PM3392

S/UNI 1x10GE

SATURN User Interface For 10 Gigabit Ethernet Lan Phy

Data Sheet

Proprietary and Confidential

Released

Issue No. 7: December, 2006

Legal Information

Copyright

PMC-2001304 (R7).

Disclaimer

None of the information contained in this document constitutes an express or implied warranty by PMC-Sierra, Inc. as to the sufficiency, fitness or suitability for a particular purpose of any such information or the fitness, or suitability for a particular purpose, merchantability, performance, compatibility with other parts or systems, of any of the products of PMC-Sierra, Inc., or any portion thereof, referred to in this document. PMC-Sierra, Inc. expressly disclaims all representations and warranties of any kind regarding the contents or use of the information, including, but not limited to, express and implied warranties of accuracy, completeness, merchantability, fitness for a particular use, or non-infringement.

In no event will PMC-Sierra, Inc. be liable for any direct, indirect, special, incidental or consequential damages, including, but not limited to, lost profits, lost business or lost data resulting from any use of or reliance upon the information, whether or not PMC-Sierra, Inc. has been advised of the possibility of such damage.

Trademarks

S/UNI and SATURN are registered trademarks of PMC-Sierra, Inc. CRSU-4x2488, POS-PHY Level 4, and PMC-Sierra are trademarks of PMC-Sierra, Inc. Other product and company names mentioned herein may be the trademarks of their respective owners.

Patents

The technology discussed in this document may be protected by one or more patent grants.

Contacting PMC-Sierra

PMC-Sierra
100-2700 Production Way
Burnaby, BC Canada V5A 4X1

Tel: +1 (604) 415-6000
Fax: +1 (604) 415-6200

Document Information: document@pmc-sierra.com
Corporate Information: info@pmc-sierra.com
Technical Support: apps@pmc-sierra.com
Web Site: http://www.pmc-sierra.com

Revision History

Issue No.	Issue Date	Details of Change
7	December 2006	Updated package and ordering information.
6	February 2006	Updated ordering information including RoHS-compliant device details.
5	October 2002	Issue 5 data sheet created.
4	August 2002	Issue 4 data sheet created. Converted document to new template, modified doc to fix all Doc. related PREP's.
3	January 2002	Issue 3 data sheet created.
2	October 2001	Issue 2 data sheet created.
1	September 2000	Preliminary data sheet created.

Table of Contents

List of Registers

List of Figures

List of Tables

1 Definitions

Table 1 S/UNI-1x10GE Abbreviations

PL4	Short hand notation for POS-PHY Level 4 System Interface
PL4IDU	PL4 Input Data Unpacker
PL4ODP	PL4 Output Data Packer
PL4IO	PL4 Input and Output Interface
PL4 Input	POS-PHY Level 4 Input Interface: relates to data impressed on TDCLK+/- and TDAT[15:0]+/- pins.
PL4 Output	POS-PHY Level 4 Output Interface: relates to data impressed on RDCLK+/- and RDAT[15:0]+/- pins.
PL4MOS	PL4 Multi-stream Output Scheduler
EFLX	Egress FIFO
IFLX	Ingress FIFO
TXXG	10 Gbit/s Transmit MAC
RXXG	10 Gbit/s Receive MAC
R64B66B	10.3 Gbit/s Receive PCS
T64B66B	10.3 Gbit/s Transmit PCS
XSBI	10.3 Gbit/s Serial Interface
PHY	Physical layer interface
LAN PHY	Designated by IEEE 10gigabit Ethernet standards committee as one of two variants of the 10GE standard intended for applications in which SONET compatibility is not required and data rates should be maximized.
Data Frame or Frame	Consist of Destination Address, Source Address, Length Field, Logical Link Control (LLC) Data, PAD, and Frame Check Sequence.
Full Duplex	A mode of operation that supports simultaneous communication between a pair of stations, provided that the Physical Layer is capable of supporting simultaneous transmission and reception without interference.
IPG	Inter-Packet Gap (IPG): A delay or time gap between physical packets.
MIB	Management Information Base (MIB): A repository of information to describe the operation of specific network device.
MAC	Media Access Control (MAC): The data link sub-layer that is responsible for transferring data to and from the Physical Layer.
Packet	The logical unit of data transferred across the POS-PHY Level 3 interface. This generally corresponds to the Data Frame as defined previously, although the CRC may or may not be present in the POS-PHY Level 3 egress direction.
Physical Packet	Consists of a Data Frame as defined previously, preceded by the Preamble and the Start Frame Delimiter, encoded, as appropriate, for the Physical Layer (PHY) type.
POS-PHY	SATURN compatible Packet over SONET interface specification for physical layer devices. POS-PHY level 4 defines an interface for bit rates up to and including 10Gbit/s.
SOF	Start of Frame.
SOP	Start of Packet.
EOF	End of Frame.

EOP	End of Packet.
Jumbo Frame	In the context of this document, Jumbo Frame refers to a data frame (see definition in this table) that has a frame size in number of octets that meets all of the following criteria: 1. It is greater than the maximum 802.3 standard specified frame length (1518 octets for untagged frames and 1522 octets for tagged frames) 2. It is less than or equal to the maximum configurable receive/transmit frame size (for receive, see RXXG register 2045H, Receive Max Frame Length; for transmit, see TXXG register 3045H, Transmit Max Frame Length). 3. The frame length is less than or equal to 9600 octets Other than the number of octets in the data frame, a Jumbo Frame otherwise meets all the requirements as outlined in the 802.3 Ethernet standard.
PRBS	Pseudo Random Bit Sequencing

2 Features

2.1 General

- Implements 10 Gigabit Ethernet LAN PHY according to IEEE 802.3ae standard.
- Provides direct connection to optics via a 16-bit by 644.53125MHz IEEE 802.3ae XSBI line-side interface.
- Provides standard IEEE 802.3ae 10 Gigabit Ethernet Media Access Controller (10GMAC) for frame verification.
- Implements IEEE 802.3ae standard 64B/66B Physical Coding Sub-layer (PCS).
- Provides IEEE 802.3ae standard square wave and pseudo-random test pattern generation and checking.
- Provides SATURN® POS-PHY Level 4™ 16-bit LVDS System Interface for 10 Gigabit Ethernet applications.
- Line-side and System side loopback for system level diagnostic capability.
- Internal 128 Kbyte ingress FIFO and 16kbyte egress FIFO to accommodate system latencies.
- Provides a standard 5 signal IEEE 1149.1 JTAG test port for boundary scan board test purposes.
- Provides a generic 16-bit microprocessor bus interface for configuration, control, and status monitoring.
- Low power 1.8V CMOS core logic with 3.3V CMOS/TTL compatible digital inputs and digital outputs.
- Industrial temperature range (-40C to +85C) Ambient.
- 896 pin FCBGA package.

2.2 10 Gigabit Ethernet MAC

- Provides an IEEE 802.3ae standard10 Gigabit MAC for Ethernet frame handling.
- Provides mapping to insert/extract Ethernet frames into/from the IEEE 802.3ae standard Physical Coding Sub-layer (PCS).
- Verifies frame integrity (FCS and length checks).
- Supports 64B/66B-based frame delineation.
- Supports Ethernet 2.0, IEEE 802.3 LLC and IEEE 802.3 SNAP/LLC encoding formats including VLAN tagged frames.
- In the receive direction, supports frame delineation, frame integrity (FCS and length) checks, frame filtering and passing based on erred frames, 64 byte minimum frame size and a 9600 byte maximum frame size.

- Supports address filtering on all standard Ethernet size frames up to 9.6k Bytes.
- In the transmit direction, supports frame generation (preamble, CRC), minimum frame size padding up to 64 bytes, truncation of over-length frames.
- Supports multicast and unicast address filtering using eight exact match filters and promiscuous mode. Frames can be filtered based on SA, SA/VID, DA, or DA/VID. Provides 64-bin hash based algorithm to filter multi-cast addresses.
- With in-band PAUSE flow control the PM3392 implements 3 Km lossless flow control for 9600 byte frames and 5 Km lossless flow control for 1518 byte frames.
- Provides support for out-of-band flow control for upper layer device by using dedicated pins or host signaling to cause generation of a PAUSE frame.

2.3 Statistics

- Provides statistic counters to support
  - Ethernet MIB IEEE 802.3-2000, Clause 30 and 802.3ae
  - Compatible SNMP Interface Group MIB, RFC 1213 MIB II & RFC 2233 SMIv2
  - RMON Statistics Group MIB, RFC 1757
  - Ethernet-like MIB, RFC 2665
- Provides 40-bit wide counters for statistics

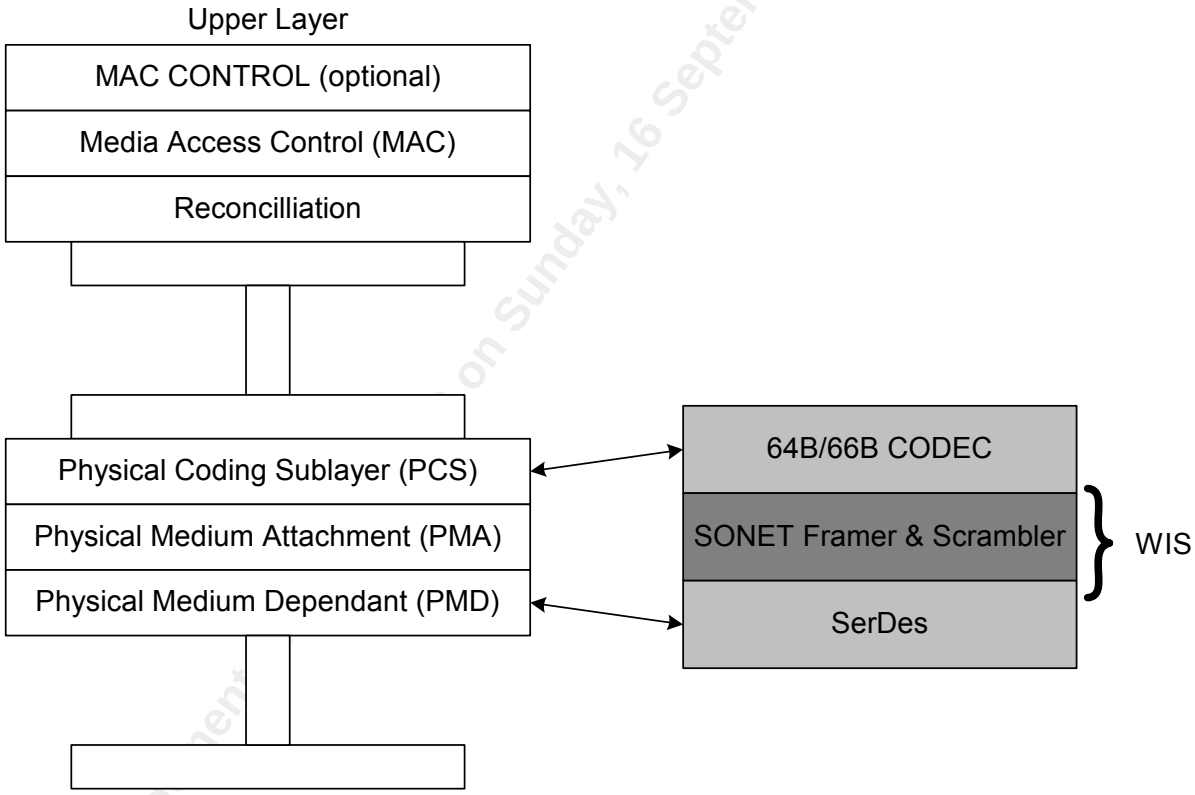
2.4 POS-PHY Level 4 Interface

- Designed to transmit cells, packets, or frames between physical and data-link layer devices.
- Compliant with the following standards:
  - ATM Forum – Frame Based ATM Interface Level 4 (ATMF0161.00)
  - Optical Internetworking Forum – System Physical Interface Level 4 Phase II (OIF2000.088)

3 Applications

- Metro POP (Point of Presence)
- Uplink Cards
- IP POP Router
- IP Services
- Multi-Service Switch

Figure 1 10 Gigabit Ethernet Reference Model



4 References

1. OIF – OIF99.102, “Proposal for a common electrical interface between SONET framers and SONET Serializer / Deserializer parts for STS-192/STM-64 interfaces.” July 1999.

2. OIF – OIF00.026.0, “SPI-4 Phase II.” baseline text.

3. PMC-991635, “POS PHY Level 4, Saturn Interface Specification For Packet and Cell Transfer Between Physical Layer and Link Layer Devices for OC-192 SONET/SDH and 10 Gbit/s Ethernet Applications”, Issue 4, April 2000.

4. IEEE 802.3aestandard; Media Access Control (MAC) Parameters, Physical Layer, and Management Parameters for 10 Gbit/s Operation

5. IEEE Std. 802.3, 2000 Edition; Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specification

6. RFC 1757 Remote Network Monitoring Management Information Base

7. RFC 1213 Management Information Base for Network Management of TCP/IP-based internets: MIB-II

8. RFC 2233 The Interfaces Group MIB using SMIv2

9. RFC 2665 Definitions of Managed Objects for the Ethernet-like Interface Types

10. PMC-Sierra, Inc, PMC-2010502 POS-PHY Level 4 Frequently Asked Questions

11. PMC-Sierra, Inc., “POS-PHY Level 4 Static Alignment Design Considerations Application Note”, Issue 1, March 2001

12. PMC-Sierra, Inc., PMC-2010198 “PMC PL4 Compliance Statement”, Issue 1, February 2001

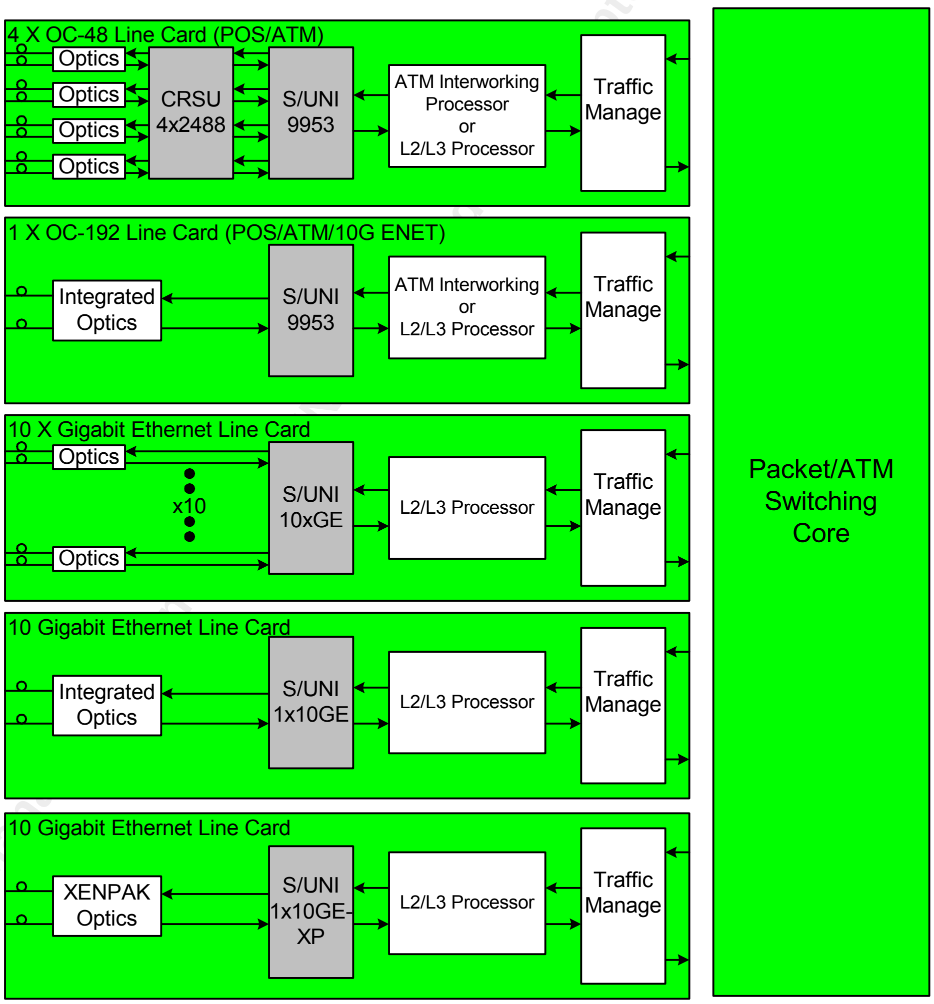
13. PMC-Sierra, Inc., PMC-2001305 “POS-PHY Level 4 Clocking and Initialization Application Note”, Issue 1, November 2000

14. PMC-Sierra, Inc., PMC-2020518 “PL4 Electrical Spec Clarifications Application Note”, Issue 1, March 2002

5 Application Examples

The PM3392 S/UNI®-1x10GE device is applicable to equipment implementing 10-Gigabit Ethernet LAN PHY interfaces. The S/UNI-1x10GE provides physical layer and MAC layer termination for connections between Edge, Enterprise Edge, and Core routers, Multi Service Switches and transport equipment at the 10 Gigabit rates. One of the most likely locations for initial deployment of 10GE LAN PHY is within the super-POP. Router to router connections is expected to be early adopters of 10GE LAN connections. Server farm connections will also provide a potential application for 10GE. It is also expected that 10GE WAN PHY will find application in connecting POPs to SONET or DWDM transport equipment.

Figure 2 Router - 10G Port Capable



6 Block Diagram

The following block diagrams give a high level view of the PM3392 S/UNI-1x10GE. The first depicts the device in normal mode exploiting all major paths. The second shows two distinct loop-back paths, system side and line side.

Figure 3 Normal Mode

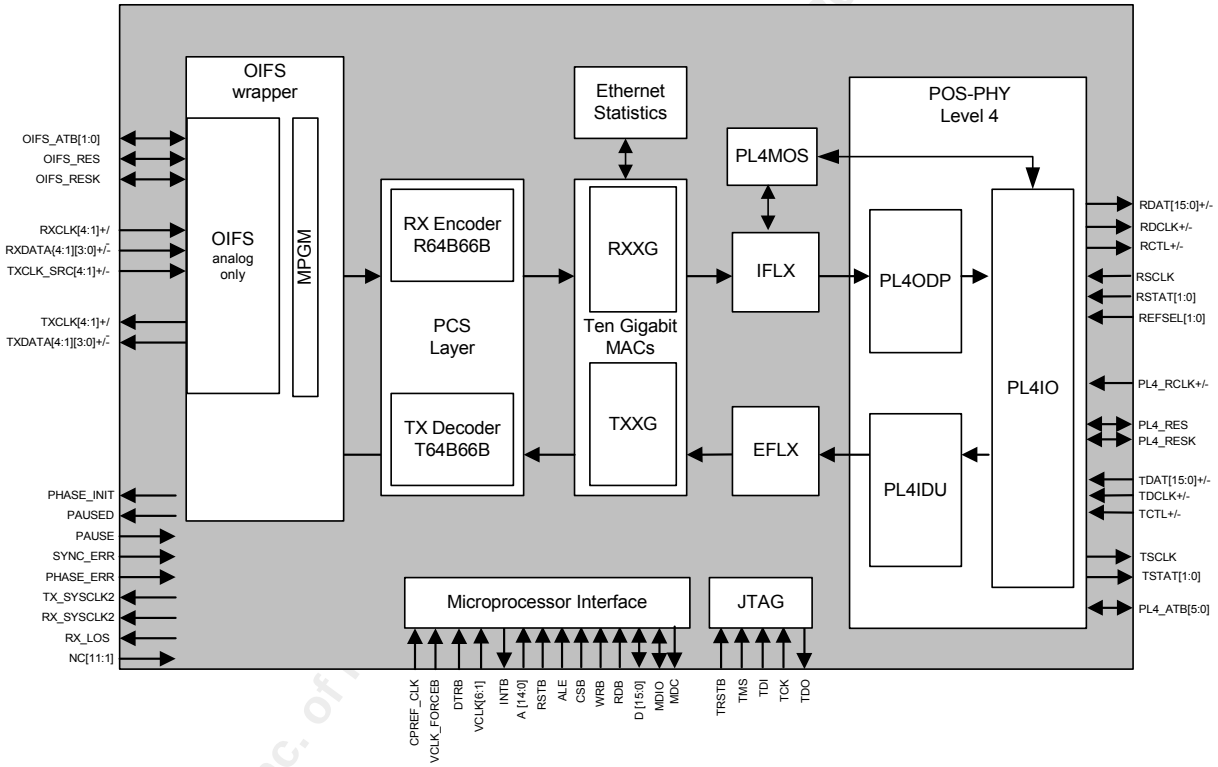


Figure 3 gives a high-level block view of the SUNI-1x10GE. All major data paths are shown with the appropriate signaling interfaces.

Figure 4 Loopback Paths

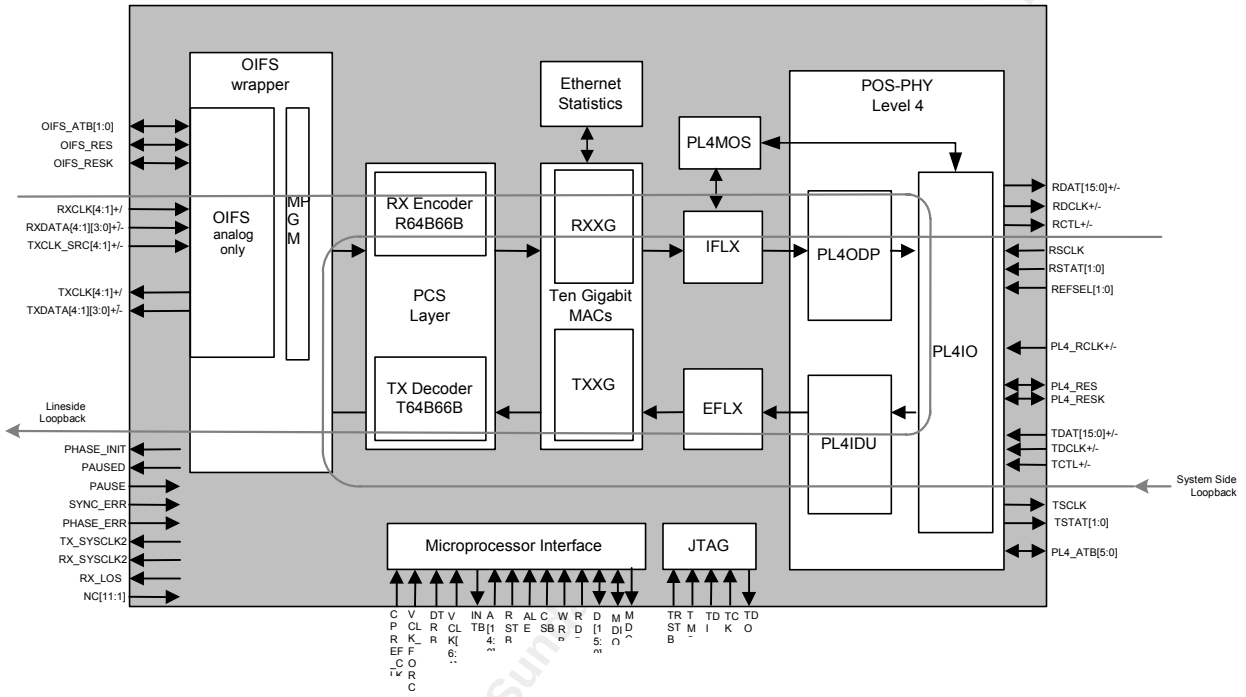


Figure 4 shows 2 possible loopback paths that are supported in the PM3392. The device supports both system side loopback through the XSBI wrapper, line side loopback through the PL4IO and system side loopback through the PL4IO.

7 Description

The PM3392 S/UNI-1x10GE SATURN User Network Interface is a monolithic integrated circuit that implements all the required functions of a 10 Gigabit IEEE 802.3ae standard compatible Ethernet LAN physical layer device (LAN PHY). The S/UNI-1x10GE Ethernet PHY device consists of a Ten Gigabit Sixteen-Bit Interface (XSBI) block, a 64B66B based physical coding sub-layer, a 10-Gigabit MAC, and a POS-PHY Level 4 interface.

The S/UNI-1x10GE transmits and receives Ethernet packets via connections to an optical module. The XSBI block accomplishes the interface bus between the optical module and the PM3392. The XSBI is derived from the OIF99.102.5 SFI-4 specification. The XSBI bus uses Low Voltage Differential Signaling (LVDS) for both clock and data receiver/transmitters. Sixteen pairs of LVDS data signals are provided in the transmit channel and another sixteen pairs in the receive channel. The source clock for the data receivers is RXCLK+/- and is used for the internal reference. In the transmit direction, a single reference clock, TXCLK_SRC+/- is provided for the 16 data channels. The source LVDS clock that is transmitted aligned with the data. The LVDS reference clock TXCLK_SRC, an input to the PM3392, is provided by the SERDES to the XSBI as a reference clock for the PCS layer, the MAC, and the Transmit Framer functionality. When the clock rate is 644.53125MHz, an aggregate of 10320Mbit/s (16x644.53125Mbit/s) is transferred in each direction.

The S/UNI 1x10GE TXXG block processes all outgoing Ethernet frames and performs the MAC functionality on the egress path. The TXXG provides Ethernet framing, insertion of an 8-byte preamble/Start Frame Delimiter sequence, plus computation and optional insertion of a 32-bit FCS. Frame timing is provided relative to the system clock reference input. The Transmit Framer will insert the correct programmable inter-frame gap between frames to ensure that the LAN-mode operation conforms to the IEEE 802.3 specification. The inter-frame gap, preamble, FCS generation, and error checking features of the Transmit Framer can be configured by means of internal configuration registers accessible via a microprocessor interface. It provides PAUSE frame generation and insertion. The PM3392 also supports an external “PAUSE” pin so that the system can force the MAC to send pause frames. Generated PAUSE frames are multiplexed into the outgoing frame stream in between data frames, with the proper spacing.

The S/UNI 1x10GE RXXG block processes all incoming Ethernet packet streams while performing basic frame checks. The MAC provides Ethernet framing detection, framing to the standard preamble/SFD sequence, removal of the preamble/SFD, checking of the 32-bit CRC field and frame validation (marking of erred frames for discard). Frame timing checks are relative to the receive input reference clock. The RXXG will verify that the inter-frame gap does not fall below a pre-set minimum and will filter frames that violate this restriction when used in LAN-mode devices. Optional received frame filtering allows frames to be discarded if they are found to contain length or CRC errors. The RXXG implements a 2048-byte full-frame buffer to facilitate this filtering. Jumbo frames that are erred will not be filtered but marked as an error and passed on. Jumbo frames will however, be address filtered.

The RXXG supplies received frame stream parsing and PAUSE MAC Control frame detection, validation and extraction. The PAUSE Timer fields of received PAUSE frames are extracted and sent to flow control logic. If enabled the flow control logic will pause data frames from begin transmitted onto the XSBI interface and assert the external “PAUSED” pin for use by the system. When the PAUSED pin is asserted this means that the TXXG is in a PAUSED state and is no longer transmitting data frames. The RXXG checks every received frame against the IEEE 802.3 frame error criteria and updates the appropriate statistics counters, which can be used to implement the standard Ethernet MIB for link management. Configuration and status maintenance, the minimum IFG, maximum frame size, preamble checking and erred frame discard functions, can be configured by means of internal configuration registers. Status registers are also implemented by the RXXG to permit a host CPU to monitor its functions. The frame check sequence is optionally verified for correctness and the extracted packets are placed in a receive FIFO. A full suite of Ethernet statistics are counted and provided for performance monitoring.

The PL4 interface provides a 16-bit wide data bus, an in-band control stream, a single control signal, and a dual phase source synchronous clock in the forward path. All forward path signals are differential LVDS. The clock signal is either loop timed from the TDCLK+/- or internally generated using the PL4_RCLK+/- reference input. The control signal is used to identify the in-band control words. In the return path, the PL4 interface provides a two-bit FIFO status bus with associated clock. These status signals are provided to the PL4MOS block to allow it to make scheduling decisions. In the case of the PM3392, decisions are limited to releasing data based on credit information. In addition, the Receive PL4 interface allows for the transmission of a training sequence to allow for dynamic de-skewing by a sink entity. Data must contain a sufficient training pattern density to allow reliable operation of the data recovery and de-skewing units. The PL4 interfaces transfer un-encoded NRZ data streams. Consequently there may be arbitrarily long runs of consecutive zeros or ones. The Transmit PL4 interface is capable of properly recovering data once training has completed.

The Receive PL4 Interface implements the PL4 protocol as described in PMC-991635. The PL4 interface consists of a PHY interface (PL4IO), Output Data Packer (PL4ODP), Input Data Packer (PL4IDU), and Multi-stream Output Scheduler (PL4MOS). The PL4ODP encapsulates the outgoing data stream originating at the IFLX FIFO interface. The PL4IDU unpacks the incoming PL4 data stream and presents the framed data to the EFLX FIFO. The PL4MOS is used to generate transfer requests to the ingress FIFO using a credit based scheduling scheme.

8 Pin Diagram

The PM3392 is packaged in an 896 Flip Chip Ball Grid array package. The body size is 31mm by 31mm with a 1.0mm ball pitch.

Figure 5 Ball View 1

	30	29	28	27	26	25	24	23	22	21	
A		VDDI	TDAT_N[1]	TDAT_N[4]	TDCLK_N	TDAT_N[9]	VSS	VDDO	TDAT_N[14]	TDAT_N[15]	A
B	VDDI	VDDI	VSS	TDAT_P[1]	TDAT_P[4]	TDCLK_P	TDAT_P[9]	VDDO	VSS	TDAT_P[14]	B
C	PL4_AVDH[2]	VSS	VSS	VDDO	TDAT_N[0]	TDAT_N[5]	TDAT_N[8]	TDAT_N[12]	VSS	VDDO	C
D	PL4_ATB[3]	PL4_AVDH[1]	VDDO	VDDO	VSS	TDAT_P[0]	TDAT_P[5]	TDAT_P[8]	TDAT_P[12]	VDDO	D
E	PL4_RESK	PL4_ATB[1]	PL4_AVDL[3]	VSS	VSS	VDDI	TDAT_N[3]	TDAT_N[6]	TDAT_N[10]	TDAT_N[11]	E
F	RDAT_P[0]	PL4_RES	PL4_AVDL[2]	PL4_AVDH[0]	VDDI	VDDI	VSS	TDAT_P[3]	TDAT_P[6]	TDAT_P[10]	F
G	VSS	RDAT_N[0]	PL4_ATB[0]	PL4_AVDL[1]	PL4_RCLK_P	VSS	VSS	VDDI	TDAT_N[2]	TDAT_N[7]	G
H	VDDO	VDDO	RDAT_P[1]	PL4_ATB[2]	PL4_ATB[5]	PL4_RCLK_N	VDDI	VDDI	VSS	TDAT_P[2]	H
J	RDAT_P[3]	VSS	VSS	RDAT_N[1]	PL4_AVDH[3]	PL4_ATB[4]	VSS	VSS	VSS	VDDI	J
K	RDAT_P[6]	RDAT_N[3]	VDDO	VDDO	RDAT_P[2]	VSS	VDDO	VDDO	VDDI	VDDI	K
L	RDAT_P[7]	RDAT_N[6]	RDAT_P[4]	VSS	VSS	RDAT_N[2]	VSS	VSS	VDDI	VSS	L
M	RDAT_P[8]	RDAT_N[7]	RDCLK_P	RDAT_N[4]	VDDO	VDDO	VDDI	VDDI	VDDI	VDDI	M
N	VDDI	RDAT_N[8]	RDAT_P[9]	RDCLK_N	RDAT_P[5]	VSS	VSS	VSS	VDDO	VDDI	N
P	VSS	VSS	RDAT_P[10]	RDAT_N[9]	RDAT_P[11]	RDAT_N[5]	VDDO	VDDO	VDDO	VDDO	P
R	VDDO	VDDO	VDDI	RDAT_N[10]	VSS	RDAT_N[11]	VSS	VSS	VSS	VDDO	R
T	VDDO	VDDI	VDDI	RDAT_N[13]	VSS	REFSEL[0]	VSS	VSS	VSS	VDDO	T
U	VSS	VSS	RDAT_P[13]	RDAT_N[14]	REFSEL[1]	VSS	VDDO	VDDO	VDDO	VDDO	U
V	VDDI	RDAT_N[12]	RDAT_P[14]	NC5	VSS	VSS	VSS	VSS	VDDO	VDDI	V
W	RDAT_P[12]	RDAT_N[15]	NC6	RSTAT[1]	VDDO	VDDO	VDDI	VDDI	VDDI	VDDI	W
Y	RDAT_P[15]	RCTL_N	RSTAT[0]	VSS	VSS	VSS	VSS	VSS	VDDI	VSS	Y
AA	RCTL_P	PAUSE	VDDO	VDDO	VDDI	VDDI	VDDO	VDDO	VDDI	VDDI	AA
AB	RSCLK	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VDDI	AB
AC	VDDO	VDDO	VDDI	VDDI	VDDO	VDDO	VDDI	VDDI	VSS	VDDI	AC
AD	VSS	NC8	VSS	VSS	VSS	VSS	VSS	VDDI	VSS	VSS	AD
AE	NC7	NC10	VDDO	VDDO	VDDI	VDDI	VSS	VDDO	VDDO	VSS	AE
AF	NC9	VSS	VSS	VSS	VSS	VDDI	VSS	VSS	VDDO	VSS	AF
AG	VDDO	VDDO	VDDO	VDDO	VSS	VDDI	VDDI	VSS	VDDO	VDDO	AG
AH	VSS	VSS	VSS	VDDO	VSS	VSS	VDDI	VSS	VSS	VDDO	AH
AJ	VDDI	VDDI	VSS	VDDO	VDDO	VSS	VDDO	VDDO	VSS	VDDI	AJ
AK		VDDI	VSS	VSS	VDDO	VSS	VSS	VDDO	VSS	VDDI	AK
	30	29	28	27	26	25	24	23	22	21	

Figure 6 Ball View 2

	20	19	18	17	16	15	14	13	12	11	
A	DTRB	RSTB	VDDI	VSS	VDDO	VDDO	VSS	VDDI	D[5]	D[9]	A
B	TDAT_P[15]	TSCLK	A[1]	VSS	VDDI	VDDO	VSS	A[10]	D[1]	D[3]	B
C	TDAT_N[13]	TCTL_N	CSB	A[9]	VDDI	VDDI	A[11]	D[2]	D[6]	D[4]	C
D	VSS	TDAT_P[13]	TCTL_P	A[3]	A[0]	A[5]	A[7]	D[0]	ALE	VSS	D
E	VSS	VDDO	INTB	TSTAT[1]	A[2]	A[14]	A[6]	A[12]	VDDO	VSS	E
F	TDAT_P[11]	VDDO	VSS	WRB	TSTAT[0]	A[4]	A[8]	VSS	VDDO	D[15]	F
G	VSS	VSS	VSS	VDDO	RDB	A[13]	VDDO	VSS	D[8]	MDC	G
H	TDAT_P[7]	VSS	VSS	VDDO	VSS	VSS	VDDO	D[11]	MDIO	NC4	H
J	VDDI	VDDI	VDDO	VDDO	VSS	VSS	VDDO	VDDO	VDDI	VDDI	J
K	VSS	VDDI	VDDI	VDDO	VDDO	VDDO	VDDO	VDDI	VDDI	VSS	K
L	VSS	VDDI	VDDI	VDDI	VDDO	VDDO	VDDI	VDDI	VDDI	VSS	L
M	VDDI	VDDI	VSS	VDDI	VDDI	VDDI	VDDI	VSS	VDDI	VDDI	M
N	VDDI	VSS	VSS	VDDI	VDDI	VDDI	VDDI	VSS	VSS	VDDI	N
P	VDDI	VDDI	VDDI	VDDI	VSS	VSS	VDDI	VDDI	VDDI	VDDI	P
R	VDDO	VDDI	VDDI	VSS	VSS	VSS	VSS	VDDI	VDDI	VDDO	R
T	VDDO	VDDI	VDDI	VSS	VSS	VSS	VSS	VDDI	VDDI	VDDO	T
U	VDDI	VDDI	VDDI	VDDI	VSS	VSS	VDDI	VDDI	VDDI	VDDI	U
V	VDDI	VSS	VSS	VDDI	VDDI	VDDI	VDDI	VSS	VSS	VDDI	V
W	VDDI	VDDI	VSS	VDDI	VDDI	VDDI	VDDI	VSS	VDDI	VDDI	W
Y	VSS	VDDI	VDDI	VDDI	VDDO	VDDO	VDDI	VDDI	VDDI	VSS	Y
AA	VSS	VDDI	VDDI	VDDO	VDDO	VDDO	VDDO	VDDI	VDDI	VSS	AA
AB	VDDI	VDDI	VDDO	VDDO	VSS	VSS	VDDO	VDDO	VDDI	VDDI	AB
AC	VDDI	VDDI	VDDO	VDDO	VSS	VSS	VDDO	VDDO	VDDI	VDDI	AC
AD	VDDI	VSS	VSS	VDDO	VDDI	VDDI	VDDO	VSS	VSS	VDDI	AD
AE	VDDO	VDDO	VSS	VDDI	VSS	VSS	VDDI	VSS	VDDO	VDDO	AE
AF	VSS	VDDO	VSS	VDDI	VDDI	VDDI	VDDI	VSS	VDDO	VSS	AF
AG	VSS	VDDO	VSS	VSS	VSS	VSS	VSS	VSS	VDDO	VSS	AG
AH	VSS	VDDO	VDDO	VDDO	VDDI	VDDI	VDDI	VDDO	VDDO	VSS	AH
AJ	VSS	VSS	VSS	VSS	VDDO	VDDI	VSS	VSS	VSS	VSS	AJ
AK	VDDI	VDDI	VDDI	VSS	VDDO	VDDO	VSS	VDDI	VDDI	VDDI	AK
	20	19	18	17	16	15	14	13	12	11	

Figure 7 Ball View 3

	10	9	8	7	6	5	4	3	2	1	
A	D[7]	D[13]	VDDO	VSS	TMS	VCLK4	RX_LOS	PHASE_INIT	VDDI		A
B	D[10]	VSS	VDDO	TRSTB	VCLK3	VCLK_FORCEB	SYNC_ERR	VSS	VDDI	VDDI	B
C	VDDO	VSS	TDI	D[12]	VCLK5	RX_SYSCLK2	VDDO	VSS	VSS	TXDATA4_P[3]	C
D	VDDO	TCK	VCLK1	VCLK6	TX_SYSCLK2	VSS	VDDO	VDDO	TXDATA4_N[3]	TXDATA4_P[1]	D

E	TDO	NC3	NC1	CPREF_CLK	VDDI	VSS	VSS	OIFS_RES	TXDATA4_N[1]	TXDATA3_P[2]	E
F	VCLK2	NC2	NC11	VSS	VDDI	VDDI	OIFS_RESK	TXCLK4_P	TXDATA3_N[2]	TXCLK_SRC4_N	F
G	D[14]	PAUSED	VDDI	VSS	VSS	TXDATA4_P[2]	TXCLK4_N	TXDATA4_P[0]	TXCLK_SRC4_P	VSS	G
H	PHASE_ERR	VSS	VDDI	VDDI	TXDATA4_N[2]	VSS	TXDATA4_N[0]	TXCLK2_P	VDDO	VDDO	H
J	VDDI	VSS	VSS	VSS	VSS	TXDATA3_P[3]	TXCLK2_N	VSS	VSS	TXCLK3_P	J
K	VDDI	VDDI	VSS	AVDHVREF	TXDATA3_N[3]	TXDATA3_P[1]	VDDO	VDDO	TXCLK3_N	TXDATA3_P[0]	K
L	VSS	VDDI	VSS	VSS	TXDATA3_N[1]	VSS	VSS	TXCLK_SRC3_N	TXDATA3_N[0]	TXDATA2_P[3]	L
M	VDDI	VDDI	VSS	VSS	VDDO	VDDO	TXCLK_SRC3_P	TXDATA2_P[2]	TXDATA2_N[3]	TXCLK_SRC2_N	M
N	VDDI	VDDO	VSS	VSS	VSS	TXCLK1_P	TXDATA2_N[2]	TXDATA2_P[1]	TXCLK_SRC2_P	VDDI	N
P	VDDO	VDDO	VDDO	VDDO	TXCLK1_N	TXDATA2_P[0]	TXDATA2_N[1]	TXDATA1_P[3]	VSS	VSS	P
R	VDDO	VSS	VSS	VSS	TXDATA2_N[0]	TXCLK_SRC1_P	TXDATA1_N[3]	VDDI	VDDI	VDDO	R
T	VDDO	VSS	VSS	VSS	TXDATA1_N[2]	TXCLK_SRC1_N	TXDATA1_N[1]	VDDI	VDDO	VDDO	T
U	VDDO	VDDO	VDDO	VDDO	TXDATA1_N[0]	TXDATA1_P[2]	OIFS_ATB[1]	TXDATA1_P[1]	VSS	VSS	U
V	VDDI	VDDO	RXDATA3_P[0]	VSS	VSS	TXDATA1_P[0]	RXDATA4_P[1]	OIFS_ATB[0]	RXDATA4_P[3]	VDDI	V
W	VDDI	VDDI	RXDATA3_P[1]	RXDATA3_N[0]	VDDO	VDDO	RXDATA3_P[3]	RXDATA4_N[1]	RXDATA4_P[0]	RXDATA4_N[3]	W
Y	VSS	VDDI	VSS	RXDATA3_N[1]	RXDATA3_P[2]	VSS	VSS	RXDATA3_N[3]	RXDATA4_P[2]	RXDATA4_N[0]	Y
AA	VDDI	VDDI	VSS	VSS	RXCLK3_P	RXDATA3_N[2]	VDDO	VDDO	RXCLK4_P	RXDATA4_N[2]	AA
AB	VDDI	VSS	VSS	VSS	RXDATA2_P[1]	RXCLK3_N	RXCLK2_P	VSS	VSS	RXCLK4_N	AB
AC	VDDI	VSS	VDDI	VDDI	RXDATA1_P[0]	RXDATA2_N[1]	RXDATA2_P[2]	RXCLK2_N	VDDO	VDDO	AC
AD	VSS	VSS	VDDI	VSS	VSS	RXDATA1_N[0]	RXCLK1_P	RXDATA2_N[2]	RXDATA2_P[3]	VSS	AD
AE	VSS	VDDO	VDDO	VSS	VDDI	VDDI	RXDATA1_P[2]	RXCLK1_N	RXDATA2_P[0]	RXDATA2_N[3]	AE
AF	VSS	VDDO	VSS	VSS	VDDI	VSS	VSS	RXDATA1_N[2]	RXDATA1_P[3]	RXDATA2_N[0]	AF
AG	VDDO	VDDO	VSS	VDDI	VDDI	VSS	VDDO	VDDO	RXDATA1_P[1]	RXDATA1_N[3]	AG
AH	VDDO	VSS	VSS	VDDI	VSS	VSS	VDDO	VSS	VSS	RXDATA1_N[1]	AH
AJ	VDDI	VSS	VDDO	VDDO	VSS	VDDO	VDDO	VSS	VDDI	VDDI	AJ
AK	VDDI	VSS	VDDO	VSS	VSS	VDDO	VSS	VSS	VDDI		AK

9 Pin Description

Table 2 S/UNI-1x10GE Line Side Signaling Interface

Pin Name	Type	Pin No.	Function
RXCLK2+ RXCLK2-	Analog LVDS Input	AB4 AC3	**Receive Clock 2** RXCLK2 is 644.53125MHz clock. RXCLK2 is used to sample the 16 receive data steams at rising edge. The RXCLK2 pins correspond to PMA_RX_CLK +/- in the IEEE 802.3ae standard
RXCLK4+ RXCLK4- RXCLK3+ RXCLK3- RXCLK1+ RXCLK1-	Analog LVDS Input	AA2 AB1 AA6 AB5 AD4 AE3	**Receive Clock [4, 3, 1] UNUSED** The UNUSED RXCLK[4,3,1] inputs need to have the Neg (-) pins tied to 3.3v through a 4.7k ohm resistor. The Pos (+) pins must be tied to ground through a zero ohm resistor.
RXDATA1[0]+ RXDATA1[0]- RXDATA1[1]+ RXDATA1[1]- RXDATA1[2]+ RXDATA1[2]- RXDATA1[3]+ RXDATA1[3]- RXDATA2[0]+ RXDATA2[0]- RXDATA2[1]+ RXDATA2[1]- RXDATA2[2]+ RXDATA2[2]- RXDATA2[3]+ RXDATA2[3]- RXDATA3[0]+ RXDATA3[0]- RXDATA3[1]+ RXDATA3[1]- RXDATA3[2]+ RXDATA3[2]- RXDATA3[3]+ RXDATA3[3]- RXDATA4[0]+ RXDATA4[0]- RXDATA4[1]+ RXDATA4[1]- RXDATA4[2]+ RXDATA4[2]- RXDATA4[3]+ RXDATA4[3]-	Analog LVDS Input	AC6 AD5 AG2 AH1 AE4 AF3 AF2 AG1 AE2 AF1 AB6 AC5 AC4 AD3 AD2 AE1 V8 W7 W8 Y7 Y6 AA5 W4 Y3 W2 Y1 V4 W3 Y2 AA1 V2 W1	The **Differential Receive Data** (RXDATA) inputs carry the byte-serial 10Gigabit Ethernet stream. Pin RXDATA1[0] corresponds to rx_data_unit<0> while RXDATA4[3] corresponds to rx_data_unit<15> as described in figure 49-2 of IEEE802.3ae standard.

Pin Name	Type	Pin No.	Function
TXCLK_SRC2+ TXCLK_SRC2-	Analog LVDS Input	N2 M1	Transmit Source Clock **TXCLK_SRC2** is a 644.53125MHz reference clock from SERDES. All 16 transmit data streams TXDATAi_j (i=1 to 4 & j=1 to 4) are transmitted with reference to TXCLK_SRC2. The TXCLK_SRC2 pins correspond to PMA_TXCLK_SRC +/- in the IEEE 802.3ae standard.
TXCLK_SRC4+ TXCLK_SRC4- TXCLK_SRC3+ TXCLK_SRC3- TXCLK_SRC1+ TXCLK_SRC1-	Analog LVDS Input	G2 F1 M4 L3 R5 T5	Transmit Source Clock **TXCLK_SRC[4, 3, 1]** UNUSED The UNUSED TXCLK_SRC[4,3,1] inputs need to have the Neg (-) pins tied to 3.3v through a 4.7k ohm resistor. The Pos (+) pins must be tied to ground through a zero ohm resistor.
TXCLK2+ TXCLK2-	Analog LVDS Output	H3 J4	Transmit Clock **TXCLK2** is a 644.53125MHz clock to synchronize the transmission of the 16 transmit data streams. The TXCLK2 pins correspond to PMA_TX_CLK +/- in the IEEE 802.3ae standard. The TXCLK2+/- LVDS outputs are internally terminated by the PM3392.
TXCLK4+ TXCLK4- TXCLK3+ TXCLK3- TXCLK1+ TXCLK1-	Analog LVDS Output	F3 G4 J1 K2 N5 P6	Transmit Clock **TXCLK[4,3,1]** are UNUSED. The TXCLK2+/- LVDS outputs are internally terminated by the PM3392.

Pin Name	Type	Pin No.	Function
TXDATA1[0]+ TXDATA1[0]- TXDATA1[1]+ TXDATA1[1]- TXDATA1[2]+ TXDATA1[2]- TXDATA1[3]+ TXDATA1[3]- TXDATA2[0]+ TXDATA2[0]- TXDATA2[1]+ TXDATA2[1]- TXDATA2[2]+ TXDATA2[2]- TXDATA2[3]+ TXDATA2[3]- TXDATA3[0]+ TXDATA3[0]- TXDATA3[1]+ TXDATA3[1]- TXDATA3[2]+ TXDATA3[2]- TXDATA3[3]+ TXDATA3[3]- TXDATA4[0]+ TXDATA4[0]- TXDATA4[1]+ TXDATA4[1]- TXDATA4[2]+ TXDATA4[2]- TXDATA4[3]+ TXDATA4[3]-	Analog LVDS Output	V5 U6 U3 T4 U5 T6 P3 R4 P5 R6 N3 P4 M3 N4 L1 M2 K1 L2 K5 L6 E1 F2 J5 K6 G3 H4 D1 E2 G5 H6 C1 D2	The **Differential Transmit Data** (TXDATA) outputs carries the byte-serial 10gigabit Ethernet stream. Pin TXDATA1[0] corresponds to tx_data_unit<0> while TXDATA4[3] corresponds to tx_data_unit<15> as described in figure 49-2 of IEEE802.3ae standard The TXDATA+/- LVDS outputs are internally terminated by the PM3392.
PHASE_ERR	Input CMOS	H10	The **Phase Error** (PHASE_ERR) input indicates when the TXCLK2+/- output is not aligned with the corresponding TXDATA+/- bus. When asserted, the receiving line side device cannot use the source synchronous TXCLK2 to sample the TXDATA bus. PHASE_ERR is treated as an asynchronous signal and is used to trigger maskable interrupt. In addition, the associated PHASE_INIT output should be asserted to reinitiate alignment under user control.
SYNC_ERR	Input CMOS Tri-state	B4	The **Synchronization Error** (SYNC_ERR) input indicates that RXDATA bus can be safely sampled. When SYNC_ERR is high, RXDATA+/- is not derived from the optical line and is suspect. When SYNC_ERR is low, RXDATA is recovered from the optical stream. The SYNC_ERR signal is treated as an asynchronous input.

Pin Name	Type	Pin No.	Function
PHASE_INIT	Output CMOS Tri-state	A3	The **Phase Initialization** (PHASE_INIT) output indicates to the receiving device that the device should start the TXCLK[2]+/- and TXDATA[N]+/- alignment process. The PHASE_INIT output is driven by a top level register programmed by software.
PAUSE	Input CMOS Internal Pull Down	AA29	The **Pause** (PAUSE) signal is controlled by the system level interface to request the internal transmit MAC to send pause frames through the line side to initiate flow control. Pause frames are sent at a prescribed interval programmed in the pause timer register. PAUSE is active high and is treated as an asynchronous input.
PAUSED	Output CMOS Tri-state	G9	The **Paused** (PAUSED) signal is controlled by the transmit MAC to indicate the MAC is currently acting upon an ingressed pause frame as indicated from the receive MAC. PAUSED is active high and is treated as an asynchronous output.

Table 3 S/UNI-1x10GE System Side Signaling Interface

Pin Name	Type	Pin No.	Function
REFSEL[1] REFSEL[0]	Input CMOS Schmitt Trigger Internal Pull Up	U26 T25	The **POS-PHY Level 4 Reference Clock Select** determines the source and frequency of the reference clock used to generate the internal clocks for the PL4 Bus interface logic on the PM3392. When REFSEL[0] is low the reference clock is derived from TDCLK+/- (slave mode). When REFSEL[0] is high the reference clock is derived from REFCLK+/- (master mode). When REFSEL[1] is low in slave mode the TRAIN_DIS and ODAT_DIS bits in the PL4IO Configuration register are initially cleared and TDCLK+/- must be valid before de-asserting RSTB. When REFSEL[1] is high in slave mode the TRAIN_DIS and ODAT_DIS bits in the PL4IO Configuration register are initially set and TDCLK+/- must be valid before these bits are cleared. When REFSEL[1] is low in master mode the selected PM3392 internal PL4 reference clock frequency is ½ the PL4 data rate. When REFSEL[1] is high in master mode the selected reference clock frequency is ¼ the PL4 data rate. REFSEL should only be switched while RSTB is asserted.

Pin Name	Type	Pin No.	Function
PL4_RCLK+ PL4_RCLK-	Differential PECL Input	G26 H25	The **POS-PHY Level 4 Reference** is an optional 155.5 MHz to 350 MHz, 40-60% duty cycle clock reference which is used to generate RDCLK+/- and provide phase references for TDAT[15:0]+/- de-skewing. PL4_RCLK is ignored when REFSEL[0] is logic 0 (slave mode) and used when REFSEL[0] is logic 1 (master mode). The actual frequency of PL4_RCLK in master mode must be between 311.0 MHz and 350 MHz when REFSEL[1] is low and is frequency locked to TDCLK+/-. The actual frequency of PL4_RCLK in master mode must be between 155.5 MHz and 175 MHz when REFSEL[1] is high and is frequency locked to TDCLK+/- divided by 2. PL4_RCLK+/- should be stable before de-asserting RSTB if REFSEL[0] is a logic 1 (master mode). If the POS-PHY Level 4 Reference clock is used, it is recommended to use PL4_RCLK as a differential PECL input. The PL4_RCLK+/- PECL inputs are not internally terminated by the PM3392.
RDCLK+ RDCLK-	Analog LVDS Output	M28 N27	The **POS-PHY Differential Receive Clock** (RDCLK+/-) is a 1*PL4_RCLK, 2*PL4_RCLK or TDCLK MHz, 45-55% duty cycle source synchronous clock. When operating in PL4 master mode, RDCLK+/- is a 1x or 2x multiplied version of the PL4_RCLK+/- inputs. When operating in PL4 slave mode, RDCLK+/- is a loop-timed version of TDCLK+/-. RDCLK+/- is provided to downstream devices to clock in RDAT[15:0]+/- and RCTL+/-. The rising and falling edges of RDCLK+/- are used to update RDAT[15:0]+/- and RCTL+/-. The RDCLK+/- LVDS outputs are internally terminated by the PM3392.

Pin Name	Type	Pin No.	Function
RDAT[15]+ RDAT[15]- RDAT[14]+ RDAT[14]- RDAT[13]+ RDAT[13]- RDAT[12]+ RDAT[12]- RDAT[11]+ RDAT[11]- RDAT[10]+ RDAT[10]- RDAT[9]+ RDAT[9]- RDAT[8]+ RDAT[8]- RDAT[7]+ RDAT[7]- RDAT[6]+ RDAT[6]- RDAT[5]+ RDAT[5]- RDAT[4]+ RDAT[4]- RDAT[3]+ RDAT[3]- RDAT[2]+ RDAT[2]- RDAT[1]+ RDAT[1]- RDAT[0]+ RDAT[0]-	Analog LVDS Output	Y30 W29 V28 U27 U28 T27 W30 V29 P26 R25 P28 R27 N28 P27 M30 N29 L30 M29 K30 L29 N26 P25 L28 M27 J30 K29 K26 L25 H28 J27 F30 G29	The PL4 **Differential Receive Data** (RDAT[15:0]+/-) bus carries the Ethernet frame data that are read from the ingress FIFO and the in-band control words which describes the stream. In-band control words are identified using the RCTL+/- output. Please refer to the Operations section for a description of the POS-PHY Level 4 protocol and the bus data structures. RDAT[15:0]+/- is updated on both edges of RDCLK+/-. The RDAT[15:0]+/- LVDS outputs are internally terminated by the PM3392.
RCTL+ RCTL-	Analog LVDS Output	AA30 Y29	The PL4 **Differential Receive Control** (RCTL+/-) signals identify control words on the RDAT[15:0]+/- bus. When RCTL+/- is high, a control word is on the RDAT[15:0]+/- bus. If RCTL+/- is low, a payload word is on the RDAT[15:0]+/- bus. RCTL+/- is updated on both edges of RDCLK+/-. The RCTL+/- LVDS outputs are internally terminated by the PM3392.

Pin Name	Type	Pin No.	Function
RSCLK	Input 3V CMOS Schmitt Trigger	AB30	The PL4 **Receive Status Clock** (RSCLK) is the optional FIFO status input clock supplied by the peer PL4 device and is used to transfer FIFO status from the peer PL4 device to another device. The actual frequency of RSCLK is determined by the peer device. The maximum frequency of RSCLK is at 1/8 of the PL4 data transfer rate (RDCLK/4 MHz). Conforming to the PL4 Bus specification requirement that implementation of the FIFO status channel in the receive interface is optional, RSCLK may be tied to a logic 0 if the Receive Status bus is not used. Refer to the Operations section for additional details (subsection Operation with PL4 Receive FIFO status unimplemented).
RSTAT[1] RSTAT[0]	Input CMOS	W27 Y28	The PL4 **Receive Status** (RSTAT[1:0]) bus is used to indicate the status of the downstream device's FIFO. For the FIFO, a Satisfied, Hungry or Starving condition can be indicated. In addition, a special two-bit code is used for framing and alignment. Please refer to the Operation section for the status definitions and the status protocol. RSTAT[1:0] is sampled on the rising edge of RSCLK. Conforming to the PL4 Bus specification requirement that implementation of the FIFO status channel in the receive interface is optional, RSTAT may be tied to a logic 0 if the Receive Status bus is not used. Refer to the Operations section for additional details (subsection Operation with PL4 Receive FIFO status unimplemented).
TDCLK+ TDCLK-	Analog LVDS Input	B25 A26	The PL4 **Differential Transmit Clock** (TDCLK+/-) is a 311 MHz to 350 MHz, 45-55% duty cycle source synchronous clock. TDCLK+/- is used to clock the transmit POS-PHY circuitry when in slave mode. When configured for PL4 master mode, TDCLK+/- must be frequency locked to the PL4_RCLK+/-. LVDS inputs are internally terminated to 100-ohm differential impedance.

Pin Name	Type	Pin No.	Function
TDAT[15]+ TDAT[15]- TDAT[14]+ TDAT[14]- TDAT[13]+ TDAT[13]- TDAT[12]+ TDAT[12]- TDAT[11]+ TDAT[11]- TDAT[10]+ TDAT[10]- TDAT[9]+ TDAT[9]- TDAT[8]+ TDAT[8]- TDAT[7]+ TDAT[7]- TDAT[6]+ TDAT[6]- TDAT[5]+ TDAT[5]- TDAT[4]+ TDAT[4]- TDAT[3]+ TDAT[3]- TDAT[2]+ TDAT[2]- TDAT[1]+ TDAT[1]- TDAT[0]+ TDAT[0]-	Analog LVDS Input	B20 A21 B21 A22 D19 C20 D22 C23 F20 E21 F21 E22 B24 A25 D23 C24 H20 G21 F22 E23 D24 C25 B26 A27 F23 E24 H21 G22 B27 A28 D25 C26	The PL4 **Differential Transmit Data** (TDAT[15:0]+/-) bus carries Ethernet frame data that are written into the egress FIFO and the in-band control words which describes the stream. In-band control words are identified using the TCTL+/- input. Please refer to the Operations section for a description of the POS-PHY Level 4 protocol and the bus data structures. TDAT[15:0]+/- is sampled on both edges of TDCLK+/-.
TCTL+ TCTL-	Analog LVDS Input	D18 C19	The PL4 **Differential Transmit Control** (TCTL+/-) signals identify control words on the TDAT[15:0]+/- bus. When TCTL+/- is high, a control word is on the TDAT[15:0]+/- bus. If TCTL+/- is low, a payload word is on the TDAT[15:0]+/- bus. TCTL+/- is sampled on both edges of TDCLK+/-.
TSTAT[1] TSTAT[0]	Output CMOS Tri-state	E17 F16	The PL4 **Transmit Status** (TSTAT[1:0]) bus is used to indicate the status of the S/UNI-1x10GE egress FIFOs. A Satisfied, Hungry or Starving condition can be indicated. Please refer to the Operation section for the status definitions and the status protocol. TSTAT[1:0] is updated on the rising edge of TSCLK.

Pin Name	Type	Pin No.	Function
TSCLK	Output CMOS Tri-state	B19	The PL4 **Transmit Status Clock** (TSCLK) is a TDCLK/4 MHz, 40/60% duty cycle source synchronous clock. TSCLK is used to clock the transmit status circuitry. The rising edge of TSCLK is used to update TSTAT[1:0].

Table 4 ECBI Signals and MDC/MDIO

Pin Name	Type	Pin No.	Function
CSB	Input CMOS	C18	The active low **Chip Select** (CSB) signal is low during S/UNI-1x10GE register accesses. If CSB is not required (i.e. register accesses controlled using the RDB and WRB signals only), CSB must be connected to an inverted version of the RSTB input.
RDB	Input CMOS	G16	The active low **Read Enable** (RDB) signal is low during a S/UNI-1x10GE read access. The S/UNI-1x10GE drives the D[15:0] bus with the contents of the addressed register while RDB and CSB are low.
WRB	Input CMOS	F17	The active low **Write Strobe** (WRB) signal is low during a S/UNI-1x10GE register write access. The D[15:0] bus contents are clocked into the addressed register on the rising WRB edge while CSB is low.
D[15] D[14] D[13] D[12] D[11] D[10] D[9] D[8] D[7] D[6] D[5] D[4] D[3] D[2] D[1] D[0]	Bi-Directional CMOS	F11 G10 A9 C7 H13 B10 A11 G12 A10 C12 A12 C11 B11 C13 B12 D13	The bi-directional **Data Bus** (D[15:0]) is used during S/UNI-1x10GE read and write accesses.
A[14] (TRS)	Input CMOS	E15	The **Test Register Select** (TRS) signal selects between normal and test mode register accesses. TRS is high during test mode register accesses, and is low during normal mode register accesses. TRS may be tied low.

Pin Name	Type	Pin No.	Function
A[13] A[12] A[11] A[10] A[9] A[8] A[7] A[6] A[5] A[4] A[3] A[2] A[1] A[0]	Input CMOS	G15 E13 C14 B13 C17 F14 D14 E14 D15 F15 D17 E16 B18 D16	The **Address** (A[13:0]) bus selects specific registers during S/UNI-1x10GE register accesses.
RSTB	Input CMOS Schmitt Trigger Internal Pull Up	A19	The active low **Reset** (RSTB) signal provides an asynchronous S/UNI-1x10GE reset. RSTB is a Schmitt triggered input with an integral pull-up resistor.
ALE	Input CMOS Internal Pull Up	D12	The **Address Latch Enable** (ALE) is an active-high signal and latches the address bus A[13:0] when low. When ALE is high, the internal address latches are transparent. It allows the S/UNI-1x10GE to interface to a multiplexed address/data bus. The ALE input has an integral pull up resistor.
INTB	Output CMOS Tri-state	E18	The active low **Interrupt** (INTB) is set low when an S/UNI-1x10GE enabled interrupt source is active. The S/UNI-1x10GE may be enabled to report many alarms or events via interrupts. INTB is tri-stated when the interrupt is acknowledged via the appropriate register access. INTB is an open drain output.
MDC	Output CMOS Tri-state	G11	**Management Data Clock** (MDC) output signal is approximately 2.5MHz with a 50% duty cycle clock for the MDIO.
MDIO	Bi-directional CMOS Internal Pull Down	H12	**Management Data IO** (MDIO), is a bi-directional signal that send and receives status data from an external MII PHY. The output management frame bit serial signal is updated on the rising edge of MDC. Input bit serial status data is sampled on the rising edge of MDC. MDIO has an internal pull down resistor.

Table 5 JTAG and Reserved Signals

Pin Name	Type	Pin No.	Function
TCK	Input Schmitt Trigger CMOS internal Pull Up	D9	The **Test Clock** (TCK) signal provides timing for test operations that can be carried out using the IEEE P1149.1 test access port. TCK has an integral pull up resistor.
TMS	Input Schmitt Trigger CMOS Internal Pull Up	A6	The **Test Mode Select** (TMS) signal controls the test operations that can be carried out using the IEEE P1149.1 test access port. TMS is sampled on the rising edge of TCK. TMS has an integral pull up resistor.
TDI	Input Schmitt Trigger CMOS Internal Pull Up	C8	When the S/UNI-1x10GE is configured for JTAG operation, the **Test Data Input** (TDI) signal carries test data into the S/UNI-1x10GE via the IEEE P1149.1 test access port. TDI is sampled on the rising edge of TCK. TDI has an integral pull up resistor.
TDO	Output CMOS Tri-state	E10	The **Test Data Output** (TDO) signal carries test data out of the S/UNI-1x10GE via the IEEE P1149.1 test access port. TDO is updated on the falling edge of TCK. TDO is a tri-state output which is inactive except when scanning of data is in progress.
TRSTB	Input Schmitt Trigger CMOS Internal Pull Up	B7	The active low **Test Reset** (TRSTB) signal provides an asynchronous S/UNI-1x10GE test access port reset via the IEEE P1149.1 test access port. TRSTB is a Schmidt triggered input with an integral pull up resistor. In the event that TRSTB is not used, it must be connected to RSTB.

Pin Name	Type	Pin No.	Function
DTRB	Input Schmitt Trigger CMOS Internal Pull Down	A20	**Digital-core Timer Reset Bypass** (DTRB**)** is used to override the internal digital-core timer reset signal from the PL4 sub-system. The timer reset allows the analog circuitry (specifically clock recovery or generation) to stabilize before the digital-core logic is release from the reset state for a more controlled initialization. DTRB is logically ORed to the internal digital-core timer reset signal. DTRB must be asserted when RSTB is asserted. When RSTB is de-asserted, DTRB can then be de-asserted with a delay of up to 10ms. When DTRB is de-asserted before the internal timer reset expires, the digital-core timer reset is overridden by DTRB. DTRB is reserved for PMC test purposes only. DTRB is a Schmitt Trigger input with an internal pull down. It is required that this pin be strapped low for normal operation.
CPREF_CLK	Input Schmitt Trigger CMOS	E7	Chopper Reference Clock, **CPREF_CLK**, is a 60-81MHz clock with a 40/60 duty cycle or better, input that must be stable before the deassertion of pin reset.
VCLK1 VCLK2 VCLK3 VCLK4 VCLK5 VCLK6	Input MOS	D8 F10 B6 A5 C6 D7	**VCLK[N]** is reserved for PMC test purposes. VCLK[N] must be strapped low for normal operation.
VCLK_FORCEB	Input CMOS	B5	**VCLK_FORCEB** is used in conjunction with VCLK[N]. VCLK_FORCEB is reserved for PMC test purposes only and must be strapped high for normal operation.
TX_SYSCLK2	Output MOS Tri-state	D6	**TX_SYSCLK2** is used for diagnostic purpose only, these should be left as no connects externally.
RX_SYSCLK2	Output MOS Tri-state	C5	**RX_SYSCLK2** is used for diagnostic purpose only, these should be left as no connects externally.
RX_LOS	Output MOS Tri-state	A4	**RX_LOS** is used for diagnostic purpose only, these should be left as no connects externally.

Pin Name	Type	Pin No.	Function
NC1 NC2 NC3 NC4 NC5 NC6 NC7 NC8 NC9 NC10 NC11	Input MOS Internal Pull Up	E8 F9 E9 H11 V27 W28 AE30 AD29 AF30 AE29 F8	Reserved, **NC1-11** are no connects externally.

Table 6 S/UNI-1x10GE Analog Pins

Pin Name	Type	Pin No.	Function
PL4_RES PL4_RESK	Differential Analog Bi-Directional	F29 E30	**PL4 Reference Resistor Connection**. An off-chip 3.16kΩ ±1% resistor is connected between the positive resistor reference pin PL4_RES and a Kelvin ground contact PL4_RESK for the TXLVREF circuitry. An on-chip negative feedback path will force an internal 0.80V reference voltage onto PL4_RES, therefore forcing 252μA of current to flow through the resistor. This current is used by the transmitter blocks.
PL4_AVDH[3] PL4_AVDH[2] PL4_AVDH[1] PL4_AVDH[0]	Analog Power	J26 C30 D29 F27	The **PL4 Analog Power High** (PL4_AVDH[0:3]) pins for the PL4 interface analog circuits. The PL4_AVDH[0:3] pins should be connected through passive filtering networks to a well decoupled +3.3V analog power supply. See operation section for detailed information.
PL4_AVDL[3] PL4_AVDL[2] PL4_AVDL[1]	Analog Power	E28 F28 G27	The **PL4 Analog Power Low** (PL4_AVDL[1:3]) pins for the PL4 interface analog circuits. PL4_AVDL[1:3] pins should be connected through passive filtering networks to a well decoupled +1.8V analog power supply. See operation section for detailed information.
PL4_ATB[5] PL4_ATB[4] PL4_ATB[3] PL4_ATB[2] PL4_ATB[1] PL4_ATB[0]	Analog Bi-Directional	H26 J25 D30 H27 E29 G28	The **PL4 Analog Test Pins** (PL4_ATB[5:0] are provided for production testing only. These pins must be tied to analog ground (VSS) during normal operation.

Pin Name	Type	Pin No.	Function
OIFS_RES OIFS_RESK	Differential Analog Bi-Directional	E3 F4	**OIFS Reference Resistor Connection**. An off-chip 3.16kΩ ±1% resistor is connected between the positive resistor reference pin OIFS_RES and a Kelvin ground contact OIFS_RESK for the TXLVREF circuitry. An on-chip negative feedback path will force an internal 0.80V reference voltage onto OIFS_RES, therefore forcing 252μA of current to flow through the resistor. This current is used by the transmitter blocks.
AVDHVREF	Analog Reference	K7	**AVDHVREF** is the quiet TXLVREF analog power (AVDHVREF) is a +3.3V power supply for the quiet analog blocks. This pin is de-coupled to VSS via an on-chip capacitor and is also de-coupled externally to GROUND via a 0.1uF ceramic de-coupling capacitor for proper HF noise shunting.
OIFS_ATB[1] OIFS_ATB[0]	Analog Bi-Directional	U4 V3	The **OIFS Analog Test Pins** (OIFS_ATB[1:0] are provided for production testing only. These pins must be tied to analog ground (VSS) during normal operation.

Table 7 Digital Power and Ground

Pin Name	Pin Type	PIN No.	Function
VDDO	Digital I/O Power	A15 A16 A23 A8 AA14 AA15 AA16 AA17 AA23 AA24 AA27 AA28 AA3 AA4 AB13 AB14 AB17 AB18 AC1 AC13 AC14 AC17 AC18 AC2	The **Digital I/O Power** (VDDO) pins should be connected to a well-decoupled +3.3V digital power supply.

Pin Name	Pin Type	PIN No.	Function
		AC25 AC26 AC29 AC30 AD14 AD17 AE11 AE12 AE19 AE20 AE22 AE23 AE27 AE28 AE8 AE9 AF12 AF19 AF22 AF9 AG10 AG12 AG19 AG21 AG22 AG27 AG28 AG29 AG3 AG30 AG4 AG9 AH10 AH12 AH13 AH17 AH18 AH19 AH21 AH27 AH4 AJ16 AJ23 AJ24 AJ26 AJ27 AJ4 AJ5 AJ7 AJ8 AK15 AK16 AK23 AK26 AK5 AK8	

Pin Name	Pin Type	PIN No.	Function
		B15 B23 B8 C10 C21 C27 C4 D10 D21 D27 D28 D3 D4 E12 E19 F12 F19 G14 G17 H1 H14 H17 H2 H29 H30 J13 J14 J17 J18 K14 K15 K16 K17 K23 K24 K27 K28 K3 K4 L15 L16 M25 M26 M5 M6 N22 N9 P10 P21 P22 P23 P24 P7 P8 P9 R1	

Pin Name	Pin Type	PIN No.	Function
		R10 R11 R20 R21 R29 R30 T1 T10 T11 T2 T20 T21 T30 U10 U21 U22 U23 U24 U7 U8 U9 V22 V9 W25 W26 W5 W6 Y15 Y16	
VDDI	Digital Core Power	A13 A18 A2 A29 AA10 AA12 AA13 AA18 AA19 AA21 AA22 AA25 AA26 AA9 AB10 AB11 AB12 AB19 AB20 AB21 AC10 AC11 AC12 AC19 AC20 AC21 AC23	The **Digital Core Power** (VDDI) pins should be connected to a well-decoupled +1.8V digital power supply.

Pin Name	Pin Type	PIN No.	Function
		AC24 AC27 AC28 AC7 AC8 AD11 AD15 AD16 AD20 AD23 AD8 AE14 AE17 AE25 AE26 AE5 AE6 AF14 AF15 AF16 AF17 AF25 AF6 AG24 AG25 AG6 AG7 AH14 AH15 AH16 AH24 AH7 AJ1 AJ10 AJ15 AJ2 AJ21 AJ29 AJ30 AK10 AK11 AK12 AK13 AK18 AK19 AK2 AK20 AK21 AK29 B1 B16 B2 B29 B30 C15 C16	

Pin Name	Pin Type	PIN No.	Function
		E25 E6 F25 F26 F5 F6 G23 G8 H23 H24 H7 H8 J10 J11 J12 J19 J20 J21 K10 K12 K13 K18 K19 K21 K22 K9 L12 L13 L14 L17 L18 L19 L22 L9 M10 M11 M12 M14 M15 M16 M17 M19 M20 M21 M22 M23 M24 M9 N1 N10 N11 N14 N15 N16 N17 N20	

Pin Name	Pin Type	PIN No.	Function
		N21 N30 P11 P12 P13 P14 P17 P18 P19 P20 R12 R13 R18 R19 R2 R28 R3 T12 T13 T18 T19 T28 T29 T3 U11 U12 U13 U14 U17 U18 U19 U20 V1 V10 V11 V14 V15 V16 V17 V20 V21 V30 W10 W11 W12 W14 W15 W16 W17 W19 W20 W21 W22 W23 W24 W9	

Pin Name	Pin Type	PIN No.	Function
		Y12 Y13 Y14 Y17 Y18 Y19 Y22 Y9	
VSS	Digital / Analog Ground	A14 A17 A24 A7 AA11 AA20 AA7 AA8 AB15 AB16 AB2 AB22 AB23 AB24 AB25 AB26 AB27 AB28 AB29 AB3 AB7 AB8 AB9 AC15 AC16 AC22 AC9 AD1 AD10 AD12 AD13 AD18 AD19 AD21 AD22 AD24 AD25 AD26 AD27 AD28 AD30 AD6 AD7 AD9 AE10 AE13 AE15 AE16	The **Digital/Analog Ground** (VSS) pins should be connected to the common digital/analog ground of the device power supply.

Pin Name	Pin Type	PIN No.	Function
		AE18 AE21 AE24 AE7 AF10 AF11 AF13 AF18 AF20 AF21 AF23 AF24 AF26 AF27 AF28 AF29 AF4 AF5 AF7 AF8 AG11 AG13 AG14 AG15 AG16 AG17 AG18 AG20 AG23 AG26 AG5 AG8 AH11 AH2 AH20 AH22 AH23 AH25 AH26 AH28 AH29 AH3 AH30 AH5 AH6 AH8 AH9 AJ11 AJ12 AJ13 AJ14 AJ17 AJ18 AJ19 AJ20 AJ22	

Pin Name	Pin Type	PIN No.	Function
		AJ25 AJ28 AJ3 AJ6 AJ9 AK14 AK17 AK22 AK24 AK25 AK27 AK28 AK3 AK4 AK6 AK7 AK9 B14 B17 B22 B28 B3 B9 C2 C22 C28 C29 C3 C9 D11 D20 D26 D5 E11 E20 E26 E27 E4 E5 F13 F18 F24 F7 G1 G13 G18 G19 G20 G24 G25 G30 G6 G7 H15 H16 H18	

Pin Name	Pin Type	PIN No.	Function
		H19 H22 H5 H9 J15 J16 J2 J22 J23 J24 J28 J29 J3 J6 J7 J8 J9 K11 K20 K25 K8 L10 L11 L20 L21 L23 L24 L26. L27 L4 L5 L7 L8 M13 M18 M7 M8 N12 N13 N18 N19 N23 N24 N25 N6 N7 N8 P1 P15 P16 P2 P29 P30 R14 R15 R16	

Pin Name	Pin Type	PIN No.	Function
		R17 R22 R23 R24 R26 R7 R8 R9 T14 T15 T16 T17 T22 T23 T24 T26 T7 T8 T9 U1 U15 U16 U2 U25 U29 U30 V12 V13 V18 V19 V23 V24 V25 V26 V6 V7 W13 W18 Y10 Y11 Y20 Y21 Y23 Y24 Y25 Y26 Y27 Y4 Y5 Y8	

Notes on Pin Description:

1. All S/UNI-1x10GE inputs and bi-directional pins present minimum capacitive loading and operate at CMOS/TTL logic levels except the LVDS signals listed above in the Pin Description, which operate at LVDS levels.

2. The S/UNI-1x10GE digital outputs are 3.3V tolerant.
3. Inputs ALE, RSTB, TMS, TDI, TCK and TRSTB have internal pull-up resistors.
4. The LVDS inputs and outputs should be terminated in a passive network and interface at LVDS levels as described in the Operations section.
5. It is mandatory that every ground pin be connected to the printed circuit board ground plane to ensure reliable device operation.
6. It is mandatory that every power pin be connected to the printed circuit board power plane to ensure reliable device operation.
7. All analog power and ground pins can be sensitive to noise. They must be isolated from the digital power and ground. Care must be taken to correctly decouple these pins. Please refer to the XENON Power Supply Filtering Recommendation Application Note. PMC-2010770.
8. Due to ESD protection structures in the pads, it is necessary to exercise caution when powering a device up or down. ESD protection devices behave as diodes between power supply pins and from I/O pins to power supply pins. Under extreme conditions it is possible to damage these ESD protection devices or trigger latch up. Please adhere to the recommended power supply sequencing as described in the Operation section of this document.
9. Do not exceed 100 mA of current on any signal pin during the power-up or power-down sequence. Refer to the Power Sequencing description in the Operations section.
10. Before any input activity occurs, ensure that the device power supplies are within their nominal voltage range.
11. Hold the device in the reset condition until the device power supplies are within their nominal voltage range.

