VSC8572-02 Datasheet
Dual-Port 10/100/1000BASE-T PHY with Synchronous Ethernet, IEEE 1588, and QSGMII/SGMII/RGMII MAC

a Microchip company

Microsemi Headquarters
One Enterprise, Aliso Viejo,
CA 92656 USA
Within the USA: +1 (800) 713-4113
Outside the USA: +1 (949) 380-6100
Sales: +1 (949) 380-6136
Fax: +1 (949) 215-4996
Email: sales.support@microsemi.com
www.microsemi.com

About Microsemi

Microsemi, a wholly owned subsidiary of Microchip Technology Inc. (Nasdaq: MCHP), offers a comprehensive portfolio of semiconductor and system solutions for aerospace & defense, communications, data center and industrial markets. Products include high-performance and radiation-hardened analog mixed-signal integrated circuits, FPGAs, SoCs and ASICs; power management products; timing and synchronization devices and precise time solutions, setting the world's standard for time; voice processing devices; RF solutions; discrete components; enterprise storage and communication solutions, security technologies and scalable anti-tamper products; Ethernet solutions; Power-over-Ethernet ICs and midspans; as well as custom design capabilities and services. Learn more at www.microsemi.com.

Contents

Figures

Tables

1 Revision History

This section describes the changes that were implemented in this document. The changes are listed by revision, starting with the most current publication.

1.1 Revision 4.2

Revision 4.2 was published in April 2019. The following is a summary of the changes in revision 4.2 of this document.

- The Block Diagram figure was updated. For more information, see Figure 4, page 5.
- The IEEE 1588 Architecture figure was updated by removing reference to the pps functionality. For more information, see Figure 27, page 26.
- The IEEE 1588 Device Synchronization section was updated by removing reference to the pps functionality. For more information, see IEEE 1588 Device Synchronization, page 43.
- The Timestamp Update section was updated by removing reference to the pps functionality. For more information, see Timestamp Update, page 43.
- The Timestamp Update section was updated by removing reference to the pps functionality. For more information, see Timestamp Update, page 43.
- The Local Time Counter section was updated by removing reference to the PPS0 functionality and the Local Time Counter Load/Save Timing figure. For more information, see Local Time Counter, page 70.
- The 1588_PPS_0/1 Mux Control section was deleted.
- The Register Bits for GPIO Control and Status table was updated by removing reference to the pps functionality. For more information, see Table 31, page 81.
- The GPIO Control 2, Address 14G (0x0E) table was updated by removing reference to the pps functionality. For more information, see Table 103, page 130.
- The GPIO Input, Address 15G (0x0F) table was updated by removing reference to the pps functionality. For more information, see Table 104, page 131.
- The GPIO Output, Address 16G (0x10) table was updated by removing reference to the pps functionality. For more information, see Table 105, page 131.
- The GPIO Input/Output Configuration, Address 17G (0x11) table was updated by removing reference to the pps functionality. For more information, see Table 106, page 132.
- The MISC_CFG table was removed.
- The Registers in IP_1588_LTC table was updated by removing reference to the pps functionality. For more information, see Table 125, page 142.
- The Fields in LTC_CTRL table was updated by removing reference to the pps functionality. For more information, see Table 143, page 147.
- The LTC 1 Pulse per Second Width Adjustment section was removed.
- The IP_1588:MISC_CFG section was removed.
- The diagrams have been updated by removing references to the pps functionality. For more information, see Figure 96, page 315 and Figure 97, page 316.
- The 1588 Support Pins table was updated by removing references to the pps functionality. For more information, see Table 522, page 316.
- The GPIO and 1588 Support Pins table was updated by removing references to the pps functionality. For more information, see Table 523, page 317.
- The GPIO and SIGDET Pins table was updated by removing references to the pps functionality. For more information, see Table 524, page 317.
- The 10BASE-T link recovery failures section was updated. For more information, see 10BASE-T link recovery failures, page 327.

1.2 Revision 4.1

Revision 4.1 was published in May 2018. The following is a summary of the changes in revision 4.1 of this document.

- Configuration procedure steps were clarified. For more information, see Configuration, page 92.
- The description of bit 10 was updated for register 0. For more information, see Table 40, page 96.

- The description of bit 0 was updated for register 22. For more information, see Extended Control and Status, page 105.
- Serial timestamp interface characteristics were updated. For more information, see Table 517, page 311.
- Design considerations were updated. For more information, see Design Considerations, page 327.

1.3 Revision 4.0

Revision 4.0 was published in November 2017. The following is a summary of the changes in revision 4.0 of this document.

- A note was added about enhanced serial LED mode using the V_{DD} LED drive state.
- Details about LED pulsing were updated.
- Information on enabling the serial clock was added.
- Register bits were designated as “sticky” where appropriate.
- A footnote was added about the fast link failure interrupt mask.
- The default for the ring resiliency status bits 4:4 was updated from 00 to 11.
- The default value for the MAC SerDes clause 37 advertised ability register was updated from 0x0000 to 0x01E0.
- Footnotes regarding required register clears were added to the SIGDET/GPIO control register.
- All GPIO input register bits marked as read-only and defaults updated.
- Global interrupt status register defaults were added.
- Register 30G changed from reserved to extended revision ID register.
- Footnotes were added for INGR_BYPASS_ON and EGR_BYPASS_ON 1588 register bits.
- Current consumption values were updated.
- Some parameter names and conditions for recovered clock AC characteristics were updated.
- Product SKUs in the package section were corrected to match the ordering information.
- Design considerations were removed and new ones added to correctly reflect device functionality.

1.4 Revision 2.0

Revision 2.0 of this datasheet was published in September 2017. This was the first publication of the document.

2 Product Overview

The VSC8572-02 is a low-power, dual-port Gigabit Ethernet transceiver with two SerDes interfaces for dual-port dual media capability. It also includes an integrated dual port two-wire serial multiplexer (MUX) to control SFPs or PoE modules. It has a low electromagnetic interference (EMI) line driver, and integrated line side termination resistors that conserve both power and printed circuit board (PCB) space.

The VSC8572-02 includes Microsemi's IEEE 1588 timestamping solution with encapsulation support. The device also includes dual recovered clock outputs to support Synchronous Ethernet applications. Programmable clock squelch control is included to inhibit undesirable clocks from propagating and to help prevent timing loops. The VSC8572-02 also supports a ring resiliency feature that allows a 1000BASE-T connected PHY port to switch between master and slave timing without having to interrupt the 1000BASE-T link.

Using Microsemi's EcoEthernet v2.0 PHY technology, the VSC8572-02 supports energy efficiency features such as Energy Efficient Ethernet (EEE), ActiPHY link down power savings, and PerfectReach that can adjust power based on the cable length. It also supports fully optimized power consumption in all link speeds.

Microsemi's mixed signal and digital signal processing (DSP) architecture is a key operational feature of the VSC8572-02, assuring robust performance even under less-than-favorable environmental conditions. It supports both half-duplex and full-duplex 10BASE-T, 100BASE-TX, and 1000BASE-T communication speeds over Category 5 (Cat5) unshielded twisted pair (UTP) cable at distances greater than 100 m, displaying excellent tolerance to NEXT, FEXT, echo, and other types of ambient environmental and system electronic noise. The device also supports two dual media ports that can support up to two 100BASE-FX, 1000BASE-X fiber, and/or triple-speed copper SFPs.

The following illustrations show a high-level, general view of typical VSC8572-02 applications.

Figure 1 • **Dual Media Application Diagram**

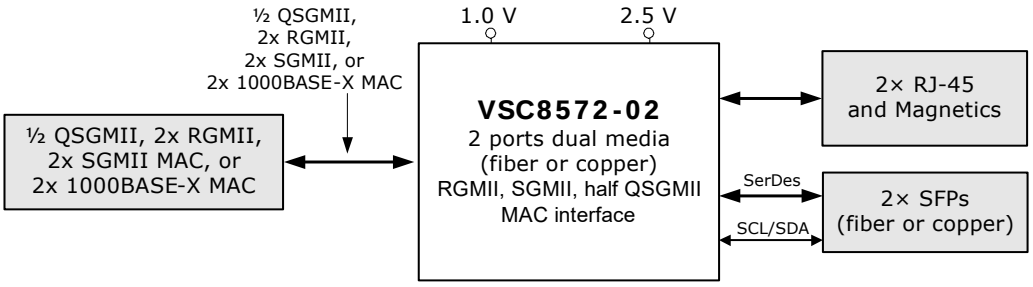


Figure 2 • **Copper Transceiver Application Diagram**

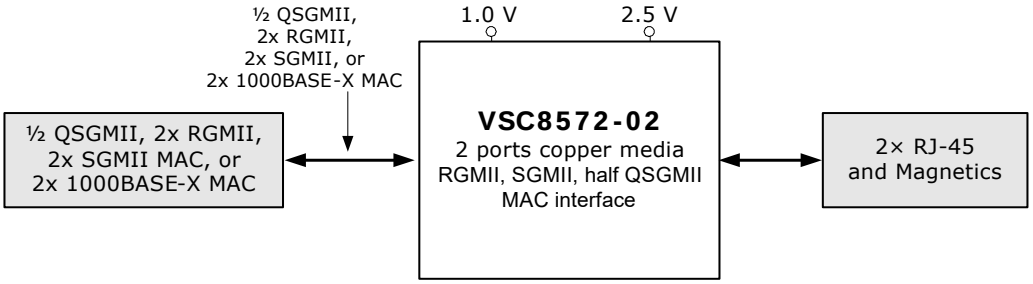
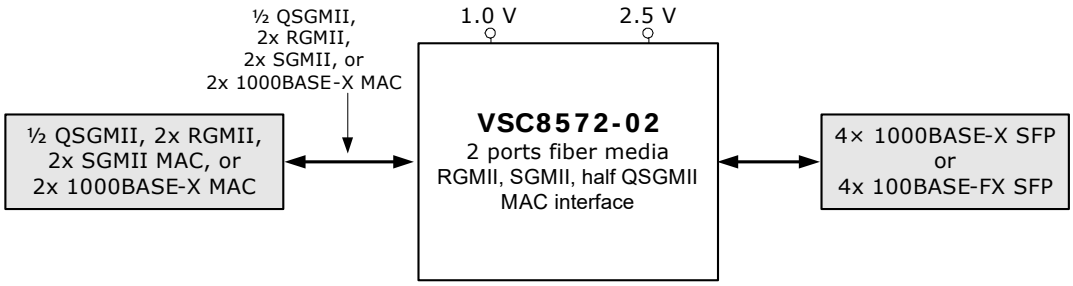


Figure 3 • **Fiber Media Transceiver Application Diagram**



2.1 Key Features

This section lists the main features and benefits of the VSC8572-02 device.

2.1.1 Low Power

- Low power consumption of approximately 425 mW per port in 1000BASE-T mode, 200 mW per port in 100BASE-TX mode, 225 mW per port in 10BASE-T mode, and less than 115 mW per port in 100BASE-FX and 1000BASE-X modes
- ActiPHY™ link down power savings
- PerfectReach™ smart cable reach algorithm
- IEEE 802.3az-2010 Energy Efficient Ethernet idle power savings

2.1.2 Advanced Carrier Ethernet Support

- Support for IEEE 1588-2008 timestamping with encapsulation support
- Recovered clock outputs with programmable clock squelch control and fast link failure indication (<1 ms; worst-case <3 ms) for G.8261 Synchronous Ethernet applications
- Ring resiliency for maintaining linkup integrity when switching between 1000BASE-T master and slave timing
- Supports IEEE 802.3bf timing and synchronization standard
- Integrated dual two-wire serial mux to control SFP and PoE modules
- Support for IEEE 802.3ah unidirectional transport for 100BASE-FX and 1000BASE-X fiber media

2.1.3 Wide Range of Support

- Compliant with IEEE 802.3 (10BASE-T, 100BASE-TX, 1000BASE-T, 100BASE-FX, and 1000BASE-X) specifications
- Support for >16 kB jumbo frames in all speeds with programmable synchronization FIFOs
- Supports Cisco QSGMII v1.3, Cisco SGMII v1.9, RGMII versions 1.3 and 2.0 (2.5 V), 1000BASE-X MACs, IEEE 1149.1 JTAG boundary scan, and IEEE 1149.6 AC-JTAG
- Available in a low-cost, 256-pin BGA package with a 17 mm × 17 mm body size

2.1.4 Flexibility

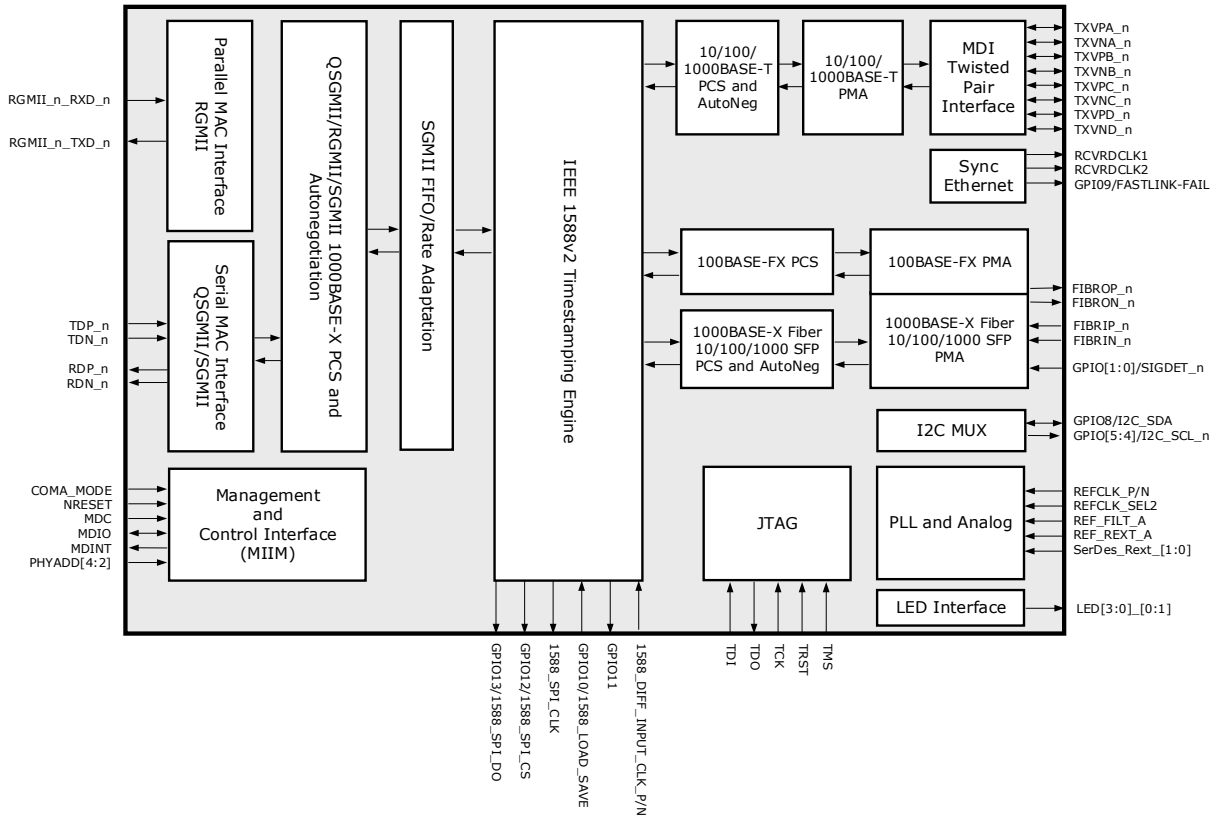
- VeriPHY® cable diagnostics suite provides extensive network cable information such as cable length, termination status, and open/short fault location
- Patented, low EMI line driver with integrated line side termination resistors
- Four programmable direct-drive LEDs per port with adjustable brightness levels using register controls; bi-color LED support using two LED pins
- Serial LED interface option
- Extensive test features including near end, far end, copper media connector, SerDes MAC/media loopback, and Ethernet packet generator with CRC error counter to decrease time-to-market

Note: All MAC interfaces must be the same — all QSGMII, RGMII, or SGMII.

2.2 Block Diagram

The following illustration shows the primary functional blocks of the VSC8572-02 device.

Figure 4 • **Block Diagram**



Note: All MAC interfaces must be the same—all QSGMII, RGMII, SGMII, or 1000BASE-X.

3 Functional Descriptions

This section describes the functional aspects of the VSC8572-02 device, including available configurations, operational features, and testing functionality. It also defines the device setup parameters that configure the device for a particular application.

3.1 Operating Modes

The following table lists the operating modes of the VSC8572-02 device.

Table 1 • **Operating Modes**

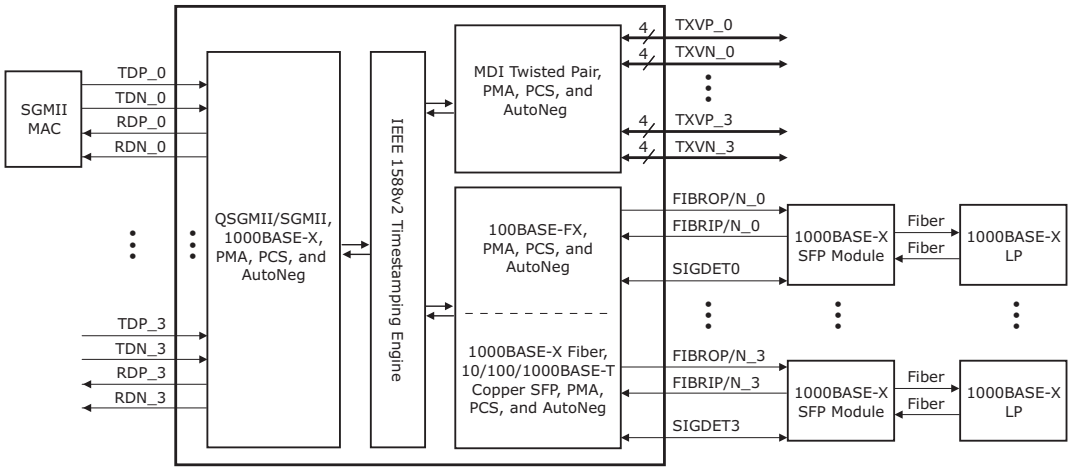
Operating Mode	Supported Media	Notes
QSGMII/RGMII/SGMII MAC-to-1000BASE-X Link Partner	1000BASE-X	See Figure 5, page 7.
QSGMII/RGMII/SGMII MAC-to-100BASE-FX Link Partner	100BASE-FX	See Figure 7, page 8.
QSGMII/RGMII/SGMII MAC-to-AMS and 1000BASE-X SerDes	1000BASE-X, 10/100/1000BASE-T	See Figure 8, page 9.
QSGMII/RGMII/SGMII MAC-to-AMS and 100BASE-FX SerDes	100BASE-FX, 10/100/1000BASE-T	See Figure 9, page 10.
QSGMII/RGMII/SGMII MAC-to-AMS and Protocol Transfer mode	SFP/Fiber Protocol Transfer mode (10/100/1000BASE-T Cu SFP), 10/100/1000BASE-T	See Figure 10, page 11.
QSGMII/RGMII/SGMII MAC-to-Cat5 Link Partner	10/100/1000BASE-T	See Figure 11, page 12.
QSGMII/RGMII/SGMII MAC-to-Protocol Transfer mode	SFP/Fiber Protocol Transfer mode (10/100/1000BASE-T Cu SFP)	See Figure 12, page 12.
1000BASE-X MAC-to-Cat5 Link Partner	1000BASE-T only	See Figure 13, page 13.

Note: All MAC interfaces must be the same — all QSGMII, RGMII, or SGMII.

3.1.1 QSGMII/SGMII MAC-to-1000BASE-X Link Partner

The following illustrations and sections show the register settings used to configure a QSGMII/SGMII MAC-to-1000BASE-X link partner.

Figure 5 • **SGMII MAC-to-1000BASE-X Link Partner**

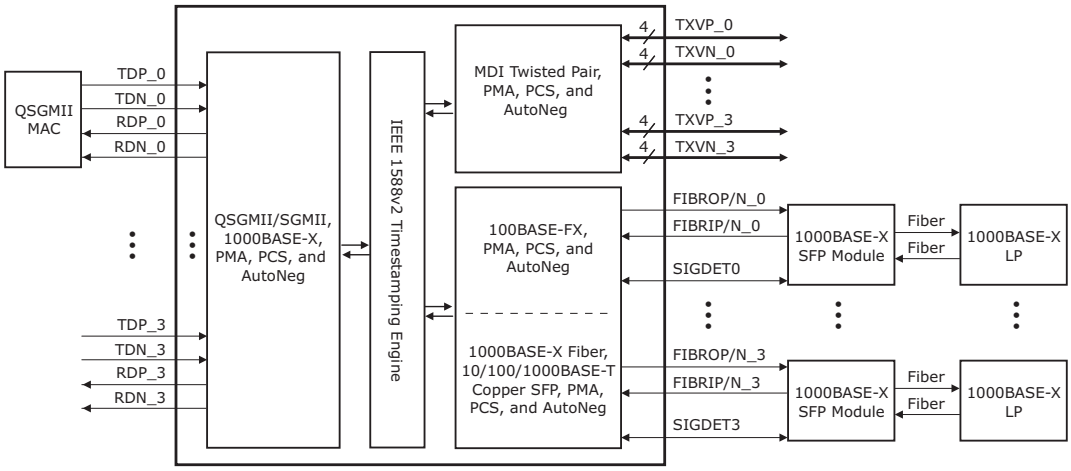


3.1.1.1 MAC interface SGMII

Use the following settings to configure the SGMII MAC interface.

- Set register 19G bits 15:14 = 00.
- Set register 23 (main register) bit 12 = 0.
- Set register 18G = 0x80F0. For more information, see Table 86, page 123.

Figure 6 • **QSGMII MAC-to-1000BASE-X Link Partner**



3.1.1.2 MAC interface QSGMII

Use the following settings to configure the QSGMII MAC interface.

- Set register 19G bits 15:14 = 01.
- Set register 23 (main register) bit 12 = 0.
- Set register 18G = 0x80F0. For more information, see Table 86, page 123.

3.1.1.3 Media interface 1000BASE-X SFP Fiber (1000BASE-X Link Partner)

Use the following settings to configure the 1000BASE-X SFP fiber media interface.

- Set register 23 bits 10:8 = 010.
- Set register 0 bit 12 = 1 (enable autonegotiation).
- Set register 18G = 0x8FC1. For more information, see Table 86, page 123.

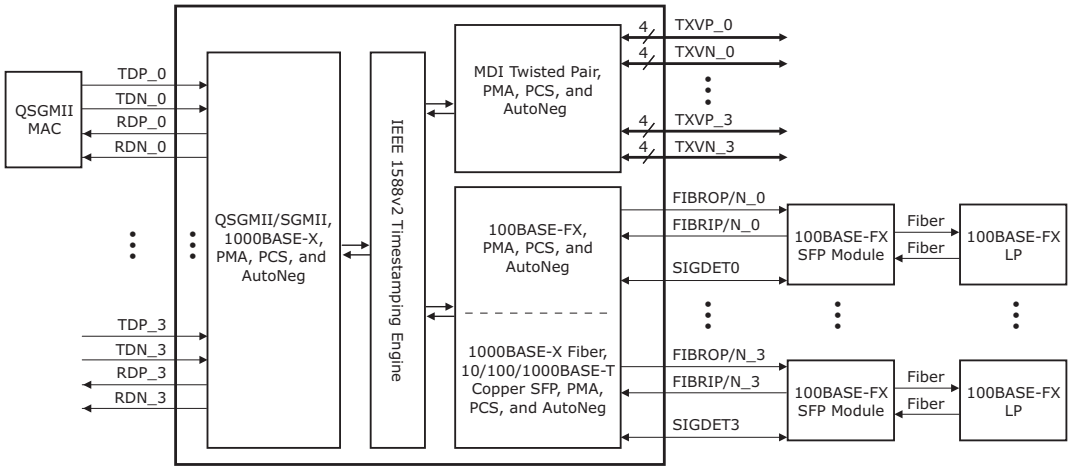
The F in 0x8FC1 identifies the port. To exclude a port from the configuration, set its bit to 0. For example, the configuration of port 0 and port 1 to 1000BASE-X is 0011 or 3, making the bit setting 0x83C1.

Note: Whenever there is a mode change a software reset (register 0 bit 15) is required to make the mode change effective. This register is cleared when read.

3.1.2 QSGMII/SGMII MAC-to-100BASE-FX Link Partner

The following illustration and sections show the register settings used to configure a QSGMII/SGMII MAC-to-100BASE-FX link partner.

Figure 7 • **QSGMII/SGMII MAC-to-100BASE-FX Link Partner**



3.1.2.1 MAC interface SGMII

Use the following settings to configure the SGMII MAC interface.

- Set register 19G bits 15:14 = 00.
- Set register 23 (main register) bit 12 = 0.
- Set register 18G = 0x80F0. For more information, see Table 86, page 123.

3.1.2.2 Media interface 100BASE-FX SFP Fiber (100BASE-FX Link Partner)

Use the following settings to configure the 100BASE-FX SFP fiber media interface.

- Set register 23 bits 10:8 = 011.
- Set register 0 bit 12 = 0 (autonegotiation not present in 100BASE-FX PHY).
- Set register 18G = 0x8FD1. For more information, see Table 86, page 123.

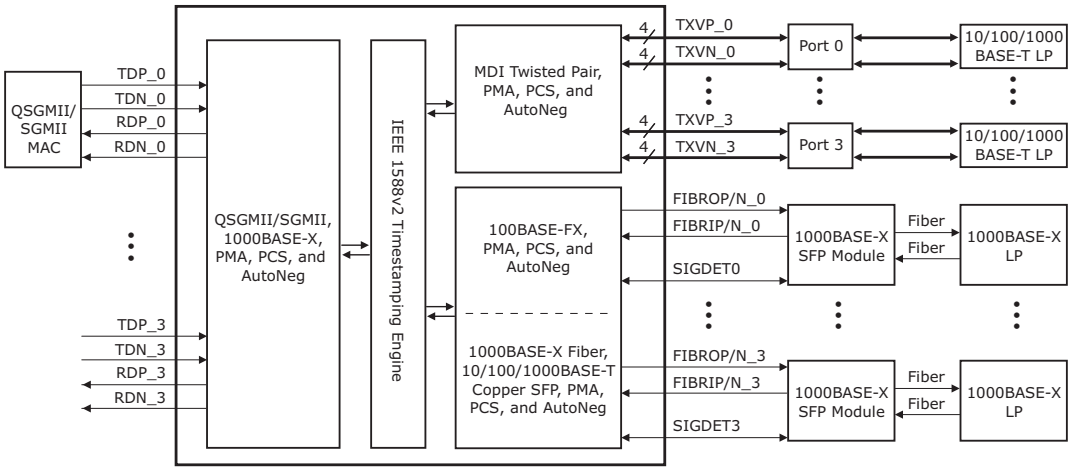
For QSGMII only port 0 is used.

Note: Whenever there is a mode change a software reset (register 0 bit 15) is required to make the mode change effective. This register is cleared when read.

3.1.3 QSGMII/SGMII MAC-to-AMS and 1000BASE-X Media SerDes

The following illustration and sections show the register settings used to configure a QSGMII/SGMII MAC-to-AMS and 1000BASE-X media SerDes.

Figure 8 • **QSGMII/SGMII MAC-to-AMS and 1000BASE-X Media SerDes**



3.1.3.1 MAC interface SGMII

Use the following settings to configure the SGMII MAC interface.

- Set register 19G bits 15:14 = 00.
- Set register 23 (main register) bit 12 = 0.
- Set register 18G = 0x80F0. For more information, see Table 86, page 123.

3.1.3.2 Media interface 1000BASE-X SFP Fiber (1000BASE-X Link Partner)

Use the following settings to configure the 1000BASE-X SFP fiber media interface.

- Set register 23 bits 10:8 = 010.
- Set register 0 bit 12 = 1 (enable autonegotiation).

3.1.3.3 AMS Preference Setup

Use the following settings for the AMS preferences setup.

- Set register 23 bit 10 = 1 (enable AMS).
- Set register 23 bit 11 to the port preferences.

The media selected by AMS can be read from register 20E1 bits 7:6. For more information, see Table 4, page 21.

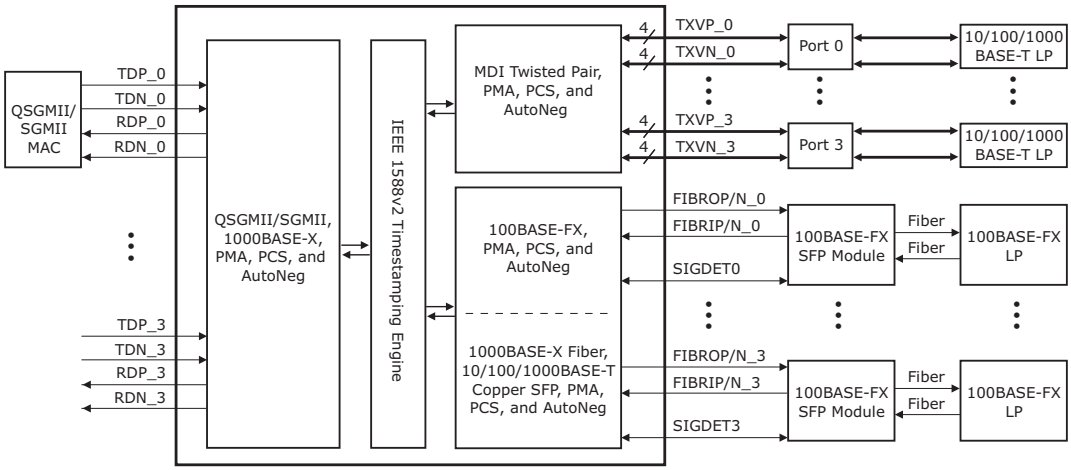
For QSGMII only port 0 is used.

Note: Whenever there is a mode change a software reset (register 0 bit 15) is required to make the mode change effective. This register is cleared when read.

3.1.4 QSGMII/SGMII MAC-to-AMS and 100BASE-FX Media SerDes

The following illustration and sections show the register settings used to configure a QSGMII/SGMII MAC-to-AMS and 100BASE-FX media SerDes.

Figure 9 • **QSGMII/SGMII MAC-to-AMS and 100BASE-FX Media SerDes**



3.1.4.1 MAC interface SGMII

Use the following settings to configure the SGMII MAC interface.

- Set register 19G bits 15:14 = 00.
- Set register 23 (main register) bit 12 = 0.
- Set register 18G = 0x80F0. For more information, see Table 86, page 123.

3.1.4.2 Media interface 100BASE-FX SFP Fiber (100BASE-FX Link Partner)

Use the following settings to configure the 100BASE-FX SFP fiber media interface.

- Set register 23 bits 10:8 = 011.
- Set register 0 bit 12 = 1 (enable autonegotiation).

3.1.4.3 AMS Preference Setup

Use the following settings for the AMS preferences setup.

- Set register 23 bit 10 = 1 (enable AMS).
- Set register 23 bit 11 to the port preferences.

The media selected by AMS can be read from register 20E1 bits 7;6. For more information, see Table 4, page 21.

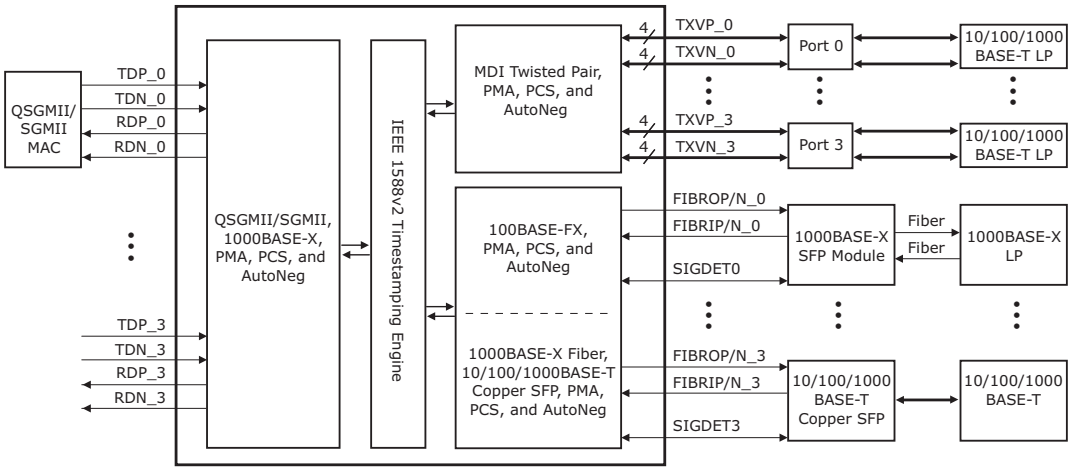
For QSGMII only port 0 is used.

Note: Whenever there is a mode change a software reset (register 0 bit 15) is required to make the mode change effective. This register is cleared when read.

3.1.5 QSGMII/SGMII MAC-to-AMS and Protocol Transfer Mode

The following illustration and sections show the register settings used to configure a QSGMII/SGMII MAC-to-AMS and Protocol Transfer mode.

Figure 10 • **QSGMII/SGMII MAC-to-AMS and Protocol Transfer Mode**



3.1.5.1 MAC interface SGMII

Use the following settings to configure the SGMII MAC interface.

- Set register 19G bits 15:14 = 00.
- Set register 23 (main register) bit 12 = 0.
- Set register 18G = 0x80F0. For more information, see Table 86, page 123.

3.1.5.2 Media interface 10/100/1000BASE-T Cu-SFP

Use the following settings to configure the fiber/SFP media interface for protocol transfer mode.

- Set register 23 bits 10:8 = 001.
- Set register 0 bit 12 = 1 (enable autonegotiation).

3.1.5.3 AMS Preference Setup

Use the following settings for the AMS preferences setup.

- Set register 23 bit 10 = 1 (enable AMS).
- Set register 23 bit 11 to the port preferences.

The media selected by AMS can be read from register 20E1 bits 7;6. For more information, see Table 4, page 21.

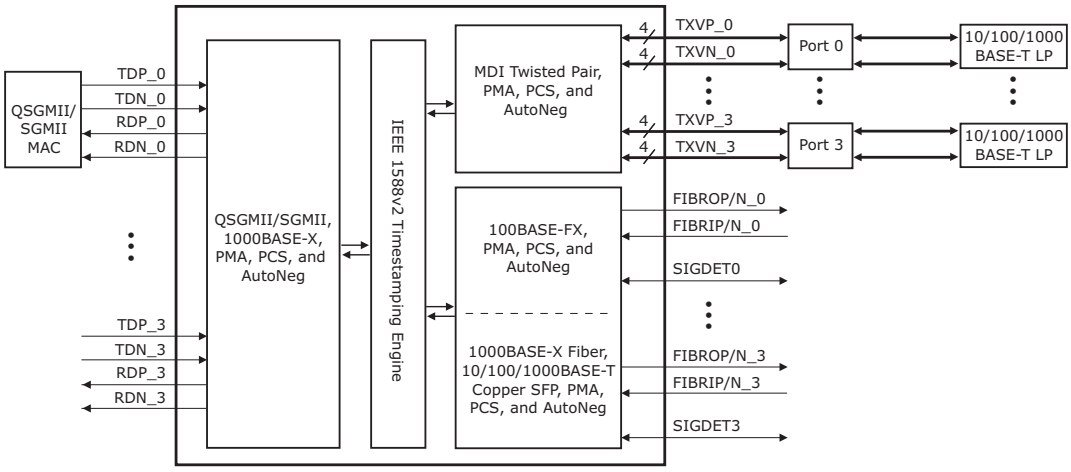
For QSGMII only port 0 is used.

Note: Whenever there is a mode change a software reset (register 0 bit 15) is required to make the mode change effective. This register is cleared when read.

3.1.6 QSGMII/SGMII MAC-to-Cat5 Link Partner

The following illustration and sections show the register settings used to configure a QSGMII/SGMII MAC-to-Cat5 link partner.

Figure 11 • **QSGMII/SGMII MAC-to-Cat5 Link Partner**



3.1.6.1 MAC interface SGMII

Use the following settings to configure the SGMII MAC interface.

- Set register 19G bits 15:14 = 00.
- Set register 23 (main register) bit 12 = 0.
- Set register 18G = 0x80F0. For more information, see Table 86, page 123.

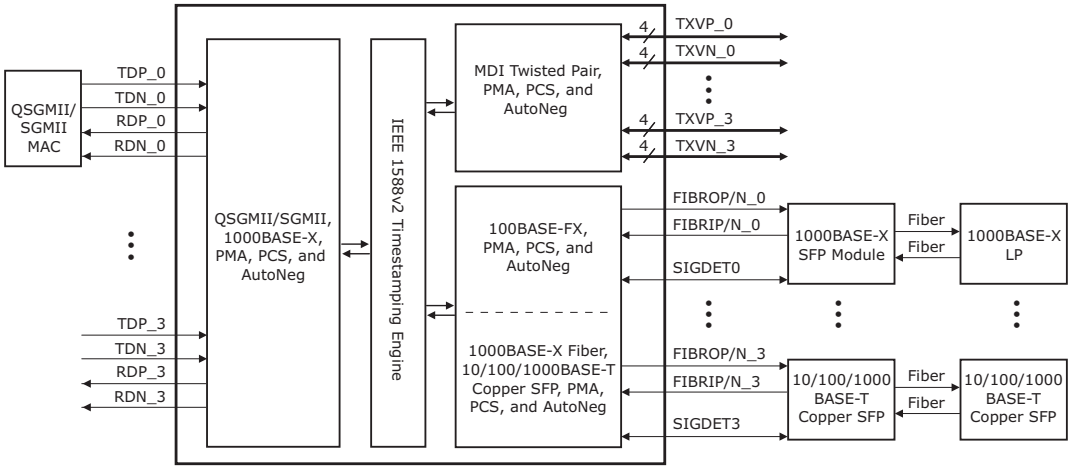
For QSGMII only port 0 is used.

Note: Whenever there is a mode change a software reset (register 0 bit 15) is required to make the mode change effective. This register is cleared when read.

3.1.7 QSGMII/SGMII MAC-to-Protocol Transfer Mode

The following illustration and sections show the register settings used to configure a QSGMII/SGMII MAC-to-Protocol Transfer mode.

Figure 12 • **QSGMII/SGMII MAC-to-Protocol Transfer Mode**



3.1.7.1 MAC interface SGMII

Use the following settings to configure the SGMII MAC interface.

- Set register 19G bits 15:14 = 00.
- Set register 23 (main register) bit 12 = 0.
- Set register 18G = 0x80F0. For more information, see Table 86, page 123.

3.1.7.2 Media interface 10/100/1000BASE-T Cu-SFP

Use the following settings to configure the fiber/SFP media interface for protocol transfer mode.

- Set register 23 bits 10:8 = 001.
- Set register 0 bit 12 = 1 (enable autonegotiation).

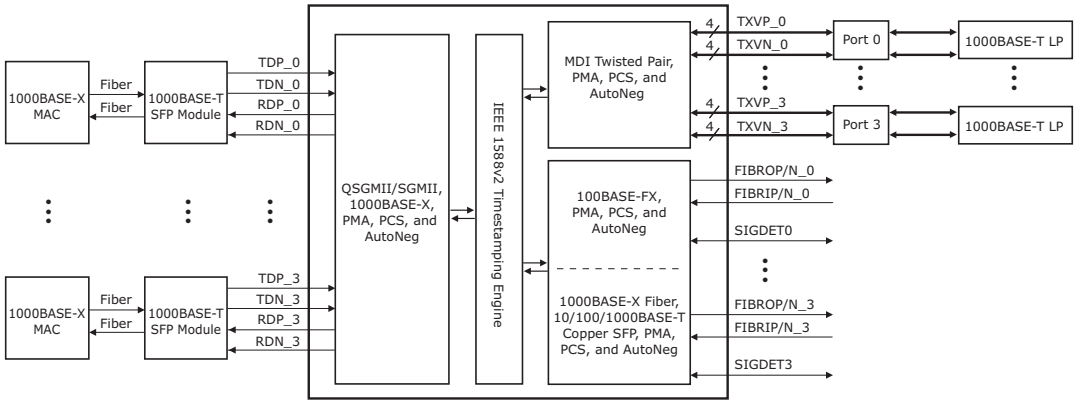
For QSGMII only port 0 is used.

Note: Whenever there is a mode change a software reset (register 0 bit 15) is required to make the mode change effective. This register is cleared when read.

3.1.8 1000BASE-X MAC-to-Cat5 Link Partner

The following illustration and sections show the register settings used to configure a 1000BASE-X MAC-to-Cat5 Link Partner.

Figure 13 • **1000BASE-X MAC-to-Cat5 Link Partner**



In this mode the device provides data throughput of 1000 Mbps only.

3.1.8.1 MAC interface

Use the following settings to configure the MAC interface.

- Set register 19G bits 15:14 = 00.
- Set register 23 (main register) bit 12 = 1.

3.1.8.2 Clause 37 MAC Autonegotiation

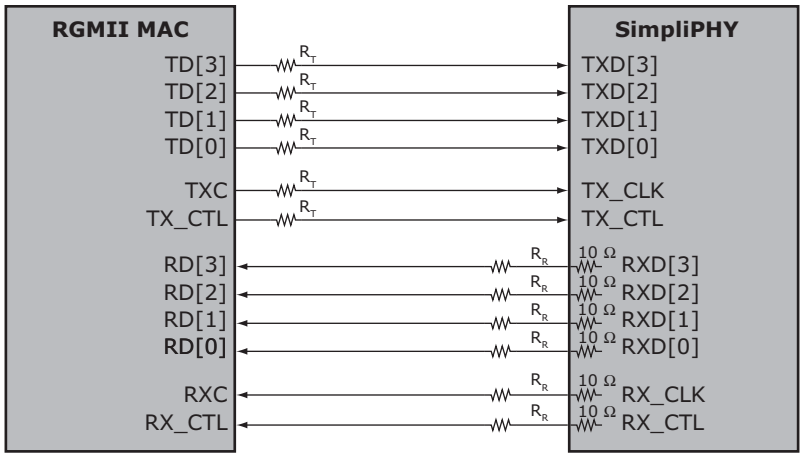
For clause 37 MAC autonegotiation, set register 16E3 bit 7 = 1.

Note: Whenever there is a mode change a software reset (register 0 bit 15) is required to make the mode change effective. This register is cleared when read.

3.2 RGMII MAC

The VSC8572-02 device supports RGMII versions 1.3 and 2.0 (2.5 V). The RGMII interface supports all three speeds (10 Mbps, 100 Mbps, and 1000 Mbps) and is used as an interface to an RGMII-compatible MAC.

Figure 14 • **RGMII MAC Interface**



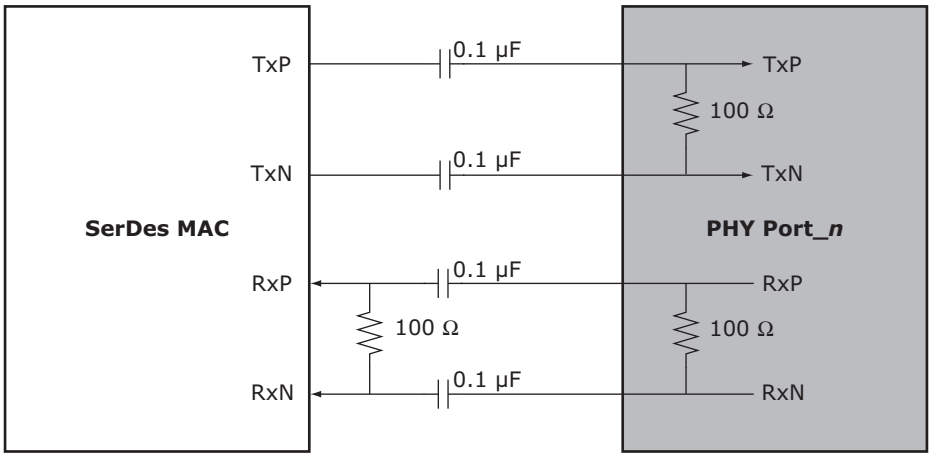
3.3 SerDes MAC Interface

The VSC8572-02 SerDes MAC interface performs data serialization and deserialization functions using an integrated SerDes block. The interface operates in 1000BASE-X compliant mode, QSGMII mode, or SGMII mode. The SerDes and enhanced SerDes block has the termination resistor integrated into the device. The SerDes block also has the AC decoupling capacitors integrated in the receive path. Register 19G is a global register and only needs to be set once to configure the device. The other register bits are configured on a per-port basis and the operation either needs to be repeated for each port, or a broadcast write needs to be used by setting register 22, bit 0 to configure all the ports simultaneously.

3.3.1 SerDes MAC

When connected to a SerDes MAC compliant to 1000BASE-X, the VSC8572-02 device provides data throughput at a rate of 1000 Mbps only; 10 Mbps and 100 Mbps rates are not supported. To configure the device for SerDes MAC mode, set register 19G, bits 15:14 = 0, and register 23, bit 12 = 1. The device also supports 1000BASE-X Clause 37 MAC-side autonegotiation and is enabled through register 16E3, bit 7. To configure the rest of the device for 1000 Mbps operation, select 1000BASE-T only by disabling the 10BASE-T/100BASE-TX advertisements in register 4.

Figure 15 • **SerDes MAC Interface**

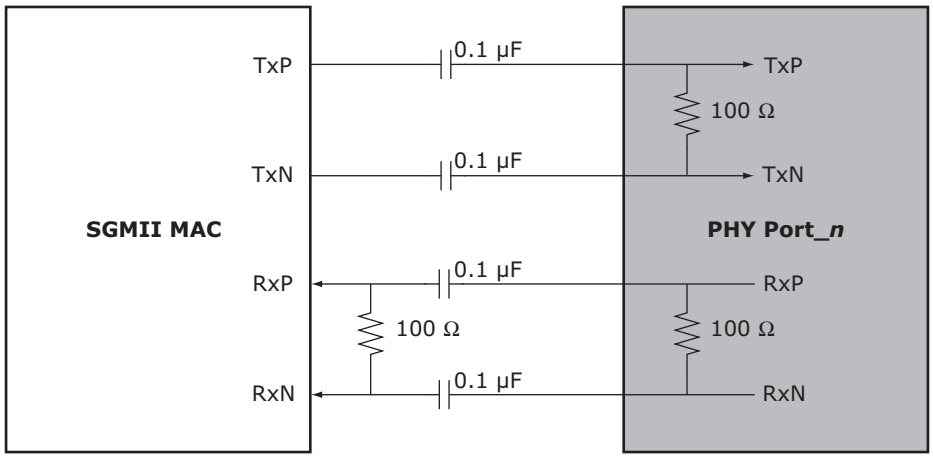


3.3.2 SGMII MAC

When configured to detect and switch between 10BASE-T, 100BASE-T, and 1000BASE-T data rates, the VSC8572-02 device can be connected to an SGMII-compatible MAC. To configure the device for SGMII MAC mode, set register 19G, bits 15:14 = 00 and register 23, bit 12 = 0. In addition, set register 18G as

desired. This device also supports SGMII MAC-side autonegotiation and is enabled through register 16E3, bit 7.

Figure 16 • **SGMII MAC Interface**

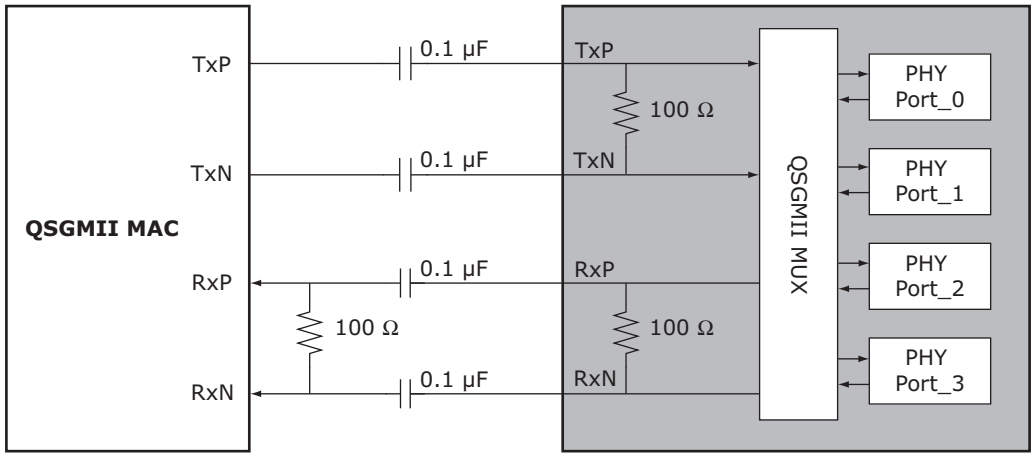


3.3.3 QSGMII MAC

The VSC8572-02 device supports a QSGMII MAC to convey two ports of network data and port speed between 10BASE-T, 100BASE-T, and 1000BASE-T data rates and operates in both half-duplex and full-duplex at all port speeds. The MAC interface protocol for each port within QSGMII can be either 1000BASE-X or SGMII, if the QSGMII MAC that the VSC8572-02 is connecting to supports this functionality. To configure the device for QSGMII MAC mode, set register 19G, bits 15:14 = 01. In addition, set register 18G as desired. The device also supports SGMII MAC-side autonegotiation on each individual port and is enabled through register 16E3, bit 7, of that port.

Note: Two of the four QSGMII channels contain data. The windows for the other two channels remain present and need to be supported in both directions by the MAC. This support is called “half QSGMII.”

Figure 17 • **QSGMII MAC Interface**



3.4 SerDes Media Interface

The VSC8572-02 device SerDes media interface performs data serialization and deserialization functions using an integrated SerDes block in the SerDes media interface. The interface operates at 1.25 Gbps speed, providing full-duplex and half-duplex for 10/100/1000 Mbps bandwidth that can connect directly to 100BASE-FX/1000BASE-X-compliant optical devices as well as to 10/100/1000BASE-T copper SFP devices. The interface also provides support for unidirectional transport as defined in IEEE 802.3-2008, Clause 66. The SerDes interface has the following operating modes:

- QSGMII/RGMII/SGMII to 1000BASE-X
- QSGMII/RGMII/SGMII to 100BASE-FX
- QSGMII/RGMII/SGMII to SGMII/1000BASE-X protocol transfer

The SerDes media block has the termination resistor integrated into the device. It also has the AC decoupling capacitors integrated in the receive path.

A software reset through register 0, bit 15 is required when changing operating modes between 100BASE-FX and 1000BASE-X.

3.4.1 QSGMII/RGMII/SGMII to 1000BASE-X

The 1000BASE-X SerDes media in QSGMII/RGMII/SGMII mode supports IEEE 802.3 Clause 36 and Clause 37, which describe 1000BASE-X fiber autonegotiation. In this mode, control and status of the SerDes media is displayed in the VSC8572-02 device registers 0 through 15 in a manner similar to what is described in IEEE 802.3 Clause 28. In this mode, connected copper SFPs can only operate at 1000BASE-T speed. A link in this mode is established using autonegotiation (enabled or disabled) between the PHY and the link partner. To configure the PHY in this mode, set register 23, bits 10:8 = 010. To configure 1000BASE-X autonegotiation for this mode, set register 0, bit 12. Setting this mode and configurations can be performed individually on each of the two ports. Ethernet packet generator (EPG), cyclical redundancy check (CRC) counters, and loopback modes are supported in 1000BASE-X mode.

3.4.2 QSGMII/RGMII/SGMII to 100BASE-FX

The VSC8572-02 supports 100BASE-FX communication speed for connecting to fiber modules such as GBICs and SFPs. This capability is facilitated by using the connections on the SerDes pins when connected to a MAC through QSGMII/RGMII/SGMII. Ethernet packet generator (EPG), cyclical redundancy check (CRC) counters, and loopback modes are supported in the 100BASE-FX mode. Setting this mode and configurations can be performed individually on each of the two ports. To configure the PHY in this mode, set register 23, bits 10:8 = 011.

3.4.3 QSGMII to SGMII Protocol Conversion

QSGMII to SGMII (protocol transfer) mode is a feature that links a fiber module or triple speed 10/100/1000-T copper SFP to the QSGMII MAC through the VSC8572-02 device. SGMII can be converted to QSGMII with protocol conversion using this mode.

To configure the PHY in this mode, set register 23, bits 10:8 = 001. To establish the link, assert the relevant signal-detect pin.

All relevant LED modes are supported except for collision, duplex, and autonegotiation fault. The triple-speed copper SFP's link status and data type plugged into the port can be indicated by the PHY's LEDs. Setting this particular mode and configuration can be performed individually on each of the two ports within a QSGMII grouping.

3.4.4 Unidirectional Transport for Fiber Media

The VSC8572-02 device supports IEEE 802.3ah for unidirectional transport across its 1000BASE-X and 100BASE-FX fiber media. This feature enables transmission across fiber media, regardless of whether the PHY has determined that a valid link has been established (register 1, bit 2). The only valid operating modes for unidirectional fiber mode are 100BASE-FX or 1000BASE-X fiber media.

To enable this feature, set register 0, bit 5 to 1. For status of the unidirectional ability, read register 1, bit 7.

Note: Automatic media sensing does not work with this feature. In addition, because unidirectional fiber media must have autonegotiation disabled, RGMII/SGMII autonegotiation must also be disabled (register 16E3, bit 7 = 0).

3.5 PHY Addressing and Port Mapping

This section contains information about PHY addressing and port mapping.

3.5.1 PHY Addressing

The VSC8572-02 includes three external PHY address pins, PHYADD[4:2], to allow control of multiple PHY devices on a system board sharing a common management bus. These pins set the most significant bits of the PHY address port map. The lower two bits of the address for each port are derived from the physical address of the port (0 to 1) and the setting of the PHY address reversal bit in register 20E1, bit 9.

3.5.2 SerDes Port Mapping

The VSC8572-02 includes seven 1.25 GHz SerDes macros and one 5 GHz enhanced SerDes macro. Three of the seven SerDes macros are configured as SGMII MAC interfaces and the remaining four are configured as 1000BASE-X/100BASE-FX SerDes media interfaces. The enhanced SerDes macro can be configured as either a QSGMII MAC interface or the fourth SGMII MAC interface. The following table shows the different operating modes based on the settings of register 19G, bits 15:14.

Table 2 • **MAC Interface Mode Mapping**

19G[15:14]	Operating Mode
00	SGMII
01	QSGMII
10	RGMII
11	Reserved

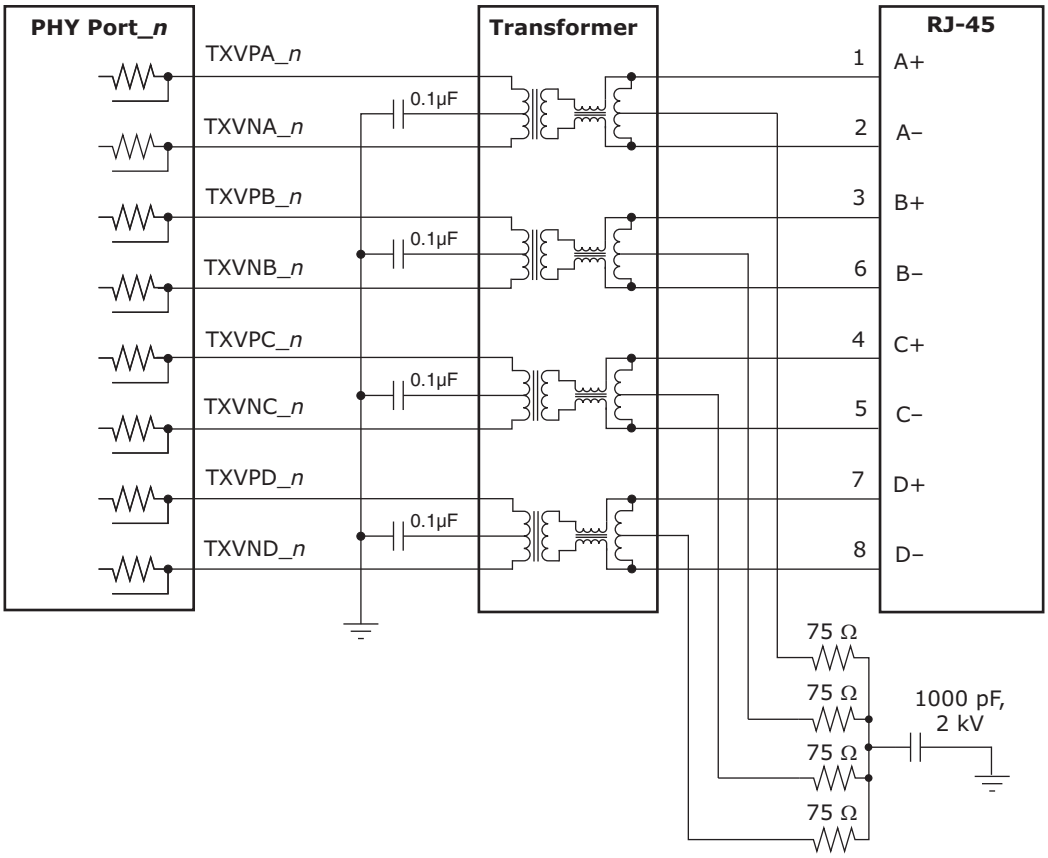
3.6 Cat5 Twisted Pair Media Interface

The VSC8572-02 twisted pair interface is compliant with IEEE 802.3-2008 and the IEEE 802.3az-2010 standard for energy efficient Ethernet.

3.6.1 Voltage Mode Line Driver

Unlike many other gigabit PHYs, the VSC8572-02 uses a patented voltage mode line driver that allows it to fully integrate the series termination resistors, which are required to connect the PHY's Cat5 interface to an external 1:1 transformer. Also, the interface does not require the user to place an external voltage on the center tap of the magnetic. The following illustration shows the connections.

Figure 18 • **Cat5 Media Interface**



3.6.2 Cat5 Autonegotiation and Parallel Detection

The VSC8572-02 supports twisted pair autonegotiation, as defined by IEEE 802.3-2008 Clause 28 and IEEE 802.3az-2010. The autonegotiation process evaluates the advertised capabilities of the local PHY and its link partner to determine the best possible operating mode. In particular, autonegotiation can determine speed, duplex configuration, and master or slave operating modes for 1000BASE-TX. Autonegotiation also enables a connected MAC to communicate with its link partner MAC through the VSC8572-02 using optional next pages, which set attributes that may not otherwise be defined by the IEEE standard.

If the Category 5 (Cat5) link partner does not support autonegotiation, the VSC8572-02 automatically uses parallel detection to select the appropriate link speed.

Autonegotiation is disabled by clearing register 0, bit 12. When autonegotiation is disabled, the state of register bits 0.6, 0.13, and 0.8 determine the device operating speed and duplex mode.

Note: While 10BASE-T and 100BASE-TX do not require autonegotiation, Clause 40 has defined 1000BASE-T to require autonegotiation.

3.6.3 Automatic Crossover and Polarity Detection

For trouble-free configuration and management of Ethernet links, the VSC8572-02 includes a robust automatic crossover detection feature for all three speeds on the twisted pair interface (10BASE-T, 100BASE-T, and 1000BASE T). Known as HP Auto-MDIX, the function is fully compliant with Clause 40 of IEEE 802.3-2008.

Additionally, the device detects and corrects polarity errors on all MDI pairs — a useful capability that exceeds the requirements of the standard.

Both HP Auto-MDIX detection and polarity correction are enabled in the device by default. Default settings can be changed using device register bits 18.5:4. Status bits for each of these functions are located in register 28.

Note: The VSC8572-02 can be configured to perform HP Auto-MDIX, even when autonegotiation is disabled and the link is forced into 10/100 speeds. To enable this feature, set register 18.7 to 0. To use the feature, also set register 0.12 to 0.

The HP Auto-MDIX algorithm successfully detects, corrects, and operates with any of the MDI wiring pair combinations listed in the following table, which shows that twisted pair A (of four twisted paris A, B, C, and D) is connected to the RJ45 connector 1,2 in normal MDI mode.

Table 3 • **Supported MDI Pair Combinations**

RJ45 Connections				
1, 2	**3, 6**	**4, 5**	**7, 8**	**Mode**
A	B	C	D	Normal MDI
B	A	D	C	Normal MDI-X
A	B	D	C	Normal MDI with pair swap on C and D pair
B	A	C	D	Normal MDI-X with pair swap on C and D pair

3.6.4 Manual HP Auto-MDIX Setting

As an alternative to HP Auto-MDIX detection, the PHY can be forced to be MDI or MDI-X using register 19E1, bits 3:2. Setting these bits to 10 forces MDI and setting 11 forces MDI-X. Leaving the bits 00 enables the HP Auto-MDIX setting to be based on register 18, bits 7 and 5.

3.6.5 Link Speed Downshift

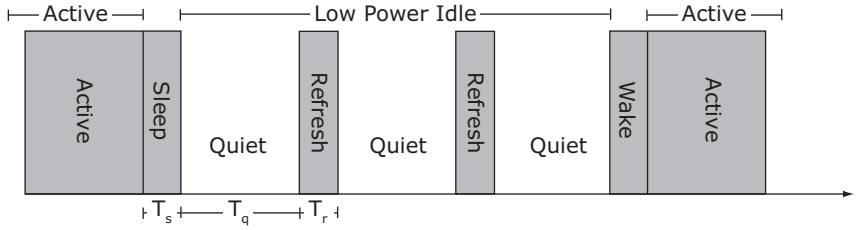
For operation in cabling environments that are incompatible with 1000BASE-T, the VSC8572-02 provides an automatic link speed downshift option. When enabled, the device automatically changes its 1000BASE-T autonegotiation advertisement to the next slower speed after a set number of failed attempts at 1000BASE-T. No reset is required to get out of this state when a subsequent link partner with 1000BASE-T support is connected. This feature is useful in setting up in networks using older cable installations that include only pairs A and B, and not pairs C and D.

To configure and monitor link speed downshifting, set register 20E1, bits 4:1. For more information, see Table 51, page 102.

3.6.6 Energy Efficient Ethernet

The VSC8572-02 supports the IEEE 802.3az-2010 Energy Efficient Ethernet standard. This standard provides a method for reducing power consumption on an Ethernet link during times of low utilization. It uses low power idles (LPI) to achieve this objective.

Figure 19 • **Low Power Idle Operation**



Using LPI, the usage model for the link is to transmit data as fast as possible and then return to a low power idle state. Energy is saved on the link by cycling between active and low power idle states. During LPI, power is reduced by turning off unused circuits and using this method, energy use scales with bandwidth utilization.

The VSC8572-02 uses LPI to optimize power dissipation in 100BASE-TX and 1000BASE-T modes of operation. In addition, the IEEE 802.3az-2010 standard defines a 10BASE-Te mode that reduces transmit signal amplitude from 5 V peak-to-peak to approximately 3.3 V peak-to-peak. This mode reduces power consumption in 10 Mbps link speed and fully interoperates with legacy 10BASE-T compliant PHYs over 100 m Cat5 cable or better.

To configure the VSC8572-02 in 10BASE-Te mode, set register 17E2.15 to 1 for each port. Additional energy efficient Ethernet features are controlled through Clause 45 registers. For more information, see Design Considerations, page 327.

3.6.7 Ring Resiliency

Ring resiliency changes the timing reference between the master and slave PHYs without altering the master/slave configuration in 1000BASE-T mode. The master PHY transmitter sends data based on the local clock and initiates timing recovery in the receiver. The slave PHY instructs node to switch the local timing reference to the recovered clock from other PHYs in the box, freezes timing recovery, and locks clock frequency for the transmitter. The master PHY makes a smooth transition to transmission from local clock to recovered clock after timing lock is achieved.

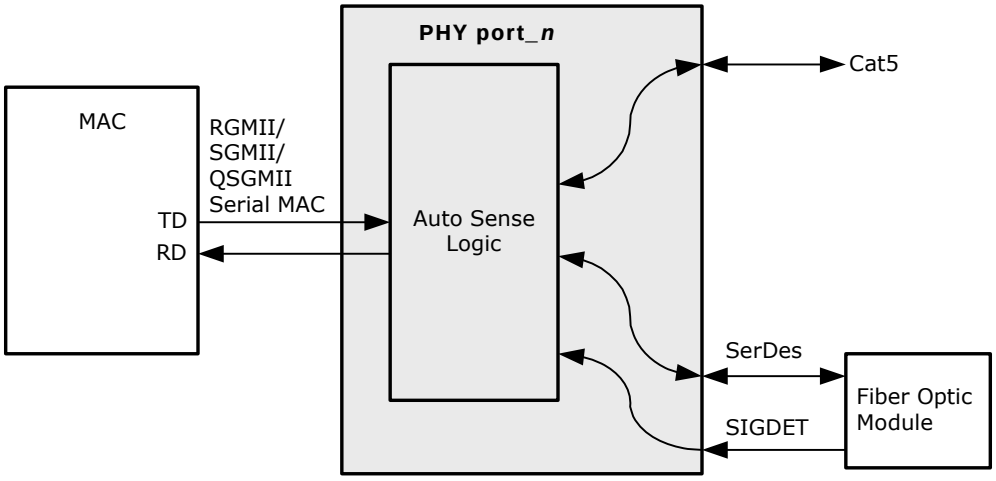
Ring resiliency can be used in synchronous Ethernet systems, because the local clocks in each node are synchronized to a grandmaster clock.

Note: For ring resiliency to successfully exchange master/slave timing over 1000BASE-T, the link partner must also support ring resiliency.

3.7 Automatic Media Sense Interface Mode

Automatic media sense (AMS) mode automatically sets the media interface to Cat5 mode or SerDes mode. The active media mode chosen is based on the automatic media sense preferences set in the device register 23, bit 11. The following illustration shows a block diagram of AMS functionality on ports 0 through 3 of the VSC8572-02 device.

Figure 20 • **Automatic Media Sense Block Diagram**



When both the SerDes and Cat5 media interfaces attempt to establish a link, the preferred media interface overrides a linkup of the nonpreferred media interface. For example, if the preference is set for SerDes mode and Cat5 media establishes a link, Cat5 becomes the active media interface. However, after the SerDes media interface establishes a link, the Cat5 interface drops its link because the preference was set for SerDes mode. In this scenario, the SerDes preference determines the active media source until the SerDes link is lost. Also, Cat5 media cannot link up unless there is no SerDes

media link established. The following table shows the possible link conditions based on preference settings.

Table 4 • **AMS Media Preferences**

Preference Setting	Cat5 Linked, Fiber Not Linked	SerDes Linked, Cat5 Not Linked	Cat5 Linked, SerDes Attempts to Link	SerDes Linked, Cat5 Attempts to Link	Both Cat5 and SerDes Attempt to Link
SerDes	Cat5	SerDes	SerDes	SerDes	SerDes
Cat5	Cat5	SerDes	Cat5	Cat5	Cat5

The status of the media mode selected by the AMS can be read from device register 20E1, bits 7:6. It indicates whether copper media, SerDes media, or no media is selected. Each PHY has four automatic media sense modes. The difference between the modes is based on the SerDes media modes:

- SGMII or QSGMII MAC to AMS and 1000BASE-X SerDes
- SGMII or QSGMII MAC to AMS and 100BASE-FX SerDes
- SGMII or QSGMII MAC to AMS and SGMII (protocol transfer)

For more information about SerDes media mode functionality with AMS enabled, see SerDes Media Interface, page 15.

3.8 Reference Clock

The device reference clock supports both 25 MHz and 125 MHz clock signals. The 1588 differential input clock supports frequencies of 125 MHz to 250 MHz. Both reference clocks can be either differential or single-ended. If differential, they must be capacitively coupled and LVDS compatible.

3.8.1 Configuring the Reference Clock

The REFCLK_SEL2 pin configures the reference clock speed. The following table shows the functionality and associated reference clock frequency.

Table 5 • **REFCLK Frequency Selection**

REFCLK_SEL2	Frequency
0	25 MHz
1	125 MHz

3.8.2 Single-Ended REFCLK Input

To use a single-ended reference clock, an external resistor network is required. The purpose of the network is to limit the amplitude and to adjust the center of the swing. The configurations for a single-ended REFCLK, with the clock centered at 1 V and a 500 mV peak-to-peak swing, are shown in the following illustrations.

Figure 21 • **2.5 V CMOS Single-Ended REFCLK Input Resistor Network**

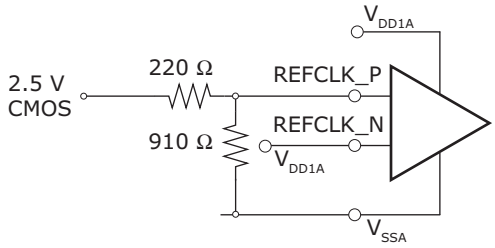


Figure 22 • **3.3 V CMOS Single-Ended REFCLK Input Resistor Network**

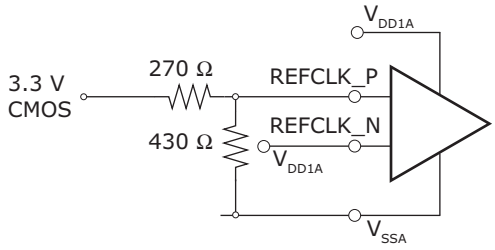
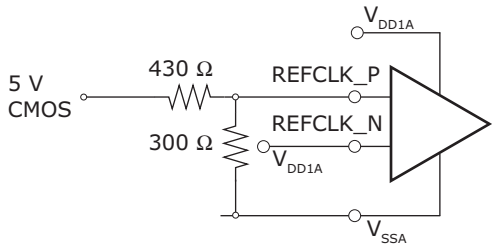


Figure 23 • **5 V CMOS Single-Ended REFCLK Input Resistor Network**

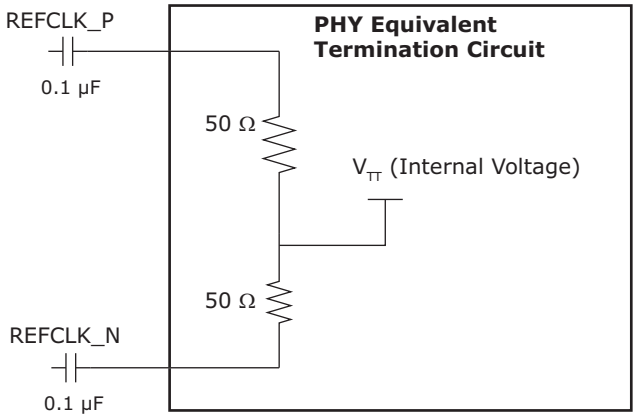


Note: A single-ended 25 MHz reference clock is not guaranteed to meet requirements for QSGMII MAC operation.

3.8.3 Differential REFCLK Input

AC coupling is required when using a differential REFCLK. Differential clocks must be capacitively coupled and LVDS compatible. The following illustration shows the configuration.

Figure 24 • **AC Coupling for REFCLK Input**



3.9 1588 Reference Clock

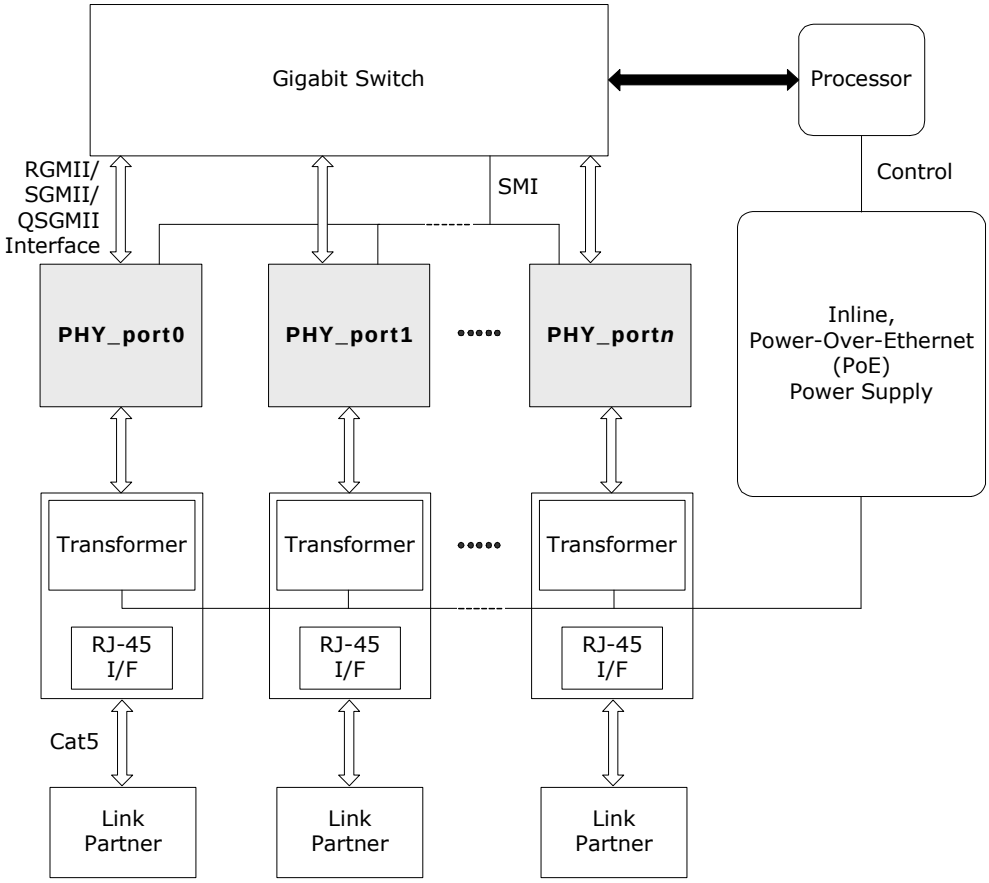
The device 1588 reference clock input supports a continuum of frequencies between 125 MHz and 250 MHz. Both single-ended and differential clocks are supported, but differential clocks are preferred for better performance. If differential, they must be capacitively coupled and compatible. For more information about configuring the clock for single-ended operation, see Reference Clock, page 21.

3.10 Ethernet Inline Powered Devices

The VSC8572-02 can detect legacy inline powered devices in Ethernet network applications. Inline powered detection capability is useful in systems that enable IP phones and other devices (such as wireless access points) to receive power directly from their Ethernet cable, similar to office digital phones receiving power from a private branch exchange (PBX) office switch over telephone cabling. This type of setup eliminates the need for an external power supply and enables the inline powered device to remain active during a power outage, assuming that the Ethernet switch is connected to an uninterrupted power supply, battery, back-up power generator, or other uninterruptable power source.

For more information about legacy inline powered device detection, visit the Cisco Web site at www.cisco.com. The following illustration shows an example of an inline powered Ethernet switch application.

Figure 25 • **Inline Powered Ethernet Switch Diagram**



The following procedure describes the process that an Ethernet switch must perform to process inline power requests made by a link partner (LP) that is, in turn, capable of receiving inline power:

1. Enable the inline powered device detection mode on each VSC8572-02 PHY using its serial management interface. Set register bit 23E1.10 to 1.
2. Ensure that the VSC8572-02 autonegotiation enable bit (register 0.12) is also set to 1. In the application, the device sends a special fast link pulse (FLP) signal to the LP. Reading register bit 23E1.9:8 returns 00 during the search for devices that require power over Ethernet (PoE).
3. The VSC8572-02 PHY monitors its inputs for the FLP signal looped back by the LP. An LP capable of receiving PoE loops back the FLP pulses when the LP is in a powered down state. This is reported when VSC8572-02 register bit 23E1.9:8 reads back 01. It can also be verified as an inline power detection interrupt by reading VSC8572-02 register bit 26.9, which should be a 1, and which is subsequently cleared and the interrupt de-asserted after the read. When an LP device does not loop back the FLP after a specific time, VSC8572-02 register bit 23E1.9:8 automatically resets to 10.
4. If the VSC8572-02 PHY reports that the LP requires PoE, the Ethernet switch must enable inline power on this port, externally of the PHY.
5. The PHY automatically disables inline powered device detection when the VSC8572-02 register bits 23E1.9:8 automatically reset to 10, and then automatically changes to its normal autonegotiation process. A link is then autonegotiated and established when the link status bit is set (register bit 1.2 is set to 1).
6. In the event of a link failure (indicated when VSC8572-02 register bit 1.2 reads 0), it is recommended that the inline power be disabled to the inline powered device external to the PHY. The VSC8572-02 PHY disables its normal autonegotiation process and re-enables its inline powered device detection mode.

3.11 IEEE 802.3af PoE Support

The VSC8572-02 device is compatible with designs that are intended for use in systems that supply power to data terminal equipment (DTE) by means of the MDI or twisted pair cable, as described in IEEE 802.3af Clause 33.

3.12 ActiPHY Power Management

In addition to the IEEE-specified power-down control bit (device register bit 0.11), the device also includes an ActiPHY power management mode for each PHY. This mode enables support for power-sensitive applications. It utilizes a signal-detect function that monitors the media interface for the presence of a link to determine when to automatically power-down the PHY. The PHY wakes up at a programmable interval and attempts to wake up the link partner PHY by sending a burst of FLP over copper media.

The ActiPHY power management mode in the VSC8572-02 is enabled on a per-port basis during normal operation at any time by setting register bit 28.6 to 1.

The following operating states are possible when ActiPHY mode is enabled:

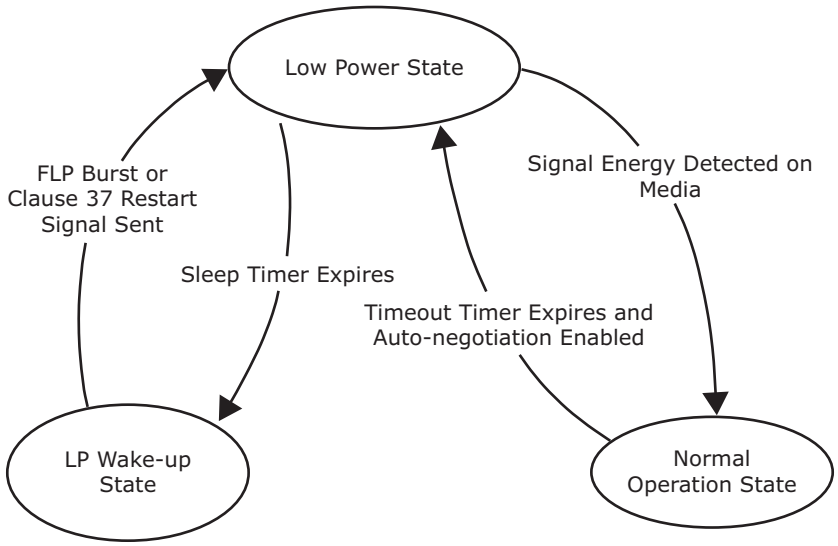
- Low power state
- Link partner wake-up state
- Normal operating state (link-up state)

The VSC8572-02 switches between the low power state and LP wake-up state at a programmable rate (the default is two seconds) until signal energy has been detected on the media interface pins. When signal energy is detected, the PHY enters the normal operating state. If the PHY is in its normal operating state and the link fails, the PHY returns to the low power state after the expiration of the link status time-out timer. After reset, the PHY enters the low power state.

When autonegotiation is enabled in the PHY, the ActiPHY state machine operates as described. When autonegotiation is disabled and the link is forced to use 10BASE-T or 100BASE-TX modes while the PHY is in its low power state, the PHY continues to transition between the low power and LP wake-up states until signal energy is detected on the media pins. At that time, the PHY transitions to the normal operating state and stays in that state even when the link is dropped. When autonegotiation is disabled while the PHY is in the normal operation state, the PHY stays in that state when the link is dropped and does not transition back to the low power state.

The following illustration shows the relationship between ActiPHY states and timers.

Figure 26 • **ActiPHY State Diagram**



3.12.1 Low Power State

In the low power state, all major digital blocks are powered down. However, the SMI interface (MDC, MDIO, and MDINT) functionality is provided.

In this state, the PHY monitors the media interface pins for signal energy. The PHY comes out of low power state and transitions to the normal operating state when signal energy is detected on the media. This happens when the PHY is connected to one of the following:

- Autonegotiation-capable link partner
- Another PHY in enhanced ActiPHY LP wake-up state

In the absence of signal energy on the media pins, the PHY periodically transitions from low-power state to LP wake-up state, based on the programmable sleep timer (register bits 20E1.14:13). The actual sleep time duration is randomized from –80 ms to 60 ms to avoid two linked PHYs in ActiPHY mode entering a lock-up state during operation.

3.12.2 Link Partner Wake-Up State

In the link partner wake-up state, the PHY attempts to wake up the link partner. Up to three complete FLP bursts are sent on alternating pairs A and B of the Cat5 media for a duration based on the wake-up timer, which is set using register bits 20E1.12:11.

In this state, SMI interface (MDC, MDIO, and MDINT) functionality is provided.

After sending signal energy on the relevant media, the PHY returns to the low power state.

3.12.3 Normal Operating State

In the normal operating state, the PHY establishes a link with a link partner. When the media is unplugged or the link partner is powered down, the PHY waits for the duration of the programmable link status time-out timer, which is set using register bit 28.7 and bit 28.2. It then enters the low power state.

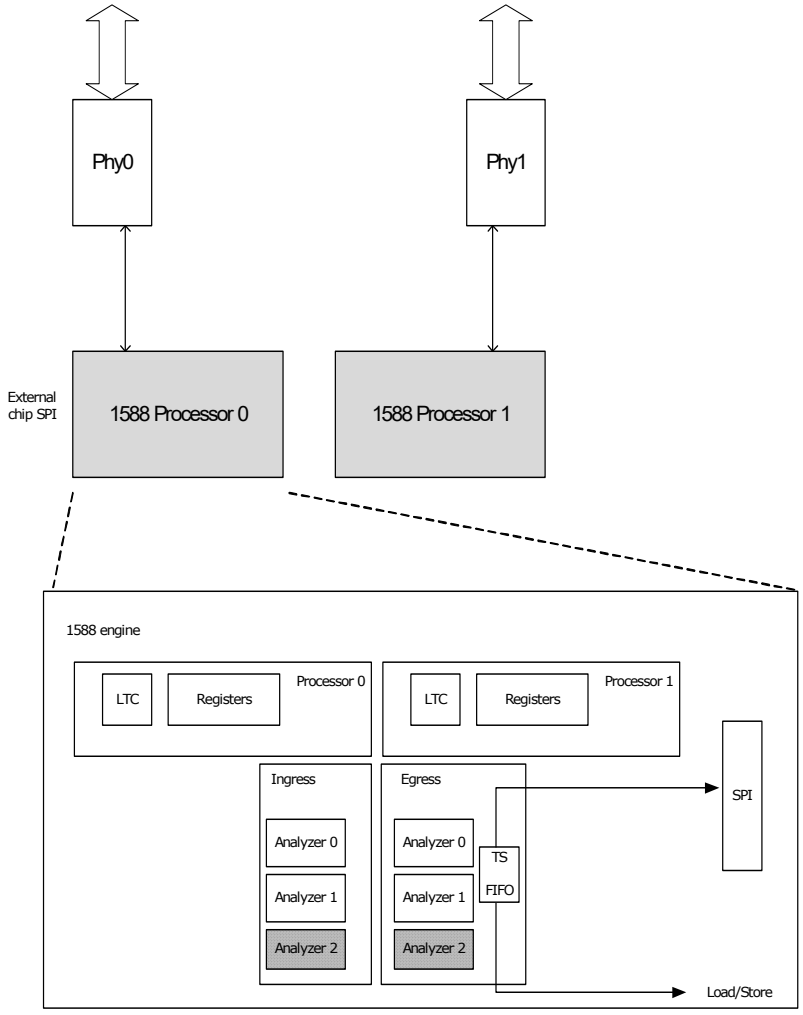
3.13 IEEE 1588 Timestamping Engine

This section provides information about the IEEE 1588 block for the VSC8572-02 device.

3.13.1 IEEE 1588 Block Operation

This section describes the basic operation of the architecture, when configured to work in each of the different IEEE 1588 operation modes. The following illustration shows a block diagram of the IEEE 1588 architecture in the VSC8572-02 device.

Figure 27 • **IEEE 1588 Architecture**



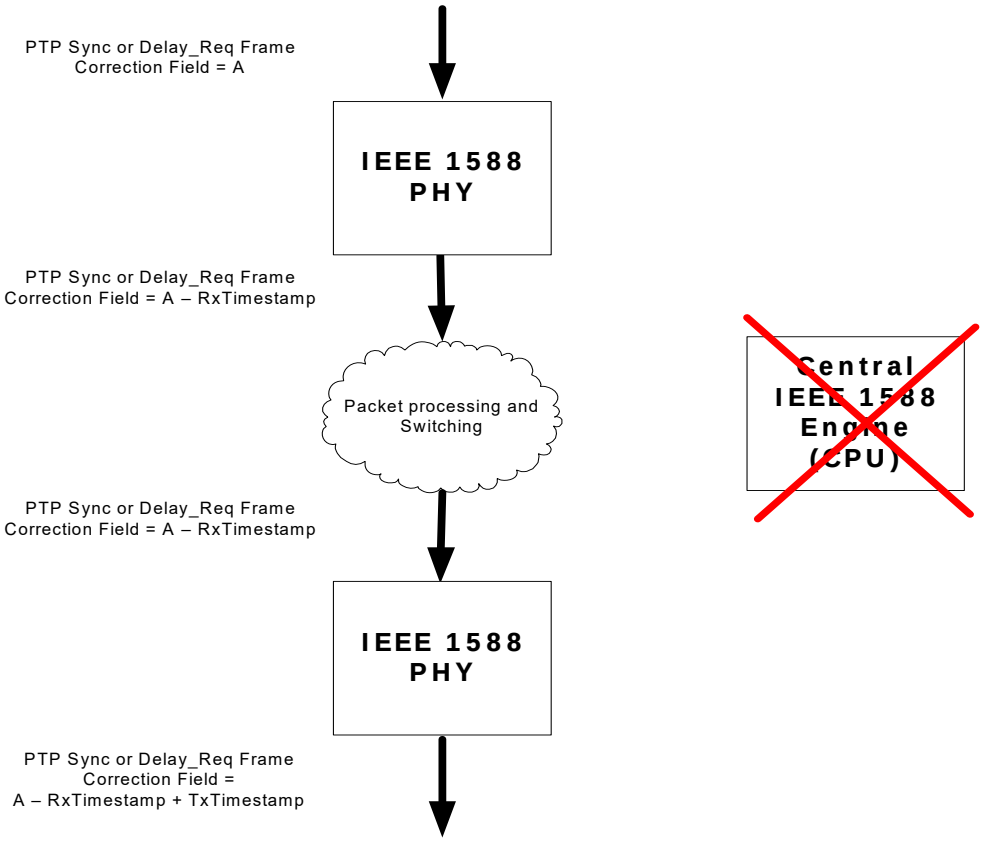
3.13.1.1 One-Step End-to-End Transparent Clock

End-to-end transparent clocks add the residence time (time it takes to traverse from the input to the output port(s) of the system) to all Sync and Delay_Req frames. It does not need to have any knowledge of the actual time, but if it is not locked to the frequency of the 1588 time, it will produce an error that is the ppm difference in frequency times the residence time.

When the TC is frequency-locked by means of 1588 or other methods (SyncE), the error is only caused by sampling inaccuracies.

When an E2E TC needs to recover the frequency using 1588, it must have a PHY with 1588 timestamping support before the 1588 engine, or another way of adding the local time to the correction field. The 1588 Engine is then able to receive Sync frames and adjust the local frequency to match the 1588 time. This can be done by adjusting the Time counter in each PHY or by adjusting the global Timetick clock.

Figure 28 • **One-Step E2E TC Mode A**



3.13.1.2 Ingress, Mode A

When the system works in one-step E2E TC mode, the system needs to forward Sync and Delay_Req frames through the system and add residence time = Egress timestamp – Ingress timestamp to the correction field in the frame before it leaves the system.

Each time the Timestamp block detects a rising edge on the Start_of_Frame_Indicator pulse (synchronized to the clock domain of the PHY core), it saves the value of the Local_Time that is receives from the Local Time Counter into a raw_timestamp register and converts this to raw_timestamp_ns. It then subtracts the value in the local_correction register from the raw_timestamp_ns value and stores the result in an active_timestamp_ns register. The local_correction register is programmed with the fixed latency from the measurement point to the place that the SFD is detected in the PCS/PMA logic. The timestamp block also contains a register that can be programmed with the known link asymmetry. This value is added or subtracted from the correction field, depending on the frame type.

When the frame leaves the PCS/PMA block it is loaded into a FIFO block that delays and stores the frame data for a few clock cycles to allow for later modifications of the frame. The data is also copied to the analyzer block that parses the incoming frame to detect whether it is a 1588 Sync or Delay_Req frame belonging to the PTP domain that the system is operating on.

If the analyzer detects that the frame is a 1588 Sync or Delay_Req frame belonging to the PTP domain(s) of system, it signals to the timestamp block which action to perform (Subtract), along with the correction field of the frame. It also delivers the write offset and data size (location of the correction field inside the frame, 8 bytes wide) to the rewriter.

If the analyzer detects that the frame is not matched, it signals to the Timestamp block and the Rewriter block to ignore the frame (NOP), which allows it to pass unmodified and flushes the saved timestamp in the Timestamp block.

If the Timestamp block gets the Subtract action, it subtracts the value in the active_timestamp_ns register from the value it receives from the analyzer (the original correction field from the frame) and outputs the value on the New_Field bus to the Rewriter block.

The Rewriter block continuously takes data out of the FIFO block and feeds it to the system side PCS/PMA block (or SGMII/RGMII interface) and has a counter that keeps track of the byte positions of the frame. When the Rewriter block receives a signal from the Timestamp block (rising edge of NF) to rewrite a specific position in the frame (that information comes from the analyzer block), it overwrites the position with the data on the New_Field bus and replaces the FCS of the frame. The rewriter also checks the original FCS of the frame to ensure that a frame that is received with a bad FCS and then modified by the rewriter is also sent out with a bad FCS. This is achieved by inverting the new FCS.

The result is that the frame send towards the system now has a correction field containing the value: original correction field – RX timestamp (converted to ns).

The following full calculations are performed:

- Sync frames: Internal Correction field = Original Correction field – (Raw_Timestamp_ns – Local_correction) + Asymmetry
- Delay_req frames: Internal Correction field = Original Correction field – (Raw_Timestamp_ns – Local_correction)

3.13.1.3 Egress, Mode A

The egress side works that same way as ingress, but the analyzer is setup to add the active_timestamp to the correction field.

When a frame is received from the system side PCS/PMA block (or SGMII/RGMII interface) it is loaded into a FIFO block that delays and stores the frame data for a few clock cycles to allow for later modifications of the frame. The data is also copied to the analyzer block that parses the incoming frame to detect whether it is a 1588 Sync or Delay_Req frame belonging to the PTP domain that the system is operating on.

If the analyzer detects that the frame is a 1588 Sync or Delay_Req frame belonging to the PTP domain(s) of system, it signals to the timestamp block which action to perform (Add), along with the correction field of the frame. It also delivers the write offset and data size (location of the correction field inside the frame, 8 bytes wide) to the rewriter.

If the analyzer detects that the frame is not matched, it signals the Timestamp block and the Rewriter block to ignore the frame (let it pass unmodified and flush the saved timestamp in the Timestamp block).

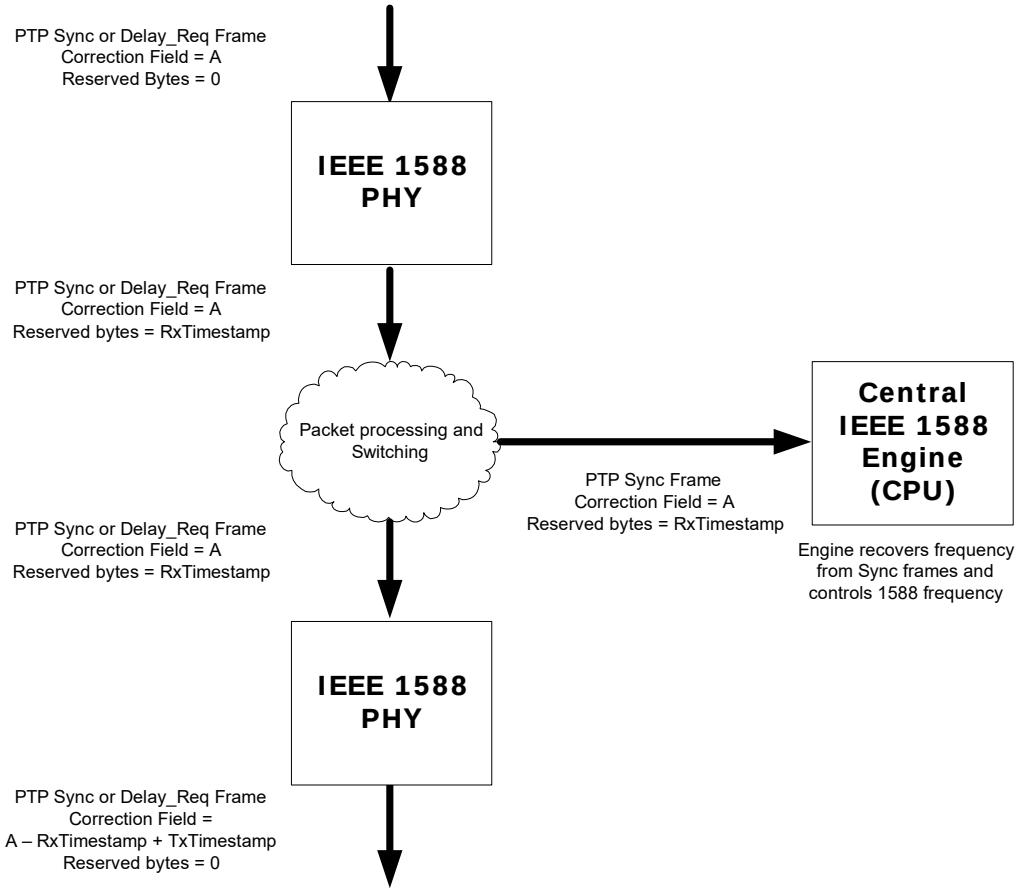
If the Timestamp block gets the Add action, it adds the current value of the active_timestamp_ns register with the value of the correction field it gets from the analyzer and outputs the value on the New_Field bus to the Rewriter block.

The Rewriter block continuously takes data out of the FIFO block and feeds it to the line side PCS/PMA block and has a counter that keeps track of the byte positions of the frame (detects SFD and counts until next SFD). When the Rewriter block receives a signal from the Analyzer block to rewrite a specific position in the frame, it overwrites the position with the data on the New_Field bus and replaces the FCS of the frame. The rewriter also checks the original FCS of the frame and ensures that a frame that is received with a bad FCS and then modified by the rewriter is also sent out with a bad FCS. This is achieved by inverting the new FCS.

The following full calculations are performed:

- Sync frames: Correction field = Internal Correction field + (Raw_Timestamp_ns + Local_correction)
- Delay_req frames: Correction field = Internal Correction field + (Raw_Timestamp_ns + Local_correction) – Asymmetry

Figure 29 • **One-Step E2E TC Mode B**



3.13.1.4 Ingress, Mode B

In Ingress Mode B, all calculations are performed at the Egress port. When the system works in one-step E2E TC mode, the system need to forward Sync and Delay_Req frames through the system and add residence time = Egress timestamp – Ingress timestamp to the correction field in the frame before it leaves the system.

On the ingress side, when the analyzer detects Sync or Delay_Req frames it adds the RX timestamp to the four reserved bytes in the PTP frame.

The following full calculations are performed:

- Sync frames: Reserved_bytes = Raw_Timestamp_ns – Local_correction + Asymmetry
- Delay_req frames: Reserved_bytes = Raw_Timestamp_ns – Local_correction

3.13.1.5 Egress, Mode B

All the calculations are done at the egress side. When the analyzer detects Sync or Delay_Req frames it performs the following calculation:

Correction field = Original correction field + TX timestamp – RX timestamp

The value of the RX timestamp is extracted from four reserved bytes in the PTP header.

The four reserved bytes are cleared back to 0 before transmission.

The result is that every Sync and Delay_Req frame that belongs to the PTP domain(s) and is configured as one-step E2E TC in the system will exit the system with a correction field that contains the following:

Correction field = Original correction field + TX timestamp – RX timestamp

All this is done without any interaction with a CPU system, other than the initial setup. There is no bandwidth expansion. Standard switching/routing tunneling can be done between the ingress and egress PHY, provided that the analyzers in the ingress PHY and egress PHY are set up to catch the sync and Delay_req on both. If the PTP sync and Delay_req frames are modified inside the system, the egress analyzer must be able to detect the egress sync and delay_req frames; otherwise, the egress sync and Delay_req frames will have an incorrect correction field.

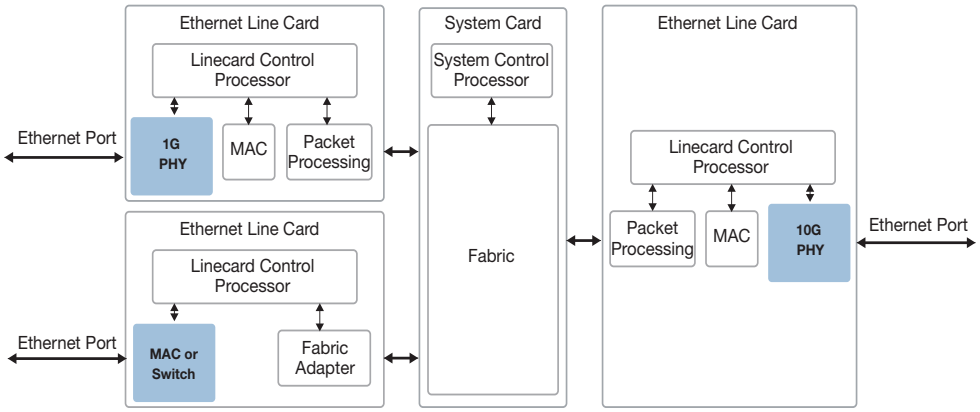
The following full calculations are performed:

- Sync frames: Correction field = Original Correction field + (Raw_Timestamp_ns + Local_correction) – Reserved_bytes
- Delay_req frames: Correction field = Original Correction field + (Raw_Timestamp_ns + Local_correction) – Reserved_bytes – Asymmetry

3.13.2 Supporting IEEE 1588 Timestamping Applications

This section describes the integrated PTP block that supports IEEE 1588-2008 timestamping with encapsulation support.

Figure 30 • **Linecard E2E TC PHY application**



3.13.3 Application 1: IEEE 1588 One-Step E2E TC in Systems

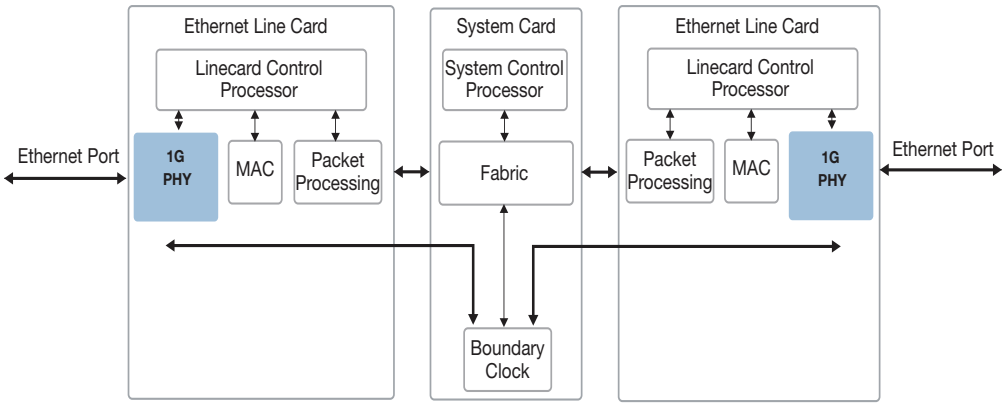
Unique advantages for implementing IEEE 1588-2008:

- When several VSC8572-02 devices or Microsemi PHYs with integrated 1588 timestamping blocks are used on all ports within the system that support IEEE 1588 one-step E2E TC, the rest of the system does not need to be IEEE 1588 aware and there is no CPU maintenance needed once the system is set up
- As all the PHYs in a system can be configured the same way, it supports fail-over of IEEE 1588 masters without any CPU intervention
- VSC8572-02 and another Microsemi PHYs with integrated 1588 timestamping block also works for pizza box solutions, where the switch/router can be upgraded to support IEEE 1588 E2E TC
- The requirements to the rest of the system are:
- Delivery of a synchronous global timetick clock (or reference clock) to the PHYs
- Delivery of a global timetick load, that synchronizes the local time counters in each port.
- CPU access to each PHY to setup the required configuration. Can be MDC/MDIO or a dedicated CPU interface.

