII mode:	Receive error signal.
		SYM mode:	Must be connected to 0.
terr{i}/lnk{i}	O	MII mode:	Transmit error signal.
		SYM mode:	Link indication.
mdc	O	MII management clock.	
mdio	I/O	MII management input/output serial data.	
JTAG Interface			
tck	I	JTAG clock. If this pin is not used, it must be connected to 0.	
tms	I	JTAG test mode. If this pin is not used, it must be connected to 1.	
tdi	I	JTAG data serial input. If this pin is not used, it must be connected to 1.	
tdo	O	JTAG data serial output.	

2.2 Pin Count

Table 3 summarizes the IXF440 pin count.

Table 3. IXF440 Pin Count

Pin	Common	Per Port	Total
CPU	26	1/1 of cint{i}_I[1]	34/27[1]
FIFO	87/92[1]	3	111/116[1]
MII/SYM	2	16	130
JTAG	4	—	4
Total I/O	119	20	279
Gnd	28	—	28
Vdd	24	—	24
Reserved	21	—	21
Total			352

1. The number to the right of the slash applies when operating in Split.

2.3 Connection Rules

- All the reserved pins must remain unconnected.
- All the OD (Open Drain) signals must be connected to a pull-up device.
- All signals connected to a fixed value should be connected through a resistive device. The following are recommended resistor values:
  - Unused I/O pins: 10 KΩ pull-up
  - Unused I pins: 1–10 KΩ pull-up or pull-down
  - Unused O pins: no pull-up required
  - OD pins: R=T/C (required signal deassertion time (T) divided by capacitive load (C))

2.4 Pin List

Table 4 lists the IXF440 pins and their associated names.

Table 4. Pin List

Pin Number	Pin Name	Pin Number	Pin Name	Pin Number	Pin Name
A1	Gnd	B1	Gnd	C1	sop/rx_sop
A2	Gnd	B2	Vdd	C2	Gnd
A3	fdat[0]/rx_fdat[0]	B3	Gnd	C3	Vdd
A4	fdat[4]/rx_fdat[4]	B4	fdat[1]/rx_fdat[1]	C4	RESERVED
A5	fdat[8]/rx_fdat[8]	B5	fdat[5]/rx_fdat[5]	C5	fdat[2]/rx_fdat[2]
A6	fdat[12]/rx_fdat[12]	B6	fdat[9]/rx_fdat[9]	C6	fdat[6]/rx_fdat[6]
A7	fdat[15]/rx_fdat[15]	B7	fdat[13]/rx_fdat[13]	C7	fdat[10]/rx_fdat[10]
A8	fdat[19/]rx_fdat[19]	B8	fdat[16]/rx_fdat[16]	C8	fdat[14]/rx_fdat[14]
A9	fdat[22]/rx_fdat[22]	B9	fdat[20]/rx_fdat[20]	C9	fdat[17]/rx_fdat[17]
A10	fdat[26]/rx_fdat[26]	B10	fdat[24]/rx_fdat[24]	C10	fdat[21]/rx_fdat[21]
A11	fdat[29]/rx_fdat[29]	B11	fdat[27]/rx_fdat[27]	C11	fdat[25]/rx_fdat[25]
A12	txsel_l	B12	fdat[31]/rx_fdat[31]	C12	fdat[30]/rx_fdat[30]
A13	Gnd	B13	rxsel_l	C13	RESERVED
A14	Gnd	B14	RESERVED	C14	RESERVED
A15	clk	B15	fdat[32]/tx_fdat[0]	C15	fdat[33]/tx_fdat[1]
A16	fdat[34]/tx_fdat[2]	B16	fdat[36]/tx_fdat[4]	C16	fdat[38]/tx_fdat[6]
A17	fdat[37]/tx_fdat[5]	B17	fdat[39]/tx_fdat[7]	C17	fdat[42]/tx_fdat[10]
A18	fdat[41]/tx_fdat[9]	B18	fdat[43]/tx_fdat[11]	C18	fdat[46]/tx_fdat[14]
A19	fdat[44]/tx_fdat[12]	B19	fdat[47]/tx_fdat[15]	C19	fdat[49]/tx_fdat[17]
A20	fdat[48]/tx_fdat[16]	B20	fdat[50]/tx_fdat[18]	C20	fdat[53]/tx_fdat[21]
A21	fdat[51]/tx_fdat[19]	B21	fdat[54]/tx_fdat[22]	C21	fdat[57]/tx_fdat[25]
A22	fdat[55]/tx_fdat[23]	B22	fdat[58]/tx_fdat[26]	C22	fdat[61]/tx_fdat[29]
A23	fdat[59]/tx_fdat[27]	B23	fdat[62]/tx_fdat[30]	C23	RESERVED
A24	fdat[63]/tx_fdat[31]	B24	Gnd	C24	Vdd
A25	Gnd	B25	Vdd	C25	Gnd
A26	Gnd	B26	Gnd	C26	fps[0]

Table 4. Pin List (Continued)

Pin Number	Pin Name	Pin Number	Pin Name	Pin Number	Pin Name
D1	fbe_l[6]/tx_fbe[2]	E4	RESERVED	J1	cdat[4]
D2	eop/rx_eop	E23	RESERVED	J2	cdat[6]
D3	RESERVED	E24	fps[2]	J3	cint1_l/tx_eop
D4	Vdd	E25	rxfail	J4	Vdd
D5	RESERVED	E26	txctl_l	J23	Vdd
D6	fdat[3]/rx_fdat[3]	F1	cint6_l/tx_fps[2]	J24	txrdy7
D7	fdat[7]/rx_fdat[7]	F2	fbe_l[1]/rx_fbe[1]	J25	rxrdy2
D8	fdat[11]/rx_fdat[11]	F3	fbe_l[4]/tx_fbe[0]	J26	rxrdy4
D9	Vdd	F4	fbe_l[7]/tx_fbe[3]	K1	cdat[0]
D10	fdat[18]/rx_fdat[18]	F23	txasis /txerr	K2	cdat[2]
D11	fdat[23]/rx_fdat[23]	F24	rxkep	K3	cdat[5]
D12	fdat[28]/rx_fdat[28]	F25	rxctl_l	K4	cint0_l/cint_any_l
D13	RESERVED	F26	txrdy2	K23	rxrdy0
D14	Vdd	G1	cint3_l/RESERVED	K24	rxrdy3
D15	fdat[35]/tx_fdat[3]	G2	cint5_l/tx_fps[1]	K25	rxrdy6
D16	fdat[40]/tx_fdat[8]	G3	fbe_l[0]/rx_fbe[0]	K26	flct0
D17	fdat[45]/tx_fdat[13]	G4	fbe_l[3]/rx_fbe[3]	L1	crd_l
D18	Vdd	G23	rxabt	L2	crdy_l
D19	fdat[52]/tx_fdat[20]	G24	txrdy0	L3	cdat[1]
D20	fdat[56]/tx_fdat[24]	G25	txrdy3	L4	cdat[3]
D21	fdat[60]/tx_fdat[28]	G26	txrdy5	L23	rxrdy5
D22	RESERVED	H1	cdat[7]	L24	rxrdy7
D23	Vdd	H2	cint2_l/tx_sop	L25	flct1
D24	RESERVED	H3	cint4_l/tx_fps[0]	L26	flct3
D25	fps[1]/rx_fps[1]	H4	cint7_l/RESERVED	M1	cps[1]
D26	RESERVED	H23	txrdy1	M2	cps[2]
E1	fbe_l[2]/rx_fbe[2]	H24	txrdy4	M3	cs_l
E2	fbe_l[5]/tx_fbe[1]	H25	txrdy6	M4	cwr_l
E3	Vdd_clmp	H26	rxrdy1	M23	flct2

Table 4. Pin List (Continued)

Pin Number	Pin Name	Pin Number	Pin Name	Pin Number	Pin Name
M24	flct4	T3	mdio	W26	txd7[2]
M25	flct5	T4	col0/act0	Y1	rclk0
M26	flct6	T23	ten7/txd7[4]	Y2	dv0/rxd0[4]
N1	Gnd	T24	rerr7	Y3	rxd0[2]
N2	cadd[9]	T25	dv7/rxd7[4]	Y4	col1/act1
N3	cps[0]	T26	rxd7[1]	Y23	terr6/lnk6
N4	Vdd	U1	mdc	Y24	dv6/rxd6[4]
N23	flct7	U2	crs0/sd0	Y25	rxd6[1]
N24	reset_l	U3	txd0[2]	Y26	rxd6[3]
N25	tdo	U4	txd0[0]	AA1	rxd0[0]
N26	Gnd	U23	txd7[3]	AA2	rxd0[3]
P1	Gnd	U24	txd7[0]	AA3	txd1[3]
P2	cadd[8]	U25	terr7/lnk7	AA4	txd1[0]
P3	cadd[7]	U26	rclk7	AA23	txd6[2]
P4	cadd[6]	V1	txd0[3]	AA24	ten6/txd6[4]
P23	Vdd	V2	txd0[1]	AA25	rerr6
P24	tms	V3	ten0/txd0[4]	AA26	rclk6
P25	tdi	V4	Vdd	AB1	crs1/sd1
P26	Gnd	V23	Vdd	AB2	txd1[2]
R1	cadd[5]	V24	col7/act7	AB3	terr1/lnk1
R2	cadd[4]	V25	txd7[1]	AB4	RESERVED
R3	cadd[3]	V26	tclk7	AB23	RESERVED
R4	cadd[1]	W1	tclk0	AB24	txd6[3]
R23	rxd7[0]	W2	terr0/lnk0	AB25	txd6[0]
R24	rxd7[2]	W3	rerr0	AB26	tclk6
R25	rxd7[3]	W4	rxd0[1]	AC1	txd1[1]
R26	tck	W23	rxd6[0]	AC2	ten1/txd1[4]
T1	cadd[2]	W24	rxd6[2]	AC3	RESERVED
T2	cadd[0]	W25	crs7/sd7	AC4	Vdd

Table 4. Pin List (Continued)

Pin Number	Pin Name	Pin Number	Pin Name	Pin Number	Pin Name
AC5	RESERVED	AD8	terr2/lnk2	AE11	txd3[1]
AC6	rxd1[0]	AD9	dv2/rxd2[4]	AE12	terr3/lnk3
AC7	crs2/sd2	AD10	rxd2[2]	AE13	dv3/rxd3[4]
AC8	txd2[1]	AD11	txd3[3]	AE14	rxd3[0]
AC9	Vdd	AD12	ten3/txd3[4]	AE15	crs4/sd4
AC10	rxd2[0]	AD13	rerr3	AE16	txd4[1]
AC11	crs3/sd3	AD14	rxd3[1]	AE17	terr4/lnk4
AC12	txd3[0]	AD15	col4/act4	AE18	dv4/rxd4[4]
AC13	Vdd	AD16	txd4[0]	AE19	rxd4[3]
AC14	rxd3[2]	AD17	rerr4	AE20	txd5[3]
AC15	txd4[2]	AD18	rxd4[2]	AE21	ten5/txd5[4]
AC16	ten4/txd4[4]	AD19	col5/act5	AE22	rerr5
AC17	rxd4[1]	AD20	txd5[0]	AE23	rxd5[2]
AC18	Vdd	AD21	dv5/rxd5[4]	AE24	Gnd
AC19	txd5[1]	AD22	rxd5[1]	AE25	Vdd
AC20	terr5/lnk5	AD23	RESERVED	AE26	Gnd
AC21	rxd5[0]	AD24	Vdd	AF1	Gnd
AC22	RESERVED	AD25	Gnd	AF2	Gnd
AC23	Vdd	AD26	crs6/sd6	AF3	rclk1
AC24	RESERVED	AE1	Gnd	AF4	rxd1[1]
AC25	col6/act6	AE2	Vdd	AF5	col2/act2
AC26	txd6[1]	AE3	Gnd	AF6	txd2[0]
AD1	tclk1	AE4	dv1/rxd1[4]	AF7	tclk2
AD2	Gnd	AE5	rxd1[2]	AF8	rclk2
AD3	Vdd	AE6	txd2[3]	AF9	rxd2[3]
AD4	RESERVED	AE7	ten2/txd2[4]	AF10	txd3[2]
AD5	rerr1	AE8	rerr2	AF11	tclk3
AD6	rxd1[3]	AE9	rxd2[1]	AF12	rclk3
AD7	txd2[2]	AE10	col3/act3	AF13	Gnd
AF14	Gnd	AF19	rxd4[0]	AF24	rxd5[3]
AF15	rxd3[3]	AF20	crs5/sd5	AF25	Gnd
AF16	txd4[3]	AF21	txd5[2]	AF26	Gnd
AF17	tclk4	AF22	tclk5		
AF18	rclk4	AF23	rclk5		

3.0 Register Descriptions

This section describes the IXF440 registers. Each of the IXF440 ports includes an independent register set, enabling maximum flexibility.

The registers in each port are divided into two groups:

- Control and status registers (CSRs)
- Network statistics counters

3.1 Register Conventions

All the registers in the IXF440 are implemented in each of its eight ports, unless mentioned otherwise in the text.

In the register description tables throughout this chapter, the following abbreviations are used to indicate register access modes:

R	Readable only.
W	Writable only.
R/W	Readable and writable.

3.1.1 Access Rules

The following access rules must be followed when accessing registers:

- Unlisted addresses are reserved and must not be accessed.
- Reserved bits on registers must be written as “0” and are unpredictable on read.
- The configuration and the serial registers are writable only when the port is in stop state (as reported in the interrupt status register – INT_STT), following transmit and receive disable programming.
- Multibyte registers are ordered from low to high (the lower address points to the lower byte).

3.2 CSR Register

This section describes the IXF440 CSR register mapping.

3.2.1 Register Mapping

Table 5 lists each CSR register name, mnemonic, address offset, and the section that describes the register.

Table 5. CSR Register Mapping

Register Description	Mnemonic	I/O Address Offset	Section
Base Registers			3.2.2
Chip interrupt summary register[1]	CHIP_INT	00H	3.2.2.1
Interrupt status register	INT_STT	01H	3.2.2.2
Interrupt enable register	INT_EN	02H	3.2.2.3
Transmit status register	TXMT_STT	03H	3.2.2.4
Receive status register	RECV_STT	04H	3.2.2.5
Port control register	PORT_CTR	05H	3.2.2.6
Device identification number register1.	DEV_ID	06H	3.2.2.7
Revision identification number register1.	REV_ID	07H	3.2.2.8
Serial command register	SER_COM	08H	3.2.2.9
Configuration Registers			3.2.3
FIFO threshold register	FFO_TSHD	11H	3.2.3.1
IX Bus mode register	FFO_BUS	12H	3.2.3.2
Transmit parameters register	TX_PARAM	13H	3.2.3.3
Transmit error mode register	TX_ERR_MOD	14H	3.2.3.4
Transmit packet sending threshold and backoff limit register	TX_TSHD_BOFF	15H	3.2.3.5
Receive parameters register	RX_PARAM	16H	3.2.3.6
Receive filtering mode register	RX_FILT_MOD	17H	3.2.3.7
Transmit flow-control pause time register	PAUSE_TIME	18H – 19H	3.2.3.8
Maximum packet size register	PKT_MAX_SIZE	1AH – 1BH	3.2.3.9
InterPacket gap register	IPG	32H – 33H	3.2.3.10
Serial Registers			3.2.4
Serial mode register	SER_MOD	21H	3.2.4.1
Link status register	LNK_STT	22H	3.2.4.2
Physical address register	PHY_ADD	28H – 2DH	3.2.4.3

1. These registers are valid in Port 0 only.

3.2.2 Base Registers

The following sections describe the individual base registers.

3.2.2.1 Chip Interrupt Summary Register

This register handles the interrupt status of all the ports. A set bit indicates that the corresponding port has generated an interrupt, and several bits located in its Interrupt Status Register have been set. A bit will reset when the corresponding interrupts have been cleared. The interrupt summary register is valid only in port 0.

CHIP_INT

7	6	5	4	3	2	1	0
INT7	INT6	INT5	INT4	INT3	INT2	INT1	INT0

Bit Name	Bit Number	Bit Description
INT7	7	Port 7 interrupt
INT6	6	Port 6 interrupt
INT5	5	Port 5 interrupt
INT4	4	Port 4 interrupt
INT3	3	Port 3 interrupt
INT2	2	Port 2 interrupt
INT1	1	Port 1 interrupt
INT0	0	Port 0 interrupt
Access Rules		
Register access		R
Value after reset		00H

3.2.2.2 Interrupt Status Register

The interrupt status register handles information regarding events that cause an interrupt. A specific event sets a bit if it was programmed to generate an interrupt in the interrupt enable register (INT_EN).

Reading this register will reset all the bits, except for the TXER and RXEV bits. These bits are reset only when the corresponding status register is read.

INT_STT

7	6	5	4	3	2	1	0
	STOP	OVFC	LNCH	TXER	TXOK	RXEV	RXOK

Bit Name	Bit Number	Bit Description
—	7	RESERVED
STOP	6	Port stop. Indicates that transmit or receive path is in the stop state, following disable programming in the port control register. If both transmit and receive paths are programmed to be stopped, this bit is set when they both reach the stop state.
OVFC	5	Counter overflow. Indicates that at least one management counter has reached its highest possible value (and is stuck there). The overflowed counter needs to be reset to prevent further interrupt.
LNCH	4	Link change. Indicates that a link change has occurred while working in SYM mode. The actual link status is reported in the link status register.
TXER	3	Transmit error. A logical OR bit for all the bits in the transmit error status register.
TXOK	2	Transmit OK. Indicates successful packet transmission.
RXEV	1	Receive event. A logical OR of all the bits in the receive status register. This bit is set only after the packet is fully received by the chip.
RXOK	0	Receive OK. Indicates that a packet has been received without error. This bit is set only after the packet is fully transferred onto the chip.
Access Rules		
Register access		R
Value after reset		00H

3.2.2.3 Interrupt Enable Register

Each bit in this register enables the generation of an interrupt when the corresponding event occurs. The event causing the interrupt will be reported in the interrupt status register (INT_STT).