10 Functional Description

10.1 Primary Interfaces, Line Side and System Side

10.1.1 OIFS and OIFS Digital Wrapper, Receive and Transmit Functionality

The OIFS performs the XSBI functionality as prescribed by 10 Gigabit IEEE 802.3ae standard compatible Ethernet LAN physical layer devices (LAN PHY). The block performs the conversion from channelized serial data to an internal parallel data stream for the ingress channel and also converts the internal parallel data stream to the resultant channelized serial bit stream for the egress channel.

The OIFS Digital Wrapper also supplies the loopback features for this device. Both line side and system side loopback can be configured. All relevant synchronization logic is contained in this block.

10.1.2 XSBI Line Interface

The XSBI Line Interface block integrates Analog Block Components (ABC), along with supporting logic to implement a 16-bit interface (XSBI compliant) between the Physical Coding Sub-layer (PCS) and the Physical Media Device (PMD).

On the receive direction, the XSBI reorders the sixteen 8-bit parallel data coming from the OIFS (16 1:8 SIPO which de-serializes a 644.53125 Mbps bit serial data stream into an 80.63 Mbps byte serial data stream) and provide a 128-bit system side interface to the PCS Sub-layer. The bit ordering (big-endian or little-endian) within each 16-bit is programmable.

On the transmit direction, the XSBI Line Interface takes a 128-bits of line data input from the PCS sub-layer and reorders it into sixteen 8-bit data to be provided to the sixteen 8:1 PISO (each PISO converts an 80.63 Mbps byte serial data stream into a 644.53125 Mbps bit serial data stream).

The interface also provides PRBS (pseudo random bit sequencing) testing capability for the receive and transmit blocks within the XSBI. When in test mode on the receive side it monitors the incoming data sequence for errors, and on the transmit side it sends a Pseudo Random Bit Sequence. For the XSBI, there is no difference in operation between TEST mode and normal operation.

10.2 Receive Channel (Ingress) – Line Side to System Side

10.2.1 Receive 10G Ethernet 64B/66B Processor: R64B66B

The Receive 64B/66B Processor implements all the required functionality of the 10 Gigabit Ethernet PCS interface as specified in the IEEE 802.3ae Standard. The R64B66B decoder performs 64B66B alignment, payload descrambling, 64B66B data/control decoding, and Ethernet frame delineation. The resulting decoded stream is then passed to the upstream device.

The R64B66B implements the following principal functions:

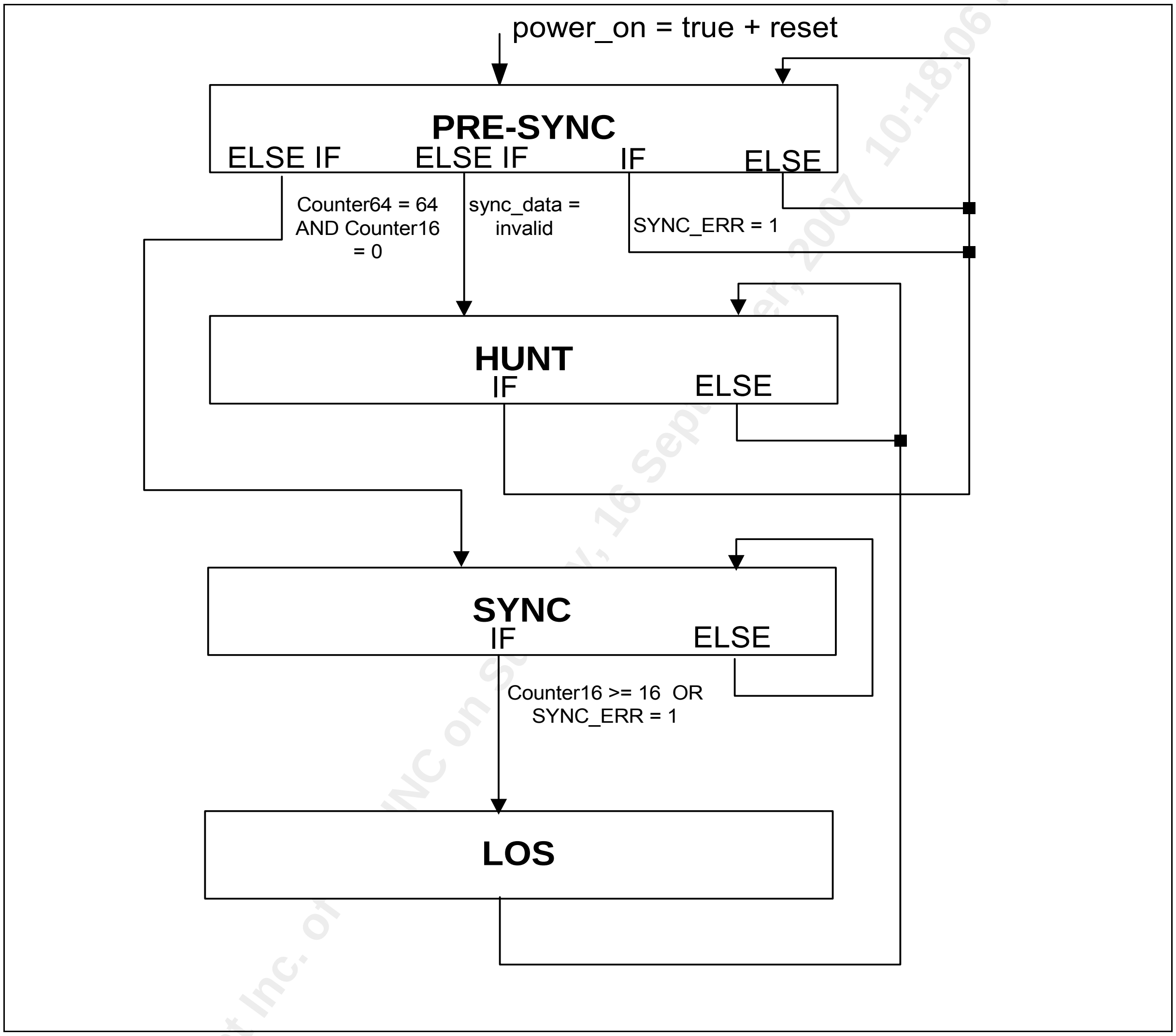
- Ethernet frame delineation
- Aligns to 32-bit SOP Boundaries
- 33:32 Rate adaptation
- 64B66B Data decoding
- 64B66B Control Code decoding
- Data De-scrambling
- Receive Link Fault signaling
- Receive Jitter test pattern checking

RX PCS

The RX PCS contains the Finite State Machines for Synchronization, Receive Data and Bit Error monitoring. It contains a de-scrambler based on $G(x) = 1 + X^{39} + X^{58}$. It provides all the timing and control necessary for capturing, aligning, and processing the downstream data. The decision(s) to leave or remain in a given state are based upon the IEEE 802.3ae Standard. The RX PCS provides Jitter Test pattern checking as per IEEE 802.3ae standard.

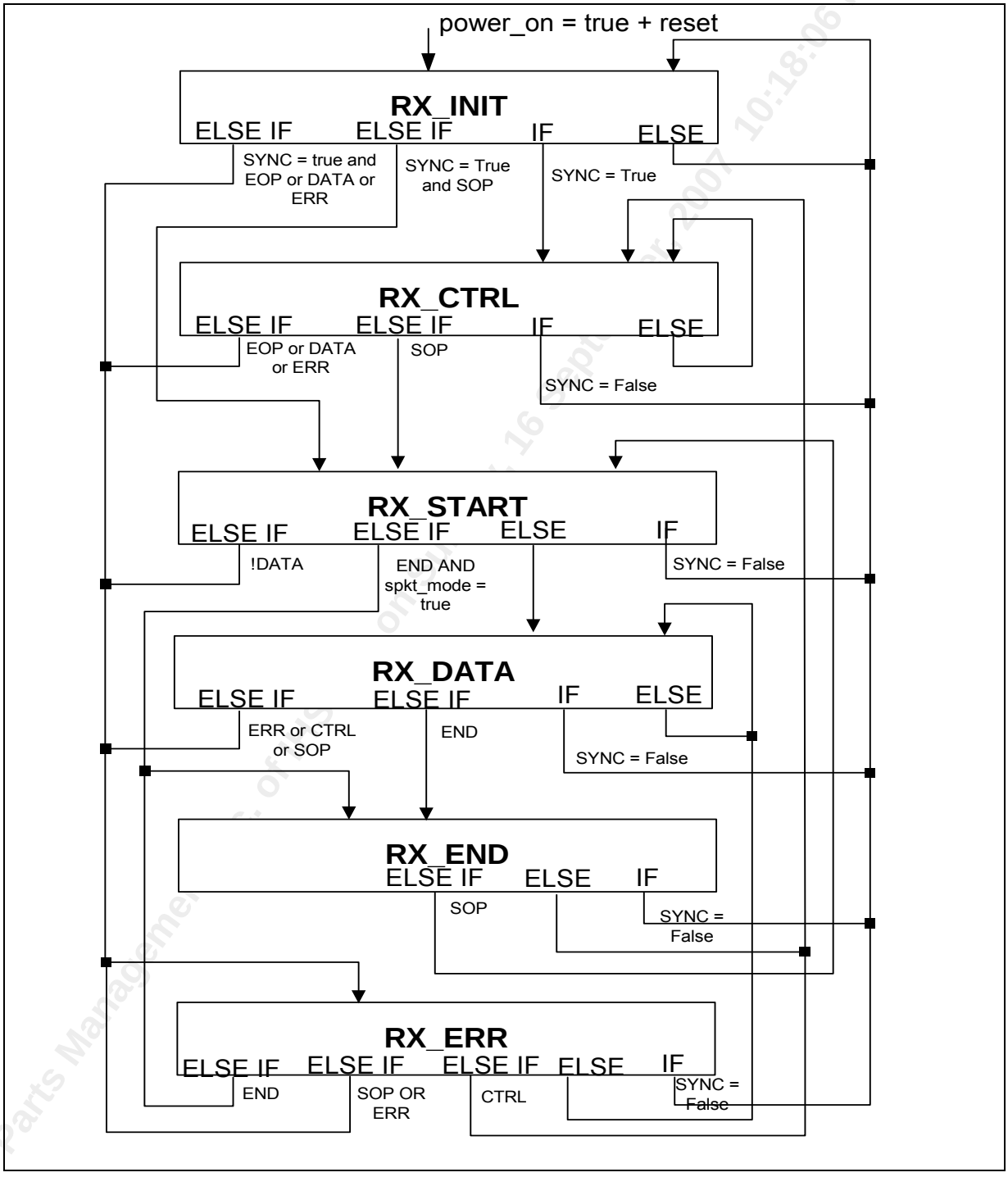
The Receive SYNC State Machine, shown in Figure 10-1 below, contains four states PRE-SYNC, HUNT, SYNC and LOS and is based on the synchronization process described in IEEE 802.3ae standard clause 49. PRE-SYNC is the start-up state, when in this state, it will attempt to sync on the incoming 2-bit sync field. If the 2-bit sync field is not a proper sync character of ‘10’ or ‘01’ then the state machine transitions to the HUNT state. The HUNT state will then shift the data by 2-bits. This shift will now provide the Receive SYNC State Machine with a new sync field and will attempt to synchronize on this new sync field. This process will continue until the current sync field and the next sync field are valid sync characters of ‘10’ or ‘01’. When this happens the Receive SYNC State Machine monitors two counters, 64 block counter and 16 bad sync counter. The 64-block counter performs the 64-block window that is needed to maintain SYNC in the RX PCS, it counts 64 66-bit blocks and rolls over and starts again. The 16 bad sync counter is needed to count the number of bad sync fields received either ‘11’ or ‘00’. If the Receive SYNC State Machine is seeing good sync and the 64 block counter = 64 and the 16 bad sync counter = 0, the state machine will transition to the SYNC state. When in the SYNC state the state machine continues to monitor the 64 block counter and the 16 bad sync counter, if the 16 bad sync counter is = 16 and the 64 block counter is less than 64 then the state machine transitions to the LOS state. The transition to the LOS state will cause the assertion of Loss Of Sync and Link Fail status, which may cause an interrupt if enabled. Loss Of Sync will cause the Receive and BER state machines to transition to their INIT states. Once in the LOS state the Receive state machine, on the next clock cycle, transition to the HUNT state to try and re-acquire sync.

Figure 8   RX SYNC State Machine



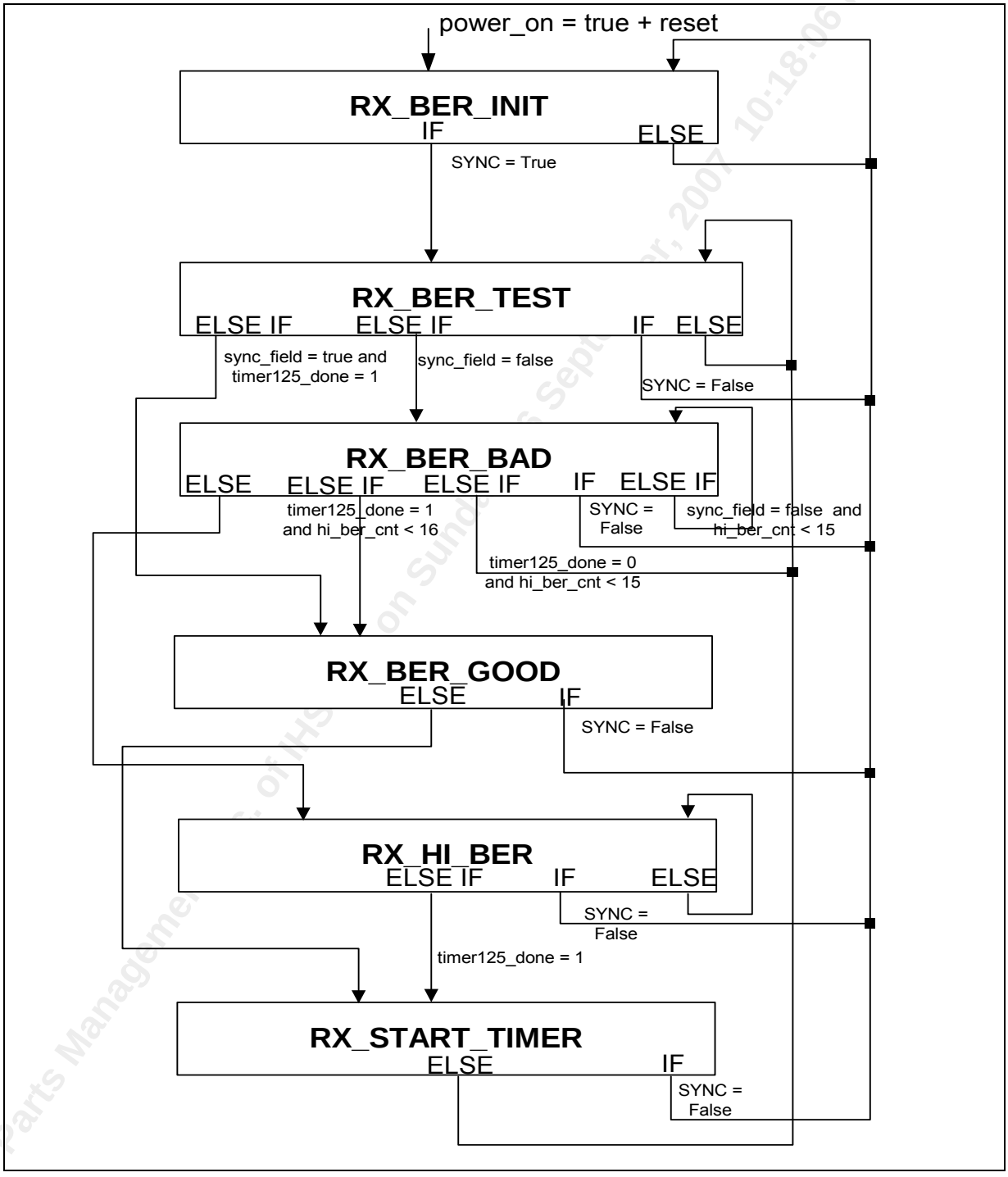
The Receive State Machine, shown in Figure 9 below, contains six states RX_INIT, RX_CTRL, RX_START, RX_DATA, RX_END, RX_ERR and is based on the Receive State Machine process described in IEEE 802.3ae standard clause 49. The RX_INIT state is the start-up state and will not transition out of this state until the Receive SYNC State Machine is in the SYNC state. When the Receive State Machine is in the RX_INIT state no data will be transferred to the MAC. Once the Receive SYNC State Machine is in the SYNC state the Receive State Machine will monitor the sync and type fields of the incoming data stream to decide on the proper transitions between states.

Figure 9 Receive State Machine



The Receive Bit Error Rate (BER) State Machine, shown in Figure 10 below, contains six states RX_BER_INIT, RX_BER_TEST, RX_BER_BAD, RX_BER_GOOD, RX_HI_BER, RX_START_TIMER and is based on the Receive BER State Machine process described in IEEE 802.3ae standard clause 49. The RX_BER_INIT state is the start-up state and will not transition out of this state until the Receive SYNC State Machine is in the SYNC state. No Bit Error monitoring will be performed when in the RX_BER_INIT state. Once the Receive SYNC State Machine is in the SYNC state the Receive BER State Machine will monitor the sync field of the incoming data stream to decide on the proper transitions between states. The Receive BER State Machine monitors two counters and the sync field of the incoming data stream. The two counters are a 125µs counter and the high bit error counter. The 125µs counter is an approximation based on the SYS_CLKx2 clock. The SYS_CLKx2 period is ~6.2 ns, therefore the counter will count up to 20,162 clock periods and hold. The high bit error counter counts the number of invalid sync fields (‘11’ or ‘00’).

Figure 10 RX BER State Machine



Jitter Test pattern checker

The Control Detector can be put into a test mode (jitter test mode) where it counts errors in the received data. The other side of the link must be put into test mode, so that it is generating the expected data patterns.

The Jitter test pattern checker utilizes the lock state machine and the descrambler operating as they do during normal data reception. The hi_ber state machine is disabled during receive test pattern mode. When block_lock is true and the receive test pattern mode is active, the test pattern checker observes the output from the descrambler. When the output of the descrambler is the data pattern or its inverse, a match is detected. Since the transmitter's scrambler is loaded with a seed value every 128 blocks (block is 64 bits) and the receiver's descrambler is running normally, a mismatch will be detected once every 128 blocks in the absence of errors. The transmitter reseeding the scrambler, and inverting the input data, causes this mismatch. The test pattern checker will count 128-block windows. The first mismatch in a window will not increment the test pattern error counter. Any subsequent mismatch in a window indicates an error and will increment the Jitter Counter(Register 0x2087).

10.2.2 Receive 10G Ethernet MAC (RXXG)

The Receive 10G Ethernet MAC (RXXG) provides 10 Gigabit Media Access Control Sub-layer processing on a single 10 Gigabit Ethernet stream. The RXXG expects data from the R64B66B block. The exact data throughput is determined by the system clock frequency, the inter-packet or intra-packet gaps received from the line interface, and the backpressure received from the FIFO interface.

The RXXG implements the following principal functions:

- Ethernet framing (detection and framing to the standard preamble/SFD sequence, removal of the preamble/SFD, and checking of the 32-bit CRC field) and frame validation (marking of errored frames for discard).
- Frame timing check (relative to the receive input clock reference): the RXXG will verify that the interframe gap does not fall below a pre-set minimum, and will filter frames that violate this restriction, when used in LAN-mode devices.
- Optional received frame filtering: frames can be discarded if they are found to contain length or CRC errors. The RXXG block implements a 2048-byte full-frame buffer to facilitate this filtering. Note that jumbo frames (>1522 bytes) will not be filtered, but will be marked for discard by a downstream entity.
- Received frame stream parsing, and PAUSE MAC Control frame detection, validation and extraction. The PAUSE Timer fields of received PAUSE frames are extracted and sent to flow control logic on a separate set of signals.
- Receive statistics support: the RXXG checks every received frame against the IEEE 802.3 frame error criteria, and outputs a statistics vector at the end of every received frame. The statistics vector is expected to be used by an external unit to update the appropriate statistics counters, which should be used to implement the standard Ethernet MIBs for link management.

- Optional Receive PAD stripping for all non-errored packets received.

The RXXG functional sub units include

- Receive MAC (RMAC) Framer
- CRC Checker
- Receive MAC (RMAC) Parser
- Receive FIFO

RMAC Framer

The RMAC Framer monitors for valid words from the R64B66B block, expecting preambles/SFD bytes. If the framer is looking for standard Ethernet preamble/ SFD, it can also be configured to validate the preamble byte contents and length. If errors are detected, RMAC Framer will discard the frame. If no errors are found, the preamble and SFD are stripped and the following frame byte marked as the MAC Header SOP. At this stage, the RMAC Framer writes the frame into an internal FIFO. The RMAC Parser starts MAC frame parsing and the CRC Checker starts accumulation. PUREP, and LONGP are register configuration bits that are set in the RXXG Configuration 1 register, address 0x2040. The device defaults to PUREP disabled where only the SFD is evaluated. If PUREP is enabled, the device will ignore frames with anything other than 0x55 or pure preamble.

Table 8 Preamble Checking

PUREP	LONGP	Preamble/SFD Framing Accepted
0	0	1-11 bytes of non-0xD5 byte values followed by 0xD5 (SFD) byte
0	1	Any number >1 byte of non-0xD5 byte values followed by 0xD5 (SFD) byte
1	0	1-11 bytes of 0x55 byte values followed by 0xD5 (SFD) byte
1	1	Any number >1 byte of 0x55 byte values followed by 0xD5 (SFD) byte

RMAC Framer maintains a packet count and will perform in-range checking for Type-Length field values between 0 and 1518 bytes. Packets with in-range errors may be optionally discarded. RMAC Framer verifies packet size against the minimum value (64 bytes) and maximum programmable frame size. Packets outside this range may be optionally discarded. Table 9 lists the range, packet size checks and resulting packet classification. All the actual length value checks must have RX_MAXFR, maximum frame, and value incremented by 4 for the case where the packet is VLAN Tagged. Also note that 'CRC Result' also takes into consideration the case where the packet EOP received from the line had LINE_ERR set, and this case is treated as if the packet had a bad CRC.

Table 9 Range, Size and CRC Result Processing

Type/Length Field (TL)	Actual Packet Size	CRC Result	Packet Classification
X	< 64	Good	Undersize

Type/Length Field (TL)	Actual Packet Size	CRC Result	Packet Classification
X	< 64	Bad	Fragment
X	> RX_MAXFR	Good	FrameTooLong
X	> RX_MAXFR	Bad	Jabber
X	>63 & <= RX_MAXFR	Bad	FrameWithFCSError
46 - 1500	TL + 18	Good	Good
46 – 1500	TL + 22 & (VLAN Tagged)	Good	Good
46 - 1500	!(TL+18) & (> 64 & <= RX_MAXFR)	Good	InRangeLengthError
< 46	64	Good	Good
< 46	>63 & <= RX_MAXFR	Good	InRangeLengthError
1501-1536	-	Good	Good
< 46	68 & (VLAN Tagged)	Good	Good
< 46	>64 & <= RX_MAXFR & !(68 & VLAN)	Good	InRangeLengthError
1501-1536	-	Good	Good
> 1536	> 64 & <= RX_MAXFR	Good	Good

Once an EOP is received from the R64B66B block, the RMAC Framer waits for a CRC result to be returned from the CRC Checker and packet filter status from the RMAC Parser. It will then inform the FIFO Controller about FIFO flushing of the packet or forwarding of the packet to the System Interface. If the packet is terminated with an error flag set with EOP, RMAC Framer may discard the packet only if it is less than 1900 bytes in length. If the frame is larger than the internal ram capacity, the frame will not be discarded on error, but marked appropriately and transmitted.

The RMAC Framer can be configured to check for a minimum inter-packet gap between received frames. If this is enabled, the RMAC Framer will ignore packets when the SOP is received during the IPG.

CRC Checker

The CRC checker performs a CRC-32 calculation on the entire data frame received from the RMAC Framer. The 32-bit CRC result is compared to a constant expected value and a good / bad CRC status is returned to the RMAC Framer.

RMAC Parser

The RMAC Parser classifies the packet according to MAC Header fields, DA, SA, Type/Length and optional VLAN Tag. If address filters have been enabled, the required address and optional VLAN ID are applied to an address filter function, which will determine if the packet should be forwarded or filtered. In addition, if Multicast Hashing has been enabled, Multicast DAs are applied to a CRC-32 Hash function, resulting in an index into a 64-bit MHASH register and a filtering/ forwarding decision made. The address filter block resets to a promiscuous mode of operation, whereby all filtering is disabled.

The RMAC Parser checks for received PAUSE Control packets and, if enabled, will transfer the extracted pause parameter to the TXXG. The RMAC Framer may be configured to forward or filter received MAC Control packets of all types. When the RMAC Parser updates the RMAC Framer with packet classification information (after EOP), RMAC Framer will make a forward/filter decision.

Receive FIFO

The purpose of the FIFO is to provide the capability to discard erred frames, up to a maximum size of 1522 bytes (maximum expected frame size). This FIFO is independent of the Ingress Flexible FIFO (IFLX), which contains the main ingress system buffering. Two modes of operation are provided: store forward and cut through.

In store-and-forward mode, frame transfer to the IFLX only begins once the complete frame has been written to the FIFO. This will not deal with a situation where an oversized frame is received. Hence a FIFO Read Threshold has been provided, which will be initialized to a value > Maximum Size Frame divided by 8. This will ensure that if the FIFO fills beyond this level, the read port will begin transferring the frame to the IFLX.

The same FIFO Read Threshold could be used to configure the FIFO for a cut-through mode, with all error frame discard functions disabled. In this case, the IFLX would be informed to begin frame transfer when a small number of entries have been written into the FIFO.

10.2.3 Ingress Flexible FIFO (IFLX)

The Ingress Flexible FIFO (IFLX) provides a FIFO to separate the line-side timing from the higher layer system timing and the associated PL4 system interface. The IFLX receives line side data from the Ten Gigabit Ethernet physical port via the XSBI interface, R64B66B and RXXG blocks.

The IFLX provides 128KB (131,072 bytes) of storage in total. The configuration of the buffer space cannot be done dynamically. Changes to the buffer are to be done during device initialization. The FIFO has a low watermark, and a high watermark. The low and high watermarks can be used for Ethernet PAUSE Flow Control operation.

When the IFLX ingress FIFO reaches the programmed high watermark flow control threshold the ingress FIFO will assert an indication to the Transmit MAC (TXXG) to start PAUSE flow control. The IFLX ingress FIFO will continue to keep the flow control signal asserted until the number of entries in the FIFO have decreased to the programmed low watermark flow control threshold level.

In the event that the PL4 system-side sink device indicates that it is no longer able to accept Ethernet frame data (so PL4 FIFO status from the system-side sink device is SATISFIED and the PL4 FIFO channel credit counter decrements to 0), the PM3392 will buffer the incoming frames from the line side XSBI interface until all the buffer facilities within the PM3392 are exhausted. This buffer capacity includes both the allocated IFLX ingress FIFO buffer space and a 2KB (2048 bytes) receive buffer in the Receive MAC (RXXG). At this time the PM3392 will no longer accept data from the line side, resulting in overflow of the RXXG receive FIFO. When buffer resources within the PM3392 become available, the RXXG will re-synchronize to the next start of a physical packet (preamble/SFD delimiter) and continue Ethernet frame reception. In the event that the PM3392 truncates a frame because of lack of buffer space the frame will be marked as erred and is a countable event in the MSTATS receive counter FramesLostDueToInternalMACError. Whether the received frame that was truncated in the RXXG receive FIFO is forwarded to the IFLX is determined by the setting of the FIFO Cut-thru Threshold register, the point at which truncation occurred, and whether any data from the frame had been transferred from the RXXG receive FIFO to the IFLX).

Any time an erred receive frame is forwarded to the IFLX from the RXXG, the control word following the transfer of the last data byte of the erred frame on the PL4 Bus will terminate with an End-Of-Packet status of ABORT.

The IFLX ingress FIFO will absorb in-flight frames regardless of the PAUSED flow-control state of the PM3392. The amount of Ethernet frame data that can be received without exhausting receive buffer capabilities is related to:

The buffer fill level at which the PAUSE request was generated (for example, the Ingress FIFO high watermark register setting)

The response of the link-partner to a PAUSE request.

The outgoing rate of frame data on the given channel on the PL4 system-side. This is controlled by the system-side sink device using the FIFO status signaling protocol and the PL4 output scheduling block (PL4MOS) on the PM3392 device.

10.2.4 PL4 Multi-Channel Output Scheduler (PL4MOS)

The PL4 Multi-Channel Output Scheduler (PL4MOS) block is used to provide fairness when the IFLX is configured for more than one channel of operation. Since the PM3392 only has one channel the fairness operation of the PL4MOS is not relevant.

The PL4MOS consists of three main blocks: the Status Calendar, the Fairness Controller, and the Scheduling Engine. The Fairness Controller will not be discussed since this function is turned off in the S/UNI-1x10GE due to only one channel being active.

Status Calendar

The Status Calendar communicates with the PL4 interface to obtain the FIFO Status (FS) on the PL4 system-side sink device. As described in the PL4 Bus specification, the FIFO Status in the PL4MOS is interpreted as a 2-bit code:

00	STV	Starving
01	HUN	Hungry
10	SAT	Satisfied
11	FRM	Framing

The Status Calendar is responsible for assigning credits based on the FIFO status information received on the RSTAT[1:0] pins of the PM3392 device. The Status Calendar operates on the latest available FIFO information.

If a loss of synchronization is detected on the RSTAT FIFO status channel the Status Calendar resets the internal FIFO status and clears the credit counters to zero.

Figure 11 State Machine of a FIFO Channel

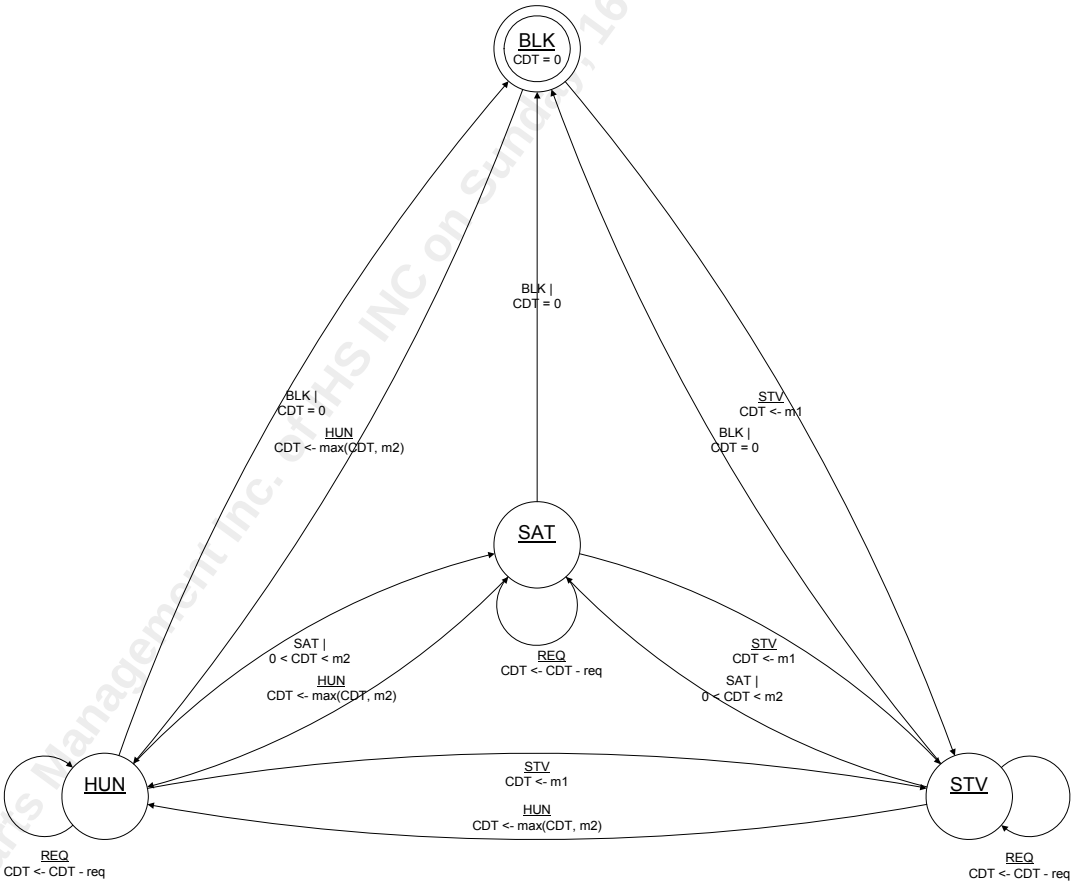


Figure 11 shows a state diagram for a single channel and how the FIFO status inputs affect the state machine of each channel. A 16-bit credit counter (CDT) is defined. This counter is set to MaxBurst1 when the FIFO status is STV. If the RSTAT FIFO status is updated to HUN, the credit counter is set to the maximum of MaxBurst2 and the current value of the credit counter. The values of MaxBurst1, MaxBurst2 are calculated based on the overall system latency and configuration and programmed into the PL4MOS MaxBurst1 and MaxBurst2 registers.

Scheduling Engine

The scheduling engine is responsible for generating data transfer requests and sending them to the IFLX. The scheduler generates the request based on two parameters. First is the status of the FIFO whether it is HUN, SAT, STV or BLK, where BLK stands for Blocked and will be explained later in this section. Second is whether the IFLX has data ready for the FIFO.

The scheduler checks the status and if the IFLX has data, if the status is HUN or STV and the IFLX has data ready, a service request is generated. Every time the scheduler issues a request it updates the credit counter: it decrements the current value of CDT by a value, MAX_WRDS, which is the minimum of the configurable PL4 maximum Data Burst length (MAX_TRANSFER register) or the CDT counter.

$$CDT = CDT - \min(MAX\_TRANSFER, CDT).$$

10.2.5 PL4 Output Interface

The PL4 Output Interface implements the PL4 protocol as described in the PL4 Bus specification (PMC-991635) via the RDCLK+/-, RCTL+/-, and RDAT[15:0]+/- device pins. In the block diagram, this datapath encompasses the PL4ODP and part of the PL4IO block. The PL4 interface provides a 16-bit wide data and in-band control stream, a single control signal and a dual-phase source synchronous clock in the forward path. The clock signal is either looptimed from the TDCLK+/- or internally generated using the PL4_RCLK+/- reference input. The control signal is used to identify the in-band control words. In the return path, the PL4 interface provides a two-bit FIFO status bus with associated clock. All transmit and receive datapath signals and their associated data clock are differential LVDS: RDCLK+/-, RCTL+/-, and RDAT[15:0]+/- for ingress; TDCLK+/-, TCTL+/-, and TDAT[15:0]+/- for egress. Please refer to PMC-991635 for protocol details.

The PL4 Output Interface takes a data stream provided by the IFLX block, inserts in-band control (SOF, EOF status, DIP4 code, channel address) and bursts the stream across the PL4 bus to a system-side device in accordance to the PL4 protocol via the RCTL+/- and RDAT[15:0]+/- device pins. The data sink provides back to the PM3392 PL4 Output Interface FIFO status indications using a 2-bit bus (RSTAT[1:0]) and associated clock (RSCLK). These status signals are provided to the PL4MOS block and are used in scheduling of data transfers from the IFLX FIFO to the PL4 bus.

The PL4 Output Interface allows for the transmission of a training sequence to allow for dynamic de-skewing by a sink entity. The training sequence can be programmed to be a configurable number of consecutive PL4 Bus training patterns. When possible, training patterns are stuffed into what would have been a sequence of twenty IDLE control characters: this does not affect normal frame data transfer. If a complete training sequence is sent within the configurable MAX_T number of PL4 Bus cycles, then normal frame data transfer will be held-off until a complete training sequence is sent.

The minimum SOP-to-SOP spacing for data to be impressed on the PL4 Bus by the PM3392 is programmable to be a minimum of 2 or 8 PL4 Bus cycles.

All PL4 outputs from the PM3392 device are strictly compliant with the POS-PHY Level 4 specification.

10.3 Transmit Channel (Egress) – System Side to Line Side

10.3.1 PL4 Input Interface

The PL4 Input Interface implements the PL4 protocol as described in the PL4 Bus specification (PMC-991635) via the TDCLK+/-, TCTL+/-, and TDAT[15:0]+/- device pins. In the block diagram, this datapath encompasses the PL4IDU and part of the PL4IO block. The PL4 interface provides a 16-bit wide data and in-band control stream, a single control signal and a dual-phase source synchronous clock in the forward path. All forward path signals are differential LVDS. The TDCLK+/- clock signal is required to be source-synchronous to the TCTL+/- and TDAT[15:0] signals. The control signal is used to identify the in-band control words. In the return path, the PL4 interface on the PM3392 drives a two-bit FIFO status bus with associated clock (TSTAT[1:0] and TSCLK). Please refer to PMC-991635 for protocol details.

The PL4 Input Interface sinks data and control on the PL4 Bus: the data stream impressed on the TDAT[15:0]+/- device pins is delineated using TCTL+/-. The incoming stream of PL4 Bus words is parsed according to the Data Path Bus state diagram included in the PL4 specification. Ethernet frame data is decapsulated in the incoming PL4 word stream and data bytes contained within a PL4 Data Burst are written into the EFLX egress FIFO. PL4 control words are checked and extracted from the incoming PL4 word stream.

Data must contain sufficient training pattern density to allow reliable operation of the data recovery and deskew units in the PL4IO input logic. The PL4 Bus specifies the transfer of un-encoded NRZ data streams. Consequently there may be arbitrarily long runs of consecutive zeros or ones. The PL4 input interface is capable of properly recovering data once training has completed. A certain minimum rate of training patterns is required to ensure that data continues to be received without error. See the Operations section for further details.

The POS-PHY Level 4 specification defines correct operation, but generally does not define error handling except where it affects interoperability between the peer devices. Consequently different devices may handle the same error differently.

The following simple rules are followed by the PL4 Input interface on the PM3392 device with respect to PL4 protocol errors:

- Erroneous PL4 inputs (presented on PL4 TCTL and TDAT[15:0] pins) that could affect data integrity are strictly enforced.
- Erroneous inputs that do not affect data integrity (length of training pattern, minimum SOP spacing, etc) are only enforced to the extent that they actually cause problems for the state machines or data path.
- No single error will cause the data path (PL4 TCTL and TDAT[15:0] pins) or the status path (PL4 RSTAT[1:0] pins) to lose synchronization.

There is a link status state machine in the PL4 Input Interface to handle the tracking of errors with respect to the reception of non-erred training sequences. Four link states are defined and can be read from the PL4IDU Status register:

- DISABLE: awaiting the decode of a non-erred training pattern. No data within a PL4 Data burst will be forwarded to the downstream EFLX from the PL4 Input interface.
- RUN_GREEN: have received at least one non-erred training pattern and there have been no detected PL4 word parsing errors. Data within a PL4 Data burst will be forwarded to the downstream EFLX from the PL4 Input interface.
- RUN_YELLOW: have detected one PL4 word parsing error since being in a RUN_GREEN state. Reception of a non-erred training pattern will result in a state change from RUN_YELLOW to RUN_GREEN. Reception of a PL4 word that has a detectable error will result in a state change from RUN_YELLOW to RUN_ORANGE. While in RUN_YELLOW, data within a PL4 data burst will be forwarded to the downstream EFLX from the PL4 Input interface.
- RUN_ORANGE: have detected two PL4 word parsing errors since being in a RUN_GREEN state. Data within a PL4 Data burst will be forwarded to the downstream EFLX from the PL4 Input interface as long as a third PL4 word parsing error is not detected. Reception of a non-erred training pattern will result in a state change from RUN_ORANGE to RUN_GREEN. Reception of a PL4 word that has a detectable error will result in a state change from RUN_ORANGE to DISABLE. While in RUN_ORANGE, data within a PL4 data burst that is non-erred will be forwarded to the downstream EFLX from the PL4 Input interface.

In the event that the word parsing on the PL4 Input Interface detects an error (for example, the DIP4 code check fails on the control word following a PL4 Data Burst) in the PL4IDU that have a PL4 Port State of ACTIVE or PAUSED will be internally aborted. This will result in any Ethernet frame data being written out to the EFLX interface with the error flag set; the EFLX FIFO will subsequently propagate the frame data and error flag to the Ethernet MAC transmit interface. Frame data terminated in this manner will there be sent out the serial link interface as an erred Ethernet transmit frame. The setting of the error flag in the data sent from the EFLX to the TXXG is a detectable and countable event (MSTAT Transmit Statistics Counter TransmitSystemError).

10.3.2 Egress Flexible FIFO (EFLX)

The Egress Flexible FIFO (EFLX) provides a FIFO to separate the system-side source timing of the PL4 Bus interface from the Ethernet MAC (TXXG) transmit line-side timing. The EFLX presents a single interface to the PL4 bus on the input side; on the output side the EFLX connects to the transmit interface of the TXXG block.

The EFLX provides 16KB (bytes of total buffering). The configuration of the buffer space cannot be done dynamically. Changes to the buffer are to be done during device initialization. There is a minimum size required in order to guarantee that an almost-full condition on the Egress FIFO signaled via the TSTAT[1:0] pins of the PM3392 device will be able to absorb the maximum number of frame data bytes that can be in-flight. The FIFO has a configurable low start threshold, a low watermark, and a high watermark. The low start threshold is used to allow accumulation of frame data before it is passed to the TXXG interface. Data is not forwarded from the egress FIFO until the first of these conditions is met:

- The EOP flag bit associated with the frame data in the Egress FIFO has been set
- The low start threshold on Ethernet frame size for has been met by the frame data (programmable via **the EFLX Indirect FIFO Cut-Through** register).

This hold-off is done in order to avoid underruns during transfer of the data to the TXXG interface. The low and high watermarks are set to provide the PL4 Bus FIFO status signaling back to the system-side source device. Refer to the Operations section for additional details.

10.3.3 Transmit 10G Ethernet MAC (TXXG)

The Transmit 10G Ethernet MAC (TXXG) provides 10 Gigabit Media Access Control Sub-layer processing on a single 10 Gigabit Ethernet stream. The TXXG reads partially formed Ethernet frames from the Egress Flexible FIFO (EFLX) and generates completely formed Ethernet frames. The resulting frames are forward to the T64B66B block.

The TXXG implements the following principal functions:

- Ethernet framing (optional insertion of an 8-byte preamble/Start Frame Delimiter sequence, plus computation and optional insertion of a 32-bit FCS).
- Frame timing relative to the system clock reference input. The TXXG will insert the correct programmable interframe gap between frames to ensure that LAN-mode operation conforms to the IEEE 802.3 specification. In WAN mode, no interframe gap is inserted; instead, the downstream payload processor is responsible for flow controlling the data stream.
- PAUSE frame generation and insertion. The TXXG block will generate and format 64-byte PAUSE MAC Control frames in response to requests from external blocks, with the specified pause timer values inserted at the proper location. Generated PAUSE frames are multiplexed into the outgoing frame stream in between data frames and with the proper spacing.

- Transmit pause implementation. PAUSE frames received from the entity at the other end of the Ethernet link must implement flow control by halting or resuming transmitted traffic. The TXXG block provides the means for this to happen: pause timer values extracted from received PAUSE frames can be input to the TXXG along with a notification signal, and will cause the TXXG block to stop or restart traffic as required at the proper boundaries.
- Transmit Statistics support. After each data or PAUSE frame has been transmitted, whether successfully or unsuccessfully, the TXXG block outputs a Statistics vector that signals the Status of the transmission. This Statistics vector is expected to be used by an external unit to update appropriate Statistics counters. These counters in turn can be used to implement Standard Ethernet MIBs for link management purposes.
- Configuration and Status maintenance. The interframe gap, preamble, FCS generation, and error checking features of the TXXG can be configured by means of internal configuration registers accessible via an ECBI bus interface.

The following sections will describe the functionality of each sub-block in more detail.

MAC Incoming Flow Control State Machine

When a Receive MAC (RXXG) decodes an error-free incoming 803.2 full-duplex flow control packet, the RXXG sends a PAUSE signal to the TXXG indicating that transmission is to be paused. The pause parameter is transferred to the TXXG unit, to apply to the local pause timer. This State-machine loads the timer (on request from the RXXG unit) and decrements the timer in units of pause interval**.** Note, PAUSE control frame transmission initiated using register bits is not affected by the PAUSE pin.

MAC Outgoing Flow Control State Machine

This State-machine monitors a high watermark FIFO threshold level indication from the IFLX or the external PAUSE pin, and will issue a request to the MAC Core to transmit a MAC Control PAUSE frames when required. When the MAC Core acknowledges this request, this State-machine will build the MAC Header and Data frame, and present it to the MAC Core for transmission. The Pause Parameter value can be programmed using an internal register. Pause Control frames are injected (by the MAC Core) at the rate programmed using the PAUSE Timer Interval register after the transmission of the current frame. This periodic transmission will request the far end to XOFF for the duration of the timer.

When sending out Control PAUSE frames and a low watermark FIFO indication from the IFLX is detected, the MAC core inserts a PAUSE control frame with a pause timer value of 0 after the transmission of the current frame. This transmission will request the far end to immediately XON.

The transmit MAC can also be directed by the ingress MAC to halt data transmission to the line side. If the receive MAC interprets a pause frame, the transmit MAC will be notified and also given a timer value. The transmit MAC will halt transmission of data until the pause timer value has expired. While the transmit MAC is halted, the external PAUSED pin to the line side will be asserted.

MAC Core

The MAC Core is the heart of the TXXG TSB and implements the principal functions of the IEEE 802.3 MAC layer. It consists of a state machine, data steering and encapsulation logic, together with a 64-bit CRC generator for computing the Standard 32-bit IEEE FCS.

At any time, it may be operating in one of three modes:

- Frame transmission disabled.
- MAC Flow Control packet transmission enabled, when the MAC Incoming Flow Control State machine indicates the transmitter is PAUSED.
- MAC Flow Control and EFLX-source packet transmission enabled, with priority being given to MAC Flow Control packet transmission, when not PAUSED.

During operation, the MAC Core waits until a frame is available to be sent, the downstream is requesting a packet for transmission and the inter-packet gap timer (IPG timer) is not active. It then generates the requisite preamble and SFD, and then transfers frame data from the packet source, 64 bits at a time. The SOF/EOF, error and valid signals supplied with the data words are processed to determine frame boundaries and control the CRC generator, and also to determine intra-frame zero-padding and post-frame IPG spacing. The processed 64-bit data words are output to the downstream block.

At the end of each frame transmission, the MAC Core inserts an inter-frame gap of the required amount before starting the transfer of the next frame. Note that the interframe gap is counted in units of bytes rather than absolute time, permitting frame timing to be done differently based on the application.

After each packet transmit is completed, the MAC Core builds and transfers a Transmit Statistics Vector to the MSTAT block.

CRC Generator

The CRC checker performs a CRC-32 calculation on the whole Ethernet frame received using the standard Ethernet polynomial.

The CRC Generator performs a CRC-32 calculation on the whole Ethernet frame using the standard Ethernet polynomial. A parallel implementation of the CRC polynomial is used. The result is optionally prepended to the frame.

10.3.4 Transmit 64B66B Encoder (T64B66B)

The Transmit 64B/66B Processor implements all the required functionality of the 10 Gigabit Ethernet PCS Sub-layer interface as specified in the unapproved IEEE 802.3aestandard. It performs frame delineation on transmitted frames from an upstream block, scrambles the frames, and 64B/66B encodes the scrambled stream. The resulting encoded stream is passed to a downstream device. In IEEE mode, the T64B66B input stream consists of standard Ethernet (Preamble/SFD, MAC Header, Data, Pad and CRC) frames with possible intra-packet gaps (128-bit word gaps).

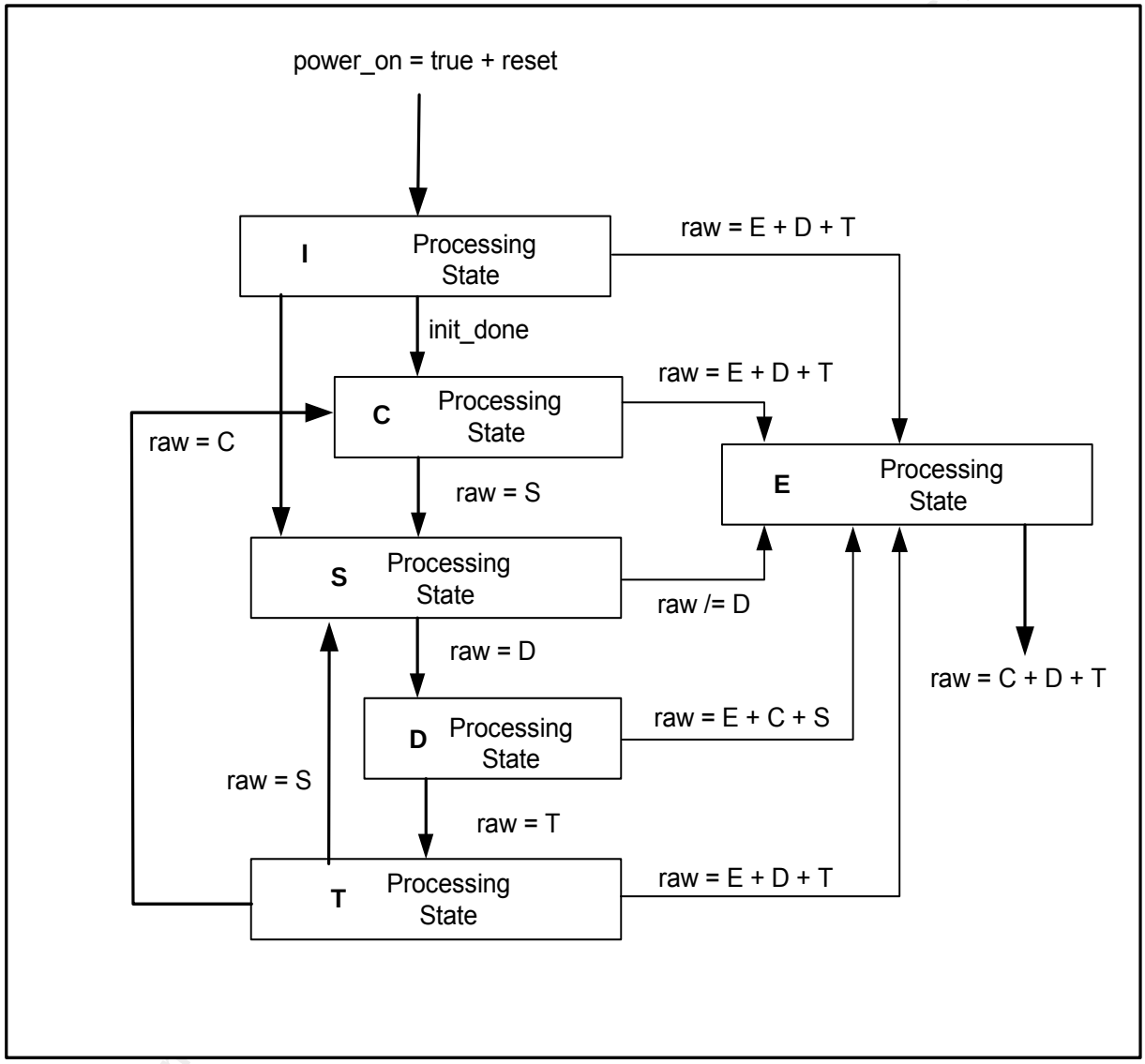
The T64B66B implements the following principal functions:

- Ethernet frame delineation
- Aligns to 32-bit SOP Boundaries
- 32 to 33 Gearbox
- 64B/66B data encoding
- Control Code Mapping
- Data Scrambling
- Bit within Byte Ordering Function
- Transmit Fault Signaling
- Transmit Jitter test pattern generator

TX PCS

The TX PCS contains a Finite State Machine, which sequences through the 6 basic states shown in Figure 12. It contains a scrambler based on $G(x) = 1 + X^{39} + X^{58}$. It provides all the timing and control necessary for capturing, aligning, and processing the upstream data. The decision(s) for the transmit state machine to leave or remain in a given state are based upon the IEEE 802.3ae standard. The TX PCS provides Jitter Test pattern generation as per IEEE 802.3ae standard.

Figure 12 TX Processing Steps For 66-Bit Codes



From Figure 12, the controller enters its Initialization (I) State during reset or during a power-up sequence. To exit the I State, the Input Controller must wait for the internal data formatter and barrel shifter to sequence completely through their first 33 clock processing cycle, it also may wait for the de-assertion of an overhead marker (depending on mode) from the downstream device and/or the assertion of a Control Detect indication from the Data Encoder Block. Given a successful execution of a proper initialization and the detection of the proper out-of-band control signals, the state machine will advance to its Control Code or (C) State. The C state processor will map only the incoming Control Codes. Upon detecting a valid SOP transition, the state machine advances to the S processing state.

By looking at Figure 12, a typical state sequence can be defined as: S, D, and T. The Start of Packet (S) State is entered via detection of the SOP byte control indication. The Data (D) State can consist of multiple (N) 66-bit frames. The End of Packet (T) State is entered via detection of the EOP byte control indication. In this case, according to the IEEE 802.3ae specification, the outgoing EOP byte can occur in anyone of eight possible locations within a 66-bit frame. An errored sequence would be any other type of transition that doesn't follow the simple sequence of Start-of-Packet state to the Data Capture state to the End-of-Packet state. For example the following sequences: Start-of-Packet state to the End-of-Packet state, Start-of-Packet state to the Start-of-Packet state, End-of-Packet state to the Data Capture state, and Data Capture state to the Start-of-Packet state would be counted as errored sequences. After detection of an errored sequence the state machine always advances to the E state.

During any Errored state the proper errored frame will be sent downstream to the Line Interface. During a Receiver Error Event: Loss Of Signal (LOS), Remote Fault (RF), or Local Fault (LF) the Input Controller will immediately advance to the E state, where it will stay until the errored event is cleared. During LOS or LF the TX PCS will start sending Remote Fault Messages continuously downstream. During RF the TX PCS will only send IDLEs downstream.

Jitter Test Pattern Generator

When this is enabled via the JITT_PAT_EN (bit 6 of register 0x3080), the scrambler's normal input data is ignored and two possible patterns are instead generated. If JTST_PAT_SEL (bit 7 of register 0x3080) is set to '1' a square wave signal of the wavelength defined by SWAVE_LEN+4 becomes the input to the scrambler. If instead JTST_PAT_SEL is set to '0' then a pseudo random signal is generated. This is done by:

1. Setting the scrambler's input to all zeros if the bit JDAT_PAT_SEL(bit 8 of register 0x3080) equals '1' or to the LF ordered set if JDAT_PAT_SEL equals '0'.

2. Seeding the scrambler every 128 blocks where the seed value repeats as follows:

    a. Seed A -> Seed A inverse -> Seed B -> Seed B inverse -> Seed A …

10.4 Management Statistics (MSTAT)

The MSTAT block is used to accumulate Ethernet specific counts used for supporting management agents such RMON, SNMP, and Etherlike interfaces. The MSTAT supports full system probing capability via the use of full counter snapshot to shadow registers. Incorporated into the MSTAT block is a fully programmable interrupt array enabling per counter rollover monitoring with interrupt reporting. Each MSTAT counter is 40-bits wide.

10.4.1 Receive Statistics Counters

With the Receive Statistics, there are 31 corresponding counters. They are defined in Table 10.

Table 10 Receive Statistics Counters

Name of Counter
FramesReceivedOK
OctetsReceivedOK
FramesReceived
OctetsReceived
UnicastFramesReceivedOK
MulticastFramesReceivedOK
BroadcastFramesReceivedOK
TaggedFramesReceived
PAUSEMACControlFrameReceived
MACControlFrameReceived
FrameCheckSequenceErrors
FramesLostDueToInternalMACError
SymbolError
InRangeLengthErrors
FramesTooLongErrors
Jabbers
Fragments
UndersizedFrames
ReceiveFrames64Octets
ReceiveFrames65to127Octets
ReceiveFrames128to255Octets
ReceiveFrames256to511Octets
ReceiveFrames512to1023Octets
ReceiveFrames1024to1518Octets (includes Tagged frames of 1522 bytes)
ReceiveFrames1519toMAXOctets
JumboOctetsReceivedOK
FilteredOctets
FilteredUnicastFrames
FilteredMulticastFrames
FilteredBroadcastFrames

10.4.2 Transmit Statistics Counters

With the Transmit Statistics, there are 22 corresponding counters. They are defined in Table 11.

Table 11 Transmit Statistics Counters

Name of Counter
FramesTransmittedOK
OctetsTransmittedOK

Name of Counter
OctetsTransmitted
FramesLostDueToInternalMACTransmissionError
TransmitSystemError
UnicastFramesTransmittedAttempted
UnicastFramesTransmittedOK
MulticastFramesTransmittedAttempted
MulticastFramesTransmittedOK
BroadcastFramesTransmittedAttempted
BroadcastFramesTransmittedOK
PAUSEMACCTRLFramesTransmitted
MACCTRLFramesTransmitted
TransmittedFrames64Octets
TransmittedFrames65to127Octets
TransmittedFrames128to255Octets
TransmittedFrames256to511Octets
TransmittedFrames512to1023Octets
TransmittedFrames1024to1518Octets (includes Tagged frames of 1522 bytes)
TransmittedFrames1519toMAXOctets
JumboOctetsTransmittedOK

10.5 Management Data Interface

10.5.1 Description

MDIO (management data input/output) provides communication between the host processor and an external physical device by means of a two-wire interface. The MDIO block generates a clock (MDC) by dividing down the internal XSBI interface clock to provide timing reference for transfer of information on the MDI and MDO signal.

MDIO provides a MII Management interface to transmit and receive management frame serially for the purpose of controlling the physical device and gathering status from the physical device. Synchronization bits, selection addresses, and control data and address to the external MII device are sent on the MDO. Status data is received on MDI. MDOEN is a tri-state driver enable for the MDO data. MDI and MDO are expected to be combined into a bi-directional pin external to this block. The port address bus PRTADR[4:0] is used to select one of the 32 unique port and the device address bus DEVADR[4:0] is used to select a particular external physical device with which to communicate.

10.6 JTAG Test Access Port Interface

The JTAG Test Access Port block provides JTAG support for boundary scan. The standard JTAG EXTEST, SAMPLE, BYPASS, IDCODE and STCTEST instructions are supported. The S/UNI-1x10GE identification code is 0x333920CD hexadecimal.

10.7 Microprocessor Interface

The Microprocessor Interface Block provides the logic required for interfacing the generic microprocessor bus with the normal mode and test mode registers within the S/UNI-1x10GE. The normal mode registers are used during normal operation to configure and monitor the S/UNI-1x10GE. The test mode registers are used to enhance the testability of the S/UNI-1x10GE. The register set is accessed as shown below. The address column of the table identifies the corresponding memory map address. Addresses that are not shown are not used and must be treated as Reserved.

Table 12 Register Memory Map

Address		Register Description
Hex Base	Hex End	
0x0000	0x0007	S/UNI-1x10GE Top Level
0x0008	0x203F	PM3392 Reserved
0x2040	0x207F	RXXG
0x2080	0x20BF	R64B66B
0x20C0	0x20FF	PM3392 Reserved
0x2100	0x21FF	MSTAT
0x2200	0x220F	IFLX
0x2210	0x223F	PM3392 Reserved
0x2240	0x224F	PL4MOS
0x2250	0x227F	PM3392 Reserved
0x2280	0x22BF	PL4ODP
0x22C0	0x22FF	PM3392 Reserved
0x2300	0x233F	PL4IO
0x2340	0x23FF	PM3392 Reserved
0x2400	0x241F	IRAM
0x2420	0x303F	PM3392 Reserved
0x3040	0x307F	TXXG
0x3080	0x30BF	T64B66B
0x30C0	0x31FF	PM3392 Reserved
0x3200	0x321F	EFLX
0x3220	0x327F	PM3392 Reserved
0x3280	0x32BF	PL4IDU
0x32C0	0x33FF	PM3392 Reserved
0x3400	0x341F	ERAM
0x3420	0xFFFF	PM3392 Reserved

11 Normal Mode Register Description

Normal mode registers are used to configure and monitor the operation of the S/UNI-1x10GE. Normal mode registers (as opposed to test mode registers) are selected when TRS (A[14]) is low.

Notes on Normal Mode Register Bits:

1. Writing values into **UNUSED** register bits has no effect. However, to ensure software compatibility with future, feature-enhanced versions of this product, unused register bits must be written with logic 0. Reading back unused bits can produce either a logic 1 or a logic 0; hence, unused register bits should be masked off by software when read.

2. All configuration bits that can be written into can also be read back. This allows the processor controlling the S/UNI-1x10GE to determine the programming state of the device.

3. Write-able normal mode register bits are cleared to logic 0 upon reset unless otherwise noted.

4. Writing into read-only normal mode register bit locations does not affect S/UNI-1x10GE operation unless otherwise noted.

5. Certain register bits are **RESERVED**. These bits are associated with mega cell functions that are reserved in this application. To ensure that the S/UNI-1x10GE operates as intended, **RESERVED** register bits must only be written with the logic level as specified. Writing to **RESERVED** registers should be avoided.

Table 13 Normal Mode Register Map

Address A[14:0]		Register Description
Top Level Registers		
0x0000H		Identification Register
0x0001H		Product Revision Register
0x0002H		Configuration and Reset Control Register
0x0003H		Master Interrupt Status Register
0x0004H		Device Status Register
0x0005H		Global Performance Monitor Update
0x0006H		MDIO Command Register
0x0007H		MDIO Interrupt Mask Register
0x0008H		MDIO Interrupt Register
0x0009H		MMD PHY Address Register
0x000AH		MMD Control Data Address Register
0x000BH		MMD Read Status Data Register
0x000CH	0x00FF	PM3392 Reserved
0x0100H		XSBI Configuration Register

Address A[14:0]		Register Description
0x0101H		Reserved
0x0102H		Reserved
0x0103H		XBSI RXFIFOSLIPI
0x0104H		XSBI Interrupt
0x0105H		Reserved
0x0106H		Reserved
0x0107H		XSBI Interrupt Mask2
0x0108H		XSBI Interrupt Mask3
0x0109H		XSBI RXOOLV
0x010AH		XSBI Analog Debug
0x0200H	0x020F	MDIO
0x010BH	0x203F	PM3392 Reserved
RXXG Specific Registers		
0x2040H		Configuration 1
0x2041H		Configuration 2
0x2042H		Configuration 3
0x2043H		Interrupt
0x2044H		Status
0x2045H		Maximum Frame Size
0x2046H		Station Address, Low 16 bits
0x2047H		Station Address, Middle 16 bits
0x2048H		Station Address, High 16 bits
0x2049H		Receive FIFO Threshold
0x204AH		Exact Match Address 0 Low Word
0x204BH		Exact Match Address 0 Mid Word
0x204CH		Exact Match Address 0 High Word
0x204DH		Exact Match Address 1 Low Word
0x204EH		Exact Match Address 1 Mid Word
0x204FH		Exact Match Address 1 High Word
0x2050H		Exact Match Address 2 Low Word
0x2051H		Exact Match Address 2 Mid Word
0x2052H		Exact Match Address 2 High Word
0x2053H		Exact Match Address 3 Low Word
0x2054H		Exact Match Address 3 Mid Word
0x2055H		Exact Match Address 3 High Word
0x2056H		Exact Match Address 4 Low Word
0x2057H		Exact Match Address 4 Mid Word
0x2058H		Exact Match Address 4 High Word
0x2059H		Exact Match Address 5 Low Word
0x205AH		Exact Match Address 5 Mid Word

Address A[14:0]		Register Description
0x205BH		Exact Match Address 5 High Word
0x205CH		Exact Match Address 6 Low Word
0x205DH		Exact Match Address 6 Mid Word
0x205EH		Exact Match Address 6 High Word
0x205FH		Exact Match Address 7 Low Word
0x2060H		Exact Match Address 7 Mid Word
0x2061H		Exact Match Address 7 High Word
0x2062H		Exact Match VID 0
0x2063H		Exact Match VID 1
0x2064H		Exact Match VID 2
0x2065H		Exact Match VID 3
0x2066H		Exact Match VID 4
0x2067H		Exact Match VID 5
0x2068H		Exact Match VID 6
0x2069H		Exact Match VID 7
0x206AH		Multicast Hash Low Word
0x206BH		Multicast Hash MidLow Word
0x206CH		Multicast Hash MidHigh Word
0x206DH		Multicast Hash High Word
0x206EH		Address Filter Control 0
0x206FH		Address Filter Control 1
0x2070H		Address Filter Control 2
0x2071H		Filter Error Counter
0x2072H	0x207F	RXXG Reserved
R64B66B Specific Registers		
0x2080H		R64B66B Configuration
0x2081H		R64B66B Interrupt Mask
0x2082H		R64B66B Interrupt Status
0x2083H		R64B66B Status
0x2084H		R64B66B Error Frame Count
0x2085H		R64B66B Error Lock Counter
0x2086H		R64B66B High Bit Error Rate
0x2087H	0x20BF	R64B66B Reserved
0x20C0H	0x20FF	PM3392 Reserved
MSTAT Specific Registers		
0x2100H		MSTAT Control
0x2101H		MSTAT Counter Rollover 0
0x2102H		MSTAT Counter Rollover 1
0x2103H		MSTAT Counter Rollover 2
0x2104H		MSTAT Counter Rollover 3

Address A[14:0]		Register Description
0x2105H		MSTAT Interrupt Mask 0
0x2106H		MSTAT Interrupt Mask 1
0x2107H		MSTAT Interrupt Mask 2
0x2108H		MSTAT Interrupt Mask 3
0x2109H		MSTAT Counter Write Address
0x210AH		MSTAT Counter Write Data Low
0x210BH		MSTAT Counter Write Data Middle
0x210CH		MSTAT Counter Write Data High
0x210DH	0x210F	MSTAT Reserved
0x2110H	Low	FramesReceivedOK
0x2111H	Med	
0x2112H	High	
0x2113H		MSTAT Reserved
0x2114H	Low	OctetsReceivedOK
0x2115H	Med	
0x2116H	High	
0x2117H		MSTAT Reserved
0x2118H	Low	FramesReceived
0x2119H	Med	
0x211AH	High	
0x211BH		MSTAT Reserved
0x211CH	Low	OctetsReceived
0x211DH	Med	
0x211EH	High	
0x211FH		MSTAT Reserved
0x2120H	Low	UnicastFramesReceivedOK
0x2121H	Med	
0x2122H	High	
0x2123H		MSTAT Reserved
0x2124H	Low	MulticastFramesReceivedOK
0x2125H	Med	
0x2126H	High	
0x2127H		MSTAT Reserved
0x2128H	Low	BroadcastFramesReceivedOK
0x2129H	Med	
0x212AH	High	
0x212BH		MSTAT Reserved
0x212CH	Low	TaggedFramesReceived
0x212DH	Med	
0x212EH	High	

Address A[14:0]		Register Description
0x212FH		MSTAT Reserved
0x2130H	Low	PAUSEMACControlFrameReceived
0x2131H	Med	
0x2132H	High	
0x2133H		MSTAT Reserved
0x2134H	Low	MACControlFrameReceived
0x2135H	Med	
0x2136H	High	
0x2137H		MSTAT Reserved
0x2138H	Low	FrameCheckSequenceErrors
0x2139H	Med	
0x213AH	High	
0x213BH		MSTAT Reserved
0x213CH	Low	FramesLostDueToInternalMACError
0x213DH	Med	
0x213EH	High	
0x213FH		MSTAT Reserved
0x2140H	Low	SymbolError
0x2141H	Med	
0x2142H	High	
0x2143H		MSTAT Reserved
0x2144H	Low	InRangeLengthErrors
0x2145H	Med	
0x2146H	High	
0x2147H		MSTAT Reserved
0x2148H	Low	Reserved
0x2149H	Med	
0x214AH	High	
0x214BH		MSTAT Reserved
0x214CH	Low	FramesTooLongErrors
0x214DH	Med	
0x214EH	High	
0x214FH		MSTAT Reserved
0x2150H	Low	Jabbers
0x2151H	Med	
0x2152H	High	
0x2153H		MSTAT Reserved
0x2154H	Low	Fragments
0x2155H	Med	
0x2156H	High	

Address A[14:0]		Register Description
0x2157H		MSTAT Reserved
0x2158H	Low	UndersizedFrames
0x2159H	Med	
0x215AH	High	
0x215BH		MSTAT Reserved
0x215CH	Low	ReceiveFrames64Octets
0x215DH	Med	
0x215EH	High	
0x215FH		MSTAT Reserved
0x2160H	Low	ReceiveFrames65to127Octets
0x2161H	Med	
0x2162H	High	
0x2163H		MSTAT Reserved
0x2164H	Low	ReceiveFrames128to255Octets
0x2165H	Med	
0x2166H	High	
0x2167H		MSTAT Reserved
0x2168H	Low	ReceiveFrames256to511Octets
0x2169H	Med	
0x216AH	High	
0x216BH		MSTAT Reserved
0x216CH	Low	ReceiveFrames512to1023Octets
0x216DH	Med	
0x216EH	High	
0x216FH		MSTAT Reserved
0x2170H	Low	ReceiveFrames1024to1518Octets
0x2171H	Med	
0x2172H	High	
0x2173H		MSTAT Reserved
0x2174H	Low	ReceiveFrames1519toMAXOctets
0x2175H	Med	
0x2176H	High	
0x2177H		MSTAT Reserved
0x2178H	Low	JumboOctetsReceivedOK
0x2179H	Med	
0x217AH	High	
0x217BH		MSTAT Reserved
0x217CH	Low	FilteredOctets
0x217DH	Med	
0x217EH	High	

Address A[14:0]		Register Description
0x217FH		MSTAT Reserved
0x2180H	Low	FilteredUnicastFrames
0x2181H	Med	
0x2182H	High	
0x2183H		MSTAT Reserved
0x2184H	Low	FilteredMulticastFrames
0x2185H	Med	
0x2186H	High	
0x2187H		MSTAT Reserved
0x2188H	Low	FilteredBroadcastFrames
0x2189H	Med	
0x218AH	High	
0x218BH		MSTAT Reserved
0x218CH	Low	Reserved
0x218DH	Med	
0x218EH	High	
0x218FH		MSTAT Reserved
0x2190H	Low	FramesTransmittedOK
0x2191H	Med	
0x2192H	High	
0x2193H		MSTAT Reserved
0x2194H	Low	OctetsTransmittedOK
0x2195H	Med	
0x2196H	High	
0x2197H		MSTAT Reserved
0x2198H	Low	OctetsTransmitted
0x2199H	Med	
0x219AH	High	
0x219BH		MSTAT Reserved
0x219CH	Low	FramesLostDueToInternalMACTransmissionError
0x219DH	Med	
0x219EH	High	
0x219FH		MSTAT Reserved
0x21A0H	Low	TransmitSystemError
0x21A1H	Med	
0x21A2H	High	
0x21A3H		MSTAT Reserved
0x21A4H	Low	UnicastFramesTransmittedAttempted
0x21A5H	Med	
0x21A6H	High	

Address A[14:0]		Register Description
0x21A7H		MSTAT Reserved
0x21A8H	Low	UnicastFramesTransmittedOK
0x21A9H	Med	
0x21AAH	High	
0x21ABH		MSTAT Reserved
0x21ACH	Low	MulticastFramesTransmittedAttempted
0x21ADH	Med	
0x21AEH	High	
0x21AFH		MSTAT Reserved
0x21B0H	Low	MulticastFramesTransmittedOK
0x21B1H	Med	
0x21B2H	High	
0x21B3H		MSTAT Reserved
0x21B4H	Low	BroadcastFramesTransmittedAttempted
0x21B5H	Med	
0x21B6H	High	
0x21B7H		MSTAT Reserved
0x21B8H	Low	BroadcastFramesTransmittedOK
0x21B9H	Med	
0x21BAH	High	
0x21BAH		MSTAT Reserved
0x21BCH	Low	PAUSEMACCTRLFramesTransmitted
0x21BDH	Med	
0x21BEH	High	
0x21BFH		MSTAT Reserved
0x21C0H	Low	MACCTRLFramesTransmitted
0x21C1H	Med	
0x21C2H	High	
0x21C3H		MSTAT Reserved
0x21C4H	Low	TransmittedFrames64Octets
0x21C5H	Med	
0x21C6H	High	
0x21C7H		MSTAT Reserved
0x21C8H	Low	TransmittedFrames65to127Octets
0x21C9H	Med	
0x21CAH	High	
0x21CBH		MSTAT Reserved
0x21CCH	Low	TransmittedFrames128to255Octets
0x21CDH	Med	
0x21CEH	High	

Address A[14:0]		Register Description
0x21CFH		MSTAT Reserved
0x21D0H	Low	TransmittedFrames256to511Octets
0x21D1H	Med	
0x21D2H	High	
0x21D3H		MSTAT Reserved
0x21D4H	Low	TransmittedFrames512to1023Octets
0x21D5H	Med	
0x21D6H	High	
0x21D7H		MSTAT Reserved
0x21D8H	Low	TransmittedFrames1024to1518Octets
0x21D9H	Med	
0x21DAH	High	
0x21DBH		MSTAT Reserved
0x21DCH	Low	TransmittedFrames1519toMAXOctets
0x21DDH	Med	
0x21DEH	High	
0x21DFH		MSTAT Reserved
0x21E0H	Low	JumboOctetsTransmittedOK
0x21E1H	Med	
0x21E2H	High	
0x21E3H		MSTAT Reserved
0x21E4H	Low	Reserved
0x21E5H	Med	
0x21E6H	High	
0x21E7H		MSTAT Reserved
0x21E8H		Reserved
0x21E9H		Reserved
0x21EAH	0x21FF	MSTAT Reserved
IFLX Specific Registers		
0x2200H		IFLX Global Configuration Register
0x2201H		IFLX Channel Provision
0x2202H		IFLX Global Status Register
0x2203H		IFLX SOF Error Enable
0x2204H		IFLX SOF Error Interrupt
0x2205H		IFLX EOF Error Enable
0x2206H		IFLX EOF Error Interrupt
0x2207H		Reserved
0x2208H		Reserved
0x2209H		IFLX FIFO Overflow Enable
0x220AH		IFLX FIFO Overflow Interrupt

Address A[14:0]		Register Description
0x220BH		IFLX Tag Processor Error Enable
0x220CH		Reserved
0x220DH		Reserved
0x220EH		IFLX Indirect Logical FIFO Low Limit & Provision
0x220FH		IFLX Indirect Logical FIFO High Limit
0x2210H		IFLX Indirect Full/Almost Full Status & Limit
0x2211H		IFLX Indirect Empty/Almost Empty Status & Limit
0x2212H		Reserved
0x2213H		Reserved
0x2214H		Reserved
0x2215H		Reserved
0x2216H		Reserved
0x2217H		Reserved
0x2218H	0x223F	IFLX Reserved
PL4MOS Specific Registers		
0x2240H		PL4MOS Configuration Register
0x2241H		PL4MOS Mask Register
0x2242H		PL4MOS Fairness Masking Register
0x2243H		PL4MOS MaxBurst1 Register
0x2244H		PL4MOS MaxBurst2 Register
0x2245H		PL4MOS Transfer Size Register
0x2247H		Reserved
0x2248H		Reserved
0x2249H		Reserved
0x224AH		Reserved
0x224BH		Reserved
0x224CH		Reserved
0x224DH		Reserved
0x224EH		Reserved
0x224FH		Reserved
0x2250H	0x225F	Reserved
0x2260H	0x226F	Reserved
0x2270H	0x227F	PL4MOS Reserved
PL4ODP Specific Registers		
0x2280H		PL4ODP Configuration
0x2281H		PL4ODP Status
0x2282H		PL4ODP Interrupt Mask
0x2283H		PL4ODP Interrupt
0x2284H		PL4ODP Configuration MAX_T Register
02285H		PL4ODP Elastic Store Limit

Address A[14:0]		Register Description
0x2286H		Reserved
0x2287H		Reserved
0x2288H		Reserved
0x2289H		Reserved
0x228AH		Reserved
0x228BH		Reserved
0x228CH		Reserved
0x228DH		Reserved
0x228EH		Reserved
0x228FH		Reserved
0x2290H		Reserved
0x2291H		Reserved
0x2292H		Reserved
0x2293H		Reserved
0x2294H		Reserved
0x2295H		Reserved
0x2296H		Reserved
0x2297H		Reserved
0x2298H		Reserved
0x2299H		Reserved
0x229AH		Reserved
0x229BH		Reserved
0x229CH		Reserved
0x229DH		Reserved
0x229EH	0x22FFF	PL4ODP Reserved
PL4IO Specific Registers		
0x2300H		PL4IO Lock Detect Status
0x2301H		PL4IO Lock Detect Change
0x2302H		PL4IO Lock Detect Mask
0x2303H		PL4IO Lock Detect Limits
0x2304H		PL4IO Calendar Repetitions
0x2305H		PL4IO Configuration
0x2306H		Reserved
0x2307H		Reserved
0x2308H		Reserved
0x2309H		Reserved
0x230AH		Reserved
0x230BH		Reserved
0x230CH		Reserved

Address A[14:0]		Register Description
0x230DH		Reserved
0x230E		Reserved
0x230FH		Reserved
0x2310H		Reserved
0x2311H		Reserved
0x2312H		
0x2313H		
0x2314H		
0x2315H		
0x2316H	0x231F	
0x2320H		
0x2321H		
0x2322H		
0x2323H		
0x2324H		
0x2325H		
0x2326H		
0x2327H		
0x2328H		
0x2329H		
0x232AH		
0x232BH		
0x232CH		
0x232DH		
0x232EH		
0x232FH		
0x2330H		
0x2331H	0x233F	PL4IO Reserved
0x2340H	0x23FF	PM3392 Reserved
TXXG Specific Registers		
0x3040H		Configuration Register 1
0x3041H		Configuration Register 2
0x3042H		Configuration Register 3
0x3043H		Interrupt Register
0x3044H		Status Register
0x3045H		Transmit Max Frame Size Register
0x3046H		Transmit Min Frame Size Register
0x3047H		Station Address, Low 16 bits
0x3048H		Station Address, Middle 16 bits
0x3049H		Station Address, High 16 bits

Address A[14:0]		Register Description
0x304AH		Diagnostic Register 1
0x304BH		Diagnostic Register 2
0x304CH		Diagnostic Status Register
0x304DH		PAUSE Timer Register
0x304EH		PAUSE Timer Interval Register
0x304FH		Packet Statistics Register Low 14 bits
0x3050H		Packet Statistics Register High 16 bits
0x3051H		Filter Error Count Register
0x3052		Pause Quantum Value Configuration
0x3053H	0x307F	TXXG Reserved
T64B66B Specific Registers		
0x3080H		T64B66B Configuration
0x3081H		T64B66B Interrupt Mask
0x3082H		T64B66B Interrupt Status
0x3083H		T64B66B Status
0x3084H		T64B66B Store Threshold
0x3085H	0x31FF	PM3392 Reserved
EFLX Specific Registers		
0x3200H		Global configuration
0x3201H		ERCU Global status
0x3202H		Indirect Channel Address
0x3203H		Indirect FIFO Low Limit
0x3204H		Indirect FIFO High Limit
0x3205H		Indirect Full/Almost-Full Status and Limit
0x3206H		Indirect Empty/Almost-Empty Status and Limit
0x3207H		Indirect FIFO Cut-Through Threshold
0x3208H		FIFO SOF Error Enable
0x3209H		FIFO SOF Error Indication
0x320AH		FIFO EOF Error Enable
0x320BH		FIFO EOF Error Indication
0x320CH		FIFO Overflow Error Enable
0x320DH		FIFO Overflow Error Indication
0x320EH		Invalid Channel Error Enable
0x320FH		Invalid Channel Error Indication
0x3210H		Channel Provision
0x3211H	0x3216	Unused, reserved
0x3217H		Reserved
0x3218H		Reserved
0x3219H		Reserved
0x321AH		Reserved

Address A[14:0]		Register Description
0x321BH		Reserved
0x321CH		Reserved
0x321DH	0x327F	PM3392 Reserved
PL4IDU Specific Registers		
0x3280H		PL4IDU Configuration
0x3281H		PL4IDU Status
0x3282H		PL4IDU Interrupt Mask
0x3283H		PL4IDU Interrupt
0x3284H		Reserved
0x3285H		Reserved
0x3286H		Reserved
0x3287H		Reserved
0x3288H		Reserved
0x3289H		Reserved
0x328AH		Reserved
0x328BH		Reserved
0x328CH		Reserved
0x328DH		Reserved
0x328EH		Reserved
0x328FH		Reserved
0x3290H		Reserved
0x3291H		Reserved
0x3292H		Reserved
0x3293H		Reserved
0x3294H		Reserved
0x3295H		Reserved
0x3296H		Reserved
0x3297H		Reserved
0x3298H		Reserved
0x3299H		Reserved
0x329AH		Reserved
0x329BH		Reserved
0x329CH		Reserved
0x329DH		Reserved
0x329EH		Reserved
0x329FH		Reserved
0x32A0H		Reserved
0x32A1H		Reserved
0x32A2H		Reserved
0x32A3H		Reserved

Address A[14:0]		Register Description
0x32A4H		Reserved
0x32A5H		Reserved
0x32A6H		Reserved
0x32A7H		Reserved
0x32A8H		Reserved
0x32A9H		Reserved
0x32AAH		Reserved
0x32ABH		Reserved
0x32ACH		Reserved
0x32ADH		Reserved
0x32AEH		Reserved
0x32AFH		Reserved
0x32B0H		Reserved
0x32B1H		Reserved
0x32B2H		Reserved
0x32B3H		Reserved
0x32B3H	0x32BF	PL4IDU Reserved
0x32C0H	0x33FF	PM3392 Reserved

Register 0x0000H:PM3392 Identification

Bit	Type	Function	Default
Bit 15	R	ID[15]	0
Bit 14	R	ID[14]	0
Bit 13	R	ID[13]	1
Bit 12	R	ID[12]	1
Bit 11	R	ID[11]	0
Bit 10	R	ID[10]	0
Bit 9	R	ID[9]	1
Bit 8	R	ID[8]	1
Bit 7	R	ID[7]	1
Bit 6	R	ID[6]	0
Bit 5	R	ID[5]	0
Bit 4	R	ID[4]	1
Bit 3	R	ID[3]	0
Bit 2	R	ID[2]	0
Bit 1	R	ID[1]	1
Bit 0	R	ID[0]	0

ID[15:0]

The Identification register presents a valid PMC product ID number for the device. This register is read only. The default value is 0x3392.