3.13.4 Application 2: IEEE 1588 TC and BC in Systems

This is the basically the same system as above, with the addition of a central IEEE 1588 engine (Boundary Clock). The 1588 engine is most likely a CPU system, possible together with hardware support functions to generate Sync frames (for BC and ordinary clock masters). The switch/fabric needs to have the ability to redirect (and copy) PTP frames to the 1588 engine for processing.

Figure 31 • **BC Linecard Application**



This solution also works for pizza boxes. To ensure that blade redundancy works, it the PHYs for the redundant blades must have the same 1588-in-the-PHY configuration.

The requirements to the rest of the system are:

- Delivery of a synchronous global timetick clock (or reference clock) to the PHYs
- Delivery of a global timetick load, that synchronizes the local time counters in each port
- CPU access to each PHY to setup the required configuration. For one-step support this can be MDC/MDIO. For two-step support, a higher speed CPU interface might be required (depending on the number of timestamps that are required to be read by the CPU). In blade systems it might be required to have a local CPU on the blade that collects the information and sends it to the 1588 engine by means of the control plane or the data plane. In advanced MAC/Switch devices this might be an internal CPU
- Fabric must be able to detect 1588 frames and redirect some of them to the central 1588 engine

The same solution can also be used to add Y.1731 delay measurement support. This does not require a local CPU on the blade, but the fabric must be able to redirect OAM frames to a local/central OAM processor

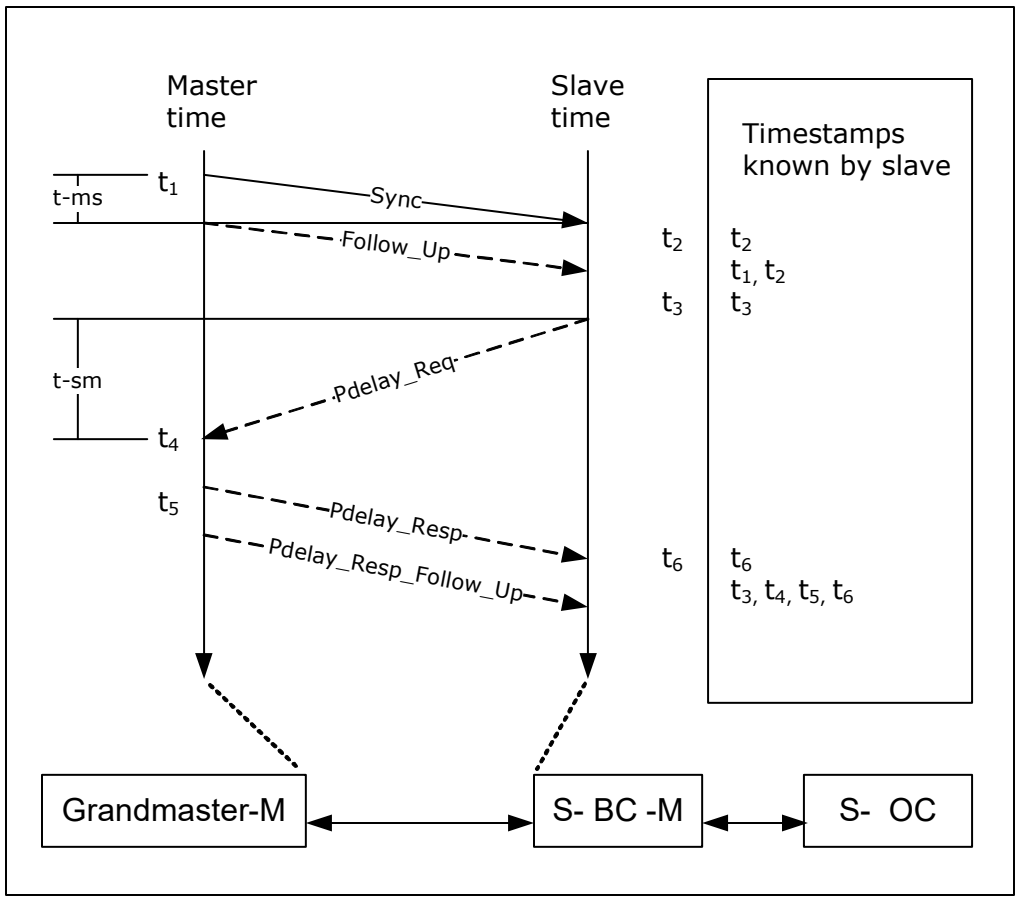
3.13.5 Application 3: Enhancing IEEE 1588 Accuracy for CE Switches and MACs

Connecting VSC8572-02 or other Microsemi PHYs have integrated 1588 time stamping in front of the CE Switches and MACs improves the accuracy of the 1588 timestamp calculation. This is due to the clock boundary for the XAUI SGMII/RGMII interface. It will also add support for one-step TC and BC on the Jaguar and Caracal family of devices.

3.13.6 Supporting One-Step Peer-to-Peer Transparent Clock

In P2P TC, the P2P TC device is actively sending out and receiving pDelay_Req and pDelay_Resp messages and calculating the path delays to each neighbor node in the PTP network. When a Sync frame traverses a P2P TC, the correction field is updated with both the residence time and the calculated path delay on the port that the Sync frame came in on.

Figure 32 • **Delay Measurements**



To calculate the path delays on a link, the 1588 engine (located somewhere in the system) generates Pdelay_Req messages on all ports. When transmitted, the actual TX timestamp t3 is saved for the CPU to read.

When a P2P TC, BC, or OC receives a Pdelay_Req frame, it saves the Rx timestamp (t4) and generates a Pdelay_Resp frame, which adds t5 – t4 to the correction field copied from the received Pdelay_Req frame, where t5 is the time that the Pdelay_Resp leaves the port (t5).

When a P2P TC receives the Pdelay_Resp frame, it saves the RX timestamp (t6) and then calculates the path delay as (t6 – t3 – the correction field of the frame)/2. The timestamp corrections are combined into a single formula as follows:

Path delay = (t6 – (t3 + (t5 – t4))/2 = (t6 – t3 – t5 + t4)/2 = ((t4 – t3) + (t6 – t5))/2

The two path delays are divided by two, but in such a way as to cancel out any timing difference between the two devices.

A slight modification can be made to the algorithm to remove the CPU processing overhead of reading the t3 time stamp. To modify the algorithm, the IEEE 1588 engine should send the Pdelay_req message with a software generated t3 value in the origin time stamp the sub-second value of the t3 time stamp in the reserved bytes of the PTP header, and a correction field of 0. The software generated t3 time stamp should just be within a second before the actual t3 time. The egress PHY should then be configured to perform E2E TC egress operation, meaning calculate the "residence time" from the inserted t3 time stamp to the actual t3 time and insert this value in the correction field of the frame. When the IEEE 1588 engine receives the corresponding Pdelay_resp frame back it can use the software generated t3 value as the correction field of the Pdelay_resp frame will contain a value that compensates for the actual t3 transmission time.

A P2P TC adds the calculated one-way path delay to the Ingress correction field, and this ensures that the timestamp + correction field in the egress Sync frames is accurate and a slave connected to the P2P TC only needs to add the link delay from the TC to the slave.

The following section describes both the standard and modified methods for taking P2P measurements.

3.13.6.1 Ingress, Special

If the analyzer detects that the frame is a 1588 Sync frame belonging to the PTP domain(s) of system, it signals to the timestamp block which action to perform (Subtract_p2p), along with the correction field of the frame. It also delivers the write offset and data size (location of the correction field inside the frame, 8 bytes wide) to the rewriter.

If the analyzer detects that the frame is a 1588 Pdelay_Req frame belonging to the PTP domain(s) of system, it signals to the timestamp block which action to perform (Subtract), along with the correction field of the frame. It also delivers the write offset and data size (location of the correction field in the PTP header, 8 bytes wide) to the rewriter.

If the analyzer detects that the frame is a 1588 Pdelay_Resp frame belonging to the PTP domain(s) of system, it signals to the timestamp block which action to perform (Add). It also delivers the write offset and data size (location of the correction field, 8 bytes wide) to the rewriter.

If the Timestamp block gets the Subtract_p2p action, it subtracts the value in the active_timestamp_ns_p2p register from the correction_field data and outputs the value on the New_Field bus to the Rewriter block.

If the Timestamp block gets the Subtract action, it subtracts the value in the active_timestamp_ns register from the correction field value and outputs the value on the New_field bus to the Rewriter block.

If the Timestamp block gets the Add action, it adds the correction field value to the value in the active_timestamp_ns register and outputs the value on the New_Field bus to the Rewriter block.

The following full calculations are performed:

- Sync frames: Internal Correction field = Original Correction field – (Raw_Timestamp_ns – Local_correction) + Path_delay + Asymmetry
- Pdelay_req frames: Internal Correction field = Original Correction field – (Raw_Timestamp_ns – Local_correction)
- Pdelay_resp frames: Internal Correction field = Original Correction field + (Raw_Timestamp_ns – Local_correction) + Asymmetry

3.13.6.2 Egress, Special

If the analyzer detects that the frame is a 1588 Sync frame belonging to the PTP domain(s) of system, it signals to the timestamp block which action to perform (Add), along with the correction field of the frame. It also delivers the write offset and data size (location of the correction field inside the frame, 8 bytes wide) to the rewriter.

If the analyzer detects that the frame is a 1588 Pdelay_Req frame belonging to the PTP domain(s) of system, it signals to the timestamp block which action to perform (Subtract), along with the original correction field of the frame (will have the value of 0). It also delivers the write offset and data size (location of the correction field inside the frame, 8 bytes wide) to the rewriter.

If the analyzer detects that the frame is a 1588 Pdelay_Resp frame belonging to the PTP domain(s) of system, it signals to the timestamp block which action to perform (Add), along with the original correction field of the frame. It also delivers the write offset and data size (location of the correction field inside the frame, 8 bytes wide) to the rewriter.

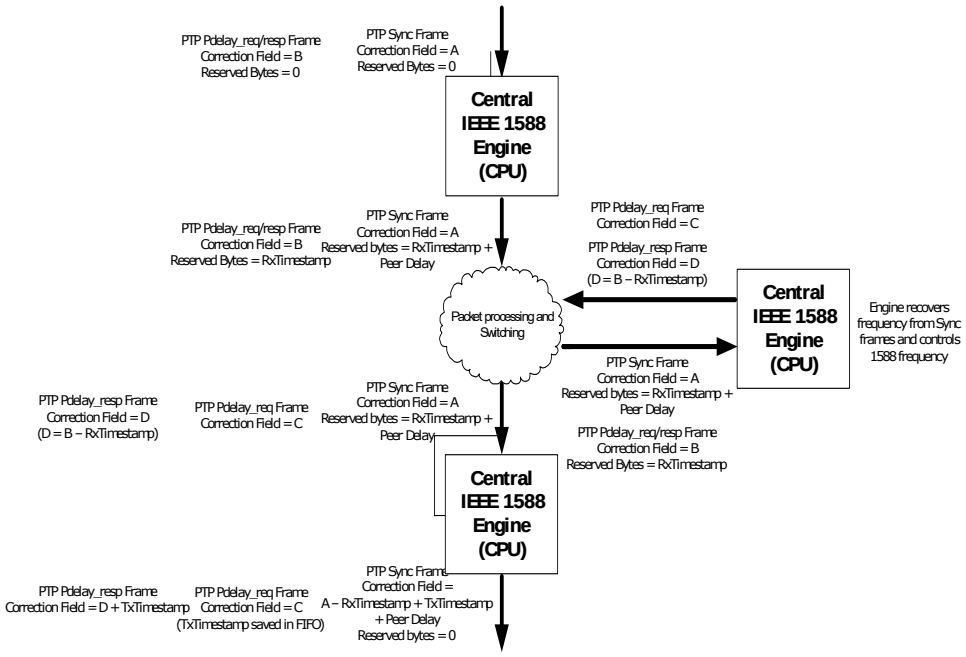
If the analyzer detects that the frame is not matched, it signals to the Timestamp block and the Rewriter block to ignore the frame (let it pass unmodified and flush the saved timestamp in the Timestamp block).

The following full calculations are performed:

- Sync frames: Correction field = Internal Correction field + (Raw_Timestamp_ns + Local_correction)
- Pdelay_req frames: Correction field = Internal Correction field – (Raw_Timestamp_ns + Local_correction) – Asymmetry

- Pdelay_resp frames: Correction field = Original Correction field + (Raw_Timestamp_ns + Local_correction)

Figure 33 • **One-Step P2P TC Standard**



3.13.6.3 Ingress, Standard

If the analyzer detects that the frame is a 1588 Sync frame belonging to the PTP domain(s) of system, it signals to the timestamp block which action to perform (subtract_p2p), along with the correction field of the frame. It also delivers the write offset and data size (location of the correction field inside the frame, 8 bytes wide) to the rewriter.

If the analyzer detects that the frame is a 1588 Pdelay_Req frame belonging to the PTP domain(s) of system, it signals to the timestamp block which action to perform (Write). It also delivers the write offset and data size (location of the reserved 4 bytes in the PTP header we use to save the ns part of the RX timestamp, 4 bytes wide) to the rewriter.

If the analyzer detects that the frame is a 1588 Pdelay_Resp frame belonging to the PTP domain(s) of system, it signals to the timestamp block which action to perform (Write). It also delivers the write offset and data size (location of the reserved 4 bytes in the PTP header we use to save the ns part of the RX timestamp, 4 bytes wide) to the rewriter.

If the Timestamp block gets the Subtract_p2p action, it subtracts the value in the active_timestamp_ns_p2p register from the correction_field data and outputs the value on the New_Field bus to the Rewriter block.

If the Timestamp block gets the Write action, it outputs the value of the active_timestamp_ns register on the New_field bus to the Rewriter block.

The following full calculations are performed:

- Sync frames: Internal Correction field = Original Correction field – (Raw_Timestamp_ns – Local_correction) + Path_delay + Asymmetry
- Pdelay_req frames: Reserved_bytes = Raw_Timestamp_ns – Local_correction
- Pdelay_resp frames: Reserved_bytes = Raw_Timestamp_ns – Local_correction + Asymmetry

3.13.6.4 Egress, Standard

If the analyzer detects that the frame is a 1588 Sync frame belonging to the PTP domain(s) of system, it signals to the timestamp block which action to perform (Add), along with the correction field of the frame.

It also delivers the write offset and data size (location of the correction field inside the frame, 8 bytes wide) to the rewriter.

If the analyzer detects that the frame is a 1588 Pdelay_Req frame belonging to the PTP domain(s) of system, it signals to the timestamp block which action to perform (Write, Save), along with the original correction field of the frame (will have the value of 0). It also delivers the write offset and data size (0 No data is actually written into the frame) to the rewriter. In addition it outputs the field that holds the frame identifier (sequenceId from the PTP header) to the timestamp FIFO, to save along with the TX timestamp.

If the analyzer detects that the frame is a 1588 Pdelay_Resp frame belonging to the PTP domain(s) of system, it signals to the timestamp block which action to perform (Add - this requires that the 1588 engine has subtracted the RX timestamp from the correction field), along with the original correction field of the frame. It also delivers the write offset and data size (location of the correction field inside the frame, 8 bytes wide) to the rewriter.

If the Timestamp block gets the Write, Save action it outputs the value of the active_timestamp_ns register on the New_field bus to the Rewriter block and sets the save_timestamp bit.

If the Timestamp block gets the Add action, it adds the correction field value to the value in the active_timestamp_ns register and outputs the value on the New_Field bus to the Rewriter block.

The TX Timestamp FIFO block contains an (implementation specific) amount of buffer memory. It simply stores the TX timestamp values that it receives from the Timestamp block together with the frame identifier data it receives from the Analyzer block and has a CPU interface that allows the 1588 engine to read out the timestamp sets (Frame identifier + New TX timestamp).

The following full calculations are performed:

- Sync frames: Correction field = Internal Correction field + (Raw_Timestamp_ns + Local_correction)
- Pdelay_req frames: FIFO = Raw_Timestamp_ns + Local_correction – Asymmetry
- Pdelay_resp frames: Correction field = Internal Correction field + (Raw_Timestamp_ns + Local_correction)

3.13.7 Supporting One-Step Boundary Clock/Ordinary Clock

In one-step boundary clock, the BC device acts as an ordinary clock slave on one port and as master on all the other ports. On the master ports, Sync frames are transmitted from the 1588 engine that holds the Origin timestamp. These frames will have the correction field or the full TX timestamp updated on the way out though the PHY.

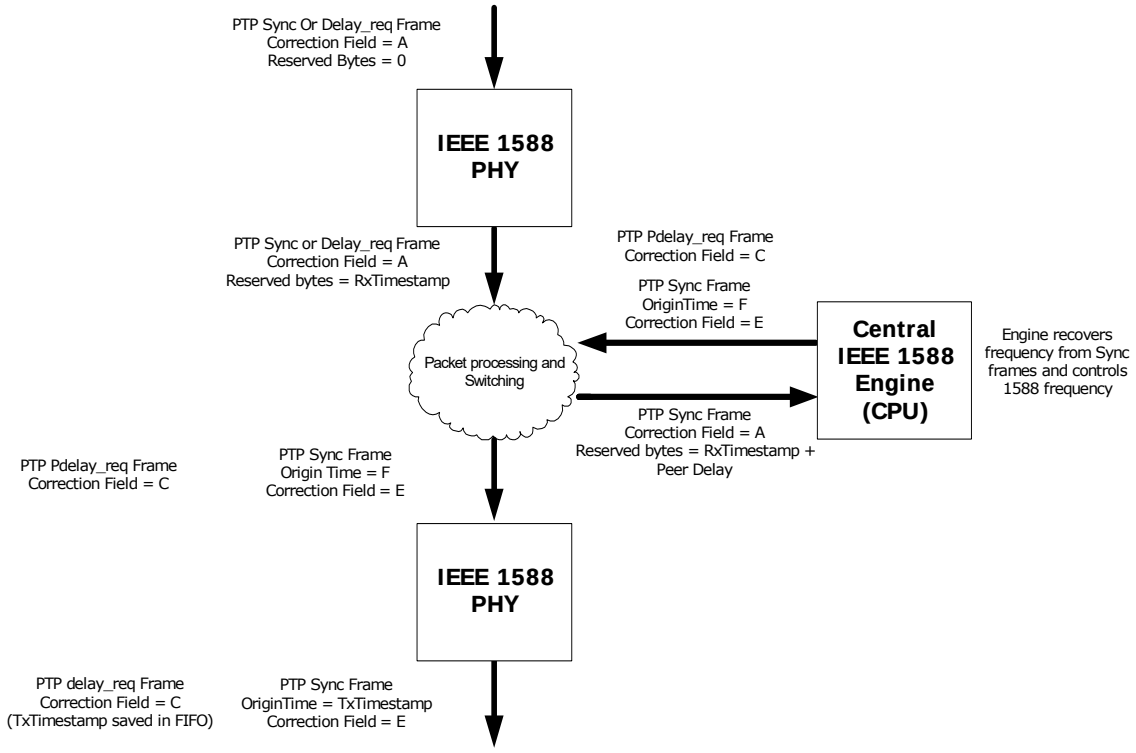
Master ports also receive Delay_req from the slaves and respond with Delay_resp messages. The Delay_req messages are time stamped on the way through the PHY and the 1588 engine receives the Delay_req frame and generates a Delay_resp message. The Delay_resp messages are not event messages and are passed though the PHY as any other frame.

The port that is configured as slave receives Sync frames from its master. The Sync frames get an RX timestamp added in the PHY and forwarded to the 1588 engine.

The 1588 engine also generates Delay_req frames that are send out on the port that is configures as slave port.

Boundary clocks and ordinary clocks must also reply to Pdelay_req messages just as P2P TC, but the procedure is the same as for the P2P TC, so this is not described here.

Figure 34 • **One-Step E2E BC**



3.13.7.1 Ingress

If the analyzer detects that the frame is a 1588 Sync or Delay_req frame belonging to the PTP domain(s) of system, it signals to the timestamp block which action to perform (Write). It also delivers the write offset and data size (location of the four reserved bytes in the PTP header, 4 bytes wide) to the rewriter.

If the Timestamp block gets the Write action, it puts the value of the active_timestamp register out on the New_field bus to the Rewriter block and the rewriter block adds this timestamp (ns part of it) to the four reserved bytes in the frame and recalculates FCS.

The following full calculations are performed:

- Sync frames: Reserved_bytes = (Raw_Timestamp_ns – Local_correction) + Asymmetry
- Delay_req frames: Reserved_bytes = (Raw_Timestamp_ns – Local_correction)

3.13.7.2 Egress

If the analyzer detects that the frame is a 1588 Sync frame belonging to the PTP domain(s) of system, it signals to the timestamp block which action to perform (Write), along with correction field of the frame (contains the offset from the system PTP domain to any other domain that the frame belong to - set by the 1588 engine that generates the frame). It also delivers the write offset and data size (location of the TX timestamp inside the frame, 10 bytes wide) to the rewriter.

If the analyzer detects that the frame is a 1588 Delay_Req frame belonging to the PTP domain(s) of system, it signals to the timestamp block which action to perform (Write, Save). It also delivers the write offset and data size (location of the TX timestamp inside the frame, 10 bytes wide to the rewriter. It also outputs 10 bytes of frame identifier to the TX Timestamp FIFO, to be saved along with the TX timestamp.

If the Timestamp block gets the Write, Save action it outputs the current value from the active_timestamp register on the New_field bus to the Rewriter Timestamp_fifo blocks and signals to the Timestamp FIFO block that it must save the New_field data along with the frame identifier data it received from the Analyzer block.

The TX Timestamp FIFO block contains an (implementation specific) amount of buffer memory. It simply stores the TX timestamp values that it receives from the Timestamp block together with the frame

identifier data it receives from the Analyzer block and has a CPU interface that allows the 1588 engine to read out the timestamp sets (Frame identifier + New TX timestamp).

The following full calculations are performed:

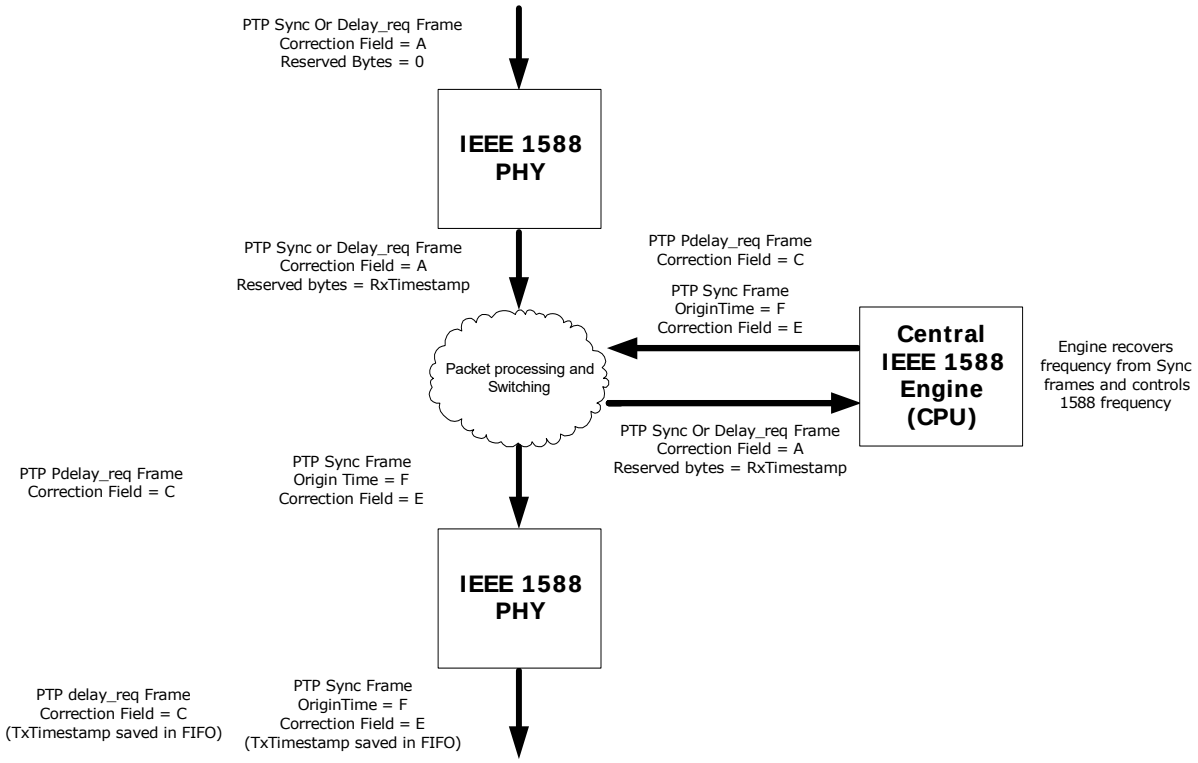
- Sync frames: OriginTimestamp = (Raw_Timestamp + Local_correction)
- Delay_req frames: OriginTimestamp = (Raw_Timestamp + Local_correction) – Asymmetry

3.13.8 Supporting Two- Step Boundary/Ordinary Clock

Two-steps clocks are used in systems that cannot update the correction field on-the-fly and this requires more CPU power.

Every time a TX timestamp is sent in a frame, the 1588 engine needs to read the actual TX transmission time from the Timestamp FIFO and issue a follow-up message containing this timestamp.

Figure 35 • **Two-Step E2E BC**



3.13.8.1 Ingress

If the analyzer detects that the frame is a 1588 Sync or Delay_req frame belonging to the PTP domain(s) of system, it signals to the timestamp block which action to perform (Write). It also delivers the write offset and data size (location of the four reserved bytes in the PTP header, 4 bytes wide) to the rewriter.

If the Timestamp block gets the Write action, it puts the value of the active_timestamp register out on the New_field bus to the Rewriter block and the rewriter block adds this timestamp (ns part of it) to the four reserved bytes in the frame and recalculates FCS.

Note: When secure timing delivery is required, the 1588 engine must revert the four reserved bytes back to 0 before performing integrity check.

The following full calculations are performed:

- Sync frames: Reserved_bytes = (Raw_Timestamp – Local_correction) + Asymmetry
- Delay_req frames: Reserved_bytes = (Raw_Timestamp – Local_correction)

3.13.8.2 Egress

If the analyzer detects that the frame is a 1588 Sync or Delay_Req frame belonging to the PTP domain(s) of system, it signals to the timestamp block which action to perform (Write, Save). The analyzer also delivers the write offset and data size (but as nothing is to be overwritten the values will be 0) to the rewriter. The analyzer outputs up to 15 bytes of frame identifier to the TX Timestamp FIFO to be saved along with the TX timestamp. The frame identifier must include, at minimum, the sequenceId field so the CPU can match the timestamp with the follow-up frame.

If the Timestamp block gets the Write, Save action it outputs the current value from the active_timestamp register on the New_field bus to the Rewriter (and timestamp FIFO) and sets the save_timestamp bit. The Timestamp_fifo block saves the New_field data along with the frame identifier data it received from the Analyzer block.

The following full calculations are performed:

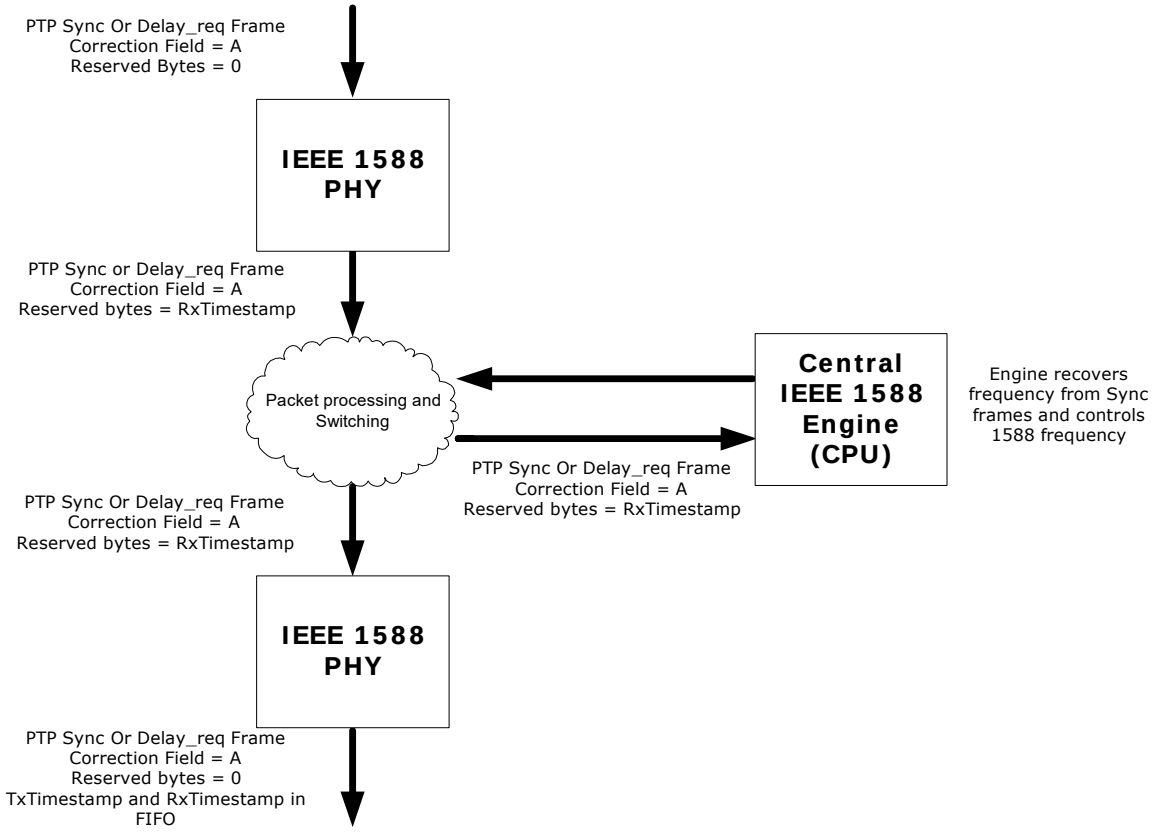
- Sync frames: FIFO = (Raw_Timestamp + Local_correction)
- Delay_req frames: FIFO = (Raw_Timestamp + Local_correction) – Asymmetry

3.13.9 Supporting Two-Step Transparent Clock

In two-step transparent clocks, the RX and TX timestamps are saved for the 1588 Engine to read and the follow-up message is redirected to the 1588 engine so that it can update the correction field with the residence time.

Even though two-step transparent clocks can be used with this architecture, it is also possible to process the frames in the same manner as a one-step TC, because the slaves are required to take both the corrections fields from the Sync frames and the follow-up frames into account. This significantly reduces the CPU load for the TC. The following illustration shows two-step transparent clock normal operation.

Figure 36 • **Two-Step E2E TC**



3.13.9.1 Ingress

If the analyzer detects that the frame is a 1588 Sync or Delay_Req frame belonging to the PTP domain(s) of system, it signals to the timestamp block which action to perform (Write). The analyzer also delivers the write offset and data size to the rewriter (four reserved bytes in the PTP header, which will be passed out on the egress port of the system). A changed reserved value may be significant in security protection. This method allowed the frames to be copied to the 1588 engine, so that it can extract the RX timestamp and that it knows that it needs to read the TX timestamps to be ready for the follow up message. It is also possible to save the RX timestamp value along with the TX timestamp in the TX timestamp FIFO.

If the Timestamp block gets the Write action, it outputs the current value from the active_timestamp register on the New_field bus to the Rewriter and the rewriter writes the ns part of the timestamp into the reserved bytes and recalculates FCS.

The following full calculations are performed:

- Sync frames: Reserved_bytes = (Raw_Timestamp_ns – Local_correction) + Asymmetry
- Delay_req frames: Reserved_bytes = Raw_Timestamp_ns – Local_correction

3.13.9.2 Egress

If the analyzer detects that the frame is a 1588 Sync or Delay_Req frame belonging to the PTP domain(s) of system, it signals to the timestamp block which action to perform (Write, Save). The analyzer also delivers the write offset and data size (but as nothing is to be overwritten the values will be 0) to the rewriter. The analyzer outputs 10 bytes of frame identifier to the TX Timestamp FIFO to be saved along with the TX timestamp. The frame identifier must include, at minimum, the sequenceId field so the CPU can match the timestamp with the follow-up frame. The analyzer also outputs the offset for the reserved fields in the PTP header to the rewriter, so that the rewriter field is reset to 0 and the temporary RX timestamp value is cleared.

If the Timestamp block gets the Write, Save action it outputs the current value from the active_timestamp register on the New_field bus to the Rewriter (and timestamp FIFO) and sets the save_timestamp bit. The Timestamp_fifo block saves the New_field data along with the frame identifier data it received from the Analyzer block. The frame identifier data that is saved can contain the reserved field in the PTP header that was written with the RX timestamp, so that the CPU now can read the set of TX and RX timestamp from the TX timestamp FIFO.

The following full calculations are performed:

- Sync frames: FIFO = Raw_Timestamp_ns + Local_correction (reserved_bytes containing the RX timestamp saved together with TX timestamp)
- Delay_req frames: FIFO = Raw_Timestamp_ns + Local_correction – Asymmetry (reserved_bytes containing the RX timestamp saved together with TX timestamp)

3.13.10 Calculating Y.1731 OAM Delay Measurements

Frame delay measurements can be made as one-way and two-way delay measurements. Microsemi recommends that the delay measurement be measured before the packets enter the queues, if the purpose is to measure the delay for different priority traffic, but it can be used with timestamping in the PHY to measure the delay through the network devices placed in the path between the measurement points.

The function is mainly an on-demand OAM function, but it can run continuously.

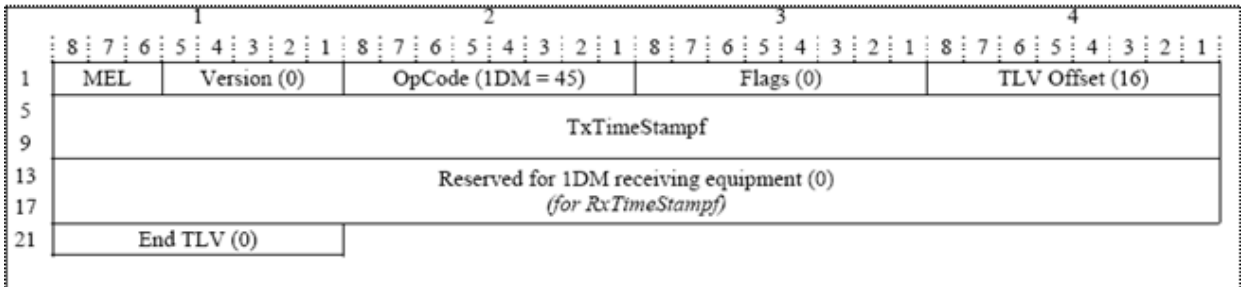
3.13.11 One-Way Delay Measurements

One-way delay measurements require that the two peers are synchronized in time. When they are not synchronized, only frame delay variations can be measured.

The MEP periodically sends out 1DM OAM frames containing a TxTimeStampf value in IEEE 1588 format.

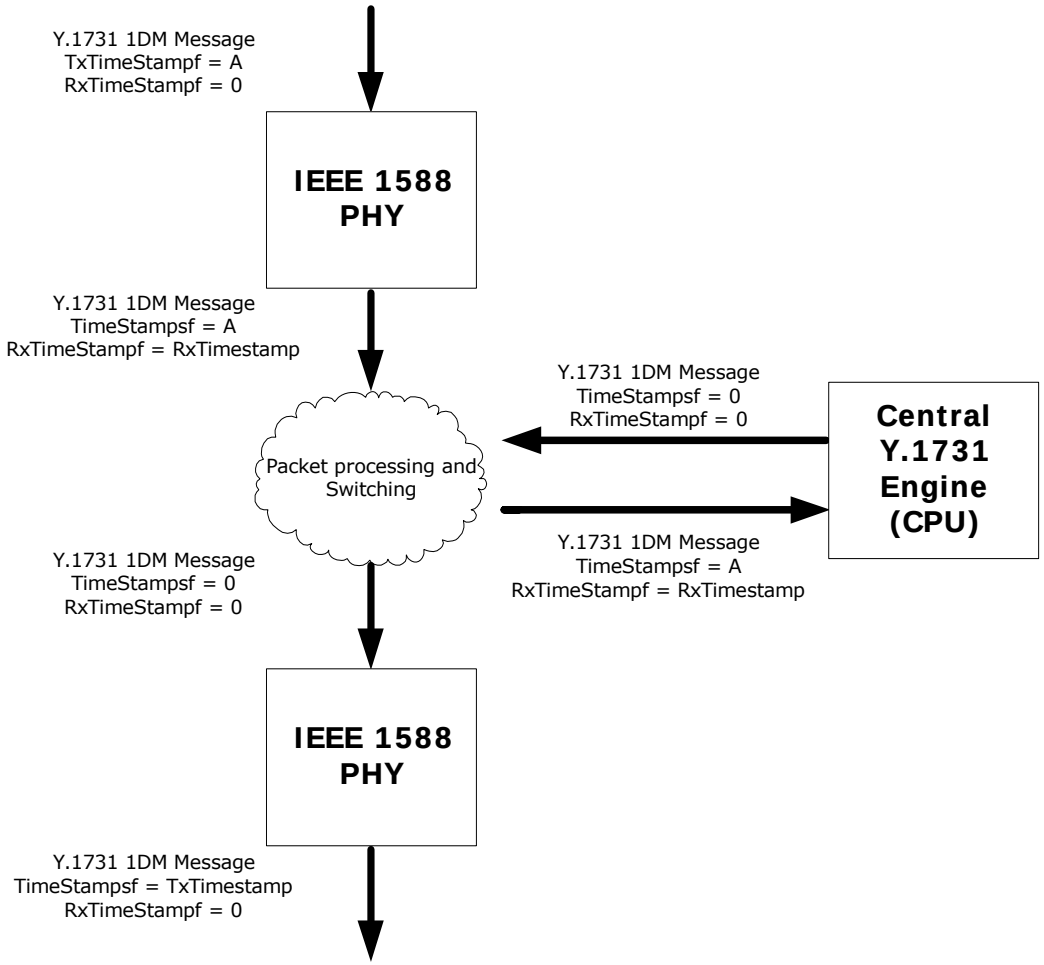
The receiver notes the time of reception of the 1DM frame and calculates the delay.

Figure 37 • **Y.1731 1DM PDU Format**



1. For one-way delay measurements, both MEPs must support IEEE 1588 and be in sync.
2. 1DM frame is generated by the CPU, but with an empty Tx timestamp.
3. The frame is transmitted by the initiating MEP.
4. The 1DM frame is classified as an outgoing 1DM frame by the Egress PHY and the PHY rewrites the frame with the time as TxFCf.
5. The receiving PHY classifies the incoming 1DM frame and writes the receive timestamp in reserved place (RxTimeStampf).
6. The frame is received by the peer MEP.
7. The frame is forwarded to the CPU that can calculate the delay.

Figure 38 • **Y.1731 One-Way Delay**



3.13.11.1 Ingress

If the analyzer detects that the frame is a Y.1731 1DM PDU frame belonging to the MEP, it signals to the timestamp block which action to perform (Write). The analyzer also delivers the write offset and data size (location of the RxTimeStampf location in the frame, 8 bytes wide) to the rewriter.

If the Timestamp block gets the Write action, it puts the value of the active_timestamp register out on the New_field bus to the Rewriter block and the rewriter block adds this timestamp to the reserved bytes in the frame and recalculates FCS.

The following calculation is performed for 1DM frames:

RxTimeStampf = (Raw_Timestamp – Local_correction)

3.13.11.2 Egress

If the analyzer detects that the frame is a Y.1731 1DM PDU frame belonging to the MEP, it signals to the timestamp block which action to perform (Write). It also delivers the write offset and data size (location of the TxTimeStampf location in the frame, 8 bytes wide) to the rewriter.

If the Timestamp block gets the Write action, it puts the value of the active_timestamp register out on the New_field bus to the Rewriter block and the rewriter block adds this timestamp to the reserved bytes in the frame and recalculates FCS.

The following calculation is performed for 1DM frames:

TxTimeStampf = (Raw_Timestamp + Local_correction)

3.13.12 Two-Way Delay Measurements

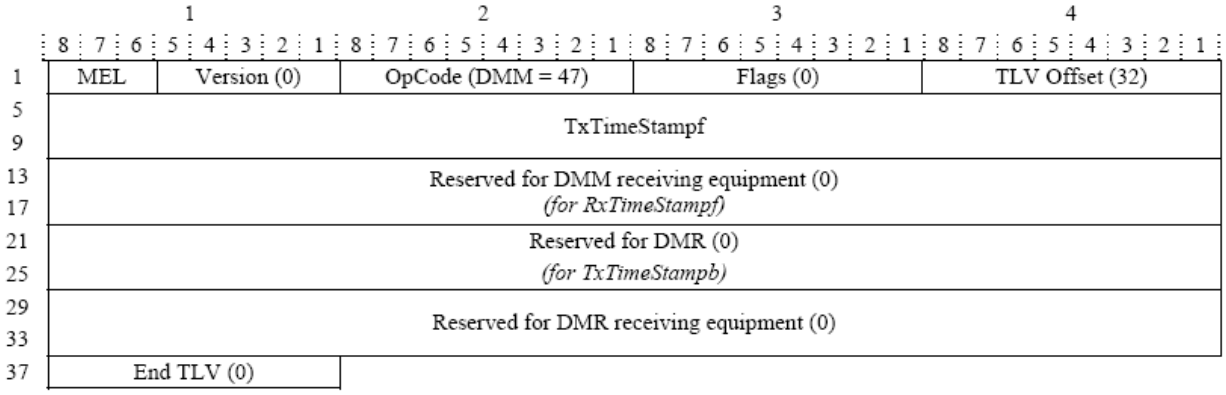
When performing two-way delay measurements, the initiating MEP transmits DMM frames containing a TxTimeStampf value. The receiving MEP replies with a DMR frame that is the same as the DMM frame, but with destination and source MAC address swapped and with a different OAMPDU opcode.

When the DMR frame is received back at the initiating MEP, the time of reception is noted and the total delay is calculated.

As an option, it is allowed to include two additional timestamps in the DMR frame: RxTimeStampf and TxTimeStampb. These contain the time that the DMM page is received for processing and the time the responding DMR reply is sent back, both in IEEE 1588 format.

Including these timestamps allow for exclusion of the processing time in the peer MEP, but it does not require that the two MEPs are synchronized.

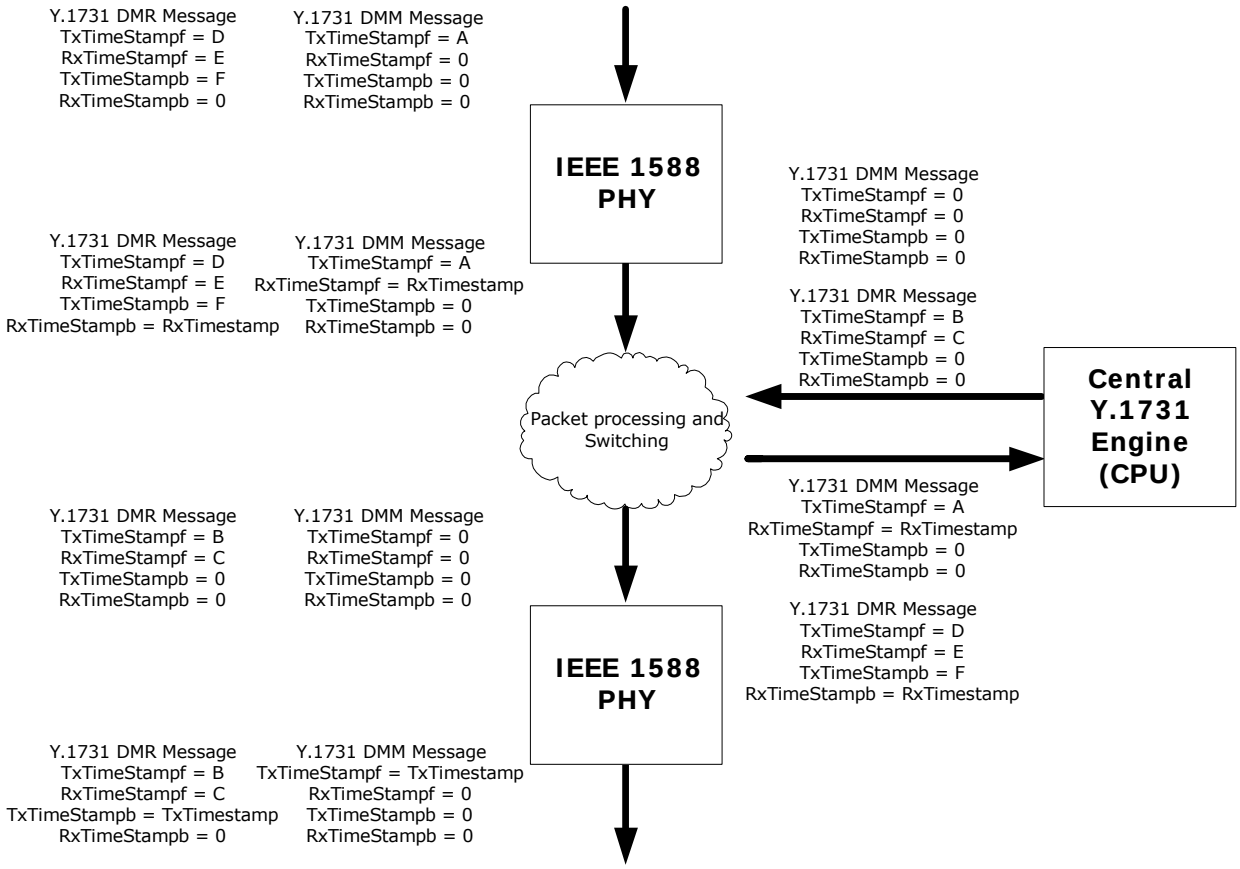
Figure 39 • **Y.1731 DMM PDU Format**



In that case, the following frame flow is needed (two-way delay measurement):

1. DMM frame is generated by the CPU (initiating MEP), but with an empty Tx timestamp.
2. In the egress PHY the DMM frame is classified as an outgoing DMM frame from the MEP and the PHY rewrites the frame with the time as TxTimeStampf.
3. In the ingress PHY the frame is classified as an incoming DMM belonging to the MEP and the RxTimeStampf in the frame is written (the frame has a reserved space for this).
4. The DMM frame is forwarded to the MEP (CPU).
5. The CPU processes the frame (swaps SA/DA MAC addresses, modifies the opcode to DMT) and sends out a DMT frame.
6. The outgoing DMT frame is detected in the egress PHY and the TxTimeStampb is written into the frame.
7. In the ingress PHY the frame is classified as an incoming DMT belonging to the MEP and the RxTimeStampb in the frame in written (the frame has a reserved space for this).
8. The frame is forwarded to the CPU that can calculate the delays.

Figure 40 • **Y.1731 Two-Way Delay**



3.13.12.1 Ingress

If the analyzer detects that the frame is a Y.1731 DMM PDU frame belonging to the MEP, it signals to the timestamp block which action to perform (Write). It also delivers the write offset and data size (location of the RxTimeStampf location in the frame, 8 bytes wide) to the rewriter.

If the analyzer detects that the frame is a Y.1731 DMT PDU frame belonging to the MEP, it signals to the timestamp block which action to perform (Write). It also delivers the write offset and data size (location of the RxTimeStampb location in the frame, 8 bytes wide) to the rewriter.

If the Timestamp block gets the Write action, it puts the value of the active_timestamp register out on the New_field bus to the Rewriter block and the rewriter block adds this timestamp to the reserved bytes in the frame and recalculates FCS.

The following calculations are performed:

- DMM frames: RxTimeStampf = (Raw_Timestamp – Local_correction)
- DMT frames: RxTimeStampb = (Raw_Timestamp – Local_correction)

3.13.12.2 Egress

If the analyzer detects that the frame is a Y.1731 DMM PDU frame belonging to the MEP, it signals to the timestamp block which action to perform (Write). It also delivers the write offset and data size (location of the TxTimeStampf location in the frame, 8 bytes wide) to the rewriter.

If the analyzer detects that the frame is a Y.1731 DMT PDU frame belonging to the MEP, it signals to the timestamp block which action to perform (Write). It also delivers the write offset and data size (location of the TxTimeStampb location in the frame, 8 bytes wide) to the rewriter.

If the Timestamp block gets the Write action, it puts the value of the active_timestamp register out on the New_field bus to the Rewriter block and the rewriter block adds the timestamp to the reserved bytes in the frame and recalculates FCS as follows:

- DMM frames: TxTimeStampf = (Raw_Timestamp + Local_correction)
- DMT frames: TxTimeStampb = (Raw_Timestamp + Local_correction)

3.13.13 IEEE 1588 Device Synchronization

It is important to keep all the Local Clock blocks synchronized to the accurate time over a complete system. To maintain ns accuracy, the signal routing and internal signal delays must be taken into account when configuring a system.

The architecture described in this document assumes that there is a global synchronous clock available in the system. If the system is a telecom system where the system is locked to a PRC, the system clock can be adjusted to match the PRC, meaning that once locked, the frequency of the system clock ensures that the local clocks are progressing (counting) with the accurate frequency. This system clock can be locked to the PRC using 1588, SyncE, SDH, or by other means.

A global timing signal must also be distributed to all the devices. This could be a 1 pps pulse or another slow synchronization pulse, like a 4 kHz synchronization frequency. It can also just be a one-shot pulse. The system CPU can load each local counter with the time value that happens next time the synchronization pulse goes high (+ the known delay of the synchronization pulse traces). It can also just load the same approximate time value into all the local clock blocks (again + the known delay of the synchronization pulse traces) and load them in parallel. Then the local time can be adjusted to match the actual time by adjusting the local clock blocks using the ±1 ns function.

If the Save signal is triggered synchronously on all PHYs of the system, software can read the saved timestamp in each PHY and correct the time accordingly.

If the global system clock is not synchronous, the PPM offset between system clock and the 1588 time progress can be calculated. This PPM offset can be used to calculate how many local-time-clocks is takes to reach a time offset of 1 ns and this value can be programmed into each local time block. The CPU still need to keep track of the smaller PPM offset and adjust the local time blocks with ± writes when necessary.

3.13.14 Timestamp Update

The IEEE1588 block is also called the Time Stamp Update block (TSU) and supports the implementation of IEEE 1588v2 and ITU-T Y.1731 in PHY hardware by providing a mechanism for time-stamp update (PTP) and time-stamping (OAM).

The TSU block works with other blocks to identify PTP/OAM messages, process these messages, and insert accurate timestamp updates/timestamps where necessary. For 1588 timing distribution the VSC8572-02 device supports ordinary clocks, boundary clocks, end-to-end transparent clocks, and peer-to-peer transparent clocks in a chassis based 1588 capable system. One-step and two-step processing is also supported. For details on the timing protocol, refer to IEEE 1588v2. For OAM details refer to ITU-T Y.1731 and G.8113.1/G.8113.2. The TSU block implements part of the functionality required for full 1588 compliance.

The 1588 protocol has four different types of messages that require action by the TSU: Sync, Delay_Req, Pdelay_Req, and Pdelay_Resp. These frames may be encapsulated in other protocols,

several layers deep. The processor is able to detect PTP messages within these other protocols. The supported encapsulations are as follows:

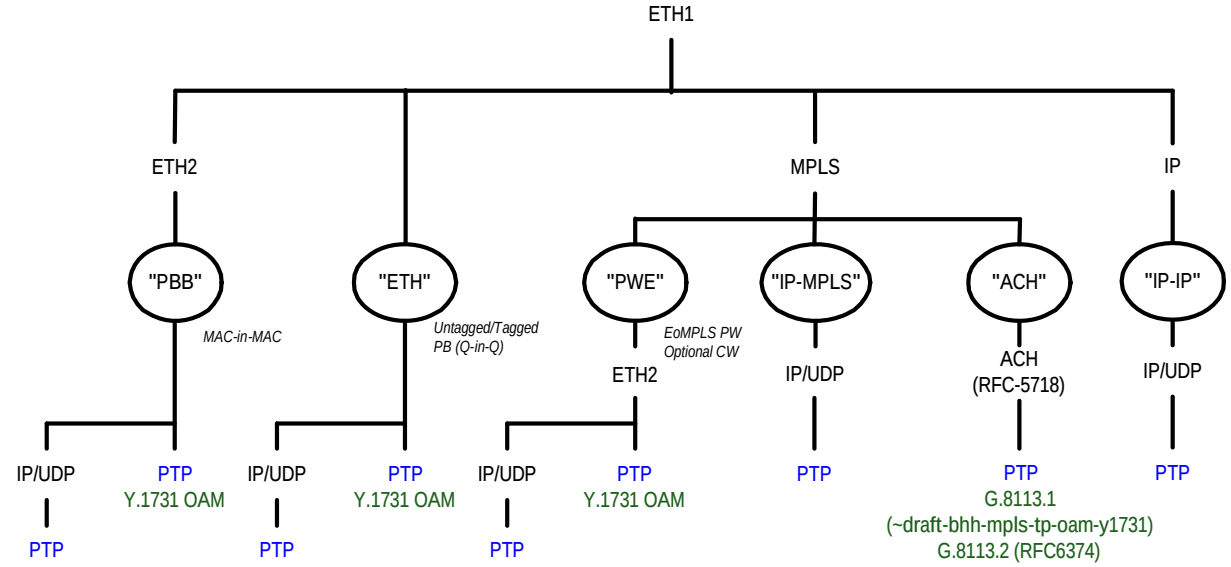
- Ethernet
- UDP over IPv4
- UDP over IPv6
- MPLS
- Pseudo-wires
- PBB and PBB-TE tunnels
- IP/IP tunnel

OAM frames for delay measurement (1DM, DMM, and DMR) with the following supported encapsulations:

- Ethernet (Y.1731 Ethernet OAM)
- Ethernet in MPLS pseudo-wires (Y.1731 Ethernet OAM)
- MPLS-TP (G.8113.1 (~draft-bhh-mpls-tp-oam-y1731) and G.8113.2 (RFC6374))

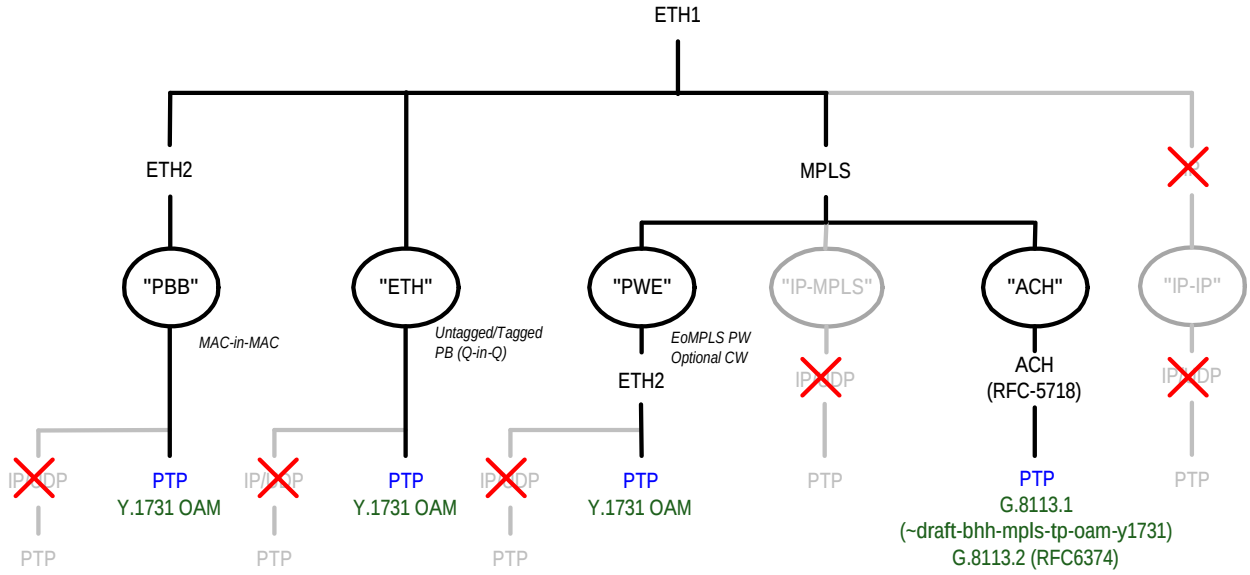
The following illustration shows an overview of the supported PTP encapsulations. Note that the implementation is flexible such that encapsulations not defined here may also be covered.

Figure 41 • **PTP Packet Encapsulations**



The following illustration shows the same overview of the supported encapsulations with the focus on OAM.

Figure 42 • **OAM Packet Encapsulations**



There is one TSU per channel in the VSC8572-02 device. The TSU detects and updates up to three different encapsulations of PTP/OAM. Non-matching frames are transferred transparently. This includes IFG, preamble, and SFD. For all frames there is no bandwidth expansion/shrink.

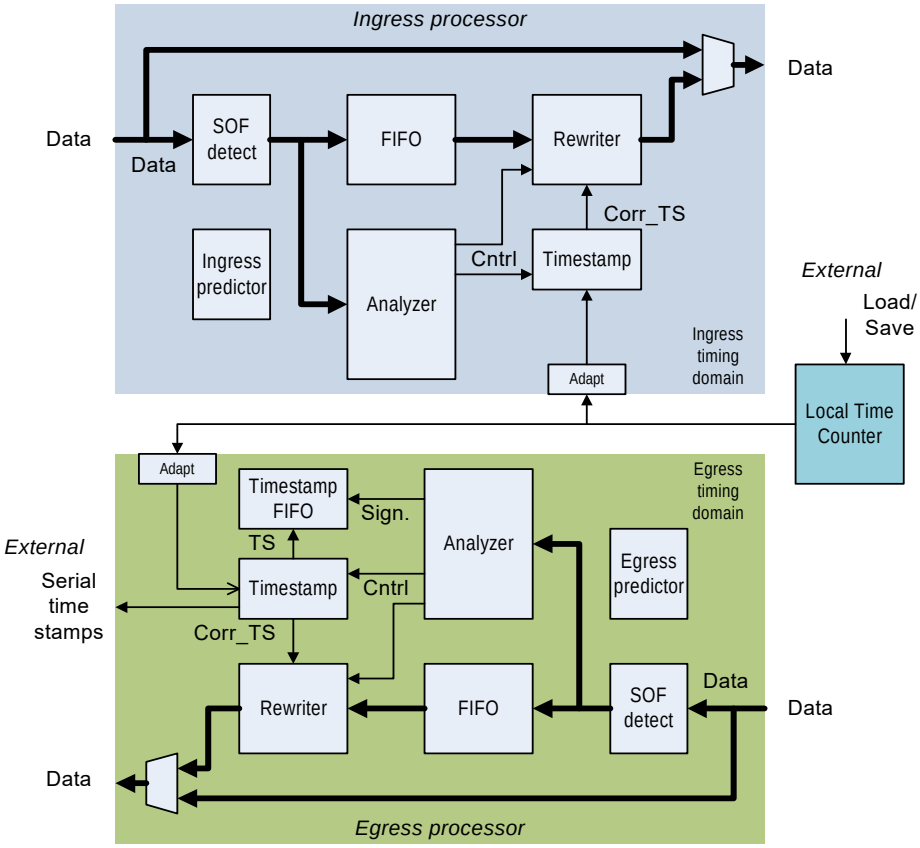
Once these frames are detected in the receive path, they are stamped with the ingress time and forwarded for further PTP/OAM processing. In the transmit path, the correction field of the appropriate PTP message (or the Rx and Tx fields of the OAM frame) is updated with the correct timestamp. A local time counter is maintained to provide the timestamps. Implementation of some of the 1588 protocol requires interaction with the TSU block over the CPU interface and external processing.

The system has an ingress processor, egress processor, and a local time counter. The ingress and egress processing logic blocks are identical except that the timestamp FIFO is only required in the egress direction because the CPU needs to know the actual timestamps of some of the transmitted PTP frames. The CPU reads the timestamps and any associated frame information out of the timestamp FIFO. The FIFO saves the generated timestamps along with information that uniquely identifies the frame to be read out by the CPU.

The ingress and egress processing blocks run on the same clock as the data paths for the corresponding directions. The local time counter is the primary reference clock for the system and it maintains the local reference time used by the TSU logic. It should be synchronized by an external entity. The block provides a method to load and view its value when the 1588_LOAD_SAVE pin is asserted. The local time counter runs at several clock frequencies.

The following illustration shows the block diagram of the TSU.

Figure 43 • **TSU Block Diagram**



In both directions, the input data from the PHY layer is first fed to an SOF detect block. Data is then fed to both the programmable time-delay FIFO and the analyzer. The FIFO delays the data by the time needed to complete the operations necessary to update the PTP frame. That is, the data is delayed to the input of the rewriter so that the rewriter operations are known when the frame arrives. This includes the analyzer and timestamp processor block's functions.

The analyzer block checks the data stream and searches for PTP/OAM frames. When one is detected, it determines the appropriate operations to be performed based on the operating mode and the type of frame detected.

Note: The analyzer blocks of different channels share configuration registers and have identical setups.

The timestamp block waits for an SOF to be detected, captures a timestamp from the local time counter, and builds the new timestamp that is to be written into the PTP/OAM frame. Captured timestamps can be read by the CPU.

The rewriter block handles the actual writing of the new timestamp into the PTP/OAM frame. It is also able to clear parts of the frame such as the UDP checksum, if required, or it can update the frame to ensure that the UDP checksum is correct (for IPv6 PTP frames). The block also calculates the new FCS to be written to the PTP frame after updating the fields with the new timestamp.

The VSC8572-02 device has variable latency in the PCS block. These variations are predicted and used to compensate/maximize the accuracy of the 1588 timestamp logic.

If the time stamp update function is not used the block can be bypassed. When the TSU is bypassed, the block can be configured and then enabled and taken out of bypass mode. The change in bypass mode takes effect only when an IDLE is in the bypass register. This allows the TSU block to be switched on without corrupting data.

Pause frames pass unmodified through the TSU, but the latency may cause a violation of the allowed pause flow-control latency limits per IEEE 802.3.

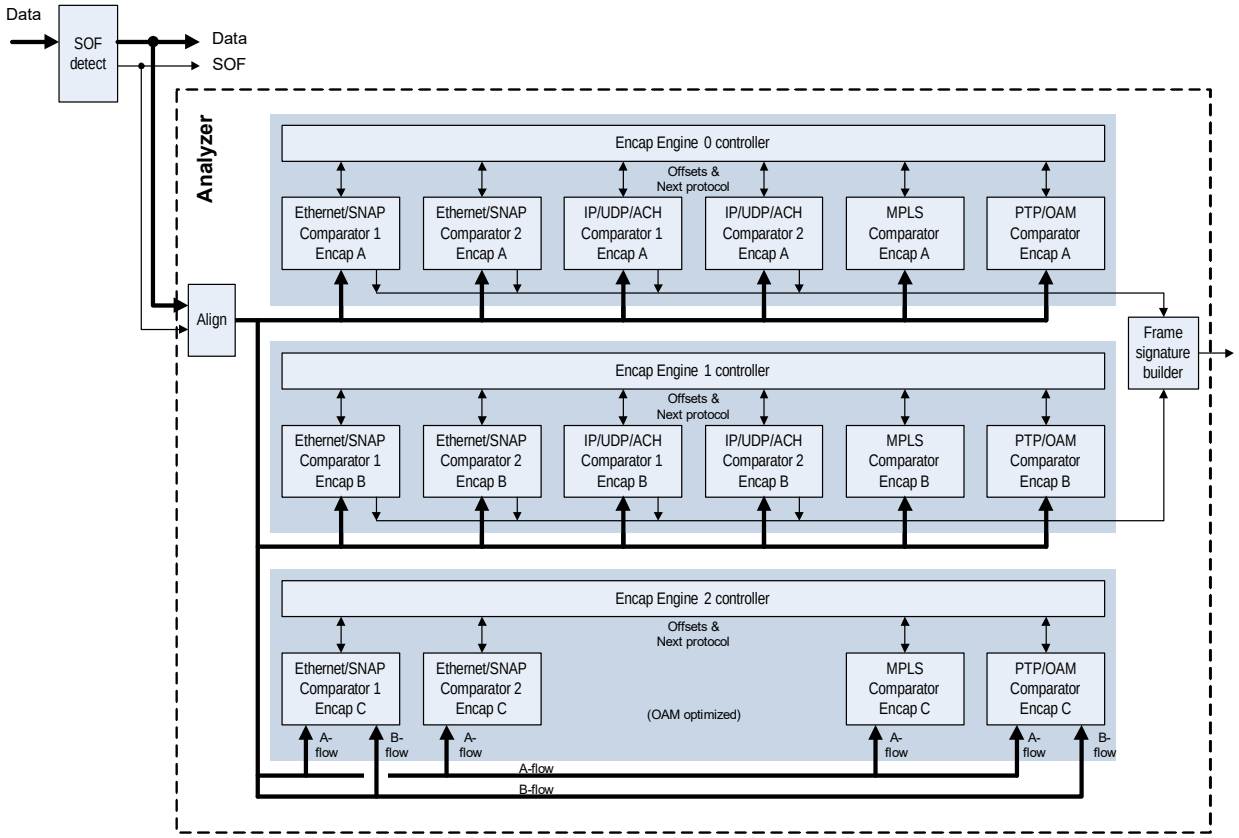
3.13.15 Analyzer

The packet analyzer parses incoming packets looking for PTP/OAM frames. It determines the offset of the correction field within the packet for all PTP frames/for the time stamp in Y.1731 OAM frames. The analyzer has the following characteristics:

- Can compare against two different filter sets plus one optimized for OAM
- Each filter targets PTP or OAM frames
- Flexible comparator sequence with fixed start (Ethernet/SNAP) and end (PTP/OAM) comparator. Configurable intermediate comparators (Ethernet/SNAP, 2x IP/UDP/ACH, and MPLS)

The following illustration shows a block diagram of the analyzer.

Figure 44 • **Analyzer Block Diagram**



The analyzer process is divided into engines and stages. Each engine represents a particular encapsulation stack that must be matched. There are up to six stages in each engine. Each stage uses a comparator block that looks for a particular protocol. The comparison is performed stage-by-stage until the entire frame header has been parsed.

Each engine has its own master enable, so that it can be shut down for major reconfiguration, such as changes in encapsulation order, without stopping traffic. Other enabled engines are not affected.

The SOF detect block searches for the SFD in the preamble and uses that to indicate the SOF position. This information is carried along in the pipeline and also passed to the analyzer.

The first stage of the analyzer is a data path aligner that aligns the first byte of the packet (without the preamble & SFD) to byte 0 of the analyzer data path.

The encapsulation engine handles numerous types of encapsulation stacks. These can be broken down to their individual protocols, and a comparator is defined for each type. The order in which these are applied is configurable. Each comparator outputs a pattern/flow match bit and an offset to the start of the next protocol. The cumulative offset points to the time stamp field.

The sequence in which the protocol comparators are applied is determined by configuration registers associated with each comparator and the transfer of parameters between comparators is controlled by the encapsulation engine controller.

It receives the pattern match and offset information from one comparator stage and feeds the start-of-protocol position to the next comparator. This continues until the entire encapsulation stack has been parsed and always ends with the PTP/OAM stage or until a particular comparator stage cannot find a match in any of its flows. If at any point along the way no valid match is found in a particular stage, the analyzer sends the NOP communication to the timestamp block indicating that this frame does not need modification and that it should discard its timestamp.

There are two types of engines in the analyzer, one optimized for PTP frames and the other optimized for OAM frames. The two engine types are mostly identical except that the IP comparators are removed from the OAM engines. The following table shows the comparator layout per engine type and the number of flows in each comparator. There are two PTP engines and one OAM engine in each analyzer. Additional differences in the Ethernet and MPLS blocks are defined in their respective sections. For more information, see Ethernet/SNAP/LLC Comparator, page 49 and MPLS Comparator, page 53.

Table 6 • **Flows Per Engine Type**

	Number of Flows	
Comparator	**PTP Engine**	**OAM Engine**
Ethernet 1	8	8
Ethernet 2	8	8
MPLS	8	8
IP/ACH 1	8	0
IP/ACH 2	8	0
PTP/OAM	6	6

Each comparator stage has an offset register that points to the beginning of the next protocol relative to the start of the current one. The offset is in bytes, and the first byte of the current protocol counts as byte 0. As an example, the offset register for a stage would be programmed to 10 when the header to match is 10 byte long. With the exception of the MPLS stage (offsets are automatically calculated in that stage), it is the responsibility of the programmer to determine the value to put in these registers. This value must be calculated based upon the expected length of the header and is not expected to change from frame-to-frame when matching a given flow.

Table 7 • **Ethernet Comparator: Next Protocol**

Parameter	Width	Description
Encap_Engine_ENA	1 bit	For each encapsulation engine and enable bit that turns the engine on or off. The engine enables and disables either during IDLE (all 8 bytes must be IDLE) or at the end of a frame. If the enable bit is changed during the middle of a frame, the engine will wait until it sees either of those conditions before turning on or off.
Encap_Flow_Mode	1 bit	There is a separate bit for each engine. For each encapsulation engine: 1 = Strict flow matching, a valid frame must use the same flow IDs in all comparators in the engine except the PTP and MPLS comparators. 0 = A valid frame may match any enabled flow in all comparators If more than one encapsulation produces a match, the analyzer sends NOP to the rewriter and sets a sticky bit.

The following table shows the ID codes comparators use in the sequencing registers. The PTP packet target encapsulations require only up to five comparators.

Table 8 • **Comparator ID Codes**

ID	Name	Sequence
0	Ethernet Comparator 1	Must be the first
1	Ethernet Comparator 2	Intermediate
2	IP/UDP/ACH Comparator 1	Intermediate
3	IP/UDP/ACH Comparator 2	Intermediate
4	MPLS Comparator	Intermediate
5	PTP/OAM Comparator	Must be the last

The following sections describe the comparators. The frame format of each comparator type is described first, followed by match/mask parameter definition. All upper and lower bound ranges are inclusive and all match/mask registers work the same way. If the corresponding mask bit is 1, then the match bit is compared to the incoming frame. If a mask bit is 0, then the corresponding match bit is ignored (a wildcard).

3.13.15.1 Ethernet/SNAP/LLC Comparator

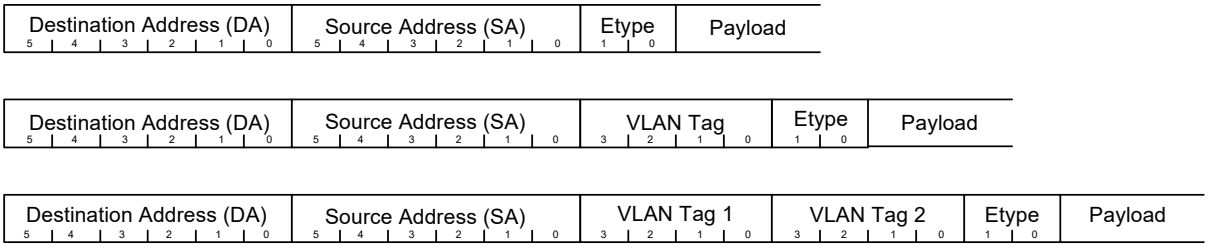
There are two such comparators in each engine. The first stage of each engine is always an Ethernet/SNAP/LLC comparator. The other comparator can be configured to be at any point in the chain.

Ethernet frames can have multiple formats. Frames that have an actual length value in the ether-type field (Ethernet type I) can have one of three formats: Ethernet with an EtherType (Ethernet type II), Ethernet with LLC, or Ethernet with LLC & SNAP. Each of these formats can be compounded by having one or two VLAN tags.

TYPE II ETHERNET

Type II Ethernet is the most common and basic type of Ethernet frame. The Length/EtherType field contains an EtherType value and either 0, 1, or 2 VLAN tags. Both VLAN can be of type S/C (with EtherType 0x8a88/0x8100). The payload would be the start of the next protocol.

Figure 45 • **Type II Ethernet Basic Frame Format**

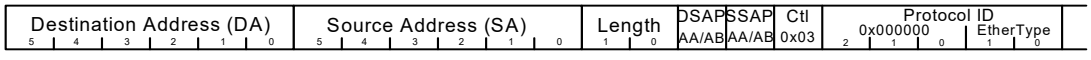


Ethernet with LLC and SNAP

If an Ethernet frame with LLC contains a SNAP header, it always follows a three-octet LLC header. The LLC values for DSAP & SSAP are either 0xAA or 0xAB and the control field contains 0x03. The SNAP header is five octets long and consists of two fields, the 3-octet OUI value and the 2-octet EtherType. As with the other types of Ethernet frames, this format can have 0, 1, or 2 VLAN tags. The OUI portion of the SNAP header is hard configured to be 0 or 0xf8.

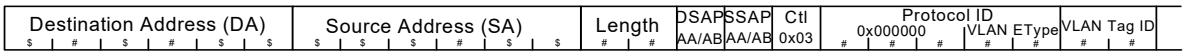
The following illustration shows an Ethernet frame with a length in the Length/EtherType field, an LLC header, and a SNAP header.

Figure 46 • **Ethernet Frame with SNAP**



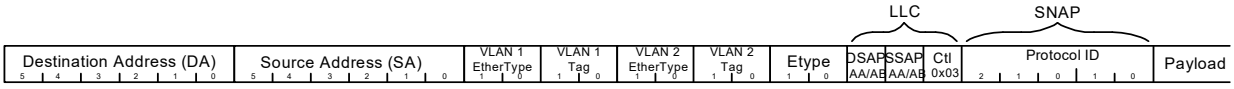
The following illustration shows an Ethernet frame with an LLC/SNAP header and a VLAN tag in the SNAP header. The Ethertype in the SNAP header is the VLAN identifier and tag immediately follows the SNAP header.

Figure 47 • **Ethernet Frame with VLAN Tag and SNAP**



The following illustration shows the longest form of the Ethernet frame header that needs to be supported: two VLAN tags, an LLC header, and a SNAP header.

Figure 48 • **Ethernet Frame with VLAN Tags and SNAP**



PBB (PROVIDER BACKBONE BRIDGING) SUPPORT

The provider backbone bridging protocol is supported using two Ethernet comparator blocks back-to-back. The first portion of the frame has a type II Ethernet frame with either 0 or 1 VLAN tags followed by an I-tag. The following illustrations show two examples of the PBB Ethernet frame format.

Figure 49 • **PBB Ethernet Frame Format (No B-Tag)**

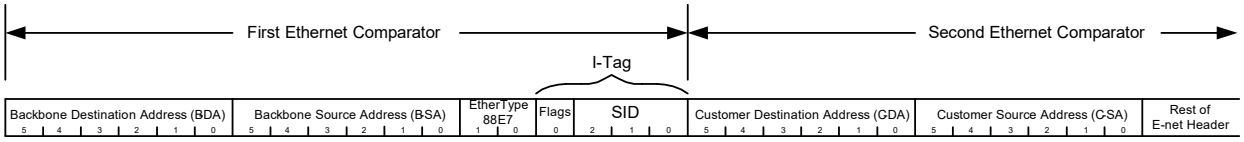
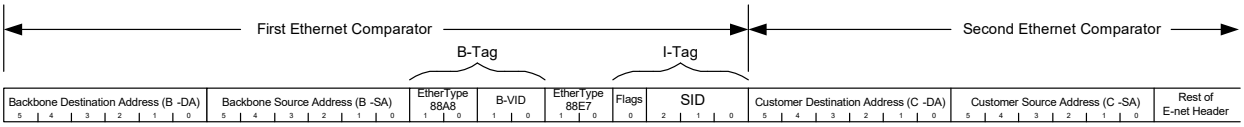


Figure 50 • **PBB Ethernet Frame Format (1 B-Tag)**



Ethernet Comparison

The Ethernet comparator block has two forms of comparison, as follows:

- Next protocol comparison is common for all flows in the comparator. It is the single set of registers and is used to verify what the next protocol in the encapsulated stack will be.
- Flow comparison is used to match any of the possible flows within the comparator.

Ethernet Next Protocol Comparison

The next protocol comparison field looks at the last EtherType field in the header (there can be multiple in the header) to verify the next protocol. It may also look at VLAN tags and the EtherType field when it is used as a length. Each has a pattern match/mask or range, and an offset.

The following table lists the next protocol parameters for the Ethernet comparator.

Table 9 • **Ethernet Comparator (Next Protocol)**

Parameter	Width	Description
Eth_Nxt_Comparator	3 bit	Pointer to the next comparator.
Eth_Frame_Sig_Offset	5 bit	Points to the start of the field used to build the frame signature.
Eth_VLAN-TPID_CFG	16 bit	Globally defines the value of the TPID for an S-tag, B-tag, or any other tag type other than a C-tag or I-tag.

Table 9 • **Ethernet Comparator (Next Protocol)**

Parameter	Width	Description
Eth_PBB_ENA	1 bit	Configures if the packet carries PBB or not. This configuration bit is only present in the first Ethernet comparator block. PBB is disabled in Ethernet comparator block 2.
Eth_Etype_Match_Enable	1 bit	Configures if the Ethertype field match register is used or not. Only valid when the packet is a type II Ethernet packet.
Eth_Etype_Match	16 bit	If the packet is a type II Ethernet packet and Eth_Etype_Match_Enable is a 1, the Ethertype field in the packet is compared against this value.

Ethernet Flow Comparison

The Ethernet flow is determined by looking at VLAN tags and either the source address (SA) or the destination address (DA). There are a configurable number of these matched sets. The following table lists the flow parameters for the Ethernet comparator.

Table 10 • **Ethernet Comparator (Flow)**

Parameter	Width	Description
Eth_Flow_Enable	1 bit/flow	0 = Flow disabled 1 = Flow enabled
Eth_Channel_Mask	1 bit/channel/flow	0 = Do not use this flow match group for this channel 1 = Use this flow match group for this channel
Eth_VLAN_Tags	2 bit	Configures the number of VLAN tags in the frame (0, 1, or 2)
Eth_VLAN_Tag1_Type	1 bit	Configures the VLAN tag type for VLAN tag 1 If PBB is not enabled: 0 = C-tag, value of 0x8100 1 = S-tag, match to the value in CONF_VLAN_TPID (global for all ports/directions) If PBB enabled: 0 = S-tag (or B-tag), to the value in CONF_VLAN_TPID (global for all ports/directions) There must be 2 VLAN tags, 1 S-tag and one I-tag 1 = I-tag
Eth_VLAN_Tag2_Type	1 bit	Configures the VLAN tag type for VLAN tag 2 If PBB is not enabled: 0 = C-tag, value of 0x8100 1 = S-tag, match to the value in CONF_VLAN_TPID (global for all ports/directions) If PBB enabled: The second tag is always an I-tag and this register control bit is not used. The second tag in PBB is always an I-tag.

Table 10 • **Ethernet Comparator (Flow)**

Parameter	Width	Description
Eth_Ethertype_Mode	1 bit	0 = Only type 2 Ethernet frames supported, no SNAP/LLC expected 1 = Type 1 & 2 Ethernet packets supported. Logic looks at the Ethertype/length field to determine the packet type. If the field is a length (less than 0x0600), then the packet is a type 1 packet and MUST include a SNAP & 3-byte LLC header. If the field is not a length, it is assumed to be an Ethertype and SNAP/LLC must not be present
Eth_VLAN_Verify_Ena	1 bit	0 = Parse for presence of VLAN tags but do not check the values. For PBB mode, the I-tag is still always checked. 1 = Verify the VLAN tag configuration including number and value of the tags.
Eth_VLAN_Tag_Mode	2 bit	0 = No range checking on either VLAN tag 1 = Range checking on VLAN tag 1 2 = Range checking on VLAN tag 2
Eth_Addr_Match	48 bit	Matches an address field selected by Eth_Addr_Match_Mode
Eth_Addr_Match_Select	2 bit	Selects the address to match 0 = Match the destination address 1 = Match the source address 2 = Match either the source or destination address 3 = Reserved, do not use
Eth_Addr_Match_Mode	3 bits per flow	Selects the address match mode. One or multiple bits can be set in this mode register allowing any combination of match types. For unicast or multicast modes, only the MSB of the address field is checked (0 = unicast; 1 = multicast). See section 3.2.3.1 of 802.3 for more details. 0 = Match the full 48-bit address 1 = Match any unicast address 2 = Match any multicast address
Eth_VLAN_Tag1_Match	12 bit	Match field for the first VLAN tag (if configured to be present).
Eth_VLAN_Tag1_Mask	12 bit	Mask for the first VLAN tag. If a match set is not used, set this register to all 0s.
Eth_VLAN_Tag2_Match	12 bit	Match field for the update VLAN tag (if configured to be present).
Eth_VLAN_Tag2_Mask	12 bit	Mask for the second VLAN tag. If a match set is not used, set this register to all 0s.
Eth_VLAN_Tag_Range_Upper	12 bit	Upper limit of the range for one of the VLAN fields selected by ETH_VLAN_TAG_MODE register. If PBB mode is enabled, this register is not used for range checking but rather is the upper 12 bit of the I-tag.
Eth_VLAN_Tag_Range_Lower	12 bit	Lower limit of the range for one of the VLAN fields selected by ETH_VLAN_TAG_MODE register. If PBB mode is enabled, this register is not used for range checking but rather is the lower 12 bit of the I-tag SID.

Table 10 • **Ethernet Comparator (Flow)**

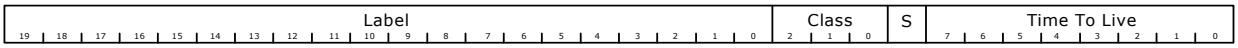
Parameter	Width	Description
Eth_Nxt_Prot_Grp_Sel	1 bit	Per flow, maps a particular flow to a next-protocol group register set. This register only appears in the Ethernet block in the OAM-optimized engine.

If the Ethernet block is part of the OAM optimized engine, there are two sets of next-protocol configuration registers. Both sets are identical except one has an _A suffix and the other has a _B suffix. In the per-flow registers an additional register, ETH_NXT_PROT_SEL, is included to map a particular flow with a set of next protocol register set. This function allows the Ethernet block within the OAM-optimized engine to act like two separate engines with a configurable number of flows assignable to each with a total maximum number of eight flows. It effectively allows two separate protocol encapsulation stacks to be handled within the engine.

3.13.15.2 MPLS Comparator

The MPLS comparator block counts MPLS labels to find the start of the next protocol. The MPLS header can have anywhere from 1 to 4 labels. Each label is 32 bit long and has the format shown in the following illustration.

Figure 51 • **MPLS Label Format**

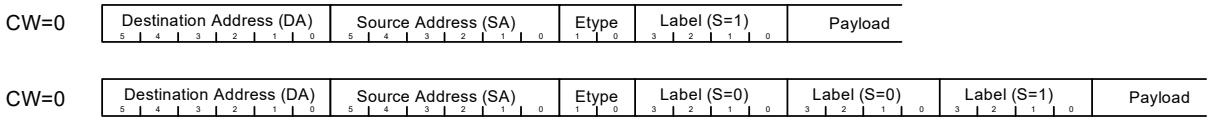


The S bit is used to indicate the last label in the stack, as follows: If S = 0, then there is another label. If S = 1, then this is the last label in the stack.

Also, the MPLS stack can optionally be followed by a control word (CW). This is configurable per flow.

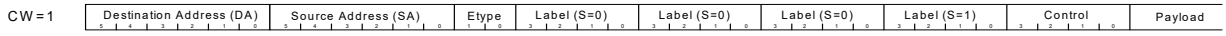
The following illustration shows a simple Ethernet packet with either one label or three labels and no control word.

Figure 52 • **MPLS Label Stack within an Ethernet Frame**



The following illustration shows an Ethernet frame with four labels and a control word. Keep in mind that this comparator is used to compare the MPLS labels and control words; the Ethernet portion is checked in the first stage.

Figure 53 • **MPLS Labels and Control Word**



There could be VLAN tags between the SA and the Etype fields and, potentially, an LLC and SNAP header before the MPLS stack, but these would be handled in the Ethernet/LLC/SNAP comparator.

The only configuration registers that apply to all flows within the comparator are the match_mode register and the nxt_comparator register. The match mode register determines how the match filters are used and there is one per stage. Each flow has it own complete set of match registers.

Table 11 • **MPLS Comparator: Next Word**

Parameter	Width	Description
MPLS_Nxt_Comparator	3 bit	Pointer to the next comparator

Table 12 • **MPLS Comparator: Per-Flow**

Parameter	Width	Description
MPLS_Flow_Enable	1 bit per flow	0 = Flow disabled 1 = Flow enabled
MPLS_Channel_Mask	1 bit per channel per flow	0 = Do not use this flow match group for this channel 1 = Use this flow match group for this channel
MPLS_Ctl_Word	1 bit	Indicates if there is a 32-bit control word after the last label. This should only be set if the control word is not expected to be an ACH header. ACH headers are checked in the IP block. If the control word is a non-ACH control word, only the upper 4 bits of the control are checked and are expected to be 0. 0 = There is no control word after the last label 1 = There is expected to be a control word after the last label
MPLS_REF_PNT	1 bit	The MPLS comparator implements a searching algorithm to properly parse the MPLS header. The search can be performed from either the top of the stack or the end of the stack. 0 = All searching is performed starting from the top of the stack 1 = All searching if performed from the end of the stack
MPLS_STACK_DEPTH	4 bit	Each bit represents a possible stack depth, as shown in the following list. MPLS_STACK_DEPTH Bit — Allowed Stack Depth 0 — 1 1 — 2 2 — 3 3 — 4

Table 13 • **MPLS Range_Upper/Lower Label Map**

Parameter	MPLS_REF_PNT = 0, top-of-stack referenced	MPLS_REF_PNT=1, end-of-stack referenced
MPLS_Range_Upper/Lower_0	Top label	Third label before the end label
MPLS_Range_Upper/Lower_1	First label after the top label	Second label before the end label
MPLS_Range_Upper/Lower_2	Second label after the top label	First label before the end label
MPLS_Range_Upper/Lower_3	Third label after the top label	End label

The offset to the next protocol is calculated automatically. It is based upon the number of labels found and whether a control word is configured to be present. It points to the first octet after the last label or after the control word, if present.

Table 14 • **Next MPLS Comparator**

Parameter	Width	Description
MPLS_Range_Lower	20 bit × 4 labels	Lower value of the label range when range checking is enabled
MPLS_Range_Upper	20 bit × 4 labels	Upper value of the label range when range checking is enabled

If an exact label match is desired, set the upper and lower range values to the same value. If a label value is a don't care, then set the upper value to the maximum value and the lower value to 0.

The MPLS comparator block used in the OAM-optimized engine differs from the one used in the PTP-optimized engine.

Just like the Ethernet comparator block, there are two sets of next protocol blocks along with a next protocol association configuration field per-flow. This allows two different encapsulations to occur in a single engine.

Table 15 • **Next-Protocol Registers in OAM-Version of MPLS Block**

Parameter	Width	Description
MPLS_Nxt_Prot_Grp_Sel	1 bit per flow	Maps each flow to next-protocol-register set A or B

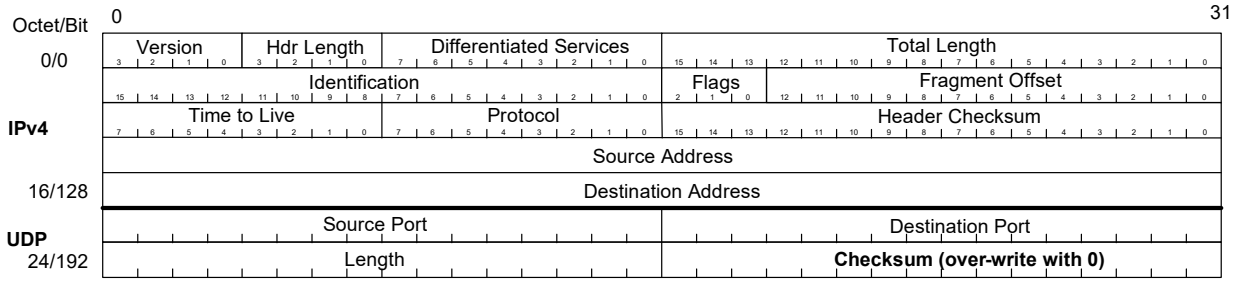
3.13.15.3 IP/UDP/ACH Comparator

The IP/UDP/ACH comparator is used to verify one of three possible formats, IPv4, IPV6, and ACH. Additionally, IPv4 and IPv6 can also have a UDP header after the IP header. There are two of these comparators and they can operate at stages 2, 3, or 4 of the analyzer pipeline. Note that if there is an IP-in-IP encapsulation, a UDP header will only exist with the inner encapsulation.

3.13.15.4 IPv4 Header Format

The following illustration shows an IPv4 frame header followed immediately by a UDP header. IPv4 does not always have the UDP header, but the comparator is designed to work with or without it. The Header Length field is used to verify the offset to the next protocol. It is a count of 32-bit words and does not include the UDP header. If the IPv4 frame contains a UDP header, the Source and Destination ports are also checked. These values are the same for all flows within the comparator. Note that IPv4 options, extended headers, and UDP fragments are not supported.

Figure 54 • **IPv4 with UDP**



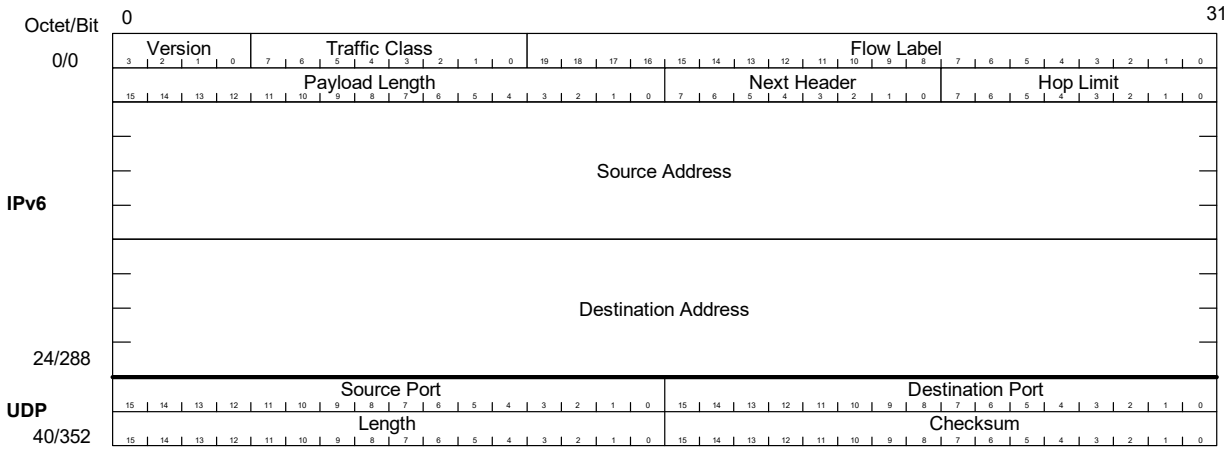
Note: Checksum over-write with 0 occurs on ingress only. PTP applications that generate 1588 frames with this format are responsible for creating IPv4/UDP frames with a zeroed checksum upon generation from the application.

Per flow validation is performed on the Source or Destination Address in the IPv4 header. The comparator can be configured to indicate a match in the flow if the source, destination, or either the source or destination fields match.

3.13.15.5 IPv6 Header Format

The following illustration shows an IPv6 frame header followed immediately by a UDP header. IPv6 does not always have the UDP header, but the comparator is designed to work with or without it. The Next Header field is used to verify the offset to the next protocol. It is a count of 32-bit words and does not include the UDP header. If the IPv6 frame contains a UDP header, the Source and Destination ports are also checked. These values are the same for all flows within the comparator.

Figure 55 • **IPv6 with UDP**



Per flow validation is performed on the Source or Destination Address in the IPv6 header. The comparator can be configured to indicate a match in the flow if the source, destination, or either the source or destination fields match.

If the IPv6 frame is the inner most IP protocol, then the checksum field must be valid. This is accomplished using a pair of pad bytes after the PTP frame. The checksum is computed using one's compliment of the one's compliment sum of the IPv6 header, UDP header, and payload including the pad bytes. If any of the fields in the frame are updated, the pad byte field at the end of the frame will be updated by the PHY so that the checksum field does not have to be modified.

Note: IPv6 extension headers are not supported.

3.13.15.6 ACH Header Format

The following illustrations show ACH headers. They can appear after a MPLS label stack in place of the control word. ACH is verified as a protocol only. There are no flows within the protocol for ACH. The ACH header can optionally have a Protocol ID field. The protocol is verified using the Version, Channel type, and optional Protocol ID field.

Figure 56 • **ACH Header Format**

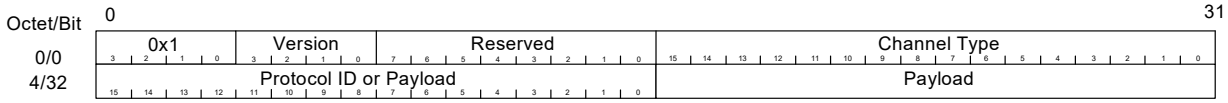
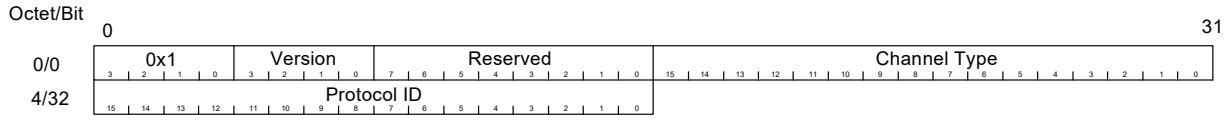


Figure 57 • **ACH Header with Protocol ID Field**

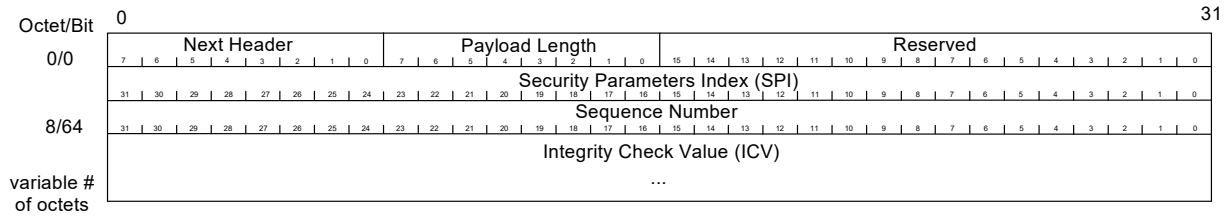


3.13.15.7 IPSec

IPSec adds security to the IP frame using an Integrity Check Value (ICV), a variable-length checksum that is encoded with a special key. The key value is known by the sender and the receiver, but not any of the devices in between. A frame must have a correct ICV to be valid. The sequence number field is a continuously incrementing value that is used to prevent replay attacks (resending a known good frame).

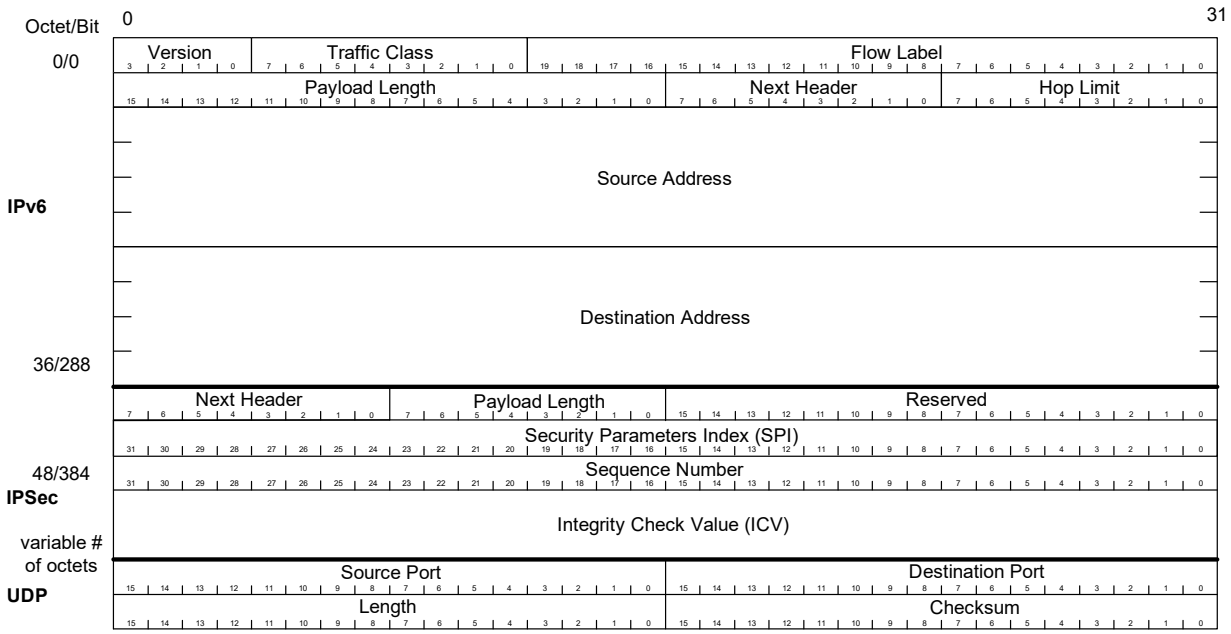
Little can be done with frames when IPSec is used because the 1588 block cannot recalculate the ICV and the frame cannot be modified on egress. Therefore, one-step processing cannot be performed, only two-step processing can be done. The only task here is to verify the presence of the protocol header. Stored timestamps in the TS FIFO are used to create follow-up messages. On ingress, the timestamp The following illustration shows the format of the IPSec frame. It normally appears between the IP header (IPv4 or IPv6) and the UDP header or at the start of the payload.

Figure 58 • **IPSec Header Format**



There is only one set of match/mask registers associated with IPSec and they are used to verify the presence of the IPSec header. The following illustration shows the largest possible IP frame header with IPv6, IPSec, and UDP.

Figure 59 • **IPv6 with UDP and IPSec**



3.13.15.8 Comparator Field Summary

The following table shows a summary of the fields and widths to verify IPv4, IPv6, and ACH protocols.

Table 16 • **Comparator Field Summary**

Protocol	Next Protocol Fields	NPF Bit Widths	Flow Fields	Flow Bit Widths
IPv4	Header length	One 4-bit field	Source/ Destination Address	One 32-bit field
	UDP Source/Destination Port	One 32-bit field		
IPv6	Next header	One 8-bit field	Source/ Destination Address	One 128-bit field
	UDP Source/Destination Port	One 32-bit field		
ACH	Entire ACH header	One 64-bit field		
IPSec	Next Header/Payload Length/ SPI	One 64-bit field		

3.13.15.8.1 IP/ACH Comparator Next Protocol

The following table shows the registers used to verify the current header protocol and the next protocol. They are universal and cover IPv4, IPv6, and ACH. They can also be used to verify other future protocols.