INT_EN

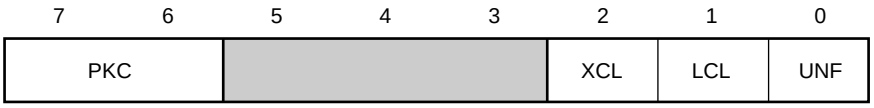
7	6	5	4	3	2	1	0
	STOPE	OVFCE	LNCHE	TXERE	TXOKE	RXEVE	RXOKE

Bit Name	Bit Number	Bit Description
—	7	RESERVED
STOPE	6	Port stop interrupt enable
OVFCE	5	Counter overflow interrupt enable
LNCHE	4	Link change interrupt enable
TXERE	3	Transmit error interrupt enable
TXOKE	2	Transmit OK interrupt enable
RXEVE	1	Receive event interrupt enable
RXOKE	0	Receive OK interrupt enable
Access Rules		
Register access		R/W
Value after reset		00H

3.2.2.4 Transmit Status Register

This register handles the causes for transmit stops. All bits, except PKC bits, are reset upon reading this register. The bits in this register are set only if packet transmission is programmed to be stopped following the corresponding error in the transmit error mode register (TX_ERR_MOD).

TXMT_STT



A5989-01

Bit Name	Bit Number	Bit Description
PKC	7:6	Packet count. Indicates the number of packets currently stored in the transmit FIFO. The packet count is incremented while loading the first byte of a packet and decremented following transmission of a packet. The packet count can get a maximum value of 2.
—	5:3	RESERVED
XCL	2	Excessive collision. Indicates that a packet was sent 16 times unsuccessfully due to consecutive collisions.
LCL	1	Late collision. Indicates that a collision has occurred after transmission of 64 bytes.
UNF	0	FIFO underflow. Indicates that data was not available in the transmit FIFO during packet transmission, after transmission of 64 bytes.
Access Rules		
Register access		R
Value after reset		00H

3.2.2.5 Receive Status Register

This register reports events that have occurred during packet reception. All bits are reset upon reading this register. The bits in this register are set only if the receive logic is programmed to pass packets with the corresponding event in the receive filtering mode register (RX_FILT_MOD). If a packet with multiple errors is to be passed and not all the corresponding bits in the receive filtering mode register (RX_FILT_MOD) are set, none of the bits in this register are set. The receive status is also appended to the end of the packet in the receive FIFO.

RECV_STT

7	6	5	4	3	2	1	0
FLC		MER	RTL	SRT	DRB	CRC	OVF

Bit Name	Bit Number	Bit Description
FLC	7	Flow control packet. Indicates that a flow-control packet was received.
—	6	RESERVED
MER	5	MII error. Indicates either that a symbol error was detected in the SYM mode or that an MII error signal was asserted in the MII mode during packet reception.
RTL	4	Too long packet. Indicates that a packet longer than the maximum allowable size has been received.
SRT	3	Short packet. Indicates that a packet shorter than 64 bytes has been received.
DRB	2	Alignment error. Indicates that a packet was received with CRC and framing errors.
CRC	1	CRC error. Indicates that a packet was received with a CRC error but without a framing error.
OVF	0	FIFO overflow. Indicates that there was not enough space available in the receive FIFO during packet reception.
Access Rules		
Register access		R
Value after reset		00H

3.2.2.6 Port Control Register

The port control register handles all the control bits of the port.

PORT_CTR

7	6	5	4	3	2	1	0
		TXSTT	TXEN	RXEN	PTRST	CTRST	CNRST

Bit Name	Bit Number	Bit Description
—	7:6	RESERVED
TXSTT	5	Transmit restart. When set, the chip restarts the transmission process after stopping due to a transmit error. The TXSTT is a trigger bit (does not require reset before setting).
TXEN	4	Transmit enable. When this bit is set, the port enters the working mode. When reset, the port completes transmission of the packet currently processed, and then stops. Stop state is reported in the STOP bit located in the interrupt status register (INT_STT).
RXEN	3	Receive enable. When this bit is set, the port enters the working mode. When reset, the port completes reception of the packet currently processed, and then stops. Stop state is reported in the STOP bit located in the interrupt status register (INT_STT).
PTRST	2	Port reset. When set, this bit causes the transmit and receive logic to reset.
CTRST	1	Control reset. When set, this bit resets all the registers, except PORT_CTR, to their default values. Transmit and receive logic are also reset.
CNRST	0	Counter reset. When set, this bit resets all the network statistic counters. The reset process takes 100 cycles. During this time, the counters and PORT_CTR must not be accessed. This bit must not be set together with the CTRST bit.
Access Rules		
Register access		R/W
Value after reset		00H

3.2.2.7 Device ID Register

The device ID register is valid only in port 0.

DEV_ID

7	6	5	4	3	2	1	0
DID							

Bit Name	Bit Number	Bit Description
DID	7:0	Device ID.
Access Rules		
Register access		R
Value after reset		01H

3.2.2.8 Revision ID Register

The revision ID register is valid only in port 0.

REV_ID

7	6	5	4	3	2	1	0
MRID				SRID			

Bit Name	Bit Number	Bit Description
MRID	7:4	Main revision ID. This number is incremented for subsequent IXF440 revisions.
SRID	3:0	Sub revision ID. This number is incremented for subsequent IXF440 steps within the current revision.
Access Rules		
Register access		R
Value after reset		11H

3.2.2.9 Serial Command Register

The serial command register handles the control of the serial interface.

SER_COM

7	6	5	4	3	2	1	0
		MDM	MDIO	MDC	BKP	FCT	

Bit Name	Bit Number	Bit Description
—	7:6	RESERVED
MDM	5	MII data mode. When set, the mdio signal is in the output mode. When reset, the mdio signal is in the input mode. This bit is effective only in port “0”.
MDIO	4	MII data IO. During write access to the MII management, the value written to this bit is driven onto the mdio signal. During read access, the mdio signal value is latched into this bit. This bit is effective only in port “0”.
MDC	3	MII management clock. The value written to this bit is driven onto the mdc signal. Each new written value must remain stable for at least 200 ns. This bit is effective only in port “0”.
BKP	2	Backpressure mode. When set, a transmission is generated upon reception of each packet to force a collision while the port is in the half-duplex mode.
FCT	1	Flow-control packet trigger. When set, this bit initiates a flow-control packet transmission to the remote node, as defined in the 802.3 IEEE Standard, while the port is in the full-duplex mode. This bit can be used only if the XON disable bit on transmit parameters register (TX_PARAM) is set.
—	0	RESERVED
Access Rules		
Register access		R/W
Value after reset		00H

3.2.3 Configuration Registers

The following sections describe the individual configuration registers.

3.2.3.1 FIFO Threshold Register

The FIFO threshold register handles the transmit and receive FIFO threshold.

FFO_TSHD

7	6	5	4	3	2	1	0
TTH				RTH			

Bit Name	Bit Number	Bit Description
TTH	7:4	This field defines the minimum amount of free space required in the transmit FIFO in order to assert the txrdy signal. The effective threshold in bytes is equal to 8 × (TTH + 1). A value of 0 is not allowed.
RTH	3:0	This field defines the minimum amount of data required in the receive FIFO in order to assert the rxrdy signal. The effective threshold in bytes is equal to 8 × (RTH + 1). A value of 0 is not allowed.
Access Rules		
Register access		R/W
Value after reset		77H

Register field value (RTH or TTH)	Effective threshold (bytes)	Register field value (RTH or TTH)	Effective threshold (bytes)
0	not allowed	8	72
1	16	9	80
2	24	10	88
3	32	11	96
4	40	12	104
5	48	13	112
6	56	14	120
7	64	15	128

3.2.3.2 IX Bus Mode Register

The IX Bus mode register controls the IX Bus mode of operation.

FFO_BUS

7	6	5	4	3	2	1	0
			SPLT		HRYD	BWID	BEND

Bit Name	Bit Number	Bit Description
—	7:5	RESERVED
SPLT	4	Split IX Bus mode. This bit is meaningful for MAC0 only. For MAC1 through MAC7, this bit is RESERVED. If cleared to 0: • And BWID (bit 1) is also 0, data is transmitted and received via fdat[31:0] (IX Bus [31:0]). • And BWID (bit 1) is 1, data is transmitted and received via fdat[63:0] (IX Bus [63:0]). If set to 1: Data is received via rx_fdat[31:0] (IX Bus [31:0]) and transmitted via tx_dat[31:0] (IX Bus [63:32]).
—	3	RESERVED
HRYD	2	Header ready disable. When set, the rxrdy signal will not be asserted when a packet header is in FIFO, but only according to FIFO threshold values.
BWID	1	Bus width. Defines the number of bits used on the IX Bus. When set, 64 bits are used. When reset, the lower 32 bits are used. Software must ensure that MAC0 FFO_BUS[4] (SPLT) is set to 0 if BWID is set.
BEND	0	Big or little endian mode. Defines the byte ordering mode on the IX Bus. When set, the port uses big endian mode. When reset, little endian mode is used.
Access Rules		
Register access		R/W
Value after reset		00H

3.2.3.3 Transmit Parameters Register

The transmit parameters register handles the control of the transmit serial interface.

TX_PARAM

7	6	5	4	3	2	1	0
			SPM	XOND	FLCE	CRCD	PADD

Bit Name	Bit Number	Bit Description
—	7:5	RESERVED
SPM	4	Single packet mode. When set, a packet is loaded in the transmit FIFO only after the previous packet was fully transmitted without any error.
XOND	3	XON disable. When set, a flow-control packet will not be sent upon flct{i} signal deassertion.
FLCE	2	Flow-control mode enable. When set, transmission is stopped in the full-duplex mode while receiving flow-control packets.
CRCD	1	CRC appending disable. When set, packets are transmitted without padding or CRC appending to the end of the packet. This field is ignored if the txasis signal is asserted during the start of packet loading.
PADD	0	Padding appending disable. When set, short packets are transmitted without the addition of bytes complementing their sizes to 64 bytes. When the CRCD bit is set, this bit is ignored. This field is ignored if the txasis signal is asserted during the start of packet loading.
Access Rules		
Register access		R/W
Value after reset		00H

3.2.3.4 Transmit Error Mode Register

The transmit error mode register controls the events that cause transmission stop. If, during packet transmission, an error occurs and the corresponding bit is set, the transmit FIFO is flushed, and transmission stops. The error is then reported in the transmit error status register (TERR_STT), and can generate an interrupt according to the interrupt enable register (INT_EN).

TX_ERR_MOD

7	6	5	4	3	2	1	0
					XCLS	LCLS	UNFS

Bit Name	Bit Number	Bit Description
—	7:3	RESERVED
XCLS	2	Stop transmission after excessive collisions. When set, packet retransmission stops after 16 attempts. When reset, packets will always be retransmitted following a collision until a successful transmission is achieved.
LCLS	1	Stop transmission after late collision. When set, transmission stops if a collision occurs after transmission of 64 bytes. When reset, the packet will be flushed from the transmit FIFO and the following packet will be transmitted.
UNFS	0	Stop transmission after FIFO underflow. When set, transmission stops if an underflow occurs after transmission of 64 bytes. When reset, the packet will be flushed from the transmit FIFO, and the following packet will be transmitted.
Access Rules		
Register access		R/W
Value after reset		00H

3.2.3.5 Transmit Threshold and Backoff Register

Packet transmission starts when the amount of data in the transmit FIFO is larger than or equal to this threshold, or the entire packet enters onto the FIFO. The effective serial transmit threshold in bytes is equal to 8 x TSD + 4. If the sum of the effective serial transmit threshold and the effective IX Bus transmit threshold (based upon FFO_TSHD<TTH>) is larger than 230 bytes, transmission can begin before the effective serial transmit threshold is reached in order to prevent deadlock. A value of 0 is not allowed.

TX_TSHD_BOFF

7	6	5	4	3	2	1	0
BKL				TSD			

Bit Name	Bit Number	Bit Description
BKL	7:4	Backoff limit. This is the maximum number used in the retransmission time algorithm following a collision. Lower numbers will cause faster retransmission times. According to Ethernet standard, this value must be programmed to be 10 (decimal).
TSD	3:0	Packet transmission threshold. Packet transmission starts when the amount of data in the transmit FIFO is larger than or equal to the threshold, or the entire packet enters onto the FIFO. The effective threshold in bytes is equal to 8 × TSD + 4. If the sum of effective serial threshold and the effective parallel threshold is larger than 230 bytes, the transmission can start before threshold is reached to prevent deadlock. A value of 0 is not allowed.
Access Rules		
Register access		R/W
Value after reset		A3H

Packet transmission threshold	Effective threshold (bytes)	Packet transmission threshold	Effective threshold (bytes)
0	not allowed	8	68
1	12	9	76
2	20	10	84
3	28	11	92
4	36	12	100
5	44	13	108
6	52	14	116
7	60	15	124

3.2.3.6 Receive Parameters Register

The receive parameters register defines the receive data flow on the IX Bus.

RX_PARAM

7	6	5	4	3	2	1	0
CRCR	HRPL	HDRS					

Bit Name	Bit Number	Bit Description
CRCR	7	CRC remove. When set, the last four bytes of the received packet will not be transferred onto the IX Bus. Packets shorter than 4 bytes will cause invalid data to appear on the IX Bus.
HRPL	6	Header replay. When set, packet header is transferred twice onto the IX Bus. If this bit is set, the RX_FILT_MOD<PCRC> bit should be reset.
HDRS	5:0	Header size. Header size is used for the header replay function and for rxrdy signal assertion (even if the header replay function is disabled). The header size is calculated in bytes as 4 × HDRS (valid values: 4 ≤ HDRS ≤ 48).
Access Rules		
Register access		R/W
Value after reset		10H

3.2.3.7 Receive Filtering Mode Register

When a packet with a specific event is programmed to be passed, the corresponding packets are not discarded and are transferred as regular packets. The status is reported in the receive status register (RECV_STT) and can generate an interrupt according to the interrupt enable register (INT_EN).

If a specific event is not programmed to enable reception of a packet, the corresponding packets are discarded from the receive FIFO. Failure will be reported onto the rxfail signal if the packet has already started to be transferred onto the IX Bus. If a packet with multiple errors is to be passed, all corresponding bits must be set.

RX_FILT_MOD

7	6	5	4	3	2	1	0
PFLC		PMER	PRTL	PSRT	PDRB	PCRC	POVF

Bit Name	Bit Number	Bit Description
PFLC	7	Pass flow-control packets.
—	6	RESERVED
PMER	5	Pass packets with MII error.
PRTL	4	Pass too long packets.
PSRT	3	Pass short packets.
PDRB	2	Pass alignment error packets.
PCRC	1	Pass CRC error packets.
POVF	0	Pass FIFO overflow.
Access Rules		
Register access		R/W
Value after reset		00H

3.2.3.8 Transmit Pause Time Register

The transmit pause time register handles the time field used in the flow-control packets.

PAUSE_TIME

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PSTM															

Bit Name	Bit Number	Bit Description
PSTM	15:0	Indicates the number of slot times during which the remote mode cannot send packets. This field is inserted into the transmitted flow-control packets.
Access Rules		
Register access		R/W
Value after reset		0H

3.2.3.9 Maximum Packet Size Register

This register controls the maximum allowed packet size.

PKT_MAX_SIZE

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MPS														

Bit Name	Bit Number	Bit Description
—	15	RESERVED
MPS	14:0	Packets received with a longer size are treated as too long packets.
Access Rules		
Register access		R/W
Value after reset		1518 (decimal)

3.2.3.10 InterPacket Gap Register

The InterPacket gap register controls the interpacket gap.

IPG

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IPS2								IPS1							

Bit Name	Bit Number	Bit Description
IPS2	15:8	This field contains the number of serial clocks for the second part of the interpacket gap. Even if a carrier is sensed on the serial line, the IXF440 continues to count time and transmit its packet, and thus forces a collision on the network. The interpacket gap value is in units of 4 bit times. Therefore, the default value (09H) corresponds to 36 bit times and matches the standards requirements. The minimum value is 02H.
IPS1	7:0	This field contains the number of serial clock cycles for the first part of the interpacket gap. If a carrier is sensed on the serial line during these clock cycles (excluding the first 40-bit time after transmission), the IXF440 defers its transmission and waits until the line is idle. The interpacket gap value is in units of 4 bit times. Therefore, the default value (0FH) corresponds to 60 bit times and matches the standards requirements. The minimum value is 0AH.
Access Rules		
Register access		R/W
Value after reset		090F (hexadecimal)

The minimum value for IPS1 and IPS2 combined should be 22 (decimal) while the maximum value for IPS1 and IPS2 combined should be 100 (decimal). Note that this means IPS1 and IPS2 may not both be their respective minimum values.

3.2.4 Serial Registers

The following sections describe the individual serial registers.

3.2.4.1 Serial Mode Register

The serial mode register controls the serial interface mode of work.

SER_MOD

7	6	5	4	3	2	1	0
		ILPK	ELPK	FDX		SCR	SYP

Bit Name	Bit Number	Bit Description
—	7:6	RESERVED
ILPK	5	Internal loopback mode. When set, the port is disconnected from the line and all the transmitted packets are sent back to the receive side. Packets are not transmitted onto the line (except in the SYM mode) and packets received from the line are rejected.
ELPK	4	External loopback mode. When set, packets are being received and transmitted simultaneously. This mode is usable only when the external logic is programmed to loopback packets.
FDX	3	Full-duplex mode. When set, packets are being received and transmitted simultaneously.
—	2	RESERVED
SCR	1	Scrambler mode. If a SYM-100Base-TX PHY is connected, this bit must be set, and if a SYM-100Base-FX PHY is connected, this bit must be reset. This bit is used only if the SYP bit is set.
SYP	0	MII/SYM port mode. This bit defines the MII/SYM port mode. If an MII PHY device is connected, this bit must be reset. If a SYM PHY device is connected, this bit must be set.
Access Rules		
Register access		R/W
Value after reset		00H

3.2.4.2 Link Status Register

The link status register indicates the link status

LNK_STT

7	6	5	4	3	2	1	0
							LNK100

Bit Name	Bit Number	Bit Description
—	7:1	RESERVED
LNK100	0	100Base-X link status. Set if the line link is OK while in the SYM mode.
Access Rules		
Register access		R

3.2.4.3 Physical Address Register

The physical address register contains the Ethernet physical address of the port. This address is inserted to transmitted flow-control packets in the source address field.