Register 0x0001H:PM3392 Product Revision

Bit	Type	Function	Default
Bit 15	R	REVISION[15]	0
Bit 14	R	REVISION[14]	0
Bit 13	R	REVISION[13]	0
Bit 12	R	REVISION[12]	0
Bit 11	R	REVISION[11]	0
Bit 10	R	REVISION[10]	0
Bit 9	R	REVISION[9]	0
Bit 8	R	REVISION[8]	0
Bit 7	R	REVISION[7]	0
Bit 6	R	REVISION[6]	0
Bit 5	R	REVISION[5]	0
Bit 4	R	REVISION[4]	0
Bit 3	R	REVISION[3]	0
Bit 2	R	REVISION[2]	0
Bit 1	R	REVISION[1]	1
Bit 0	R	REVISION[0]	1

REVISION[15:0]

This register is read only. This register presents the current device revision number. The default value is 0x0003.

Register 0x0002H:PM3392 Configuration and Reset Control

Bit	Type	Function	Default
Bit 15	R	TOP_CHAN[3]	0
Bit 14	R	TOP_CHAN[2]	0
Bit 13	R	TOP_CHAN[1]	0
Bit 12	R	TOP_CHAN[0]	0
Bit 11	R	Unused	0
Bit 10	R	Unused	0
Bit 9	R	Unused	0
Bit 8	R	Unused	0
Bit 7	R/W	Reserved	0
Bit 6	R	Unused	x
Bit 5	R	Unused	x
Bit 4	R	Unused	x
Bit 3	R	Unused	x
Bit 2	R/W	XSBI ARESETB	1
Bit 1	R/W	ARESETB	1
Bit 0	R/W	DRESETB	1

The Reset Control Register provides configuration related to reset operation on the PM3392.

DRESETB

The DRESETB bit (low-true) allows the digital circuitry in the PM3392 to be reset under software control. Setting this bit to logic 0 will cause the digital portion of the device to be held in reset. This bit is not self-clearing. Please refer to the operations section of this document for instructions concerning resetting the PM3392 device. Performing a hardware reset will set this bit to logic 1, thus negating the digital software reset.

ARESETB

The ARESETB bit (low-true) allows all analog circuitry related to the PL4 Bus analog circuitry to be reset under software control. Writing this bit to a logic 0 will cause the PL4 analog circuitry to be held in reset. This bit is not self-clearing and must be written to logic 1 to de-assert. Please refer to the operations section of this document for instructions concerning resetting the PM3392 device. Performing a hardware reset (that is, forcing the device RSTB pin to a logic 0) will set the ARESETB bit to logic 1, thus negating the analog software reset.

XSBI ARESETB

The XSBI ARESETB bit (low-true) allows all analog circuitry related to the XSBI Bus analog circuitry to be reset under software control. Writing this bit to a logic 0 will cause the XSBI analog circuitry to be held in reset. Once set to 0, internal clocks generated from the line side to the core will not be reliable. This bit is not self-clearing and must be written to logic 1 to de-assert.

TOP_CHAN[3:0]

TOP_CHAN[3:0] is the ID number of the highest active channel on the PM3392 device. This field is read only and set to 0.

Register 0x0003H:PM3392 Master Interrupt Status

Bit	Type	Function	Default
Bit 15	R/W	INTE	0
Bit 14	R	IRAM_INT	
Bit 13	R	ERAM_INT	
Bit 12	R	XSBI_INT	
Bit 11	R	MSTAT_INT	
Bit 10	R	RXXG_INT	
Bit 9	R	TXXG_INT	
Bit 8	R	R64B66B_INT	
Bit 7	R	T64B66B_INT	
Bit 6	R	MDIO_BUSY_INT	
Bit 5	R	PL4_DOOL_INT	
Bit 4	R	PL4_ROOL_INT	
Bit 3	R	IFLX_INT	
Bit 2	R	EFLX_INT	
Bit 1	R	PL4ODP_INT	
Bit 0	R	PL4IDU_INT	

This Master Interrupt Status register allows the source of an active interrupt to be identified down to the block level. Further register accesses are required to the block in question to determine the cause of an active interrupt and to acknowledge the interrupt source. Reading of this register has no side effects. Interrupt status is set to logic 1 to indicate a pending interrupt of the specified type.

PL4IDU_INT

Interrupt Status indicator for the PL4IDU block.

PL4ODP_INT

Interrupt Status indicator for the PL4ODP block.

EFLX_INT

Interrupt Status indicator for the EFLX block.

IFLX_INT

Interrupt Status indicator for the IFLX block.

PL4_ROOL_INT

PL4 Reference Out Of Lock Interrupt Status. This is the Interrupt Status indicator for the PL4IO block indicating that the Reference Out Of Lock condition has changed state.

PL4IO_DOOL_INT

PL4IO Data Out Of Lock Interrupt Status. This is the Interrupt Status indicator for the PL4IO block indicating that the Data Out Of Lock condition has changed state.

MDIO_BUSY_INT

MDIO Busy Int is asserted with the rising edge of LCTLD, LCTLA, RSTAT, or RDINC in the MDIO Command Register. This indicates that the management interface has an operation in progress.

T64B66B_INT

Interrupt Status indicator for the T64B66B block.

R64B66B_INT

Interrupt Status indicator for the R64B66B block.

MSTAT_INT

Interrupt Status indicator for one of the ten MSTAT blocks. Identification of which MSTAT channels have active interrupts can be done using the PM3392 MSTAT Interrupt Status register.

XSBI_INT

Interrupt Status indicator for the XSBI block.

ERAM_INT

Interrupt Status indicator for the EFLX RAM block.

IRAM_INT

Interrupt Status indicator for the IFLX RAM block.

INTE

The interrupt enable (INTE) bit controls the assertion of the interrupt (INTB) output. When logic 1 is written to INTE, the pending interrupt(s) listed in this register will assert the interrupt (INTB) output. When logic 0 is written to INTE, the pending interrupt(s) will not assert the interrupt (INTB) output.

Register 0x0004H:PM3392 Device Status

Bit	Type	Function	Default
Bit 15	R	Unused	
Bit 14	R	Unused	
Bit 13	R	Unused	
Bit 12	R	Unused	
Bit 11	R	Unused	
Bit 10	R	Unused	
Bit 9	R	Unused	
Bit 8	R	MDIO_BUSY	
Bit 7	R	DTRB	
Bit 6	R	EXPIRED	
Bit 5	R	PAUSED	
Bit 4	R	PL4_ID_DOOL	
Bit 3	R	PL4_IS_DOOL	
Bit 2	R	PL4_ID_ROOL	
Bit 1	R	PL4_IS_ROOL	
Bit 0	R	PL4_OUT_ROOL	

The PM3392 Device Status Register provides the ability to monitor device operation.

PL4_OUT_ROOL

PL4 Output Reference Out Of Lock status. PL4_OUT_ROOL is a logic 1 if the synthesized clock associated with the PL4 output interface is not trained to the reference frequency. PL4_OUT_ROOL is a logic 0 otherwise. The PL4 data and status output interfaces (PM3392 device pins RDCLK+/-, RDAT[15:0]+/-, RCTL+/-, TSTAT[1:0], and TSCLK) are normally disabled when PL4_OUT_ROOL is asserted. All of the PL4IO interfaces share a single clock synthesizer.

PL4_IS_ROOL

PL4 Input Status Reference Out Of Lock Condition. PL4_IS_ROOL is a logic 1 if the input FIFO status clock (PM3392 device pin RSCLK) frequency exceeds ¼ the frequency of device pin RDCLK+/-. PL4_IS_ROOL is a logic 0 otherwise. The PL4 input status interface (PM3392 device pins RSTAT[1:0]) is normally disabled when PL4_IS_ROOL is a logic 1.

PL4_ID_ROOL

PL4 Input Data Reference Out Of Lock Condition. PL4_ID_ROOL is a logic 1 if the parallel data input clock (PM3392 device pin TDCLK+/-) or one or more of the parallel data input streams (PM3392 device pins TDAT[15:0]+/- and TCTL+/-) is not trained to the local synthesized clock. PL4_ID_ROOL is a logic 0 otherwise. The PL4 data input interface is normally disabled when PL4_ID_ROOL is asserted.

PL4_IS_DOOL

PL4 Input Status Out Of Lock Condition. PL4_IS_DOOL is a logic 1 if the parallel FIFO status input stream is not properly synchronized. PL4_IS_DOOL is a logic 0 otherwise.

PL4_ID_DOOL

PL4 Input Data Out Of Lock Condition. PL4_ID_DOOL is a logic 1 if the parallel input data stream is not properly aligned. PL4_ID_DOOL is a logic 0 otherwise.

PAUSED

The PAUSED signal indicates the reception and execution of MAC Control PAUSE frames on the PM3392. If PAUSED is logic 1, then the PM3392 is in the PAUSED state and MAC Data frame transmission is blocked. If PAUSED is a logic 0, then it is not in a PAUSED state.

EXPIRED

EXPIRED is used by the device to deassert digital reset (DRSTB) after the analog clocks have had time to stabilize. This indicates that the PL4 analog block has come out of reset. EXPIRED is deasserted when PRSTB is asserted. EXPIRED is asserted 10.0 to 13.5 ms (nominal) after PRSTB is deasserted, and remains asserted thereafter.

DTRB

Digital Timer Reset, DTRB, is used to override the internal timer supplied by the PL4 analog. The internal timer determines the amount of time before clocks are reliable from the clock recovery logic. DTRB is an external pin used during power-up reset to override this timer. This pin is normally held low, but driven high to override.

MDIO_BUSY

MDIO BUSY is used to indicate that the MDIO block has an operation in progress. The register can be used for polling.

Register 0x0005H:PM3392 Global Performance Monitor Update

Bit	Type	Function	Default
Bit 15	R/W	TIP	X
Bit 14	R	Unused	X
Bit 13	R	Unused	X
Bit 12	R	Unused	X
Bit 11	R	Unused	X
Bit 10	R	Unused	X
Bit 9	R	Unused	X
Bit 8	R	Unused	X
Bit 7	R	Unused	X
Bit 6	R	Unused	X
Bit 5	R	Unused	X
Bit 4	R	Unused	X
Bit 3	R	Unused	X
Bit 2	R	Unused	X
Bit 1	R	Unused	X
Bit 0	R	Unused	X

Writing to this register performs a global performance monitor update by simultaneously loading all the performance meter registers in the PL4IDU and PL4ODP blocks.

TIP

The TIP bit is set to a logic one when the performance meter registers are being loaded. Writing to this register with DRESET equal to logic 0 initiates an accumulation interval transfer and loads all the performance meter registers in the PM3392.

TIP remains logic 1 while the transfer is in progress, and is set to a logic zero when the transfer is complete. TIP can be polled by a microprocessor to determine when the accumulation interval transfer is complete.

Register 0x0006H:MDIO Command Register

Bit	Type	Function	Default
Bit 15	R	Unused	X
Bit 14	R	Unused	X
Bit 13	R	Unused	X
Bit 12	R	Unused	X
Bit 11	R	Unused	X
Bit 10	R	Unused	X
Bit 9	R	Unused	X
Bit 8	R	Unused	X
Bit 7	R	Unused	X
Bit 6	R	Unused	X
Bit 5	R	Unused	X
Bit 4	R/W	RDINC	0
Bit3	R/W	RSTAT	0
Bit 2	R/W	LCTLD	0
Bit 1	R/W	LCTLA	0
Bit 0	R/W	SPRE	0

SPRE

This bit controls the preamble suppression in the MDIO block.

When ‘0’ the MDIO sends a sequence of 32 contiguous logic one bit on the MDO pin that can be used to establish synchronization. When ‘1’ the MDIO suppresses the 32-bit preamble. This bit must be written to the inactive state to complete the operation or begin a new operation.

LCTLA

A 0 to 1 transition on this bit causes the MDIO block to send a management frame with payload containing address of the register to be accessed in the subsequent operation. This bit must be written to the inactive state to complete the operation or begin a new operation.

LCTLD

A 0 to 1 transition on this bit causes the MDIO block to send a management frame with payload containing control data to be written to the register whose address was provided in the previous address frame. This bit must be written to the inactive state to complete the operation or begin a new operation.

RSTAT

A 0 to 1 transition on this bit causes the MDIO block to send a read operation management frame. The MDIO reads status from the external MII PHY register whose address is specified in the previous address frame. This bit must be written to the inactive state to complete the operation or begin a new operation.

RDINC

A 0 to 1 transition on this bit causes the MDIO block to send a post-read-increment-address operation management frame. This bit must be written to the inactive state to complete the operation or begin a new operation.

Register 0x0007H:MDIO Interrupt Mask Register

Bit	Type	Function	Default
Bit 15	R	Unused	X
Bit 14	R	Unused	X
Bit 13	R	Unused	X
Bit 12	R	Unused	X
Bit 11	R	Unused	X
Bit 10	R	Unused	X
Bit 9	R	Unused	X
Bit 8	R	Unused	X
Bit 7	R	Unused	X
Bit 6	R	Unused	X
Bit 5	R	Unused	X
Bit 4	R	Unused	X
Bit 3	R	Unused	X
Bit 2	R	Unused	X
Bit 1	R	Unused	X
Bit 0	R/W	BUSYE	0

BUSYE

When ‘1’, BUSYI in the MDIO Interrupt Register will assert the INT pin when interrupts are enabled.

When ‘0’, the BUSYI interrupt will have no effect on the interrupt pin.

Register 0x0008H:MDIO Interrupt Register

Bit	Type	Function	Default
Bit 15	R	Unused	X
Bit 14	R	Unused	X
Bit 13	R	Unused	X
Bit 12	R	Unused	X
Bit 11	R	Unused	X
Bit 10	R	Unused	X
Bit 9	R	Unused	X
Bit 8	R	Unused	X
Bit 7	R	Unused	X
Bit 6	R	Unused	X
Bit 5	R	Unused	X
Bit 4	R	Unused	X
Bit 3	R	Unused	X
Bit 2	R	Unused	X
Bit 1	R	Unused	X
Bit 0	R	BUSYI	0

BUSYI

When ‘1’ indicates MDIO operation is in progress, Busy is asserted with the rising edge of LCTLD, LCTLA, RSTAT, or RDINC in the MDIO Command Register.

When ‘0’ indicates the end of operation.

Register 0x0009H:MMD PHY Address Register

Bit	Type	Function	Default
Bit 15	R	Unused	X
Bit 14	R	Unused	X
Bit 13	R	Unused	X
Bit 12	R/W	DEVADR[4]	0
Bit 11	R/W	DEVADR[3]	0
Bit 10	R/W	DEVADR[2]	0
Bit 9	R/W	DEVADR[1]	0
Bit 8	R/W	DEVADR[0]	0
Bit 7	R	Unused	X
Bit 6	R	Unused	X
Bit 5	R	Unused	X
Bit 4	R/W	PRTADR[4]	0
Bit3	R/W	PRTADR[3]	0
Bit 2	R/W	PRTADR[2]	0
Bit 1	R/W	PRTADR[1]	0
Bit 0	R/W	PRTADR[0]	0

PRTAD[4:0]

This is a 5-bit port address allowing 32 unique port addresses per MDIO. The first port address bit to be transmitted is the MSB of the address.

DEVADR[4:0]

This is a 5-bit device address, allows 32 unique devices per port. The first device address bit to be transmitted is the MSB of the address.

Register 0x000AH:MMD Control Address Data Register

Bit	Type	Function	Default
Bit 15	R/W	CTLAD[15]	0
Bit 14	R/W	CTLAD[14]	0
Bit 13	R/W	CTLAD[13]	0
Bit 12	R/W	CTLAD[12]	0
Bit 11	R/W	CTLAD[11]	0
Bit 10	R/W	CTLAD[10]	0
Bit 9	R/W	CTLAD[9]	0
Bit 8	R/W	CTLAD[8]	0
Bit 7	R/W	CTLAD[7]	0
Bit 6	R/W	CTLAD[6]	0
Bit 5	R/W	CTLAD[5]	0
Bit 4	R/W	CTLAD[4]	0
Bit 3	R/W	CTLAD[3]	0
Bit 2	R/W	CTLAD[2]	0
Bit 1	R/W	CTLAD[1]	0
Bit 0	R/W	CTLAD[0]	0

CTLAD[15:0]

Control Address/Data input bus. This register is used for indexing a 16-bit address into the external device during a control sequence. When expecting a data during a write command, this register will contain the resultant data.

Register 0x000BH:MDIO Read Status Data Register

Bit	Type	Function	Default
Bit 15	R	PRSD[15]	0
Bit 14	R	PRSD[14]	0
Bit 13	R	PRSD[13]	0
Bit 12	R	PRSD[12]	0
Bit 11	R	PRSD[11]	0
Bit 10	R	PRSD[10]	0
Bit 9	R	PRSD[9]	0
Bit 8	R	PRSD[8]	0
Bit 7	R	PRSD[7]	0
Bit 6	R	PRSD[6]	0
Bit 5	R	PRSD[5]	0
Bit 4	R	PRSD[4]	0
Bit 3	R	PRSD[3]	0
Bit 2	R	PRSD[2]	0
Bit 1	R	PRSD[1]	0
Bit 0	R	PRSD[0]	0

PRSD[15:0]

Read Status Data. The 16-bit result from the MDIO read operation whose address is specified in the previous address frame.

Register 0x0100H:XSBI Wrapper Configuration Register

Bit	Type	Function	Default
Bit 15	R	Reserved	
Bit 14	R	Reserved	
Bit 13	R	Unused	
Bit 12	R	Unused	
Bit 11	R	Unused	
Bit 10	R	Reserved	
Bit 9	R/W	SYNC_ERR_INV	0
Bit 8	R/W	XSBI_EN	0
Bit 7	R/W	ATMEN	0
Bit 6	R	Reserved	0
Bit 5	R/W	LOCAL_LOOPBACK_EN	0
Bit 4	R/W	OIFS_EN	1
Bit3	R/W	MPGM_MON_EN	0
Bit 2	R/W	MPGM_GEN_EN	0
Bit 1	R/W	BIT_SWIZZLE	0
Bit 0	R/W	PHASE_INIT	0

PHASE_INIT

This bit controls the logic level of the PHASE_INIT output pin.

When '0' sets the PHASE_INIT output pin low.
When '1' sets the PHASE_INIT output pin high

BIT_SWIZZLE

When '0' the chip is in the default LAN mode and the bit order is not altered

When '1', it reverses the bit order in the 16-bit RXDATAi_j (I=1 to 4, j=1 to 4) on the RX side and TXDATAi_j (I=1 to 4, j=1 to 4) on the TX side.

RX side:

RXDATA4_4 becomes RXDATA1_1
RXDATA4_3 becomes RXDATA1_2
RXDATA1_1 becomes RXDATA4_4

MPGM_GEN_EN

When ‘0’, the XSBI_Wrapper is in normal functional mode and the MPGM generator on the transmit side is disabled.

When ‘1’, the XSBI_Wrapper is in diagnostic mode and the MPGM generator on the transmit side is enabled. This bit enables the transmit PRBS testing.

MPGM_MON_EN

When ‘0’, the XSBI_Wrapper is in normal functional mode and the MPGM Monitor on the receive side is disabled.

When ‘1’, the XSBI_Wrapper is in diagnostic mode and the MPGM generator on the receive side is enabled. This bit enables the receive PRBS testing.

OIFS_EN (input to the OIFS ABC)

When ‘1’, OIFS ABC is in normal operation.

When ‘0’, Power to the OIFS ABC is shut down.

LOCAL_LOOPBACK_EN

When '0' the XSBI_Wrapper is in normal functional mode.

When '1' the XSBI_Wrapper is in diagnostic mode. The Local System side loopback mode is enabled. The transmit data from the transmit PCS is looped back to the Receive PCS. To write to this bit the XSBI_EN bit (bit 8) must be set to 0 first, then write the LOCAL_LOOPBACK_EN bit.

ATMEN (input to the OIFS ABC)

When ‘0’ OIFS ABC is in Normal Operational Mode.

When ‘1’ OIFS ABC is in Analog Test mode.

XSBI_EN

When '0 the data will not be forwarded to and from the PCS block. When '1' the XSBI_Wrapper block will be in normal operational mode.

SYNC_ERR_INV

SYNC_ERR_INV is used to supply an inversion feature when the external driving pin is SIGNAL_DETECT instead of SYNC_ERR. This will make the PM3392 more compatible with other optics.

When '0', the sync error input signal will be unaltered.

When '1', the sync error input will be inverted.

Bit	Type	Function	Default
Bit 15	R	RXOOLI15	0
Bit 14	R	RXOOLI14	0
Bit 13	R	RXOOLI13	0
Bit 12	R	RXOOLI12	0
Bit 11	R	RXOOLI11	0
Bit 10	R	RXOOL10	0
Bit 9	R	RXOOLI9	0
Bit 8	R	RXOOLI8	0
Bit 7	R	RXOOLI7	0
Bit 6	R	RXOOLI6	0
Bit 5	R	RXOOLI5	0
Bit 4	R	RXOOLI4	0
Bit 3	R	RXOOLI3	0
Bit 2	R	RXOOLI2	0
Bit 1	R	RXOOLI1	0
Bit 0	R	RXOOLI0	0

Register 0x0104H:XSBI Interrupt Status

Bit	Type	Function	Default
Bit 15	R	Reserved	0:high
Bit 14	R	Reserved	0:high
Bit 13	R	Unused	X:high
Bit 12	R	Unused	X:high
Bit 11	R	Unused	X:high
Bit 10	R	Unused	X:high
Bit 9	R	Unused	X:high
Bit 8	R	Unused	X:high
Bit 7	R	Unused	X:high
Bit 6	R	Unused	X:high
Bit 5	R	Unused	X:high
Bit 4	R	Unused	X:high
Bit 3	R	Unused	X:high
Bit 2	R	TXFIFOSLIPI	0
Bit 1	R	SYNC_ERRI	0
Bit 0	R	PHASE_ERRI	0

Phase_erri

When ‘1’ indicates that the input pin PHASE_ERR is asserted and output clock pin TXCLK2+/- is not aligned with the corresponding output data bus TXDATA+/-.

Sync_erri

When '1' indicates that the input pin SYNC_ERR is asserted and input data, RXDATA+/- is not derived from the optical line and is suspect.

Txfifoslipi

When ‘1’ indicates that there has been an underflow or overflow in the TX_SYNC_FIFO when the local system side loopback mode is enabled.

Bit	Type	Function	Default
Bit 15	R/W	ENABLE0[15]	0
Bit 14	R/W	ENABLE0[14]	0
Bit 13	R/W	ENABLE0[13]	0
Bit 12	R/W	ENABLE0[12]	0
Bit 11	R/W	ENABLE0[11]	0
Bit 10	R/W	ENABLE0[10]	0
Bit 9	R/W	ENABLE0[9]	0

Bit	Type	Function	Default
Bit 8	R/W	ENABLE0[8]	0
Bit 7	R/W	ENABLE0[7]	0
Bit 6	R/W	ENABLE0[6]	0
Bit 5	R/W	ENABLE0[5]	0
Bit 4	R/W	ENABLE0[4]	0
Bit 3	R/W	ENABLE0[3]	0
Bit 2	R/W	ENABLE0[2]	0
Bit 1	R/W	ENABLE0[1]	0
Bit 0	R/W	ENABLE0[0]	0

Register 0x0107H:XSBI Interrupt ENABLE2

Bit	Type	Function	Default
Bit 15	R/W	ENABLE2[15]	0
Bit 14	R/W	ENABLE2[14]	0
Bit 13	R/W	ENABLE2[13]	0
Bit 12	R/W	ENABLE2[12]	0
Bit 11	R/W	ENABLE2[11]	0
Bit 10	R/W	ENABLE2[10]	0
Bit 9	R/W	ENABLE2[9]	0
Bit 8	R/W	ENABLE2[8]	0
Bit 7	R/W	ENABLE2[7]	0
Bit 6	R/W	ENABLE2[6]	0
Bit 5	R/W	ENABLE2[5]	0
Bit 4	R/W	ENABLE2[4]	0
Bit 3	R/W	ENABLE2[3]	0
Bit 2	R/W	ENABLE2[2]	0
Bit 1	R/W	ENABLE2[1]	0
Bit 0	R/W	ENABLE2[0]	0

ENABLE2[15:0]

When ‘1’, the corresponding RXFIFOSLIPI interrupt will assert the INT pin when interrupts are enabled.

When ‘0’, the RXFIFOSLIPI [0-15] interrupt will have no effect on the interrupt pin.

Register 0x0108H:XSBI Interrupt ENABLE3

Bit	Type	Function	Default
Bit 15	R/W	Unused	X:high
Bit 14	R/W	Unused	X:high
Bit 13	R/W	Unused	X:high
Bit 12	R/W	Unused	X:high
Bit 11	R/W	Unused	X:high
Bit 10	R/W	Unused	X:high
Bit 9	R/W	Unused	X:high
Bit 8	R/W	Unused	X:high
Bit 7	R/W	Unused	X:high
Bit 6	R/W	Unused	X:high
Bit 5	R/W	Unused	X:high
Bit 4	R/W	Unused	X:high
Bit 3	R/W	Unused	X:high
Bit 2	R/W	ENABLE3[2]	0
Bit 1	R/W	ENABLE3[1]	0
Bit 0	R/W	ENABLE3[0]	0

ENABLE3[0]

When ‘1’, PHASE_ERRI interrupt will assert the INT pin when interrupts are enabled.

When ‘0’, the PHASE_ERRI interrupt will have no effect on the interrupt pin.

ENABLE3[1]

When ‘1’, SYNC_ERRI interrupt will assert the INT pin when interrupts are enabled.

When ‘0’, the SYNC_ERRI interrupt will have no effect on the interrupt pin.

ENABLE3[0]

When ‘1’, TXFIFOSLIPI interrupt will assert the INT pin when interrupts are enabled.

When ‘0’, the TXFIFOSLIPI interrupt will have no effect on the interrupt pin.

Register 0x0109H:XSBI RXOOLV

Bit	Type	Function	Default
Bit 15	R	RXOOLV15	0
Bit 14	R	RXOOLV14	0
Bit 13	R	RXOOLV13	0
Bit 12	R	RXOOLV12	0
Bit 11	R	RXOOLV11	0
Bit 10	R	RXLOCK10	0
Bit 9	R	RXOOLV9	0
Bit 8	R	RXOOLV8	0
Bit 7	R	RXOOLV7	0
Bit 6	R	RXOOLV6	0
Bit 5	R	RXOOLV5	0
Bit 4	R	RXOOLV4	0
Bit 3	R	RXOOLV3	0
Bit 2	R	RXOOLV2	0
Bit 1	R	RXOOLV1	0
Bit 0	R	RXOOLV0	0

RXOOLV0 – RXOOLV15

The Out of Lock Status (RXOOLV) reflects the state of the MPGM monitor's state machine. When RXOOLV [0-15] is set high, the MPGM [0-15] monitor state machine is in the Out of Locked State. When OOLV is low, the monitor is in Lock State.

Register 0x010AH:XSBI Analog Debug Register

Bit	Type	Function	Default
Bit 15	R/W	Unused	X:high
Bit 14	R/W	Unused	X:high
Bit 13	R/W	Unused	X:high
Bit 12	R/W	Unused	X:high
Bit 11	R/W	Unused	X:high
Bit 10	R/W	Unused	X:high
Bit 9	R/W	TIN[3]	0
Bit 8	R/W	TIN[2]	0
Bit 7	R/W	TIN[1]	0
Bit 6	R/W	TIN[0]	0
Bit 5	R/W	ATMS[5]	0
Bit 4	R/W	ATMS[4]	0
Bit 3	R/W	ATMS[3]	0
Bit 2	R/W	ATMS[2]	0
Bit 1	R/W	ATMS[1]	0
Bit 0	R/W	ATMS[0]	0

ATMS[5:0]

Analog Test Mode Select bus ATMS[5:0] is used to select the block to be tested in the OIFS ABC in analog test mode.

TIN[3:0]

TIN[3:0] is used to select the test points in the block selected by ATMS[5:0] in the OIFS ABC in analog test mode.

Register 0x2040H:RXXG Configuration 1

Bit	Type	Function	Default
Bit 15	R/W	RXEN	0
Bit 14	R/W	ROCF	0:high
Bit 13	R/W	PAD_STRIP	0:high
Bit 12	R/W	Reserved	0
Bit 11	R/W	Reserved	0
Bit 10	R/W	PUREP	1
Bit 9	R/W	LONGP	0
Bit 8	R/W	PARF	0
Bit 7	R/W	FLCHK	0
Bit 6	R/W	Reserved	0
Bit 5	R/W	PASS_CTRL	0
Bit 4	R/W	RX_CONTIG	1
Bit 3	R/W	CRC_STRIP	0
Bit 2	R/W	Reserved	1
Bit 1	R/W	Reserved	0
Bit 0	R/W	Reserved	0

RXEN

When ‘1’, Receive packet parsing and transfer is enabled. When ‘0’, no packet parsing will be initiated, but a packet reception in progress will be completed. RXXG should be configured before writing this bit as ‘1’.

ROCF

Respond to Oversize Control Frames. When ‘1’, the RXXG will pause when an oversize (>=64) byte pause frame is received. When ‘0’ the RXXG will ONLY pause when 64 byte pause frames are received.

PAD_STRIP

When ‘1’, the RXXG will pad strip all non-errored and non-vlan tagged packets received. Pad stripping means that all frames will be truncated to the length field specified in the frame. When ‘0’, pad stripping will be disabled. Note that if CRC_STRIP must be set to ‘1’ when PAD_STRIP = ‘1’. Behavior is not defined if CRC_STRIP is ‘0’ and PAD_STRIP = ‘1’.

NOTE: if Frame Length Check is enabled and pad strip is also enabled, packets padded up to 64 bytes are the only padded packets that will not generate a frame length check error. VLAN tagged frames are not pad stripped.

RESERVED

This bit must be programmed to 0. Operation is not guaranteed otherwise.

PUREP

Pure Preamble. Set this bit to cause RXXG to check the contents of the preamble field of the packet, ensuring a data pattern of 0x55. Clear this bit if no preamble checking is desired.

LONGP

Long Preamble. Set this bit to permit reception of packets with any number of preamble. Clear this bit to cause RXXG to discard packets with >11 bytes of preamble.

PARF

Pass All Receive Frames. When this bit is '0', RXXG will process PAUSE Control frames, extracting the pause parameter from the packet to transfer to the TXXG and then discard the packet. When this bit is '1', RXXG will not process PAUSE Control frames and will forward the packet (provided it is not filtered for error reasons).

NOTE: If PARF= 1; The forwarding of PAUSE frames only relies on the MAC Control frame type of 0x8808 and the PAUSE opcode of 0x0001 and no errors associated with the received frame.

FLCHK

Frame Length Check. When this bit is '1', RXXG will compare the actual frame length against the value contained in the Type/Length field of the MAC header. This check is only done if the Type/Length field is in the range (0-1518). If a length mismatch occurs, the frame will be discarded. When this bit is '0', this check is not performed.

NOTE: if Frame Length Check is enabled and pad strip is also enabled, packets padded up to 64 bytes are the only padded packets that will not generate a frame length check error. Also VLAN tagged frames will not generate a frame length check error.

PASS_CTRL

Pass MAC Control frames. When this bit is '1', RXXG will forward MAC Control non-PAUSE frames to the System interface. When this bit is '0', RXXG will filter these frames. MAC Control PAUSE frame processing and filtering is determined by PARF bit.

RX_CONTIG

Receive Packet Contiguous. This configuration register bit must ALWAYS be sent to 0 on initialization. Operation is not guaranteed if this bit is set to 1.

CRC_STRIP

CRC Strip Register Bit. When this bit is '1', RXXG will strip the CRC from the packet. When this bit is '0', the packet is forwarded through RXXG with the CRC appended

Register 0x2042H:RXXG Configuration 3

Bit	Type	Function	Default
Bit 15	R/W	MIN_LERRE	0
Bit 14	R/W	MAX_LERRE	0
Bit 13	R/W	ODD_ALIGNE	0
Bit 12	R/W	LINE_ERRE	0
Bit 11	R/W	Reserved	0
Bit 10	R/W	RX_OVRE	0
Bit 9	R/W	ADR_FILTERE	0
	R/W	ERR_DETECTE	0
Bit 7	R/W	Reserved	0
Bit 6	R/W	Reserved	0
Bit 5	R/W	PRMB_ERRE	0
Bit 4	R	Unused	X:high
Bit 3	R	Unused	X:high
Bit 2	R	Unused	X:high
Bit 1	R	Unused	X:high
Bit 0	R	Unused	X:high

MIN_LERRE

The MIN_LERRE bit enables the generation of an interrupt due to a packet sourced from the Line Interface being less that the minimum frame size of 64 bytes. The frame size does not include preamble/ SFD bytes or prepended frame delineation headers. It just covers the MAC frame contents including CRC.

MAX_LERRE

The MAX_LERRE bit enables the generation of an interrupt due to a packet sourced from the Line interface exceeding the maximum frame size programmed in the Maximum Frame Length register. The frame size does not include preamble/ SFD bytes or prepended frame delineation headers. It just covers the MAC frame contents including CRC.

ODD_ALIGNE

The ODD_ALIGNE bit enables the generation of an interrupt due to a packet sourced from the Line interface not starting on a 32-bit alignment. This is in support of the requirements for the 64b/66b PCS framing schemes, which are constrained to 32-bit start alignment. This interrupt indicates a possible framing error.

LINE_ERRE

The LINE_ERRE bit enables the generation of an interrupt due to any Line interface errors. See the description of LINE_ERRI in the RXXG Interrupt register for a description of errors.

RX_OVRE

The RX_OVRE bit enables the generation of an interrupt due to a receive overrun causes by upstream blockage subsequently causing an overrun of the RXXG internal FIFO.

ADR_FILTERE

The ADR_FILTERE bit enables the generation of an interrupt whenever a packet is filtered by RXXG, due to MAC Address filter functions.

ERR_DETECTE

This bit is set when a protocol error(CRC, Frame Length, Range Check) is seen in a packet. The packet will be filtered by RXXG when PASS_ERRORS = '0' and the packet is less than the programmable FIFO Threshold value, otherwise the packet is forwarded and marked as errored

PRMB_ERRE

PRMB_ERRE enables the PRMB_ERRI bit generating an interrupt output (INT). PRMB_ERRI is set when a packet is received that violates the preamble expected. The expected preamble is dependent on PUREP and LONGP in RXXG Configuration Register 1. If PRMB_ERRE is ‘1’ in RXXG Configuration 3 register, an interrupt will also be generated (INT output asserted).

Register 0x2043H:RXXG Interrupt

Bit	Type	Function	Default
Bit 15	R	MIN_LERRI	0
Bit 14	R	MAX_LERRI	0
Bit 13	R	ODD_ALIGNI	0
Bit 12	R	LINE_ERRI	0
Bit 11	R	Reserved	0
Bit 10	R	RX_OVRI	0
Bit 9	R	ADR_FILTERI	0
Bit 8	R	ERR_DETECTI	0
Bit 7	R	Reserved	0
Bit 6	R	Reserved	0
Bit 5	R	PRMB_ERRI	0
Bit 4	R	Unused	X:high
Bit 3	R	Unused	X:high
Bit 2	R	Unused	X:high
Bit 1	R	Unused	X:high
Bit 0	R	Unused	X:high

MIN_LERRI

The MIN_LERRI bit will be set when the packet sourced from the Line Interface is less than the legal minimum of 64 bytes. The frame size does not include preamble/ SFD bytes or prepended frame delineation headers. It just covers the MAC frame contents including CRC. If MIN_LERRE is ‘1’ in RXXG Configuration 3 register, an interrupt will also be generated (INT output asserted).

MAX_LERRI

The MAX_LERRI bit will be set when the packet sourced from the Line Interface exceeds the maximum frame size programmed in the Maximum Frame Length register. The frame size does not include preamble/ SFD bytes or prepended frame delineation headers. If MAX_LERRE is ‘1’ in RXXG Configuration 3 register, an interrupt will also be generated (INT output asserted)

ODD_ALIGNI

The ODD_ALIGNI bit is set when the packet sourced from the Line interface does not start on an even 32-bit alignment. This is in support of the requirements for XAUI and 64b/66b PCS framing schemes which are constrained to this start alignment. RXXG will accept frames on any byte alignment. This interrupt is just an indication of a possible framing error. If ODD_ALIGNE is ‘1’ in RXXG Configuration 3 register, an interrupt will also be generated (INT output asserted).

LINE_ERRI

The LINE_ERRI bit is set when a Line Interface error is detected. If LINE_ERRE is '1' in RXXG Configuration 3 register, an interrupt will also be generated (INT output asserted). Failure modes are (1) Breakdown in alternating SOP-EOP sequence (2) Invalid byte(s) between SOP and EOP not part of a completely invalid word. (3) Reception of frames less than 14 bytes.

RX_OVRI

The RX_OVRI bit is set when a receive overrun is caused by the FIFO_FULL[15:0] inputs being asserted and subsequently causing an overrun of the RXXG internal FIFO. If RX_OVRE is '1' in RXXG Configuration 3 register, an interrupt will also be generated (INT output asserted).

ADR_FILTERI

The ADR_FILTERI bit is set whenever a packet is filtered by RXXG, due to MAC Address filter functions. If ADR_FILTERE is '1' in RXXG Configuration 3 register, an interrupt will also be generated (INT output asserted).

ERR_DETECTI

The ERR_DETECTI bit is set whenever a packet is filtered by RXXG, due protocol error conditions. If ERR_DETECTI is '1' in RXXG Configuration 3 register, an interrupt will also be generated (INT output asserted).

PRMB_ERRI

PRMB_ERRI is set when a packet is received that violates the preamble expected. The expected preamble is dependent on PUREP and LONGP in RXXG Configuration Register 1. If PRMB_ERRE is '1' in RXXG Configuration 3 register, an interrupt will also be generated (INT output asserted).

Register 0x2044H:RXXG Status

Bit	Type	Function	Default
Bit 15	R	Reserved	0
Bit 14	R	Unused	X:high
Bit 13	R	Unused	X:high
Bit 12	R	Unused	X:high
Bit 11	R	Unused	X:high
Bit 10	R	Unused	X:high
Bit 9	R	Unused	X:high
Bit 8	R	Unused	X:high
Bit 7	R	Unused	X:high
Bit 6	R	Unused	X:high
Bit 5	R	Unused	X:high
Bit 4	R	Unused	X:high
Bit 3	R	Unused	X:high
Bit 2	R	Unused	X:high
Bit 1	R	Unused	X:high
Bit 0	R	Unused	X:high

Register 0x2045H:RXXG Maximum Frame Length

Bit	Type	Function	Default
Bit 15	R/W	RX_MAXFR[15]	0
Bit 14	R/W	RX_MAXFR[14]	0
Bit 13	R/W	RX_MAXFR[13]	0
Bit 12	R/W	RX_MAXFR[12]	0
Bit 11	R/W	RX_MAXFR[11]	0
Bit 10	R/W	RX_MAXFR[10]	1
Bit 9	R/W	RX_MAXFR[9]	0
Bit 8	R/W	RX_MAXFR[8]	1
Bit 7	R/W	RX_MAXFR[7]	1
Bit 6	R/W	RX_MAXFR[6]	1
Bit 5	R/W	RX_MAXFR[5]	1
Bit 4	R/W	RX_MAXFR[4]	0
Bit 3	R/W	RX_MAXFR[3]	1
Bit 2	R/W	RX_MAXFR[2]	1
Bit 1	R/W	RX_MAXFR[1]	1
Bit 0	R/W	RX_MAXFR[0]	0

RX_MAXFR[15:0]

This field resets to 0x05EE, which represents a maximum receive frame of 1518 octets. An untagged maximum size Ethernet frame is 1518 octets in length. A tagged frame adds four octets for a total of 1522 octets. If a different maximum length restriction is desired, program this 16-bit field. Frames which exceed this length will be truncated to match the specified length. Note: This does not include the length of preamble/ SFD bytes.

Register 0x2046H:RXXG SA[15:0] – Station Address

Bit	Type	Function	Default
Bit 15	R/W	SA[15]	0
Bit 14	R/W	SA [14]	0
Bit 13	R/W	SA [13]	0
Bit 12	R/W	SA [12]	0
Bit 11	R/W	SA [11]	0
Bit 10	R/W	SA [10]	0
Bit 9	R/W	SA [9]	0
Bit 8	R/W	SA [8]	0
Bit 7	R/W	SA [7]	0
Bit 6	R/W	SA [6]	0
Bit 5	R/W	SA [5]	0
Bit 4	R/W	SA [4]	0
Bit 3	R/W	SA [3]	0
Bit 2	R/W	SA [2]	0
Bit 1	R/W	SA[1]	0
Bit 0	R/W	SA[0]	0

SA[15:0]

Station Address Low Word. MAC Control PAUSE frames may be addressed to the well-known Multicast address assigned for this purpose, or to the Station Address directly i.e. SA[47 :0]. The MAC Control Sublayer expects either the Multicast address or the Station address as criteria to respond to the PAUSE frame received.

Register 0x2047H:RXXG SA[31:16] – Station Address

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

SA[31:16]

Station Address Mid Word. MAC Control PAUSE frames may be addressed to the well-known Multicast address assigned for this purpose, or to the Station Address directly i.e. SA[47 :0] . The MAC Control Sublayer expects either the Multicast address or the Station address as criteria to respond to the PAUSE frame received.

Register 0x2048H:RXXG SA[47:32] – Station Address

Bit	Type	Function	Default
Bit 15	R/W	SA[47]	0
Bit 14	R/W	SA [46]	0
Bit 13	R/W	SA [45]	0
Bit 12	R/W	SA [44]	0
Bit 11	R/W	SA [43]	0
Bit 10	R/W	SA [42]	0
Bit 9	R/W	SA [41]	0
Bit 8	R/W	SA [40]	0
Bit 7	R/W	SA [39]	0
Bit 6	R/W	SA [38]	0
Bit 5	R/W	SA [37]	0
Bit 4	R/W	SA [36]	0
Bit 3	R/W	SA [35]	0
Bit 2	R/W	SA [34]	0
Bit 1	R/W	SA[33]	0
Bit 0	R/W	SA[32]	0

SA[47:32]

Station Address High Word. MAC Control PAUSE frames may be addressed to the well-known Multicast address assigned for this purpose, or to the Station Address directly i.e. SA[47 :0] . The MAC Control Sublayer expects either the Multicast address or the Station address as criteria to respond to the PAUSE frame received.

Register 0x2049H:RXXG Receive FIFO Threshold

Bit	Type	Function	Default
Bit 15	R	Unused	X:high
Bit 14	R	Unused	X:high
Bit 13	R	Unused	X:high
Bit 12	R	Unused	X:high
Bit 11	R	Unused	X:high
Bit 10	R	Unused	X:high
Bit 9	R	Unused	X:high
Bit 8	R	Unused	X:high
Bit 7	R	Unused	X:high
Bit 6	R	Unused	X:high
Bit 5	R	Unused	X:high
Bit 4	R	Unused	X:high
Bit 3	R	Unused	X:high
Bit 2	R/W	CUT_THRU_THRES_SEL[2]	1
Bit 1	R/W	CUT_THRU_THRES_SEL[1]	0
Bit 0	R/W	CUT_THRU_THRES_SEL[0]	1

CUT_THRU_THRES_SEL[2:0]:

Cut_thru_thres_sel	Pkt_Fifo_Threshold
0	8 (64 bytes)
1	64 (512 bytes)
2	128 (1024 bytes)
3	192 (1536 bytes)
4	224 (1792 bytes)
5	EOP write triggers reading from PKT fifo.

Cut-Thru threshold select, sets the forwarding threshold in the internal receive FIFO for initiating packet transfer to the System FIFO interface. Default is 0x5 or 255 x 64-bit entries or 2040 bytes (+/-7 bytes). This defines a store-and-forward mode of operation. This mode of operation can be used only when all packets received are expected to be less than or equal to 2000. Packets greater than 2000 will be truncated (to prevent FIFO filling) and marked as errored. To configure RXXG for cut-through mode, program cut_thru_thres_sel to 0,1, 2, 3, or 4. Increasing the threshold value, increases the errored packet size RXXG can filter. This mode of operation can be used where packets greater than the FIFO size (~2000 bytes) are expected, i.e. jumbo packets, but errored packets less than the threshold value are to be filtered. If a packet breaks the cut-through threshold, RXXG begins to read the packet out. As RXXG can only determine if a packet is errored after the EOP is received, it is not possible to filter this packet. Since each word in the RAM can occupy 8 bytes of data, packets that are under the minimum size to fill the threshold can only be filtered by RXXG. For example,

CUT_THRU_SEL = 4
=> 224 Words
=> 1 byte on first

+ 222 x 8 bytes

+ 2 x 8 bytes (Latency)

- 8 bytes (Possible Line Gap inside a packet, i.e. dead-band

1786 byte packet

Hence, the absolute maximum size packet RXXG can guarantee to filter, when CUT_THRU_SEL = 4, is a 1785 byte packet. Depending on the offset of the SOP in the RAM, i.e. is the SOP in the MSByte of the RAM, the RXXG will filter or forward packets between 1786 bytes and 1808 (224x8) bytes. Below is a Table illustrating the acceptable packet sizes for filtering based on the CUT_THRU_SEL value.

Table 14 CUT_THRU_THRES Packet Sizes

CUT_THRU_SEL	0	1	2	3	4	5
Words to Threshold	8	64	128	192	224	N/A
Guaranteed maximum size packet RXXG will filter	57	505	1017	1529	1785	1992
Filter or forward packets in this range	58 - 80	–506 - 528	–1018 - 1040	–1530 - 1552	–1786 - 1808	N/A
Guaranteed minimum size packet RXXG will not filter	>80	>528	>1041	>1552	>1808	N/A

A cut_thru_thres_sel value of 5 will mean that the Pkt FIFO will never go into cut_thru mode, i.e. no support for jumbo packets in RXXG. When the packet size goes over 2040 bytes the Pkt FIFO will overrun.

Register 0x204AH:RXXG Exact Match Address 0 Low Word

Bit	Type	Function	Default
Bit 15	R/W	ADR_MATCH0[15]	0
Bit 14	R/W	ADR_MATCH0 [14]	0
Bit 13	R/W	ADR_MATCH0 [13]	0
Bit 12	R/W	ADR_MATCH0 [12]	0
Bit 11	R/W	ADR_MATCH0 [11]	0
Bit 10	R/W	ADR_MATCH0 [10]	0
Bit 9	R/W	ADR_MATCH0 [9]	0
Bit 8	R/W	ADR_MATCH0 [8]	0
Bit 7	R/W	ADR_MATCH0 [7]	0
Bit 6	R/W	ADR_MATCH0 [6]	0
Bit 5	R/W	ADR_MATCH0 [5]	0
Bit 4	R/W	ADR_MATCH0 [4]	0
Bit 3	R/W	ADR_MATCH0 [3]	0
Bit 2	R/W	ADR_MATCH0 [2]	0
Bit 1	R/W	ADR_MATCH0 [1]	0
Bit 0	R/W	ADR_MATCH0 [0]	0

ADR_MATCH0[15:0]

The Exact Match Address 0 Low Word is used for the Low Word [15:0] of the 48-bit MAC Address that the Address Filter Logic uses to compare on. This register is one of eight separate Exact Match Address registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x204BH:RXXG Exact Match Address 0 Mid Word

Bit	Type	Function	Default
Bit 15	R/W	ADR_MATCH0[31]	0
Bit 14	R/W	ADR_MATCH0 [30]	0
Bit 13	R/W	ADR_MATCH0 [29]	0
Bit 12	R/W	ADR_MATCH0 [28]	0
Bit 11	R/W	ADR_MATCH0 [27]	0
Bit 10	R/W	ADR_MATCH0 [26]	0
Bit 9	R/W	ADR_MATCH0 [25	0
Bit 8	R/W	ADR_MATCH0 [24]	0
Bit 7	R/W	ADR_MATCH0 [23]	0
Bit 6	R/W	ADR_MATCH0 [22]	0
Bit 5	R/W	ADR_MATCH0 [21]	0
Bit 4	R/W	ADR_MATCH0 [20]	0
Bit 3	R/W	ADR_MATCH0 [19]	0
Bit 2	R/W	ADR_MATCH0 [18]	0
Bit 1	R/W	ADR_MATCH0 [17]	0
Bit 0	R/W	ADR_MATCH0 [16]	0

ADR_MATCH0[31:16]

The Exact Match Address 0 Mid Word is used for the Mid Word [31:16] of the 48-bit MAC Address that the Address Filter Logic uses to compare on. This register is one of eight separate Exact Match Address registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x204CH:RXXG Exact Match Address 0 High Word

Bit	Type	Function	Default
Bit 15	R/W	ADR_MATCH0[47]	0
Bit 14	R/W	ADR_MATCH0 [46]	0
Bit 13	R/W	ADR_MATCH0 [45]	0
Bit 12	R/W	ADR_MATCH0 [44]	0
Bit 11	R/W	ADR_MATCH0 [43]	0
Bit 10	R/W	ADR_MATCH0 [42]	0
Bit 9	R/W	ADR_MATCH0 [41]	0
Bit 8	R/W	ADR_MATCH0 [40]	0
Bit 7	R/W	ADR_MATCH0 [39]	0
Bit 6	R/W	ADR_MATCH0 [38]	0
Bit 5	R/W	ADR_MATCH0 [37]	0
Bit 4	R/W	ADR_MATCH0 [36]	0
Bit 3	R/W	ADR_MATCH0 [35]	0
Bit 2	R/W	ADR_MATCH0 [34]	0
Bit 1	R/W	ADR_MATCH0 [33]	0
Bit 0	R/W	ADR_MATCH0 [32]	0

ADR_MATCH0[47:32]

The Exact Match Address 0 High Word is used for the High Word [47:32] of the 48-bit MAC Address that the Address Filter Logic uses to compare on. This register is one of eight separate Exact Match Address registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x204DH:RXXG Exact Match Address 1 Low Word

Bit	Type	Function	Default
Bit 15	R/W	ADR_MATCH1[15]	0
Bit 14	R/W	ADR_MATCH1 [14]	0
Bit 13	R/W	ADR_MATCH1 [13]	0
Bit 12	R/W	ADR_MATCH1 [12]	0
Bit 11	R/W	ADR_MATCH1 [11]	0
Bit 10	R/W	ADR_MATCH1 [10]	0
Bit 9	R/W	ADR_MATCH1 [9]	0
Bit 8	R/W	ADR_MATCH1 [8]	0
Bit 7	R/W	ADR_MATCH1 [7]	0
Bit 6	R/W	ADR_MATCH1 [6]	0
Bit 5	R/W	ADR_MATCH1 [5]	0
Bit 4	R/W	ADR_MATCH1 [4]	0
Bit 3	R/W	ADR_MATCH1 [3]	0
Bit 2	R/W	ADR_MATCH1 [2]	0
Bit 1	R/W	ADR_MATCH1 [1]	0
Bit 0	R/W	ADR_MATCH1 [0]	0

ADR_MATCH1[15:0]

The Exact Match Address 1 Low Word is used for the Low Word [15:0] of the 48-bit MAC Address that the Address Filter Logic uses to compare on. This register is one of eight separate Exact Match Address registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x204EH: RXXG Exact Match Address 1 Mid Word

Bit	Type	Function	Default
Bit 15	R/W	ADR_MATCH1[31]	0
Bit 14	R/W	ADR_MATCH1 [30]	0
Bit 13	R/W	ADR_MATCH1 [29]	0
Bit 12	R/W	ADR_MATCH1 [28]	0
Bit 11	R/W	ADR_MATCH1 [27]	0
Bit 10	R/W	ADR_MATCH1 [26]	0
Bit 9	R/W	ADR_MATCH1 [25]	0
Bit 8	R/W	ADR_MATCH1 [24]	0
Bit 7	R/W	ADR_MATCH1 [23]	0
Bit 6	R/W	ADR_MATCH1 [22]	0
Bit 5	R/W	ADR_MATCH1 [21]	0
Bit 4	R/W	ADR_MATCH1 [20]	0
Bit 3	R/W	ADR_MATCH1 [19]	0
Bit 2	R/W	ADR_MATCH1 [18]	0
Bit 1	R/W	ADR_MATCH1 [17]	0
Bit 0	R/W	ADR_MATCH1 [16]	0

ADR_MATCH1[31:16]

The Exact Match Address 1 Mid Word is used for the Mid Word [31:16] of the 48-bit MAC Address that the Address Filter Logic uses to compare on. This register is one of eight separate Exact Match Address registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x204FH:RXXG Exact Match Address 1 High Word

Bit	Type	Function	Default
Bit 15	R/W	ADR_MATCH1[47]	0
Bit 14	R/W	ADR_MATCH1 [46]	0
Bit 13	R/W	ADR_MATCH1 [45]	0
Bit 12	R/W	ADR_MATCH1 [44]	0
Bit 11	R/W	ADR_MATCH1 [43]	0
Bit 10	R/W	ADR_MATCH1 [42]	0
Bit 9	R/W	ADR_MATCH1 [41]	0
Bit 8	R/W	ADR_MATCH1 [40]	0
Bit 7	R/W	ADR_MATCH1 [39]	0
Bit 6	R/W	ADR_MATCH1 [38]	0
Bit 5	R/W	ADR_MATCH1 [37]	0
Bit 4	R/W	ADR_MATCH1 [36]	0
Bit 3	R/W	ADR_MATCH1 [35]	0
Bit 2	R/W	ADR_MATCH1 [34]	0
Bit 1	R/W	ADR_MATCH1 [33]	0
Bit 0	R/W	ADR_MATCH1 [32]	0

ADR_MATCH1[47:32]

The Exact Match Address 1 High Word is used for the High Word [47:32] of the 48-bit MAC Address that the Address Filter Logic uses to compare on. This register is one of eight separate Exact Match Address registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x2050H:RXXG Exact Match Address 2 Low Word

Bit	Type	Function	Default
Bit 15	R/W	ADR_MATCH2[15]	0
Bit 14	R/W	ADR_MATCH2 [14]	0
Bit 13	R/W	ADR_MATCH2 [13]	0
Bit 12	R/W	ADR_MATCH2 [12]	0
Bit 11	R/W	ADR_MATCH2 [11]	0
Bit 10	R/W	ADR_MATCH2 [10]	0
Bit 9	R/W	ADR_MATCH2 [9]	0
Bit 8	R/W	ADR_MATCH2 [8]	0
Bit 7	R/W	ADR_MATCH2 [7]	0
Bit 6	R/W	ADR_MATCH2 [6]	0
Bit 5	R/W	ADR_MATCH2 [5]	0
Bit 4	R/W	ADR_MATCH2 [4]	0
Bit 3	R/W	ADR_MATCH2 [3]	0
Bit 2	R/W	ADR_MATCH2 [2]	0
Bit 1	R/W	ADR_MATCH2 [1]	0
Bit 0	R/W	ADR_MATCH2 [0]	0

ADR_MATCH2[15:0]

The Exact Match Address 2 Low Word is used for the Low Word [15:0] of the 48-bit MAC Address that the Address Filter Logic uses to compare on. This register is one of eight separate Exact Match Address registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x2051H:RXXG Exact Match Address 2 Mid Word

Bit	Type	Function	Default
Bit 15	R/W	ADR_MATCH2[31]	0
Bit 14	R/W	ADR_MATCH2 [30]	0
Bit 13	R/W	ADR_MATCH2 [29]	0
Bit 12	R/W	ADR_MATCH2 [28]	0
Bit 11	R/W	ADR_MATCH2 [27]	0
Bit 10	R/W	ADR_MATCH2 [26]	0
Bit 9	R/W	ADR_MATCH2 [25]	0
Bit 8	R/W	ADR_MATCH2 [24]	0
Bit 7	R/W	ADR_MATCH2 [23]	0
Bit 6	R/W	ADR_MATCH2 [22]	0
Bit 5	R/W	ADR_MATCH2 [21]	0
Bit 4	R/W	ADR_MATCH2 [20]	0
Bit 3	R/W	ADR_MATCH2 [19]	0
Bit 2	R/W	ADR_MATCH2 [18]	0
Bit 1	R/W	ADR_MATCH2 [17]	0
Bit 0	R/W	ADR_MATCH2 [16]	0

ADR_MATCH2[31:16]

The Exact Match Address 2 Mid Word is used for the Mid Word [31:16] of the 48-bit MAC Address that the Address Filter Logic uses to compare on. This register is one of eight separate Exact Match Address registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x2052H:RXXG Exact Match Address 2 High Word

Bit	Type	Function	Default
Bit 15	R/W	ADR_MATCH2[47]	0
Bit 14	R/W	ADR_MATCH2 [46]	0
Bit 13	R/W	ADR_MATCH2 [45]	0
Bit 12	R/W	ADR_MATCH2 [44]	0
Bit 11	R/W	ADR_MATCH2 [43]	0
Bit 10	R/W	ADR_MATCH2 [42]	0
Bit 9	R/W	ADR_MATCH2 [41]	0
Bit 8	R/W	ADR_MATCH2 [40]	0
Bit 7	R/W	ADR_MATCH2[39]	0
Bit 6	R/W	ADR_MATCH2 [38]	0
Bit 5	R/W	ADR_MATCH2 [37]	0
Bit 4	R/W	ADR_MATCH2 [36]	0
Bit 3	R/W	ADR_MATCH2 [35]	0
Bit 2	R/W	ADR_MATCH2 [34]	0
Bit 1	R/W	ADR_MATCH2 [33]	0
Bit 0	R/W	ADR_MATCH2 [32]	0

ADR_MATCH2[47:32]

The Exact Match Address 2 High Word is used for the High Word [47:32] of the 48-bit MAC Address that the Address Filter Logic uses to compare on. This register is one of eight separate Exact Match Address registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x2053H:RXXG Exact Match Address 3 Low Word

Bit	Type	Function	Default
Bit 15	R/W	ADR_MATCH3[15]	0
Bit 14	R/W	ADR_MATCH3 [14]	0
Bit 13	R/W	ADR_MATCH3 [13]	0
Bit 12	R/W	ADR_MATCH3 [12]	0
Bit 11	R/W	ADR_MATCH3 [11]	0
Bit 10	R/W	ADR_MATCH3 [10]	0
Bit 9	R/W	ADR_MATCH3 [9]	0
Bit 8	R/W	ADR_MATCH3 [8]	0
Bit 7	R/W	ADR_MATCH3 [7]	0
Bit 6	R/W	ADR_MATCH3 [6]	0
Bit 5	R/W	ADR_MATCH3 [5]	0
Bit 4	R/W	ADR_MATCH3 [4]	0
Bit 3	R/W	ADR_MATCH3 [3]	0
Bit 2	R/W	ADR_MATCH3 [2]	0
Bit 1	R/W	ADR_MATCH3 [1]	0
Bit 0	R/W	ADR_MATCH3 [0]	0

ADR_MATCH3[15:0]

The Exact Match Address 3 Low Word is used for the Low Word [15:0] of the 48-bit MAC Address that the Address Filter Logic uses to compare on. This register is one of eight separate Exact Match Address registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x2054H:RXXG Exact Match Address 3 Mid Word

Bit	Type	Function	Default
Bit 15	R/W	ADR_MATCH3[31]	0
Bit 14	R/W	ADR_MATCH3 [30]	0
Bit 13	R/W	ADR_MATCH3 [29]	0
Bit 12	R/W	ADR_MATCH3 [28]	0
Bit 11	R/W	ADR_MATCH3 [27]	0
Bit 10	R/W	ADR_MATCH3 [26]	0
Bit 9	R/W	ADR_MATCH3 [25]	0
Bit 8	R/W	ADR_MATCH3 [24]	0
Bit 7	R/W	ADR_MATCH3 [23]	0
Bit 6	R/W	ADR_MATCH3 [22]	0
Bit 5	R/W	ADR_MATCH3 [21]	0
Bit 4	R/W	ADR_MATCH3 [20]	0
Bit 3	R/W	ADR_MATCH3 [19]	0
Bit 2	R/W	ADR_MATCH3 [18]	0
Bit 1	R/W	ADR_MATCH3 [17]	0
Bit 0	R/W	ADR_MATCH3 [16]	0

ADR_MATCH3[31:16]

The Exact Match Address 3 Mid Word is used for the Mid Word [31:16] of the 48-bit MAC Address that the Address Filter Logic uses to compare on. This register is one of eight separate Exact Match Address registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x2055H:RXXG Exact Match Address 3 High Word

Bit	Type	Function	Default
Bit 15	R/W	ADR_MATCH3[47]	0
Bit 14	R/W	ADR_MATCH3 [46]	0
Bit 13	R/W	ADR_MATCH3 [45]	0
Bit 12	R/W	ADR_MATCH3 [44]	0
Bit 11	R/W	ADR_MATCH3 [43]	0
Bit 10	R/W	ADR_MATCH3 [42]	0
Bit 9	R/W	ADR_MATCH3 [41]	0
Bit 8	R/W	ADR_MATCH3 [40]	0
Bit 7	R/W	ADR_MATCH3 [39]	0
Bit 6	R/W	ADR_MATCH3 [38]	0
Bit 5	R/W	ADR_MATCH3 [37]	0
Bit 4	R/W	ADR_MATCH3 [36]	0
Bit 3	R/W	ADR_MATCH3 [35]	0
Bit 2	R/W	ADR_MATCH3 [34]	0
Bit 1	R/W	ADR_MATCH3 [33]	0
Bit 0	R/W	ADR_MATCH3 [32]	0

ADR_MATCH3[47:32]

The Exact Match Address 3 High Word is used for the High Word [47:32] of the 48-bit MAC Address that the Address Filter Logic uses to compare on. This register is one of eight separate Exact Match Address registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x2056H:RXXG Exact Match Address 4 Low Word

Bit	Type	Function	Default
Bit 15	R/W	ADR_MATCH4[15]	0
Bit 14	R/W	ADR_MATCH4 [14]	0
Bit 13	R/W	ADR_MATCH4 [13]	0
Bit 12	R/W	ADR_MATCH4 [12]	0
Bit 11	R/W	ADR_MATCH4 [11]	0
Bit 10	R/W	ADR_MATCH4 [10]	0
Bit 9	R/W	ADR_MATCH4 [9]	0
Bit 8	R/W	ADR_MATCH4 [8]	0
Bit 7	R/W	ADR_MATCH4 [7]	0
Bit 6	R/W	ADR_MATCH4 [6]	0
Bit 5	R/W	ADR_MATCH4 [5]	0
Bit 4	R/W	ADR_MATCH4 [4]	0
Bit 3	R/W	ADR_MATCH4 [3]	0
Bit 2	R/W	ADR_MATCH4 [2]	0
Bit 1	R/W	ADR_MATCH4 [1]	0
Bit 0	R/W	ADR_MATCH4 [0]	0

ADR_MATCH4[15:0]

The Exact Match Address 4 Low Word is used for the Low Word [15:0] of the 48-bit MAC Address that the Address Filter Logic uses to compare on. This register is one of eight separate Exact Match Address registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x2057H:RXXG Exact Match Address 4 Mid Word

Bit	Type	Function	Default
Bit 15	R/W	ADR_MATCH4[31]	0
Bit 14	R/W	ADR_MATCH4 [30]	0
Bit 13	R/W	ADR_MATCH4 [29]	0
Bit 12	R/W	ADR_MATCH4 [28]	0
Bit 11	R/W	ADR_MATCH4 [27]	0
Bit 10	R/W	ADR_MATCH4 [26]	0
Bit 9	R/W	ADR_MATCH4 [25]	0
Bit 8	R/W	ADR_MATCH4 [24]	0
Bit 7	R/W	ADR_MATCH4 [23]	0
Bit 6	R/W	ADR_MATCH4 [22]	0
Bit 5	R/W	ADR_MATCH4 [21]	0
Bit 4	R/W	ADR_MATCH4 [20]	0
Bit 3	R/W	ADR_MATCH4 [19]	0
Bit 2	R/W	ADR_MATCH4 [18]	0
Bit 1	R/W	ADR_MATCH4 [17]	0
Bit 0	R/W	ADR_MATCH4 [16]	0

ADR_MATCH4[31:16]

The Exact Match Address 4 Mid Word is used for the Mid Word [31:16] of the 48-bit MAC Address that the Address Filter Logic uses to compare on. This register is one of eight separate Exact Match Address registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x2058H:RXXG Exact Match Address 4 High Word

Bit	Type	Function	Default
Bit 15	R/W	ADR_MATCH4[47]	0
Bit 14	R/W	ADR_MATCH4 [46]	0
Bit 13	R/W	ADR_MATCH4 [45]	0
Bit 12	R/W	ADR_MATCH4 [44]	0
Bit 11	R/W	ADR_MATCH4 [43]	0
Bit 10	R/W	ADR_MATCH4 [42]	0
Bit 9	R/W	ADR_MATCH4 [41]	0
Bit 8	R/W	ADR_MATCH4 [40]	0
Bit 7	R/W	ADR_MATCH4 [39]	0
Bit 6	R/W	ADR_MATCH4 [38]	0
Bit 5	R/W	ADR_MATCH4 [37]	0
Bit 4	R/W	ADR_MATCH4 [36]	0
Bit 3	R/W	ADR_MATCH4 [35]	0
Bit 2	R/W	ADR_MATCH4 [34]	0
Bit 1	R/W	ADR_MATCH4 [33]	0
Bit 0	R/W	ADR_MATCH4 [32]	0

ADR_MATCH4[47:32]

The Exact Match Address 4 High Word is used for the High Word [47:32] of the 48-bit MAC Address that the Address Filter Logic uses to compare on. This register is one of eight separate Exact Match Address registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x2059H:RXXG Exact Match Address 5 Low Word

Bit	Type	Function	Default
Bit 15	R/W	ADR_MATCH5[15]	0
Bit 14	R/W	ADR_MATCH5 [14]	0
Bit 13	R/W	ADR_MATCH5 [13]	0
Bit 12	R/W	ADR_MATCH5 [12]	0
Bit 11	R/W	ADR_MATCH5 [11]	0
Bit 10	R/W	ADR_MATCH5 [10]	0
Bit 9	R/W	ADR_MATCH5 [9]	0
Bit 8	R/W	ADR_MATCH5 [8]	0
Bit 7	R/W	ADR_MATCH5 [7]	0
Bit 6	R/W	ADR_MATCH5 [6]	0
Bit 5	R/W	ADR_MATCH5 [5]	0
Bit 4	R/W	ADR_MATCH5 [4]	0
Bit 3	R/W	ADR_MATCH5 [3]	0
Bit 2	R/W	ADR_MATCH5 [2]	0
Bit 1	R/W	ADR_MATCH5 [1]	0
Bit 0	R/W	ADR_MATCH5 [0]	0

ADR_MATCH5[15:0]

The Exact Match Address 5 Low Word is used for the Low Word [15:0] of the 48-bit MAC Address that the Address Filter Logic uses to compare on. This register is one of eight separate Exact Match Address registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x205AH:RXXG Exact Match Address 5 Mid Word

Bit	Type	Function	Default
Bit 15	R/W	ADR_MATCH5[31]	0
Bit 14	R/W	ADR_MATCH5 [30]	0
Bit 13	R/W	ADR_MATCH5 [29]	0
Bit 12	R/W	ADR_MATCH5 [28]	0
Bit 11	R/W	ADR_MATCH5 [27]	0
Bit 10	R/W	ADR_MATCH5 [26]	0
Bit 9	R/W	ADR_MATCH5 [25]	0
Bit 8	R/W	ADR_MATCH5 [24]	0
Bit 7	R/W	ADR_MATCH5 [23]	0
Bit 6	R/W	ADR_MATCH5 [22]	0
Bit 5	R/W	ADR_MATCH5 [21]	0
Bit 4	R/W	ADR_MATCH5 [20]	0
Bit 3	R/W	ADR_MATCH5 [19]	0
Bit 2	R/W	ADR_MATCH5 [18]	0
Bit 1	R/W	ADR_MATCH5 [17]	0
Bit 0	R/W	ADR_MATCH5 [16]	0

ADR_MATCH5[31:16]

The Exact Match Address 5 Mid Word is used for the Mid Word [31:16] of the 48-bit MAC Address that the Address Filter Logic uses to compare on. This register is one of eight separate Exact Match Address registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x205BH:RXXG Exact Match Address 5 High Word

Bit	Type	Function	Default
Bit 15	R/W	ADR_MATCH5[47]	0
Bit 14	R/W	ADR_MATCH5 [46]	0
Bit 13	R/W	ADR_MATCH5 [45]	0
Bit 12	R/W	ADR_MATCH5 [44]	0
Bit 11	R/W	ADR_MATCH5 [43]	0
Bit 10	R/W	ADR_MATCH5 [42]	0
Bit 9	R/W	ADR_MATCH5 [41]	0
Bit 8	R/W	ADR_MATCH5 [40]	0
Bit 7	R/W	ADR_MATCH5 [39]	0
Bit 6	R/W	ADR_MATCH5 [38]	0
Bit 5	R/W	ADR_MATCH5 [37]	0
Bit 4	R/W	ADR_MATCH5 [36]	0
Bit 3	R/W	ADR_MATCH5 [35]	0
Bit 2	R/W	ADR_MATCH5 [34]	0
Bit 1	R/W	ADR_MATCH5 [33]	0
Bit 0	R/W	ADR_MATCH5 [32]	0

ADR_MATCH5[47:32]

The Exact Match Address 5 High Word is used for the High Word [47:32] of the 48-bit MAC Address that the Address Filter Logic uses to compare on. This register is one of eight separate Exact Match Address registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x205CH:RXXG Exact Match Address 6 Low Word

Bit	Type	Function	Default
Bit 15	R/W	ADR_MATCH6[15]	0
Bit 14	R/W	ADR_MATCH6 [14]	0
Bit 13	R/W	ADR_MATCH6 [13]	0
Bit 12	R/W	ADR_MATCH6 [12]	0
Bit 11	R/W	ADR_MATCH6 [11]	0
Bit 10	R/W	ADR_MATCH6 [10]	0
Bit 9	R/W	ADR_MATCH6 [9]	0
Bit 8	R/W	ADR_MATCH6 [8]	0
Bit 7	R/W	ADR_MATCH6 [7]	0
Bit 6	R/W	ADR_MATCH6 [6]	0
Bit 5	R/W	ADR_MATCH6 [5]	0
Bit 4	R/W	ADR_MATCH6 [4]	0
Bit 3	R/W	ADR_MATCH6 [3]	0
Bit 2	R/W	ADR_MATCH6 [2]	0
Bit 1	R/W	ADR_MATCH6 [1]	0
Bit 0	R/W	ADR_MATCH6 [0]	0