Table 17 • **IP/ACH Next-Protocol Comparison**

Parameter	Width	Description
IP_Mode	2 bit	Specifies the mode of the comparator. If IPv4 or IPv6 is selected, the version field is automatically checked to be either 4 or 6 respectively. If another protocol mode is selected, then the version field is not automatically checked. In IPv4, the fragment offset field must be 0, and the MF flag bit (LSB of the flag field) must be 0. 0 = IPv4 1 = IPv6 2 = Other protocol, 32-bit address match 3 = Other protocol, 128-bit address match
IP_Prot_Match_1	8 bit	Match bit for Protocol field in IPv4 or next header field in IPv6
IP_Prot_Mask_1	8 bit	Mask bits for IP_Prot_Match_1. For each bit, if it is a 1, the corresponding match bit is valid. If it is 0, the corresponding match bit is ignored. Disable this match/mask set by setting the mask register to all 0's.
IP_Prot_Offset_1	5 bit	Indicates the starting position relative to the beginning of the IP frame header to start matching for the match/mask 1 register pair.
IP_Prot_Match_2	64 bit	Match bits for the IPSec header or any other desired field. For ACH, this register should be used to match the ACH header.
IP_Prot_Mask_2	64 bit	Mask bits for IP_Prot_Match_2. For each bit, if it is a 1, the corresponding match bit is valid. If it is 0, the corresponding match bit is ignored. Disable this match/mask set by setting the mask register to all 0's.
IP_Prot_Offset_2	7 bit	Indicates the starting position relative to the beginning of the IP frame header to start matching for the match/mask two-register pair.
IP_Nxt_Protocol	8 bit	Points to the start of the next protocol relative to the beginning of this header. It is the responsibility of the programmer to determine this offset, it is not calculated automatically. Each flow within an encapsulation engine must have the same encapsulation order and each header must be the same length. This field is current protocol header length in bytes.
IP_Nxt_Comparator	3 bit	Pointer to the next comparator. 0 = Reserved 1 = Ethernet comparator 2 2 = IP/UDP/ACH comparator 1 3 = IP/UDP/ACH comparator 2 4 = Reserved 5 = PTP/OAM comparator 6,7 = Reserved
IP_Flow_Offset	5 bit	Indicates the starting position relative to the beginning of the IP frame header to start matching for the flow match/mask register pair. When used with IPv4 or 6, this will point to the first byte of the source address. When used with a protocol other that IPv4 or 6, this register points to the beginning of the field that will be used for flow matching.

Table 17 • **IP/ACH Next-Protocol Comparison**

Parameter	Width	Description
IP_UDP_Checksum_Clear_Ena	1 bit	If set, the 2-byte UDP checksum should be cleared (written with zeroes). This would only be used for UDP in IPv4.
IP_UDP_Checksum_Update_Ena	1 bit	If set, the last two bytes in the UDP frame must be updated to reflect changes in the PTP or OAM frame. This is necessary to preserve the validity of the IPv6 UDP checksum. Note that IP_UDP_Checksum_Clear_Ena & IP_UDP_Checksum_Update_Ena should never be set at the same time.
IP_UDP_Checksum_Offset	8 bit	This configuration field is only used if the protocol is IPv4. This register points to the location of the UDP checksum relative to the start of this header. This info is used later by the PTP/Y.1731 block to inform the rewriter of the location of the checksum in a UDP frame. This is normally right after the Log Message Interval field.
IP_UDP_Checksum_Width	2 bit	Specifies the length of the UDP checksum in bytes (normally 2 bytes)

The IP/ACH Comparator Flow Verification registers are used to verify the current frame against a particular flow within the engine. When this engine is used to verify IPv4 or IPv6 protocol, the flow is verified using either the source or destination address in the frame.

If the protocol is something other than IPv4 or IPv6, then the flow match can be used to match either a 32 or 128 bit field pointed to by the IP_Flow_Offset register. Mask bits can be used to shorten the length of the match, but there is no concept of source or destination address in this mode.

Table 18 • **IP/ACH Comparator Flow Verification Registers**

Parameter	Width	Description
IP_Flow_Ena	1 bit per flow	0 = Flow disabled 1 = Flow enabled
IP_Flow_Match_Mode	2 bit per flow	This register is only valid when the comparator block is configured to match on IPv4 or IPv6. It allows the match to be performed on the source address, destination address, or either address. 0 = Match on the source address 1 = Match on the destination address 2 = Match on either the source or the destination address
IP_Flow_Match	128 bit	Match bits for source & destination address in IPv4 & 6. Also used as the flow match for protocols other than IPv4 or 6. When used with IPv4, only the upper 32 bits are used and the remaining bits are not used.
IP_Flow_Mask	128 bit	Mask bits for IP_Flow_Match. For each bit, if it is a 1, the corresponding match bit is valid. If it is 0, the corresponding match bit is ignored.
IP_Channel_Mask	1 bit per channel per flow	Enable for this match set for this channel

Table 18 • **IP/ACH Comparator Flow Verification Registers**

Parameter	Width	Description
IP_Frame_Sig_Offset	5 bit	Points to the start of the field that will be used to build the frame signature. This register is only present in comparators where frame signature is supported. In other words, if there is no frame signature FIFO in a particular direction, this register will be removed.

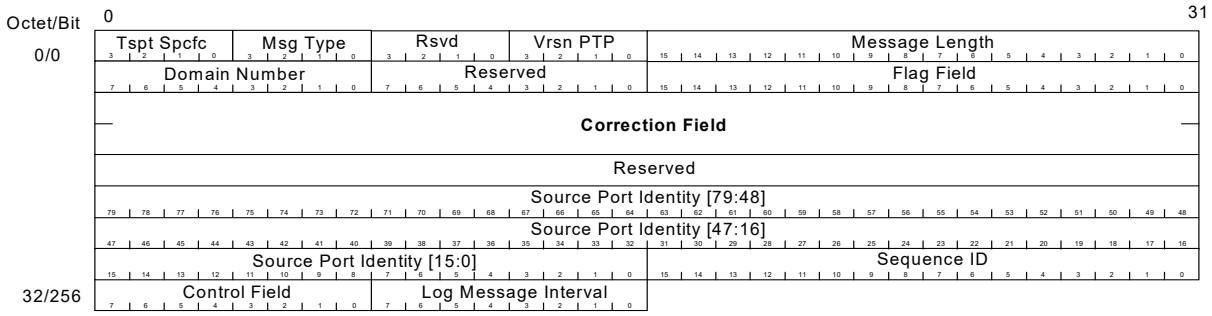
3.13.15.9 PT/Y.1731 OAM Comparator

The PTP/OAM comparator is always the last stage in the analyzer for each encapsulation engine. It can validate IEEE 1588 PTP frames or Y.1731 OAM frames.

3.13.15.10PTP Frame Header

The following illustration shows the header of a PTP frame.

Figure 60 • **PTP Frame Layout**

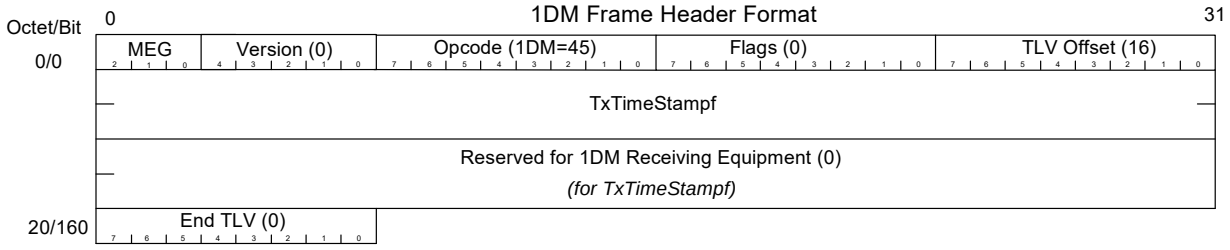


Unlike most of the other stages, there is no protocol validation for PTP frames; only interpretation of the header to determine what action to take. The first eight bytes of the header are used to determine the action to be taken. These match fields in the flow comparison registers with a corresponding set of command registers for each flow.

3.13.15.11Y.1731 OAM Frame Header

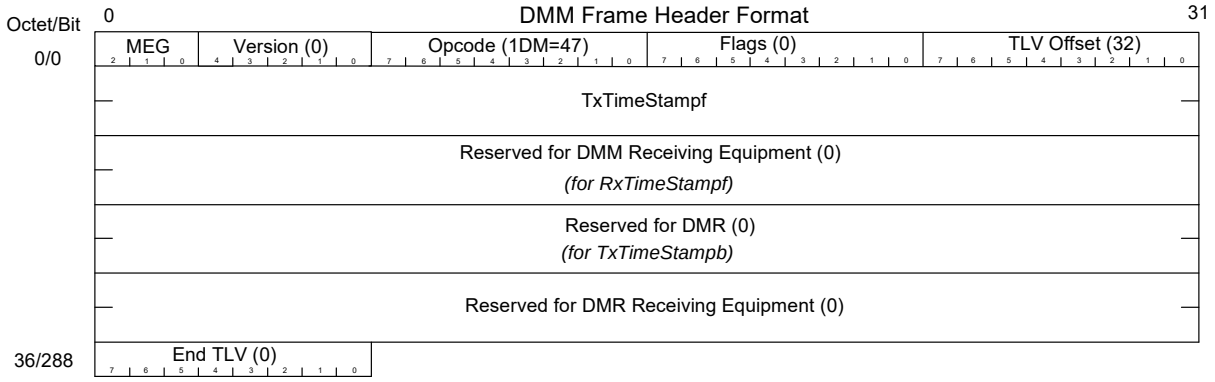
1DM, DMM, and DMR are the three supported Y.1731 frame headers. The following illustration shows the header part of a 1DM Y.1731 OAM frame.

Figure 61 • **OAM 1DM Frame Header Format**



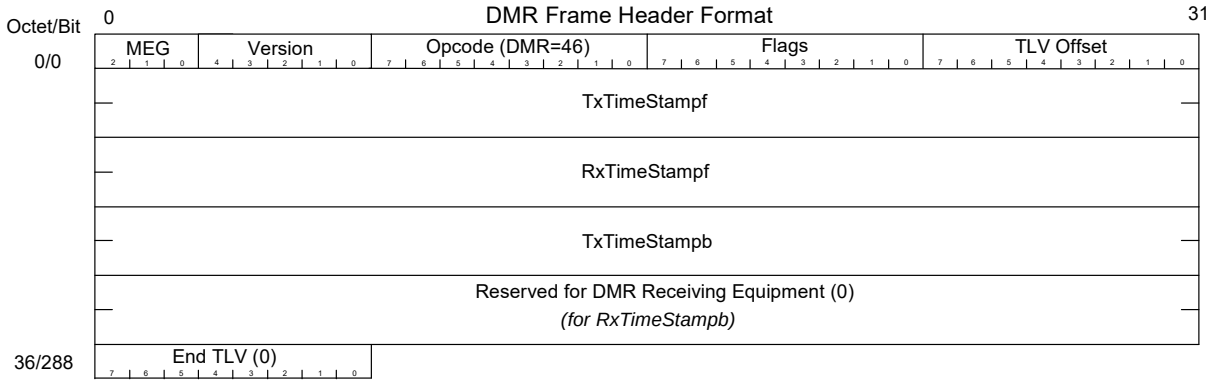
The following illustration shows a DMM frame header.

Figure 62 • **OAM DMM Frame Header Format**



The following illustration shows a DMR frame header.

Figure 63 • **OAM DMR Frame Header Format**



As with PTP, there is no protocol validation for Y.1731 frames; only interpretation of the header to determine what action to take. The first four bytes of the header are used to determine the action to be taken.

3.13.15.12PTP Comparator Action Control Registers

The following registers perform matching on the frame header and define what action is to be taken based upon the match. There is one mask register for all flows, and the rest of the registers are unique for each flow.

Table 19 • **PTP Comparison**

Parameter	Width	Description
PTP_Flow_Match	64 bit	Matches bits in the PTP/Y.1731 frame starting at the beginning of the protocol header
PTP_Flow_Mask	64 bit	Mask bits for PTP_Flow_Match
PTP_Domain_Range_Lower	8 bit	Lower range of the domain field to match
PTP_Domain_Range_Upper	8 bit	Upper range of the domain field to match
PTP_Domain_Range_ Enable	1 bit	Enable for range checking
PTP_Domain_Offset	5 bit	Pointer to the domain field, or whatever field is to be used for range checking

Table 19 • **PTP Comparison**

Parameter	Width	Description		
PTP_Action_Command	3 bit	**Command Value**	**Mnemonic**	**Action**
		0	NOP	Do nothing
		1	SUB	New correction field = Current correction field – Captured local time
		2	SUB_P2P	New correction field = Current correction field – Local latency + path_delay
		3	ADD	New correction field = Current correction field + Captured local time
		4	SUB_ADD	New correction field = Current correction field + (Captured local time + Local latency – Time storage field)
		5	WRITE_1588	Write captured local time to time storage field
		6	WRITE_P2P	Active_timestamp_ns = captured local time and path_delay written to time storage field and correction field (deprecated command)
		7	WRITE_NS	Write local time in nanoseconds to the new field
		8	WRITE_NS_P2P	Write local time in nanoseconds + p2p_delay to the new field and correction field
PTP_Save_Local_Time	1 bit	When set, saves the local time to the timestamp FIFO (only valid for egress ports).		
PTP_Correction_Field_Offset	5 bit	Points to the location of the correction field. Location is relative to the first byte of the PTP/OAM header.		
PTP_Time_Storage_Field_Offset	6 bit	Points to a location in a PTP frame where a time value can be stored or read.		
PTP_Add_Delay_Asymmetry_Enable	1 bit	When enabled, the value in the delay asymmetry register is added to the correction field of the frame.		
PTP_Subtract_Delay_Asymmetry_Enable	1 bit	When enabled, the value in the delay asymmetry register is subtracted from the correction field of the frame.		
PTP_Zero_Field_Offset	6 bit	Points to a location in the PTP/OAM frame to be zeroed if this function is enabled		
PTP_Zero_Field_Byte_Count	4 bit	The number of bytes to be zeroed. If this field is 0, then this function is not enabled.		

Table 19 • **PTP Comparison**

Parameter	Width	Description
PTP_Modified_Frame_Byte_Offset	3 bit	Indicates the position relative to the start of the PTP frame in bytes where the Modified_Frame_Status bit resides. This value is also used to calculate the offset from the beginning of the Ethernet packet to this field for use by the Rewriter.
PTP_Modified_Frame_Status_Update	1 bit	If set, tells the rewriter to update the value of this bit. Configuration registers inside the rewriter indicate if the bit will be set to 0 or 1.
PTP_Rewrite_Bytes	4 bits	Number of bytes in the PTP or OAM frame that must be modified by the Rewriter for the timestamp
PTP_Rewrite_Offset	8 bits	Points to where in the frame relative to the SFD that the timestamp should be updated
PTP_New_CF_Loc	8 bits	Location where the updated correction field value is written relative to the PTP header start
PTP_Channel_Mask	1 bit per channel per flow	Enable for this match set for this channel
PTP_Flow_Enable	1 bit	When set, the fields associated with this flow are all valid

The following table shows controls that are common to all flows.

Table 20 • **PTP Comparison: Common Controls**

Parameter	Width	Description
PTP_IP_CHKSUM_Sel	1 bit	0 = Use IP checksum controls from comparator 1 1 = Use IP checksum controls from comparator 2
FSB_Adr_Sel	2 bits	Selects the source of the address for use in the frame signature builder

The following table shows the one addition, per-flow, register.

Table 21 • **PTP Comparison: Additions for OAM-Optimized Engine**

Parameter	Width	Description
PTP_NXT_Prot_Group_Mask	2 bits	There are two bits for each flow. Each bit indicates if the flow can be associated with next-protocol group A or B. One or both bits may be set. If a bit is 1 for a particular next-protocol group, then a flow match is valid if the prior comparator stages also produced matches with the same next-protocol group.

3.13.15.13Future Protocol Compatibility

Except for MPLS, the comparators are not hardwired to their intended protocols. They can be used as generic field and range comparators because all of the offsets or pointers to the beginning of the fields are configurable. The IP comparator is the most generic and would probably be the first choice for validating a new protocol.

Additionally, if there are not enough comparison resources in a single comparator block to handle a new protocol, two comparators back-to-back can used by splitting up the comparison work. One portion can be validated in one comparator and then handed off to another. The only restriction is that there must

beat least one 64-bit word of separation between the start of the protocol and where the second starts to operate.

3.13.15.14Reconfiguration

There are three ways to perform reconfiguration:

1. Disable an entire encapsulation engine.
   Once an engine has been disabled, any of the configuration registers associated with it may be modified in any order. If other encapsulation engines are still active, they will still operate normally.
2. Disable a flow in an active engine.
   Each stage in the engine has an enable bit for each flow. If a flow is disabled in a stage, its registers may be modified. Once reconfiguration for a flow in a stage is complete, it can be enabled.
3. Disable a comparator.
   Each comparator within the active encapsulation engine can be disabled. The ETH1 comparator in all active encapsulation engines is always on and looks for the Ethernet header. If an Ethernet header according to the configuration TypeI or Type II with SNAP/LLC is not found then subsequent flows will not be matched. The ETH1 comparator cannot be disabled.

The disabling of engines and flows is always done in a clean manner so that partial matches do not occur. Flows and engines are always enabled or disabled during inter-packet gaps or at the end of a packet. This guarantees that when a new packet is received that it will be analyzed cleanly.

If strict flow matching is enabled and a flow is disabled in one of the stages, then the entire flow is automatically disabled.

If any register in a stage that applies to all flows needs to be modified, then the entire encapsulation engine must be disabled.

3.13.15.15Frame Signature Builder

Along with timestamp and CRC updates, the analyzer outputs a frame signature that can be stored in the timestamp FIFO to help match frames with other info in the FIFO. This information is used by the CPU so that it can match timestamps in the timestamp FIFO with actual frames. The frame signature is up to 16 bytes long and contains information from the Ethernet header (SA or DA), IP header (SA or DA), and from the PTP or OAM frame. The frame signature is only used in the egress direction.

The PTP block contains a set of mapping registers to configure which bytes are mapped into the frame signature. The following tables show the mapping for each byte.

Table 22 • **Frame Signature Byte Mapping**

Select	Source Byte
0-23	PTP header byte number = (31-select)
24	PTP header byte number 6
25	PTP header byte number 4
26	PTP header byte number 0
27	Reserved
28-35	Selected address byte (select-28)

Table 23 • **Frame Signature Address Source**

Parameter	Width	Description
FSB_Map_Reg_0-15	6 bits	For each byte of the frame signature, use Table 22, page 64 to select which available byte is used. Frame signature byte 0 is the LSB. If not all 16 bytes are needed, the frame signature should be packed towards the LSB and the upper unused byte configuration values do not need to be programmed.

Table 23 • **Frame Signature Address Source**

Parameter	Width	Description	
FSB_Adr_Sel	2 bits	Selects the source of the address for use in the frame signature builder according to the following list	
		Select Value	**Address Source**
		0	Ethernet block 1
		1	Ethernet block 2
		2	IP block 1
		3	IP block 2

Configuration registers in each comparator block supply an address to select if it is the source address or the destination address.

3.13.15.16 Configuration Sharing

The analyzer configuration services both channels. Each flow within each comparator has a channel-mask register that indicates which channels the flow is valid for. Each flow can be valid for channel A, channel B, or both channels.

The total of eight flows can be allocated the two channels if the analyzer configuration cannot be shared. They can each have four distinct flows (or three for the one, and five for the other, etc.).

3.13.15.17 OAM-Optimized Engine

In addition to the descriptions of the Ethernet and MPLS blocks in the OAM optimized engine, there is the notion of protocol-A/protocol-B. When a match occurs in the Ethernet 1 block the status of the protocol set that produced the match is indicated. There are two bits, one for protocol A and another for protocol B. If both sets produce a match, then both bits are set.

These bits are then carried to the next comparison block and only allow flow matches for the protocol sets that produced matches in the prior block. This block also produces a set of protocol match bits that are also carried forward.

This feature is provided to prevent a match with protocol set A in the first block and protocol set B in the second block.

3.13.16 Timestamp Processor

The primary function of the timestamp processor block is to generate a new Timestamp_field or new Correction_field (Transparent clocks) for the rewriter block. The timestamp block generates an output that is either a snapshot of the corrected Local Time (struct Timestamp) or a signed (two's complement) 64 bit Correction_field.

In the ingress direction the timestamp block calculates a new timestamp for the rewriter that indicates the earlier time when the corresponding PTP event frame entered the chip (crossed the reference plane referred to in the IEEE1588 standard).

In the egress direction the timestamp block calculates a new timestamp for the rewriter in time for the PCS block to transmit the new timestamp field in the frame. In this case the Timestamp field indicates when the corresponding PTP event frame will exit the chip.

Transparent clocks correct PTP event messages for the time resided in the transparent clock. Peer-to-Peer transparent clocks additionally correct for the propagation time on the inbound link (Path_delay). The Path_delay [ns] input to the timestamp block is software programmed based upon 1588 path delay measurements.

In general, the IEEE 1588 standard allows for a transparent clock to update the Correction_Field for both PTP event messages as well as the associated follow up message (for two-step operation). However, the TSP only updates PTP event messages. Also, the 1588 standard allows that end-to-end transparent clocks correct and forward all PTP-timing messages while Peer-to-Peer transparent clocks only correct and forward Sync and Follow_Up messages. Again, the TSP only updates PTP event messages (not Follow_Up messages).

Internally the timestamp block generates an Active_timestamp from the captured/timestamped Local time (Raw_timestamp). The Active_time stamp is the Raw_timestamp corrected for the both fixed (programmed) local chip, and variable chip latencies relative to where the Start_of_Frame_Indicator captures the local time. The timestamp block operates on the Active_timestamp based on the Command code.

The Active_timestamp is calculated differently in the Ingress and Egress directions and the equations are given below.

In the ingress direction:

```
Active_timestamp = Raw_timestamp - Local_latency - Variable_latency
```

In the egress direction:

```
Active_timestamp = Raw_timestamp + Local_latency + Variable_latency
```

In addition, the following values are also calculated for use by the commands:

```
Active_timestamp_ns = Active_timestamp converted to nanoseconds
Active_timestamp_p2p_ns = active_timestamp_ns + path delay
```

The Local_latency is a programmed fixed value while the Variable_latency is predicted from the PCS logic based upon the current state of the ingress or egress data pipeline.

For the option of Peer-to-Peer transparent clocks, the ingress Active_timestamp calculation includes an additional Path_delay component. The path delay is always added for a transparent clock per the standard. The path delay is always added to the correction field.

The signed 32-bit two's complement Delay Asymmetry register (bits 31–0) can be programmed by the user. Bit 31 is the sign bit. Bits 15–0 are scaled nanoseconds just like for the CorrectionField format. The DelayAsymmetry register (whether it be positive or negative) will be sign extended and added to the 64-bit correction field (signed add) if the Add_Delay_Asymmetry bit is set. The DelayAsymmetry register (whether it be positive or negative) will be sign extended and subtracted from the 64-bit correction field (signed Subtract) if the Subtract_Delay_Asymmetry bit is set.

The timestamp block keeps a shadow copy of the programmed latency values (Local_latency, Path_delay, and Delay_Asymmetry) to protect against CPU updates.

3.13.17 Timestamp FIFO

The Timestamp FIFO stores timestamps along with frame signature information. This information can be read out by a CPU or pushed out on a dedicated Serial Timestamp Output Interface and used in 2-step processing mode to create follow-up messages. The timestamp FIFO is only present in the egress data path.

The timestamp FIFO takes a frame signature from the analyzer and the updated correction field, and the full data set for that timestamp is saved to the FIFO. This creates an interrupt to the CPU. If the FIFO ever overflows this is indicated with an interrupt.

The stored frame signature can be of varying sizes controlled by the EGR_TSFIFO_CSR.EGR_TS_SIGNAT_BYTES register. Only the indicated number of signature bytes is saved with each timestamp. The saved values are packed so that reducing the number of signature bytes allows more timestamps to be saved.

The packing of the timestamp data is done by logic before the write occurs to the FIFO. When no compression is used, each timestamp may contain 208 bits of information consisting of 128 bits of frame signature and 80 bits of timestamp data. Therefore a full sized timestamp is 26 bytes long. Compressing the frame signature can reduce this to as little as 10 bytes (or 4 bytes if EGR_TSFIFO_CSR.EGR_TS_4BYTES = 1) if no signature information is saved (EGR_TSFIFO_CSR.EGR_TS_SIGNAT_BYTES = 0). The value to store is built up in an internal register. When the register contains 26 valid bytes, that data is written to the timestamp FIFO. Data in the FIFO is packed end-to-end. It is up to the reader of the data to unpack the data.

The timestamps in the FIFO are visible and accessible for the CPU as a set of 32-bit registers. Multiple register reads are required to read a full timestamp if all bits are used. Bit 31 in register EGR_TSFIFO_0 contains the current FIFO empty flag, which can be used by the CPU to determine if the current

timestamps are available for reading. If the bit is set, the FIFO is empty and no timestamps are available. The value that was read can be discarded because it does not contain any valid timestamp data. If the bit is 0 (deasserted), the value contains 16 valid data bits of a timestamp. The remaining bits should be read from the other registers in the other locations and properly unpacked to recreate the timestamp. Care should be taken to read the timestamps one at a time as each read of the last (7th) address will trigger a pop of the FIFO.

Timestamps are packed into seven registers named EGR_TSFIFO_0 to EGR_TSFIFO_6. If the timestamp FIFO registers are read to the point that the FIFO goes empty and there are remaining valid bytes in the internal packing register, then the packing register is written to the FIFO. In this case the registers may not be fully packed with timestamps. Flag bits are used to indicate where the valid data ends within the set of seven registers. The flag bits are in register EGR_TSFIFO_0.EGR_TS_FLAGS (together with the empty flag) and are encoded as follows:

000 = Only a partial timestamp is valid in the seven register set

001 = One timestamp begins in the current seven register set

010 = Two timestamps begin in the current seven register set.

011 = Three timestamps begin in the current seven register set (4-byte mode)

100 = Four timestamps begin in the current seven register set (4-byte mode)

101 = Five timestamps begin in the current seven register set (4-byte mode)

110 = Six timestamps begin in the current seven register set (4-byte mode)

111 = The current seven register set is fully packed with valid timestamp data

The FIFO empty bit is visible in the EGR_TSFIFO_0.EGR_TS_EMPTY register so the CPU can poll this bit to know when timestamps are available. There is also a maskable interrupt which will assert whenever the timestamp FIFO level reaches the threshold given in EGR_TSFIFO_CSR.EGR_TS_THRESH register. The FIFO level is also visible in the EGR_TSFIFO_CSR.EGR_TS_LEVEL register. If the timestamp FIFO overflows, writes to the FIFO are inhibited. The data in the FIFO is still available for reading but new timestamps are dropped.

Note: Timestamp FIFO exists only in the Egress direction. There is no Timestamp FIFO in the Ingress direction

3.13.18 Serial Timestamp Output Interface

For each 1588 Processor 0 and 1, timestamp information stored in the Egress direction can be read through either the register interface or through the Serial Timestamp interface. These two ways to read registers are mutually exclusive. While enabling/disabling the serial interface is done on a Processor level, only one serial interface exists. This means the serial interface can be enabled for Processor 0, while the timestamp FIFO can be read through registers for Processor 1. If the serial interface is enabled for both Processor 0 and 1, then the serial interface will arbitrate between two Egress Timestamp FIFOs in Processor 0 and 1 and push the data out.

The timestamp FIFO serial interface block writes, or pushes, timestamp/frame signature pairs that have been enqueued and packed into timestamp FIFOs to the external chip interface consisting of three output pins: 1588_SPI_DO, 1588_SPI_CLK, and 1588_SPI_CS. There is one interface for all channels.

When the serial interface (SPI) is enabled, the timestamp/frame signature pairs are dequeued from timestamp FIFO(s) and unpacked. Unpacked timestamp/frame signature pairs are then serialized and sent one at a time to the external interface. Unpacking shifts the timestamp/frame signature into alignment considering the configured size of the timestamps and frame signatures (a single SI write may require multiple reads from a timestamp FIFO). The timestamp FIFO serial interface is an alternative to the MDIO register interface described in the timestamp FIFO section. When the serial timestamp interface is enabled in register TS_FIFO_SI_CFG.TS_FIFO_SI_ENA, data read from the timestamp FIFO registers described in Timestamp FIFO, page 66 are invalid.

Timestamp/Frame signature pairs from two egress timestamp FIFOs are serialized one at a time and transmitted to the interface pins. The TS_FIFO_SI arbitrates in a round-robin fashion between the ports that have non-empty timestamp FIFOs. The port associated with each transmitted timestamp/frame signature pair is indicated in a serial address that precedes the data phase of the serial transmission.

Because the timestamp FIFOs are instantiated in the per port clock domains, a small single entry asynchronous SI FIFO (per port) ensures that the timestamp/frame signature pairs are synchronized, staged, and ready for serial transmission. When an SI FIFO is empty, the SI FIFO control fetches and/or unpacks a single timestamp/frame signature performing any timestamp FIFO dequeues necessary. The SI FIFO goes empty following the completion of the last data bit of the serial transmission. Enabled ports (TS_FIFO_SI_CFG.TS_FIFO_SI_ENA) participate in the round-robin selection.

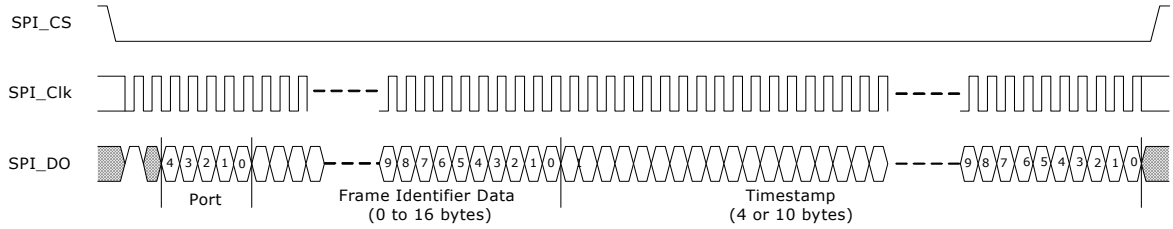
Register TS_FIFO_SI_TX_CNT accumulates the number of timestamp/frame signature pairs transmitted from the serial timestamp interface for each channel. Register EGR_TS_FIFO_DROP_CNT accumulates the number of timestamp/frame signature pairs that have been dropped per channel due to a timestamp FIFO overflow.

The SPI compatible interface asserts a chip select (SPI_CS) for each write followed by a write command data bit equal to 1, followed by a “don't care” bit (0), followed by an address phase, followed by a data phase, followed by a deselect where SPI_CS is negated. Each write command corresponds to a single timestamp/frame signature pair. The length of the data phase depends upon the sum of the configured lengths of the timestamp and signature, respectively. The address phase is fixed at five bits. The SPI_CLK is toggled to transfer each SPI_DO bit (as well as the command and address bits). The “Timestamp” and “Frame Identifier Data” from the following illustration are sent MSB first down to LSB (bit 0) in the same format as stored in the seven registers of TS FIFO CSRs. For more information, see Timestamp FIFO, page 66 and Figure 64, page 68.

The frequency of the generated output 1588_SPI_CLK can be flexibly programmed from 10 MHz up to 62.5 MHz using TS_FIFO_SI_CFG to set the number of CSR clocks that the 1588_SPI_CLK is both high and low. For example, to generate a 1588_SPI_CLK that is a divide-by-6 of the CSR clock, the CSR register would be set such that both SI_CLK_LO_CYCS and SI_CLK_HI_CYCS equal 3. Also, the number of CSR clocks after SPI_CS asserts before the first 1588_SPI_CLK is programmable (SI_EN_ON_CYCS), as is the number of clocks before SI_EN negates after the last 1588_SPI_CLK (SI_EN_OFF_CYCS). The number of clocks during which SI_EN is negated between writes is also programmable (SI_EN_DES_CYCS). The 1588_SPI_CLK may also be configured to be inverted (SI_CLK_POL).

Without considering de-selection between writes, if the PTP 16-byte SequenceID (frame signature) is used as frame identifier each 10 byte time stamp write take 2 + 55 + 10 × 8 + 16 × 8 = 265 clocks (at 40 MHz) ~6625 ns. This corresponds to a time stamp bandwidth of > 0.15 M time stamp/second/port. The following illustration shows the serial time stamp/frame signature output.

Figure 64 • **Serial Time Stamp/Frame Signature Output**



3.13.19 Rewriter

When the rewriter block gets a valid indication it overwrites the input data starting at the offset specified in Rewrite_offset and replaces N bytes of the input data with updated N bytes. Frames are modified by the rewriter as indicated by the analyzer-only PTP/OAM frames are modified by the rewriter.

The output of the rewriter block is the frame data stream that includes both unmodified frames and modified PTP frames. The block also outputs a count of the number of modified PTP frames in INGR_RW_MODFRM_CNT/EGR_RW_MODFRM_CNT, depending upon the direction. This counter accumulates the number of PTP frames to which a write was performed and includes errored frames.

3.13.19.1 Rewriter Ethernet FCS Calculation

The rewriter block has to recalculate the Ethernet CRC for the PTP message to modify the contents by writing a new timestamp or clear bytes. Two versions of the Ethernet CRC are calculated in accordance

with IEEE 802.3 Clause 3.2.9: one on the unmodified input data stream and one on the modified output data stream. The input frame FCS is checked against the input calculated FCS and if the values match, the frame is good. If they do not, then the frame is considered a bad or errored frame. The new calculated output FCS is used to update the FCS value in the output data frame. If the frame was good, then the FCS is used directly. If the frame was bad, the calculated output FCS is inverted before writing to the frame. Each version of the FCS is calculated in parallel by a separate FCS engine.

A count of the number of PTP/OAM frames that are in error is kept in the INGR_RW_FCS_ERR_CNT or EGR_RW_FCS_ERR_CNT register, depending upon the direction.

3.13.19.2 Rewriter UDP Checksum Calculation

For IPv6/UDP, the rewriter also calculates the value to write into the dummy blocks to correct the UDP checksum. The checksum correction is calculated by taking the original frame's checksum, the value in the dummy bytes, and the new data to be written; and using them to modify the existing value in the dummy byte location. The new dummy byte value is then written to the frame to ensure a valid checksum. The location of the dummy bytes is given by the analyzer. The UDP checksum correction is only performed when enabled using the following register bits:

- INGR_IP1_UDP_CHKSUM_UPDATE_ENABLE
- INGR_IP2_UDP_CHKSUM_UPDATE_ENABLE
- EGR_IP1_UDP_CHKSUM_UPDATE_ENABLE
- EGR_IP2_UDP_CHKSUM_UPDATE_ENABLE

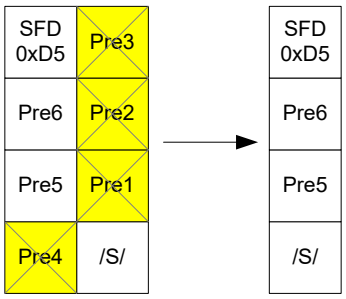
Based upon the analyzer command and the rewriter configuration, the rewriter writes the timestamp in one of the following ways:

- Using PTP_REWRITE_BYTES to choose four bytes write to PTP_REWRITE_OFFSET. This method is similar to other PTP frame modifications and the timestamp is typically written to the reserved field in the PTP header.
- Using PTP_REWRITE_BYTES and RW_REDUCE_PREAMBLE to select the mode of operation when writing Rx timestamps into the frame.
  In these modes, it cannot do both a time stamp write/append and a PTP operation in the same frame. If PTP_REWRITE_BYTES = 0xE and RW_REDUCE_PREAMBLE = 1, it does it by overwriting the existing FCS with the timestamp in the lowest four bytes of the calculated timestamp and generating a new FCS and appending it.

Because the rewriter cannot modify the IFG or change the size of the frame, if the original FCS is overwritten with timestamp data a new FCS needs to be appended and the frame shortened by reducing the preamble. The preamble length includes the
/S/ character and all preamble characters up to but not including the SFD. In this mode, it is assumed that all incoming preambles are of sufficient (5 to 7-byte) length to delete four bytes and the preamble of every frame (not only PTP frames) will be reduced by four bytes by deleting four bytes of the preamble. Then, the new FCS is written at the end of the matched frame. For unmatched frames, or if the PTP_REWRITE_BYTES is anything but 0xE, the IFG is increased by adding four IDLE (/I/) characters after the /T/ which ends the packet.

To timestamp a frame in one of the modes, the actual length of the preamble is then checked and if the preamble is too short to allow a deletion of four bytes (if the preamble is not five bytes or more) then no operations are performed on the preamble, the FCS is not overwritten, and no timestamp is appended. For all such frames, a counter is maintained and every time an unsuccessful operation is encountered, the counter is incremented. This counter is read through register:
INGR_RW_PREAMBLE_ERR_CNT/EGR_RW_PREAMBLE_ERR_CNT. The following illustration shows the deleted preamble bytes.

Figure 65 • **Preamble Reduction in Rewriter**



If PTP_REWRITE_BYTES = 0xF and RW_REDUCE_PREAMBLE = 0, the rewriter replaces the FCS of the frame with the four lowest bytes of the calculated time stamp and does not write the FCS to the frame. In this mode, all the frames have corrupted FCSs and the MAC needs to be configured to handle this case. In the case of a CRC error in the original frame, the rewriter writes all ones (0xFFFFFFFF) to the FCS instead of the timestamp. This indicates an invalid CRC to the MAC because this is reserved to indicate an invalid timestamp. In the rare case that the actual timestamp has the value 0xFFFFFFFF and the CRC is valid, the rewriter increments the timestamp to 0x0 and writes that value instead. This causes an error of 1 ns but is required to reserve the timestamp value of 0xFFFFFFFF for frames with an invalid CRC.

A flag bit may also be set in the PTP message header to indicate that the TSU has modified the frame (when set) or to clear the bit (on egress). The analyzer sends the byte offset of the flag byte to the rewriter in PTP_MOD_FRAME_BYTE_OFFSET and indicates whether the bit should be modified or not using PTP_MOD_FRAME_STATUS_UPDATE. The bit offset within the byte is programmed in the configuration register RW_FLAG_BIT. When the PTP frame is being modified, the selected bit is set to the value in the RW_FLAG_VAL. This only occurs when the frame is being modified by the rewriter; when the PTP frame matches and the command is not NOP.

3.13.20 Local Time Counter

The local time counter keeps the local time for the device and the time is monitored and synchronized to an external reference by the CPU. The source clock for the counter is selected externally to be a 250 MHz, 200 MHz, 125 MHz, or some other frequency. The clock may be a line clock or the dedicated 1588_DIFF_INPUT_CLK_P/N pins. The clock source is selected in register LTC_CTRL.LTC_CLK_SEL.

To support other frequencies, a flexible counter system is used that can convert almost any frequency in the 125–250 MHz range into a usable source clock. The frequency is programmed in terms of the clock period. Set the LTC_SEQUENCE.LTC_SEQUENCE_A register to the clock period to the nearest whole number of nanoseconds to be added to the local time counter on each clock cycle. Set LTC_SEQ.LTC_SEQ_E to the amount of error between the actual clock period and the LTC_SEQUENCE.LTC_SEQUENCE_A setting in femtoseconds. Register LTC_SEQ.LTC_SEQ_ADD_SUB indicates the direction of the error. An internal counter keeps track of the accumulated error. When the accumulated error exceeds 1 nanosecond, an extra nanosecond is either added or subtracted from the local time counter. Use the following as an example to program a 5.9 ns period:

```
LTC_SEQUENCE.LTC_SEQUENCE_A = 6 (6 ns)
LTC_SEQ.LTC_SEQ_E = 100000 (0.1 ns)
LTC_SEQ.LTC_SEQ_ADD_SUB = 0 (subtract an extra nanosecond, i.e add 5 ns)
```

To support automatic PPM adjustments, an internal counter runs on the same clock as the local time counter, and increments using the same sequence to count nanoseconds. The maximum (rollover) value of the internal counter in nanoseconds is given in register LTC_AUTO_ADJUST.LTC_AUTO_ADJUST_NS. At rollover, the next increment of the local time counter is increased by one additional or one less nanosecond as determined by the LTC_AUTO_ADJUST.LTC_AUTO_ADD_SUB_1NS register. When LTC_AUTO_ADJUST.LTC_AUTO_ADD_SUB_1NS is set to 0x1, an additional nanosecond is added to the local time counter. When it is set to 0x2, one less nanosecond is added to the local timer counter. No PPM adjustments are made when the register is set to 0x0 or 0x3.

PPM adjustments to the local time counter can be made on an as-needed basis by writing to the one-shot LTC_CTRL.LTC_ADD_SUB_1NS_REQ register. One nanosecond is added or subtracted from the local time counter each time LTC_CTRL.LTC_ADD_SUB_1NS_REQ is asserted. The LTC_CTRL.LTC_ADD_SUB_1NS register setting controls whether the local time counter adjustment is an addition or a subtraction.

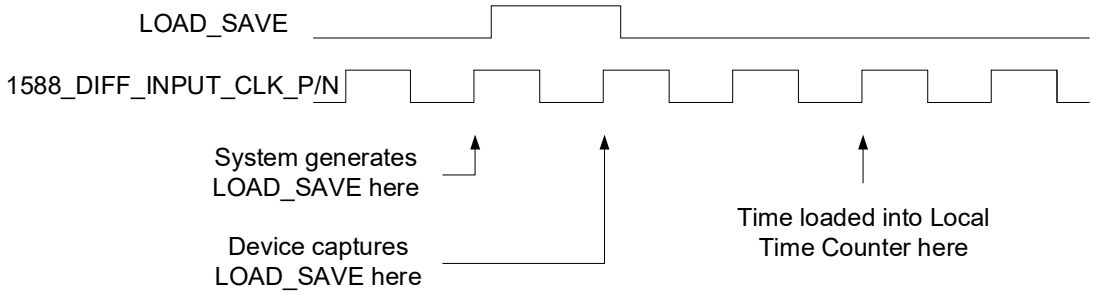
The current time is loaded into the local time counter with the following procedure.

1. Configure the 1588_LOAD_SAVE pin.
2. Write the time to be loaded into the local time counter in registers LTC_LOAD_SEC_H, LTC_LOAD_SEC_L and LTC_LOAD_NS.
3. Program LTC_CTRL.LTC_LOAD_ENA to a 1.
4. Drive the 1588_LOAD_SAVE pin from low to high.

The time in registers LTC_LOAD_SEC_H, LTC_LOAD_SEC_L and LTC_LOAD_NS is loaded into the local time counter when the rising edge of the 1588 LOAD_SAVE strobe is detected. The LOAD_SAVE strobe is synchronized to the local time counter clock domain.

When the 1588_DIFF_INPUT_CLK_P/N pins are the clock source for the local time counter, and the LOAD_SAVE strobe is synchronous to 1588_DIFF_INPUT_CLK_P/N, the LTC_LOAD* registers are loaded into the local time counter, as shown in the following illustration.

Figure 66 • **Local Time Counter Load/Save Timing**



When the LOAD_SAVE strobe is not synchronous to the 1588_DIFF_INPUT_CLK_P/N pins or an internal clock drives the local time counter, there is some uncertainty as to when the local time counter is loaded. This reduces the accuracy of the time stamping function by the period of the local time counter clock.

Note: There is a local time counter in each channel. The counter is initialized in both channels if the LTC_CTRL.LTC_LOAD_ENA register in each channel is asserted when the LOAD_SAVE strobe occurs.

If the LTC_CTRL.LTC_SAVE_ENA register is asserted when the 1588 LOAD_SAVE input transitions from low to high, the state of the local time counter is stored in the LTC_SAVED_SEC_H, LTC_SAVED_SEC_L, and LTC_SAVED_NS registers. As with loading the local time counter, there is one clock cycle of uncertainty as to when the time is saved if the LOAD_SAVE strobe is not synchronous to the clock driving the counter.

The following is an example of an automatic adjustment calculation:

If a 250 MHz local time counter clock is off by 100 PPM (0.01%), then the 4 ns period is off by 0.0004 ns every clock cycle. A 1 ns adjustment would need to be made to the local time counter every 1 ns/(0.0004 ns) cycles, or every 2500 cycles. Because the clock period is 4 ns, the LTC_AUTO_ADJUST.LTC_AUTO_ADJUST_NS register would be set to 2500 x 4 which is 10,000 ns.

3.13.21 Accuracy and Resolution

Contact Microsemi with any questions regarding PTP accuracy calculations. The timestamp accuracy is a system-level property and may depend upon oscillator selection, port type and speed, system configuration, and calibration decisions.

Supported frequencies of the local time counter are 125 MHz, 156.25 MHz, 200 MHz, and 250 MHz. The time stamp resolution is equal to the local time counter clock period. For example, a 250 MHz local time counter clock will provide a 4 ns time stamp resolution.

3.13.22 Accessing 1588 IP Registers

The following sections describe how the 1588 IP registers are accessed in the VSC8572-02 device.

Note: Contact Microsemi for an initialization script that supports the quick initialization of 1588 registers.

3.13.22.1 1588 Register Access Using SMI (MDC/MDIO) Mechanism

The SMI mechanism is an IEEE defined register access mechanism (refer to Clause 22 of IEEE 802.3). The registers are arranged as 16 bits per register address with a 5 bit address field as defined by IEEE. However Microsemi has extended this register address space by creating a register page key in register 31. When writing a particular key to register 31, a different set of 5 bit address space register bank can be accessed through the SMI mechanism. (extended page, GPIO page, etc).

The 1588 registers are organized on page 0x1588. Setting Register 31 to 0x1588 provides a window to CSR registers through registers 16,17, and 18.

The 1588 IP registers are arranged as 32 bits of data. The access method through SMI is done by breaking up the 32 bits of each 1588 register into the high 16 bits into register 18 and lower 16 bits into register 17. Then register 16 is used as a command register. Phy02 automatically reads/writes to engine A. Phy1 automatically reads/writes to engine B. For more information, see Figure 27, page 26. The following tables show the bit descriptions for register 16, 17, and 18.

Table 24 • **Register 16**

Bit	Access	Description
15	RWSC	Command bit. 1: Must be set to execute the command. It is set back to 1 when done. 0: Command busy, do not do any write to register 16. Register 17 and 18 maintain previous write values.
14	RW	1: Execute a read on the CSR registers 0: Execute a write on the CSR registers
13:11	RW	Target block code 000: Analyzer 0 Ingress 001: Analyzer 0 Egress 010: Analyzer 1 Ingress 011: Analyzer 1 Egress 100: Analyzer 2 Ingress 101: Analyzer 2 Egress 110: Processor 0 111: Processor 1
10:0	RW	CRS register address[10:0](1)

1. For more information about valid 1588 registers for this field, see 1588 IP Registers, page 142.

Table 25 • **Register 17**

Bit	Access	Description
15:0	RWSC	CSR Data_LSB[15:0]

Table 26 • **Register 18**

Bit	Access	Description
15:0	RWSC	CSR Data_MSB[31:16]

3.13.23 1588_DIFF_INPUT_CLK Configuration

The default configuration of the 1588_DIFF_INPUT_CLK_P/N pins sets the device to use an internal clock for the LTC. To configure these pins correctly to use an external clock for LTC, write 0xb71c to register 30E1588 and 0x7ae0 to register 29E1588. Set these two registers to 0x0 when an internal clock is used for LTC.

3.14 Media Recovered Clock Outputs

For Synchronous Ethernet applications, the VSC8572-02 includes two recovered clock output pins, RCVRDCLK1 and RCVRDCLK2, controlled by registers 23G and 24G, respectively. The recovered clock pins are synchronized to the clock of the active media link.

To enable recovered clock output, set register 23G or 24G, bit 15, to 1. By default, the recovered clock output pins are disabled and held low, including when NRESET is asserted. Registers 23G and 24G also control the PHY port for clock output, the clock source, the clock frequency (either 25 MHz, 31.25 MHz, or 125 MHz), and squelch conditions.

Note: When EEE is enabled on a link, the use of the recovered clock output is not recommended due to long holdovers occurring during EEE Quiet/Refresh cycles.

3.14.1 Clock Selection Settings

On each pin, the recovered clock supports the following sources, as set by registers 23G or 24G, bits 2:0:

- Fiber SerDes media recovered clock
- Copper PHY recovered clock
- Copper PHY media transmitter TCLK output (RCVRDCLK1 only.) For more information, see Table 91, page 125 and Table 92, page 126.

Note: When using the automatic media sense feature, the recovered clock output cannot automatically change between each active media. Changing the media source must be managed through the recovered clock register settings.

Adjust the squelch level to enable 1000BASE-T master mode recovered clock for SyncE operation. This is accomplished by changing the 23G and 24G register bits 5:4 to 01. This setting also provides clock out for 10BASE-T operation. For 1000BASE-T master mode, the clock is based on the VSC8572-02 REFCLK input, which is a local clock.

3.14.2 Clock Output Squelch

Under certain conditions, the PHY outputs a clock based on the REFCLK_P and REFCLK_N pins, such as when there is no link present or during autonegotiation. To prevent an undesirable clock from appearing on the recovered clock pins, the VSC8572-02 squelches, or inhibits, the clock output based on any of the following criteria:

- No link is detected (the link status register 1, bit 2 = 0). In fiber media modes, sync status is required to unsquelch the recovered clock output instead of link status.
- The link is found to be unstable using the fast link failure detection feature. The GPIO9/FASTLINK-FAIL pin is asserted high when enabled.
- The active link is in 10BASE-T or in 1000BASE-T master mode. These modes produce unreliable recovered clock sources.
- CLK_SQUELCH_IN is enabled to squelch the clock.

Use registers 23G or 24G, bits 5:4 to configure the clock squelch criteria. These registers can also disable the squelch feature. The CLK_SQUELCH_IN pin controls the squelching of the clock. Both RCVRDCLK1 and RCVRDCLK2 are squelched when the CLK_SQUELCH_IN pin is high.

3.15 Serial Management Interface

The VSC8572-02 device includes an IEEE 802.3-compliant serial management interface (SMI) that is affected by use of its MDC and MDIO pins. The SMI provides access to device control and status registers. The register set that controls the SMI consists of 32 16-bit registers, including all required IEEE-specified registers. Also, there are additional pages of registers accessible using device register 31.

Energy efficient Ethernet control registers are available through the SMI using Clause 45 registers and Clause 22 register access in registers 13 through 14. For more information, see Table 28, page 76 and Table 95, page 127.

The SMI is a synchronous serial interface with input data to the VSC8572-02 on the MDIO pin that is clocked on the rising edge of the MDC pin. The output data is sent on the MDIO pin on the rising edge of the MDC signal. The interface can be clocked at a rate from 0 MHz to 12.5 MHz, depending on the total load on MDIO. An external 2-kΩ pull-up resistor is required on the MDIO pin.

3.15.1 SMI Frames

Data is transferred over the SMI using 32-bit frames with an optional, arbitrary-length preamble. Before the first frame can be sent, at least two clock pulses on MDC must be provided with the MDIO signal at logic one to initialize the SMI state machine. The following illustrations show the SMI frame format for read and write operations.

Figure 67 • **SMI Read Frame**

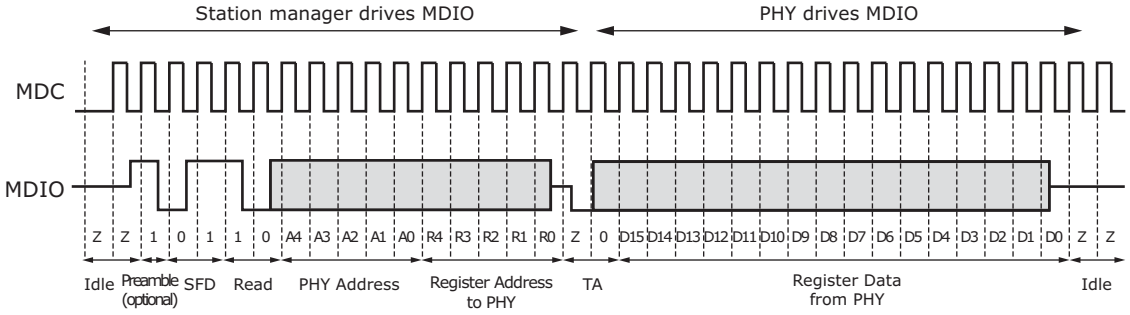
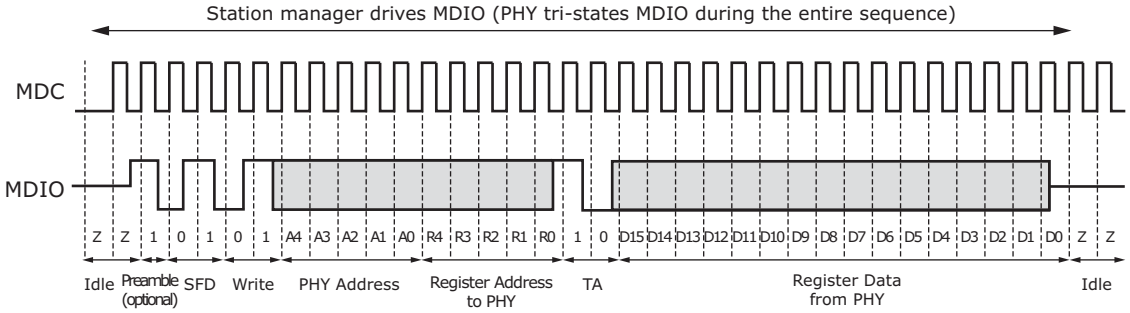


Figure 68 • **SMI Write Frame**



The following list provides additional information about the terms used in the SMI read and write timing diagrams.

- Idle—During idle, the MDIO node goes to a high-impedance state. This allows an external pull-up resistor to pull the MDIO node up to a logical 1 state. Because the idle mode does not contain any transitions on MDIO, the number of bits is undefined during idle.
- Preamble—By default, preambles are not expected or required. The preamble is a string of ones. If it exists, the preamble must be at least 1 bit; otherwise, it can be of an arbitrary length.
- Start of Frame (SFD)—A pattern of 01 indicates the start of frame. If the pattern is not 01, all following bits are ignored until the next preamble pattern is detected.
- Read or Write Opcode—A pattern of 10 indicates a read. A 01 pattern indicates a write. If the bits are not either 01 or 10, all following bits are ignored until the next preamble pattern is detected.
- PHY Address—The particular VSC8572-02 responds to a message frame only when the received PHY address matches its physical address. The physical address is 5 bits long (4:0).
- Register Address—The next five bits are the register address.
- Turnaround—The two bits used to avoid signal contention when a read operation is performed on the MDIO are called the turnaround (TA) bits. During read operations, the VSC8572-02 drives the second TA bit, a logical 0.

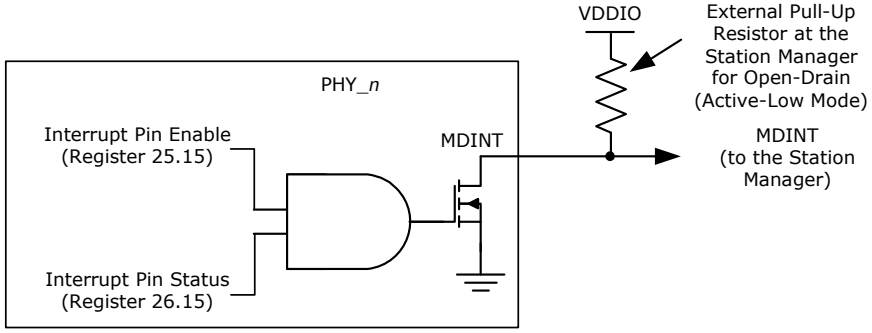
- Data—The 16-bits read from or written to the device are considered the data or data stream. When data is read from a PHY, it is valid at the output from one rising edge of MDC to the next rising edge of MDC. When data is written to the PHY, it must be valid around the rising edge of MDC.
- Idle—The sequence is repeated.

3.15.2 SMI Interrupt

The SMI includes an output interrupt signal, MDINT, for signaling the station manager when certain events occur in the VSC8572-02.

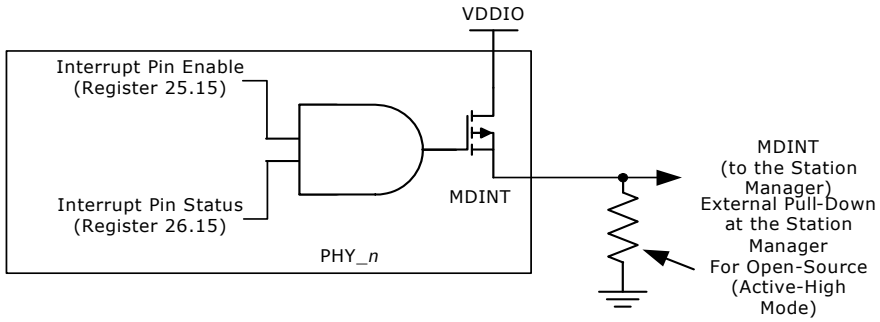
The MDINT pin can be configured for open-drain (active-low) by tying the pin to a pull-up resistor and to VDDIO. The following illustration shows this configuration.

Figure 69 • **MDINT Configured as an Open-Drain (Active-Low) Pin**



Alternatively, the MDINT pin can be configured for open-source (active-high) by tying the pin to a pull-down resistor and to VSS. The following illustration shows this configuration.

Figure 70 • **MDINT Configured as an Open-Source (Active-High) Pin**



When a PHY generates an interrupt, the MDINT pin is asserted (driven high or low, depending on resistor connection) if the interrupt pin enable bit (MII register 25.15) is set.

3.16 LED Interface

The LED interface supports the following configurations: direct drive, basic serial LED mode, and enhanced serial LED mode. The polarity of the LED outputs is programmable and can be changed through register 17E2, bits 13:10. The default polarity is active low.

Direct drive mode provides four LED signals per port, LED0_[0:3] through LED3_[0:3]. The mode and function of each LED signal can be configured independently. When serial LED mode is enabled, the direct drive pins not used by the serial LED interface remain available.

In basic serial LED mode, all signals that can be displayed on LEDs are sent as LED_Data and LED_CLK for external processing. In enhanced serial LED mode, up to four LED signals per port can be sent as LED_Data, LED_CLK, LED_LD, and LED_Pulse. The following sections provide detailed information about the various LED modes.

Note: LED number is listed using the convention, LED<LED#>_<Port#>.

The following table shows the bit 9 settings for register 14G that are used to control the LED behavior for all the LEDs in VSC8572-02.

Note: Enhanced serial LED mode will use the V_{DD} LED drive state (alternate setting), regardless of the setting in register 14G.9.

Table 27 • **LED Drive State**

Setting	Active	Not Active
14G.9 = 1 (default)	Ground	Tristate
14G.9 = 0 (alternate setting)	Ground	V_{DD}

3.16.1 LED Modes

Each LED pin can be configured to display different status information that can be selected by setting the LED mode in register 29. The modes listed in the following table are equivalent to the setting used in register 29 to configure each LED pin. The default LED state is active low and can be changed by modifying the value in register 17E2, bits 13:10. The blink/pulse-stretch is dependent on the LED behavior setting in register 30.

The following table provides a summary of the LED modes and functions.

Table 28 • **LED Mode and Function Summary**

Mode	Function Name	LED State and Description
0	Link/Activity	1: No link in any speed on any media interface. 0: Valid link at any speed on any media interface. Blink or pulse-stretch = Valid link at any speed on any media interface with activity present.
1	Link1000/Activity	1: No link in 1000BASE-T or 1000BASE-X. 0: Valid 1000BASE-T or 1000BASE-X. Blink or pulse-stretch = Valid 1000BASE-T or 1000BASE-X link with activity present.
2	Link100/Activity	1: No link in 100BASE-TX or 100BASE-FX. 0: Valid 100BASE-TX or 100BASE-FX. Blink or pulse-stretch = Valid 100BASE-TX or 100BASE-FX link with activity present.
3	Link10/Activity	1: No link in 10BASE-T. 0: Valid 10BASE-T link. Blink or pulse-stretch = Valid 10BASE-T link with activity present.
4	Link100/1000/Activity	1: No link in 100BASE-TX, 100BASE-FX, 1000BASE-X, or 1000BASE-T. 0: Valid 100BASE-TX, 100BASE-FX, 1000BASE-X, or 1000BASE-T link. Blink or pulse-stretch = Valid 100BASE-TX, 100BASE-FX, 1000BASE-X, or 1000BASE-T link with activity present.
5	Link10/1000/Activity	1: No link in 10BASE-T, 1000BASE-X, or 1000BASE-T. 0: Valid 10BASE-T, 1000BASE-X, or 1000BASE-T link. Blink or pulse-stretch = Valid 10BASE-T, 1000BASE-X, or 1000BASE-T link with activity present.
6	Link10/100/Activity	1: No link in 10BASE-T, 100BASE-FX, or 100BASE-TX. 0: Valid 10BASE-T, 100BASE-FX, or 100BASE-TX link. Blink or pulse-stretch = Valid 10BASE-T, 100BASE-FX, or 100BASE-TX link with activity present.

Table 28 • **LED Mode and Function Summary *(continued)***

Mode	Function Name	LED State and Description
7	Link100BASE-FX/1000BASE-X/Activity	1: No link in 100BASE-FX or 1000BASE-X. 0: Valid 100BASE-FX or 1000BASE-X link. Blink or pulse-stretch = Valid 100BASE-FX or 1000BASE-X link with activity present.
8	Duplex/Collision	1: Link established in half-duplex mode, or no link established. 0: Link established in full-duplex mode. Blink or pulse-stretch = Link established in half-duplex mode but collisions are present.
9	Collision	1: No collision detected. Blink or pulse-stretch = Collision detected.
10	Activity	1: No activity present. Blink or pulse-stretch = Activity present (becomes TX activity present when register bit 30.14 is set to 1).
11	100BASE-FX/1000BASE-X Fiber Activity	1: No 100BASE-FX or 1000BASE-X activity present. Blink or pulse-stretch = 100BASE-FX or 1000BASE-X activity present (becomes RX activity present when register bit 30.14 is set to 1).
12	Autonegotiation Fault	1: No autonegotiation fault present. 0: Autonegotiation fault occurred.
13	Serial Mode	Serial stream. See Basic Serial LED Mode, page 78. Only relevant on PHY port 0 and reserved in others.
14	Force LED Off	1: De-asserts the LED(1).
15	Force LED On	0: Asserts the LED(1).

1. Setting this mode suppresses LED blinking after reset.

3.16.2 Extended LED Modes

In addition to the LED modes in register 29, there are also additional LED modes that are enabled on the LED0_[1:0] pins whenever the corresponding register 19E1, bits 15 to 12 are set to 1. Each of these bits enables extended modes on a specific LED pin and these extended modes are shown in the following table. For example, LED0 = mode 17 means that register 19E1 bit 12 = 1 and register 29 bits 3 to 0 = 0001.

The following table provides a summary of the extended LED modes and functions.

Table 29 • **Extended LED Mode and Function Summary**

Mode	Function Name	LED State and Description
16	Link1000BASE-X Activity	1: No link in 1000BASE-X. 0: Valid 1000BASE-X link.
17	Link100BASE-FX Activity	1: No link in 100BASE-FX. 0: Valid 100BASE-FX link.
18	1000BASE-X Activity	1: No 1000BASE-X activity present. Blink or pulse-stretch = 1000BASE-X activity present.
19	100BASE-FX Activity	1: No 100BASE-FX activity present. Blink or pulse-stretch = 100BASE-FX activity present.
20	Force LED Off	1: De-asserts the LED.
21	Force LED On	0: Asserts the LED. LED pulsing is disabled in this mode.

Table 29 • **Extended LED Mode and Function Summary *(continued)***

Mode	Function Name	LED State and Description
22	Fast Link Fail	1: Enable fast link fail on the LED pin 0: Disable

3.16.3 LED Behavior

Several LED behaviors can be programmed into the VSC8572-02. Use the settings in register 30 and 19E1 to program LED behavior, which includes the following.

3.16.3.1 LED Combine

Enables an LED to display the status for a combination of primary and secondary modes. This can be enabled or disabled for each LED pin. For example, a copper link running in 1000BASE-T mode and activity present can be displayed with one LED by configuring an LED pin to Link1000/Activity mode. The LED asserts when linked to a 1000BASE-T partner and also blinks or performs pulse-stretch when activity is either transmitted by the PHY or received by the Link Partner. When disabled, the combine feature only provides status of the selected primary function. In this example, only Link1000 asserts the LED, and the secondary mode, activity, does not display when the combine feature is disabled.

3.16.3.2 LED Blink or Pulse-Stretch

This behavior is used for activity and collision indication. This can be uniquely configured for each LED pin. Activity and collision events can occur randomly and intermittently throughout the link-up period. Blink is a 50% duty cycle oscillation of asserting and de-asserting an LED pin. Pulse-stretch guarantees that an LED is asserted and de-asserted for a specific period of time when activity is either present or not present. These rates can also be configured using a register setting.

3.16.3.3 Rate of LED Blink or Pulse-Stretch

This behavior controls the LED blink rate or pulse-stretch length when blink/pulse-stretch is enabled on an LED pin. The blink rate, which alternates between a high and low voltage level at a 50% duty cycle, can be set to 2.5 Hz, 5 Hz, 10 Hz, or 20 Hz. For pulse-stretch, the rate can be set to 50 ms, 100 ms, 200 ms, or 400 ms. The blink rate selection for PHY0 globally sets the rate used for all LED pins on all PHY ports.

3.16.3.4 LED Pulsing Enable

To provide additional power savings, the LEDs (when asserted) can be pulsed at 5 kHz with a programmable duty cycle through register 25G bits 15:8. For duty cycle details, see Table 114, page 137.

3.16.3.5 LED Blink After Reset

The LEDs will blink for one second after power-up and after any time all resets have been de-asserted. This can be disabled through register 19E1, bit 11 = 0.

3.16.3.6 Fiber LED Disable

This bit controls whether the LEDs indicate the fiber and copper status (default) or the copper status only.

3.16.3.7 Pulse Programmable Control

These bits add the ability to width and frequency of LED pulses. This feature facilitates power reduction options.

3.16.3.8 Fast Link Failure

For more information about this feature, see Fast Link Failure Indication, page 80.

3.16.4 Basic Serial LED Mode

Optionally, the VSC8572-02 can be configured so that access to all its LED signals is available through two pins. This option is enabled by setting LED0 on PHY0 to serial LED mode in register 29, bits 7:0 to 0xDD. When setting 3:0 to 0xD on the serial data signal is enabled, setting 7:4 to 0xD enables the serial clock. When serial LED mode is enabled, the LED0_0 pin becomes the serial data pin, and the LED1_0

pin becomes the serial clock pin. All other LED pins can still be configured normally. The serial LED mode clocks the 48 LED status bits on the rising edge of the serial clock where bits 25:48 are ignored.

The LED behavior settings can also be used in serial LED mode. The controls are used on a per-PHY basis, where the LED combine and LED blink or pulse-stretch setting of LED0_n for each PHY is used to control the behavior of each bit of the serial LED stream for each corresponding PHY. To configure LED behavior, set device register 30.

The following table shows the 48-bit serial output bitstream of each LED signal where bits 25:48 are ignored. The individual signals can be clocked in the following order.

Table 30 • **LED Serial Bitstream Order**

Output	PHY0	PHY1
Link/activity	1	13
Link1000/activity	2	14
Link100/activity	3	15
Link10/activity	4	16
Fiber link/activity	5	17
Duplex/collision	6	18
Collision	7	19
Activity	8	20
Fiber activity	9	21
Tx activity	10	22
Rx activity	11	23
Autonegotiation fault	12	24

3.16.5 Enhanced Serial LED Mode

VSC8572-02 can be configured to output up to four LED signals per port on a serial stream that can be de-serialized externally to drive LEDs on the system board. In enhanced serial LED mode, the port 0 and port 1 LED output pins serve the following functions:

- LED0_0/LED0_1: LED_DATA
- LED1_0/LED1_1: LED_CLK
- LED2_0/LED2_1: LED_LD
- LED3_0/LED3_1: LED_PULSE

The serial LED_DATA is shifted out on the falling edge of LED_CLK and is latched in the external serial-to-parallel converter on the rising edge of LED_CLK. The falling edge of LED_LD signal can be used to shift the data from the shift register in the converter to the parallel output drive register. When a separate parallel output drive register is not used in the external serial-to-parallel converter, the LEDs will blink at a high frequency as the data bits are being shifted through, which may be undesirable. LED pin functionality is controlled by setting register 25G, bits 7:1.

The LED_PULSE signal provides a 5 kHz pulse stream whose duty cycle can be modulated to turn on/off LEDs at a high rate. This signal can be tied to the output enable signal of the serial-to-parallel converter to provide the LED dimming functionality to save energy. The LED_PULSE duty cycle is controlled by setting register 25G, bits 15:8.

3.16.6 LED Port Swapping

For additional hardware configurations, the VSC8572-02 can have its LED port order swapped. This is a useful feature to help simplify PCB layout design. Register 25G bit 0 controls the LED port swapping mode.

Note: LED port swapping only applies to the parallel LED outputs and does not affect the serial LED outputs.

3.17 Fast Link Failure Indication

To aid Synchronous Ethernet applications, the VSC8572-02 can indicate the onset of a link failure in less than 1 ms (worst-case <3 ms). By comparison, the IEEE 802.3 standard establishes a delay of up to 750 ms before indicating that a 1000BASE-T link is no longer present. A fast link failure indication is critical to support ports used in a synchronization timing link application. The fast link failure indication works for all copper media speeds, but not for fiber media. Fast link failure is supported for each PHY port through the GPIO9/FASTLINK-FAIL pin. For details on how to use the FASTLINK-FAIL pin, see Table 72, page 114 and Table 108, page 133.

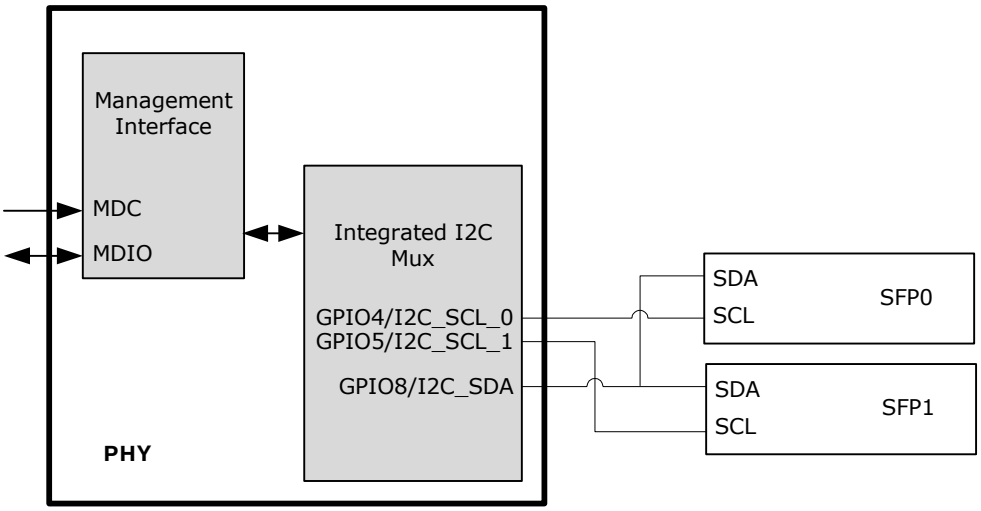
Note: For all links except 1000BASE-T, the fast link failure indication matches the link status register (address 1, bit 2). For 1000BASE-T links, the link failure is based on a circuit that analyzes the integrity of the link, and at the indication of failure, will assert.

Note: The Fast Link Failure Indication should not be used when EEE is enabled on a link.

3.18 Integrated Two-Wire Serial Multiplexer

The VSC8572-02 includes an integrated dual two-wire serial multiplexer (MUX), eliminating the need for an external two-wire serial device for the control and status of SFP or PoE modules. There are three two-wire serial controller pins: two clocks and one shared data pin. Each SFP or PoE connects to the multipurpose GPIO[5:4]_I2C_SCL_[1:0] and GPIO8/I2C_SDA device pins, which must be configured to the corresponding two-wire serial function. For more information about configuring the pins, see Two-Wire Serial MUX Control 1, page 133. For SFP modules, VSC8572-02 can also provide control for the MODULE_DETECT and TX_DIS module pins using the multipurpose LED and GPIO pins.

Figure 71 • **Two-Wire Serial MUX with SFP Control and Status**



3.18.1 Read/Write Access Using the Two-Wire Serial MUX

Using the integrated two-wire serial MUX, the VSC8572-02 device can read and write to an SFP or PoE module through the SCL and SDA pins. If the ability is required to write to the slave two-wire serial device, refer to the device's specific datasheet for more information.

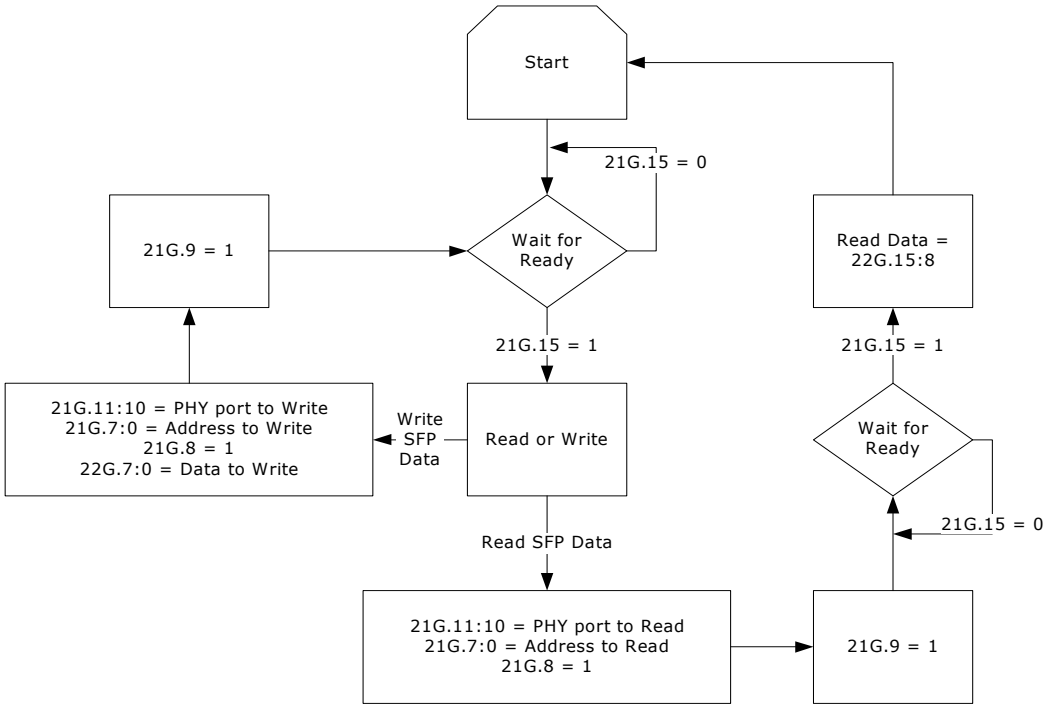
Note: The VSC8572-02 device does not automatically increment the two-wire serial address. Each desired address must be intentionally set.

Main control of the integrated two-wire serial MUX is available through register 20G. The two-wire serial MUX pins are enabled or disabled using register 20G bits 3:0. Register 20G bits 15:9 set the two-wire serial device address (the default is 0xA0). Using register 20G bits 5:4, the two-wire serial frequency can be changed from 100 kHz to other speeds, such as 50 kHz, 100 kHz (the default), 400 kHz, and 2 MHz.

Note: The frequencies listed are available when the PHY API is used.

Registers 21G and 22G provide status and control of the read/write process. The following illustration shows the read and write register flow.

Figure 72 • **Two-Wire Serial MUX Read and Write Register Flow**



To read a value from a specific address of the two-wire serial slave device:

1. Read the VSC8572-02 device register 21G bit 15, and ensure that it is set.
2. Write the PHY port address to be read to register 21G bits 11:10.
3. Write the two-wire serial address to be read to register 21G bits 7:0.
4. Set both register 21G bits 8 and 9 to 1.
5. When register 21G bit 15 changes to 1, read the 8-bit data value found at register 22G bits 15:8. This is the contents of the address just read by the PHY.

To write a value to a specific address of the two-wire serial slave device:

1. Read the VSC8572-02 device register 21G bit 15 and ensure that it is set.
2. Write the PHY port address to be written to register 21G bits 11:10.
3. Write the address to be written to register 21G bits 7:0.
4. Set register 21 bit 8 to 0.
5. Set register 22G bits 7:0 with the 8-bit value to be written to the slave device.
6. Set register 21G bit 9 to 1.

To avoid collisions during read and write transactions on the two-wire serial bus, always wait until register 21G bit 15 changes to 1 before performing another two-wire serial read or write operation.

3.19 GPIO Pins

The VSC8572-02 provides 6 multiplexed general purpose input/output (GPIO) pins. All device GPIO pins and their behavior are controlled using registers. The following table shows an overview of the register controls for GPIO pins. For more information, see General Purpose Registers, page 128.

Table 31 • **Register Bits for GPIO Control and Status**

GPIO Pin	GPIO_ctrl	GPIO Input	GPIO Output	GPIO Output Enable
GPIO0/SIGDET0	13G.1:0	15G.0	16G.0	17G.0
GPIO1/SIGDET1	13G.3:2	15G.1	16G.1	17G.1

Table 31 • **Register Bits for GPIO Control and Status *(continued)***

GPIO Pin	GPIO_ctrl	GPIO Input	GPIO Output	GPIO Output Enable
GPIO4/I2C_SCL_0	13G.9:8.	15G.4	16G.4	17G.4
GPIO5/I2C_SCL_1	13G.11:10	15G.5	16G.5	17G.5
GPIO8/I2C_SDA	14G.1:0	15G.8	16G.8	17G.8
GPIO9/FASTLINK_FAIL	14G.3:2	15G.9	16G.9	17G.9
GPIO10/1588_LOAD_SAVE	14G.5:4	15G.10	16G.10	17G.10
GPIO11	14G.7:6	15G.11	16G.11	17G.11
GPIO12/1588_SPI_CS	14G.15:14	15G.12	16G.12	17G.12
GPIO13/1588_SPI_DO	14G.15:14	15G.13	16G.13	17G.13

3.20 Testing Features

The VSC8572-02 device includes several testing features designed to facilitate performing system-level debugging and in-system production testing. This section describes the available features.

3.20.1 Ethernet Packet Generator

The Ethernet packet generator (EPG) can be used at each of the 10/100/1000BASE-T speed settings for copper Cat5 media and fiber media to isolate problems between the MAC and the VSC8572-02, or between a locally connected PHY and its remote link partner. Enabling the EPG feature effectively disables all MAC interface transmit pins and selects the EPG as the source for all data transmitted onto the twisted pair interface. This feature is not used when the SerDes media is set to pass-through mode.

Important The EPG is intended for use with laboratory or in-system testing equipment only. Do not use the EPG testing feature when the VSC8572-02 is connected to a live network.

To enable the VSC8572-02 EPG feature, set the device register bit 29E1.15 to 1.

When the EPG is enabled, packet loss occurs during transmission of packets from the MAC to the PHY. However, the PHY receive output pins to the MAC are still active when the EPG is enabled. When it is necessary to disable the MAC receive pins as well, set the register bit 0.10 to 1.

When the device register bit 29E1.14 is set to 1, the PHY begins transmitting Ethernet packets based on the settings in registers 29E1 and 30E1. These registers set:

- Source and destination addresses for each packet
- Packet size
- Interpacket gap
- FCS state
- Transmit duration
- Payload pattern

When register bit 29E1.13 is set to 0, register bit 29E1.14 is cleared automatically after 30,000,000 packets are transmitted.

3.20.2 CRC Counters

Two sets of cyclical redundancy check (CRC) counters are available in all PHYs in VSC8572-02. One set monitors traffic on the copper interface and the other set monitors traffic on the SerDes interface.

The device CRC counters operate in the 100BASE-FX/1000BASE-X over SerDes mode as well as in the 10/100/1000BASE-T mode as follows:

After receiving a packet on the media interface, register bit 15 in register 18E1 or register 28E3 is set and cleared after being read.

The packet then is counted by either the good CRC counter or the bad CRC counter.

Both CRC counters are also automatically cleared when read.

The good CRC counter's highest value is 9,999 packets. After this value is reached, the counter clears on the 10,000th packet and continues to count additional packets beyond that value. The bad CRC counter stops counting when it reaches its maximum counter limit of 255 packets.

3.20.2.1 Copper Interface CRC Counters

Two separate CRC counters are available and reside between the copper interface PCSs and SerDes MAC interface. There is a 14-bit good CRC counter available through register bits 18E1.13:0 and a separate 8-bit bad CRC counter available in register bits 23E1.7:0.

3.20.2.2 SerDes Interface CRC Counters

Two separate CRC counters are available and reside between the SerDes media interface PCSs and SerDes MAC interface. There is a 14-bit good CRC counter available through register bits 28E3.13:0 and a separate 8-bit bad CRC counter available in register bits 29E3.7:0.

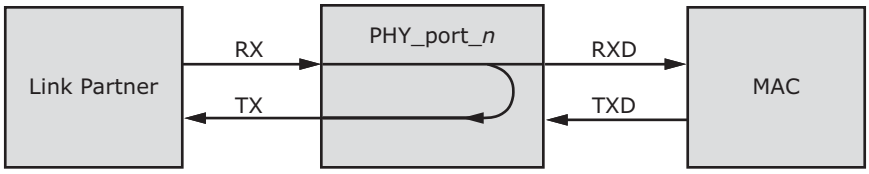
3.20.2.3 SerDes Fiber Media Transmit Counters

Two fiber media transmit counters are available to verify packets being transmitted on the fiber media. Register bits 21E3.13:0 are the good CRC packet counters and register bits 22E3.7:0 are the CRC error counters.

3.20.3 Far-End Loopback

The far-end loopback testing feature is enabled by setting register bit 23.3 to 1. When enabled, it forces incoming data from a link partner on the current media interface, into the MAC interface of the PHY, to be retransmitted back to the link partner on the media interface as shown in the following illustration. In addition, the incoming data also appears on the receive data pins of the MAC interface. Data present on the transmit data pins of the MAC interface is ignored when using this testing feature.

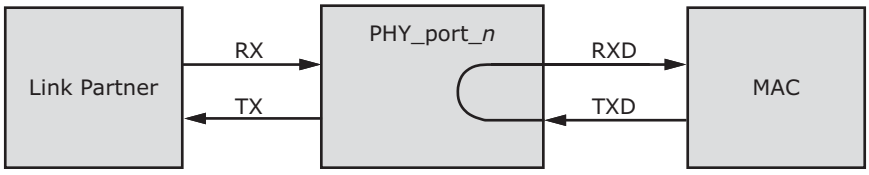
Figure 73 • **Far-End Loopback Diagram**



3.20.4 Near-End Loopback

When the near-end loopback testing feature is enabled, transmitted data (TXD) is looped back in the PCS block onto the receive data signals (RXD), as shown in the following illustration. When using this testing feature, no data is transmitted over the network. To enable near-end loopback, set the device register bit 0.14 to 1.

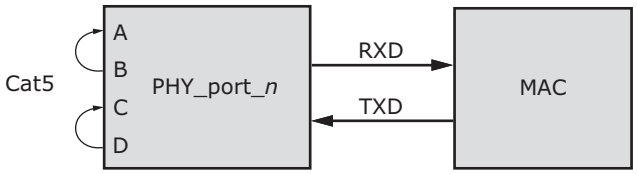
Figure 74 • **Near-End Loopback Diagram**



3.20.5 Connector Loopback

The connector loopback testing feature allows the twisted pair interface to be looped back externally. When using this feature, the PHY must be connected to a loopback connector or a loopback cable. Connect pair A to pair B, and pair C to pair D, as shown in the following illustration. The connector loopback feature functions at all available interface speeds.

Figure 75 • **Connector Loopback Diagram**



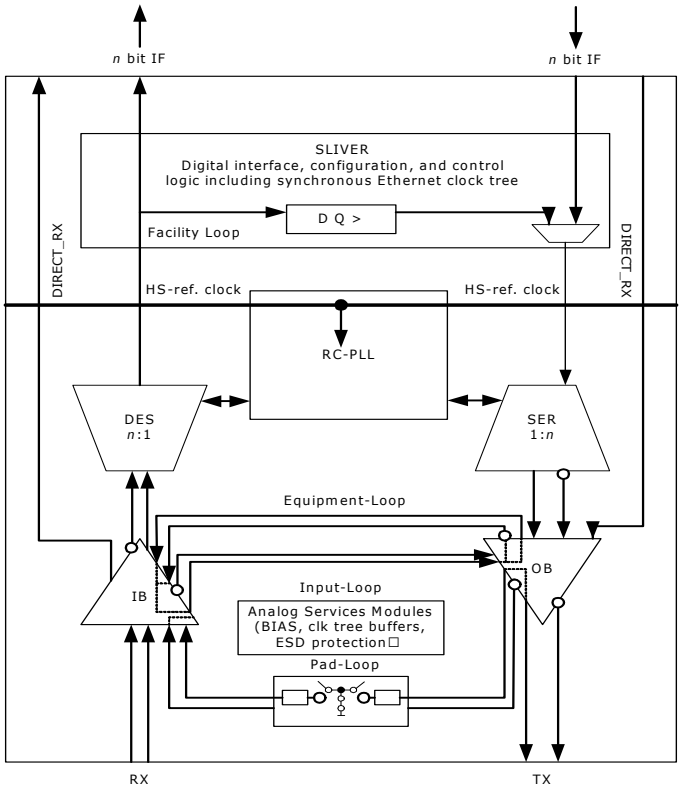
When using the connector loopback testing feature, the device autonegotiation, speed, and duplex configuration is set using device registers 0, 4, and 9. For 1000BASE-T connector loopback, the following additional writes are required. Execute the additional writes in the following order:

1. Enable the 1000BASE-T connector loopback. Set register bit 24.0 to 1.
2. Disable pair swap correction. Set register bit 18.5 to 1.

3.20.6 SerDes Loopbacks

For test purposes, the SerDes and SerDes macro interfaces provides several data loops. The following illustration shows the SerDes loopbacks.

Figure 76 • **Data Loops of the SerDes Macro**



3.20.6.1 RGMII/SGMII Mode

When the MAC interface is configured in RGMII/SGMII mode, write the following 16-bit value to register 18G:

Bits 15:12 0x9

Bits 11:8: Port address (0x0 to 0x3)

Bits 7:4: Loopback type

Bits 3:0: 0x2

where loopback type is:

0x0: No loopback

0x2: Input loopback

0x4: Facility loopback

0x8: Equipment loopback

3.20.6.2 QSGMII Mode

When the MAC interface is configured in QSGMII mode, write the following 16-bit value to register 18G:

Bits 15:12 0x9

Bits 11:8: Port address (0x0)

Bits 7:4: Loopback type

Bits 3:0: 0x2

where loopback type is:

0x0: No loopback

0x2: Input loopback

0x4: Facility loopback

0x8: Equipment loopback

Note: Loopback configuration affects all ports associated with a QSGMII. Individual port loopback within a QSGMII is not possible.

3.20.6.3 Fiber Media Port Mode

When the SerDes is configured as a fiber media port, write the following 16-bit value to register 18G:

Bits 15:12: 0x8

Bits 11:8: Port address

Bits 7:4: Loopback type

Bits 3:0: 0x2

where port address is:

0x1: Fiber0 port

0x2: Fiber1 port

0x4: Fiber2 port

0x8: Fiber3 port

Port addresses for fiber media SerDes can be OR'ed together to address multiple ports using a single command. bit 18G.15 will be cleared when the internal configuration is complete.

3.20.6.4 Facility Loop

The recovered and de-multiplexer deserializer data output is looped back to the serializer data input and replaces the data delivered by the digital core. This test loop provides the possibility to test the complete analog macro data path from outside including input buffer, clock and data recovery, serialization and output buffer. The data received by the input buffer must be transmitted by the output buffer after some delay.

Additional configuration of the macro is required for facility loopback mode. When entering facility loopback mode, the set = 1 option should be run; when exiting facility loopback mode, the set = 0 option should be run.

```
PhyWrite(PhyBaseAddr, 31, 0x0010);
PhyWrite(PhyBaseAddr, 18, 0x8s03);
// where “s” is the physical address of the SerDes macro
```

```
PhyWrite(PhyBaseAddr, 18, 0xd7cb);
PhyWrite(PhyBaseAddr, 18, 0x8007);
tmp1 = PhyRead(PhyBaseAddr, 18);
tmp2 = tmp1 & 0x0ff0;
if (set)
    tmp3 = tmp2 | 0x0010;
else
    tmp3 = tmp2 & 0x0fe0;
tmp4 = tmp3 | 0x8006;
PhyWrite(PhyBaseAddr, 18, tmp4);
if (SGMII)
    PhyWrite(PhyBaseAddr, 18, 0x9p40);
// where “p” is the logical address of the SGMII interface
else
    PhyWrite(PhyBaseAddr, 18, 0x8p40);
// where “p” is the logical address of the Fiber media interface
// PhyBaseAddr is the 5-bit base address of the internal PHYs.
// The upper 3 bits are set by the PHYADD[4:2] pins and the
// lower 2 bits are 0.
```

Additional configuration of the enhanced SerDes macro is required for facility loopback mode. When entering facility loopback mode, the set = 1 option should be run; when exiting facility loopback mode, the set = 0 option should be run.

```
PhyWrite(PhyBaseAddr, 31, 0x0010);
PhyWrite(PhyBaseAddr, 18, 0x8013);
PhyWrite(PhyBaseAddr, 18, 0xd7cb);
PhyWrite(PhyBaseAddr, 18, 0x8007);
tmp1 = PhyRead(PhyBaseAddr, 18);
tmp2 = tmp1 & 0x0ff0;
if (set)
    tmp3 = tmp2 | 0x0100;
else
    tmp3 = tmp2 & 0x0ef0;
tmp4 = tmp3 | 0x8006;
PhyWrite(PhyBaseAddr, 18, tmp4);
PhyWrite(PhyBaseAddr, 18, 0x9c40);
// PhyBaseAddr is the 5-bit base address of the internal PHYs.
// The upper 3 bits are set by the PHYADD[4:2] pins and the
// lower 2 bits are 0.
```

3.20.6.5 Equipment Loop

The 1-bit data stream at the serializer output is looped back to the deserializer and replaces the received data stream from the input buffer. This test loop provides the possibility to verify the digital data path internally. The transmit data goes through the serialization, the clock and data recovery and deserialization before the data is fed back to the digital core.

Note: After entering equipment loopback mode, the following workaround should be run with set = 1 option in case external signal is not present; when exiting equipment loopback mode, the set = 0 option should be run:

SGMII/QSGMII SerDes

```
PhyWrite(<phy>, 31, 0x52b5);
PhyWrite(<phy>, 16, 0xa68c);
tmp17 = PhyRead(<phy>,17);
if (set)
tmp17 |= 0x0010; //Set SigDet as desired, Set bit 4
else // clear SigDet
tmp17 &= 0xffef; //Clear SigDet, bit 4
PhyWrite(<phy>, 17, tmp17);
```

```
PhyWrite(<phy>, 16, 0x868c);
PhyWrite(<phy>, 31, 0x0);
```

Fiber media SerDes

```
PhyWrite(<phy>, 31, 0x52b5);
PhyWrite(<phy>, 16, 0xa68a);
tmp17 = PhyRead(<phy>,17);
if (set)
tmp17 |= 0x0010; //Set SigDet as desired, Set bit 4
else // clear SigDet
tmp17 &= 0xffef; //Clear SigDet, bit 4
PhyWrite(<phy>, 17, tmp17);
PhyWrite(<phy>, 16, 0x868a);
PhyWrite(<phy>, 31, 0x0);
```

3.20.6.6 Input Loop

The received 1-bit data stream of the input buffer is looped back asynchronously to the output buffer. This test loop provides the possibility to test only the analog parts of the RGMII/SGMII interface because only the input and output buffer are part of this loop.

Note: When the enhanced SerDes macro is in input loopback, the output is inverted relative to the input.

The following table shows the SerDes macro address map.

Table 32 • **SerDes Macro Address Map**

SerDes Macro	Physical Address (s)	Interface Logical Type (p)	Address
SerDes0	0x0	Fiber0	0x1
SerDes1	0x1	SGMII1	0x1
SerDes2	0x2	Fiber1	0x2
SerDes3	0x3	SGMII2	0x2
SerDes4	0x4	Fiber2	0x4
SerDes5	0x5	SGMII3	0x3
SerDes6	0x6	Fiber3	0x8

3.20.7 VeriPHY Cable Diagnostics

The VSC8572-02 includes a comprehensive suite of cable diagnostic functions that are available using SMI reads and writes. These functions enable a variety of cable operating conditions and status to be accessed and checked. The VeriPHY suite has the ability to identify the cable length and operating conditions and to isolate a variety of common faults that can occur on the Cat5 twisted pair cabling.

Note: When a link is established on the twisted pair interface in the 1000BASE-T mode, VeriPHY can run without disrupting the link or disrupting any data transfer. However, when a link is established in 100BASE-TX or 10BASE-T modes, VeriPHY causes the link to drop while the diagnostics are running. After diagnostics are finished, the link is re-established.

The following diagnostic functions are part of the VeriPHY suite:

- Detecting coupling between cable pairs
- Detecting cable pair termination
- Determining cable length

3.20.7.1 Coupling Between Cable Pairs

Shorted wires, improper termination, or high crosstalk resulting from an incorrect wire map can cause error conditions, such as anomalous coupling between cable pairs. These conditions can prevent the device from establishing a link in any speed.

3.20.7.2 Cable Pair Termination

Proper termination of Cat5 cable requires a 100 Ω differential impedance between the positive and negative cable terminals. IEEE 802.3 allows for a termination of 115 Ω maximum and 85 Ω minimum. VeriPHY diagnostics can report anomalous termination that falls outside of this range. The diagnostics can also determine the presence of an open or shorted cable pair.

3.20.7.3 Cable Length

When the Cat5 cable in an installation is properly terminated, VeriPHY reports the approximate cable length in meters. If there is a cable fault the distance to the fault is reported.

3.20.7.4 Mean Square Error Noise

The average absolute error can be read out when either a 100BASE-TX or 1000BASE-T link is established. In the case of 1000BASE-T link, there are two average absolute error terms, one for each twisted pair over which signal is received. Use the following script to read average absolute error for 100BASE-TX:

```
PhyWrite(<phy>, 31, 0x52b5);
PhyWrite(<phy>, 16, 0xa3c0);
PhyRead(<phy>, 16);
tmp17 = PhyRead(<phy>, 17);
tmp18 = PhyRead(<phy>, 18);
mse = (tmp18 << 4) | (tmp17 >> 12);
PhyWrite(<phy>, 31, 0);
```

The returned average absolute error is in units of 1/2,048 and can be found in the mse variable.

```
PhyWrite(<phy>, 31, 0x52b5);
PhyWrite(<phy>, 16, 0xa3c0);
PhyRead(<phy>, 16);
tmp17 = PhyRead(<phy>, 17);
tmp18 = PhyRead(<phy>, 18);
mseA = (tmp18 << 4) | (tmp17 >> 12);
mseB = tmp17 & 0x0fff;
PhyWrite(<phy>, 16, 0xa3c2);
PhyRead(<phy>, 16);
tmp17 = PhyRead(<phy>, 17);
tmp18 = PhyRead(<phy>, 18);
mseC = (tmp18 << 4) | (tmp17 >> 12);
mseD = tmp17 & 0x0fff;
PhyWrite(<phy>, 31, 0);
```

The returned average absolute error is in units of 1/2,048 and can be found in the mseA, mseB, mseC, and mseD variables for each twisted pair.

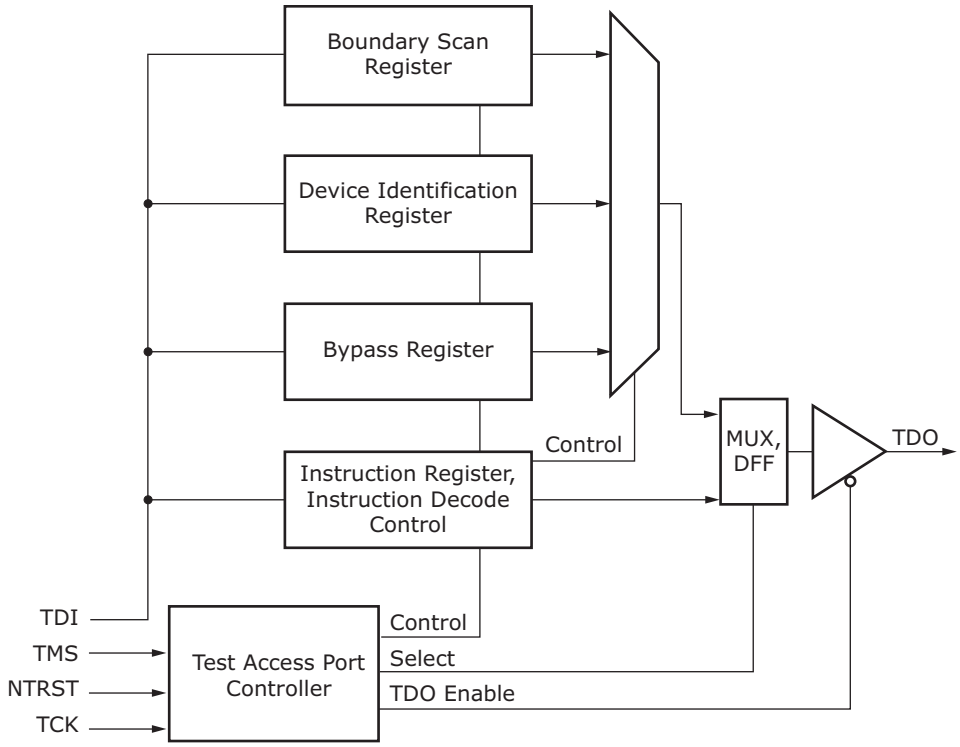
3.20.8 JTAG Boundary Scan

The VSC8572-02 supports the test access port (TAP) and boundary scan architecture described in IEEE 1149.1. The device includes an IEEE 1149.1-compliant test interface, referred to as a JTAG TAP interface.

The JTAG boundary scan logic on the VSC8572-02, accessed using its TAP interface, consists of a boundary scan register and other logic control blocks. The TAP controller includes all IEEE-required signals (TMS, TCK, TDI, and TDO), in addition to the optional asynchronous reset signal TRST. The following illustration shows the TAP and boundary scan architecture.

Important When JTAG is not in use, the TRST pin must be tied to ground with a pull-down resistor for normal operation.

Figure 77 • **Test Access Port and Boundary Scan Architecture**



After a TAP reset, the device identification register is serially connected between TDI and TDO by default. The TAP instruction register is loaded either from a shift register when a new instruction is shifted in, or, if there is no new instruction in the shift register, a default value of 6'b100000 (IDCODE) is loaded. Using this method, there is always a valid code in the instruction register, and the problem of toggling instruction bits during a shift is avoided. Unused codes are mapped to the BYPASS instruction.

3.20.9 JTAG Instruction Codes

The VSC8572-02 supports the following instruction codes:

Table 33 • **JTAG Instruction Codes**

Instruction Code	Description
BYPASS	The bypass register contains a single shift-register stage and is used to provide a minimum-length serial path (one TCK clock period) between TDI and TDO to bypass the device when no test operation is required.
CLAMP	Allows the state of the signals driven from the component pins to be determined from the boundary scan register while the bypass register is selected as the serial path between TDI and TDO. While the CLAMP instruction is selected, the signals driven from the component pins do not change.
EXTEST	Allows tests of the off-chip circuitry and board-level interconnections by sampling input pins and loading data onto output pins. Outputs are driven by the contents of the boundary scan cells, which have to be updated with valid values, with the PRELOAD instruction, prior to the EXTEST instruction.