PHY_ADD

47 46 45 44 43 42 41 40 39 38 37 36 35 34 33 32 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
EAD

Bit Name	Bit Number	Bit Description
EAD	47:0	Port Ethernet address.
Access Rules		
Register access		R/W
Value after reset		0H

3.3 Network Statistic Counter Mapping

This section describes the IXF440 statistic counter mapping.

3.3.1 Register Mapping

Table 6 lists each network statistic IXF440 register name, mnemonic, and offset.

I/O Base address: 100H or 200H.

Table 6. Network Statistic Register Mapping (Sheet 1 of 3)

Register Description	Mnemonic	I/O Address Offset
Transmit Statistic Counters		
The number of unicast packets transmitted without any errors.	TX_UNI_OK_CNT	00H – 03H
The number of multicast packets that are not broadcast, transmitted without any errors.	TX_MLT_OK_CNT	04H – 07H
The number of broadcast packets transmitted without any errors.	TX_BRD_OK_CNT	08H – 0BH
The number of packets deferred upon the first transmit attempt due to a busy line.	TX_DEFER_CNT	0CH – 0FH
The total number of regular collision events occurring during transmission.	TX_COL_CNT	10H – 13H
The number of packets transmitted without any error following a single collision.	TX_SCOL_CNT	14H – 17H
The number of packets transmitted without any error following multiple collisions.	TX_MCOL_CNT	18H – 1BH
The number of packets that have experienced 16 consecutive collisions or more.	TX_XCOL_CNT	1CH – 1FH
The number of transmission abortions due to a collision occurring after transmission of packets that are 64 bytes in length.	TX_LCOL_CNT	20H – 23H
The number of transmitted packets, 64 bytes in length, including bad packets.	TX_PKT_64_CNT	24H – 27H
The number of transmitted packets, 65 to 127 bytes in length, including bad packets.	TX_PKT_65_CNT	28H – 2BH
The number of transmitted packets, 128 to 255 bytes in length, including bad packets.	TX_PKT_128_CNT	2CH – 2FH
The number of transmitted packets, 256 to 511 bytes in length, including bad packets.	TX_PKT_256_CNT	30H – 33H
The number of transmitted packets, 512 to 1023 bytes in length, including bad packets.	TX_PKT_512_CNT	34H – 37H
The number of transmitted packets, 1024 to 1518 bytes in length, including bad packets.	TX_PKT_1024_CNT	38H – 3BH
The number of transmitted packets with length larger than 1519 bytes, including bad packets.	TX_PKT_1519_CNT	3CH – 3FH
The number of correct transmitted flow-control packets.	TX_PAUSE_CNT	40H – 43H
The number of packets transmitted with an error due to transmit FIFO underflow or txerr signal assertion.	TX_ERR_CNT	44H – 47H
Transmit Byte Counters		
The number of bytes transmitted in good packets.	TX_OCT_OK_CNT	60H – 64H

Table 6. Network Statistic Register Mapping (Sheet 2 of 3)

Register Description	Mnemonic	I/O Address Offset
The number of bytes transmitted in packets with errors.	TX_OCT_BAD_CNT	68H – 6CH
Receive Byte Counters		
The number of bytes received in good packets.	RX_OCT_OK_CNT	70H – 74H
The number of bytes received in packets with errors.	RX_OCT_BAD_CNT	78H – 7CH
Receive Statistic Counters		
The number of frames received without SFD detection but with carrier assertion. This counter must be reset after moving to the SYM mode.	RX_RUNT_CNT	80H – 83H
The number of receive packets not fully accepted due to receive FIFO overflow.	RX_OVF_CNT	84H – 87H
The number of packets, less than 64 bytes in length, received without any error.	RX_SHORT_OK_CNT	88H – 8BH
The number of packets less than 64 bytes in length, received with CRC error.	RX_SHORT_CRC_CNT	8CH – 8FH
The number of unicast packets with lengths between 64 bytes and the maximum packet size, received without any errors.	RX_UNI_OK_CNT	90H – 93H
The number of multicast packets with lengths between 64 bytes and the maximum packet size, received without any errors.	RX_MLT_OK_CN	94H – 97H
The number of broadcast packets with lengths between 64 bytes and the maximum packet size, received without any errors.	RX_BRD_OK_CNT	98H – 9BH
The number of packets with lengths between 64 bytes and the maximum packet size, received with an integral number of bytes and a CRC error.	RX_NORM_CRC_CNT	9CH – 9FH
The number of packets with lengths between 64 bytes and the maximum packet size, received with a nonintegral number of bytes and a CRC error.	RX_NORM_ALI_CNT	A0H – A3H
The number of packets, larger than the maximum packet size, received without any error.	RX_LONG_OK_CNT	A4H – A7H
The number of packets, larger than the maximum packet size, received with a CRC error.	RX_LONG_CRC_CNT	A8H – ABH
The number of received packets, 64 bytes in length, including bad packets.	RX_PKT_64_CNT	ACH – AFH
The number of received packets, 65 to 127 bytes in length, including bad packets.	RX_PKT_65_CNT	B0H – B3H
The number of received packets, 128 to 255 bytes in length, including bad packets.	RX_PKT_128_CNT	B4H – B7H
The number of received packets, 256 to 511 bytes in length, including bad packets.	RX_PKT_256_CNT	B8H – BBH
The number of received packets, 512 to 1023 bytes in length, including bad packets.	RX_PKT_512_CNT	BCH – BFH
The number of received packets, 1024 to 1518 bytes in length, including bad packets.	RX_PKT_1024_CNT	C0H – C3H
The number of received packets, with lengths between 1519 bytes and the maximum packet size (programmable value), including bad packets.	RX_PKT_1519_CNT	C4H – C7H

Table 6. Network Statistic Register Mapping (Sheet 3 of 3)

Register Description	Mnemonic	I/O Address Offset
The number of correct received flow-control packets.	RX_PAUSE_CNT	C8H – CBH
The number of false carrier events detected.	RX_FALS_CRS_CNT	CCH – CFH
The number of received packets during which PHY symbol errors were detected.	RX_SYM_ERR_CNT	D0H – D3H

3.3.2 Network Statistic Counters Access Rules

The network statistic counters access rules are as follows:

- The counters can be accessed with a base address equal to 100H or 200H. Accessing the counters with the 100H base address causes them to reset. Accessing them with the 200H will not reset the counter.
- Any of the counter bytes can be read without reading the other bytes. Only the accessed bytes are reset (if accessed with the 100H base address).
- The counters must be read from the lower to the upper bytes, in consecutive accesses.
- The statistic counters are not writable.

Note: Counter overflow causes it to remain fixed at the highest value it has reached.

Receive statistic counters are updated even upon receive FIFO overflow.

All the byte counters take CRC bytes into account, but exclude the framing bits in the packets.

3.4 Access Sequences

This section describes the initialization, mode change, and interrupt handling sequences for the IXF440.

3.4.1 Initialization Sequence

Each IXF440 port must be initialized according to the following sequence:

1. Disable the port and reset it by writing:
   06h to the port control register (PORT_CTR), and then
   01h to the same register
2. Initialize the port by writing to the relevant configuration and serial registers.
3. Enable port operation by writing a value of 18h to the port control register (PORT_CTR).

3.4.2 Mode Change Sequence

In order to change the IXF440 working mode without impacting packet transfer, the following sequence must be used:

1. Disable the port by writing a value of 00h to the port control register (PORT_CTR).

2. Wait until the port enters the stop state. The stop state entry may generate an interrupt if enabled, and is reported by the stop bit in the interrupt status register (INT_STT<STOP>).
3. Change the configuration and serial register values.
4. Enable port operation by writing a value of 18h to the port control register (PORT_CTR).

Note: The IX Bus mode register (FFO_BUS) can only be updated by using the initialization sequence, which will cause an interruption of packet transfer.

3.4.3 Interrupt Handling Sequence

Following interrupt, the subsequent sequence must be used to reset the interrupt:

1. If all ports are sharing the same interrupt line, read the chip interrupt summary register (CHIP_INT), and perform steps 2 to 4 for each port that generates an interrupt.
2. Read the port interrupt status register (INT_STT) and act according to the interrupt cause.
3. If the receive event bit is set (INT_STT<RXEV>), read the receive status register (RECV_STT) and act according to the interrupt cause.
4. If the transmit error bit is set (INT_STT<TXER>), read the transmit status register (TXMT_STT) and act according to the interrupt cause. To resume transmission, write a value of 38h to the port control register (PORT_CTR).

4.0 FIFO Interface Operation

This chapter describes the IXF440 FIFO interface operation, including the transmission and reception flows.

4.1 FIFO Interface

The IXF440 uses a generic bus interface for data transfer to and from its FIFOs. The data bus is 64 bits wide but can be programmed to use only the lower 32-bits (FFO_BUS<BWID>). Big and little endian byte ordering are both supported on 32-bit boundaries (FFO_BUS<BEND>). The different FIFOs are accessed according to port selection signals (fps[2:0]) as well as transmit or receive enabling signals (txsel_l, rxsel_l). Data transfer is synchronized to the main clock (clk), and new data may be sent or received on each clock cycle. Each FIFO has a dedicated signal for each direction (txrdy, rxrdy), reporting if it is ready for data transfer according to predetermined thresholds (FFO_TSHD<TTH,RTH>). The burst size should be shorter than or equal to the effective threshold. The amount of data transferred during a FIFO access may be dynamically changed from one access to the other. On receive, if the FFO_BUS<HRYD> bit is reset, each first burst of a packet should be shorter than or equal to the header size.

4.1.1 Byte Ordering on IX Bus

On the IX Bus, bytes are ordered according to the endian mode (FFO_BUS<BEND>) and the bus width (FFO_BUS<BWID>). Table 7 and Table 8 show the different options for little endian mode. Table 9 and Table 10 show the different options for big endian mode.

Table 7. Little Endian, 64-Bit Bus (BEND=0, BWID=1)

63			32	31			0
8th byte	7th byte	6th byte	5th byte	4th byte	3rd byte	2nd byte	1st byte

Table 8. Little Endian, 32-Bit Bus (BEND=0, BWID=0)

31			0*
4th byte	3rd byte	2nd byte	1st byte

Table 9. Big Endian, 64-Bit Bus (BEND=1, BWID=1)

63			32	31			0
5th byte	6th byte	7th byte	8th byte	1st byte	2nd byte	3rd byte	4th byte

Table 10. Big Endian, 32-Bit Bus (BEND=1, BWID=0)

31			0*
1st byte	2nd byte	3rd byte	4th byte

4.1.2 FIFO Status Signaling

The IXF440 reports the status of each FIFO through dedicated signals. Each transmit FIFO has a txrdy signal indicating that there is enough free space to load new data. Each receive FIFO has a rxrdy signal indicating that there is enough data to be transferred onto the IX Bus. The txrdy signals are driven by the IXF440 when the txctl_l signal is asserted, and the rxrdy signals are enabled by the rxclt_l signal.The txrdy signal of a specific port is asserted when the txctl_l signal is asserted and the specific port is selected (fps[2:0] or, alternatively, tx_fps[2:0] in Split). The same applies for the rxrdy signal of a specific port, which is asserted with rxctl_l assertion and the specific port selection (fps[2:0]).

4.2 Packet Transmission

The following sections describe the packet transmission policy.

4.2.1 Packet Loading

The IXF440 loads packets from the IX Bus into the transmit FIFO during burst accesses. In order to guarantee a minimal amount of data transfer, the transmit FIFO txrdy signal reports minimal space availability according to a programmable threshold (FFO_TSHD<TTH>).

When a new packet is loaded in the FIFO, the first cycle of the first burst must be signalled with sop signal assertion. If the txasis signal is asserted together with sop, the packet will be sent onto the network without padding or CRC addition. At the end of a packet load, the last data must be signalled with the assertion of the eop signal in the last cycle of the last burst. If the txerr signal is asserted together with eop while sending the packet onto the network, the MII error signal terr will be asserted and the CRC will be damaged if it was requested to be appended by the IXF440.

Up to two packets can be loaded in the transmit FIFO, although the IXF440 may be programmed to handle only a single packet at a time (TX_PARAM<SPM>).

Byte masking signals (fbe_l[7:0]) may be used to load selective bytes. They can be used during packet transfer to load packet segments on byte boundaries and for loading the exact number of bytes at the end of a packet. Valid bytes may start at any byte boundary, while all valid bytes in a given cycle need to be contiguous.

For example, a packet may be built up from the following buffers, with each one being transferred in a different burst:

Buffer 1:

B3	B2	B1	X	X	X	X	X
B11	B10	B9	B8	B7	B6	B5	B4
X	X	X	X	X	X	X	B12

Buffer 2:

X	X	X	B14	B13	X	X	X

Buffer 3:

X	B19	B18	B17	B16	B15	X	X
X	X	B21	B20	X	X	X	X

4.2.2 Network Transmission

The IXF440 will start to transmit the packet if there is enough data in the transmit FIFO according to the programmable transmission threshold (TX_TSHD_BOFF<TSD>), or if the full packet is loaded into the transmit FIFO.

During the first phase of transmission, which includes the first 64 bytes, the data is kept on the transmit FIFO to ensure retransmission of the packet in case of collision without needing to reload the packet onto the transmit FIFO.

If the packet is transmitted without error, the next packet is continuously loaded into the FIFO and transmitted onto the network, with a minimum gap between them. Normal packet transmission may also generate an interrupt, if programmed to do so.

During packet transmission, the following errors may occur:

- Excessive collision
  A packet has collided during 16 consecutive attempts.
- Late collision
  A collision occurred after transmission of 64 bytes.
- FIFO underflow
  Data was not ready in the FIFO during packet transmission.

The IXF440 can be programmed to stop or to continue working when a transmit error occurs. In the error continuing mode, the IX Bus behaves the same way with or without an error. In both modes, according to the error, the appropriate statistic counters will be updated.

4.2.3 Excessive Collisions

By default, the IXF440 automatically retransmits a packet until it is successfully transmitted, even after excessive collisions. If the IXF440 is programmed to stop after excessive collisions (TX_ERR_MOD<XCLS>), no more packets will be transmitted.

4.2.4 Late Collision

In standard networks, late collisions are not expected to occur. In cases of late collisions, the IXF440 aborts the current transmission and continues with the next packet. If the IXF440 is programmed to stop after late collision (TX_ERR_MOD<LCLS>), it will flush the transmit FIFO and no more packets will be transmitted.

4.2.5 FIFO Underflow

Following an underflow, packet transmission is truncated, an MII error is generated (terr signal assertion), and a bad CRC is appended to the packet. If the underflow occurs before the transmission of 60 bytes, padding is added first and then the bad CRC.

If the underflow occurs during transmission of the first 64 bytes of a packet, the IXF440 will automatically retransmit the packet. If the underflow occurs later, by default the IXF440 will continue with the next packet. If the IXF440 is programmed to stop after underflow (TX_ERR_MOD<UNFS>), the transmit FIFO will be flushed and no more packets will be transmitted.

4.2.6 Stopping Mode on Transmission Errors

If an error is programmed to cause the transmit process to stop, the IXF440 will flush all the data found in the transmit FIFO. The number of packets flushed from the transmit FIFO and the transmission stop cause are reported in the transmit status register (TERR_STT). The subsequent packets will be loaded only after the FIFO is restarted (PORT_CTR<TXSTT>).

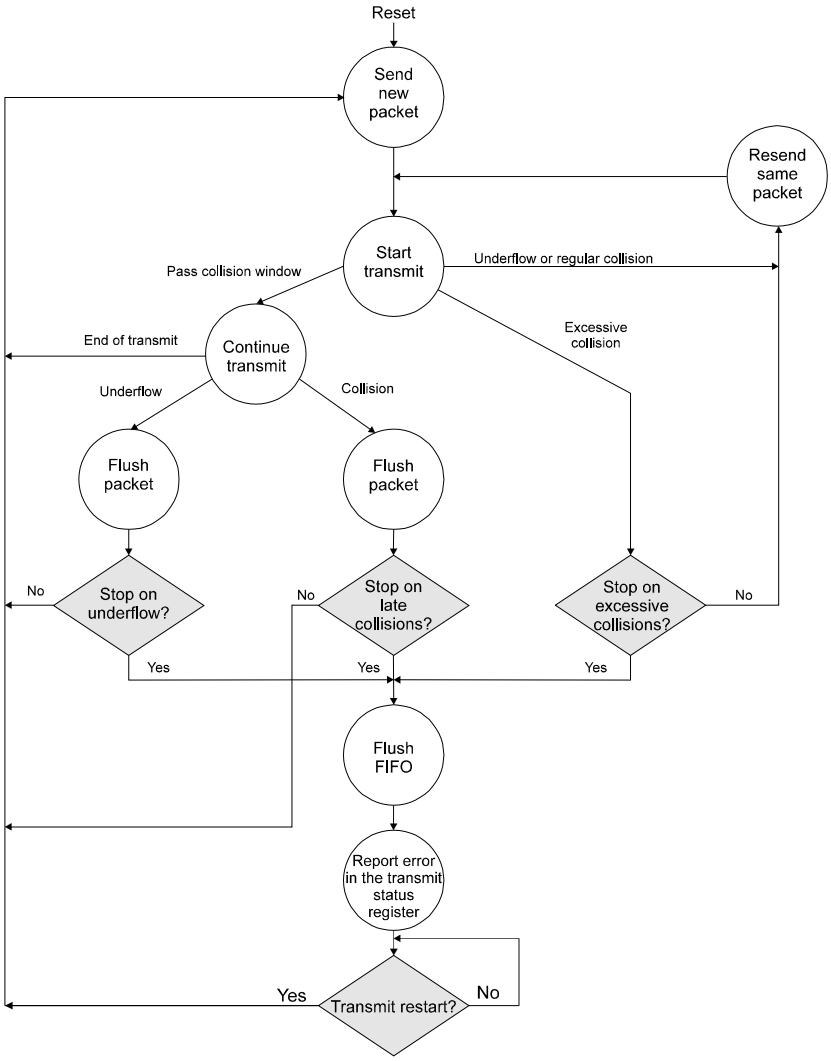
Transmission errors may also generate an interrupt, if programmed to do so.