ADR_MATCH6[15:0]

The Exact Match Address 6 Low Word is used for the Low Word [15:0] of the 48-bit MAC Address that the Address Filter Logic uses to compare on. This register is one of eight separate Exact Match Address registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x205DH:RXXG Exact Match Address 6 Mid Word

Bit	Type	Function	Default
Bit 15	R/W	ADR_MATCH6[31]	0
Bit 14	R/W	ADR_MATCH6 [30]	0
Bit 13	R/W	ADR_MATCH6 [29]	0
Bit 12	R/W	ADR_MATCH6 [28]	0
Bit 11	R/W	ADR_MATCH6 [27]	0
Bit 10	R/W	ADR_MATCH6 [26]	0
Bit 9	R/W	ADR_MATCH6 [25]	0
Bit 8	R/W	ADR_MATCH6 [24]	0
Bit 7	R/W	ADR_MATCH6 [23]	0
Bit 6	R/W	ADR_MATCH6 [22]	0
Bit 5	R/W	ADR_MATCH6 [21]	0
Bit 4	R/W	ADR_MATCH6 [20]	0
Bit 3	R/W	ADR_MATCH6 [19]	0
Bit 2	R/W	ADR_MATCH6 [18]	0
Bit 1	R/W	ADR_MATCH6 [17]	0
Bit 0	R/W	ADR_MATCH6 [16]	0

ADR_MATCH6[31:16]

The Exact Match Address 6 Mid Word is used for the Mid Word [31:16] of the 48-bit MAC Address that the Address Filter Logic uses to compare on. This register is one of eight separate Exact Match Address registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x205EH:RXXG Exact Match Address 6 High Word

Bit	Type	Function	Default
Bit 15	R/W	ADR_MATCH6[47]	0
Bit 14	R/W	ADR_MATCH6 [46]	0
Bit 13	R/W	ADR_MATCH6 [45]	0
Bit 12	R/W	ADR_MATCH6 [44]	0
Bit 11	R/W	ADR_MATCH6 [43]	0
Bit 10	R/W	ADR_MATCH6 [42]	0
Bit 9	R/W	ADR_MATCH6 [41]	0
Bit 8	R/W	ADR_MATCH6 [40]	0
Bit 7	R/W	ADR_MATCH6 [39]	0
Bit 6	R/W	ADR_MATCH6 [38]	0
Bit 5	R/W	ADR_MATCH6 [37]	0
Bit 4	R/W	ADR_MATCH6 [36]	0
Bit 3	R/W	ADR_MATCH6 [35]	0
Bit 2	R/W	ADR_MATCH6 [34]	0
Bit 1	R/W	ADR_MATCH6 [33]	0
Bit 0	R/W	ADR_MATCH6 [32]	0

ADR_MATCH6[47:32]

The Exact Match Address 6 High Word is used for the High Word [47:32] of the 48-bit MAC Address that the Address Filter Logic uses to compare on. This register is one of eight separate Exact Match Address registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x205FH: RXXG Exact Match Address 7 Low Word

Bit	Type	Function	Default
Bit 15	R/W	ADR_MATCH7[15]	0
Bit 14	R/W	ADR_MATCH7 [14]	0
Bit 13	R/W	ADR_MATCH7 [13]	0
Bit 12	R/W	ADR_MATCH7 [12]	0
Bit 11	R/W	ADR_MATCH7 [11]	0
Bit 10	R/W	ADR_MATCH7 [10]	0
Bit 9	R/W	ADR_MATCH7 [9]	0
Bit 8	R/W	ADR_MATCH7 [8]	0
Bit 7	R/W	ADR_MATCH7 [7]	0
Bit 6	R/W	ADR_MATCH7 [6]	0
Bit 5	R/W	ADR_MATCH7 [5]	0
Bit 4	R/W	ADR_MATCH7 [4]	0
Bit 3	R/W	ADR_MATCH7 [3]	0
Bit 2	R/W	ADR_MATCH7 [2]	0
Bit 1	R/W	ADR_MATCH7 [1]	0
Bit 0	R/W	ADR_MATCH7 [0]	0

ADR_MATCH7[15:0]

The Exact Match Address 7 Low Word is used for the Low Word [15:0] of the 48-bit MAC Address that the Address Filter Logic uses to compare on. This register is one of eight separate Exact Match Address registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x2060H:RXXG Exact Match Address 7 Mid Word

Bit	Type	Function	Default
Bit 15	R/W	ADR_MATCH7[31]	0
Bit 14	R/W	ADR_MATCH7 [30]	0
Bit 13	R/W	ADR_MATCH7 [29]	0
Bit 12	R/W	ADR_MATCH7 [28]	0
Bit 11	R/W	ADR_MATCH7 [27]	0
Bit 10	R/W	ADR_MATCH7 [26]	0
Bit 9	R/W	ADR_MATCH7 [25]	0
Bit 8	R/W	ADR_MATCH7 [24]	0
Bit 7	R/W	ADR_MATCH7 [23]	0
Bit 6	R/W	ADR_MATCH7 [22]	0
Bit 5	R/W	ADR_MATCH7 [21]	0
Bit 4	R/W	ADR_MATCH7 [20]	0
Bit 3	R/W	ADR_MATCH7 [19]	0
Bit 2	R/W	ADR_MATCH7 [18]	0
Bit 1	R/W	ADR_MATCH7 [17]	0
Bit 0	R/W	ADR_MATCH7 [16]	0

ADR_MATCH7[31:16]

The Exact Match Address 7 Mid Word is used for the Mid Word [31:16] of the 48-bit MAC Address that the Address Filter Logic uses to compare on. This register is one of eight separate Exact Match Address registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x2061H:RXXG Exact Match Address 7 High Word

Bit	Type	Function	Default
Bit 15	R/W	ADR_MATCH7[47]	0
Bit 14	R/W	ADR_MATCH7 [46]	0
Bit 13	R/W	ADR_MATCH7 [45]	0
Bit 12	R/W	ADR_MATCH7 [44]	0
Bit 11	R/W	ADR_MATCH7 [43]	0
Bit 10	R/W	ADR_MATCH7 [42]	0
Bit 9	R/W	ADR_MATCH7 [41]	0
Bit 8	R/W	ADR_MATCH7 [40]	0
Bit 7	R/W	ADR_MATCH7 [39]	0
Bit 6	R/W	ADR_MATCH7 [38]	0
Bit 5	R/W	ADR_MATCH7 [37]	0
Bit 4	R/W	ADR_MATCH7 [36]	0
Bit 3	R/W	ADR_MATCH7 [35]	0
Bit 2	R/W	ADR_MATCH7 [34]	0
Bit 1	R/W	ADR_MATCH7 [33]	0
Bit 0	R/W	ADR_MATCH7 [32]	0

ADR_MATCH7[47:32]

The Exact Match Address 7 High Word is used for the High Word [47:32] of the 48-bit MAC Address that the Address Filter Logic uses to compare on. This register is one of eight separate Exact Match Address registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x2062H:RXXG Exact Match VID 0

Bit	Type	Function	Default
Bit 15	R	Unused	X:high
Bit 14	R	Unused	X:high
Bit 13	R	Unused	X:high
Bit 12	R	Unused	X:high
Bit 11	R/W	VID_MATCH0[11]	0
Bit 10	R/W	VID_MATCH0[10]	0
Bit 9	R/W	VID_MATCH0[9]	0
Bit 8	R/W	VID_MATCH0[8]	0
Bit 7	R/W	VID_MATCH0[7]	0
Bit 6	R/W	VID_MATCH0[6]	0
Bit 5	R/W	VID_MATCH0[5]	0
Bit 4	R/W	VID_MATCH0[4]	0
Bit 3	R/W	VID_MATCH0[3]	0
Bit 2	R/W	VID_MATCH0[2]	0
Bit 1	R/W	VID_MATCH0[1]	0
Bit 0	R/W	VID_MATCH0[0]	0

VID_MATCH0[11:0]

The Exact Match VID 0 register is used by the Address Filter Logic to compare on the 12 bit VID field on VLAN tagged frames. This register is one of eight separate Exact Match VID registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x2063H:RXXG Exact Match VID 1

Bit	Type	Function	Default
Bit 15	R	Unused	X:high
Bit 14	R	Unused	X:high
Bit 13	R	Unused	X:high
Bit 12	R	Unused	X:high
Bit 11	R/W	VID_MATCH1[11]	0
Bit 10	R/W	VID_MATCH1[10]	0
Bit 9	R/W	VID_MATCH1[9]	0
Bit 8	R/W	VID_MATCH1[8]	0
Bit 7	R/W	VID_MATCH1[7]	0
Bit 6	R/W	VID_MATCH1[6]	0
Bit 5	R/W	VID_MATCH1[5]	0
Bit 4	R/W	VID_MATCH1[4]	0
Bit 3	R/W	VID_MATCH1[3]	0
Bit 2	R/W	VID_MATCH1[2]	0
Bit 1	R/W	VID_MATCH1[1]	0
Bit 0	R/W	VID_MATCH1[0]	0

VID_MATCH1[11:0]

The Exact Match VID 1 register is used by the Address Filter Logic to compare on the 12 bit VID field on VLAN tagged frames. This register is one of eight separate Exact Match VID registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x2064H:RXXG Exact Match VID 2

Bit	Type	Function	Default
Bit 15	R	Unused	X:high
Bit 14	R	Unused	X:high
Bit 13	R	Unused	X:high
Bit 12	R	Unused	X:high
Bit 11	R/W	VID_MATCH2[11]	0
Bit 10	R/W	VID_MATCH2[10]	0
Bit 9	R/W	VID_MATCH2[9]	0
Bit 8	R/W	VID_MATCH2[8]	0
Bit 7	R/W	VID_MATCH2[7]	0
Bit 6	R/W	VID_MATCH2[6]	0
Bit 5	R/W	VID_MATCH2[5]	0
Bit 4	R/W	VID_MATCH2[4]	0
Bit 3	R/W	VID_MATCH2[3]	0
Bit 2	R/W	VID_MATCH2[2]	0
Bit 1	R/W	VID_MATCH2[1]	0
Bit 0	R/W	VID_MATCH2[0]	0

VID_MATCH2[11:0]

The Exact Match VID 2 register is used by the Address Filter Logic to compare on the 12 bit VID field on VLAN tagged frames. This register is one of eight separate Exact Match VID registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x2065H:RXXG Exact Match VID 3

Bit	Type	Function	Default
Bit 15	R	Unused	X:high
Bit 14	R	Unused	X:high
Bit 13	R	Unused	X:high
Bit 12	R	Unused	X:high
Bit 11	R/W	VID_MATCH3[11]	0
Bit 10	R/W	VID_MATCH3[10]	0
Bit 9	R/W	VID_MATCH3[9]	0
Bit 8	R/W	VID_MATCH3[8]	0
Bit 7	R/W	VID_MATCH3[7]	0
Bit 6	R/W	VID_MATCH3[6]	0
Bit 5	R/W	VID_MATCH3[5]	0
Bit 4	R/W	VID_MATCH3[4]	0
Bit 3	R/W	VID_MATCH3[3]	0
Bit 2	R/W	VID_MATCH3[2]	0
Bit 1	R/W	VID_MATCH3[1]	0
Bit 0	R/W	VID_MATCH3[0]	0

VID_MATCH3[11:0]

The Exact Match VID 3 register is used by the Address Filter Logic to compare on the 12 bit VID field on VLAN tagged frames. This register is one of eight separate Exact Match VID registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x2066H:RXXG Exact Match VID 4

Bit	Type	Function	Default
Bit 15	R	Unused	X:high
Bit 14	R	Unused	X:high
Bit 13	R	Unused	X:high
Bit 12	R	Unused	X:high
Bit 11	R/W	VID_MATCH4[11]	0
Bit 10	R/W	VID_MATCH4[10]	0
Bit 9	R/W	VID_MATCH4[9]	0
Bit 8	R/W	VID_MATCH4[8]	0
Bit 7	R/W	VID_MATCH4[7]	0
Bit 6	R/W	VID_MATCH4[6]	0
Bit 5	R/W	VID_MATCH4[5]	0
Bit 4	R/W	VID_MATCH4[4]	0
Bit 3	R/W	VID_MATCH4[3]	0
Bit 2	R/W	VID_MATCH4[2]	0
Bit 1	R/W	VID_MATCH4[1]	0
Bit 0	R/W	VID_MATCH4[0]	0

VID_MATCH4[11:0]

The Exact Match VID 4 register is used by the Address Filter Logic to compare on the 12 bit VID field on VLAN tagged frames. This register is one of eight separate Exact Match VID registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x2067H:RXXG Exact Match VID 5

Bit	Type	Function	Default
Bit 15	R	Unused	X:high
Bit 14	R	Unused	X:high
Bit 13	R	Unused	X:high
Bit 12	R	Unused	X:high
Bit 11	R/W	VID_MATCH5[11]	0
Bit 10	R/W	VID_MATCH5[10]	0
Bit 9	R/W	VID_MATCH5[9]	0
Bit 8	R/W	VID_MATCH5[8]	0
Bit 7	R/W	VID_MATCH5[7]	0
Bit 6	R/W	VID_MATCH5[6]	0
Bit 5	R/W	VID_MATCH5[5]	0
Bit 4	R/W	VID_MATCH5[4]	0
Bit 3	R/W	VID_MATCH5[3]	0
Bit 2	R/W	VID_MATCH5[2]	0
Bit 1	R/W	VID_MATCH5[1]	0
Bit 0	R/W	VID_MATCH5[0]	0

VID_MATCH5[11:0]

The Exact Match VID 5 register is used by the Address Filter Logic to compare on the 12 bit VID field on VLAN tagged frames. This register is one of eight separate Exact Match VID registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x2068H:RXXG Exact Match VID 6

Bit	Type	Function	Default
Bit 15	R	Unused	X:high
Bit 14	R	Unused	X:high
Bit 13	R	Unused	X:high
Bit 12	R	Unused	X:high
Bit 11	R/W	VID_MATCH6[11]	0
Bit 10	R/W	VID_MATCH6[10]	0
Bit 9	R/W	VID_MATCH6[9]	0
Bit 8	R/W	VID_MATCH6[8]	0
Bit 7	R/W	VID_MATCH6[7]	0
Bit 6	R/W	VID_MATCH6[6]	0
Bit 5	R/W	VID_MATCH6[5]	0
Bit 4	R/W	VID_MATCH6[4]	0
Bit 3	R/W	VID_MATCH6[3]	0
Bit 2	R/W	VID_MATCH6[2]	0
Bit 1	R/W	VID_MATCH6[1]	0
Bit 0	R/W	VID_MATCH6[0]	0

VID_MATCH6[11:0]

The Exact Match VID 6 register is used by the Address Filter Logic to compare on the 12 bit VID field on VLAN tagged frames. This register is one of eight separate Exact Match VID registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x2069H:RXXG Exact Match VID 7

Bit	Type	Function	Default
Bit 15	R	Unused	
Bit 14	R	Unused	X:high
Bit 13	R	Unused	X:high
Bit 12	R	Unused	X:high
Bit 11	R/W	VID_MATCH7[11]	0
Bit 10	R/W	VID_MATCH7[10]	0
Bit 9	R/W	VID_MATCH7[9]	0
Bit 8	R/W	VID_MATCH7[8]	0
Bit 7	R/W	VID_MATCH7[7]	0
Bit 6	R/W	VID_MATCH7[6]	0
Bit 5	R/W	VID_MATCH7[5]	0
Bit 4	R/W	VID_MATCH7[4]	0
Bit 3	R/W	VID_MATCH7[3]	0
Bit 2	R/W	VID_MATCH7[2]	0
Bit 1	R/W	VID_MATCH7[1]	0
Bit 0	R/W	VID_MATCH7[0]	0

VID_MATCH7[11:0]

The Exact Match VID 7 register is used by the Address Filter Logic to compare on the 12 bit VID field on VLAN tagged frames. This register is one of eight separate Exact Match VID registers that the Address Filter Logic can use to compare on.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x206AH: RXXG Multicast HASH Low Word

Bit	Type	Function	Default
Bit 15	R/W	MHASH[15]	0
Bit 14	R/W	MHASH[14]	0
Bit 13	R/W	MHASH[13]	0
Bit 12	R/W	MHASH[12]	0
Bit 11	R/W	MHASH[11]	0
Bit 10	R/W	MHASH[10]	0
Bit 9	R/W	MHASH[9]	0
Bit 8	R/W	MHASH[8]	0
Bit 7	R/W	MHASH[7]	0
Bit 6	R/W	MHASH[6]	0
Bit 5	R/W	MHASH[5]	0
Bit 4	R/W	MHASH[4]	0
Bit 3	R/W	MHASH[3]	0
Bit 2	R/W	MHASH[2]	0
Bit 1	R/W	MHASH[1]	0
Bit 0	R/W	MHASH[0]	0

MHASH[15:0]

The MHASH[15:0] is the Low Word of the 64-bit Multicast Hash bin. This is used by the Address Filter Logic for filtering of Multicast Addressed frames when MHASH_EN is ‘1’ in Address Filter Control 2 register.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x206BH:RXXG Multicast HASH MidLow Word

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

MHASH[31:16]

The MHASH[31:16] is the MidLow Word of the 64-bit Multicast Hash bin. This is used by the Address Filter Logic for filtering of Multicast Addressed frames when MHASH_EN is '1' in Address Filter Control 2 register.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x206CH:RXXG Multicast HASH MidHigh Word

Bit	Type	Function	Default
Bit 15	R/W	MHASH[47]	0
Bit 14	R/W	MHASH[46]	0
Bit 13	R/W	MHASH[45]	0
Bit 12	R/W	MHASH[44]	0
Bit 11	R/W	MHASH[43]	0
Bit 10	R/W	MHASH[42]	0
Bit 9	R/W	MHASH[41]	0
Bit 8	R/W	MHASH[40]	0
Bit 7	R/W	MHASH[39]	0
Bit 6	R/W	MHASH[38]	0
Bit 5	R/W	MHASH[37]	0
Bit 4	R/W	MHASH[36]	0
Bit 3	R/W	MHASH[35]	0
Bit 2	R/W	MHASH[34]	0
Bit 1	R/W	MHASH[33]	0
Bit 0	R/W	MHASH[32]	0

MHASH[47:32]

The MHASH[47:32] is the MidHigh Word of the 64-bit Multicast Hash bin. This is used by the Address Filter Logic for filtering of Multicast Addressed frames when MHASH_EN is '1' in Address Filter Control 2 register.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x206DH:RXXG Multicast HASH High Word

Bit	Type	Function	Default
Bit 15	R/W	MHASH[63]	0
Bit 14	R/W	MHASH[62]	0
Bit 13	R/W	MHASH[61]	0
Bit 12	R/W	MHASH[60]	0
Bit 11	R/W	MHASH[59]	0
Bit 10	R/W	MHASH[58]	0
Bit 9	R/W	MHASH[57]	0
Bit 8	R/W	MHASH[56]	0
Bit 7	R/W	MHASH[55]	0
Bit 6	R/W	MHASH[54]	0
Bit 5	R/W	MHASH[53]	0
Bit 4	R/W	MHASH[52]	0
Bit 3	R/W	MHASH[51]	0
Bit 2	R/W	MHASH[50]	0
Bit 1	R/W	MHASH[49]	0
Bit 0	R/W	MHASH[48]	0

MHASH[63:48]

The MHASH[63:48] is the High Word of the 64-bit Multicast Hash bin. This is used by the Address Filter Logic for filtering of Multicast Addressed frames when MHASH_EN is ‘1’ in Address Filter Control 2 register.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x206EH:RXXG Address Filter Control 0

Bit	Type	Function	Default
Bit 15	R/W	ADRFILT_CTRL3[3]	0
Bit 14	R/W	ADRFILT_CTRL3[2]	0
Bit 13	R/W	ADRFILT_CTRL3[1]	0
Bit 12	R/W	ADRFILT_CTRL3[0]	0
Bit 11	R/W	ADRFILT_CTRL2[3]	0
Bit 10	R/W	ADRFILT_CTRL2[2]	0
Bit 9	R/W	ADRFILT_CTRL2[1]	0
Bit 8	R/W	ADRFILT_CTRL2[1]	0
Bit 7	R/W	ADRFILT_CTRL1[3]	0
Bit 6	R/W	ADRFILT_CTRL1[2]	0
Bit 5	R/W	ADRFILT_CTRL1[1]	0
Bit 4	R/W	ADRFILT_CTRL1[0]	0
Bit 3	R/W	ADRFILT_CTRL0[3]	0
Bit 2	R/W	ADRFILT_CTRL0[2]	0
Bit 1	R/W	ADRFILT_CTRL0[1]	0
Bit 0	R/W	ADRFILT_CTRL0[0]	0

The Address Filter Control 0 register contains the Control bits for the first 4 filters 0-3, each filter needs 4 bits of control information.

ADRFILT_CTRL [3]

Forward Enable bit. When this bit is ‘1’, the Address Filter Logic will only **accept** frames that match the corresponding Exact Match Address register, and if the VLAN enable bit is set the corresponding VID_MATCH register. All other frames are filtered. When this bit is ‘0’, the Address Filter Logic will only **discard** frames that match the corresponding Exact Match Address register, and if the VLAN enable bit is set the corresponding VID_MATCH register. All other frames are filtered.

ADRFILT_CTRL[2]

VLAN Enable bit. When this bit is ‘1’, the Address Filter Logic will use the corresponding 12-bit VID_MATCH register along with the corresponding Exact Match Address register to perform the compare. When this bit is ‘0’, the Address Filter logic will only use the corresponding Exact Match Address register to perform the compare.

ADRFILT_CTRL[1]

Source Address Enable bit. When this bit is ‘0’, the Address Filter Logic will use the Destination Address to perform a compare to the corresponding Exact Match Address register. When this bit is ‘1’, the Address Filter Logic will use the Source Address to perform a compare to the corresponding Exact Match Address register.

ADRFILT_CTRL[0]

Match Enable bit. When this bit is ‘0’, the Address Filter Logic will not use the corresponding filter. When this bit is ‘1’, the Address Filter Logic will use the corresponding filter based on ADRFILT_CTRL?[3:1]. This bit must be written to ‘0’ before making any updates to the corresponding Exact Match Address and VID registers., and can then be written as ‘1’ again.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x206FH:RXXG Address Filter Control 1

Bit	Type	Function	Default
Bit 15	R/W	ADRFILT_CTRL7[3]	0
Bit 14	R/W	ADRFILT_CTRL7[2]	0
Bit 13	R/W	ADRFILT_CTRL7[1]	0
Bit 12	R/W	ADRFILT_CTRL7[0]	0
Bit 11	R/W	ADRFILT_CTRL6[3]	0
Bit 10	R/W	ADRFILT_CTRL6[2]	0
Bit 9	R/W	ADRFILT_CTRL6[1]	0
Bit 8	R/W	ADRFILT_CTRL6[1]	0
Bit 7	R/W	ADRFILT_CTRL5[3]	0
Bit 6	R/W	ADRFILT_CTRL5[2]	0
Bit 5	R/W	ADRFILT_CTRL5[1]	0
Bit 4	R/W	ADRFILT_CTRL5[0]	0
Bit 3	R/W	ADRFILT_CTRL4[3]	0
Bit 2	R/W	ADRFILT_CTRL4[2]	0
Bit 1	R/W	ADRFILT_CTRL4[1]	0
Bit 0	R/W	ADRFILT_CTRL4[0]	0

The Address Filter Control 1 register contains the Control bits for the second 4 filters 4-7, each filter needs 4 bits of control information.

ADRFILT_CTRL[3]

Forward Enable bit. When this bit is ‘1’, the Address Filter Logic will only **accept** frames that match the corresponding Exact Match Address register, and if the VLAN enable bit is set the corresponding VID_MATCH register. All other frames are filtered. When this bit is ‘0’, the Address Filter Logic will only **discard** frames that match the corresponding Exact Match Address register, and if the VLAN enable bit is set the corresponding VID_MATCH register. All other frames are filtered.

ADRFILT_CTRL[2]

VLAN Enable bit. When this bit is ‘1’, the Address Filter Logic will use the corresponding 12-bit VID_MATCH register along with the corresponding Exact Match Address register to perform the compare. When this bit is ‘1’, the Address Filter logic will only use the corresponding Exact Match Address register to perform the compare.

ADRFILT_CTRL[1]

Source Address Enable bit. When this bit is ‘0’, the Address Filter Logic will use the Destination Address to perform a compare to the corresponding Exact Match Address register. When this bit is ‘1’, the Address Filter Logic will use the Source Address to perform a compare to the corresponding Exact Match Address register.

ADRFILT_CTRL[0]

Match Enable bit. When this bit is ‘0’, the Address Filter Logic will not use the corresponding filter. When this bit is ‘1’, the Address Filter Logic will use the corresponding filter based on ADRFILT_CTRL?[3:1]. This bit must be written to ‘0’ before making any updates to the corresponding Exact Match Address and VID registers., and can then be written as ‘1’ again.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x2070H:RXXG Address Filter Control 2

Bit	Type	Function	Default
Bit 15	R	Unused	X:high
Bit 14	R	Unused	X:high
Bit 13	R	Unused	X:high
Bit 12	R	Unused	X:high
Bit 11	R	Unused	X:high
Bit 10	R	Unused	X:high
Bit 9	R	Unused	X:high
Bit 8	R	Unused	X:high
Bit 7	R	Unused	X:high
Bit 6	R	Unused	X:high
Bit 5	R	Unused	X:high
Bit 4	R	Unused	X:high
Bit 3	R	Unused	X:high
Bit 2	R	Unused	X:high
Bit 1	R/W	PMODE	1
Bit 0	R/W	MHASH_EN	0

MHASH_EN

Multicast Hash Filter Enable. When this bit is '1', the 64-bit Multicast Hash Filter function will look at all Multicast -addressed frames for filter processing. When this bit is '0', no Multicast Hash look-ups are performed.

PMODE

Promiscuous Mode. When this bit is '1', the PM3392 will allow all frames to pass through to the PL4 interface regardless of the DA/SA Addresses, unless Address Filtering Logic is enabled and there is a match to filter the incoming frame based on the DA/SA/VID fields. When this bit is '0', the PM3392 will not allow any frames to pass through to the PL4 interface unless Address Filtering Logic is enabled and there is a match to the incoming frame.

Note: Please refer to 13.17 for the proper use of these registers.

Register 0x2080H:R64B66B Configuration

Bit	Type	Function	Default
Bit 15	R	Unused	X:high
Bit 14	R	Unused	X:high
Bit 13	R/W	TIP	0
Bit 12	R	Reserved	0
Bit 11	R/W	JITTER_PATTERN_SLCT	0
Bit 10	R/W	JITTER_TEST_ENABLE	0
Bit 9	R	Reserved	0
Bit 8	R	Reserved	0
Bit 7	R	Unused	0
Bit 6	R	Unused	0
Bit 5	R	Unused	0
Bit 4	R	Reserved	0
Bit 3	R	Unused	1
Bit 2	R	Unused	0
Bit 1	R/W	INT_EN	0
Bit 0	R	Reserved	1

INT_EN

When ‘1’, the Interrupts are enabled.

When ‘0’, the Interrupts are disabled.

JITTER_TEST_ENABLE

Setting this bit to ‘1’ enables jitter test mode. This disables the BER and Receive state machines.

JITTER_PATTERN_SLCT

Setting this to ‘1’ causes the R64B66B to look for the all zeroes pattern (128 zeroes). Setting it to ‘0’ causes it to look for one of the three local fault patterns, types 0x2d, 0x4b, and 0x55. Type 0x66 is left out because it codes for start of frame.

TIP

Writing a ‘1’ to this bit copies the current value of registers 0x3 – 0x7 to a set of shadow registers. When this bit is read as ‘1’ the transfer is still in progress, the transfer typically takes a few sys_clkx2 cycles. When this bit is read as ‘0’, the transfer is complete. During the transfer the BUSY TSB output is also ‘1’.

Register 0x2081H:R64B66B Interrupt Mask

Bit	Type	Function	Default
Bit 15	R	Unused	X:high
Bit 14	R	Unused	X:high
Bit 13	R	Unused	X:high
Bit 12	R	Unused	X:high
Bit 11	R	Unused	X:high
Bit 10	R	Unused	X:high
Bit 9	R	Unused	X:high
Bit 8	R	Unused	X:high
Bit 7	R	Unused	X:high
Bit 6	R	Unused	X:high
Bit 5	R/W	LINK_FAILE	0
Bit 4	R/W	RX_LFE	0
Bit 3	R/W	RX_RFE	0
Bit 2	R/W	LOSE	0
Bit 1	R/W	HI_BERE	0
Bit 0	R/W	SYNC_ERRE	0

SYNC_ERRE: Sync Error Mask bit

When ‘0’, masks the interrupt status that the SYNC_ERR pin asserted due to a loss of signal from the optics.

When ‘1’, will cause the interrupt pin to assert when the SYNC_ERR pin asserted due to a loss of signal from the optics.

HI_BERE: High Bit-Error-Rate Mask bit

When ‘0’, masks the interrupt status indicating a High Bit-Error-Rate state.

When ‘1’, will cause the interrupt pin to assert during a High Bit-Error-Rate state.

LOSE: Loss Of Sync Mask bit

When ‘0’, masks the interrupt status indicating an Out Of Sync state.

When ‘1’, will cause the interrupt pin to assert for an Out Of Sync state.

RX_RFE: Receive Remote Fault Mask bit

When ‘0’, masks the interrupt status that the R64B66B block has just received at least 3 Remote Fault messages.

When ‘1’, will cause the interrupt pin to assert when the R64B66B block has receives at least 3 Remote Fault messages.

RX_LFE: Receive Local Fault Mask bit

When ‘0’, masks the interrupt status that the R64B66B block has just received at least 3 Local Fault messages.

When ‘1’, will cause the interrupt pin to assert when the R64B66B block has receives at least 3 Local Fault messages.

LINK_FAILE: Link Failed Mask bit

When ‘0’, masks the interrupt status that the R64B66B link has failed due to a high bit error rate or a loss of sync.

When ‘1’, will cause the interrupt pin to assert when the R64B66B link fails due to a high bit error rate or a loss of sync.

Register 0x2082H:R64B66B Interrupt Status

Bit	Type	Function	Default
Bit 15	R	Unused	X:high
Bit 14	R	Unused	X:high
Bit 13	R	Unused	X:high
Bit 12	R	Unused	X:high
Bit 11	R	Unused	X:high
Bit 10	R	Unused	X:high
Bit 9	R	Unused	X:high
Bit 8	R	Unused	X:high
Bit 7	R	Unused	X:high
Bit 6	R	Unused	X:high
Bit 5	R	LINK_FAILI	0
Bit 4	R	RX_LFI	0
Bit 3	R	RX_RFI	0
Bit 2	R	LOSI	0
Bit 1	R	HI_BERI	0
Bit 0	R	SYNC_ERRI	0

SYNC_ERRI: Sync Error status bit

When ‘1’, signals status that the SYNC_ERR pin asserted due to a loss of signal from the optics.

When ‘0’, the SYNC_ERR pin is low.

Note: This bit is a latching high bit and will assert on a rising edge event. This bit will clear on a read from this register

HI_BERI: High Bit-Error-Rate status bit

When ‘1’, signals a HI Bit-Error-Rate state.

When ‘0’, the BER is below 16 errors per 125us.

Note: This bit is a latching high bit and will assert on a rising edge event. This bit will clear on a read from this register

LOSI: Loss Of Sync interrupt status bit

When ‘1’, signals an Out Of Sync state. Sync can be lost with reset, sync_err input, or more than 32 invalid sync fields in 64 66-bit frames.

When ‘0’, the Sync detector logic is sync’d.

Note: This bit is a latching high bit and will assert on a rising edge event. This bit will clear on a read from this register

RX_RFI: Receive Remote Fault Interrupt status bit

When ‘1’, signals status that the R64B66B block has just received at least 3 Remote Fault messages.

When ‘0’, means that the PCS is not in a Remote Fault state.

Note: This bit is a latching high bit and will assert on a rising edge event. This bit will clear on a read from this register

RX_LFI: Receive Local Fault Interrupt status bit

When ‘1’, signal status that the R64B66B block has just received at least 3 Local Fault messages.

When ‘0’, means that the PCS is not in a Local Fault state.

Note: This bit is a latching high bit and will assert on a rising edge event. This bit will clear on a read from this register

LINK_FAIL: Link Failed Interrupt status bit

When ‘1’, signals status that the R64B66B link has failed due to either Loss Of Sync or a Local Fault status was observed.

When ‘0’, means that the Receive State Machine is receiving good data.

Note: This bit is a latching high bit and will assert on a rising edge event. This bit will clear on a read from this register

Register 0x2083H:R64B66B Status

Bit	Type	Function	Default
Bit 15	R	Unused	X:high
Bit 14	R	Unused	X:high
Bit 13	R	Unused	X:high
Bit 12	R	Unused	X:high
Bit 11	R	Unused	X:high
Bit 10	R	Unused	X:high
Bit 9	R	Unused	X:high
Bit 8	R	Unused	X:high
Bit 7	R	Unused	X:high
Bit 6	R	Unused	X:high
Bit 5	R	LINK_FAIL	1
Bit 4	R	RX_LF	0
Bit 3	R	RX_RF	0
Bit 2	R	LOS	1
Bit 1	R	HI_BER	0
Bit 0	R	SYNC_ERR	X:high

NOTE: To update these status bits one must first write to the R64B66B TIP bit (bit-15) in Register 0x2080 R64B66B Configuration. Then a read can be performed to get the status.

SYNC_ERR: Sync Error current status bit

When ‘1’, signals status that the SYNC_ERR pin asserted due to a loss of signal from the optics.

When ‘0’, the SYNC_ERR pin is low.

HI_BER: High Bit-Error-Rate current status bit

When ‘1’, signals a HI Bit-Error-Rate state.

When ‘0’, the BER is below 16 errors per 125us.

LOS: Loss Of Sync current status bit

When ‘1’, signals an Out Of Sync state. Sync can be lost with reset, sync_err input, or more than 32 invalid sync fields in 64 66-bit frames.

When ‘0’, the Sync detector logic is sync’d.

RX_RF: Receive Remote Fault current status bit

When ‘1’, signals status that the R64B66B block has just received at least 3 Remote Fault messages.

When ‘0’, means that the PCS is not in a Remote Fault state. RX_LF: Receive Local Fault current status bit

When ‘1’, signal status that the R64B66B block has just received at least 3 Local Fault messages.

When ‘0’, means that the PCS is not in a Local Fault state.

LINK_FAIL: Link Failed current status bit

When ‘1’, signals status that the R64B66B link has failed due to the either Loss Of Sync or a Local Fault status was observed.

When ‘0’, means that the R64B66B Link is up.

Register 0x2084H:R64B66B ERROR_FRAME_CNT

Bit	Type	Function	Default
Bit 15	R	Unused	X:high
Bit 14	R	Unused	X:high
Bit 13	R	Unused	X:high
Bit 12	R	Unused	X:high
Bit 11	R	Unused	X:high
Bit 10	R	Unused	X:high
Bit 9	R	Unused	X:high
Bit 8	R	Unused	X:high
Bit 7	R	ERROR_FRAME_CNT[7]	X:high
Bit 6	R	ERROR_FRAME_CNT[6]	X:high
Bit 5	R	ERROR_FRAME_CNT[5]	X:high
Bit 4	R	ERROR_FRAME_CNT[4]	X:high
Bit 3	R	ERROR_FRAME_CNT[3]	X:high
Bit 2	R	ERROR_FRAME_CNT[2]	X:high
Bit 1	R	ERROR_FRAME_CNT[1]	X:high
Bit 0	R	ERROR_FRAME_CNT[0]	X:high

ERROR_FRAME_CNT[7:0]

These bits indicate the number of invalid PCS code sequences detected in the Ingress path. The counter only increments when the R64B66B block is in a SYNC'd state. The counter should be polled regularly to avoid saturating. The counter can saturate at values 0x00fe or 0x00ff. This register is cleared on read.

Register 0x2085H:R64B66B FRAME_LOCK_COUNT[15:0]

Bit	Type	Function	Default
Bit 15	R	FRAME_LOCK_COUNT[15]	X:high
Bit 14	R	FRAME_LOCK_COUNT[14]	X:high
Bit 13	R	FRAME_LOCK_COUNT[13]	X:high
Bit 12	R	FRAME_LOCK_COUNT[12]	X:high
Bit 11	R	FRAME_LOCK_COUNT[11]	X:high
Bit 10	R	FRAME_LOCK_COUNT[10]	X:high
Bit 9	R	FRAME_LOCK_COUNT[9]	X:high
Bit 8	R	FRAME_LOCK_COUNT[8]	X:high
Bit 7	R	FRAME_LOCK_COUNT[7]	X:high
Bit 6	R	FRAME_LOCK_COUNT[6]	X:high
Bit 5	R	FRAME_LOCK_COUNT[5]	X:high
Bit 4	R	FRAME_LOCK_COUNT[4]	X:high
Bit 3	R	FRAME_LOCK_COUNT[3]	X:high
Bit 2	R	FRAME_LOCK_COUNT[2]	X:high
Bit 1	R	FRAME_LOCK_COUNT[1]	X:high
Bit 0	R	FRAME_LOCK_COUNT[0]	X:high

FRAME_LOCK_COUNT[15:0]

These bits indicate the number times the Synchronizer state machine has gone from a non-sync'd state to a sync'd state. The counter should be polled regularly to avoid saturating. The counter can saturate at values 0xffff or 0xfffe. This register is cleared on read.

Register 0x2086H:R64B66B HI_BER_CNT[4:0]

Bit	Type	Function	Default
Bit 15	R	Unused	X:high
Bit 14	R	Unused	X:high
Bit 13	R	Unused	X:high
Bit 12	R	Unused	X:high
Bit 11	R	Unused	X:high
Bit 10	R	Unused	X:high
Bit 9	R	Unused	X:high
Bit 8	R	Unused	X:high
Bit 7	R	Unused	X:high
Bit 6	R	Unused	X:high
Bit 5	R	HI_BER_CNT[5]	X:high
Bit 4	R	HI_BER_CNT[4]	X:high
Bit 3	R	HI_BER_CNT[3]	X:high
Bit 2	R	HI_BER_CNT[2]	X:high
Bit 1	R	HI_BER_CNT[1]	X:high
Bit 0	R	HI_BER_CNT[0]	X:high

HI_BER_CNT[5:0]

These bits indicate the number of Bit errors during a 125us interval. The counter resets to zero after every 125us. The counter only increments when the R64B66B block is in a SYNC'd state. This register is cleared on read.

Register 0x2087H: R64B66B JITTER_CNT[15:0]

Bit	Type	Function	Default
Bit 15	R	JITTER_CNT[15]	x
Bit 14	R	JITTER_CNT[14]	x
Bit 13	R	JITTER_CNT[13]	x
Bit 12	R	JITTER_CNT[12]	x
Bit 11	R	JITTER_CNT[11]	x
Bit 10	R	JITTER_CNT[10]	x
Bit 9	R	JITTER_CNT[9]	x
Bit 8	R	JITTER_CNT[8]	x
Bit 7	R	JITTER_CNT[7]	x
Bit 6	R	JITTER_CNT[6]	x
Bit 5	R	JITTER_CNT[5]	x
Bit 4	R	JITTER_CNT[4]	x
Bit 3	R	JITTER_CNT[3]	x
Bit 2	R	JITTER_CNT[2]	x
Bit 1	R	JITTER_CNT[1]	x
Bit 0	R	JITTER_CNT[0]	x

This register is updated by setting the TIP bit (bit-13) in Register 0x2080 R64B668 Configuration.

JITTER_CNT[15:0]

Putting the R64B66B into jitter mode (register 0, bit 10) enables this counter. This counter is incremented for every unscrambled 66 bit frame that doesn't match the expected frame data. The expected frame type is set by register 0, bit 11. Register 7H is cleared upon read. See section 49.2.12 of the IEEE 802.3ae standard for a description of the receive jitter algorithm. The counter can saturate at values 0xffff or 0xfffe.

Register 0x2100H:MSTAT Control

Bit	Type	Function	Default
Bit 15	R	Reserved	0
Bit 14	R	Reserved	0
Bit 13	R	Reserved	0
Bit 12	R	Reserved	0
Bit 11	R	Reserved	0
Bit 10	R	Reserved	0
Bit 9	R	Reserved	0
Bit 8	R	Reserved	0
Bit 7	R	Reserved	0
Bit 6	R	Reserved	0
Bit 5	R	Reserved	0
Bit 4	R	Reserved	0
Bit 3	R	Reserved	0
Bit 2	R/W	WRITE	0
Bit 1	R/W	CLEAR	0
Bit 0	R/W	SNAP	0

The MSTAT Control Register provides general control over the MSTAT.

SNAP

The SNAP bit is used to snap all management statistics counters into their complimentary system probe shadow registers for full static system probes. The SNAP bit will perform the copy operation when set high (logic 1). The SNAP bit will automatically clear itself (logic 0) after two positive clock edges (the clock can be RDCKI or TDCKI).

CLEAR

The CLEAR bit is used to clear all management statistic registers. The CLEAR bit clear all registers when set high (logic 1). The CLEAR bit will automatically clear itself (logic 0) after two positive clock edges (the clock can be RDCKI or TDCKI).

WRITE

The WRITE bit is used to initiate a data update write to the selected counter indicated by the MSTAT Counter Write Address Register. The contents of the MSTAT Counter Write Data Registers will be copied into the associative counter. The write is initiated by setting this bit high (logic 1). The WRITE bit will automatically clear itself (logic 0) after two positive clock edges (the clock can be RDCKI or TDCKI).

Register 0x2101H:MSTAT Counter Rollover 0

Bit	Type	Function	Default
Bit 15	R	FramesTooLongErrors	0
Bit 14	R	Reserved	0
Bit 13	R	InRangeLengthErrors	0
Bit 12	R	SymbolError	0
Bit 11	R	FramesLostDueToInternalMACError	0
Bit 10	R	FrameCheckSequenceErrors	0
Bit 9	R	MACControlFrameReceived	0
Bit 8	R	PAUSEMACControlFrameReceived	0
Bit 7	R	TaggedFramesReceived	0
Bit 6	R	BroadcastFramesReceivedOK	0
Bit 5	R	MulticastFramesReceivedOK	0
Bit 4	R	UnicastFramesReceivedOK	0
Bit 3	R	OctetsReceived	0
Bit 2	R	FramesReceived	0
Bit 1	R	OctetsReceivedOK	0
Bit 0	R	FramesReceivedOK	0

The MSTAT Counter Rollover Registers provide sticky indication of counter roll over conditions. Reading this register clears all bits within this register. Each bit represents the given counter name as documented.

Register 0x2102H:MSTAT Counter Rollover 1

Bit	Type	Function	Default
Bit 15	R	Reserved	0
Bit 14	R	FilteredBroadcastFrames	0
Bit 13	R	FilteredMulticastFrames	0
Bit 12	R	FilteredUnicastFrames	0
Bit 11	R	FilteredOctets	0
Bit 10	R	JumboOctetsReceivedOK	0
Bit 9	R	ReceiveFrames1519toMAXOctets	0
Bit 8	R	ReceiveFrames1024to1518Octets	0
Bit 7	R	ReceiveFrames512to1023Octets	0
Bit 6	R	ReceiveFrames256to511Octets	0
Bit 5	R	ReceiveFrames128to255Octets	0
Bit 4	R	ReceiveFrames65to127Octets	0
Bit 3	R	ReceiveFrames64Octets	0
Bit 2	R	UndersizedFrames	0
Bit 1	R	Fragments	0
Bit 0	R	Jabbers	0

The MSTAT Counter Rollover Registers provide sticky indication of counter roll over conditions. Reading this register clears all bits within this register. Each bit represents the given counter name as documented.

Register 0x2103H:MSTAT Counter Rollover 2

Bit	Type	Function	Default
Bit 15	R	TransmittedFrames128to255Octets	0
Bit 14	R	TransmittedFrames65to127Octets	0
Bit 13	R	TransmittedFrames64Octets	0
Bit 12	R	MACCTRLFramesTransmitted	0
Bit 11	R	PAUSEMACCTRLFramesTransmitted	0
Bit 10	R	BroadcastFramesTransmittedOK	0
Bit 9	R	BroadcastFramesTranmittedAttempted	0
Bit 8	R	MulticastFramesTransmittedOK	0
Bit 7	R	MulticastFramesTransmittedAttempted	0
Bit 6	R	UnicastFramesTransmittedOK	0
Bit 5	R	UnicastFramesTransmittedAttempted	0
Bit 4	R	TransmitSystemError	0
Bit 3	R	FramesLostDueToInternalMacTransmissionError	0
Bit 2	R	OctetsTransmitted	0
Bit 1	R	OctetsTransmittedOK	0
Bit 0	R	FramesTransmitteOK	0

The MSTAT Counter Rollover Registers provide sticky indication of counter roll over conditions. Reading this register clears all bits within this register. Each bit represents the given counter name as documented.

Register 0x2104H:MSTAT Counter Rollover 3

Bit	Type	Function	Default
Bit 15	R	Reserved	0
Bit 14	R	Reserved	0
Bit 13	R	Reserved	0
Bit 12	R	Reserved	0
Bit 11	R	Reserved	0
Bit 10	R	Reserved	0
Bit 9	R	Reserved	0
Bit 8	R	Reserved	0
Bit 7	R	Reserved	0
Bit 6	R	Reserved	0
Bit 5	R	Reserved	0
Bit 4	R	JumboOctetsTransmittedOK	0
Bit 3	R	TransmittedFrames1519toMAXOctets	0
Bit 2	R	TransmittedFrames1024to1518Octets	0
Bit 1	R	TransmittedFrames512to1023Octets	0
Bit 0	R	TransmittedFrames256to511Octets	0

The MSTAT Counter Rollover Registers provide sticky indication of counter roll over conditions. Reading this register clears all bits within this register. Each bit represents the given counter name as documented.

Register 0x2105H:MSTAT Interrupt Mask 0

Bit	Type	Function	Default
Bit 15	R/W	MASK0[15]	0
Bit 14	R/W	MASK0[14]	0
Bit 13	R/W	MASK0[13]	0
Bit 12	R/W	MASK0[12]	0
Bit 11	R/W	MASK0[11]	0
Bit 10	R/W	MASK0[10]	0
Bit 9	R/W	MASK0[9]	0
Bit 8	R/W	MASK0[8]	0
Bit 7	R/W	MASK0[7]	0
Bit 6	R/W	MASK0[6]	0
Bit 5	R/W	MASK0[5]	0
Bit 4	R/W	MASK0[4]	0
Bit 3	R/W	MASK0[3]	0
Bit 2	R/W	MASK0[2]	0
Bit 1	R/W	MASK0[1]	0
Bit 0	R/W	MASK0[0]	0

The MSTAT Interrupt Mask Registers provide programmable interrupt masking of the MSTAT Counter Rollover Register bits.

MASK0[15:0]

The MASK0[15:0] bits are used as a logical mask for each corresponding bit in the **MSTAT Counter Rollover Register 0**. If the MASK bit is high (logic 1), the given counter overflow condition in the **MSTAT Counter Rollover Register 0** will cause the MSTAT to assert the INT pin high (logic 1). If the MASK bit is low (logic 0), the corresponding **MSTAT Counter Rollover Register 0** bit state has no effect on the MSTAT INT pin.

Register 0x2106H:MSTAT Interrupt Mask 1

Bit	Type	Function	Default
Bit 15	R	Reserved	0
Bit 14	R/W	MASK1[14]	0
Bit 13	R/W	MASK1[13]	0
Bit 12	R/W	MASK1[12]	0
Bit 11	R/W	MASK1[11]	0
Bit 10	R/W	MASK1[10]	0
Bit 9	R/W	MASK1[9]	0
Bit 8	R/W	MASK1[8]	0
Bit 7	R/W	MASK1[7]	0
Bit 6	R/W	MASK1[6]	0
Bit 5	R/W	MASK1[5]	0
Bit 4	R/W	MASK1[4]	0
Bit 3	R/W	MASK1[3]	0
Bit 2	R/W	MASK1[2]	0
Bit 1	R/W	MASK1[1]	0
Bit 0	R/W	MASK1[0]	0

The MSTAT Interrupt Mask Registers provide programmable interrupt masking of the MSTAT Counter Rollover Register bits.

MASK1[14:0]

The MASK1[14:0] bits are used as a logical mask for each corresponding bit in the **MSTAT Counter Rollover Register 1**. If the MASK bit is high (logic 1), the given counter overflow condition in the **MSTAT Counter Rollover Register 1** will cause the MSTAT to assert the INT pin high (logic 1). If the MASK bit is low (logic 0), the corresponding **MSTAT Counter Rollover Register 1** bit state has no effect on the MSTAT INT pin.

Register 0x2107H:MSTAT Interrupt Mask 2

Bit	Type	Function	Default
Bit 15	R/W	MASK2[15]	0
Bit 14	R/W	MASK2[14]	0
Bit 13	R/W	MASK2[13]	0
Bit 12	R/W	MASK2[12]	0
Bit 11	R/W	MASK2[11]	0
Bit 10	R/W	MASK2[10]	0
Bit 9	R/W	MASK2[9]	0
Bit 8	R/W	MASK2[8]	0
Bit 7	R/W	MASK2[7]	0
Bit 6	R/W	MASK2[6]	0
Bit 5	R/W	MASK2[5]	0
Bit 4	R/W	MASK2[4]	0
Bit 3	R/W	MASK2[3]	0
Bit 2	R/W	MASK2[2]	0
Bit 1	R/W	MASK2[1]	0
Bit 0	R/W	MASK2[0]	0

The MSTAT Interrupt Mask Registers provide programmable interrupt masking of the MSTAT Counter Rollover Register bits.

MASK2[15:0]

The MASK2[15:0] bits are used as a logical mask for each corresponding bit in the **MSTAT Counter Rollover Register 2**. If the MASK bit is high (logic 1), the given counter overflow condition in the **MSTAT Counter Rollover Register 2** will cause the MSTAT to assert the INT pin high (logic 1). If the MASK bit is low (logic 0), the corresponding **MSTAT Counter Rollover Register 2** bit state has no effect on the MSTAT INT pin.

Register 0x2108H:MSTAT Interrupt Mask 3

Bit	Type	Function	Default
Bit 15	R	Reserved	0
Bit 14	R	Reserved	0
Bit 13	R	Reserved	0
Bit 12	R	Reserved	0
Bit 11	R	Reserved	0
Bit 10	R	Reserved	0
Bit 9	R	Reserved	0
Bit 8	R	Reserved	0
Bit 7	R	Reserved	0
Bit 6	R	Reserved	0
Bit 5	R/W	MASK3[5]	0
Bit 4	R/W	MASK3[4]	0
Bit 3	R/W	MASK3[3]	0
Bit 2	R/W	MASK3[2]	0
Bit 1	R/W	MASK3[1]	0
Bit 0	R/W	MASK3[0]	0

The MSTAT Interrupt Mask Registers provide programmable interrupt masking of the MSTAT Counter Rollover Register bits.

MASK3[5:0]

The MASK3[5:0] bits are used as a logical mask for each corresponding bit in the **MSTAT Counter Rollover Register 3**. If the MASK bit is high (logic 1), the given counter overflow condition in the **MSTAT Counter Rollover Register 3** will cause the MSTAT to assert the INT pin high (logic 1). If the MASK bit is low (logic 0), the corresponding **MSTAT Counter Rollover Register 3** bit state has no effect on the MSTAT INT pin.

Register 0x2109H:MSTAT Counter Write Address

Bit	Type	Function	Default
Bit 15	R	Reserved	0
Bit 14	R	Reserved	0
Bit 13	R	Reserved	0
Bit 12	R	Reserved	0
Bit 11	R	Reserved	0
Bit 10	R	Reserved	0
Bit 9	R	Reserved	0
Bit 8	R	Reserved	0
Bit 7	R	Reserved	0
Bit 6	R	Reserved	0
Bit 5	R/W	ADDRESS[5]	0
Bit 4	R/W	ADDRESS[4]	0
Bit 3	R/W	ADDRESS[3]	0
Bit 2	R/W	ADDRESS[2]	0
Bit 1	R/W	ADDRESS[1]	0
Bit 0	R/W	ADDRESS[0]	0

The MSTAT Counter Write Address Register provides the write address used during a write operation to the MSTAT counters.

ADDRESS[5:0]

The ADDRESS[5:0] bits are used as the write address during a write operation to the MSTAT counters. A proper counter address must be written to the MSTAT Counter Write Address prior to initiating a write operation via the WRITE bit in the MSTAT Control Register.

Register 0x210AH:MSTAT Counter Write Data Low

Bit	Type	Function	Default
Bit 15	R/W	DATA[15]	0
Bit 14	R/W	DATA[14]	0
Bit 13	R/W	DATA[13]	0
Bit 12	R/W	DATA[12]	0
Bit 11	R/W	DATA[11]	0
Bit 10	R/W	DATA[10]	0
Bit 9	R/W	DATA[9]	0
Bit 8	R/W	DATA[8]	0
Bit 7	R/W	DATA[7]	0
Bit 6	R/W	DATA[6]	0
Bit 5	R/W	DATA[5]	0
Bit 4	R/W	DATA[4]	0
Bit 3	R/W	DATA[3]	0
Bit 2	R/W	DATA[2]	0
Bit 1	R/W	DATA[1]	0
Bit 0	R/W	DATA[0]	0

The MSTAT Counter Write Data Registers provide the write data used during a write operation to the MSTAT counters. The MSTAT Counter Write Data Registers are partitioned into low, middle, and high register entities.

DATA[15:0]

The DATA[15:0] bits are used as the write data during a write operation to the MSTAT counters. The proper counter data must be written to the MSTAT Counter Write Data Registers prior to initiating a write operation via the WRITE bit in the MSTAT Control Register.

Register 0x210BH:MSTAT Counter Write Data Middle

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

The MSTAT Counter Write Data Registers provide the write data used during a write operation to the MSTAT counters. The MSTAT Counter Write Data Registers are partitioned into low, middle, and high register entities.

DATA[31:16]

The DATA[31:16] bits are used as the write data during a write operation to the MSTAT counters. The proper counter data must be written to the MSTAT Counter Write Data Registers prior to initiating a write operation via the WRITE bit in the MSTAT Control Register.

Register 0x210CH:MSTAT Counter Write Data High

Bit	Type	Function	Default
Bit 15	R	Reserved	0
Bit 14	R	Reserved	0
Bit 13	R	Reserved	0
Bit 12	R	Reserved	0
Bit 11	R	Reserved	0
Bit 10	R	Reserved	0
Bit 9	R	Reserved	0
Bit 8	R	Reserved	0
Bit 7	R/W	DATA[39]	0
Bit 6	R/W	DATA[38]	0
Bit 5	R/W	DATA[37]	0
Bit 4	R/W	DATA[36]	0
Bit 3	R/W	DATA[35]	0
Bit 2	R/W	DATA[34]	0
Bit 1	R/W	DATA[33]	0
Bit 0	R/W	DATA[32]	0

The MSTAT Counter Write Data Registers provide the write data used during a write operation to the MSTAT counters. The MSTAT Counter Write Data Registers are partitioned into low, middle, and high register entities.

DATA[39:32]

The DATA[39:32] bits are used as the write data during a write operation to the MSTAT counters. The proper counter data must be written to the MSTAT Counter Write Data Registers prior to initiating a write operation via the WRITE bit in the MSTAT Control Register.

Register 0x2110H to 0x21E6H:MSTAT Receive Statistical Counters' Low

Bit	Type	Function	Default
Bit 15	R	COUNT[15]	0
Bit 14	R	COUNT[14]	0
Bit 13	R	COUNT[13]	0
Bit 12	R	COUNT[12]	0
Bit 11	R	COUNT[11]	0
Bit 10	R	COUNT[10]	0
Bit 9	R	COUNT[9]	0
Bit 8	R	COUNT[8]	0
Bit 7	R	COUNT[7]	0
Bit 6	R	COUNT[6]	0
Bit 5	R	COUNT[5]	0
Bit 4	R	COUNT[4]	0
Bit 3	R	COUNT[3]	0
Bit 2	R	COUNT[2]	0
Bit 1	R	COUNT[1]	0
Bit 0	R	COUNT[0]	0

The MSTAT Statistical Counters are defined in Table 15. The MSTAT Statistical Counter is a 40-bits counter. The MSTAT Statistical Counters represent the individual counters split between high, middle, and low registers. The low register contains bits 15:0, the middle register contains bits 31:16, and the high register contains bits 39:32 as well as 8 unused or reserved bits in the MSB of the high register.

COUNT[15:0]

The COUNT[15:0] bits are used as the low 16-bit of the counter.

Register 0x2110H to 0x21E6H:MSTAT Receive Statistical Counters' Middle

Bit	Type	Function	Default
Bit 15	R	COUNT[31]	0
Bit 14	R	COUNT[30]	0
Bit 13	R	COUNT[29]	0
Bit 12	R	COUNT[28]	0
Bit 11	R	COUNT[27]	0
Bit 10	R	COUNT[26]	0
Bit 9	R	COUNT[25]	0
Bit 8	R	COUNT[24]	0
Bit 7	R	COUNT[23]	0
Bit 6	R	COUNT[22]	0
Bit 5	R	COUNT[21]	0
Bit 4	R	COUNT[20]	0
Bit 3	R	COUNT[19]	0
Bit 2	R	COUNT[18]	0
Bit 1	R	COUNT[17]	0
Bit 0	R	COUNT[16]	0

The MSTAT Statistical Counters are defined in Table 15. The MSTAT Statistical Counter is a 40-bits counter. The MSTAT Statistical Counters represent the individual counters split between high, middle, and low registers. The low register contains bits 15:0, the middle register contains bits 31:16, and the high register contains bits 39:32 as well as 8 unused or reserved bits in the MSB of the high register.

COUNT[31:16]

The COUNT[31:16] bits are used as the middle 16-bit of the counter.

Register 0x2110H to 0x21E6H:MSTAT Receive Statistical Counters' High

Bit	Type	Function	Default
Bit 15	R	Reserved	0
Bit 14	R	Reserved	0
Bit 13	R	Reserved	0
Bit 12	R	Reserved	0
Bit 11	R	Reserved	0
Bit 10	R	Reserved	0
Bit 9	R	Reserved	0
Bit 8	R	Reserved	0
Bit 7	R/W	COUNT[39]	0
Bit 6	R/W	COUNT[38]	0
Bit 5	R/W	COUNT[37]	0
Bit 4	R/W	COUNT[36]	0
Bit 3	R/W	COUNT[35]	0
Bit 2	R/W	COUNT[34]	0
Bit 1	R/W	COUNT[33]	0
Bit 0	R/W	COUNT[32]	0

The MSTAT Statistical Counters are defined in Table 15. The MSTAT Statistical Counter is a 40-bits counter. The MSTAT Statistical Counters represent the individual counters split between high, middle, and low registers. The low register contains bits 15:0, the middle register contains bits 31:16, and the high register contains bits 39:32 as well as 8 unused or reserved bits in the MSB of the high register.

COUNT[39:32]

The COUNT[39:32] bits are used as the high 8-bit of the counter.

Table 15 MSTAT Counter Description

MSTAT Counter Registers		
Read Address		
0x2110	Low	FramesReceivedOK Contains a count of frames that are successfully received. This does not include frames received that are classified under: FrameCheckSequenceErrors, FramesLostDueToInternalMACError, SymbolError, InRangeLengthErrors, OutofRangeLengthErrors, FramesTooLongErrors, Jabbers, Fragments, or UndersizedFrames. MSTAT Counter Write Address = 0x0
0x2111	Mid	
0x2112	High	
0x2114	Low	OctetsReceivedOK Contains a count of data and padding octets in frames (not including Preamble, SFD, destination/source address, type/length field, Q-Tag prefix or FCS) that are successfully received. This does not include octets in frames received that are classified under: FrameCheckSequenceErrors, FramesLostDueToInternalMACError, SymbolError, InRangeLengthErrors, OutofRangeLengthErrors, FramesTooLongErrors, Jabbers, Fragments, UndersizedFrames ifInOctets (MIB-II) can be computed using the following: ifInOctets = OctetsReceivedOK + (18 * FramesReceivedOK) ifInOctets includes the count of data, padding, destination/source address, length/type field, Q-Tag prefix, and FCS. (excludes preamble and SFD). MSTAT Counter Write Address = 0x1
0x2115	Mid	
0x2116	High	
0x2118	Low	FramesReceived The total number of frames (including bad frames, unicast frames, broadcast frames, and multicast frames) received. This count includes those frames of Jumbo Size. MSTAT Counter Write Address = 0x2
0x2119	Mid	
0x211A	High	
0x211C	Low	OctetsReceived The total number of octets of data (including those in bad frames) received (excluding framing bits but including FCS octets). This includes the count of bytes from the first byte of the Destination address to the last byte of the FCS field. MSTAT Counter Write Address = 0x3
0x211D	Mid	
0x211E	High	

MSTAT Counter Registers		
Read Address		
0x2120	Low	UnicastFramesReceivedOK Contains a count of frames that are successfully received and are directed to a unicast group address. This does not include octets in frames received that are classified under: FrameCheckSequenceErrors, FramesLostDueToInternalMACError, SymbolError, InRangeLengthErrors, OutofRangeLengthErrors, FramesTooLongErrors, Jabbers, Fragments, or UndersizedFrames. MSTAT Counter Write Address = 0x4
0x2121	Mid	
0x2122	High	
0x2124	Low	MulticastFramesReceivedOK Contains count of frames that are successfully received and are directed to a multicast group address. This counter will not increment for frames classified as unicast or broadcast. This does not include frames received that are classified under: FrameCheckSequenceErrors, FramesLostDueToInternalMACError, SymbolError, InRangeLengthErrors, OutofRangeLengthErrors, FramesTooLongErrors, Jabbers, Fragments, or UndersizedFrames. MSTAT Counter Write Address = 0x5
0x2125	Mid	
0x2126	High	
0x2128	Low	BroadcastFramesReceivedOK Contains a count of frames that are successfully received and are directed to the broadcast group address. This counter will not increment for frames classified as unicast or multicast. This does not include frames received that are classified under: FrameCheckSequenceErrors, FramesLostDueToInternalMACError, SymbolError, InRangeLengthErrors, OutofRangeLengthErrors, FramesTooLongErrors, Jabbers, Fragments, or UndersizedFrames. MSTAT Counter Write Address = 0x6
0x2129	Mid	
0x212A	High	
0x212C	Low	TaggedFramesReceived Contains a count of all tagged frames that are received. MSTAT Counter Write Address = 0x7
0x212D	Mid	
0x212E	High	

MSTAT Counter Registers		
Read Address		
0x2130	Low	PAUSEMACControlFrameReceived Contains a count of MAC Control frames passed by the MAC sublayer to the MAC Control sublayer. This counter is incremented when a ReceiveFrame function call returns a valid frame with: A lengthOrType field value equal to the reserved Type for 802.3_MAC_Control as specified in 802.3-1998 (31.4.1.3), and An opcode indicating the PAUSE operation. This does not include frames received that are classified under: FrameCheckSequenceErrors, FramesLostDueToInternalMACError, SymbolError, InRangeLengthErrors, OutofRangeLengthErrors, FramesTooLongErrors, Jabbers, Fragments, or UndersizedFrames. MSTAT Counter Write Address = 0x8
0x2131	Mid	
0x2132	High	
0x2134	Low	MACControlFrameReceived Contains a count of MAC Control frames passed by the MAC sublayer to the MAC Control sublayer. This counter is incremented when a ReceiveFrame function call returns a valid frame with: (1) a lengthOrType field value equal to the reserved Type for 802.3_MAC_Control as specified in 802.3-1998 (31.4.1.3). This does not include frames received that are classified under: FrameCheckSequenceErrors, FramesLostDueToInternalMACError, SymbolError, InRangeLengthErrors, OutofRangeLengthErrors, FramesTooLongErrors, Jabbers, Fragments, or UndersizedFrames. MSTAT Counter Write Address = 0x9
0x2135	Mid	
0x2136	High	
0x2138	Low	FrameCheckSequenceErrors Contains a count of receive frames that are an integral number of octets in length and do not pass the FCS check. This does not include frames received that are too long(jabbers), or too short (fragments). MSTAT Counter Write Address = 0xA
0x2139	Mid	
0x213A	High	
0x213C	Low	FramesLostDueToInternalMACError Contains a count of frames that would otherwise be received by the device, but could not be accepted due to an internal MAC sublayer receive error (I.E. FIFO overrun). If this counter is incremented, then none of the other error counters in this section are incremented. MSTAT Counter Write Address = 0xB
0x213D	Mid	
0x213E	High	

MSTAT Counter Registers		
Read Address		
0x2140	Low	SymbolError A count of the number of times when valid length frame was received at the port and during which time there was at least one occurrence of an event that causes the PHY to indicate “Data reception error” or invalid “Data symbol error.” This counter shall be incremented only once per valid CarrierEvent. This counter will increment when the Receive PCS experiences the following errors: • Receive State Machine transitions to the ERROR state as per the 802.3ae standard • Loss of sync • HI_BER state • Receptions of Ordered sets MSTAT Counter Write Address = 0xC
0x2141	Mid	
0x2142	High	
0x2144	Low	InRangeLengthErrors Contains a count of frames with a length/type field value between 46 and 1500 that does not match the number of MAC client data octets received. The counter also increments for frames whose length/type field value is from 0 to 45 regardless of the number of MAC client data octets received. MSTAT Counter Write Address = 0xD
0x2145	Mid	
0x2146	High	
0x214C	Low	FramesTooLongErrors Contains a count of frames received that exceed the maximum permitted frame size and have no other errors. This counter is aware of both tagged and non tagged frames as well as frames of Jumbo size. MSTAT Counter Write Address = 0xF
0x214D	Mid	
0x214E	High	
0x2150	Low	Jabbers Contains a count of the total number of frames received that were longer than the maximum permitted frame size and had a bad Frame Check Sequence (FCS). MSTAT Counter Write Address = 0x10
0x2151	Mid	
0x2152	High	

MSTAT Counter Registers		
Read Address		
0x2154	Low	Fragments The total number of frames received that were less than minimum permitted frame size (64 octets long excluding framing bits, but including FCS octets) and had a bad frame check sequence (FCS). MSTAT Counter Write Address = 0x11
0x2155	Mid	
0x2156	High	
0x2158	Low	UndersizedFrames The total number of frames received that were less than the minimum permitted frame size (64 octets long excluding framing bits, but including FCS octets) and were otherwise well formed. MSTAT Counter Write Address = 0x12
0x2159	Mid	
0x215A	High	
0x215C	Low	ReceiveFrames64Octets The total number of frames (including bad frames) received that were 64 octets in length (excluding framing bits but including FCS octets). MSTAT Counter Write Address = 0x13
0x215D	Mid	
0x215E	High	
0x2160	Low	ReceiveFrames65to127Octets The total number of frames (including bad frames) received that were between 65 and 127 octets in length inclusive (excluding framing bits but including FCS octets). MSTAT Counter Write Address = 0x14
0x2161	Mid	
0x2162	High	
0x2164	Low	ReceiveFrames128to255Octets The total number of frames (including bad frames) received that were between 128 and 255 octets in length inclusive (excluding framing bits but including FCS octets). MSTAT Counter Write Address = 0x15
0x2165	Mid	
0x2166	High	
0x2168	Low	ReceiveFrames256to511Octets The total number of frames (including bad frames) received that were between 256 and 511 octets in length inclusive (excluding framing bits but including FCS octets). MSTAT Counter Write Address = 0x16
0x2169	Mid	
0x216A	High	
0x216C	Low	ReceiveFrames512to1023Octets The total number of frames (including bad frames) received that were between 512 and 1023 octets in length inclusive (excluding framing bits but including FCS octets). MSTAT Counter Write Address = 0x17
0x216D	Mid	
0x216E	High	

MSTAT Counter Registers		
Read Address		
0x2170	Low	ReceiveFrames1024to1518Octets The total number of frames (including bad frames) received that were between 1024 and (1518 octets for untagged frames and 1522 octets for VLAN tagged frames) in length inclusive (excluding framing bits but including FCS octets). MSTAT Counter Write Address = 0x18
0x2171	Mid	
0x2172	High	
0x2174	Low	ReceiveFrames1519toMAXOctets The total number of frames (including bad frames) received that were between the maximum normal frame lengths (1518 octets for untagged frames and 1522 octets for tagged frames) and maximum Jumbo frame lengths (i.e. 9600 octets) (excluding framing bits but including FCS octets). MSTAT Counter Write Address = 0x19
0x2175	Mid	
0x2176	High	
0x2178	Low	JumboOctetsReceivedOK The total number of octets (excluding bad frames) received that were between the maximum normal frame lengths (1518 octets for untagged frames and 1522 octets for tagged frames) and maximum Jumbo frame lengths (i.e. up to MaxFrameSize) (excluding framing bits but including FCS octets). MSTAT Counter Write Address = 0x1A
0x2179	Mid	
0x217A	High	
0x217C	Low	FilteredOctets The total number of octets that would normally be passed to the link that are dropped because of filtering rules. MSTAT Counter Write Address = 0x1B
0x217D	Mid	
0x217E	High	
0x2180	Low	FilteredUnicastFrames The total number of Unicast classified fames that would normally be passed to the link that are dropped because of filtering rules. MSTAT Counter Write Address = 0x1C
0x2181	Mid	
0x2182	High	
0x2184	Low	FilteredMulticastFrames The total number of Multicast frames that would normally be passed to the link that are dropped because of filtering rules. MSTAT Counter Write Address = 0x1D
0x2185	Mid	
0x2186	High	
0x2188	Low	FilteredBroadcastFrames The total number of Broadcast frames that would normally be passed to the link that are dropped because of filtering rules. MSTAT Counter Write Address = 0x1E
0x2189	Mid	
0x218A	High	
0x2190	Low	FramesTransmittedOK

MSTAT Counter Registers		
Read Address		
0x2191	Mid	Contains the count of frames that are successfully transmitted over the MAC interface. MSTAT Counter Write Address = 0x20
0x2192	High	
0x2194	Low	OctetsTransmittedOK Contains a count of data and padding (excluding preamble, SFD, destination/source address, length/type field, Q-Tag prefix, and FCS) octets of frames that are successfully transmitted over the MAC interface. ifOutOctets (MIB-II) can be computed using the following: ifOutOctets = OctetsTransmittedOK + (18 * FramesTransmittedOK) ifOutOctets includes the count of data, padding, destination/source address, length/type field, Q-Tag prefix, and FCS. (excludes preamble and SFD). MSTAT Counter Write Address = 0x21
0x2195	Mid	
0x2196	High	
0x2198	Low	OctetsTransmitted Contains a count of data and padding (excluding preamble, SFD, destination/source address, length/type field, Q-Tag prefix, and FCS) octets of frames that are attempted to be transmitted over the MAC interface. MSTAT Counter Write Address = 0x22
0x2199	Mid	
0x219A	High	
0x219C	Low	FramesLostDueToInternalMACTransmissionError Contains a count of frames that would otherwise be transmitted by the device, but because of a MAC FIFO underrun could not be sent correctly. If this counter is incremented, then none of the other error counters in this section are incremented. MSTAT Counter Write Address = 0x23
0x219D	Mid	
0x219E	High	
0x21A0	Low	TransmitSystemError Contains a count of frames that would otherwise be transmitted by the device, but could not be sent due to an indication from the POS-PHY Level 4 ERROR signal being asserted, an oversize frame being transmitted, or an internal CRC error discovered that was generated from the upstream device. If this counter is incremented, then none of the other error counters in this section are incremented. MSTAT Counter Write Address = 0x24
0x21A1	Mid	
0x21A2	High	
0x21A4	Low	UnicastFramesTransmittedAttempted Contains a count of frames that are requested to be transmitted to a group unicast destination address. This count includes those frames that
0x21A5	Mid	

MSTAT Counter Registers		
Read Address		
0x21A6	High	were discarded or not sent. MSTAT Counter Write Address = 0x25
0x21A8	Low	UnicastFramesTransmittedOK Contains a count of frames that are successfully transmitted via the MAC interface to a group unicast destination address. MSTAT Counter Write Address = 0x26
0x21A9	Mid	
0x21AA	High	
0x21AC	Low	MulticastFramesTransmittedAttempted Contains a count of frames that are requested to be transmitted to a group multicast destination address. This count includes those frames that were discarded or not sent. This count is not updated by broadcast frame transmission. MSTAT Counter Write Address = 0x27
0x21AD	Mid	
0x21AE	High	
0x21B0	Low	MulticastFramesTransmittedOK Contains a count of frames that are successfully transmitted to a group multicast destination. This count is not updated by broadcast frame transmission. MSTAT Counter Write Address = 0x28
0x21B1	Mid	
0x21B2	High	
0x21B4	Low	BroadcastFramesTransmittedAttempted Contains a count of the frames that were requested to be transmitted to a broadcast address. This count includes those frames that were discarded or not sent. This count is not updated by multicast frame transmission. MSTAT Counter Write Address = 0x29
0x21B5	Mid	
0x21B6	High	
0x21B8	Low	BroadcastFramesTransmittedOK Contains a count of the frames that were successfully transmitted to the broadcast address. This count is not updated by multicast frame transmission. MSTAT Counter Write Address = 0x2A
0x21B9	Mid	
0x21BA	High	

MSTAT Counter Registers		
Read Address		
0x21BC	Low	PAUSEMACCTRLFramesTransmitted Contains a count of PAUSE frames passed to the MAC Control sublayer for transmission. This counter is incremented when a request to send the PAUSE control frame is generated. The count only includes PAUSE MAC Control Frames that are generated by the Transmit MAC within the PM3392, i.e. IFLX FIFO almost full threshold is reached, the external PAUSE pin is asserted or the HOSTPAUSE bit is asserted in the MAC. This does not include any PAUSE MAC Control Frames coming across the PL4 interface. MSTAT Counter Write Address = 0x2B
0x21BD	Mid	
0x21BE	High	
0x21C0	Low	MACCTRLFramesTransmitted Contains a count of frames passed to the MAC sublayer for transmission. This counter is incremented when a control frame is transmitted out of the MAC. This does not include any MAC Control Frames coming across the PL4 interface. MSTAT Counter Write Address = 0x2C
0x21C1	Mid	
0x21C2	High	
0x21C4	Low	TransmittedFrames64Octets The total number of frames (including bad frames) transmitted that were 64 octets in length (excluding framing bits but including FCS octets). MSTAT Counter Write Address = 0x2D
0x21C5	Mid	
0x21C6	High	
0x21C8	Low	TransmittedFrames65to127Octets The total number of frames (including bad frames) transmitted that were between 65 and 127 octets in length inclusive (excluding framing bits but including FCS octets). MSTAT Counter Write Address = 0x2E
0x21C9	Mid	
0x21CA	High	
0x21CC	Low	TransmittedFrames128to255Octets The total number of frames (including bad frames) transmitted that were between 128 and 255 octets in length inclusive (excluding framing bits but including FCS octets). MSTAT Counter Write Address = 0x2F
0x21CD	Mid	
0x21CE	High	
0x21D0	Low	TransmittedFrames256to511Octets The total number of frames (including bad frames) transmitted that were between 256 and 511 octets in length inclusive (excluding framing bits but including FCS octets). MSTAT Counter Write Address = 0x30
0x21D1	Mid	
0x21D2	High	

MSTAT Counter Registers		
Read Address		
0x21D4	Low	TransmittedFrames512to1023Octets The total number of frames (including bad frames) transmitted that were between 512 and 1023 octets in length inclusive (excluding framing bits but including FCS octets). MSTAT Counter Write Address = 0x31
0x21D5	Mid	
0x21D6	High	
0x21D8	Low	TransmittedFrames1024to1518Octets The total number of frames (including bad frames) transmitted that were between 1024 and (1518 octets for untagged frames and 1522 octets for VLAN tagged frames) in length inclusive (excluding framing bits but including FCS octets). MSTAT Counter Write Address = 0x32
0x21D9	Mid	
0x21DA	High	
0x21DC	Low	TransmittedFrames1519toMAXOctets The total number of frames (including bad frames) transmitted that were between the normal maximum length (1518 octets for un-tagged frames and 1522 octets for tagged frames) and the max Jumbo frame length (i.e. up to MaxFrameSize) (excluding framing bits but including FCS octets). MSTAT Counter Write Address = 0x33
0x21DD	Mid	
0x21DE	High	
0x21E0	Low	JumboOctetsTransmittedOK The total number of octets (excluding bad frames) transmitted that were between the normal maximum length (1518 octets for un-tagged frames and 1522 octets for tagged frames) and the max Jumbo frame length (i.e. up to MaxFrameSize) (excluding framing bits but including FCS octets). MSTAT Counter Write Address = 0x34
0x21E1	Mid	
0x21E2	High	

Register 0x2200H:IFLX Global Configuration Register

Bit	Type	Function	Default
Bit 15	R/W	IRCU_ENABLE	0
Bit 14	R/W	IDSWT_ENABLE	0
Bit 13	R/W	Reserved	0
Bit 12	R/W	Reserved	0
Bit 11	R/W	Reserved	0
Bit 10	R/W	Reserved	0
Bit 9	R/W	Reserved	0
Bit 8	R/W	Reserved	0
Bit 7	R/W	Reserved	0
Bit 6	R/W	Reserved	0
Bit 5	R/W	Reserved	0
Bit 4	R/W	Reserved	0
Bit 3	R/W	Reserved	0
Bit 2	R/W	Reserved	0
Bit 1	R/W	Reserved	0
Bit 0	R/W	Reserved	0

IRCU_ENABLE

The ENABLE bit must be set to enable the operation of the IRCU block. Clearing this bit causes the entire IRCU block to freeze and stop processing requests from either the write datapath or the system scheduler.