Table 33 • **JTAG Instruction Codes *(continued)***

Instruction Code	Description
HIGHZ	Places the component in a state in which all of its system logic outputs are placed in a high-impedance state. In this state, an in-circuit test system can drive signals onto the connections normally driven by a component output without incurring a risk of damage to the component. This makes it possible to use a board where not all of the components are compatible with the IEEE 1149.1 standard.
IDCODE	Provides the version number (bits 31:28), device family ID (bits 27:12), and the manufacturer identity (bits 11:1) to be serially read from the device.
SAMPLE/PRELOAD	Allows a snapshot of inputs and outputs during normal system operation to be taken and examined. It also allows data values to be loaded into the boundary scan cells prior to the selection of other boundary scan test instructions.
USERCODE	Provides the version number (bits 31:28), part number (bits 27:12), and the manufacturer identity (bits 11:1) to be serially read from the device.

The following tables provide information about the IDCODE and USERCODE binary values stored in the device JTAG registers.

Table 34 • **IDCODE JTAG Device Identification Register Descriptions**

Description	Device Version	Family ID	Manufacturing Identity	LSB
Bit field	31–28	27–12	11–1	0
Binary value	0000	1000 0101 0111 0100	000 0111 0100	1

Table 35 • **USERCODE JTAG Device Identification Register Descriptions**

Description	Device Version	Model Number	Manufacturing Identity	LSB
Bit field	31–28	27–12	11–1	0
Binary value	0010	1000 0101 0111 0010	000 0111 0100	1

The following table provides information about the location and IEEE compliance of the JTAG instruction codes used in the VSC8572-02. Instructions not explicitly listed in the table are reserved. For more information about these IEEE specifications, visit the IEEE Web site at www.IEEE.org.

Table 36 • **JTAG Instruction Code IEEE Compliance**

Instruction	Code	Selected Register	Register Width	IEEE 1149.1
EXTEST	6'b000000	Boundary Scan	161	Mandatory
SAMPLE/PRELOAD	6'b000001	Boundary Scan	161	Mandatory
IDCODE	6'b100000	Device Identification	32	Optional
USERCODE	6'b100101	Device Identification	32	Optional
CLAMP	6'b000010	Bypass Register	1	Optional
HIGHZ	6'b000101	Bypass Register	1	Optional
BYPASS	6'b111111	Bypass Register	1	Mandatory

3.20.10 Boundary Scan Register Cell Order

All inputs and outputs are observed in the boundary scan register cells. All outputs are additionally driven by the contents of boundary scan register cells. Bidirectional pins have all three related boundary scan register cells: input, output, and control.

The complete boundary scan cell order is available as a BSDL file format on the Microsemi Web site at www.Microsemi.com.

3.21 100BASE-FX Halt Code Transmission and Reception

The VSC8572-02 device supports transmission and reception of halt code words in 100BASE-FX mode. There are three separate scripts provided to initiate transmission of halt code words, stop transmission of halt code words and detect reception of halt code words. Use the following scripts to implement each of these functions:

Sending the HALT codeword:

```
PhyWrite(<phy>, 31, 0x52b5);
PhyWrite(<phy>, 16, 0xac82);
reg18 = PhyRead(<phy>, 18);
reg18 = (reg18 & 0xf0) | 0x0c;
PhyWrite(<phy>, 18, reg18);
PhyWrite(<phy>, 17, 0xe739);
PhyWrite(<phy>, 16, 0x8c82);

PhyWrite< <phy>, 16, 0xbe80);
reg17 = PhyRead(<phy>, 17);
reg18 = PhyRead(<phy>, 18);
reg17 = reg17 | 0x0040;
PhyWrite(<phy>, 18, reg18);
PhyWrite(<phy>, 17, reg17);
PhyWrite(<phy>, 16, 0x9e80);
PhyWrite(<phy>, 31, 0);
```

Stop sending the HALT codeword:

```
PhyWrite(<phy>, 31, 0x52b5);
PhyWrite< <phy>, 16, 0xbe80);
reg17 = PhyRead(<phy>, 17);
reg18 = PhyRead(<phy>, 18);
reg17 = reg17 & ~0x0040;
PhyWrite(<phy>, 18, reg18);
PhyWrite(<phy>, 17, reg17);
PhyWrite(<phy>, 16, 0x9e80);
PhyWrite(<phy>, 31, 0);
```

Detecting whether the HALT codeword is being sent by the link partner:

```
long patternset[5] = {
     0xce739,
     0xe739c,
     0x739ce,
     0x39ce7,
     0x9ce73
};
```

Turning on the pattern checker:

```
PhyWrite(<phy>, 31, 0x52b5);
PhyWrite(<phy>, 16, 0xbe80);
reg18 = PhyRead(<phy>, 18);
reg17 = PhyRead(<phy>, 17);
reg17 = reg17 | 4;
```

```
PhyWrite(<phy>, 18, reg18);
PhyWrite(<phy>, 17, reg17);
PhyWrite(<phy>, 16, 0x9e80);
```

Sweeping through all five pattern shifts checking for a match:

```
for (i = 0, matchfailed = 1; i < 5 && matchfailed; ++i) {
PhyWrite(<phy>, 16, 0xac84);
reg18 = PhyRead(<phy>, 18);
reg18 = (reg18 & 0xf0) | (patternset[i] >> 16)
PhyWrite(<phy>, 18, reg18);
PhyWrite(<phy>, 17, patternset[i] & 0xffff);
PhyWrite(<phy>, 16, 0x8c84);

PhyWrite(<phy>, 16, 0xbe84); // Dummy read to clear latched mismatch
PhyWrite(<phy>, 16, 0xbe84); // Read pattern check failure status
matchfailed = PhyRead(<phy>, 17) & 1; // Extract pattern check failure status
}
```

Turning off the pattern checker:

```
PhyWrite(<phy>, 16, 0xbe80);
reg18 = PhyRead(<phy>, 18);
reg17 = PhyRead(<phy>, 17);
reg17 = reg17 & ~4;
PhyWrite(<phy>, 18, reg18);
PhyWrite(<phy>, 17, reg17);
PhyWrite(<phy>, 16, 0x9e80);

PhyWrite(<phy>, 31, 0);

HALT_codeword_detected =!matchfailed;
```

3.22 Configuration

The VSC8572-02 can be configured by setting internal memory registers using the management interface. To configure the device, perform the following steps:

1. COMA_MODE active, drive high (optional).
2. Apply power.
3. Apply RefCLK and IEEE 1588 reference clock.
4. Release reset, drive high. Power and clock must be stable before releasing reset.
5. Wait 120 ms minimum.
6. Apply patch from PHY_API (required for production released optional for board testing).
7. Configure register 19G for MAC mode (to access register 19G, register 31 must be 0x10). Read register 19G. Set bits 15:14, MAC configuration as follows:
   00: SGMII
   01: QSGMII
   10: RGMII
   11: Reserved
   Write new register 19G.
8. Set RGMII (optional)

Table 37 • **Register 18E2 Settings for RGMII**

Bit	Name	Setting
6:4	rgmii_skew_tx	000
3:1	rgmii_skew_rx	000
0	rgmii_bit_rev	0

9. Configure register 18G for MAC on all 4 PHYs write:
   SGMII: 0x80F0
   QSGMII: 0x80E0
   RGMII: set 19G[15:14] = 0x10 to set PHY0 and PHY1 MAC to be RGMII
10. Read register 18G until bit 15 equals 0.
11. If Fiber Media on all 4 PHYs configure register 18G by writing:
    Media 1000BASE-X: 0x8FC1
    Media 100BASE-FX: 0x8FD1
12. If Fiber Media read register 18G till bit 15 equals 0.
13. Configure register 23 for MAC and Media mode (to access register 23, register 31 must be 0). Read register 23. Set bits 10:8 as follows:
    000: Copper
    010: 1000BASE-X
    011: 100BASE-FX
    Write new register 23.
14. Software reset. Read register 0 (to access register 0, register 31 must be 0). Set bit 15 to 1. Write new register 0.
15. Read register 0 until bit 15 equals 0.
16. Release the COMA_MODE pin, drive low (only necessary if COMA_MODE pin is driven high or unconnected).

Note: All MAC interfaces must be the same — all QSGMII, RGMII, or SGMII.

3.22.1 Initialization

The COMA_MODE pin provides an optional feature that may be used to control when the PHYs become active. The typical usage is to keep the PHYs from becoming active before they have been fully initialized. For more information, see Configuration, page 92. By not being active until after complete initialization keeps links from going up and down. Alternatively the COMA_MODE pin may be connected low (ground) and the PHYs will be fully active once out of reset.

4 Registers

This section provides information about how to configure the VSC8572-02 using its internal memory registers and the management interface. The registers marked reserved and factory test should not be read or written to, because doing so may produce undesired effects.

The default value documented for registers is based on the value at reset; however, in some cases, that value may change immediately after reset.

The access type for each register is shown using the following abbreviations:

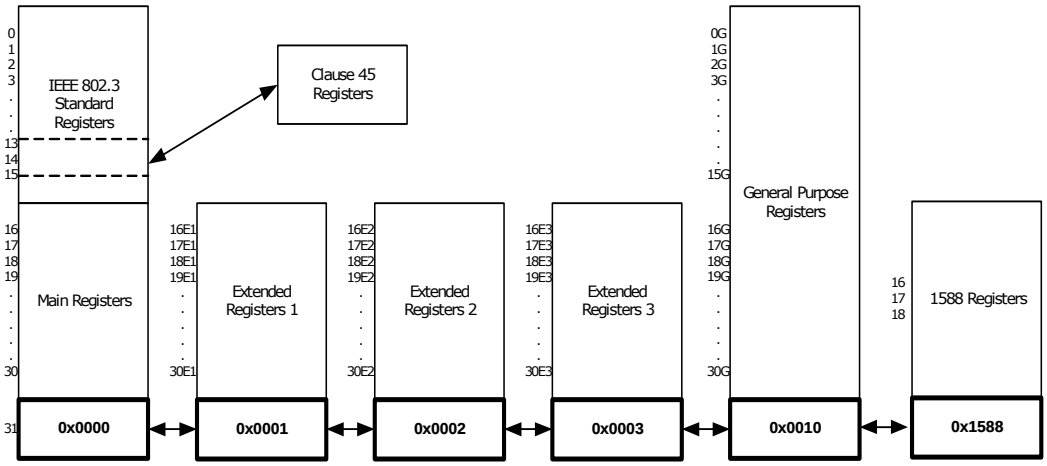
- RO: Read Only
- ROCR: Read Only, Clear on Read
- RO/LH: Read Only, Latch High
- RO/LL: Read Only, Latch Low
- R/W: Read and Write
- RWSC: Read Write Self Clearing

The VSC8572-02 uses several different types of registers:

- IEEE Clause 22 device registers with addresses from 0 to 31
- Three pages of extended registers with addresses from 16E1–30E1, 16E2–30E2, and 16E3–30E3
- General-purpose registers with addresses from 0G to 30G
- IEEE Clause 45 devices registers accessible through the Clause 22 registers 13 and 14 to support IEEE 802.3az-2010 energy efficient Ethernet registers

The following illustration shows the relationship between the device registers and their address spaces.

Figure 78 • **Register Space Diagram**



Reserved Registers—For main registers 16–31, extended registers 16E1–30E1, 16E2–30E2, 16E3–30E3, and general purpose registers 0G–30G, any bits marked as Reserved should be processed as read-only and their states as undefined.

Reserved Bits—In writing to registers with reserved bits, use a read-modify-then-write technique, where the entire register is read but only the intended bits to be changed are modified. Reserved bits cannot be changed and their read state cannot be considered static or unchanging.

4.1 Register and Bit Conventions

Registers are referred to by their address and bit number in decimal notation. A range of bits is indicated with a colon. For example, a reference to address 26, bits 15 through 14 is shown as 26.15:14.

A register with an E and a number attached (example 27E1) means it is a register contained within extended register page number 1. A register with a G attached (example 13G) means it is a GPIO page register.

Bit numbering follows the IEEE standard with bit 15 being the most significant bit and bit 0 being the least significant bit.

4.2 IEEE 802.3 and Main Registers

In the VSC8572-02, the page space of the standard registers consists of the IEEE 802.3 standard registers and the Microsemi standard registers. The following table lists the names of the registers associated with the addresses as specified by IEEE 802.3.

Table 38 • **IEEE 802.3 Registers**

Address	Name
0	Mode Control
1	Mode Status
2	PHY Identifier 1
3	PHY Identifier 2
4	Autonegotiation Advertisement
5	Autonegotiation Link Partner Ability
6	Autonegotiation Expansion
7	Autonegotiation Next-Page Transmit
8	Autonegotiation Link Partner Next-Page Receive
9	1000BASE-T Control
10	1000BASE-T Status
11–12	Reserved
13	Clause 45 Access Registers from IEEE 802.3 Table 22-6 and 22.24.3.11-12 and Annex 22D
14	Clause 45 Access Registers from IEEE 802.3 Table 22-6 and 22.24.3.11-12 and Annex 22D
15	1000BASE-T Status Extension 1

The following table lists the names of the registers in the main page space of the device. These registers are accessible only when register address 31 is set to 0x0000.

Table 39 • **Main Registers**

Address	Name
16	100BASE-TX status extension
17	1000BASE-T status extension 2
18	Bypass control
19	Error Counter 1
20	Error Counter 2
21	Error Counter 3
22	Extended control and status
23	Extended PHY control 1

Table 39 • **Main Registers *(continued)***

Address	Name
24	Extended PHY control 2
25	Interrupt mask
26	Interrupt status
27	Reserved
28	Auxiliary control and status
29	LED mode select
30	LED behavior
31	Extended register page access

4.2.1 Mode Control

The device register at memory address 0 controls several aspects of VSC8572-02 functionality. The following table shows the available bit settings in this register and what they control.

Table 40 • **Mode Control, Address 0 (0x00)**

Bit	Name	Access	Description	Default
15	Software reset	R/W	Self-clearing. Restores all serial management interface (SMI) registers to default state, except for sticky and super-sticky bits. 1: Reset asserted. 0: Reset de-asserted. Wait [X] after setting this bit to initiate another SMI register access.	0
14	Loopback	R/W	1: Loopback enabled. 0: Loopback disabled. When loop back is enabled, the device functions at the current speed setting and with the current duplex mode setting (bits 6, 8, and 13 of this register).	0
13	Forced speed selection LSB	R/W	Least significant bit. MSB is bit 6. 00: 10 Mbps. 01: 100 Mbps. 10: 1000 Mbps. 11: Reserved.	0
12	Autonegotiation enable	R/W	1: Autonegotiation enabled. 0: Autonegotiation disabled.	1
11	Power-down	R/W	1: Power-down enabled.	0
10	Isolate	R/W	1: Disconnect the MAC-side interface of the device from the rest of the datapath. Traffic entering the PHY from either the MAC-side or media-side interface will terminate inside the PHY.	0
9	Restart autonegotiation	R/W	Self-clearing bit. 1: Restart autonegotiation on media interface.	0
8	Duplex(1)	R/W	1: Full-duplex. 0: Half-duplex.	0
7	Collision test enable	R/W	1: Collision test enabled.	0

Table 40 • **Mode Control, Address 0 (0x00) *(continued)***

Bit	Name	Access	Description	Default
6	Forced speed selection MSB	R/W	Most significant bit. LSB is bit 13.(2) 00: 10 Mbps. 01: 100 Mbps. 10: 1000 Mbps. 11: Reserved.	10
5	Unidirectional enable	R/W	When bit 0.12 = 1 or bit 0.8 = 0, this bit is ignored. When bit 0.12 = 0 and bit 0.8 = 1, the behavior is as follows: 1: Enable transmit from media independent interface regardless of whether the PHY has determined that a valid link has been established. 0: Enable transmit from media independent interface only when the PHY has determined that a valid link has been established. **Note:** This bit is only applicable in 100BASE-FX and 1000BASE-X fiber media modes.	0
4:0	Reserved		Reserved.	00000

1. Half-duplex is not supported when the 1588 unit is operating.
2. Before selecting the 1000 Mbps forced speed mode, manually configure the PHY as master or slave by setting bit 11 in register 9 (1000BASE-T Control). Each time the link drops, the PHY needs to be powered down manually to enable it to link up again using the master/slave setting specified in register 9.11.

4.2.2 Mode Status

The register at address 1 in the device main registers space allows you to read the currently enabled mode setting. The following table shows possible readouts of this register.

Table 41 • **Mode Status, Address 1 (0x01)**

Bit	Name	Access	Description	Default
15	100BASE-T4 capability	RO	1: 100BASE-T4 capable.	0
14	100BASE-TX FDX capability	RO	1: 100BASE-TX FDX capable.	1
13	100BASE-TX HDX capability	RO	1: 100BASE-TX HDX capable.	1
12	10BASE-T FDX capability	RO	1: 10BASE-T FDX capable.	1
11	10BASE-T HDX capability	RO	1: 10BASE-T HDX capable.	1
10	100BASE-T2 FDX capability	RO	1: 100BASE-T2 FDX capable.	0
9	100BASE-T2 HDX capability	RO	1: 100BASE-T2 HDX capable.	0
8	Extended status enable	RO	1: Extended status information present in register 15.	1

Table 41 • **Mode Status, Address 1 (0x01) *(continued)***

Bit	Name	Access	Description	Default
7	Unidirectional ability	RO	1: PHY able to transmit from media independent interface regardless of whether the PHY has determined that a valid link has been established. 0: PHY able to transmit from media independent interface only when the PHY has determined that a valid link has been established. **Note:** This bit is only applicable to 100BASE-FX and 1000BASE-X fiber media modes.	1
6	Preamble suppression capability	RO	1: MF preamble can be suppressed. 0: MF required.	1
5	Autonegotiation complete	RO	1: Autonegotiation complete.	0
4	Remote fault	RO	Latches high. 1: Far-end fault detected.	0
3	Autonegotiation capability	RO	1: Autonegotiation capable.	1
2	Link status	RO	Latches low. 1: Link is up.	0
1	Jabber detect	RO	Latches high. 1: Jabber condition detected.	0
0	Extended capability	RO	1: Extended register capable.	1

4.2.3 Device Identification

All 16 bits in both register 2 and register 3 in the VSC8572-02 are used to provide information associated with aspects of the device identification. The following tables list the expected readouts.

Table 42 • **Identifier 1, Address 2 (0x02)**

Bit	Name	Access	Description	Default
15:0	Organizationally unique identifier (OUI)	RO	OUI most significant bits (3:18)	0×0007

Table 43 • **Identifier 2, Address 3 (0x03)**

Bit	Name	Access	Description	Default
15:10	OUI	RO	OUI least significant bits (19:24)	000001
9:4	Microsemi model number	RO	VSC8572-02 (0xD)	001101
3:0	Device revision number	RO	See register 30G for the extended revision identification of this device.	0010

4.2.4 Autonegotiation Advertisement

The bits in address 4 in the main registers space control the VSC8572-02 ability to notify other devices of the status of its autonegotiation feature. The following table shows the available settings and readouts.

Table 44 • **Device Autonegotiation Advertisement, Address 4 (0x04)**

Bit	Name	Access	Description	Default
15	Next page transmission request	R/W	1: Request enabled	0
14	Reserved	RO	Reserved	0
13	Transmit remote fault	R/W	1: Enabled	0
12	Reserved	R/W	Reserved	0
11	Advertise asymmetric pause	R/W	1: Advertises asymmetric pause	0
10	Advertise symmetric pause	R/W	1: Advertises symmetric pause	0
9	Advertise100BASE-T4	R/W	1: Advertises 100BASE-T4	0
8	Advertise100BASE-TX FDX	R/W	1: Advertise 100BASE-TX FDX	1
7	Advertise100BASE-TX HDX	R/W	1: Advertises 100BASE-TX HDX	1
6	Advertise10BASE-T FDX	R/W	1: Advertises 10BASE-T FDX	1
5	Advertise10BASE-T HDX	R/W	1: Advertises 10BASE-T HDX	1
4:0	Advertise selector	R/W		00001

4.2.5 Link Partner Autonegotiation Capability

The bits in main register 5 can be used to determine if the Cat5 link partner (LP) used with the VSC8572-02 is compatible with the autonegotiation functionality.

Table 45 • **Autonegotiation Link Partner Ability, Address 5 (0x05)**

Bit	Name	Access	Description	Default
15	LP next page transmission request	RO	1: Requested	0
14	LP acknowledge	RO	1: Acknowledge	0
13	LP remote fault	RO	1: Remote fault	0
12	Reserved	RO	Reserved	0
11	LP advertise asymmetric pause	RO	1: Capable of asymmetric pause	0
10	LP advertise symmetric pause	RO	1: Capable of symmetric pause	0
9	LP advertise 100BASE-T4	RO	1: Capable of 100BASE-T4	0
8	LP advertise 100BASE-TX FDX	RO	1: Capable of 100BASE-TX FDX	0
7	LP advertise 100BASE-TX HDX	RO	1: Capable of 100BASE-TX HDX	0
6	LP advertise 10BASE-T FDX	RO	1: Capable of 10BASE-T FDX	0
5	LP advertise 10BASE-T HDX	RO	1: Capable of 10BASE-T HDX	0
4:0	LP advertise selector	RO		00000

4.2.6 Autonegotiation Expansion

The bits in main register 6 work together with those in register 5 to indicate the status of the LP autonegotiation functioning. The following table shows the available settings and readouts.

Table 46 • **Autonegotiation Expansion, Address 6 (0x06)**

Bit	Name	Access	Description	Default
15:5	Reserved	RO	Reserved.	All zeros
4	Parallel detection fault	RO	This bit latches high. 1: Parallel detection fault.	0
3	LP next page capable	RO	1: LP is next page capable.	0
2	Local PHY next page capable	RO	1: Local PHY is next page capable.	1
1	Page received	RO	This bit latches low. 1: New page is received.	0
0	LP is autonegotiation capable	RO	1: LP is capable of autonegotiation.	0

4.2.7 Transmit Autonegotiation Next Page

The settings in register 7 in the main registers space provide information about the number of pages in an autonegotiation sequence. The following table shows the settings available.

Table 47 • **Autonegotiation Next Page Transmit, Address 7 (0x07)**

Bit	Name	Access	Description	Default
15	Next page	R/W	1: More pages follow	0
14	Reserved	RO	Reserved	0
13	Message page	R/W	1: Message page 0: Unformatted page	1
12	Acknowledge 2	R/W	1: Complies with request 0: Cannot comply with request	0
11	Toggle	RO	1: Previous transmitted LCW = 0 0: Previous transmitted LCW = 1	0
10:0	Message/unformatted code	R/W		00000000001

4.2.8 Autonegotiation Link Partner Next Page Receive

The bits in register 8 of the main register space work together with register 7 to determine certain aspects of the LP autonegotiation. The following table shows the possible readouts.

Table 48 • **Autonegotiation LP Next Page Receive, Address 8 (0x08)**

Bit	Name	Access	Description	Default
15	LP next page	RO	1: More pages follow	0
14	Acknowledge	RO	1: LP acknowledge	0
13	LP message page	RO	1: Message page 0: Unformatted page	0
12	LP acknowledge 2	RO	1: LP complies with request	0
11	LP toggle	RO	1: Previous transmitted LCW = 0 0: Previous transmitted LCW = 1	0
10:0	LP message/unformatted code	RO		All zeros

4.2.9 1000BASE-T Control

The VSC8572-02's 1000BASE-T functionality is controlled by the bits in register 9 of the main register space. The following table shows the settings and readouts available.

Table 49 • **1000BASE-T Control, Address 9 (0x09)**

Bit	Name	Access	Description	Default
15:13	Transmitter test mode	R/W	000: Normal 001: Mode 1: Transmit waveform test 010: Mode 2: Transmit jitter test as master 011: Mode 3: Transmit jitter test as slave 100: Mode 4: Transmitter distortion test 101–111: Reserved	000
12	Master/slave manual configuration	R/W	1: Master/slave manual configuration enabled	0
11	Master/slave value	R/W	This register is only valid when bit 9.12 is set to 1. 1: Configure PHY as master during negotiation 0: Configure PHY as slave during negotiation	0
10	Port type	R/W	1: Multi-port device 0: Single-port device	1
9	1000BASE-T FDX capability	R/W	1: PHY is 1000BASE-T FDX capable	1
8	1000BASE-T HDX capability	R/W	1: PHY is 1000BASE-T HDX capable	1
7:0	Reserved	R/W	Reserved	0x00

Note: Transmitter test mode (bits 15:13) operates in the manner described in IEEE 802.3 section 40.6.1.1.2. When using any of the transmitter test modes, the automatic media sense feature must be disabled. For more information, see Extended PHY Control Set 1, page 106.

4.2.10 1000BASE-T Status

The bits in register 10 of the main register space can be read to obtain the status of the 1000BASE-T communications enabled in the device. The following table shows the readouts.

Table 50 • **1000BASE-T Status, Address 10 (0x0A)**

Bit	Name	Access	Description	Default
15	Master/slave configuration fault	RO	This bit latches high. 1: Master/slave configuration fault detected 0: No master/slave configuration fault detected	0
14	Master/slave configuration resolution	RO	1: Local PHY configuration resolved to master 0: Local PHY configuration resolved to slave	1
13	Local receiver status	RO	1: Local receiver is operating normally	0
12	Remote receiver status	RO	1: Remote receiver OK	0
11	LP 1000BASE-T FDX capability	RO	1: LP 1000BASE-T FDX capable	0
10	LP 1000BASE-T HDX capability	RO	1: LP 1000BASE-T HDX capable	0
9:8	Reserved	RO	Reserved	00

Table 50 • **1000BASE-T Status, Address 10 (0x0A) *(continued)***

Bit	Name	Access	Description	Default
7:0	Idle error count	RO	Self-clearing register	0x00

4.2.11 MMD Access Control Register

The bits in register 13 of the main register space are a window to the EEE registers as defined in IEEE 802.3az-2010 Clause 45.

Table 51 • **MMD EEE Access, Address 13 (0x0D)**

Bit	Name	Access	Description
15:14	Function	R/W	00: Address 01: Data, no post increment 10: Data, post increment for read and write 11: Data, post increment for write only
13:5	Reserved	R/W	Reserved
4:0	DVAD	R/W	Device address as defined in IEEE 802.3az-2010 table 45–1

4.2.12 MMD Address or Data Register

The bits in register 14 of the main register space are a window to the EEE registers as defined in IEEE 802.3az-2010 Clause 45.

Table 52 • **MMD Address or Data Register, Address 14 (0x0E)**

Bit	Name	Access	Description
15:0	Register Address/Data	R/W	When register 13.15:14 = 2'b00, address of register of the device that is specified by 13.4:0. Otherwise, the data to be written to or read from the register.

4.2.13 1000BASE-T Status Extension 1

Register 15 provides additional information about the operation of the device 1000BASE-T communications. The following table shows the readouts available.

Table 53 • **1000BASE-T Status Extension 1, Address 15 (0x0F)**

Bit	Name	Access	Description	Default
15	1000BASE-X FDX capability	RO	1: PHY is 1000BASE-X FDX capable	1
14	1000BASE-X HDX capability	RO	1: PHY is 1000BASE-X HDX capable	1
13	1000BASE-T FDX capability	RO	1: PHY is 1000BASE-T FDX capable	1
12	1000BASE-T HDX capability	RO	1: PHY is 1000BASE-T HDX capable	1
11:0	Reserved	RO	Reserved	0x000

4.2.14 100BASE-TX/FX Status Extension

Register 16 in the main registers page space of the VSC8572-02 provides additional information about the status of the device's 100BASE-TX/100BASE-FX operation.

Table 54 • **100BASE-TX/FX Status Extension, Address 16 (0x10)**

Bit	Name	Access	Description	Default
15	100BASE-TX/FX Descrambler	RO	1: Descrambler locked	0
14	100BASE-TX/FX lock error	RO	Self-clearing bit. 1: Lock error detected	0
13	100BASE-TX/FX disconnect state	RO	Self-clearing bit. 1: PHY 100BASE-TX link disconnect detected	0
12	100BASE-TX/FX current link status	RO	1: PHY 100BASE-TX link active	0
11	100BASE-TX/FX receive error	RO	Self-clearing bit. 1: Receive error detected	0
10	100BASE-TX/FX transmit error	RO	Self-clearing bit. 1: Transmit error detected	0
9	100BASE-TX/FX SSD error	RO	Self-clearing bit. 1: Start-of-stream delimiter error detected	0
8	100BASE-TX/FX ESD error	RO	Self-clearing bit. 1: End-of-stream delimiter error detected	0
7:0	Reserved	RO	Reserved	

4.2.15 1000BASE-T Status Extension 2

The second status extension register is at address 17 in the device main registers space. It provides information about another set of parameters associated with 1000BASE-T communications. For information about the first status extension register, see Table 53, page 102.

Table 55 • **1000BASE-T Status Extension 2, Address 17 (0x11)**

Bit	Name	Access	Description	Default
15	1000BASE-T descrambler	RO	1: Descrambler locked.	0
14	1000BASE-T lock error	RO	Self-clearing bit. 1: Lock error detected	0
13	1000BASE-T disconnect state	RO	Self-clearing bit. 1: PHY 1000BASE-T link disconnect detected	0
12	1000BASE-T current link status	RO	1: PHY 1000BASE-T link active	0
11	1000BASE-T receive error	RO	Self-clearing bit. 1: Receive error detected	0
10	1000BASE-T transmit error	RO	Self-clearing bit. 1: Transmit error detected	0
9	1000BASE-T SSD error	RO	Self-clearing bit. 1: Start-of-stream delimiter error detected	0

Table 55 • **1000BASE-T Status Extension 2, Address 17 (0x11) *(continued)***

Bit	Name	Access	Description	Default
8	1000BASE-T ESD error	RO	Self-clearing bit. 1: End-of-stream delimiter error detected	0
7	1000BASE-T carrier extension error	RO	Self-clearing bit. 1: Carrier extension error detected	0
6	Non-compliant BCM5400 detected	RO	1: Non-compliant BCM5400 link partner detected	0
5	MDI crossover error	RO	1: MDI crossover error was detected	0
4:0	Reserved	RO	Reserved	

4.2.16 Bypass Control

The bits in this register control aspects of functionality in effect when the device is disabled for the purpose of traffic bypass. The following table shows the settings available.

Table 56 • **Bypass Control, Address 18 (0x12)**

Bit	Name	Access	Description	Default
15	Transmit disable	R/W	1: PHY transmitter disabled	0
14	4B5B encoder/decoder	R/W	1: Bypass 4B/5B encoder/decoder	0
13	Scrambler	R/W	1: Bypass scrambler	0
12	Descrambler	R/W	1: Bypass descrambler	0
11	PCS receive	R/W	1: Bypass PCS receiver	0
10	PCS transmit	R/W	1: Bypass PCS transmit	0
9	LFI timer	R/W	1: Bypass Link Fail Inhibit (LFI) timer	0
8	Reserved	RO	Reserved	
7	HP Auto-MDIX at forced 10/100	R/W	Sticky bit. 1: Disable HP Auto-MDIX at forced 10/100 speeds	1
6	Non-compliant BCM5400 detect disable	R/W	Sticky bit. 1: Disable non-compliant BCM5400 detection	0
5	Disable pair swap correction (HP Auto-MDIX when autonegotiation enabled)	R/W	Sticky bit. 1: Disable the automatic pair swap correction	0
4	Disable polarity correction	R/W	Sticky bit. 1: Disable polarity inversion correction on each subchannel	0
3	Parallel detect control	R/W	Sticky bit. 1: Do not ignore advertised ability 0: Ignore advertised ability	1
2	Pulse shaping filter	R/W	1: Disable pulse shaping filter	0
1	Disable automatic 1000BASE-T next page exchange	R/W	Sticky bit. 1: Disable automatic 1000BASE T next page exchanges	0
0	Reserved	RO	Reserved	

Note: If bit 18.1 is set to 1 in this register, automatic exchange of next pages is disabled, and control is returned to the user through the SMI after the base page is exchanged. The user then must send the correct sequence of next pages to the link partner, determine the common capabilities, and force the device into the correct configuration following the successful exchange of pages.

4.2.17 Error Counter 1

The bits in register 19 provide an error counter. The following table shows the settings available.

Table 57 • **Error Counter 1, Address 19 (0x13)**

Bit	Name	Access	Description	Default
15:8	Reserved	RO	Reserved.	
7:0	100/1000 receive error counter	RO	8-bit counter that saturates when it reaches 255. These bits are self-clearing when read.	0x00

4.2.18 Error Counter 2

The bits in register 20 provide an error counter. The following table shows the settings available.

Table 58 • **Error Counter 2, Address 20 (0x14)**

Bit	Name	Access	Description	Default
15:8	Reserved	RO	Reserved.	
7:0	100/1000 false carrier counter	RO	8-bit counter that saturates when it reaches 255. These bits are self-clearing when read.	0x00

4.2.19 Error Counter 3

The bits in register 21 provide an error counter. The following table shows the settings available.

Table 59 • **Error Counter 3, Address 21 (0x15)**

Bit	Name	Access	Description	Default
15:8	Reserved	RO	Reserved.	
7:0	Copper media link disconnect counter	RO	8-bit counter that saturates when it reaches 255. These bits are self-clearing when read.	0x00

4.2.20 Extended Control and Status

The bits in register 22 provide additional device control and readouts. The following table shows the settings available.

Table 60 • **Extended Control and Status, Address 22 (0x16)**

Bit	Name	Access	Description	Default
15	Force 10BASE-T link high	R/W	Sticky bit. 1: Bypass link integrity test 0: Enable link integrity test	0
14	Jabber detect disable	R/W	Sticky bit. 1: Disable jabber detect	0
13	Disable 10BASE-T echo	R/W	Sticky bit. 1: Disable 10BASE-T echo	1
12	Disable SQE mode	R/W	Sticky bit. 1: Disable SQE mode	1

Table 60 • **Extended Control and Status, Address 22 (0x16) *(continued)***

Bit	Name	Access	Description	Default
11:10	10BASE-T squelch control	R/W	Sticky bit. 00: Normal squelch 01: Low squelch 10: High squelch 11: Reserved	00
9	Sticky reset enable	R/W	Super-sticky bit. 1: Enabled	1
8	EOF Error	RO	This bit is self-clearing. 1: EOF error detected	0
7	10BASE-T disconnect state	RO	This bit is self-clearing. 1: 10BASE-T link disconnect detected	0
6	10BASE-T link status	RO	1: 10BASE-T link active	0
5:1	Reserved	RO	Reserved	
0	SMI broadcast write	R/W	Sticky bit. 1: Enabled	0

The following information applies to the extended control and status bits:

- When bit 22.15 is set, the link integrity state machine is bypassed and the PHY is forced into a link pass status.
- When bits 22.11:10 are set to 00, the squelch threshold levels are based on the IEEE standard for 10BASE-T. When set to 01, the squelch level is decreased, which can improve the bit error rate performance on long loops. When set to 10, the squelch level is increased and can improve the bit error rate in high-noise environments.
- When bit 22.9 is set, all sticky register bits retain their values during a software reset. Clearing this bit causes all sticky register bits to change to their default values upon software reset. Super-sticky bits retain their values upon software reset regardless of the setting of bit 22.9.
- When bit 22.0 is set, if a write to any PHY register (registers 0–31, including extended registers), the same write is broadcast to all PHYs. For example, if bit 22.0 is set to 1 and a write to PHY0 is executed (register 0 is set to 0x1040), all PHYs' register 0s are set to 0x1040. This bit must be disabled before performing a software reset of any PHY port (see register 0 bit 15, Table 40, page 96). Disabling this bit restores normal PHY write operation. Reads are still possible when this bit is set, but the value that is read corresponds only to the particular PHY being addressed.

4.2.21 Extended PHY Control Set 1

The following table shows the settings available.

Table 61 • **Extended PHY Control 1, Address 23 (0x17)**

Bit	Name	Access	Description	Default
15:13	Reserved	R/W	Reserved	0
12	MAC interface mode	R/W	Super-sticky bit. 0: RGMII/SGMII 1: 1000BASE-X. **Note:** Register 19G.15:14 must be = 00 for this selection to be valid.	0
11	AMS preference	R/W	Super-sticky bit. 1: Cat5 copper preferred. 0: SerDes fiber/SFP preferred.	0

Table 61 • **Extended PHY Control 1, Address 23 (0x17) *(continued)***

Bit	Name	Access	Description	Default
10:8	Media operating mode	R/W	Super-sticky bits. 000: Cat5 copper only. 001: SerDes fiber/SFP protocol transfer mode only. 010: 1000BASE-X fiber/SFP media only with autonegotiation performed by the PHY. 011: 100BASE-FX fiber/SFP on the fiber media pins only. 101: Automatic media sense (AMS) with Cat5 media or SerDes fiber/SFP protocol transfer mode. 110: AMS with Cat5 media or 1000BASE-X fiber/SFP media with autonegotiation performed by PHY. 111: AMS with Cat5 media or 100BASE-FX fiber/SFP media. 100: AMS.	000
7:6	Force AMS override	R/W	Sticky bits. 00: Normal AMS selection 01: Force AMS to select SerDes media only 10: Force AMS to select copper media only 11: Reserved	00
5:4	Reserved	RO	Reserved.	
3	Far-end loopback mode	R/W	1: Enabled.	0
2:0	Reserved	RO	Reserved.	

Note: After configuring bits 13:8 of the extended PHY control register set 1, a software reset (register 0, bit 15) must be written to change the device operating mode. On read, these bits only indicate the actual operating mode and not the pending operating mode setting before a software reset has taken place.

4.2.22 Extended PHY Control Set 2

The second set of extended controls is located in register 24 in the main register space for the device. The following table shows the settings and readouts available.

Table 62 • **Extended PHY Control 2, Address 24 (0x18)**

Bit	Name	Access	Description	Default
15:13	100BASE-TX edge rate control	R/W	Sticky bit. 011: +5 edge rate (slowest) 010: +4 edge rate 001: +3 edge rate 000: +2 edge rate 111: +1 edge rate 110: Default edge rate 101: –1 edge rate 100: –2 edge rate (fastest)	001
12	PICMG 2.16 reduced power mode	R/W	Sticky bit. 1: Enabled	0
11:6	Reserved	RO	Reserved	

Table 62 • **Extended PHY Control 2, Address 24 (0x18) *(continued)***

Bit	Name	Access	Description	Default
5:4	Jumbo packet mode	R/W	Sticky bit. 00: Normal IEEE 1.5 kB packet length 01: 9 kB jumbo packet length (12 kB with 60 ppm or better reference clock) 10: 12 kB jumbo packet length (16 kB with 70 ppm or better reference clock) 11: Reserved	00
3:1	Reserved	RO	Reserved	
0	1000BASE-T connector loopback	R/W	1: Enabled	0

Note: When bits 5:4 are set to jumbo packet mode, the default maximum packet values are based on 100 ppm driven reference clock to the device. Controlling the ppm offset between the MAC and the PHY as specified in the bit description results in a higher jumbo packet length.

4.2.23 Interrupt Mask

These bits control the device interrupt mask. The following table shows the settings available.

Table 63 • **Interrupt Mask, Address 25 (0x19)**

Bit	Name	Access	Description	Default
15	MDINT interrupt status enable	R/W	Sticky bit. 1: Enabled.	0
14	Speed state change mask	R/W	Sticky bit. 1: Enabled.	0
13	Link state change mask	R/W	Sticky bit. 1: Enabled.	0
12	FDX state change mask	R/W	Sticky bit. 1: Enabled.	0
11	Autonegotiation error mask	R/W	Sticky bit. 1: Enabled.	0
10	Autonegotiation complete mask	R/W	Sticky bit. 1: Enabled.	0
9	Inline powered device (PoE) detect mask	R/W	Sticky bit. 1: Enabled.	0
8	Symbol error interrupt mask	R/W	Sticky bit. 1: Enabled.	0
7	Fast link failure interrupt mask(1)	R/W	Sticky bit. 1: Enabled.	0
6:5	Reserved	R/W		0
4	AMS media changed mask(2)	R/W	Sticky bit. 1: Enabled.	0
3	False carrier interrupt mask	R/W	Sticky bit. 1: Enabled.	0
2	Link speed downshift detect mask	R/W	Sticky bit. 1: Enabled.	0
1	Master/Slave resolution error mask	R/W	Sticky bit. 1: Enabled.	0
0	RX_ER interrupt mask	R/W	Sticky bit. 1: Enabled.	0

1. The interrupt is only valid for 100 Mbps and 1000 Mbps speeds. Notification at 10 Mbps speed requires use of the FASTLINK-FAIL pin.
2. If hardware interrupts are not used, the mask can still be set and the status polled for changes.

Note: When bit 25.15 is set, the MDINT pin is enabled. When enabled, the state of this pin reflects the state of bit 26.15. Clearing this bit only inhibits the MDINT pin from being asserted. Also, before enabling this bit, read register 26 to clear any previously inactive interrupts pending that will cause bit 25.15 to be set.

4.2.24 Interrupt Status

The status of interrupts already written to the device is available for reading from register 26 in the main registers space. The following table shows the expected readouts.

Table 64 • **Interrupt Status, Address 26 (0x1A)**

Bit	Name	Access	Description	Default
15	Interrupt status	RO	Self-clearing bit. 1: Interrupt pending.	0
14	Speed state change status	RO	Self-clearing bit. 1: Interrupt pending.	0
13	Link state change status	RO	Self-clearing bit. 1: Interrupt pending.	0
12	FDX state change status	RO	Self-clearing bit. 1: Interrupt pending.	0
11	Autonegotiation error status	RO	Self-clearing bit. 1: Interrupt pending.	0
10	Autonegotiation complete status	RO	Self-clearing bit. 1: Interrupt pending.	0
9	Inline powered device detect status	RO	Self-clearing bit. 1: Interrupt pending.	0
8	Symbol error status	RO	Self-clearing bit. 1: Interrupt pending.	0
7	Fast link failure detect status	RO	Self-clearing bit. 1: Interrupt pending.	0
6:5	Reserved	RO		0
4	AMS media changed mask(1)	RO	Self-clearing bit. 1: Interrupt pending.	0
3	False carrier interrupt status	RO	Self-clearing bit. 1: Interrupt pending.	0
2	Link speed downshift detect status	RO	Self-clearing bit. 1: Interrupt pending.	0
1	Master/Slave resolution error status	RO	Self-clearing bit. 1: Interrupt pending.	0
0	RX_ER interrupt status	RO	Self-clearing bit. 1: Interrupt pending.	0

1. If hardware interrupts are not used, the mask can still be set and the status polled for changes.

The following information applies to the interrupt status bits:

- All set bits in this register are cleared after being read (self-clearing). If bit 26.15 is set, the cause of the interrupt can be read by reading bits 26.14:0.
- For bits 26.14 and 26.12, bit 0.12 must be set for this interrupt to assert.
- For bit 26.2, bits 4.8:5 must be set for this interrupt to assert.
- For bit 26.0, this interrupt will not occur when RX_ER is used for carrier-extension decoding of a link partner's data transmission.

4.2.25 Device Auxiliary Control and Status

Register 28 provides control and status information for several device functions not controlled or monitored by other device registers. The following table shows the settings available and the expected readouts.

Table 65 • **Auxiliary Control and Status, Address 28 (0x1C)**

Bit	Name	Access	Description	Default
15	Autonegotiation complete	RO	Duplicate of bit 1.5	0
14	Autonegotiation disabled	RO	Inverted duplicate of bit 0.12	0
13[1]	HP Auto-MDIX crossover indication	RO	1: HP Auto-MDIX crossover performed internally	0

Table 65 • **Auxiliary Control and Status, Address 28 (0x1C) *(continued)***

Bit	Name	Access	Description	Default
12	CD pair swap	RO	1: CD pairs are swapped	0
11	A polarity inversion	RO	1: Polarity swap on pair A	0
10	B polarity inversion	RO	1: Polarity swap on pair B	0
9	C polarity inversion	RO	1: Polarity swap on pair C	0
8	D polarity inversion	RO	1: Polarity swap on pair D	0
7	ActiPHY link status time-out control [1]	R/W	Sticky bit. Bits 7 and 2 are part of the ActiPHY Link Status time-out control. Bit 7 is the MSB. 00: 2.3 seconds 01: 3.3 seconds 10: 4.3 seconds 11: 5.3 seconds	0
6	ActiPHY mode enable	R/W	Sticky bit. 1: Enabled	0
5	FDX status	RO	1: Full-duplex 0: Half-duplex	00
4:3	Speed status	RO	00: Speed is 10BASE-T 01: Speed is 100BASE-TX or 100BASE-FX 10: Speed is 1000BASE-T or 1000BASE-X 11: Reserved	0
2	ActiPHY link status time-out control [0]	R/W	Sticky bit. Bits 7 and 2 are part of the ActiPHY Link Status time-out control. Bit 7 is the MSB. 00: 2.3 seconds 01: 3.3 seconds 10: 4.3 seconds 11: 5.3 seconds	1
1:0	Media mode status	RO	00: No media selected 01: Copper media selected 10: SerDes (Fiber) media selected 11: Reserved	00

1. In 1000BT mode, if Force MDI crossover is performed while link is up, the 1000BT link must be re-negotiated in order for this bit to reflect the actual Auto-MDIX setting.

4.2.26 LED Mode Select

The device LED outputs are controlled using the bits in register 29 of the main register space. The following table shows the information needed to access the functionality of each of the outputs. For more information about LED modes, see Table 28, page 76. For information about enabling the extended LED mode bits in Register 19E1 bits 13 to 12, see Table 29, page 77.

Table 66 • **LED Mode Select, Address 29 (0x1D)**

Bit	Name	Access	Description	Default
15:12	LED3 mode select	R/W	Sticky bit. Select from LED modes 0–15.	1000
11:8	LED2 mode select	R/W	Sticky bit. Select from LED modes 0–15.	0000
7:4	LED1 mode select	R/W	Sticky bit. Select from LED modes 0–15.	0010
3:0	LED0 mode select	R/W	Sticky bit. Select from LED modes 0–15.	0001

4.2.27 LED Behavior

The bits in register 30 control and enable you to read the status of the pulse or blink rate of the device LEDs. The following table shows the settings you can write to the register or read from the register.

Table 67 • **LED Behavior, Address 30 (0x1E)**

Bit	Name	Access	Description	Default
15	Copper and fiber LED combine disable	R/W	Sticky bit 0: Combine enabled (Copper/Fiber on link/linkXXXX/activity LED) 1: Disable combination (link/linkXXXX/activity LED; indicates copper only)	0
14	Activity output select	R/W	Sticky bit 1: Activity LED becomes TX_Activity and fiber activity LED becomes RX_Activity 0: TX and RX activity both displayed on activity LEDs	0
13	Reserved	RO	Reserved	
12	LED pulsing enable	R/W	Sticky bit 0: Normal operation 1: LEDs pulse with a 5 kHz, programmable duty cycle when active	0
11:10	LED blink/pulse-stretch rate	R/W	Sticky bit 00: 2.5 Hz blink rate/400 ms pulse-stretch 01: 5 Hz blink rate/200 ms pulse-stretch 10: 10 Hz blink rate/100 ms pulse-stretch 11: 20 Hz blink rate/50 ms pulse-stretch The blink rate selection for PHY0 globally sets the rate used for all LED pins on all PHY ports	01
9	Reserved	RO	Reserved	
8	LED3 pulse-stretch/blink select	R/W	Sticky bit 1: Pulse-stretch 0: Blink	0
7	LED2 pulse-stretch/blink select	R/W	Sticky bit 1: Pulse-stretch 0: Blink	0
6	LED1 pulse-stretch/blink select	R/W	Sticky bit 1: Pulse-stretch 0: Blink	0
5	LED0 pulse-stretch/blink select	R/W	Sticky bit 1: Pulse-stretch 0: Blink	0
4:2	Reserved	RO	Reserved	
3	LED3 combine feature disable	R/W	Sticky bit 0: Combine enabled (link/activity, duplex/collision) 1: Disable combination (link only, duplex only)	0
2	LED2 combine feature disable	R/W	Sticky bit 0: Combine enabled (link/activity, duplex/collision) 1: Disable combination (link only, duplex only)	0

Table 67 • **LED Behavior, Address 30 (0x1E) *(continued)***

Bit	Name	Access	Description	Default
1	LED1 combine feature disable	R/W	Sticky bit 0: Combine enabled (link/activity, duplex/collision) 1: Disable combination (link only, duplex only)	0
0	LED0 combine feature disable	R/W	Sticky bit 0: Combine enabled (link/activity, duplex/collision) 1: Disable combination (link only, duplex only)	0

Note: Bits 30.11:10 are active only in port 0 and affect the behavior of LEDs for all the ports.

4.2.28 Extended Page Access

To provide functionality beyond the IEEE 802.3-specified registers and main device registers, the VSC8572-02 includes an extended set of registers that provide an additional 15 register spaces.

The register at address 31 controls the access to the extended registers for the VSC8572-02. Accessing the GPIO page register space is similar to accessing the extended page registers. The following table shows the settings available.

Table 68 • **Extended/GPIO Register Page Access, Address 31 (0x1F)**

Bit	Name	Access	Description	Default
15:0	Extended/GPIO page register access	R/W	0x0000: Register 16–30 accesses main register space. Writing 0x0000 to register 31 restores the main register access. 0x0001: Registers 16–30 access extended register space 1 0x0002: Registers 16–30 access extended register space 2 0x0003: Registers 16–30 access extended register space 3 0x0010: Registers 0–30 access GPIO register space 0x1588: Registers 16-18 1588 registers	0x0000

4.3 Extended Page 1 Registers

To access the extended page 1 registers (16E1–30E1), enable extended register access by writing 0x0001 to register 31. Writing 0x0000 to register 31 restores the main register access.

When extended page 1 register access is enabled, reads and writes to registers 16–30 affect the extended registers 16E1–30E1 instead of those same registers in the IEEE-specified register space. Registers 0–15 are not affected by the state of the extended page register access.

Table 69 • **Extended Registers Page 1 Space**

Address	Name
16E1	SerDes Media Control
17E1	Reserved
18E1	Cu Media CRC good counter
19E1	Extended mode and SIGDET control
20E1	Extended PHY control 3 (ActiPHY)

Table 69 • **Extended Registers Page 1 Space *(continued)***

Address	Name
21E1–22E1	Reserved
23E1	Extended PHY control 4 (PoE and CRC error counter)
24E1	VeriPHY 1
25E1	VeriPHY 2
26E1	VeriPHY 3
27E1–28E1	Reserved
29E1	Ethernet packet generator (EPG) 1
30E1	EPG 2

4.3.1 SerDes Media Control

Register 16E1 controls some functions of the SerDes media interface on ports 0–3. These settings are only valid for those ports. The following table shows the setting available in this register.

Table 70 • **SerDes Media Control, Address 16E1 (0x10)**

Bit	Name	Access	Description	Default
15:14	Transmit remote fault	R/W	Remote fault indication sent to link partner (LP)	00
13:12	Link partner (LP) remote fault	RO	Remote fault bits sent by LP during autonegotiation	00
11:10	Reserved	RO	Reserved	
9	Allow 1000BASE-X link-up	R/W	Sticky bit. 1: Allow 1000BASE-X fiber media link-up capability 0: Suppress 1000BASE-X fiber media link-up capability	1
8	Allow 100BASE-FX link-up	R/W	Sticky bit. 1: Allow 100BASE-FX fiber media link-up capability 0: Suppress 100BASE-FX fiber media link-up capability	1
7	Reserved	RO	Reserved	
6	Far end fault detected in 100BASE-FX	RO	Self-clearing bit. 1: Far end fault in 100BASE-FX detected	0
5:0	Reserved	RO	Reserved	

4.3.2 Cu Media CRC Good Counter

Register 18E1 makes it possible to read the contents of the CRC good counter for packets that are received on the Cu media interface; the number of CRC routines that have executed successfully. The following table shows the expected readouts.

Table 71 • **Cu Media CRC Good Counter, Address 18E1 (0x12)**

Bit	Name	Access	Description	Default
15	Packet since last read	RO	Self-clearing bit. 1: Packet received since last read.	0

Table 71 • **Cu Media CRC Good Counter, Address 18E1 (0x12) *(continued)***

Bit	Name	Access	Description	Default
14	Reserved	RO	Reserved.	
13:0	Cu Media CRC good counter contents	RO	Self-clearing bit. Counter containing the number of packets with valid CRCs modulo 10,000; this counter does not saturate and will roll over to zero on the next good packet received after 9,999.	0x000

4.3.3 Extended Mode Control

Register 19E1 controls the extended LED and other chip modes. The following table shows the settings available.

Table 72 • **Extended Mode Control, Address 19E1 (0x13)**

Bit	Name	Access	Description	Default
15	LED3 Extended Mode	R/W	Sticky bit. 1: See Extended LED Modes, page 77	0
14	LED2 Extended Mode	R/W	Sticky bit. 1: See Extended LED Modes, page 77	0
13	LED1 Extended Mode	R/W	Sticky bit. 1: See Extended LED Modes, page 77	0
12	LED0 Extended Mode	R/W	Sticky bit. 1: See Extended LED Modes, page 77	0
11	LED Reset Blink Suppress	R/W	Sticky bit. 1: Blink LEDs after COMA_MODE is de-asserted 0: Suppress LED blink after COMA_MODE is de-asserted	0
10:5	Reserved	RO	Reserved	0
4	Fast link failure	R/W	Sticky bit. Enable fast link failure pin. This must be done from PHY0 only. 1: Enabled 0: Disabled (GPIO9 pin becomes general purpose I/O)	0
3:2	Force MDI crossover	R/W	Sticky bits. 00: Normal HP Auto-MDIX operation 01: Reserved 10: Copper media forced to MDI 11: Copper media forced MDI-X	00
1	Reserved	RO	Reserved	
0	GPIO[1:0]/SIGDET[1:0] pin polarity	R/W	Sticky bit. 1: Active low 0: Active high	0

4.3.4 ActiPHY Control

Register 20E1 controls the device ActiPHY sleep timer, its wake-up timer, and its link speed downshifting feature. The following table shows the settings available.

Table 73 • **Extended PHY Control 3, Address 20E1 (0x14)**

Bit	Name	Access	Description	Default
15	Disable carrier extension	R/W	1: Disable carrier extension in RGMII/1000BASE-T copper links	1
14:13	ActiPHY sleep timer	R/W	Sticky bit. 00: 1 second 01: 2 seconds 10: 3 seconds 11: 4 seconds	01
12:11	ActiPHY wake-up timer	R/W	Sticky bit. 00: 160 ms 01: 400 ms 10: 800 ms 11: 2 seconds	00
10	Reserved	RO	Reserved	
9	PHY address reversal	R/W	Sticky bit. Reverse PHY address Enabling causes physical PHY 0 to have address of 3, PHY 1 address of 2, PHY 2 address of 1, and PHY 3 address of 0. Changing this bit to 1 should initially be done from PHY 0 and changing to 0 from PHY3 1: Enabled 0: Disabled	0
8	Reserved	RO	Valid only on PHY0	
7:6	Media mode status	RO	00: No media selected 01: Copper media selected 10: SerDes media selected 11: Reserved	00
5	Enable 10BASE-T no preamble mode	R/W	Sticky bit. 1: 10BASE-T will assert RX_DV indication when data is presented to the receiver even without a preamble preceding it	0
4	Enable link speed autodownshift feature	R/W	Sticky bit. 1: Enable auto link speed downshift from 1000BASE-T	0
3:2	Link speed auto downshift control	R/W	Sticky bits. 00: Downshift after 2 failed 1000BASE-T autonegotiation attempts 01: Downshift after 3 failed 1000BASE-T autonegotiation attempts 10: Downshift after 4 failed 1000BASE-T autonegotiation attempts 11: Downshift after 5 failed 1000BASE-T autonegotiation attempts	01
1	Link speed auto downshift status	RO	0: No downshift 1: Downshift is required or has occurred	0

Table 73 • **Extended PHY Control 3, Address 20E1 (0x14) *(continued)***

Bit	Name	Access	Description	Default
0	Reserved	RO	Reserved	

4.3.5 PoE and Miscellaneous Functionality

The register at address 23E1 controls various aspects of inline powering and the CRC error counter in the VSC8572-02.

Table 74 • **Extended PHY Control 4, Address 23E1 (0x17)**

Bit	Name	Access	Description	Default
15:11	PHY address	RO	PHY address; latched on reset	
10	Inline powered device detection	R/W	Sticky bit. 1: Enabled	0
9:8	Inline powered device detection status	RO	Only valid when bit 10 is set. 00: Searching for devices 01: Device found; requires inline power 10: Device found; does not require inline power 11: Reserved	00
7:0	Cu Media CRC error counter	RO	Self-clearing bit	

RC error counter for packets received on the Cu media interface. The value saturates at 0xFF and subsequently clears when read and restarts count.0x00

4.3.6 VeriPHY Control 1

Register 24E1 in the extended register space provides control over the device VeriPHY diagnostics features. There are three separate VeriPHY control registers. The following table shows the settings available and describes the expected readouts.

Table 75 • **VeriPHY Control Register 1, Address 24E1 (0x18)**

Bit	Name	Access	Description	Default
15	VeriPHY trigger	R/W	Self-clearing bit. 1: Triggers the VeriPHY algorithm and clears when VeriPHY has completed. Settings in registers 24E–26E become valid after this bit clears.	0
14	VeriPHY valid	RO	1: VeriPHY results in registers 24E–26E are valid.	0
13:8	Pair A (1, 2) distance	RO	Loop length or distance to anomaly for pair A (1, 2).	0x00
7:6	Reserved	RO	Reserved.	
5:0	Pair B (3, 6) distance	RO	Loop length or distance to anomaly for pair B (3, 6).	0x00

Note: The resolution of the 6-bit length field is 3 meters.

4.3.7 VeriPHY Control 2

The register at address 25E1 consists of the second of the three device registers that provide control over VeriPHY diagnostics features. The following table shows the expected readouts.

Table 76 • **VeriPHY Control Register 2, Address 25E1 (0x19)**

Bit	Name	Access	Description	Default
15:14	Reserved	RO	Reserved	
13:8	Pair C (4, 5) distance	RO	Loop length or distance to anomaly for pair C (4, 5)	0x00
7:6	Reserved	RO	Reserved	
5:0	Pair D (7, 8) distance	RO	Loop length or distance to anomaly for pair D (7, 8)	0x00

Note: The resolution of the 6-bit length field is 3 meters.

4.3.8 VeriPHY Control 3

The register at address 26E1 consists of the third of the three device registers that provide control over VeriPHY diagnostics features. Specifically, this register provides information about the termination status (fault condition) for all two link partner pairs. The following table shows the expected readouts.

Table 77 • **VeriPHY Control Register 3, Address 26E1 (0x1A)**

Bit	Name	Access	Description	Default
15:12	Pair A (1, 2) termination status	RO	Termination fault for pair A (1, 2)	0x00
11:8	Pair B (3, 6) termination status	RO	Termination fault for pair B (3, 4)	0x00
7:4	Pair C (4, 5) termination status	RO	Termination fault for pair C (4, 5)	0x00
3:0	Pair D (7, 8) termination status	RO	Termination fault for pair D (7, 8)	0x00

The following table shows the meanings for the various fault codes.

Table 78 • **VeriPHY Control Register 3 Fault Codes**

Code	Denotes
0000	Correctly terminated pair
0001	Open pair
0010	Shorted pair
0100	Abnormal termination
1000	Cross-pair short to pair A
1001	Cross-pair short to pair B
1010	Cross-pair short to pair C
1011	Cross-pair short to pair D
1100	Abnormal cross-pair coupling with pair A
1101	Abnormal cross-pair coupling with pair B
1110	Abnormal cross-pair coupling with pair C
1111	Abnormal cross-pair coupling with pair D

4.3.9 Ethernet Packet Generator Control 1

The EPG control register provides access to and control of various aspects of the EPG testing feature. There are two separate EPG control registers. The following table shows the settings available in the first register.

Table 79 • **EPG Control Register 1, Address 29E1 (0x1D)**

Bit	Name	Access	Description	Default
15	EPG enable	R/W	1: Enable EPG	0
14	EPG run or stop	R/W	1: Run EPG	0
13	Transmission duration	R/W	1: Continuous (sends in 10,000-packet increments) 0: Send 30,000,000 packets and stop	0
12:11	Packet length	R/W	00: 125 bytes 01: 64 bytes 10: 1518 bytes 11: 10,000 bytes (jumbo packet)	0
10	Interpacket gap	R/W	1: 8,192 ns 0: 96 ns	0
9:6	Destination address	R/W	Lowest nibble of the 6-byte destination address	0001
5:2	Source address	R/W	Lowest nibble of the 6-byte destination address	0000
1	Payload type	R/W	1: Randomly generated payload pattern 0: Fixed based on payload pattern	0
0	Bad frame check sequence (FCS) generation	R/W	1: Generate packets with bad FCS 0: Generate packets with good FCS	0

The following information applies to the EPG control number 1:

- Do not run the EPG when the VSC8572-02 is connected to a live network.
- bit 29E1.13 (continuous EPG mode control): When enabled, this mode causes the device to send continuous packets. When disabled, the device continues to send packets only until it reaches the next 10,000-packet increment mark. It then ceases to send packets.
- The 6-byte destination address in bits 9:6 is assigned one of 16 addresses in the range of 0xFF FF FF FF FF F0 through 0xFF FF FF FF FF FF.
- The 6-byte source address in bits 5:2 is assigned one of 16 addresses in the range of 0xFF FF FF FF FF F0 through 0xFF FF FF FF FF FF.
- If any of bits 13:0 are changed while the EPG is running (bit 14 is set to 1), bit 14 must be cleared and then set back to 1 for the change to take effect and to restart the EPG.

4.3.10 Ethernet Packet Generator Control 2

Register 30E1 consists of the second set of bits that provide access to and control over the various aspects of the EPG testing feature. The following table shows the settings available.

Table 80 • **EPG Control Register 2, Address 30E1 (0x1E)**

Bit	Name	Access	Description	Default
15:0	EPG packet payload	R/W	Data pattern repeated in the payload of packets generated by the EPG	0x00

Note: If any of bits 15:0 in this register are changed while the EPG is running (bit 14 of register 29E1 is set to 1), that bit (29E1.14) must first be cleared and then set back to 1 for the change to take effect and to restart the EPG.

4.4 Extended Page 2 Registers

To access the extended page 2 registers (16E2–30E2), enable extended register access by writing 0x0002 to register 31. For more information, see Table 68, page 112.

When extended page 2 register access is enabled, reads and writes to registers 16–30 affect the extended registers 16E2–30E2 instead of those same registers in the IEEE-specified register space. Registers 0–15 are not affected by the state of the extended page register access.

Writing 0x0000 to register 31 restores the main register access.

The following table lists the addresses and register names in the extended register page 2 space. These registers are accessible only when the device register 31 is set to 0x0002.

Table 81 • **Extended Registers Page 2 Space**

Address	Name
16E2	Cu PMD Transmit Control
17E2	EEE Control
18E2	RGMII Settings
19E2-29E2	Reserved
30E2	Ring Resiliency Control

4.4.1 Cu PMD Transmit Control

The register at address 16E2 consists of the bits that provide control over the amplitude settings for the transmit side Cu PMD interface. These bits provide the ability to make small adjustments in the signal amplitude to compensate for minor variations in the magnetics from different vendors. Extreme caution must be exercised when changing these settings from the default values as they have a direct impact on the signal quality. Changing these settings also affects the linearity and harmonic distortion of the transmitted signals. For help with changing these values, contact your Microsemi representative.

Table 82 • **Cu PMD Transmit Control, Address 16E2 (0x10)**

Bit	Name	Access	Description	Default
15:12	1000BASE-T signal amplitude trim(1)	R/W	Sticky bits. 1000BASE-T signal amplitude 1111: -1.7% 1110: -2.6% 1101: -3.5% 1100: -4.4% 1011: -5.3% 1010: -7% 1001: -8.8% 1000: -10.6% 0111: 5.5% 0110: 4.6% 0101: 3.7% 0100: 2.8% 0011: 1.9% 0010: 1% 0001: 0.1% 0000: -0.8%	0000

Table 82 • **Cu PMD Transmit Control, Address 16E2 (0x10) *(continued)***

Bit	Name	Access	Description	Default
11:8	100BASE-TX signal amplitude trim(2)	R/W	Sticky bits. 100BASE-TX signal amplitude 1111: -1.7% 1110: -2.6% 1101: -3.5% 1100: -4.4% 1011: -5.3% 1010: -7% 1001: -8.8% 1000: -10.6% 0111 5.5% 0110: 4.6% 0101: 3.7% 0100: 2.8% 0011: 1.9% 0010: 1% 0001: 0.1% 0000: -0.8%	0010
7:4	10BASE-T signal amplitude trim(3)	R/W	Sticky bits. 10BASE-T signal amplitude 1111: -7% 1110: -7.9% 1101: -8.8% 1100: -9.7% 1011: -10.6% 1010: -11.5% 1001: -12.4% 1000: -13.3% 0111: 0% 0110: -0.7% 0101: -1.6% 0100: -2.5% 0011: -3.4% 0010: -4.3% 0001: -5.2% 0000: -6.1%	1011
3:0	10BASE-Te signal amplitude trim	R/W	Sticky bits. 10BASE-Te signal amplitude 1111: -30.45% 1110: -31.1% 1101: -31.75% 1100: -32.4% 1011: -33.05% 1010: -33.7% 1001: -34.35% 1000: -35% 0111: -25.25% 0110: -25.9% 0101: -26.55% 0100: -27.2% 0011: -27.85% 0010: -28.5% 0001: -29.15% 0000: -29.8%	1110

1. Changes to 1000BASE-T amplitude may result in unpredictable side effects.
2. Adjust 100BASE-TX to specific magnetics.
3. Amplitude is limited by V_{CC} (2.5 V).

4.4.2 EEE Control

The register at address 17E2 consists of the bits that provide additional control over the chip behavior in energy efficient Ethernet (IEEE 802.3az-2010) mode.

Table 83 • **EEE Control, Address 17E2 (0x11)**

Bit	Name	Access	Description	Default
15	Enable 10BASE-Te	R/W	Sticky bit. Enable energy efficient (IEEE 802.3az-2010) 10BASE-Te operating mode.	0
14	Enable LED in fiber unidirectional mode	R/W	Sticky bit. 1: Enable LED functions in fiber unidirectional mode.	0
13:10	Invert LED polarity	R/W	Sticky bits. Invert polarity of LED[3:0]_[1:0] signals. Default is to drive an active low signal on the LED pins. This also applies to enhanced serial LED mode. For more information, see Enhanced Serial LED Mode, page 79.	0000
9:6	Reserved	RO	Reserved.	
5	Enable 1000BASE-T force mode	R/W	Sticky bit. 1: Enable 1000BASE-T force mode to allow PHY to link-up in 1000BASE-T mode without forcing master/slave when register 0, bits 6 and 13 are set to 2'b10.	0
4[1]	Force transmit LPI	R/W	Sticky bit. 1: Enable the EPG to transmit LPI on the MDI, ignore data from the MAC interface. 0: Transmit idles being received from the MAC.	0
3	Inhibit 100BASE-TX transmit EEE LPI	R/W	Sticky bit. 1: Disable transmission of EEE LPI on transmit path MDI in 100BASE-TX mode when receiving LPI from MAC.	0
2	Inhibit 100BASE-TX receive EEE LPI	R/W	Sticky bit. 1: Disable transmission of EEE LPI on receive path MAC interface in 100BASE-TX mode when receiving LPI from the MDI.	0
1	Inhibit 1000BASE-T transmit EEE LPI	R/W	Sticky bit. 1: Disable transmission of EEE LPI on transmit path MDI in 1000BASE-T mode when receiving LPI from MAC.	0
0	Inhibit 1000BASE-T receive EEE LPI	R/W	Sticky bit. 1: Disable transmission of EEE LPI on receive path MAC interface in 1000BASE-T mode when receiving LPI from the MDI.	0

1. 17E2 bits 4:0 are for debugging purposes only, not for operational use.

4.4.3 RGMII Settings

The following table shows the register settings for the RGMII setting controls at address 18E2.

Table 84 • **RGMII Settings, Address 18E2**

Bit	Name	Access	Description	Default
6:4	rgmii_skew_tx	R/W	000: 0.2 ns delay 001: 0.8 ns delay 010: 1.1 ns delay 011: 1.7 ns delay 100: 2.0 ns delay 101: 2.3 ns delay 110: 2.6 ns delay 111: 3.4 ns delay	000
3:1	rgmii_skew_rx	R/W	000: 0.2 ns delay 001: 0.8 ns delay 010: 1.1 ns delay 011: 1.7 ns delay 100: 2.0 ns delay 101: 2.3 ns delay 110: 2.6 ns delay 111: 3.4 ns delay	000
0	rgmii_bit_rev	RO		0

4.4.4 Ring Resiliency Control

The following table shows the register settings for the ring resiliency controls at address 30E2.

Table 85 • **Ring Resiliency, Address 30E2**

Bit	Name	Access	Description	Default
15	Ring resiliency startup enable (master TR enable)	R/W	Sticky	0
14	Advertise ring resiliency	R/W	Sticky	0
13	LP ring resiliency advertisement	RO		0
12	Force ring resiliency enable (override autoneg)	R/W	Sticky	0
11:6	Reserved	RO	Reserved	000000
5:4	Ring resiliency status	RO	Ring resiliency status (from r1000 DSP SM) 00: Timing slave(1) 10: Timing slave becoming master 11: Timing master(1) 01: Timing master becoming slave	11
3:1	Reserved	RO	Reserved	000
0	Start switchover (only when not in progress)	RWSC		0

1. Reflects autoneg master/slave at initial link-up

4.5 Extended Page 3 Registers

To access the extended page 3 registers (16E3–30E3), enable extended register access by writing 0x0003 to register 31. For more information, see Table 68, page 112.

When extended page 3 register access is enabled, reads and writes to registers 16–30 affect the extended registers 16E3–30E3 instead of those same registers in the IEEE-specified register space. Registers 0–15 are not affected by the state of the extended page register access.

Writing 0x0000 to register 31 restores the main register access.

The following table lists the addresses and register names in the extended register page 3 space. These registers are accessible only when the device register 31 is set to 0x0003.

Table 86 • **Extended Registers Page 3 Space**

Address	Name
16E3	MAC SerDes PCS Control
17E3	MAC SerDes PCS Status
18E3	MAC SerDes Clause 37 Advertised Ability
19E3	MAC SerDes Clause 37 Link Partner Ability
20E3	MAC SerDes Status
21E3	Media SerDes Transmit Good Packet Counter
22E3	Media SerDes Transmit CRC Error Counter
23E3	Media SerDes PCS Control
24E3	Media SerDes PCS Status
25E3	Media SerDes Clause 37 Advertised Ability
26E3	Media SerDes Clause 37 Link Partner Ability
27E3	Media SerDes status
28E3	Fiber Media CRC Good Counter
29E3	Fiber Media CRC Error Counter
30E3	Reserved

4.5.1 MAC SerDes PCS Control

The register at address 16E3 consists of the bits that provide access to and control over MAC SerDes PCS block. The following table shows the settings available.

Table 87 • **MAC SerDes PCS Control, Address 16E3 (0x10)**

Bit	Name	Access	Description	Default
15	MAC interface disable	R/W	Sticky bit. 1: 1000BASE-X MAC interface disable when media link down.	0
14	MAC interface restart	R/W	Sticky bit. 1: 1000BASE-X MAC interface restart on media link change.	0
13	MAC interface PD enable	R/W	Sticky bit. 1: MAC interface autonegotiation parallel detect enable.	0

Table 87 • **MAC SerDes PCS Control, Address 16E3 (0x10) *(continued)***

Bit	Name	Access	Description	Default
12	MAC interface autonegotiation restart	R/W	Self-clearing bit. 1: Restart MAC interface autonegotiation.	0
11	Force advertised ability	R/W	1: Force 16-bit advertised ability from register 18E3.	0
10:8	SGMII preamble control	R/W	000: No effect on the start of packet. 001: If both the first two nibbles of the 10/100 packet are not 0x5, a byte of 0x55 must be prefixed to the output, otherwise there will be no effect on the start of packet. 010: If both the first two nibbles of the 10/100 packet are not 0x5, a byte of 0x55 must be prefixed to the output. An additional byte of 0x55 must be prefixed to the output if the next two nibbles are also not 0x5. 011–111: Reserved.	001
7	MAC SerDes autonegotiation enable	R/W	Sticky bit. 1: MAC SerDes ANEG enable.	0
6	SerDes polarity at input of MAC	R/W	1: Invert polarity of signal received at input of MAC.	0
5	SerDes polarity at output of MAC	R/W	1: Invert polarity of signal at output of MAC.	
4	Fast link status enable	R/W	1: Use fast link fail indication as link status indication to MAC SerDes. 0: Use normal link status indication to MAC SerDes.	0
3	Reserved	R/W	Reserved.	0
2:0	Reserved	RO	Reserved.	

4.5.2 MAC SerDes PCS Status

The register at address 17E3 consists of the bits that provide status from the MAC SerDes PCS block. The following table shows the settings available.

Table 88 • **MAC SerDes PCS Status, Address 17E3 (0x11)**

Bit	Name	Access	Description
15:13	Reserved	RO	Reserved
12	SGMII alignment error	RO	1: RGMII/SGMII alignment error occurred
11	MAC interface LP autonegotiation restart	RO	1: MAC interface link partner autonegotiation restart request occurred
10	Reserved	RO	Reserved
9:8	MAC remote fault	RO	01, 10, and 11: Remote fault detected from MAC 00: No remote fault detected from MAC
7	Asymmetric pause advertisement	RO	1: Asymmetric pause advertised by MAC
6	Symmetric pause advertisement	RO	1: Symmetric pause advertised by MAC
5	Full duplex advertisement	RO	1: Full duplex advertised by MAC
4	Half duplex advertisement	RO	1: Half duplex advertised by MAC

Table 88 • **MAC SerDes PCS Status, Address 17E3 (0x11) *(continued)***

Bit	Name	Access	Description
3	MAC interface LP autonegotiation capable	RO	1: MAC interface link partner autonegotiation capable
2	MAC interface link status	RO	1: MAC interface link status connected
1	MAC interface autonegotiation complete	RO	1: MAC interface autonegotiation complete
0	MAC comma detect	RO	1: Comma currently detected 0: comma currently not detected

4.5.3 MAC SerDes Clause 37 Advertised Ability

The register at address 18E3 consists of the bits that provide access to and control over MAC SerDes Clause 37 advertised ability. The following table shows the settings available.

Table 89 • **MAC SerDes Cl37 Advertised Ability, Address 18E3 (0x12)**

Bit	Name	Access	Description	Default
15:0	MAC SerDes advertised ability	R/W	Current configuration code word being advertised (this register is read/write if 16E3.11 = 1)	0x01E0

4.5.4 MAC SerDes Clause 37 Link Partner Ability

The register at address 19E3 consists of the bits that provide status of the MAC SerDes link partner's Clause 37 advertised ability. The following table shows the settings available.

Table 90 • **MAC SerDes Cl37 LP Ability, Address 19E3 (0x13)**

Bit	Name	Access	Description
15:0	MAC SerDes LP ability	RO	Last configuration code word received from link partner

4.5.5 MAC SerDes Status

The register at address 20E3 consists of the bits that provide access to MAC SerDes status. The following table shows the settings available.

Table 91 • **MAC SerDes Status, Address 20E3 (0x14)**

Bit	Name	Access	Description
15	Reserved	RO	Reserved
14	MAC comma detect	RO	Super-sticky bit. Cleared upon SW reset. 1: Comma detected 0: Comma not detected
13	QSGMII sync status	RO	
12:0	Reserved	RO	Reserved

4.5.6 Media SerDes Transmit Good Packet Counter

The register at address 21E3 consists of the bits that provide status of the media SerDes transmit good packet counter. The following table shows the settings available.

Table 92 • **Media SerDes Tx Good Packet Counter, Address 21E3 (0x15)**

Bit	Name	Access	Description
15	Tx good packet counter active	RO	1: Transmit good packet counter active
14	Reserved	RO	Reserved
13:0	Tx good packet count	RO	Transmit good packet count modulo 10000

4.5.7 Media SerDes Transmit CRC Error Counter

The register at address 22E3 consists of the bits that provide status of the media SerDes transmit packet count that had a CRC error. The following table shows the settings available.

Table 93 • **Media SerDes Tx CRC Error Counter, Address 22E3 (0x16)**

Bit	Name	Access	Description
15:8	Reserved	RO	Reserved
7:0	Tx CRC packet count	RO	Transmit CRC packet count (saturates at 255)

4.5.8 Media SerDes PCS Control

The register at address 23E3 consists of the bits that provide access to and control over Media SerDes PCS control. The following table shows the settings available.

Table 94 • **Media SerDes PCS Control, Address 23E3 (0x17)**

Bit	Name	Access	Description	Default
15:14	Reserved	RO	Reserved	
13	Media interface autonegotiation parallel-detection	R/W	Sticky bit. 1: SerDes media autonegotiation parallel detect enabled	0
12	Reserved	RO	Reserved	
11	Force advertised ability	R/W	1: Force 16-bit advertised ability from register 25E3.15:0	0
10:7	Reserved	RO	Reserved	
6	Polarity reversal input		Media SerDes polarity reversal input 0: No polarity reversal (default) 1: Polarity reversed	0
5	Polarity reversal output		Media SerDes polarity reversal output 0: No polarity reversal (default) 1: Polarity reversed	0
4:0	Reserved	RO	Reserved	

4.5.9 Media SerDes PCS Status

The register at address 24E3 consists of the bits that provide status of the Media SerDes PCS block. The following table shows the settings available.

Table 95 • **Media SerDes PCS Status, Address 24E3 (0x18)**

Bit	Name	Access	Description
15:14	Reserved	RO	Reserved
13	SerDes protocol transfer	RO	100 Mb or 100BASE-FX link status
12	SerDes protocol transfer	RO	10 Mb link status
11	Media interface link partner autonegotiation restart	RO	1: Media interface link partner autonegotiation restart request occurred
10	Reserved	RO	Reserved
9:8	Remote fault detected	RO	01, 10, 11: Remote fault detected from link partner
7	Link partner asymmetric pause	RO	1: Asymmetric pause advertised by link partner
6	Link partner symmetric pause	RO	1: Symmetric pause advertised by link partner
5	Link partner full duplex advertisement	RO	1: Full duplex advertised by link partner
4	Link partner half duplex advertisement	RO	1: Half duplex advertised by link partner
3	Link partner autonegotiation capable	RO	1: Media interface link partner autonegotiation capable
2	Media interface link status	RO	1: Media interface link status
1	Media interface autonegotiation complete	RO	1: Media interface autonegotiation complete
0	Reserved		Reserved

4.5.10 Media SerDes Clause 37 Advertised Ability

The register at address 25E3 consists of the bits that provide access to and control over Media SerDes Clause 37 advertised ability. The following table shows the settings available.

Table 96 • **Media SerDes Cl37 Advertised Ability, Address 25E3 (0x19)**

Bit	Name	Access	Description	Default
15:0	Media SerDes advertised ability	R/W	Current configuration code word being advertised. This register is read/write when 23E3.11 = 1.	0x0000

4.5.11 Media SerDes Clause 37 Link Partner Ability

The register at address 26E3 consists of the bits that provide status of the media SerDes link partner's Clause 37 advertised ability. The following table shows the settings available.

Table 97 • **MAC SerDes Cl37 LP Ability, Address 26E3 (0x1A)**

Bit	Name	Access	Description
15:0	Media SerDes LP ability	RO	Last configuration code word received from link partner

4.5.12 Media SerDes Status

The register at address 27E3 consists of the bits that provide access to Media SerDes status. The following table shows the settings available.

Table 98 • **Media SerDes Status, Address 27E3 (0x1B)**

Bit	Name	Access	Description
15	K28.5 comma realignment	RO	Self-clearing bit. 1: K28.5 comma re-alignment has occurred
14	Signal detect	RO	Self-clearing bit. Sticky bit. 1: SerDes media signal detect
13:0	Reserved	RO	Reserved

4.5.13 Fiber Media CRC Good Counter

Register 28E3 makes it possible to read the contents of the CRC good counter for packets that are received on the Fiber media interface; the number of CRC routines that have executed successfully. The following table shows the expected readouts.

Table 99 • **Fiber Media CRC Good Counter, Address 28E3 (0x1C)**

Bit	Name	Access	Description	Default
15	Packet since last read	RO	Self-clearing bit. 1: Packet received since last read.	0
14	Reserved	RO	Reserved.	
13:0	Fiber media CRC good counter contents	RO	Self-clearing bit. Counter containing the number of packets with valid CRCs. This counter does not saturate and will roll over.	0x000

4.5.14 Fiber Media CRC Error Counter

Register 29E3 makes it possible to read the contents of the CRC error counter for packets that are received on the Fiber media interface. The following table shows the expected readouts.

Table 100 • **Fiber Media CRC Error Counter, Address 29E3 (0x1D)**

Bit	Name	Access	Description	Default
15:8	Reserved	RO	Reserved.	
7:0	Fiber Media CRC error counter	RO	Self-clearing bit. CRC error counter for packets received on the Fiber media interface. The value saturates at 0xFF and subsequently clears when read and restarts count.	0x00

4.6 General Purpose Registers

Accessing the general purpose register space is similar to accessing the extended page registers. Set register 31 to 0x0010. This sets all 32 registers to the general purpose register space.

To restore main register page access, write 0x0000 to register 31.

The following table lists the addresses and register names in the general purpose register page space. These registers are accessible only when the device register 31 is set to 0x0010. All general purpose register bits are super-sticky.

Table 101 • **General Purpose Registers Page Space**

Address	Name
0G–12G	Reserved
13G	LED/SIGDET/GPIO Control
14G	GPIO Control 2
15G	GPIO Input
16G	GPIO Output
17G	GPIO Output Enable
18G	Micro Command
19G	MAC Mode and Fast Link Configuration
20G	Two-Wire Serial MUX Control 1
21G	Two-Wire Serial MUX Control 2
22G	Two-Wire Serial MUX Data Read/Write
23G	Recovered Clock 0 Control
24G	Recovered Clock 1 Control
25G	Enhanced LED Control
26G	Reserved
27G	Reserved
28G	Reserved
29G	Global Interrupt Status
30G	Extended Revision ID

4.6.1 Reserved General Purpose Address Space

The bits in registers 0G to 12G of the general purpose register space are reserved.

4.6.2 SIGDET/GPIO Control

The SIGDET control bits configure the GPIO[1:0]/SIGDET[1:0] pins to function either as signal detect pins for each fiber media port, or as GPIOs. The following table shows the values that can be written.

Table 102 • **SIGDET/GPIO Control, Address 13G (0x0D)**

Bit	Name	Access	Description	Default
15:12	Reserved	RO	Reserved	00
11:10	GPIO5/I2C_SCL_1	R/W	00: SCL for PHY1 01: Reserved 10: Reserved 11: Controlled by MII registers 15G to 17G[1]	00
9:8	GPIO4/I2C_SCL_0	R/W	00: SCL for PHY0 01: Reserved 10: Reserved 11: Controlled by MII registers 15G to 17G[2]	00
7:4	Reserved	RO	Reserved	00

Table 102 • **SIGDET/GPIO Control, Address 13G (0x0D) *(continued)***

Bit	Name	Access	Description	Default
3:2	GPIO1/SIGDET1 control	R/W	00: SIGDET operation 01: Reserved 10: Reserved 11: Controlled by MII registers 15G to 17G	00
1:0	GPIO0/SIGDET0 control	R/W	00: SIGDET operation 01: Reserved 10: Reserved 11: Controlled by MII registers 15G to 17G	00

1. Register 20G bit 1 must be clear in order for this setting to take effect.
2. Register 20G bit 0 must be clear in order for this setting to take effect.

4.6.3 GPIO Control 2

The GPIO control 2 register configures the functionality of the COMA_MODE and 1588 control input pins, and provides control for possible GPIO pin options.

Table 103 • **GPIO Control 2, Address 14G (0x0E)**

Bit	Name	Access	Description	Default
15:14	GPIO12/1588_SPI_CS and GPIO13/1588_SPI_DO	R/W	GPIO12/1588_SPI_CS and GPIO13/1588_SPI_DO control. 00: 1588_SPI_CS/1588_SPI_DO operation. 01: Reserved. 10: Reserved. 11: GPIO12/GPIO13 operation. Controlled by MII registers 15G to 17G.	
13	COMA_MODE output enable (active low)	R/W	1: COMA_MODE pin is an input. 0: COMA_MODE pin is an output.	1
12	COMA_MODE output data	R/W	Value to output on the COMA_MODE pin when it is configured as an output.	0
11	COMA_MODE input data	RO	Data read from the COMA_MODE pin.	
10	Tri-state enable for two-wire serial bus	R/W	1: Tri-states two-wire serial bus output signals instead of driving them high. This allows those signals to be pulled above VDD25 using an external pull-up resistor. 0: Drive two-wire serial bus output signals to high and low values as appropriate.	1
9	Tri-state enable for LEDs	R/W	1: Tri-state LED output signals instead of driving them high. This allows the signals to be pulled above V_{DDIO} using an external pull-up resistor. 0: Drive LED bus output signals to high and low values.	1
8	Reserved	RO	Reserved	0
7:6	GPIO11	R/W	GPIO11 control. 00: Reserved 01: Reserved 10: Reserved 11: Controlled by MII registers 15G to 17G	

Table 103 • **GPIO Control 2, Address 14G (0x0E)** ***(continued)***

Bit	Name	Access	Description	Default
5:4	GPIO10/1588_LOAD_SAVE	R/W	GPIO10/1588_LOAD_SAVE control. 00: 1588_LOAD_SAVE operation 01: Reserved 10: Reserved 11: Controlled by MII registers 15G to 17G	
3:2	GPIO9/FASTLINK_FAIL	R/W	GPIO9/FASTLINK_FAIL control. 00: FASTLINK_FAIL operation 01: Reserved 10: Reserved 11: Controlled by MII registers 15G to 17G	
1:0	GPIO8/I2C_SDA	R/W	GPIO8/I2C_SDA control. 00: I2C_SDA operation 01: Reserved 10: Reserved 11: Controlled by MII registers 15G to 17G	

4.6.4 GPIO Input

The input register contains information about the input to the device GPIO pins. Read from this register to access the data on the device GPIO pins. The following table shows the readout you can expect.

Table 104 • **GPIO Input, Address 15G (0x0F)**

Bit	Name	Access	Description	Default
15:14	Reserved	RO	Reserved	00
13	GPIO13/1588_SPI_DO	RO	GPIO13/1588_SPI_DO input	1
12	GPIO12/1588_SPI_CS	RO	GPIO12/1588_SPI_CS input	1
11	GPIO11	RO	GPIO11 input	0
10	GPIO10/1588_LOAD_SAVE	RO	GPIO10/1588_LOAD_SAVE input	0
9	GPIO9/FASTLINK_FAIL	RO	GPIO9/FASTLINK_FAIL input	1
8	GPIO8/I2C_SDA	RO	GPIO8/I2C_SDA input	1
7:6	Reserved	RO	Reserved	0
5	GPIO5/I2C_SCL_1	RO	GPIO5/I2C_SCL_1 input	1
4	GPIO4/I2C_SCL_0	RO	GPIO4/I2C_SCL_0 input	1
3:2	Reserved	RO	Reserved	
1	GPIO1/SIGDET1	RO	GPIO1/SIGDET1 input	1
0	GPIO0/SIGDET0	RO	GPIO0/SIGDET0 input	0

4.6.5 GPIO Output

The output register allows you to access and control the output from the device GPIO pins. The following table shows the values you can write.

Table 105 • **GPIO Output, Address 16G (0x10)**

Bit	Name	Access	Description	Default
15:14	Reserved	RO	Reserved	
13	GPIO13/1588_SPI_DO	R/W	GPIO13/1588_SPI_DO output	0

Table 105 • **GPIO Output, Address 16G (0x10) *(continued)***

Bit	Name	Access	Description	Default
12	GPIO12/1588_SPI_CS	R/W	GPIO12/1588_SPI_CS output	0
11	GPIO11	R/W	GPIO11 output	0
10	GPIO10/1588_LOAD_SAVE	R/W	GPIO10/1588_LOAD_SAVE output	0
9	GPIO9/FASTLINK_FAIL	R/W	GPIO9/FASTLINK_FAIL output	0
8	GPIO8/I2C_SDA	R/W	GPIO8/I2C_SDA output	0
7:6	Reserved	RO	Reserved	0
5	GPIO5/I2C_SCL_1	R/W	GPIO5/I2C_SCL_1 output	0
4	GPIO4/I2C_SCL_0	R/W	GPIO4/I2C_SCL_0 output	0
3:2	Reserved	RO	Reserved	
1	GPIO1/SIGDET1	R/W	GPIO1/SIGDET1 output	0
0	GPIO0/SIGDET0	R/W	GPIO0/SIGDET0 output	0

4.6.6 GPIO Pin Configuration

Register 17G in the GPIO register space controls whether a particular GPIO pin functions as an input or an output. The following table shows the settings available.

Table 106 • **GPIO Input/Output Configuration, Address 17G (0x11)**

Bit	Name	Access	Description	Default
15:14	Reserved	RO	Reserved	
13	GPIO13/1588_SPI_DO	R/W	GPIO13/1588_SPI_DO output enable	0
12	GPIO12/1588_SPI_CS	R/W	GPIO12/1588_SPI_CS output enable	0
11	GPIO11	R/W	GPIO11 output enable	0
10	GPIO10/1588_LOAD_SAVE	R/W	GPIO10/1588_LOAD_SAVE output enable	0
9	GPIO9/FASTLINK_FAIL	R/W	GPIO9/FASTLINK_FAIL output enable	0
8	GPIO8/I2C_SDA	R/W	GPIO8/I2C_SDA output enable	0
7:6	Reserved	RO	Reserved	0
5	GPIO5/I2C_SCL_1	R/W	GPIO5/I2C_SCL_1 output enable	0
4	GPIO4/I2C_SCL_0	R/W	GPIO4/I2C_SCL_0 output enable	0
3:2	Reserved	RO	Reserved	
1	GPIO1/SIGDET1	R/W	GPIO1/SIGDET1 output enable	0
0	GPIO0/SIGDET0	R/W	GPIO0/SIGDET0 output	0

4.6.7 Microprocessor Command

Register 18G is a command register. Bit 15 tells the internal processor to execute the command. When bit 15 is cleared the command has completed. Software needs to wait until bit 15 = 0 before proceeding with the next PHY register access. Bit 14 = 1 typically indicates an error condition where the squelch patch was not loaded. Use the following steps to execute the command:

1. Write desired command
2. Check bit 15 (move existing text)
3. Check bit 14 (if set, then error)

Commands may take up to 25 ms to complete before bit 15 changes to 0.

Note: All MAC interfaces must be the same — all QSGMII, RGMII, or SGMII.

Table 107 • **Microprocessor Command Register, Address 18G**

Command	Setting
Enable 2 ports MAC SGMII	0x80F0
Enable 2 ports MAC 1/2 QSGMII	0x80E0
QSGMII transmitter control(1)	
1588 initialization(2)	0x801A
Enable 2 ports Media 1000BASE-X	0x8FC1(3)
Enable 2 ports Media 100BASE-FX	0x8FD1(3)

1. Contact your Microsemi representative for an initialization script that greatly simplifies the programming of QSGMII transmit controls.
2. Initializes six analyzers in both 1588 IP blocks A and B. This needs to be done after reset and before the 1588 blocks are used.
3. The “F” in the command has a bit representing each of the four PHYs. To exclude a PHY from the configuration, set its bit to 0. For example, the configuration of PHY 3 and PHY 2 to 1000BASE-X would be 1100 or a “C” and the command would be 0x8CC1.

4.6.8 MAC Configuration and Fast Link

Register 19G in the GPIO register space controls the MAC interface mode and the selection of the source PHY for the fast link failure indication. The following table shows the settings available for the GPIO9/FASTLINK-FAIL pin.

Table 108 • **MAC Configuration and Fast Link Register, Address 19G (0x13)**

Bit	Name	Access	Description	Default
15:14	MAC configuration	R/W	Select MAC interface mode 00: SGMII 01: QSGMII 10: RGMII 11: Reserved	00
13:4	Reserved	RO	Reserved	
3:0	Fast link failure port setting	R/W	Select fast link failure PHY source 0000: Port0 0001: Port1 0010: Reserved 0011: Reserved 1100–1111: Output disabled	0xF

4.6.9 Two-Wire Serial MUX Control 1

The following table shows the settings available to control the integrated two-wire serial MUX.

Table 109 • **Two-Wire Serial MUX Control 1, Address 20G (0x14)**

Bit	Name	Access	Description	Default
15:9	Two-wire serial device address	R/W	Top 7 bits of the 8-bit address sent out on the two wire serial stream. The bottom bit is the read/write signal, which is controlled by register 21G, bit 8. SFPs use 0xA0.	0xA0

Table 109 • **Two-Wire Serial MUX Control 1, Address 20G (0x14) *(continued)***

Bit	Name	Access	Description	Default
8:6	Reserved	RO	Reserved.	
5:4	Two-wire serial SCL clock frequency	R/W	00: 50 kHz 01: 100 kHz 10: 400 kHz 11: 2 MHz	01
3	Two-wire serial MUX port 3 enable	R/W	1: Enabled. 0: Two-wire serial disabled. Becomes GPIO pin.	0
2	Two-wire serial MUX port 2 enable	R/W	1: Enabled. 0: Two-wire serial disabled. Becomes GPIO pin.	0
1	Two-wire serial MUX port 1 enable	R/W	1: Enabled. 0: Two-wire serial disabled. Becomes GPIO pin.	0
0	Two-wire serial MUX port 0 enable	R/W	1: Enabled. 0: Two-wire serial disabled. Two-wire serial MUX port 0 becomes GPIO pin if serial LED function is enabled, regardless of the settings of this bit.	0

4.6.10 Two-Wire Serial MUX Control 2

Register 21G is used to control the two-wire serial MUX for status and control of two-wire serial slave devices.

Table 110 • **Two-Wire Serial MUX Interface Status and Control, Address 21G (0x15)**

Bit	Name	Access	Description	Default
15	Two-wire serial MUX ready	RO	1: Two-wire serial MUX is ready for read or write	1
14:12	Reserved	RO	Reserved	
11:10	PHY port Address	R/W	Specific VSC8572-02 PHY port being addressed.	00
9	Enable two-wire serial MUX access	R/W	Self-clearing bit. 1: Execute read or write through the two-wire serial MUX based on the settings of register bit 21G.8	0
8	Two-wire serial MUX read or write	R/W	1: Read from two-wire serial MUX 0: Write to two-wire serial MUX	1
7:0	Two-wire serial MUX address	R/W	Sets the address of the two-wire serial MUX used to direct read or write operations.	0x00

4.6.11 Two-Wire Serial MUX Data Read/Write

Register 22G in the extended register space enables access to the two-wire serial MUX.

Table 111 • **Two-Wire Serial MUX Data Read/Write, Address 22G (0x16)**

Bit	Name	Access	Description	Default
15:8	Two-wire serial MUX read data	RO	Eight-bit data read from two-wire serial MUX; requires setting both register 21G.9 and 21G.8 to 1.	
7:0	Two-wire serial MUX write data	R/W	Eight-bit data to be written to two-wire serial MUX.	0x00

4.6.12 Recovered Clock 1 Control

Register 23G in the extended register space controls the functionality of the recovered clock 1 output signal.

Table 112 • **Recovered Clock 1 Control, Address 23G (0x17)**

Bit	Name	Access	Description	Default
15	Enable RCVRDCLK1	R/W	1: Enable recovered clock 1 output 0: Disable recovered clock 1 output	0
14:11	Clock source select	R/W	Select bits for source PHY for recovered clock: 0000: PHY0 0001: PHY1 0100–1111: Reserved	0000
10:8	Clock frequency select	R/W	Select output clock frequency: 000: 25 MHz output clock 001: 125 MHz output clock 010: 31.25 MHz output clock 011–111: Reserved	000
7:6	Reserved	RO	Reserved.	
5:4	Clock squelch level	R/W	Select clock squelch level 00: Automatically squelch clock to low when the link is not up, is unstable, is up in a mode that does not support the generation of a recovered clock (1000BASE-T master or 10BASE-T), or is up in EEE mode (100BASE-TX or 1000BASE-T slave). 01: Same as 00 except that the clock is also generated in 1000BASE-T master and 10BASE-T link-up modes. This mode also generates a recovered clock output in EEE mode during reception of LP_IDLE. 10: Squelch only when the link is not up. 11: Disable clock squelch. **Note:** A clock from the SerDes or Cu PHY will be output on the recovered clock output in this mode when the link is down. When the CLK_SQUELCH_IN pin is set high, it squelches the recovered clocks regardless of bit settings.	
3	Reserved	RO	Reserved.	
2:0	Clock selection for specified PHY	R/W	000: Serial media recovered clock 001: Copper PHY recovered clock 010: Copper PHY transmitter TCLK 011–111: Reserved	000

4.6.13 Recovered Clock 2 Control

Register 24G in the extended register space controls the functionality of the recovered clock 2 output signal.

Table 113 • **Recovered Clock 2 Control, Address 24G (0x18)**

Bit	Name	Access	Description	Default
15	Enable RCVRDCLK2	R/W	Enable recovered clock 2 output	0
14:11	Clock source select	R/W	Select bits for source PHY for recovered clock: 0000: PHY0 0001: PHY1 0100–1111: Reserved	0000
10:8	Clock frequency select	R/W	Select output clock frequency: 000: 25 MHz output clock 001: 125 MHz output clock 010: 31.25 MHz output clock 011–111: Reserved	000
7:6	Reserved	RO	Reserved	
5:4	Clock squelch level	R/W	Select clock squelch level: 00: Automatically squelch clock to low when the link is not up, is unstable, is up in a mode that does not support the generation of a recovered clock (1000BASE-T master or 10BASE-T), or is up in EEE mode (100BASE-TX or 1000BASE-T slave). 01: Same as 00 except that the clock is also generated in 1000BASE-T master and 10BASE-T link-up modes. This mode also generates a recovered clock output in EEE mode during reception of LP_IDLE 10: Squelch only when the link is not up 11: Disable clock squelch. **Note:** A clock from the SerDes or Cu PHY will be output on the recovered clock output in this mode when the link is down. **Note:** A clock from the SerDes or Cu PHY will be output on the recovered clock output in this mode when the link is down. When the CLK_SQUELCH_IN pin is set high, it squelches the recovered clocks regardless of bit settings.	
3	Reserved	RO	Reserved	
2:0	Clock selection for specified PHY	R/W	000: Serial media recovered clock 001: Copper PHY recovered clock 010–111: Reserved	000

4.6.14 Enhanced LED Control

The following table contains the bits to control advanced functionality of the parallel and serial LED signals.

Table 114 • **Enhanced LED Control, Address 25G (0x19)**

Bit	Name	Access	Description	Default
15:8	LED pulsing duty cycle control	R/W	Programmable control for LED pulsing duty cycle when bit 30.12 is set to 1. Valid settings are between 0 and 198. A setting of 0 corresponds to a 0.5% duty cycle and 198 corresponds to a 99.5% duty cycle. Intermediate values change the duty cycle in 0.5% increments	00
7	Port 1 enhanced serial LED output enable	R/W	Enable the enhanced serial LED output functionality for port 1 LED pins. 1: Enhanced serial LED outputs 0: Normal function	0
6	Port 0 enhanced serial LED output enable	R/W	Enable the enhanced serial LED output functionality for port 0 LED pins. 1: Enhanced serial LED outputs 0: Normal function	0
5:3	Serial LED frame rate selection	R/W	Select frame rate of serial LED stream 000: 2500 Hz frame rate 001: 1000 Hz frame rate 010: 500 Hz frame rate 011: 250 Hz frame rate 100: 200 Hz frame rate 101: 125 Hz frame rate 110: 40 Hz frame rate 111: Output basic serial LED stream See Table 30, page 79.	
2:1	Serial LED select	R/W	Select which LEDs from each PHY to enable on the serial stream 00: Enable all four LEDs of each PHY 01: Enable LEDs 2, 1 and 0 of each PHY 10: Enable LEDs 1 and 0 of each PHY 11: Enable LED 0 of each PHY	00
0	LED port swapping	R/W	See LED Port Swapping, page 79.	