In the single-packet mode (TX_PARAM<SPM>), new packets are loaded into the transmit FIFO only after successful transmit completion of the previous packet. In case of an error, only the erroneous packet is flushed from the FIFO.

4.2.7 Transmit Flow Diagram

Figure 2 shows the transmit flow diagram.

Figure 2. Transmit Flow Diagram



4.3 Packet Reception

The following sections describe the packet reception policy.

4.3.1 Packet Storing

Packets received from the network are loaded into the receive FIFO. Received packets are transferred from the receive FIFO onto the IX Bus during burst accesses. To indicate the minimal amount of data available in the receive FIFO, the rxrdy signal is asserted according to a programmable threshold (FFO_TSHD<RTH>). If the end of a packet is loaded onto the FIFO, the rxrdy signal is asserted even if the amount of data available is below the threshold value.

The fbe_l[7:0] signals are used to report which bytes driven onto the bus are valid. For example, they indicate which bytes are valid in the last bus transfer of a packet.

The first packet data is signalled with the sop signal, while the last data is signalled with the eop signal. Following the last data transfer of a received packet, a field describing the packet status is driven on the data bus, reporting the status of the packet and its length. The packet status is driven on the IX Bus according to the description in Table 11. In 64-bit IX Bus mode, the byte enables for the status word will report all eight data bytes as valid (fbe_l<7:0>=00h). In 32-bit IX Bus mode, the lower four bytes are valid (fbe_l<7:0>=f0h). Further reads from the receive FIFO during the same access are ignored to prevent transfer of the next packet in the same burst. The byte enables for these additional read transactions will report that all bytes contain invalid data (fbe_l<7:0>=ffh). If the IXF440 is programmed to remove CRC (RX_PARAM<CRCR>), the last four bytes of the packet will not be transferred on the IX Bus, and eop will be asserted on the packet's last data word.

At a given time, multiple packets may be loaded in the receive FIFO.

The packet status is appended to any packet completely transferred onto the IX Bus, and is driven onto the IX Bus in the access following the last byte transfer. If the last byte was transferred on the last cycle of the burst, then the rxrdy signal is asserted again to indicate that the packet status was not transferred. The packet length indicates the number of bytes received in this packet on the serial line, independent of the packet transfer on the IX Bus. For the 64-bit bus mode (FFO_BUS<BWID>), the packet status is passed on twice on the FBUS: on the 32 low bits and on the 32 high bits. The status will always be driven as little endian data as described in Table 7 and Table 8. During status transfer, the value of the fbe_l<7:0> signals can be ignored, as the full status word is always valid.

Table 11. IX Bus Receive Packet Status

Bit Name	Bit Number	Bit Description
LEN	31:16	Packet length
—	15:11	RESERVED
MLT	10	Multicast packet
BRD	9	Broadcast packet
ROK	8	Receive OK
FLW	7	Flow-control packet
—	6	RESERVED
MER	5	MII error
RTL	4	Too long packet
RNT	3	Runt packet
DRB	2	Alignment error
CRC	1	CRC error
OVF	0	Receive FIFO overflow (if set, LEN field is not valid)

4.3.2 Header Preprocessing

The IXF440 supports the ability to process the packet header in several ways. The header size is programmable (RX_PARAM<HDRS>) and may be changed according to the required processing (for example, MAC header, VLAN header, or Layer3 header).

When header ready mode is enabled (FFO_BUS<HRYD>=0) and a packet header has been fully loaded into the receive FIFO, the IXF440 will assert the rxrdy signal, even if the header size is smaller than the programmed receive threshold. In this instance, the first burst of a packet must be shorter than or equal to the header size.

The packet header may also be read from the receive FIFO for processing without removing it from the FIFO. If the IXF440 is programmed to work in the header replay mode (RX_PARAM<HRPL>), the packet header will be transferred twice onto the IX Bus: the first time for header processing and the second time with the packet transfer.

4.3.3 Packet Segmentation

The IXF440 supports receive packet segmentation on any byte boundaries. When the rxkep signal is asserted on the last cycle of a burst, the last data transfer of that burst will be reissued on the next burst.

The rxkep signal is ignored when it is asserted in one of the following cases: nonvalid data, last data of the packet, or last data of the header on header replay mode (RX_PARAM<HRPL>).

In the example, rxkep is asserted on the last cycle of a three octal word burst from the IXF440, causing the third octal word to be retained in the receive FIFO. During the next receive burst, this same octal word will be driven as the first data word of the burst. Any masking of data bytes to the buffers is performed by the host. In the example, the host places bytes 1–19 in the first buffer (a result of the first burst) and bytes 20–28 in the second buffer (a result of the second burst).

Buffer 1:

B8	B7	B6	B5	B4	B3	B2	B1
B16	B15	B14	B13	B12	B11	B10	B9
X	X	X	X	X	B19	B18	B17

Buffer 2:

B24	B23	B22	B21	B20	X	X	X	– replayed
X	X	X	X	B28	B27	B26	B25	

4.3.4 Packet Abortion

During the transfer of a received packet onto the IX Bus, the IXF440 supports the ability to prevent any further transfer of this packet. At any time during packet reception, the packet may be dynamically discarded from the receive FIFO by asserting the rxabt signal during packet reading. Any subsequent packet loaded into the receive FIFO is not affected by rxabt assertion. The next FIFO access will be the next packet.

4.3.5 Network Reception

A packet received from the network is loaded into the receive FIFO. If the packet is received without any error, it is transferred to the IX Bus. If an error occurs during reception, the packet is handled according onto the programming.

The IXF440 may be programmed to work in two modes: reject the erroneous packet or accept it (RX_FILT_MOD). In both modes, according to the error, the appropriate statistic counters will be updated, even if the packet was rejected due to packet error or FIFO overflow. Even in case of errors, any following packets continue to be accepted and loaded to the receive FIFO.

The following events are considered as reception errors:

- FIFO overflow
- CRC error
- Alignment error
- Short packet
- Too long packet
- MII error

4.3.6 Rejecting Mode on Reception Errors

If a packet with a reception error is programmed to be rejected, the IXF440 discards the packet from the receive FIFO without affecting previous packets that may still be in the receive FIFO. If the packet had not yet started to be transferred on the IX Bus, it will be discarded without affecting IX Bus activity. If the packet had already started to be transferred onto the IX Bus, or rxrdy was already asserted, the rxfail signal will be asserted on the next FIFO access, indicating that the currently transferred packet was discarded from the receive FIFO. Packet status will not be driven for such a packet.

Runt packets that may be received due to collision on the network can be filtered by the IXF440 by programming the receive threshold to be greater than 64 bytes.

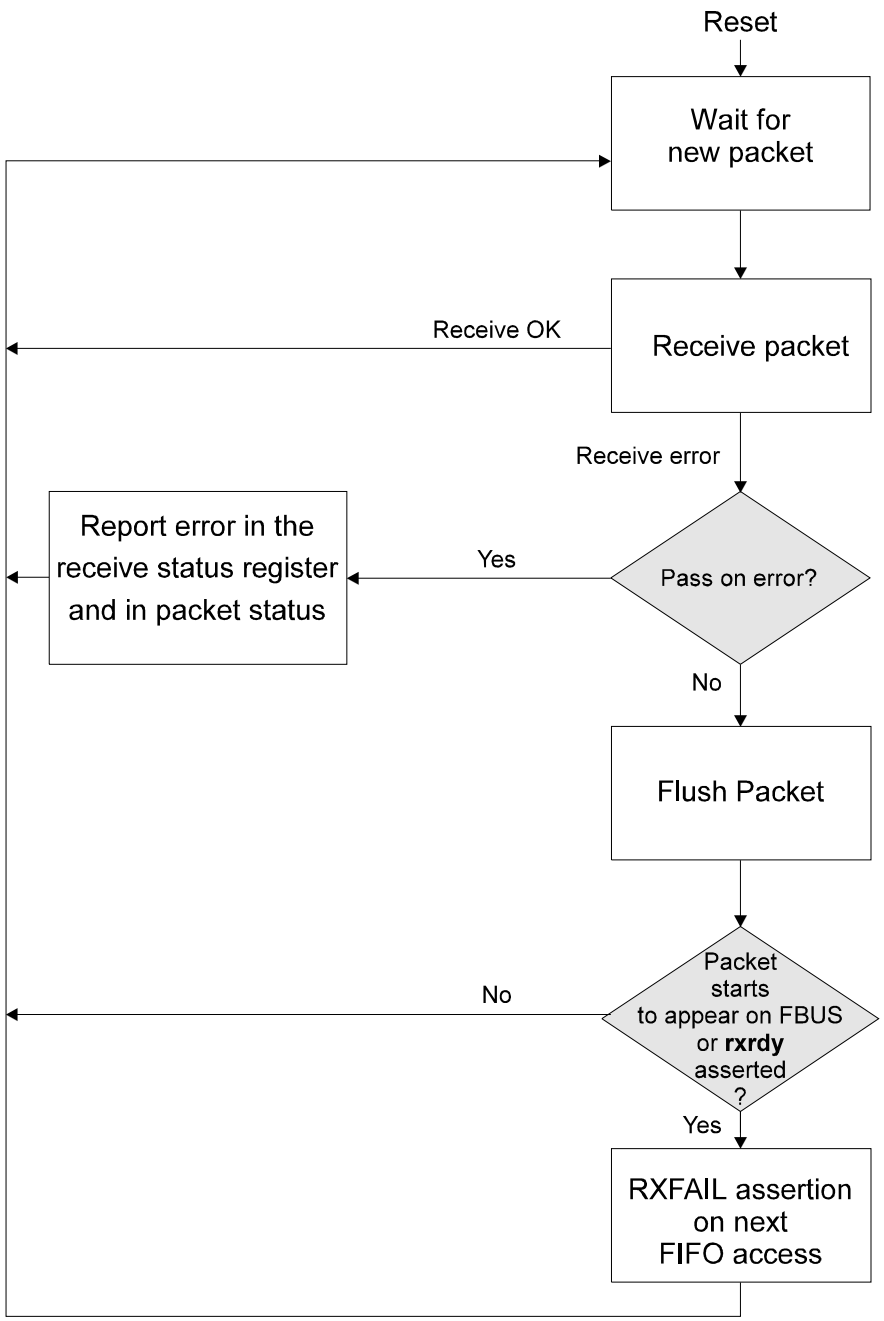
4.3.7 Accepting Mode on Reception Errors

If a packet with a specific event is programmed to be accepted, it is transferred to the IX Bus as a regular packet and the event type is reported in the packet status appended to the end of the packet. Such a packet may also generate an interrupt, if programmed to do so. The cause of the interrupt is reported in the receive status register (RECV_STT).

4.3.8 Receive Flow Diagram

Figure 3 shows the receive flow diagram.

Figure 3. Receive Flow Diagram



5.0 CPU Interface Operation

The following sections describe the CPU interface operation.

5.1 CPU Interface

The IXF440 has a dedicated port for a CPU interface, enabling access to the different registers without interfering with packet transfer through the FIFOs. The CPU interface is generic and supports a wide range of standard controllers. Each of the eight IXF440 ports has its own independent registers. Each of the port registers is accessible through an 8-bit data bus and a 10-bit address bus. A specific port is addressed by using the port select signals (cps[2:0]), which may be considered a part of the address bus. Each port has a dedicated interrupt signal (cint_l) to report special events to the CPU.

Each control and status register is 1 byte wide and accessible through a single CPU access. Network statistic counters are multibyte wide and require multiple CPU accesses to be fully read.

5.2 Network Management

The IXF440 includes network statistic counters defined by Ethernet SNMP MIB and RMON MIB standards. Each event counter is 32 bits wide and each byte counter is 40 bits wide.

To assemble each counter value, its bytes must be read from the lower to the upper addresses.

Partial byte reading is also possible. If the exact counter value is not required, the lower counter byte may not be read. If the counter is read often, the upper byte will always remain null and may not be read.

Each counter is accessible through two different addresses. One address will cause the read byte to reset, while the other will not. When a counter overflows, it remains stuck on the highest value it has reached, and can generate an interrupt if programmed accordingly.

Receive statistic counters are updated according to analysis of the received packet, while ignoring the IXF440 filtering mode or the receive FIFO status. When the port is in the disable mode, the counters are not updated.

5.2.1 SNMP MIB Support

The IXF440 supports Ethernet MIB according to Table 12.

Table 12. SNMP MIB to IXF440 Counters Mapping

SNMP MAC Counters	IXF440 Statistic Counters
FramesTransmittedOK	TX_UNI_OK_CNT + TX_MLT_OK_CNT + TX_BRD_OK_CNT
SingleCollisionFrames	TX_SCOL_CNT
MultipleCollisionFrames	TX_MCOL_CNT
FramesReceivedOK	RX_UNI_OK_CNT + RX_MLT_OK_CNT + RX_BRD_OK_CNT
FramesCheckSequenceErrors	RX_NORM_CRC_CNT
AlignmentErrors	RX_NORM_ALI_CNT
OctetsTransmittedOK	TX_OCT_OK_CNT
FramesWithDeferredXmissions	TX_DEFER_CNT
LateCollisions	TX_LCOL_CNT
FramesAbortedDueToXSColls	TX_XCOL_CNT
OctetsReceivedOK	RX_OCT_OK_CNT
FrameTooLongErrors	RX_LONG_OK_CNT + RX_LONG_CRC_CNT
MulticastFramesXmittedOK	TX_MLT_OK_CNT
BroadcastFramesReceivedOK	RX_BRD_OK_CNT
MulticastFramesReceivedOK	RX_MLT_OK_CNT
BroadcastFramesXmittedOK	TX_BRD_OK_CNT
PauseFramesTransmitted	TX_PAUSE_CNT
PauseFramesReceived	RX_PAUSE_CNT

5.2.2 RMON Statistic Group Support

Table 13 describes the IXF440 support of the RMON statistic group. If packets are loaded into the IXF440 to be transmitted as bad packets, they must be counted in the error counters, too.

Table 13. RMON Statistics to IXF440 Counters Mapping

RMON Statistic Counters	IXF440 Statistic Counters
etherStatsDropEvents	RX_OVF_CNT
etherStatsOctets	TX_OCT_OK_CNT + TX_OCT_BAD_CNT + RX_OCT_OK_CNT + RX_OCT_BAD_CNT
etherStatsPkts	TX_UNI_OK_CNT + TX_LCOL_CNT + RX_UNI_OK_CNT + etherStatsBroadcastPkts + etherStatsMulticastPkts + etherStatsCRCAlignErrors + etherStatsUndersizePkts + etherStatsFragments + etherStatsOversizePkts + etherStatsJabber
etherStatsBroadcastPkts	TX_BRD_OK_CNT + RX_BRD_OK_CNT
etherStatsMulticastPkts	TX_MLT_OK_CNT + RX_MLT_OK_CNT
etherStatsCRCAlignErrors	RX_NORM_ALI_CNT + RX_NORM_CRC_CNT + TX_ERR_CNT
etherStatsUndersizePkts	RX_SHORT_OK_CNT
etherStatsOversizePkts	RX_LONG_OK_CNT
etherStatsFragments	RX_SHORT_CRC_CNT + TX_COL_CNT
etherStatsJabber	RX_LONG_CRC_CNT
etherStatsCollisions	RX_RUNT_CNT + RX_SHORT_CRC_CNT + TX_COL_CNT + TX_LCOL_CNT
etherStatsPkts64Octets	TX_PKT_64_CNT + RX_PKT_64_CNT
etherStatsPkts65to127Octets	TX_PKT_65_CNT + RX_PKT_65_CNT
etherStatsPkts128to255Octets	TX_PKT_128_CNT + RX_PKT_128_CNT
etherStatsPkts256to511Octets	TX_PKT_256_CNT + RX_PKT_256_CNT
etherStatsPkts512to1023Octets	TX_PKT_512_CNT + RX_PKT_512_CNT
etherStatsPkts1024to1518Octets	TX_PKT_1024_CNT + RX_PKT_1024_CNT
etherStatsDropEvents	RX_OVF_CNT
etherStatsOctets	TX_OCT_OK_CNT + TX_OCT_BAD_CNT + RX_OCT_OK_CNT + RX_OCT_BAD_CNT

5.2.3 RMON Host Group Support

Table 14 describes the IXF440 support of the RMON host group when a single node is connected to a port. If packets are loaded into the IXF440 to be transmitted as bad packets, they must be counted in the error counters, too.

Table 14. RMON Host to IXF440 Counters Mapping

RMON Host Counters	IXF440 Statistic Counters
hostInPkts	TX_UNI_OK_CNT + TX_MLT_OK_CNT + TX_BRD_OK_CNT
hostOutPkts	RX_UNI_OK_CNT + RX_MLT_OK_CNT + RX_BRD_OK_CNT+ RX_NORM_ALI_CNT + RX_NORM_CRC_CNT + RX_SHORT_OK_CNT + RX_SHORT_CRC_CNT + RX_LONG_OK_CNT + RX_LONG_CRC_CNT + TX_COL_CNT + TX_LCOL_CNT
hostInOctets	TX_OCT_OK_CNT
hostOutOctets	RX_OCT_OK_CNT + RX_OCT_BAD_CNT
hostOutErrors	RX_NORM_ALI_CNT + RX_NORM_CRC_CNT + RX_SHORT_OK_CNT + RX_SHORT_CRC_CNT + RX_LONG_OK_CNT + RX_LONG_CRC_CNT + TX_COL_CNT + TX_LCOL_CNT
hostOutBroadcastPkts	RX_BRD_OK_CNT
hostOutMulticastPkts	RX_MLT_OK_CNT

6.0 Network Interface Operation

This chapter describes the MII/SYM port operation. It also describes media access control (MAC), flow control, full-duplex, and loopback operations. The IXF440 supports full implementation of the MAC sublayer according to the IEEE 802.3 Standard.