IDSWT_ENABLE

The ENABLE bit must be set to enable the operation of the IDSWT block. This bit must be set to a logic ‘1’ at all times.

Register 0x2201H:IFLX Channel Provision

Bit	Type	Function	Default
Bit 15	R/W	Reserved	0
Bit 14	R/W	Reserved	0
Bit 13	R/W	Reserved	0
Bit 12	R/W	Reserved	0
Bit 11	R/W	Reserved	0
Bit 10	R/W	Reserved	0
Bit 9	R/W	Reserved	0
Bit 8	R/W	Reserved	0
Bit 7	R/W	Reserved	0
Bit 6	R/W	Reserved	0
Bit 5	R/W	Reserved	0
Bit 4	R/W	Reserved	0
Bit 3	R/W	Reserved	0
Bit 2	R/W	Reserved	0
Bit 1	R/W	Reserved	0
Bit 0	R/W	PROV[0]	0

PROV

The channel provision bit (PROV) specifies the active status of the channel being accessed. When PROV is asserted high, the channel is active. When PROV is asserted low, the channel is inactive and is not used. The PROV bit is used for enabling and disabling the single channels within the IFLX, as well as to initialize the data and tag RAM read/write addresses. A transition from the channel being active (PROV is logic 1) to inactive (PROV is logic 0) will reset the rate adaptation buffer, the frame buffer and freeze the logical FIFO. A transition from the channel being inactive (PROV is logic 0) to active (PROV is logic 1) will reset the logical FIFO read/write pointers to an empty state and begin to process data independent of other provisioned channels. A zero value in this register will disable the channel, rendering it inactive. **In the case of the PM3392 the only valid value is 0x0001h.**

Register 0x2209H:IFLX FIFO Overflow Enable

Bit	Type	Function	Default
Bit 15	R/W	Reserved	0
Bit 14	R/W	Reserved	0
Bit 13	R/W	Reserved	0
Bit 12	R/W	Reserved	0
Bit 11	R/W	Reserved	0
Bit 10	R/W	Reserved	0
Bit 9	R/W	Reserved	0
Bit 8	R/W	Reserved	0
Bit 7	R/W	Reserved	0
Bit 6	R/W	Reserved	0
Bit 5	R/W	Reserved	0
Bit 4	R/W	Reserved	0
Bit 3	R/W	Reserved	0
Bit 2	R/W	Reserved	0
Bit 1	R/W	Reserved	0
Bit 0	R/W	Reserved	0

Register 0x220AH:IFLX FIFO Overflow Interrupt

Bit	Type	Function	Default
Bit 15	R	Reserved	0
Bit 14	R	Reserved	0
Bit 13	R	Reserved	0
Bit 12	R	Reserved	0
Bit 11	R	Reserved	0
Bit 10	R	Reserved	0
Bit 9	R	Reserved	0
Bit 8	R	Reserved	0
Bit 7	R	Reserved	0
Bit 6	R	Reserved	0
Bit 5	R	Reserved	0
Bit 4	R	Reserved	0
Bit 3	R	Reserved	0
Bit 2	R	Reserved	0
Bit 1	R	Reserved	0
Bit 0	R	Reserved	0

Register 0x220DH:IFLX Indirect Channel Address

Bit	Type	Function	Default
Bit 15	R	BUSY	0
Bit 14	R/W	RWB	1
Bit 13	R	Unused	0
Bit 12	R	Unused	0
Bit 11	R	Unused	0
Bit 10	R	Unused	0
Bit 9	R	Unused	0
Bit 8	R	Unused	0
Bit 7	R	Unused	0
Bit 6	R	Unused	0
Bit 5	R	Unused	0
Bit 4	R	Unused	0
Bit 3	R/W	Reserved	0
Bit 2	R/W	Reserved	0
Bit 1	R/W	Reserved	0
Bit 0	R/W	Reserved	0

RWB

The read/write bar (RWB) bit selects between an update operation (write) and a query operation (read). Writing logic 0 to RWB triggers the update operation with the information in the IFLX Indirect Registers (0x220E to 0x2217). Writing a logic 1 to RWB triggers a query and the information is placed in all of the IFLX Indirect Registers.

BUSY

The indirect access status bit (BUSY) reports the progress of an indirect access. BUSY is set high when a write to the Indirect Channel Select register triggers an indirect access and will stay high until the access is complete. This register should be polled to determine when data from an indirect read operation is available in all the Indirect Data registers or to determine when a new indirect write operation may commence.

Register 0x220EH:IFLX Indirect Logical FIFO Low Limit & Provision

Bit	Type	Function	Default
Bit 15	R	Unused	0
Bit 14	R	Unused	0
Bit 13	R	Unused	0
Bit 12	R	Unused	0
Bit 11	R	Unused	0
Bit 10	R	Unused	0
Bit 9	R/W	LOLIM[9]	0
Bit 8	R/W	LOLIM[8]	0
Bit 7	R/W	LOLIM[7]	0
Bit 6	R/W	LOLIM[6]	0
Bit 5	R/W	LOLIM[5]	0
Bit 4	R/W	LOLIM[4]	0
Bit 3	R/W	LOLIM[3]	0
Bit 2	R/W	LOLIM[2]	0
Bit 1	R/W	LOLIM[1]	0
Bit 0	R/W	LOLIM[0]	0

LOLIM[9:0]

The lower address boundary (LOLIM[9:0]) specifies the lower address limit in the ring buffer for the logical FIFO defined in the Indirect Channel Address Register. The address value specified is in units of 256 bytes, or 16 128-bit words, and must point to the first byte of the first location in the RAM that belongs to the FIFO. For example, to set up a logical FIFO with 4096 bytes of space lying between locations 256 and 511 128-bit words (inclusive) in the RAM, the LOLIM[9:0] field must be set to 16 decimal. This value should be programmed to zero.

Register 0x220FH:IFLX Indirect Logical FIFO High Limit

Bit	Type	Function	Default
Bit 15	R	Unused	0
Bit 14	R	Unused	0
Bit 13	R	Unused	0
Bit 12	R	Unused	0
Bit 11	R	Unused	0
Bit 10	R	Unused	0
Bit 9	R/W	HILIM[9]	0
Bit 8	R/W	HILIM[8]	0
Bit 7	R/W	HILIM[7]	0
Bit 6	R/W	HILIM[6]	0
Bit 5	R/W	HILIM[5]	0
Bit 4	R/W	HILIM[4]	0
Bit 3	R/W	HILIM[3]	0
Bit 2	R/W	HILIM[2]	0
Bit 1	R/W	HILIM[1]	0
Bit 0	R/W	HILIM[0]	0

HILIM[9:0]

The upper address boundary, (HILIM[9:0]) specifies the upper address limit in the ring buffer for the logical FIFO defined in the Indirect Channel Address Register. HILIM[9:0] is specified in units of 256 bytes, or 16 128-bit words, and must point to the first byte *after* the last RAM location for the FIFO. For example, to set up a logical FIFO with 4096 bytes of space lying between locations 256 and 511 128-bit words (inclusive) in the RAM, the HILIM[9:0] field must be set to 32 decimal.

Register 0x2210H:IFLX Indirect Full/Almost Full Status & Limit

Bit	Type	Function	Default
Bit 15	R	FULL	0
Bit 14	R	AFULL	0
Bit 13	R/W	AFTH[13]	0
Bit 12	R/W	AFTH[12]	0
Bit 11	R/W	AFTH[11]	0
Bit 10	R/W	AFTH[10]	0
Bit 9	R/W	AFTH[9]	0
Bit 8	R/W	AFTH[8]	0
Bit 7	R/W	AFTH[7]	0
Bit 6	R/W	AFTH[6]	0
Bit 5	R/W	AFTH[5]	0
Bit 4	R/W	AFTH[4]	0
Bit 3	R/W	AFTH[3]	0
Bit 2	R/W	AFTH[2]	0
Bit 1	R/W	AFTH[1]	0
Bit 0	R/W	AFTH[0]	0

AFTH [13:0]

The almost full threshold field (AFTH[13:0]) is the number of 128-bit words that can be held in the logical FIFO before the AFULL signal is asserted for the specified channel in the Indirect Channel Address Register. The AFULL signal is reported to the upstream datapath through the Line side Interface. When the number of 128-bit words held in this FIFO (as measured by the word counter) is equal to or greater than AFTH[13:0], the AFULL output from the asserted, otherwise AFULL is de-asserted. It is recommended that AFTH[13:0] be written only at initialization time.

AFULL

The almost full status bit (AFULL) is set when the number of words within the logical FIFO is greater than the AFTH[13:0] value. This status information is provided for diagnostic purposes.

FULL

The full status bit (FULL) is set when the logical FIFO is reporting a full status to the upstream write datapath. This status information is provided for diagnostic purposes.

Register 0x2211H:IFLX Indirect Empty/Almost Empty Status & Limit

Bit	Type	Function	Default
Bit 15	R	EMPTY	1
Bit 14	R	AEMPTY	1
Bit 13	R/W	AETH[13]	0
Bit 12	R/W	AETH[12]	0
Bit 11	R/W	AETH[11]	0
Bit 10	R/W	AETH[10]	0
Bit 9	R/W	AETH[9]	0
Bit 8	R/W	AETH[8]	0
Bit 7	R/W	AETH[7]	0
Bit 6	R/W	AETH[6]	0
Bit 5	R/W	AETH[5]	0
Bit 4	R/W	AETH[4]	0
Bit 3	R/W	AETH[3]	0
Bit 2	R/W	AETH[2]	0
Bit 1	R/W	AETH[1]	0
Bit 0	R/W	AETH[0]	0

AETH [13:0]

The almost empty threshold field (AETH[13:0]) is the number of 128-bit words that are held in the logical FIFO before the AFULL signal is de-asserted. The AEMPTY signal is reported to the upstream datapath through the Line side Interface. When the number of 128-bit words held in this FIFO (as measured by the word counter) is equal to or less than AETH[13:0], the AEMPTY output is asserted, otherwise AEMPTY is de-asserted. It is recommended that AETH[13:0] be written only at initialization time.

AEMPTY

The almost empty status bit (AEMPTY) is set when the number of words within the logical FIFO is less than or equal to the AETH[13:0] value. This status information is provided for diagnostic purposes.

EMPTY

The empty status bit (EMPTY) is set whenever the logical FIFO is completely empty. This flag is provided for diagnostic purposes.

Register 0x2240H:PL4MOS Configuration Register

Bit	Type	Function	Default
Bit 15	R	Unused	X
Bit 14	R	Unused	X
Bit 13	R	Unused	X
Bit 12	R	Unused	X
Bit 11	R	Unused	X
Bit 10	R	Unused	X
Bit 9	R	Unused	X
Bit 8	R	Unused	X
Bit 7	R	Unused	X
Bit 6	R	Unused	X
Bit 5	R	Unused	X
Bit 4	R	Unused	X
Bit 3	R/W	RE_INIT	0
Bit 2	R/W	PL4MOS_EN	0
Bit 1	R/W	NO_STATUS	0
Bit 0	R/W	RESERVE	1

NO_STATUS

The NO_STATUS bit is set to 1 to indicate that status information is not available. In this case, infinite credit is assigned. When NO_STATUS bit is set to 0, the PL4 status information is available and the PL4MOS may use it to implement the credit based scheduling algorithm. NO_STATUS bit should be updated before enabling the PL4MOS (PL4MOS_EN).

NOTE: If the NO_STATUS bit is set to 1 then for proper operation the PL4IO NO_ISTAT bit, in the PL4IO Configuration Register 0x2305H, must also be set to 1. If not then the MAC's can overflow their respective FIFO's due to improper calendar operation.

PL4MOS_EN

When the PL4MOS_EN bit is set low, the PL4MOS is disabled and no traffic can be passed. When the PL4MOS_EN bit is set high, the PL4MOS operates in normal mode. Changing this bit does not affect the internal status or counters of the PL4MOS. This bit should be set high after the IFLX and EFLX have been initialized. If the upstream logic has to be reconfigured during normal operation, the PL4MOS_EN bit should be low and the RE_INIT bit should be high to clear all the internal counters and previous status. For diagnostic purposes, this bit should be set low to disable the PL4MOS before reading the internal status registers. This guarantees the consistency of the data read.

RE_INIT

RE_INIT bit is used to clear all the internal status registers. It does not affect the configuration registers. When the RE_INIT bit is set high, all the internal counters and status calendar are reset to 0. When it is set low, the PL4MOS operates in normal mode.

Register 0x2241H:PL4MOS Reserved

Bit	Type	Function	Default
Bit 15	R/W	Reserved	0
Bit 14	R/W	Reserved	0
Bit 13	R/W	Reserved	0
Bit 12	R/W	Reserved	0
Bit 11	R/W	Reserved	0
Bit 10	R/W	Reserved	0
Bit 9	R/W	Reserved	0
Bit 8	R/W	Reserved	0
Bit 7	R/W	Reserved	0
Bit 6	R/W	Reserved	0
Bit 5	R/W	Reserved	0
Bit 4	R/W	Reserved	0
Bit 3	R/W	Reserved	0
Bit 2	R/W	Reserved	0
Bit 1	R/W	Reserved	0
Bit 0	R/W	Reserved	0

Reserved

This register should be programmed to zero at all times.

Register 0x2242H:PL4MOS Fairness Mask

Bit	Type	Function	Default
Bit 15	R/W	Reserved	1
Bit 14	R/W	Reserved	1
Bit 13	R/W	Reserved	1
Bit 12	R/W	Reserved	1
Bit 11	R/W	Reserved	1
Bit 10	R/W	Reserved	1
Bit 9	R/W	Reserved	1
Bit 8	R/W	Reserved	1
Bit 7	R/W	Reserved	1
Bit 6	R/W	Reserved	1
Bit 5	R/W	Reserved	1
Bit 4	R/W	Reserved	1
Bit 3	R/W	Reserved	1
Bit 2	R/W	Reserved	1
Bit 1	R/W	Reserved	1
Bit 0	R/W	Reserved	1

PL4MOS FAIRNESS MASK REGISTER

This register should be programmed to zero at all times.

Register 0x2243H:PL4MOS MaxBurst1 Register

Bit	Type	Function	Default
Bit 15	R	Unused	X
Bit 14	R	Unused	X
Bit 13	R	Unused	X
Bit 12	R	Unused	X
Bit 11	R/W	MAX_BURST1[11]	0
Bit 10	R/W	MAX_BURST1[10]	0
Bit 9	R/W	MAX_BURST1[9]	0
Bit 8	R/W	MAX_BURST1[8]	0
Bit 7	R/W	MAX_BURST1[7]	0
Bit 6	R/W	MAX_BURST1[6]	0
Bit 5	R/W	MAX_BURST1[5]	0
Bit 4	R/W	MAX_BURST1[4]	0
Bit 3	R/W	MAX_BURST1[3]	1
Bit 2	R/W	MAX_BURST1[2]	0
Bit 1	R	MAX_BURST1[1]	0
Bit 0	R	MAX_BURST1[0]	0

MAX_BURST1

MAX_BURST1 defines the amount of credit, in 128-bit words, granted for a starving channel. This value depends on the system latency and architecture. Note that the lower 2 bits are forced to zero in order to constrain MAX_BURST1 to modulo 4 values. The default value is 8 words; i.e. 128 bytes. PL4MOS_EN bit should be deasserted before writing to this register.

Register 0x2244H:PL4MOS MaxBurst2 Register

Bit	Type	Function	Default
Bit 15	R	Unused	X
Bit 14	R	Unused	X
Bit 13	R	Unused	X
Bit 12	R	Unused	X
Bit 11	R/W	MAX_BURST2[11]	0
Bit 10	R/W	MAX_BURST2[10]	0
Bit 9	R/W	MAX_BURST2[9]	0
Bit 8	R/W	MAX_BURST2[8]	0
Bit 7	R/W	MAX_BURST2[7]	0
Bit 6	R/W	MAX_BURST2[6]	0
Bit 5	R/W	MAX_BURST2[5]	0
Bit 4	R/W	MAX_BURST2[4]	0
Bit 3	R/W	MAX_BURST2[3]	
Bit 2	R/W	MAX_BURST2[2]	0
Bit 1	R	MAX_BURST2[1]	0
Bit 0	R	MAX_BURST2[0]	0

MAX_BURST2

MAX_BURST2 defines the amount of credit, in 128-bit words, granted for a hungry channel. This value depends on the system latency and architecture. Note that the lower 2 bits are forced to zero in order to constrain MAX_BURST2 to modulo 4 values. The default value is 8 words; i.e. 128 bytes. PL4MOS_EN bit should be deasserted before writing to this register.

Register 0x2245H:PL4MOS Transfer Size Register

Bit	Type	Function	Default
Bit 15	R	Unused	X
Bit 14	R	Unused	X
Bit 13	R	Unused	X
Bit 12	R	Unused	X
Bit 11	R	Unused	X
Bit 10	R	Unused	X
Bit 9	R	Unused	X
Bit 8	R	Unused	X
Bit 7	R/W	MAX_TRANSFER[7]	0
Bit 6	R/W	MAX_TRANSFER[6]	0
Bit 5	R/W	MAX_TRANSFER[5]	0
Bit 4	R/W	MAX_TRANSFER[4]	0
Bit 3	R/W	MAX_TRANSFER[3]	1
Bit 2	R/W	MAX_TRANSFER[2]	0
Bit 1	R	MAX_TRANSFER[1]	0
Bit 0	R	MAX_TRANSFER[0]	0

MAX_TRANSFER

MAX_TRANSFER defines the maximum number of 128-bit words that may be transferred during one service request. Note that the lower 2 bits are forced to zero in order to constrain MAX_TRANSFER to modulo 4 values. PL4MOS_EN bit should be deasserted before writing to this register.

Register 0x2280H:PL4ODP Configuration

Bit	Type	Function	Default
Bit 15	R/W	REPEAT_T[3]	0
Bit 14	R/W	REPEAT_T[2]	0
Bit 13	R/W	REPEAT_T[1]	0
Bit 12	R/W	REPEAT_T[0]	0
Bit 11	R	Unused	X
Bit 10	R	Unused	X
Bit 9	R	Unused	X
Bit 8	R/W	SOP_RULE	1
Bit 7	R/W	Reserved	0
Bit 6	R/W	Reserved	0
Bit 5	R/W	Reserved	0
Bit 4	R	Unused	X
Bit 3	R	Unused	X
Bit 2	R	Reserved	0
Bit 1	R/W	EN_PORTS	0
Bit 0	R/W	EN_DFWD	0

EN_DFWD

Enable Data Forward. When ‘1’, data will be forwarded. When ‘0’, data will not be forwarded. Modifying this register bit during packet data reception can result in the creation of partial packets

EN_PORTS

Enable Port. When ‘1’, data will be forwarded. When ‘0’, data will not be forwarded by the PL4ODP. This bit only takes effect on packet boundaries. No partial packets can be created.

SOP_RULE

The value of SOP_RULE determines the minimum number of PL4 Bus cycles between successive Start-Of-Packet control words.

SOP_RULE	Minimum SOP spacing (in terms of PL4 Bus cycles)
0	2
1	8

REPEAT_T[3:0]

REPEAT_T is the PL4 Bus configuration parameter defining how many back-to-back training patterns define a training sequence. The value of this input is actually the number of training patterns plus one: a value of REPEAT_T of 0x0 means that a single 20-cycle (PL4 Bus cycle) training pattern defines a training sequence; a value of 0xF for REPEAT_T implies that the training sequence will be 320-cycles, with the 20-cycle training pattern repeated 16 times consecutively.

Register 0x2282H:PL4ODP Interrupt Mask

Bit	Type	Function	Default
Bit 15	R	Unused	X:high
Bit 14	R	Unused	X:high
Bit 13	R	Unused	X:high
Bit 12	R	Unused	X:high
Bit 11	R	Unused	X:high
Bit 10	R	Unused	X:high
Bit 9	R	Unused	X:high
Bit 8	R	Unused	X:high
Bit 7	R/W		0
Bit 6	R/W		0
Bit 5	R/W	Reserved	0
Bit 4	R/W	Reserved	0
Bit 3	R/W		0
Bit 2	R/W		0
Bit 1	R/W		0
Bit 0	R/W	OUT_DISE	0

OUT_DISE

The OUT_DISE bit enables the generation of an interrupt due to the primary input signal OUT_DIS being deasserted when the Link State is PL4ODP_FWD.

Register 0x2283H:PL4ODP Interrupt

Bit	Type	Function	Default
Bit 15	R	Unused	X:high
Bit 14	R	Unused	X:high
Bit 13	R	Unused	X:high
Bit 12	R	Unused	X:high
Bit 11	R	Unused	X:high
Bit 10	R	Unused	X:high
Bit 9	R	Unused	X:high
Bit 8	R	Unused	X:high
Bit 7	R		0
Bit 6	R		0
Bit 5	R	Reserved	0
Bit 4	R	Reserved	0
Bit 3	R		0
Bit 2	R		0
Bit 1	R		0
Bit 0	R	OUT_DISI	0

OUT_DISI

The OUT_DISI bit will be set when the primary input signal OUT_DIS is deasserted when the Link State is PL4ODP_FWD. If OUT_DISE is '1' in the PL4ODP Interrupt Mask register, an interrupt will also be generated (INT output asserted).

Register 0x2284H:PL4ODP Configuration MAX_T Register

Bit	Type	Function	Default
Bit 15	R/W	MAX_T[15]	0
Bit 14	R/W	MAX_T[14]	0
Bit 13	R/W	MAX_T[13]	0
Bit 12	R/W	MAX_T[12]	0
Bit 11	R/W	MAX_T[11]	0
Bit 10	R/W	MAX_T[10]	0
Bit 9	R/W	MAX_T[9]	0
Bit 8	R/W	MAX_T[8]	0
Bit 7	R/W	MAX_T[7]	0
Bit 6	R/W	MAX_T[6]	0
Bit 5	R/W	MAX_T[5]	0
Bit 4	R/W	MAX_T[4]	1
Bit 3	R/W	MAX_T[3]	0
Bit 2	R/W	MAX_T[2]	0
Bit 1	R/W	MAX_T[1]	0
Bit 0	R/W	MAX_T[0]	0

MAX_T[15:0]

MAX_T defines the bounded interval in time over which the PL4 training sequence is to be sent. The time value of MAX_T is given in terms of 16 internal CLK cycles. MAX_T setting of 0x000 has a special meaning

NOTES: The maximum value in time defined by MAX_T for a PL4 cycle frequency of 833 MHz is: (2**16)*(16 CLK cycles)*(4/833 MHz per CLK cycle) = 5.0 mS. For a 700 MHz clock this is about 6 ms.

A value of 0x000 for MAX_T will result in disabling the timer associated with the sending of training patterns. When the MAX_T register is set to 0x000, and the OUT_DIS input is deasserted, no training patterns will be sent from the PL4ODP. If OUT_DIS is asserted, the PL4ODP transmits idle patterns regardless of the value of MAX_T.

Default value of 16 in MAX_T for a PL4 cycle frequency of 833 MHz means that a training sequence will occur every (16)*(16 CLK cycles)*(4/833 MHz per CLK cycle) = 1.23 uS. For a 700 MHz clock this is 1.46 us.

The PM3392 waits until the current data transfer is complete before transmitting a training sequence when the MAX_T timer expires. This may result in intervals greater than MAX_T between successive training sequences when data throughput is high. Furthermore, it may result in an average interval between training sequences greater than MAX_T over a long period of time. If a system has a minimum value for this interval that is critical for correct operation, then MAX_T should be set lower than the minimum value. The required setting depends on data transfer size and PL4 ingress path bandwidth utilization. A guardband of 5% should be sufficient

Register 0x2300H:PL4IO Lock Detect Status

Bits	Type	Function	Default
15	R	OUT_ROOL	1
14	R	Unused	X
13	R	Unused	X
12	R	IS_ROOL	1
11	R	DIP2_ERR	X
10	R	Unused	X
9	R	Unused	X
8	R	ID_ROOL	1
7	R	Unused	X
6	R	Unused	X
5	R	Unused	X
4	R	IS_DOOL	1
3	R	Unused	X
2	R	Unused	X
1	R	Unused	X
0	R	ID_DOOL	1

The current Reference Out Of Lock (ROOL) and Data Out Of Lock (DOOL) conditions for the PL4 interface.

ID_DOOL

Depending on the setting of the DLSEL bit in the PL4IO Configuration register, either the input data FIFOs are not aligned to the incoming data stream (in normal operation) or a valid PRBS pattern is not being received in the input data FIFOs (in serial data link testing.)

IS_DOOL

The parallel FIFO status input stream is not properly synchronized.

DIP2_ERR

A DIP2 error occurred on the FIFO status input stream.

ID_ROOL

The parallel data input clock (TDCLK) or one or more of the parallel data input streams is not trained to the local synthesized clock. The PL4 data input interface is normally disabled when ID_ROOL is asserted.

IS_ROOL

The input FIFO status clock (RSCLK) frequency exceeds ¼ the frequency of RDCLK. The PL4 input status interface is normally disabled when IS_ROOL is asserted.

OUT_ROOL

The local synthesized clock is not trained to the reference frequency. The PL4 data and status output interfaces are normally disabled when OUT_ROOL is asserted.

All of the PL4IO interfaces share a single clock synthesizer.

Register 0x2301H:PL4IO Lock Detect Change

Bits	Type	Function	Default
15	R	OUT_ROOLI	0
14	R	Unused	X
13	R	Unused	X
12	R	IS_ROOLI	0
11	R	DIP2_ERRI	X
10	R	Unused	X
9	R	Unused	X
8	R	ID_ROOLI	0
7	R	Unused	X
6	R	Unused	X
5	R	Unused	X
4	R	IS_DOOLI	0
3	R	Unused	X
2	R	Unused	X
1	R	Unused	X
0	R	ID_DOOLI	0

Indicates whether any of the Reference Out Of Lock (ROOL) or Data Out Of Lock (DOOL) conditions on the PL4 interface have changed since the previous ECBI read from the Lock Detect Change register. An interrupt request (ROOL_INT or DOOL_INT) will be asserted when any pair of corresponding bits in the Lock Detect Change and Lock Detect Mask registers are both set to logic "1".

ID_DOOLI

The Input Data Out Of Lock condition has changed.

ID_DOOLI is set to logic "1" when the value of the ID_DOOL condition changes.

IS_DOOLI

The Input Status Out Of Lock condition has changed.

IS_DOOLI is set to logic "1" when the value of the IS_DOOL condition changes.

ID_ROOLI

The Input Data Reference Out Of Lock condition has changed.

ID_ROOLI is set to logic "1" when the value of the ID_ROOL condition changes.

DIP2_ERRI

The DIP2_ERRI status condition has changed.

DIP2_ERRI is set to logic "1" when the value of the DIP2_ERR condition changes.

IS_ROOLI

The Input Status Reference Out Of Lock condition has changed.

IS_ROOLI is set to logic "1" when the value of the IS_ROOL condition changes.

OUT_ROOLI

The Output Reference Out Of Lock condition has changed.

OUT_ROOLI is set to logic "1" when the value of the OUT_ROOL condition changes.

Register 0x2302H:PL4IO Lock Detect Mask

Bits	Type	Function	Default
15	R/W	OUT_ROOLE	0
14	R	Unused	X
13	R	Unused	X
12	R	Unused	X
12	R/W	IS_ROOL	0
11	R	DIP2_ERRE	X
10	R	Unused	X
9	R	Unused	X
8	R/W	ID_ROOLE	0
7	R	Unused	X
6	R	Unused	X
5	R	Unused	X
4	R/W	IS_DOOLE	0
3	R	Unused	X
2	R	Unused	X
1	R	Unused	X
0	R/W	ID_DOOLE	0

Arms the PL4IO interrupt requests (ROOL_INT and DOOL_INT) when any pair of corresponding bits in the Lock Detect Change and Lock Detect Mask registers are both set to logic "1".

ID_DOOLE

Enables the triggering of DOOL_INT. The DOOL_INT signal is asserted when the ID_DOOL bits in the Lock Detect Change and Lock Detect Mask registers are both set to logic "1".

IS_DOOLE

Enables the triggering of DOOL_INT. The DOOL_INT signal is asserted when the IS_DOOL bits in the Lock Detect Change and Lock Detect Mask registers are both set to logic "1".

ID_ROOLE

Enables the triggering of ROOL_INT. The ROOL_INT signal is asserted when the ID_ROOL bits in the Lock Detect Change and Lock Detect Mask registers are both set to logic "1".

DIP2_ERRE

Enables the triggering of DIP2_ERR interrupt. The DIP2_ERR interrupt signal is asserted when the DIP2_ERR bits in the Lock Detect Change and Lock Detect Mask registers are both set to logic "1".

IS_ROOLE

Enables the triggering of ROOL_INT. The ROOL_INT signal is asserted when the IS_ROOL bits in the Lock Detect Change and Lock Detect Mask registers are both set to logic "1".

OUT_ROOLE

Enables the triggering of ROOL_INT. The ROOL_INT signal is asserted when the OUT_ROOL bits in the Lock Detect Change and Lock Detect Mask registers are both set to logic "1".

Register 0x2303H:PL4IO Lock Detect Limits

Bits	Type	Function	Default
15	R/W	REF_LIMIT[7]	0
14	R/W	REF_LIMIT[6]	0
13	R/W	REF_LIMIT[5]	0
12	R/W	REF_LIMIT[4]	0
11	R/W	REF_LIMIT[3]	0
10	R/W	REF_LIMIT[2]	0
9	R/W	REF_LIMIT[1]	0
8	R/W	REF_LIMIT[0]	1
7	R/W	TRAN_LIMIT[7]	0
6	R/W	TRAN_LIMIT[6]	0
5	R/W	TRAN_LIMIT[5]	1
4	R/W	TRAN_LIMIT[4]	1
3	R/W	TRAN_LIMIT[3]	1
2	R/W	TRAN_LIMIT[2]	1
1	R/W	TRAN_LIMIT[1]	1
0	R/W	TRAN_LIMIT[0]	1

Specifies the permitted range of measured clock frequency deviation, relative to the selected reference clock.

REF_LIMIT[7:0]

The maximum permitted deviation for any clock derived from the reference clock, in multiples of 30 ppm (nominal - 30.773 ppm actual).

REF_LIMIT applies to the CSU and to the input data clock (IDCLK).

REF_LIMIT is set to logic "0000_0001" (30 ppm nominal) when DRSTB is asserted.

TRAN_LIMIT[7:0]

The minimum number of input data transition events (in multiples of 16) required to synchronize the DRU to the incoming data stream.

TRAN_LIMIT applies to the DRUs.

TRAN_LIMIT is set to logic "0011_1111" (1024 transitions) when DRSTB is asserted.

Register 0x2304H:PL4IO Calendar Repetitions

Bits	Type	Function	Default
15	R/W	IN_MUL[7]	0
14	R/W	IN_MUL[6]	0
13	R/W	IN_MUL[5]	0
12	R/W	IN_MUL[4]	0
11	R/W	IN_MUL[3]	0
10	R/W	IN_MUL[2]	0
9	R/W	IN_MUL[1]	0
8	R/W	IN_MUL[0]	0
7	R/W	OUT_MUL[7]	0
6	R/W	OUT_MUL[6]	0
5	R/W	OUT_MUL[5]	0
4	R/W	OUT_MUL[4]	0
3	R/W	OUT_MUL[3]	0
2	R/W	OUT_MUL[2]	0
1	R/W	OUT_MUL[1]	0
0	R/W	OUT_MUL[0]	0

Specifies the number of repetitions in the FIFO status calendars.

IN_MUL[7:0]

The number of repetitions of the input FIFO status channel within a complete status calendar sequence. The PL4 input status parameter CALENDAR_M = IN_MUL + 1. The total length of the input status calendar sequence = CALENDAR_LEN x CALENDAR_M.

IN_MUL should only be changed while the input status interface is disabled in the Configuration register. The result of changing IN_MUL while the status interface is enabled is unspecified.

OUT_MUL[7:0]

The number of repetitions of the output FIFO status channels within a complete status calendar sequence. The PL4 output status parameter CALENDAR_M = OUT_MUL + 1. The total length of the output status calendar sequence = CALENDAR_LEN x CALENDAR_M.

OUT_MUL should only be changed while the output status interface is disabled in the Configuration register. The result of changing OUT_MUL while the status interface is enabled is unspecified.

Register 0x2305H:PL4IO Configuration

Bits	Type	Function	Default
15	R/W	DIP2ERR_CHK	0
14	R/W	RESET	0
13	R/W	IDDQ	0
12	R/W	ENABLE	1
11	R/W	ODAT_DIS	X
10	R/W	TRAIN_DIS	X
9	R/W	OSTAT_DIS	1
8	R/W	ISTAT_DIS	1
7	R/W	NO_ISTAT	0
6	R/W	STAT_OUTSEL	X
5	R/W	INSEL	0
4	R/W	DLSEL	0
3		Reserved	X
2		Reserved	X
1	R/W	OUTSEL[1]	0
0	R/W	OUTSEL[0]	1

Specifies the requested configuration of the PL4IO interface. In normal operation the PL4IO control logic will sequence the internal PL4IO components toward this configuration.

OUTSEL[1:0]

Selects the source of the 68 bit parallel data stream for the PISOs.

00 – None
01 – PL4ODP output data stream (OUTD[67:0])
— normal mode
10 – PRBS generate data stream
– used for PL4 data link testing
11 – FIFO read data stream enables PL4 system-side (local) loopback. Data is looped-back from the device TCTL and TDAT[15:0] input pins to the RCTL and RDAT[15:0] output pins.

OUTSEL is set to logic "01" when DRSTB is asserted.

DLSEL

Selects the source for the Input Data Out Of Lock (ID_DOOL) condition.

0 – The input data FIFOs are not aligned to the incoming data stream
– used for normal parallel data link operation
1 – A valid PRBS pattern is not being received in the input data FIFOs
– used for serial data link testing

INSEL

Selects the source of the 68 bit parallel data stream for the Unpacker (IND[67:0]), together with related handshake signals.

0 – DRU input data stream
1 – Packer output data stream (OUTD[67:0])
– enables device-side (local) loopback
– The following connections are looped back:
– OUT_DIS is driven from IN_DIS
– INS is driven from OUTS
– IND_VALID is driven from OUTD_VALID
– IND is driven from OUTD

If PL4IO INSEL is logic 1 (remote loopback from PL4ODP to PL4IDU) and PL4IO OUTSEL[1:0] is b01 (normal operation) data will be impressed on the PL4 RDAT[15:0], RCTL pins. If the user would like to disable the transfer of data on the PL4 RDAT[15:0],RCTL interface pins then the following can be done:

STEP_1: program PL4IO OUTSEL[1:0] to b00. This will result in the RDAT[15:0],RCTL pins being driven to a logic 0.

STEP_2: PL4IO INSEL can be programmed to logic 1 to configure the PL4 interface for remote loopback.

NOTE:

When INSEL is set to 1 the PL4 interface must be in Master mode and have a valid reference clock for the loopback operation to work properly. The PL4IO lock status logic continues to check the lock status on the device's PL4 interface pins. This will not effect the remote loopback operation even if the PL4IO reports IS/ID ROOL/DOOL out-of-lock, it means the PL4 partner is not configured or present.

STAT_OUTSEL

Selects the source of the output status impressed on the device TSTAT[1:0] output pins.

0 – PL4IO output status state machine (normal operation)
1 – RSTAT input pins (PL4 local loopback operation)

STAT_OUTSEL is set to logic 0 when DRSTB is asserted.

NO_ISTAT

When NO_ISTAT is cleared to logic "0" the input FIFO status channel operates normally. When NO_ISTAT is set to logic "1" the input FIFO status channel is disabled and the RSTAT pins are ignored. The output data path can operate normally while NO_ISTAT is set to logic "1" and ISTAT_DIS is cleared to logic "0".

This bit should not be set until the output data path has been configured for normal operation.

NOTE: If the NO_ISTAT bit is set to 1 then for proper operation the PL4MOS NO_STATUS bit, in the PL4MOS Configuration Register 0x2240H, must also be set to 1. If not then the MAC's can overflow their respective FIFO's due to improper calendar operation.

ISTAT_DIS

When ISTAT_DIS is cleared to logic "0" the input FIFO status channel operates normally. When ISTAT_DIS is set to logic "1" the input FIFO status channel is disabled and the RSTAT pins are ignored. Neither the output data path nor the input status path will operate normally while ISTAT_DIS is set to logic "1".

ISTAT_DIS is set to logic "1" when DRSTB is asserted. This bit should not be cleared until the output data path and input status path have been configured for normal operation. This bit should not be cleared in slave mode until the Input Data Clock (IDCLK) is valid.

OSTAT_DIS

When OSTAT_DIS is cleared to logic "0" the output FIFO status channel operates normally. When OSTAT_DIS is set to logic "1" the output FIFO status channel is disabled and the OSTAT pins are driven high. The input data path can operate normally while OSTAT_DIS is set to logic "1".

OSTAT_DIS is set to logic "1" when DRSTB is asserted. This bit should not be cleared until the input data path and output status path have been configured for normal operation. This bit should not be cleared in slave mode until the Input Data Clock (IDCLK) is valid.

TRAIN_DIS

When TRAIN_DIS is cleared to logic "0" the PL4 input data deskew function operates normally. When TRAIN_DIS is set to logic "1" the input data deskew function is disabled and the PL4 training pattern is ignored. Neither the input data path nor the output status path will operate normally while TRAIN_DIS is set to logic "1".

TRAIN_DIS is cleared to logic "0" when DRSTB is asserted while REFSEL[0] is high (master mode). TRAIN_DIS is loaded from REFSEL[1] when DRSTB is asserted while REFSEL[0] is low (slave mode). This bit should not be cleared in slave mode until the Input Data Clock (IDCLK) is valid.

ODAT_DIS

When ODAT_DIS is cleared to logic "0" the output data bus (ODAT) operates normally. When ODAT_DIS is set to logic "1" the output data bus is disabled and the OCTL / ODAT pins are driven low. The input status path can operate normally while ODAT_DIS is set to logic "1".

ODAT_DIS is cleared to logic "0" when DRSTB is asserted while REFSEL[0] is high (master mode). ODAT_DIS is loaded from REFSEL[1] when DRSTB is asserted while REFSEL[0] is low (slave mode). This bit should not be cleared in slave mode until the Input Data Clock (IDCLK) is valid.

ENABLE

When ENABLE is set to logic "1" normal operation of the PL4IO is enabled. When ENABLE is cleared to logic "0" the PL4IO is disabled and the ENB signals to the ABCs are deasserted to minimize power consumption.

ENABLE is set to logic "1" when DRSTB is asserted.

IDDQ

When IDDQ is set to logic "1" the common IDDQ signal to the PL4IO Analog Block Components (ABCs) is asserted. This causes all digital ABC outputs going to core logic to be held static and all ABC circuitry powered from the core power supply to be held static. For normal operation IDDQ should be cleared to logic "0".

RESET

When RESET is set to logic "1" the PL4IO control logic (“wrapper”) is initialized. The effect is nearly identical to asserting DRSTB to this TSB. The Port Configuration register itself is also initialized, except for the RESET bit.

RESET does not reset the Analog Block Components (ABCs) of the PL4IO.

DIP2ERR_CHK

When DIP2ERR_CHK is set to logic ‘1’ the receive status state machine will go out of lock when an error occurs in consecutive frames. An error is defined as a dip2 error, a ‘11’ channel status, or missing an expected sync pattern. Rev A returned to the HOLD state after an error when the next end of calendar was found. A bug this caused is that the state machine would not lose lock in the case of constant invalid DIP2 fields. Setting this bit changes the state machine. After one error is detected, the state machine will return to the HOLD state after two end of calendar are found without another error.

Register 0x3040H:TXXG Configuration Register 1

Bit	Type	Function	Default
Bit 15	R/W	TXEN0	0
Bit 14	R	Unused	X
Bit 13	R/W	HOSTPAUSE	0
Bit 12	R/W	IPGT[5]	0
Bit 11	R/W	IPGT[4]	0
Bit 10	R/W	IPGT[3]	1
Bit 9	R/W	IPGT[2]	1
Bit 8	R/W	IPGT[1]	0
Bit 7	R/W	IPGT[0]	0
Bit 6	R	Unused	
Bit 5	R/W	32BIT_ALIGN	0
Bit 4	R/W	CRCEN	0
Bit 3	R/W	FCTX	0
Bit 2	R/W	FCRX	0
Bit 1	R/W	PADEN	0
Bit 0	R	Reserved	0

PADEN

If set, the TXXG will pad all IEEE 802.3 transmitted frames that are less than the minimum size up to the minimum size that is programmed in the TX_MINFR register. The minimum size is set by the Transmit Min Frame Size Register. Padding is done by inserting zero-filled octets between the end of the payload and the Start of the FCS field. If an IEEE 802.3 frame has a VLAN tag inserted, it will be padded to the minimum size + 4 bytes.

FCRX

If set, the TXXG will respond to transmit pausing requests from the RXXG when a proper PAUSE frame is received; otherwise, the TXXG ignores these signals and never interrupts the outgoing transmit stream.

FCTX

If set, the TXXG will respond to PAUSE frame transmit requests that come from either the Ingress Buffer Almost full threshold or the external PAUSE pin input; otherwise, the TXXG ignores these signals and never injects PAUSE frames into the transmit stream.

CRCEN

If CRCEN is set then all frames transmitted will have a 4 byte CRC appended. If CRCEN is not set, frames transmitted will not have a CRC appended. Note: MAC Control Pause Packets generated internally will always have a valid CRC appended, regardless of the setting of CRCEN. Corrupt packets [error bit set] will always have a corrupt CRC appended, regardless of the setting of CRCEN.

32BIT_ALIGN

If 32BIT_ALIGN is set, the packet that is output to the line must start 32-bit aligned i.e. LINE_SOP[15:0] can only occur in byte lanes [15], [11], [7] or [3]. This 32-bit alignment is for IEEE 802.3ae only. As a requirement, this bit MUST ALWAYS be programmed to 1.

IPGT [5:0]

Back-to-Back Transmit IPG. This is a programmable field representing the IPG between back-to-back packets. This is the IPG parameter used in Full-Duplex mode. Set this field to the number of octets of IPG desired. A setting of 12 decimal represents the minimum IPG of 12 bytes. The IPG setting is a average setting due to the 32-bit alignment requirement. The IPG may be +/- 3 bytes from the programmed IPG, and on average the IPG is the programmed value.

NOTE:**Because of the 32-bit alignment requirement the minimum programmed IPG that will not cause errors is 8 bytes +/- 3 bytes. So the minimum allowed setting of IPGT is 0x08H.**

Table 16 InterPacket Gap Encoding

IPGT[5:0]	IPG In Bytes	IPGT[5:0]	IPG In Bytes
00h	0	20h	32
01h	1	21h	33
02h	2	22h	34
03h	3	23h	35
04h	4	24h	36
05h	5	25h	37
06h	6	26h	38
07h	7	27h	39
08h	8	28h	40
09h	9	29h	41
0ah	10	2ah	42
0bh	11	2bh	43
0ch	12	2ch	44
0dh	13	2dh	45

IPGT[5:0]	IPG In Bytes	IPGT[5:0]	IPG In Bytes
0eh	14	2eh	46
0fh	15	2fh	47
10h	16	30h	48
11h	17	31h	49
12h	18	32h	50
13h	19	33h	51
14h	20	34h	52
15h	21	35h	53
16h	22	36h	54
17h	23	37h	55
18h	24	38h	56
19h	25	39h	57
1ah	26	3ah	58
1bh	27	3bh	59
1ch	28	3ch	60
1dh	29	3dh	61
1eh	30	3eh	62
1fh	31	3fh	63

HOSTPAUSE

HOSTPAUSE enable bit. When this bit is set to a 1 the TXXG will send PAUSE Control Frames based on the PAUSE timer and PAUSE interval registers. The operation of HOSTPAUSE is masked by FCTX.

TXEN0

This bit must be set to enable TXXG to transmit packets. All TXXG programming should be performed before setting this bit. If this bit is cleared during packet transmission, the current packet transmit will proceed to completion, and no new packet transmission initiated.

Register 0x3042H:TXXG Configuration Register 3

	Type	Function	Default
Bit 15	R/W	FIFO_ERRE	0
Bit 14	R/W	FIFO_UDRE	0
Bit 13	R/W	MAX_LERRE	0
Bit 12	R/W	MIN_LERRE	0
Bit 11	R/W	XFERE	0
Bit 10	R	Unused	X
Bit 9	R	Unused	X
Bit 8	R	Unused	X
Bit 7	R	Unused	X
Bit 6	R	Unused	X
Bit 5	R	Unused	X
Bit 4	R	Unused	X
Bit 3	R	Unused	X
Bit 2	R	Unused	X
Bit 1	R	Unused	X
Bit 0	R	Unused	X

XFERE

The XFERE bit enables the generation of an interrupt when an accumulation interval is completed and new values are stored in the filter error counter holding register. When XFERE is set to logic 1, the interrupt is enabled.

MIN_LERRE

The MIN_LERRE bit enables the generation of an interrupt for frames less than the minimum frame size programmed in the Transmit Min Frame Size Register.

MAX_LERRE

The MAX_LERRE bit enables the generation of an interrupt for frames exceeding the maximum frame size programmed in the Transmit Max Frame Size Register.

FIFO_UDRE

The FIFO_UDRE bit enables the generation of an interrupt due to a TXXG FIFO underrun event. When the FIFO_ERRE bit is set to logic 1, an underrun event will cause the FIFO_UDRI bit to be set in the Interrupt register.

FIFO_ERRE

The FIFO_ERRE bit enables the generation of an interrupt due to a FIFO error event. When the FIFO_ERRE bit is set to logic 1, a FIFO error event will cause the FIFO_ERRI bit to be set in the Interrupt register. FIFO error events are:

- FIFO_ERR asserted with FIFO_EOP.
- Invalid SOP/ EOP sequence.
- Invalid gapping during data transfer.
- Frame length less than 14 bytes.
- FIFO overrun.

Register 0x3043H:TXXG Interrupt

Bit	Type	Function	Default
Bit 15	R	FIFO_ERRI	0
Bit 14	R	FIFO_UDRI	0
Bit 13	R	MAX_LERRI	0
Bit 12	R	MIN_LERRI	0
Bit 11	R	XFERI	0
Bit 10	R	Unused	X
Bit 9	R	Unused	X
Bit 8	R	Unused	X
Bit 7	R	Unused	X
Bit 6	R	Unused	X
Bit 5	R	Unused	X
Bit 4	R	Unused	X
Bit 3	R	Unused	X
Bit 2	R	Unused	X
Bit 1	R	Unused	X
Bit 0	R	Unused	X

XFERI

The XFERI bit indicates that a transfer of accumulated counter data has occurred. A logic 1 in this bit position indicates that the filter error counter holding register has been updated. This update is initiated by writing to one of the counter register locations, or by. If XFERE is set in Configuration Register, then an interrupt will also be generated (INT output asserted). The bit will clear on read.

MIN_LERRI

The MIN_LERRI bit is set for frames less than the value programmed in the Transmit Min Frame Size Register and if PADEN (bit 1 of the TXXG Configuration register 0x3040) is set to 0. If MIN_LERRE is set in TXXG Configuration Register 3, then an interrupt will also be generated (INT output asserted). The bit will clear on read.

MAX_LERRI

The MAX_LERRI bit is for frames exceeding the value programmed in the Transmit Max Frame Size Register. If MAX_LERRE is set in TXXG Configuration Register 3, then an interrupt will also be generated (INT output asserted). The bit will clear on read.

FIFO_UDRI

The FIFO_UDRI bit is set when a FIFO underrun event occurs. An underrun condition exists if the TXXG is in the middle of transmitting a packet and no additional data is available to be transmitted. If FIFO_UDRE is set in TXXG Configuration Register 2, then an interrupt will also be generated (INT output asserted). The bit will clear on read.

The FIFO_ERRI bit is set when a FIFO error event occurs. FIFO error events are:

- FIFO_ERR asserted with FIFO_EOP.
- Invalid SOP/ EOP sequence.
- Invalid gapping during data transfer.
- Frame length less than 14 bytes.
- FIFO overrun.

Register 0x3044H:TXXG Status Register

Bit	Type	Function	Default
Bit 15	R	Unused	X
Bit 14	R	Unused	X
Bit 13	R	Unused	X
Bit 12	R	Unused	X
Bit 11	R	Unused	X
Bit 10	R	Unused	X
Bit 9	R	Unused	X
Bit 8	R	Unused	X
Bit 7	R	Unused	X
Bit 6	R	Unused	X
Bit 5	R	Unused	X
Bit 4	R	Unused	X
Bit 3	R	Unused	X
Bit 2	R	Unused	X
Bit 1	R	TXACTIVE	0
Bit 0	R	PAUSED	0

PAUSED

When set, indicates that the PAUSE timer is non-zero. This will happen whenever the RXXG interface transfers a non-zero pause value, or the CPU test mode loads a non-zero value, and the timer has not yet decremented to 0.

TXACTIVE

When set, indicates that the TXXG is currently in the process of transmitting a packet from the System interface to the Line interface, or is emitting a PAUSE packet.

Register 0x3045H:TXXG TX_MAXFR Transmit Max Frame Size Register

Bit	Type	Function	Default
Bit 15	R/W	TX_MAXFR[15]	0
Bit 14	R/W	TX_MAXFR[14]	0
Bit 13	R/W	TX_MAXFR[13]	0
Bit 12	R/W	TX_MAXFR[12]	0
Bit 11	R/W	TX_MAXFR[11]	0
Bit 10	R/W	TX_MAXFR[10]	1
Bit 9	R/W	TX_MAXFR[9]	0
Bit 8	R/W	TX_MAXFR[8]	1
Bit 7	R/W	TX_MAXFR[7]	1
Bit 6	R/W	TX_MAXFR[6]	1
Bit 5	R/W	TX_MAXFR[5]	1
Bit 4	R/W	TX_MAXFR[4]	0
Bit 3	R/W	TX_MAXFR[3]	1
Bit 2	R/W	TX_MAXFR[2]	1
Bit 1	R/W	TX_MAXFR[1]	1
Bit 0	R/W	TX_MAXFR[0]	0

TX_MAXFR[15:0]

TX_MAXFR[15:0] places an upper bound on a transmitted Ethernet frame, in octets. The length is measured from the first byte of the destination address to the last byte of the FCS field. The default is 0x05EE, corresponding to the Standard setting of 1518 bytes or a maximum untagged Ethernet frame. Four bytes are automatically added for VLAN tagged Ethernet frames. Maximum setting is 9600 bytes. If the pre-set frame length is exceeded by the frame source, the TXXG unit will truncate the outgoing frame, force a CRC error, and discard all subsequently received bytes until the next SOF.

Register 0x3046H:TXXG TX_MINFR Transmit Min Frame Size Register

Bit	Type	Function	Default
Bit 15	R	Unused	X
Bit 14	R	Unused	X
Bit 13	R	Unused	X
Bit 12	R	Unused	X
Bit 11	R	Unused	X
Bit 10	R	Unused	X
Bit 9	R	Unused	X
Bit 8	R	Unused	X
Bit 7	R/W	TX_MINFR[7]	0
Bit 6	R/W	TX_MINFR[6]	0
Bit 5	R/W	TX_MINFR[5]	1
Bit 4	R/W	TX_MINFR[4]	1
Bit 3	R/W	TX_MINFR[3]	1
Bit 2	R/W	TX_MINFR[2]	1
Bit 1	R/W	TX_MINFR[1]	0
Bit 0	R/W	TX_MINFR[0]	0

TX_MINFR[7:0]

The TX_MINFR[7:0] field is used to set a lower limit on the size of a transmitted Ethernet frame. The length is measured from the first byte of the destination address to the last byte of the payload, excluding the 4-byte FCS field. The default is set to 60 decimal, corresponding to the nominal 64-byte minimum-size frame. If the TXXG receives an IEEE 802.3 frame (i.e., with the Length/ Type field set to 0x05DC or less) that is smaller than TX_MINFR[7:0], and the PADEN bit is set in the TXXG Configuration Register 1, it will pad the frame to TX_MINFR[7:0] with zeros. PADEN **MUST** be enabled for frame sizes less than 32 bytes.

Register 0x3047H:TXXG SA[15:0] Station Address

Bit	Type	Function	Default
Bit 15	R/W	SA[15]	0
Bit 14	R/W	SA[14]	0
Bit 13	R/W	SA[13]	0
Bit 12	R/W	SA[12]	0
Bit 11	R/W	SA[11]	0
Bit 10	R/W	SA[10]	0
Bit 9	R/W	SA[9]	0
Bit 8	R/W	SA[8]	0
Bit 7	R/W	SA[7]	0
Bit 6	R/W	SA[6]	0
Bit 5	R/W	SA[5]	0
Bit 4	R/W	SA[4]	0
Bit 3	R/W	SA[3]	0
Bit 2	R/W	SA[2]	0
Bit 1	R/W	SA[1]	0
Bit 0	R/W	SA[0]	0

SA[15:0]

The SA[15:0] register sets the low 16 bits of the 48-bit Station address used for PAUSE frame generation.

Register 0x3048H:TXXG SA[31:16] Station Address

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

SA[31:16]

The SA[31:16] register sets the middle 16 bits of the 48-bit Station address used for PAUSE frame generation.

Register 0x3049H:TXXG SA[47:32] Station Address

Bit	Type	Function	Default
Bit 15	R/W	SA[47]	0
Bit 14	R/W	SA[46]	0
Bit 13	R/W	SA[45]	0
Bit 12	R/W	SA[44]	0
Bit 11	R/W	SA[43]	0
Bit 10	R/W	SA[42]	0
Bit 9	R/W	SA[41]	0
Bit 8	R/W	SA[40]	0
Bit 7	R/W	SA[39]	0
Bit 6	R/W	SA[38]	0
Bit 5	R/W	SA[37]	0
Bit 4	R/W	SA[36]	0
Bit 3	R/W	SA[35]	0
Bit 2	R/W	SA[34]	0
Bit 1	R/W	SA[33]	0
Bit 0	R/W	SA[32]	0

SA[47:32]

The SA[47:32] register sets the high 16 bits of the 48-bit Station address used for PAUSE frame generation.

Register 0x304DH:TXXG PAUSE_TIME – PAUSE TIMER Register

Bit	Type	Function	Default
Bit 15	R/W	PAUSE_TIME[15]	1
Bit 14	R/W	PAUSE_TIME [14]	1
Bit 13	R/W	PAUSE_TIME [13]	1
Bit 12	R/W	PAUSE_TIME [12]	1
Bit 11	R/W	PAUSE_TIME [11]	1
Bit 10	R/W	PAUSE_TIME [10]	1
Bit 9	R/W	PAUSE_TIME [9]	1
Bit 8	R/W	PAUSE_TIME [8]	1
Bit 7	R/W	PAUSE_TIME [7]	1
Bit 6	R/W	PAUSE_TIME [6]	1
Bit 5	R/W	PAUSE_TIME [5]	1
Bit 4	R/W	PAUSE_TIME [4]	1
Bit 3	R/W	PAUSE_TIME [3]	1
Bit 2	R/W	PAUSE_TIME [2]	1
Bit 1	R/W	PAUSE_TIME [1]	1
Bit 0	R/W	PAUSE_TIME [0]	1

PAUSE_TIME [15:0]

This register contains the PAUSE timer value that is used on the PAUSE Control Frames that are sent to the Line Interface. The default is 0xFFFF for an XON/XOFF type of protocol. In diagnostic mode, this register must be programmed before DIAG_HOSTPAUSE is toggled from 0 to 1. In normal mode, this register must be programmed before HOSTPAUSE is toggled from 0 to 1.

Register 0x304EH:TXXG PAUSE_IVAL PAUSE Timer Interval Register

Bit	Type	Function	Default
Bit 15	R/W	PAUSE_IVAL[15]	1
Bit 14	R/W	PAUSE_ IVAL [14]	1
Bit 13	R/W	PAUSE_ IVAL [13]	0
Bit 12	R/W	PAUSE_ IVAL[12]	0
Bit 11	R/W	PAUSE_ IVAL [11]	1
Bit 10	R/W	PAUSE_ IVAL [10]	1
Bit 9	R/W	PAUSE_ IVAL [9]	1
Bit 8	R/W	PAUSE_ IVAL [8]	1
Bit 7	R/W	PAUSE_ IVAL [7]	1
Bit 6	R/W	PAUSE_ IVAL [6]	1
Bit 5	R/W	PAUSE_ IVAL [5]	1
Bit 4	R/W	PAUSE_ IVAL [4]	1
Bit 3	R/W	PAUSE_ IVAL [3]	1
Bit 2	R/W	PAUSE_ IVAL [2]	1
Bit 1	R/W	PAUSE_ IVAL [1]	1
Bit 0	R/W	PAUSE_ IVAL [0]	1

PAUSE_ IVAL [15:0]

This register contains the Pause Timer Interval value that is used by the PAUSE Generation Logic to control how often a PAUSE Control frame is sent. The PAUSE_IVAL decrements every 512 bit times (PAUSE quanta). The default value is 0xCFFF.

Register 0x3052H:TXXG Pause Quantum Value Configuration Register

Bit	Type	Function	Default
Bit 15	R/W	Unused	X:high
Bit 14	R/W	Unused	X:high
Bit 13	R/W	Unused	X:high
Bit 12	R/W	Unused	X:high
Bit 11	R/W	Unused	X:high
Bit 10	R/W	Unused	X:high
Bit 9	R/W	Unused	X:high
Bit 8	R/W	FC_PHY_PACE_EN	0
Bit 7	R/W	FC_PAUSE_QNTM [7]	0
Bit 6	R/W	FC_PAUSE_QNTM [6]	0
Bit 5	R/W	FC_PAUSE_QNTM [5]	0
Bit 4	R/W	FC_PAUSE_QNTM [4]	0
Bit 3	R/W	FC_PAUSE_QNTM [3]	0
Bit 2	R/W	FC_PAUSE_QNTM [2]	1
Bit 1	R/W	FC_PAUSE_QNTM [1]	1
Bit 0	R/W	FC_PAUSE_QNTM [0]	1

FC_PAUSE_QNTM [7:0]

The FC_PAUSE_QNTM [15:0] register indicates the number of sysclkX2 cycles that are to be used for scaling the pause_quantum value (minus 1). A value of 7 means that every 8 sysclkX2 cycles, the local pause timer value decrements by 1.

FC_PHY_PACE_EN If FC_PHY_PACE_EN is set, the local pause timer in the incoming_fc module will stall decrementing [when loaded] once in every 33 sysclkX2 cycles.

Register 0x3080H:T64B66B Configuration 1

Bit	Type	Function	Default
Bit 15	R/W	TIP	0:high
Bit 14	R/W	SWAVE_LEN[2]	0:high
Bit 13	R/W	SWAVE_LEN[1]	0:high
Bit 12	R/W	SWAVE_LEN[0]	0:high
Bit 11	R	Unused	0:high
Bit 10	R	Unused	0:high
Bit 9	R	Unused	0:high
Bit 8	R/W	JDAT_PAT_SEL	0:high
Bit 7	R/W	JTST_PAT_SEL	0:high
Bit 6	R/W	JITT_PAT_EN	0:high
Bit 5	R/W	RXERR_DIS	0
Bit 4	R/W	Reserved	0
Bit 3	R/W		0
Bit 2	R/W	Reserved	1
Bit 1	R/W	INT_EN	0
Bit 0	R/W	Reserved	

INT_EN

When ‘1’, the Interrupts are enabled.

When ‘0’, the Interrupts are disabled.

RESERVED

This bit is reserved and must be programmed to 1.

RX_ERRDIS

When ‘1’, The T64B66B Block will ignore all of the receiver detected error events: Loss Of Signal (LOS), Remote Fault (RF), and Local Fault (LF). All data received from the upstream device will be processed as normal. The associated interrupts (RX_LFI, RX_RFI and RX_LOSI) will still assert even if the RX_ERRDIS bit is set to ‘1’.

When ‘0’, The T64B66B Block will NOT ignore the receiver detected error events: Loss Of Signal (LOS), Remote Fault (RF), Local Fault (LF). All data received from the upstream device will be processed as normal only if the receiver is not indicating any error(s).

JITT_PAT_EN

When ‘1’, the T64B66B block will ignore all input data and transmit a jitter test pattern using the scrambler and the data defined by the JDAT_PAT_SEL bit.

When ‘0’ the T64BT66B transmits data normally.

JTST_PAT_SEL

When ‘1’ the square wave test pattern is used for jitter testing. In this mode the T64BT66B transmits a square wave which consists of 4 – 11 alternating series of 1’s and 0’s where the 4 – 11 is defined by the SWAVE_LEN bits.

When ‘0’ the pseudo random test pattern is used for jitter testing. The input to the scrambler is then defined by JDAT_PAT_SEL and the seeds used are defined by the JITTER TEST SEED registers.

JDAT_PAT_SEL

When ‘1’ the zeros data pattern is used for jitter testing. If we are in pseudo random test pattern mode also, the input to the scrambler is all zeros.

When ‘0’ the LF ordered set is used for jitter testing. When JITT_PAT_EN is ‘1’ also the input to the scrambler is a constant LF.

SWAVE_LEN

Number of consecutive 0’s and 1’s that constitute the square wave generated when we are in square wave test pattern mode. The value is offset by 4 so that the range is from 4 – 11 i.e. SWAVE_LEN=000 means 4 0’s and 1’s.

TIP

When read this bit indicates that a transfer is in progress of the status register 0x03h. When this bit is written to, a transfer of the status bits occurs.

Register 0x3083H:T64B66B Status

Bit	Type	Function	Default
Bit 15	R	Unused	X
Bit 14	R	Unused	X
Bit 13	R	Unused	X
Bit 12	R	Unused	X
Bit 11	R	Unused	X
Bit 10	R	Unused	X
Bit 9	R	Unused	X
Bit 8	R	Unused	X
Bit 7	R	Reserved	0
Bit 6	R	RX_LF	0
Bit 5	R	RX_RF	0
Bit 4	R	RX_LOS	0
Bit 3	R	FIFO_UNRUN	0
Bit 2	R	FIFO_OVRUN	0
Bit 1	R	SEQ	0
Bit 0	R	SOP	0

NOTE: To update these status bits one must first write to the T64B66B TIP bit (bit-15) in Register 0x3080 T64B66B Configuration 1. Then a read can be performed to get the status.

SOP

When ‘1’, indicates that a non 32 bit aligned packet was received.

When ‘0’, indicates that only aligned packets were received.

SEQ

When ‘1’, indicates that the T64B66B has detected a incorrect SOP/EOP or EOP/SOP sequence.

When ‘0’, indicates that the T64B66B has not detected any incorrect SOP/EOP or EOP/SOP sequences.

FIFO_OVRUN

When ‘1’, indicates that the T64B66B’s FIFO has experienced an overflow since the last status read. An overflow is set when the FIFO is full and a write cycle is executed ahead of the next read cycle.

When ‘0’, indicates that the T64B66B’s FIFO has not experienced an overflow since the last status read.

FIFO_UNRUN

When ‘1’, indicates that the T64B66B’s FIFO has experienced an under run since the last status read. An under run is set when the FIFO is empty and a read cycle is executed ahead of the next write cycle.

When ‘0’, indicates that the T64B66B’s FIFO has not experienced an overflow since the last status read.

RX_LOS

When ‘1’, indicates that the T64B66B block has received an external:

Receiver Loss of Signal (LOS) event and is still sending an RF message.

When ‘0’, indicates that the T64B66B has not received an external:

Receiver Loss of Signal event and is capturing data from the upstream device.

RX_RF

When ‘1’, indicates that the T64B66B block has received an external: Receiver Remote Fault (RF) event and is still sending an RF message.

When ‘0’, indicates that the T64B66B has not received an external:

Receiver Remote Fault event and is capturing data from the upstream device.

RX_LF

When ‘1’, indicates that the T64B66B block has received an external: Receiver Local Fault (LF) event and is still sending an idle message.

When ‘0’, indicates that the T64B66B has not received an external:

Receiver Local Fault event and is capturing data from the upstream device.

Register 0x3085H: JITTER TEST SEED A 3

Bit	Type	Function	Default
Bit 15	R	Unused	X
Bit 14	R	Unused	X
Bit 13	R	Unused	X
Bit 12	R	Unused	X
Bit 11	R	Unused	X
Bit 10	R	Unused	X
Bit 9	R/W	JITT_SEED_A[57]	0
Bit 8	R/W	JITT_SEED_A[56]	0
Bit 7	R/W	JITT_SEED_A[55]	0
Bit 6	R/W	JITT_SEED_A[54]	0
Bit 5	R/W	JITT_SEED_A[53]	0
Bit 4	R/W	JITT_SEED_A[52]	0
Bit 3	R/W	JITT_SEED_A[51]	0
Bit 2	R/W	JITT_SEED_A[50]	0
Bit 1	R/W	JITT_SEED_A[49]	0
Bit 0	R/W	JITT_SEED_A[48]	0

JITT_SEED_A[57:48]

Contains bits 57 down to 48 of the 58 bit pseudo random jitter test pattern seed A.

NOTE: These bits can only be written when jitter pattern generation is turned off, i.e. JITT_PAT_EN is written low.

Register 0x3086H: JITTER TEST SEED A 2

Bit	Type	Function	Default
Bit 15	R/W	JITT_SEED_A[47]	1
Bit 14	R/W	JITT_SEED_A[46]	1
Bit 13	R/W	JITT_SEED_A[45]	1
Bit 12	R/W	JITT_SEED_A[44]	1
Bit 11	R/W	JITT_SEED_A[43]	1
Bit 10	R/W	JITT_SEED_A[42]	1
Bit 9	R/W	JITT_SEED_A[41]	1
Bit 8	R/W	JITT_SEED_A[40]	1
Bit 7	R/W	JITT_SEED_A[39]	1
Bit 6	R/W	JITT_SEED_A[38]	1
Bit 5	R/W	JITT_SEED_A[37]	1
Bit 4	R/W	JITT_SEED_A[36]	1
Bit 3	R/W	JITT_SEED_A[35]	1
Bit 2	R/W	JITT_SEED_A[34]	1
Bit 1	R/W	JITT_SEED_A[33]	1
Bit 0	R/W	JITT_SEED_A[32]	1

JITT_SEED_A[47:32]

Contains bits 47 down to 32 of the 58 bit pseudo random jitter test pattern seed A.

NOTE: These bits can only be written when jitter pattern generation is turned off, i.e. JITT_PAT_EN is written low.

Register 0x3087H: JITTER TEST SEED A 1

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

JITT_SEED_A[31:16]

Contains bits 31 down to 16 of the 58 bit pseudo random jitter test pattern seed A.

NOTE: These bits can only be written when jitter pattern generation is turned off, i.e. JITT_PAT_EN is written low.

Register 0x3088H: JITTER TEST SEED A 0

Bit	Type	Function	Default
Bit 15	R/W	JITT_SEED_A[15]	1
Bit 14	R/W	JITT_SEED_A[14]	1
Bit 13	R/W	JITT_SEED_A[13]	1
Bit 12	R/W	JITT_SEED_A[12]	1
Bit 11	R/W	JITT_SEED_A[11]	1
Bit 10	R/W	JITT_SEED_A[10]	1
Bit 9	R/W	JITT_SEED_A[9]	1
Bit 8	R/W	JITT_SEED_A[8]	1
Bit 7	R/W	JITT_SEED_A[7]	1
Bit 6	R/W	JITT_SEED_A[6]	1
Bit 5	R/W	JITT_SEED_A[5]	1
Bit 4	R/W	JITT_SEED_A[4]	1
Bit 3	R/W	JITT_SEED_A[3]	1
Bit 2	R/W	JITT_SEED_A[2]	1
Bit 1	R/W	JITT_SEED_A[1]	1
Bit 0	R/W	JITT_SEED_A[0]	1

JITT_SEED_A[15:0]

Contains bits 15 down to 0 of the 58 bit pseudo random jitter test pattern seed A.

NOTE: These bits can only be written when jitter pattern generation is turned off, i.e. JITT_PAT_EN is written low.

Register 0x3089H: JITTER TEST SEED B 3

Bit	Type	Function	Default
Bit 15	R/W	unused	X
Bit 14	R	Unused	X
Bit 13	R	Unused	X
Bit 12	R	Unused	X
Bit 11	R	Unused	X
Bit 10	R	Unused	X
Bit 9	R/W	JITT_SEED_B[57]	0
Bit 8	R/W	JITT_SEED_B[56]	0
Bit 7	R/W	JITT_SEED_B[55]	0
Bit 6	R/W	JITT_SEED_B[54]	0
Bit 5	R/W	JITT_SEED_B[53]	0
Bit 4	R/W	JITT_SEED_B[52]	0
Bit 3	R/W	JITT_SEED_B[51]	0
Bit 2	R/W	JITT_SEED_B[50]	0
Bit 1	R/W	JITT_SEED_B[49]	0
Bit 0	R/W	JITT_SEED_B[48]	0

JITT_SEED_B[57:48]

Contains bits 57 down to 48 of the 58 bit pseudo random jitter test pattern seed B.

NOTE: These bits can only be written when jitter pattern generation is turned off, i.e. JITT_PAT_EN is written low.

Register 0x308AH: JITTER TEST SEED B 2

Bit	Type	Function	Default
Bit 15	R/W	JITT_SEED_B[47]	1
Bit 14	R/W	JITT_SEED_B[46]	1
Bit 13	R/W	JITT_SEED_B[45]	1
Bit 12	R/W	JITT_SEED_B[44]	1
Bit 11	R/W	JITT_SEED_B[43]	1
Bit 10	R/W	JITT_SEED_B[42]	1
Bit 9	R/W	JITT_SEED_B[41]	1
Bit 8	R/W	JITT_SEED_B[40]	1
Bit 7	R/W	JITT_SEED_B[39]	1
Bit 6	R/W	JITT_SEED_B[38]	1
Bit 5	R/W	JITT_SEED_B[37]	1
Bit 4	R/W	JITT_SEED_B[36]	1
Bit 3	R/W	JITT_SEED_B[35]	1
Bit 2	R/W	JITT_SEED_B[34]	1
Bit 1	R/W	JITT_SEED_B[33]	1
Bit 0	R/W	JITT_SEED_B[32]	1

JITT_SEED_B[47:32]

Contains bits 47 down to 32 of the 58 bit pseudo random jitter test pattern seed B.

NOTE: These bits can only be written when jitter pattern generation is turned off, i.e. JITT_PAT_EN is written low.

Register 0x308BH: JITTER TEST SEED B 1

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

JITT_SEED_B[31:16]

Contains bits 31 down to 16 of the 58 bit pseudo random jitter test pattern seed B.

NOTE: These bits can only be written when jitter pattern generation is turned off, i.e. JITT_PAT_EN is written low.

Register 0x308CH: JITTER TEST SEED B 0

Bit	Type	Function	Default
Bit 15	R/W	JITT_SEED_B[15]	1
Bit 14	R/W	JITT_SEED_B[14]	1
Bit 13	R/W	JITT_SEED_B[13]	1
Bit 12	R/W	JITT_SEED_B[12]	1
Bit 11	R/W	JITT_SEED_B[11]	1
Bit 10	R/W	JITT_SEED_B[10]	1
Bit 9	R/W	JITT_SEED_B[9]	1
Bit 8	R/W	JITT_SEED_B[8]	1
Bit 7	R/W	JITT_SEED_B[7]	1
Bit 6	R/W	JITT_SEED_B[6]	1
Bit 5	R/W	JITT_SEED_B[5]	1
Bit 4	R/W	JITT_SEED_B[4]	1
Bit 3	R/W	JITT_SEED_B[3]	1
Bit 2	R/W	JITT_SEED_B[2]	1
Bit 1	R/W	JITT_SEED_B[1]	1
Bit 0	R/W	JITT_SEED_B[0]	1

JITT_SEED_B[15:0]

Contains bits 15 down to 0 of the 58 bit pseudo random jitter test pattern seed B.

NOTE: These bits can only be written when jitter pattern generation is turned off, i.e. JITT_PAT_EN is written low.

Register 0x3200H:EFLX Global Configuration

Bit	Type	Function	Default
Bit 15	R/W	ERCU_EN	0
Bit 14	R	Unused	0
Bit 13	R	Unused	0
Bit 12	R	Unused	0
Bit 11	R	Unused	0
Bit 10	R	Unused	0
Bit 9	R	Unused	0
Bit 8	R	Unused	0
Bit 7	R/W	EN_EDSWT	0
Bit 6	R	Unused	0
Bit 5	R	Unused	0
Bit 4	R	Unused	0
Bit 3	R	Unused	0
Bit 2	R	Unused	0
Bit 1	R	Unused	0
Bit 0	R	Unused	0

ERCU_EN

The ERCU_EN bit must be set for the normal operation

EDSWT_EN

The EDSWT_EN bit must be set for normal operation.

Register 0x3201H:EFLX ERCU Global Status

Bit	Type	Function	Default
Bit 15	R	Reserved	0
Bit 14	R	Reserved	0
Bit 13	R	OVF_ERR	0
Bit 12	R	Reserved	0
Bit 11	R	Unused	0
Bit 10	R	Unused	0
Bit 9	R	Unused	0
Bit 8	R	Unused	0
Bit 7	R	Reserved	0
Bit 6	R	Reserved	0
Bit 5	R	Reserved	0
Bit 4	R	Reserved	0
Bit 3	R	Reserved	0
Bit 2	R	Reserved	0
Bit 1	R	Reserved	0
Bit 0	R	Reserved	0

OVF_ERR

When OVR_ERR is asserted the address register and FIFO flag unit has detected an overflow (i.e., a write to a FIFO that could not be performed because the FIFO was completely full). The OVF_ERR status bit is cleared when the offending channel status is read.

Register 0x3202H:EFLX Indirect Channel Address

Bit	Type	Function	Default
Bit 15	R	BUSY	0
Bit 14	R/W	RWB	1
Bit 13	R	Unused	0
Bit 12	R	Unused	0
Bit 11	R	Unused	0
Bit 10	R	Unused	0
Bit 9	R	Unused	0
Bit 8	R	Unused	0
Bit 7	R	Unused	0
Bit 6	R	Unused	0
Bit 5	R	Unused	0
Bit 4	R	Unused	0
Bit 3	R/W	Reserved	0
Bit 2	R/W	Reserved	0
Bit 1	R/W	Reserved	0
Bit 0	R/W	Reserved	0

RESERVED

The reserved bits in this register must remain 0.

RWB

The read/write bar (RWB) bit selects between an update operation (write) and a query operation (read). Writing logic 0 to RWB triggers the update operation with the information in Indirect Registers. Writing a logic 1 to RWB triggers a query and the information is placed in all of the Indirect Registers.

BUSY

The indirect access status bit (BUSY) reports the progress of an indirect access. BUSY is set high when a write to the Indirect Channel Select register triggers an indirect access and will stay high until the access is complete. This register should be polled to determine when data from an indirect read operation is available in all the Indirect Data registers or to determine when a new indirect write operation may commence.