4.6.15 Global Interrupt Status

The following table contains the interrupt status from the various sources to indicate which one caused that last interrupt on the pin.

Table 115 • **Global Interrupt Status, Address 29G (0x1D)**

Bit	Name	Access	Default	Description
15:10	Reserved	RO	000111	Reserved
9	PHY1 1588[(1)]	RO	1	PHY 1 1588 interrupt source indication 0: PHY1 1588 caused the interrupt 1: PHY1 1588 did not cause the interrupt

Table 115 • **Global Interrupt Status, Address 29G (0x1D) *(continued)***

Bit	Name	Access	Default	Description
8	PHY0 1588[1]	RO	1	PHY 0 1588 interrupt source indication 0: PHY0 1588 caused the interrupt 1: PHY0 1588 did not cause the interrupt
7:2	Reserved	R	111111	Reserved
1	PHY1 interrupt source[1]	RO	1	PHY1 interrupt source indication 0: PHY1 caused the interrupt 1: PHY1 did not cause the interrupt
0	PHY0 interrupt source[1]	RO	1	PHY0 interrupt source indication 0: PHY0 caused the interrupt 1: PHY0 did not cause the interrupt

1. This bit is set to 1 when the corresponding PHY's Interrupt Status register 26 (0x1A) is read.

For information about 1588 IP register access, see Accessing 1588 IP Registers, page 72.

4.6.16 Extended Revision ID

The following table lists the extended revision ID information.

Table 116 • **Extended Revision ID, Address 30G (0x1E)**

Bit	Name	Access	Default	Description
15:1	Reserved	RO	0x0000	Reserved
0	Ext Rev ID	RO	0x1	Revision E

4.7 Clause 45 Registers to Support Energy Efficient Ethernet and 802.3bf

This section describes the Clause 45 registers that are required to support energy efficient Ethernet. Access to these registers is through the IEEE standard registers 13 and 14 (MMD access control and MMD data or address registers) as described in section 4.2.11 and 4.2.12.

The following table lists the addresses and register names in the Clause 45 register page space. When the link is down, 0 is the value returned for the x.180x addresses.

Table 117 • **Clause 45 Registers Page Space**

Address	Name
1.1801	Tx maximum delay through PHY (PMA/PMD/PCS, until 1588 block)
1.1803	Tx minimum delay through PHY (PMA/PMD/PCS, until 1588 block)
1.1805	Rx maximum delay through PHY (PMA/PMD/PCS, until 1588 block)
1.1807	Rx minimum delay through PHY (PMA/PMD/PCS, until 1588 block)
3.1	PCS status 1
3.1801	Tx maximum delay through 1588
3.1803	Tx minimum delay through 1588
3.1805	Rx maximum delay through 1588
3.1807	Rx minimum delay through 1588
3.20	EEE capability
3.22	EEE wake error counter

Table 117 • **Clause 45 Registers Page Space *(continued)***

Address	Name
4.1801	Tx maximum delay through xMII (RGMII, SGMII, QSGMII, including FIFO variations)
4.1803	Tx minimum delay through xMII (RGMII, SGMII, QSGMII, including FIFO variations)
4.1805	Rx maximum delay through xMII (RGMII, SGMII, QSGMII, including FIFO variations)
4.1807	Rx minimum delay through xMII (RGMII, SGMII, QSGMII, including FIFO variations)
7.60	EEE advertisement
7.61	EEE link partner advertisement

4.7.1 PCS Status 1

The bits in the PCS Status 1 register provide a status of the EEE operation from the PCS for the link that is currently active.

Table 118 • **PCS Status 1, Address 3.1**

Bit	Name	Access	Description
15:12	Reserved	RO	Reserved
11	Tx LPI received	RO/LH	1: Tx PCS has received LPI 0: LPI not received
10	Rx LPI received	RO/LH	1: Rx PCS has received LPI 0: LPI not received
9	Tx LPI indication	RO	1: Tx PCS is currently receiving LPI 0: PCS is not currently receiving LPI
8	Rx LPI indication	RO	1: Rx PCS is currently receiving LPI 0: PCS is not currently receiving LPI
7:3	Reserved	RO	Reserved
2	PCS receive link status	RO	1: PCS receive link up 0: PCS receive link down
1:0	Reserved	RO	Reserved

4.7.2 EEE Capability

This register is used to indicate the capability of the PCS to support EEE functions for each PHY type. The following table shows the bit assignments for the EEE capability register.

Table 119 • **EEE Capability, Address 3.20**

Bit	Name	Access	Description
15:3	Reserved	RO	Reserved
2	1000BASE-T EEE	RO	1: EEE is supported for 1000BASE-T 0: EEE is not supported for 1000BASE-T
1	100BASE-TX EEE	RO	1: EEE is supported for 100BASE-TX 0: EEE is not supported for 100BASE-TX
0	Reserved	RO	Reserved

4.7.3 EEE Wake Error Counter

This register is used by PHY types that support EEE to count wake time faults where the PHY fails to complete its normal wake sequence within the time required for the specific PHY type. The definition of

the fault event to be counted is defined for each PHY and can occur during a refresh or a wakeup as defined by the PHY. This 16-bit counter is reset to all zeros when the EEE wake error counter is read or when the PHY undergoes hardware or software reset.

Table 120 • **EEE Wake Error Counter, Address 3.22**

Bit	Name	Access	Description
15:0	Wake error counter	RO	Count of wake time faults for a PHY

4.7.4 EEE Advertisement

This register defines the EEE advertisement that is sent in the unformatted next page following a EEE technology message code. The following table shows the bit assignments for the EEE advertisement register.

Table 121 • **EEE Advertisement, Address 7.60**

Bit	Name	Access	Description	Default
15:3	Reserved	RO	Reserved	
2	1000BASE-T EEE	R/W	1: Advertise that the 1000BASE-T has EEE capability 0: Do not advertise that the 1000BASE-T has EEE capability	0
1	100BASE-TX EEE	R/W	1: Advertise that the 100BASE-TX has EEE capability 0: Do not advertise that the 100BASE-TX has EEE capability	0
0	Reserved	RO	Reserved	

4.7.5 EEE Link Partner Advertisement

All the bits in the EEE LP Advertisement register are read only. A write to the EEE LP advertisement register has no effect. When the AN process has been completed, this register will reflect the contents of the link partner's EEE advertisement register. The following table shows the bit assignments for the EEE advertisement register.

Table 122 • **EEE Advertisement, Address 7.61**

Bit	Name	Access	Description
15:3	Reserved	RO	Reserved
2	1000BASE-T EEE	RO	1: Link partner is advertising EEE capability for 1000BASE-T 0: Link partner is not advertising EEE capability for 1000BASE-T
1	100BASE-TX EEE	RO	1: Link partner is advertising EEE capability for 100BASE-TX 0: Link partner is not advertising EEE capability for 100BASE-TX
0	Reserved	RO	Reserved

The following table shows the bit assignments for the 802.3bf registers. When the link is down, 0 is the value returned. cl45reg1_1801 would be device address of 1 and register address of 1801.

Table 123 • **802.3bf Registers**

Register	Name	Function
1.1801	cl45reg1_1801_val[15:0]	Tx maximum delay through PHY (PMA/PMD/PCS, until 1588 block)
1.1803	cl45reg1_1803_val[15:0]	Tx minimum delay through PHY (PMA/PMD/PCS, until 1588 block)
1.1805	cl45reg1_1805_val[15:0]	Rx maximum delay through PHY (PMA/PMD/PCS, until 1588 block)
1.1807	cl45reg1_1807_val[15:0]	Rx minimum delay through PHY (PMA/PMD/PCS, until 1588 block)
3.1801	cl45reg3_1801_val[15:0]	Tx maximum delay through 1588
3.1803	cl45reg3_1803_val[15:0]	Tx minimum delay through 1588
3.1805	cl45reg3_1805_val[15:0]	Rx maximum delay through 1588
3.1807	cl45reg3_1807_val[15:0]	Rx minimum delay through 1588
4.1801	cl45reg4_1801_val[15:0]	Tx maximum delay through xMII (RGMII, SGMII, QSGMII, including FIFO variations)
4.1803	cl45reg4_1803_val[15:0]	Tx minimum delay through xMII (RGMII, SGMII, QSGMII, including FIFO variations)
4.1805	cl45reg4_1805_val[15:0]	Rx maximum delay through xMII (RGMII, SGMII, QSGMII, including FIFO variations)
4.1807	cl45reg4_1807_val[15:0]	Rx minimum delay through xMII (RGMII, SGMII, QSGMII, including FIFO variations)

4.8 1588 IP Registers

This section lists the 1588 IP registers.

4.9 1588 IP Block Configuration and Status Registers

This section lists the overviews for the 1588 IP block configuration and status registers. The registers documented in this section are present in both channels.

Note: For more information about accessing the 1588 IP registers, see Accessing 1588 IP Registers, page 72.

Table 124 • **IP_1588_TOP_CFG_STAT**

Address	Name	Details
0x00	INTERFACE_CTL	Interface Control, page 145
0x01	ANALYZER_MODE	Analyzer Mode, page 146
0x02	SPARE_REGISTER	Spare Scratchpad, page 146

Table 125 • **IP_1588_LTC**

Address	Name	Details
0x10	LTC_CTRL	LTC Control, page 146
0x11	LTC_LOAD_SEC_H	LTC Load Seconds (High), page 148
0x12	LTC_LOAD_SEC_L	LTC Load Seconds (Low), page 148
0x13	LTC_LOAD_NS	LTC Load Nanoseconds, page 148
0x14	LTC_SAVED_SEC_H	LTC Saved Seconds (High), page 148
0x15	LTC_SAVED_SEC_L	LTC Saved Seconds (Low), page 148
0x16	LTC_SAVED_NS	LTC Saved Nanoseconds, page 149
0x17	LTC_SEQUENCE	LTC Sequence Configuration, page 149
0x18	LTC_SEQ	LTC Sequence Configuration, page 149
0x1A	LTC_AUTO_ADJUST	LTC Auto Adjustment, page 150

Table 126 • **TS_FIFO_SI**

Address	Name	Details
0x20	TS_FIFO_SI_CFG	Timestamp FIFO Serial Interface Configuration, page 150
0x21	TS_FIFO_SI_TX_CNT	Transmitted Timestamp Count, page 151

Table 127 • **INGR_PREDICTOR**

Address	Name	Details
0x22	IG_CFG	Ingress Configuration, page 151

Table 128 • **EGR_PREDICTOR**

Address	Name	Details
0x26	EG_CFG	Egress Configuration, page 151

Table 129 • **INGR_IP_1588_CFG_STAT**

Address	Name	Details
0x2D	INGR_INT_STATUS	IP 1588 Interrupt Status, page 152
0x2E	INGR_INT_MASK	IP 1588 Interrupt Mask, page 153
0x2F	INGR_SPARE_REGISTER	Spare Scratchpad, page 153

Table 130 • **INGR_IP_1588_TSP**

Address	Name	Details
0x35	INGR_TSP_CTRL	TSP Control, page 154
0x36	INGR_TSP_STAT	TSP Status, page 154
0x37	INGR_LOCAL_LATENCY	Local Latency, page 154
0x38	INGR_PATH_DELAY	Path Delay, page 155
0x39	INGR_DELAY_ASYMMETRY	DelayAsymmetry, page 155

Table 131 • **INGR_IP_1588_DF**

Address	Name	Details
0x3A	INGR_DF_CTRL	Configuration and Control for the Delay FIFO, page 155

Table 132 • **INGR_IP_1588_RW**

Address	Name	Details
0x44	INGR_RW_CTRL	Rewriter Configuration and Control, page 156
0x45	INGR_RW_MODFRM_CNT	Count of Modified Frames, page 156
0x46	INGR_RW_FCS_ERR_CNT	Count of FCS Errors, page 156
0x47	INGR_RW_PREAMBLE_ERR_CNT	Count of the Number of Preamble Errors, page 157

Table 133 • **EGR_IP_1588_CFG_STAT**

Address	Name	Details
0x4D	EGR_INT_STATUS	IP 1588 Interrupt Status, page 157

Table 133 • **EGR_IP_1588_CFG_STAT *(continued)***

Address	Name	Details
0x4E	EGR_INT_MASK	IP 1588 Interrupt Mask, page 158
0x4F	EGR_SPARE_REGISTER	Spare Scratchpad, page 159

Table 134 • **EGR_IP_1588_TSP**

Address	Name	Details
0x55	EGR_TSP_CTRL	TSP Control, page 159
0x56	EGR_TSP_STAT	TSP Status, page 159
0x57	EGR_LOCAL_LATENCY	Local Latency, page 160
0x58	EGR_PATH_DELAY	Path Delay, page 160
0x59	EGR_DELAY_ASYMMETRY	DelayAsymmetry, page 160

Table 135 • **EGR_IP_1588_DF**

Address	Name	Details
0x5A	EGR_DF_CTRL	Configuration and Control for the Delay FIFO, page 161

Table 136 • **EGR_IP_1588_TSFIFO**

Address	Name	Details
0x5B	EGR_TSFIFO_CSR	Timestamp FIFO Configuration and Status, page 161
0x5C	EGR_TSFIFO_0	Data Value from the Timestamp FIFO, page 161
0x5D	EGR_TSFIFO_1	Data Value from the Timestamp FIFO, page 162
0x5E	EGR_TSFIFO_2	Data Value from the Timestamp FIFO, page 163
0x5F	EGR_TSFIFO_3	Data Value from the Timestamp FIFO, page 163
0x60	EGR_TSFIFO_4	Data Value from the Timestamp FIFO, page 163
0x61	EGR_TSFIFO_5	Data Value from the Timestamp FIFO, page 163
0x62	EGR_TSFIFO_6	Data Value from the Timestamp FIFO, page 163
0x63	EGR_TSFIFO_DROP_CNT	Count of Dropped Timestamps, page 164

Table 137 • **EGR_IP_1588_RW**

Address	Name	Details
0x64	EGR_RW_CTRL	Rewriter Configuration and Control, page 164
0x65	EGR_RW_MODFRM_CNT	Count of Modified Frames, page 164
0x66	EGR_RW_FCS_ERR_CNT	Count of FCS Errors, page 165

Table 137 • **EGR_IP_1588_RW *(continued)***

Address	Name	Details
0x67	EGR_RW_PREAMBLE_ERR_CNT	Count of the Number of Preamble Errors, page 165

Table 138 • **INGR_IP_1588_DEBUG_REGISTERS**

Address	Name	Details
0x9F	INGR_SW_POP_FIFO	1588 IP Ingress Debug Register, page 165

Table 139 • **EGR_IP_1588_DEBUG_REGISTERS**

Address	Name	Details
0xC0	EGR_SW_POP_FIFO	1588 IP Egress Debug Registers, page 166

4.10 1588 IP Control and Status Registers

This section provides information about the 1588 IP control and status registers.

4.10.1 Interface Control

Short Name: INTERFACE_CTL

Address: 0x00

Table 140 • **Interface Control Register**

Bit	Name	Description	Access	Default
6	CLK_ENA	Enables the data path clocks in the 1588 IP block. The 1588 logic, including all configuration registers, is held in a reset state when the clocks are disabled. 0: Clocks Disabled 1: Clocks Enabled	R/W	0x0
2	BYPASS	When 1, the 1588 IP block is bypassed. This is the default state. Changing this bit to 0 will allow 1588 processed data to flow out of the block. This bit is internally registered so that it only takes effect during an IDLE period in the data stream. This allows for a more seamless transition from bypass to data passing modes. 0: Data mode 1: Bypass mode Data flows through the bypass data path even if register bit CLK_ENA = 0.	R/W	0x1
1:0	MII_PROTOCOL	Defines the operating mode that the attached PCS block operates in 0: reserved 1: reserved 2: GMII 3: reserved **Note:** These bits must be set to 0x2	R/W	0x0

4.10.2 Analyzer Mode

Short Name: ANALYZER_MODE

Address: 0x01

Table 141 • **Analyzer Mode Register**

Bit	Name	Description	Access	Default
18:16	ENCAP_FLOW_MODE	Defines how flow matching is performed in each encapsulation engine. For each engine 0: Match any flow 1: Strict matching	R/W	0x0
6:4	EGR_ENCAP_ENGINE_ENA	Enables for the egress encapsulation engines. Enable bit 0 & 1 are for the PTP engines and bit 2 is for the OAM engine. For each engine 0: Disabled 1: Enabled	R/W	0x0
2:0	INGR_ENCAP_ENGINE_ENA	Enables for the ingress encapsulation engines. Enable bit 0 & 1 are for the PTP engines and bit 2 is for the OAM engine. For each engine 0: Disabled 1: Enabled	R/W	0x0

4.10.3 Spare Scratchpad

Short Name: SPARE_REGISTER

Address: 0x02

Table 142 • **Spare Scratchpad Register**

Bit	Name	Description	Access	Default
31:0	SPARE_REGISTER	Spare scratchpad register	R/W	0x00000000

4.11 1588 IP Local Time Counter Registers

This section provides information about the 1588 IP local time counter configuration and status registers.

4.11.1 LTC Control

Short Name: LTC_CTRL

Address: 0x10

Table 143 • **LTC Control Register**

Bit	Name	Description	Access	Default
14:12	LTC_CLK_SEL	This field is used to select the clock source for the LTC block. The actual clock mux is external to the IP block, this field merely provides the select lines to the clock mux. These 3 select lines are outputs of the IP block and are not used internally. The single clock signal is then fed to the LTC input pin. The 3 bits allows for one of up to 8 possible clock sources to be selected. 0: External clock (supports 125 MHz, 156.25 MHz, and 250 MHz) 1: Client Rx Clock (QSGMII/SGMII recovered clock), 125 MHz 2: Client Tx Clock, 125 MHz 3: Line Rx Clock, 125 MHz 4: Line Tx Clock, 125 MHz 5: Local reference clock, 250 MHz 6: INVALID 7: INVALID	R/W	0x0
10:6	Reserved	Reserved	RO	
5	Reserved	Reserved	RO	
4	LTC_AUTO_ADJUST_UPDATE	When written to a '1' causes the Local Time Counter to update the automatic adjustment values from the LTC_AUTO_ADJUST register. The current automatic adjustment is reset to start with the new values. Automatically cleared. 0: No change to any previous updates (write), or update has completed (read) 1: Use new values from LTC_AUTO_ADJUST register.	One-shot	0x0
3	LTC_ADD_SUB_1NS_REQ	When written to a '1' causes a request for 1ns to be added or subtracted (depending upon the LTC_ADD_SUB_1NS field) from the Local time. Automatically cleared. 0: No Add/Subtract from local time (write), Bit has auto cleared (read) 1: Add/Subtract 1ns from the local time.	One-shot	0x0
2	LTC_ADD_SUB_1NS	This bit selects whether a write to the LTC_ADD_SUB_1NS_REQ register causes an add or subtract. 0: Subtract 1 ns 1: Add 1ns to the local time.	R/W	0x0
1	LTC_SAVE_ENA	LTC save enable. Enables the chip save pin to save the LTC_SAVE seconds/nanoseconds registers.	R/W	0x0
0	LTC_LOAD_ENA	LTC load enable. Enables the chip load pin to load the LTC_LOAD seconds/nanoseconds registers.	R/W	0x0

4.11.2 LTC Load Seconds (High)

Short Name: LTC_LOAD_SEC_H

Address: 0x11

LTC load seconds (high)

Table 144 • **LTC Load Seconds (High) Register**

Bit	Name	Description	Access	Default
15:0	LTC_LOAD_SEC_H	LTC load seconds (high)	R/W	0x0000

4.11.3 LTC Load Seconds (Low)

Short Name: LTC_LOAD_SEC_L

Address: 0x12

LTC load seconds (low)

Table 145 • **LTC Load Seconds (Low) Register**

Bit	Name	Description	Access	Default
31:0	LTC_LOAD_SEC_L	LTC load seconds (low)	R/W	0x00000000

4.11.4 LTC Load Nanoseconds

Short Name: LTC_LOAD_NS

Address: 0x13

LTC load nanoseconds

Table 146 • **LTC Load Nanoseconds Register**

Bit	Name	Description	Access	Default
31:0	LTC_LOAD_NS	LTC load nanoseconds	R/W	0x00000000

4.11.5 LTC Saved Seconds (High)

Short Name: LTC_SAVED_SEC_H

Address: 0x14

LTC saved seconds (high)

Table 147 • **LTC Saved Seconds (High) Register**

Bit	Name	Description	Access	Default
15:0	LTC_SAVED_SEC_H	LTC saved seconds (high)	R/O	0x0000

4.11.6 LTC Saved Seconds (Low)

Short Name: LTC_SAVED_SEC_L

Address: 0x15

LTC saved seconds (low)

Table 148 • **LTC Saved Seconds (Low) Register**

Bit	Name	Description	Access	Default
31:0	LTC_SAVED_SEC_L	LTC saved seconds (low)	R/O	0x00000000

4.11.7 LTC Saved Nanoseconds

Short Name: LTC_SAVED_NS

Address: 0x16

LTC saved nanoseconds

Table 149 • **LTC Saved Nanoseconds Register**

Bit	Name	Description	Access	Default
31:0	LTC_SAVED_NS	LTC load nanoseconds	R/O	0x00000000

4.11.8 LTC Sequence Configuration

Short Name: LTC_SEQUENCE

Address: 0x17

LTC sequence configuration

Table 150 • **LTC Sequence Configuration Register**

Bit	Name	Description	Access	Default
19:12	Reserved	Must be set to its default.	R/W	0x01
11:8	Reserved	Must be set to its default.	R/W	0x4
7:4	Reserved	Must be set to its default.	R/W	0x4
3:0	LTC_SEQUENCE_A	LTC sequence of increments (nanoseconds)	R/W	0x4

4.11.9 LTC Sequence Configuration

Short Name: LTC_SEQ

Address: 0x18

LTC sequence configuration

Table 151 • **LTC Sequence Configuration Register**

Bit	Name	Description	Access	Default
20	Reserved	Must be set to its default.	R/W	0x1
19	LTC_SEQ_ADD_SUB	LTC sequence correction sign 0: subtract 1ns adjustment 1: add 1ns adjustment	R/W	0x1

Table 151 • **LTC Sequence Configuration Register *(continued)***

Bit	Name	Description	Access	Default
18:0	LTC_SEQ_E	LTC sequence correction (nanoseconds * 1 million) Example for 6.4 ns period (156.25MHz): LTC_SEQUENCE.LTC_SEQUENCE_A = 6 (6 ns) LTC_SEQ.LTC_SEQ_ADD_SUB = 1 (add 1ns) LTC_SEQ.LTC_SEQ_E = 400000 (0.4ns * 1,000,000)	R/W	0x00000

4.11.10 LTC Auto Adjustment

Short Name: LTC_AUTO_ADJUST

Address: 0x1A

LTC auto adjustment

Table 152 • **LTC Auto Adjustment Register**

Bit	Name	Description	Access	Default
31:30	LTC_AUTO_ADD_SUB_1NS	LTC auto adjustment add/subtract 1ns 0,3: No adjustment 1: Adjust by adding 1ns upon rollover. 2: Adjust by subtracting 1ns upon rollover.	R/W	0x0
29:0	LTC_AUTO_ADJUST_NS	LTC auto adjustment rollover (nanoseconds)	R/W	0x00000000

4.12 Timestamp FIFO Serial Interface Registers

This section provides information about the timestamp FIFO serial interface registers.

4.12.1 Timestamp FIFO Serial Interface Configuration

Short Name: TS_FIFO_SI_CFG

Address: 0x20

Polarity and cycle counts are configurable from port 0 only.

Table 153 • **Timestamp FIFO Serial Interface Configuration Register**

Bit	Name	Description	Access	Default
25	SI_CLK_PHA	Timestamp serial interface clock phase control. 0=SPI_CLK falling edge changes output data 1=SPI_CLK rising edge changes output data	R/W	0x0
24	SI_CLK_POL	Timestamp FIFO serial interface clock polarity control. 0=SPI_CLK starts and ends (idles) low 1=SPI_CLK starts and ends (idles) high	R/W	0x0
23:20	SI_EN_DES_CYCS	Number of CSR clock periods SPI_CS negates between writes (deselected). The CSR clock frequency is one-half the XREFCK frequency.	R/W	0x0

Table 153 • **Timestamp FIFO Serial Interface Configuration Register *(continued)***

Bit	Name	Description	Access	Default
10:6	SI_CLK_HI_CYCS	Number of CSR clock periods that the SPI_CLK is high. Registers SI_CLK_HI_CYCS and SI_CLK_LO_CYCS determine the frequency of the SPI_CLK pin when the timestamp FIFO serial interface is enabled. The CSR clock frequency is one-half the XREFCK frequency. Zero is an invalid setting.	R/W	0x02
5:1	SI_CLK_LO_CYCS	Number of CSR clock periods that the SPI_CLK is low. Registers SI_CLK_HI_CYCS and SI_CLK_LO_CYCS determine the frequency of the SPI_CLK pin when the timestamp FIFO serial interface is enabled. The CSR clock frequency is one-half the XREFCK frequency. Zero is an invalid setting.	R/W	0x02
0	TS_FIFO_SI_ENA	When 1, the Timestamp FIFO Serial Interface block is enabled 0=Disabled 1=Enabled	R/W	0x0

4.12.2 Transmitted Timestamp Count

Short Name: TS_FIFO_SI_TX_CNT

Address: 0x21

Counter for the number of timestamps transmitted to the interface.

Table 154 • **Transmitted Timestamp Count Register**

Bit	Name	Description	Access	Default
31:0	TS_FIFO_SI_TX_CNT	Counter value	R/W	0x00000000

4.13 Ingress (Rx) Registers

This section provides information about the ingress registers.

4.13.1 Ingress Configuration

Short Name: IG_CFG

Address: 0x22

Table 155 • **Ingress Configuration Register**

Bit	Name	Description	Access	Default
0	IG_ENABLE	When 1, the Ingress prediction block is enabled 0=Disabled 1=Enabled	R/W	0x0

4.14 Egress (Tx) Registers

This section provides information about the egress registers.

4.14.1 Egress Configuration

Short Name: EG_CFG

Address: 0x26

Table 156 • **Egress Configuration Register**

Bit	Name	Description	Access	Default
0	EG_ENABLE	When 1, the Egress prediction block is enabled. 0=Disabled 1=Enabled	R/W	0x0

4.15 1588 IP Ingress Control and Status Registers

This section provides information about the 1588 IP control and status registers.

4.15.1 IP 1588 Interrupt Status

Short Name: INGR_INT_STATUS

Address: 0x2D

Status sticky conditions for the 1588 IP

Table 157 • **IP 1588 Interrupt Status Register**

Bit	Name	Description	Access	Default
6	INGR_ANALYZER_ERROR_STICKY	Indicates that more than one engine has produced a match 0: No error found 1: Duplicate match found	Sticky	0x0
5	INGR_RW_PREAMBLE_ERR_STICKY	When set, indicates that a preamble that was too short to modify was detected in a PTP frame. Write to 0 to clear. This occurs when the Rewriter needs to shrink the preamble to append a timestamp but cannot because the preamble is too short. A short preamble is any preamble that is less than 8 characters long including the XGMII /S/ character and the ending SFD of 0xD5. Other preamble values are not checked, only the length. 0: No error 1: Preamble too short error	Sticky	0x0
4	INGR_RW_FCS_ERR_STICKY	When set, indicates that an FCS error was detected in a PTP frame. Write to 0 to clear. 0: No error 1: FCS error	Sticky	0x0
3	INGR_TS_LEVEL_STICKY	Reserved	Sticky	0x0
2	INGR_TS_LOADED_STICKY	When set, indicates a timestamp was captured in the Timestamp FIFO. The sticky bit should be reset by writing it to zero. 0: No overflow 1: Overflow	Sticky	0x0
1	INGR_TS_UNDERFLOW_STICKY	When set, indicates an underflow in the Timestamp FIFO. The sticky bit should be reset by writing it to zero. 0: No overflow 1: Overflow	Sticky	0x0

Table 157 • **IP 1588 Interrupt Status Register *(continued)***

Bit	Name	Description	Access	Default
0	INGR_TS_OVERFLOW_STICKY	When set, indicates an overflow in the Timestamp FIFO. The sticky bit should be reset by writing it to zero. 0: No overflow 1: Overflow	Sticky	0x0

4.15.2 IP 1588 Interrupt Mask

Short Name: INGR_INT_MASK

Address: 0x2E

Masks that enable and disable the interrupts

Table 158 • **IP 1588 Interrupt Mask Register**

Bit	Name	Description	Access	Default
6	INGR_ANALYZER_ERROR_MASK	Mask bit for ANALYZER_ERROR_STICKY bit. 0: Interrupt disabled 1: Interrupt enabled	R/W	0x0
5	INGR_RW_PREAMBLE_ERR_MASK	Mask for the RW_PREAMBLE_ERR_STICKY bit. 0: Interrupt disabled 1: Interrupt enabled	R/W	0x0
4	INGR_RW_FCS_ERR_MASK	Mask for the RW_FCS_ERR_STICKY bit. 0: Interrupt disabled 1: Interrupt enabled	R/W	0x0
3	INGR_TS_LEVEL_MASK	Reserved. Do not modify setting.	R/W	0x0
2	INGR_TS_LOADED_MASK	Mask bit for TS_LOADED_STICKY. When 1, the interrupt is enabled. 0: Interrupt disabled 1: Interrupt enabled	R/W	0x0
1	INGR_TS_UNDERFLOW_MASK	Mask bit for TS_UNDERFLOW_STICKY. When 1, the interrupt is enabled. 0: Interrupt disabled 1: Interrupt enabled	R/W	0x0
0	INGR_TS_OVERFLOW_MASK	Mask bit for TS_OVERFLOW_STICKY. When 1, the interrupt is enabled. 0: Interrupt disabled 1: Interrupt enabled	R/W	0x0

4.15.3 Spare Scratchpad

Short Name: INGR_SPARE_REGISTER

Address: 0x2F

Table 159 • **Spare Scratchpad Register**

Bit	Name	Description	Access	Default
31:0	INGR_SPARE_REGISTER	Spare scratchpad register	R/W	0x00000000

4.16 1588 IP Ingress Timestamp Processor Registers

This section provides information about the 1588 IP timestamp processor registers.

4.16.1 TSP Control

Short Name: INGR_TSP_CTRL

Address: 0x35

Table 160 • **TSP Control Register**

Bit	Name	Description	Access	Default
2	INGR_FRACT_NS_MODE	Selects a mode in which the fractional portion of a second (in units of nanoseconds) is used for timestamping. Only the operation of the WRITE_NS, WRITE_NS_P2P, and SUB_ADD PTP commands are affected by the setting of this mode bit. 0: Select the total (summed) nanoseconds for timestamping. 1: Select the fractional portion in nanoseconds for timestamping.	R/W	0x0
1	INGR_SEL_EXT_SOF_IND	Select external pin start of frame indicator. 0: Select internal PCS as the source of SOF 1: Select external pin as the source of SOF.	R/W	0x0
0	INGR_LOAD_DELAYS	One-shot loads Local latency, Path delay, and DelayAsymmetry values into the Timestamp Processor	One-shot	0x0

4.16.2 TSP Status

Short Name: INGR_TSP_STAT

Address: 0x36

Table 161 • **TSP Status Register**

Bit	Name	Description	Access	Default
0	INGR_CF_TOO_BIG_STICKY	Timestamp processor marked a calculated correction field as too big. 0: A calculated correction field that was too big did occur. 1: A calculated correction field that was too big did not occur.	Sticky	0x0

4.16.3 Local Latency

Short Name: INGR_LOCAL_LATENCY

Address: 0x37

Table 162 • **Local Latency Register**

Bit	Name	Description	Access	Default
15:0	INGR_LOCAL_LATENCY	Local latency (nanoseconds) The value programmed in this register is dependent upon the frequency of the clock driving the Local Time Counter (LTC) and upon LAN mode of operation. When in LAN mode and the LTC clock frequency is 250 MHz, set this register to 106. When in LAN mode and the LTC clock frequency is 125 MHz, set this register to 112.	R/W	0x0000

4.16.4 Path Delay

Short Name: INGR_PATH_DELAY

Address: 0x38

Table 163 • **Path Delay Register**

Bit	Name	Description	Access	Default
31:0	INGR_PATH_DELAY	Path delay (nanoseconds)	R/W	0x00000000

4.16.5 DelayAsymmetry

Short Name: INGR_DELAY_ASYMMETRY

Address: 0x39

Table 164 • **DelayAsymmetry Register**

Bit	Name	Description	Access	Default
31:0	INGR_DELAY_ASYMMETRY	DelayAsymmetry (scaled nanoseconds)	R/W	0x00000000

4.17 1588 IP Ingress Delay FIFO Registers

This section provides information about the 1588 delay FIFO registers. The delay FIFO delays the data in a pipeline governed by these settings.

4.17.1 Configuration and Control for the Delay FIFO

Short Name: INGR_DF_CTRL

Address: 0x3A

Table 165 • **Configuration and Control Register for the Delay FIFO**

Bit	Name	Description	Access	Default
4:0	INGR_DF_DEPTH	The index of the register stage in the Delay FIFO that is used for output. The actual delay through the block is one more than the depth. If depth is set to 2, then the delay is 3 clocks as data is taken from stage 2. The depth MUST be greater than 0 (depth of 0 is not allowed). This bit group must be set to 0x0F in the device.	R/W	0x00

4.18 1588 IP Ingress Rewriter Registers

This section provides information about the 1588 IP rewriter registers.

4.18.1 Rewriter Configuration and Control

Short Name: INGR_RW_CTRL

Address: 0x44

Configuration for the Rewriter

Table 166 • **Rewriter Configuration and Control Register**

Bit	Name	Description	Access	Default
4	INGR_RW_REDUCE_PREAMBLE	When set, the 1588 IP will reduce the preamble of ALL incoming frames by 4 bytes to allow a timestamp to be appended to the ingress data frames. This bit must be set along with proper configuration of the Analyzer to ensure proper operation. ** VALID IN INGRESS DIRECTION ONLY ** 0: No preamble modification 1: Reduce preamble by 4 bytes	R/W	0x0
3	INGR_RW_FLAG_VAL	Value to write to the flag bit when it is overwritten. 0: '0' will be written to the flag bit 1: '1' will be written to the flag bit	R/W	0x0
2:0	INGR_RW_FLAG_BIT	Bit offset within a byte of the flag bit which indicates if the frame has been modified or not. Binary number	R/W	0x0

4.18.2 Count of Modified Frames

Short Name: INGR_RW_MODFRM_CNT

Address: 0x45

Table 167 • **Count of Modified Frames Register**

Bit	Name	Description	Access	Default
31:0	INGR_RW_MODFRM_CNT	Count of the number of frames modified by the 1588 IP. The counter wraps. Binary number	R/W	0x00000000

4.18.3 Count of FCS Errors

Short Name: INGR_RW_FCS_ERR_CNT

Address: 0x46

Table 168 • **Count of FCS Errors Register**

Bit	Name	Description	Access	Default
31:0	INGR_RW_FCS_ERR_CNT	Count of the number of FCS errored frames detected by the Rewriter. Binary number	R/W	0x00000000

4.18.4 Count of the Number of Preamble Errors

Short Name: INGR_RW_PREAMBLE_ERR_CNT

Address: 0x47

Table 169 • **Count of the Number of Preamble Errors Register**

Bit	Name	Description	Access	Default
31:0	INGR_RW_PREAMBLE_ERR_CNT	Count of the number of errored preambles detected. The counter wraps. An errored preamble is a preamble that is too short to shrink that is encountered when RW_REDUCE_PREAMBLE is set. Binary number	R/W	0x00000000

4.19 1588 IP Egress Control & Status Registers

This section provides information about the 1588 IP control and status registers.

4.19.1 IP 1588 Interrupt Status

Short Name: EGR_INT_STATUS

Address: 0x4D

Status sticky conditions for the 1588 IP

Table 170 • **IP 1588 Interrupt Status Register**

Bit	Name	Description	Access	Default
6	EGR_ANALYZER_ERROR_STICKY	Indicates that more than one engine has produced a match 0: No error found 1: Duplicate match found	Sticky	0x0
5	EGR_RW_PREAMBLE_ERR_STICKY	When set, indicates that a preamble that was too short to modify was detected in a PTP frame. Write to 0 to clear. This occurs when the Rewriter needs to shrink the preamble to append a timestamp but cannot because the preamble is too short. A short preamble is any preamble that is less than 8 characters long including the XGMII /S/ character and the ending SFD of 0xD5. Other preamble values are not checked, only the length. 0: No error 1: Preamble too short error	Sticky	0x0
4	EGR_RW_FCS_ERR_STICKY	When set, indicates that an FCS error was detected in a PTP frame. Write to 0 to clear. 0: No error 1: FCS error	Sticky	0x0
3	EGR_TS_LEVEL_STICKY	When set, indicates that the level in the Timestamp FIFO has reached the threshold EGR_TS_THRESH. The sticky bit should be reset by writing it to zero. 0: Egress timestamp FIFO threshold not reached 1: Egress timestamp FIFO threshold reached	Sticky	0x0

Table 170 • **IP 1588 Interrupt Status Register *(continued)***

Bit	Name	Description	Access	Default
2	EGR_TS_LOADED_STICKY	When set, indicates a timestamp was captured in the Timestamp FIFO. The sticky bit should be reset by writing it to zero. 0: Egress timestamp FIFO not loaded 1: Egress timestamp FIFO loaded	Sticky	0x0
1	EGR_TS_UNDERFLOW_STICKY	When set, indicates an underflow in the Timestamp FIFO. The sticky bit should be reset by writing it to zero. 0: No underflow 1: Underflow	Sticky	0x0
0	EGR_TS_OVERFLOW_STICKY	When set, indicates an overflow in the Timestamp FIFO. The sticky bit should be reset by writing it to zero. 0: No overflow 1: Overflow	Sticky	0x0

4.19.2 IP 1588 Interrupt Mask

Short Name: EGR_INT_MASK

Address: 0x4E

Masks that enable and disable the interrupts

Table 171 • **IP 1588 Interrupt Mask Register**

Bit	Name	Description	Access	Default
6	EGR_ANALYZER_ERROR_MASK	Mask bit for ANALYZER_ERROR_STICKY bit. 0: Interrupt disabled 1: Interrupt enabled	R/W	0x0
5	EGR_RW_PREAMBLE_ERR_MASK	Mask for the RW_PREAMBLE_ERR_STICKY bit. 0: Interrupt disabled 1: Interrupt enabled	R/W	0x0
4	EGR_RW_FCS_ERR_MASK	Mask for the RW_FCS_ERR_STICKY bit. 0: Interrupt disabled 1: Interrupt enabled	R/W	0x0
3	EGR_TS_LEVEL_MASK	Mask bit for EGR_TS_LEVEL_STICKY. When 1, the interrupt is enabled. 0: Interrupt disabled 1: Interrupt enabled	R/W	0x0
2	EGR_TS_LOADED_MASK	Mask bit for TS_LOADED_STICKY. When 1, the interrupt is enabled. 0: Interrupt disabled 1: Interrupt enabled	R/W	0x0
1	EGR_TS_UNDERFLOW_MASK	Mask bit for TS_UNDERFLOW_STICKY. When 1, the interrupt is enabled. 0: Interrupt disabled 1: Interrupt enabled	R/W	0x0

Table 171 • **IP 1588 Interrupt Mask Register *(continued)***

Bit	Name	Description	Access	Default
0	EGR_TS_OVERFLOW_MASK	Mask bit for TS_OVERFLOW_STICKY. When 1, the interrupt is enabled. 0: Interrupt disabled 1: Interrupt enabled	R/W	0x0

4.19.3 Spare Scratchpad

Short Name: EGR_SPARE_REGISTER

Address: 0x4F

Table 172 • **Spare Scratchpad Register**

Bit	Name	Description	Access	Default
31:0	EGR_SPARE_REGISTER	Spare scratchpad register	R/W	0x00000000

4.20 1588 IP Egress Timestamp Processor Registers

This section provides information about the 1588 IP timestamp processor registers.

4.20.1 TSP Control

Short Name: EGR_TSP_CTRL

Address: 0x55

Table 173 • **TSP Control Register**

Bit	Name	Description	Access	Default
2	EGR_FRACT_NS_MODE	Selects a mode in which the fractional portion of a second (in units of nanoseconds) is used for timestamping. Only the operation of the WRITE_NS, WRITE_NS_P2P, and SUB_ADD PTP commands are affected by the setting of this mode bit. 0: Select the total (summed) nanoseconds for timestamping. 1: Select the fractional portion in nanoseconds for timestamping.	R/W	0x0
1	EGR_SEL_EXT_SOF_IND	Select external pin start of frame indicator. 0: Select internal PCS as the source of SOF 1: Select external pin as the source of SOF.	R/W	0x0
0	EGR_LOAD_DELAYS	One-shot loads Local latency, Path delay, and DelayAsymmetry values into the Timestamp Processor	One-shot	0x0

4.20.2 TSP Status

Short Name: EGR_TSP_STAT

Address: 0x56

Table 174 • **TSP Status Register**

Bit	Name	Description	Access	Default
0	EGR_CF_TOO_BIG_STICKY	Timestamp processor marked a calculated correction field as too big. 0: A calculated correction field that was too big did occur. 1: A calculated correction field that was too big did not occur.	Sticky	0x0

4.20.3 Local Latency

Short Name: EGR_LOCAL_LATENCY

Address: 0x57

Table 175 • **Local Latency Register**

Bit	Name	Description	Access	Default
15:0	EGR_LOCAL_LATENCY	Local latency (nanoseconds) The value programmed in this register is dependent upon the frequency of the clock driving the Local Time Counter (LTC) and upon LAN mode of operation. When in LAN mode and the LTC clock frequency is 250 MHz, set this register to 206. When in LAN mode and the LTC clock frequency is 125 MHz, set this register to 200.	R/W	0x0000

4.20.4 Path Delay

Short Name: EGR_PATH_DELAY

Address: 0x58

Table 176 • **Path Delay Register**

Bit	Name	Description	Access	Default
31:0	EGR_PATH_DELAY	Path delay (nanoseconds)	R/W	0x00000000

4.20.5 DelayAsymmetry

Short Name: EGR_DELAY_ASYMMETRY

Address: 0x59

Table 177 • **DelayAsymmetry Register**

Bit	Name	Description	Access	Default
31:0	EGR_DELAY_ASYMMETRY	DelayAsymmetry (scaled nanoseconds)	R/W	0x00000000

4.21 1588 IP Egress Delay FIFO Registers

This section provides information about the 1588 IP delay FIFO registers. The delay FIFO delays the data in a pipeline governed by these settings.

4.21.1 Configuration and Control for the Delay FIFO

Short Name: EGR_DF_CTRL

Address: 0x5A

Table 178 • **Configuration and Control Register for the Delay FIFO**

Bit	Name	Description	Access	Default
4:0	EGR_DF_DEPTH	The index of the register stage in the Delay FIFO that is used for output. The actual delay through the block is one more than the depth. If depth is set to 2, then the delay is 3 clocks as data is taken from stage 2. The depth MUST be greater than 0 (depth of 0 is not allowed). This bit group must be set to 0x0F in the device.	R/W	0x00

4.22 1588 IP Egress Timestamp FIFO Registers

This section provides information about the egress timestamp FIFO.

4.22.1 Timestamp FIFO Configuration and Status

Short Name: EGR_TSFIFO_CSR

Address: 0x5B

Configuration and status register for the Timestamp FIFO

Table 179 • **Timestamp FIFO Configuration and Status Register**

Bit	Name	Description	Access	Default
17	EGR_TS_4BYTES	Selects a smaller timestamp size to be stored in the Timestamp FIFO (4 bytes vs. the default 10 bytes). 0: full 10 byte timestamps are stored 1: Only 4 bytes of each timestamp are stored.	R/W	0x0
16	EGR_TS_FIFO_RESET	Forces the Timestamp_FIFO into the reset state.	R/W	0x0
15:12	EGR_TS_LEVEL	The FIFO level associated with the last read of the EGR_TS_EMPTY status field of the EGR_TSFIFO_0 register. Binary number (0-8)	R/O	0x0
11:8	EGR_TS_THRESH	The threshold at which the Timestamp FIFO interrupt EGR_TS_LEVEL_STICKY will be set. If the FIFO level reaches the threshold, the sticky bit EGR TS_LEVEL_STICKY will be set. Binary number (1-8)	R/W	0x3
4:0	EGR_TS_SIGNAT_BYTES	Indicates the number of signature bytes used for timestamps in the Timestamp FIFO (0-16).	R/W	0x00

4.22.2 Data Value from the Timestamp FIFO

Short Name: EGR_TSFIFO_0

Address: 0x5C

Read the data from the timestamp FIFO along with the FIFO empty flag in the MSB

Table 180 • **Data Value from the Timestamp FIFO Register**

Bit	Name	Description	Access	Default
31	EGR_TS_EMPTY	The FIFO empty flag from the Timestamp FIFO. If this bit is set, there is no FIFO data to be read from the FIFO. The data in the TSFIFO_x registers is not valid and should be discarded. When 0, the FIFO has data and the TSFIFO_x has a valid set of data. This register can be polled and when the bit is cleared, the other registers should be read to get a full timestamp. When 1, the last data has already been read out and the current read data should be discarded. Timestamp/Frame signature bytes are packed such that the 10 or 4 byte Timestamp resides in the LEAST significant bytes while the Frame signature (0 to 16 bytes) resides in the MOST significant bytes. The order of the bytes within each Timestamp/Frame signature field is also most significant to least significant. For example, a 26 byte timestamp/frame signature pairs are packed with the 10 byte timestamp field ([79:0]) corresponding to Bits 79:0 in the registers below, and a 16 byte frame signature field ([127:0]) corresponding to Bits 207:80 in the registers below. 0: FIFO not empty, data valid 1: FIFO empty, data invalid	R/O	0x1
30:28	EGR_TS_FLAGS	The FIFO flags from the Timestamp FIFO. These bits indicate how many timestamps are valid in the current (not empty) 26 byte FIFO entry. 000: Only the end of a partial timestamp is valid in the current FIFO entry (any remaining data is invalid) 001: 1 valid timestamp begins in the current FIFO entry (any remaining data is invalid) 010: 2 valid timestamps begin in the current FIFO entry (any remaining data is invalid) 011: 3 valid timestamps begin in the current FIFO entry (any remaining data is invalid) 100: 4 valid timestamps begin in the current FIFO entry (any remaining data is invalid) 101: 5 valid timestamps begin in the current FIFO entry (any remaining data is invalid) 110: 6 valid timestamps begin in the current FIFO entry (any remaining data is invalid) 111: The current FIFO entry is fully packed with timestamps (all data is valid)	R/O	N/A
15:0	EGR_TSFIFO_0	16 bits from the Timestamp FIFO. Bits 15:0.	R/O	N/A

4.22.3 Data Value from the Timestamp FIFO

Short Name: EGR_TSFIFO_1

Address: 0x5D

Read the data from the timestamp FIFO

Table 181 • **Data Value from the Timestamp FIFO Register**

Bit	Name	Description	Access	Default
31:0	EGR_TSFIFO_1	32 bits from the Timestamp FIFO. Bits 47:16.	R/O	N/A

4.22.4 Data Value from the Timestamp FIFO

Short Name: EGR_TSFIFO_2

Address: 0x5E

Read the data from the timestamp FIFO

Table 182 • **Data Value from the Timestamp FIFO Register**

Bit	Name	Description	Access	Default
31:0	EGR_TSFIFO_2	32 bits from the Timestamp FIFO. Bits 79:48.	R/O	N/A

4.22.5 Data Value from the Timestamp FIFO

Short Name: EGR_TSFIFO_3

Address: 0x5F

Read the data from the timestamp FIFO

Table 183 • **Data Value from the Timestamp FIFO Register**

Bit	Name	Description	Access	Default
31:0	EGR_TSFIFO_3	32 bits from the Timestamp FIFO. Bits 111:80.	R/O	N/A

4.22.6 Data Value from the Timestamp FIFO

Short Name: EGR_TSFIFO_4

Address: 0x60

Read the data from the timestamp FIFO

Table 184 • **Data Value from the Timestamp FIFO Register**

Bit	Name	Description	Access	Default
31:0	EGR_TSFIFO_4	32 bits from the Timestamp FIFO. Bits 143:112.	R/O	N/A

4.22.7 Data Value from the Timestamp FIFO

Short Name: EGR_TSFIFO_5

Address: 0x61

Read the data from the timestamp FIFO

Table 185 • **Data Value from the Timestamp FIFO Register**

Bit	Name	Description	Access	Default
31:0	EGR_TSFIFO_5	32 bits from the Timestamp FIFO. Bits 175:144.	R/O	N/A

4.22.8 Data Value from the Timestamp FIFO

Short Name: EGR_TSFIFO_6

Address: 0x62

Read the data from the timestamp FIFO

Table 186 • **Data Value from the Timestamp FIFO Register**

Bit	Name	Description	Access	Default
31:0	EGR_TSFIFO_6	32 bits from the Timestamp FIFO. Bits 207:176.	R/O	N/A

4.22.9 Count of Dropped Timestamps

Short Name: EGR_TSFIFO_DROP_CNT

Address: 0x63

Count of dropped Timestamps not enqueued to the TS FIFO

Table 187 • **Count of Dropped Timestamps Register**

Bit	Name	Description	Access	Default
31:0	EGR_TS_FIFO_DROP_CNT	Timestamps dropped count	R/W	0x00000000

4.23 1588 IP Egress Rewriter Registers

This section provides information about the 1588 IP rewriter configuration and status registers.

4.23.1 Rewriter Configuration and Control

Short Name: EGR_RW_CTRL

Address: 0x64

Configuration for the Rewriter

Table 188 • **Rewriter Configuration and Control Register**

Bit	Name	Description	Access	Default
4	EGR_RW_REDUCE_PREAMBLE	When set, the 1588 IP will reduce the preamble of ALL incoming frames by 4 bytes to allow a timestamp to be appended to the ingress data frames. This bit must be set along with proper configuration of the Analyzer to ensure proper operation. ** VALID IN INGRESS DIRECTION ONLY ** 0: No preamble modification 1: Reduce preamble by 4 bytes	R/W	0x0
3	EGR_RW_FLAG_VAL	Value to write to the flag bit when it is overwritten. 0: 0 will be written to the flag bit 1: 1 will be written to the flag bit	R/W	0x0
2:0	EGR_RW_FLAG_BIT	Bit offset within a byte of the flag bit which indicates if the frame has been modified or not. Binary number	R/W	0x0

4.23.2 Count of Modified Frames

Short Name: EGR_RW_MODFRM_CNT

Address: 0x65

Table 189 • **Count of Modified Frames Register**

Bit	Name	Description	Access	Default
31:0	EGR_RW_MODFRM_CNT	Count of the number of frames modified by the 1588 IP. The counter wraps. Binary number	R/W	0x00000000

4.23.3 Count of FCS Errors

Short Name: EGR_RW_FCS_ERR_CNT

Address: 0x66

Table 190 • **Count of FCS Errors Register**

Bit	Name	Description	Access	Default
31:0	EGR_RW_FCS_ERR_CNT	Count of the number of FCS errored frames detected by the Rewriter. Binary number	R/W	0x00000000

4.23.4 Count of the Number of Preamble Errors

Short Name: EGR_RW_PREAMBLE_ERR_CNT

Address: 0x67

Table 191 • **Count of the Number of Preamble Errors Register**

Bit	Name	Description	Access	Default
31:0	EGR_RW_PREAMBLE_ERR_CNT	Count of the number of errored preambles detected. The counter wraps. An errored preamble is a preamble that is too short to shrink that is encountered when RW_REDUCE_PREAMBLE is set. Binary number	R/W	0x00000000

4.24 1588 IP Ingress Debug Register

This section provides information about reading the 1588 IP internal registers.

4.24.1 Software Pop FIFO

Short Name: INGR_SW_POP_FIFO

Address: 0x9F

Table 192 • **INGR_SW_POP_FIFO**

Field Name	Bit	Access	Description	Default
INGR_SAFE_MODCHG_DIS	13	R/W	When set low, the mode change is a controlled process that first forces bypass mode, then takes effect after processing a single IDLE, then releases bypass mode. When asserted high, changes to PROTOCOL_MODE take immediate effect. 0= Follow controlled mode changes 1= Mode changes take immediate effect	0x0
Reserved	12	R/W		0x0

Table 192 • **INGR_SW_POP_FIFO *(continued)***

Field Name	Bit	Access	Description	Default
INGR_BYPASS_ON	11	R/W	Sticky bit. Indicates 1588 bypass is ON. Always clears by writing 1.	0x0[1]
INGR_BYPASS_IDLE	10	R/W	Sticky bit. Indicates 1588 has encountered at least one IDLE at input and in process of asserting bypass. Always clears by writing 1.	0x0
INGR_FIFO_LVL_OFF	9	R/W	Sticky bit. Indicates FIFO levels are non-zero in steady state and were automatically emptied. Always clears by writing 1.	0x0
INGR_AUTO_CLR_DONE	8	R/W	Sticky bit. If 1, indicates auto-clear operation was completed. Always clears by writing 1.	0x0
INGR_AUTO_CLR_CLKS	7:2	R/W	Only valid when AUTO_CLR_EN is high. After encountering AUTO_CLR_CLKS (‘n’ number of IDLEs), 1588 datapath FIFOs will empty. Only valid for number greater than or equal to 0x1A. 1G: needs ‘n+1’*8 clock cycles 100M: needs ‘n+1’*80 clock cycles 10M: needs ‘n+1’*800 clock cycles	0x1A
INGR_AUTO_CLR_EN	1	R/W	If enabled, 1588 will detect when FIFOs contain stale entries. If contains entries in steady-state, then after encountering AUTO_CLR_CLKS number of IDLEs, all 1588 FIFOs will automatically empty.	0x1
INGR_SW_POP_FIFO	0	R/W	Self-clearing bit to manually drain all FIFOs within 1588. Should only be asserted when 1588 is in the steady-state.	0x0

1. Bit may not be valid until a clear-by-write has been performed after a device power-on.

4.25 1588 IP Egress Debug Registers

This section provides information about reading the 1588 IP internal registers.

4.25.1 Software Pop FIFO

Short Name: EGR_SW_POP_FIFO

Address: 0xC0

Table 193 • **EGR_SW_POP_FIFO**

Field Name	Bit	Access	Description	Default
EGR_SAFE_MODCHG_DIS	13	R/W	When set low, the mode change is a controlled process that first forces bypass mode, then takes effect after processing a single IDLE, then releases bypass mode. When asserted high, changes to PROTOCOL_MODE take immediate effect. 0= Follow controlled mode changes 1= Mode changes take immediate effect	0x0
Reserved	12	R/W		0x0
EGR_BYPASS_ON	11	R/W	Sticky bit. Indicates 1588 bypass is ON. Always clears by writing 1.	0x0[1]

Table 193 • **EGR_SW_POP_FIFO *(continued)***

Field Name	Bit	Access	Description	Default
EGR_BYPASS_IDLE	10	R/W	Sticky bit. Indicates 1588 has encountered at least one IDLE at input and in process of asserting bypass. Always clears by writing 1.	0x0
EGR_FIFO_LVL_OFF	9	R/W	Sticky bit. Indicates FIFO levels are non-zero in steady state and were automatically emptied. Always clears by writing 1.	0x0
EGR_AUTO_CLR_DONE	8	R/W	Sticky bit. If 1, indicates auto-clear operation was completed. Always clears by writing 1.	0x0
EGR_AUTO_CLR_CLKS	7:2	R/W	Only valid when AUTO_CLR_EN is high. After encountering AUTO_CLR_CLKS ('n' number of IDLEs), 1588 datapath FIFOs will empty. Only valid for number greater than or equal to 0x1A. 1G: needs 'n+1'*8 clock cycles 100M: needs 'n+1'*80 clock cycles 10M: needs 'n+1'*800 clock cycles	0x1A
EGR_AUTO_CLR_EN	1	R/W	If enabled, 1588 will detect when FIFOs contain stale entries. If contains entries in steady-state, then after encountering AUTO_CLR_CLKS number of IDLEs, all 1588 FIFOs will automatically empty.	0x1
EGR_SW_POP_FIFO	0	R/W	Self-clearing bit to manually drain all FIFOs within 1588. Should only be asserted when 1588 is in the steady-state.	0x0

1. Bit may not be valid until a clear-by-write has been performed after a device power-on.

4.26 Ingress0 Analyzer Engine Configuration Registers

This section lists overviews for the ingress0 analyzer engine configuration registers.Ingress1 analyzer engine registers are identical to the ones defined for ingress0.

Note: The analyzer engine configuration registers are not initialized to the default values during chip reset. Software must configure these registers to their default value.

Note: For more information about accessing the 1588 IP registers, see Accessing 1588 IP Registers, page 72.

Table 194 • **INGR0_ETH1_NXT_PROTOCOL**

Address	Name	Details
0x00	INGR0_ETH1_NXT_PROTOCOL	Ethernet Next Protocol, page 171
0x01	INGR0_ETH1_VLAN_TPID_CFG	VLAN TPID Configuration, page 172
0x02	INGR0_ETH1_TAG_MODE	Ethernet Tag Mode, page 172
0x03	INGR0_ETH1_ETYPE_MATCH	Ethertype Match, page 172

Table 195 • **INGR0_ETH1_FLOW_CFG (8 instances)**

Address	Name	Details
0x10	INGR0_ETH1_FLOW_ENABLE	Ethernet Flow Enable, page 173
0x11	INGR0_ETH1_MATCH_MODE	Ethernet Protocol Match Mode, page 173
0x12	INGR0_ETH1_ADDR_MATCH_1	Ethernet Address Match Part 1, page 174

Table 195 • **INGR0_ETH1_FLOW_CFG (8 instances)** ***(continued)***

Address	Name	Details
0x13	INGR0_ETH1_ADDR_MATCH_2	Ethernet Address Match Part 2, page 175
0x14	INGR0_ETH1_VLAN_TAG_RANGE_I_TAG	Ethernet VLAN Tag Range Match, page 175
0x15	INGR0_ETH1_VLAN_TAG1	VLAN Tag 1 Match/Mask, page 176
0x16	INGR0_ETH1_VLAN_TAG2_I_TAG	Match/Mask For VLAN Tag 2 or I-Tag Match, page 176

Table 196 • **INGR0_ETH2_NXT_PROTOCOL**

Address	Name	Details
0x90	INGR0_ETH2_NXT_PROTOCOL	Ethernet Next Protocol, page 177
0x91	INGR0_ETH2_VLAN_TPID_CFG	VLAN TPID Configuration, page 177
0x92	INGR0_ETH2_ETYPE_MATCH	Ethertype Match, page 177

Table 197 • **INGR0_ETH2_FLOW_CFG (8 instances)**

Address	Name	Details
0xA0	INGR0_ETH2_FLOW_ENABLE	Ethernet Flow Enable, page 178
0xA1	INGR0_ETH2_MATCH_MODE	Ethernet Protocol Match Mode, page 178
0xA2	INGR0_ETH2_ADDR_MATCH_1	Ethernet Address Match Part 1, page 179
0xA3	INGR0_ETH2_ADDR_MATCH_2	Ethernet Address Match Part 2, page 180
0xA4	INGR0_ETH2_VLAN_TAG_RANGE_I_TAG	Ethernet VLAN Tag Range Match, page 180
0xA5	INGR0_ETH2_VLAN_TAG1	VLAN Tag 1 Match/Mask, page 181
0xA6	INGR0_ETH2_VLAN_TAG2_I_TAG	Match/Mask for VLAN Tag 2 or I-Tag Match, page 181

Table 198 • **INGR0_MPLS_NXT_COMPARATOR**

Address	Name	Details
0x120	INGR0_MPLS_NXT_COMPARATOR	MPLS Next Protocol Comparator, page 182

Table 199 • **INGR0_MPLS_FLOW_CFG (8 instances)**

Address	Name	Details
0x130	INGR0_MPLS_FLOW_CONTROL	MPLS Flow Control, page 182
0x132	INGR0_MPLS_LABEL_RANGE_LOWER_0	MPLS Label 0 Match Range Lower Value, page 183

Table 199 • **INGR0_MPLS_FLOW_CFG (8 instances)** ***(continued)***

Address	Name	Details
0x133	INGR0_MPLS_LABEL_RANGE_UPPER_0	MPLS Label 0 Match Range Upper Value, page 183
0x134	INGR0_MPLS_LABEL_RANGE_LOWER_1	MPLS Label 1 Match Range Lower Value, page 184
0x135	INGR0_MPLS_LABEL_RANGE_UPPER_1	MPLS Label 1 Match Range Lower Value, page 184
0x136	INGR0_MPLS_LABEL_RANGE_LOWER_2	MPLS Label 2 Match Range Lower Value, page 185
0x137	INGR0_MPLS_LABEL_RANGE_UPPER_2	MPLS Label 2 Match Range Lower Value, page 185
0x138	INGR0_MPLS_LABEL_RANGE_LOWER_3	MPLS Label 3 Match Range Lower Value, page 185
0x139	INGR0_MPLS_LABEL_RANGE_UPPER_3	MPLS Label 3 Match Range Lower Value, page 186

Table 200 • **INGR0_IP1_NXT_PROTOCOL**

Address	Name	Details
0x1B0	INGR0_IP1_NXT_COMPARATOR	IP Next Comparator Control, page 186
0x1B1	INGR0_IP1_MODE	IP Comparator Mode, page 187
0x1B2	INGR0_IP1_PROT_MATCH_1	IP Match Register Set 1, page 187
0x1B3	INGR0_IP1_PROT_MATCH_2_UPPER	Upper Portion of Match 2, page 187
0x1B4	INGR0_IP1_PROT_MATCH_2_LOWER	Lower Portion of Match 2, page 188
0x1B5	INGR0_IP1_PROT_MASK_2_UPPER	Upper Portion of Match Mask 2, page 188
0x1B6	INGR0_IP1_PROT_MASK_2_LOWER	Lower Portion of Match Mask 2, page 188
0x1B7	INGR0_IP1_PROT_OFFSET_2	Match Offset 2, page 188
0x1B8	INGR0_IP1_UDP_CHKSUM_CFG	IP/UDP Checksum Control, page 188

Table 201 • **INGR0_IP1_FLOW_CFG (8 instances)**

Address	Name	Details
0x1C0	INGR0_IP1_FLOW_ENA	IP Flow Enable, page 189
0x1C1	INGR0_IP1_FLOW_MATCH_UPPER	Upper Portion of the IP Flow Match, page 190
0x1C2	INGR0_IP1_FLOW_MATCH_UPPER_MID	Upper Mid Portion of the IP Flow Match, page 190
0x1C3	INGR0_IP1_FLOW_MATCH_LOWER_MID	Lower Mid Portion of the IP Flow Match, page 191
0x1C4	INGR0_IP1_FLOW_MATCH_LOWER	Lower Portion of the IP Flow Match, page 191

Table 201 • **INGR0_IP1_FLOW_CFG (8 instances)** ***(continued)***

Address	Name	Details
0x1C5	INGR0_IP1_FLOW_MASK_UPPER	IP Flow Match Mask, page 191
0x1C6	INGR0_IP1_FLOW_MASK_UPPER_MID	Upper Mid Portion of the IP Flow Mask, page 192
0x1C7	INGR0_IP1_FLOW_MASK_LOWER_MID	Lower Mid Portion of the IP Flow Mask, page 192
0x1C8	INGR0_IP1_FLOW_MASK_LOWER	Lower Portion of the IP Flow Mask, page 193

Table 202 • **INGR0_IP2_NXT_PROTOCOL**

Address	Name	Details
0x240	INGR0_IP2_NXT_COMPARATOR	IP Next Comparator Control, page 193
0x241	INGR0_IP2_MODE	IP Comparator Mode, page 193
0x242	INGR0_IP2_PROT_MATCH_1	IP Match Set 1, page 194
0x243	INGR0_IP2_PROT_MATCH_2_UPPER	Upper Portion of Match 2, page 194
0x244	INGR0_IP2_PROT_MATCH_2_LOWER	Lower Portion of Match 2, page 194
0x245	INGR0_IP2_PROT_MASK_2_UPPER	Upper Portion of Match Mask 2, page 194
0x246	INGR0_IP2_PROT_MASK_2_LOWER	Lower Portion of Match Mask 2, page 195
0x247	INGR0_IP2_PROT_OFFSET_2	Match Offset 2, page 195
0x248	INGR0_IP2_UDP_CHKSUM_CFG	IP/UDP Checksum Control, page 195

Table 203 • **INGR0_IP2_FLOW_CFG (8 instances)**

Address	Name	Details
0x250	INGR0_IP2_FLOW_ENA	IP Flow Enable, page 196
0x251	INGR0_IP2_FLOW_MATCH_UPPER	Upper Portion of the IP Flow Match, page 196
0x252	INGR0_IP2_FLOW_MATCH_UPPER_MID	Upper Mid Portion of the IP Flow Match, page 197
0x253	INGR0_IP2_FLOW_MATCH_LOWER_MID	Lower Mid Portion of the IP Flow Match, page 197
0x254	INGR0_IP2_FLOW_MATCH_LOWER	Lower Portion of the IP Flow Match, page 198
0x255	INGR0_IP2_FLOW_MASK_UPPER	IP Flow Match Mask, page 198
0x256	INGR0_IP2_FLOW_MASK_UPPER_MID	Upper Mid Portion of the IP Flow Mask, page 198
0x257	INGR0_IP2_FLOW_MASK_LOWER_MID	Lower Mid Portion of the IP Flow Mask, page 199

Table 203 • **INGR0_IP2_FLOW_CFG (8 instances) *(continued)***

Address	Name	Details
0x258	INGR0_IP2_FLOW_MASK_LOWER	Lower Portion of the IP Flow Mask, page 199

Table 204 • **INGR0_PTP_FLOW (6 instances)**

Address	Name	Details
0x2D0	INGR0_PTP_FLOW_ENA	PTP/OAM Flow Enable, page 200
0x2D1	INGR0_PTP_FLOW_MATCH_UPPER	Upper Half of PTP/OAM Flow Match Field, page 200
0x2D2	INGR0_PTP_FLOW_MATCH_LOWER	Lower Half of PTP/OAM Flow Match Field, page 200
0x2D3	INGR0_PTP_FLOW_MASK_UPPER	Upper Half of PTP/OAM Flow Match Mask, page 201
0x2D4	INGR0_PTP_FLOW_MASK_LOWER	Lower Half of PTP/OAM Flow Match Mask, page 201
0x2D5	INGR0_PTP_DOMAIN_RANGE	PTP/OAM Range Match, page 201
0x2D6	INGR0_PTP_ACTION	PTP Action Control, page 202
0x2D7	INGR0_PTP_ACTION_2	PTP Action Control 2, page 203
0x2D8	INGR0_PTP_ZERO_FIELD_CTL	Zero Field Control, page 203

Table 205 • **INGR0_PTP_IP_CHKSUM_CTL**

Address	Name	Details
0x330	INGR0_PTP_IP_CKSUM_SEL	IP Checksum Block Select, page 204

4.27 Ingress0 Ethernet Next Protocol Configuration Registers

This section provides information about the Ethernet next protocol configuration registers.

4.27.1 Ethernet Next Protocol

Short Name: INGR0_ETH1_NXT_PROTOCOL

Address: 0x00

Table 206 • **Ethernet Next Protocol Register**

Bit	Name	Description	Access	Default
20:16	INGR0_ETH1_FRAME_SIG_OFFSET	Frame signature offset. Points to the start of the byte field in the Ethernet frame that will be used for the frame signature	R/W	0x00

Table 206 • **Ethernet Next Protocol Register *(continued)***

Bit	Name	Description	Access	Default
2:0	INGR0_ETH1_NXT_COMPARATOR	Points to the next comparator block after this Ethernet block 0: Reserved 1: Ethernet comparator 2 2: IP/UDP/ACH comparator 1 3: IP/UDP/ACH comparator 2 4: MPLS comparator 5: PTP/OAM comparator 6,7: Reserved	R/W	0x0

4.27.2 VLAN TPID Configuration

Short Name: INGR0_ETH1_VLAN_TPID_CFG

Address: 0x01

Table 207 • **VLAN TPID Configuration Register**

Bit	Name	Description	Access	Default
31:16	INGR0_ETH1_VLAN_TPID_CFG	Configurable VLAN TPID (S or B-tag)	R/W	0x88A8

4.27.3 Ethernet Tag Mode

Short Name: INGR0_ETH1_TAG_MODE

Address: 0x02

Table 208 • **Ethernet Tag Mode Register**

Bit	Name	Description	Access	Default
0	INGR0_ETH1_PBB_ENA	This bit enables the presence of PBB. The I-tag match bits are programmed in the ETH1_VLAN_TAG_RANGE registers. The mask bits are programmed in the ETH1_VLAN_TAG2 registers. A B-tag if present is configured in the ETH1_VLAN_TAG1 registers. 0: PBB not enabled 1: Always expect PBB, last tag is always an I-tag	R/W	0x0

4.27.4 Ethertype Match

Short Name: INGR0_ETH1_ETYPE_MATCH

Address: 0x03

Table 209 • **Ethertype Match Register**

Bit	Name	Description	Access	Default
15:0	INGR0_ETH1_ETYPE_MATCH	If the Ethertype/length field is an Ethertype, then this register is compared against the value. If the field is a length, the length value is not checked.	R/W	0x0000

4.27.5

Instance offsets: 0x10 INGR0_ETH1_FLOW_CFG_0

0x20 INGR0_ETH1_FLOW_CFG_1

0x30 INGR0_ETH1_FLOW_CFG_2

0x40 INGR0_ETH1_FLOW_CFG_3

0x50 INGR0_ETH1_FLOW_CFG_4

0x60 INGR0_ETH1_FLOW_CFG_5

0x70 INGR0_ETH1_FLOW_CFG_6

0x80 INGR0_ETH1_FLOW_CFG_7

4.27.6 Ethernet Flow Enable

Short Name: INGR0_ETH1_FLOW_ENABLE

Addresses: 0x10 INGR0_ETH1_FLOW_CFG_0

0x20 INGR0_ETH1_FLOW_CFG_1

0x30 INGR0_ETH1_FLOW_CFG_2

0x40 INGR0_ETH1_FLOW_CFG_3

0x50 INGR0_ETH1_FLOW_CFG_4

0x60 INGR0_ETH1_FLOW_CFG_5

0x70 INGR0_ETH1_FLOW_CFG_6

0x80 INGR0_ETH1_FLOW_CFG_7

Table 210 • **Ethernet Flow Enable Register**

Bit	Name	Description	Access	Default
9:8	INGR0_ETH1_CHANNEL_MASK	Channel mask bit 0: Flow valid for channel 0 bit 1: Flow valid for channel 1	R/W	0x3
0	INGR0_ETH1_FLOW_ENABLE	Flow enable 0: Flow is disabled 1: Flow is enabled	R/W	0x0

4.27.7 Ethernet Protocol Match Mode

Short Name: INGR0_ETH1_MATCH_MODE

Addresses: 0x11 INGR0_ETH1_FLOW_CFG_0

0x21 INGR0_ETH1_FLOW_CFG_1

0x31 INGR0_ETH1_FLOW_CFG_2

0x41 INGR0_ETH1_FLOW_CFG_3

0x51 INGR0_ETH1_FLOW_CFG_4

0x61 INGR0_ETH1_FLOW_CFG_5

0x71 INGR0_ETH1_FLOW_CFG_6

0x81 INGR0_ETH1_FLOW_CFG_7

Table 211 • **Ethernet Protocol Match Mode Register**

Bit	Name	Description	Access	Default
13:12	INGR0_ETH1_VLAN_TAG_MODE	0: VLAN range checking disabled 1: VLAN range checking on tag 1 2: VLAN range checking on tag 2 (not supported with PBB) 3: reserved	R/W	0x0
9	INGR0_ETH1_VLAN_TAG2_TYPE	This register is only used if ETH1_VLAN_VERIFY_ENA = 1 If PBB not enabled: 0: C tag (TPID of 0x8100) 1: S tag (match to CONF_VLAN_TPID) If PBB enabled: 0,1: I tag (use range registers)	R/W	0x1
8	INGR0_ETH1_VLAN_TAG1_TYPE	This register is only used if ETH1_VLAN_VERIFY_ENA = 1 0: C tag (TPID of 0x8100) 1: S or B tag (match to CONF_VLAN_TPID)	R/W	0x0
7:6	INGR0_ETH1_VLAN_TAGS	This register is only used if ETH1_VLAN_VERIFY_ENA = 1 0: No VLAN tags (not valid for PBB) 1: 1 VLAN tag (for PBB this would be the I-tag) 2: 2 VLAN tags (for PBB expect a B-tag and an I-tag) 3: Reserved	R/W	0x0
4	INGR0_ETH1_VLAN_VERIFY_ENA	0: Parse for VLAN tags, do not check values. For PBB the I-tag is always checked. 1: Verify configured VLAN tag configuration.	R/W	0x0
0	INGR0_ETH1_ETHERTYPE_MODE	When checking for presence of SNAP/LLC based upon ETH1_MATCH_MODE, this field indicates if SNAP & 3-byte LLC is expected to be present 0: Only Ethernet type II supported, no SNAP/LLC 1: Ethernet type II & Ethernet type I with SNAP/LLC, determine if SNAP/LLC is present or not. Type I always assumes that SNAP/LLC is present	R/W	0x0

4.27.8 Ethernet Address Match Part 1

Short Name: INGR0_ETH1_ADDR_MATCH_1

Addresses: 0x12 INGR0_ETH1_FLOW_CFG_0

0x22 INGR0_ETH1_FLOW_CFG_1

0x32 INGR0_ETH1_FLOW_CFG_2

0x42 INGR0_ETH1_FLOW_CFG_3

0x52 INGR0_ETH1_FLOW_CFG_4

0x62 INGR0_ETH1_FLOW_CFG_5

0x72 INGR0_ETH1_FLOW_CFG_6

0x82 INGR0_ETH1_FLOW_CFG_7

Table 212 • **Ethernet Address Match Part 1 Register**

Bit	Name	Description	Access	Default
31:0	INGR0_ETH1_ADDR_MATCH_1	First 32 bits of the address match value	R/W	0x00000000

4.27.9 Ethernet Address Match Part 2

Short Name: INGR0_ETH1_ADDR_MATCH_2

Addresses: 0x13 INGR0_ETH1_FLOW_CFG_0

0x23 INGR0_ETH1_FLOW_CFG_1

0x33 INGR0_ETH1_FLOW_CFG_2

0x43 INGR0_ETH1_FLOW_CFG_3

0x53 INGR0_ETH1_FLOW_CFG_4

0x63 INGR0_ETH1_FLOW_CFG_5

0x73 INGR0_ETH1_FLOW_CFG_6

0x83 INGR0_ETH1_FLOW_CFG_7

Table 213 • **Ethernet Address Match Part 2 Register**

Bit	Name	Description	Access	Default
22:20	INGR0_ETH1_ADDR_MATCH_MODE	Selects how the addresses are matched. Multiple bits can be set at once bit 0: Full 48-bit address match bit 1: Match any unicast address bit 2: Match any multicast address	R/W	0x1
17:16	INGR0_ETH1_ADDR_MATCH_SELECT	Selects which address to match 0: Match the Destination Address 1: Match the Source Address 2: Match either the Source of Destination Address 3: Reserved	R/W	0x0
15:0	INGR0_ETH1_ADDR_MATCH_2	Last 16 bits of the Ethernet address match field	R/W	0x0000

4.27.10 Ethernet VLAN Tag Range Match

Short Name: INGR0_ETH1_VLAN_TAG_RANGE_I_TAG

Addresses: 0x14 INGR0_ETH1_FLOW_CFG_0

0x24 INGR0_ETH1_FLOW_CFG_1

0x34 INGR0_ETH1_FLOW_CFG_2

0x44 INGR0_ETH1_FLOW_CFG_3

0x54 INGR0_ETH1_FLOW_CFG_4

0x64 INGR0_ETH1_FLOW_CFG_5

0x74 INGR0_ETH1_FLOW_CFG_6

0x84 INGR0_ETH1_FLOW_CFG_7

Table 214 • **Ethernet VLAN Tag Range Match Register**

Bit	Name	Description	Access	Default
27:16	INGR0_ETH1_VLAN_TAG_RANGE_UPPER	If PBB mode is not enabled, then this register contains the upper range of the VLAN tag range match. If PBB mode is enabled, then this register contains the upper 12 bits of the I-tag	R/W	0xFFF
11:0	INGR0_ETH1_VLAN_TAG_RANGE_LOWER	If PBB mode is not enabled, then this register contains the lower range of the VLAN tag range match. If PBB mode is enabled, then this register contains the lower 12 bits of the I-tag	R/W	0x000

4.27.11 VLAN Tag 1 Match/Mask

Short Name: INGR0_ETH1_VLAN_TAG1

Addresses: 0x15 INGR0_ETH1_FLOW_CFG_0

0x25 INGR0_ETH1_FLOW_CFG_1

0x35 INGR0_ETH1_FLOW_CFG_2

0x45 INGR0_ETH1_FLOW_CFG_3

0x55 INGR0_ETH1_FLOW_CFG_4

0x65 INGR0_ETH1_FLOW_CFG_5

0x75 INGR0_ETH1_FLOW_CFG_6

0x85 INGR0_ETH1_FLOW_CFG_7

Table 215 • **VLAN Tag 1 Match/Mask Register**

Bit	Name	Description	Access	Default
27:16	INGR0_ETH1_VLAN_TAG1_MASK	Mask value for VLAN tag 1	R/W	0xFFF
11:0	INGR0_ETH1_VLAN_TAG1_MATCH	Match value for the first VLAN tag	R/W	0x000

4.27.12 Match/Mask For VLAN Tag 2 or I-Tag Match

Short Name: INGR0_ETH1_VLAN_TAG2_I_TAG

Addresses: 0x16 INGR0_ETH1_FLOW_CFG_0

0x26 INGR0_ETH1_FLOW_CFG_1

0x36 INGR0_ETH1_FLOW_CFG_2

0x46 INGR0_ETH1_FLOW_CFG_3

0x56 INGR0_ETH1_FLOW_CFG_4

0x66 INGR0_ETH1_FLOW_CFG_5

0x76 INGR0_ETH1_FLOW_CFG_6

0x86 INGR0_ETH1_FLOW_CFG_7

Table 216 • **Match/Mask For VLAN Tag 2 or I-Tag Match Register**

Bit	Name	Description	Access	Default
27:16	INGR0_ETH1_VLAN_TAG2_MASK	When PBB is not enabled, the mask field for VLAN tag 2 When PBB is enabled, the upper 12 bits of the I-tag mask	R/W	0xFFF
11:0	INGR0_ETH1_VLAN_TAG2_MATCH	When PBB is not enabled, the match field for VLAN Tag 2 When PBB is enabled, the lower 12 bits of the I-tag mask field	R/W	0x000

4.27.13 Ethernet Next Protocol

Short Name: INGR0_ETH2_NXT_PROTOCOL

Address: 0x90

Table 217 • **Ethernet Next Protocol Register**

Bit	Name	Description	Access	Default
20:16	INGR0_ETH2_FRAME_SIG_OFFSET	Frame signature offset. Points to the start of the byte field in the Ethernet frame that will be used for the frame signature	R/W	0x00
2:0	INGR0_ETH2_NXT_COMPARATOR	Points to the next comparator block after this Ethernet block. If this comparator block is not used, this field must be set to 0. 0: Comparator block not used 1: Ethernet comparator 2 2: IP/UDP/ACH comparator 1 3: IP/UDP/ACH comparator 2 4: MPLS comparator 5: PTP/OAM comparator 6,7: Reserved	R/W	0x0

4.27.14 VLAN TPID Configuration

Short Name: INGR0_ETH2_VLAN_TPID_CFG

Address: 0x91

Table 218 • **VLAN TPID Configuration Register**

Bit	Name	Description	Access	Default
31:16	INGR0_ETH2_VLAN_TPID_CFG	Configurable S-tag TPID	R/W	0x88A8

4.27.15 Ethertype Match

Short Name: INGR0_ETH2_ETYPE_MATCH

Address: 0x92

Table 219 • **Ethertype Match Register**

Bit	Name	Description	Access	Default
15:0	INGR0_ETH2_ETYPE_MATCH	If the Ethertype/length field is an Ethertype, then this register is compared against the value. If the field is a length, the length value is not checked.	R/W	0x0000

4.27.16

Instance offsets: 0xA0 INGR0_ETH2_FLOW_CFG_0

0xB0 INGR0_ETH2_FLOW_CFG_1

0xC0 INGR0_ETH2_FLOW_CFG_2

0xD0 INGR0_ETH2_FLOW_CFG_3

0xE0 INGR0_ETH2_FLOW_CFG_4

0xF0 INGR0_ETH2_FLOW_CFG_5

0x100 INGR0_ETH2_FLOW_CFG_6

0x110 INGR0_ETH2_FLOW_CFG_7

4.27.17 Ethernet Flow Enable

Short Name: INGR0_ETH2_FLOW_ENABLE

Addresses: 0xA0 INGR0_ETH2_FLOW_CFG_0

0xB0 INGR0_ETH2_FLOW_CFG_1

0xC0 INGR0_ETH2_FLOW_CFG_2

0xD0 INGR0_ETH2_FLOW_CFG_3

0xE0 INGR0_ETH2_FLOW_CFG_4

0xF0 INGR0_ETH2_FLOW_CFG_5

0x100 INGR0_ETH2_FLOW_CFG_6

0x110 INGR0_ETH2_FLOW_CFG_7

Table 220 • **Ethernet Flow Enable Register**

Bit	Name	Description	Access	Default
9:8	INGR0_ETH2_CHANNEL_MASK	bit 0: Flow valid for channel 0 bit 1: Flow valid for channel 1	R/W	0x3
0	INGR0_ETH2_FLOW_ENABLE	Flow enable. If this comparator block is not used, all flow enable bits must be set to 0. 0: Flow is disabled 1: Flow is enabled	R/W	0x0

4.27.18 Ethernet Protocol Match Mode

Short Name: INGR0_ETH2_MATCH_MODE

Addresses: 0xA1 INGR0_ETH2_FLOW_CFG_0

0xB1 INGR0_ETH2_FLOW_CFG_1

0xC1 INGR0_ETH2_FLOW_CFG_2

0xD1 INGR0_ETH2_FLOW_CFG_3

0xE1 INGR0_ETH2_FLOW_CFG_4

0xF1 INGR0_ETH2_FLOW_CFG_5

0x101 INGR0_ETH2_FLOW_CFG_6

0x111 INGR0_ETH2_FLOW_CFG_7

Table 221 • **Ethernet Protocol Match Mode Register**

Bit	Name	Description	Access	Default
13:12	INGR0_ETH2_VLAN_TAG_MODE	0: VLAN range checking disabled 1: VLAN range checking on tag 1 2: VLAN range checking on tag 2 (not supported with PBB) 3: reserved	R/W	0x0
9	INGR0_ETH2_VLAN_TAG2_TYPE	This register is only used if ETH1_VLAN_VERIFY_ENA = 1 0: C tag (TPID of 0x8100) 1: S tag (match to CONF_VLAN_TPID)	R/W	0x1
8	INGR0_ETH2_VLAN_TAG1_TYPE	This register is only used if ETH1_VLAN_VERIFY_ENA = 1 0: C tag (TPID of 0x8100) 1: S or B tag (match to CONF_VLAN_TPID)	R/W	0x0
7:6	INGR0_ETH2_VLAN_TAGS	This register is only used if ETH2_VLAN_VERIFY_ENA = 1 0: No VLAN tags 1: 1 VLAN tag 2: 2 VLAN tags 3: Reserved	R/W	0x0
4	INGR0_ETH2_VLAN_VERIFY_ENA	0: Parse for VLAN tags, do not check values. 1: Verify configured VLAN tag configuration.	R/W	0x0
0	INGR0_ETH2_ETHERTYPE_MODE	When checking for presence of SNAP/LLC based upon ETH1_MATCH_MODE, this field indicates if SNAP & 3-byte LLC is expected to be present 0: Only Ethernet type II supported, no SNAP/LLC 1: Ethernet type II & Ethernet type I with SNAP/LLC, determine if SNAP/LLC is present or not. Type I always assumes that SNAP/LLC is present	R/W	0x0

4.27.19 Ethernet Address Match Part 1

Short Name: INGR0_ETH2_ADDR_MATCH_1

Addresses: 0xA2 INGR0_ETH2_FLOW_CFG_0

0xB2 INGR0_ETH2_FLOW_CFG_1

0xC2 INGR0_ETH2_FLOW_CFG_2

0xD2 INGR0_ETH2_FLOW_CFG_3

0xE2 INGR0_ETH2_FLOW_CFG_4

0xF2 INGR0_ETH2_FLOW_CFG_5

0x102 INGR0_ETH2_FLOW_CFG_6

0x112 INGR0_ETH2_FLOW_CFG_7

Table 222 • **Ethernet Address Match Part 1 Register**

Bit	Name	Description	Access	Default
31:0	INGR0_ETH2_ADDR_MATCH_1	First 32 bits of the address match value	R/W	0x00000000

4.27.20 Ethernet Address Match Part 2

Short Name: INGR0_ETH2_ADDR_MATCH_2

Addresses: 0xA3 INGR0_ETH2_FLOW_CFG_0

0xB3 INGR0_ETH2_FLOW_CFG_1

0xC3 INGR0_ETH2_FLOW_CFG_2

0xD3 INGR0_ETH2_FLOW_CFG_3

0xE3 INGR0_ETH2_FLOW_CFG_4

0xF3 INGR0_ETH2_FLOW_CFG_5

0x103 INGR0_ETH2_FLOW_CFG_6

0x113 INGR0_ETH2_FLOW_CFG_7

Table 223 • **Ethernet Address Match Part 2 Register**

Bit	Name	Description	Access	Default
22:20	INGR0_ETH2_ADDR_MATCH_MODE	Selects how the addresses are matched. Multiple bits can be set at once bit 0: Full 48-bit address match bit 1: Match any unicast address bit 2: Match any multicast address	R/W	0x1
17:16	INGR0_ETH2_ADDR_MATCH_SELECT	Selects which address to match 0: Match the Destination Address 1: Match the Source Address 2: Match either the Source of Destination Address 3: Reserved	R/W	0x0
15:0	INGR0_ETH2_ADDR_MATCH_2	Last 16 bits of the Ethernet address match field	R/W	0x0000

4.27.21 Ethernet VLAN Tag Range Match

Short Name: INGR0_ETH2_VLAN_TAG_RANGE_I_TAG

Addresses: 0xA4 INGR0_ETH2_FLOW_CFG_0

0xB4 INGR0_ETH2_FLOW_CFG_1

0xC4 INGR0_ETH2_FLOW_CFG_2

0xD4 INGR0_ETH2_FLOW_CFG_3

0xE4 INGR0_ETH2_FLOW_CFG_4

0xF4 INGR0_ETH2_FLOW_CFG_5

0x104 INGR0_ETH2_FLOW_CFG_6

0x114 INGR0_ETH2_FLOW_CFG_7

Table 224 • **Ethernet VLAN Tag Range Match Register**

Bit	Name	Description	Access	Default
27:16	INGR0_ETH2_VLAN_TAG_RANGE_UPPER	This register contains the upper range of the VLAN tag range match.	R/W	0xFFF
11:0	INGR0_ETH2_VLAN_TAG_RANGE_LOWER	This register contains the lower range of the VLAN tag range match.	R/W	0x000

4.27.22 VLAN Tag 1 Match/Mask

Short Name: INGR0_ETH2_VLAN_TAG1

Addresses: 0xA5 INGR0_ETH2_FLOW_CFG_0

0xB5 INGR0_ETH2_FLOW_CFG_1

0xC5 INGR0_ETH2_FLOW_CFG_2

0xD5 INGR0_ETH2_FLOW_CFG_3

0xE5 INGR0_ETH2_FLOW_CFG_4

0xF5 INGR0_ETH2_FLOW_CFG_5

0x105 INGR0_ETH2_FLOW_CFG_6

0x115 INGR0_ETH2_FLOW_CFG_7

Table 225 • **VLAN Tag 1 Match/Mask Register**

Bit	Name	Description	Access	Default
27:16	INGR0_ETH2_VLAN_TAG1_MASK	Mask value for VLAN tag 1	R/W	0xFFF
11:0	INGR0_ETH2_VLAN_TAG1_MATCH	Match value for the first VLAN tag	R/W	0x000

4.27.23 Match/Mask for VLAN Tag 2 or I-Tag Match

Short Name: INGR0_ETH2_VLAN_TAG2_I_TAG

Addresses: 0xA6 INGR0_ETH2_FLOW_CFG_0

0xB6 INGR0_ETH2_FLOW_CFG_1

0xC6 INGR0_ETH2_FLOW_CFG_2

0xD6 INGR0_ETH2_FLOW_CFG_3

0xE6 INGR0_ETH2_FLOW_CFG_4

0xF6 INGR0_ETH2_FLOW_CFG_5

0x106 INGR0_ETH2_FLOW_CFG_6

0x116 INGR0_ETH2_FLOW_CFG_7

Table 226 • **Match/Mask for VLAN Tag 2 or I-Tag Match Register**

Bit	Name	Description	Access	Default
27:16	INGR0_ETH2_VLAN_TAG2_MASK	Mask field for VLAN tag 2	R/W	0xFFF

Table 226 • **Match/Mask for VLAN Tag 2 or I-Tag Match Register *(continued)***

Bit	Name	Description	Access	Default
11:0	INGR0_ETH2_VLAN_TAG2_MATCH	Match field for VLAN Tag 2	R/W	0x000

4.28 Ingress0 MPLS Next Protocol Registers

This section provides information about the MPLS next protocol registers.