6.1 Operating Modes

Each of the IXF440 eight ports supports MII or SYM interfaces.

In the MII mode, the MII port provides a standard and simple interconnection between the MAC sublayer and the PHY layer. In this mode, the IXF440 can be used with any device with an MII interface that implements the 100BASE-TX, 100BASE-FX, 100BASE-T4, or 10BASE-T standards. Through the MII interface, PHYs can also be connected beyond a connector.

The MII interface comprises the following characteristics:

- Supports both 100 Mbps and 10 Mbps data rates
- Includes data and delimiters that are synchronous to clock references
- Provides independent, 4-bit transmit and receive data paths
- Utilizes TTL signal levels
- Provides a simple management interface
- Provides the capability to drive a limited length of shielded cable

In the SYM mode, the PCS encoding/decoding is done by the IXF440, which enables connection of simple FDDI PHY devices implementing the 100BASE-TX or 100BASE-FX standards. For 100BASE-TX connections, the IXF440 also implements scrambler and descrambler functions according to the ANSI TP-PMD standard. The SYM port is multiplexed together with the MII port.

6.2 MII Port Interface

In the MII mode (SER_MOD<SYP>=0), the MII/SYM port implements the IEEE 802.3 Standard MII interface. Table 15 describes the MII port signal names as they refer to the appropriate IEEE 802.3 signal names.

The MII management signals (mdc and mdio) are common to all eight ports.

Table 15 describes the MII port signals versus standard signals.

Table 15. MII Port Signals versus Standard Signals

MII Signals	IEEE 802.3 Signals	Purpose
tclk{i}	tx_clk	Transmit clock, synchronizes all transmit signals (ten{i}, txd{i}[3:0], terr{i}). In the 100 Mbps data rate, operates at 25 MHz. In the 10 Mbps data rate, operates at 2.5 MHz.
rclk{i}	rx_clk	Receive clock, synchronizes all receive signals (dv{i}, rxd{i}[3:0], rerr{i}). In the 100 Mbps data rate, operates at 25 MHz. In the 10 Mbps data rate, operates at 2.5 MHz.
ten{i}	tx_en	Transmit enable, asserted by the MAC sublayer when the first transmit preamble nibble is driven over the MII. It remains asserted for the remainder of the frame, up to the last CRC nibble.
txd{i}[3:0]	txd[3:0]	These lines provide transmit data, driving a nibble on each tclk{i} cycle when ten{i} is asserted.
terr{i}	tx_err	Transmit error, asserted by the MAC layer to generate a coding error on the nibble currently being transferred over txd{i}[3:0].
dv{i}	rx_dv	Receive data valid, asserted by the PHY layer when the first received preamble nibble is driven over the MII. It remains asserted for the remainder of the frame, up to the last CRC nibble.
rxd{i}[3:0]	rxd[3:0]	These lines provide receive data, driving a nibble on each rclk{i} cycle when dv{i} is asserted.
rerr{i}	rx_err	Receive error, asserted by the PHY layer to indicate an error the MAC cannot detect. If asserted during packet reception, indicates a coding error on the frame currently being transferred on rxd{i}[3:0]. If asserted while dv{i} is deasserted with rxd{i}[3:0] equal to 1110, indicates that a false carrier was detected by the PHY layer.
crs{i}	crs	Carrier sense, asserted by the PHY layer when either the transmit or receive medium is active (not idle).
col{i}	col	Collision, asserted by the PHY layer when it detects a collision on the medium. Remains asserted while this condition persists.
mdc	mdc	Management data clock, the mdio signal clock reference.
mdio	mdio	Management data input/output, used to transfer control signals between the PHY layer and the manager entity. The IXF440 is capable of initiating control signal transfer between the IXF440 and the PHY devices.

6.3 MAC Frame Format

Ethernet is the generic name for the network type implementing the IEEE 802.3 Standard. An Ethernet frame has a minimum length of 64 bytes and a maximum length of 1518 bytes, excluding the preamble and the SFD bytes.

An Ethernet frame format consists of the following fields:

- Preamble
- Start frame delimiter (SFD)
- Two address fields
- Type or length field
- Data field
- Frame check sequence (CRC value)

Preamble	SFD	Destination Address	Source Address	Type/Length	Data	FCS
(7)	(1)	(6)	(6)	(2)	(46..1500)	(4)

Note: Numbers in parentheses indicate field length in bytes.

Table 16 describes the Ethernet frame fields.

Table 16. Ethernet Frame Description

Field	Description
Preamble	A 7-byte field of alternating 1s and 0s: 10101010.
SFD (Start Frame Delimiter)	A single-byte field containing the value 10101011.
Destination address	A 6-byte field containing either a specific station address, or the broadcast address, or a multicast (logical) address, all of which indicate the frame's destination.
Source address	A 6-byte field containing the specific station address of frame origin.
Type/length	A 2-byte field indicating whether the frame is in the IEEE 802.3 format or the Ethernet format. A field greater than 1500 is interpreted as a type field, which defines the protocol type. A field smaller than or equal to 1500 (05-DC) is interpreted as a length field, indicating the number of data bytes in the frame.
Data	A data field consisting of 46 to 1500 information bytes. This data field is fully transparent because any arbitrary sequence of bits can occur. A data field shorter than 46 bytes, specified by the length field, is allowed. In its default mode, padding is enabled and up to 46 bytes are added to the data field by the IXF440 when transmitting.
FCS (Frame Check Sequence)	A 32-bit cyclic redundancy check (CRC), computed as a function of the destination address field, source address field, type field, and data field. The FCS is appended to each transmitted frame and is used during reception to determine if the received frame is valid.

The CRC polynomial, as specified in the 802.3 Standard, is as follows:

$$FCS(X) = X^{31}+X^{26}+X^{23}+X^{22}+X^{16}+X^{12}+X^{11}+X^{10}+X^{8}+X^{7}+X^{5}+X^{4}+X^{2}+X^{1}+1$$

The 32 bits of the CRC value are placed in the FCS field so that the X31 term is the right-most bit of the first octet. The CRC bits are thus transmitted in the following order: X31, X30,..., X1, X0.

A frame octet is transferred on the serial line from the LSB to the MSB. The octet $D_7D_6D_5D_4D_3D_2D_1D_0$, where D_7 is the MSB, is transferred on the MII data signals as two consecutive nibbles: first $D_3D_2D_1D_0$ and then $D_7D_6D_5D_4$.

6.4 MAC Transmit Operation

This section describes the transmit operation in detail, as supported by the IXF440. Transmit activities are registered into the network management registers, which are accessible through the CPU port.

6.4.1 Transmit Initiation

After the transmit FIFO is adequately filled up to the programmed threshold level (TX_TSHD_BOFF<TSD>), or after there is a full frame loaded into the transmit FIFO, the IXF440 starts to encapsulate the frame. The transmit encapsulation is performed by the transmit controller, which delays the actual transmission of the data onto the network until it has been idle for a minimum interpacket gap (IPG) time.

6.4.2 Initial Deferral

The IXF440 constantly monitors the line. Actual transmission of the data onto the network occurs only if it has been idle for a 96-bit time period and any backoff time requirements have been satisfied. In accordance with the standard, the IXF440 begins to measure the IPG in full-duplex mode from the tendeassertion, and from crs deassertion in half-duplex mode. Because CRS deassertion is an asynchronous event, it may be necessary to adjust the IPG used by the IXF440 in half-duplex mode (depending on the MII or SYM physical-layer selection). Care should be taken to maintain standards compliance if these fields are modified.

The IPG time includes two parts, IPS1 and IPS2:

- IPS1 Time (60–bit time)

  The IXF440 monitors the network for an idle state. If a carrier is sensed on the serial line during this time, the IXF440 defers and waits until the line is idle again before restarting the IPS1 time count. If a carrier is sensed during the first 40-bit time after transmission of the IXF440, the IXF440 does not defer (carrier sense inhibition period).

- IPS2 Time (36–bit time)

  The IXF440 continues to count time even if a carrier has been sensed on the network, and thus forces collisions on the network. This enables all network stations to fairly access the serial line.

6.4.3 Frame Encapsulation

The transmit data frame encapsulation stream includes the appending of the 56 preamble bits and the SFD to the basic frame beginning, and the FCS to the basic frame end. The basic frame loaded from the bus includes the destination address field, the source address field, the type/length field, and the data field. If the data field length is shorter than 46 bytes, and padding as well as CRC appending are not disabled (TX_PARAM<CRCD,PADD>), the IXF440 pads the basic frame with the pattern 00 for up to 46 bytes. At the end of the frame, the IXF440 appends the FCS field if CRC appending is not disabled. If during the beginning of the packet load, the txasis signal is asserted, the IXF440 ignores the programmed mode and transmits the frame without the padding and the FCS field.

In the MII mode, the transmit enable signal (ten{i}) is asserted together with the first preamble byte transmission and is deasserted with the last CRC byte transmission.

6.4.4 Collision

A collision occurs on a half-duplex network when concurrent transmissions from two or more nodes take place. During transmission, the IXF440 monitors the line condition.

In the MII mode, the IXF440 detects a collision when the collision detect signal (col{i}) asserts.

When the IXF440 detects a collision while transmitting, it halts transmission of the data and transmits a 32-bit jam pattern. If the collision was detected during the preamble or the SFD transmission, the jam pattern is transmitted after completing the SFD. This results in a minimum 96-bit fragment.

Note: The jam pattern is a fixed pattern that is not compared to the actual frame CRC. It has a very low probability (0.5^{32}) of having a jam pattern identical to the CRC.

In standard networks, collisions will always occur before transmission of 64 bytes (including preamble and SFD), in which case the IXF440 begins the backoff wait period.

The IXF440 scheduling of retransmission is determined through a controlled randomization process, termed the truncated binary exponential backoff. The delay time is represented by an integer multiple of slot times (1 slot is equal to a 512-bit time period). The number of the delay slot times, before the *n*th retransmission attempt, is chosen as a uniformly distributed random integer in the range:

$$0 \le r < 2^k$$

$$k = \min(n, N)$$

The maximum backoff time is programmable by limiting the “N” number (TX_TSHD_BOFF<BKL>). In the IEEE 802.3 Standard, “N” is equal to 10.

When 16 transmission attempts have been made, all terminated by collisions, the IXF440 reports an excessive collision event. The IXF440 then stops transmission if programmed to do so (TX_ERR_MOD<XLCS>). Otherwise, it will continue to transmit the same packet again and again, while restarting the backoff algorithm as if it were a new packet (n = 0).

In network topology violating standard requirements, collision may occur after transmission of 64 bytes. If the IXF440 was programmed to stop upon late collision (TX_ERR_MOD<LCLS>), it will flush the transmit FIFO and stop transmitting. Otherwise, the IXF440 will discard the packet from its FIFO, and resume transmission with the next packet.

6.4.5 Terminating Transmission

A specific frame transmission is terminated under any of the following conditions:

- Normal

  The frame has been transmitted successfully. After the last byte is serialized, the pad and CRC are optionally appended and transmitted, thus concluding frame transmission.

- CRC error

  The txerr signal was asserted during packet loading. The IXF440 infects the CRC it is building and sends a bad CRC onto the network. An MII error is generated as well (terr assertion in MII mode).

- Underflow

  Transmit data is not ready when needed for transmission. The packet is terminated on the network with a bad CRC and an MII error generation (terr assertion in MII mode).

- Excessive collisions

A collision occurs 16 consecutive times during transmission attempt of the same frame.

- Late collision

  A collision occurs after the collision window (transmitting at least 64 bytes). The transmission is cut off.

6.4.6 Backpressure

The IXF440 provides the ability to prevent packet reception in the half-duplex mode. The backpressure mode can be activated by programming (SER_COM<BKP>) or by asserting the flct{i} signal. If any one of these is asserted, each receive activity detection triggers a transmit activity that will cause a collision. This transmission consists of a 96-bit pattern. In the backpressure mode, following a collision, the IXF440 resends packets with minimal IPG and without using the random backoff algorithm.

6.4.7 Flow Control

In the full-duplex mode, the IXF440 supports the standard flow control defined in the IEEE 802.3 Standard, enabling the stopping of remote node transmissions. Upon triggering, the IXF440 sends a flow control frame in the following format:

	Destination Address	Source Address	Type	Op-Code	Pause Time	Padding	FCS
Byte Count	6	6	2	2	2	42	4
Value (Canonic Form)	01-80-C2-00-00-01	EA1-EA2-..-EA6	88 - 08	00 - 01	PT1 - PT2		

The source address field (EA1 – EA6) is taken from the PHY_ADD register.

Upon assertion of the flct{i} signal or setting of the flow control trigger bit (SER_COM<FCT>), a flow control frame is sent with the pause time field (PT1 – PT2) equal to the PAUSE_TIME register. Upon deassertion, another flow control frame will be sent with the pause time field equal to zero, meaning that the remote node may resume transmission. If the XON mode is disabled (TX_PARAM<XOND>), the flow control frame will not be transmitted on deassertion. If the chip is transmitting, the next flow control frame sent will be according to the last flow control trigger.

6.5 MAC Receive Operation

This section describes the detailed receive operation as supported by the IXF440. Receive activities are registered into network management registers, which are accessible through the CPU port.

6.5.1 Receive Initiation

The IXF440 continuously monitors the network when reception is enabled. When an activity is recognized, the IXF440 starts to process the incoming data.

In the MII mode, the IXF440 detects activity when the data valid signal (dv{i}) asserts.

After detecting receive activity on the line, the IXF440 starts to process the preamble bytes.

6.5.2 Preamble Processing

The IEEE 802.3 Standard allows a maximum size of 56 bits (7 bytes) for the preamble, while the IXF440 allows any arbitrary preamble length. The IXF440 checks for the start frame delimiter (SFD) byte. If the IXF440 receives a binary 11 before receiving 6 bits or a binary 00 anywhere while checking for SFD, the reception of the current frame is aborted. The frame is not received, and the IXF440 waits until the network activity stops before monitoring the network activity for a new preamble.

The interpacket gap (IPG) between received frames should be at least 32-bit times.

6.5.3 Frame Decapsulation

While the frame is being assembled, the IXF440 continues to monitor the line condition.

In the MII mode, the IXF440 detects the end of frame when the data valid signal (dv{i}) deasserts.

Reception terminates with a frame error if the frame is not a valid MAC frame, or if an MII error was detected during frame reception.

In the MII mode, an MII error is detected when the receive error signal (rerr{i}) asserts during frame reception.

The IXF440 refers to the last 4 full bytes received as the CRC. It checks the CRC bytes of all received frames and reports all errors. Only whole bytes are run through the CRC check.

6.5.4 Terminating Reception

When reception terminates, the IXF440 determines the status of the received frame and loads the status into the receive FIFO. The IXF440 can report the following events at the end of frame reception:

- Overflow

  The IXF440 receive FIFO is not emptied as rapidly as it is filled, and an error occurs as frame data is lost. If the FIFO is already full when a new frame is received, it will not be loaded in the FIFO.

- CRC error

  The 32-bit CRC, transmitted with the frame, did not match the CRC calculated upon reception.

- Alignment error

  The frame did not end on a byte boundary, but with a spare nibble (4 bits), and a CRC error has also occurred. If the frame was not composed of an integral number of bytes, but the CRC was okay, the frame is considered good and no error is reported.

- MII error

  An MII error was detected during frame reception.

- Frame too short

  A frame containing less than 64 bytes was received (including CRC).

- Frame too long

A frame containing more than the programmed maximum size was received.

6.5.5 Flow Control

In the full-duplex mode, the IXF440 identifies standard flow control frames during reception. If a flow control frame is received and flow control mode is enabled (TX_PARAM<FLCE>), frame transmission will be stopped until expiration of the pause time. If flow control frames are received during frame transmission, the frame will be completely transmitted on the line, and then transmission will stop. The IXF440 identifies flow control frames according to the field matching described in Table 17, and correctness of the CRC.

Table 17. Flow control Field Matching

	Byte Number	Value (Canonic Form)
Destination Address	1 – 6	01-80-C2-00-00-01
Type	13 – 14	88-08
Op-Code	15 – 16	00-01

The value of fields 17 – 18 in the frame indicates the transmit pause time, represented in units of slot times.

Flow control frames are not loaded into the receive FIFO. They are discarded following identification, unless the IXF440 is programmed to pass them (RX_FILT_ MOD<PFLC>).

6.6 MAC Full-Duplex Operation

In full-duplex mode (SER_MOD<FDX>), the IXF440 activates the transmit and receive processes simultaneously. It also supports receive back-to-back packets with a minimal interpacket gap (IPG), in parallel with the transmission of back-to-back packets with an IPG of 96 bit times. In this mode, the MAC will ignore both the carrier sense (crs{i}) and the collision detect (col{i}) signals.

6.7 MAC Loopback Operations

The IXF440 supports two loopback modes:

- Internal loopback
- External loopback

6.7.1 Internal Loopback Mode

The internal loopback mode enables verification that the internal logic operates correctly. In this mode, frames loaded in the transmit FIFO are transferred to the receive FIFO through the transmit logic and receive logic. In the internal loopback mode, the IXF440 disconnects from the network. Frames are not transmitted onto the line (except in SYM mode), and frames received from the line are rejected.

6.7.2 External Loopback Mode

The external loopback mode enables verification that the logic up to the wire operates correctly. In the external loopback mode (SER_MOD<ELPK>), the external logic must be programmed to loop back frames from the transmit side to the receive side. In SYM mode, the loopback will work properly even if the link detection signal (sd) is not asserted.

6.8 SYM Mode

In the SYM mode (SER_MOD<SYP>=1), the 100BASE-X PCS encoding and decoding is performed by the IXF440. The functions implemented in this mode include:

- 4-bit to 5-bit encoding in the transmit path
- 5-bit to 4-bit decoding in the receive path
- Start-of-stream delimiter (SSD) and end-of-stream delimiter (ESD) detection and generation
- Bit alignment
- Carrier detect
- Collision detect
- Symbol error detection
- Link timer

In the 100BASE-TX mode (SER_MOD<SCR>=1), the IXF440 also performs the scrambling and descrambling functions.