Register 0x3203H:EFLX Indirect Logical FIFO Low Limit

Bit	Type	Function	Default
Bit 15	R	Unused	0
Bit 14	R	Unused	0
Bit 13	R	Unused	0
Bit 12	R	Unused	0
Bit 11	R	Unused	0
Bit 10	R	Unused	0
Bit 9	R/W	LOLIM[9]	0
Bit 8	R/W	LOLIM[8]	0
Bit 7	R/W	LOLIM[7]	0
Bit 6	R/W	LOLIM[6]	0
Bit 5	R/W	LOLIM[5]	0
Bit 4	R/W	LOLIM[4]	0
Bit 3	R/W	LOLIM[3]	0
Bit 2	R/W	LOLIM[2]	0
Bit 1	R/W	LOLIM[1]	0
Bit 0	R/W	LOLIM[0]	0

LOLIM[9:0]

The lower address boundary of the ring buffer for the logical FIFO. This value should remain zero.

Register 0x3204H:EFLX Indirect Logical FIFO High Limit

Bit	Type	Function	Default
Bit 15	R	Unused	0
Bit 14	R	Unused	0
Bit 13	R	Unused	0
Bit 12	R	Unused	0
Bit 11	R	Unused	0
Bit 10	R	Unused	0
Bit 9	R/W	HILIM[9]	0
Bit 8	R/W	HILIM[8]	0
Bit 7	R/W	HILIM[7]	0
Bit 6	R/W	HILIM[6]	0
Bit 5	R/W	HILIM[5]	0
Bit 4	R/W	HILIM[4]	0
Bit 3	R/W	HILIM[3]	0
Bit 2	R/W	HILIM[2]	0
Bit 1	R/W	HILIM[1]	0
Bit 0	R/W	HILIM[0]	0

HILIM[9:0]

The upper address boundary, in units of 256 bytes, of the ring buffer for the logical FIFO. In the PM3392, this should be programmed to 64 decimal or 40 hex.

Register 0x3205H:EFLX Indirect Full/Almost-Full Status and Limit

Bit	Type	Function	Default
Bit 15	R	FULL	0
Bit 14	R	AFULL	0
Bit 13	R/W	AFTH[13]	0
Bit 12	R/W	AFTH[12]	0
Bit 11	R/W	AFTH[11]	0
Bit 10	R/W	AFTH [10]	0
Bit 9	R/W	AFTH [9]	0
Bit 8	R/W	AFTH [8]	0
Bit 7	R/W	AFTH [7]	0
Bit 6	R/W	AFTH [6]	0
Bit 5	R/W	AFTH [5]	0
Bit 4	R/W	AFTH [4]	0
Bit 3	R/W	AFTH [3]	0
Bit 2	R/W	AFTH [2]	0
Bit 1	R/W	AFTH [1]	0
Bit 0	R/W	AFTH [0]	0

AFTH [13:0]

The AFTH [13:0] field holds the threshold, in terms number of 128-bit words currently present in the logical FIFO before an almost-full status is reported. AFTH [13:0] should be written at initialization time (ERCUEN = '0') with the total number of words allocated to the logical FIFO, minus 3 for the latency through the ERCU block itself, minus the required amount for the latency in the system bus controller. It is not changed by the ERCU.

AFULL

The almost full status bit (AFULL) is set when the number of words within the logical FIFO is greater than the AFTH[13:0] value. This status information is provided for diagnostic purposes.

FULL

The full status bit (FULL) is set when the logical FIFO is reporting a full status to the upstream write datapath. This status information is provided for diagnostic purposes.

Register 0x3206H:EFLX Indirect Empty/Almost-Empty Status and Limit

Bit	Type	Function	Default
Bit 15	R	EMPTY	0
Bit 14	R	AEMPTY	0
Bit 13	R/W	AETH[13]	0
Bit 12	R/W	AETH [12]	0
Bit 11	R/W	AETH[11]	0
Bit 10	R/W	AETH [10]	0
Bit 9	R/W	AETH [9]	0
Bit 8	R/W	AETH [8]	0
Bit 7	R/W	AETH [7]	0
Bit 6	R/W	AETH [6]	0
Bit 5	R/W	AETH [5]	0
Bit 4	R/W	AETH [4]	0
Bit 3	R/W	AETH [3]	0
Bit 2	R/W	AETH [2]	0
Bit 1	R/W	AETH [1]	0
Bit 0	R/W	AETH [0]	0

AETH [13:0]

The AETH [13:0] field holds the threshold, in terms number of 128-bit words currently present in the logical before an almost-empty status is reported. AETH [13:0] should be written at initialization time (ERCUEN = '0') with a value greater than 0, but less than the total number of words allocated to the logical FIFO. It is not changed by the ERCU.

AEMPTY

The almost empty status bit (AEMPTY) is set when the number of words within the logical FIFO is less than or equal to the AETH[13:0] value. This status information is provided for diagnostic purposes.

EMPTY

The empty status bit (EMPTY) is set whenever the logical FIFO is completely empty. This flag is provided for diagnostic purposes.

Register 0x3207H:EFLX Indirect FIFO Cut-Through Threshold

Bit	Type	Function	Default
Bit 15	R	Unused	0
Bit 14	R	Unused	0
Bit 13	R/W	CUT_THRU[13]	0
Bit 12	R/W	CUT_THRU [12]	0
Bit 11	R/W	CUT_THRU [11]	0
Bit 10	R/W	CUT_THRU [10]	0
Bit 9	R/W	CUT_THRU [9]	0
Bit 8	R/W	CUT_THRU [8]	0
Bit 7	R/W	CUT_THRU [7]	0
Bit 6	R/W	CUT_THRU [6]	0
Bit 5	R/W	CUT_THRU [5]	0
Bit 4	R/W	CUT_THRU [4]	0
Bit 3	R/W	CUT_THRU [3]	0
Bit 2	R/W	CUT_THRU [2]	0
Bit 1	R/W	CUT_THRU [1]	0
Bit 0	R/W	CUT_THRU [0]	0

CUT_THRU [13:0]

The CUT_THRU [13:0] field holds the threshold, in terms number of 128-bit words currently present in the logical before a frame is released. It should be written at initialization time (ERCU_EN = '0' and PROV = '1') with the total number of words of a given frame that should be written for the logical FIFO before reads are issued on the line side to minimize the probability of an underflow situation. It is not changed by the ERCU.

Note: The EFLX will begin transmitting when the fill level is within 208 bytes of the cut through threshold. Also note that depending on traffic flow across the PL4 interface the value of the CUT_THRU register can cause transmit underruns in the transmit MAC. Please refer to section 13.13.1 in the Operation section to help clarify the use of the CUT_THRU register.

Register 0x320CH:EFLX FIFO Overflow Error Enable

Bit	Type	Function	Default
Bit 15	R/W	Reserved	0
Bit 14	R/W	Reserved	0
Bit 13	R/W	Reserved	0
Bit 12	R/W	Reserved	0
Bit 11	R/W	Reserved	0
Bit 10	R/W	Reserved	0
Bit 9	R/W	Reserved	0
Bit 8	R/W	Reserved	0
Bit 7	R/W	Reserved	0
Bit 6	R/W	Reserved	0
Bit 5	R/W	Reserved	0
Bit 4	R/W	Reserved	0
Bit 3	R/W	Reserved	0
Bit 2	R/W	Reserved	0
Bit 1	R/W	Reserved	0
Bit 0	R/W	OVFE	0

OVFE

The overflow interrupt enable controls the assertion of the INT output when OVFI. When OVFE is set high, an interrupt is generated upon assertion event of the OVFI register. When OVFE is set low, changes in the OVFI status do not generate an interrupt.

Register 0x320DH: EFLX FIFO Overflow Error Indication

Bit	Type	Function	Default
Bit 15	R	Reserved	0
Bit 14	R	Reserved	0
Bit 13	R	Reserved	0
Bit 12	R	Reserved	0
Bit 11	R	Reserved	0
Bit 10	R	Reserved	0
Bit 9	R	Reserved	0
Bit 8	R	Reserved	0
Bit 7	R	Reserved	0
Bit 6	R	Reserved	0
Bit 5	R	Reserved	0
Bit 4	R	Reserved	0
Bit 3	R	Reserved	0
Bit 2	R	Reserved	0
Bit 1	R	Reserved	0
Bit 0	R	OVFI	0

OVFI

The OVFI interrupt flag is set whenever an overflow (i.e., an attempted write to a completely full FIFO) is detected. The OVFI register bit is cleared immediately after it is read

Register 0x3210H:EFLX Channel Provision

Bit	Type	Function	Default
Bit 15	R/W	Reserved	0
Bit 14	R/W	Reserved	0
Bit 13	R/W	Reserved	0
Bit 12	R/W	Reserved	0
Bit 11	R/W	Reserved	0
Bit 10	R/W	Reserved	0
Bit 9	R/W	Reserved	0
Bit 8	R/W	Reserved	0
Bit 7	R/W	Reserved	0
Bit 6	R/W	Reserved	0
Bit 5	R/W	Reserved	0
Bit 4	R/W	Reserved	0
Bit 3	R/W	Reserved	0
Bit 2	R/W	Reserved	0
Bit 1	R/W	Reserved	0
Bit 0	R/W	PROV	0

PROV

The channel provision bit (PROV) specifies the active status of the channel being accessed. The PM3392 has only a single channel. When PROV is asserted high, the channel is active. When PROV is asserted low, the channel is inactive and is not used. The PROV bit is used for enabling and disabling the single channel within the EFLX, as well as to initialize the data and tag RAM read/write addresses. A transition from the channel being active (PROV is logic 1) to inactive (PROV is logic 0) will reset the rate adaptation buffer and freeze the FIFO. A transition from the channel being inactive (PROV is logic 0) to active (PROV is logic 1) will latch in the LO_LIM[9:0], HI_LIM[9:0], AFTH[13:0], AETH[13:0] and CUT_THRU[13:0] to the FIFO and begin to process data.

Register 0x3280H:PL4IDU Configuration

Bit	Type	Function	Default
Bit 15	R	Unused	X:high
Bit 14	R	Unused	X:high
Bit 13	R	Unused	X:high
Bit 12	R	Unused	X:high
Bit 11	R	Unused	X:high
Bit 10	R	Unused	X:high
Bit 9	R/W	Reserved	0:high
Bit 8	R/W	Reserved	0:high
Bit 7	R/W	Reserved	0:high
Bit 6	R/W	Reserved	0
Bit 5	R/W	Reserved	0
Bit 4	R/W	Reserved	0
Bit 3	R/W	Reserved	0
Bit 2	R/W	SYNCH_ON_TRAIN	1
Bit 1	R/W	EN_PORTS	0
Bit 0	R/W	EN_DFWD	0

EN_DFWD

Enable Data Forward. This bit is used to allow PL4 data words to be forwarded from the PL4IDU. When ‘1’, data will be forwarded. When ‘0’, data will not be forwarded. The assertion and deassertion can result in partial packets. Port Statistics are incremented only if EN_DFWD is a ‘1’.

EN_PORTS

Enable Port State machine. This bit is used to allow PL4 data words to be forwarded from the PL4IDU. When ‘1’, data will be forwarded. When ‘0’, data will not be forwarded. The assertion and deassertion of this signal will not result in partial packets. Port Statistics are incremented only if EN_PORTS is a ‘1’.

SYNCH_ON_TRAIN

Should be set to 1.

Register 0x3281H:PL4IDU Status

Bit	Type	Function	Default
Bit 15		Reserved	X:high
Bit 14	R	Reserved	X:high
Bit 13	R	Reserved	X:high
Bit 12	R	Reserved	X:high
Bit 11	R	Reserved	0
Bit 10	R	Reserved	0
Bit 9	R	Reserved	0
Bit 8	R	Reserved	0
Bit 7	R	Unused	X:high
Bit 6	R	Reserved	
Bit 5	R	RUN_ORANGE	
Bit 4	R	RUN_YELLOW	0
Bit 3	R	RUN_GREEN	0
Bit 2	R	PL4IDU_DISABLE	1
Bit 1	R	INT	0
Bit 0	R	Reserved	0

PL4IDU_DISABLE

This is a registered value of the Link State PL4IDU_DISABLE. This bit is provided for diagnostic purposes only.

RUN_GREEN

This is a registered value of the Link State RUN_GREEN. This bit is provided for diagnostic purposes only.

RUN_YELLOW

This is a registered value of the Link State RUN_YELLOW. This bit is provided for diagnostic purposes only.

RUN_ORANGE

This is a registered value of the Link State RUN_ORANGE. This bit is provided for diagnostic purposes only.

Register 0x3282H:PL4IDU Interrupt Mask

Bit	Type	Function	Default
Bit 15	R	Unused	X:high
Bit 14	R	Unused	X:high
Bit 13	R	Unused	X:high
Bit 12	R	Unused	X:high
Bit 11	R	Unused	X:high
Bit 10	R	Unused	X:high
Bit 9	R	Unused	X:high
Bit 8	R/W	Reserved	0:high
Bit 7	R/W	Reserved	0:high
Bit 6	R/W	Reserved	0:high
Bit 5	R/W	Reserved	0:high
Bit 4	R/W	Reserved	0
Bit 3	R/W	Reserved	0
Bit 2	R/W	Reserved	0
Bit 1	R/W	DIP4E	0
Bit 0	R/W	IND_VALIDE	0

IND_VALIDE

The IND_VALIDE bit enables the generation of an interrupt due to the primary input signal IND_VALID transitioning from a ‘1’ to a ‘0’.

DIP4E

The DIP4E bit enables the generation of an interrupt due to a DIP4 check error on a PL4 control word.

Register 0x3283H:PL4IDU Interrupt

Bit	Type	Function	Default
Bit 15	R	Unused	X:high
Bit 14	R	Unused	X:high
Bit 13	R	Unused	X:high
Bit 12	R	Unused	X:high
Bit 11	R	Unused	X:high
Bit 10	R	Unused	X:high
Bit 9	R	Unused	X:high
Bit 8	R	Reserved	0:high
Bit 7	R	Reserved	0:high
Bit 6	R	Reserved	0:high
Bit 5	R	Reserved	0:high
Bit 4	R	Reserved	
Bit 3	R	Reserved	0
Bit 2	R	Reserved	0
Bit 1	R	DIP4I	0
Bit 0	R	IND_VALIDI	0

IND_VALIDI

The IND_VALIDI bit will be set when the primary input signal IND_VALID transitions from a ‘1’ to a ‘0. If IND_VALIDE is ‘1’ in the PL4IDU Interrupt Mask register, an interrupt will also be generated (INT output asserted).

DIP4I

The DIP4I bit will be set when there is a DIP4 check error. The DIP4 checking is only done if the primary input signal IND_VALID is asserted on the PL4 input interface. If DIP4E is ‘1’ in the PL4IDU Interrupt Mask register, an interrupt will also be generated (INT output asserted).

12 Test Features Description

The following section will describe the test features implemented in the PM3392.

12.1 High Impedance State for IO

Test features to facilitate board testing include being able to place all output pins, including analog, in a high-impedance state. To achieve a high impedance state for the digital outputs the following steps can be taken:

Assert CSB, RSTB, and TRSTB to “0”

De-assert TRSTB to “1” after 200ns

Write 0x0010H to register 0x4000H

The combination of bit 4 of register 0x4000H and RSTB low, sets the digital output to a HIZ state. TRSTB is an asynchronous reset to register 0x4000H, therefore must be deasserted before the register write.

Another way to set digital outputs to high-impedance is to simultaneously assert (low) the CSB, RDB and WRB inputs. All digital output pins and the data bus (d[15:0]) will be held in a high-impedance state. This test feature may be used for board testing.

To set the analog output pins to high impedance, the following steps can be taken:

Assert CSB, RSTB, and TRSTB to “0”

De-assert RSTB and TRSTB to “1” after 200ns

Write 0x2000H to register 0x2305H

12.2 Test Registers

PM3392 Test Register 0

Bit	Type	Function	Default
Bit 15	R	Unused	0
Bit 14	R	Unused	0
Bit 13	R	Unused	0
Bit 12	R	Unused	0
Bit 11	R	Unused	0
Bit 10	R	Unused	0
Bit 9	R	Unused	0
Bit 8	R	Unused	0
Bit 7	R	Unused	0
Bit 6	R	Unused	0
Bit 5	R	Unused	0
Bit 4	R/W	PMCTSTB	0
Bit 3	R/W	Reserved	0
Bit 2	R	Unused	0
Bit 1	R/W	Reserved	0
Bit 0	R/W	HIZIO	0

HIZIO

HIZIO works in conjunction with WRB, CSB, and RDB. When these signals are low together and HIZIO is written to '1', the digital pins are set to a high impedance state. This does not effect the analog pins.

PMCTSTB

PMCTSTB is used for placing the device in test mode to be used for production testing.

PM3392 Test Register 4

Bit	Type	Function	Default
Bit 15	R	Unused	0
Bit 14	R	Unused	0
Bit 13	R	Unused	0
Bit 12	R	Unused	0
Bit 11	R	Unused	0
Bit 10	R	Unused	0
Bit 9	R	Unused	0
Bit 8	R	Unused	0
Bit 7	R	Unused	0
Bit 6	R	Unused	0
Bit 5	R/W	VCLK6	0
Bit 4	R/W	VCLK5	0
Bit 3	R/W	VCLK4	0
Bit 2	R/W	VCLK3	0
Bit 1	R/W	VCLK2	0
Bit 0	R/W	VCLK1	0

VCLK1 – VCLK6

VLCK is used as test clock inputs to help with production test vectors.

12.3 JTAG Test Port

The S/UNI-1x10GE(PM3392) JTAG Test Access Port (TAP) allows access to the TAP controller and the 4 TAP registers: instruction, bypass, device identification and boundary scan. Using the TAP, device input logic levels can be read, device outputs can be forced, the device can be identified and the device scan path can be bypassed. For more details on the JTAG port, please refer to section 13.

Table 17 Instruction Register (Length - 3 Bits)

Instructions	Selected Register	Instruction Codes, IR[2:0]
EXTEST	Boundary Scan	000
IDCODE	Identification	001
SAMPLE	Boundary Scan	010
BYPASS	Bypass	011
BYPASS	Bypass	100
STCTEST	Boundary Scan	101
BYPASS	Bypass	110
BYPASS	Bypass	111

Table 18 Identification Register

Length	32 bits
Version Number	3H
Part Number	3392H
Manufacturer's Identification Code	0CDH
Device Identification	333920CDH

Table 19 Boundary Scan Register

Name	Register Bit	Cell Type	Device ID
RCLK	171	IN_CELL	-
RDAT[1]	170	OUT_CELL	-
RDAT[0]	169	OUT_CELL	-
RDAT[3]	168	OUT_CELL	-
RDAT[2]	167	OUT_CELL	-
RDAT[5]	166	OUT_CELL	-
RDAT[4]	165	OUT_CELL	-
RDAT[7]	164	OUT_CELL	-
RDAT[6]	163	OUT_CELL	-
RCTL	162	OUT_CELL	-
RDAT[8]	161	OUT_CELL	-
RDAT[9]	160	OUT_CELL	-
RDCLK	159	OUT_CELL	-
RDAT[11]	158	OUT_CELL	-
RDAT[10]	157	OUT_CELL	-
RDAT[13]	156	OUT_CELL	-
RDAT[12]	155	OUT_CELL	-
RDAT[15]	154	OUT_CELL	-
RDAT[14]	153	OUT_CELL	-
REFSEL[0]	152	IN_CELL	-
REFSEL[1]	151	IN_CELL	-
PAUSE	150	IN_CELL	-
RSCLK	149	IN_CELL	-
RSTAT[1]	148	IN_CELL	-
RSTAT[0]	147	IN_CELL	-
TXDATA4[3]	146	OUT_CELL	-
TXDATA4[1]	145	OUT_CELL	-
TXDATA4[2]	144	OUT_CELL	-
TXDATA4[0]	143	OUT_CELL	-
TXCLK4	142	OUT_CELL	-
TXCLK4_SRC	141	IN_CELL	-
TXDATA3[3]	140	OUT_CELL	-

Name	Register Bit	Cell Type	Device ID
TXDATA3[1]	139	OUT_CELL	-
TXDATA3[2]	138	OUT_CELL	-
TXDATA3[0]	137	OUT_CELL	-
TXCLK3	136	OUT_CELL	-
TXCLK3_SRC	135	IN_CELL	-
TXCLK2	134	OUT_CELL	-
TXCLK2_SRC	133	IN_CELL	-
TXDATA2[3]	132	OUT_CELL	-
TXDATA2[1]	131	OUT_CELL	-
TXDATA2[2]	130	OUT_CELL	-
TXDATA2[0]	129	OUT_CELL	-
TXCLK1	128	OUT_CELL	-
TXCLK1_SRC	127	IN_CELL	-
TXDATA1[3]	126	OUT_CELL	-
TXDATA1[1]	125	OUT_CELL	-
TXDATA1[2]	124	OUT_CELL	-
TXDATA1[0]	123	OUT_CELL	-
RXDATA4[3]	122	IN_CELL	-
RXDATA4[1]	121	IN_CELL	-
RXDATA4[2]	120	IN_CELL	-
RXDATA4[0]	119	IN_CELL	-
RXDATA3[3]	118	IN_CELL	-
RXCLK4	117	IN_CELL	-
RXDATA3[2]	116	IN_CELL	-
RXDATA3[0]	115	IN_CELL	-
RXDATA3[1]	114	IN_CELL	-
RXCLK3	113	IN_CELL	-
RXDATA2[3]	112	IN_CELL	-
RXCLK2	111	IN_CELL	-
RXDATA2[2]	110	IN_CELL	-
RXDATA2[0]	109	IN_CELL	-
RXDATA2[1]	108	IN_CELL	-
RXCLK1	107	IN_CELL	-
RXDATA1[3]	106	IN_CELL	-
RXDATA1[1]	105	IN_CELL	-
RXDATA1[2]	104	IN_CELL	-
RXDATA1[0]	103	IN_CELL	-
DTRB	102	IN_CELL	-
OEB_TSCLK	101	OUT_CELL	-
TSCLK	100	OUT_CELL	-

Name	Register Bit	Cell Type	Device ID
OEB_TSTAT[1]	99	OUT_CELL	-
TSTAT[1]	98	OUT_CELL	-
OEB_TSTAT[0]	97	OUT_CELL	-
TSTAT[0]	96	OUT_CELL	-
OEB_INTB	95	OUT_CELL	-
INTB	94	OUT_CELL	-
WRB	93	IN_CELL	-
RSTB	92	IN_CELL	-
CSB	91	IN_CELL	-
RDB	90	IN_CELL	-
A[0]	89	IN_CELL	-
A[1]	88	IN_CELL	-
A[3]	87	IN_CELL	-
A[9]	86	IN_CELL	-
A[13]	85	IN_CELL	-
A[2]	84	IN_CELL	-
A[5]	83	IN_CELL	-
A[4]	82	IN_CELL	-
A[7]	81	IN_CELL	-
A[14]	80	IN_CELL	-
OEB_D[1]	79	OUT_CELL	-
D[1]	78	IO_CELL	-
A[6]	77	IN_CELL	-
A[10]	76	IN_CELL	-
A[11]	75	IN_CELL	-
OEB_D[0]	74	OUT_CELL	-
D[0]	73	IO_CELL	-
OEB_D[2]	72	OUT_CELL	-
D[2]	71	IO_CELL	-
OEB_D[5]	70	OUT_CELL	-
D[5]	69	IO_CELL	-
A[8]	68	IN_CELL	-
A[12]	67	IN_CELL	-
ALE	66	IN_CELL	-
OEB_D[3]	65	OUT_CELL	-
D[3]	64	IO_CELL	-
OEB_D[6]	63	OUT_CELL	-
D[6]	62	IO_CELL	-
OEB_D[9]	61	OUT_CELL	-
D[9]	60	IO_CELL	-

Name	Register Bit	Cell Type	Device ID
OEB_D[4]	59	OUT_CELL	-
D[4]	58	IO_CELL	-
OEB_D[7]	57	OUT_CELL	-
D[7]	56	IO_CELL	-
OEB_D[10]	55	OUT_CELL	-
D[10]	54	IO_CELL	-
OEB_D[13]	53	OUT_CELL	-
D[13]	52	IO_CELL	-
OEB_D[15]	51	OUT_CELL	-
D[15]	50	IO_CELL	-
OEB_D[11]	49	OUT_CELL	-
D[11]	48	IO_CELL	-
OEB_D[8]	47	OUT_CELL	-
D[8]	46	IO_CELL	-
OEB_MDIO	45	OUT_CELL	-
MDIO	44	IO_CELL	-
OEB_MDC	43	OUT_CELL	-
MDC	42	OUT_CELL	-
VCLK2	41	IN_CELL	-
VCLK1	40	IN_CELL	-
OEB_D[12]	39	OUT_CELL	-
D[12]	38	IO_CELL	-
VCLK3	37	IN_CELL	-
VCLK4	36	IN_CELL	-
VCLK5	35	IN_CELL	-
VCLK6	34	IN_CELL	-
VCLK_FORCEB	33	IN_CELL	-
OEB_RX_LOS	32	OUT_CELL	-
RX_LOS	31	OUT_CELL	-
OEB_D[14]	30	OUT_CELL	-
D[14]	29	IO_CELL	-
OEB_PAUSED	28	OUT_CELL	-
PAUSED	27	OUT_CELL	-
PHASE_ERR	26	IN_CELL	-
OEB_PHASE_INIT	25	OUT_CELL	-
PHASE_INIT	24	OUT_CELL	-
SYNC_ERR	23	IN_CELL	-
OEB_RX_SYSCLK2	22	OUT_CELL	-
RX_SYSCLK2	21	OUT_CELL	-
OEB_TX_SYSCLK2	20	OUT_CELL	-

Name	Register Bit	Cell Type	Device ID
TX_SYSCLK2	19	OUT_CELL	-
CPREF_CLK	18	IN_CELL	-
TDAT[0]	17	IN_CELL	-
TDAT[2]	16	IN_CELL	-
TDAT[4]	15	IN_CELL	-
TDAT[6]	14	IN_CELL	-
TDAT[8]	13	IN_CELL	-
TDAT[9]	12	IN_CELL	-
TDAT[11]	11	IN_CELL	-
TDAT[13]	10	IN_CELL	-
TDAT[15]	9	IN_CELL	-
TDAT[1]	8	IN_CELL	-
TDAT[3]	7	IN_CELL	-
TDAT[5]	6	IN_CELL	-
TDAT[7]	5	IN_CELL	-
TDCLK	4	IN_CELL	-
TDAT[10]	3	IN_CELL	-
TDAT[12]	2	IN_CELL	-
TDAT[14]	1	IN_CELL	-
TCTL	0	IN_CELL	-

Note 1: When set high, INTB will be set to high impedance.

Note 2: Each output cell has its own output enable (OEB_*), except for the differential outputs.

Note 3: TCTL is the first bit in the boundary scan chain, and RCLK is the first bit out of the boundary scan chain.

Note 4: It is assumed that the differential I/O will drive or will be driven differentially. A logic 1 being driven on a differential output via the JTAG scan chain will cause a logic 1 to be driven onto "positive" or "_P" pin, and a logic 0 to be driven onto the "negative" or "_N" pin. A similar convention is true for differential inputs.

Boundary Scan Cells

In the following diagrams, CLOCK-DR is equal to TCK when the current controller state is SHIFT-DR or CAPTURE-DR, and unchanging otherwise. The multiplexer in the center of the diagram selects one of four inputs, depending on the status of select lines G1 and G2. The ID Code bit is as listed in the Boundary Scan Register table located above.

Figure 13 Input Observation Cell (IN_CELL)

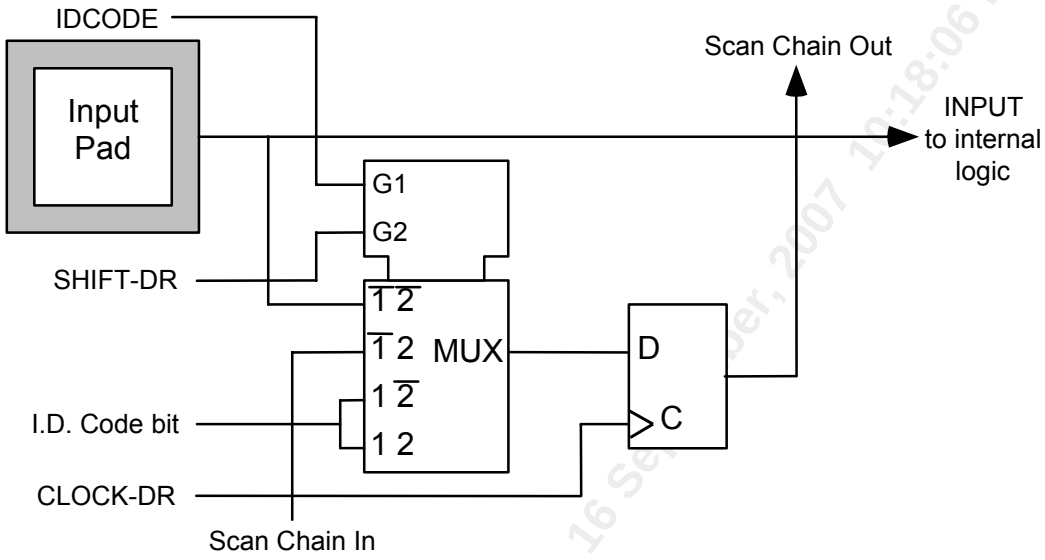


Figure 14 Output Cell (OUT_CELL)

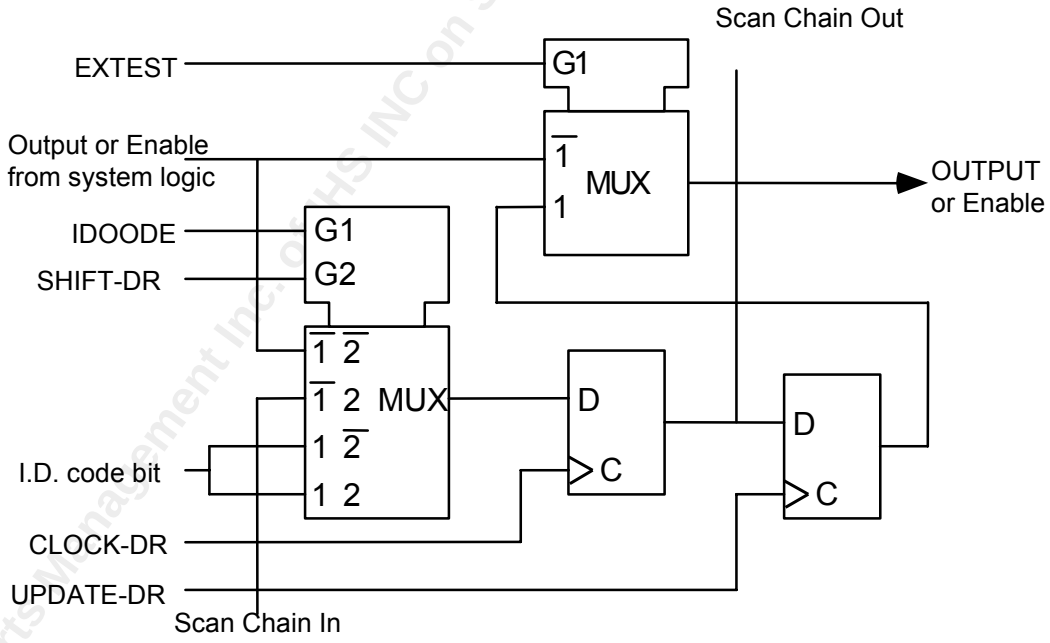


Figure 15 Bi-directional Cell (IO_CELL)

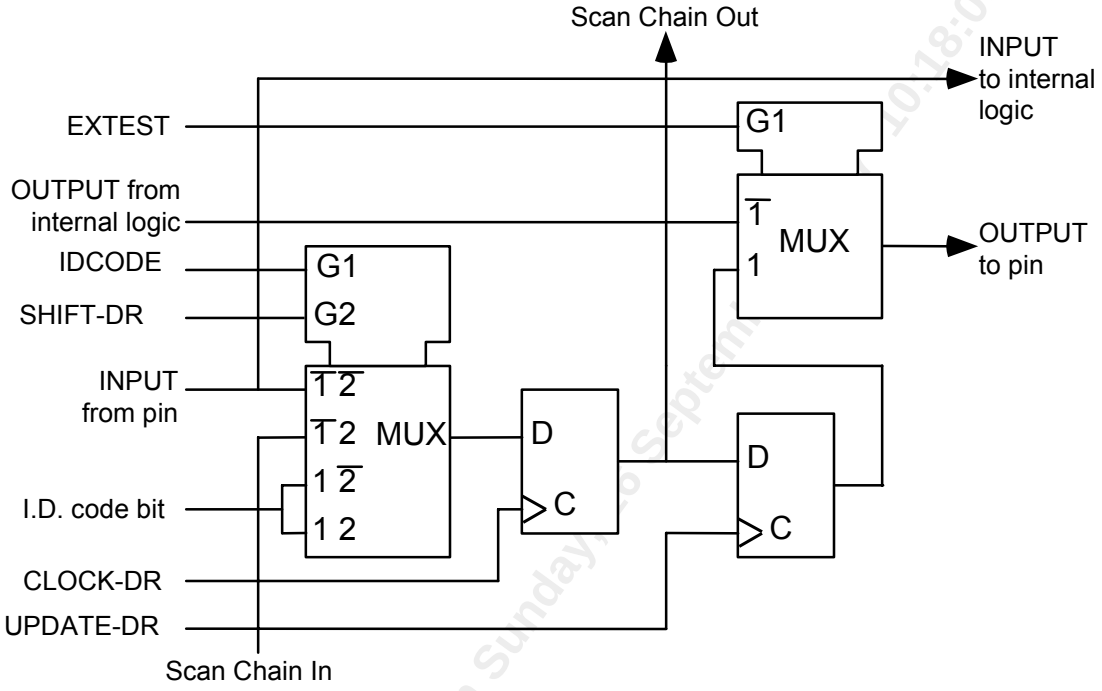
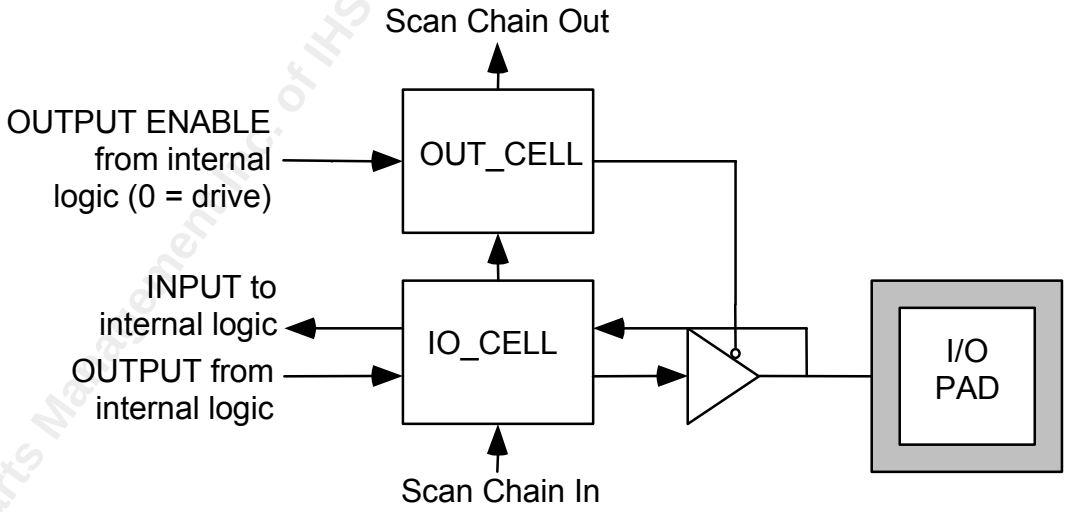


Figure 16 Layout of Output Enable and Bi-directional Cells



13 Operation

13.1 Power Sequencing

The PM3392 uses five separate power sources: VDDO, VDDI, AVDHVREF, PL4_AVDH, and PL4_AVDL. The PM3392 specifies the ground pins VSS.

The analog high power, PL4_AVDH, must be connected to a properly de-coupled +3.3V supply. The analog high power, AVDHVREF, must be connected to a properly de-coupled +3.3V supply. The analog low power,PL4_ AVDL, must be connected to a properly de-coupled +1.8V supply. The digital switching power pins, VDDO, must be connected to a properly de-coupled +3.3V supply. The digital core power, VDDI, must be connected to a properly de-coupled +1.8V supply. The digital and analog power pins that are of the same supply voltage can be sourced from the same physical power supply source.

The ground pins can be connected to a common uninterrupted physical ground plane. The digital core and switching power pins, VDDO and VDDI, are to be de-coupled to the VSS ground. Each analog power pin is to be independently de-coupled to the ground plane.

The power-on sequence is as follows:

- VDDO must come up before or simultaneously with PL4_AVDH
- AVDHVREF and PL4_AVDH must come up together and before or simultaneously with PL4_AVDL
- VDDO must come up before or simultaneously with VDDI

13.2 Device Reset

The reset pin of the PM3392 device (RSTB – active low) should be asserted for at least 1 ms to initiate a complete initialization, or re-initialization, of the device. While RSTB is held low (logic 0) both the digital and the analog portions of the chip are being reset. Before the de-assertion of RSTB, it is required that all external clocks and REFSEL be stable for a minimum of 1ms.

After de-assertion of the RSTB pin the device will continue to hold its internal digital reset asserted until an internal timer expires after 10 to 14 ms.. This will allow the analog clock synthesizer (CSU) for the high-speed device interface (the PL4 interface) to stabilize to the selected reference frequency before allowing the digital portions of the device to operate. The line side or XSBI interface recovers internal clocks by dividing down the supplied reference clocks. The divided clocks are stable with in a 200ns. To override the internal initialization sequencing and timer, DTRB can be toggled high. This is only used for chip testing and debug. It is not recommended that DTRB be toggled by the user. This pin should be tied low at the board level.

When the RSTB pins is de-asserted (logic 1), the device reset can be controlled by writing to the Configuration and Reset Control Register. System status of analog training and progress can be viewed via the top level PM3392 **Device Status** register.

The system programmer may also elect to reset the PM3392 via software commands. This is accomplished by writing to the PM3392 **Device Configuration and Reset Control** register. The programmer is to write the PL4_ARESETB, XSBI ARESETB and DRESETB bits to a logic 0. This asserts a full device reset. The programmer must pause no less than 1ms (there is no upper limit) then de-assert by writing a logic 1 to PL4_ARESETB and XSBI_ARESETB. The programmer is to wait no less than 10ms (there is no upper limit) then de-assert DRESETB by writing a logic 1 to the **Device Configuration and Reset Control** register DRESETB bit. As with assertion of the RSTB pin the programmer must also insure that the REFSEL[1:0] pins are in a stable state and that all clocks for the device are present for a minimum of 1ms prior to initiating a software reset sequence. Note that the internal 10ms digital reset delay timer is only initiated after an appropriate RSTB pin reset sequence. Asserting software reset via ARESETB or DRESETB will not properly sequence the delay timer. It is not recommended that software attempt a full device reset. However, a soft reset may be accomplished by de-asserting DRESETB then re-asserting DRESETB. This will reset all internal digital logic without effecting the clocks. Block configuration

After the device reset and 10ms wait time for clocks to become stable, it is recommended to configure and enable all blocks except the XSBI and PL4IO. Once all other blocks are configured and enabled, then the XSBI and the PL4IO blocks can be configured and enabled. This allows for a clean start-up of the line and system interfaces.

13.3 Line-Side LVDS Interface Overview

A generic LVDS link is implemented accordance with IEEE 1596.3-1996. The transmitter drives a differential signal through a pair of 50Ω characteristic interconnects, such as board traces, back plane traces, or short lengths of cable. The receiver presents a 100Ω differential termination impedance to terminate the lines. Included in the standard is sufficient common-mode range for the receiver to accommodate as much as 925mV of common-mode ground difference.

13.4 POS-PHY Level 4 Introduction

The PM3392 device, one of PMC-Sierra's 10 Gigabit physical layer devices, utilizes the Industry standard POS-PHY Level 4 system interface to provide a common interface across multiple devices supporting various protocols and rates with an aggregate throughput of 12.5Gbit/s.

The POS-PHY Level 4 system interface consists of a pair of 16 bit data paths and a pair of 2 bit FIFO status paths. For a detailed description of this interface please refer to the POS-PHY Level 4 specification. Both the transmit and receive data paths operate at the same frequency (*i.e.* RDCLK frequency = TDCLK frequency).

The POS-PHY Level 4 interface FIFO status paths typically operate at 1/8th the data path-rate with LVCMOS I/O. The transmit FIFO status path can operate at any speed up to 1/8th the data path rate. The receive FIFO status path operates at 1/8th the data path rate.

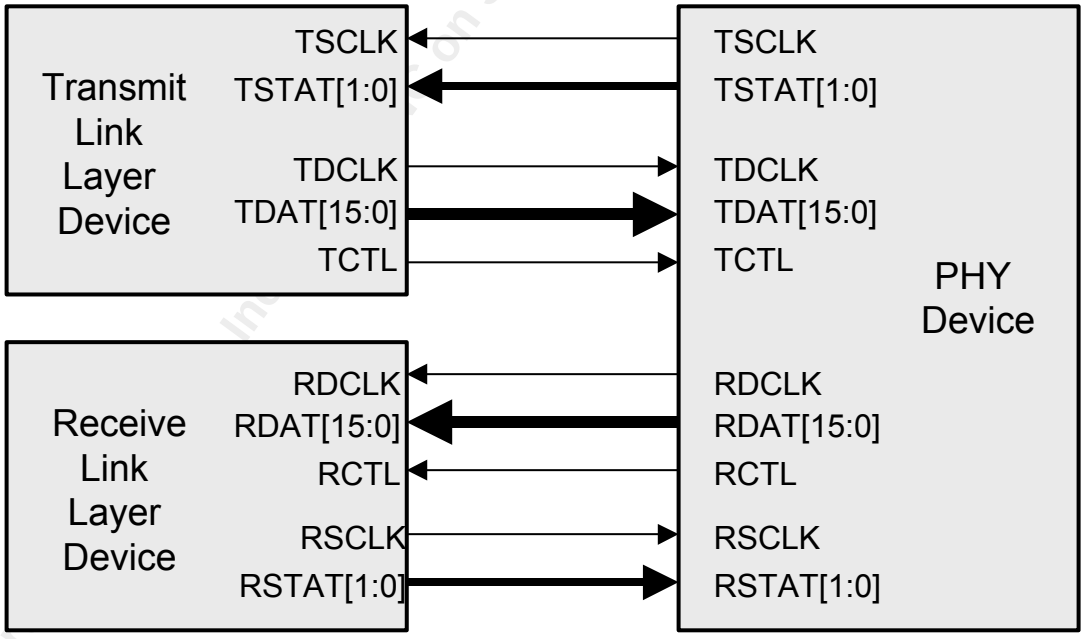
The terms 'egress', 'transmit', 'ingress' and 'receive' denote the system relative flow of status or data across the POS-PHY Level 4 interfaces whereas the terms 'in', 'input', 'out' and 'output' denote the device relative flow of status or data across the device interfaces.

Figure 17 shows the relationships between devices communicating via the POS-PHY Level 4 interface. On the PM3392, the transmit interface consists of TSCLK / TSTAT outputs and TDCLK / TCTL / TDAT inputs, and the receive interface consists of RSCLK / RSTAT inputs and RDCLK / RCTL / RDAT outputs. For link layer devices the transmit interface consists of TSCLK / TSTAT inputs and TDCLK / TCTL / TDAT outputs, and the receive interface consists of (optional) RSCLK / RSTAT outputs and RDCLK / RCTL / RDAT inputs.

13.4.1 External Components Required for PL4 Bus Interface on the PM3392

The analog power pins on the PM3392 device that are associated with the PL4 Bus interface must be properly decoupled. Please refer to PMC application note PMC-2010770 (Power Supply Filtering Recommendations for XENON Devices) for additional details.

Figure 17 POS-PHY Level 4 Interfaces



13.5 POS-PHY Level 4 Clocking

13.5.1 POS-PHY Level 4 Clocking Modes

The POS-PHY Level 4 interface supports two data path-clocking modes: a master and a slave mode. In master mode, the PMC-Sierra physical layer device (PHY) derives the PL4 clock from reference input (REFCLK) and provides clocking to the customer ASIC. In slave mode, the PHY derives the PL4 clock from the transmit data clock (TDCLK) received from the customer ASIC.

The choice between master and slave clocking modes depends on board design considerations and the jitter performance expected on the customer ASIC device.

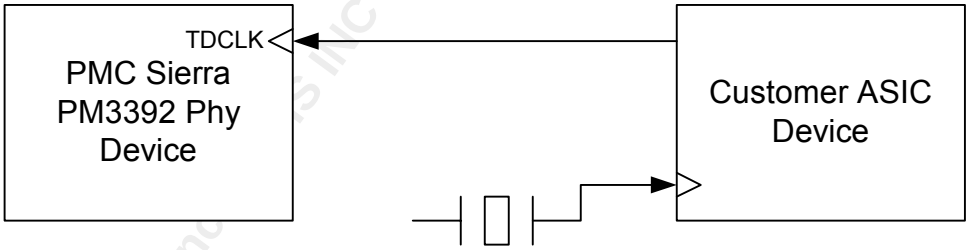
The PL4 interface needs to be frequency locked regardless of which mode is chosen.

Slave Clocking Mode

When in Slave clocking mode the PMC-Sierra PHY derives its data clocks from the transmit data clock (TDCLK) received from the customer ASIC device.

Figure 18 shows a PMC-Sierra PHY device connected to a customer ASIC device in slave clocking mode.

Figure 18 POS-PHY Level 4 Slave Clocking Mode



Master Clocking Mode

When in Master clocking mode the PMC-Sierra PHY derives its data clocks directly from an external reference clock (REFCLK) and provides a frequency reference to the customer ASIC device via the receive data clock (RDCLK).

Figure 19 shows a PMC-Sierra PHY device connected to a customer ASIC device in master clocking mode.

Figure 19 POS-PHY Level 4 Master Clocking Mode

13.5.2 POS-PHY Level 4 Clock Mode Configuration

This section describes how to configure the clocking mode for the POS-PHY Level 4 interface.

The choice of clocking mode is indicated to the PM3392 device by the REFSEL[0] input pin on the device as follows:

0 = Slave mode (TDCLK is the frequency reference)

1 = Master mode (REFCLK is the frequency reference)

In slave mode the availability of the transmit data clock (TDCLK) is indicated to the PM3392 device by the REFSEL[1] input pin on the device as follows:

0 = TDCLK is valid when the RSTB pin is de-asserted (immediate mode)
Clock initialization proceeds automatically.

1 = TDCLK is not valid when the RSTB pin is de-asserted (deferred mode)
Software intervention is required to complete clock initialization.

In master mode the frequency of the reference clock (REFCLK) is indicated to the PM3392 device by the REFSEL[1] input pin on the device as follows:

0 = ½ the data rate (311 to 350 MHz)

1 = ¼ the data rate (155.5 to 175 MHz)

In master mode the REFCLK must be available when the RSTB pin is de-asserted (*i.e.* there is no deferred master mode).

The REFSEL settings must not change after the RSTB pin is de-asserted.

13.5.3 PL4 Bus Clock Frequency Selection for Non-Blocking Operation

The required minimum PL4 Bus reference clock required to support non-blocking operation of the PM3392 (that is, to support continuous line-rate Ethernet frame data transfer in both the ingress and egress directions) is a function of numerous factors including the following:

- The maximum size of Ethernet frames in the application.
- The selection of the PL4 maximum burst length (for the PM3392, this is configurable in PL4MOS MAX_TRANSFER register)
- The minimum SOP-to-SOP spacing rule (configurable in the PL4ODP Configuration register).
- The PL4 Training sequence frequency (configurable in the PL4ODP MAX_T register

The following table lists the minimum and maximum supported PL4 Bus data transfer rates to support non-blocking operation. It is recommended that the PM3392 be operated with the PL4 Bus transferring data at 700 Mega-cycles per second.

Table 20 PL4 Bus Data Transfer Rate For Non-Blocking Operation

Maximum Ethernet Frame Size (In Bytes)	Minimum PL4 Bus Data Transfer Rate (Mcps)
1518 (standard 802.3)	660
9600 (jumbo)	680

In the table above, the Maximum Ethernet Frame size is for an untagged frame; a tagged frame will have a frame size 4 octets larger and is already taken into consideration. The MAX_TRANSFER size programmed is assumed to be 128 bytes or greater. The MAX_T value is assumed to be programmed to at least 64 for non-blocking operation.

PL4 Data Alignment modes (Dynamic versus Static)

The egress data path of the PM3392 (TCTL+/- and TDAT[15:0]+/- pins) only supports the PL4 Bus specification dynamic alignment mode. This requires that the peer PL4 device must send the PL4 training sequence at a certain minimum rate. The minimum rate should be 4000 to 10000 PL4 bus clock cycles or 5.7us to 14.3us. It is also required that the peer PL4 device comply to the dynamic mode jitter specifications that will also be referenced in the application note.

The ingress data path of the PM3392 (RCTL +/- and RDAT[15:0]+/-) support both PL4 Bus specification static and dynamic alignment modes. Please refer to PMC-2010198 “PMC PL4 Compliance Statement” for a detailed application note. In static mode, training is disabled. Also, in static alignment mode, careful attention must be paid to skew and jitter specification as outlined by the PL4 Bus specification.

13.6 POS-PHY Level 4 Initialization

Initialization of PMC-Sierra’s POS-PHY Level 4 interface proceeds in several phases:

1. Device reset (previously discussed)
2. Clock acquisition
3. Training

4. Configuration

5. Enabling

These phases represent the conceptual initialization sequence. Depending on the configured mode of operation there may be some overlap between phases.

13.6.1 Clock Acquisition

Clock acquisition begins during device reset with the de-assertion of the RSTB pin. In immediate (master or slave) mode the selected reference clock (REFCLK or TDCLK) should be valid at this point, and the CSU will begin training to the supplied reference frequency.

In deferred (slave) mode the transmit data clock (TDCLK) input should be active but may not be operating at the correct frequency, *e.g.* because it is being driven from a CSU which is still in frequency acquisition. In this case the training of the slave CSU should track the training of the master CSU with a moderate amount of time lag.

Completion of clock acquisition is determined by monitoring the frequency of the CSU relative to the transmit data clock (TDCLK) in both slave and master modes, and relative to the reference clock (REFCLK) in master mode only. In immediate (master or slave) mode the frequency lock detection functions are enabled immediately upon de-assertion of the device internal digital reset. At this point the CSU should be properly trained to the selected reference frequency. The frequency lock detection functions should indicate reference in-lock within 500ms after the device internal digital reset is de-asserted.

In deferred (slave) mode the output of the lock detect function must be ignored until the master device is driving a valid frequency reference (TDCLK) into the slave device, and the slave CSU has had time to acquire frequency lock. This is accomplished by automatically setting two configuration flags (TRAIN_DIS and ODAT_DIS) at reset time when REFSEL[1:0] = 10, and delaying the training phase until software clears these flags, as described below.

13.6.2 Training

Both the transmit data path (TCTL / TDAT) and the receive data path (RCTL / RDAT) are de-skewed during this phase. During training the output end of the data path continuously sends the training pattern defined in the POS-PHY Level 4 standard (10 Training Control Words / 10 Training Data Words) to the input end.

Training proceeds somewhat independently in each direction. Transmit data path training begins when the TRAIN_DIS flag is cleared (by the device in immediate mode or by software in deferred mode) and the frequency lock detect function indicates that the CSU is locked to the transmit data clock (TDCLK), and in master mode only to the reference clock (REFCLK). During training the input data de-skew function aligns the 17 parallel bit lanes of the transmit data interface to each other. The transmit data alignment function should indicate data in-lock within 6 μs after transmit data path training begins when a valid training pattern is present at the transmit data interface (TCTL / TDAT). Transmit data path training continues until the transmit data path is enabled, as described below.

Receive data path training begins when the ODAT_DIS flag is cleared (by the device in immediate mode or by software in deferred mode) and the frequency lock detect function indicates that the CSU is locked to the selected frequency reference (TDCLK or REFCLK). During training the device continuously generates the training pattern at the receive data interface (RCTL / RDAT) to allow the input end of the interface in the peer device to properly de-skew. Receive data path training continues until the receive data path is enabled, as described below.

13.6.3 Configuration

The operation of PMC-Sierra's POS-PHY Level 4 interface requires several configuration parameters, including:

1. Highest channel number used by the device (Calendar_Len)
2. Transmit and receive FIFO allocations and thresholds for each channel.
3. Receive data scheduling parameters (Max_Burst1, Max_Burst2, Max_Transfer)
4. Receive training scheduling parameters (Data_Max_T, Alpha)
5. Optional disabling of receive FIFO status channel (NO_STATUS flag)

It is recommended that parameters be configured in the order shown to avoid anomalous behavior. Rules for setting the PL4 Bus configuration parameters is described in a later section.

13.6.4 Enabling

Enabling proceeds somewhat independently in each direction. After configuring all transmit interface parameters software must clear the OSTAT_DIS flag. (As described in the training phase, the TRAIN_DIS flag must also have been cleared automatically or by software.) Normal operation of the transmit FIFO status and data interfaces commences when the OSTAT_DIS flag is cleared and the transmit data alignment function indicates data in-lock.

After configuring all receive interface parameters software must either set the NO_STATUS flag or clear the ISTAT_DIS flag. (As described in the training phase, the ODAT_DIS flag must also have been cleared automatically or by software.) Normal operation of the receive FIFO status interface commences when the ISTAT_DIS flag is cleared and the receive status alignment function indicates status in-lock. Normal operation of the receive data interface commences when the NO_STATUS flag is set or the receive status alignment function indicates status in-lock.

13.6.5 PL4 Bus Configuration Parameters

The PL4 Bus specification specifies a number of startup parameters that are configurable on a per-interface basis. All of the PL4 Bus configuration parameters can be accessed via the microprocessor interface of the PM3392 device.

PL4 Bus CALENDAR_LEN

The PL4 Bus CALENDAR_LEN parameter is represented by the TOP_CHAN[3:0] field of the **PM3392 Configuration and Reset Control** register. The TOP_CHAN[3:0] is read only and set to 0. Since the PM3392 device has only 1 internal pipe, CALENDAR_LEN will always be 1.

PL4 Bus MaxBurst1 and MaxBurst2 For Receive

The PL4 Bus MaxBurst1 parameter is held in the **PL4MOS MaxBurst1** register. On the PM3392 device **PL4MOS MaxBurst1** can take the value 8 to 4095 and represents a credit amount in blocks of 16 bytes.

Likewise, the PL4 Bus MaxBurst2 parameter is held in the **PL4MOS MaxBurst2** register and can take the value 8 to 4095. It also represents a credit amount in blocks of 16 bytes.

As per the PL4 Bus specification, MaxBurst1 is to be programmed to be either higher than or equal to the value than MaxBurst2. On the PM3392 device there is no hardware bounds checking to ensure that this configuration requirement is met.

The selection of MaxBurst1 and MaxBurst2 in a system application is related to the size of the receive buffering. A more detailed description can be found in the PMC-2010502 POS-PHY Level 4 Frequently Asked Questions (PL4 FAQ).

PL4 Bus Maximum Burst Length (MAX_TRANSFER)

MAX_TRANSFER determines the maximum length of a PL4 data burst. The Maximum DataBurst Length parameter for output data (PM3392 device RDAT[15:0] and RCTL) is held in the **PL4MOS Transfer Size** register as MAX_TRANSFER[7:0]. MAX_TRANSFER defines the maximum size in a PL4 DataBurst in terms of 16-byte data blocks for the single pipe. The register is implemented so that the low two bits are read-only and hardwired to 2'b00, thereby constraining the configurable size to be modulo 4 (or at a 64-byte boundary). For example, a MAX_TRANSFER value of 8 limits the PL4 DataBurst to transfers of at most 128 bytes or, equivalently, 64 PL4 Bus cycles.

The Maximum Burst Length parameter for incoming data (device pins TDAT[15:0] and TCTL) is not held on the PM3392 and there is no requirement that the value of the parameter in the ingress and egress directions be the same. The Maximum Burst Length parameter for incoming data can range from 32 to 1024 bytes, in increments of 32 bytes.

PL4 FIFO Threshold Issues

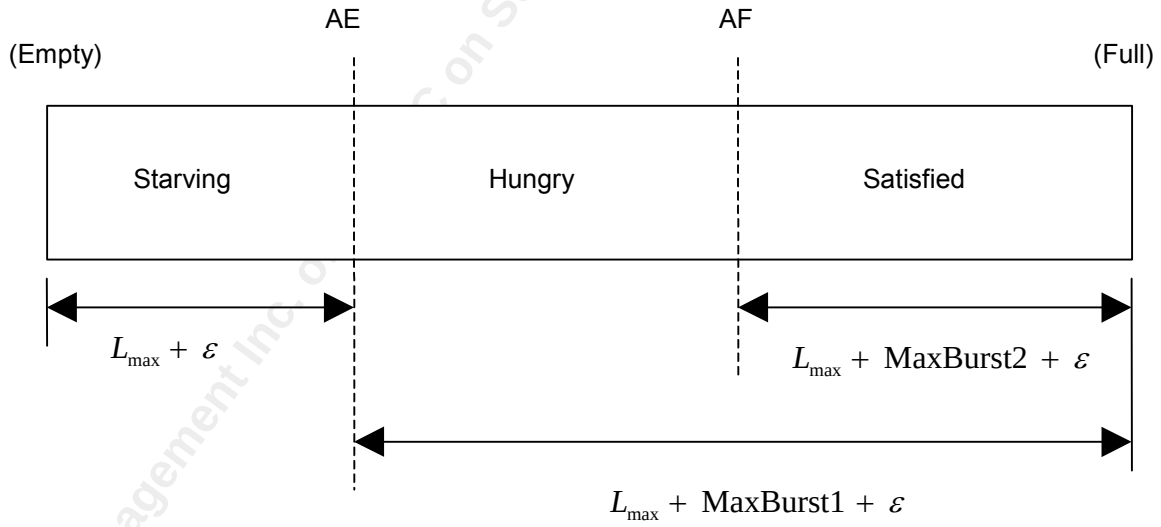
Appendix C of PMC-991635 discusses the FIFO status bandwidth requirements and FIFO threshold issues. Further details are provided below as they apply to the PM3392 device.

The thresholds for STARVING and HUNGRY in the peer PL4 device are set such that the peer FIFO can accept at least MaxBurst1 and MaxBurst2 (16-byte) blocks respectively, plus an additional amount to account for feedback-response delay. In order to guard against potential buffer underflow (which could affect total receive throughput if the sink device is unable to drain its FIFO at a rate greater than the line rate), the lowest threshold must be set high enough to allow the other end to respond to transitions to the state of lowest FIFO occupancy in a reasonable length of time (to the order of the status update interval, plus scheduler response time). MaxBurst2 and MaxBurst1 (if applicable) must be provisioned to allow adequate utilization of transfer bandwidth between status updates for the given port.

Figure 20 shows one possible way for relating worst case FIFO thresholds to MaxBurst1 and MaxBurst2 as well as the response latency. For receive the receive and transmit directions, Lmax and epsilon can take different value. Lmax corresponds to the worst-case response time, starting from the delay in receiving a status update over the FIFO status channel, and observing the reaction to that update on the corresponding data path. The margin, epsilon, is an implementation-specific quantity that accounts for differences between the calculated/designed and actual Lmax.

Figure 20 Sample FIFO Thresholds

(AE = Almost Empty waterline, AF = Almost Full waterline)



The contribution to Lmax by the PM3392 device in the receive direction (data flow on RDAT pins) is approximately 1200 bytes for a MAX_TRANSFER size of 128 bytes. Increasing MAX_TRANSFER will result in an increase in Lmax and can be approximated as follows:

Lmax ~ 1200 + 3*(MAX_TRANSFER –128) bytes;

The contribution to Lmax by the PM3392 device in the transmit direction (data flow on TDAT pins) is approximately 800 bytes.

PL4 Bus SOP-to-SOP Spacing Rule

The PL4 Bus parameter that specifies the Start-Of-Packet to Start-Of-Packet spacing rule for output data is held in the **PL4ODP Configuration** register SOP_RULE[1:0] field. The PM3392 supports a configurable minimum SOP-to-SOP spacing of 2 and 8 PL4 Bus cycles on the output data (PM3392 device RDAT[15:0] and RCTL).

For PL4 Bus input data (TDAT[15:0] and TCTL) the PM3392 device requires a minimum SOP-to-SOP spacing rule of 4. If the spacing is less than 4, this can cause the FIFO to fill erratically. If the condition persist, a FIFO overflow could result.

PL4 Bus MaxBurst1 and MaxBurst2 For Transmit

The MaxBurst1 and MaxBurst2 parameters for input data are not held on the PM3392 device. These parameters are required to be implemented either as a programmable or fixed value on the peer PL4 device that sources data on the PL4 Bus TDAT[15:0] pins.

The value of MaxBurst1 and MaxBurst2 for input data and output data do not have to be equal.

PL4 Bus CALENDAR_M

The PL4 Bus CALENDAR_M parameter is represented by the IN_MUL[7:0] and OUT_MUL[7:0] fields of the PL4IO Calendar Repetitions register.

The number of repetitions of the input FIFO status channels (impressed on PM3392 device input pins RSTAT[1:0]) within a complete status calendar sequence is controlled by IN_MUL. The PL4 input status parameter CALENDAR_M = IN_MUL + 1. The total length of the input status calendar sequence = CALENDAR_LEN x CALENDAR_M. IN_MUL must be changed while the input status interface is disabled in the PL4IO Configuration register. The result of changing IN_MUL while the status interface is enabled is unspecified. IN_MUL can take the value 1 to 255 if the input FIFO status channel is enabled. As per the PL4 Bus specification, the input FIFO status channel is optional, but will only reflect a single channel status on the PM3392.

The number of repetitions of the output FIFO status channels (impressed on the PM3392 device output pins TSTAT[1:0]) within a complete status calendar sequence is controlled by OUT_MUL. The PL4 output status parameter CALENDAR_M = OUT_MUL + 1. The total length of the output status calendar sequence = CALENDAR_LEN x CALENDAR_M. OUT_MUL must be changed while the output status interface is disabled in the PL4IO Configuration register. The result of changing OUT_MUL while the status interface is enabled is unspecified. OUT_MUL can take the value 1 to 255.

PL4 Bus MAX_CALENDAR_LEN

The maximum supported value for MAX_CALENDAR_LEN follows from the maximum supported value of both CALENDAR_LEN and CALENDAR_M:

MAX_CALENDAR_LEN = max(CALENDAR_LEN) x max(CALENDAR_M);

= 10 x 256 = 2560;

PL4 Bus FIFO_MAX_T

The PL4 Bus parameter FIFO_MAX_T is the product of CALENDAR_LEN and CALENDAR_M.

PL4 Bus DATA_MAX_T

As per the PL4 Bus specification, "For the data path de-skew procedure, DATA_MAX_T is configured only on the sending side of the data paths on the transmit and receive interfaces. DATA_MAX_T need not be identical over both interfaces."

The PL4 Bus DATA_MAX_T parameter for the output data path is held in the **PL4ODP MAX_T** register. MAX_T[11:0] defines the bounded time interval over which the PL4 training sequence is to be sent. The value of MAX_T is in terms of 1024 PL4 Bus cycles. A value of 0 in the MAX_T register disables the sending of the training sequence (no training patterns sent).

Although any integer value of MAX_T can be programmed, in order to maintain the non-blocking operation of the PM3392 a maximum of 0.1% of the PL4 bus cycles should contain a forced training data pattern or training control pattern. This limits the lowest practical value for which MAX_T can be programmed (if non-zero) to 16; for a α value of 1, this would imply that at a minimum the 20 cycle training pattern would be sent every 1000 - 4000 PL4 Bus cycles.

PL4 Bus Data Training Sequence Repetitions (α)

The PL4 Bus parameter α that specifies the number of back-to-back training patterns that define a training sequence for outgoing data (PM3392 device RDAT[15:0] and RCTL) is held in the **PL4ODP Configuration** register REPEAT_T[3:0] field. Numerically:

$$\alpha = \text{REPEAT\_T} + 1;$$

where α can take the value of 1 to 16.

For incoming data (PM3392 device TDAT[15:0] and TCTL) the value of α is not held. For correct operation of the PM3392 input deskew logic, α can take on any value greater than or equal to 1.

Operation With PL4 Bus Receive FIFO Status Unimplemented

The PM3392 device supports the PL4 Bus specification requirement that the receive FIFO status is optional. In the event that the peer PL4 sink device does not implement the receive FIFO status, the NO_STATUS bit in the **PL4MOS Configuration** register is to be written to a logic 1 prior to enabling the PL4MOS (which is done by writing PL4MOS_EN to a logic 1). In this case, each the PM3392 is assigned an infinite credit. The scheduler on the PL4MOS works as previously described but with an implied FIFO status of STA (starving).

13.7 PL4 Bus Operation

This section provides some additional operational details of the PL4 Bus interface on the PM3392 device.

13.7.1 PL4IO Initialization

After power up the PL4IO functional block on the PM3392 device must be properly initialized to ensure normal operation. The normal initialization sequence is:

- The device is reset, either through assertion of the RSTB pin or by a software reset via the ARESET and DRESET register bits, as previously described (see Device Reset).
- The PL4IO control logic trains the on-chip PL4 CSU to the selected reference clock. The Reference Clock Select (REFSEL[1:0]) selects the source (PL4_RCLK in master mode or TDCLK in slave mode) and frequency of the reference clock used to generate the internal clocks for the entire PL4IO subsystem. Training to the reference clock begins when the internal analog reset is deasserted and continues until reference lock has been detected.

  The PL4IO control logic continuously checks for reference lock while digital reset is deasserted, and generates an internal LOCKED status for the CSU when the synthesized CSU clocks are locked to the selected reference frequency. The PL4IO control logic also updates an internal LOCKED status for the Input Data when the input data clock (TDCLK) is locked to the synthesized CSU clock frequency
- The PL4IO control logic trains the on-chip DRU to the input data stream from the peer PL4 device. Training to the input data stream begins when digital reset is deasserted and continues until the input data stream has been locked. The PL4IO control logic continuously checks for input data rate lock while digital reset is deasserted and the input data clock (TDCLK) is locked to the synthesized CSU clock frequency, and updates the LOCKED bit maintained for each in each, per-bit, internal Input FIFO.

  The LOCKED bit for a given internal Input FIFO will be cleared at any time it is determined that there is an overflow or underflow on the FIFO.
- The PL4IO control logic automatically sequences the internal analog and digital components through the initialization process without system intervention. The external logic is only required to provide the defined sequencing of the reset signals.

Software can track the progress of PL4IO initialization by monitoring the PL4_OUT_ROOL and PL4_ID_ROOL in the **PM3392 Device Status** register. Software can periodically poll these registers or it can selectively arm an interrupt (ROOL_INT) when their contents change: see **PL4IO Lock Detect Change** register.

Software can also override the initial configuration for link testing by writing a non-default configuration to the PL4IO registers.

13.7.2 PL4IO CSU Lock Detect Logic

The PL4IO Lock Detect logic controls and monitors the CSU ABC. The CSU is continuously trained to the selected reference clock (PL4_RCLK or TDCLK as determined by REFSEL[1:0]). After training is complete the LVDS Transmitter ABCs may be enabled for normal data transmission.

The Lock Detect logic produces the Output Reference Out Of Lock (PL4_OUT_ROOL) and Input Data Reference Out Of Lock (PL4_ID_ROOL) outputs, which are available via the **PM3392 Device Status** register. Optional interrupts can be produced whenever PL4_OUT_ROOL or PL4_ID_ROOL changes: see **PL4IO Lock Detect Change** register.

The clock speed comparators continuously monitor the CSU frequency, relative to the reference clock (PL4_RCLK) and the input data clock (TDCLK), to determine if the both the current device and the peer PL4 device have achieved frequency lock. This is achieved by counting clock edges in a fixed window and then comparing this to the allowed value specified in the REF_LIMIT field of the Lock Detect Limits register. If the difference exceeds this value PL4_OUT_ROOL or PL4_ID_ROOL will be asserted, causing the internal status OUT_DIS to be asserted or IND_VALID to be deasserted. Assertion of OUT_DIS internal to the PM3392 device will result in sending of the error sequence on the TSTAT[1:0] status bus and, on the egress datapath, the training pattern will be continuously generated. Deassertion of IND_VALID results in the PL4IDU invalidating data transferred from the PL4IO and aborting all active channels (that is, all partial packets that have not been terminated by an EOP will internally be terminated with an EOP_ABORT status in the PL4IDU and be transferred downstream to the EFLX, which will result in the terminated Ethernet frame being sent on the Ethernet transmit media with an error indication). Further packet reception by the PL4IDU is held off until IND_VALID becomes asserted.

13.7.3 PL4IO Input Data Reception

The PL4IO Data Input logic that processes the data received on the TDAT[15:0]+/- and TCTL+/- device pins produces the Input Data Reference Out Of Lock (PL4_ID_ROOL) and Input Data Out Of Lock (PL4_ID_DOOL) outputs, which are available via the **PM3392 Device Status** register. Optional interrupts can be produced whenever PL4_ID_ROOL or PL4_ID_DOOL changes value, under control of the **PL4IO Lock Detect Status** register.

The PL4IO Data Input logic performs the PL4 deskew function. By use of individual Receive FIFOs for each of the 16 data channels, data can be skewed relative to the clocks. This allows for both jitter on the PL4 data bus and quantization error in the on-chip DRU.

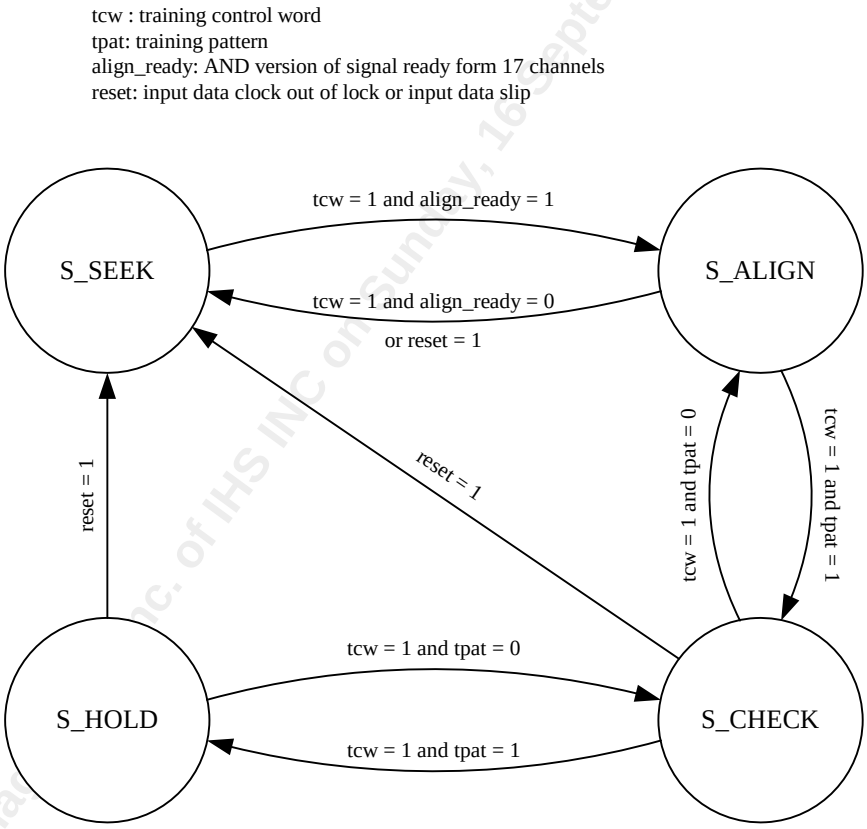
Software can track the progress of the PL4IO input data deskew operation by monitoring PL4_ID_DOOL. Software can also enable / disable the input data deskew operation by clearing / setting the TRAIN_DIS bit in the **PL4IO Configuration** register. This bit should not be cleared in slave mode until the Input Data Clock (TDCLK device pin) is valid and stable.

To ensure that the on-chip DRU is synched to the incoming data stream, the PL4IO Data Input logic waits for the number of transition events specified in the TRAN_LIMIT bits in the **PL4IO Lock Detect Limits** register before transferring data to an internal receive FIFO. After DRU training is complete PL4_ID_ROOL is de-asserted. PL4_ID_ROOL will be asserted if it is determined that the internal per-bit receive FIFO has overrun or under-run (also referred to as Input Data SLIP).

After DRU training is complete the training pattern logic is enabled. When the training pattern is concurrently detected in each of the 16 Receive FIFOs the ALIGN_READY signal is asserted. This causes each of the FIFOs to initialize its read pointer, so the data will be read in phase from all channels. When the de-skew process is complete PL4_ID_DOOL is de-asserted and IND_VALID is asserted, enabling the input data path in the downstream PL4IDU block.

- The following diagram provides an indication of the PL4IO Data In State machine. Data encapsulated in PL4 Data Bursts transferred by the peer PL4 entity across the TDAT[15:0] and TCTL device pins will be transferred in the PM3392 from the PL4IO to the downstream PL4IDU block when the PL4IO is in the "S_ALIGN" state.

Figure 21 PL4IO Data In State Diagram



13.7.4 PL4IO TSTAT FIFO Status Generation

The PL4IO Status Output logic internally accepts two bit status inputs (OUTS0, OUTS1), and encapsulated them into the two bit parallel FIFO status output stream (device TSTAT[1:0] pins) and divides the PL4 subsystem input data clock (a divided-by-4 version of TDCLK, hereafter referred to as INCLK) by 2 to produce the output status clock (TSCLK). All of the status processing logic uses INCLK, which is common to the input data side PL4 subsystem.

The main status output state machine generates the FIFO status framing structure. When IN_DIS is asserted the state machine outputs continuous "11" framing words. This inhibits the input state machine in the peer device from synchronizing to the incoming framing structure in the peer device.

The calendar state machine traverses the TDM status calendar and multiplexes the parallel channel status from the PL4 input FIFO into the FIFO status stream.

13.7.5 PL4IO RSTAT FIFO Status Reception

The PL4IO Status Input logic parses the two bit parallel FIFO status input stream (RSTAT[1:0] device pins) and internally produces two bit status outputs (INS0, INS1). The status stream is resampled by a divide-by-four version of RDCLK (hereafter called OUTCLK), which is always at least twice the speed of RSCLK. All of the status processing logic uses this OUTCLK which is common to the output data side of the PL4 subsystem internal to the PM3392.

The Status Input logic produces the Input Status Reference Out Of Lock (PL4_IS_ROOL) and Input Status Data Out Of Lock (PL4_IS_DOOL) outputs, which are available via the **PM3392 Device Status** register. Optional interrupts can be produced whenever PL4_IS_ROOL or PL4_IS_DOOL changes value: see **PL4IO Lock Detect Status** register.

The clock speed comparator continuously monitors the RSCLK frequency, relative to the output data clock (OUTCLK), to determine if the RSCLK is running faster than ½ the frequency of OUTCLK. If the RSCLK frequency exceeds this limit PL4_IS_ROOL will be asserted, causing OUT_DIS to be asserted.

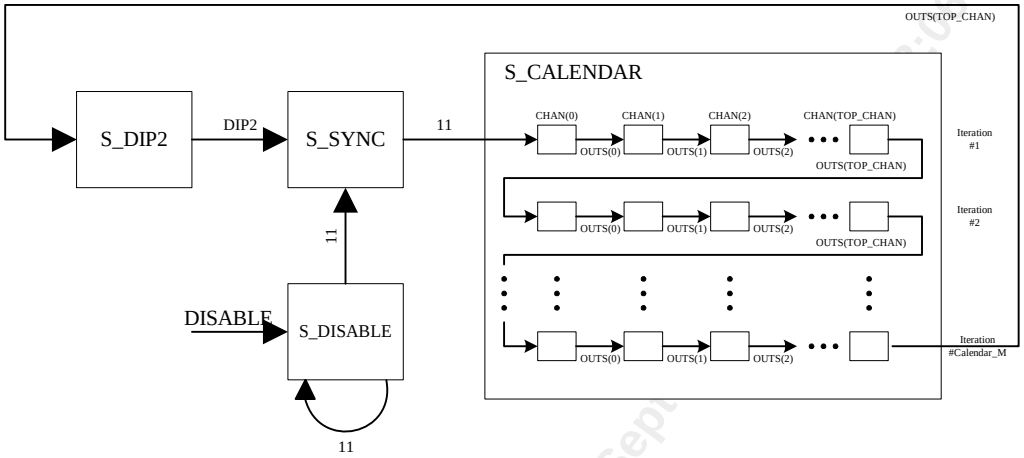
The main status input state machine synchronizes to the FIFO status framing structure. If the state machine cannot synchronize to the incoming framing structure PL4_IS_DOOL will be asserted, causing OUT_DIS to be asserted.

The calendar state machine traverses the TDM status calendar and demultiplexes the channel status stream for parallel presentation to the PL4 output data scheduler (PL4MOS).