4.28.1 MPLS Next Protocol Comparator

Short Name: INGR0_MPLS_NXT_COMPARATOR

Address: 0x120

Table 227 • **MPLS Next Protocol Comparator Register**

Bit	Name	Description	Access	Default
16	INGR0_MPLS_CTL_WORD	Indicates the presence of a control word after the last label. The first 4 bits of the control word are always 0. 0: There is no control word after the last label 1: There is a control word after the last label	R/W	0x0
2:0	INGR0_MPLS_NXT_COMPARATOR	Points to the next comparator stage. If this comparator block is not used, this field must be set to 0. 0: Comparator block not used. 1: Ethernet comparator 2 2: IP/UDP/ACH comparator 1 3: IP/UDP/ACH comparator 2 4: Reserved 5: PTP/OAM comparator 6,7: Reserved	R/W	0x0

4.28.2

Instance offsets: 0x130 INGR0_MPLS_FLOW_CFG_0

0x140 INGR0_MPLS_FLOW_CFG_1

0x150 INGR0_MPLS_FLOW_CFG_2

0x160 INGR0_MPLS_FLOW_CFG_3

0x170 INGR0_MPLS_FLOW_CFG_4

0x180 INGR0_MPLS_FLOW_CFG_5

0x190 INGR0_MPLS_FLOW_CFG_6

0x1A0 INGR0_MPLS_FLOW_CFG_7

4.28.3 MPLS Flow Control

Short Name: INGR0_MPLS_FLOW_CONTROL

Addresses: 0x130 INGR0_MPLS_FLOW_CFG_0

0x140 INGR0_MPLS_FLOW_CFG_1

0x150 INGR0_MPLS_FLOW_CFG_2

0x160 INGR0_MPLS_FLOW_CFG_3

0x170 INGR0_MPLS_FLOW_CFG_4

0x180 INGR0_MPLS_FLOW_CFG_5

0x190 INGR0_MPLS_FLOW_CFG_6

0x1A0 INGR0_MPLS_FLOW_CFG_7

Table 228 • **MPLS Flow Control Register**

Bit	Name	Description	Access	Default
25:24	INGR0_MPLS_CHANNEL_MASK	bit 0: Flow valid for channel 0 bit 1: Flow valid for channel 1	R/W	0x3
19:16	INGR0_MPLS_STACK_DEPTH	Defines the allowable stack depths for searches. The direction that the stack is referenced is determined by the setting of MPLS_REF_PNT. For each bit set, The following table maps bits to stack depths: bit 0: stack allowed to be 1 label deep bit 1: stack allowed to be 2 labels deep bit 2: stack allowed to be 3 labels deep bit 3: stack allowed to be 4 labels deep	R/W	0x0
4	INGR0_MPLS_REF_PNT	Defines the search direction for label matching 0: All searching is performed starting from the top of the stack 1: All searching is performed from the end of the stack	R/W	0x0
0	INGR0_MPLS_FLOW_ENA	Flow enable. If this comparator block is not used, all flow enable bits must be set to 0. 0: Flow is disabled 1: Flow is enabled	R/W	0x0

4.28.4 MPLS Label 0 Match Range Lower Value

Short Name: INGR0_MPLS_LABEL_RANGE_LOWER_0

Addresses: 0x132 INGR0_MPLS_FLOW_CFG_0

0x142 INGR0_MPLS_FLOW_CFG_1

0x152 INGR0_MPLS_FLOW_CFG_2

0x162 INGR0_MPLS_FLOW_CFG_3

0x172 INGR0_MPLS_FLOW_CFG_4

0x182 INGR0_MPLS_FLOW_CFG_5

0x192 INGR0_MPLS_FLOW_CFG_6

0x1A2 INGR0_MPLS_FLOW_CFG_7

Table 229 • **MPLS Label 0 Match Range Lower Value Register**

Bit	Name	Description	Access	Default
19:0	INGR0_MPLS_LABEL_RANGE_LOWER_0	Lower value for label 0 match range	R/W	0x00000

4.28.5 MPLS Label 0 Match Range Upper Value

Short Name: INGR0_MPLS_LABEL_RANGE_UPPER_0

Addresses: 0x133 INGR0_MPLS_FLOW_CFG_0

0x143 INGR0_MPLS_FLOW_CFG_1

0x153 INGR0_MPLS_FLOW_CFG_2

0x163 INGR0_MPLS_FLOW_CFG_3

0x173 INGR0_MPLS_FLOW_CFG_4

0x183 INGR0_MPLS_FLOW_CFG_5

0x193 INGR0_MPLS_FLOW_CFG_6

0x1A3 INGR0_MPLS_FLOW_CFG_7

Table 230 • **MPLS Label 0 Match Range Upper Value Register**

Bit	Name	Description	Access	Default
19:0	INGR0_MPLS_LABEL_RANGE_UPPER_0	Upper value for label 0 match range	R/W	0xFFFFF

4.28.6 MPLS Label 1 Match Range Lower Value

Short Name: INGR0_MPLS_LABEL_RANGE_LOWER_1

Addresses: 0x134 INGR0_MPLS_FLOW_CFG_0

0x144 INGR0_MPLS_FLOW_CFG_1

0x154 INGR0_MPLS_FLOW_CFG_2

0x164 INGR0_MPLS_FLOW_CFG_3

0x174 INGR0_MPLS_FLOW_CFG_4

0x184 INGR0_MPLS_FLOW_CFG_5

0x194 INGR0_MPLS_FLOW_CFG_6

0x1A4 INGR0_MPLS_FLOW_CFG_7

Table 231 • **MPLS Label 1 Match Range Lower Value Register**

Bit	Name	Description	Access	Default
19:0	INGR0_MPLS_LABEL_RANGE_LOWER_1	Lower value for label 1 match range	R/W	0x00000

4.28.7 MPLS Label 1 Match Range Lower Value

Short Name: INGR0_MPLS_LABEL_RANGE_UPPER_1

Addresses: 0x135 INGR0_MPLS_FLOW_CFG_0

0x145 INGR0_MPLS_FLOW_CFG_1

0x155 INGR0_MPLS_FLOW_CFG_2

0x165 INGR0_MPLS_FLOW_CFG_3

0x175 INGR0_MPLS_FLOW_CFG_4

0x185 INGR0_MPLS_FLOW_CFG_5

0x195 INGR0_MPLS_FLOW_CFG_6

0x1A5 INGR0_MPLS_FLOW_CFG_7

Table 232 • **MPLS Label 1 Match Range Lower Value Register**

Bit	Name	Description	Access	Default
19:0	INGR0_MPLS_LABEL_RANGE_UPPER_1	Upper value for label 1 match range	R/W	0xFFFFF

4.28.8 MPLS Label 2 Match Range Lower Value

Short Name: INGR0_MPLS_LABEL_RANGE_LOWER_2

Addresses: 0x136 INGR0_MPLS_FLOW_CFG_0

0x146 INGR0_MPLS_FLOW_CFG_1

0x156 INGR0_MPLS_FLOW_CFG_2

0x166 INGR0_MPLS_FLOW_CFG_3

0x176 INGR0_MPLS_FLOW_CFG_4

0x186 INGR0_MPLS_FLOW_CFG_5

0x196 INGR0_MPLS_FLOW_CFG_6

0x1A6 INGR0_MPLS_FLOW_CFG_7

Table 233 • **MPLS Label 2 Match Range Lower Value Register**

Bit	Name	Description	Access	Default
19:0	INGR0_MPLS_LABEL_RANGE_LOWER_2	Lower value for label 2 match range	R/W	0x00000

4.28.9 MPLS Label 2 Match Range Lower Value

Short Name: INGR0_MPLS_LABEL_RANGE_UPPER_2

Addresses: 0x137 INGR0_MPLS_FLOW_CFG_0

0x147 INGR0_MPLS_FLOW_CFG_1

0x157 INGR0_MPLS_FLOW_CFG_2

0x167 INGR0_MPLS_FLOW_CFG_3

0x177 INGR0_MPLS_FLOW_CFG_4

0x187 INGR0_MPLS_FLOW_CFG_5

0x197 INGR0_MPLS_FLOW_CFG_6

0x1A7 INGR0_MPLS_FLOW_CFG_7

Table 234 • **MPLS Label 2 Match Range Lower Value Register**

Bit	Name	Description	Access	Default
19:0	INGR0_MPLS_LABEL_RANGE_UPPER_2	Upper value for label 2 match range	R/W	0xFFFFF

4.28.10 MPLS Label 3 Match Range Lower Value

Short Name: INGR0_MPLS_LABEL_RANGE_LOWER_3

Addresses: 0x138 INGR0_MPLS_FLOW_CFG_0

0x148 INGR0_MPLS_FLOW_CFG_1

0x158 INGR0_MPLS_FLOW_CFG_2

0x168 INGR0_MPLS_FLOW_CFG_3

0x178 INGR0_MPLS_FLOW_CFG_4

0x188 INGR0_MPLS_FLOW_CFG_5

0x198 INGR0_MPLS_FLOW_CFG_6

0x1A8 INGR0_MPLS_FLOW_CFG_7

Table 235 • **MPLS Label 3 Match Range Lower Value Register**

Bit	Name	Description	Access	Default
19:0	INGR0_MPLS_LABEL_RANGE_LOWER_3	Lower value for label 3 match range	R/W	0x00000

4.28.11 MPLS Label 3 Match Range Lower Value

Short Name: INGR0_MPLS_LABEL_RANGE_UPPER_3

Addresses: 0x139 INGR0_MPLS_FLOW_CFG_0

0x149 INGR0_MPLS_FLOW_CFG_1

0x159 INGR0_MPLS_FLOW_CFG_2

0x169 INGR0_MPLS_FLOW_CFG_3

0x179 INGR0_MPLS_FLOW_CFG_4

0x189 INGR0_MPLS_FLOW_CFG_5

0x199 INGR0_MPLS_FLOW_CFG_6

0x1A9 INGR0_MPLS_FLOW_CFG_7

Table 236 • **MPLS Label 3 Match Range Lower Value Register**

Bit	Name	Description	Access	Default
19:0	INGR0_MPLS_LABEL_RANGE_UPPER_3	Upper value for label 3 match range	R/W	0xFFFFF

4.28.12 IP Next Comparator Control

Short Name: INGR0_IP1_NXT_COMPARATOR

Address: 0x1B0

Table 237 • **IP Next Comparator Control Register**

Bit	Name	Description	Access	Default
15:8	INGR0_IP1_NXT_PROTOCOL	Number of bytes in this header, points to the beginning of the next protocol	R/W	0x00

Table 237 • **IP Next Comparator Control Register *(continued)***

Bit	Name	Description	Access	Default
2:0	INGR0_IP1_NXT_COMPARATOR	Points to the next comparator stage. If this comparator block is not used, this field must be set to 0. 0: Comparator block not used 1: Reserved 2: Reserved 3: IP/UDP/ACH comparator 2 4: Reserved 5: PTP/OAM comparator 6,7: Reserved	R/W	0x0

4.28.13 IP Comparator Mode

Short Name: INGR0_IP1_MODE

Address: 0x1B1

Table 238 • **IP Comparator Mode Register**

Bit	Name	Description	Access	Default
12:8	INGR0_IP1_FLOW_OFFSET	Points to the source address field in the IP frame. Use 12 for IPv4 and 8 for IPv6	R/W	0x0C
1:0	INGR0_IP1_MODE	0: IPv4 1: IPv6 2: Other protocol, 32-bit address match 3: Other protocol, 128-bit address match	R/W	0x0

4.28.14 IP Match Register Set 1

Short Name: INGR0_IP1_PROT_MATCH_1

Address: 0x1B2

Table 239 • **IP Match Register Set 1 Register**

Bit	Name	Description	Access	Default
20:16	INGR0_IP1_PROT_OFFSET_1	Points to the start of this match field relative to the first byte of this protocol	R/W	0x00
15:8	INGR0_IP1_PROT_MASK_1	Mask field for IP_PROT_MATCH_1	R/W	0x00
7:0	INGR0_IP1_PROT_MATCH_1	8-bit match field	R/W	0x00

4.28.15 Upper Portion of Match 2

Short Name: INGR0_IP1_PROT_MATCH_2_UPPER

Address: 0x1B3

Table 240 • **Upper Portion of Match 2 Register**

Bit	Name	Description	Access	Default
31:0	INGR0_IP1_PROT_MATCH_2_UPPER	64-bit match register for advancing to the next protocol, upper portion	R/W	0x00000000

4.28.16 Lower Portion of Match 2

Short Name: INGR0_IP1_PROT_MATCH_2_LOWER

Address: 0x1B4

Table 241 • **Lower Portion of Match 2 Register**

Bit	Name	Description	Access	Default
31:0	INGR0_IP1_PROT_MATCH_2_LOWER	64-bit match register for advancing to the next protocol, lower portion	R/W	0x00000000

4.28.17 Upper Portion of Match Mask 2

Short Name: INGR0_IP1_PROT_MASK_2_UPPER

Address: 0x1B5

Table 242 • **Upper Portion of Match Mask 2 Register**

Bit	Name	Description	Access	Default
31:0	INGR0_IP1_PROT_MASK_2_UPPER		R/W	0x00000000

4.28.18 Lower Portion of Match Mask 2

Short Name: INGR0_IP1_PROT_MASK_2_LOWER

Address: 0x1B6

Table 243 • **Lower Portion of Match Mask 2 Register**

Bit	Name	Description	Access	Default
31:0	INGR0_IP1_PROT_MASK_2_LOWER		R/W	0x00000000

4.28.19 Match Offset 2

Short Name: INGR0_IP1_PROT_OFFSET_2

Address: 0x1B7

Table 244 • **Match Offset 2 Register**

Bit	Name	Description	Access	Default
6:0	INGR0_IP1_PROT_OFFSET_2	Points to the start of match field 2 relative to the first byte of this protocol	R/W	0x00

4.28.20 IP/UDP Checksum Control

Short Name: INGR0_IP1_UDP_CHKSUM_CFG

Address: 0x1B8

Table 245 • **IP/UDP Checksum Control Register**

Bit	Name	Description	Access	Default
15:8	INGR0_IP1_UDP_CHKSUM_OFFSET	Pointer to the IP/UDP checksum field FOR IPv4 frames or to the pad bytes of a IPv6/UDP frame. For IPv4, it points to the bytes that will be cleared. For IPv6, it points to the bytes that will be updated to fix the CRC	R/W	0x00
5:4	INGR0_IP1_UDP_CHKSUM_WIDTH	Specifies the length of the checksum field in bytes	R/W	0x2
1	INGR0_IP1_UDP_CHKSUM_UPDATE_ENA	This bit and IP_UDP_CHKSUM_CLEAR_ENA CANNOT be set together. 1: Update the pad bytes at the end of the frame 0: No pad byte field update	R/W	0x0
0	INGR0_IP1_UDP_CHKSUM_CLEAR_ENA	This bit and IP_UDP_CHKSUM_UPDATE_ENA CANNOT be set together. 1: Clear the UDP checksum field in an IPv4 frame 0: Do not clear the checksum	R/W	0x0

4.28.21

Instance offsets: 0x1C0 INGR0_IP1_FLOW_CFG_0

0x1D0 INGR0_IP1_FLOW_CFG_1

0x1E0 INGR0_IP1_FLOW_CFG_2

0x1F0 INGR0_IP1_FLOW_CFG_3

0x200 INGR0_IP1_FLOW_CFG_4

0x210 INGR0_IP1_FLOW_CFG_5

0x220 INGR0_IP1_FLOW_CFG_6

0x230 INGR0_IP1_FLOW_CFG_7

4.28.22 IP Flow Enable

Short Name: INGR0_IP1_FLOW_ENA

Addresses: 0x1C0 INGR0_IP1_FLOW_CFG_0

0x1D0 INGR0_IP1_FLOW_CFG_1

0x1E0 INGR0_IP1_FLOW_CFG_2

0x1F0 INGR0_IP1_FLOW_CFG_3

0x200 INGR0_IP1_FLOW_CFG_4

0x210 INGR0_IP1_FLOW_CFG_5

0x220 INGR0_IP1_FLOW_CFG_6

0x230 INGR0_IP1_FLOW_CFG_7

Table 246 • **IP Flow Enable Register**

Bit	Name	Description	Access	Default
9:8	INGR0_IP1_FLOW_MATCH_MODE	0: Match on source address 1: Match on destination address 2: Match on either source or destination address 3: reserved	R/W	0x0
5:4	INGR0_IP1_CHANNEL_MASK	bit 0: Flow valid for channel 0 bit 1: Flow valid for channel 1	R/W	0x3
0	INGR0_IP1_FLOW_ENA	Flow enable. If this comparator block is not used, all flow enable bits must be set to 0. 1: This flow is enabled 0: This flow is not enabled	R/W	0x0

4.28.23 Upper Portion of the IP Flow Match

Short Name: INGR0_IP1_FLOW_MATCH_UPPER

Addresses: 0x1C1 INGR0_IP1_FLOW_CFG_0

0x1D1 INGR0_IP1_FLOW_CFG_1

0x1E1 INGR0_IP1_FLOW_CFG_2

0x1F1 INGR0_IP1_FLOW_CFG_3

0x201 INGR0_IP1_FLOW_CFG_4

0x211 INGR0_IP1_FLOW_CFG_5

0x221 INGR0_IP1_FLOW_CFG_6

0x231 INGR0_IP1_FLOW_CFG_7

Table 247 • **Upper Portion of the IP Flow Match Register**

Bit	Name	Description	Access	Default
31:0	INGR0_IP1_FLOW_MATCH_UPPER	Match field for either the entire 32-bit selected address for IPv4 or the upper 32 bits of the selected address for IPv6	R/W	0x00000000

4.28.24 Upper Mid Portion of the IP Flow Match

Short Name: INGR0_IP1_FLOW_MATCH_UPPER_MID

Addresses: 0x1C2 INGR0_IP1_FLOW_CFG_0

0x1D2 INGR0_IP1_FLOW_CFG_1

0x1E2 INGR0_IP1_FLOW_CFG_2

0x1F2 INGR0_IP1_FLOW_CFG_3

0x202 INGR0_IP1_FLOW_CFG_4

0x212 INGR0_IP1_FLOW_CFG_5

0x222 INGR0_IP1_FLOW_CFG_6

0x232 INGR0_IP1_FLOW_CFG_7

Table 248 • **Upper Mid Portion of the IP Flow Match Register**

Bit	Name	Description	Access	Default
31:0	INGR0_IP1_FLOW_MATCH_UPPER_MID	Match bits for the upper middle 32 bits of the IPv6 address	R/W	0x00000000

4.28.25 Lower Mid Portion of the IP Flow Match

Short Name: INGR0_IP1_FLOW_MATCH_LOWER_MID

Addresses: 0x1C3 INGR0_IP1_FLOW_CFG_0

0x1D3 INGR0_IP1_FLOW_CFG_1

0x1E3 INGR0_IP1_FLOW_CFG_2

0x1F3 INGR0_IP1_FLOW_CFG_3

0x203 INGR0_IP1_FLOW_CFG_4

0x213 INGR0_IP1_FLOW_CFG_5

0x223 INGR0_IP1_FLOW_CFG_6

0x233 INGR0_IP1_FLOW_CFG_7

Table 249 • **Lower Mid Portion of the IP Flow Match Register**

Bit	Name	Description	Access	Default
31:0	INGR0_IP1_FLOW_MATCH_LOWER_MID	Match bits for the lower middle 32 bits of the IPv6 address	R/W	0x00000000

4.28.26 Lower Portion of the IP Flow Match

Short Name: INGR0_IP1_FLOW_MATCH_LOWER

Addresses: 0x1C4 INGR0_IP1_FLOW_CFG_0

0x1D4 INGR0_IP1_FLOW_CFG_1

0x1E4 INGR0_IP1_FLOW_CFG_2

0x1F4 INGR0_IP1_FLOW_CFG_3

0x204 INGR0_IP1_FLOW_CFG_4

0x214 INGR0_IP1_FLOW_CFG_5

0x224 INGR0_IP1_FLOW_CFG_6

0x234 INGR0_IP1_FLOW_CFG_7

Table 250 • **Lower Portion of the IP Flow Match Register**

Bit	Name	Description	Access	Default
31:0	INGR0_IP1_FLOW_MATCH_LOWER	Match bits for the lower 32 bits of the IPv6 address	R/W	0x00000000

4.28.27 IP Flow Match Mask

Short Name: INGR0_IP1_FLOW_MASK_UPPER

Addresses: 0x1C5 INGR0_IP1_FLOW_CFG_0

0x1D5 INGR0_IP1_FLOW_CFG_1

0x1E5 INGR0_IP1_FLOW_CFG_2

0x1F5 INGR0_IP1_FLOW_CFG_3

0x205 INGR0_IP1_FLOW_CFG_4

0x215 INGR0_IP1_FLOW_CFG_5

0x225 INGR0_IP1_FLOW_CFG_6

0x235 INGR0_IP1_FLOW_CFG_7

Table 251 • **IP Flow Match Mask Register**

Bit	Name	Description	Access	Default
31:0	INGR0_IP1_FLOW_MASK_UPPER	This is the address mask for the IP address	R/W	0x00000000

4.28.28 Upper Mid Portion of the IP Flow Mask

Short Name: INGR0_IP1_FLOW_MASK_UPPER_MID

Addresses: 0x1C6 INGR0_IP1_FLOW_CFG_0

0x1D6 INGR0_IP1_FLOW_CFG_1

0x1E6 INGR0_IP1_FLOW_CFG_2

0x1F6 INGR0_IP1_FLOW_CFG_3

0x206 INGR0_IP1_FLOW_CFG_4

0x216 INGR0_IP1_FLOW_CFG_5

0x226 INGR0_IP1_FLOW_CFG_6

0x236 INGR0_IP1_FLOW_CFG_7

Table 252 • **Upper Mid Portion of the IP Flow Mask Register**

Bit	Name	Description	Access	Default
31:0	INGR0_IP1_FLOW_MASK_UPPER_MID	These bits must be all 0 for IPv4 and any 32-bit address match mode	R/W	0x00000000

4.28.29 Lower Mid Portion of the IP Flow Mask

Short Name: INGR0_IP1_FLOW_MASK_LOWER_MID

Addresses: 0x1C7 INGR0_IP1_FLOW_CFG_0

0x1D7 INGR0_IP1_FLOW_CFG_1

0x1E7 INGR0_IP1_FLOW_CFG_2

0x1F7 INGR0_IP1_FLOW_CFG_3

0x207 INGR0_IP1_FLOW_CFG_4

0x217 INGR0_IP1_FLOW_CFG_5

0x227 INGR0_IP1_FLOW_CFG_6

0x237 INGR0_IP1_FLOW_CFG_7

Table 253 • **Lower Mid Portion of the IP Flow Mask Register**

Bit	Name	Description	Access	Default
31:0	INGR0_IP1_FLOW_MASK_LOWER_MID	These bits must be all 0 for IPv4 and any 32-bit address match mode	R/W	0x00000000

4.28.30 Lower Portion of the IP Flow Mask

Short Name: INGR0_IP1_FLOW_MASK_LOWER

Addresses: 0x1C8 INGR0_IP1_FLOW_CFG_0

0x1D8 INGR0_IP1_FLOW_CFG_1

0x1E8 INGR0_IP1_FLOW_CFG_2

0x1F8 INGR0_IP1_FLOW_CFG_3

0x208 INGR0_IP1_FLOW_CFG_4

0x218 INGR0_IP1_FLOW_CFG_5

0x228 INGR0_IP1_FLOW_CFG_6

0x238 INGR0_IP1_FLOW_CFG_7

Table 254 • **Lower Portion of the IP Flow Mask Register**

Bit	Name	Description	Access	Default
31:0	INGR0_IP1_FLOW_MASK_LOWER	These bits must be all 0 for IPv4 and any 32-bit address match mode	R/W	0x00000000

4.28.31 IP Next Comparator Control

Short Name: INGR0_IP2_NXT_COMPARATOR

Address: 0x240

Table 255 • **IP Next Comparator Control Register**

Bit	Name	Description	Access	Default
15:8	INGR0_IP2_NXT_PROTOCOL	Number of bytes in this header, points to the beginning of the next protocol	R/W	0x00
2:0	INGR0_IP2_NXT_COMPARATOR	Points to the next comparator stage. If this comparator block is not used, this field must be set to 0. 0: Comparator block not used 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: PTP/OAM comparator 6,7: Reserved	R/W	0x0

4.28.32 IP Comparator Mode

Short Name: INGR0_IP2_MODE

Address: 0x241

Table 256 • **IP Comparator Mode Register**

Bit	Name	Description	Access	Default
12:8	INGR0_IP2_FLOW_OFFSET	Points to the source address field in the IP frame. Use 12 for IPv4 and 8 for IPv6	R/W	0x0C
1:0	INGR0_IP2_MODE	0: IPv4 1: IPv6 2: Other protocol, 32-bit address match 3: Other protocol, 128-bit address match	R/W	0x0

4.28.33 IP Match Set 1

Short Name: INGR0_IP2_PROT_MATCH_1

Address: 0x242

Table 257 • **IP Match Set 1 Register**

Bit	Name	Description	Access	Default
20:16	INGR0_IP2_PROT_OFFSET_1	Points to the start of this match field relative to the first byte of this protocol	R/W	0x00
15:8	INGR0_IP2_PROT_MASK_1	Mask field for IP_PROT_MATCH_1	R/W	0x00
7:0	INGR0_IP2_PROT_MATCH_1	8-bit match field	R/W	0x00

4.28.34 Upper Portion of Match 2

Short Name: INGR0_IP2_PROT_MATCH_2_UPPER

Address: 0x243

Table 258 • **Upper Portion of Match 2 Register**

Bit	Name	Description	Access	Default
31:0	INGR0_IP2_PROT_MATCH_2_UPPER	64-bit match register for advancing to the next protocol, upper portion	R/W	0x00000000

4.28.35 Lower Portion of Match 2

Short Name: INGR0_IP2_PROT_MATCH_2_LOWER

Address: 0x244

Table 259 • **Lower Portion of Match 2 Register**

Bit	Name	Description	Access	Default
31:0	INGR0_IP2_PROT_MATCH_2_LOWER	64-bit match register for advancing to the next protocol, lower portion	R/W	0x00000000

4.28.36 Upper Portion of Match Mask 2

Short Name: INGR0_IP2_PROT_MASK_2_UPPER

Address: 0x245

Table 260 • **Upper Portion of Match Mask 2 Register**

Bit	Name	Description	Access	Default
31:0	INGR0_IP2_PROT_MASK_2_UPPER		R/W	0x00000000

4.28.37 Lower Portion of Match Mask 2

Short Name: INGR0_IP2_PROT_MASK_2_LOWER

Address: 0x246

Table 261 • **Lower Portion of Match Mask 2 Register**

Bit	Name	Description	Access	Default
31:0	INGR0_IP2_PROT_MASK_2_LOWER		R/W	0x00000000

4.28.38 Match Offset 2

Short Name: INGR0_IP2_PROT_OFFSET_2

Address: 0x247

Table 262 • **Match Offset 2 Register**

Bit	Name	Description	Access	Default
6:0	INGR0_IP2_PROT_OFFSET_2	Points to the start of match field 2 relative to the first byte of this protocol	R/W	0x00

4.28.39 IP/UDP Checksum Control

Short Name: INGR0_IP2_UDP_CHKSUM_CFG

Address: 0x248

Table 263 • **IP/UDP Checksum Control Register**

Bit	Name	Description	Access	Default
15:8	INGR0_IP2_UDP_CHKSUM_OFFSET	Pointer to the IP/UDP checksum field FOR IPv4 frames or to the pad bytes of a IPv6/UDP frame. For IPv4, it points to the bytes that will be cleared. For IPv6, it points to the bytes that will be updated to fix the CRC	R/W	0x00
5:4	INGR0_IP2_UDP_CHKSUM_WIDTH	Specifies the length of the checksum field in bytes	R/W	0x2
1	INGR0_IP2_UDP_CHKSUM_UPDATE_ENA	This bit and IP_UDP_CHKSUM_CLEAR_ENA CANNOT be set together. 1: Update the pad bytes at the end of the frame 0: No pad byte field update	R/W	0x0
0	INGR0_IP2_UDP_CHKSUM_CLEAR_ENA	This bit and IP_UDP_CHKSUM_UPDATE_ENA CANNOT be set together. 1: Clear the UDP checksum field in an IPv4 frame 0: Do not clear the checksum	R/W	0x0

4.28.40

Instance offsets: 0x250 INGR0_IP2_FLOW_CFG_0

0x260 INGR0_IP2_FLOW_CFG_1

0x270 INGR0_IP2_FLOW_CFG_2

0x280 INGR0_IP2_FLOW_CFG_3

0x290 INGR0_IP2_FLOW_CFG_4

0x2A0 INGR0_IP2_FLOW_CFG_5

0x2B0 INGR0_IP2_FLOW_CFG_6

0x2C0 INGR0_IP2_FLOW_CFG_7

4.28.41 IP Flow Enable

Short Name: INGR0_IP2_FLOW_ENA

Addresses: 0x250 INGR0_IP2_FLOW_CFG_0

0x260 INGR0_IP2_FLOW_CFG_1

0x270 INGR0_IP2_FLOW_CFG_2

0x280 INGR0_IP2_FLOW_CFG_3

0x290 INGR0_IP2_FLOW_CFG_4

0x2A0 INGR0_IP2_FLOW_CFG_5

0x2B0 INGR0_IP2_FLOW_CFG_6

0x2C0 INGR0_IP2_FLOW_CFG_7

Table 264 • **IP Flow Enable Register**

Bit	Name	Description	Access	Default
9:8	INGR0_IP2_FLOW_MATCH_MODE	0: Match on source address 1: Match on destination address 2: Match on either source or destination address 3: reserved	R/W	0x0
5:4	INGR0_IP2_CHANNEL_MASK	bit 0: Flow valid for channel 0 bit 1: Flow valid for channel 1	R/W	0x3
0	INGR0_IP2_FLOW_ENA	Flow enable. If this comparator block is not used, all flow enable bits must be set to 0. 1: This flow is enabled 0: This flow is not enabled	R/W	0x0

4.28.42 Upper Portion of the IP Flow Match

Short Name: INGR0_IP2_FLOW_MATCH_UPPER

Addresses: 0x251 INGR0_IP2_FLOW_CFG_0

0x261 INGR0_IP2_FLOW_CFG_1

0x271 INGR0_IP2_FLOW_CFG_2

0x281 INGR0_IP2_FLOW_CFG_3

0x291 INGR0_IP2_FLOW_CFG_4

0x2A1 INGR0_IP2_FLOW_CFG_5

0x2B1 INGR0_IP2_FLOW_CFG_6

0x2C1 INGR0_IP2_FLOW_CFG_7

Table 265 • **Upper Portion of the IP Flow Match Register**

Bit	Name	Description	Access	Default
31:0	INGR0_IP2_FLOW_MATCH_UPPER	Match field for either the entire 32-bit selected address for IPv4 or the upper 32 bits of the selected address for IPv6	R/W	0x00000000

4.28.43 Upper Mid Portion of the IP Flow Match

Short Name: INGR0_IP2_FLOW_MATCH_UPPER_MID

Addresses: 0x252 INGR0_IP2_FLOW_CFG_0

0x262 INGR0_IP2_FLOW_CFG_1

0x272 INGR0_IP2_FLOW_CFG_2

0x282 INGR0_IP2_FLOW_CFG_3

0x292 INGR0_IP2_FLOW_CFG_4

0x2A2 INGR0_IP2_FLOW_CFG_5

0x2B2 INGR0_IP2_FLOW_CFG_6

0x2C2 INGR0_IP2_FLOW_CFG_7

Table 266 • **Upper Mid Portion of the IP Flow Match Register**

Bit	Name	Description	Access	Default
31:0	INGR0_IP2_FLOW_MATCH_UPPER_MID	Match bits for the upper middle 32 bits of the IPv6 address	R/W	0x00000000

4.28.44 Lower Mid Portion of the IP Flow Match

Short Name: INGR0_IP2_FLOW_MATCH_LOWER_MID

Addresses: 0x253 INGR0_IP2_FLOW_CFG_0

0x263 INGR0_IP2_FLOW_CFG_1

0x273 INGR0_IP2_FLOW_CFG_2

0x283 INGR0_IP2_FLOW_CFG_3

0x293 INGR0_IP2_FLOW_CFG_4

0x2A3 INGR0_IP2_FLOW_CFG_5

0x2B3 INGR0_IP2_FLOW_CFG_6

0x2C3 INGR0_IP2_FLOW_CFG_7

Table 267 • **Lower Mid Portion of the IP Flow Match Register**

Bit	Name	Description	Access	Default
31:0	INGR0_IP2_FLOW_MATCH_LOWER_MID	Match bits for the lower middle 32 bits of the IPv6 address	R/W	0x00000000

4.28.45 Lower Portion of the IP Flow Match

Short Name: INGR0_IP2_FLOW_MATCH_LOWER

Addresses: 0x254 INGR0_IP2_FLOW_CFG_0

0x264 INGR0_IP2_FLOW_CFG_1

0x274 INGR0_IP2_FLOW_CFG_2

0x284 INGR0_IP2_FLOW_CFG_3

0x294 INGR0_IP2_FLOW_CFG_4

0x2A4 INGR0_IP2_FLOW_CFG_5

0x2B4 INGR0_IP2_FLOW_CFG_6

0x2C4 INGR0_IP2_FLOW_CFG_7

Table 268 • **Lower Portion of the IP Flow Match Register**

Bit	Name	Description	Access	Default
31:0	INGR0_IP2_FLOW_MATCH_LOWER	Match bits for the lower 32 bits of the IPv6 address	R/W	0x00000000

4.28.46 IP Flow Match Mask

Short Name: INGR0_IP2_FLOW_MASK_UPPER

Addresses: 0x255 INGR0_IP2_FLOW_CFG_0

0x265 INGR0_IP2_FLOW_CFG_1

0x275 INGR0_IP2_FLOW_CFG_2

0x285 INGR0_IP2_FLOW_CFG_3

0x295 INGR0_IP2_FLOW_CFG_4

0x2A5 INGR0_IP2_FLOW_CFG_5

0x2B5 INGR0_IP2_FLOW_CFG_6

0x2C5 INGR0_IP2_FLOW_CFG_7

Table 269 • **IP Flow Match Mask Register**

Bit	Name	Description	Access	Default
31:0	INGR0_IP2_FLOW_MASK_UPPER	This is the address mask for the IP address.	R/W	0x00000000

4.28.47 Upper Mid Portion of the IP Flow Mask

Short Name: INGR0_IP2_FLOW_MASK_UPPER_MID

Addresses: 0x256 INGR0_IP2_FLOW_CFG_0

0x266 INGR0_IP2_FLOW_CFG_1

0x276 INGR0_IP2_FLOW_CFG_2

0x286 INGR0_IP2_FLOW_CFG_3

0x296 INGR0_IP2_FLOW_CFG_4

0x2A6 INGR0_IP2_FLOW_CFG_5

0x2B6 INGR0_IP2_FLOW_CFG_6

0x2C6 INGR0_IP2_FLOW_CFG_7

Table 270 • **Upper Mid Portion of the IP Flow Mask Register**

Bit	Name	Description	Access	Default
31:0	INGR0_IP2_FLOW_MASK_UPPER_MID	These bits must be all 0 for IPv4 and any 32-bit address match mode	R/W	0x00000000

4.28.48 Lower Mid Portion of the IP Flow Mask

Short Name: INGR0_IP2_FLOW_MASK_LOWER_MID

Addresses: 0x257 INGR0_IP2_FLOW_CFG_0

0x267 INGR0_IP2_FLOW_CFG_1

0x277 INGR0_IP2_FLOW_CFG_2

0x287 INGR0_IP2_FLOW_CFG_3

0x297 INGR0_IP2_FLOW_CFG_4

0x2A7 INGR0_IP2_FLOW_CFG_5

0x2B7 INGR0_IP2_FLOW_CFG_6

0x2C7 INGR0_IP2_FLOW_CFG_7

Table 271 • **Lower Mid Portion of the IP Flow Mask Register**

Bit	Name	Description	Access	Default
31:0	INGR0_IP2_FLOW_MASK_LOWER_MID	These bits must be all 0 for IPv4 and any 32-bit address match mode	R/W	0x00000000

4.28.49 Lower Portion of the IP Flow Mask

Short Name: INGR0_IP2_FLOW_MASK_LOWER

Addresses: 0x258 INGR0_IP2_FLOW_CFG_0

0x268 INGR0_IP2_FLOW_CFG_1

0x278 INGR0_IP2_FLOW_CFG_2

0x288 INGR0_IP2_FLOW_CFG_3

0x298 INGR0_IP2_FLOW_CFG_4

0x2A8 INGR0_IP2_FLOW_CFG_5

0x2B8 INGR0_IP2_FLOW_CFG_6

0x2C8 INGR0_IP2_FLOW_CFG_7

Table 272 • **Lower Portion of the IP Flow Mask Register**

Bit	Name	Description	Access	Default
31:0	INGR0_IP2_FLOW_MASK_LOWER	These bits must be all 0 for IPv4 and any 32-bit address match mode	R/W	0x00000000

4.28.50

Instance offsets: 0x2D0 INGR0_PTP_FLOW_0

0x2E0 INGR0_PTP_FLOW_1

0x2F0 INGR0_PTP_FLOW_2

0x300 INGR0_PTP_FLOW_3

0x310 INGR0_PTP_FLOW_4

0x320 INGR0_PTP_FLOW_5

4.28.51 PTP/OAM Flow Enable

Short Name: INGR0_PTP_FLOW_ENA

Addresses: 0x2D0 INGR0_PTP_FLOW_0

0x2E0 INGR0_PTP_FLOW_1

0x2F0 INGR0_PTP_FLOW_2

0x300 INGR0_PTP_FLOW_3

0x310 INGR0_PTP_FLOW_4

0x320 INGR0_PTP_FLOW_5

Table 273 • **PTP/OAM Flow Enable Register**

Bit	Name	Description	Access	Default
5:4	INGR0_PTP_CHANNEL_MASK	bit 0: Flow valid for channel 0 bit 1: Flow valid for channel 1	R/W	0x3
0	INGR0_PTP_FLOW_ENA		R/W	0x0

4.28.52 Upper Half of PTP/OAM Flow Match Field

Short Name: INGR0_PTP_FLOW_MATCH_UPPER

Addresses: 0x2D1 INGR0_PTP_FLOW_0

0x2E1 INGR0_PTP_FLOW_1

0x2F1 INGR0_PTP_FLOW_2

0x301 INGR0_PTP_FLOW_3

0x311 INGR0_PTP_FLOW_4

0x321 INGR0_PTP_FLOW_5

Table 274 • **Upper Half of PTP/OAM Flow Match Field Register**

Bit	Name	Description	Access	Default
31:0	INGR0_PTP_FLOW_MATCH_UPPER		R/W	0x00000000

4.28.53 Lower Half of PTP/OAM Flow Match Field

Short Name: INGR0_PTP_FLOW_MATCH_LOWER

Addresses: 0x2D2 INGR0_PTP_FLOW_0

0x2E2 INGR0_PTP_FLOW_1

0x2F2 INGR0_PTP_FLOW_2

0x302 INGR0_PTP_FLOW_3

0x312 INGR0_PTP_FLOW_4

0x322 INGR0_PTP_FLOW_5

Table 275 • **Lower Half of PTP/OAM Flow Match Field Register**

Bit	Name	Description	Access	Default
31:0	INGR0_PTP_FLOW_MATCH_LOWER		R/W	0x00000000

4.28.54 Upper Half of PTP/OAM Flow Match Mask

Short Name: INGR0_PTP_FLOW_MASK_UPPER

Addresses: 0x2D3 INGR0_PTP_FLOW_0

0x2E3 INGR0_PTP_FLOW_1

0x2F3 INGR0_PTP_FLOW_2

0x303 INGR0_PTP_FLOW_3

0x313 INGR0_PTP_FLOW_4

0x323 INGR0_PTP_FLOW_5

Table 276 • **Upper Half of PTP/OAM Flow Match Mask Register**

Bit	Name	Description	Access	Default
31:0	INGR0_PTP_FLOW_MASK_UPPER		R/W	0x00000000

4.28.55 Lower Half of PTP/OAM Flow Match Mask

Short Name: INGR0_PTP_FLOW_MASK_LOWER

Addresses: 0x2D4 INGR0_PTP_FLOW_0

0x2E4 INGR0_PTP_FLOW_1

0x2F4 INGR0_PTP_FLOW_2

0x304 INGR0_PTP_FLOW_3

0x314 INGR0_PTP_FLOW_4

0x324 INGR0_PTP_FLOW_5

Table 277 • **Lower Half of PTP/OAM Flow Match Mask Register**

Bit	Name	Description	Access	Default
31:0	INGR0_PTP_FLOW_MASK_LOWER		R/W	0x00000000

4.28.56 PTP/OAM Range Match

Short Name: INGR0_PTP_DOMAIN_RANGE

Addresses: 0x2D5 INGR0_PTP_FLOW_0

0x2E5 INGR0_PTP_FLOW_1

0x2F5 INGR0_PTP_FLOW_2

0x305 INGR0_PTP_FLOW_3

0x315 INGR0_PTP_FLOW_4

0x325 INGR0_PTP_FLOW_5

Table 278 • **PTP/OAM Range Match Register**

Bit	Name	Description	Access	Default
28:24	INGR0_PTP_DOMAIN_RANGE_OFFSET		R/W	0x00
23:16	INGR0_PTP_DOMAIN_RANGE_UPPER		R/W	0xFF
15:8	INGR0_PTP_DOMAIN_RANGE_LOWER		R/W	0x00
0	INGR0_PTP_DOMAIN_RANGE_ENA		R/W	0x0

4.28.57 PTP Action Control

Short Name: INGR0_PTP_ACTION

Addresses: 0x2D6 INGR0_PTP_FLOW_0

0x2E6 INGR0_PTP_FLOW_1

0x2F6 INGR0_PTP_FLOW_2

0x306 INGR0_PTP_FLOW_3

0x316 INGR0_PTP_FLOW_4

0x326 INGR0_PTP_FLOW_5

Table 279 • **PTP Action Control Register**

Bit	Name	Description	Access	Default
28	INGR0_PTP_MOD_FRAME_STAT_UPDATE	1: Tell the Rewriter to update the value of the Modified Frame Status bit 0: Do not update the bit	R/W	0x0
26:24	INGR0_PTP_MOD_FRAME_BYTE_OFFSET	Indicates the position relative to the start of the PTP frame in bytes where the Modified_Frame_Status bit resides	R/W	0x0
21	INGR0_PTP_SUB_DELAY_ASYM_ENA	1: Signal the Timestamp block to subtract the asymmetry delay 0: Do not signal the Timestamp block to subtract the asymmetry delay	R/W	0x0
20	INGR0_PTP_ADD_DELAY_ASYM_ENA	1: Signal the Timestamp block to add the asymmetry delay 0: Do not signal the Timestamp block to add the asymmetry delay	R/W	0x0
15:10	INGR0_PTP_TIME_STRG_FIELD_OFFSET	Points to the reserved 32-bit field where the Rx timestamp is saved. The location is relative to the first byte of the PTP/OAM header.	R/W	0x00
9:5	INGR0_PTP_CORR_FIELD_OFFSET	Points to the location of the correction field for updating the timestamp. Location is relative to the first byte of the PTP/OAM header. **Note:** If this flow is being used to match OAM frames, set this register to 4	R/W	0x00

Table 279 • **PTP Action Control Register *(continued)***

Bit	Name	Description	Access	Default
4	INGR0_PTP_SAVE_LOCAL_TIME	1: Save the local time to the Timestamp FIFO 0: Do not save the time to the Timestamp FIFO	R/W	0x0
3:0	INGR0_PTP_COMMAND	0: NoP 1: SUB 2: SUB_P2P 3: ADD 4: SUB_ADD 5: WRITE_1588 6: WRITE_P2P (deprecated) 7: WRITE_NS 8: WRITE_NS_P2P	R/W	0x0

4.28.58 PTP Action Control 2

Short Name: INGR0_PTP_ACTION_2

Addresses: 0x2D7 INGR0_PTP_FLOW_0

0x2E7 INGR0_PTP_FLOW_1

0x2F7 INGR0_PTP_FLOW_2

0x307 INGR0_PTP_FLOW_3

0x317 INGR0_PTP_FLOW_4

0x327 INGR0_PTP_FLOW_5

Table 280 • **PTP Action Control 2 Register**

Bit	Name	Description	Access	Default
23:16	INGR0_PTP_NEW_CF_LOC	Location of the new correction field relative to the PTP header start. Only even values are allowed.	R/W	0x00
15:8	INGR0_PTP_REWRITE_OFFSET	Byte offset relative to the start of the PTP frame where the ingress timestamp value can be stored.	R/W	0x00
3:0	INGR0_PTP_REWRITE_BYTES	Number of bytes in the PTP or OAM frame that must be modified by the Rewriter for the timestamp	R/W	0x0

4.28.59 Zero Field Control

Short Name: INGR0_PTP_ZERO_FIELD_CTL

Addresses: 0x2D8 INGR0_PTP_FLOW_0

0x2E8 INGR0_PTP_FLOW_1

0x2F8 INGR0_PTP_FLOW_2

0x308 INGR0_PTP_FLOW_3

0x318 INGR0_PTP_FLOW_4

0x328 INGR0_PTP_FLOW_5

Table 281 • **Zero Field Control Register**

Bit	Name	Description	Access	Default
13:8	INGR0_PTP_ZERO_FIELD_OFFSET	Points to a location in the PTP/OAM frame relative to the start of the PTP header that will be zeroed if this function is enabled	R/W	0x00
3:0	INGR0_PTP_ZERO_FIELD_BYTE_CNT	The number of bytes to be zeroed. If this field is 0, then this function is not enabled.	R/W	0x0

4.29 Ingress0 IP Checksum Field Control Registers

This section provides information about the IP checksum field control registers.

4.29.1 IP Checksum Block Select

Short Name: INGR0_PTP_IP_CKSUM_SEL

Address: 0x330

Table 282 • **IP Checksum Block Select Register**

Bit	Name	Description	Access	Default
0	INGR0_PTP_IP_CHKSUM_SEL	0: Use the IP checksum controls from IP comparator 1 1: Use the IP checksum controls from IP comparator 2	R/W	0x0

4.30 Egress0 Analyzer Engine Configuration Registers

This section lists overviews for the egress0 analyzer engine configuration registers.Egress1 analyzer engine registers are identical to the ones defined for egress0.

Note: The analyzer engine configuration registers are not initialized to the default values during chip reset. Software must configure these registers to their default value.

Note: For more information about accessing the 1588 IP registers, see Accessing 1588 IP Registers, page 72.

Table 283 • **EGR0_ETH1_NXT_PROTOCOL**

Address	Name	Details
0x00	EGR0_ETH1_NXT_PROTOCOL	Ethernet Next Protocol, page 209
0x01	EGR0_ETH1_VLAN_TPID_CFG	VLAN TPID Configuration, page 209
0x02	EGR0_ETH1_TAG_MODE	Ethernet Tag Mode, page 209
0x03	EGR0_ETH1_ETYPE_MATCH	Ethertype Match, page 210

Table 284 • **EGR0_ETH1_FLOW_CFG (8 instances)**

Address	Name	Details
0x10	EGR0_ETH1_FLOW_ENABLE	Ethernet Flow Enable, page 210
0x11	EGR0_ETH1_MATCH_MODE	Ethernet Protocol Match Mode, page 211

Table 284 • **EGR0_ETH1_FLOW_CFG (8 instances)** ***(continued)***

Address	Name	Details
0x12	EGR0_ETH1_ADDR_MATCH_1	Ethernet Address Match Part 1, page 212
0x13	EGR0_ETH1_ADDR_MATCH_2	Ethernet Address Match Part 2, page 212
0x14	EGR0_ETH1_VLAN_TAG_RANGE_I_TAG	Ethernet VLAN Tag Range Match, page 213
0x15	EGR0_ETH1_VLAN_TAG1	VLAN Tag 1 Match/Mask, page 213
0x16	EGR0_ETH1_VLAN_TAG2_I_TAG	Match/Mask For VLAN Tag 2 or I-Tag Match, page 214

Table 285 • **EGR0_ETH2_NXT_PROTOCOL**

Address	Name	Details
0x90	EGR0_ETH2_NXT_PROTOCOL	Ethernet Next Protocol, page 214
0x91	EGR0_ETH2_VLAN_TPID_CFG	VLAN TPID Configuration, page 215
0x92	EGR0_ETH2_ETYPE_MATCH	Ethertype Match, page 215

Table 286 • **EGR0_ETH2_FLOW_CFG (8 instances)**

Address	Name	Details
0xA0	EGR0_ETH2_FLOW_ENABLE	Ethernet Flow Enable, page 215
0xA1	EGR0_ETH2_MATCH_MODE	Ethernet Protocol Match Mode, page 216
0xA2	EGR0_ETH2_ADDR_MATCH_1	Ethernet Address Match Part 1, page 217
0xA3	EGR0_ETH2_ADDR_MATCH_2	Ethernet Address Match Part 2, page 217
0xA4	EGR0_ETH2_VLAN_TAG_RANGE_I_TAG	Ethernet VLAN Tag Range Match, page 218
0xA5	EGR0_ETH2_VLAN_TAG1	VLAN Tag 1 Match/Mask, page 218
0xA6	EGR0_ETH2_VLAN_TAG2_I_TAG	Match/Mask For VLAN Tag 2 or I-Tag Match, page 219

Table 287 • **EGR0_MPLS_NXT_COMPARATOR**

Address	Name	Details
0x120	EGR0_MPLS_NXT_COMPARATOR	MPLS Next Protocol Comparator, page 219

Table 288 • **EGR0_MPLS_FLOW_CFG (8 instances)**

Address	Name	Details
0x130	EGR0_MPLS_FLOW_CONTROL	MPLS Flow Control, page 220

Table 288 • **EGR0_MPLS_FLOW_CFG (8 instances)** ***(continued)***

Address	Name	Details
0x132	EGR0_MPLS_LABEL_RANGE_LOWER_0	MPLS Label 0 Match Range Lower Value, page 221
0x133	EGR0_MPLS_LABEL_RANGE_UPPER_0	MPLS Label 0 Match Range Upper Value, page 221
0x134	EGR0_MPLS_LABEL_RANGE_LOWER_1	MPLS Label 1 Match Range Lower Value, page 222
0x135	EGR0_MPLS_LABEL_RANGE_UPPER_1	MPLS Label 1 Match Range Lower Value, page 222
0x136	EGR0_MPLS_LABEL_RANGE_LOWER_2	MPLS Label 2 Match Range Lower Value, page 222
0x137	EGR0_MPLS_LABEL_RANGE_UPPER_2	MPLS Label 2 Match Range Lower Value, page 223
0x138	EGR0_MPLS_LABEL_RANGE_LOWER_3	MPLS Label 3 Match Range Lower Value, page 223
0x139	EGR0_MPLS_LABEL_RANGE_UPPER_3	MPLS Label 3 Match Range Lower Value, page 224

Table 289 • **EGR0_IP1_NXT_PROTOCOL**

Address	Name	Details
0x1B0	EGR0_IP1_NXT_COMPARATOR	IP Next Comparator Control, page 224
0x1B1	EGR0_IP1_MODE	IP Comparator Mode, page 224
0x1B2	EGR0_IP1_PROT_MATCH_1	IP Match Set 1, page 225
0x1B3	EGR0_IP1_PROT_MATCH_2_UPPER	Upper Portion of Match 2, page 225
0x1B4	EGR0_IP1_PROT_MATCH_2_LOWER	Lower Portion of Match 2, page 225
0x1B5	EGR0_IP1_PROT_MASK_2_UPPER	Upper Portion of Match Mask 2, page 225
0x1B6	EGR0_IP1_PROT_MASK_2_LOWER	Lower Portion of Match Mask 2, page 226
0x1B7	EGR0_IP1_PROT_OFFSET_2	Match Offset 2, page 226
0x1B8	EGR0_IP1_UDP_CHKSUM_CFG	IP/UDP Checksum Control, page 226
0x1B9	EGR0_IP1_FRAME_SIG_CFG	IP Frame Signature Control, page 227

Table 290 • **EGR0_IP1_FLOW_CFG (8 instances)**

Address	Name	Details
0x1C0	EGR0_IP1_FLOW_ENA	IP Flow Enable, page 227
0x1C1	EGR0_IP1_FLOW_MATCH_UPPER	Upper Portion of the IP Flow Match, page 228
0x1C2	EGR0_IP1_FLOW_MATCH_UPPER_MID	Upper Mid Portion of the IP Flow Match, page 228

Table 290 • **EGR0_IP1_FLOW_CFG (8 instances)** ***(continued)***

Address	Name	Details
0x1C3	EGR0_IP1_FLOW_MATCH_LOWER_MID	Lower Mid Portion of the IP Flow Match, page 229
0x1C4	EGR0_IP1_FLOW_MATCH_LOWER	Lower Portion of the IP Flow Match, page 229
0x1C5	EGR0_IP1_FLOW_MASK_UPPER	IP Flow Match Mask, page 229
0x1C6	EGR0_IP1_FLOW_MASK_UPPER_MID	Upper Mid Portion Of IP Flow Mask, page 230
0x1C7	EGR0_IP1_FLOW_MASK_LOWER_MID	Lower Mid Portion of IP Flow Mask, page 230
0x1C8	EGR0_IP1_FLOW_MASK_LOWER	Lower Portion of IP Flow Mask, page 231

Table 291 • **EGR0_IP2_NXT_PROTOCOL**

Address	Name	Details
0x240	EGR0_IP2_NXT_COMPARATOR	IP Next Comparator Control, page 231
0x241	EGR0_IP2_MODE	IP Comparator Mode, page 231
0x242	EGR0_IP2_PROT_MATCH_1	IP Match Register Set 1, page 232
0x243	EGR0_IP2_PROT_MATCH_2_UPPER	Upper Portion of Match 2, page 232
0x244	EGR0_IP2_PROT_MATCH_2_LOWER	Lower Portion of Match 2, page 232
0x245	EGR0_IP2_PROT_MASK_2_UPPER	Upper Portion of Match Mask 2, page 232
0x246	EGR0_IP2_PROT_MASK_2_LOWER	Lower Portion of Match Mask 2, page 233
0x247	EGR0_IP2_PROT_OFFSET_2	Match Offset Register 2, page 233
0x248	EGR0_IP2_UDP_CHKSUM_CFG	IP/UDP Checksum Control, page 233
0x249	EGR0_IP2_FRAME_SIG_CFG	IP Frame Signature Control, page 234

Table 292 • **EGR0_IP2_FLOW_CFG (8 instances)**

Address	Name	Details
0x250	EGR0_IP2_FLOW_ENA	IP Flow Enable, page 234
0x251	EGR0_IP2_FLOW_MATCH_UPPER	Upper Portion of the IP Flow Match, page 235
0x252	EGR0_IP2_FLOW_MATCH_UPPER_MID	Upper Mid Portion of the IP Flow Match, page 235
0x253	EGR0_IP2_FLOW_MATCH_LOWER_MID	Lower Mid Portion of the IP Flow Match, page 236
0x254	EGR0_IP2_FLOW_MATCH_LOWER	Lower Portion of the IP Flow Match, page 236

Table 292 • **EGR0_IP2_FLOW_CFG (8 instances)** ***(continued)***

Address	Name	Details
0x255	EGR0_IP2_FLOW_MASK_UPPER	Upper Portion of the IP Flow Match Mask, page 236
0x256	EGR0_IP2_FLOW_MASK_UPPER_MID	Upper Mid Portion of the IP Flow Match Mask, page 237
0x257	EGR0_IP2_FLOW_MASK_LOWER_MID	Lower Mid Portion of the IP Flow Match Mask, page 237
0x258	EGR0_IP2_FLOW_MASK_LOWER	Lower Portion of the IP Flow Match Mask, page 238

Table 293 • **EGR0_PTP_FLOW (6 instances)**

Address	Name	Details
0x2D0	EGR0_PTP_FLOW_ENA	PTP/OAM Flow Enable, page 238
0x2D1	EGR0_PTP_FLOW_MATCH_UPPER	Upper Half of PTP/OAM Flow Match Field, page 239
0x2D2	EGR0_PTP_FLOW_MATCH_LOWER	Lower Half of PTP/OAM Flow Match Field, page 239
0x2D3	EGR0_PTP_FLOW_MASK_UPPER	Upper Half of PTP/OAM Flow Match Mask, page 239
0x2D4	EGR0_PTP_FLOW_MASK_LOWER	Lower Half of PTP/OAM Flow Match Mask, page 240
0x2D5	EGR0_PTP_DOMAIN_RANGE	PTP/OAM Range Match, page 240
0x2D6	EGR0_PTP_ACTION	PTP Action Control, page 240
0x2D7	EGR0_PTP_ACTION_2	PTP Action Control 2, page 242
0x2D8	EGR0_PTP_ZERO_FIELD_CTL	Zero Field Control, page 242

Table 294 • **EGR0_PTP_IP_CHKSUM_CTL**

Address	Name	Details
0x330	EGR0_PTP_IP_CKSUM_SEL	IP Checksum Block Select, page 242

Table 295 • **EGR0_FRAME_SIG_CFG**

Address	Register Name	Details
0x331	EGR0_FSB_CFG	Frame Signature Builder Mode Configuration, page 243
0x332	EGR0_FSB_MAP_REG_0	Frame Signature Builder Mapping 0, page 243
0x333	EGR0_FSB_MAP_REG_1	Frame Signature Builder Mapping 1, page 244
0x334	EGR0_FSB_MAP_REG_2	Frame Signature Builder Mapping 2, page 244

Table 295 • **EGR0_FRAME_SIG_CFG *(continued)***

Address	Register Name	Details
0x335	EGR0_FSB_MAP_REG_3	Frame Signature Builder Mapping 3, page 244

4.31 Egress0 Ethernet Next Protocol Configuration Registers

This section provides information about the Ethernet next protocol configuration registers.

4.31.1 Ethernet Next Protocol

Short Name: EGR0_ETH1_NXT_PROTOCOL

Address: 0x00

Table 296 • **Ethernet Next Protocol Register**

Bit	Name	Description	Access	Default
20:16	EGR0_ETH1_FRAME_SIG_OFFSET	Frame signature offset. Points to the start of the byte field in the Ethernet frame that will be used for the frame signature	R/W	0x00
2:0	EGR0_ETH1_NXT_COMPARATOR	Points to the next comparator block after this Ethernet block 0: Reserved 1: Ethernet comparator 2 2: IP/UDP/ACH comparator 1 3: IP/UDP/ACH comparator 2 4: MPLS comparator 5: PTP/OAM comparator 6,7: Reserved	R/W	0x0

4.31.2 VLAN TPID Configuration

Short Name: EGR0_ETH1_VLAN_TPID_CFG

Address: 0x01

Table 297 • **VLAN TPID Configuration Register**

Bit	Name	Description	Access	Default
31:16	EGR0_ETH1_VLAN_TPID_CFG	Configurable VLAN TPID (S or B-tag)	R/W	0x88A8

4.31.3 Ethernet Tag Mode

Short Name: EGR0_ETH1_TAG_MODE

Address: 0x02

Table 298 • **Ethernet Tag Mode Register**

Bit	Name	Description	Access	Default
0	EGR0_ETH1_PBB_ENA	This bit enables the presence of PBB. The I-tag match bits are programmed in the ETH1_VLAN_TAG_RANGE registers. The mask bits are programmed in the ETH1_VLAN_TAG2 registers. A B-tag if present is configured in the ETH1_VLAN_TAG1 registers. 0: PBB not enabled 1: Always expect PBB, last tag is always an I-tag	R/W	0x0

4.31.4 Ethertype Match

Short Name: EGR0_ETH1_ETYPE_MATCH

Address: 0x03

Table 299 • **Ethertype Match Register**

Bit	Name	Description	Access	Default
15:0	EGR0_ETH1_ETYPE_MATCH	If the Ethertype/length field is an Ethertype, then this register is compared against the value. If the field is a length, the length value is not checked.	R/W	0x0000

4.31.5

Instance offsets: 0x10 EGR0_ETH1_FLOW_CFG_0

0x20 EGR0_ETH1_FLOW_CFG_1

0x30 EGR0_ETH1_FLOW_CFG_2

0x40 EGR0_ETH1_FLOW_CFG_3

0x50 EGR0_ETH1_FLOW_CFG_4

0x60 EGR0_ETH1_FLOW_CFG_5

0x70 EGR0_ETH1_FLOW_CFG_6

0x80 EGR0_ETH1_FLOW_CFG_7

4.31.6 Ethernet Flow Enable

Short Name: EGR0_ETH1_FLOW_ENABLE

Addresses: 0x10 EGR0_ETH1_FLOW_CFG_0

0x20 EGR0_ETH1_FLOW_CFG_1

0x30 EGR0_ETH1_FLOW_CFG_2

0x40 EGR0_ETH1_FLOW_CFG_3

0x50 EGR0_ETH1_FLOW_CFG_4

0x60 EGR0_ETH1_FLOW_CFG_5

0x70 EGR0_ETH1_FLOW_CFG_6

0x80 EGR0_ETH1_FLOW_CFG_7

Table 300 • **Ethernet Flow Enable Register**

Bit	Name	Description	Access	Default
9:8	EGR0_ETH1_CHANNEL_MASK	0: Flow valid for channel 0 1: Flow valid for channel 1	R/W	0x3
0	EGR0_ETH1_FLOW_ENABLE	Flow enable 0: Flow is disabled 1: Flow is enabled	R/W	0x0

4.31.7 Ethernet Protocol Match Mode

Short Name: EGR0_ETH1_MATCH_MODE

Addresses: 0x11 EGR0_ETH1_FLOW_CFG_0

0x21 EGR0_ETH1_FLOW_CFG_1

0x31 EGR0_ETH1_FLOW_CFG_2

0x41 EGR0_ETH1_FLOW_CFG_3

0x51 EGR0_ETH1_FLOW_CFG_4

0x61 EGR0_ETH1_FLOW_CFG_5

0x71 EGR0_ETH1_FLOW_CFG_6

0x81 EGR0_ETH1_FLOW_CFG_7

Table 301 • **Ethernet Protocol Match Mode Register**

Bit	Name	Description	Access	Default
13:12	EGR0_ETH1_VLAN_TAG_MODE	0: VLAN range checking disabled 1: VLAN range checking on tag 1 2: VLAN range checking on tag 2 (not supported with PBB) 3: reserved	R/W	0x0
9	EGR0_ETH1_VLAN_TAG2_TYPE	This register is only used if ETH1_VLAN_VERIFY_ENA = 1 If PBB not enabled: 0: C tag (TPID of 0x8100) 1: S tag (match to CONF_VLAN_TPID) If PBB enabled: 0,1: I tag (use range registers)	R/W	0x1
8	EGR0_ETH1_VLAN_TAG1_TYPE	This register is only used if ETH1_VLAN_VERIFY_ENA = 1 0: C tag (TPID of 0x8100) 1: S or B tag (match to CONF_VLAN_TPID)	R/W	0x0
7:6	EGR0_ETH1_VLAN_TAGS	This register is only used if ETH1_VLAN_VERIFY_ENA = 1 0: No VLAN tags (not valid for PBB) 1: 1 VLAN tag (for PBB this would be the I-tag) 2: 2 VLAN tags (for PBB expect a B-tag and an I-tag) 3: Reserved	R/W	0x0

Table 301 • **Ethernet Protocol Match Mode Register *(continued)***

Bit	Name	Description	Access	Default
4	EGR0_ETH1_VLAN_VERIFY_ENA	0: Parse for VLAN tags, do not check values. For PBB the I-tag is always checked. 1: Verify configured VLAN tag configuration.	R/W	0x0
0	EGR0_ETH1_ETHERTYPE_MODE	When checking for presence of SNAP/LLC based upon ETH1_MATCH_MODE, this field indicates if SNAP & 3-byte LLC is expected to be present 0: Only Ethernet type II supported, no SNAP/LLC 1: Ethernet type II & Ethernet type I with SNAP/LLC, determine if SNAP/LLC is present or not. Type I always assumes that SNAP/LLC is present	R/W	0x0

4.31.8 Ethernet Address Match Part 1

Short Name: EGR0_ETH1_ADDR_MATCH_1

Addresses: 0x12 EGR0_ETH1_FLOW_CFG_0

0x22 EGR0_ETH1_FLOW_CFG_1

0x32 EGR0_ETH1_FLOW_CFG_2

0x42 EGR0_ETH1_FLOW_CFG_3

0x52 EGR0_ETH1_FLOW_CFG_4

0x62 EGR0_ETH1_FLOW_CFG_5

0x72 EGR0_ETH1_FLOW_CFG_6

0x82 EGR0_ETH1_FLOW_CFG_7

Table 302 • **Ethernet Address Match Part 1 Register**

Bit	Name	Description	Access	Default
31:0	EGR0_ETH1_ADDR_MATCH_1	First 32 bits of the address match value	R/W	0x00000000

4.31.9 Ethernet Address Match Part 2

Short Name: EGR0_ETH1_ADDR_MATCH_2

Addresses: 0x13 EGR0_ETH1_FLOW_CFG_0

0x23 EGR0_ETH1_FLOW_CFG_1

0x33 EGR0_ETH1_FLOW_CFG_2

0x43 EGR0_ETH1_FLOW_CFG_3

0x53 EGR0_ETH1_FLOW_CFG_4

0x63 EGR0_ETH1_FLOW_CFG_5

0x73 EGR0_ETH1_FLOW_CFG_6

0x83 EGR0_ETH1_FLOW_CFG_7

Table 303 • **Ethernet Address Match Part 2 Register**

Bit	Name	Description	Access	Default
22:20	EGR0_ETH1_ADDR_MATCH_MODE	Selects how the addresses are matched. Multiple bits can be set at once bit 0: Full 48-bit address match bit 1: Match any unicast address bit 2: Match any multicast address	R/W	0x1
17:16	EGR0_ETH1_ADDR_MATCH_SELECT	Selects which address to match 0: Match the Destination Address 1: Match the Source Address 2: Match either the Source of Destination Address 3: Reserved	R/W	0x0
15:0	EGR0_ETH1_ADDR_MATCH_2	Last 16 bits of the Ethernet address match field	R/W	0x0000

4.31.10 Ethernet VLAN Tag Range Match

Short Name: EGR0_ETH1_VLAN_TAG_RANGE_I_TAG

Addresses: 0x14 EGR0_ETH1_FLOW_CFG_0

0x24 EGR0_ETH1_FLOW_CFG_1

0x34 EGR0_ETH1_FLOW_CFG_2

0x44 EGR0_ETH1_FLOW_CFG_3

0x54 EGR0_ETH1_FLOW_CFG_4

0x64 EGR0_ETH1_FLOW_CFG_5

0x74 EGR0_ETH1_FLOW_CFG_6

0x84 EGR0_ETH1_FLOW_CFG_7

Table 304 • **Ethernet VLAN Tag Range Match Register**

Bit	Name	Description	Access	Default
27:16	EGR0_ETH1_VLAN_TAG_RANGE_UPPER	If PBB mode is not enabled, then this register contains the upper range of the VLAN tag range match. If PBB mode is enabled, then this register contains the upper 12 bits of the I-tag	R/W	0xFFF
11:0	EGR0_ETH1_VLAN_TAG_RANGE_LOWER	If PBB mode is not enabled, then this register contains the lower range of the VLAN tag range match. If PBB mode is enabled, then this register contains the lower 12 bits of the I-tag	R/W	0x000

4.31.11 VLAN Tag 1 Match/Mask

Short Name: EGR0_ETH1_VLAN_TAG1

Addresses: 0x15 EGR0_ETH1_FLOW_CFG_0

0x25 EGR0_ETH1_FLOW_CFG_1

0x35 EGR0_ETH1_FLOW_CFG_2

0x45 EGR0_ETH1_FLOW_CFG_3

0x55 EGR0_ETH1_FLOW_CFG_4

0x65 EGR0_ETH1_FLOW_CFG_5

0x75 EGR0_ETH1_FLOW_CFG_6

0x85 EGR0_ETH1_FLOW_CFG_7

Table 305 • **VLAN Tag 1 Match/Mask Register**

Bit	Name	Description	Access	Default
27:16	EGR0_ETH1_VLAN_TAG1_MASK	Mask value for VLAN tag 1	R/W	0xFFF
11:0	EGR0_ETH1_VLAN_TAG1_MATCH	Match value for the first VLAN tag	R/W	0x000

4.31.12 Match/Mask For VLAN Tag 2 or I-Tag Match

Short Name: EGR0_ETH1_VLAN_TAG2_I_TAG

Addresses: 0x16 EGR0_ETH1_FLOW_CFG_0

0x26 EGR0_ETH1_FLOW_CFG_1

0x36 EGR0_ETH1_FLOW_CFG_2

0x46 EGR0_ETH1_FLOW_CFG_3

0x56 EGR0_ETH1_FLOW_CFG_4

0x66 EGR0_ETH1_FLOW_CFG_5

0x76 EGR0_ETH1_FLOW_CFG_6

0x86 EGR0_ETH1_FLOW_CFG_7

Table 306 • **Match/Mask For VLAN Tag 2 or I-Tag Match Register**

Bit	Name	Description	Access	Default
27:16	EGR0_ETH1_VLAN_TAG2_MASK	When PBB is not enabled, the mask field for VLAN tag 2 When PBB is enabled, the upper 12 bits of the I-tag mask	R/W	0xFFF
11:0	EGR0_ETH1_VLAN_TAG2_MATCH	When PBB is not enabled, the match field for VLAN Tag 2 When PBB is enabled, the lower 12 bits of the I-tag mask field	R/W	0x000

4.31.13 Ethernet Next Protocol

Short Name: EGR0_ETH2_NXT_PROTOCOL

Address: 0x90

Table 307 • **Ethernet Next Protocol Register**

Bit	Name	Description	Access	Default
20:16	EGR0_ETH2_FRAME_SIG_OFFSET	Frame signature offset. Points to the start of the byte field in the Ethernet frame that will be used for the frame signature	R/W	0x00

Table 307 • **Ethernet Next Protocol Register *(continued)***

Bit	Name	Description	Access	Default
2:0	EGR0_ETH2_NXT_COMPARATOR	Points to the next comparator block after this Ethernet block. If this comparator block is not used, this field must be set to 0. 0: Comparator block not used 1: Ethernet comparator 2 2: IP/UDP/ACH comparator 1 3: IP/UDP/ACH comparator 2 4: MPLS comparator 5: PTP/OAM comparator 6,7: Reserved	R/W	0x0

4.31.14 VLAN TPID Configuration

Short Name: EGR0_ETH2_VLAN_TPID_CFG

Address: 0x91

Table 308 • **VLAN TPID Configuration Register**

Bit	Name	Description	Access	Default
31:16	EGR0_ETH2_VLAN_TPID_CFG	Configurable S-tag TPID	R/W	0x88A8

4.31.15 Ethertype Match

Short Name: EGR0_ETH2_ETYPE_MATCH

Address: 0x92

Table 309 • **Ethertype Match Register**

Bit	Name	Description	Access	Default
15:0	EGR0_ETH2_ETYPE_MATCH	If the Ethertype/length field is an Ethertype, then this register is compared against the value. If the field is a length, the length value is not checked.	R/W	0x0000

4.31.16

Instance offsets: 0xA0 EGR0_ETH2_FLOW_CFG_0

0xB0 EGR0_ETH2_FLOW_CFG_1

0xC0 EGR0_ETH2_FLOW_CFG_2

0xD0 EGR0_ETH2_FLOW_CFG_3

0xE0 EGR0_ETH2_FLOW_CFG_4

0xF0 EGR0_ETH2_FLOW_CFG_5

0x100 EGR0_ETH2_FLOW_CFG_6

0x110 EGR0_ETH2_FLOW_CFG_7

4.31.17 Ethernet Flow Enable

Short Name: EGR0_ETH2_FLOW_ENABLE

Addresses: 0xA0 EGR0_ETH2_FLOW_CFG_0

0xB0 EGR0_ETH2_FLOW_CFG_1

0xC0 EGR0_ETH2_FLOW_CFG_2

0xD0 EGR0_ETH2_FLOW_CFG_3

0xE0 EGR0_ETH2_FLOW_CFG_4

0xF0 EGR0_ETH2_FLOW_CFG_5

0x100 EGR0_ETH2_FLOW_CFG_6

0x110 EGR0_ETH2_FLOW_CFG_7

Table 310 • **Ethernet Flow Enable Register**

Bit	Name	Description	Access	Default
9:8	EGR0_ETH2_CHANNEL_MASK	bit 0: Flow valid for channel 0 bit 1: Flow valid for channel 1	R/W	0x3
0	EGR0_ETH2_FLOW_ENABLE	Flow enable. If this comparator block is not used, all flow enable bits must be set to 0. 0: Flow is disabled 1: Flow is enabled	R/W	0x0

4.31.18 Ethernet Protocol Match Mode

Short Name: EGR0_ETH2_MATCH_MODE

Addresses: 0xA1 EGR0_ETH2_FLOW_CFG_0

0xB1 EGR0_ETH2_FLOW_CFG_1

0xC1 EGR0_ETH2_FLOW_CFG_2

0xD1 EGR0_ETH2_FLOW_CFG_3

0xE1 EGR0_ETH2_FLOW_CFG_4

0xF1 EGR0_ETH2_FLOW_CFG_5

0x101 EGR0_ETH2_FLOW_CFG_6

0x111 EGR0_ETH2_FLOW_CFG_7

Table 311 • **Ethernet Protocol Match Mode Register**

Bit	Name	Description	Access	Default
13:12	EGR0_ETH2_VLAN_TAG_MODE	0: VLAN range checking disabled 1: VLAN range checking on tag 1 2: VLAN range checking on tag 2 (not supported with PBB) 3: reserved	R/W	0x0
9	EGR0_ETH2_VLAN_TAG2_TYPE	This register is only used if ETH1_VLAN_VERIFY_ENA = 1 0: C tag (TPID of 0x8100) 1: S tag (match to CONF_VLAN_TPID)	R/W	0x1
8	EGR0_ETH2_VLAN_TAG1_TYPE	This register is only used if ETH1_VLAN_VERIFY_ENA = 1 0: C tag (TPID of 0x8100) 1: S or B tag (match to CONF_VLAN_TPID)	R/W	0x0

Table 311 • **Ethernet Protocol Match Mode Register *(continued)***

Bit	Name	Description	Access	Default
7:6	EGR0_ETH2_VLAN_TAGS	This register is only used if ETH2_VLAN_VERIFY_ENA = 1 0: No VLAN tags 1: 1 VLAN tag 2: 2 VLAN tags 3: Reserved	R/W	0x0
4	EGR0_ETH2_VLAN_VERIFY_ENA	0: Parse for VLAN tags, do not check values. 1: Verify configured VLAN tag configuration.	R/W	0x0
0	EGR0_ETH2_ETHERTYPE_MODE	When checking for presence of SNAP/LLC based upon ETH1_MATCH_MODE, this field indicates if SNAP & 3-byte LLC is expected to be present 0: Only Ethernet type II supported, no SNAP/LLC 1: Ethernet type II & Ethernet type I with SNAP/LLC, determine if SNAP/LLC is present or not. Type I always assumes that SNAP/LLC is present	R/W	0x0

4.31.19 Ethernet Address Match Part 1

Short Name: EGR0_ETH2_ADDR_MATCH_1

Addresses: 0xA2 EGR0_ETH2_FLOW_CFG_0

0xB2 EGR0_ETH2_FLOW_CFG_1

0xC2 EGR0_ETH2_FLOW_CFG_2

0xD2 EGR0_ETH2_FLOW_CFG_3

0xE2 EGR0_ETH2_FLOW_CFG_4

0xF2 EGR0_ETH2_FLOW_CFG_5

0x102 EGR0_ETH2_FLOW_CFG_6

0x112 EGR0_ETH2_FLOW_CFG_7

Table 312 • **Ethernet Address Match Part 1 Register**

Bit	Name	Description	Access	Default
31:0	EGR0_ETH2_ADDR_MATCH_1	First 32 bits of the address match value	R/W	0x00000000

4.31.20 Ethernet Address Match Part 2

Short Name: EGR0_ETH2_ADDR_MATCH_2

Addresses: 0xA3 EGR0_ETH2_FLOW_CFG_0

0xB3 EGR0_ETH2_FLOW_CFG_1

0xC3 EGR0_ETH2_FLOW_CFG_2

0xD3 EGR0_ETH2_FLOW_CFG_3

0xE3 EGR0_ETH2_FLOW_CFG_4

0xF3 EGR0_ETH2_FLOW_CFG_5

0x103 EGR0_ETH2_FLOW_CFG_6

0x113 EGR0_ETH2_FLOW_CFG_7

Table 313 • **Ethernet Address Match Part 2 Register**

Bit	Name	Description	Access	Default
22:20	EGR0_ETH2_ADDR_MATCH_MODE	Selects how the addresses are matched. Multiple bits can be set at once bit 0: Full 48-bit address match bit 1: Match any unicast address bit 2: Match any multicast address	R/W	0x1
17:16	EGR0_ETH2_ADDR_MATCH_SELECT	Selects which address to match 0: Match the Destination Address 1: Match the Source Address 2: Match either the Source of Destination Address 3: Reserved	R/W	0x0
15:0	EGR0_ETH2_ADDR_MATCH_2	Last 16 bits of the Ethernet address match field	R/W	0x0000

4.31.21 Ethernet VLAN Tag Range Match

Short Name: EGR0_ETH2_VLAN_TAG_RANGE_I_TAG

Addresses: 0xA4 EGR0_ETH2_FLOW_CFG_0

0xB4 EGR0_ETH2_FLOW_CFG_1

0xC4 EGR0_ETH2_FLOW_CFG_2

0xD4 EGR0_ETH2_FLOW_CFG_3

0xE4 EGR0_ETH2_FLOW_CFG_4

0xF4 EGR0_ETH2_FLOW_CFG_5

0x104 EGR0_ETH2_FLOW_CFG_6

0x114 EGR0_ETH2_FLOW_CFG_7

Table 314 • **Ethernet VLAN Tag Range Match Register**

Bit	Name	Description	Access	Default
27:16	EGR0_ETH2_VLAN_TAG_RANGE_UPPER	This register contains the upper range of the VLAN tag range match.	R/W	0xFFF
11:0	EGR0_ETH2_VLAN_TAG_RANGE_LOWER	This register contains the lower range of the VLAN tag range match.	R/W	0x000

4.31.22 VLAN Tag 1 Match/Mask

Short Name: EGR0_ETH2_VLAN_TAG1

Addresses: 0xA5 EGR0_ETH2_FLOW_CFG_0

0xB5 EGR0_ETH2_FLOW_CFG_1

0xC5 EGR0_ETH2_FLOW_CFG_2

0xD5 EGR0_ETH2_FLOW_CFG_3

0xE5 EGR0_ETH2_FLOW_CFG_4

0xF5 EGR0_ETH2_FLOW_CFG_5

0x105 EGR0_ETH2_FLOW_CFG_6

0x115 EGR0_ETH2_FLOW_CFG_7

Table 315 • **VLAN Tag 1 Match/Mask Register**

Bit	Name	Description	Access	Default
27:16	EGR0_ETH2_VLAN_TAG1_MASK	Mask value for VLAN tag 1	R/W	0xFFF
11:0	EGR0_ETH2_VLAN_TAG1_MATCH	Match value for the first VLAN tag	R/W	0x000

4.31.23 Match/Mask For VLAN Tag 2 or I-Tag Match

Short Name: EGR0_ETH2_VLAN_TAG2_I_TAG

Addresses: 0xA6 EGR0_ETH2_FLOW_CFG_0

0xB6 EGR0_ETH2_FLOW_CFG_1

0xC6 EGR0_ETH2_FLOW_CFG_2

0xD6 EGR0_ETH2_FLOW_CFG_3

0xE6 EGR0_ETH2_FLOW_CFG_4

0xF6 EGR0_ETH2_FLOW_CFG_5

0x106 EGR0_ETH2_FLOW_CFG_6

0x116 EGR0_ETH2_FLOW_CFG_7

Table 316 • **Match/Mask For VLAN Tag 2 or I-Tag Match Register**

Bit	Name	Description	Access	Default
27:16	EGR0_ETH2_VLAN_TAG2_MASK	Mask field for VLAN tag 2	R/W	0xFFF
11:0	EGR0_ETH2_VLAN_TAG2_MATCH	Match field for VLAN Tag 2	R/W	0x000

4.32 Egress0 MPLS Next Protocol Registers

This section provides information about the MPLS next protocol registers.

4.32.1 MPLS Next Protocol Comparator

Short Name: EGR0_MPLS_NXT_COMPARATOR

Address: 0x120

Table 317 • **MPLS Next Protocol Comparator Register**

Bit	Name	Description	Access	Default
16	EGR0_MPLS_CTL_WORD	Indicates the presence of a control word after the last label. The first 4 bits of the control word are always 0. 0: There is no control word after the last label 1: There is a control word after the last label	R/W	0x0

Table 317 • **MPLS Next Protocol Comparator Register *(continued)***

Bit	Name	Description	Access	Default
2:0	EGR0_MPLS_NXT_COMPARATOR	Points to the next comparator stage. If this comparator block is not used, this field must be set to 0. 0: Comparator block not used. 1: Ethernet comparator 2 2: IP/UDP/ACH comparator 1 3: IP/UDP/ACH comparator 2 4: Reserved 5: PTP/OAM comparator 6,7: Reserved	R/W	0x0

4.32.2

Instance offsets: 0x130 EGR0_MPLS_FLOW_CFG_0

0x140 EGR0_MPLS_FLOW_CFG_1

0x150 EGR0_MPLS_FLOW_CFG_2

0x160 EGR0_MPLS_FLOW_CFG_3

0x170 EGR0_MPLS_FLOW_CFG_4

0x180 EGR0_MPLS_FLOW_CFG_5

0x190 EGR0_MPLS_FLOW_CFG_6

0x1A0 EGR0_MPLS_FLOW_CFG_7

4.32.3 MPLS Flow Control

Short Name: EGR0_MPLS_FLOW_CONTROL

Addresses: 0x130 EGR0_MPLS_FLOW_CFG_0

0x140 EGR0_MPLS_FLOW_CFG_1

0x150 EGR0_MPLS_FLOW_CFG_2

0x160 EGR0_MPLS_FLOW_CFG_3

0x170 EGR0_MPLS_FLOW_CFG_4

0x180 EGR0_MPLS_FLOW_CFG_5

0x190 EGR0_MPLS_FLOW_CFG_6

0x1A0 EGR0_MPLS_FLOW_CFG_7

Table 318 • **MPLS Flow Control Register**

Bit	Name	Description	Access	Default
25:24	EGR0_MPLS_CHANNEL_MASK	0: Flow valid for channel 0 1: Flow valid for channel 1	R/W	0x3

Table 318 • **MPLS Flow Control Register *(continued)***

Bit	Name	Description	Access	Default
19:16	EGR0_MPLS_STACK_DEPTH	Defines the allowable stack depths for searches. The direction that the stack is referenced is determined by the setting of MPLS_REF_PNT. For each bit set, The following table maps bits to stack depths: 0: stack allowed to be 1 label deep 1: stack allowed to be 2 labels deep 2: stack allowed to be 3 labels deep 3: stack allowed to be 4 labels deep	R/W	0x0
4	EGR0_MPLS_REF_PNT	Defines the search direction for label matching 0: All searching is performed starting from the top of the stack 1: All searching is performed from the end of the stack	R/W	0x0
0	EGR0_MPLS_FLOW_ENA	Flow enable. If this comparator block is not used, all flow enable bits must be set to 0. 0: Flow is disabled 1: Flow is enabled	R/W	0x0

4.32.4 MPLS Label 0 Match Range Lower Value

Short Name: EGR0_MPLS_LABEL_RANGE_LOWER_0

Addresses: 0x132 EGR0_MPLS_FLOW_CFG_0

0x142 EGR0_MPLS_FLOW_CFG_1

0x152 EGR0_MPLS_FLOW_CFG_2

0x162 EGR0_MPLS_FLOW_CFG_3

0x172 EGR0_MPLS_FLOW_CFG_4

0x182 EGR0_MPLS_FLOW_CFG_5

0x192 EGR0_MPLS_FLOW_CFG_6

0x1A2 EGR0_MPLS_FLOW_CFG_7

Table 319 • **MPLS Label 0 Match Range Lower Value Register**

Bit	Name	Description	Access	Default
19:0	EGR0_MPLS_LABEL_RANGE_LOWER_0	Lower value for label 0 match range	R/W	0x00000

4.32.5 MPLS Label 0 Match Range Upper Value

Short Name: EGR0_MPLS_LABEL_RANGE_UPPER_0

Addresses: 0x133 EGR0_MPLS_FLOW_CFG_0

0x143 EGR0_MPLS_FLOW_CFG_1

0x153 EGR0_MPLS_FLOW_CFG_2

0x163 EGR0_MPLS_FLOW_CFG_3

0x173 EGR0_MPLS_FLOW_CFG_4

0x183 EGR0_MPLS_FLOW_CFG_5

0x193 EGR0_MPLS_FLOW_CFG_6

0x1A3 EGR0_MPLS_FLOW_CFG_7

Table 320 • **MPLS Label 0 Match Range Upper Value Register**

Bit	Name	Description	Access	Default
19:0	EGR0_MPLS_LABEL_RANGE_UPPER_0	Upper value for label 0 match range	R/W	0xFFFFF

4.32.6 MPLS Label 1 Match Range Lower Value

Short Name: EGR0_MPLS_LABEL_RANGE_LOWER_1

Addresses: 0x134 EGR0_MPLS_FLOW_CFG_0

0x144 EGR0_MPLS_FLOW_CFG_1

0x154 EGR0_MPLS_FLOW_CFG_2

0x164 EGR0_MPLS_FLOW_CFG_3

0x174 EGR0_MPLS_FLOW_CFG_4

0x184 EGR0_MPLS_FLOW_CFG_5

0x194 EGR0_MPLS_FLOW_CFG_6

0x1A4 EGR0_MPLS_FLOW_CFG_7

Table 321 • **MPLS Label 1 Match Range Lower Value Register**

Bit	Name	Description	Access	Default
19:0	EGR0_MPLS_LABEL_RANGE_LOWER_1	Lower value for label 1 match range	R/W	0x00000

4.32.7 MPLS Label 1 Match Range Lower Value

Short Name: EGR0_MPLS_LABEL_RANGE_UPPER_1

Addresses: 0x135 EGR0_MPLS_FLOW_CFG_0

0x145 EGR0_MPLS_FLOW_CFG_1

0x155 EGR0_MPLS_FLOW_CFG_2

0x165 EGR0_MPLS_FLOW_CFG_3

0x175 EGR0_MPLS_FLOW_CFG_4

0x185 EGR0_MPLS_FLOW_CFG_5

0x195 EGR0_MPLS_FLOW_CFG_6

0x1A5 EGR0_MPLS_FLOW_CFG_7

Table 322 • **MPLS Label 1 Match Range Lower Value Register**

Bit	Name	Description	Access	Default
19:0	EGR0_MPLS_LABEL_RANGE_UPPER_1	Upper value for label 1 match range	R/W	0xFFFFF

4.32.8 MPLS Label 2 Match Range Lower Value

Short Name: EGR0_MPLS_LABEL_RANGE_LOWER_2

Addresses: 0x136 EGR0_MPLS_FLOW_CFG_0

0x146 EGR0_MPLS_FLOW_CFG_1

0x156 EGR0_MPLS_FLOW_CFG_2

0x166 EGR0_MPLS_FLOW_CFG_3

0x176 EGR0_MPLS_FLOW_CFG_4

0x186 EGR0_MPLS_FLOW_CFG_5

0x196 EGR0_MPLS_FLOW_CFG_6

0x1A6 EGR0_MPLS_FLOW_CFG_7

Table 323 • **MPLS Label 2 Match Range Lower Value Register**

Bit	Name	Description	Access	Default
19:0	EGR0_MPLS_LABEL_RANGE_LOWER_2	Lower value for label 2 match range	R/W	0x00000

4.32.9 MPLS Label 2 Match Range Lower Value

Short Name: EGR0_MPLS_LABEL_RANGE_UPPER_2

Addresses: 0x137 EGR0_MPLS_FLOW_CFG_0

0x147 EGR0_MPLS_FLOW_CFG_1

0x157 EGR0_MPLS_FLOW_CFG_2

0x167 EGR0_MPLS_FLOW_CFG_3

0x177 EGR0_MPLS_FLOW_CFG_4

0x187 EGR0_MPLS_FLOW_CFG_5

0x197 EGR0_MPLS_FLOW_CFG_6

0x1A7 EGR0_MPLS_FLOW_CFG_7

Table 324 • **MPLS Label 2 Match Range Lower Value Register**

Bit	Name	Description	Access	Default
19:0	EGR0_MPLS_LABEL_RANGE_UPPER_2	Upper value for label 2 match range	R/W	0xFFFFF

4.32.10 MPLS Label 3 Match Range Lower Value

Short Name: EGR0_MPLS_LABEL_RANGE_LOWER_3

Addresses: 0x138 EGR0_MPLS_FLOW_CFG_0

0x148 EGR0_MPLS_FLOW_CFG_1

0x158 EGR0_MPLS_FLOW_CFG_2

0x168 EGR0_MPLS_FLOW_CFG_3

0x178 EGR0_MPLS_FLOW_CFG_4

0x188 EGR0_MPLS_FLOW_CFG_5

0x198 EGR0_MPLS_FLOW_CFG_6

0x1A8 EGR0_MPLS_FLOW_CFG_7

Table 325 • **MPLS Label 3 Match Range Lower Value Register**

Bit	Name	Description	Access	Default
19:0	EGR0_MPLS_LABEL_RANGE_LOWER_3	Lower value for label 3 match range	R/W	0x00000

4.32.11 MPLS Label 3 Match Range Lower Value

Short Name: EGR0_MPLS_LABEL_RANGE_UPPER_3

Addresses: 0x139 EGR0_MPLS_FLOW_CFG_0

0x149 EGR0_MPLS_FLOW_CFG_1

0x159 EGR0_MPLS_FLOW_CFG_2

0x169 EGR0_MPLS_FLOW_CFG_3

0x179 EGR0_MPLS_FLOW_CFG_4

0x189 EGR0_MPLS_FLOW_CFG_5

0x199 EGR0_MPLS_FLOW_CFG_6

0x1A9 EGR0_MPLS_FLOW_CFG_7

Table 326 • **MPLS Label 3 Match Range Lower Value Register**

Bit	Name	Description	Access	Default
19:0	EGR0_MPLS_LABEL_RANGE_UPPER_3	Upper value for label 3 match range	R/W	0xFFFFF

4.32.12 IP Next Comparator Control

Short Name: EGR0_IP1_NXT_COMPARATOR

Address: 0x1B0

Table 327 • **IP Next Comparator Control Register**

Bit	Name	Description	Access	Default
15:8	EGR0_IP1_NXT_PROTOCOL	Number of bytes in this header, points to the beginning of the next protocol	R/W	0x00
2:0	EGR0_IP1_NXT_COMPARATOR	Points to the next comparator stage. If this comparator block is not used, this field must be set to 0. 0: Comparator block not used 1: Reserved 2: Reserved 3: IP/UDP/ACH comparator 2 4: Reserved 5: PTP/OAM comparator 6,7: Reserved	R/W	0x0

4.32.13 IP Comparator Mode

Short Name: EGR0_IP1_MODE

Address: 0x1B1

Table 328 • **IP Comparator Mode Register**

Bit	Name	Description	Access	Default
12:8	EGR0_IP1_FLOW_OFFSET	Points to the source address field in the IP frame. Use 12 for IPv4 and 8 for IPv6	R/W	0x0C
1:0	EGR0_IP1_MODE	0: IPv4 1: IPv6 2: Other protocol, 32-bit address match 3: Other protocol, 128-bit address match	R/W	0x0

4.32.14 IP Match Set 1

Short Name: EGR0_IP1_PROT_MATCH_1

Address: 0x1B2

Table 329 • **IP Match Set 1 Register**

Bit	Name	Description	Access	Default
20:16	EGR0_IP1_PROT_OFFSET_1	Points to the start of this match field relative to the first byte of this protocol	R/W	0x00
15:8	EGR0_IP1_PROT_MASK_1	Mask field for IP_PROT_MATCH_1	R/W	0x00
7:0	EGR0_IP1_PROT_MATCH_1	8-bit match field	R/W	0x00

4.32.15 Upper Portion of Match 2

Short Name: EGR0_IP1_PROT_MATCH_2_UPPER

Address: 0x1B3

Table 330 • **Upper Portion of Match 2 Register**

Bit	Name	Description	Access	Default
31:0	EGR0_IP1_PROT_MATCH_2_UPPER	64-bit match register for advancing to the next protocol, upper portion	R/W	0x00000000

4.32.16 Lower Portion of Match 2

Short Name: EGR0_IP1_PROT_MATCH_2_LOWER

Address: 0x1B4

Table 331 • **Lower Portion of Match 2 Register**

Bit	Name	Description	Access	Default
31:0	EGR0_IP1_PROT_MATCH_2_LOWER	64-bit match register for advancing to the next protocol, lower portion	R/W	0x00000000

4.32.17 Upper Portion of Match Mask 2

Short Name: EGR0_IP1_PROT_MASK_2_UPPER

Address: 0x1B5

Table 332 • **Upper Portion of Match Mask 2 Register**

Bit	Name	Description	Access	Default
31:0	EGR0_IP1_PROT_MASK_2_UPPER		R/W	0x00000000

4.32.18 Lower Portion of Match Mask 2

Short Name: EGR0_IP1_PROT_MASK_2_LOWER

Address: 0x1B6

Table 333 • **Lower Portion of Match Mask 2 Register**

Bit	Name	Description	Access	Default
31:0	EGR0_IP1_PROT_MASK_2_LOWER		R/W	0x00000000

4.32.19 Match Offset 2

Short Name: EGR0_IP1_PROT_OFFSET_2

Address: 0x1B7

Table 334 • **Match Offset 2 Register**

Bit	Name	Description	Access	Default
6:0	EGR0_IP1_PROT_OFFSET_2	Points to the start of match field 2 relative to the first byte of this protocol	R/W	0x00

4.32.20 IP/UDP Checksum Control

Short Name: EGR0_IP1_UDP_CHKSUM_CFG

Address: 0x1B8

Table 335 • **IP/UDP Checksum Control Register**

Bit	Name	Description	Access	Default
15:8	EGR0_IP1_UDP_CHKSUM_OFFSET	Pointer to the IP/UDP checksum field FOR IPv4 frames or to the pad bytes of a IPv6/UDP frame. For IPv4, it points to the bytes that will be cleared. For IPv6, it points to the bytes that will be updated to fix the CRC	R/W	0x00
5:4	EGR0_IP1_UDP_CHKSUM_WIDTH	Specifies the length of the checksum field in bytes	R/W	0x2
1	EGR0_IP1_UDP_CHKSUM_UPDATE_ENA	This bit and IP_UDP_CHKSUM_CLEAR_ENA CANNOT be set together. 1: Update the pad bytes at the end of the frame 0: No pad byte field update	R/W	0x0

Table 335 • **IP/UDP Checksum Control Register *(continued)***

Bit	Name	Description	Access	Default
0	EGR0_IP1_UDP_CHKSUM_CLEAR_ENA	This bit and IP_UDP_CHKSUM_UPDATE_ENA CANNOT be set together. 1: Clear the UDP checksum field in an IPv4 frame 0: Do not clear the checksum	R/W	0x0

4.32.21 IP Frame Signature Control

Short Name: EGR0_IP1_FRAME_SIG_CFG

Address: 0x1B9

Table 336 • **IP Frame Signature Control Register**

Bit	Name	Description	Access	Default
4:0	EGR0_IP1_FRAME_SIG_OFFSET	Pointer to the start of the field that will be used for the frame signature. Position is relative to the first header byte of this IP protocol. Only even values are allowed.	R/W	0x00

4.32.22

Instance offsets: 0x1C0 EGR0_IP1_FLOW_CFG_0

0x1D0 EGR0_IP1_FLOW_CFG_1

0x1E0 EGR0_IP1_FLOW_CFG_2

0x1F0 EGR0_IP1_FLOW_CFG_3

0x200 EGR0_IP1_FLOW_CFG_4

0x210 EGR0_IP1_FLOW_CFG_5

0x220 EGR0_IP1_FLOW_CFG_6

0x230 EGR0_IP1_FLOW_CFG_7

4.32.23 IP Flow Enable

Short Name: EGR0_IP1_FLOW_ENA

Addresses: 0x1C0 EGR0_IP1_FLOW_CFG_0

0x1D0 EGR0_IP1_FLOW_CFG_1

0x1E0 EGR0_IP1_FLOW_CFG_2

0x1F0 EGR0_IP1_FLOW_CFG_3

0x200 EGR0_IP1_FLOW_CFG_4

0x210 EGR0_IP1_FLOW_CFG_5

0x220 EGR0_IP1_FLOW_CFG_6

0x230 EGR0_IP1_FLOW_CFG_7

Table 337 • **IP Flow Enabler Register**

Bit	Name	Description	Access	Default
9:8	EGR0_IP1_FLOW_MATCH_MODE	0: Match on source address 1: Match on destination address 2: Match on either source or destination address 3: reserved	R/W	0x0
5:4	EGR0_IP1_CHANNEL_MASK	0: Flow valid for channel 0 1: Flow valid for channel 1	R/W	0x3
0	EGR0_IP1_FLOW_ENA	Flow enable. If this comparator block is not used, all flow enable bits must be set to 0. 1: This flow is enabled 0: This flow is not enabled	R/W	0x0

4.32.24 Upper Portion of the IP Flow Match

Short Name: EGR0_IP1_FLOW_MATCH_UPPER

Addresses: 0x1C1 EGR0_IP1_FLOW_CFG_0

0x1D1 EGR0_IP1_FLOW_CFG_1

0x1E1 EGR0_IP1_FLOW_CFG_2

0x1F1 EGR0_IP1_FLOW_CFG_3

0x201 EGR0_IP1_FLOW_CFG_4

0x211 EGR0_IP1_FLOW_CFG_5

0x221 EGR0_IP1_FLOW_CFG_6

0x231 EGR0_IP1_FLOW_CFG_7

Table 338 • **Upper Portion of the IP Flow Match Register**

Bit	Name	Description	Access	Default
31:0	EGR0_IP1_FLOW_MATCH_UPPER	Match field for either the entire 32-bit selected address for IPv4 or the upper 32 bits of the selected address for IPv6	R/W	0x00000000

4.32.25 Upper Mid Portion of the IP Flow Match

Short Name: EGR0_IP1_FLOW_MATCH_UPPER_MID

Addresses: 0x1C2 EGR0_IP1_FLOW_CFG_0

0x1D2 EGR0_IP1_FLOW_CFG_1

0x1E2 EGR0_IP1_FLOW_CFG_2

0x1F2 EGR0_IP1_FLOW_CFG_3

0x202 EGR0_IP1_FLOW_CFG_4

0x212 EGR0_IP1_FLOW_CFG_5

0x222 EGR0_IP1_FLOW_CFG_6

0x232 EGR0_IP1_FLOW_CFG_7

Table 339 • **Upper Mid Portion of the IP Flow Match Register**

Bit	Name	Description	Access	Default
31:0	EGR0_IP1_FLOW_MATCH_UPPER_MID	Match bits for the upper middle 32 bits of the IPv6 address	R/W	0x00000000

4.32.26 Lower Mid Portion of the IP Flow Match

Short Name: EGR0_IP1_FLOW_MATCH_LOWER_MID

Addresses: 0x1C3 EGR0_IP1_FLOW_CFG_0

0x1D3 EGR0_IP1_FLOW_CFG_1

0x1E3 EGR0_IP1_FLOW_CFG_2

0x1F3 EGR0_IP1_FLOW_CFG_3

0x203 EGR0_IP1_FLOW_CFG_4

0x213 EGR0_IP1_FLOW_CFG_5

0x223 EGR0_IP1_FLOW_CFG_6

0x233 EGR0_IP1_FLOW_CFG_7

Table 340 • **Lower Mid Portion of the IP Flow Match Register**

Bit	Name	Description	Access	Default
31:0	EGR0_IP1_FLOW_MATCH_LOWER_MID	Match bits for the lower middle 32 bits of the IPv6 address	R/W	0x00000000

4.32.27 Lower Portion of the IP Flow Match

Short Name: EGR0_IP1_FLOW_MATCH_LOWER

Addresses: 0x1C4 EGR0_IP1_FLOW_CFG_0

0x1D4 EGR0_IP1_FLOW_CFG_1

0x1E4 EGR0_IP1_FLOW_CFG_2

0x1F4 EGR0_IP1_FLOW_CFG_3

0x204 EGR0_IP1_FLOW_CFG_4

0x214 EGR0_IP1_FLOW_CFG_5

0x224 EGR0_IP1_FLOW_CFG_6

0x234 EGR0_IP1_FLOW_CFG_7

Table 341 • **Lower Portion of the IP Flow Match Register**

Bit	Name	Description	Access	Default
31:0	EGR0_IP1_FLOW_MATCH_LOWER	Match bits for the lower 32 bits of the IPv6 address	R/W	0x00000000

4.32.28 IP Flow Match Mask

Short Name: EGR0_IP1_FLOW_MASK_UPPER

Addresses: 0x1C5 EGR0_IP1_FLOW_CFG_0

0x1D5 EGR0_IP1_FLOW_CFG_1

0x1E5 EGR0_IP1_FLOW_CFG_2

0x1F5 EGR0_IP1_FLOW_CFG_3

0x205 EGR0_IP1_FLOW_CFG_4

0x215 EGR0_IP1_FLOW_CFG_5

0x225 EGR0_IP1_FLOW_CFG_6

0x235 EGR0_IP1_FLOW_CFG_7

Table 342 • **IP Flow Match Mask Register**

Bit	Name	Description	Access	Default
31:0	EGR0_IP1_FLOW_MASK_UPPER	This is the address mask for the IP address.	R/W	0x00000000

4.32.29 Upper Mid Portion Of IP Flow Mask

Short Name: EGR0_IP1_FLOW_MASK_UPPER_MID

Addresses: 0x1C6 EGR0_IP1_FLOW_CFG_0

0x1D6 EGR0_IP1_FLOW_CFG_1

0x1E6 EGR0_IP1_FLOW_CFG_2

0x1F6 EGR0_IP1_FLOW_CFG_3

0x206 EGR0_IP1_FLOW_CFG_4

0x216 EGR0_IP1_FLOW_CFG_5

0x226 EGR0_IP1_FLOW_CFG_6

0x236 EGR0_IP1_FLOW_CFG_7

Table 343 • **Upper Mid Portion of IP Flow Mask Register**

Bit	Name	Description	Access	Default
31:0	EGR0_IP1_FLOW_MASK_UPPER_MID	These bits must be all 0 for IPv4 and any 32-bit address match mode	R/W	0x00000000

4.32.30 Lower Mid Portion of IP Flow Mask

Short Name: EGR0_IP1_FLOW_MASK_LOWER_MID

Addresses: 0x1C7 EGR0_IP1_FLOW_CFG_0

0x1D7 EGR0_IP1_FLOW_CFG_1

0x1E7 EGR0_IP1_FLOW_CFG_2

0x1F7 EGR0_IP1_FLOW_CFG_3

0x207 EGR0_IP1_FLOW_CFG_4

0x217 EGR0_IP1_FLOW_CFG_5

0x227 EGR0_IP1_FLOW_CFG_6

0x237 EGR0_IP1_FLOW_CFG_7

Table 344 • **Lower Mid Portion of IP Flow Mask Register**

Bit	Name	Description	Access	Default
31:0	EGR0_IP1_FLOW_MASK_LOWER_MID	These bits must be all 0 for IPv4 and any 32-bit address match mode	R/W	0x00000000

4.32.31 Lower Portion of IP Flow Mask

Short Name: EGR0_IP1_FLOW_MASK_LOWER

Addresses: 0x1C8 EGR0_IP1_FLOW_CFG_0

0x1D8 EGR0_IP1_FLOW_CFG_1

0x1E8 EGR0_IP1_FLOW_CFG_2

0x1F8 EGR0_IP1_FLOW_CFG_3

0x208 EGR0_IP1_FLOW_CFG_4

0x218 EGR0_IP1_FLOW_CFG_5

0x228 EGR0_IP1_FLOW_CFG_6

0x238 EGR0_IP1_FLOW_CFG_7

Table 345 • **Lower Portion of IP Flow Mask Register**

Bit	Name	Description	Access	Default
31:0	EGR0_IP1_FLOW_MASK_LOWER	These bits must be all 0 for IPv4 and any 32-bit address match mode	R/W	0x00000000

4.32.32 IP Next Comparator Control

Short Name: EGR0_IP2_NXT_COMPARATOR

Address: 0x240

Table 346 • **IP Next Comparator Control Register**

Bit	Name	Description	Access	Default
15:8	EGR0_IP2_NXT_PROTOCOL	Number of bytes in this header, points to the beginning of the next protocol	R/W	0x00
2:0	EGR0_IP2_NXT_COMPARATOR	Points to the next comparator stage. If this comparator block is not used, this field must be set to 0. 0: Comparator block not used 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: PTP/OAM comparator 6,7: Reserved	R/W	0x0

4.32.33 IP Comparator Mode

Short Name: EGR0_IP2_MODE

Address: 0x241

Table 347 • **IP Comparator Mode Register**

Bit	Name	Description	Access	Default
12:8	EGR0_IP2_FLOW_OFFSET	Points to the source address field in the IP frame. Use 12 for IPv4 and 8 for IPv6	R/W	0x0C
1:0	EGR0_IP2_MODE	0: IPv4 1: IPv6 2: Other protocol, 32-bit address match 3: Other protocol, 128-bit address match	R/W	0x0

4.32.34 IP Match Register Set 1

Short Name: EGR0_IP2_PROT_MATCH_1

Address: 0x242

Table 348 • **IP Match Register Set 1 Register**

Bit	Name	Description	Access	Default
20:16	EGR0_IP2_PROT_OFFSET_1	Points to the start of this match field relative to the first byte of this protocol	R/W	0x00
15:8	EGR0_IP2_PROT_MASK_1	Mask field for IP_PROT_MATCH_1	R/W	0x00
7:0	EGR0_IP2_PROT_MATCH_1	8-bit match field	R/W	0x00

4.32.35 Upper Portion of Match 2

Short Name: EGR0_IP2_PROT_MATCH_2_UPPER

Address: 0x243

Table 349 • **Upper Portion of Match 2 Register**

Bit	Name	Description	Access	Default
31:0	EGR0_IP2_PROT_MATCH_2_UPPER	64-bit match register for advancing to the next protocol, upper portion	R/W	0x00000000

4.32.36 Lower Portion of Match 2

Short Name: EGR0_IP2_PROT_MATCH_2_LOWER

Address: 0x244

Table 350 • **Lower Portion of Match 2 Register**

Bit	Name	Description	Access	Default
31:0	EGR0_IP2_PROT_MATCH_2_LOWER	64-bit match register for advancing to the next protocol, lower portion	R/W	0x00000000

4.32.37 Upper Portion of Match Mask 2

Short Name: EGR0_IP2_PROT_MASK_2_UPPER

Address: 0x245

Table 351 • **Upper Portion of Match Mask 2 Register**

Bit	Name	Description	Access	Default
31:0	EGR0_IP2_PROT_MASK_2_UPPER		R/W	0x00000000

4.32.38 Lower Portion of Match Mask 2

Short Name: EGR0_IP2_PROT_MASK_2_LOWER

Address: 0x246

Table 352 • **Lower Portion of Match Mask 2 Register**

Bit	Name	Description	Access	Default
31:0	EGR0_IP2_PROT_MASK_2_LOWER		R/W	0x00000000

4.32.39 Match Offset Register 2

Short Name: EGR0_IP2_PROT_OFFSET_2

Address: 0x247

Table 353 • **Match Offset 2 Register**

Bit	Name	Description	Access	Default
6:0	EGR0_IP2_PROT_OFFSET_2	Points to the start of match field 2 relative to the first byte of this protocol	R/W	0x00

4.32.40 IP/UDP Checksum Control

Short Name: EGR0_IP2_UDP_CHKSUM_CFG

Address: 0x248

Table 354 • **IP/UDP Checksum Control Register**

Bit	Name	Description	Access	Default
15:8	EGR0_IP2_UDP_CHKSUM_OFFSET	Pointer to the IP/UDP checksum field FOR IPv4 frames or to the pad bytes of a IPv6/UDP frame. For IPv4, it points to the bytes that will be cleared. For IPv6, it points to the bytes that will be updated to fix the CRC	R/W	0x00
5:4	EGR0_IP2_UDP_CHKSUM_WIDTH	Specifies the length of the checksum field in bytes	R/W	0x2
1	EGR0_IP2_UDP_CHKSUM_UPDATE_ENA	This bit and IP_UDP_CHKSUM_CLEAR_ENA CANNOT be set together. 1: Update the pad bytes at the end of the frame 0: No pad byte field update	R/W	0x0

Table 354 • **IP/UDP Checksum Control Register *(continued)***

Bit	Name	Description	Access	Default
0	EGR0_IP2_UDP_CHKSUM_CLEAR_ENA	This bit and IP_UDP_CHKSUM_UPDATE_ENA CANNOT be set together. 1: Clear the UDP checksum field in an IPv4 frame 0: Do not clear the checksum	R/W	0x0

4.32.41 IP Frame Signature Control

Short Name: EGR0_IP2_FRAME_SIG_CFG

Address: 0x249

Table 355 • **IP Frame Signature Control Register**

Bit	Name	Description	Access	Default
4:0	EGR0_IP2_FRAME_SIG_OFFSET	Pointer to the start of the field that will be used for the frame signature. Position is relative to the first header byte of this IP protocol. Only even values are allowed.	R/W	0x00