In the SYM mode, the MII/SYM port works as a SYM port. Table 18 describes the SYM port signal names and their appropriate functions.

Table 18. SYM Port Signal Description

SYM Signals	Description
tclk{i}	25 MHz transmit clock, synchronizing the txd{i} signals.
rclk{i}	25 MHz receive clock, synchronizing the rxd{i} signals.
txd{i}[4:0]	Transmit data lines, driving a symbol on each tclk{i} cycle.
rxd{i}[4:0]	Receive data lines, driving a symbol on each rclk{i} cycle.
lnk{i}	Link signal, asserted by the IXF440 when the PCS logic detects a link to a remote node. If the descrambler is enabled, the signal will not be asserted unless it is locked.
sd{i}	Signal detect, asserted by the PHY layer when it detects link connection to the remote mode.
act{i}	Activity signal, asserted by IXF440 when it transmits a frame or is receiving a frame.
rerr{i}	Not used. Must be connected to 0.

When operating in the SYM mode, the IXF440 encapsulates and decapsulates the frames according to the IEEE 802.3 100BASE-X standard. The MII port then serves as an internal port between the MAC layer and the SYM port.

During transmit, encapsulation is performed according to the following rules:

- The first byte of the preamble in the MAC frame is replaced with the JK symbol pair.
- All of the MAC frame data is encoded according to 4B/5B standard encoding.
- After the FCS byte of the MAC frame, the TR symbol pair is inserted.
- An IDLE symbol is transmitted between frames.
- MII error generation is translated to illegal symbol generation.

During receive, decapsulation is performed according to the following rules:

- A non IDLE symbol will cause the internal carrier sense signal to be asserted.
- An IJK symbol sequence causes the internal data valid signal to be asserted (start of receive activity).
- The JK symbol pair is replaced by a preamble byte.
- All of the data symbol stream is decoded according to 5B/4B standard decoding.
- The TR symbol sequence will cause the internal data valid signal to be deasserted (end of receive activity).

In the SYM mode, collision is detected when the receive input is active while the IXF440 transmits. When the collision is detected, the internal collision detect signal is asserted.

During receive, the IXF440 expects the frame to start with the symbol sequence IJK followed by the preamble. If an IJK symbol sequence is not detected, the reception of the current frame is aborted (not received), and the IXF440 waits until the network activity stops before monitoring the network activity for a new frame. During reception, the IXF440 also checks symbol validity. If an invalid symbol is being received, or if the frame does not end with the TRI symbol sequence, the IXF440 reports an internal MII error.

7.0 Timing Diagrams

This section contains the IX Bus port and MII/SYM port timing diagrams.

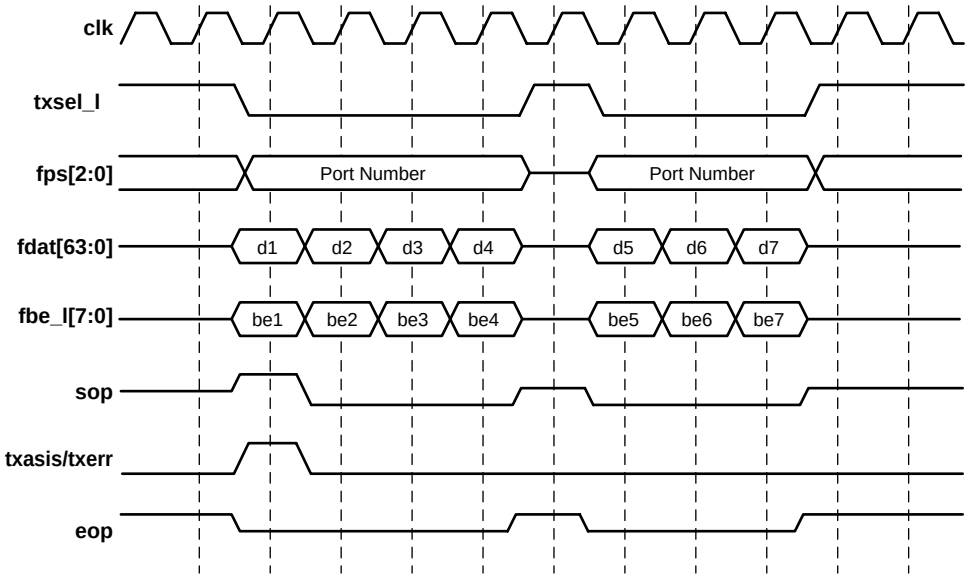
7.1 IX Bus Port Timing Diagrams

This section describes the IX Bus port timing diagrams.

7.1.1 Transmit Start-of-Packet Timing

Figure 4 shows the transmit start-of-packet timing.

Figure 4. Transmit Start-of-Packet Timing



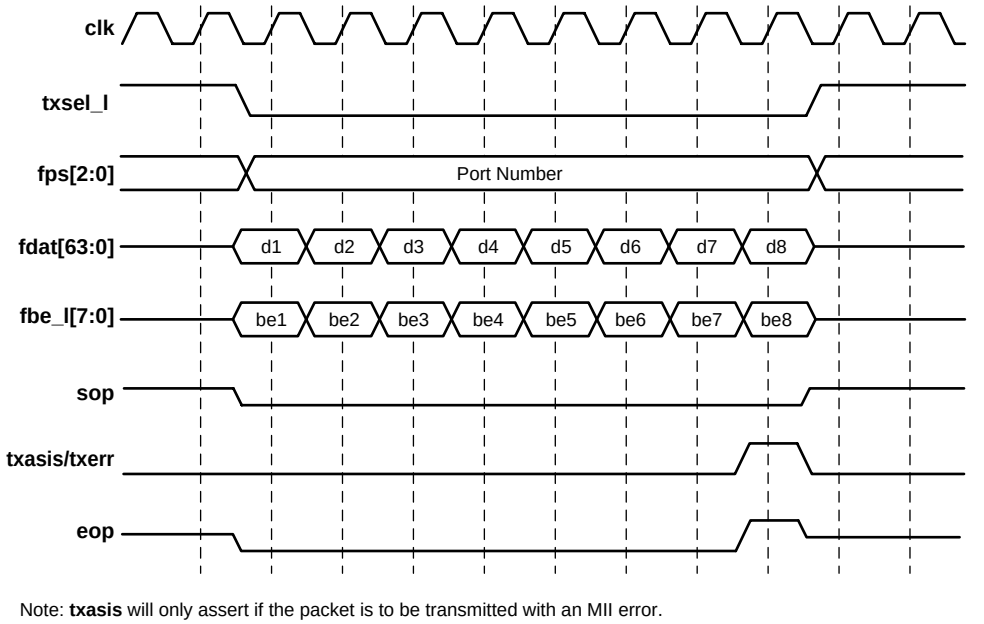
Note: **txasis** will only assert if no padding or CRC error is to be appended.

A4916-01

7.1.2 Transmit End-of-Packet Timing

Figure 5 shows the transmit end-of-packet timing.

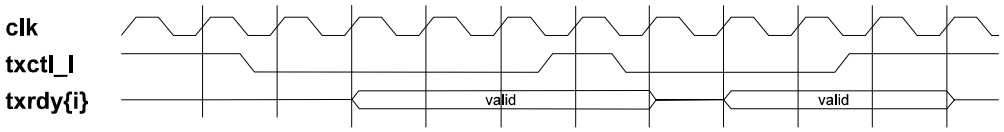
Figure 5. Transmit End-of-Packet Timing



7.1.3 Transmit FIFO Control Timing

Figure 6 shows the transmit FIFO control timing.

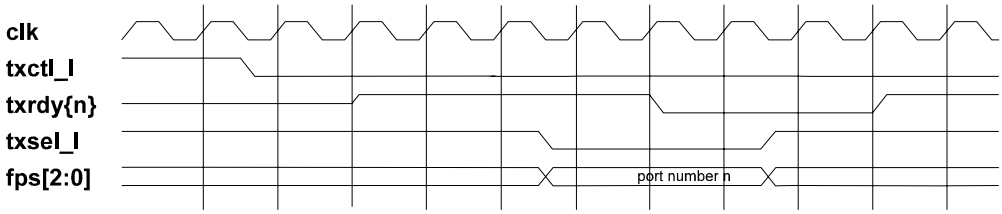
Figure 6. Transmit FIFO Control Timing



7.1.4 Transmit Ready (txrdy) Timing

Figure 7 shows the transmit ready (txrdy) timing.

Figure 7. Transmit txrdy Timing



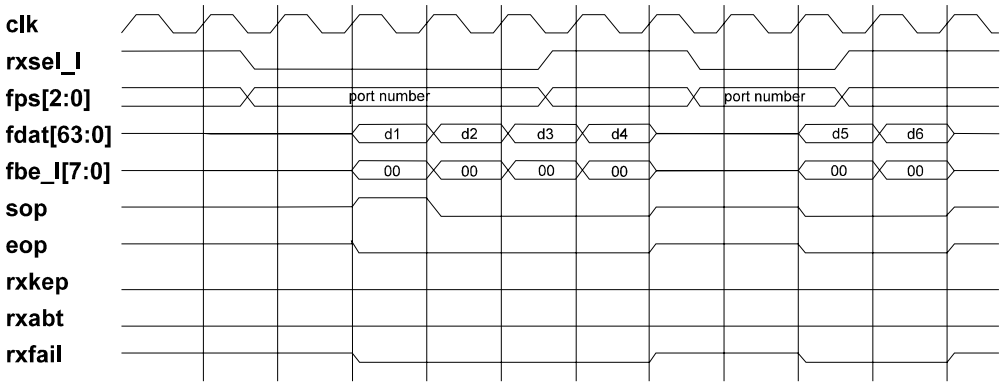
txrdy will assert a minimum of two cycles after the assertion of txctl_l, provided the available space in the transmit FIFO meets the programmed threshold. txrdy assertion must also be at least two cycles after the deassertion of txsel_l.

txrdy deasserts two cycles after the assertion of txsel_l.

7.1.5 Receive Start-of-Packet Timing

Figure 8 shows receive start-of-packet timing.

Figure 8. Receive Start-of-Packet Timing

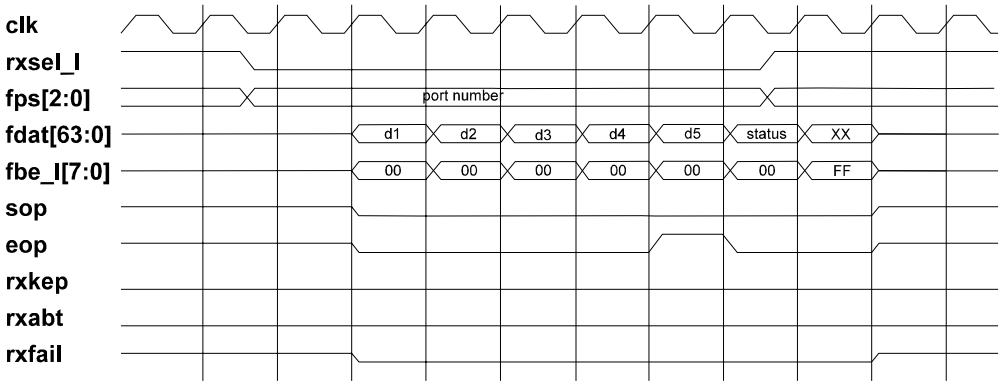


Note: rxsel_l must be deasserted for at least two cycles between two accesses to the same port.

7.1.6 Receive End-of-Packet Timing

Figure 9 shows the receive end-of-packet timing.

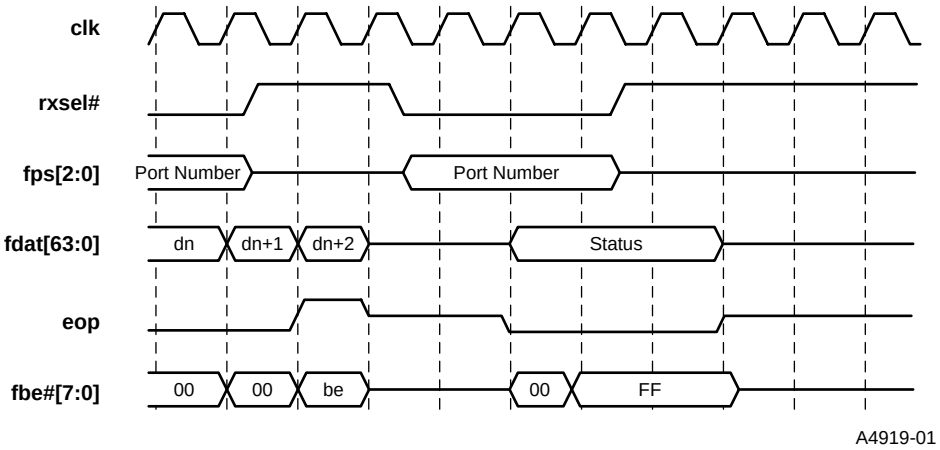
Figure 9. Receive End-of-Packet Timing



7.1.7 Fastest Receive Reaccess after EOP

Figure 10 shows the fastest receive reaccess to the same port after EOP (to get status)

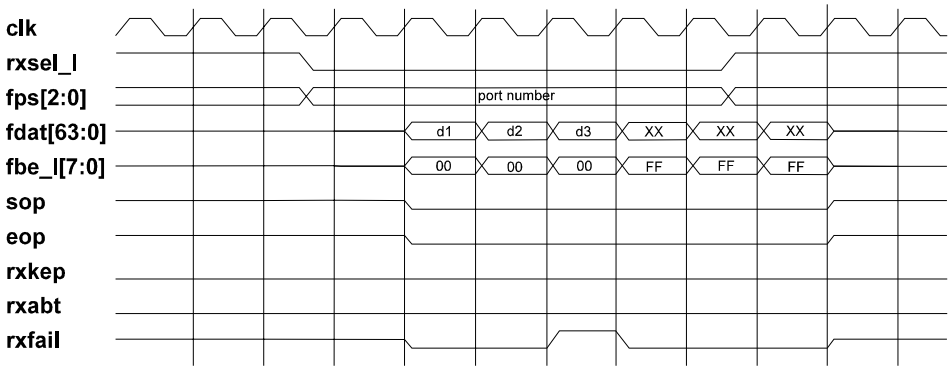
Figure 10. Fastest Receive Reaccess After EOP



7.1.8 Receive rxfail Timing

Figure 11 shows the receive rxfail timing.

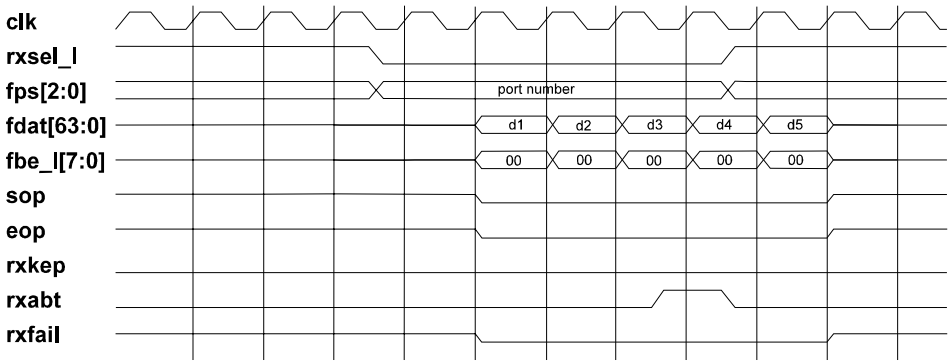
Figure 11. Receive rxfail Timing



7.1.9 Receive rxabt Timing

Figure 12 shows the receive rxabt timing.

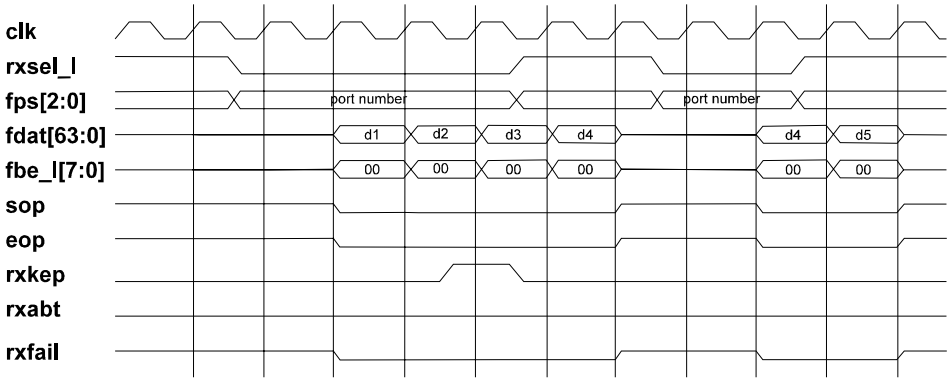
Figure 12. Receive rxabt Timing



7.1.10 Receive rxkep Timing

Figure 13 shows the receive rxkep timing.

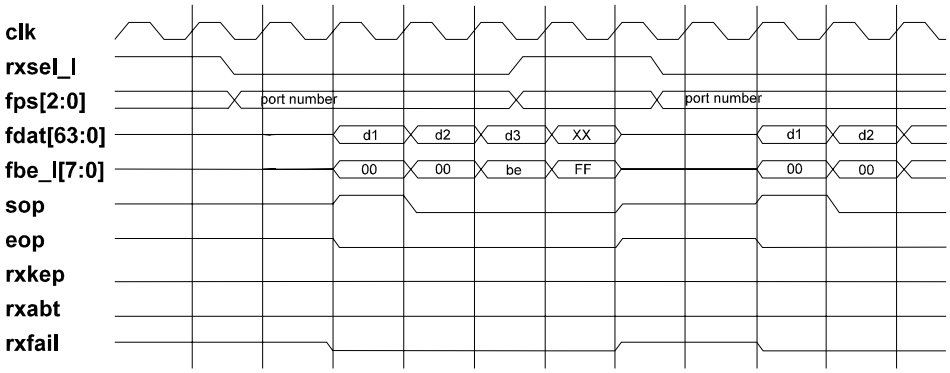
Figure 13. Receive rxkep Timing



7.1.11 Receive Header Replay Timing

Figure 14 shows the receive header replay timing.