Monitoring of RSTAT FIFO Status

Upon deassertion of digital reset, the PL4IO control logic continuously checks for the frequency of Input Status Clock (RSCLK). The sending side of the FIFO status channel is initially in the DISABLE state and sends the "11" pattern repeatedly. When FIFO status transmission is enabled, there is a transition to the SYNC state and the "11" framing pattern is sent. FIFO status words are then sent according to the calendar sequence (of length TOP_CHAN[3:0] + 1), repeating the sequence followed by the DIP-2 code. The repetition of the calendar sequence is determined by the OUT_MUL[7:0] in the Calendar Repetitions register.

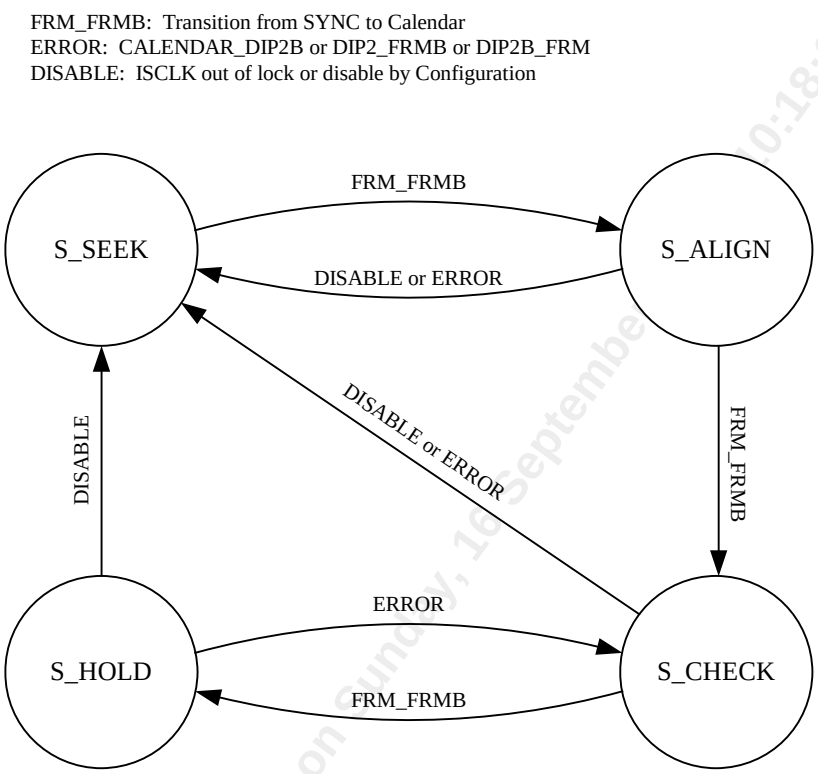
Figure 22 PL4IO FIFO Status State Diagram



The input FIFO status operation begins after digital reset is de-asserted and the PL4_IS_ROOL is logic 0 (indicating RSCLK is in the required frequency range).

The PL4IO logic follows the status stream pattern and determines if an error occurs in the incoming FIFO status stream. The ALIGNED bit in the **PL4IO Input FIFO Mode** register is set to logic “1” after a complete status pattern from SYNC to DIP-2 is detected successfully. An PL4_IS_ROOL assertion (logic 1) clears the ALIGNED bit and resets the PL4IO data_in state machine.

Figure 23 PL4IO Status In State Diagram



Software can track the progress of PL4IO input FIFO status operation by monitoring the PL4_IS_ROOL and PL4_IS_DOOL in **PM3392 Device Status** register. Software can periodically poll these registers or it can selectively arm interrupts (ROOL_INT and DOOL_INT of **PL4IO Lock Detect Status** register) when their contents change.

Software can also enable / disable the normal input FIFO status operation by clearing / setting the NO_ISTAT and ISTAT_DIS in the PL4IO **Configuration** register.

13.7.6 PL4 Ingress Training Pattern Generation

At device startup the output data path (PL4 RCTL and RDAT[15:0] pins) is disabled. During device initialization the PL4 output data state machine begins to generate training patterns; however other state machines handle other parts of the PL4 initializations. When the LVDS transmitters become active the output data state machine may be at any arbitrary point in the training pattern. Therefore a customer ASIC device must be prepared for the initial training sequence to start on any arbitrary boundary within the training pattern.

During normal operation after initialization completes, the PL4 output data state machine generates 'opportunistic' training patterns whenever the data pipeline is empty. The generated training sequence begins with an Idle Control word, followed by 10 Training Control words and 10 Training Data words. The 20 word training pattern may be repeated an arbitrary number of times, depending on when data becomes available in the pipeline. Therefore a customer ASIC device must be prepared for an arbitrary number of repetitions of the training pattern without intervening Idle Control words.

The PL4 output data state machine ensures that a training sequence of minimum length Repeat_T (the PL4 α parameter) is generated at least once in every (Max_T + MaxTransfer) cycles.

13.7.7 PL4IDU Error Handling

As stated previously, the POS-PHY Level 4 specification defines correct operation, but generally does not define error handling except where it affects interoperability between the peer devices. Consequently different devices may handle the same error differently.

The following section outlines error handling as done on the PL4IDU interface that may affect higher-layer protocols in the event that errors are detected at the PL4 Bus TDAT[15:0], TCTL, and TDCLK interface.

The PL4IDU is designed so that all single cycle PL4 bus errors are non-catastrophic and recoverable. The basic classes of errors are as follows:

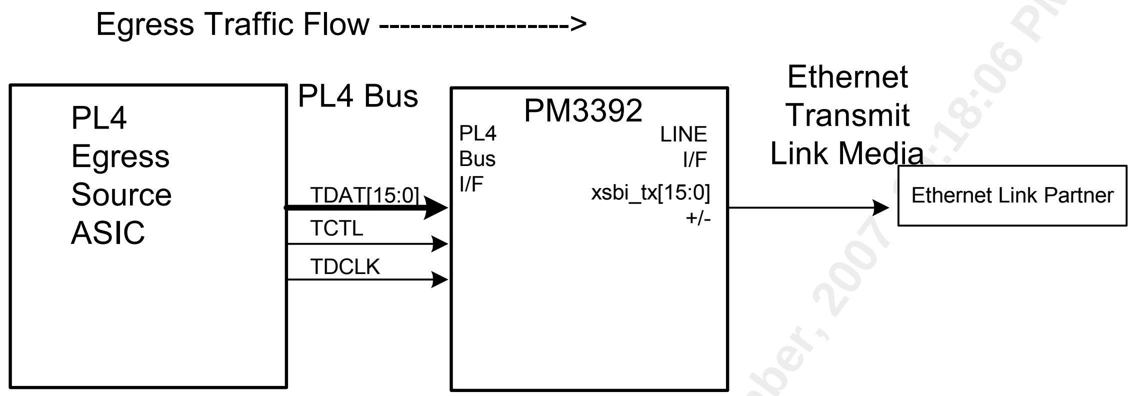
1. there is an error detected in the PL4 word parsing

2. there is a breakdown in the SOP->DATA->EOP protocol on the input data

The PL4IDU always re-synchronizes on the next valid EOP when a SOP->DATA->EOP protocol error occurs. This behavior will cause the next SOP->DATA->EOP packet to be aborted until there is a valid EOP to re-synchronize to.

Behavior in the case that MAC is Configured to append FrameCheckSequence on Transmit and a detectable error occurs on the PL4 Bus during the data transfer

In order to guarantee that any corruption of data that may occur during PL4 Bus DataBurst transfers is detected at the Ethernet link partner (data sink) with respect to the data as present internal to the data source (the PL4 Egress Source ASIC, see figure below) it is necessary that the MAC must not be programmed to modify the frame contents. Specifically, the PADEN bit must be logic 0 and the CRCEN bit must be logic 0. In this mode, any data corruption that may result during the transfer of data will be detectable at the data sink during the transfer of the Ethernet physical packet: either the packet is marked as erred during the data transfer from the PM3392 device or the FrameCheckSequence field of the physical packet will be erred.



A corollorary to the above statement applies. Specifically, if either the MAC PADEN bit is set to logic 1 or the CRCEN bit is set to logic 1 it is possible that an Ethernet frame is received at the Ethernet Link Partner with a correct FrameCheckSequence but that the internal Ethernet Frame contents as represented at the data source device may have been modified. The error rate of this occurring is less than that of the data transfer error rate on the PL4 bus media itself assuming that the PL4 Egress Source ASIC is itself error-free. Data integrity can be assured at the data sink device by doing additional verification of the received Ethernet frame: these checks can include a length check and any higher-protocol layer CRC checks.

In summary, to guarantee the highest probability of end-to-end error detection at the lowest-possible level of the communications protocol stack then data transfer across each possible communications link (in this case the PL4 Bus) must include an error protection code for each protocol layer. For the case of interfacing to the PM3392 device, this means that Ethernet Frame transfers across the PL4 Bus will have a higher probability of being detected as erred if they include the FrameCheckSequence field and the PM3392 device checks the FCS field during frame transmission.

13.8 XSBI Wrapper Line Interface

13.8.1 Initialization

Upon power up both analog reset (XSBI areset) and digital reset (dreset) are asserted simultaneously or analog reset followed by digital reset. Analog reset initializes the analog circuits to a known state, whereas digital reset initialize all normal mode ecbi registers and XSBI_Wrapper retime registers to their default value.

13.8.2 Configuration

The Configuration Register bit XSBI_EN must be set to ‘1’ in order for the data to be forwarded to and from the PCS. By default the bit-swizzle is turned off and the OIFS_EN is set to ‘1’. Bit-swizzle defaults to the Ethernet LAN mode of operation. To program the configuration register xsbi_en must be disabled. No additional configuration is required in normal operation.

13.8.3 PRBS Testing

This is a diagnostic mode for testing the SIPO/PISO (Serial In Parallel Out/ Parallel in Serial Out) analog blocks of the OIFS ABC.

PRBS Generation

PRBS generation mode overwrites incoming data stream and provides PRBS sequence to the 16 PISO's which in turn outputs the generated PRBS through the transmit side of the chip. To enable this mode the MPGM_GEN_EN bit in the configuration register must be set to '1'.

PRBS Monitor

PRBS Monitor mode detects PRBS sequence on the output of the 16 SIPO's and reports a lock condition. To enable PRBS monitor mode the MPGM_MON_EN bit in the configuration register must be set to '1'.

13.8.4 Interrupts

Reporting of XSBI_Wrapper interrupt conditions are enabled or disabled by writing to the corresponding Mask register. The interrupt pin is a logical or of all the enabled interrupts. Please refer to the register description section for various interrupts and interrupt enables(mask registers). By default the interrupts are disabled.

13.9 Loopback Operation

XSBI Local Loopback Mode

Local loop-back (system side) ties the output of the PCS transmit 64B66B decoder block to the input of the PCS Receive 64B66B encoder block. To enable this mode, set bit 5 (local_loopback_en) of Register 0x0100 XSBI Wrapper Configuration register to '1'. When the local_loopback_en bit is set to '1' the output pins TXDATA[15:0] will output the value 0x00FF until local_loopback_en is set to '0' which will resume normal operation.

PL4 System-side (Local) Loopback

The source of the 68-bit parallel data stream for the PISO in the output interface of the PL4IO logic can be configured to support system-side (local) loopback by setting the OUTSEL[1:0] bits to 2'b11. In this loopback mode, the data stream impressed on the PL4 TDAT[15:0]+/- and TCTL+/- pins goes through a SIPO and is written as a 68-bit parallel data stream comprising four PL4 bus words into an internal FIFO. The source of the 68-bit parallel data stream used by the PISO for impressing on the PL4 RDAT[15:0]+/- and RCTL+/- device pins is read from this internal FIFO. The PL4 status bus may optionally be looped back by setting the STAT_OUTSEL bit to 1. In this mode the status calender impressed on the RSTAT[1:0] device input pins is looped directly back onto the TSTAT[1:0] output pins. If STAT_OUTSEL is set to 0 and OUTSEL[1:0] is set to "11", the status that gets driven onto TSTAT[1:0] reflects the state of the device EFLX. In this mode of operation, the potential exists for the PL4 partner's receive buffer to be overflowed if cannot keep up with its (the partner's) transmitter.

Note that in this loopback mode it is the actual serial bit stream from the TDAT[15:0]+/- and TCTL+/- device pins that is being looped back. Therefore it is not required that the injected data stream represent an Ethernet frame.In this mode, Tx frame data impressed on PL4 TDAT gets forwarded to the MAC in addition to the loopback paths. As a consequence, test frames that a PL4 partner wishes to use to verify the operation of its PL4 interface get egressed to the PM3392's link partner and counted by the MSTAT block as valid network traffic. If this is not desired, one may disable the PL4IDU via the PL4IDU configuration register EN_PORTS and EN_DFWD register bits before configuring the PL4IO into local loopback mode. The PL4IDU must be re-enabled to allow line side egress if loopback is subsequently disabled.

PL4 Remote (Line-side) Loopback

The INSEL bit of the **PL4IO Configuration** register selects the source of the 68 bit parallel data stream for the PL4IDU. The default setting of INSEL, logic 0, selects the PL4 incoming DRU input data stream from the SIPO as the source of the PL4IDU 68-bit data word, IND[67:0]. Remote (that is, line-side) loopback of the PL4 internal datapath is accomplished by setting INSEL to logic 1: the source of the PL4IDU 68-bit data word, is selected to be the PL4ODP output data stream; this results in data being looped-back from the device RXDATA[15:0]+/- input pins to the TXDATA[15:0]+/- output pins.

Note that in this loopback mode that the injected data stream represents an Ethernet frame.

If PL4IO INSEL is logic 1 (remote loopback from PL4ODP to PL4IDU) and PL4IO OUTSEL[1:0] is b01 (normal operation) data will be impressed on the PL4 RDAT[15:0], RCTL pins. If the user would like to disable the transfer of data on the PL4 RDAT[15:0],RCTL interface pins then the following can be done:

STEP 1: program PL4IO OUTSEL[1:0] to b00. This will result in the RDAT[15:0],RCTL pins being driven to a logic 0.

STEP 2: PL4IO INSEL can be programmed to logic 1 to configure the PL4 interface for remote loopback.

When INSEL is set to 1 the PL4 interface must be in Master mode and have a valid reference clock for the loopback operation to work properly. The PL4IO lock status logic continues to check the lock status on the device’s PL4 interface pins. This will not effect the remote loopback operation even if the PL4IO reports IS/ID ROOL/DOOL out-of-lock, it means the PL4 partner is not configured or present.

13.10 Controlling Ethernet Frame Reception and Transmission

The reception and transmission of Ethernet frames within the PM3392 can be halted or enabled under software control. The TXEN0 and RXEN0 bits within the transmit and receive MACs (TXXG and RXXG) Configuration register enables and disables the transmit and receive data flows respectively. In addition, the FCTX, FCRX and PARF bits determine the response of the PM3392 to MAC Control frames.

13.10.1 Enabling and Disabling Reception

When the RXEN0 bit is deasserted (logic 0), the channel will cease data reception of both MAC Data frames and MAC Control frames at the next frame boundary (that is, during the interframe gap). If the channel is in the middle of receiving a frame, the frame reception will complete. All further frames, both MAC Data frames and MAC Control frames, that ingress at the RXXG line side interface will be ignored. All frames that have been received prior to halting and are buffered within either the RXXG receive FIFO or the IFLX FIFO for this channel will continue to ingress. These frames will be transferred across the PM3392 device RDAT[15:0] pins as per the PL4 Bus protocol. By default the PM3392 comes out of reset with the RXEN0 bit logic 0 (i.e. reception disabled)

It is possible to select whether a MAC Control frame is forwarded to the system side PL4 interface. The PARF bit of the RXXG GMACC1 register controls this feature. By default, MAC Control frames are not forwarded.

The FCRX bit in the TXXG Configuration 1 register determines whether the MAC Control sublayer for this channel will respond to a received MAC PAUSE Control frame. If FCRX is a logic 1 and a MAC PAUSE Control frame is received, the pause timer counter will be loaded with the PAUSE_TIME value of the received frame. If FCRX is a logic 0, the load of the pause timer counter is inhibited and the transmit of MAC Data frames from this channel is not paused.

13.10.2 Enabling and Disabling MAC Transmission

The Ethernet MAC transmit interface will be disabled if the TXEN0 bit of the **TXXG Configuration 1** register is logic 0. TXEN0 is logic 0 when the PM3392 comes out of reset. To enable the MAC interface to transmit either MAC Data or MAC Control frames, TXEN0 must be logic 1. The TXEN0 is used to enabled and disable the transmission of MAC Data frames and MAC Control frames.

When the TXEN0 bit is set to logic 1, the TXXG will cease data transfer for the transmit or egress direction on the PM3392 device. If the TXXG is in the middle of sending a frame, that frame will be finished without error. The PM3392 will then cease to transmit further MAC Data frames. If the system-side source device continues to transfer data to the PM3392 for this channel (over the PL4 TDAT[15:0] pins), the data will be buffered until all egress buffer resources have been used on the EFLX FIFO. The egress buffer resource levels are reported to the system source device using the PL4 Bus FIFO status signals over the TSTAT[1:0] pins.

13.11 Ten Gigabit Ethernet InterFrameGap Support

The PM3392 operates on frames having a range of InterFrameGap (IFG) at the Ten Gigabit Ethernet MAC interface. The receive IFG setting is fixed whereas the transmit IFG spacing is programmable.

13.11.1 Ten Gigabit Ethernet Receive IFG

The PM3392 can receive frames continuously with an IFG of equal to or greater than 40 bit times or 5 bytes. The normal receive interval is specified as the time between the last octet of the FrameCheckSequence on the previous frame and the sampling of the Start of Frame Delimiter (SFD). Note that the 9.6 ns minimum receive interval encompasses the time required for the interframe gap, preamble octets and start frame delimiter.

13.11.2 Ten Gigabit Ethernet Transmit IFG

For transmit or egress traffic the PM3392 will insert a minimum IFG of 12 bytes or 9.6 ns by default. The transmit IFG can also be programmed to allow a minimum IFG or 5 bytes or 4 ns. The IPGT[5:0] field in the TXXG Configuration 1 register defines the back-to-back IFG between frames and can be set to one of the non-reserved values as per the table below.

Table 21 Transmit InterPacket Gap Encoding

IPGT[5:0]	IPG (In Bytes)	Comment
05h	5	
06h	6	
07h	7	
08h	8	
09h	9	
0Ah	10	
0Bh	11	
0Ch	**12**	**Default value**
10h	16	
14h	20	
18h	24	
1Ch	28	
20h	32	
24h	36	
28h	40	
2Ch	44	
30h	48	
34h	52	
38h	56	
3Ch	60	
Any other value	--	Reserved

To ensure 32-bit alignment while transmitting back-to-back frames with minimum IFG, the TXXG does the following:

- For LAN mode the 32bit_align bit in the TXXG configuration 1 register needs to be set. This will cause the MAC to then insert and delete Gaps in the data stream to maintain 32bit alignment and to maintain, on average, the programmed IFG for all frames sizes.

In all cases, back-to-back frames will be sent with a transmit interval from the last octet of the FCS of the previous frame to the first octet of data in the next MAC frame of 160 bit-times (which is 96 + 64) if the IPGT encoding is 0x0C; in the case that the IPGT encoding is 0x0B, the transmit interval is 152 bit-times and so on as per the above table.

13.12 Ten Gigabit Ethernet Preamble Support

13.12.1 Transmit Preamble

On Ethernet transmit frames, the TXXG will always insert the 802.3ae specified preamble: the preamble is 7 bytes, aligned on 32-bit boundaries having the octet value 0x55 (serialized bit stream of 10101010 with serial transmission occurring from left to right). The 7-byte preamble is followed by a one-byte StartOfFrameDelimiter, SFD: having the octet value 0xD5 (serialized bit stream of 10101011 with serial transmission occurring from left to right).

13.12.2 Receive Preamble

The TXXG includes configurable options on how it will interpret the preamble and StartOfFrame delimiter (SFD) at the reconciliation sublayer. This is done through the **RXXG - Configuration 1** register: PUREP and LONGP.

Preamble checking of the content of the preamble field of the packet, ensuring a data pattern of 0x55, is done only if PUREP is a logic 1. If PUREP is a logic 0, then the data pattern during the preamble is not checked. In either case, PUREP does not affect any length check on the preamble. The 802.3ae specification for full duplex operation does not require that PUREP be set to logic 1. Note that the default mode of this register bit is logic 0

The ability of the MAC receive process to ignore frames based on the length of the preamble is controlled by the LONGP register bit. If set to logic 1, the MAC receive process will accept frames having preambles greater than 11 Bytes in length. If LONGP is set to logic 0, packets with preambles greater than 12 bytes will be ignored. The 802.3ae specification for full duplex operation does not require that LONGP is ever set to logic 1. Note that the default mode of this register bit is logic 0: by convention Ethernet frames having preambles of greater than 11 bytes are ignored.

The preamble and SFD are stripped on every received frame, converting the physical packet to an Ethernet Frame. Only the DATA octets from the internal reconciliation frame stream are transferred:

Internal reconciliation frame stream:

<Inter-frame><preamble><sfd><DATA><efd>

Ethernet Frame as transferred on PL4 Bus system interface:

<DATA>

13.13 Ten Gigabit Ethernet MAC Transmit Padding and CRC Generation

The TXXG can pad a frame for transmission that is forwarded from the PL4 system interface and is greater-than-or-equal-to 14 bytes. This is accomplished only if the PADEN and CRCEN bits are set in the **TXXG Configuration 1** register. The frame is padded with data octets having the value 0x0 to 60 bytes if not tagged or 64 bytes if tagged; a 4 octet FCS is appended to the frame prior to transmit.

The PM3392 can append a proper four-octet FCS to each and every frame prior to transmission if the CRCEN bit within the **TXXG Configuration 1** register is set.

As previously described, frames without an error indicator having a length of at least 9 bytes to 14 bytes and if the PADEN bit is set will be padded to 60 bytes and then a bad CRC is appended to the frame and the Error flag is also asserted to the PCS layer to guarantee that the frame is properly marked as bad. 14 bytes are required for the Ethernet frame destination address (6 octets), source address (6 octets), and LENGTH/TYPE field (2 octets).

Frames that are less than 9 bytes will be discarded and an internal debug counter will count the number of times a packet is discarded due to less than 9 bytes.

Table 22 PM3392 Minimum Transmit Frame Size Padding

Frame Length at internal Transmit MAC interface	Frame Type	PADEN State	CRCEN State	Pad Action	CRC Action
<9 bytes	Normal or tagged	X	X	Frame discarded by Transmit MAC control logic and debug counter incremented	
9 to < TX MIN Frame Size Reg.	Normal	0	0	No Pad	Append 4 byte CRC with ERROR[1]
9 to < TX MIN Frame Size Reg	Normal	0	1	No Pad	Append 4 byte CRC with ERROR
9 to < TX MIN Frame Size Reg	Normal	1	0	Pad with 0's to TX MIN Frame Size Reg value	Append 4 byte CRC with ERROR[2]
9 to < TX MIN Frame Size Reg	Normal	1	1	Pad with 0's to TX MIN Frame Size Reg value[3]	Append 4 byte CRC
>= TX MIN Frame Size	Normal	0	0	No Pad	No CRC Append

[1] Packet is less than the minimum frame size. Transmit MAC will assert and Error to the Transmit PCS at the End of Packet and add a Corrupt CRC to guarantee a downstream receiver will discard this packet.
[2] Same comment as above.
[3] If Frame is VLAN tagged the Pad will increase by 4 bytes.

Frame Length at internal Transmit MAC interface	Frame Type	PADEN State	CRCEN State	Pad Action	CRC Action
Reg.					
>= TX MIN Frame Size Reg.	Normal	0	1	No Pad	Append 4 byte CRC
>= TX MIN Frame Size Reg.	Normal	1	0	No Pad	No CRC Append
>= TX MIN Frame Size Reg.	Normal	1	1	No Pad	Append 4 byte CRC

13.14 Ethernet Frame Transmit Errors

The PM3392 device will use the 64-bit Error Block as defined in Table 49-1 in the 802.3ae standard, to indicate a transmission error to its peer entity across the Ethernet media.

13.14.1 Transmit Frame Error Catalysts

The minimum frame length at the transmit system interface of the TXXG for propagating an error indication is 9 bytes. Any frame indication forwarded by the PL4 system side via the EFLX that has a length less than 9 bytes will be discarded by the transmit control logic internal to the TXXG: hence, no external MAC carrier event will be signaled.

The catalysts for frame transmission with error are listed below. At least one catalyst needs to be true for asserting a frame transmit error on the Ethernet media:

1. If the PL4 system sourcing device asserts an End-Of-Packet status of ABORT during egress frame transfer (PM3392 device pins TDAT[15:0]+/-, TCTL+/-). This error indication is forwarded on to the internal TXXG transmit interface.

2. If the PL4 incoming data word parser (in the PL4IDU logic) indicates that the packet should be aborted. This could be either because a PL4 Bus error occurred (case EOP ABORT) or because a PL4 Bus error was detected (case EOP ABORT). This error indication is forwarded on to the internal TXXG transmit interface.

3. If the egress frame is considered short (see note below), long, or having an internal transmit MAC error (i.e. transmit underrun). Another way of stating this is that the frame is transmit with an error indication if the MAC sublayer within the TXXG detects a transmit frame having a status that will be interpreted by the peer device across the Ethernet link as erred. NOTE: Short frames at the TXXG internal transmit system interface will assert the Error Propagation character only if the following register settings are in effect:
   - The PADEN bit is set to logic 0

Note that detection of an error in the FrameCheckSequence (FCS) field of a frame during transmission does not result in a transmit error indication on the Ethernet media for the given channel. Restated, an FCS error detected during frame transmission is not a catalyst for asserting the Error_Propagation character. The frame is expected to be detected during the frame receive process on the peer entity across the Ethernet media.

Transmit Underrun

An internal transmit MAC error is detected if the Transmit FIFO internal to the TXXG at the EFLX interface underruns: that is, the FIFO goes empty before an internal End-Of-Packet indicator is read. In a typical system application transmit underrun will not occur as long as the peer device that is sourcing traffic on the PL4 Bus is able to maintain an adequate rate of data transfers. As previously described, the egress FIFO interface on the PM3392 device maintains a cut-through threshold that is used for determining whether data is to be forwarded on to the TXXG for transmission on the Ethernet media. For a system application that is operating on standard Ethernet sized frames (those having a maximum length of 1522 bytes), the cut-through threshold can be set for 1536 bytes (see **EFLX Cut-Through Threshold** register), thereby guaranteeing that frames will not be transferred from the egress FIFO to the TXXG until the complete frame is buffered. Internal to the PM3392 device, the rate of data transfer from the TXXG for transmit data is limited by the 645 MHz device reference clocks, with two octets of data being transferred every reference clock period. The difference in internal clock rates between the TXXG transmit interface and the egress FIFO accounts for at most a two-byte contribution to the transmit underrun calculation:

9600 bytes * 8-bits/byte * 200 ppm = 15.4 bit-times

Therefore, for a system application that is transferring Ethernet frames of 9600 bytes the requirement to prevent underrun is, on a frame-by-frame basis that the traffic sourcing device must transfer the remaining (frame_length – cut-through) number of data bytes within the amount of time that it takes to impress the frame data on the Ethernet media, not counting the inserted IFG and preamble times.

For example, to guarantee that transmit underrun does not occur on a frame of 9600 bytes - assuming [1] that the 645 MHz reference clock has an approximate 1.5 ns period; [2] that the EFLX cut-through threshold is set for 1536 bytes; [3] that there are no errors detected on the PL4 Bus incoming word stream at the PL4IDU block of the PM3392 device - the system interface egress device must transfer the last 8604 (i.e. 9600-1536) bytes of the frame on the PL4 Bus within a 6 us (8064/2 * 1.5 ns) time window starting from the time that the 1536^{th} byte was transferred on the PL4 Bus.

The response of the TXXG to transmit underrun is:

1. All bytes that had been forwarded to the TXXG at the time the underrun occurred will be impressed on the XSBI pins as long as the underrun condition occurs after the 9^{th} byte of the frame (if before 9 bytes, the data is discarded as described previously since there is not a valid Ethernet DA, SA, and IEEE/TYPE field in the frame)

2. At the point of underrun, the error is signaled on the XSBI pins using the Ethernet Transmit Frame Error protocol described below

3. All subsequent data sent by the EFLX to the TXXG will be discarded until the next valid Start-Of-Packet flag is set.

13.14.2 Ethernet Transmit Frame Error Protocol

All frames that are transferred by the MAC interface with a transmit error calculation exhibit the following behavior:

- A 64-bit Control block is formed with the ERROR Control code as defined in Table 49-1 in 802.3ae, all eight character locations.

In the event that the frame being transmit is detected as having an error at the input transmit control interface of the TXXG and prior to an End-Of-Frame indicator, the frame will be truncated by the transmit control logic and the MAC layer will be signaled to assert transmit error. The following transmit errors are detectable at the TXXG transmit control interface and can result in truncation during transmission:

1. An EOP ABORT status is detected on the PL4 system interface. This EOP ABORT could have been indicated on the PL4 Bus as an End-Of-Packet status of ABORT or it could be the result of the PL4IDU logic detecting a word parsing error.

2. Transmit underrun is detected

3. TX Max Frame length is violated

13.15 Frame Length Support

The PM3392 supports a programmable maximum threshold for MAC frame length and a programmable forwarding threshold; the thresholds in both the ingress and egress directions are fully independent and programmable. In addition, there is a minimum fixed size of a frame that is supported in each direction; frames of less than this fixed size are always discarded.

The PM3392 supports jumbo frames up to 9600 bytes in both the ingress and egress directions.

13.15.1 Ingress (Ethernet Receive) Frame Length

The RXXG supports a minimum ingress frame fragment size of 9 bytes (fixed) and a programmable maximum ingress frame size of up to 9600 bytes.

The minimum frame fragment size requirement is the result of supporting address filtering: any received frame that has less than 9 bytes will be not be forwarded on from the RXXG. If a frame fragment of less than 9 bytes is received it will be filtered and the proper receive statistics information on the frame recorded.

The **RXXG Minimum Receive Frame Length** register control the minimum size of the ingress frame. If the frame is less than the programmed Min frame length the frame will be treated as a Short or Runt frame depending on if the frame had a non-erred or erred FCS respectively. The Short or Runt frame is then filtered and the appropriate status vector is generated for the frame to update the Ethernet Statistics.

The **RXXG Maximum Receive Frame Length** register controls the maximum size of the ingress frame. If the frame is greater than the programmed size the frame will be treated as a long or jabber frame, depending on if the frame had a non-erred or erred FCS respectively. If the length of the received frame is greater than the **RXXG Receive Maximum Frame Length** then the incoming frame is truncated to the length programmed in that register and the frame is flagged as erred.

The **RXXG Receive FIFO Threshold** register sets the forwarding threshold used for ingress frame gathering and error reporting. Frames are passed from the RXXG to the PL4 ingress FIFO if an end-of-frame indication has been received by the RXXG or if the number of bytes received by the RXXG is greater than the **RXXG Receive FIFO Threshold** register. Ethernet MAC frames that are received as erred and that are forwarded on to the system interface (the POS-PHY Level 4 interface) cause the affected packet to have an End-of-Packet status of ABORT in the PL4 Payload Control word following the last byte of the Ethernet frame that is in the PL4 Data word.

This mechanism provides for two different frame error reporting capabilities. First if the forwarding threshold is set higher than the received frame size the RXXG will drop and not forward erred frames. Second if the forwarding threshold is set lower than the received frame size the RXXG will immediately start passing the incoming frame as soon as the threshold is reached. In this case the RXXG passes the state of the ReceiveError flag to the downstream logic and the PM3392 will assert an End-Of-Packet status of ABORT during data transfer on the PL4 bus.

There is no additional frame forwarding threshold register in the ingress datapath of the PM3392 device other than the RXXG **Receive FIFO Threshold** register.

13.15.2 Receive Frame Length Checking

The PM3392 device supports frame length checking based on the value of the sixteen bit Length/Type field in the received frame. Receive frame length checking is enabled by setting the FLCHK bit of the **RXXG Configuration 1** register**,** to a logic 1. If FLCHK is set to a logic 1, a received frame will fail the frame length check if the Length/Type field of the frame has a Length interpretation and the number of received octets in the data frame less 18 (Length field indicates the number of MAC client data octets and does not count the 6 octets of destination address nor 6 octets of source address nor 2 octets of Length/Type field itself nor 4 octets of FCS) is GREATER than the numeric value of the Length/Type field. The Length/Type field of a receive frame has a Length interpretation on the PM3392 device if the Ethernet frame is untagged (as per 802.3 standard, clause 3.5.4- “The Length/Type field of a tagged MAC frame always uses the Type interpretation”) and the Length/Type field is greater than or equal to 0 decimal and less than or equal to 1500 decimal.

13.15.3 Egress (Ethernet Transmit) Frame Length

The **EFLX FIFO Cut-Through Threshold** register sets the forwarding threshold used for egress frame gathering on a per-channel basis. Packets passed to the PM3392 on the PL4 bus will be gathered in the egress FIFO until an end of packet indication or until the number of bytes transferred to the PM3392 and present in the egress buffer are greater than or equal to the **EFLX FIFO Cut-Through Threshold** register (this register counts in terms of 16-byte data blocks). This forwarding threshold allows frame buffering required to ensure that a frame will not underrun once frame transmission begins on the Ethernet link.

The TXXG supports a minimum frame length of 9 bytes (fixed) at the transmit system interface. The programmable maximum egress frame size is 9600 bytes. A frame that is received with a length of less than 9 bytes, from the PL4 system side (i.e. via the EFLX) will be discarded, regardless as to the setting of the PADEN and CRCEN mode bits. Frames having a length of 9 or more bytes at the system interface will always be transferred by the Ethernet MAC and impressed on the PM3392 XSBI pins.

In order to aid diagnostics, a count of frames that are discarded in the TXXG transmit interface because they had a length of less than 9 bytes is maintained in the **TXXG Filter Error Count** register.

The minimum frame size on the egress channel for MAC frames after any optional padding or FCS appending is 64 bytes: this is a fixed limit. If a frame is transferred to the TXXG transmit interface having a length of between 9 and 64 bytes after any optional padding or FCS appending, then the frame will be transmit as a short erred frame: the error control block will be sent at the end of the frame.

The **TXXG Transmit Max Frame Length** register controls the maximum size of the egress frame. The **Transmit Max Frame Length** register specifies the maximum number of bytes transmitted before truncation in an outgoing non-tagged Ethernet frame (in the 802.3-2000 specification, this parameter is termed maxUntaggedFrameSize). The default setting of this register is 0x05EE (1518 decimal), which results in a maximum frame size before frame truncation of 1518 bytes if untagged and 1522 bytes if VLAN tagged. Frames that have exceeded the setting of **Transmit Max Frame Length** are truncated and have the error control block impressed at the End-Of-Packet. VLAN tagged frames have a 4 byte offset (i.e. 1522 bytes by default) before being considered as violating the frame length setting.

The length of the frame as transmit by the Ethernet MAC is determined by the length of the frame transferred by the egress FIFO, the setting of the PADEN and CRCEN mode bits (as described in the section “MAC Transmit Padding and CRC Generation”), and the value of the **TXXG Transmit Max Frame Length** register.

13.16 Frame Data and Byte Format

The PM3392 transfers octets of Ethernet Data Frames in the same order as they are transmitted and received on the Ethernet media. As noted in the IEEE Std 802-1990 (Local and Metropolitan Area Networks: Overview and Architecture):

1. The transmission of data for IEEE 802.3 occurs LSB (Least Significant Bit) first. This is true for the entire packet, LAN MAC address fields (source and destination), MAC-specific fields (e.g. length/type field in IEEE 802.3 LANs) and the MAC Information Field. (See also 802.3 standard, clause 3.3).

2. The 48-bit address (universal or local) is represented as a string of six octets. The octets are displayed from left to right, in the order that they are transmitted on the LAN medium, separated by hyphens. Each octet of the address is displayed as two hexadecimal digits. The bits within the octets are transmitted from left to right as shown below. In the display of octets, the first bit transmitted of each octet on the LAN medium is the LSB of that octet (e.g. I/G address Bit is LSB of octet 0). The Organizationally Unique Identifier is contained in octets 0,1,2 with octets 3,4,5 being administered by the assignee. The example given in Std 802-1990 is:

Table 23 Std 802-1990, Figure 5-3 Universal Address

Octet	0	1	2	3	4	5
	0011_0101	0111_1011	0001_0010	0000_0000	0000_0000	0000_0001
	^					
	First bit transmitted on the LAN medium (also the I/G Address Bit).					

The hexadecimal representation of this example is: AC-DE-48-00-00-80

This hexadecimal representation is often referred to as the canonical format.

For the purposes of the PM3392 engineering document, the address example given above in the Std 802-1990 is represented as follows:

DA[0] is the 1st bit on the LAN medium: 0
DA[1] is the 2nd bit on the LAN medium: 0
DA[2] is the 3rd bit on the LAN medium: 1
DA[3] is the 4th bit on the LAN medium: 1
DA[4] is the 5th bit on the LAN medium: 0
DA[5] is the 6th bit on the LAN medium: 1
DA[6] is the 7th bit on the LAN medium: 0
DA[7] is the 8th bit on the LAN medium: 1

DA[8] is the 9th bit on the LAN medium: 0
DA[9] is the 10th bit on the LAN medium: 1
DA[10] is the 11th bit on the LAN medium: 1
DA[11] is the 12th bit on the LAN medium: 1
DA[12] is the 13th bit on the LAN medium: 1
DA[13] is the 14th bit on the LAN medium: 0
DA[14] is the 15th bit on the LAN medium: 1
DA[15] is the 16th bit on the LAN medium: 1
And so forth

DA[40] is the 41st bit on the LAN medium: 0
DA[41] is the 42nd bit on the LAN medium: 0
DA[42] is the 43rd bit on the LAN medium: 0
DA[43] is the 44th bit on the LAN medium: 0
DA[44] is the 45th bit on the LAN medium: 0
DA[45] is the 46th bit on the LAN medium: 0
DA[46] is the 47th bit on the LAN medium: 0
DA[47] is the 48th bit on the LAN medium: 1

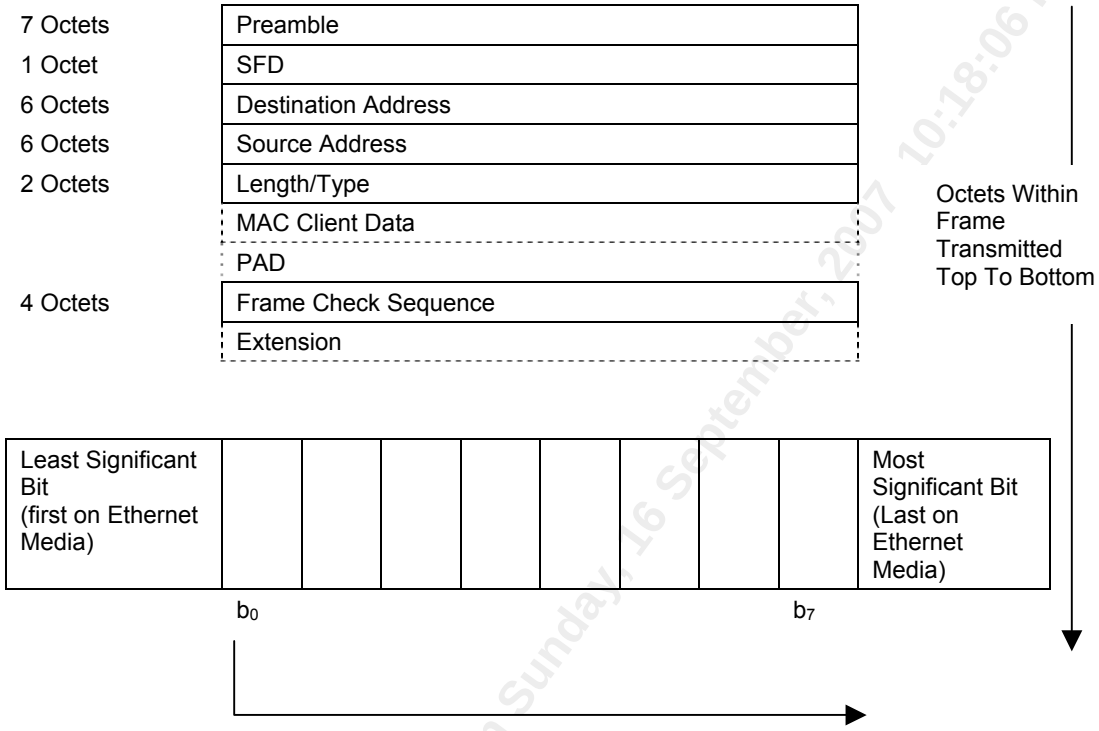
The PM3392 will represent the address shown above (example in Figure 5-3 of Std 802-1990: AC-DE-48-00-00-80) as

DA[7:0] = AC
DA[15:8] = DE
DA[23:16] = 48
DA[31:24] = 00
DA[39:32] = 00
DA[47:40] = 80

13.16.1 Frame Data and Byte Format On Ethernet Line Side

Bit-in-byte ordering in the 802.3 Ethernet standard is bit 0 (Least Significant Bit) to bit 7 (Most Significant Bit) with bit 0 being the first bit transferred on the Ethernet media. Ethernet data is always transmitted and received via the MAC line side in the following format. Bits are transmitted and received from the top to bottom and from left to right. For example, for the destination address (DA[47:0]), bit DA[0] is transmitted first and bit DA[47] is transmitted last.

Table 24 MAC Frame Format



13.16.2 Frame Data and Byte Format At PL4 Interface

Bit in Byte Ordering within the entire PL4 data path is from bit 7 (Most Significant Bit) to bit 0 (Least Significant Bit). In the ingress direction, the first byte of the frame that is received on the Ethernet line side is impressed on the PL4 RDAT[15:8] byte lane, with the most significant bit aligned to RDAT[15]. Likewise in the egress direction, the first byte of the frame that is to be transmit on the Ethernet line side is impressed on the PL4 TDAT[15:8] byte lane, with the most significant bit aligned to TDAT[15].

Two examples are provided: one for a non-VLAN tagged Ethernet frame having a length of 1518 bytes (so the IEEE correct Length/Type field of the frame is 05-DC) and one for a VLAN tagged Ethernet frame.

Table 25 PM3392 Data Order On PL4 Interface, Non-VLAN Ethernet Frame

Bits 15:8	Bits 7:0
DA[7:0]	DA[15:8]
DA[23:16]	DA[31:24]
DA[39:32]	DA[47:40]
SA[7:0]	SA[15:8]
SA[23:16]	SA[31:24]
SA[39:32]	SA[47:40]
Length/Type[7:0]: 0x05	Length/Type[15:8] 0xDC
Data[7:0]	Data[15:8]
Data[23:16]	…
…	…
FCS[24:31] 1st octet of **FRAME CHECK SEQUENCE** as received on Ethernet media. As per 802.3 clause 3.2.8, the FCS is impressed on the media with bit x31 first, bit x30 2nd, and so on with bit x0 being transmit last, RDAT[15:8] corresponds to bits [x24,x25,x26,x27,x28,x29,x30,x31].	FCS[16:23] 2nd octet of **FRAME CHECK SEQUENCE** as received on Ethernet media. As per 802.3 clause 3.2.8, the FCS is impressed on the media with bit x31 first, bit x30 2nd, and so on with bit x0 being transmit last, RDAT[7:0] corresponds to bits [x16,x17,x18,x19,x20,x21,x22,x23].
FCS[8:15] 3rd octet of **FRAME CHECK SEQUENCE** as received on Ethernet media. As per 802.3 clause 3.2.8, the FCS is impressed on the media with bit x31 first, bit x30 2nd, and so on with bit x0 being transmit last, RDAT[15:8] corresponds to bits [x08,x09,x10,x11,x12,x13,x14,x15].	FCS[0:7] 1st octet of **FRAME CHECK SEQUENCE** as received on Ethernet media. As per 802.3 clause 3.2.8, the FCS is impressed on the media with bit x31 first, bit x30 2nd, and so on with bit x0 being transmit last, RDAT[15:8] corresponds to bits [x00,x01,x02,x03,x04,x05,x06,x07].

Table 26 PM3392 Data Order On PL4 Interface, VLAN Ethernet Frame Type

Bits 15:8	Bits 7:0	Comment
DA[7:0]	DA[15:8]	Octets 1,2
DA[23:16]	DA[31:24]	Octets 3,4
DA[39:32]	DA[47:40]	Octets 5,6
SA[7:0]	SA[15:8]	Octets 7,8
SA[23:16]	SA[31:24]	Octets 9,10
SA[39:32]	SA[47:40]	Octets 11,12
0x81	0x00	VLAN Qtag Prefix
TAG_CONTROL [7:0]	TAG_CONTROL [15:8]	Where the 12 bit VID field is given by the most significant 12 bits of TAG_CONTROL; CFI bit is TAG_CONTROL[4], and USER_PRIORITYis TAG_CONTROL[7:5].
Length/Type[7:0]	Length/Type[15:8]	MAC Client Length/Type
Data[7:0]	Data[15:8]	Octets 19,20
Data[23:16]	…	Octets 21,22
…	…	…
FCS[24:31]	FCS[16:23]	FCS[24:31] is the first octet of FCS impressed on the Ethernet media
FCS[8:15]	FCS[0:7]	FCS[0:7] is the fourth octet of FCS impressed on the Ethernet media

13.16.3 Ethernet Frame Segmentation On The PL4 Bus System Interface

The PM3392 device in the ingress direction makes efficient usage of PL4 Bus transfers from the ingress FIFO logic. Scheduling by the PL4MOS is not guaranteed to be done at a frame boundary. The PM3392 device in the ingress direction cannot be programmed to guarantee that entire Ethernet frames will be transmit contiguously.

Note that in the case that the RXXG marked the ingress Ethernet frame with an err delimiter, then the frame will be transferred on the PL4 Bus and terminated with a EOPS[1:0] status of EOP_ABORT). Because of the encoding of the EOP by the PL4 Bus specification in this manner, it is impossible for the PL4 peer entity to determine the delimited length of the erred frame to within one byte. That is, the PL4 peer entity does not know whether one or two bytes of the frame terminated with EOP_ABORT were received at the RXXG link interface.

13.16.4 Example of Data Representation, From Ethernet Physical Packet To PL4 Bus Interface

The following example shows how a Physical Packet corresponding to a 64 byte MAC Data Frame would appear on the Ethernet media and when transferred on the PL4 Bus interface. This holds true for the ingress direction (Ethernet receive media to PL4 RDAT[15:0], RCTL pins) and egress direction (PL4 TDAT[15:0], TCTL pins to Ethernet transmit media).

This example utilizes a 64-byte Ethernet Data Frame that is represented in canonical format as:

DA = 12-34-56-78-9A-BC
SA = 01-23-45-67-89-AB

LENGTH/TYPE field = 0C-0D (so type interpretation)

MAC CLIENT DATA = 0E-0F-10-11-12-13-14-15-

16-17-18-19-1A-1B-1C-1D-
1E-1F-20-21-22-23-24-25-
26-27-28-29-2A-2B-2C-2D-
2E-2F-30-31-32-33-34-35-
36-37-38-39-3A-3B

FrameCheckSequence field = FD-C9-0F-E2

This is also the representation of the frame at the reconciliation sublayer (e.g. XGMII):

Table 27 Ten Gigabit Ethernet Frame Example

Ethernet Media Serial Bit Stream (first -> last)	PM3392 Octet Internal Representation	PL4 Bus DAT[] pins and DataBurst cycle	PM3392 Interpretation
/S/ Start_Of_Packet	Internal PCS replaces /S/ with the first preamble octet: 0101_0101	Not transferred	Start_Of_Packet: Physical packet only so not transferred on PL4 Bus
10101010	0101_0101	Not transferred	2nd octet of preamble
10101010	0101_0101	Not transferred	3rd octet of preamble
10101010	0101_0101	Not transferred	4th octet of preamble
10101010	0101_0101	Not transferred	5th octet of preamble
10101010	0101_0101	Not transferred	6th octet of preamble
10101010	0101_0101	Not transferred	7th octet of preamble
10101011	1101_0101	Not transferred	SFD (Start Frame Delimiter) Physical packet only so not transferred on PL4 Bus
01001000	0001_0010	DAT[15:8], cycle 1	Destination Address [7:0]
00101100	0011_0100	DAT[7:0], cycle 1	Destination Address [15:8]
01101010	0101_0110	DAT[15:8], cycle 2	Destination Address [23:16]
00011110	0111_1000	DAT[7:0], cycle 2	Destination Address [31:24]
01011001	1001_1010	DAT[15:8], cycle 3	Destination Address [39:32]
00111101	1011_1100	DAT[7:0], cycle 3	Destination Address [47:40]
10000000	0000_0001	DAT[15:8], cycle 4	Source Address [7:0]
11000100	0010_0011_	DAT[7:0], cycle 4	Source Address [15:8]
10100010	0100_0101	DAT[15:8], cycle 5	Source Address [23:16]

Ethernet Media Serial Bit Stream (first -> last)	PM3392 Octet Internal Representation	PL4 Bus DAT[] pins and DataBurst cycle	PM3392 Interpretation
11100110	0110_0111	DAT[7:0], cycle 5	Source Address [31:24]
10010001	1000_1001	DAT[15:8], cycle 6	Source Address [39:32]
11010101	1010_1011	DAT[7:0], cycle 6	Source Address [47:40]
00110000	0000_1100	DAT[15:8], cycle 7	Length/Type Field [7:0]
00110000	0000_1100	DAT[15:8], cycle 7	Length/Type Field [7:0]
10110000	0000_1101	DAT[7:0], cycle 7	Length/Type Field [15:8] Canonical Format of Length/Type Field is 0C-0D
01110000	0000_1110	DAT[15:8], cycle 8	MAC Client Data, 1st octet
11110000	0000_1111	DAT[7:0], cycle 8	MAC Client Data, 2nd octet
00001000	0001_0000	DAT[15:8], cycle 9	MAC Client Data, 3rd octet
10001000	0001_0001	DAT[7:0], cycle 9	MAC Client Data, 4th octet
……	……	……	……
01011100	0011_1010	DAT[15:8], cycle 30	MAC Client Data, 45th octet
11011100	0011_1011	DAT[7:0], cycle 30	MAC Client Data, 46th octet
10111111	1111_1101	DAT[7:0], cycle 31	FrameCheckSequence, 1st octet
10010011	1100_1001	DAT[7:0], cycle 31	FrameCheckSequence, 2nd octet
11110000	0000_1111	DAT[7:0], cycle 32	FrameCheckSequence, 3rd octet
01000111	1110_0010	DAT[7:0], cycle 32	FrameCheckSequence, 4th octet Canonical Format of FrameCheckSequence Field is FD-C9-0F-E2
/T/ End_Of_Packet	Internal PCS appends /T/ on transmit, strips on receive	Not transferred	End_Of_Packet: Physical packet only so not transferred on PL4 Bus

13.17 Frame Filtering

The PM3392 has simple programmable options to filter or forward ingress frames to the upstream link device. The PM3392 Receive Address Filtering Logic consists of eight exact-match MAC/VID filters, one 64-bin hash based multicast filter and four address filtering control registers that control the state of the forwarding for each filter. Each exact match filter includes one 48-bit MAC Address register and one 12-bit VID register that can be programmed through the microprocessor interface to the appropriate values. The filter logic is controlled by the four RXXG Address Filter Control registers. The host microprocessor has complete programmable access to all filtering features. Each address filter option is per-port independently programmable.

13.17.1 Group Multicast Address Filtering

In parallel with the exact address match, the PM3392 performs multicast filter lookups. Within the PM3392 there resides a 64-bin hash based multicast filter consisting of one 64-bit mask register that is programmable from the Microprocessor interface (**RXXG Multicast Hash** register). This register is used in conjunction with a 6-bit value which is derived from bits [28:23] of the 32-bit CRC computed over the Destination Address. This 6-bit CRC value is used to index into the 64-bit mask register. The 64-bit mask register is used to determine if a multicast address that hashes to a given bin will be accepted for forwarding The 64-bin hash based multicast filtering is enabled by the MHASH_EN bit in the **RXXG Address Control 2** register. If the MHASH_EN bit is 0 then there is no hash based multicast filtering, however if MHASH_EN is 1 then hash based multicast filtering is enabled.

The multicast hash filter operation operates only on multicast-type frames: those with the IEEE Group/Functional bit set in the DA of the frame (least significant bit of the most significant byte of the MAC DA. The final forward versus filter decision for a frame is a combination of both the Group Multicast address filter result and the results from the eight possible exact match filter operations.

The 48-bit destination address of the received frame is passed through the standard 802.3 CRC function in the same order in which the destination address octets are received. Making reference to the 802.3 specification, section 3.2.8 Frame Check Sequence field, the CRC function generating polynomial and function is:

$$G(x) = x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^{8}+x^{7}+x^{5}+x^{4}+x^{2}+x^{1}+1$$

1. The first 32-bits of the frame (which is the first 32-bits of the destination address received) are complemented.

2. The 48 bits of the destination address are then considered to be the coefficients of a polynomial M(x) of degree 47

3. M(x) is multiplied by x^{32} and divided by G(x), producing a remainder R(x) of degree <= 31.

4. The coefficients of R(x) are considered to be a 32-bit sequence.

Bits [28:23] of the resultant 32-bit CRC remainder (call this crc_rem[28:23]) are used as the index into the MHASH[63:0] register. The result of the Group Multicast address filter is logically represented by the variable MHASH_ACCEPT:

MHASH_ACCEPT = (MHASH_EN == 1) & (MHASH[crc_rem[28:23]] ==1);

13.17.2 Exact Match Filter Operation

Each of the eight exact match filters on the RXXG has four bits of associated configuration. These are found in the Address Filter Control 1 register:

1. ADRFILT_CTRL[0] enables the exact match operation. If this bit is a logic 0 then the EXACT_MATCH operation returns a logic 0.

2. ADRFILT_CTRL[1] selects whether the source address or destination address of the received frame is used as the address for matching.

3. ADRFILT_CTRL2] enables the match function to also compare the VLAN Tag VID[11:0] field of the receive frame if the two bytes following the receive frame source address are equal to the VLAN Tag ID register

4. ADRFILT_CTRL[3] is a configuration bit that determines whether an exact match will affect the variable ACCEPT or DISCARD.

The exact match filter operation is a two step process. The first step is to determine whether the address match criteria is logically true:

- EXACT_MATCH is logic 1 if the exact match filter is enabled and the selected frame address (and optional VID field of a VLAN tagged frame) are equal; otherwise, EXACT_MATCH is logic 0.

The second step is to set the EXACT_MATCH_ACCEPT or EXACT_MATCH_DISCARD variable for the given (one of eight) exact match filters based on the setting of ADRFILT_CTRL[3]:

- EXACT_MATCH_ACCEPT = EXACT_MATCH & (ADRFILT_CTRL[3] == 1);
- EXACT_MATCH_DISCARD = EXACT_MATCH & (ADRFILT_CTRL[3] == 0);

13.17.3 Address Filter ACCEPT / DISCARD Evaluation

The final result of the address filter function is a single filter versus forward decision. The result of the Group Multicast Address filter is combined with the result of the eight possible exact match filter operations to determine a final filter versus forward decision. The address filter logic can be configured so that a frame has a higher priority for being forwarded or filtered: this decision is based on the configuration bit PMODE in the Address Filter Control 2 register.

The results of the above filter operations are logically OR'ed together and then evaluated based on PMODE. Let EXACT_MATCH_ACCEPT[7:0] and EXACT_MATCH_DISCARD[7:0] represent the ACCEPT and DISCARD variables for the eight independent exact match filters respectively. The final combined value of ACCEPT and DISCARD for all address filters is logically:

- ACCEPT = (EXACT_MATCH_ACCEPT[7:0] != 0) | MHASH_ACCEPT;
- DISCARD = (EXACT_MATCH_DISCARD[7:0] != 0);

13.17.4 Address Filtering in Non-Promiscuous Mode

DISCARD has priority over ACCEPT in non-Promiscuous mode (PMODE a logic 0). A frame will be filtered only if ACCEPT is true and DISCARD is false. This is shown in the following table. It should be noted that if all filters are disabled, then all frames are filtered.

Table 28 Address Filter Result in Non-Promiscuous Mode

PMODE	Discard	Accept	Result of Address Filter Function
0	0	0	Filter frame
0	0	1	Forward frame
0	1	0	Filter frame
0	1	1	Filter frame

13.17.5 Address Filtering in Promiscuous Mode

ACCEPT has priority over DISCARD in Promiscuous mode (PMODE a logic 1). A frame will be filtered only if DISCARD is true and ACCEPT is false. This is shown in the following table. It should be noted that if all filters are disabled, then all frames are accepted. See clause 5.2.4.3 of 802.3 standard, function LayerMgmtRecognizeAddress, for a reference to “promiscuous”.

Table 29 Address Filter in Promiscuous Mode

PMODE	Discard	Accept	Result of Address Filter Function
1	0	0	Forward frame
1	0	1	Forward frame
1	1	0	Filter frame
1	1	1	Forward frame

13.17.6 Address Filter Programming

The PM3392 frame filtering is programmed in the following manner.

1. Disable receive Ethernet traffic by setting RXEN0 bit to 0 in the RXXG Configuration Register.

   Note: When RXEN0 is set to 0 there is a possible wait time for a frame to complete being received since the disable is a graceful disable and will only disable the port during an IFG period.

2. Set the ADRFILT_CTL[0] to 0 (disable state) for all exact match filters that need to be programmed or changed, including the MHASH_EN if programming or changing the Multicast Hash Filters.

3. Program all desired filters with the desired contents.

   - Program the **RXXG Exact Match Address** and **RXXG Match VID** registers and respective **RXXG Address Control 0** or **RXXG Address Control 1** registers for the desired Exact match options.

- o Program the **RXXG Multicast Hash** register with the desired bit mask and enable by programming the **RXXG Address Filter Control 2** register.

4. Set the ADRFILT_CTL[0] to 1 (enable state) for all exact match filters that need to be enabled, including the MHASH_EN if enabling the Multicast Hash Filters.

5. Enable receive Ethernet traffic be setting RXEN0 bit to 1 in the RXXG Configuration Register

13.17.7 Receive Address Frame Filtering Byte Order

The address filtering registers are programmed a certain way to achieve an exact match on the DA or SA.

Assume a DA[47:0] = 0x01 0x02 0x03 0x04 0x05 0x06 where bit 40 of the DA[47:0] is the multicast bit.

The address filter register is broken up into three 16-bit registers labeled as HIGH[15:0], MID[15:0], and LOW[15:0].

A byte swap needs to be performed when programming the address filter registers. Hence:

HIGH[15:0] = 0x0605

MID[15:0] = 0x0403

LOW[15:0] = 0x0201

13.18 PAUSE Flow Control

The PM3392 allows 802.3 PAUSE frames to be transmitted out the egress MAC port based on three separate PAUSE frame catalysts. These conditions are discussed further in this section but first a general description of the PM3392 PAUSE frame generation is provided.

The Transmit PAUSE Control Frame logic responds to a Transmit PAUSE Control Request caused by one of these three catalysts:

- Internal ingress FIFO flow control (pause request based on IFLX FIFO fill level)
- External side-band PAUSE Request using the PAUSE and PAUSED pins.
- External host based PAUSE Request.

Each of the three catalysts, if enabled, are logically OR'ed together to form a transmit PAUSE control request to the TXXG associated with a specific channel.

The PM3392 responds by entering a Transmit PAUSE Frame State. After waiting for any current frame transmission to end, a MAC PAUSE control frame will be transmitted. The PAUSE control frame is formatted as follows:

Table 30 PAUSE Control Frame Format

Octets	Frame Field	Source of Information
7 Octets	Preamble	Auto-generated
1 Octet	SFD	Auto-generated
6 Octets	Destination Address	Auto-generated (01-80-c2-00-00-01) Note that DA[7:0] = 0x01, DA[15:8] = 0x80…etc.
6 Octets	Source Address	**TXXG Station Address** register. User defined (TXXG Station Address Registers 3047H, 3048H, 3049H)
2 Octets	Length/Type Field	Auto-generated (88-08)
2 Octets	Opcode Field	Auto-generated (00-01)
2 Octets	PAUSETimer Field	**TXXG PAUSE Timer** register: By default FF-FF. Defined by PAUSE_TIME[0..15], register 304DH.
42 Octets	PAD	Auto-generated
4 Octets	FCS	Auto-generated

The PAUSE frame is stitched together using register-based information and a series of auto-generated fields. As long as the PM3392 is in the Transmit PAUSE Frame State the TXXG will continually send PAUSE control frames every time the internal **TXXG PAUSE Timer Interval** (PAUSE_IVAL[15..0], register 304EH) counts down to zero. In this fashion the egress data-pipe will not be blocked for normal egress data traffic. The **TXXG PAUSE Timer** and **TXXG PAUSE Timer Interval** registers are both programmable. By default the **TXXG PAUSE Timer** register defaults to 0xFFFF and the **TXXG PAUSE Timer Interval** register defaults to 0xCFFF. Both are representative of the number of PAUSE Quanta used in the system. Note that PAUSE Quanta is defined as 512 bit times. The **TXXG PAUSE Timer Interval** will reload to the programmed state when it reaches zero. It is the responsibility of the PAUSE catalyst to hold the input to the TXXG until normal ingress traffic can be resumed. When the catalyst removes the request for PAUSE the TXXG will send out a PAUSE Control frame with the PAUSE timer value of zero.

The three different PAUSE frame catalysts are discussed in more detail below.

13.18.1 Internal Ingress FIFO Flow Control

The POS-PHY ingress FIFO logic (IFLX) has a per-channel, programmable almost-full threshold: register **IFLX Indirect Full/Almost Full Status & Limit** (indirect register 2210H) . When the ingress FIFO fill level for the logical FIFO exceeds the programmed almost-full threshold value an ***internal*** PAUSE flow control request signal is asserted to the TXXG. The TXXG can be programmed to accept POS-PHY ingress FIFO PAUSE flow control requests for egress traffic if the **FCTX** bit is set in the **TXXG Configuration register 1** (register 3040H). When enabled and the internal FIFO PAUSE flow control signal is asserted the TXXG will commence sending 802.3 PAUSE frames. The IFLX logic continues to hold the internal FIFO PAUSE flow control request signal to the TXXG until the FIFO fill level for the logical FIFO of that channel is below the almost-empty threshold value programmed in the **IFLX Indirect Empty/Almost Empty Status & Limit** register (indirect register 2211H). At this time the internal FIFO PAUSE flow control request signal is de-asserted informing the TXXG to cease PAUSE frame flow control by sending a PAUSE Control frame with the PAUSE timer value of zero.

The almost full threshold field (AFTH[13:0]) and almost empty threshold field (AETH[13:0]) in the IFLX registers are in terms of the number of 128-bit words that can be held in the logical FIFO . The table below is one example of what the IFLX settings should be to guarantee a 3 Km (for 9604 byte frames) and 5 Km (for 1522 byte frames) loss less flow control domain.

Table 31 IFLX FIFO Settings for lossless flow control

Register	Value
LOLIM[9:0] (220EH)	0x0000
HILIM[9:0] (220FH)	0x01FE
AFTH[13:0] (2210H)	0x0E66
AETH[13:0] (2211H)	0x0E06

13.18.2 External Side-Band PAUSE Request

The PM3392 has a sideband PAUSE request signal or PAUSE pin, ball AA29. Asserting the PAUSE signal (active high) places the TXXG into a Transmit PAUSE Frame State. The PAUSE signal is to be asserted and held as long as MAC Control PAUSE frames are required to be transmitted from the TXXG. When normal frame reception is desired the PAUSE signal is de-asserted. Upon de-assertion a MAC Control frame with the PAUSE timer value of zero will be transmitted.

The PL4 Bus protocol and implementation of the output scheduling done by the PM3392 device using the PL4MOS does not require intervention by an external agent to support low-level flow control and support of non-blocking operation. The PAUSE signal is provided to allow an external agent to indicate a “coarse” level of flow control independent of that provided by the internal ingress FIFO flow control mechanism.

13.18.3 External Host Based PAUSE Request

The PM3392 allows an external microprocessor to set a register bit to initiate a PAUSE flow control request on a per-channel basis. This is done via the **HOSTPAUSE** bit in the **TXXG Configuration 1 register**, (register 3040H). When the **HOSTPAUSE** bit is set to logic 1 the TXXG is placed in a Transmit PAUSE Frame State. When normal frame reception is desired the HOSTPAUSE register bit is de-asserted by setting it to logic 0. Pause frames are formatted based on the PAUSE_TIME[15..0] value (register 304DH) and the PAUSE_IVAL[15..0] value (register 304EH). Upon de-assertion of HOSTPAUSE a MAC Control frame with the PAUSE timer value of zero will be transmitted.

The PL4 Bus protocol and implementation of the output scheduling done by the PM3392 device using the PL4MOS does not require intervention by an external agent to support low-level flow control and support of non-blocking operation. The HOSTPAUSE register bit is provided to allow an external agent to indicate a "coarse" level of flow control independent of that provided by the internal ingress FIFO flow control mechanism.

13.18.4 Reception of 802.3 PAUSE Frames

As per the 802.3-2000 standard a valid PAUSE frame shall contain following for the PM3392 to respond to the received PAUSE MAC Control Frame:

1. The globally assigned 48-bit multicast address (01-80-C2-00-00-01) or the unicast station address of the MAC.

2. The LENGTH/TYPE field = 8808

3. The PAUSE Opcode of 0001

4. A non-zero value in the PAUSE timer field

As per the 802.3-2000 standard only the LENGTH/TYPE field and the Opcode is needed to increment the PAUSEMACCtrlFramesReceived, regardless of the DA field.

The PM3392 can be programmed to handle ingress PAUSE control frames in the manner as outlined below. This programming is done via the **PARF** bit in the **RXXG Configuration 1 register** (register 2040H) and the **FCRX** bit in the **TXXG Configuration 1** register (register 3040H). The **PARF** bit programs whether or not control frames are passed to the upper layer device. The **FCRX** bit allows the MAC Control sub-layer to respond to a received PAUSE Control frames by pausing the transmitter from transmitting data frames (this response is referred to as "executed" in the following table).

Table 32 PAUSE Frame Programmable Control

PARF	FCRX	PM3392 Action
0	0	PAUSE Frames are ignored and dropped at the PM3392 level.
0	1	PAUSE Frames are executed but are not passed to the upper layer.
1	X	PAUSE Frames are ignored and forwarded to the upper layer device.

Please note as per 802.3-2000 that if the PM3392 is currently executing reception of a PAUSE frame and is currently blocking the data-path from transmission of MAC Data frames (i.e. normal data traffic) that MAC Control PAUSE frames can still be sent. While the TXXG has PAUSED transmission of data frames, the PAUSED pin, ball G9, will be asserted high.

13.19 Ethernet MAC Receive FIFO Overrun Condition

The 802.3 specification defined MAC Control PAUSE frames to inhibit transmission of MAC Data frames between two full-duplex peer Ethernet devices across the Ethernet media. However, support of Ethernet MAC Control PAUSE frames by a device that is a peer of the PM3392 device is an optional feature and so is not guaranteed to be supported on the peer device. In addition, a given application may use the PM3392 device in a manner that does not guarantee loss-less flow control (as described elsewhere in the Operation section, loss-less flow control is a function of the round-trip length of the fiber between the peer Ethernet devices, the maximum frame size of the application, the provisioned amount of buffer space in the PM3392 device for the channel, and the response time of the peer device to the PAUSE frame request and subsequent cessation of frame transmission). This section describes the behavior of the PM3392 Ethernet MAC (RXXG) to a receive FIFO overrun condition.

The response of the Ethernet MAC receive FIFO to a overrun condition depends upon whether any of the data from the frame has been transferred to the downstream ingress FIFO. As mentioned earlier, frame data transfer from the RXXG receive FIFO can start either at an End-Of-Frame indication or when the number of bytes in the frame exceeds the **RXXG Receive FIFO Threshold**.

An Ethernet MAC receive FIFO overrun condition occurs if the FIFO is full and data continues to be received from the XSBI interface(this would be data from within an Ethernet MAC Data frame). The actual FIFO buffer used by the RXXG receive logic does not overflow: the internal pointers used for determining the location in the FIFO to read and write are prevented from actually overrunning. Instead frame data is discarded until there is sufficient buffer space in the receive FIFO for the next frame and reception of subsequent frames resumes on a Start-Of-Frame boundary.

Frame Data Transfer To Ingress FIFO Had Not Started When Ethernet MAC Receive FIFO Overrun Occurs

The receive control logic in the RXXG is responsible for the detection and recovery from a receive FIFO overrun condition. If the overrun is detected before any data from the frame is transferred to the ingress FIFO (which will always be the case if the received frame size is less than the **RXXG Receive FIFO Threshold)** the receive control logic will flag the frame as an overrun frame and it will be discarded in its entirety without being transferred to the ingress FIFO. This is a countable event in the MSTATS receive counter FramesLostDueToInternalMACError. The RXXG will re-synchronize to the next start of a physical packet (preamble/SFD delimiter) and continue Ethernet frame reception. If a receive FIFO overrun condition occurs again, the response of the RXXG is identical to that described above until there is sufficient buffer space to accept either the entire frame or until the number of bytes of the received frame exceeds the **RXXG Receive FIFO Threshold.** Note that reception resumes on a Start-Of-Frame boundary.

Frame Data Transfer To Ingress FIFO Started When Ethernet MAC Receive FIFO Overrun Occurs

In the case that the receive FIFO has started transferring data when the overrun condition occurs, the receive control logic will flag the frame in transfer as an overrun frame and discard all subsequent data from the frame until the next Start-Of-Frame boundary is detected. The frame in transfer when the overrun condition occurs (which is after the **RXXG Receive FIFO Threshold** has been exceeded), will assert an internal receive error status signal to the ingress FIFO. On the PL4 Bus output interface (device pins RDAT[15:0]+/-, RCTL+/-) the frame will be truncated at the location that the overrun occurred within the RXXG receiver and the frame will be marked as erred by having a PL4 End-Of-Packet status of ABORT.

As discussed previously, this is a countable event in the MSTATS receive counter FramesLostDueToInternalMACError. The RXXG will re-synchronize to the next start of a physical packet (preamble/SFD delimiter) and continue Ethernet frame reception. If a receive FIFO overrun condition occurs again, the response of the RXXG is identical to that described above until there is sufficient buffer space to accept either the entire frame or until the number of bytes of the received frame exceeds the **RXXG Receive FIFO Threshold.** Note that reception resumes on a Start-Of-Frame boundary.

13.20 Using the Performance Monitoring Features

The PM3392 has associated with it an MSTAT block that is used to accumulate Ethernet specific counts for supporting management agents such RMON, SNMP, and Etherlike interfaces. The MSTAT provides counter width support for compliance with 802.3-2000 rollover requirements of 58 minutes, except the **OctetsReceived, OctetsReceivedOK, OctetsTransmitted, OctetsTransmittedOK**. These counters will rollover in approximately 15 minutes. The MSTAT supports full system probing capability via the use of full counter snapshot to shadow registers. The Ethernet statistics counters maintained on the PM3392 device can be transferred to shadow registers by writing to logic 1 the SNAP bit of the MSTAT Control register. This supports software polling of the MSTAT registers with a snapshot being taken at fixed time intervals; the data can then be read out of the shadow registers as a background software process over the PM3392 microprocessor interface.

Incorporated into the MSTAT block is a fully programmable interrupt array enabling per counter rollover monitoring with interrupt reporting. This supports customer software that requires an “on-demand” read of the Ethernet statistics.

13.21 Interrupt Handling

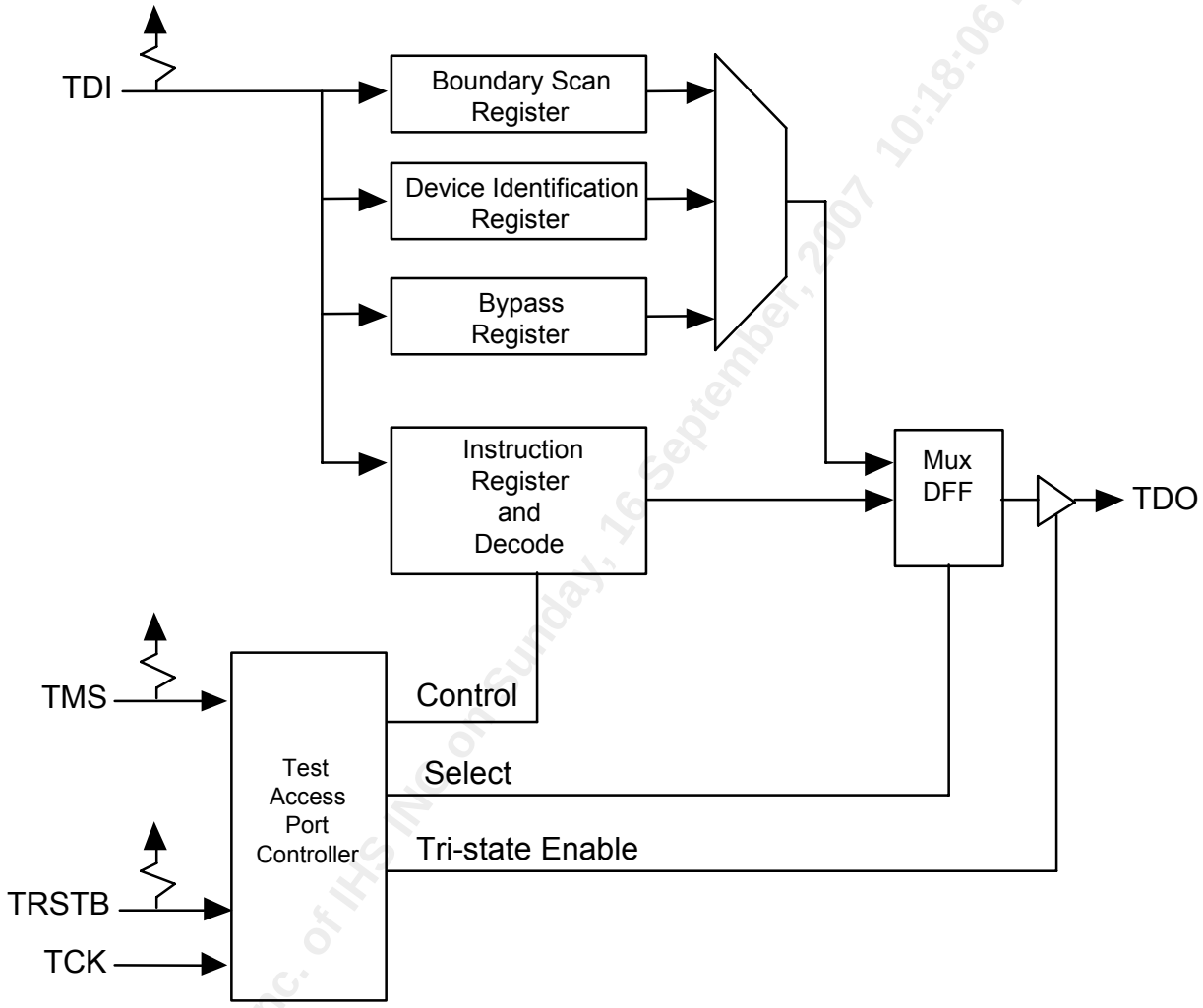
The PM3392 signals the host processor via the use of the INTB active low pin. When INTB is asserted (logic 0) the host processor can interrogate the PM3392 for the source of the active interrupt by reading the **PM3392 Master Interrupt Status** register: this identifies the interrupt source at the block level. Further register accesses are required to the block in question to determine the cause of an active interrupt and to acknowledge the interrupt source. Reading of the **PM3392 Master Interrupt Status** register has no side-effects. Interrupt status is set to logic 1 to indicate a pending interrupt of the specified type. The interrupt signal is asserted only if the interrupt status is true and the interrupt source is not masked.

To clear an interrupt the host processor must acknowledge the interrupt sources. This is done by reading the block-level interrupt source as decoded by the Master Interrupt Status Register. A read from the block-level interrupt source will clear the block level interrupt. Note that there may be more than one block level interrupt asserted. To clear the device level interrupt all block level interrupts must either be cleared or masked off. Every block level interrupt source has a corresponding interrupt mask bit.

13.22 JTAG Support

The PM3392 supports the IEEE Boundary Scan Specification as described in the IEEE 1149.1 standards. The Test Access Port (TAP) consists of the five standard pins, TRSTB, TCK, TMS, TDI and TDO used to control the TAP controller and the boundary scan registers. The TRSTB input is the active-low reset signal used to reset the TAP controller. TCK is the test clock used to sample data on input, TDI and to output data on output, TDO. The TMS input is used to direct the TAP controller through its states. The basic boundary scan architecture is shown below.