4.32.42

Instance offsets: 0x250 EGR0_IP2_FLOW_CFG_0

0x260 EGR0_IP2_FLOW_CFG_1

0x270 EGR0_IP2_FLOW_CFG_2

0x280 EGR0_IP2_FLOW_CFG_3

0x290 EGR0_IP2_FLOW_CFG_4

0x2A0 EGR0_IP2_FLOW_CFG_5

0x2B0 EGR0_IP2_FLOW_CFG_6

0x2C0 EGR0_IP2_FLOW_CFG_7

4.32.43 IP Flow Enable

Short Name: EGR0_IP2_FLOW_ENA

Addresses: 0x250 EGR0_IP2_FLOW_CFG_0

0x260 EGR0_IP2_FLOW_CFG_1

0x270 EGR0_IP2_FLOW_CFG_2

0x280 EGR0_IP2_FLOW_CFG_3

0x290 EGR0_IP2_FLOW_CFG_4

0x2A0 EGR0_IP2_FLOW_CFG_5

0x2B0 EGR0_IP2_FLOW_CFG_6

0x2C0 EGR0_IP2_FLOW_CFG_7

Table 356 • **IP Flow Enable Register**

Bit	Name	Description	Access	Default
9:8	EGR0_IP2_FLOW_MATCH_MODE	0: Match on source address 1: Match on destination address 2: Match on either source or destination address 3: reserved	R/W	0x0
5:4	EGR0_IP2_CHANNEL_MASK	bit 0: Flow valid for channel 0 bit 1: Flow valid for channel 1	R/W	0x3
0	EGR0_IP2_FLOW_ENA	Flow enable. If this comparator block is not used, all flow enable bits must be set to 0. 1: This flow is enabled 0: This flow is not enabled	R/W	0x0

4.32.44 Upper Portion of the IP Flow Match

Short Name: EGR0_IP2_FLOW_MATCH_UPPER

Addresses: 0x251 EGR0_IP2_FLOW_CFG_0

0x261 EGR0_IP2_FLOW_CFG_1

0x271 EGR0_IP2_FLOW_CFG_2

0x281 EGR0_IP2_FLOW_CFG_3

0x291 EGR0_IP2_FLOW_CFG_4

0x2A1 EGR0_IP2_FLOW_CFG_5

0x2B1 EGR0_IP2_FLOW_CFG_6

0x2C1 EGR0_IP2_FLOW_CFG_7

Table 357 • **Upper Portion of the IP Flow Match Register**

Bit	Name	Description	Access	Default
31:0	EGR0_IP2_FLOW_MATCH_UPPER	Match field for either the entire 32-bit selected address for IPv4 or the upper 32 bits of the selected address for IPv6	R/W	0x00000000

4.32.45 Upper Mid Portion of the IP Flow Match

Short Name: EGR0_IP2_FLOW_MATCH_UPPER_MID

Addresses: 0x252 EGR0_IP2_FLOW_CFG_0

0x262 EGR0_IP2_FLOW_CFG_1

0x272 EGR0_IP2_FLOW_CFG_2

0x282 EGR0_IP2_FLOW_CFG_3

0x292 EGR0_IP2_FLOW_CFG_4

0x2A2 EGR0_IP2_FLOW_CFG_5

0x2B2 EGR0_IP2_FLOW_CFG_6

0x2C2 EGR0_IP2_FLOW_CFG_7

Table 358 • **Upper Mid Portion of the IP Flow Match Register**

Bit	Name	Description	Access	Default
31:0	EGR0_IP2_FLOW_MATCH_UPPER_MID	Match bits for the upper middle 32 bits of the IPv6 address	R/W	0x00000000

4.32.46 Lower Mid Portion of the IP Flow Match

Short Name: EGR0_IP2_FLOW_MATCH_LOWER_MID

Addresses: 0x253 EGR0_IP2_FLOW_CFG_0

0x263 EGR0_IP2_FLOW_CFG_1

0x273 EGR0_IP2_FLOW_CFG_2

0x283 EGR0_IP2_FLOW_CFG_3

0x293 EGR0_IP2_FLOW_CFG_4

0x2A3 EGR0_IP2_FLOW_CFG_5

0x2B3 EGR0_IP2_FLOW_CFG_6

0x2C3 EGR0_IP2_FLOW_CFG_7

Table 359 • **Lower Mid Portion of the IP Flow Match Register**

Bit	Name	Description	Access	Default
31:0	EGR0_IP2_FLOW_MATCH_LOWER_MID	Match bits for the lower middle 32 bits of the IPv6 address	R/W	0x00000000

4.32.47 Lower Portion of the IP Flow Match

Short Name: EGR0_IP2_FLOW_MATCH_LOWER

Addresses: 0x254 EGR0_IP2_FLOW_CFG_0

0x264 EGR0_IP2_FLOW_CFG_1

0x274 EGR0_IP2_FLOW_CFG_2

0x284 EGR0_IP2_FLOW_CFG_3

0x294 EGR0_IP2_FLOW_CFG_4

0x2A4 EGR0_IP2_FLOW_CFG_5

0x2B4 EGR0_IP2_FLOW_CFG_6

0x2C4 EGR0_IP2_FLOW_CFG_7

Table 360 • **Lower Portion of the IP Flow Match Register**

Bit	Name	Description	Access	Default
31:0	EGR0_IP2_FLOW_MATCH_LOWER	Match bits for the lower 32 bits of the IPv6 address	R/W	0x00000000

4.32.48 Upper Portion of the IP Flow Match Mask

Short Name: EGR0_IP2_FLOW_MASK_UPPER

Addresses: 0x255 EGR0_IP2_FLOW_CFG_0

0x265 EGR0_IP2_FLOW_CFG_1

0x275 EGR0_IP2_FLOW_CFG_2

0x285 EGR0_IP2_FLOW_CFG_3

0x295 EGR0_IP2_FLOW_CFG_4

0x2A5 EGR0_IP2_FLOW_CFG_5

0x2B5 EGR0_IP2_FLOW_CFG_6

0x2C5 EGR0_IP2_FLOW_CFG_7

Table 361 • **Upper Portion of the IP Flow Match Mask Register**

Bit	Name	Description	Access	Default
31:0	EGR0_IP2_FLOW_MASK_UPPER	This is the address mask for the IP address.	R/W	0x00000000

4.32.49 Upper Mid Portion of the IP Flow Match Mask

Short Name: EGR0_IP2_FLOW_MASK_UPPER_MID

Addresses: 0x256 EGR0_IP2_FLOW_CFG_0

0x266 EGR0_IP2_FLOW_CFG_1

0x276 EGR0_IP2_FLOW_CFG_2

0x286 EGR0_IP2_FLOW_CFG_3

0x296 EGR0_IP2_FLOW_CFG_4

0x2A6 EGR0_IP2_FLOW_CFG_5

0x2B6 EGR0_IP2_FLOW_CFG_6

0x2C6 EGR0_IP2_FLOW_CFG_7

Table 362 • **Upper Mid Portion of the IP Flow Match Mask Register**

Bit	Name	Description	Access	Default
31:0	EGR0_IP2_FLOW_MASK_UPPER_MID	These bits must be all 0 for IPv4 and any 32-bit address match mode	R/W	0x00000000

4.32.50 Lower Mid Portion of the IP Flow Match Mask

Short Name: EGR0_IP2_FLOW_MASK_LOWER_MID

Addresses: 0x257 EGR0_IP2_FLOW_CFG_0

0x267 EGR0_IP2_FLOW_CFG_1

0x277 EGR0_IP2_FLOW_CFG_2

0x287 EGR0_IP2_FLOW_CFG_3

0x297 EGR0_IP2_FLOW_CFG_4

0x2A7 EGR0_IP2_FLOW_CFG_5

0x2B7 EGR0_IP2_FLOW_CFG_6

0x2C7 EGR0_IP2_FLOW_CFG_7

Table 363 • **Lower Mid Portion of the IP Flow Match Mask Register**

Bit	Name	Description	Access	Default
31:0	EGR0_IP2_FLOW_MASK_LOWER_MID	These bits must be all 0 for IPv4 and any 32-bit address match mode	R/W	0x00000000

4.32.51 Lower Portion of the IP Flow Match Mask

Short Name: EGR0_IP2_FLOW_MASK_LOWER

Addresses: 0x258 EGR0_IP2_FLOW_CFG_0

0x268 EGR0_IP2_FLOW_CFG_1

0x278 EGR0_IP2_FLOW_CFG_2

0x288 EGR0_IP2_FLOW_CFG_3

0x298 EGR0_IP2_FLOW_CFG_4

0x2A8 EGR0_IP2_FLOW_CFG_5

0x2B8 EGR0_IP2_FLOW_CFG_6

0x2C8 EGR0_IP2_FLOW_CFG_7

Table 364 • **Lower Portion of the IP Flow Match Mask Register**

Bit	Name	Description	Access	Default
31:0	EGR0_IP2_FLOW_MASK_LOWER	These bits must be all 0 for IPv4 and any 32-bit address match mode	R/W	0x00000000

4.32.52

Instance offsets: 0x2D0 EGR0_PTP_FLOW_0

0x2E0 EGR0_PTP_FLOW_1

0x2F0 EGR0_PTP_FLOW_2

0x300 EGR0_PTP_FLOW_3

0x310 EGR0_PTP_FLOW_4

0x320 EGR0_PTP_FLOW_5

4.32.53 PTP/OAM Flow Enable

Short Name: EGR0_PTP_FLOW_ENA

Addresses: 0x2D0 EGR0_PTP_FLOW_0

0x2E0 EGR0_PTP_FLOW_1

0x2F0 EGR0_PTP_FLOW_2

0x300 EGR0_PTP_FLOW_3

0x310 EGR0_PTP_FLOW_4

0x320 EGR0_PTP_FLOW_5

Table 365 • **PTP/OAM Flow Enable Register**

Bit	Name	Description	Access	Default
5:4	EGR0_PTP_CHANNEL_MASK	0: Flow valid for channel 0 1: Flow valid for channel 1	R/W	0x3
0	EGR0_PTP_FLOW_ENA		R/W	0x0

4.32.54 Upper Half of PTP/OAM Flow Match Field

Short Name: EGR0_PTP_FLOW_MATCH_UPPER

Addresses: 0x2D1 EGR0_PTP_FLOW_0

0x2E1 EGR0_PTP_FLOW_1

0x2F1 EGR0_PTP_FLOW_2

0x301 EGR0_PTP_FLOW_3

0x311 EGR0_PTP_FLOW_4

0x321 EGR0_PTP_FLOW_5

Table 366 • **Upper Half of PTP/OAM Flow Match Field Register**

Bit	Name	Description	Access	Default
31:0	EGR0_PTP_FLOW_MATCH_UPPER		R/W	0x00000000

4.32.55 Lower Half of PTP/OAM Flow Match Field

Short Name: EGR0_PTP_FLOW_MATCH_LOWER

Addresses: 0x2D2 EGR0_PTP_FLOW_0

0x2E2 EGR0_PTP_FLOW_1

0x2F2 EGR0_PTP_FLOW_2

0x302 EGR0_PTP_FLOW_3

0x312 EGR0_PTP_FLOW_4

0x322 EGR0_PTP_FLOW_5

Table 367 • **Lower Half of PTP/OAM Flow Match Field Register**

Bit	Name	Description	Access	Default
31:0	EGR0_PTP_FLOW_MATCH_LOWER		R/W	0x00000000

4.32.56 Upper Half of PTP/OAM Flow Match Mask

Short Name: EGR0_PTP_FLOW_MASK_UPPER

Addresses: 0x2D3 EGR0_PTP_FLOW_0

0x2E3 EGR0_PTP_FLOW_1

0x2F3 EGR0_PTP_FLOW_2

0x303 EGR0_PTP_FLOW_3

0x313 EGR0_PTP_FLOW_4

0x323 EGR0_PTP_FLOW_5

Table 368 • **Upper Half of PTP/OAM Flow Match Mask Register**

Bit	Name	Description	Access	Default
31:0	EGR0_PTP_FLOW_MASK_UPPER		R/W	0x00000000

4.32.57 Lower Half of PTP/OAM Flow Match Mask

Short Name: EGR0_PTP_FLOW_MASK_LOWER

Addresses: 0x2D4 EGR0_PTP_FLOW_0

0x2E4 EGR0_PTP_FLOW_1

0x2F4 EGR0_PTP_FLOW_2

0x304 EGR0_PTP_FLOW_3

0x314 EGR0_PTP_FLOW_4

0x324 EGR0_PTP_FLOW_5

Table 369 • **Lower Half of PTP/OAM Flow Match Mask Register**

Bit	Name	Description	Access	Default
31:0	EGR0_PTP_FLOW_MASK_LOWER		R/W	0x00000000

4.32.58 PTP/OAM Range Match

Short Name: EGR0_PTP_DOMAIN_RANGE

Addresses: 0x2D5 EGR0_PTP_FLOW_0

0x2E5 EGR0_PTP_FLOW_1

0x2F5 EGR0_PTP_FLOW_2

0x305 EGR0_PTP_FLOW_3

0x315 EGR0_PTP_FLOW_4

0x325 EGR0_PTP_FLOW_5

Table 370 • **PTP/OAM Range Match Register**

Bit	Name	Description	Access	Default
28:24	EGR0_PTP_DOMAIN_RANGE_OFFSET		R/W	0x00
23:16	EGR0_PTP_DOMAIN_RANGE_UPPER		R/W	0xFF
15:8	EGR0_PTP_DOMAIN_RANGE_LOWER		R/W	0x00
0	EGR0_PTP_DOMAIN_RANGE_ENA		R/W	0x0

4.32.59 PTP Action Control

Short Name: EGR0_PTP_ACTION

Addresses: 0x2D6 EGR0_PTP_FLOW_0

0x2E6 EGR0_PTP_FLOW_1

0x2F6 EGR0_PTP_FLOW_2

0x306 EGR0_PTP_FLOW_3

0x316 EGR0_PTP_FLOW_4

0x326 EGR0_PTP_FLOW_5

Table 371 • **PTP Action Control Register**

Bit	Name	Description	Access	Default
28	EGR0_PTP_MOD_FRAME_STAT_UPDATE	1: Tell the Rewriter to update the value of the Modified Frame Status bit 0: Do not update the bit	R/W	0x0
26:24	EGR0_PTP_MOD_FRAME_BYTE_OFFSET	Indicates the position relative to the start of the PTP frame in bytes where the Modified_Frame_Status bit resides	R/W	0x0
21	EGR0_PTP_SUB_DELAY_ASYM_ENA	1: Signal the Timestamp block to subtract the asymmetry delay 0: Do not signal the Timestamp block to subtract the asymmetry delay	R/W	0x0
20	EGR0_PTP_ADD_DELAY_ASYM_ENA	1: Signal the Timestamp block to add the asymmetry delay 0: Do not signal the Timestamp block to add the asymmetry delay	R/W	0x0
15:10	EGR0_PTP_TIME_STRG_FIELD_OFFSET	Points to the reserved 32-bit field where the Rx timestamp is saved. The location is relative to the first byte of the PTP/OAM header.	R/W	0x00
9:5	EGR0_PTP_CORR_FIELD_OFFSET	Points to the location of the correction field for updating the timestamp. Location is relative to the first byte of the PTP/OAM header. **Note:** If this flow is being used to match OAM frames, set this register to 4	R/W	0x00
4	EGR0_PTP_SAVE_LOCAL_TIME	1: Save the local time to the Timestamp FIFO 0: Do not save the time to the Timestamp FIFO	R/W	0x0
3:0	EGR0_PTP_COMMAND	0: NoP 1: SUB 2: SUB_P2P 3: ADD 4: SUB_ADD 5: WRITE_1588 6: WRITE_P2P (deprecated) 7: WRITE_NS 8: WRITE_NS_P2P	R/W	0x0

4.32.60 PTP Action Control 2

Short Name: EGR0_PTP_ACTION_2

Addresses: 0x2D7 EGR0_PTP_FLOW_0

0x2E7 EGR0_PTP_FLOW_1

0x2F7 EGR0_PTP_FLOW_2

0x307 EGR0_PTP_FLOW_3

0x317 EGR0_PTP_FLOW_4

0x327 EGR0_PTP_FLOW_5

Table 372 • **PTP Action Control 2 Register**

Bit	Name	Description	Access	Default
23:16	EGR0_PTP_NEW_CF_LOC	Location of the new correction field relative to the PTP header start. Only even values are allowed.	R/W	0x00
15:8	EGR0_PTP_REWRITE_OFFSET	Byte offset relative to the start of the PTP frame where the ingress timestamp value can be read.	R/W	0x00
3:0	EGR0_PTP_REWRITE_BYTES	Number of bytes in the PTP or OAM frame that must be modified by the Rewriter for the timestamp	R/W	0x0

4.32.61 Zero Field Control

Short Name: EGR0_PTP_ZERO_FIELD_CTL

Addresses: 0x2D8 EGR0_PTP_FLOW_0

0x2E8 EGR0_PTP_FLOW_1

0x2F8 EGR0_PTP_FLOW_2

0x308 EGR0_PTP_FLOW_3

0x318 EGR0_PTP_FLOW_4

0x328 EGR0_PTP_FLOW_5

Table 373 • **Zero Field Control Register**

Bit	Name	Description	Access	Default
13:8	EGR0_PTP_ZERO_FIELD_OFFSET	Points to a location in the PTP/OAM frame relative to the start of the PTP header that will be zeroed if this function is enabled	R/W	0x00
3:0	EGR0_PTP_ZERO_FIELD_BYTE_CNT	The number of bytes to be zeroed. If this field is 0, then this function is not enabled.	R/W	0x0

4.33 Egress0 IP Checksum Field Control Registers

This section provides information about the IP checksum field control registers.

4.33.1 IP Checksum Block Select

Short Name: EGR0_PTP_IP_CKSUM_SEL

Address: 0x330

Table 374 • **IP Checksum Block Select Register**

Bit	Name	Description	Access	Default
0	EGR0_PTP_IP_CHKSUM_SEL	0: Use the IP checksum controls from IP comparator 1 1: Use the IP checksum controls from IP comparator 2	R/W	0x0

4.34 Egress0 Frame Signature Builder Configuration Registers

4.34.1 Frame Signature Builder Mode Configuration

Short Name: EGR0_FSB_CFG

Address: 0x331

Table 375 • **Frame Signature Builder Mode Configuration Register**

Bit	Name	Description	Access	Default
1:0	EGR0_FSB_ADR_SEL	0: Use the address from Ethernet block 1 1: Use the address from Ethernet block 2 2: Use the address from IP block 1 3: Use the address from IP block 2	R/W	0x0

4.34.2 Frame Signature Builder Mapping 0

Short Name: EGR0_FSB_MAP_REG_0

Address: 0x332

This register selects bytes to pack into the frame signature vector. The frame signature vector is 16 bytes long. The following table lists the source bytes; all other select values are reserved.

Table 376 • **Source Bytes**

Select	Source	Select	Source	Select	Source	Select	Source
0	PTP hdr byte 31	1	PTP hdr byte 30	2	PTP hdr byte 29	3	PTP hdr byte 28
4	PTP hdr byte 27	5	PTP hdr byte 26	6	PTP hdr byte 25	7	PTP hdr byte 24
8	PTP hdr byte 23	9	PTP hdr byte 22	10	PTP hdr byte 21	11	PTP hdr byte 20
12	PTP hdr byte 19	13	PTP hdr byte 18	14	PTP hdr byte 17	15	PTP hdr byte 16
16	PTP hdr byte 15	17	PTP hdr byte 14	18	PTP hdr byte 13	19	PTP hdr byte 12
20	PTP hdr byte 11	21	PTP hdr byte 10	22	PTP hdr byte 9	23	PTP hdr byte 8
24	PTP hdr byte 6	25	PTP hdr byte 4	26	PTP hdr byte 0	27	reserved
28	address byte 0	29	address byte 1	30	address byte 2	31	address byte 3
32	address byte 4	33	address byte 5	34	address byte 6	35	address byte 7

Table 377 • **Frame Signature Builder Mapping 0 Register**

Bit	Name	Description	Access	Default
29:24	EGR0_FSB_MAP_4	Frame signature byte 4 select	R/W	0x04
23:18	EGR0_FSB_MAP_3	Frame signature byte 3 select	R/W	0x03
17:12	EGR0_FSB_MAP_2	Frame signature byte 2 select	R/W	0x02
11:6	EGR0_FSB_MAP_1	Frame signature byte 1 select	R/W	0x01
5:0	EGR0_FSB_MAP_0	Frame signature byte 0 select	R/W	0x00

4.34.3 Frame Signature Builder Mapping 1

Short Name: EGR0_FSB_MAP_REG_1

Address: 0x333

Table 378 • **Frame Signature Builder Mapping 1 Register**

Bit	Name	Description	Access	Default
29:24	EGR0_FSB_MAP_9	Frame signature byte 9 select	R/W	0x09
23:18	EGR0_FSB_MAP_8	Frame signature byte 8 select	R/W	0x08
17:12	EGR0_FSB_MAP_7	Frame signature byte 7 select	R/W	0x07
11:6	EGR0_FSB_MAP_6	Frame signature byte 6 select	R/W	0x06
5:0	EGR0_FSB_MAP_5	Frame signature byte 5 select	R/W	0x05

4.34.4 Frame Signature Builder Mapping 2

Short Name: EGR0_FSB_MAP_REG_2

Address: 0x334

Table 379 • **Frame Signature Builder Mapping 2 Register**

Bit	Name	Description	Access	Default
29:24	EGR0_FSB_MAP_14	Frame signature byte 14 select	R/W	0x0E
23:18	EGR0_FSB_MAP_13	Frame signature byte 13 select	R/W	0x0D
17:12	EGR0_FSB_MAP_12	Frame signature byte 12 select	R/W	0x0C
11:6	EGR0_FSB_MAP_11	Frame signature byte 11 select	R/W	0x0B
5:0	EGR0_FSB_MAP_10	Frame signature byte 10 select	R/W	0x0A

4.34.5 Frame Signature Builder Mapping 3

Short Name: EGR0_FSB_MAP_REG_3

Address: 0x335

Table 380 • **Frame Signature Builder Mapping 3 Register**

Bit	Name	Description	Access	Default
5:0	EGR0_FSB_MAP_15	Frame signature byte 15 select	R/W	0x0F

4.35 Ingress2 Analyzer Engine Configuration Registers

This section lists the register overviews for the analyzer engine configuration ingress2 registers.

Note: The analyzer engine configuration registers are not initialized to the default values during chip reset. Software must configure these registers to their default value.

Note: For more information about accessing the 1588 IP registers, see Accessing 1588 IP Registers, page 72.

Table 381 • **INGR2_ETH1_NXT_PROTOCOL_A**

Address	Name	Details
0x00	INGR2_ETH1_NXT_PROTOCOL_A	Ethernet Next Protocol, page 247
0x01	INGR2_ETH1_VLAN_TPID_CFG_A	VLAN TPID Configuration, page 247
0x02	INGR2_ETH1_TAG_MODE_A	Ethernet Tag Mode, page 248
0x03	INGR2_ETH1_ETYPE_MATCH_A	Ethertype Match, page 248

Table 382 • **INGR2_ETH1_NXT_PROTOCOL_B**

Address	Name	Details
0x10	INGR2_ETH1_NXT_PROTOCOL_B	Ethernet Next Protocol, page 248
0x11	INGR2_ETH1_VLAN_TPID_CFG_B	VLAN TPID Configuration B, page 249
0x12	INGR2_ETH1_TAG_MODE_B	Ethernet Tag Mode, page 249
0x13	INGR2_ETH1_ETYPE_MATCH_B	Ethertype Match, page 249

Table 383 • **INGR2_ETH1_FLOW_CFG (8 instances)**

Address	Name	Details
0x20	INGR2_ETH1_FLOW_ENABLE	Ethernet Flow Enable, page 250
0x21	INGR2_ETH1_MATCH_MODE	Ethernet Protocol Match Mode, page 250
0x22	INGR2_ETH1_ADDR_MATCH_1	Ethernet Address Match Part 1, page 251
0x23	INGR2_ETH1_ADDR_MATCH_2	Ethernet Address Match Part 2, page 252
0x24	INGR2_ETH1_VLAN_TAG_RANGE_I_TAG	Ethernet VLAN Tag Range Match, page 252
0x25	INGR2_ETH1_VLAN_TAG1	VLAN Tag 1 Match/Mask, page 253
0x26	INGR2_ETH1_VLAN_TAG2_I_TAG	Match/Mask For VLAN Tag 2 or I-Tag Match, page 253

Table 384 • **INGR2_ETH2_NXT_PROTOCOL_A**

Address	Name	Details
0xA0	INGR2_ETH2_NXT_PROTOCOL_A	Ethernet Next Protocol, page 254

Table 384 • **INGR2_ETH2_NXT_PROTOCOL_A *(continued)***

Address	Name	Details
0xA1	INGR2_ETH2_VLAN_TPID_CFG_A	VLAN TPID Configuration, page 254
0xA2	INGR2_ETH2_ETYPE_MATCH_A	Ethertype Match, page 254

Table 385 • **INGR2_ETH2_FLOW_CFG (8 instances)**

Address	Name	Details
0xC0	INGR2_ETH2_FLOW_ENABLE	Ethernet Flow Enable, page 255
0xC1	INGR2_ETH2_MATCH_MODE	Ethernet Protocol Match Mode, page 255
0xC2	INGR2_ETH2_ADDR_MATCH_1	Ethernet Address Match Part 1, page 256
0xC3	INGR2_ETH2_ADDR_MATCH_2	Ethernet Address Match Part 2, page 257
0xC4	INGR2_ETH2_VLAN_TAG_RANGE_I_TAG	Ethernet VLAN Tag Range Match, page 257
0xC5	INGR2_ETH2_VLAN_TAG1	VLAN Tag 1 Match/Mask, page 258
0xC6	INGR2_ETH2_VLAN_TAG2_I_TAG	Match/Mask For VLAN Tag 2 or I-Tag Match, page 258

Table 386 • **INGR2_MPLS_NXT_COMPARATOR_A**

Address	Name	Details
0x140	INGR2_MPLS_NXT_COMPARATOR_A	MPLS Next Protocol Comparator, page 259

Table 387 • **INGR2_MPLS_FLOW_CFG (8 instances)**

Address	Name	Details
0x160	INGR2_MPLS_FLOW_CONTROL	MPLS Flow Control, page 259
0x161	INGR2_MPLS_LABEL_RANGE_LOWER_0	MPLS Label 0 Match Range Lower Value, page 260
0x162	INGR2_MPLS_LABEL_RANGE_UPPER_0	MPLS Label 0 Match Range Lower Value, page 260
0x163	INGR2_MPLS_LABEL_RANGE_LOWER_1	MPLS Label 1 Match Range Lower Value, page 261
0x164	INGR2_MPLS_LABEL_RANGE_UPPER_1	MPLS Label 1 Match Range Lower Value, page 261
0x165	INGR2_MPLS_LABEL_RANGE_LOWER_2	MPLS Label 2 Match Range Lower Value, page 262
0x166	INGR2_MPLS_LABEL_RANGE_UPPER_2	MPLS Label 2 Match Range Lower Value, page 262
0x167	INGR2_MPLS_LABEL_RANGE_LOWER_3	MPLS Label 3 Match Range Lower Value, page 262

Table 387 • **INGR2_MPLS_FLOW_CFG (8 instances) *(continued)***

Address	Name	Details
0x168	INGR2_MPLS_LABEL_RANGE_UPPER_3	MPLS Label 3 Match Range Lower Value, page 263

Table 388 • **INGR2_PTP_FLOW (6 instances)**

Address	Register Name	Details
0x1E0	INGR2_PTP_FLOW_ENA	PTP/OAM Flow Enable, page 263
0x1E1	INGR2_PTP_FLOW_MATCH_UPPER	Upper Half of PTP/OAM Flow Match Field, page 264
0x1E2	INGR2_PTP_FLOW_MATCH_LOWER	Lower Half of PTP/OAM Flow Match Field, page 264
0x1E3	INGR2_PTP_FLOW_MASK_UPPER	Upper Half of PTP/OAM Flow Match Mask, page 265
0x1E4	INGR2_PTP_FLOW_MASK_LOWER	Lower Half of PTP/OAM Flow Match Mask, page 265
0x1E5	INGR2_PTP_DOMAIN_RANGE	PTP/OAM Range Match, page 265
0x1E6	INGR2_PTP_ACTION	PTP Action Control, page 266
0x1E7	INGR2_PTP_ACTION_2	PTP Action Control 2, page 267
0x1E8	INGR2_PTP_ZERO_FIELD_CTL	Zero Field Control, page 267

4.36 Ingress2 Ethernet Next Protocol Configuration Registers

This section provides information about the Ethernet next protocol configuration registers.

4.36.1 Ethernet Next Protocol

Short Name: INGR2_ETH1_NXT_PROTOCOL_A

Address: 0x00

Table 389 • **Ethernet Next Protocol Register**

Bit	Name	Description	Access	Default
2:0	INGR2_ETH1_NXT_COMPARATOR_A	Points to the next comparator block after this Ethernet block 0: Reserved 1: Ethernet comparator 2 2: IP/UDP/ACH comparator 1 3: IP/UDP/ACH comparator 2 4: MPLS comparator 5: PTP/OAM comparator 6,7: Reserved	R/W	0x0

4.36.2 VLAN TPID Configuration

Short Name: INGR2_ETH1_VLAN_TPID_CFG_A

Address: 0x01

Table 390 • **VLAN TPID Configuration Register**

Bit	Name	Description	Access	Default
31:16	INGR2_ETH1_VLAN_TPID_CFG_A	Configurable VLAN TPID (S or B-tag).	R/W	0x88A8

4.36.3 Ethernet Tag Mode

Short Name: INGR2_ETH1_TAG_MODE_A

Address: 0x02

Table 391 • **Ethernet Tag Mode Register**

Bit	Name	Description	Access	Default
0	INGR2_ETH1_PBB_ENA_A	This bit enables the presence of PBB. The I-tag match bits are programmed in the ETH1_VLAN_TAG_RANGE registers. The mask bits are programmed in the ETH1_VLAN_TAG2 registers. A B-tag if present is configured in the ETH1_VLAN_TAG1 registers. 0: PBB not enabled 1: Always expect PBB, last tag is always an I-tag	R/W	0x0

4.36.4 Ethertype Match

Short Name: INGR2_ETH1_ETYPE_MATCH_A

Address: 0x03

Table 392 • **Ethertype Match Register**

Bit	Name	Description	Access	Default
15:0	INGR2_ETH1_ETYPE_MATCH_A	If the Ethertype/length field is an Ethertype, then this register is compared against the value. If the field is a length, the length value is not checked.	R/W	0x0000

4.36.5 Ethernet Next Protocol

Short Name: INGR2_ETH1_NXT_PROTOCOL_B

Address: 0x10

Table 393 • **Ethernet Next Protocol Register**

Bit	Name	Description	Access	Default
2:0	INGR2_ETH1_NXT_COMPARATOR_B	Points to the next comparator block after this Ethernet block 0: Reserved 1: Ethernet comparator 2 2: IP/UDP/ACH comparator 1 3: IP/UDP/ACH comparator 2 4: MPLS comparator 5: PTP/OAM comparator 6,7: Reserved	R/W	0x0

4.36.6 VLAN TPID Configuration B

Short Name: INGR2_ETH1_VLAN_TPID_CFG_B

Address: 0x11

Table 394 • **VLAN TPID Configuration B Register**

Bit	Name	Description	Access	Default
15:0	INGR2_ETH1_VLAN_TPID_CFG_B	Configurable VLAN TPID (S or B-tag).	R/W	0x88A8

4.36.7 Ethernet Tag Mode

Short Name: INGR2_ETH1_TAG_MODE_B

Address: 0x12

Table 395 • **Ethernet Tag Mode Register**

Bit	Name	Description	Access	Default
0	INGR2_ETH1_PBB_ENA_B	This bit enables the presence of PBB. The I-tag match bits are programmed in the ETH1_VLAN_TAG_RANGE registers. The mask bits are programmed in the ETH1_VLAN_TAG2 registers. A B-tag if present is configured in the ETH1_VLAN_TAG1 registers. 0: PBB not enabled 1: Always expect PBB, last tag is always an I-tag	R/W	0x0

4.36.8 Ethertype Match

Short Name: INGR2_ETH1_ETYPE_MATCH_B

Address: 0x13

Table 396 • **Ethertype Match Register**

Bit	Name	Description	Access	Default
15:0	INGR2_ETH1_ETYPE_MATCH_B	If the Ethertype/length field is an Ethertype, then this register is compared against the value. If the field is a length, the length value is not checked.	R/W	0x0000

4.36.9

Instance offsets: 0x20 INGR2_ETH1_FLOW_CFG_0

0x30 INGR2_ETH1_FLOW_CFG_1

0x40 INGR2_ETH1_FLOW_CFG_2

0x50 INGR2_ETH1_FLOW_CFG_3

0x60 INGR2_ETH1_FLOW_CFG_4

0x70 INGR2_ETH1_FLOW_CFG_5

0x80 INGR2_ETH1_FLOW_CFG_6

0x90 INGR2_ETH1_FLOW_CFG_7

4.36.10 Ethernet Flow Enable

Short Name: INGR2_ETH1_FLOW_ENABLE

Addresses: 0x20 INGR2_ETH1_FLOW_CFG_0

0x30 INGR2_ETH1_FLOW_CFG_1

0x40 INGR2_ETH1_FLOW_CFG_2

0x50 INGR2_ETH1_FLOW_CFG_3

0x60 INGR2_ETH1_FLOW_CFG_4

0x70 INGR2_ETH1_FLOW_CFG_5

0x80 INGR2_ETH1_FLOW_CFG_6

0x90 INGR2_ETH1_FLOW_CFG_7

Table 397 • **Ethernet Flow Enable Register**

Bit	Name	Description	Access	Default
16	INGR2_ETH1_NXT_PROT_GRP_SEL	Indicates which next-protocol configuration group is valid with this flow 0: Associate this flow with next-protocol group A 1: Associate this flow with next-protocol group B	R/W	0x0
9:8	INGR2_ETH1_CHANNEL_MASK	0: Flow valid for channel 0 1: Flow valid for channel 1	R/W	0x3
0	INGR2_ETH1_FLOW_ENABLE	Flow enable 0: Flow is disabled 1: Flow is enabled	R/W	0x0

4.36.11 Ethernet Protocol Match Mode

Short Name: INGR2_ETH1_MATCH_MODE

Addresses: 0x21 INGR2_ETH1_FLOW_CFG_0

0x31 INGR2_ETH1_FLOW_CFG_1

0x41 INGR2_ETH1_FLOW_CFG_2

0x51 INGR2_ETH1_FLOW_CFG_3

0x61 INGR2_ETH1_FLOW_CFG_4

0x71 INGR2_ETH1_FLOW_CFG_5

0x81 INGR2_ETH1_FLOW_CFG_6

0x91 INGR2_ETH1_FLOW_CFG_7

Table 398 • **Ethernet Protocol Match Mode Register**

Bit	Name	Description	Access	Default
13:12	INGR2_ETH1_VLAN_TAG_MODE	0: VLAN range checking disabled 1: VLAN range checking on tag 1 2: VLAN range checking on tag 2 (not supported with PBB) 3: reserved	R/W	0x0

Table 398 • **Ethernet Protocol Match Mode Register *(continued)***

Bit	Name	Description	Access	Default
9	INGR2_ETH1_VLAN_TAG2_TYPE	This register is only used if ETH1_VLAN_VERIFY_ENA = 1 If PBB not enabled: 0: C tag (TPID of 0x8100) 1: S tag (match to CONF_VLAN_TPID) If PBB enabled: 0,1: I tag (use range registers)	R/W	0x1
8	INGR2_ETH1_VLAN_TAG1_TYPE	This register is only used if ETH1_VLAN_VERIFY_ENA = 1 0: C tag (TPID of 0x8100) 1: S or B tag (match to CONF_VLAN_TPID)	R/W	0x0
7:6	INGR2_ETH1_VLAN_TAGS	This register is only used if ETH1_VLAN_VERIFY_ENA = 1 0: No VLAN tags (not valid for PBB) 1: 1 VLAN tag (for PBB this would be the I-tag) 2: 2 VLAN tags (for PBB expect a B-tag and an I-tag) 3: Reserved	R/W	0x0
4	INGR2_ETH1_VLAN_VERIFY_ENA	0: Parse for VLAN tags, do not check values. For PBB the I-tag is always checked. 1: Verify configured VLAN tag configuration.	R/W	0x0
0	INGR2_ETH1_ETHERTYPE_MODE	When checking for presence of SNAP/LLC based upon ETH1_MATCH_MODE, this field indicates if SNAP & 3-byte LLC is expected to be present 0: Only Ethernet type II supported, no SNAP/LLC 1: Ethernet type II & Ethernet type I with SNAP/LLC, determine if SNAP/LLC is present or not. Type I always assumes that SNAP/LLC is present	R/W	0x0

4.36.12 Ethernet Address Match Part 1

Short Name: INGR2_ETH1_ADDR_MATCH_1

Addresses: 0x22 INGR2_ETH1_FLOW_CFG_0

0x32 INGR2_ETH1_FLOW_CFG_1

0x42 INGR2_ETH1_FLOW_CFG_2

0x52 INGR2_ETH1_FLOW_CFG_3

0x62 INGR2_ETH1_FLOW_CFG_4

0x72 INGR2_ETH1_FLOW_CFG_5

0x82 INGR2_ETH1_FLOW_CFG_6

0x92 INGR2_ETH1_FLOW_CFG_7

Table 399 • **Ethernet Address Match Part 1 Register**

Bit	Name	Description	Access	Default
31:0	INGR2_ETH1_ADDR_MATCH_1	First 32 bits of the address match value	R/W	0x00000000

4.36.13 Ethernet Address Match Part 2

Short Name: INGR2_ETH1_ADDR_MATCH_2

Addresses: 0x23 INGR2_ETH1_FLOW_CFG_0

0x33 INGR2_ETH1_FLOW_CFG_1

0x43 INGR2_ETH1_FLOW_CFG_2

0x53 INGR2_ETH1_FLOW_CFG_3

0x63 INGR2_ETH1_FLOW_CFG_4

0x73 INGR2_ETH1_FLOW_CFG_5

0x83 INGR2_ETH1_FLOW_CFG_6

0x93 INGR2_ETH1_FLOW_CFG_7

Table 400 • **Ethernet Address Match Part 2 Register**

Bit	Name	Description	Access	Default
22:20	INGR2_ETH1_ADDR_MATCH_MODE	Selects how the addresses are matched. Multiple bits can be set at once bit 0: Full 48-bit address match bit 1: Match any unicast address bit 2: Match any multicast address	R/W	0x1
17:16	INGR2_ETH1_ADDR_MATCH_SELECT	Selects which address to match 0: Match the Destination Address 1: Match the Source Address 2: Match either the Source of Destination Address 3: Reserved	R/W	0x0
15:0	INGR2_ETH1_ADDR_MATCH_2	Last 16 bits of the Ethernet address match field	R/W	0x0000

4.36.14 Ethernet VLAN Tag Range Match

Short Name: INGR2_ETH1_VLAN_TAG_RANGE_I_TAG

Addresses: 0x24 INGR2_ETH1_FLOW_CFG_0

0x34 INGR2_ETH1_FLOW_CFG_1

0x44 INGR2_ETH1_FLOW_CFG_2

0x54 INGR2_ETH1_FLOW_CFG_3

0x64 INGR2_ETH1_FLOW_CFG_4

0x74 INGR2_ETH1_FLOW_CFG_5

0x84 INGR2_ETH1_FLOW_CFG_6

0x94 INGR2_ETH1_FLOW_CFG_7

Table 401 • **Ethernet VLAN Tag Range Match Register**

Bit	Name	Description	Access	Default
27:16	INGR2_ETH1_VLAN_TAG_RANGE_UPPER	If PBB mode is not enabled, then this register contains the upper range of the VLAN tag range match. If PBB mode is enabled, then this register contains the upper 12 bits of the I-tag	R/W	0xFFF

Table 401 • **Ethernet VLAN Tag Range Match Register *(continued)***

Bit	Name	Description	Access	Default
11:0	INGR2_ETH1_VLAN_TAG_RANGE_LOWER	If PBB mode is not enabled, then this register contains the lower range of the VLAN tag range match. If PBB mode is enabled, then this register contains the lower 12 bits of the I-tag	R/W	0x000

4.36.15 VLAN Tag 1 Match/Mask

Short Name: INGR2_ETH1_VLAN_TAG1

Addresses: 0x25 INGR2_ETH1_FLOW_CFG_0

0x35 INGR2_ETH1_FLOW_CFG_1

0x45 INGR2_ETH1_FLOW_CFG_2

0x55 INGR2_ETH1_FLOW_CFG_3

0x65 INGR2_ETH1_FLOW_CFG_4

0x75 INGR2_ETH1_FLOW_CFG_5

0x85 INGR2_ETH1_FLOW_CFG_6

0x95 INGR2_ETH1_FLOW_CFG_7

Table 402 • **VLAN Tag 1 Match/Mask Register**

Bit	Name	Description	Access	Default
27:16	INGR2_ETH1_VLAN_TAG1_MASK	Mask value for VLAN tag 1	R/W	0xFFF
11:0	INGR2_ETH1_VLAN_TAG1_MATCH	Match value for the first VLAN tag	R/W	0x000

4.36.16 Match/Mask For VLAN Tag 2 or I-Tag Match

Short Name: INGR2_ETH1_VLAN_TAG2_I_TAG

Addresses: 0x26 INGR2_ETH1_FLOW_CFG_0

0x36 INGR2_ETH1_FLOW_CFG_1

0x46 INGR2_ETH1_FLOW_CFG_2

0x56 INGR2_ETH1_FLOW_CFG_3

0x66 INGR2_ETH1_FLOW_CFG_4

0x76 INGR2_ETH1_FLOW_CFG_5

0x86 INGR2_ETH1_FLOW_CFG_6

0x96 INGR2_ETH1_FLOW_CFG_7

Table 403 • **Match/Mask For VLAN Tag 2 or I-Tag Match Register**

Bit	Name	Description	Access	Default
27:16	INGR2_ETH1_VLAN_TAG2_MASK	When PBB is not enabled, the mask field for VLAN tag 2 When PBB is enabled, the upper 12 bits of the I-tag mask	R/W	0xFFF

Table 403 • **Match/Mask For VLAN Tag 2 or I-Tag Match Register *(continued)***

Bit	Name	Description	Access	Default
11:0	INGR2_ETH1_VLAN_TAG2_MATCH	When PBB is not enabled, the match field for VLAN Tag 2 When PBB is enabled, the lower 12 bits of the I-tag mask field	R/W	0x000

4.36.17 Ethernet Next Protocol

Short Name: INGR2_ETH2_NXT_PROTOCOL_A

Address: 0xA0

Table 404 • **Ethernet Next Protocol Register**

Bit	Name	Description	Access	Default
2:0	INGR2_ETH2_NXT_COMPARATOR_A	Points to the next comparator block after this Ethernet block. If this comparator block is not used, this field must be set to 0. 0: Comparator block not used 1: Ethernet comparator 2 2: IP/UDP/ACH comparator 1 3: IP/UDP/ACH comparator 2 4: MPLS comparator 5: PTP/OAM comparator 6,7: Reserved	R/W	0x0

4.36.18 VLAN TPID Configuration

Short Name: INGR2_ETH2_VLAN_TPID_CFG_A

Address: 0xA1

Table 405 • **VLAN TPID Configuration Register**

Bit	Name	Description	Access	Default
31:16	INGR2_ETH2_VLAN_TPID_CFG_A	Configurable S-tag TPID	R/W	0x88A8

4.36.19 Ethertype Match

Short Name: INGR2_ETH2_ETYPE_MATCH_A

Address: 0xA2

Table 406 • **Ethertype Match Register**

Bit	Name	Description	Access	Default
15:0	INGR2_ETH2_ETYPE_MATCH_A	If the Ethertype/length field is an Ethertype, then this register is compared against the value. If the field is a length, the length value is not checked.	R/W	0x0000

4.36.20

Instance offsets: 0xC0 INGR2_ETH2_FLOW_CFG_0

0xD0 INGR2_ETH2_FLOW_CFG_1

0xE0 INGR2_ETH2_FLOW_CFG_2

0xF0 INGR2_ETH2_FLOW_CFG_3

0x100 INGR2_ETH2_FLOW_CFG_4

0x110 INGR2_ETH2_FLOW_CFG_5

0x120 INGR2_ETH2_FLOW_CFG_6

0x130 INGR2_ETH2_FLOW_CFG_7

4.36.21 Ethernet Flow Enable

Short Name: INGR2_ETH2_FLOW_ENABLE

Addresses: 0xC0 INGR2_ETH2_FLOW_CFG_0

0xD0 INGR2_ETH2_FLOW_CFG_1

0xE0 INGR2_ETH2_FLOW_CFG_2

0xF0 INGR2_ETH2_FLOW_CFG_3

0x100 INGR2_ETH2_FLOW_CFG_4

0x110 INGR2_ETH2_FLOW_CFG_5

0x120 INGR2_ETH2_FLOW_CFG_6

0x130 INGR2_ETH2_FLOW_CFG_7

Table 407 • **Ethernet Flow Enable Register**

Bit	Name	Description	Access	Default
9:8	INGR2_ETH2_CHANNEL_MASK	bit 0: Flow valid for channel 0 bit 1: Flow valid for channel 1	R/W	0x3
0	INGR2_ETH2_FLOW_ENABLE	Flow enable. If this comparator block is not used, all flow enable bits must be set to 0. 0: Flow is disabled 1: Flow is enabled	R/W	0x0

4.36.22 Ethernet Protocol Match Mode

Short Name: INGR2_ETH2_MATCH_MODE

Addresses: 0xC1 INGR2_ETH2_FLOW_CFG_0

0xD1 INGR2_ETH2_FLOW_CFG_1

0xE1 INGR2_ETH2_FLOW_CFG_2

0xF1 INGR2_ETH2_FLOW_CFG_3

0x101 INGR2_ETH2_FLOW_CFG_4

0x111 INGR2_ETH2_FLOW_CFG_5

0x121 INGR2_ETH2_FLOW_CFG_6

0x131 INGR2_ETH2_FLOW_CFG_7

Table 408 • **Ethernet Protocol Match Mode Register**

Bit	Name	Description	Access	Default
13:12	INGR2_ETH2_VLAN_TAG_MODE	0: VLAN range checking disabled 1: VLAN range checking on tag 1 2: VLAN range checking on tag 2 (not supported with PBB) 3: reserved	R/W	0x0
9	INGR2_ETH2_VLAN_TAG2_TYPE	This register is only used if ETH1_VLAN_VERIFY_ENA = 1 0: C tag (TPID of 0x8100) 1: S tag (match to CONF_VLAN_TPID)	R/W	0x1
8	INGR2_ETH2_VLAN_TAG1_TYPE	This register is only used if ETH1_VLAN_VERIFY_ENA = 1 0: C tag (TPID of 0x8100) 1: S or B tag (match to CONF_VLAN_TPID)	R/W	0x0
7:6	INGR2_ETH2_VLAN_TAGS	This register is only used if ETH2_VLAN_VERIFY_ENA = 1 0: No VLAN tags 1: 1 VLAN tag 2: 2 VLAN tags 3: Reserved	R/W	0x0
4	INGR2_ETH2_VLAN_VERIFY_ENA	0: Parse for VLAN tags, do not check values. 1: Verify configured VLAN tag configuration.	R/W	0x0
0	INGR2_ETH2_ETHERTYPE_MODE	When checking for presence of SNAP/LLC based upon ETH1_MATCH_MODE, this field indicates if SNAP & 3-byte LLC is expected to be present 0: Only Ethernet type II supported, no SNAP/LLC 1: Ethernet type II & Ethernet type I with SNAP/LLC, determine if SNAP/LLC is present or not. Type I always assumes that SNAP/LLC is present	R/W	0x0

4.36.23 Ethernet Address Match Part 1

Short Name: INGR2_ETH2_ADDR_MATCH_1

Addresses: 0xC2 INGR2_ETH2_FLOW_CFG_0

0xD2 INGR2_ETH2_FLOW_CFG_1

0xE2 INGR2_ETH2_FLOW_CFG_2

0xF2 INGR2_ETH2_FLOW_CFG_3

0x102 INGR2_ETH2_FLOW_CFG_4

0x112 INGR2_ETH2_FLOW_CFG_5

0x122 INGR2_ETH2_FLOW_CFG_6

0x132 INGR2_ETH2_FLOW_CFG_7

Table 409 • **Ethernet Address Match Part 1 Register**

Bit	Name	Description	Access	Default
31:0	INGR2_ETH2_ADDR_MATCH_1	First 32 bits of the address match value	R/W	0x00000000

4.36.24 Ethernet Address Match Part 2

Short Name: INGR2_ETH2_ADDR_MATCH_2

Addresses: 0xC3 INGR2_ETH2_FLOW_CFG_0

0xD3 INGR2_ETH2_FLOW_CFG_1

0xE3 INGR2_ETH2_FLOW_CFG_2

0xF3 INGR2_ETH2_FLOW_CFG_3

0x103 INGR2_ETH2_FLOW_CFG_4

0x113 INGR2_ETH2_FLOW_CFG_5

0x123 INGR2_ETH2_FLOW_CFG_6

0x133 INGR2_ETH2_FLOW_CFG_7

Table 410 • **Ethernet Address Match Part 2 Register**

Bit	Name	Description	Access	Default
22:20	INGR2_ETH2_ADDR_MATCH_MODE	Selects how the addresses are matched. Multiple bits can be set at once bit 0: Full 48-bit address match bit 1: Match any unicast address bit 2: Match any multicast address	R/W	0x1
17:16	INGR2_ETH2_ADDR_MATCH_SELECT	Selects which address to match 0: Match the Destination Address 1: Match the Source Address 2: Match either the Source of Destination Address 3: Reserved	R/W	0x0
15:0	INGR2_ETH2_ADDR_MATCH_2	Last 16 bits of the Ethernet address match field	R/W	0x0000

4.36.25 Ethernet VLAN Tag Range Match

Short Name: INGR2_ETH2_VLAN_TAG_RANGE_I_TAG

Addresses: 0xC4 INGR2_ETH2_FLOW_CFG_0

0xD4 INGR2_ETH2_FLOW_CFG_1

0xE4 INGR2_ETH2_FLOW_CFG_2

0xF4 INGR2_ETH2_FLOW_CFG_3

0x104 INGR2_ETH2_FLOW_CFG_4

0x114 INGR2_ETH2_FLOW_CFG_5

0x124 INGR2_ETH2_FLOW_CFG_6

0x134 INGR2_ETH2_FLOW_CFG_7

Table 411 • **Ethernet VLAN Tag Range Match Register**

Bit	Name	Description	Access	Default
27:16	INGR2_ETH2_VLAN_TAG_RANGE_UPPER	This register contains the upper range of the VLAN tag range match.	R/W	0xFFF
11:0	INGR2_ETH2_VLAN_TAG_RANGE_LOWER	This register contains the lower range of the VLAN tag range match.	R/W	0x000

4.36.26 VLAN Tag 1 Match/Mask

Short Name: INGR2_ETH2_VLAN_TAG1

Addresses: 0xC5 INGR2_ETH2_FLOW_CFG_0

0xD5 INGR2_ETH2_FLOW_CFG_1

0xE5 INGR2_ETH2_FLOW_CFG_2

0xF5 INGR2_ETH2_FLOW_CFG_3

0x105 INGR2_ETH2_FLOW_CFG_4

0x115 INGR2_ETH2_FLOW_CFG_5

0x125 INGR2_ETH2_FLOW_CFG_6

0x135 INGR2_ETH2_FLOW_CFG_7

Table 412 • **VLAN Tag 1 Match/Mask Register**

Bit	Name	Description	Access	Default
27:16	INGR2_ETH2_VLAN_TAG1_MASK	Mask value for VLAN tag 1	R/W	0xFFF
11:0	INGR2_ETH2_VLAN_TAG1_MATCH	Match value for the first VLAN tag	R/W	0x000

4.36.27 Match/Mask For VLAN Tag 2 or I-Tag Match

Short Name: INGR2_ETH2_VLAN_TAG2_I_TAG

Addresses: 0xC6 INGR2_ETH2_FLOW_CFG_0

0xD6 INGR2_ETH2_FLOW_CFG_1

0xE6 INGR2_ETH2_FLOW_CFG_2

0xF6 INGR2_ETH2_FLOW_CFG_3

0x106 INGR2_ETH2_FLOW_CFG_4

0x116 INGR2_ETH2_FLOW_CFG_5

0x126 INGR2_ETH2_FLOW_CFG_6

0x136 INGR2_ETH2_FLOW_CFG_7

Table 413 • **Match/Mask For VLAN Tag 2 or I-Tag Match Register**

Bit	Name	Description	Access	Default
27:16	INGR2_ETH2_VLAN_TAG2_MASK	Mask field for VLAN tag 2	R/W	0xFFF
11:0	INGR2_ETH2_VLAN_TAG2_MATCH	Match field for VLAN Tag 2	R/W	0x000

4.37 Ingress2 MPLS Next Protocol Registers

This section provides information about the MPLS next protocol registers.

4.37.1 MPLS Next Protocol Comparator

Short Name: INGR2_MPLS_NXT_COMPARATOR_A

Address: 0x140

Table 414 • **MPLS Next Protocol Comparator Register**

Bit	Name	Description	Access	Default
16	INGR2_MPLS_CTL_WORD_A	Indicates the presence of a control word after the last label 0: There is no control ward after the last label 1: There is a control word after the last label	R/W	0x0
2:0	INGR2_MPLS_NXT_COMPARATOR_A	Points to the next comparator stage. If this comparator block is not used, this field must be set to 0. 0: Comparator block not used 1: Ethernet comparator 2 2: IP/UDP/ACH comparator 1 3: IP/UDP/ACH comparator 2 4: Reserved 5: PTP/OAM comparator 6,7: Reserved	R/W	0x0

4.37.2

Instance offsets: 0x160 INGR2_MPLS_FLOW_CFG_0

0x170 INGR2_MPLS_FLOW_CFG_1

0x180 INGR2_MPLS_FLOW_CFG_2

0x190 INGR2_MPLS_FLOW_CFG_3

0x1A0 INGR2_MPLS_FLOW_CFG_4

0x1B0 INGR2_MPLS_FLOW_CFG_5

0x1C0 INGR2_MPLS_FLOW_CFG_6

0x1D0 INGR2_MPLS_FLOW_CFG_7

4.37.3 MPLS Flow Control

Short Name: INGR2_MPLS_FLOW_CONTROL

Addresses: 0x160 INGR2_MPLS_FLOW_CFG_0

0x170 INGR2_MPLS_FLOW_CFG_1

0x180 INGR2_MPLS_FLOW_CFG_2

0x190 INGR2_MPLS_FLOW_CFG_3

0x1A0 INGR2_MPLS_FLOW_CFG_4

0x1B0 INGR2_MPLS_FLOW_CFG_5

0x1C0 INGR2_MPLS_FLOW_CFG_6

0x1D0 INGR2_MPLS_FLOW_CFG_7

Table 415 • **MPLS Flow Control Register**

Bit	Name	Description	Access	Default
25:24	INGR2_MPLS_CHANNEL_MASK	bit 0: Flow valid for channel 0 bit 1: Flow valid for channel 1	R/W	0x3
19:16	INGR2_MPLS_STACK_DEPTH	Defines the allowable stack depths for searches. The direction that the stack is referenced is determined by the setting of MPLS_REF_PNT The following table maps bits to stack depths: bit 0: stack allowed to be 1 label deep bit 1: stack allowed to be 2 labels deep bit 2: stack allowed to be 3 labels deep bit 3: stack allowed to be 4 labels deep	R/W	0x0
4	INGR2_MPLS_REF_PNT	Defines the search direction for label matching 0: All searching is performed starting from the top of the stack 1: All searching is performed from the end of the stack	R/W	0x0
0	INGR2_MPLS_FLOW_ENA	Flow enable. If this comparator block is not used, all flow enable bits must be set to 0. 0: Flow is disabled 1: Flow is enabled	R/W	0x0

4.37.4 MPLS Label 0 Match Range Lower Value

Short Name: INGR2_MPLS_LABEL_RANGE_LOWER_0

Addresses: 0x161 INGR2_MPLS_FLOW_CFG_0

0x171 INGR2_MPLS_FLOW_CFG_1

0x181 INGR2_MPLS_FLOW_CFG_2

0x191 INGR2_MPLS_FLOW_CFG_3

0x1A1 INGR2_MPLS_FLOW_CFG_4

0x1B1 INGR2_MPLS_FLOW_CFG_5

0x1C1 INGR2_MPLS_FLOW_CFG_6

0x1D1 INGR2_MPLS_FLOW_CFG_7

Table 416 • **MPLS Label 0 Match Range Lower Value Register**

Bit	Name	Description	Access	Default
19:0	INGR2_MPLS_LABEL_RANGE_LOWER_0	Lower value for label 0 match range	R/W	0x00000

4.37.5 MPLS Label 0 Match Range Lower Value

Short Name: INGR2_MPLS_LABEL_RANGE_UPPER_0

Addresses: 0x162 INGR2_MPLS_FLOW_CFG_0

0x172 INGR2_MPLS_FLOW_CFG_1

0x182 INGR2_MPLS_FLOW_CFG_2

0x192 INGR2_MPLS_FLOW_CFG_3

0x1A2 INGR2_MPLS_FLOW_CFG_4

0x1B2 INGR2_MPLS_FLOW_CFG_5

0x1C2 INGR2_MPLS_FLOW_CFG_6

0x1D2 INGR2_MPLS_FLOW_CFG_7

Table 417 • **MPLS Label 0 Match Range Lower Value Register**

Bit	Name	Description	Access	Default
19:0	INGR2_MPLS_LABEL_RANGE_UPPER_0	Upper value for label 0 match range	R/W	0xFFFFF

4.37.6 MPLS Label 1 Match Range Lower Value

Short Name: INGR2_MPLS_LABEL_RANGE_LOWER_1

Addresses: 0x163 INGR2_MPLS_FLOW_CFG_0

0x173 INGR2_MPLS_FLOW_CFG_1

0x183 INGR2_MPLS_FLOW_CFG_2

0x193 INGR2_MPLS_FLOW_CFG_3

0x1A3 INGR2_MPLS_FLOW_CFG_4

0x1B3 INGR2_MPLS_FLOW_CFG_5

0x1C3 INGR2_MPLS_FLOW_CFG_6

0x1D3 INGR2_MPLS_FLOW_CFG_7

Table 418 • **MPLS Label 1 Match Range Lower Value Register**

Bit	Name	Description	Access	Default
19:0	INGR2_MPLS_LABEL_RANGE_LOWER_1	Lower value for label 1 match range	R/W	0x00000

4.37.7 MPLS Label 1 Match Range Lower Value

Short Name: INGR2_MPLS_LABEL_RANGE_UPPER_1

Addresses: 0x164 INGR2_MPLS_FLOW_CFG_0

0x174 INGR2_MPLS_FLOW_CFG_1

0x184 INGR2_MPLS_FLOW_CFG_2

0x194 INGR2_MPLS_FLOW_CFG_3

0x1A4 INGR2_MPLS_FLOW_CFG_4

0x1B4 INGR2_MPLS_FLOW_CFG_5

0x1C4 INGR2_MPLS_FLOW_CFG_6

0x1D4 INGR2_MPLS_FLOW_CFG_7

Table 419 • **MPLS Label 1 Match Range Lower Value Register**

Bit	Name	Description	Access	Default
19:0	INGR2_MPLS_LABEL_RANGE_UPPER_1	Upper value for label 1 match range	R/W	0xFFFFF

4.37.8 MPLS Label 2 Match Range Lower Value

Short Name: INGR2_MPLS_LABEL_RANGE_LOWER_2

Addresses: 0x165 INGR2_MPLS_FLOW_CFG_0

0x175 INGR2_MPLS_FLOW_CFG_1

0x185 INGR2_MPLS_FLOW_CFG_2

0x195 INGR2_MPLS_FLOW_CFG_3

0x1A5 INGR2_MPLS_FLOW_CFG_4

0x1B5 INGR2_MPLS_FLOW_CFG_5

0x1C5 INGR2_MPLS_FLOW_CFG_6

0x1D5 INGR2_MPLS_FLOW_CFG_7

Table 420 • **MPLS Label 2 Match Range Lower Value Register**

Bit	Name	Description	Access	Default
19:0	INGR2_MPLS_LABEL_RANGE_LOWER_2	Lower value for label 2 match range	R/W	0x00000

4.37.9 MPLS Label 2 Match Range Lower Value

Short Name: INGR2_MPLS_LABEL_RANGE_UPPER_2

Addresses: 0x166 INGR2_MPLS_FLOW_CFG_0

0x176 INGR2_MPLS_FLOW_CFG_1

0x186 INGR2_MPLS_FLOW_CFG_2

0x196 INGR2_MPLS_FLOW_CFG_3

0x1A6 INGR2_MPLS_FLOW_CFG_4

0x1B6 INGR2_MPLS_FLOW_CFG_5

0x1C6 INGR2_MPLS_FLOW_CFG_6

0x1D6 INGR2_MPLS_FLOW_CFG_7

Table 421 • **MPLS Label 2 Match Range Lower Value Register**

Bit	Name	Description	Access	Default
19:0	INGR2_MPLS_LABEL_RANGE_UPPER_2	Upper value for label 2 match range	R/W	0xFFFFF

4.37.10 MPLS Label 3 Match Range Lower Value

Short Name: INGR2_MPLS_LABEL_RANGE_LOWER_3

Addresses: 0x167 INGR2_MPLS_FLOW_CFG_0

0x177 INGR2_MPLS_FLOW_CFG_1

0x187 INGR2_MPLS_FLOW_CFG_2

0x197 INGR2_MPLS_FLOW_CFG_3

0x1A7 INGR2_MPLS_FLOW_CFG_4

0x1B7 INGR2_MPLS_FLOW_CFG_5

0x1C7 INGR2_MPLS_FLOW_CFG_6

0x1D7 INGR2_MPLS_FLOW_CFG_7

Table 422 • **MPLS Label 3 Match Range Lower Value Register**

Bit	Name	Description	Access	Default
19:0	INGR2_MPLS_LABEL_RANGE_LOWER_3	Lower value for label 3 match range	R/W	0x00000

4.37.11 MPLS Label 3 Match Range Lower Value

Short Name: INGR2_MPLS_LABEL_RANGE_UPPER_3

Addresses: 0x168 INGR2_MPLS_FLOW_CFG_0

0x178 INGR2_MPLS_FLOW_CFG_1

0x188 INGR2_MPLS_FLOW_CFG_2

0x198 INGR2_MPLS_FLOW_CFG_3

0x1A8 INGR2_MPLS_FLOW_CFG_4

0x1B8 INGR2_MPLS_FLOW_CFG_5

0x1C8 INGR2_MPLS_FLOW_CFG_6

0x1D8 INGR2_MPLS_FLOW_CFG_7

Table 423 • **MPLS Label 3 Match Range Lower Value Register**

Bit	Name	Description	Access	Default
19:0	INGR2_MPLS_LABEL_RANGE_UPPER_3	Upper value for label 3 match range	R/W	0xFFFFF

4.37.12

Instance offsets: 0x1E0 INGR2_PTP_FLOW_0

0x1F0 INGR2_PTP_FLOW_1

0x200 INGR2_PTP_FLOW_2

0x210 INGR2_PTP_FLOW_3

0x220 INGR2_PTP_FLOW_4

0x230 INGR2_PTP_FLOW_5

4.37.13 PTP/OAM Flow Enable

Short Name: INGR2_PTP_FLOW_ENA

Addresses: 0x1E0 INGR2_PTP_FLOW_0

0x1F0 INGR2_PTP_FLOW_1

0x200 INGR2_PTP_FLOW_2

0x210 INGR2_PTP_FLOW_3

0x220 INGR2_PTP_FLOW_4

0x230 INGR2_PTP_FLOW_5

Table 424 • **PTP/OAM Flow Enable Register**

Bit	Name	Description	Access	Default
17:16	INGR2_PTP_NXT_PROT_GRP_MASK	Indicates which next protocol groups that this flow is valid for. For each next protocol group, if the bit is 1, then this flow is valid for that group. If it is 0, then it is not valid for the group. 0: Mask bit for next protocol group A 1: Mask bit for next protocol group B	R/W	0x3
5:4	INGR2_PTP_CHANNEL_MASK	0: Flow valid for channel 0 1: Flow valid for channel 1	R/W	0x3
0	INGR2_PTP_FLOW_ENA		R/W	0x0

4.37.14 Upper Half of PTP/OAM Flow Match Field

Short Name: INGR2_PTP_FLOW_MATCH_UPPER

Addresses: 0x1E1 INGR2_PTP_FLOW_0

0x1F1 INGR2_PTP_FLOW_1

0x201 INGR2_PTP_FLOW_2

0x211 INGR2_PTP_FLOW_3

0x221 INGR2_PTP_FLOW_4

0x231 INGR2_PTP_FLOW_5

Table 425 • **Upper Half of PTP/OAM Flow Match Field Register**

Bit	Name	Description	Access	Default
31:0	INGR2_PTP_FLOW_MATCH_UPPER		R/W	0x00000000

4.37.15 Lower Half of PTP/OAM Flow Match Field

Short Name: INGR2_PTP_FLOW_MATCH_LOWER

Addresses: 0x1E2 INGR2_PTP_FLOW_0

0x1F2 INGR2_PTP_FLOW_1

0x202 INGR2_PTP_FLOW_2

0x212 INGR2_PTP_FLOW_3

0x222 INGR2_PTP_FLOW_4

0x232 INGR2_PTP_FLOW_5

Table 426 • **Lower Half of PTP/OAM Flow Match Field Register**

Bit	Name	Description	Access	Default
31:0	INGR2_PTP_FLOW_MATCH_LOWER		R/W	0x00000000

4.37.16 Upper Half of PTP/OAM Flow Match Mask

Short Name: INGR2_PTP_FLOW_MASK_UPPER

Addresses: 0x1E3 INGR2_PTP_FLOW_0

0x1F3 INGR2_PTP_FLOW_1

0x203 INGR2_PTP_FLOW_2

0x213 INGR2_PTP_FLOW_3

0x223 INGR2_PTP_FLOW_4

0x233 INGR2_PTP_FLOW_5

Table 427 • **Upper Half of PTP/OAM Flow Match Mask Register**

Bit	Name	Description	Access	Default
31:0	INGR2_PTP_FLOW_MASK_UPPER		R/W	0x00000000

4.37.17 Lower Half of PTP/OAM Flow Match Mask

Short Name: INGR2_PTP_FLOW_MASK_LOWER

Addresses: 0x1E4 INGR2_PTP_FLOW_0

0x1F4 INGR2_PTP_FLOW_1

0x204 INGR2_PTP_FLOW_2

0x214 INGR2_PTP_FLOW_3

0x224 INGR2_PTP_FLOW_4

0x234 INGR2_PTP_FLOW_5

Table 428 • **Lower Half of PTP/OAM Flow Match Mask Register**

Bit	Name	Description	Access	Default
31:0	INGR2_PTP_FLOW_MASK_LOWER		R/W	0x00000000

4.37.18 PTP/OAM Range Match

Short Name: INGR2_PTP_DOMAIN_RANGE

Addresses: 0x1E5 INGR2_PTP_FLOW_0

0x1F5 INGR2_PTP_FLOW_1

0x205 INGR2_PTP_FLOW_2

0x215 INGR2_PTP_FLOW_3

0x225 INGR2_PTP_FLOW_4

0x235 INGR2_PTP_FLOW_5

Table 429 • **PTP/OAM Range Match Register**

Bit	Name	Description	Access	Default
28:24	INGR2_PTP_DOMAIN_RANGE_OFFSET		R/W	0x00

Table 429 • **PTP/OAM Range Match Register *(continued)***

Bit	Name	Description	Access	Default
23:16	INGR2_PTP_DOMAIN_RANGE_UPPER		R/W	0xFF
15:8	INGR2_PTP_DOMAIN_RANGE_LOWER		R/W	0x00
0	INGR2_PTP_DOMAIN_RANGE_ENA		R/W	0x0

4.37.19 PTP Action Control

Short Name: INGR2_PTP_ACTION

Addresses: 0x1E6 INGR2_PTP_FLOW_0

0x1F6 INGR2_PTP_FLOW_1

0x206 INGR2_PTP_FLOW_2

0x216 INGR2_PTP_FLOW_3

0x226 INGR2_PTP_FLOW_4

0x236 INGR2_PTP_FLOW_5

Table 430 • **PTP Action Control Register**

Bit	Name	Description	Access	Default
28	INGR2_PTP_MOD_FRAME_STAT_UPDATE	1: Tell the Rewriter to update the value of the Modified Frame Status bit 0: Do not update the bit	R/W	0x0
26:24	INGR2_PTP_MOD_FRAME_BYTE_OFFSET	Indicates the position relative to the start of the PTP frame in bytes where the Modified_Frame_Status bit resides	R/W	0x0
21	INGR2_PTP_SUB_DELAY_ASYM_ENA	1: Signal the Timestamp block to subtract the asymmetry delay 0: Do not signal the Timestamp block to subtract the asymmetry delay	R/W	0x0
20	INGR2_PTP_ADD_DELAY_ASYM_ENA	1: Signal the Timestamp block to add the asymmetry delay 0: Do not signal the Timestamp block to add the asymmetry delay	R/W	0x0
15:10	INGR2_PTP_TIME_STRG_FIELD_OFFSET	Points to the reserved 32-bit field where the Rx timestamp is saved. The location is relative to the first byte of the PTP/OAM header.	R/W	0x00
9:5	INGR2_PTP_CORR_FIELD_OFFSET	Points to the location of the correction field for updating the timestamp. Location is relative to the first byte of the PTP/OAM header. **Note:** If this flow is being used to match OAM frames, set this register to 4	R/W	0x00
4	INGR2_PTP_SAVE_LOCAL_TIME	1: Save the local time to the Timestamp FIFO 0: Do not save the time to the Timestamp FIFO	R/W	0x0

Table 430 • **PTP Action Control Register *(continued)***

Bit	Name	Description	Access	Default
3:0	INGR2_PTP_COMMAND	0: NoP 1: SUB 2: SUB_P2P 3: ADD 4: SUB_ADD 5: WRITE_1588 6: WRITE_P2P (deprecated) 7: WRITE_NS 8: WRITE_NS_P2P	R/W	0x0

4.37.20 PTP Action Control 2

Short Name: INGR2_PTP_ACTION_2

Addresses: 0x1E7 INGR2_PTP_FLOW_0

0x1F7 INGR2_PTP_FLOW_1

0x207 INGR2_PTP_FLOW_2

0x217 INGR2_PTP_FLOW_3

0x227 INGR2_PTP_FLOW_4

0x237 INGR2_PTP_FLOW_5

Table 431 • **PTP Action Control 2 Register**

Bit	Name	Description	Access	Default
23:16	INGR2_PTP_NEW_CF_LOC	Location of the new correction field relative to the PTP header start. Only even values are allowed.	R/W	0x00
15:8	INGR2_PTP_REWRITE_OFFSET	Byte offset relative to the start of the PTP frame where the ingress timestamp value can be stored.	R/W	0x00
3:0	INGR2_PTP_REWRITE_BYTES	Number of bytes in the PTP or OAM frame that must be modified by the Rewriter for the timestamp	R/W	0x0

4.37.21 Zero Field Control

Short Name: INGR2_PTP_ZERO_FIELD_CTL

Addresses: 0x1E8 INGR2_PTP_FLOW_0

0x1F8 INGR2_PTP_FLOW_1

0x208 INGR2_PTP_FLOW_2

0x218 INGR2_PTP_FLOW_3

0x228 INGR2_PTP_FLOW_4

0x238 INGR2_PTP_FLOW_5

Table 432 • **Zero Field Control Register**

Bit	Name	Description	Access	Default
13:8	INGR2_PTP_ZERO_FIELD_OFFSET	Points to a location in the PTP/OAM frame relative to the start of the PTP header that will be zeroed if this function is enabled	R/W	0x00
3:0	INGR2_PTP_ZERO_FIELD_BYTE_CNT	The number of bytes to be zeroed. If this field is 0, then this function is not enabled.	R/W	0x0

4.38 Egress2 Analyzer Engine Configuration Registers

This section lists the overviews for the analyzer engine configuration egress2 registers.

Note: The analyzer engine configuration registers are not initialized to the default values during chip reset. Software must configure these registers to their default value.

Note: For more information about accessing the 1588 IP registers, see Accessing 1588 IP Registers, page 72.

Table 433 • **EGR2_ETH1_NXT_PROTOCOL_A**

Address	Name	Details
0x00	EGR2_ETH1_NXT_PROTOCOL_A	Ethernet Next Protocol, page 270
0x01	EGR2_ETH1_VLAN_TPID_CFG_A	VLAN TPID Configuration, page 271
0x02	EGR2_ETH1_TAG_MODE_A	Ethernet Tag Mode, page 271
0x03	EGR2_ETH1_ETYPE_MATCH_A	Ethertype Match, page 271

Table 434 • **EGR2_ETH1_NXT_PROTOCOL_B**

Address	Name	Details
0x10	EGR2_ETH1_NXT_PROTOCOL_B	Ethernet Next Protocol, page 272
0x11	EGR2_ETH1_VLAN_TPID_CFG_B	VLAN TPID Configuration, page 272
0x12	EGR2_ETH1_TAG_MODE_B	Ethernet Tag Mode, page 272
0x13	EGR2_ETH1_ETYPE_MATCH_B	Ethertype Match, page 272

Table 435 • **EGR2_ETH1_FLOW_CFG (8 instances)**

Address	Name	Details
0x20	EGR2_ETH1_FLOW_ENABLE	Ethernet Flow Enable, page 273
0x21	EGR2_ETH1_MATCH_MODE	Ethernet Protocol Match Mode, page 273
0x22	EGR2_ETH1_ADDR_MATCH_1	Ethernet Address Match Part 1, page 275
0x23	EGR2_ETH1_ADDR_MATCH_2	Ethernet Address Match Part 2, page 275
0x24	EGR2_ETH1_VLAN_TAG_RANGE_I_TAG	Ethernet VLAN Tag Range Match, page 275

Table 435 • **EGR2_ETH1_FLOW_CFG (8 instances)** ***(continued)***

Address	Name	Details
0x25	EGR2_ETH1_VLAN_TAG1	VLAN Tag 1 Match/Mask, page 276
0x26	EGR2_ETH1_VLAN_TAG2_I_TAG	Match/Mask For VLAN Tag 2 or I-Tag Match, page 276

Table 436 • **EGR2_ETH2_NXT_PROTOCOL_A**

Address	Name	Details
0xA0	EGR2_ETH2_NXT_PROTOCOL_A	Ethernet Next Protocol, page 277
0xA1	EGR2_ETH2_VLAN_TPID_CFG_A	VLAN TPID Configuration, page 277
0xA2	EGR2_ETH2_ETYPE_MATCH_A	Ethertype Match, page 277

Table 437 • **EGR2_ETH2_FLOW_CFG (8 instances)**

Address	Name	Details
0xC0	EGR2_ETH2_FLOW_ENABLE	Ethernet Flow Enable, page 278
0xC1	EGR2_ETH2_MATCH_MODE	Ethernet Protocol Match Mode, page 278
0xC2	EGR2_ETH2_ADDR_MATCH_1	Ethernet Address Match Part 1, page 279
0xC3	EGR2_ETH2_ADDR_MATCH_2	Ethernet Address Match Part 2, page 280
0xC4	EGR2_ETH2_VLAN_TAG_RANGE_I_TAG	Ethernet VLAN Tag Range Match, page 280
0xC5	EGR2_ETH2_VLAN_TAG1	VLAN Tag 1 Match/Mask, page 281
0xC6	EGR2_ETH2_VLAN_TAG2_I_TAG	Match/Mask For VLAN Tag 2 or I-Tag Match, page 281

Table 438 • **EGR2_MPLS_NXT_COMPARATOR_A**

Address	Name	Details
0x140	EGR2_MPLS_NXT_COMPARATOR_A	MPLS Next Protocol Comparator, page 282

Table 439 • **EGR2_MPLS_FLOW_CFG (8 instances)**

Address	Name	Details
0x160	EGR2_MPLS_FLOW_CONTROL	MPLS Flow Control, page 282
0x161	EGR2_MPLS_LABEL_RANGE_LOWER_0	MPLS Label 0 Match Range Lower Value, page 283
0x162	EGR2_MPLS_LABEL_RANGE_UPPER_0	MPLS Label 0 Match Range Lower Value, page 283
0x163	EGR2_MPLS_LABEL_RANGE_LOWER_1	MPLS Label 1 Match Range Lower Value, page 284

Table 439 • **EGR2_MPLS_FLOW_CFG (8 instances)** ***(continued)***

Address	Name	Details
0x164	EGR2_MPLS_LABEL_RANGE_UPPER_1	MPLS Label 1 Match Range Lower Value, page 284
0x165	EGR2_MPLS_LABEL_RANGE_LOWER_2	MPLS Label 2 Match Range Lower Value, page 285
0x166	EGR2_MPLS_LABEL_RANGE_UPPER_2	MPLS Label 2 Match Range Lower Value, page 285
0x167	EGR2_MPLS_LABEL_RANGE_LOWER_3	MPLS Label 3 Match Range Lower Value, page 285
0x168	EGR2_MPLS_LABEL_RANGE_UPPER_3	MPLS Label 3 Match Range Lower Value, page 286

Table 440 • **EGR2_PTP_FLOW (6 instances)**

Address	Register Name	Details
0x1E0	EGR2_PTP_FLOW_ENA	PTP/OAM Flow Enable, page 286
0x1E1	EGR2_PTP_FLOW_MATCH_UPPER	Upper Half of PTP/OAM Flow Match Field, page 287
0x1E2	EGR2_PTP_FLOW_MATCH_LOWER	Lower Half of PTP/OAM Flow Match Field, page 287
0x1E3	EGR2_PTP_FLOW_MASK_UPPER	Upper Half of PTP/OAM Flow Match Mask, page 288
0x1E4	EGR2_PTP_FLOW_MASK_LOWER	Lower Half of PTP/OAM Flow Match Mask, page 288
0x1E5	EGR2_PTP_DOMAIN_RANGE	PTP/OAM Range Match, page 288
0x1E6	EGR2_PTP_ACTION	PTP Action Control, page 289
0x1E7	EGR2_PTP_ACTION_2	PTP Action Control 2, page 290
0x1E8	EGR2_PTP_ZERO_FIELD_CTL	Zero Field Control, page 290

4.39 Egress2 Ethernet Next Protocol Configuration Registers

This section provides information about the Ethernet next protocol configuration registers.

4.39.1 Ethernet Next Protocol

Short Name: EGR2_ETH1_NXT_PROTOCOL_A

Address: 0x00

Table 441 • **Ethernet Next Protocol Register**

Bit	Name	Description	Access	Default
2:0	EGR2_ETH1_NXT_COMPARATOR_A	Points to the next comparator block after this Ethernet block 0: Reserved 1: Ethernet comparator 2 2: IP/UDP/ACH comparator 1 3: IP/UDP/ACH comparator 2 4: MPLS comparator 5: PTP/OAM comparator 6,7: Reserved	R/W	0x0

4.39.2 VLAN TPID Configuration

Short Name: EGR2_ETH1_VLAN_TPID_CFG_A

Address: 0x01

Table 442 • **VLAN TPID Configuration Register**

Bit	Name	Description	Access	Default
31:16	EGR2_ETH1_VLAN_TPID_CFG_A	Configurable VLAN TPID (S or B-tag).	R/W	0x88A8

4.39.3 Ethernet Tag Mode

Short Name: EGR2_ETH1_TAG_MODE_A

Address: 0x02

Table 443 • **Ethernet Tag Mode Register**

Bit	Name	Description	Access	Default
0	EGR2_ETH1_PBB_ENA_A	This bit enables the presence of PBB. The I-tag match bits are programmed in the ETH1_VLAN_TAG_RANGE registers. The mask bits are programmed in the ETH1_VLAN_TAG2 registers. A B-tag if present is configured in the ETH1_VLAN_TAG1 registers. 0: PBB not enabled 1: Always expect PBB, last tag is always an I-tag	R/W	0x0

4.39.4 Ethertype Match

Short Name: EGR2_ETH1_ETYPE_MATCH_A

Address: 0x03

Table 444 • **Ethertype Match Register**

Bit	Name	Description	Access	Default
15:0	EGR2_ETH1_ETYPE_MATCH_A	If the Ethertype/length field is an Ethertype, then this register is compared against the value. If the field is a length, the length value is not checked.	R/W	0x0000

4.39.5 Ethernet Next Protocol

Short Name: EGR2_ETH1_NXT_PROTOCOL_B

Address: 0x10

Table 445 • **Ethernet Next Protocol Register**

Bit	Name	Description	Access	Default
2:0	EGR2_ETH1_NXT_COMPARATOR_B	Points to the next comparator block after this Ethernet block 0: Reserved 1: Ethernet comparator 2 2: IP/UDP/ACH comparator 1 3: IP/UDP/ACH comparator 2 4: MPLS comparator 5: PTP/OAM comparator 6,7: Reserved	R/W	0x0

4.39.6 VLAN TPID Configuration

Short Name: EGR2_ETH1_VLAN_TPID_CFG_B

Address: 0x11

Table 446 • **VLAN TPID Configuration Register**

Bit	Name	Description	Access	Default
15:0	EGR2_ETH1_VLAN_TPID_CFG_B	Configurable VLAN TPID (S or B-tag).	R/W	0x88A8

4.39.7 Ethernet Tag Mode

Short Name: EGR2_ETH1_TAG_MODE_B

Address: 0x12

Table 447 • **Ethernet Tag Mode Register**

Bit	Name	Description	Access	Default
0	EGR2_ETH1_PBB_ENA_B	This bit enables the presence of PBB. The I-tag match bits are programmed in the ETH1_VLAN_TAG_RANGE registers. The mask bits are programmed in the ETH1_VLAN_TAG2 registers. A B-tag if present is configured in the ETH1_VLAN_TAG1 registers. 0: PBB not enabled 1: Always expect PBB, last tag is always an I-tag	R/W	0x0

4.39.8 Ethertype Match

Short Name: EGR2_ETH1_ETYPE_MATCH_B

Address: 0x13

Table 448 • **Ethertype Match Register**

Bit	Name	Description	Access	Default
15:0	EGR2_ETH1_ETYPE_MATCH_B	If the Ethertype/length field is an Ethertype, then this register is compared against the value. If the field is a length, the length value is not checked.	R/W	0x0000

4.39.9

Instance offsets: 0x20 EGR2_ETH1_FLOW_CFG_0

0x30 EGR2_ETH1_FLOW_CFG_1

0x40 EGR2_ETH1_FLOW_CFG_2

0x50 EGR2_ETH1_FLOW_CFG_3

0x60 EGR2_ETH1_FLOW_CFG_4

0x70 EGR2_ETH1_FLOW_CFG_5

0x80 EGR2_ETH1_FLOW_CFG_6

0x90 EGR2_ETH1_FLOW_CFG_7

4.39.10 Ethernet Flow Enable

Short Name: EGR2_ETH1_FLOW_ENABLE

Addresses: 0x20 EGR2_ETH1_FLOW_CFG_0

0x30 EGR2_ETH1_FLOW_CFG_1

0x40 EGR2_ETH1_FLOW_CFG_2

0x50 EGR2_ETH1_FLOW_CFG_3

0x60 EGR2_ETH1_FLOW_CFG_4

0x70 EGR2_ETH1_FLOW_CFG_5

0x80 EGR2_ETH1_FLOW_CFG_6

0x90 EGR2_ETH1_FLOW_CFG_7

Table 449 • **Ethernet Flow Enable Register**

Bit	Name	Description	Access	Default
16	EGR2_ETH1_NXT_PROT_GRP_SEL	Indicates which next-protocol configuration group is valid with this flow 0: Associate this flow with next-protocol group A 1: Associate this flow with next-protocol group B	R/W	0x0
9:8	EGR2_ETH1_CHANNEL_MASK	0: Flow valid for channel 0 1: Flow valid for channel 1	R/W	0x3
0	EGR2_ETH1_FLOW_ENABLE	Flow enable 0: Flow is disabled 1: Flow is enabled	R/W	0x0

4.39.11 Ethernet Protocol Match Mode

Short Name: EGR2_ETH1_MATCH_MODE

Addresses: 0x21 EGR2_ETH1_FLOW_CFG_0

0x31 EGR2_ETH1_FLOW_CFG_1

0x41 EGR2_ETH1_FLOW_CFG_2

0x51 EGR2_ETH1_FLOW_CFG_3

0x61 EGR2_ETH1_FLOW_CFG_4

0x71 EGR2_ETH1_FLOW_CFG_5

0x81 EGR2_ETH1_FLOW_CFG_6

0x91 EGR2_ETH1_FLOW_CFG_7

Table 450 • **Ethernet Protocol Match Mode Register**

Bit	Name	Description	Access	Default
13:12	EGR2_ETH1_VLAN_TAG_MODE	0: VLAN range checking disabled 1: VLAN range checking on tag 1 2: VLAN range checking on tag 2 (not supported with PBB) 3: reserved	R/W	0x0
9	EGR2_ETH1_VLAN_TAG2_TYPE	This register is only used if ETH1_VLAN_VERIFY_ENA = 1 If PBB not enabled: 0: C tag (TPID of 0x8100) 1: S tag (match to CONF_VLAN_TPID) If PBB enabled: 0,1: I tag (use range registers)	R/W	0x1
8	EGR2_ETH1_VLAN_TAG1_TYPE	This register is only used if ETH1_VLAN_VERIFY_ENA = 1 0: C tag (TPID of 0x8100) 1: S or B tag (match to CONF_VLAN_TPID)	R/W	0x0
7:6	EGR2_ETH1_VLAN_TAGS	This register is only used if ETH1_VLAN_VERIFY_ENA = 1 0: No VLAN tags (not valid for PBB) 1: 1 VLAN tag (for PBB this would be the I-tag) 2: 2 VLAN tags (for PBB expect a B-tag and an I-tag) 3: Reserved	R/W	0x0
4	EGR2_ETH1_VLAN_VERIFY_ENA	0: Parse for VLAN tags, do not check values. For PBB the I-tag is always checked. 1: Verify configured VLAN tag configuration.	R/W	0x0
0	EGR2_ETH1_ETHERTYPE_MODE	When checking for presence of SNAP/LLC based upon ETH1_MATCH_MODE, this field indicates if SNAP & 3-byte LLC is expected to be present 0: Only Ethernet type II supported, no SNAP/LLC 1: Ethernet type II & Ethernet type I with SNAP/LLC, determine if SNAP/LLC is present or not. Type I always assumes that SNAP/LLC is present	R/W	0x0

4.39.12 Ethernet Address Match Part 1

Short Name: EGR2_ETH1_ADDR_MATCH_1

Addresses: 0x22 EGR2_ETH1_FLOW_CFG_0

0x32 EGR2_ETH1_FLOW_CFG_1

0x42 EGR2_ETH1_FLOW_CFG_2

0x52 EGR2_ETH1_FLOW_CFG_3

0x62 EGR2_ETH1_FLOW_CFG_4

0x72 EGR2_ETH1_FLOW_CFG_5

0x82 EGR2_ETH1_FLOW_CFG_6

0x92 EGR2_ETH1_FLOW_CFG_7

Table 451 • **Ethernet Address Match Part 1 Register**

Bit	Name	Description	Access	Default
31:0	EGR2_ETH1_ADDR_MATCH_1	First 32 bits of the address match value	R/W	0x00000000

4.39.13 Ethernet Address Match Part 2

Short Name: EGR2_ETH1_ADDR_MATCH_2

Addresses: 0x23 EGR2_ETH1_FLOW_CFG_0

0x33 EGR2_ETH1_FLOW_CFG_1

0x43 EGR2_ETH1_FLOW_CFG_2

0x53 EGR2_ETH1_FLOW_CFG_3

0x63 EGR2_ETH1_FLOW_CFG_4

0x73 EGR2_ETH1_FLOW_CFG_5

0x83 EGR2_ETH1_FLOW_CFG_6

0x93 EGR2_ETH1_FLOW_CFG_7

Table 452 • **Ethernet Address Match Part 2 Register**

Bit	Name	Description	Access	Default
22:20	EGR2_ETH1_ADDR_MATCH_MODE	Selects how the addresses are matched. Multiple bits can be set at once bit 0: Full 48-bit address match bit 1: Match any unicast address bit 2: Match any multicast address	R/W	0x1
17:16	EGR2_ETH1_ADDR_MATCH_SELECT	Selects which address to match 0: Match the Destination Address 1: Match the Source Address 2: Match either the Source of Destination Address 3: Reserved	R/W	0x0
15:0	EGR2_ETH1_ADDR_MATCH_2	Last 16 bits of the Ethernet address match field	R/W	0x0000

4.39.14 Ethernet VLAN Tag Range Match

Short Name: EGR2_ETH1_VLAN_TAG_RANGE_I_TAG

Addresses: 0x24 EGR2_ETH1_FLOW_CFG_0

0x34 EGR2_ETH1_FLOW_CFG_1

0x44 EGR2_ETH1_FLOW_CFG_2

0x54 EGR2_ETH1_FLOW_CFG_3

0x64 EGR2_ETH1_FLOW_CFG_4

0x74 EGR2_ETH1_FLOW_CFG_5

0x84 EGR2_ETH1_FLOW_CFG_6

0x94 EGR2_ETH1_FLOW_CFG_7

Table 453 • **Ethernet VLAN Tag Range Match Register**

Bit	Name	Description	Access	Default
27:16	EGR2_ETH1_VLAN_TAG_RANGE_UPPER	If PBB mode is not enabled, then this register contains the upper range of the VLAN tag range match. If PBB mode is enabled, then this register contains the upper 12 bits of the I-tag	R/W	0xFFF
11:0	EGR2_ETH1_VLAN_TAG_RANGE_LOWER	If PBB mode is not enabled, then this register contains the lower range of the VLAN tag range match. If PBB mode is enabled, then this register contains the lower 12 bits of the I-tag	R/W	0x000

4.39.15 VLAN Tag 1 Match/Mask

Short Name: EGR2_ETH1_VLAN_TAG1

Addresses: 0x25 EGR2_ETH1_FLOW_CFG_0

0x35 EGR2_ETH1_FLOW_CFG_1

0x45 EGR2_ETH1_FLOW_CFG_2

0x55 EGR2_ETH1_FLOW_CFG_3

0x65 EGR2_ETH1_FLOW_CFG_4

0x75 EGR2_ETH1_FLOW_CFG_5

0x85 EGR2_ETH1_FLOW_CFG_6

0x95 EGR2_ETH1_FLOW_CFG_7

Table 454 • **VLAN Tag 1 Match/Mask Register**

Bit	Name	Description	Access	Default
27:16	EGR2_ETH1_VLAN_TAG1_MASK	Mask value for VLAN tag 1	R/W	0xFFF
11:0	EGR2_ETH1_VLAN_TAG1_MATCH	Match value for the first VLAN tag	R/W	0x000

4.39.16 Match/Mask For VLAN Tag 2 or I-Tag Match

Short Name: EGR2_ETH1_VLAN_TAG2_I_TAG

Addresses: 0x26 EGR2_ETH1_FLOW_CFG_0

0x36 EGR2_ETH1_FLOW_CFG_1

0x46 EGR2_ETH1_FLOW_CFG_2

0x56 EGR2_ETH1_FLOW_CFG_3

0x66 EGR2_ETH1_FLOW_CFG_4

0x76 EGR2_ETH1_FLOW_CFG_5

0x86 EGR2_ETH1_FLOW_CFG_6

0x96 EGR2_ETH1_FLOW_CFG_7

Table 455 • **Match/Mask For VLAN Tag 2 or I-Tag Match Register**

Bit	Name	Description	Access	Default
27:16	EGR2_ETH1_VLAN_TAG2_MASK	When PBB is not enabled, the mask field for VLAN tag 2 When PBB is enabled, the upper 12 bits of the I-tag mask	R/W	0xFFF
11:0	EGR2_ETH1_VLAN_TAG2_MATCH	When PBB is not enabled, the match field for VLAN Tag 2 When PBB is enabled, the lower 12 bits of the I-tag mask field	R/W	0x000

4.39.17 Ethernet Next Protocol

Short Name: EGR2_ETH2_NXT_PROTOCOL_A

Address: 0xA0

Table 456 • **Ethernet Next Protocol Register**

Bit	Name	Description	Access	Default
2:0	EGR2_ETH2_NXT_COMPARATOR_A	Points to the next comparator block after this Ethernet block. If this comparator block is not used, this field must be set to 0. 0: Comparator block not used 1: Ethernet comparator 2 2: IP/UDP/ACH comparator 1 3: IP/UDP/ACH comparator 2 4: MPLS comparator 5: PTP/OAM comparator 6,7: Reserved	R/W	0x0

4.39.18 VLAN TPID Configuration

Short Name: EGR2_ETH2_VLAN_TPID_CFG_A

Address: 0xA1

Table 457 • **VLAN TPID Configuration Register**

Bit	Name	Description	Access	Default
31:16	EGR2_ETH2_VLAN_TPID_CFG_A	Configurable S-tag TPID	R/W	0x88A8

4.39.19 Ethertype Match

Short Name: EGR2_ETH2_ETYPE_MATCH_A

Address: 0xA2

Table 458 • **Ethertype Match Register**

Bit	Name	Description	Access	Default
15:0	EGR2_ETH2_ETYPE_MATCH_A	If the Ethertype/length field is an Ethertype, then this register is compared against the value. If the field is a length, the length value is not checked.	R/W	0x0000

4.39.20

Instance offsets: 0xC0 EGR2_ETH2_FLOW_CFG_0

0xD0 EGR2_ETH2_FLOW_CFG_1

0xE0 EGR2_ETH2_FLOW_CFG_2

0xF0 EGR2_ETH2_FLOW_CFG_3

0x100 EGR2_ETH2_FLOW_CFG_4

0x110 EGR2_ETH2_FLOW_CFG_5

0x120 EGR2_ETH2_FLOW_CFG_6

0x130 EGR2_ETH2_FLOW_CFG_7

4.39.21 Ethernet Flow Enable

Short Name: EGR2_ETH2_FLOW_ENABLE

Addresses: 0xC0 EGR2_ETH2_FLOW_CFG_0

0xD0 EGR2_ETH2_FLOW_CFG_1

0xE0 EGR2_ETH2_FLOW_CFG_2

0xF0 EGR2_ETH2_FLOW_CFG_3

0x100 EGR2_ETH2_FLOW_CFG_4

0x110 EGR2_ETH2_FLOW_CFG_5

0x120 EGR2_ETH2_FLOW_CFG_6

0x130 EGR2_ETH2_FLOW_CFG_7

Table 459 • **Ethernet Flow Enable Register**

Bit	Name	Description	Access	Default
9:8	EGR2_ETH2_CHANNEL_MASK	bit 0: Flow valid for channel 0 bit 1: Flow valid for channel 1	R/W	0x3
0	EGR2_ETH2_FLOW_ENABLE	Flow enable. If this comparator block is not used, all flow enable bits must be set to 0. 0: Flow is disabled 1: Flow is enabled	R/W	0x0

4.39.22 Ethernet Protocol Match Mode

Short Name: EGR2_ETH2_MATCH_MODE

Addresses: 0xC1 EGR2_ETH2_FLOW_CFG_0

0xD1 EGR2_ETH2_FLOW_CFG_1

0xE1 EGR2_ETH2_FLOW_CFG_2

0xF1 EGR2_ETH2_FLOW_CFG_3

0x101 EGR2_ETH2_FLOW_CFG_4

0x111 EGR2_ETH2_FLOW_CFG_5

0x121 EGR2_ETH2_FLOW_CFG_6

0x131 EGR2_ETH2_FLOW_CFG_7

Table 460 • **Ethernet Protocol Match Mode Register**

Bit	Name	Description	Access	Default
13:12	EGR2_ETH2_VLAN_TAG_MODE	0: VLAN range checking disabled 1: VLAN range checking on tag 1 2: VLAN range checking on tag 2 (not supported with PBB) 3: reserved	R/W	0x0
9	EGR2_ETH2_VLAN_TAG2_TYPE	This register is only used if ETH1_VLAN_VERIFY_ENA = 1 0: C tag (TPID of 0x8100) 1: S tag (match to CONF_VLAN_TPID)	R/W	0x1
8	EGR2_ETH2_VLAN_TAG1_TYPE	This register is only used if ETH1_VLAN_VERIFY_ENA = 1 0: C tag (TPID of 0x8100) 1: S or B tag (match to CONF_VLAN_TPID)	R/W	0x0
7:6	EGR2_ETH2_VLAN_TAGS	This register is only used if ETH2_VLAN_VERIFY_ENA = 1 0: No VLAN tags 1: 1 VLAN tag 2: 2 VLAN tags 3: Reserved	R/W	0x0
4	EGR2_ETH2_VLAN_VERIFY_ENA	0: Parse for VLAN tags, do not check values. 1: Verify configured VLAN tag configuration.	R/W	0x0
0	EGR2_ETH2_ETHERTYPE_MODE	When checking for presence of SNAP/LLC based upon ETH1_MATCH_MODE, this field indicates if SNAP & 3-byte LLC is expected to be present 0: Only Ethernet type II supported, no SNAP/LLC 1: Ethernet type II & Ethernet type I with SNAP/LLC, determine if SNAP/LLC is present or not. Type I always assumes that SNAP/LLC is present	R/W	0x0

4.39.23 Ethernet Address Match Part 1

Short Name: EGR2_ETH2_ADDR_MATCH_1

Addresses: 0xC2 EGR2_ETH2_FLOW_CFG_0

0xD2 EGR2_ETH2_FLOW_CFG_1

0xE2 EGR2_ETH2_FLOW_CFG_2

0xF2 EGR2_ETH2_FLOW_CFG_3

0x102 EGR2_ETH2_FLOW_CFG_4

0x112 EGR2_ETH2_FLOW_CFG_5

0x122 EGR2_ETH2_FLOW_CFG_6

0x132 EGR2_ETH2_FLOW_CFG_7

Table 461 • **Ethernet Address Match Part 1 Register**

Bit	Name	Description	Access	Default
31:0	EGR2_ETH2_ADDR_MATCH_1	First 32 bits of the address match value	R/W	0x00000000

4.39.24 Ethernet Address Match Part 2

Short Name: EGR2_ETH2_ADDR_MATCH_2

Addresses: 0xC3 EGR2_ETH2_FLOW_CFG_0

0xD3 EGR2_ETH2_FLOW_CFG_1

0xE3 EGR2_ETH2_FLOW_CFG_2

0xF3 EGR2_ETH2_FLOW_CFG_3

0x103 EGR2_ETH2_FLOW_CFG_4

0x113 EGR2_ETH2_FLOW_CFG_5

0x123 EGR2_ETH2_FLOW_CFG_6

0x133 EGR2_ETH2_FLOW_CFG_7

Table 462 • **Ethernet Address Match Part 2 Register**

Bit	Name	Description	Access	Default
22:20	EGR2_ETH2_ADDR_MATCH_MODE	Selects how the addresses are matched. Multiple bits can be set at once bit 0: Full 48-bit address match bit 1: Match any unicast address bit 2: Match any multicast address	R/W	0x1
17:16	EGR2_ETH2_ADDR_MATCH_SELECT	Selects which address to match 0: Match the Destination Address 1: Match the Source Address 2: Match either the Source of Destination Address 3: Reserved	R/W	0x0
15:0	EGR2_ETH2_ADDR_MATCH_2	Last 16 bits of the Ethernet address match field	R/W	0x0000

4.39.25 Ethernet VLAN Tag Range Match

Short Name: EGR2_ETH2_VLAN_TAG_RANGE_I_TAG

Addresses: 0xC4 EGR2_ETH2_FLOW_CFG_0

0xD4 EGR2_ETH2_FLOW_CFG_1

0xE4 EGR2_ETH2_FLOW_CFG_2

0xF4 EGR2_ETH2_FLOW_CFG_3

0x104 EGR2_ETH2_FLOW_CFG_4

0x114 EGR2_ETH2_FLOW_CFG_5

0x124 EGR2_ETH2_FLOW_CFG_6

0x134 EGR2_ETH2_FLOW_CFG_7

Table 463 • **Ethernet VLAN Tag Range Match Register**

Bit	Name	Description	Access	Default
27:16	EGR2_ETH2_VLAN_TAG_RANGE_UPPER	This register contains the upper range of the VLAN tag range match.	R/W	0xFFF
11:0	EGR2_ETH2_VLAN_TAG_RANGE_LOWER	This register contains the lower range of the VLAN tag range match.	R/W	0x000

4.39.26 VLAN Tag 1 Match/Mask

Short Name: EGR2_ETH2_VLAN_TAG1

Addresses: 0xC5 EGR2_ETH2_FLOW_CFG_0

0xD5 EGR2_ETH2_FLOW_CFG_1

0xE5 EGR2_ETH2_FLOW_CFG_2

0xF5 EGR2_ETH2_FLOW_CFG_3

0x105 EGR2_ETH2_FLOW_CFG_4

0x115 EGR2_ETH2_FLOW_CFG_5

0x125 EGR2_ETH2_FLOW_CFG_6

0x135 EGR2_ETH2_FLOW_CFG_7

Table 464 • **VLAN Tag 1 Match/Mask Register**

Bit	Name	Description	Access	Default
27:16	EGR2_ETH2_VLAN_TAG1_MASK	Mask value for VLAN tag 1	R/W	0xFFF
11:0	EGR2_ETH2_VLAN_TAG1_MATCH	Match value for the first VLAN tag	R/W	0x000

4.39.27 Match/Mask For VLAN Tag 2 or I-Tag Match

Short Name: EGR2_ETH2_VLAN_TAG2_I_TAG

Addresses: 0xC6 EGR2_ETH2_FLOW_CFG_0

0xD6 EGR2_ETH2_FLOW_CFG_1

0xE6 EGR2_ETH2_FLOW_CFG_2

0xF6 EGR2_ETH2_FLOW_CFG_3

0x106 EGR2_ETH2_FLOW_CFG_4

0x116 EGR2_ETH2_FLOW_CFG_5

0x126 EGR2_ETH2_FLOW_CFG_6

0x136 EGR2_ETH2_FLOW_CFG_7

Table 465 • **Match/Mask For VLAN Tag 2 or I-Tag Match Register**

Bit	Name	Description	Access	Default
27:16	EGR2_ETH2_VLAN_TAG2_MASK	Mask field for VLAN tag 2	R/W	0xFFF
11:0	EGR2_ETH2_VLAN_TAG2_MATCH	Match field for VLAN Tag 2	R/W	0x000

4.40 Egress2 MPLS Next Protocol Registers

This section provides information about the MPLS next protocol registers.