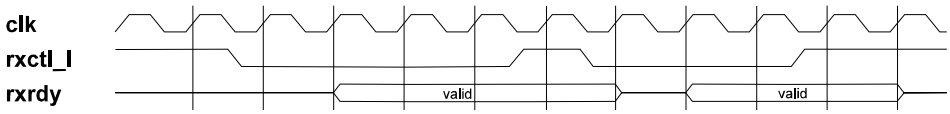
Figure 14. Receive Header Replay Timing



7.1.12 Receive FIFO Control Timing

Figure 15 shows the receive FIFO control timing.

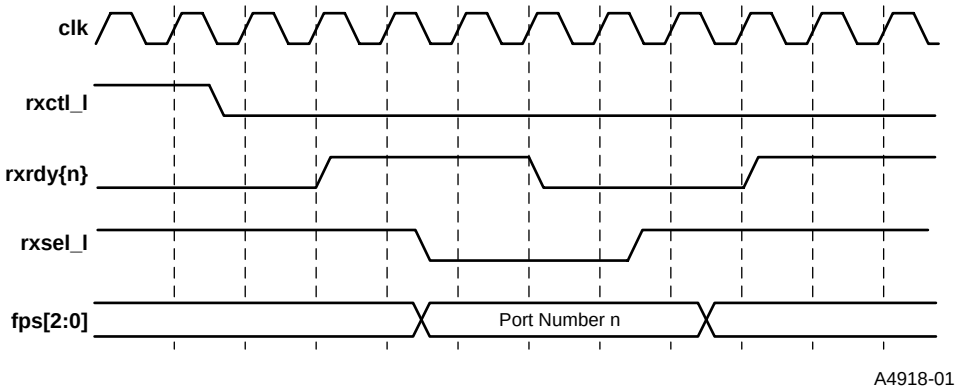
Figure 15. Receive FIFO Control Timing



7.1.13 Receive Ready (rxrdy) Control Timing

Figure 16 shows the receive ready (rxrdy) control timing.

Figure 16. Receive rxrdy Timing



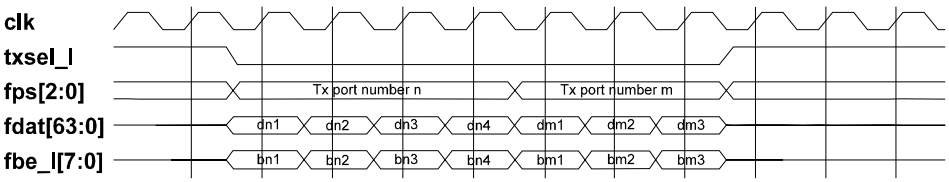
rxrdy will asset a minimum of two cycles after the assertion of rxctl_l, provided the amount of data in the receive FIFO meets the programmed threshold. rxrdy assertion must also be at least two cycles after the deassertion of rxsel_l.

rxrdy deasserts two cycles after the assertion of rxsel_l.

7.1.14 Consecutive Transmit-Transmit Timing

Figure 17 shows the consecutive transmit-transmit timing.

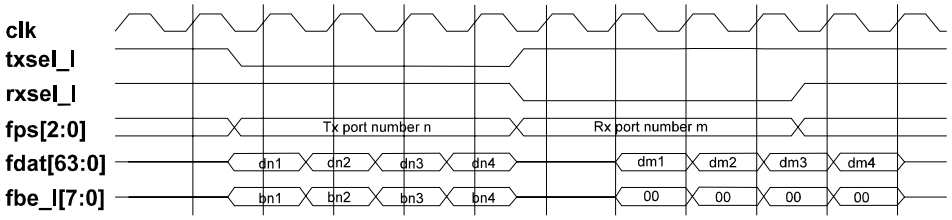
Figure 17. Consecutive Transmit-Transmit Timing



7.1.15 Consecutive Transmit-Receive Timing

Figure 18 shows the consecutive transmit-receive timing.

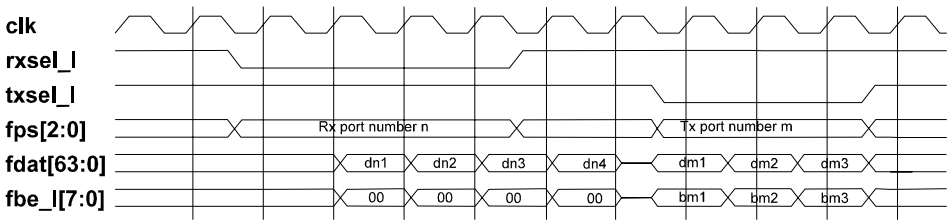
Figure 18. Consecutive Transmit-Receive Timing



7.1.16 Consecutive Receive-Transmit Timing

Figure 19 shows the consecutive receive-transmit timing.

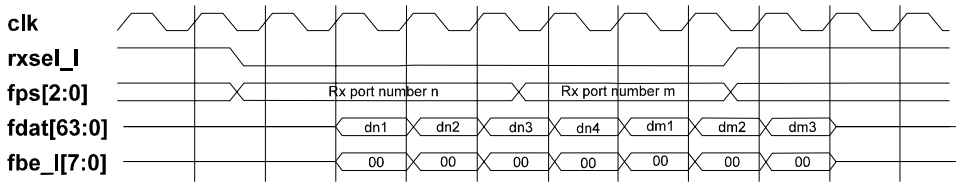
Figure 19. Consecutive Receive-Transmit Timing



7.1.17 Consecutive Receive-Receive Timing

Figure 20 shows the consecutive receive-receive timing.

Figure 20. Consecutive Receive-Receive Timing



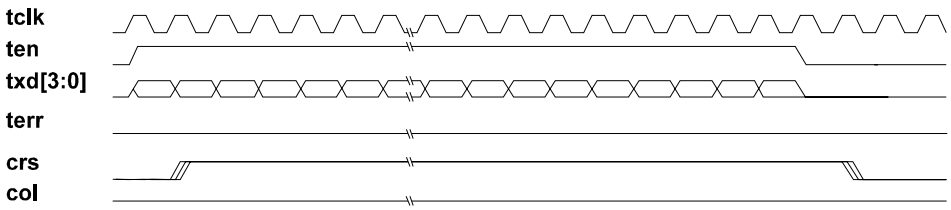
7.2 MII/SYM Port Timing Diagrams

This section shows the MII MII/SYM port timing diagrams. The MII/SYM port timing specification is compliant with the IEEE 802.3 Standard.

7.2.1 Packet Transmission Timing

Figure 21 shows the packet transmission timing.

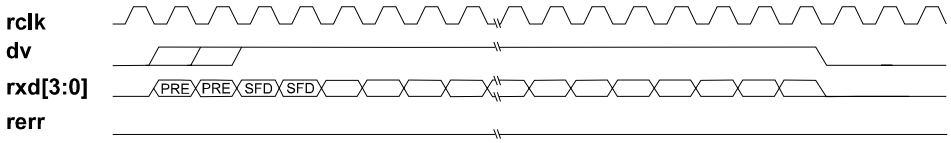
Figure 21. Packet Transmission Timing



7.2.2 Packet Reception Timing

Figure 22 shows the packet reception timing.

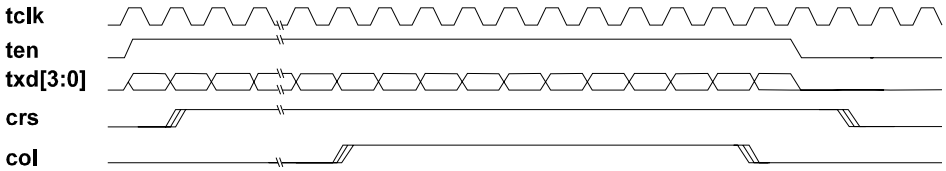
Figure 22. Packet Reception Timing



7.2.3 Transmission with Collision Timing

Figure 23 shows the transmission with collision timing.

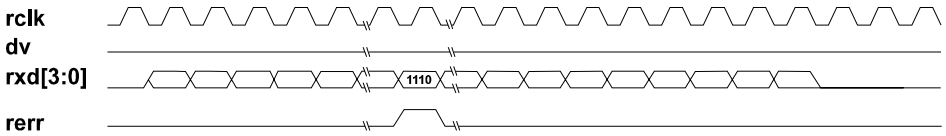
Figure 23. Transmission with Collision Timing



7.2.4 False Carrier Timing

Figure 24 shows the false carrier timing.

Figure 24. False Carrier Timing



8.0 Electrical and Environmental Specifications

This section contains the electrical and environmental specifications for the IXF440. The IXF440 supports both 5 V and 3.3 V signaling environments for all pins.

8.1 Functional Operating Range

Table 19 lists the functional operating range. Refer to Section 8.2 for details about the absolute maximum ratings. The signalling environment for the IX Bus pins are dependent on vdd_clmp.

Table 19. Functional Operating Range

Parameter	Minimum	Maximum
Power supply (Vdd)	3 V	3.6 V
Vdd_clmp (5.0 V signaling)	4.75 V	5.25 V
Vdd_clmp (3.3 V signaling)	3 V	Min. (3.6 V, Vdd + 0.3 V)
ESD protection voltage	—	2000 V
ESD protection voltage for Vdd_clmp	—	1900 V

Table 20 lists the AC parameters for the 3.3 V signaling levels.

Table 20. 3.3 V AC Signaling Specifications

Parameter	Condition	Minimum
Icl (Low clamp current)	–3 V < Vin < –1 V	–25 + (Vin + 1)/0.015 mA
Ich (High clamp current)	Vdd + 4 > Vin > Vdd + 1	25 + (Vin – Vdd –1)/0.015 mA

Table 21 lists the AC parameters for the 5 V signaling levels.

Table 21. 5 V AC Signaling Specifications

Parameter	Condition	Minimum
Icl (Low clamp current)	–5 V < Vin < –1 V	–25 + (Vin + 1)/0.015 mA

8.2 Absolute Maximum Rating

Applying stresses beyond the absolute maximum may cause unrecoverable damage to the device. Operation of this product is not implied for any condition beyond the ranges specified in the functional operation range described in Section 8.1. Operating this product at the absolute maximum rating for a prolonged period can negatively impact device reliability.

Table 22 lists the absolute maximum rating for the IXF440.

Table 22. Absolute Maximum Rating

Parameter	Maximum Rating
Supply voltage (Vdd)	3.9 V
Signal pins (Vsig)	5.5 V
Junction temperature, commercial (Tj)	125°C
Junction temperature, extended (Tj)	110°C

8.3 Supply Current and Power Dissipation

The values listed in Table 23 are based on a 66 MHz IX Bus clock frequency.

Table 23. Supply Current and Power Dissipation

Power Supply	Maximum Current	Maximum Power
3.3 V	750 mA	2.5 W
3.6 V	830 mA	3 W

8.4 Temperature Limit Ratings

Table 24 lists the temperature limit ratings.

Table 24. Temperature Limit Ratings

Parameter	Minimum	Maximum	Notes
Storage temperature	–55°C	125°C	
Operating temperature, commercial	0°C	70°C	
Operating temperature, extended	–40°C	85°C	Available only in 66 MHz device.

8.5 Reset Specification

The IXF440 reset signal (reset_l) is an asynchronous signal that must be active for at least 10 IX Bus clock (clk) cycles with stable power.

8.6 FIFO Port Specifications

This section describes the FIFO port specifications.

8.6.1 Clock Specification

Figure 25 shows the IX Bus clock timing diagram.

Figure 25. IX Bus Clock Timing Diagram

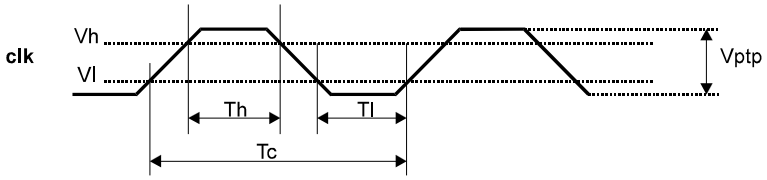


Table 25 lists the specifications of the IX Bus clock.

Table 25. IX Bus Clock Timing Specifications

Symbol	Parameter	Condition	Minimum	Maximum
Freq	Clock frequency	—	25 MHz[1]	66.6 MHz
Tc	Cycle time	—	15 ns	40 ns
Th	Clock high time	—	6 ns	—
Tl	Clock low time	—	6 ns	—
Vptp	Clock peak to peak (0.2 × Vdd to 0.6 × Vdd)	3.3 V clock	0.4 × Vdd	—
Vh	Clock high threshold	3.3 V clock	0.5 × Vdd	—
Vl	Clock low threshold	3.3 V clock	—	0.3 × Vdd
Vptp	Clock peak to peak (0.4 V to 2.4 V)	5 V clock	2 V	—
Vh	Clock high threshold	5 V clock	2 V	—
Vl	Clock low threshold	5 V clock	—	0.8 V

1. For testing purposes, the minimum frequency is 16 MHz.

8.6.2 3.3 Volt DC Specifications

Table 26 lists the DC parameters for the IX Bus 3.3 V signaling levels.

Table 26. IX Bus 3.3 V Signaling Specifications

Symbol	Parameter	Condition	Minimum	Maximum
Vih	Input high voltage	—	0.5 × Vdd	Vdd_clmp + 0.5 V[1]
Vil	Input low voltage	—	–0.5 V[2]	0.3 × Vdd
Ii	Input leakage current	0 < Vin < Vdd	–15 µA	15 µA
Voh	Output high voltage	Iout = –500 µA	0.9 × Vdd	—
Vol	Output low voltage	Iout = 1500 µA	—	0.1 × Vdd
Cin	Pin capacitance	—	5 pF	10 pF

1. Overvoltage protection maximum is +7.1 V through a series 29 Ω resistor for 11 ns.

2. Overvoltage protection minimum is –3.5 V through a series 28 Ω resistor for 11 ns.

8.6.3 5 Volt DC Specifications

Table 27 lists the DC parameters for the IX Bus 5 V signaling levels.

Table 27. IX Bus 5-V Signaling Specifications

Symbol	Parameter	Condition	Minimum	Maximum
Vih	Input high voltage	—	2.0 V	Vdd_clmp + 0.5 V[1]
Vil	Input low voltage	—	–0.5 V[2]	0.8 V
Ii	Input leakage current	0.5 V < Vin < 2.7 V	–15 µA	15 µA
Voh	Output high voltage	Iout = –2 mA	2.4 V	—
Vol	Output low voltage	Iout = 6 mA	—	0.55 V
Cin	Pin capacitance	—	5 pF	10 pF

1. Overvoltage protection maximum is +11 V through a series 29 Ω resistor for 11 ns.
2. Overvoltage protection minimum is –5.5 V through a series 28 Ω resistor for 11 ns.

8.6.4 IX Bus Signals Timing

Figure 26 shows the IX Bus signals timing diagram.

Figure 26. IX Bus Signals Timing Diagram

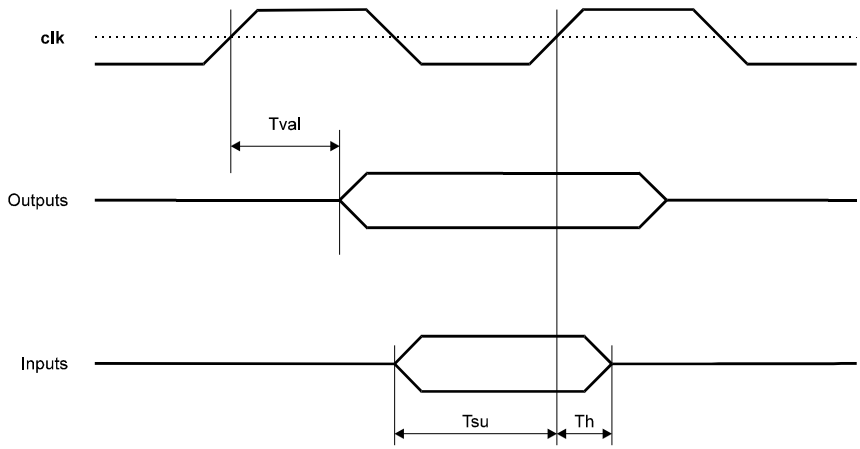


Table 28 lists the timing specifications of the IX Bus signals.

Table 28. IX Bus Signals Timing Specifications

Symbol	Parameter	Minimum (ns)	Maximum (ns)
Tval	Clock-to-signal valid delay	1	7
Tsu	Input signal valid setup time before clock	4	—
Th	Input signal hold time from clock	1	—

8.7 CPU Port Specifications

This section describes the CPU port electrical specifications.

8.7.1 DC Specifications

Table 29 lists the CPU port DC specifications.