Figure 24 Boundary Scan Architecture



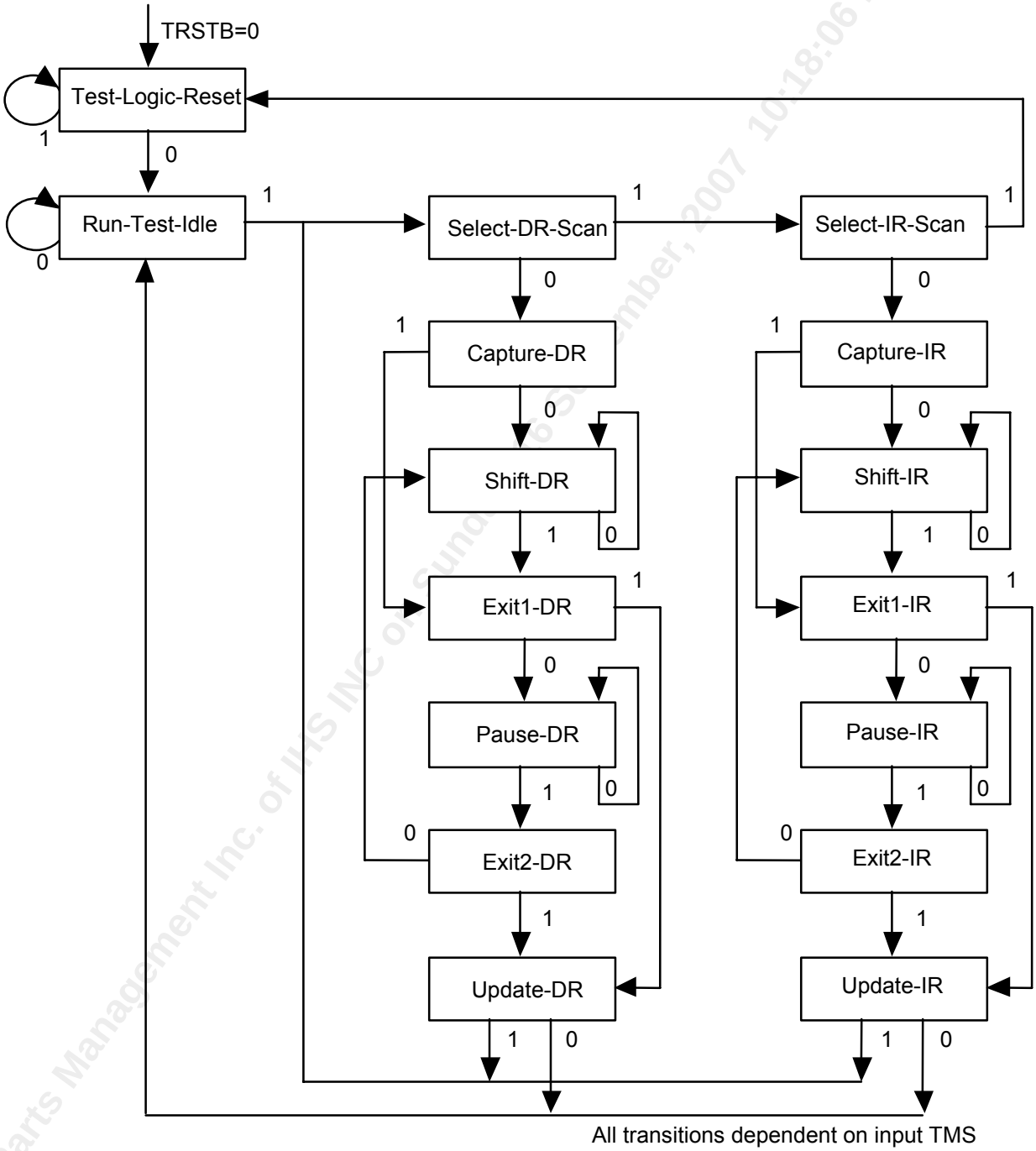
The boundary scan architecture consists of a TAP controller, an instruction register with instruction decode, a bypass register, a device identification register and a boundary scan register. The TAP controller interprets the TMS input and generates control signals to load the instruction and data registers. The instruction register with instruction decode block is used to select the test to be executed and/or the register to be accessed. The bypass register offers a single-bit delay from primary input, TDI to primary output, TDO. The device identification register contains the device identification code.

The boundary scan register allows testing of board inter-connectivity. The boundary scan register consists of a shift register place in series with device inputs and outputs. Using the boundary scan register, all digital inputs can be sampled and shifted out on primary output, TDO. In addition, patterns can be shifted in on primary input, TDI and forced onto all digital outputs.

13.22.1 TAP Controller

The TAP controller is a synchronous finite state machine clocked by the rising edge of primary input, TCK. All state transitions are controlled using primary input, TMS. The finite state machine is described below.

Figure 25 TAP Controller Finite State Machine



Test-Logic-Reset

The test logic reset state is used to disable the TAP logic when the device is in normal mode operation. The state is entered asynchronously by asserting input, TRSTB. The state is entered synchronously regardless of the current TAP controller state by forcing input, TMS high for 5 TCK clock cycles. While in this state, the instruction register is set to the IDCODE instruction.

Run-Test-Idle

The run test/idle state is used to execute tests.

Capture-DR

The capture data register state is used to load parallel data into the test data registers selected by the current instruction. If the selected register does not allow parallel loads or no loading is required by the current instruction, the test register maintains its value. Loading occurs on the rising edge of TCK.

Shift-DR

The shift data register state is used to shift the selected test data registers by one stage. Shifting is from MSB to LSB and occurs on the rising edge of TCK.

Update-DR

The update data register state is used to load a test register's parallel output latch. In general, the output latches are used to control the device. For example, for the EXTEST instruction, the boundary scan test register's parallel output latches are used to control the device's outputs. The parallel output latches are updated on the falling edge of TCK.

Capture-IR

The capture instruction register state is used to load the instruction register with a fixed instruction. The load occurs on the rising edge of TCK.

Shift-IR

The shift instruction register state is used to shift both the instruction register and the selected test data registers by one stage. Shifting is from MSB to LSB and occurs on the rising edge of TCK.

Update-IR

The update instruction register state is used to load a new instruction into the instruction register. The new instruction must be scanned in using the Shift-IR state. The load occurs on the falling edge of TCK.

The Pause-DR and Pause-IR states are provided to allow shifting through the test data and/or instruction registers to be momentarily paused.

Boundary Scan Instructions

The following is a description of the standard instructions. Each instruction selects an serial test data register path between input, TDI and output, TDO.

BYPASS

The bypass instruction shifts data from input, TDI to output, TDO with one TCK clock period delay. The instruction is used to bypass the device.

EXTEST

The external test instruction allows testing of the interconnection to other devices. When the current instruction is the EXTEST instruction, the boundary scan register is place between input, TDI and output, TDO. Primary device inputs can be sampled by loading the boundary scan register using the Capture-DR state. The sampled values can then be viewed by shifting the boundary scan register using the Shift-DR state. Primary device outputs can be controlled by loading patterns shifted in through input TDI into the boundary scan register using the Update-DR state.

SAMPLE

The sample instruction samples all the device inputs and outputs. For this instruction, the boundary scan register is placed between TDI and TDO. Primary device inputs and outputs can be sampled by loading the boundary scan register using the Capture-DR state. The sampled values can then be viewed by shifting the boundary scan register using the Shift-DR state.

IDCODE

The identification instruction is used to connect the identification register between TDI and TDO. The device's identification code can then be shifted out using the Shift-DR state.

STCTEST

The single transport chain instruction is used to test out the TAP controller and the boundary scan register during production test. When this instruction is the current instruction, the boundary scan register is connected between TDI and TDO. During the Capture-DR state, the device identification code is loaded into the boundary scan register. The code can then be shifted out output, TDO using the Shift-DR state.

13.23 Receive PCS Layer Error Handling

This section describes the behavior of the RX PCS during error conditions and during a fault condition. The RX PCS conforms to the 802.3ae standard for all error and fault conditions as described in Clauses 49 and 46 of the 802.3ae standard. All errors that the RX PCS encounters are communicated to the RX MAC by latching the error condition until the end of packet is received. All fault conditions are terminated in the RX PCS block (i.e. will not be propagated to the system interface).

The possible error conditions that exist are: Receive State Machine Errors, Invalid 66-bit blocks, Error control code, Loss of sync, Bit Error State Machine.

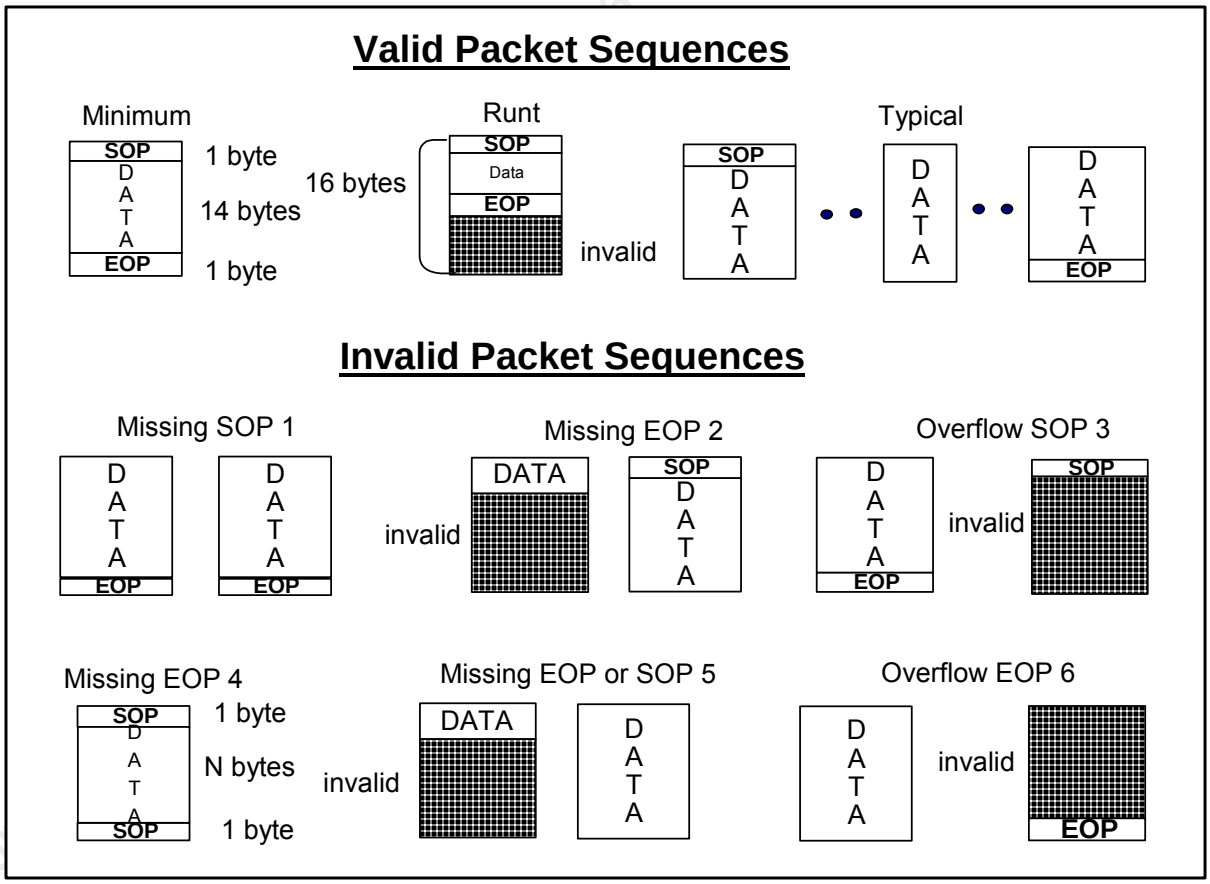
The possible fault conditions are Local Fault or Remote Fault.

The sections to follow will describe that details of the error conditions and the fault conditions.

13.23.1 Receive State Machine Errors

The RX PCS follows the Receive State Machine in Figure 49-14 of 802.3ae. When the Receive State Machine transitions to RX_E (Error state), the RX PCS will replace the payload with all 0's, but still drive the byte_valid signals for all 8 bytes. The RX PCS will latch the error condition and when a good end-of-frame occurs, will drive the appropriate PCS layer error signal to the RX MAC.

Figure 26 Expected Packet Formats



13.23.2 Invalid 66-Bit Blocks

There are three types of invalid 66-bit blocks:

- The sync field has a value of 00 or 11
- The type field contains a reserved value
- Any control character contains a value not in Table 49-1 of 802.3ae

When the RX PCS encounters an invalid 66-bit block the Receive State Machine will transition to the RX_E(Error) state.

13.23.3 Error Control Code

The 802.3ae defines an Error control code to be eight control characters and its first character is an Error control code (0x1e). This condition can only occur when the type field is 0x1e and the first control code (C0[6:0]) is 0x1e and the rest of the control codes (C1-C7) are either 0x0 or 0x1e. When this is encountered the Receive State Machine will transition to the RX_E(Error) state.

13.23.4 Loss Of Sync

Once the RX PCS has achieved the initial synchronization there are two ways for the RX PCS to lose sync. If within receiving 64 66-bit blocks there are 32 or more invalid sync fields(00,11) or if the SYNC_ERR signal is asserted

When a Loss of sync occurs, the RX PCS will terminate and error any packet that is being transferred across the system interface and assert the RX_LOS status signal and LOS status bit. The assertion of the LOS status bit can cause an interrupt to be generated if enabled.

If the Loss of sync is due to the assertion of SYNC_ERR a sync_err status bit will also be asserted. The assertion of the sync_err status bit can cause an interrupt to be generated if enabled.

13.23.5 Bit Error State Machine

The RX PCS monitors bit errors in the sync field. The Bit Error State Machine is defined in Figure 49-12 of 802.3ae. The Bit Error State Machine uses an 125us timer to give a window for a high bit error indication. A high bit error is defined as 16 bit errors within an 125us window. When a high bit error occurs the hi_ber status bit will be asserted, this can cause an interrupt to be generated if enabled. When the BER State Machine goes into a HI_BER state the Receive State Machine will jump to the RX_INIT state causing no data to be transferred to the RX MAC. The Receive State Machine will remain in the RX_INIT state until the HI_BER state is cleared.

The 125us timer is a 14-bit counter that rolls over to 0 when the counter reaches the default done value. The default done value on reset is 20162 or 0x4EC2. The calculation for the 125us timer is 125us/6.2ns(period of sys_clkx2).

13.23.6 Fault Conditions

Fault signaling is communicated by using the defined Sequence Ordered Set in 802.3ae The RX PCS also includes part of the RS layer, therefore the section in clause 46 Link fault signaling also applies.

Local Fault

A Local Fault(LF) is defined as being one of the Order Set Type Control codes defined in figure 49-7 of 802.3ae with the D3 or D7 byte field being 0x01. When at least 4 LF messages are received the RX PCS will assert the RX_LF status signal and assert the Link Fail status bit. The assertion of the RX_LF status signal or Link Fail status bit can cause an interrupt to be generated if enabled. After achieving a Local Fault state, when there is an absence of Local Fault messages for at least 64 sys_clkx2's the RX PCS will de-assert RX_LF status signal and the Link Fail status bit. The Local Fault messages are terminated in the RX PCS, only IDLE's are passed to the RX MAC.

Remote Fault

A Remote Fault(RF) is defined as being one of the Order Set Type Control codes defined in figure 49-7 of 802.3ae with the D3 or D7 byte field being 0x02. When at least 4 RF messages are received the RX PCS will assert the RX_RF status signal. The assertion of the RX_RF status signal can cause an interrupt to be generated if enabled. After achieving a Remote Fault state, when there is an absence of Remote Fault messages for at least 64 sys_clkx2's the RX PCS will de-assert RX_RF status signal. The Remote Fault messages are terminated in the RX PCS, only IDLE's are passed to the RX MAC.

13.24 Transmit PCS Layer Error Handling

This section describes the behavior of the TX PCS during error conditions and during a fault condition. The TX PCS conforms to the 802.3ae standard for all error and fault conditions as described in Clauses 49 and 46 of the 802.3ae standard.

The possible error conditions that exist are: Transmit State Machine Errors, MAC transmit errors, Loss of sync from the RX PCS, HI_BER from the RX PCS.

The possible fault conditions are Local Fault or Remote Fault.

The sections to follow will describe that details of the error conditions and the fault conditions.

13.24.1 Receive State Machine Errors

The TX PCS follows the Transmit State Machine in Figure 49-14 of 802.3ae. When the Transmit State Machine transitions to TX_E (Error state), the TX PCS will insert the 66-bit ERROR code as defined by the 802.3ae.

13.24.2 MAC Transmit Errors

The TX PCS will also insert the 66-bit ERROR code when the TX MAC asserts the SYSTEM_ERROR signal. The SYSTEM_ERROR signal is asserted when the TX MAC has encountered an error.

13.24.3 Loss of Sync From The RX PCS

When the RX PCS encounters a loss of sync the TX PCS will treat this as a Local Fault condition. When in a Local Fault condition the TX PCS will hold off data from the TX MAC and stream out the Remote Fault code alternating between IDLE's and Remote Fault until the Local Fault condition is cleared.

13.24.4 HI BER From The RX PCS

When the RX PCS encounters a High Bit ERROR the TX PCS will treat this as a Local Fault condition. When in a Local Fault condition the TX PCS will hold off data from the TX MAC and stream out the Remote Fault code alternating between IDLE's and Remote Fault until the Local Fault condition is cleared.

13.24.5 Local Fault Received From The RX PCS

When the RX PCS receives the Local Fault code the TX PCS will treat this as a Local Fault condition. When in a Local Fault condition the TX PCS will hold off data from the TX MAC and stream out the Remote Fault code alternating between IDLE's and Remote Fault until the Local Fault condition is cleared.

13.24.6 Remote Fault Received From The RX PCS

When the RX PCS receives the Remote Fault code the TX PCS will treat this as a Remote Fault condition. When in a Remote Fault condition the TX PCS will hold off data from the TX MAC and stream out IDLE's until the Remote Fault condition is cleared.

13.25 MDIO Access

The MDIO block communicates between the host processor and an external MII physical device by means of a two wire interface. The MDIO block produces a 2.5 Mhz MDC (Management Data Clock) clock by dividing down the system clock. The control information is synchronously driven by the MDIO with respect to MDC and is sampled synchronously by the PHY. Status information is driven by the PHY synchronously with respect to MDC and is sampled synchronously by MDIO.

This section describes how the host can use on-chip registers to access the external MII PHY. The bit definitions and details of these registers are defined in the Normal Mode Register Description. There are four different operation namely address, write, read and read-post-increment.

13.25.1 MII Address Operation

1. Check the busy bit in the status register (poll the busy bit in the status register or wait for the busyi interrupt) make sure it is low.

2. Program the port address, and device address for the particular MDIO manageable device (MMD) to be accessed in the MMD PHY address register.

3. Program the CTLAD Register with the address of the register to be accessed by subsequent data transaction frame.

4. Set the LCTLA bit in the MDIO command register. This will initiate the MII address operation and the busy bit in the status register will be asserted until the operation is complete.

13.25.2 MII Data Write Operation

1. Transmit the address frame as specified in the previous section.

2. Wait for the busy signal to be deasserted.

3. Program the CTLAD Register with the control data to be transmitted.

4. Set the LCTLD bit in the MDIO command register. This will initiate the MII data write operation and the busy bit in the status register will be asserted until the operation in complete.

13.25.3 MII Data Read Operation

1. Transmit the address frame as specified above

2. Wait for the busy signal to be deasserted.

3. Set the RSTAT bit in the MDIO command register. This will initiate the read operation and set the busy bit in the status register until the operation is complete. Once the busy bit is deasserted read the mdio read status data register PRSD.

13.25.4 MII RDINC Operation

1. Wait for the busy signal to be deasserted.

2. Program the port address, and device address for the particular MDIO manageable device (MMD) to be accessed in the MMD PHY address register.

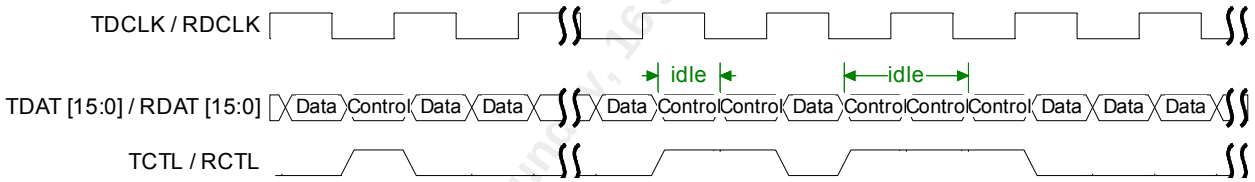
3. Set the RDINC bit in the MDIO command register. This will initiate a post-read-increment-address operation and set the busy bit in the status register until the operation is complete. Upon completion of the read operation the MMD will increment the address register by one. Once the busy bit is deasserted read the PRSD.

14 Functional Timing

14.1 PL4 Interface Data Path and FIFO Status Timing

A timing diagram of the PL4 interface data path signals is shown in Figure 27. This diagram is applicable to either the transmit or the receive interface. TCTL/RCTL is high when TDAT[15:0]/RDAT[15:0] contains control words. Idle periods correspond to back-to-back control words. Data and control words are updated on both the rising and falling edges of the TDCLK/RDCLK. The actual clock rate used in practice is determined by the application at hand (up to a maximum of 700 MHz data rate divided by 2).

Figure 27 PL4 Interface Data Path Functional Timing



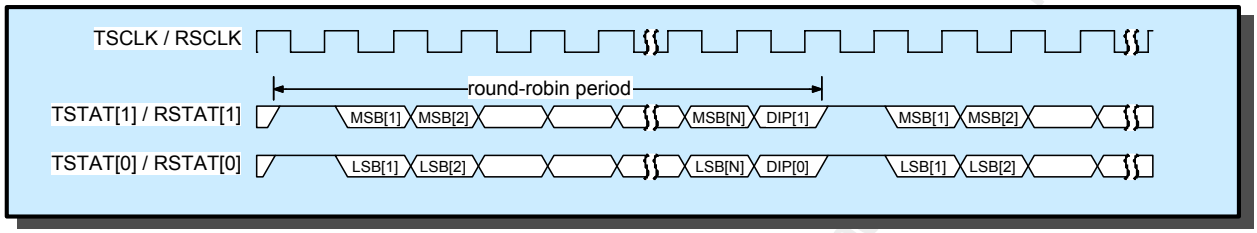
Complete packets or shorter bursts (fragment of larger packets) may be transferred. The maximum configured payload data transfer size must be a multiple of 16 bytes. Control words are inserted only between burst transfers. Once a transfer has begun, data words are sent uninterrupted until end-of-packet or the burst transfer size is reached, whichever comes first. The interval between the end of a given transfer and the next payload control word (marking the start of another transfer) consists of zero or more idle control words.

The minimum and maximum supported packet lengths are determined by the application. For ease of implementation however, successive start-of-packets must occur not less than 8 cycles apart, where a cycle is one control or data word. The gap between shorter packets is filled with idle control words.

Payload data bytes are transferred over the interface in the same order as they would be transmitted or received on the line side. The most significant bits (MSBs) of the constituent bytes correspond to bits 15 and 7. The byte with MSB at bit 15 is transmitted or received first on the line side relative to the byte with MSB at bit 7. On payload transfers that do not end on an even byte boundary, the unused byte (after the last valid byte) is set to all zeroes.

Figure 28 shows the PL4 Interface FIFO Status functional timing for the S/UNI-1x10GE device. FIFO status information is sent periodically over TSTAT[1:0] from the S/UNI-1x10GE to the Link Layer device, and over the RSTAT[1:0] from the Link Layer to the S/UNI-1x10GE. The transmit and receive status channels operate independently of each other. STAT[1:0]/RSTAT[1:0] is updated on the rising edge of TSCLK/RSCLK. TSCLK/RSCLK runs at a nominal rate of $1/8^{th}$ the data rate of the PL4 transmit/receive interface.

Figure 28 PL4 Interface FIFO Status Functional Timing



The FIFO status of each port is encoded in a 2-bit data structure as per PL4 specification. The port sequences on the transmit and receive interfaces may be configured differently from each other. The "1 1" pattern is reserved for in-band framing. It must be sent once prior to the start of the FIFO status sequence. A DIP-2 odd parity checksum is sent at the end of each complete sequence, immediately before the "1 1" framing pattern. The DIP-2 code is computed over all preceding FIFO status indications sent after the last "1 1" framing pattern as per PL4 specification. A continuous stream of repeated "1 1" framing patterns may be sent to indicate an error condition.

14.2 XSBI Functional Timing

This section shows the functional timing relationship for the Ingress and Egress paths. No propagation delays are shown.

Figure 29 Ingress Timing Diagram

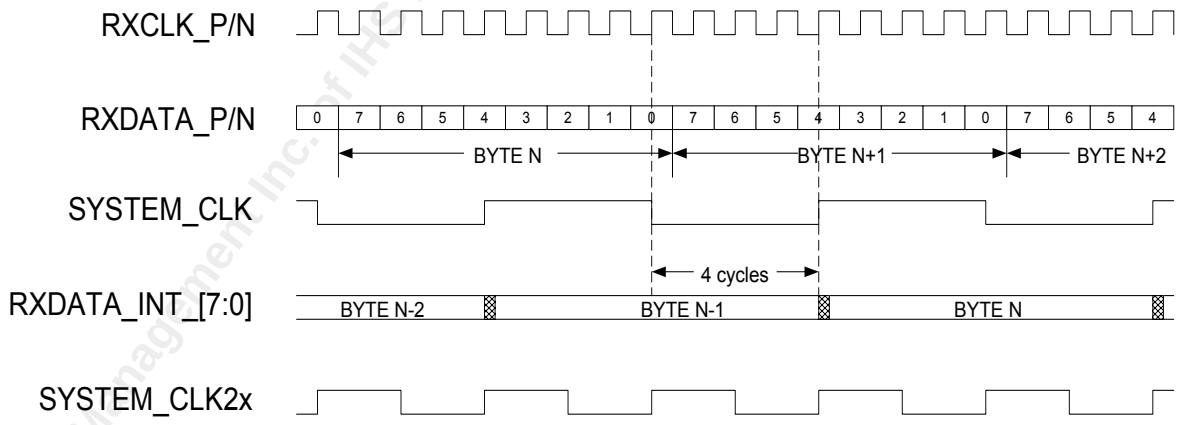
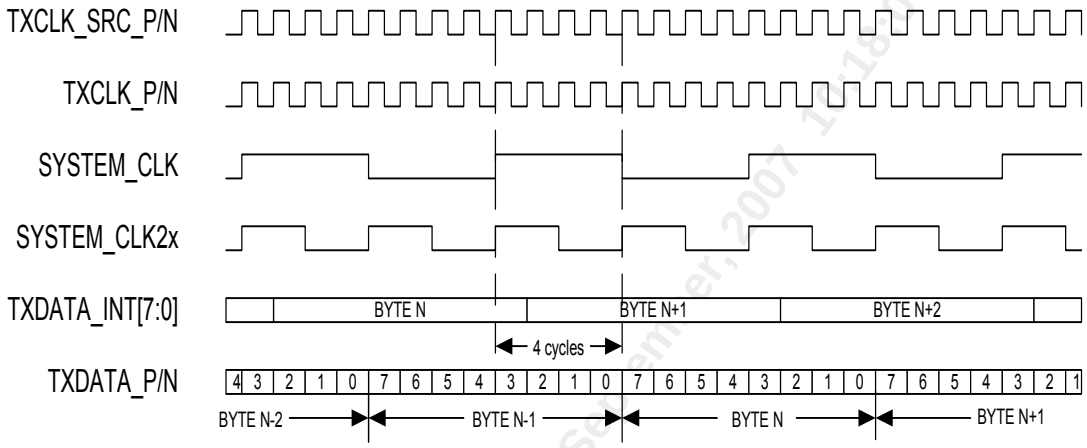


Figure 30 Egress Timing Diagram



15 Absolute Maximum Ratings

Maximum rating is the worst-case limits that the device can withstand without sustaining permanent damage. They are not indicative of normal mode operation conditions.

Table 33 Maximum Ratings

Storage Temperature	-40°C to +125°C
Supply Voltage	See Table 40 below
Voltage on Any Pin	See Table 41 below
Static Discharge Voltage	±1000 V
Latch-Up Current	±100 mA
DC Input Current	±20 mA
Lead Temperature	+225°C
Voltage of Overshoot of duration <10ns on any pin (unless otherwise specified)	-2.0V to VDDo+2.0 V
Junction Temperature under bias (correct operation not necessarily guaranteed)	-40°C to +125°C

Table 34 Core Supply Voltage Specs

Process	Absolute Min VDD [V]	Absolute Max VDD [V]
0.18um CMOS Core Supply	-0.5	2.2

Table 35 I/O Supply Voltage Specs

Process	Absolute Min VDDo [V]	Absolute Max VDDo [V]	Min Voltage on any pin Vx [V]	Max Voltage on any pin Vx [V]
3.3V Tolerant 3.3V Supply	VDD (core)- 0.5	4.0	-0.5	VDDo +0.5

16 D.C. Characteristics

T_A = -40°C to T_J = +125°C, V_{VDDI} = $V_{VDDItypical}$ ± 5%, V_{VDDO} = $V_{DDOtypical}$ ± 5% (Typical Conditions: T_A = 25°C, V_{VDDI} = 1.8V, V_{VDDO} = 3.3V, V_{AVDL} = 1.8V, V_{AVDH} = 3.3V)

Table 36 D.C. Characteristics

Symbol	Parameter	Min	Typ	Max	Units	Conditions
V_{VDDI}	Power Supply	1.71	1.8	1.89	Volts	
V_{VDDO}	Power Supply	3.135	3.3	3.465	Volts	
V_{AVDL}	Power Supply	1.71	1.8	1.89	Volts	
V_{AVDH}	Power Supply	3.135	3.3	3.465	Volts	
VIL	Input Low Voltage			0.8	Volts	Guaranteed Input Low voltage
VIH	Input High Voltage	2.0			Volts	Guaranteed Input High voltage
VOL	Output or Bi-directional Low Voltage		0.1	0.4	Volts	Guaranteed output Low voltage at VDDO=2.97V and IOL=maximum rated for pad.
VOH	Output or Bi-directional High Voltage	2.4	2.7		Volts	Guaranteed output High voltage at VDDO=2.97V and IOH=maximum rated current for pad.
VT-	Reset Input Low Voltage			0.8	Volts	Applies to RSTB and TRSTB only.
VT+	Reset Input High Voltage	2.2			Volts	Applies to RSTB and TRSTB only.
IILPU	Input Low Current	-300	-120	-10	µA	VIL = GND. Notes 1 and 3.
IIHPU	Input High Current	-10	0	+10	µA	VIH = VDDO. Notes 1 and 3.
IILPD	Input Low Current	-10	0	+10	µA	VIL = GND. Notes 5 and 3.
IIHPD	Input High Current	-300	-120	-10	µA	VIH = VDDO. Notes 5 and 3.
IIL	Input Low Current	-10	0	+10	µA	VIL = GND. Notes 2 and 3.
IIH	Input High Current	-10	0	+10	µA	VIH = VDDO. Notes 2 and 3.
CIN	Input Capacitance		5		pF	tA=25°C, f = 1 MHz
COUT	Output Capacitance		5		pF	tA=25°C, f = 1 MHz
CIO	Bi-directional Capacitance		5		pF	tA=25°C, f = 1 MHz
LVDS RECEIVER DC SPECIFICATIONS (PINS: TDCLK+/-,TDAT[15:0]+/-, TCTL+/-)						
VICM	LVDS Input Common-Mode Range	0		2.4	V	Applies to LVDS inputs.

Symbol	Parameter	Min	Typ	Max	Units	Conditions
|VIDM|	LVDS Input Differential Sensitivity	100		600	mV	Applies to LVDS inputs.
RIN	LVDS Differential Input Impedance	80	100	120	Ω	Applies to LVDS inputs.
LVDS OUTPUT DC SPECIFICATIONS (PINS: RDCLK+/-, RDAT[15:0]+/-, RCTL+/-)						
VLOH	LVDS Output voltage high		1375	1475	mV	RLOAD=100Ω ±1%
VLOL	LVDS Output voltage low	925	1025		mV	RLOAD=100Ω ±1%
VODM	LVDS Output Differential Voltage	100	350	600	mV	RLOAD=100Ω ±1%
VOCM	LVDS Output Common-Mode Voltage	1125	1200	1275	mV	RLOAD=100Ω ±1%
DC-COUPLED PECL: DC SPECIFICATION (PINS: PL4_RCLK+/-)						
VPECLI+	Input DC PECL High Voltage	VAVDH – 1.165	VAVDH – 0.955	VAVDH – 0.880	Volts	Applies to PL4_RCLK+/- inputs only.
VPECLI-	Input DC PECL Low Voltage	VAVDH – 1.810	VAVDH – 1.700	VAVDH – 1.470	Volts	Applies to PL4_RCLK+/- inputs only.
LVDS XSBI DC SPECIFICATIONS (PINS: TXDATA1-4[0-3]+/-, RXDATA1-4[0-3]+/-)						
V_I	LVDS Input Voltage Range	900		1600	mV	|Vgpd| <50mV See note 4
V_{ID}	LVDS Input Differential Voltage	100		600	mV	|Vgpd| <50mV See note 4
R_{IN}	LVDS Differential Input Impedance	80		120	Ω	
V_{OD}	LVDS Output Differential Voltage	250		400	mV	R_{LOAD}=100Ω ±1%
V_{OS}	LVDS Output Offset Voltage	1125		1375	mV	R_{LOAD}=100Ω ±1%
$\|V_{OD}\|$	Change in $\|V_{OD}\|$ between “0” and “1”			50	mV	R_{LOAD}=100Ω ±1%
$\|V_{OS}\|$	Change in $\|V_{OS}\|$ between “0” and “1”			50	mV	R_{LOAD}=100Ω ±1%
t_R, t_F	20%-80% rise and fall times	100		400	ps	

Notes on D.C. Characteristics:

1. Input pin or bi-directional pin with internal pull-up resistor.
2. Input pin or bi-directional pin without internal pull-up resistor.
3. Negative currents flow into the device (sinking), positive currents flow out of the device (sourcing).
4. |Vgpd| is the ground potential differential between PMA client and PMA

5. Input pin or bi-directional pin with internal pull-down resistor.

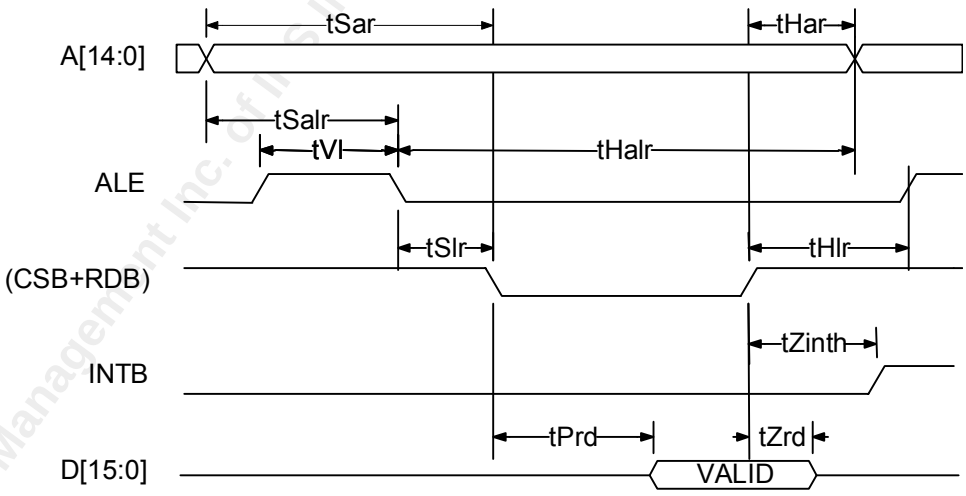
17 Microprocessor Interface Timing Characteristics

T_A = -40°C to T_J = +125°C, V_{VDDI} = $V_{VDDItypical}$ ± 5%, V_{VDDO} = $V_{DDOtypical}$ ± 5%
(Typical Conditions: T_J = 25°C, V_{VDDI} = 1.8V, V_{VDDO} = 3.3V)

Table 37 Microprocessor Interface Multiplexed Read Timing

Symbol	Parameter	Min	Max	Units
tS_{AR}	Address to Valid Read Set-up Time	10		ns
tH_{AR}	Address to Valid Read Hold Time	5		ns
tS_{ALR}	Address to Latch Set-up Time	10		ns
tH_{ALR}	Address to Latch Hold Time	10		ns
tV_{L}	Valid Latch Pulse Width	5		ns
tS_{LR}	Latch to Read Set-up	0		ns
tH_{LR}	Latch to Read Hold	5		ns
tP_{RD}	Valid Read to Valid Data Propagation Delay		70	ns
tZ_{RD}	Valid Read Negated to Output Tri-state		20	ns
tZ_{INTH}	Valid Read Negated to INTB High		50	Ns

Figure 31 Microprocessor Interface Read Timing



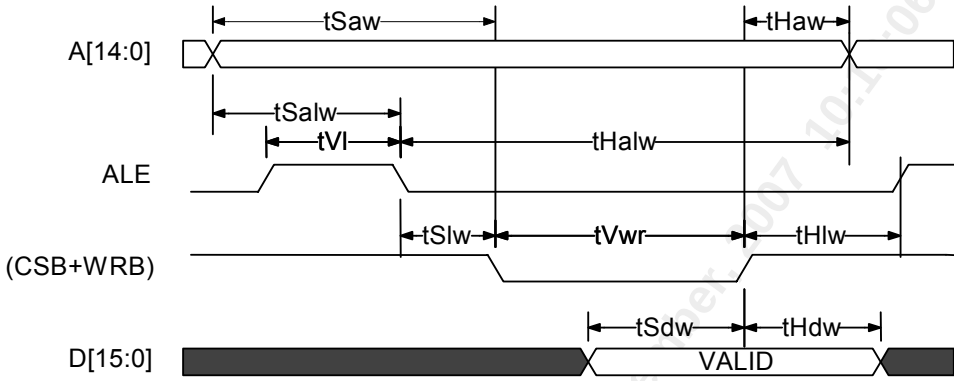
Notes on Microprocessor Interface Read Timing:

1. Output propagation delay time is the time in nanoseconds from the 1.4 Volt point of the reference signal to the 1.4 Volt point of the output.
2. Maximum output propagation delays are measured with a 100 pF load on the Microprocessor Interface data bus, (D[15:0]).
3. A valid read cycle is defined as a logical AND of the CSB and the RDB signals while the WRB signal is not asserted.
4. In non-multiplexed address/data bus architectures, ALE should be held high so parameters tS_{ALR}, tH_{ALR}, tV_L, tS_{LR}, and tH_{LR} are not applicable.
5. Parameter tH_{AR} is not applicable if address latching is used.
6. When a set-up time is specified between an input and a clock, the set-up time is the time in nanoseconds from the 1.4 Volt point of the input to the 1.4 Volt point of the clock.
7. When a hold time is specified between an input and a clock, the hold time is the time in nanoseconds from the 1.4 Volt point of the input to the 1.4 Volt point of the clock.

Table 38 Microprocessor Interface Write Access

Symbol	Parameter	Min	Max	Units
tS_{AW}	Address to Valid Write Set-up Time	10		ns
tS_{DW}	Data to Valid Write Set-up Time	20		ns
tS_{ALW}	Address to Latch Set-up Time	10		ns
tH_{ALW}	Address to Latch Hold Time	10		ns
tV_L	Valid Latch Pulse Width	5		ns
tS_{LW}	Latch to Write Set-up	0		ns
tH_{LW}	Latch to Write Hold	5		ns
tH_{DW}	Data to Valid Write Hold Time	5		ns
tH_{AW}	Address to Valid Write Hold Time	5		ns
tV_{WR}	Valid Write Pulse Width	40		ns

Figure 32 Microprocessor Interface Write Timing



Notes on Microprocessor Interface Write Timing:

1. A valid write cycle is defined as a logical AND of the CSB and the WRB signals.
2. In non-multiplexed address/data bus architectures, ALE should be held high so parameters tS_{ALW}, tH_{ALW}, tV_L, tS_{LW}, and tH_{LW} are not applicable.
3. Parameter tH_{AW} is not applicable if address latching is used.
4. When a set-up time is specified between an input and a clock, the set-up time is the time in nanoseconds from the 1.4 Volt point of the input to the 1.4 Volt point of the clock.
5. When a hold time is specified between an input and a clock, the hold time is the time in nanoseconds from the 1.4 Volt point of the input to the 1.4 Volt

18 AC Timing Characteristics

T_A = -40°C to T_J = +125°C, V_{VDDI} = $V_{VDDI typical}$ ± 5%, V_{VDDO} = $V_{DDO\ typical}$ ± 5%
(Typical Conditions: T_A = 25°C, V_{VDDI} = 1.8V, V_{VDDO} = 3.3V)

18.1 PL4 Interface Timing

18.1.1 Interpretation of the SPI-4-II/PL4 Standards for LVDS Electrical, Jitter and Skew Specifications

This section provides clarification and supplemental information regarding the Electrical Specifications of PMC-Sierra's POS-PHY Level 4 interface implemented on the Xenon Family of devices. The information provided in this section is specific to PMC-Sierra's implementation of the PL4/SPI-4 Phase II standard and provides additional clarification of performance specifications outlined in the standards that are ambiguous or incomplete.

LVDS Electrical

The LVDS electrical specifications for the PMC PL4/SPI-4-II interfaces are given in Table 39. It should be noted that the PL4/SPI-4-II standards just reference the IEEE and TIA LVDS standards for the LVDS electrical specifications. In some cases these two standards differ, as well as being different than the SFI-4-I standard.

Table 39 LVDS Electrical Specifications

Symbol	Parameter	Comments		TYP	Units
t_R, t_F	Input 20%-80% Rise & Fall Times	See Notes 1 & 2		0.36	UI
t_R, t_F	Output 20%-80% Rise & Fall Times			175	ps

Notes:
1. Same as the PL4/SPI-4-II standards.
2. Rise/fall times are measured from the 20% to 80% thresholds

18.1.2 Clock and Data Jitter Methods

18.1.3 Reference Clock Interface

The data in Table 40 and Table 41 below is for the reference clock when in Master mode. This data is not part of the PL4/SPI-4-II standards (no specifications are given for refclk input).

Table 40 Reference Clock Timing Specifications for divBy2 (PL4_REFCLK)

Symbol	Parameter	Min	Typ	Max	Units	Comments
fD	Reference Clock Frequency	311		350	MHz	
DCref	Reference Clock Duty Cycle	40		60	%	

DJref	Reference Clock Deterministic Jitter (pk-to-pk between fD/1000 & 8MHz) (pk-to-pk between fD/1000 & fD)			 0.01 0.05	 UI UI	Not in the PL4/SPI-4-II standards
TJref	Reference Clock Total Jitter (pk-to-pk between fD/1000 & fD)			0.10	UI	
tRFref	Reference Clock Rise / Fall Times		1.0		ns	

Notes on Reference Clock Timing:

1. Master Mode and divBy2 (REFSEL[0] = "1" & REFSEL[1] = “0”).
2. The Unit Interval (UI) is the reciprocal of the symbol rate.
3. Total jitter includes both deterministic jitter and random jitter. The random jitter is the total jitter minus the actual deterministic jitter.
4. Values are measured with each PECL/LVDS input DC coupled into a 50 Ohm impedance (100 Ohms differential impedance).
5. Rise time is measured from the 10% threshold of the reference signal to the 90% threshold of the reference signal.
6. Fall time is measured from the 90% threshold of the reference signal to the 10% threshold of the reference signal.
7. Duty cycle and jitter are specified between differential crossings of the 50% threshold of the reference signal.

Table 41 Reference Clock Timing Specifications for divBy4 (PL4_REFCLK)

Symbol	Parameter	Min	Typ	Max	Units	Comments
fD	Reference Clock Frequency	155.5		175	MHz	Not in the PL4/SPI-4-II standards
DCref	Reference Clock Duty Cycle	40		60	%	
DJref	Reference Clock Deterministic Jitter (pk-to-pk between fD/500 & 8MHz) (pk-to-pk between fD/500 & 2*fD)			 0.01 0.05	 UI UI	
TJref	Reference Clock Total Jitter (pk-to-pk between fD/500 & 2*fD)			0.10	UI	
tRFref	Reference Clock Rise / Fall Times		1.0		ns	

Notes on Reference Clock Timing:

1. Master Mode divBy4 (REFSEL[0] = "1" & REFSEL[1] = “1”).
2. The Unit Interval (UI) is the reciprocal of the symbol rate.
3. Total jitter includes both deterministic jitter and random jitter. The random jitter is the total jitter minus the actual deterministic jitter.
4. Values are measured with each PECL/LVDS input DC coupled into a 50 Ohm impedance (100 Ohms differential impedance).
5. Rise time is measured from the 10% threshold of the reference signal to the 90% threshold of the reference signal.
6. Fall time is measured from the 90% threshold of the reference signal to the 10% threshold of the reference signal.
7. Duty cycle and jitter are specified between differential crossings of the 50% threshold of the reference signal.

18.1.4 Data Interface

The data in Table 42 and Table 43 below give output and input specifications for the data and clock lanes.

Table 42 Output Data Timing (RDCLK, RCTL, RDAT)

Symbol	Parameter	Min	Typ	Max	Units	Comments
Dcoc	Output Clock Duty Cycle	45		55	%	
fD	Output Clock Frequency	311		350	MHz	Max is not in the PL4/SPI-4-II standards
TJoc	Output Clock Total Jitter (pk-to-pk between fD/1000 & fD)			0.10	UI	With max allowed jitter on PL4_REFCLK/TDCLK
DJod	Output Data Deterministic Jitter (pk-to-pk between fD/1000 & fD)			0.12	UI	Not in the PL4/SPI-4-II standards With max allowed jitter on PL4_REFCLK/TDCLK
TJod	Output Data Total Jitter (pk-to-pk between fD/1000 & fD)			0.19	UI	PL4/SPI-4-II standards give a max of 0.24UI With max allowed jitter on PL4_REFCLK/TDCLK
tCDSout	Output Clock lane to any Data lane Skew			±280	ps	
tDSout	Output Differential Skew		±20		ps	

Notes on Output Timing:

1. The Unit Interval (UI) is the reciprocal of the symbol rate for both clock and data.
2. Total jitter includes both deterministic jitter and random jitter. The random jitter is the total jitter minus the actual deterministic jitter.
3. Values are measured with each LVDS output DC coupled into a 50 Ohm impedance (100 Ohms differential impedance).
4. Jitter and skew are specified between differential crossings of the 50% threshold of the reference signal.

Table 43 Input Data Timing (TDCLK, TCTL, TDAT)

Symbol	Parameter	Min	Typ	Max	Units	Comments
Dcic	Input Clock Duty Cycle	40		60	%	See note #1
fD	Input Clock Frequency	311		350	MHz	
DJic	Input Clock Deterministic Jitter (pk-to-pk between fD/1000 & 8MHz) (pk-to-pk between fD/1000 & fD)			 0.01 0.10	 UI UI	See note #1 Max not in the PL4/SPI-4-II standards
TJic	Input Clock Total Jitter (pk-to-pk between fD/1000 & fD)			0.17	UI	
DJid	Input Data Deterministic Jitter (pk-to-pk between fD/1000 & fD, and no jitter on PL4_REFCLK/TDCLK)			0.36	UI	See note #2 Not in the PL4/SPI-4-II

TJid	Input Data Total Jitter (pk-to-pk between fD/1000 & fD, and no jitter on PL4_REFCLK/TDCLK)			0.65	UI	standards
tDDSin	Input Data to Data Skew (any pair of data signals)			±1.0	UI	
tDSin	Input Differential Skew		±60		ps	

Notes on Input Timing:

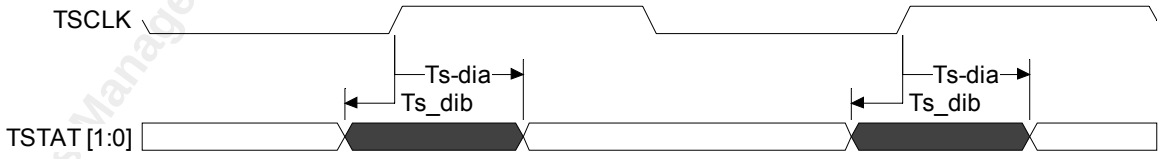
1. Spec's on TDCLK (Dcoc, fD, DJic & TJic) only valid when in Slave Mode (REFSEL[0] = "0" & REFSEL[1] = "0"), as TDCLK is not required when in either Master mode.
2. When in Master Mode and divBy4, the "fD" in DJid & TJid is twice value of the PL4_REFCLK frequency. When in Master Mode and divBy2, the "fD" in DJid & TJid is the same as the PL4_REFCLK frequency.
3. The Unit Interval (UI) is the reciprocal of the symbol rate for both clock and data.
4. Total jitter includes both deterministic jitter and random jitter. The random jitter is the total jitter minus the actual deterministic jitter.
5. Values are measured with each LVDS input DC coupled into a 50 Ohm impedance (100 Ohms differential impedance).
6. Jitter and skew are specified between differential crossings of the 50% threshold of the reference signal.

18.1.5 PL4 FIFO Status Interface

Table 44 Output Status Timing (TSCLK, TSTAT[1:0])

Symbol	Parameter	Min	Typ	Max	Units
PL4_Fos	TSCLK Frequency	1/4		1/4	PL4_Fin[1]
PL4_Dcos	TSCLK Duty Cycle	40		60	%
PL4_Ts_dib	TSTAT[1:0] Invalid Window Before Rising Edge of TSCLK (at PM3388 pins)			1.0	ns
PL4_Ts_dia	TSTAT[1:0] Invalid Window After Rising Edge of TSCLK (at PM3388 pins)			2.5	ns

Figure 33 PL4 Bus Output Status AC Timing Diagram



Notes on PL4 Output Status Timing:

1. The TSCLK frequency is one-quarter of the TDCLK frequency.
2. Assumes a load of 30 pF on the TSCLK and TSTAT[1:0] outputs.
3. Rise time is measured from the 0.8 Volt threshold of the reference signal to the 2.0 Volt threshold of the reference signal.

4. Fall time is measured from the 2.0 Volt threshold of the reference signal to the 0.8 Volt threshold of the reference signal.
5. Duty cycle and skew are specified between crossings of the 1.4 Volt threshold of the reference signal.

Table 45 Input Status Timing (RSCLK, RSTAT[1:0])

Symbol	Parameter	Min	Typ	Max	Units
PL4_Fis	RSCLK Frequency			1/4	PL4_Fout[1]
PL4_DCis	RSCLK Duty Cycle	35		65	%
PL4_tSis	RSTAT[1:0] to Rising Edge RSCLK Setup (at PM3388 pins)	2.0			ns
PL4_tHis	RSTAT[1:0] to Rising Edge RSCLK Hold (at PM3388 pins)	0.5			ns

Figure 34 PL4 Bus Input Status AC Timing Diagram

Notes on PL4 Input StatusTiming:

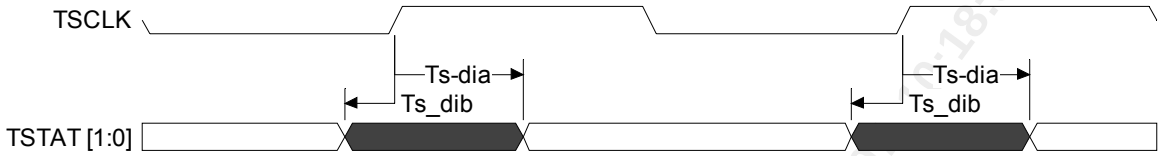
1. The maximum RSCLK frequency shall not exceed one-quarter of the RDCLK frequency.
2. Rise time is measured from the 0.8 Volt threshold of the reference signal to the 2.0 Volt threshold of the reference signal.
3. Fall time is measured from the 2.0 Volt threshold of the reference signal to the 0.8 Volt threshold of the reference signal.
4. Duty cycle, setup and hold are specified between crossings of the 1.4 Volt threshold of the reference signal.

18.1.6 PL4 FIFO Status Interface

Table 46 Output Status Timing (TSCLK, TSTAT[1:0])

Symbol	Parameter	Min	Typ	Max	Units
PL4_Fos	TSCLK Frequency	1/4		1/4	See Note #1
PL4_Dcos	TSCLK Duty Cycle	40		60	%
PL4_Ts_dib	TSTAT[1:0] Invalid Window Before Rising Edge of TSCLK (at PM3392 pins)			1.0	ns
PL4_Ts_dia	TSTAT[1:0] Invalid Window After Rising Edge of TSCLK (at PM3392 pins)			2.5	ns

Figure 35 PL4 Bus Output Status AC Timing Diagram



Notes on PL4 Output Status Timing:

1. The TSCLK frequency is one-quarter of the TDCLK frequency.
2. Assumes a load of 30 pF on the TSCLK and TSTAT[1:0] outputs.
3. Rise time is measured from the 0.8 Volt threshold of the reference signal to the 2.0 Volt threshold of the reference signal.
4. Fall time is measured from the 2.0 Volt threshold of the reference signal to the 0.8 Volt threshold of the reference signal.
5. Duty cycle and skew are specified between crossings of the 1.4 Volt threshold of the reference signal.

Table 47 Input Status Timing (RSCLK, RSTAT[1:0])

Symbol	Parameter	Min	Typ	Max	Units
PL4_Fis	RSCLK Frequency			1/4	See Note #1
PL4_DCis	RSCLK Duty Cycle	35		65	%
PL4_tSis	RSTAT[1:0] to Rising Edge RSCLK Setup (at PM3392 pins)	2.0			ns
PL4_tHis	RSTAT[1:0] to Rising Edge RSCLK Hold (at PM3392 pins)	0.5			ns

Figure 36 PL4 Bus Input Status AC Timing Diagram

Notes on PL4 Input StatusTiming:

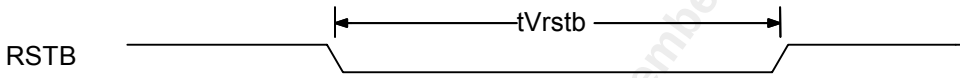
1. The maximum RSCLK frequency shall not exceed one-quarter of the RDCLK frequency.
2. Rise time is measured from the 0.8 Volt threshold of the reference signal to the 2.0 Volt threshold of the reference signal.
3. Fall time is measured from the 2.0 Volt threshold of the reference signal to the 0.8 Volt threshold of the reference signal.
4. Duty cycle, setup and hold are specified between crossings of the 1.4 Volt threshold of the reference signal.

18.2 System Miscellaneous Timing

Table 48 System Miscellaneous Timing

Symbol	Description	Min	Max	Units
tVRSTB	RSTB input pulse width	100		ns

Figure 37 System Miscellaneous Timing Diagram

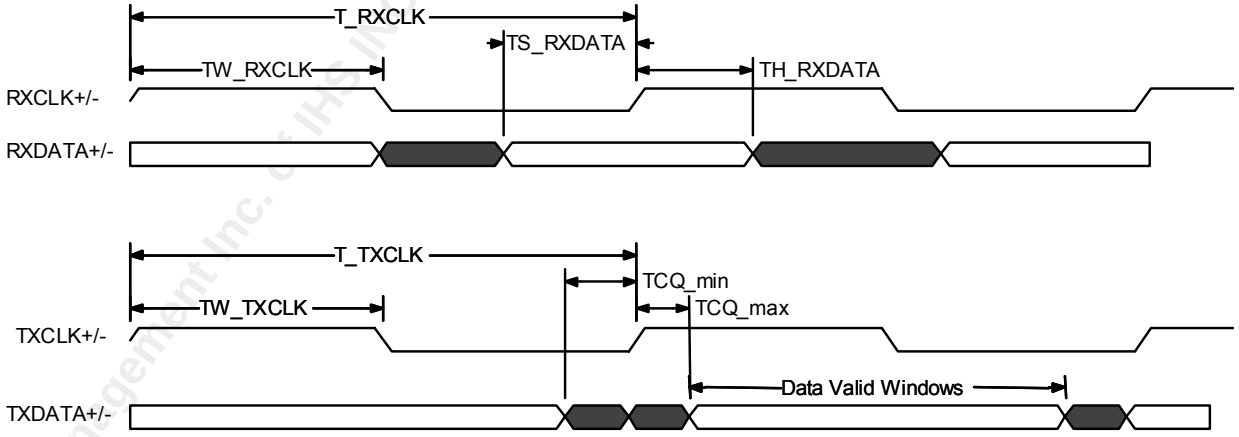


18.3 XSBI Interface Timing

Table 49 XSBI Interface Timing

Symbol	Description	Min	Typ	Max	Units
fRXCLK	RXCLK Frequency (nominally 644.53125MHz)	643.5		645	MHz
TRXCLK	RXCLK period (nominally 1.5515 ns)	1.550		1.56	ns
TWRXCLK	RXCLK duty cycle	45		55	%
TRRXCLK	RXCLK rise time (20%-80%)	100	200	300	ps
TFRXCLK	RXCLK fall time (20%-80%)	100	200	300	ps
TSRXCLK	RXDATA Setup time	300			ps
THRXCLK	RXDATA hold time	300			ps
fTXCLK	TXCLK Frequency (nominally 644.53125MHz)	643.5		645	MHz
TTXCLK	TXCLK period (nominally 1.55151 ns)	1.550		1.56	ns
TWTXCLK	TXCLK duty cycle	40		60	%
TRTXCLK	TXCLK rise time (20%-80%)	100	175	250	ps
TFTXCLK	TXCLK fall time (20%-80%)	100	175	250	ps
TCQTXCLK	TXDATA propagation delay is measured into the recommended termination network	-200		200	ps

Figure 38 Line Interface Timing

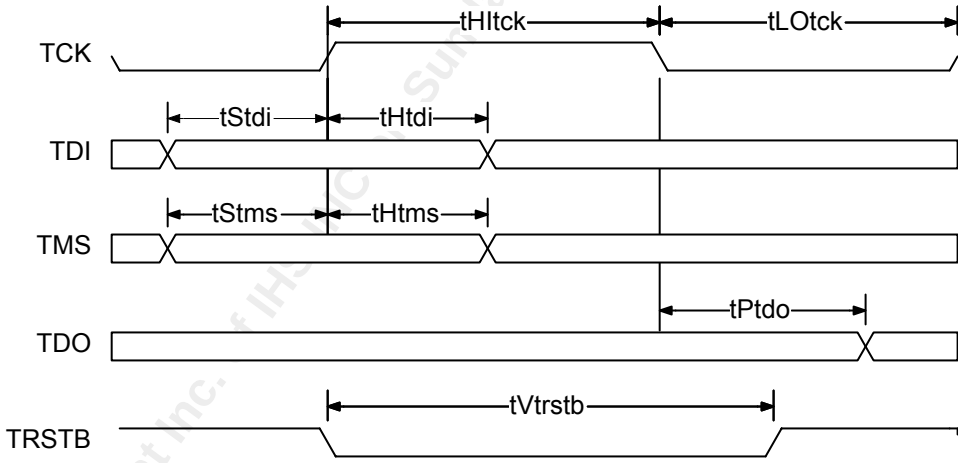


18.4 JTAG Port Timing

Table 50 JTAG Port Interface

Symbol	Description	Min	Max	Units
fTCK	TCK Frequency		4	MHz
tHITCK	TCK HI Pulse Width	100		ns
tLOTCK	TCK LO Pulse Width	100		ns
tSTMS	TMS Set-up time to TCK	25		ns
tHTMS	TMS Hold time to TCK	25		ns
tSTDI	TDI Set-up time to TCK	25		ns
tHTDI	TDI Hold time to TCK	25		ns
tPTDO	TCK Low to TDO Valid	2	25	ns
tVTRSTB	TRSTB Pulse Width	100		ns

Figure 39 JTAG Port Interface Timing



Notes on Input Timing:

1. When a set-up time is specified between an input and a clock, the set-up time is the time in nanoseconds from the 1.4 Volt point of the input to the 1.4 Volt point of the clock.
2. When a hold time is specified between an input and a clock, the hold time is the time in nanoseconds from the 1.4 Volt point of the clock to the 1.4 Volt point of the input.

Notes on Output Timing:

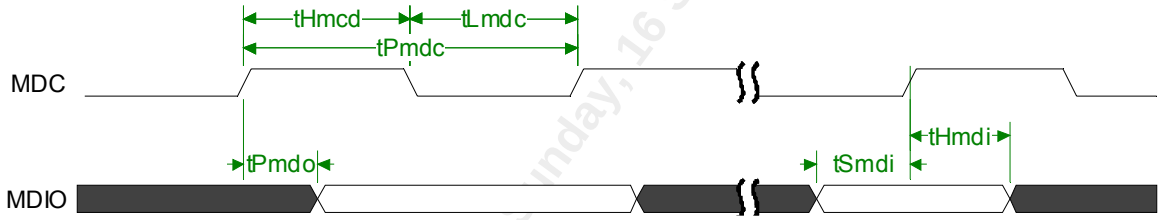
1. Output propagation delay time is the time in nanoseconds from the 1.4 Volt point of the reference signal to the 1.4 Volt point of the output.
2. Maximum and minimum output propagation delays are measured with a 30 pF load on the outputs except when otherwise specified.

18.5 MDIO/MDC Timing

Table 51 MDC / MDIO Interface Timing

Symbol	Description	Min	Typ	Max	Units
tPmcd	MDC Period		2.4		MHz
tHmcd	Time High MDC		211		ns
tLmcd	Time Low MDC		211		ns
tPmdo	MDC High to Valid MDIO Data		25		ns
tSmdi	MDIO Setup Time to MDC		100		ns
tHmdi	MDIO Hold Time to MDC		0		ns

Figure 40 MDC / MDIO Physical Timing



Notes on MDC/MDIO I/O Timing:

1. When a set-up time is specified between an input and a clock, the set-up time is the time in nanoseconds from the 1.4 Volt point of the input to the 1.4 Volt point of the clock.
2. When a hold time is specified between an input and a clock, the hold time is the time in nanoseconds from the 1.4 Volt point of the clock to the 1.4 Volt point of the input.
3. Output propagation delay time is the time in nanoseconds from the 1.4 Volt point of the reference signal to the 1.4 Volt point of the output.

19 Thermal Information

This product is designed to operate over a wide temperature range when used with a heat sink and is suited for outside plant equipment[1].

Table 52 Outside Plant Thermal Information

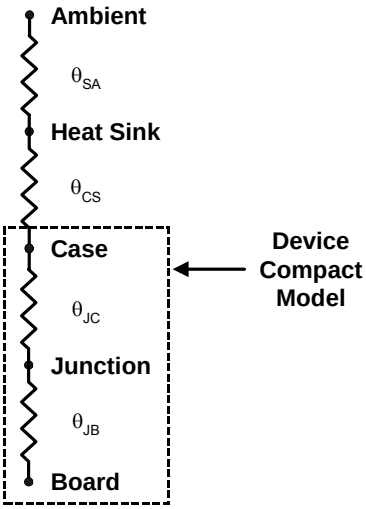
Maximum long-term operating junction temperature (T_J) to ensure adequate long-term life.	105 °C
Maximum junction temperature (T_J) for short-term excursions with guaranteed continued functional performance[2]. This condition will typically be reached when the local ambient temperature reaches 85 °C.	125 °C
Minimum ambient temperature (T_A)	-40 °C

Table 53 Device Compact Model[3]

Junction-to-Case Thermal Resistance, θ_{JC}	0.39 °C/W
Junction-to-Board Thermal Resistance, θ_{JB}	4.8 °C/W

Table 54 Heat Sink Requirements

$\theta_{SA}+\theta_{CS}$[4]	The sum of θ_{SA} + θ_{CS} must be less than or equal to: $[(105 - T_A) / P_D] - \theta_{JC}$] °C/W where: T_A is the ambient temperature at the heat sink location P_D is the operating power dissipated in the package
	θ_{SA} and θ_{CS} are required for long-term operation



Power depends upon the operating mode. To obtain power information, refer to 'High' power values in section Section 18.1 Power Requirements.

Notes

1. The minimum ambient temperature requirement for Outside Plant Equipment meets the minimum ambient temperature requirement for Industrial Equipment
2. Short-term is used as defined in Telcordia Technologies Generic Requirements GR-63-Core Core
3. θ_{JC}, the junction-to-case thermal resistance, is a measured nominal value plus two sigma. θ_{JB}, the junction-to-board thermal resistance, is obtained by simulating conditions described in JEDEC Standard JESD 51-8; for more information about this standard,
4. θ_{SA} is the thermal resistance of the heat sink to ambient. θ_{CS} is the thermal resistance of the heat sink attached material. The maximum θ_{SA} required for the airspeed at the location of the device in the system with all components in place

19.1 Power Requirements

Table 55 Power Consumption

Mode	Parameter	Typ	High	Max	Units
Master by 2	IDDOP (VDDI)	2.583	3.43	3.544	A
	IDDOP (VDDO)	0.465	0.43	0.467	A
	IDDOP (AVDL)	0.078	0.10	0.104	A
	IDDOP (AVDH)	0.043	0.06	0.066	A
	Total Power	**6.469**	**8.36**	**-**	W
Master by 4	IDDOP (VDDI)	2.583	3.48	3.597	A
	IDDOP (VDDO)	0.465	0.43	0.467	A
	IDDOP (AVDL)	0.078	0.10	0.105	A
	IDDOP (AVDH)	0.043	0.06	0.067	A
	Total Power	**6.469**	**8.47**	**-**	W
Slave-Immediate	IDDOP (VDDI)	2.583	3.43	3.544	A
	IDDOP (VDDO)	0.465	0.43	0.467	A
	IDDOP (AVDL)	0.078	0.10	0.104	A
	IDDOP (AVDH)	0.043	0.06	0.066	A
	Total Power	**6.469**	**8.36**	**-**	W
Slave-Deferred	IDDOP (VDDI)	2.583	3.43	3.544	A
	IDDOP (VDDO)	0.465	0.43	0.467	A
	IDDOP (AVDL)	0.078	0.10	0.104	A
	IDDOP (AVDH)	0.043	0.06	0.066	A
	Total Power	**6.469**	**8.36**	**-**	W

Notes:

1. Typical IDD values are calculated as the mean value of current under the following conditions: typically processed silicon, nominal supply voltage, Tj=60 °C, outputs loaded with 30 pF, and a normal amount of traffic or signal activity. These values are suitable for evaluating typical device performance in a system.
2. Max IDD values are currents guaranteed by the production test program, product validation results and/or characterization over process for operating currents at the maximum operating voltage and operating temperature that yields the highest current (including outputs loaded to 30pF).
3. Typical power values are calculated using the formula:

   Power = ∑i(VDDNomi x IDDTypi)

   Where i denotes all the various power supplies on the device, VDDNomi is the nominal voltage for supply i, and IDDTypi is the typical current for supply i (as defined in note 1 above). These values are suitable for evaluating typical device performance in a system.

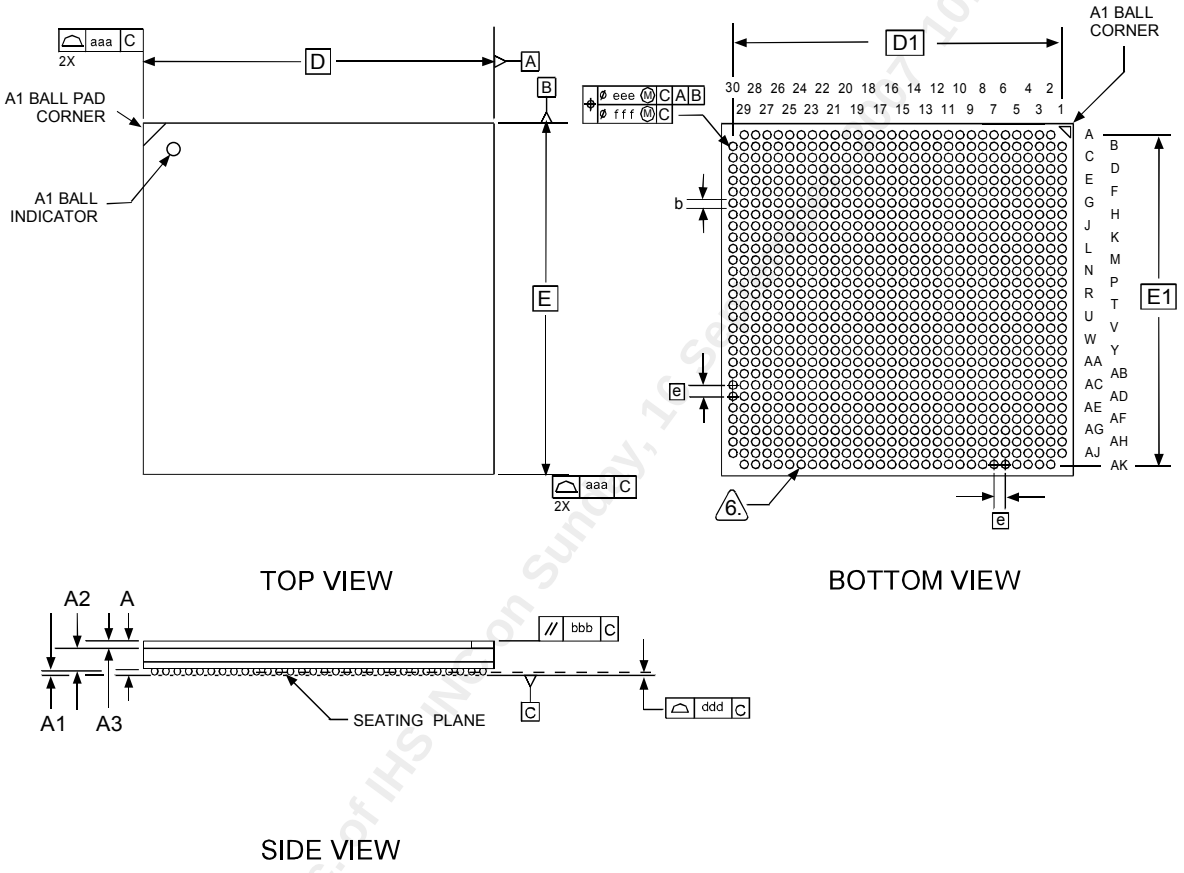
4. High power values are a “normal high power” estimate, calculated using the formula:

   Power = ∑i(VDDMaxi x IDDHighi)

   Where i denotes all the various power supplies on the device, VDDMaxi is the maximum operating voltage for supply i, and IDDHighi is the current for supply i. IDDHigh values are calculated as the mean value plus two sigmas (2σ) of measured current under the following conditions: Tj=105° C, outputs loaded with 30 pF. These values are suitable for evaluating board and device thermal characteristics.

20 Mechanical Information

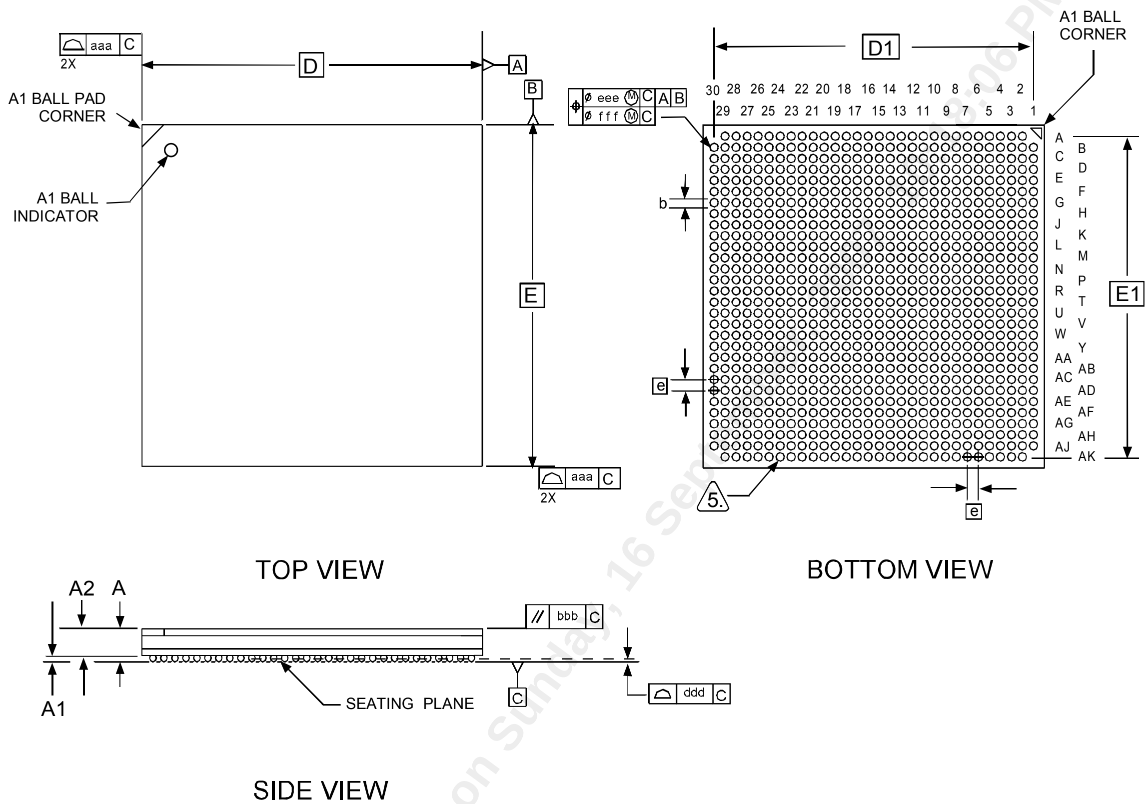
Figure 41 896 PIN FCBGA -31x31 MM BODY - (3M Substrate)



NOTES: 1) ALL DIMENSIONS IN MILLIMETER.
2) DIMENSION aaa DENOTES PACKAGE BODY PROFILE.
3) DIMENSION bbb DENOTES PARALLEL.
4) DIMENSION ccc DENOTES FLATNESS.
5) DIMENSION ddd DENOTES COPLANARITY.
6) DIAMETER OF SOLDER MASK OPENING IS 0.530mm (SMD).

PACKAGE TYPE : 896 FLIP CHIP BALL GRID ARRAY - FCBGA															
BODY SIZE : 31 x 31 x 2.39 MM															
Dim.	**A**	**A1**	**A2**	**A3**	**D**	**D1**	**E**	**E1**	**b**	**e**	**aaa**	**bbb**	**ddd**	**eee**	**f f f**
Min.	2.11	0.40	1.08	0.63	-	-	-	-	0.50	-	-	-	-	-	-
Nom.	2.39	0.50	1.23	0.66	31.00 BSC	29.00 BSC	31.00 BSC	29.00 BSC	0.65	1.00 BSC	-	-	-	-	-
Max.	2.67	0.60	1.38	0.69	-	-	-	-	0.80	-	0.20	0.25	0.20	0.30	0.15

Figure 42 896 PIN FCBGA -31x31 MM BODY - (HDBU Substrate)



NOTES: 1) ALL DIMENSIONS IN MILLIMETER.
2) DIMENSION aaa DENOTES PACKAGE BODY PROFILE.
3) DIMENSION bbb DENOTES PARALLEL.
4) DIMENSION ddd DENOTES COPLANARITY.
5) DIAMETER OF SOLDER MASK OPENING IS 0.530mm (SMD).
6) PACKAGE COMPLIANT TO JEDEC REGISTERED OUTLINE MS-034, VARIATION AAN-1 .

PACKAGE TYPE : 896 FLIP CHIP BALL GRID ARRAY - FCBGA (HDBU 3_2_3)														
BODY SIZE : 31 x 31 x 3.04 MM														
Dim.	A	A1	A2	D	D1	E	E1	b	e	aaa	bbb	ddd	eee	f f f
Min.	2.76	0.40	2.36	-	-	-	-	0.50	-	-	-	-	-	-
Nom.	3.04	0.50	2.54	31.00 BSC	29.00 BSC	31.00 BSC	29.00 BSC	0.64	1.00 BSC	-	-	-	-	-
Max.	3.32	0.60	2.72	-	-	-	-	0.70	-	0.20	0.25	0.20	0.25	0.10

21 Ordering Information

Table 56 Ordering Information

Part No.	Description
PM3392-FI	S/UNI-1x10GE 896-Pin FCBGA (3M Substrate)
PM3392H-FI	S/UNI-1x10GE 896-Pin FCBGA (HDBU Substrate)
PM3392-FGI	S/UNI-1x10GE 896-Pin FCBGA (HDBU Substrate, RoHS-compliant)

Notes