Table 29. CPU Port DC Specifications

Symbol	Parameter	Condition	Minimum	Maximum
Vih	Input high voltage	—	2 V	—
Vil	Input low voltage	—	—	0.8 V
Voh	Output high voltage	Ioh = – 4 mA	2.4 V	—
Vol	Output low voltage	Iol = 4 mA	—	0.4 V
Ii	Input leakage current	—	–15 µA	15 µA
Cin	Pin capacitance	—	5 pF	10 pF
Iol	crdy_l, cint{i}_l Output low current	Vol = 0.55 V	20 mA	50 mA

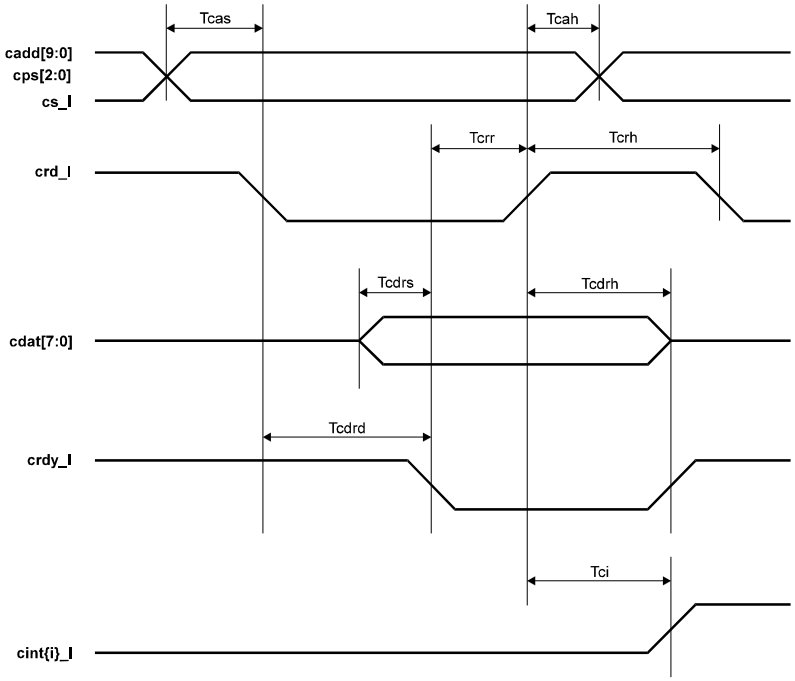
8.7.2 Signals Timing

This section describes the timing diagram of the CPU port.

8.7.2.1 Read Timing

Figure 27 shows the CPU port read timing diagram.

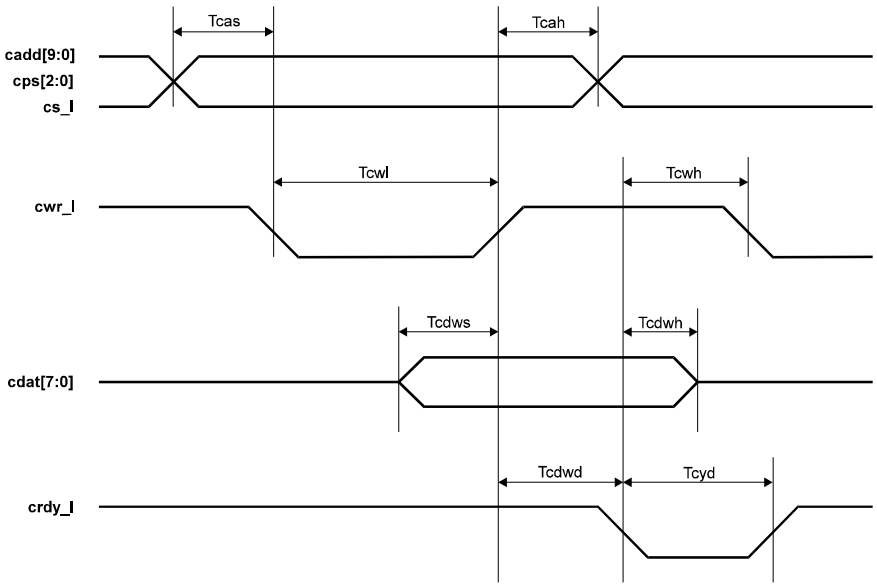
Figure 27. CPU Port Read Timing Diagram



8.7.2.2 Write Timing

Figure 8-4 shows the CPU port write timing diagram.

Figure 28. CPU Port Write Timing Diagram



8.7.2.3 Timing Parameters

Table 30 lists the CPU bus clock timing parameters.

Table 30. Timing Parameters

Symbol	Parameter	Minimum	Maximum
Tcas	cadd[9:0], cps[2:0], cs_l setup time	10 ns	—
Tcah	cadd[9:0], cps[2:0], cs_l hold time	10 ns	—
Tcrr	crdy_l assertion to crd_l deassertion	10 ns	—
Tcrh	crd_l high width	3 × TC[1]	—
Tcdrs	cdat[7:0] to crdy_l setup time	10 ns	—
Tcdrh	crd_l to cdat[7:0] hold time	TC	4 × TC
Tcdrd	Read cdat[7:0] driving delay	3 × TC	15 × TC
Tci	crd_l to cint_l clear delay	—	5 × TC
Tcwl	cwr_l low width	3 × TC	—
Tcwh	crdy_l to cwr_l hold time	2 × TC	—
Tcdws	cdat[7:0] to cwr_l setup time	10 ns	—
Tcdwh	crdy_l to cdat[7:0] hold time	10 ns	—
Tcdwd	Write cdat[7:0] latching delay	2 × TC	4 × TC
Tcyd	crdy_l width in write cycle	4 × TC	4 × TC
Trtw	Read crdy_l deassertion to cwr_l assertion	4 × TC	—
Twtr	Write crdy_l deassertion to crd_l assertion	4 × TC	—

1. TC is the IX Bus clock cycle time.

8.8 MII/SYM Port Specifications

The MII/SYM port electrical specifications are compliant with the IEEE 802.3 Standard.

8.8.1 DC Specifications

Table 31 lists the MII/SYM port DC specifications.

Table 31. MII/SYM Port DC Specifications

Symbol	Parameter	Condition	Minimum	Maximum
Vih	Input high voltage	—	2 V	—
Vil	Input low voltage	—	—	0.8 V
Voh	Output high voltage	Ioh = – 4 mA	2.4 V	—
Vol	Output low voltage	Iol = 4 mA	—	0.4 V
Ii	Input leakage current	—	–15 µA	15 µA
Cin	Pin capacitance	—	5 pF	10 pF

8.8.2 Signals Timing

This section describes the timing diagram of the MII/SYM port.

8.8.2.1 Clocks Specifications

Figure 29 shows the MII/SYM port clocks timing diagram.

Figure 29. MII/SYM Clock Timing Diagram

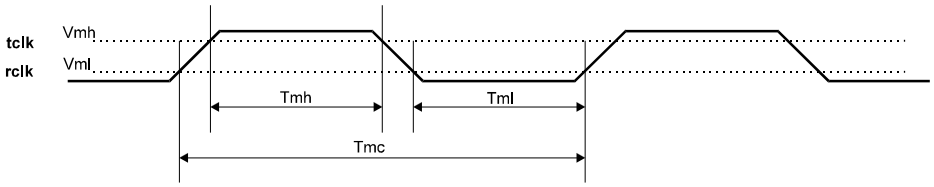


Table 32 lists the MII/SYM port signals timing specifications.

Table 32. MII/SYM Port Signals Timing Specifications

Symbol	Parameter	Minimum	Maximum
Fm10	tclk, rclk frequency in 10 Mbps	2.5 MHz –100 ppm	2.5 MHz + 100 ppm
Fm100	tclk, rclk frequency in 100 Mbps	25 MHz – 100 ppm	25 MHz + 100 ppm
Tmc	tclk, rclk cycle time	—	—
Tmh	tclk, rclk high time	0.35 × Tmc	0.65 × Tmc
Tml	tclk, rclk low time	0.35 × Tmc	0.65 × Tmc
Vmh	tclk, rclk high threshold	2 V	—
Vml	tclk, rclk low threshold	—	0.8 V

8.8.2.2 Signals Timing Diagrams

Figure 30 shows the MII/SYM port transmit timing characteristics.

Figure 30. MII/SYM Port Transmit Timing Diagram

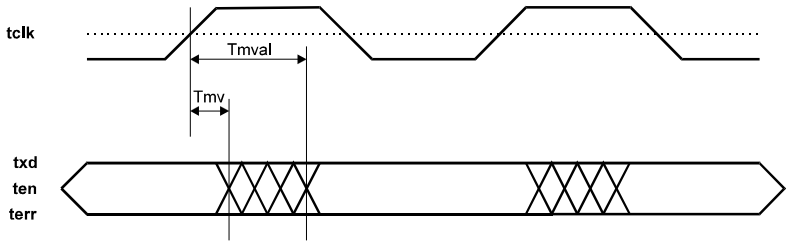


Figure 31 shows the MII/SYM port receive timing characteristics.

Figure 31. MII/SYM Port Receive Timing Diagram

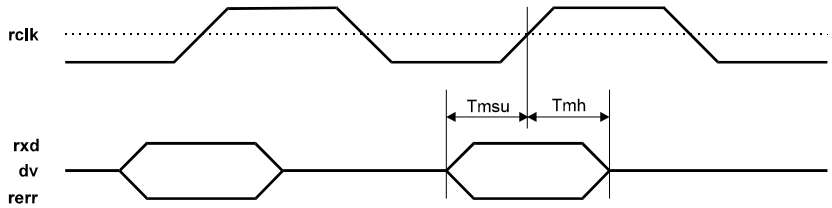
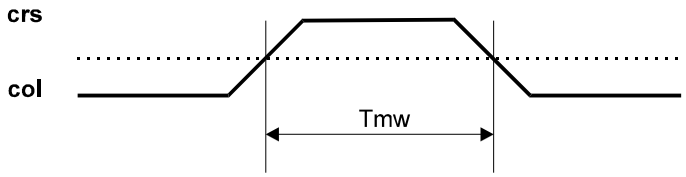


Figure 32 shows the MII/SYM port carrier sense and collision timing diagram.

Figure 32. MII/SYM Port Carrier Sense and Collision Timing Diagram



8.8.2.3 Data Timing Parameters

Table 33 describes the data timing parameters.

Table 33. Data Timing Parameters

Symbol	Parameter	Minimum	Maximum
Tmv	1. txd, terr, output valid hold after tclk	0 ns	—
	2. ten, output valid hold after tclk	0 ns	—
Tmval	tclk to output valid delay	—	16 ns
Tmsu	Input setup time before rclk	10 ns	—
Tmh	Input hold time after rclk	10 ns	—
Tmw	col, crs high time	20 ns	—

8.9 JTAG Port Specifications

This section describes the JTAG port electrical specifications.

8.9.1 DC Specifications

Table 34 lists the JTAG port DC specifications.

Table 34. JTAG Port DC Specifications

Symbol	Parameter	Condition	Minimum	Maximum
Vih	Input high voltage	—	2 V	—
Vil	Input low voltage	—	—	0.8 V
Voh	Output high voltage	Ioh = –4 mA	2.4 V	—
Vol	Output low voltage	Iol = 4 mA	—	0.4 V
Ii	Input leakage current (tck)	—	–20 µA	20 µA
Iip	Input leakage current with internal pull-up (tdi, tms)	—	–1500 µA	20 µA
Io	Tristate output leakage current (tdo)	—	–20 µA	20 µA
Cin	Pin capacitance	—	5 pF	10 pF

8.9.2 Signals Timing

Figure 33 shows the JTAG port timing characteristics.

Figure 33. JTAG Port Timing Diagram

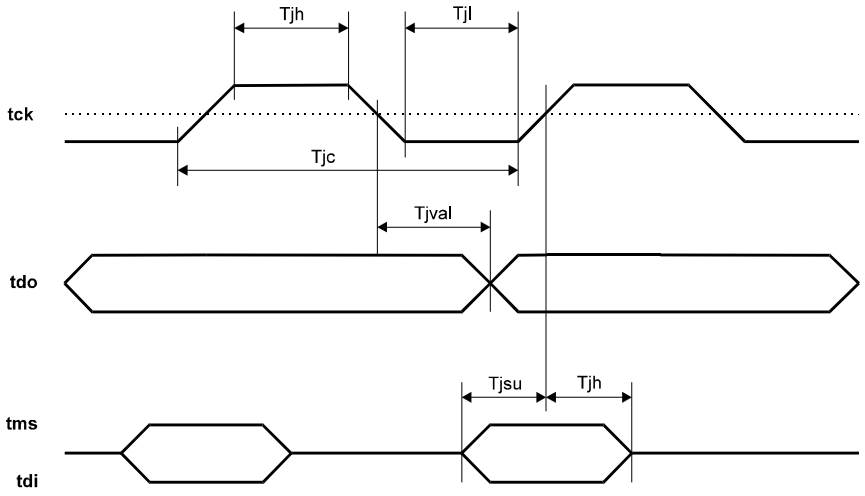


Table 35 lists the JTAG port timing specifications.

Table 35. JTAG Port Timing Specifications

Symbol	Parameter	Minimum	Maximum
Tjc	tck cycle time	90 ns	—
Tjh	tck high time	0.4 × Tjc	0.6 × Tjc
Tjl	tck low time	0.4 × Tjc	0.6 × Tjc
Tjval	tck fall to tdo valid delay	—	20 ns
Tjsu	tms and tdi setup time before tck	20 ns	—
Tjh	tms and tdi hold time from tck	5 ns	—

9.0 Mechanical Specifications

The IXF440 is contained in a 352-BGA package. Figure 34 shows the part marking.

Product Name	Stepping	Marketing Part Number	Notes
GCIXF440AC	A0	831001	
GCIXF440ACT	A0	838093	Extended temperature version. Available in 66 MHz device only.

Figure 34. Part Marking

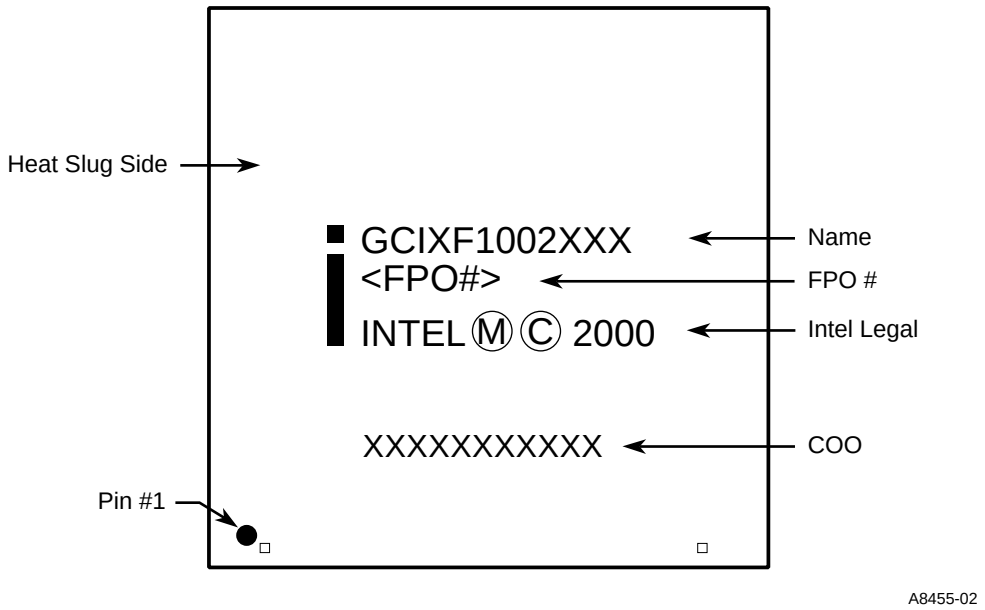


Figure 35, Figure 36, and Figure 37 show the 352-BGA package. Table 36 lists the dimensions in millimeters.

Figure 35. 352-BGA Package - Bottom View

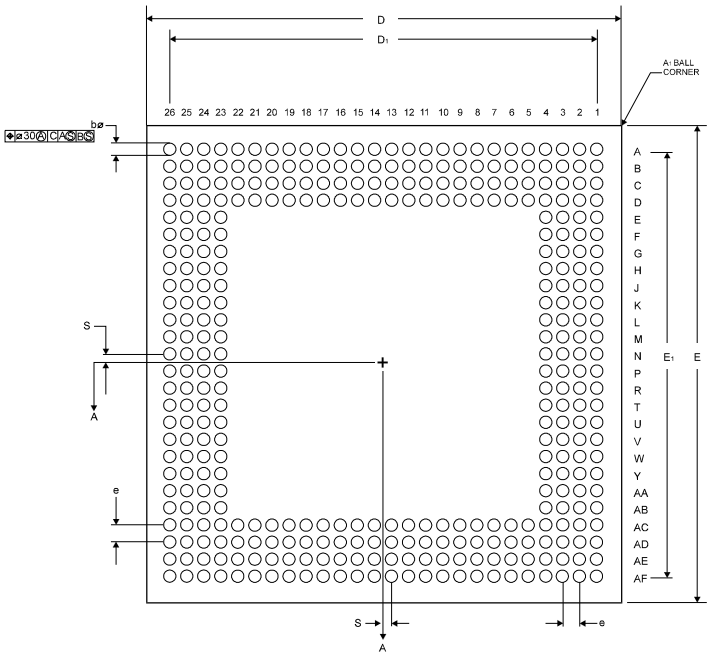


Figure 36. Side View

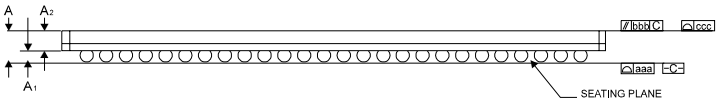


Figure 37. A–A Section View

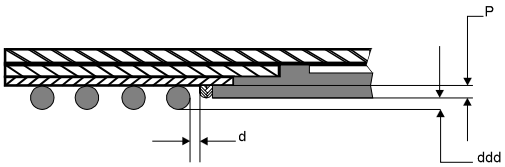


Table 36. 352-BGA Dimensional Attributes

Dimension	Symbol	Minimum	Nominal	Maximum
Package overall thickness	A	1.41	1.54	1.67
Ball height	A1	0.56	0.63	0.7
Body thickness	A2	0.85	0.91	0.97
Body size	E	34.9	35	35.1

Table 36. 352-BGA Dimensional Attributes

Dimension	Symbol	Minimum	Nominal	Maximum
Ball footprint	E1	31.65	31.75	31.85
Ball matrix	—	—	26 × 26	—
Number of rows deep	—	—	4	—
Ball diameter	b	0.6	0.75	0.9
Minimum distance encapsulation to balls	d	—	0.6	—
Ball pitch	e	—	1.27	—
Coplanarity	aaa	—	—	0.20
Parallel	bbb	—	—	0.15
Top flatness	ccc	—	—	0.2
Seating plane clearance	ddd	0.15	0.33	0.5
Encapsulation height	P	0.2	0.3	0.35
Solder ball placement	S	—	—	0.635