4.40.1 MPLS Next Protocol Comparator

Short Name: EGR2_MPLS_NXT_COMPARATOR_A

Address: 0x140

Table 466 • **MPLS Next Protocol Comparator Register**

Bit	Name	Description	Access	Default
16	EGR2_MPLS_CTL_WORD_A	Indicates the presence of a control word after the last label 0: There is no control ward after the last label 1: There is a control word after the last label	R/W	0x0
2:0	EGR2_MPLS_NXT_COMPARATOR_A	Points to the next comparator stage. If this comparator block is not used, this field must be set to 0. 0: Comparator block not used 1: Ethernet comparator 2 2: IP/UDP/ACH comparator 1 3: IP/UDP/ACH comparator 2 4: Reserved 5: PTP/OAM comparator 6,7: Reserved	R/W	0x0

4.40.2

Instance offsets: 0x160 EGR2_MPLS_FLOW_CFG_0

0x170 EGR2_MPLS_FLOW_CFG_1

0x180 EGR2_MPLS_FLOW_CFG_2

0x190 EGR2_MPLS_FLOW_CFG_3

0x1A0 EGR2_MPLS_FLOW_CFG_4

0x1B0 EGR2_MPLS_FLOW_CFG_5

0x1C0 EGR2_MPLS_FLOW_CFG_6

0x1D0 EGR2_MPLS_FLOW_CFG_7

4.40.3 MPLS Flow Control

Short Name: EGR2_MPLS_FLOW_CONTROL

Addresses: 0x160 EGR2_MPLS_FLOW_CFG_0

0x170 EGR2_MPLS_FLOW_CFG_1

0x180 EGR2_MPLS_FLOW_CFG_2

0x190 EGR2_MPLS_FLOW_CFG_3

0x1A0 EGR2_MPLS_FLOW_CFG_4

0x1B0 EGR2_MPLS_FLOW_CFG_5

0x1C0 EGR2_MPLS_FLOW_CFG_6

0x1D0 EGR2_MPLS_FLOW_CFG_7

Table 467 • **MPLS Flow Control Register**

Bit	Name	Description	Access	Default
25:24	EGR2_MPLS_CHANNEL_MASK	0: Flow valid for channel 0 1: Flow valid for channel 1	R/W	0x3
19:16	EGR2_MPLS_STACK_DEPTH	Defines the allowable stack depths for searches. The direction that the stack is referenced is determined by the setting of MPLS_REF_PNT The following table maps bits to stack depths: 0: stack allowed to be 1 label deep 1: stack allowed to be 2 labels deep 2: stack allowed to be 3 labels deep 3: stack allowed to be 4 labels deep	R/W	0x0
4	EGR2_MPLS_REF_PNT	Defines the search direction for label matching 0: All searching is performed starting from the top of the stack 1: All searching is performed from the end of the stack	R/W	0x0
0	EGR2_MPLS_FLOW_ENA	Flow enable. If this comparator block is not used, all flow enable bits must be set to 0. 0: Flow is disabled 1: Flow is enabled	R/W	0x0

4.40.4 MPLS Label 0 Match Range Lower Value

Short Name: EGR2_MPLS_LABEL_RANGE_LOWER_0

Addresses: 0x161 EGR2_MPLS_FLOW_CFG_0

0x171 EGR2_MPLS_FLOW_CFG_1

0x181 EGR2_MPLS_FLOW_CFG_2

0x191 EGR2_MPLS_FLOW_CFG_3

0x1A1 EGR2_MPLS_FLOW_CFG_4

0x1B1 EGR2_MPLS_FLOW_CFG_5

0x1C1 EGR2_MPLS_FLOW_CFG_6

0x1D1 EGR2_MPLS_FLOW_CFG_7

Table 468 • **MPLS Label 0 Match Range Lower Value Register**

Bit	Name	Description	Access	Default
19:0	EGR2_MPLS_LABEL_RANGE_LOWER_0	Lower value for label 0 match range	R/W	0x00000

4.40.5 MPLS Label 0 Match Range Lower Value

Short Name: EGR2_MPLS_LABEL_RANGE_UPPER_0

Addresses: 0x162 EGR2_MPLS_FLOW_CFG_0

0x172 EGR2_MPLS_FLOW_CFG_1

0x182 EGR2_MPLS_FLOW_CFG_2

0x192 EGR2_MPLS_FLOW_CFG_3

0x1A2 EGR2_MPLS_FLOW_CFG_4

0x1B2 EGR2_MPLS_FLOW_CFG_5

0x1C2 EGR2_MPLS_FLOW_CFG_6

0x1D2 EGR2_MPLS_FLOW_CFG_7

Table 469 • **MPLS Label 0 Match Range Lower Value Register**

Bit	Name	Description	Access	Default
19:0	EGR2_MPLS_LABEL_RANGE_UPPER_0	Upper value for label 0 match range	R/W	0xFFFFF

4.40.6 MPLS Label 1 Match Range Lower Value

Short Name: EGR2_MPLS_LABEL_RANGE_LOWER_1

Addresses: 0x163 EGR2_MPLS_FLOW_CFG_0

0x173 EGR2_MPLS_FLOW_CFG_1

0x183 EGR2_MPLS_FLOW_CFG_2

0x193 EGR2_MPLS_FLOW_CFG_3

0x1A3 EGR2_MPLS_FLOW_CFG_4

0x1B3 EGR2_MPLS_FLOW_CFG_5

0x1C3 EGR2_MPLS_FLOW_CFG_6

0x1D3 EGR2_MPLS_FLOW_CFG_7

Table 470 • **MPLS Label 1 Match Range Lower Value Register**

Bit	Name	Description	Access	Default
19:0	EGR2_MPLS_LABEL_RANGE_LOWER_1	Lower value for label 1 match range	R/W	0x00000

4.40.7 MPLS Label 1 Match Range Lower Value

Short Name: EGR2_MPLS_LABEL_RANGE_UPPER_1

Addresses: 0x164 EGR2_MPLS_FLOW_CFG_0

0x174 EGR2_MPLS_FLOW_CFG_1

0x184 EGR2_MPLS_FLOW_CFG_2

0x194 EGR2_MPLS_FLOW_CFG_3

0x1A4 EGR2_MPLS_FLOW_CFG_4

0x1B4 EGR2_MPLS_FLOW_CFG_5

0x1C4 EGR2_MPLS_FLOW_CFG_6

0x1D4 EGR2_MPLS_FLOW_CFG_7

Table 471 • **MPLS Label 1 Match Range Lower Value Register**

Bit	Name	Description	Access	Default
19:0	EGR2_MPLS_LABEL_RANGE_UPPER_1	Upper value for label 1 match range	R/W	0xFFFFF

4.40.8 MPLS Label 2 Match Range Lower Value

Short Name: EGR2_MPLS_LABEL_RANGE_LOWER_2

Addresses: 0x165 EGR2_MPLS_FLOW_CFG_0

0x175 EGR2_MPLS_FLOW_CFG_1

0x185 EGR2_MPLS_FLOW_CFG_2

0x195 EGR2_MPLS_FLOW_CFG_3

0x1A5 EGR2_MPLS_FLOW_CFG_4

0x1B5 EGR2_MPLS_FLOW_CFG_5

0x1C5 EGR2_MPLS_FLOW_CFG_6

0x1D5 EGR2_MPLS_FLOW_CFG_7

Table 472 • **MPLS Label 2 Match Range Lower Value Register**

Bit	Name	Description	Access	Default
19:0	EGR2_MPLS_LABEL_RANGE_LOWER_2	Lower value for label 2 match range	R/W	0x00000

4.40.9 MPLS Label 2 Match Range Lower Value

Short Name: EGR2_MPLS_LABEL_RANGE_UPPER_2

Addresses: 0x166 EGR2_MPLS_FLOW_CFG_0

0x176 EGR2_MPLS_FLOW_CFG_1

0x186 EGR2_MPLS_FLOW_CFG_2

0x196 EGR2_MPLS_FLOW_CFG_3

0x1A6 EGR2_MPLS_FLOW_CFG_4

0x1B6 EGR2_MPLS_FLOW_CFG_5

0x1C6 EGR2_MPLS_FLOW_CFG_6

0x1D6 EGR2_MPLS_FLOW_CFG_7

Table 473 • **MPLS Label 2 Match Range Lower Value Register**

Bit	Name	Description	Access	Default
19:0	EGR2_MPLS_LABEL_RANGE_UPPER_2	Upper value for label 2 match range	R/W	0xFFFFF

4.40.10 MPLS Label 3 Match Range Lower Value

Short Name: EGR2_MPLS_LABEL_RANGE_LOWER_3

Addresses: 0x167 EGR2_MPLS_FLOW_CFG_0

0x177 EGR2_MPLS_FLOW_CFG_1

0x187 EGR2_MPLS_FLOW_CFG_2

0x197 EGR2_MPLS_FLOW_CFG_3

0x1A7 EGR2_MPLS_FLOW_CFG_4

0x1B7 EGR2_MPLS_FLOW_CFG_5

0x1C7 EGR2_MPLS_FLOW_CFG_6

0x1D7 EGR2_MPLS_FLOW_CFG_7

Table 474 • **MPLS Label 3 Match Range Lower Value Register**

Bit	Name	Description	Access	Default
19:0	EGR2_MPLS_LABEL_RANGE_LOWER_3	Lower value for label 3 match range	R/W	0x00000

4.40.11 MPLS Label 3 Match Range Lower Value

Short Name: EGR2_MPLS_LABEL_RANGE_UPPER_3

Addresses: 0x168 EGR2_MPLS_FLOW_CFG_0

0x178 EGR2_MPLS_FLOW_CFG_1

0x188 EGR2_MPLS_FLOW_CFG_2

0x198 EGR2_MPLS_FLOW_CFG_3

0x1A8 EGR2_MPLS_FLOW_CFG_4

0x1B8 EGR2_MPLS_FLOW_CFG_5

0x1C8 EGR2_MPLS_FLOW_CFG_6

0x1D8 EGR2_MPLS_FLOW_CFG_7

Table 475 • **MPLS Label 3 Match Range Lower Value Register**

Bit	Name	Description	Access	Default
19:0	EGR2_MPLS_LABEL_RANGE_UPPER_3	Upper value for label 3 match range	R/W	0xFFFFF

4.40.12

Instance offsets: 0x1E0 EGR2_PTP_FLOW_0

0x1F0 EGR2_PTP_FLOW_1

0x200 EGR2_PTP_FLOW_2

0x210 EGR2_PTP_FLOW_3

0x220 EGR2_PTP_FLOW_4

0x230 EGR2_PTP_FLOW_5

4.40.13 PTP/OAM Flow Enable

Short Name: EGR2_PTP_FLOW_ENA

Addresses: 0x1E0 EGR2_PTP_FLOW_0

0x1F0 EGR2_PTP_FLOW_1

0x200 EGR2_PTP_FLOW_2

0x210 EGR2_PTP_FLOW_3

0x220 EGR2_PTP_FLOW_4

0x230 EGR2_PTP_FLOW_5

Table 476 • **PTP/OAM Flow Enable Register**

Bit	Name	Description	Access	Default
17:16	EGR2_PTP_NXT_PROT_GRP_MASK	Indicates which next protocol groups that this flow is valid for. For each next protocol group, if the bit is 1, then this flow is valid for that group. If it is 0, then it is not valid for the group. bit 0: Mask bit for next protocol group A bit 1: Mask bit for next protocol group B	R/W	0x3
5:4	EGR2_PTP_CHANNEL_MASK	bit 0: Flow valid for channel 0 bit 1: Flow valid for channel 1	R/W	0x3
0	EGR2_PTP_FLOW_ENA		R/W	0x0

4.40.14 Upper Half of PTP/OAM Flow Match Field

Short Name: EGR2_PTP_FLOW_MATCH_UPPER

Addresses: 0x1E1 EGR2_PTP_FLOW_0

0x1F1 EGR2_PTP_FLOW_1

0x201 EGR2_PTP_FLOW_2

0x211 EGR2_PTP_FLOW_3

0x221 EGR2_PTP_FLOW_4

0x231 EGR2_PTP_FLOW_5

Table 477 • **Upper Half of PTP/OAM Flow Match Field Register**

Bit	Name	Description	Access	Default
31:0	EGR2_PTP_FLOW_MATCH_UPPER		R/W	0x00000000

4.40.15 Lower Half of PTP/OAM Flow Match Field

Short Name: EGR2_PTP_FLOW_MATCH_LOWER

Addresses: 0x1E2 EGR2_PTP_FLOW_0

0x1F2 EGR2_PTP_FLOW_1

0x202 EGR2_PTP_FLOW_2

0x212 EGR2_PTP_FLOW_3

0x222 EGR2_PTP_FLOW_4

0x232 EGR2_PTP_FLOW_5

Table 478 • **Lower Half of PTP/OAM Flow Match Field Register**

Bit	Name	Description	Access	Default
31:0	EGR2_PTP_FLOW_MATCH_LOWER		R/W	0x00000000

4.40.16 Upper Half of PTP/OAM Flow Match Mask

Short Name: EGR2_PTP_FLOW_MASK_UPPER

Addresses: 0x1E3 EGR2_PTP_FLOW_0

0x1F3 EGR2_PTP_FLOW_1

0x203 EGR2_PTP_FLOW_2

0x213 EGR2_PTP_FLOW_3

0x223 EGR2_PTP_FLOW_4

0x233 EGR2_PTP_FLOW_5

Table 479 • **Upper Half of PTP/OAM Flow Match Mask Register**

Bit	Name	Description	Access	Default
31:0	EGR2_PTP_FLOW_MASK_UPPER		R/W	0x00000000

4.40.17 Lower Half of PTP/OAM Flow Match Mask

Short Name: EGR2_PTP_FLOW_MASK_LOWER

Addresses: 0x1E4 EGR2_PTP_FLOW_0

0x1F4 EGR2_PTP_FLOW_1

0x204 EGR2_PTP_FLOW_2

0x214 EGR2_PTP_FLOW_3

0x224 EGR2_PTP_FLOW_4

0x234 EGR2_PTP_FLOW_5

Table 480 • **Lower Half of PTP/OAM Flow Match Mask Register**

Bit	Name	Description	Access	Default
31:0	EGR2_PTP_FLOW_MASK_LOWER		R/W	0x00000000

4.40.18 PTP/OAM Range Match

Short Name: EGR2_PTP_DOMAIN_RANGE

Addresses: 0x1E5 EGR2_PTP_FLOW_0

0x1F5 EGR2_PTP_FLOW_1

0x205 EGR2_PTP_FLOW_2

0x215 EGR2_PTP_FLOW_3

0x225 EGR2_PTP_FLOW_4

0x235 EGR2_PTP_FLOW_5

Table 481 • **PTP/OAM Range Match Register**

Bit	Name	Description	Access	Default
28:24	EGR2_PTP_DOMAIN_RANGE_OFFSET		R/W	0x00

Table 481 • **PTP/OAM Range Match Register *(continued)***

Bit	Name	Description	Access	Default
23:16	EGR2_PTP_DOMAIN_RANGE_UPPER		R/W	0xFF
15:8	EGR2_PTP_DOMAIN_RANGE_LOWER		R/W	0x00
0	EGR2_PTP_DOMAIN_RANGE_ENA		R/W	0x0

4.40.19 PTP Action Control

Short Name: EGR2_PTP_ACTION

Addresses: 0x1E6 EGR2_PTP_FLOW_0

0x1F6 EGR2_PTP_FLOW_1

0x206 EGR2_PTP_FLOW_2

0x216 EGR2_PTP_FLOW_3

0x226 EGR2_PTP_FLOW_4

0x236 EGR2_PTP_FLOW_5

Table 482 • **PTP Action Control Register**

Bit	Name	Description	Access	Default
28	EGR2_PTP_MOD_FRAME_STAT_UPDATE	1: Tell the Rewriter to update the value of the Modified Frame Status bit 0: Do not update the bit	R/W	0x0
26:24	EGR2_PTP_MOD_FRAME_BYTE_OFFSET	Indicates the position relative to the start of the PTP frame in bytes where the Modified_Frame_Status bit resides	R/W	0x0
21	EGR2_PTP_SUB_DELAY_ASYM_ENA	1: Signal the Timestamp block to subtract the asymmetry delay 0: Do not signal the Timestamp block to subtract the asymmetry delay	R/W	0x0
20	EGR2_PTP_ADD_DELAY_ASYM_ENA	1: Signal the Timestamp block to add the asymmetry delay 0: Do not signal the Timestamp block to add the asymmetry delay	R/W	0x0
15:10	EGR2_PTP_TIME_STRG_FIELD_OFFSET	Points to the reserved 32-bit field where the Rx timestamp is saved. The location is relative to the first byte of the PTP/OAM header.	R/W	0x00
9:5	EGR2_PTP_CORR_FIELD_OFFSET	Points to the location of the correction field for updating the timestamp. Location is relative to the first byte of the PTP/OAM header. **Note:** If this flow is being used to match OAM frames, set this register to 4	R/W	0x00
4	EGR2_PTP_SAVE_LOCAL_TIME	1: Save the local time to the Timestamp FIFO 0: Do not save the time to the Timestamp FIFO	R/W	0x0

Table 482 • **PTP Action Control Register *(continued)***

Bit	Name	Description	Access	Default
3:0	EGR2_PTP_COMMAND	0: NoP 1: SUB 2: SUB_P2P 3: ADD 4: SUB_ADD 5: WRITE_1588 6: WRITE_P2P (deprecated) 7: WRITE_NS 8: WRITE_NS_P2P	R/W	0x0

4.40.20 PTP Action Control 2

Short Name: EGR2_PTP_ACTION_2

Addresses: 0x1E7 EGR2_PTP_FLOW_0

0x1F7 EGR2_PTP_FLOW_1

0x207 EGR2_PTP_FLOW_2

0x217 EGR2_PTP_FLOW_3

0x227 EGR2_PTP_FLOW_4

0x237 EGR2_PTP_FLOW_5

Table 483 • **PTP Action Control 2 Register**

Bit	Name	Description	Access	Default
23:16	EGR2_PTP_NEW_CF_LOC	Location of the new correction field relative to the PTP header start. Only even values are allowed.	R/W	0x00
15:8	EGR2_PTP_REWRITE_OFFSET	Byte offset relative to the start of the PTP frame where the ingress timestamp value can be read.	R/W	0x00
3:0	EGR2_PTP_REWRITE_BYTES	Number of bytes in the PTP or OAM frame that must be modified by the Rewriter for the timestamp	R/W	0x0

4.40.21 Zero Field Control

Short Name: EGR2_PTP_ZERO_FIELD_CTL

Addresses: 0x1E8 EGR2_PTP_FLOW_0

0x1F8 EGR2_PTP_FLOW_1

0x208 EGR2_PTP_FLOW_2

0x218 EGR2_PTP_FLOW_3

0x228 EGR2_PTP_FLOW_4

Table 484 • **Zero Field Control Register**

Bit	Name	Description	Access	Default
13:8	EGR2_PTP_ZERO_FIELD_OFFSET	Points to a location in the PTP/OAM frame relative to the start of the PTP header that will be zeroed if this function is enabled	R/W	0x00
3:0	EGR2_PTP_ZERO_FIELD_BYTE_CNT	The number of bytes to be zeroed. If this field is 0, then this function is not enabled.	R/W	0x0

5 Electrical Specifications

This section provides the DC characteristics, AC characteristics, recommended operating conditions, and stress ratings for the VSC8572-02 device.

5.1 DC Characteristics

This section contains the DC specifications for the VSC8572-02 device.

5.1.1 VDD25

The following table shows the DC specifications for the pins referenced to VDD25. The specifications listed in the following table are valid only when V_{DD1} = 1.0 V, V_{DD1A} = 1.0 V, or V_{DD25A} = 2.5 V.

Table 485 • **VDD25 DC Characteristics**

Parameter	Symbol	Minimum	Maximum	Unit	Condition
Output high voltage	V_{OH}	2.0		V	IOH = –1.0 mA
Output low voltage	V_{OL}		0.4	V	IOL = 1.0 mA
Input high voltage	V_{IH}	1.85	3.3	V	
Input low voltage	V_{IL}	–0.3	0.7	V	
Input leakage current	I_{ILEAK}	–32	32	µA	Internal resistor included
Output leakage current	I_{OLEAK}	–32	32	µA	Internal resistor included
Output low current drive strength	I_{OL}		6	mA	
Output high current drive strength	I_{OH}	–6		mA	

5.1.2 LED and GPIO

The following table shows the DC specifications for the LED and GPIO pins.

Table 486 • **LED and GPIO Characteristics**

Pin	Symbol	Minimum	Maximum	Unit
LED	I_{OH}		24	mA
LED	I_{OL}	–24		mA
GPIO	I_{OH}		12	mA
GPIO	I_{OL}	–12		mA

5.1.3 Internal Pull-Up or Pull-Down Resistors

Internal pull-up or pull-down resistors are specified in the following table. For more information about signals with internal pull-up or pull-down resistors, see Pins by Function, page 316.

All internal pull-up resistors are connected to their respective I/O supply.

Table 487 • **Internal Pull-Up or Pull-Down Resistors**

Parameter	Symbol	Minimum	Typical	Maximum	Unit
Internal pull-up resistor, GPIO	R_{PU_GPIO}	33	53	90	kΩ

Table 487 • **Internal Pull-Up or Pull-Down Resistors *(continued)***

Parameter	Symbol	Minimum	Typical	Maximum	Unit
Internal pull-up resistor, all others	R_{PU}	96	120	144	kΩ
Internal pull-down resistor	R_{PD}	96	120	144	kΩ

5.1.4 Reference Clock

The following table shows the DC specifications for a differential reference clock input signal

Table 488 • **Reference Clock DC Characteristics**

Parameter	Symbol	Minimum	Typical	Maximum	Unit
Input voltage range	V_{IP},V_{IN}	–25		1260	mV
Input differential peak-to-peak voltage	$\|V_{ID}\|$	150[1]		1200	mV
Input common-mode voltage	V_{ICM}	0		1200[2]	mV
Differential input impedance	R_I		100		Ω

1. To meet jitter specifications, the minimum $|V_{ID}|$ must be 400 mV. When using a single-ended clock input, the REFCLK_P low voltage must be less than V_{DDA} – 200 mV, and the high voltage level must be greater than V_{DDA} + 200 mV
2. The maximum common-mode voltage is provided without a differential signal. The common-mode voltage is only limited by the maximum and minimum input voltage range and by the differential amplitude of the input signal.

5.1.5 1588 Reference Clock

The following table shows the DC specifications for a differential 1588 reference clock input signal.

Table 489 • **1588 Reference Clock DC Characteristics**

Parameter	Symbol	Minimum	Typical	Maximum	Unit
Input voltage range	V_{IP},V_{IN}	–25		1260	mV
Input differential peak-to-peak voltage	$\|V_{ID}\|$	150		1200	mV
Input common-mode voltage	V_{ICM}	0		1200[1]	mV
Differential input impedance	R_I		100		Ω

1. The maximum common-mode voltage is provided without a differential signal. The common-mode voltage is only limited by the maximum and minimum input voltage range and by the differential amplitude of the input signal.

5.1.6 SerDes Interface (SGMII)

The SerDes output drivers are designed to operate in SGMII/LVDS mode. The SGMII/LVDS mode meets or exceeds the DC requirements of Serial-GMII Specification Revision 1.9 (ENG-46158), unless otherwise noted. The following table lists the DC specifications for the SGMII driver. The values are valid for all configurations, unless stated otherwise.

Table 490 • **SerDes Driver DC Specifications**

Parameter	Symbol	Minimum	Maximum	Unit	Condition
Output high voltage, V_{OA} or V_{OB}	V_{OH}		1050	mV	R_L = 100 Ω ±1%
Output low voltage, V_{OA} or V_{OB}	V_{OL}	0		mV	R_L = 100 Ω ±1%

Table 490 • **SerDes Driver DC Specifications *(continued)***

Parameter	Symbol	Minimum	Maximum	Unit	Condition
Output differential peak voltage	$\|V_{OD}\|$	350	450	mV	V_{DD_VS} = 1.0 V R_L = 100 Ω ±1%
Output differential peak voltage, fiber media 1000BASE-X	$\|V_{OD}\|$	350	450	mV	V_{DD_VS} = 1.0 V R_L = 100 Ω ±1%
Output offset voltage(1)	V_{OS}	420	580	mV	V_{DD_VS} = 1.0 V R_L = 100 Ω ±1%
DC output impedance, single-ended, SGMII mode	R_O	40	140	Ω	V_C = 1.0 V See Figure 81, page 295
R_O mismatch between A and B, SGMII mode(2)	ΔR_O		10	%	V_C = 1.0 V See Figure 81, page 295
Change in $\|V_{OD}\|$ between 0 and 1, SGMII mode	$\Delta\|V_{OD}\|$		25	mV	R_L = 100 Ω ±1%
Change in V_{OS} between 0 and 1, SGMII mode	ΔV_{OS}		25	mV	R_L = 100 Ω ±1%
Output current, driver shorted to GND, SGMII mode	$\|I_{OSA}\|$, $\|I_{OSB}\|$		40	mA	
Output current, drivers shorted together, SGMII mode	$\|I_{OSAB}\|$		12	mA	

1. Requires AC-coupling for SGMII compliance.
2. Matching of reflection coefficients. For more information about test methods, see IEEE Std 1596.3-1996.

Figure 79 • **SGMII DC Transmit Test Circuit**

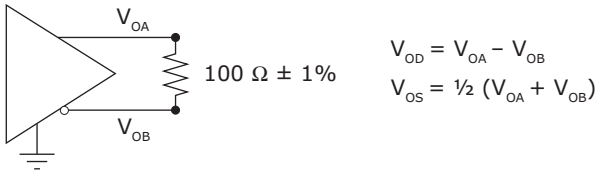
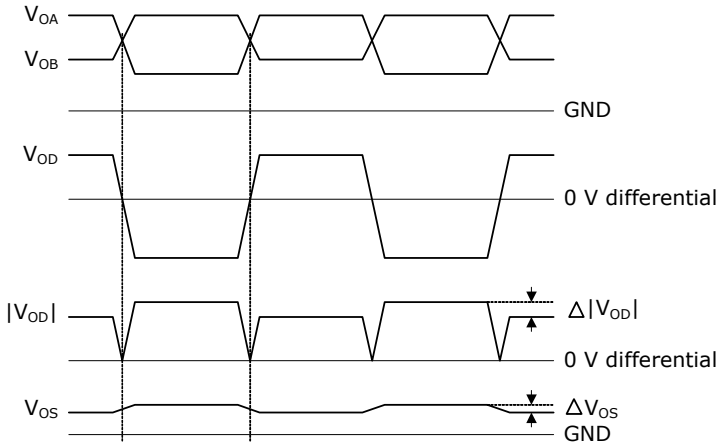


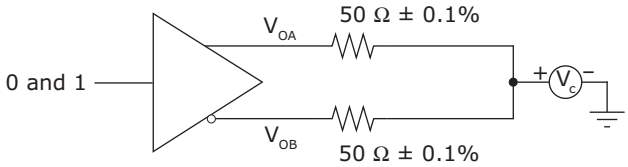
Figure 80 • **SGMII DC Definitions**



$\Delta|V_{OD}| = | \, |V_{OAH} - V_{OBL}| - |V_{OBH} - V_{OAL}| \, |$

$$\Delta V_{OS} = | \frac{1}{2}(V_{OAH} + V_{OBL}) - \frac{1}{2}(V_{OAL} + V_{OBH}) |$$

Figure 81 • **SGMII DC Driver Output Impedance Test Circuit**



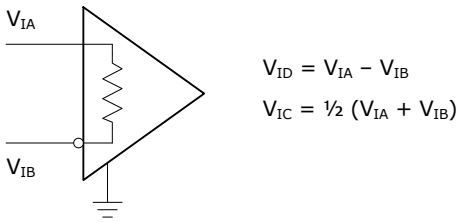
The following table lists the DC specifications for the SGMII receivers.

Table 491 • **SerDes Receiver DC Specifications**

Parameter	Symbol	Minimum	Maximum	Unit	Condition
Input voltage range, V_{IA} or V_{IB}	V_I	–25	1250	mV	
Input differential peak-to-peak voltage	$\|V_{ID}\|$	100	1000	mV	
Input common-mode voltage(1)	V_{ICM}	0	V_{DD_A}(2)	mV	Without any differential signal
Receiver differential input impedance	R_I	80	120	Ω	
Input differential hysteresis, SGMII mode	V_{HYST}	25		mV	

1. SGMII compliancy requires external AC-coupling. When interfacing with specific Microsemi devices, DC-coupling is possible. For more information, contact your local Microsemi sales representative.
2. The common-mode voltage is only limited by the maximum and minimum input voltage range and the input signal's differential amplitude.

Figure 82 • **SGMII DC Input Definitions**



5.1.7 Enhanced SerDes Interface (QSGMII)

All DC specifications for the enhanced SerDes interface are compliant with QSGMII Specification Revision 1.3 and meet or exceed the requirements in the standard. They are also compliant with OIF-CEI-02.0 requirements where applicable.

The enhanced SerDes interface supports the following operating modes: SGMII, QSGMII, and SFP. The values in the following table apply to the modes specified in the condition column.

The following table shows the DC specifications for the enhanced SerDes driver.

Table 492 • **Enhanced SerDes Driver DC Specifications**

Parameter	Symbol	Minimum	Maximum	Unit	Condition
Output differential peak voltage, SFP and QSGMII modes	$\|V_{ODp}\|$	250	400	mV	V_{DD_VS} = 1.0 V R_L = 100 Ω ±1% maximum drive

Table 492 • **Enhanced SerDes Driver DC Specifications *(continued)***

Parameter	Symbol	Minimum	Maximum	Unit	Condition
Output differential peak voltage, SGMII mode(1)	$\|V_{ODp}\|$	150	400	mV	V_{DD_VS} = 1.0 V R_L = 100 Ω ±1%
DC output impedance, single-ended, SGMII mode	R_O	40	140	Ω	V_C = 1.0 V See Figure 81, page 295
R_O mismatch between A and B, SGMII mode(2)	ΔR_O		10	%	V_C = 1.0 V See Figure 81, page 295
Change in $\|V_{OD}\|$ between 0 and 1, SGMII mode	$\Delta\|V_{OD}\|$		25	mV	R_L = 100 Ω ±1%
Change in V_{OS} between 0 and 1, SGMII mode	ΔV_{OS}		25	mV	R_L = 100 Ω ±1%
Output current, drivers shorted to ground, SGMII and QSGMII modes	$\|I_{OSA}\|$, $\|I_{OSB}\|$		40	mA	
Output current, drivers shorted together, SGMII and QSGMII modes	$\|I_{OSAB}\|$		12	mA	

1. Voltage is adjustable in 64 steps.
2. Matching of reflection coefficients. For more information about test methods, see IEEE Std 1596.3-1996.

The following table lists the DC specifications for the enhanced SerDes receiver.

Table 493 • **Enhanced SerDes Receiver DC Specifications**

Parameter	Symbol	Minimum	Typical	Maximum	Unit
Input voltage range, V_{IA} or V_{IB}(1)	V_I	–0.25		1.2	V
Input differential peak-to-peak voltage	$\|V_{ID}\|$	100		1600	mV
Input common-mode voltage	V_{ICM}	0		1200	mV
Receiver differential input impedance	R_I	80	100	120	Ω

1. QSGMII DC input sensitivity is less than 400 mV.

5.1.8 Current Consumption

The following tables show the current consumption values for each mode. Add significant margin above the values for sizing power supplies.

Table 494 • **Current Consumption**

Mode	Typical				Maximum				Unit	Condition
	1 V Digital	1 V Analog	2.5 V Digital	2.5 V Analog	1 V Digital	1 V Analog	2.5 V Digital	2.5 V Analog		
Reset	52	55	9	1	460	110	13	5	mA	
Power down	110	170	10	20	525	220	15	25	mA	
1000BASE-T	250	180	15	250	755	245	15	265	mA	2-port SGMII
100BASE-TX	155	175	15	170	645	235	15	190	mA	2-port SGMII

Table 494 • **Current Consumption *(continued)***

Mode	Typical				Maximum				Unit	Condition
	1 V Digital	1 V Analog	2.5 V Digital	2.5 V Analog	1 V Digital	1 V Analog	2.5 V Digital	2.5 V Analog		
10BASE-T	130	170	15	145	615	230	15	150	mA	2-port SGMII
10BASE-Te	130	170	15	135	615	230	15	140	mA	2-port SGMII
1000BASE-X	145	205	15	35	685	265	15	40	mA	2-port SGMII
100BASE-FX	135	200	15	35	645	260	15	40	mA	2-port SGMII
1000BASE-T	300	180	15	250	800	245	15	265	mA	2-port SGMII + 1588
100BASE-TX	175	175	15	170	665	235	15	190	mA	2-port SGMII + 1588
10BASE-T	150	170	15	145	625	230	15	150	mA	2-port SGMII + 1588
10BASE-Te	150	170	15	135	625	230	15	140	mA	2-port SGMII + 1588
1000BASE-X	190	205	15	35	715	265	15	40	mA	2-port SGMII + 1588
100BASE-FX	155	200	15	35	665	260	15	40	mA	2-port SGMII + 1588
1000BASE-T	235	100	55	250	740	160	65	265	mA	2-port RGMII
100BASE-TX	140	95	20	170	630	150	20	190	mA	2-port RGMII
10BASE-T	115	90	15	145	600	145	15	150	mA	2-port RGMII
10BASE-Te	115	90	15	135	600	145	15	140	mA	2-port RGMII
1000BASE-X	130	120	55	35	670	180	65	40	mA	2-port RGMII
100BASE-FX	120	115	20	35	630	175	20	40	mA	2-port RGMII
1000BASE-T	290	100	60	250	795	160	65	265	mA	2-port RGMII + 1588
100BASE-TX	165	95	20	170	650	150	20	190	mA	2-port RGMII + 1588
10BASE-T	140	90	15	145	610	145	15	150	mA	2-port RGMII + 1588
10BASE-Te	140	90	15	135	610	145	15	140	mA	2-port RGMII + 1588
1000BASE-X	180	120	60	35	700	180	65	40	mA	2-port RGMII + 1588
100BASE-FX	145	115	20	35	650	175	20	40	mA	2-port RGMII + 1588
1000BASE-T	250	145	10	250	755	210	15	265	mA	2-port half QSGMII
100BASE-TX	155	140	10	170	645	200	15	190	mA	2-port half QSGMII
10BASE-T	130	135	10	145	615	195	15	150	mA	2-port half QSGMII
10BASE-Te	130	135	10	135	615	195	15	140	mA	2-port half QSGMII
1000BASE-X	145	170	10	35	685	230	15	40	mA	2-port half QSGMII
100BASE-FX	135	165	10	35	645	225	15	40	mA	2-port half QSGMII
1000BASE-T	300	145	10	250	800	210	15	265	mA	2-port half QSGMII + 1588
100BASE-TX	175	140	10	170	665	200	15	190	mA	2-port half QSGMII + 1588
10BASE-T	150	135	10	145	625	195	15	150	mA	2-port half QSGMII + 1588
10BASE-Te	150	135	10	135	625	195	15	140	mA	2-port half QSGMII + 1588
1000BASE-X	190	175	10	35	715	230	15	40	mA	2-port half QSGMII + 1588

Table 494 • **Current Consumption *(continued)***

Mode	Typical				Maximum				Unit	Condition
	1 V Digital	1 V Analog	2.5 V Digital	2.5 V Analog	1 V Digital	1 V Analog	2.5 V Digital	2.5 V Analog		
100BASE-FX	155	170	10	35	665	225	15	40	mA	2-port half QSGMII + 1588

5.1.9 Thermal Diode

The VSC8572-02 device includes an on-die diode and internal circuitry for monitoring die temperature (junction temperature). The operation and accuracy of the diode is not guaranteed and should only be used as a reference. Care should be taken to find compatible grounded cathode temperature monitoring device.

A thermal sensor, located on the board or in a stand-alone measurement kit, can monitor and display the die temperature of the switch for thermal management or instrumentation purposes.

Temperature measurement using a thermal diode is very sensitive to noise.

The following table provides the diode parameter and interface specifications. Note that the ThermDC pin is connected to VSS internally in the device.

Table 495 • **Thermal Diode Parameters**

Parameter	Symbol	Typical	Maximum	Unit
Forward bias current	IFW		1	mA
Diode ideality factor	n	1.008		

Note: Microsemi does not support or recommend operation of the thermal diode under reverse bias.

The ideality factor, n, represents the deviation from ideal diode behavior as exemplified by the diode equation:

$$I_{FW} = I_S \times \left(e^{V_d \times \frac{q}{nkT}} - 1 \right)$$

where, Is = saturation current, q = electronic charge, Vd = voltage across the diode, k = Boltzmann Constant, and T = absolute temperature (Kelvin).

5.2 AC Characteristics

This section provides the AC specifications for the VSC8572-02 device.

5.2.1 Reference Clock

The following table shows the AC specifications for a 125 MHz differential reference clock source. Performance is guaranteed for 125 MHz differential clocks only; however, 125 MHz single-ended clocks are also supported for QSGMII interfaces.

25 MHz clock implementations are available but are limited to SGMII interfaces. For more information, contact your Microsemi representative.

Table 496 • **Reference Clock AC Characteristics for QSGMII 125 MHz Differential Clock**

Parameter	Symbol	Minimum	Typical	Maximum	Unit	Condition
Reference clock frequency, REFCLK_SEL2 = 1	*f*		125.00		MHz	±100 ppm

Table 496 • **Reference Clock AC Characteristics for QSGMII 125 MHz Differential Clock *(continued)***

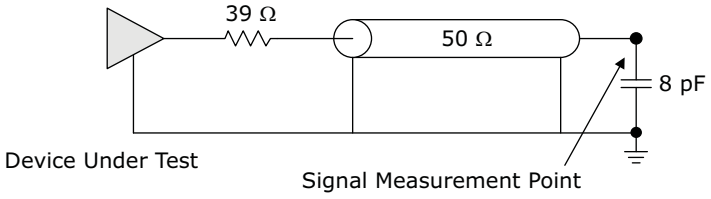
Parameter	Symbol	Minimum	Typical	Maximum	Unit	Condition
Duty cycle	DC	40	50	60	%	
Rise time and fall time	t_r, t_f			1.5	ns	20% to 80% threshold
RefClk input RMS jitter requirement, bandwidth between 12 kHz and 500 kHz(1)				20	ps	To meet jitter generation of 1G output data per IEEE 802.3z
RefClk input RMS jitter requirement, bandwidth between 500 kHz and 15 MHz(1)				4	ps	To meet jitter generation of 1G output data per IEEE 802.3z
RefClk input RMS jitter requirement, bandwidth between 15 MHz and 40 MHz(1)				20	ps	To meet jitter generation of 1G output data per IEEE 802.3z
RefClk input RMS jitter requirement, bandwidth between 40 MHz and 80 MHz(1)				100	ps	To meet jitter generation of 1G output data per IEEE 802.3z
Jitter gain from RefClk to SerDes output, bandwidth between 0 MHz and 0.1 MHz				0.3	dB	
Jitter gain from RefClk to SerDes output, bandwidth between 0.1 MHz and 7 MHz			1	3	dB	
Jitter gain from RefClk to SerDes output, bandwidth above 7 MHz		1–20 × log (f/7 MHz)		3–20 × log (f/7 MHz)	dB	

1. Maximum RMS jitter allowed at the RefClk input for the given bandwidth.

5.2.2 Recovered Clock

This section provides the AC characteristics for the recovered clock output signals. The following illustration shows the test circuit for the recovered clock output signals.

Figure 83 • **Test Circuit for Recovered Clock Output Signals**



The following table shows the AC specifications for the RCVRDCLK1 and RCVRDCLK2 outputs.

Table 497 • **Recovered Clock AC Characteristics**

Parameter	Symbol	Minimum	Typical	Maximum	Unit	Condition
Recovered clock frequency	f		125.00		MHz	
Recovered clock frequency	f		31.25		MHz	
Recovered clock frequency	f		25.00		MHz	
Recovered clock cycle time	t_{RCYC}		8.0		ns	
Recovered clock cycle time	t_{RCYC}		32.0		ns	
Recovered clock cycle time	t_{RCYC}		40.0		ns	
Frequency stability	$f_{STABILITY}$			50	ppm	
Duty cycle, master mode	DC	40	50	60	%	
Clock rise time and fall time	t_R, t_F		600		ps	20% to 80%
Peak-to-peak jitter, copper media interface (1000BASE-T slave mode)	JPP_{CLK_Cu}			400	ps	10K samples
Peak-to-peak jitter, fiber media interface, 100BASE-FX	JPP_{CLK_FiFX}			1.2	ns	10K samples
Peak-to-peak jitter, fiber media interface, 1000BASE-X	JPP_{CLK_FiX}			250	ps	10K samples

5.2.3 SerDes Outputs

The values listed in the following table are valid for all configurations, unless otherwise noted.

Table 498 • **SerDes Outputs AC Specifications**

Parameter	Symbol	Minimum	Maximum	Unit	Condition
V_{OD} ringing compared to V_S, RGMII/SGMII mode	V_{RING}		±10	%	RL = 100 Ω ±1%
V_{OD} rise time and fall time, RGMII/SGMII mode	t_R, t_F	100	200	ps	20% to 80% of V_S RL = 100 Ω ±1%
Differential peak-to-peak output voltage	V_{OD}		30	mV	Tx disabled
Differential output return loss, 50 MHz to 625 MHz	R_{LO_DIFF}	≥10		dB	RL = 100 Ω ±1%
Differential output return loss, 625 MHz to 1250 MHz	R_{LO_DIFF}	10–10 × log (f/625 MHz)		dB	RL = 100 Ω ±1%

Table 498 • **SerDes Outputs AC Specifications *(continued)***

Parameter	Symbol	Minimum	Maximum	Unit	Condition
Common-mode return loss, 50 MHz to 625 MHz	RL_{OCM}	6		dB	
Interpair skew, RGMII/SGMII mode	t_{SKEW}		20	ps	

5.2.4 SerDes Driver Jitter

The following table lists the jitter characteristics for the SerDes output driver.

Table 499 • **SerDes Driver Jitter Characteristics**

Parameter	Symbol	Maximum	Unit	Condition
Total jitter	TJ_O	192	ps	Measured according to IEEE 802.3.38.5
Deterministic jitter	DJ_O	80	ps	Measured according to IEEE 802.3.38.5

5.2.5 SerDes Inputs

The following table lists the AC specifications for the SerDes inputs.

Table 500 • **SerDes Input AC Specifications**

Parameter	Maximum	Unit	Condition
Differential input return loss, 50 MHz to 625 MHz	≥10	dB	RL = 100 Ω ±1%
Differential input return loss, 625 MHz to 1250 MHz	10–10 × log (*f*/625 MHz)	dB	RL = 100 Ω ±1%

5.2.6 SerDes Receiver Jitter Tolerance

The following table lists jitter tolerances for the SerDes receiver.

Table 501 • **SerDes Receiver Jitter Tolerance**

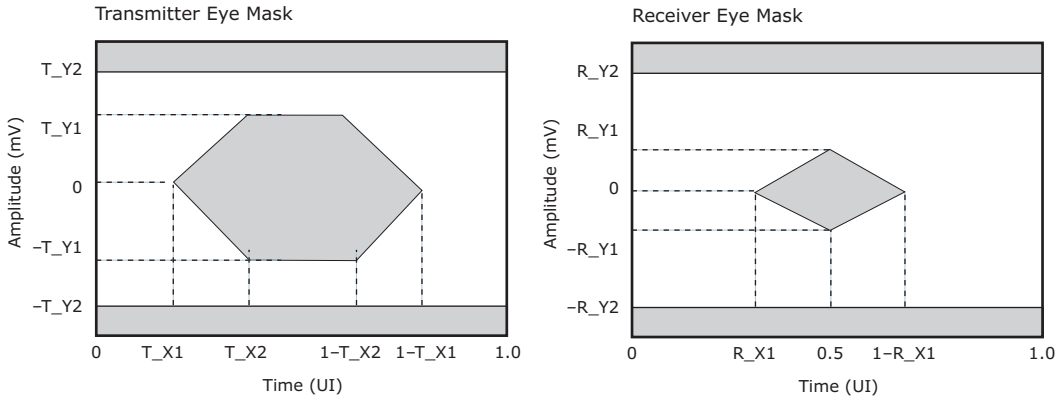
Parameter	Symbol	Minimum	Unit	Condition
Total jitter tolerance, greater than 637 kHz, SFP mode	TJT_I	600	ps	Measured according to IEEE 802.3 38.6.8
Deterministic jitter tolerance, greater than 637 kHz, SFP mode	DJT_I	370	ps	Measured according to IEEE 802.3 38.6.8
Cycle distortion jitter tolerance, 100BASE-FX mode	JT_{CD}	1.4	ns	Measured according to ISO/IEC 9314-3:1990
Data-dependent jitter tolerance, 100BASE-FX mode	DDJ	2.2	ns	Measured according to ISO/IEC 9314-3:1990
Random peak-to-peak jitter tolerance, 100BASE-FX mode	RJT	2.27	ns	Measured according to ISO/IEC 9314-3:1990

5.2.7 Enhanced SerDes Interface

All AC specifications for the enhanced SerDes interface are compliant with QSGMII Specification Revision 1.3 and meet or exceed the requirements in the standard. They are also compliant with the OIF-CEI-02.0 requirements where applicable.

The enhanced SerDes interface supports the following modes of operation: SGMII, QSGMII, and SFP. The values in the tables in the following sections apply to the QSGMII modes listed in the condition column and are based on the test circuit shown in Figure 79, page 294. The transmit and receive eye specifications relate to the eye diagrams shown in the following illustration, with the compliance load as defined in the test circuit.

Figure 84 • **QSGMII Transient Parameters**



5.2.7.1 Enhanced SerDes Outputs

The following table provides the AC specifications for the enhanced SerDes outputs in SGMII mode.

Table 502 • **Enhanced SerDes Outputs AC Specifications, SGMII Mode**

Parameter	Symbol	Minimum	Maximum	Unit	Condition
Unit interval, 1.25G mode	UI				800 ps
V_{OD} ringing compared to V_S	V_{RING}		±10	%	R_L = 100 Ω ±1%
V_{OD} rise time and fall time	t_R, t_F	100	200	ps	20% to 80% of V_S R_L = 100 Ω ±1%
Differential peak-to-peak output voltage	V_{OD}		30	mV	Tx disabled
Differential output return loss, 50 MHz to 625 MHz	RL_{O_DIFF}	≥10		dB	R_L = 100 Ω ±1%
Differential output return loss, 625 MHz to 1250 MHz	RL_{O_DIFF}	10–10 × log (f/625 MHz)		dB	R_L = 100 Ω ±1%
Common-mode return loss, 50 MHz to 625 MHz	RL_{OCM}	6		dB	
Intrapair skew	t_{SKEW}		20	ps	

The following table provides the AC specifications for the enhanced SerDes outputs in QSGMII mode.

Table 503 • **Enhanced SerDes Outputs AC Specifications, QSGMII Mode**

Parameter	Symbol	Minimum	Maximum	Unit	Condition
Unit interval, 5G	UI				200 ps
V_{OD} rise time and fall time	t_R, t_F	30	96	ps	20% to 80% of V_S R_L = 100 Ω ±1%
Differential peak-to-peak output voltage	V_{OD}		30	mV	Tx disabled

Table 503 • **Enhanced SerDes Outputs AC Specifications, QSGMII Mode**

Parameter	Symbol	Minimum	Maximum	Unit	Condition
Differential output return loss, 100 MHz to 2.5 GHz	RL_{O_DIFF}	8		dB	R_L = 100 Ω ±1%
Differential output return loss, 2.5 GHz to 5 GHz	RL_{O_DIFF}	8 dB – 16.6 log (*f*/2.5 GHz)		dB	R_L = 100 Ω ±1%
Eye mask X1	T_X1		0.15	UI	
Eye mask X2	T_X2		0.4	UI	
Eye mask Y1	T_Y1	200		mV	
Eye mask Y2	T_Y2		450	mV	

5.2.7.2 Enhanced SerDes Driver Jitter Characteristics

The following table lists the jitter characteristics for the enhanced SerDes driver in QSGMII mode. For information about jitter characteristics for the enhanced SerDes driver in SGMII mode, see Table 499, page 301.

Table 504 • **Enhanced SerDes Driver Jitter Characteristics, QSGMII Mode**

Parameter	Symbol	Maximum	Unit	Condition
Total output jitter	TJ_O	60	ps	Measured according to IEEE 802.3.38.5.
Deterministic output jitter	DJ_O	10	ps	Measured according to IEEE 802.3.38.5.

5.2.7.3 Enhanced SerDes Inputs

The following table lists the AC specifications for the enhanced SerDes inputs in SGMII mode.

Table 505 • **Enhanced SerDes Input AC Specifications, SGMII Mode**

Parameter	Symbol	Minimum	Unit	Condition
Unit interval, 1.25G	UI		ps	800 ps
Differential input return loss, 50 MHz to 625 MHz	RL_{I_DIFF}	10	dB	R_L = 100 Ω ±1%
Common-mode input return loss, 50 MHz to 625 MHz	RL_{ICM}	6	dB	

The following table lists the AC specifications for the enhanced SerDes inputs in QSGMII mode.

Table 506 • **Enhanced SerDes Inputs AC Specifications, QSGMII Mode**

Parameter	Symbol	Minimum	Maximum	Unit	Condition
Unit interval, 5G	UI				200 ps
Differential input return loss, 100 MHz to 2.5 GHz	RL_{I_DIFF}	8		dB	R_L = 100 Ω ±1%
Differential input return loss, 2.5 GHz to 5 GHz	RL_{I_DIFF}	8 dB – 16.6 log (*f*/2.5 GHz)		dB	R_L = 100 Ω ±1%
Common-mode input return loss, 100 MHz to 2.5 GHz	RL_{ICM}	6		dB	
Eye mask X1	R_X1		0.3	UI	

Table 506 • **Enhanced SerDes Inputs AC Specifications, QSGMII Mode**

Parameter	Symbol	Minimum	Maximum	Unit	Condition
Eye mask Y1	R_Y1		50	mV	
Eye mask Y2	R_Y2		450	mV	

5.2.7.4 Enhanced SerDes Receiver Jitter Tolerance

The following table lists the jitter tolerance for the enhanced SerDes receiver in QSGMII mode. For information about jitter tolerance for the enhanced SerDes receiver in SGMII mode, see Table 501, page 301.

Table 507 • **Enhanced SerDes Receiver Jitter Tolerance, QSGMII Mode**

Parameter	Symbol	Maximum	Unit	Condition
Bounded high-probability jitter(1)	BHPJ	90	ps	92 ps peak-to-peak random jitter and 38 ps sinusoidal jitter (SJHF).
Sinusoidal jitter, maximum	SJ_{MAX}	1000	ps	
Sinusoidal jitter, high frequency	SJ_{HF}	10	ps	
Total jitter tolerance	TJT_{I}	120	ps	92 ps peak-to-peak random jitter and 38 ps sinusoidal jitter (SJHF).

1. This is the sum of uncorrelated bounded high probability jitter (0.15 UI), and correlated bounded high probability jitter (0.30 UI). Uncorrelated bounded high probability jitter is distribution where the value of the jitter shows no correlation to any signal level being transmitted, formally defined as deterministic jitter (DJ). Correlated bounded high probability jitter is jitter distribution where the value of the jitter shows a strong correlation to the signal level being transmitted.

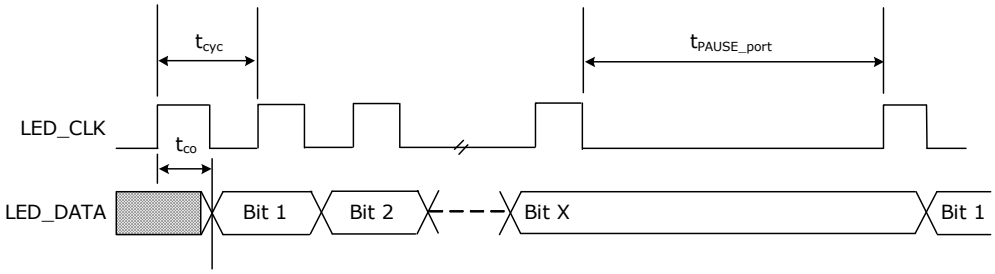
5.2.8 Basic Serial LEDs

This section contains the AC specifications for the basic serial LEDs.

Table 508 • **Basic Serial LEDs AC Characteristics**

Parameter	Symbol	Typical	Unit
LED_CLK cycle time	t_{CYC}	1024	ns
Pause between LED port sequences	t_{PAUSE_port}	3072	ns
Pause between LED bit sequences	t_{PAUSE_bit}	25.541632	ms
LED_CLK to LED_DATA	t_{CO}	1	ns

Figure 85 • **Basic Serial LED Timing**



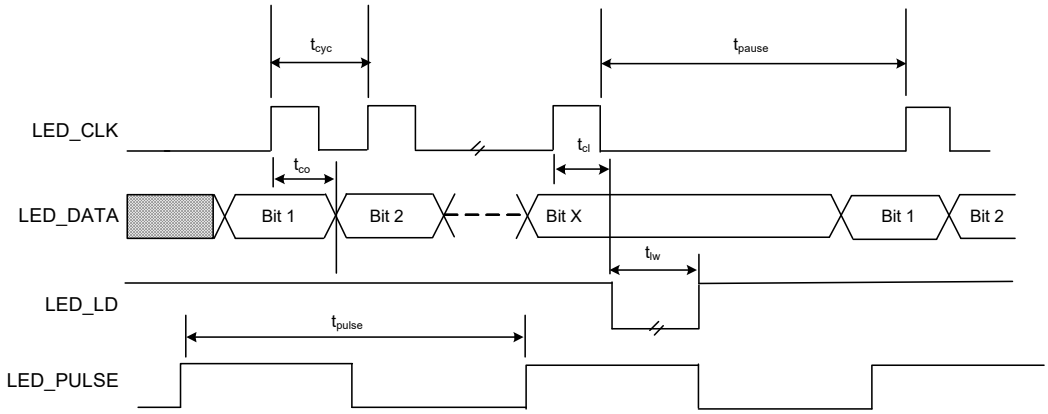
5.2.9 Enhanced Serial LEDs

This section contains the AC specifications for the enhanced serial LEDs. The duty cycle of the LED_PULSE signal is programmable and can be varied between 0.5% and 99.5%.

Table 509 • **Enhanced Serial LEDs AC Characteristics**

Parameter	Symbol	Minimum	Typical	Maximum	Unit
LED_CLK cycle time	t_{CYC}		256		ns
Pause between LED_DATA bit sequences	t_{PAUSE}	0.396		24.996	ms
LED_CLK to LED_DATA	t_{CO}		127		ns
LED_CLK to LED_LD	t_{CL}		256		ns
LED_LD pulse width	t_{LW}		128		ns
LED_PULSE cycle time	t_{PULSE}	199		201	µs

Figure 86 • **Enhanced Serial LED Timing**



5.2.10 JTAG Interface

This section provides the AC specifications for the JTAG interface. The specifications meet or exceed the requirements of IEEE 1149.1-2001. The JTAG receive signal requirements are requested at the pin of the device. The JTAG_TRST signal is asynchronous to the clock, and does not have a setup or hold time requirement.

Table 510 • **JTAG Interface AC Specifications**

Parameter	Symbol	Minimum	Maximum	Unit	Condition
TCK frequency	f		10	MHz	
TCK cycle time	t_C	100		ns	
TCK high time	$t_{W(CH)}$	40		ns	
TCK low time	$t_{W(CL)}$	40		ns	
Setup time to TCK rising	t_{SU}	10		ns	
Hold time from TCK rising	t_H	10		ns	
TDO valid after TCK falling	$t_{V(C)}$		28	ns	C_L = 10 pF
TDO hold time from TCK falling	$t_{H(TDO)}$	0		ns	C_L = 0 pF
TDO disable time[1]	t_{DIS}		30	ns	See Figure 88, page 306.

Table 510 • **JTAG Interface AC Specifications *(continued)***

Parameter	Symbol	Minimum	Maximum	Unit	Condition
TRST time low	$t_{W(TL)}$	30		ns	

1. The pin begins to float when a 300 mV change from the actual V_{OH}/V_{OL} level occurs.

Figure 87 • **JTAG Interface Timing Diagram**

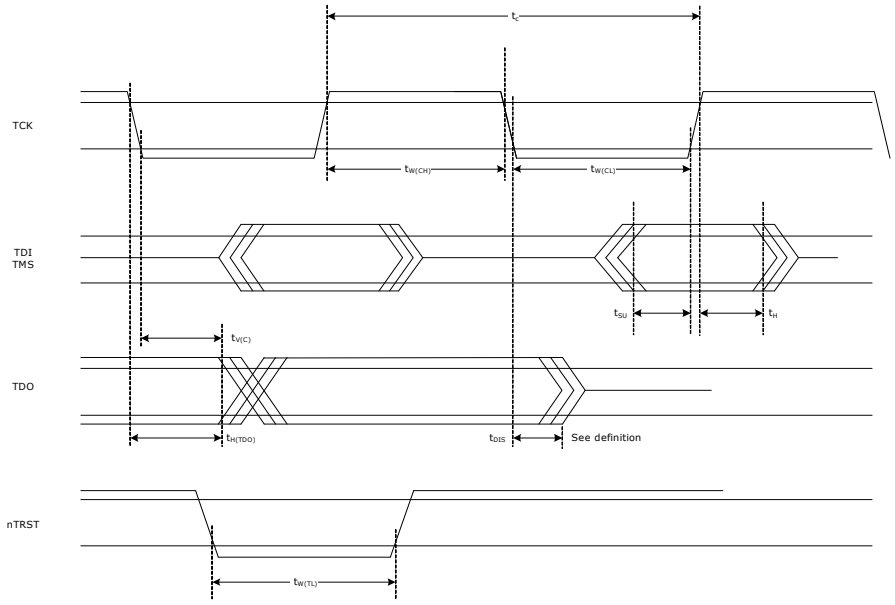
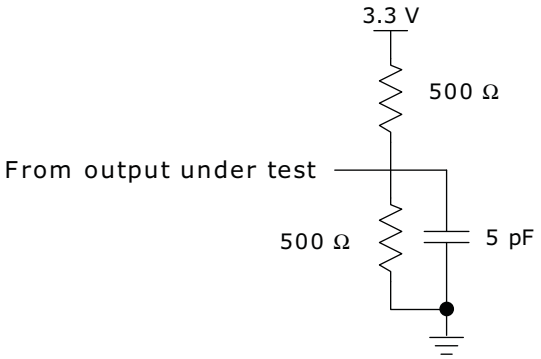


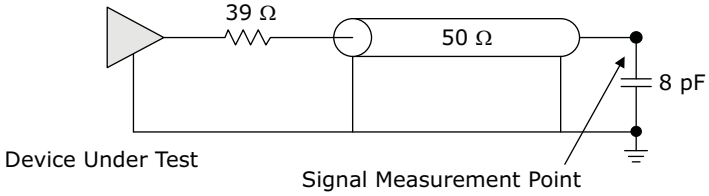
Figure 88 • **Test Circuit for TDO Disable Time**



5.2.11 RGMII, Uncompensated

The following illustration shows the test circuit for the RGMII output signals.

Figure 89 • **Test Circuit for RGMII Output Signals**



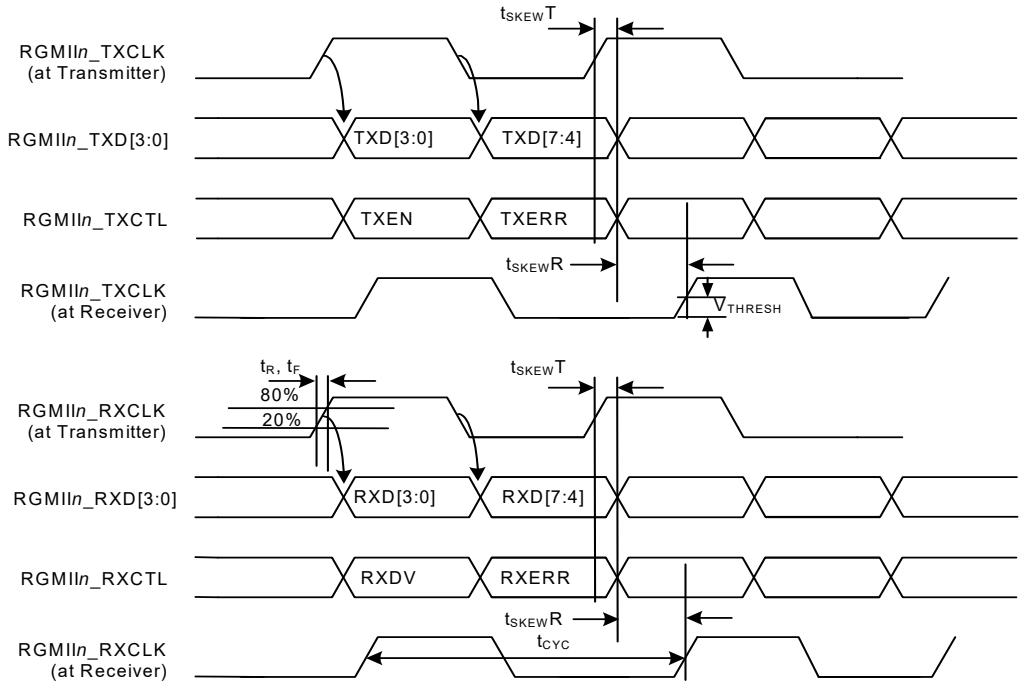
The following table lists the characteristics when using the device in RGMII uncompensated mode. For more information about the RGMII uncompensated timing, see Figure 90, page 308.

Table 511 • **AC Characteristics for RGMII Uncompensated**

Parameter	Symbol	Minimum	Typical	Maximum	Unit	Condition
Clock frequency			125 25 2.5		MHz	1000BASE-T operation 100BASE-TX operation 10BASE-T operation
1000BASE-T duty cycle	$t_{DUTY1000}$	40	50	60	%	
10/100BASE-T duty cycle	$t_{DUTY10/100}$	35	38	65	%	10BASE-T
10/100BASE-T duty cycle	$t_{DUTY10/100}$	40	50	60	%	100BASE-TX
Data to clock output skew[1] (at transmitter)	$t_{SKEW}T$	–500	0	500	ps	
Data to clock output skew[1] (at receiver)	$t_{SKEW}R$	1.0	1.8	2.6	ns	
TX_CLK switching threshold	V_{THRESH}		1.25		V	V_{DD25} = 2.5
Clock/data output rise and fall times	t_R and t_F			750	ps	20% to 80%, 1000BASE-T
Clock/data output rise and fall times	t_R and t_F			1000	ps	20% to 80%, 10BASE-T/ 100BASE-TX

1. When operating in uncompensated mode, the PC board design requires a clock to be routed so that an additional trace delay of greater than 1.5 ns is added to the associated clock signal.

Figure 90 • **RGMII Uncompensated Timing**



n= 0 or 1, corresponding to PHY *n*.

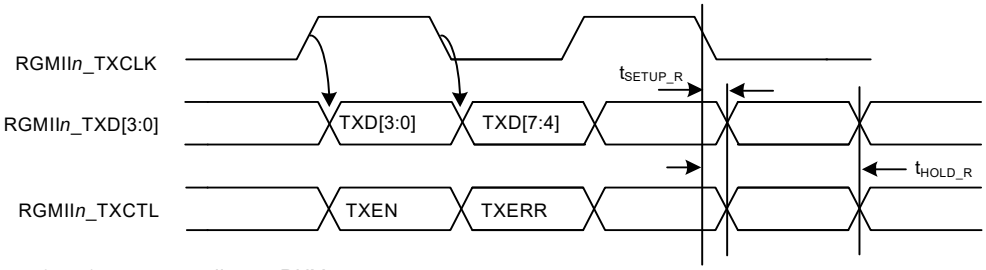
5.2.12 RGMII, Compensated

The following table lists the characteristics when using the device in RGMII compensated mode.

Table 512 • **PHY Input (RGMII*n*_TXCLK Delay When Register 18E2.[6:4]=011'b)**

Parameter	Symbol	Minimum	Typical	Maximum	Unit
Data to clock setup (TX_CLK delay = 011'b)	t_{SETUP_R}	–1.0			ns
Clock to data hold (TX_CLK delay = 011'b)	t_{HOLD_R}	2.8			ns

Figure 91 • **Compensated Input RGMII Timing**

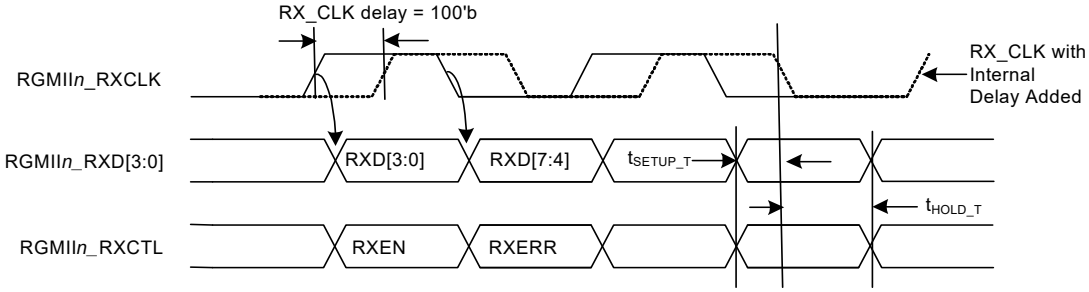


n= 0 or 1, corresponding to PHY *n*.

Table 513 • **PHY Output (RGMII*n*_RXCLK Delay When Register 18E2.[3:1]=100'b)**

Parameter	Symbol	Minimum	Typical	Maximum	Unit
Data to clock setup (RX_CLK delay = 100'b)	t_{SETUP_T}	1.3	2.0		ns
Clock to data hold (RX_CLK delay = 100'b)	t_{HOLD_T}	1.4	2.0		ns

Figure 92 • **Compensated Output RGMII Timing**



n= 0 or 1, corresponding to PHY *n*.

5.2.13 Serial Management Interface

This section contains the AC specifications for the serial management interface (SMI).

Table 514 • **Serial Management Interface AC Characteristics**

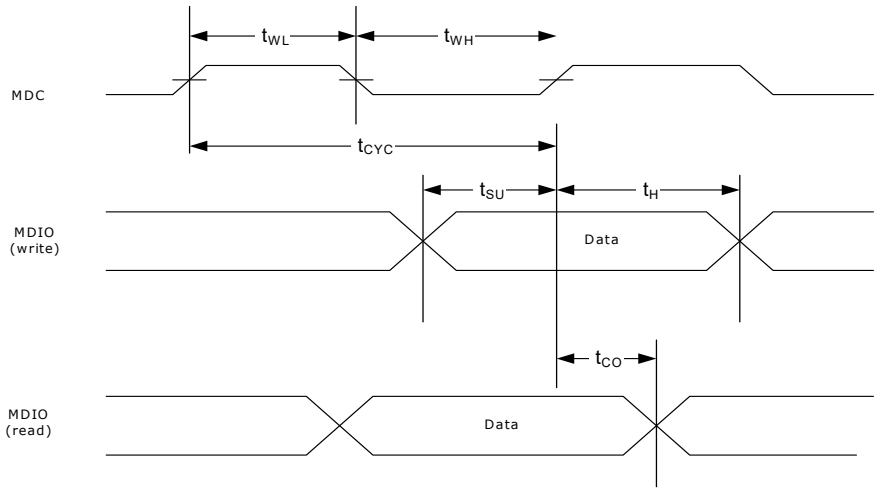
Parameter	Symbol	Minimum	Typical	Maximum	Unit	Condition
MDC frequency(1)	f_{CLK}		2.5	12.5	MHz	
MDC cycle time	t_{CYC}	80	400		ns	
MDC time high	t_{WH}	20	50		ns	
MDC time low	t_{WL}	20	50		ns	
Setup to MDC rising	t_{SU}	10			ns	
Hold from MDC rising	t_H	10			ns	
MDC rise time	t_R			100 t_{CYC} × 10%(1)	ns	MDC = 0: 1 MHz MDC = 1: MHz – f_{CLK} maximum

Table 514 • **Serial Management Interface AC Characteristics *(continued)***

Parameter	Symbol	Minimum	Typical	Maximum	Unit	Condition
MDC fall time	t_F			100 t_{CYC} × 10%(1)		
MDC to MDIO valid	t_{CO}		10	300	ns	Time-dependant on the value of the external pull-up resistor on the MDIO pin

1. For f_{CLK} above 1 MHz, the minimum rise time and fall time is in relation to the frequency of the MDC clock period. For example, if f_{CLK} is 2 MHz, the minimum clock rise time and fall time is 50 ns.

Figure 93 • **Serial Management Interface Timing**



5.2.14 Reset Timing

This section contains the AC specifications that apply to device reset functionality. The signal applied to the NRESET input must comply with the specifications listed in the following table.

Table 515 • **Reset Timing Specifications**

Parameter	Symbol	Minimum	Maximum	Unit
NRESET assertion time after power supplies and clock stabilize	t_W	2		ms
Recovery time from reset inactive to device fully active	t_{REC}		105	ms
NRESET pulse width	$t_{W(RL)}$	100		ns
Wait time between NRESET de-assert and access of the SMI interface	t_{WAIT}	105		ms

5.2.15 1588 Timing Specifications

This section contains the AC specifications for the 1588 clock pins.

Table 516 • **1588 Timing Specifications AC Characteristics**

Parameter	Symbol	Minimum	Typical	Maximum	Unit	Condition
1588 reference clock frequency[1]	f	125		250	MHz	±100 ppm Jitter < 10 ps RMS
Duty cycle	DC	40	50	60	%	
Rise time and fall time	t_R, t_F		1.5		ns	20% to 80% threshold

1. Supports a continuum of frequencies between 125 MHz and 250 MHz.

5.2.16 Serial Timestamp Interface

This section contains information about the AC specifications for the serial timestamp interface.

Table 517 • **Serial Timestamp Interface**

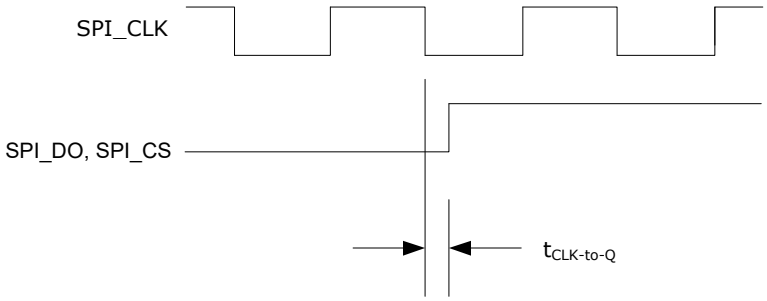
Parameter	Symbol	Minimum	Typical	Maximum	Unit	Condition
1588_SPI_CLK frequency				62.5[1]	MHz	
1588_SPI_DO clock-to-Q timing	$t_{CLK\text{-}to\text{-}Q}$	–5		0	ns	
1588_SPI_CS clock-to-Q timing	$t_{CLK\text{-}to\text{-}Q}$	–5		0	ns	

1. SPI clock low time programmed through SI_CLK_LO_CYCs must always equal 0x1 (8 nanoseconds) for correct bus operation. Duty cycle is dependent on SI_CLK_HI_CYCs configuration.

The following illustration shows the serial timestamp interface timing diagram.

Note: Data changes state on a falling 1588_SPI_CLK edge in the default configuration. 1588_SPI_CLK can be inverted by setting the 1588 register bit TS_FIFO_SI_CFG:SI_CLK_PHA.

Figure 94 • **Serial Timestamp Interface Timing Diagram**



5.2.17 Local Time Counter Load/Save Timing

This section contains information about the AC specifications for the local time counter load/save signal.

Figure 95 • **Local Time Counter Load/Save Timing Diagram**

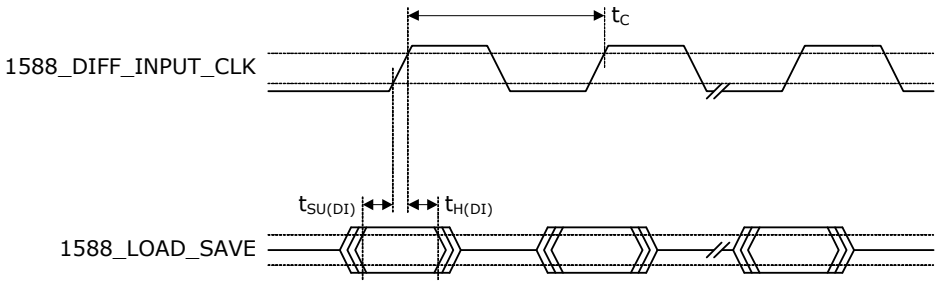


Table 518 • **Local Time Counter Load/Save Timing Specifications**

Parameter	Symbol	Minimum	Maximum	Unit
Clock frequency	*f*		250	MHz
Clock cycle time	t_C	4		ns
DI setup time to clock	$t_{SU(DI)}$	2.8		ns
DI hold time from clock	$t_{H(DI)}$	0.3		ns

5.3 Operating Conditions

The following table shows the recommended operating conditions for the VSC8572-02 device.

Table 519 • **Recommended Operating Conditions**

Parameter	Symbol	Minimum	Typical	Maximum	Unit
Power supply voltage for V_{DD1}	V_{DD1}	0.95	1.00	1.05	V
Power supply voltage for V_{DD1A}	V_{DD1A}	0.95	1.00	1.05	V
Power supply voltage for V_{DD25}	V_{DD25}	2.38	2.50	2.62	V
Power supply voltage for V_{DD25A}	V_{DD25A}	2.38	2.50	2.62	V
VSC8572-02 operating temperature(1)	T	0		125	°C
VSC8572-05 operating temperature(1)	T	–40		125	°C

1. Minimum specification is ambient temperature, and the maximum is junction temperature. For carrier class applications, the maximum operating temperature is 110 °C junction.

5.4 Stress Ratings

This section contains the stress ratings for the VSC8572-02 device.

Warning Stresses listed in the following table may be applied to devices one at a time without causing permanent damage. Functionality at or exceeding the values listed is not implied. Exposure to these values for extended periods may affect device reliability.

Table 520 • **Stress Ratings**

Parameter	Symbol	Minimum	Maximum	Unit
Power supply voltage for core supply	V_{VDD1}	–0.3	1.10	V
Power supply voltage for analog circuits	V_{VDD1A}	–0.3	1.10	V
Power supply voltage for analog circuits	V_{VDD25A}	–0.3	2.75	V
Power supply voltage for digital I/O	V_{VDD25}	–0.3	2.75	V

Table 520 • **Stress Ratings *(continued)***

Parameter	Symbol	Minimum	Maximum	Unit
Input voltage for GPIO and logic input pins			3.3	V
Storage temperature	T_S	–55	125	°C
Electrostatic discharge voltage, charged device model	V_{ESD_CDM}	–250	250	V
Electrostatic discharge voltage, human body model	V_{ESD_HBM}	See note(1)		V

1. This device has completed all required testing as specified in the JEDEC standard JESD22-A114, *Electrostatic Discharge (ESD) Sensitivity Testing Human Body Model (HBM)*, and complies with a Class 2 rating. The definition of Class 2 is any part that passes an ESD pulse of 2000 V, but fails an ESD pulse of 4000 V.

Warning This device can be damaged by electrostatic discharge (ESD) voltage. Microsemi recommends that all integrated circuits be handled with appropriate precautions. Failure to observe proper handling and installation procedures may adversely affect reliability of the device.

6 Pin Descriptions

The VSC8572-02 device has 256 pins, which are described in this section.

The pin information is also provided as an attached Microsoft Excel file so that you can copy it electronically. In Acrobat, double-click the attachment icon.

6.1 Pin Identifications

This section contains the pin descriptions for the VSC8572-02 device. The following table provides notations for definitions of the various pin types.

Table 521 • **Pin Type Symbol Definitions**

Symbol	Pin Type	Description
3V		3.3 V-tolerant pin.
ABIAS	Analog bias	Analog bias pin.
ADIFF	Analog differential	Analog differential signal pair.
I	Input	Input without on-chip pull-up or pull-down resistor.
I/O	Bidirectional	Bidirectional input or output signal.
NC	No connect	No connect pins must be left floating.
O	Output	Output signal.
OD	Open drain	Open drain output.
OS	Open source	Open source output.
PD	Pull-down	On-chip pull-down resistor to VSS.
PU	Pull-up	On-chip pull-up resistor to VDD_IO.
ST	Schmitt-trigger	Input has Schmitt-trigger circuitry.

6.2 Pin Diagram

The following illustrations show the pin diagram for the VSC8572-02 device. For clarity, the device is shown in two halves, the top left and top right.

Figure 96 • **Pin Diagram, Top Left**

	1	2	3	4	5	6	7	8
A	NC_1	RESERVED_54	RESERVED_56	RESERVED_58	RESERVED_60	RESERVED_62	RESERVED_64	RESERVED_66
B	VSS_1	RESERVED_55	RESERVED_57	RESERVED_59	RESERVED_61	RESERVED_63	RESERVED_65	RESERVED_67
C	REFCLK_N	VDD25A_1	THERMDA	VDD25A_2	VSS_3	VDD25A_3	VDD1A_1	VDD1A_2
D	REFCLK_P	THERMDC_VSS	REF_FILT_A	REF_REXT_A	VSS_6	VSS_7	VSS_8	VSS_9
E	REFCLK_SEL2	TMS	TRST	VDD25A_6	VDD1_1	VSS_14	VSS_15	VSS_16
F	TDO	TDI	TCK	VSS_20	VDD1_3	VSS_21	VSS_22	VSS_23
G	LED0_0	LED1_0	LED2_0	LED3_0	VDD1_5	VSS_27	VSS_28	VSS_29
H	LED0_1	LED1_1	LED2_1	LED3_1	VDD1_7	VSS_33	VSS_34	VSS_35
J	RGMII1_RXD0	RGMII1_RXD1	RGMII1_RXD2	RGMII1_RXD3	VDD1_9	VSS_39	VSS_40	VSS_41
K	RGMII1_TXCLK	RGMII1_TXCTL	RGMII1_RXCTL	RGMII1_RXCLK	VDD1_11	VSS_45	VSS_46	VSS_47
L	RGMII1_TXD0	RESERVED_73	COMA_MODE	RESERVED_3	VDD1_13	VSS_51	VSS_52	VSS_53
M	RGMII1_TXD1	MDINT	NRESET	VDD25_2	VDD1_15	VSS_57	VSS_58	VSS_59
N	RGMII1_TXD2	MDIO	RESERVED_9	RESERVED_72	VDD1_17	VSS_63	VSS_64	VSS_65
P	RGMII1_TXD3	MDC	VDD25_4	RESERVED_4	VDD25A_8	VDD1A_5	VDD1A_6	VDD1A_7
R	VSS_69	RESERVED_22	RESERVED_24	RESERVED_26	RESERVED_28	RESERVED_30	RESERVED_32	RESERVED_34
T	NC_3	RESERVED_23	RESERVED_25	RESERVED_27	RESERVED_29	RESERVED_31	RESERVED_33	RESERVED_35

Figure 97 • **Pin Diagram, Top Right**

9	10	11	12	13	14	15	16	
RESERVED_68	TXVPA_1	TXVPB_1	TXVPC_1	TXVPD_1	TXVPA_0	TXVPB_0	NC_2	A
RESERVED_69	TXVNA_1	TXVNB_1	TXVNC_1	TXVND_1	TXVNA_0	TXVNB_0	VSS_2	B
VDD1A_3	RESERVED_1	VDD25A_4	VSS_4	VDD1A_4	VDD25A_5	TXVNC_0	TXVPC_0	C
VSS_10	VSS_11	VSS_12	VSS_13	RESERVED_2	RGMII0_TXD3	TXVND_0	TXVPD_0	D
VSS_17	VSS_18	VSS_19	VDD1_2	VDD25A_7	RGMII0_TXD2	CLK_SQUELCH_IN	1588_SPI_CLK	E
VSS_24	VSS_25	VSS_26	VDD1_4	RGMII0_TXD1	PHYADD4	RGMII0_TXD0	RCVRDCLK1	F
VSS_30	VSS_31	VSS_32	VDD1_6	PHYADD2	PHYADD3	RGMII0_TXCLK	RCVRDCLK2	G
VSS_36	VSS_37	VSS_38	VDD1_8	VDD25_1	GPIO13/1588_SPI_DO	RGMII0_TXCTL	RGMII0_RXCLK	H
VSS_42	VSS_43	VSS_44	VDD1_10	RGMII0_RXCTL	GPIO12/1588_SPI_CS	1588_DIFF_INPUT_CLK_P	1588_DIFF_INPUT_CLK_N	J
VSS_48	VSS_49	VSS_50	VDD1_12	GPIO8/I2C_SDA	GPIO9/FASTLINK-FAIL	GPIO10/1588_LOAD_SAVE	GPIO11	K
VSS_54	VSS_55	VSS_56	VDD1_14	GPIO4/I2C_SCL_0	GPIO5/I2C_SCL_1	RGMII0_RXD0	RGMII0_RXD1	L
VSS_60	VSS_61	VSS_62	VDD1_16	VDD25_3	GPIO1/SIGDET1	RGMII0_RXD2	RGMII0_RXD3	M
VSS_66	VSS_67	VSS_68	VDD1_18	SerDes_Rext_1	GPIO0/SIGDET0	TDP_0	TDN_0	N
VDD1A_8	VDD1A_9	VDD1A_10	VDD25A_9	VDD25A_10	SerDes_Rext_0	RDP_0	RDN_0	P
RESERVED_36	FIBROP_1	FIBRIP_1	RDP_1	TDP_1	FIBROP_0	FIBRIP_0	VSS_70	R
RESERVED_37	FIBRON_1	FIBRIN_1	RDN_1	TDN_1	FIBRON_0	FIBRIN_0	NC_4	T

6.3 Pins by Function

This section contains the functional pin descriptions for the VSC8572-02 device.

6.3.1 1588 Support

The following table lists the 1588 support pins.

Table 522 • **1588 Support Pins**

Name	Pin	Type	Description
1588_DIFF_INPUT_CLK_N 1588_DIFF_INPUT_CLK_P	J16 J15	ADIFF	Differential reference clock input pair.
RESERVED_9	N3	NC	Leave pin unconnected (floating).

Table 522 • **1588 Support Pins *(continued)***

Name	Pin	Type	Description
1588_SPI_CLK	E16	O	1588 SPI clock.

6.3.2 GPIO and 1588 Support

The following table lists the GPIO and 1588 support pins.

Table 523 • **GPIO and 1588 Support Pins**

Name	Pin	Type	Description
GPIO10/1588_LOAD_SAVE	K15	I/O, PU, 3 V	Sync signal to load the time to the 1588 engine. Rising edge triggered.
GPIO12/1588_SPI_CS	J14	I/O, PU, 3 V	1588 SPI chip select.
GPIO13/1588_SPI_DO	H14	I/O, PU, 3 V	1588 SPI data output.

6.3.3 GPIO and SIGDET

The following table lists the GPIO and SIGDET pins.

Table 524 • **GPIO and SIGDET Pins**

Name	Pin	Type	Description
GPIO0/SIGDET0 GPIO1/SIGDET1 GPIO4/I2C_SCL_0 GPIO5/I2C_SCL_1 GPIO8/I2C_SDA GPIO9/FASTLINK-FAIL GPIO11	N14 M14 L13 L14 K13 K14 K16	I/O, PU, 3 V	General purpose input/output (GPIO). The multipurpose SIGDET pins, two-wire serial controller pins, and fast link fail pin can be configured to serve as GPIOs.

6.3.4 JTAG

The following table lists the JTAG test pins.

Table 525 • **JTAG Pins**

Name	Pin	Type	Description
TCK	F3	I, PU, ST, 3 V	JTAG test clock input.
TDI	F2	I, PU, ST, 3 V	JTAG test serial data input.
TDO	F1	O	JTAG test serial data output.
TMS	E2	I, PU, ST, 3 V	JTAG test mode select.
TRST	E3	I, PU, ST, 3 V	JTAG reset. **Important** When JTAG is not in use, this pin must be tied to ground with a pull-down resistor for normal operation.

6.3.5 Miscellaneous

The following table lists the miscellaneous pins.

Table 526 • **Miscellaneous Pins**

Name	Pin	Type	Description
CLK_SQUELCH_IN	E15	I, PU, 3 V	Input control to squelch recovered clock.
COMA_MODE	L3	I, PU, 3 V	When this pin is asserted high, all PHYs are held in a powered down state. When de-asserted low, all PHYs are powered up and resume normal operation. This signal is also used to synchronize the operation of multiple chips on the same PCB to provide visual synchronization for LEDs driven by separate chips.(1)
LED0_[0:1] LED1_[0:1] LED2_[0:1] LED3_[0:1]	G1, H1 G2, H2 G3, H3, G4, H4,	O	LED direct-drive outputs. All LEDs pins are active-low. A serial LED stream can also be implemented. See LED Mode Select, page 110. **Note:** LEDbit_port, where port = PHY port number and bit = the particular LED for the port.
NC_1 NC_2 NC_3 NC_4	A1 A16 T1 T16	NC	No connect.
PHYADD2 PHYADD3 PHYADD4	G13 G14 F14	I, PD, 3 V	Device SMI address bits 4:2.
RCVRDCLK1 RCVRDCLK2	F16 G16	O	Clock output can be enabled or disabled and also output a clock frequency of 125 MHz or 25 MHz based on the selected active recovered media programmed for this pin. This pin is not active when NRESET is asserted. When disabled, the pin is held low.
REF_FILT_A	D3	ABIAS	Reference filter connects to an external 1 µF capacitor to analog ground.
REF_REXT_A	D4	ABIAS	Reference external connects to an external 2 kΩ (1%) resistor to analog ground.
REFCLK_N REFCLK_P	C1 D1	I, ADIFF	125 MHz or 25 MHz reference clock input pair. Must be capacitively coupled and LVDS compatible.
REFCLK_SEL2	E1	I, PU, 3 V	Selects the reference clock speed: 0: 25 MHz (VSS) 1: 125 MHz (2.5 V) Use 125 MHz for typical applications.
RESERVED_[1:4]	C10, D13, L4, P4	NC	Leave these pins unconnected (floating).
RESERVED_[22:37]	R2, T2, R3, T3, R4, T4, R5, T5, R6, T6, R7, T7, R8, T8, R9, T9	NC	Leave these pins unconnected (floating).

Table 526 • **Miscellaneous Pins *(continued)***

Name	Pin	Type	Description
RESERVED_[54:69]	A2, B2, A3, B3, A4, B4, A5, B5, A6, B6, A7, B7, A8, B8, A9, B9	NC	Leave these pins unconnected (floating).
RESERVED_[72:73]	N4, L2	NC	Leave these pins unconnected (floating).
THERMDA	C3	A	Thermal diode anode.
THERMDC_VSS	D2	A	Thermal diode cathode connected to device ground. Temperature sensor must be chosen accordingly.

1. For more information, see Initialization, page 93. For a typical bring-up example, see Configuration, page 92.

6.3.6 Power Supply

The following table lists the power supply pins and associated functional pins. All power supply pins must be connected to their respective voltage input, even if certain functions are not used for a specific application. No power supply sequencing is required. However, clock and power must be stable before releasing Reset.

Table 527 • **Power Supply Pins**

Name	Pin	Type	Description
VDD1_[1:18]	E5, E12, F5, F12, G5, G12, H5, H12, J5, J12, K5, K12, L5, L12, M5, M12, N5, N12	1.0 V	1.0 V internal digital logic.
VDD1A_[1:10]	C7, C8, C9, C13, P6, P7, P8, P9, P10, P11	1.0 V	1.0 V analog power requiring additional PCB power supply filtering. Associated with the QSGMII/SGMII MAC receiver output pins.
VDD25_[1:4]	H13, M4, M13, P3	2.5 V	2.5 V general digital power supply. Associated with the LED, GPIO, JTAG, twisted pair interface, reference filter, reference external supply connect, and recovered clock pins.
VDD25A_[1:10]	C2, C4, C6, C11, C14, E4, E13, P5, P12, P13	2.5 V	2.5 V general analog power supply.
VSS_[1:4] VSS_[6:70]	B1, B16, C5, C12 D5, D6, D7, D8, D9, D10, D11, D12, E6, E7, E8, E9, E10, E11, F4, F6, F7, F8, F9, F10, F11, G6, G7, G8, G9, G10, G11, H6, H7, H8, H9, H10, H11, J6, J7, J8, J9, J10, J11, K6, K7, K8, K9, K10, K11, L6, L7, L8, L9, L10, L11, M6, M7, M8, M9, M10, M11, N6, N7, N8, N9, N10, N11, R1, R16	0 V	General device ground.

6.3.6.1 RGMII Interface

The following table lists the RGMII interface pins.

Note: Unused RGMII port pins cannot be used as GPIOs.

Table 528 • **RGMII Interface Pins**

Name	Pin	Type	Description
RGMII0_RXCLK	H16	O	Receive clock. Receive data is sourced from the PHY synchronously on the rising edge of RXCLK and is the recovered clock from the media.
RGMII0_RXCTL	J13	O	Multiplexed receive data valid, receive error. This output is sampled by the MAC on opposite edges of RXCLK to indicate two receive conditions from the PHY: 1. On the rising edge of RXCLK, this output serves as RXDV and signals valid data is available on the RXD input data bus. 2. On the falling edge of RXCLK, this output signals a receive error from the PHY, based on a logical derivative of RXDV and RXER, as stated by the RGMII specification.
RGMII0_RXD0 RGMII0_RXD1 RGMII0_RXD2 RGMII0_RXD3	L15 L16 M15 M16	O	Multiplexed receive data. Bits 3:0 are synchronously output on the rising edge of RXCLK and bits 7:4 on the falling edge of RXCLK.
RGMII0_TXCLK	G15	I	Transmit clock. This clock is 2.5 MHz for 10 Mbps mode, 25 MHz for 100 Mbps mode, and 125 MHz for 1000 Mbps mode. If left unconnected, these pins require a pull-down resistor to ground.
RGMII0_TXCTL	H15	I	Multiplexed transmit enable, transmit error. This input is sampled by the PHY on opposite edges of TXCLK to indicate two transmit conditions of the MAC: 1. On the rising edge of TXCLK, this input serves as TXEN, indicating valid data is available on the TXD input data bus. 2. On the falling edge of TXCLK, this input signals a transmit error from the MAC, based on a logical derivative of TXEN and TXER, as stated by the RGMII specification.
RGMII0_TXD0 RGMII0_TXD1 RGMII0_TXD2 RGMII0_TXD3	F15 F13 E14 D14	I	Multiplexed transmit data. Bits 3:0 are synchronously output on the rising edge of TXCLK and bits 7:4 on the falling edge of TXCLK.
RGMII1_RXCLK	K4	O	Receive clock. Receive data is sourced from the PHY synchronously on the rising edge of RXCLK and is the recovered clock from the media.
RGMII1_RXCTL	K3	O	Multiplexed receive data valid, receive error. This output is sampled by the MAC on opposite edges of RXCLK to indicate two receive conditions from the PHY: 1. On the rising edge of RXCLK, this output serves as RXDV and signals valid data is available on the RXD input data bus. 2. On the falling edge of RXCLK, this output signals a receive error from the PHY, based on a logical derivative of RXDV and RXER, as stated by the RGMII specification.

Table 528 • **RGMII Interface Pins *(continued)***

Name	Pin	Type	Description
RGMII1_RXD0 RGMII1_RXD1 RGMII1_RXD2 RGMII1_RXD3	J1 J2 J3 J4	O	Multiplexed receive data. Bits 3:0 are synchronously output on the rising edge of RXCLK and bits 7:4 on the falling edge of RXCLK.
RGMII1_TXCLK	K1	I	Transmit clock. This clock is 2.5 MHz for 10 Mbps mode, 25 MHz for 100 Mbps mode, and 125 MHz for 1000 Mbps mode. If left unconnected, these pins require a pull-down resistor to ground.
RGMII1_TXCTL	K2	I	Multiplexed transmit enable, transmit error. This input is sampled by the PHY on opposite edges of TXCLK to indicate two transmit conditions of the MAC: 1. On the rising edge of TXCLK, this input serves as TXEN, indicating valid data is available on the TXD input data bus. 2. On the falling edge of TXCLK, this input signals a transmit error from the MAC, based on a logical derivative of TXEN and TXER, as stated by the RGMII specification.
RGMII1_TXD0 RGMII1_TXD1 RGMII1_TXD2 RGMII1_TXD3	L1 M1 N1 P1	I	Multiplexed transmit data. Bits 3:0 are synchronously output on the rising edge of TXCLK and bits 7:4 on the falling edge of TXCLK.

6.3.7 SGMII/SerDes/QSGMII MAC Interface

The following table lists the SerDes MAC interface pins.

Table 529 • **SerDes MAC Interface Pins**

Name	Pin	Type	Description
RDN_0 RDP_0	P16 P15	O, ADIFF	PHY0 QSGMII/SGMII/SerDes MAC receiver output pair.
RDN_1 RDP_1	T12 R12	O, ADIFF	SGMII/SerDes MAC receiver output pair.
SerDes_Rext_0	P14	ABIAS	SerDes bias pins. Connect to a 620 Ω 1% resistor between SerDes_Rext_0 and SerDes_Rext_1.
SerDes_Rext_1	N13	ABIAS	SerDes bias pins. Connect to a 620 Ω 1% resistor between SerDes_Rext_0 and SerDes_Rext_1.
TDN_0 TDP_0	N16 N15	I, ADIFF	PHY0 QSGMII/SGMII/SerDes MAC transmitter input pair.
TDN_1 TDP_1	T13 R13	I, ADIFF	SGMII/SerDes MAC transmitter input pair.

6.3.8 SerDes Media Interface

The following table lists the SerDes media interface pins.

Table 530 • **SerDes Media Interface Pins**

Name	Pin	Type	Description
FIBRIN_0 FIBRIN_1	T15 T11	I, ADIFF	SerDes media receiver input pair.
FIBRIP_0 FIBRIP_1	R15 R11	I, ADIFF	SerDes media receiver input pair.
FIBRON_0 FIBRON_1	T14 T10	O, ADIFF	SerDes media transmitter output pair.
FIBROP_0 FIBROP_1	R14 R10	O, ADIFF	SerDes media transmitter output pair.

6.3.9 Serial Management Interface

The following table lists the serial management interface (SMI) pins. The SMI pins are referenced to VDD25 and can be set to a 2.5 V power supply.

Table 531 • **SMI Pins**

Name	Pin	Type	Description
MDC	P2	I, PD, 3 V	Management data clock. A 0 MHz to 12.5 MHz reference input is used to clock serial MDIO data into and out of the PHY.
MDINT	M2	I/O, OS, OD	Management interrupt signal. Upon reset the device will configure these pins as active-low (open drain) or active-high (open source) based on the polarity of an external 10 kΩ resistor connection. These pins can be tied together in a wired-OR configuration with only a single pull-up or pull-down resistor.
MDIO	N2	I/O, OD	Management data input/output pin. Serial data is written or read from this pin bidirectionally between the PHY and Station Manager, synchronously on the positive edge of MDC. One external pull-up resistor is required at the Station Manager, and its value depends on the MDC clock frequency and the total sum of the capacitive loads from the MDIO pins.
NRESET	M3	I, PD, ST, 3 V	Device reset. Active low input that powers down the device and sets all register bits to their default state.

6.3.10 Twisted Pair Interface

The following table lists the twisted pair interface pins.

Table 532 • **Twisted Pair Interface Pins**

Name	Pin	Type	Description
TXVNA_0 TXVNA_1	B14 B10	ADIFF	TX/RX channel A negative signal
TXVNB_0 TXVNB_1	B15 B11	ADIFF	TX/RX channel B negative signal
TXVNC_0 TXVNC_1	C15 B12	ADIFF	TX/RX channel C negative signal

Table 532 • **Twisted Pair Interface Pins *(continued)***

Name	Pin	Type	Description
TXVND_0 TXVND_1	D15 B13	ADIFF	TX/RX channel D negative signal
TXVPA_0 TXVPA_1	A14 A10	ADIFF	TX/RX channel A positive signal
TXVPB_0 TXVPB_1	A15 A11	ADIFF	TX/RX channel B positive signal
TXVPC_0 TXVPC_1	C16 A12	ADIFF	TX/RX channel C positive signal
TXVPD_0 TXVPD_1	D16 A13	ADIFF	TX/RX channel D positive signal

7 Package Information

VSC8572XKS-02 and VSC8572XKS-05 are packaged in a lead(Pb)-free, 256-pin, plastic ball grid array (BGA) with a 17 mm × 17 mm body size, 1 mm pin pitch, and 1.8 mm maximum height.

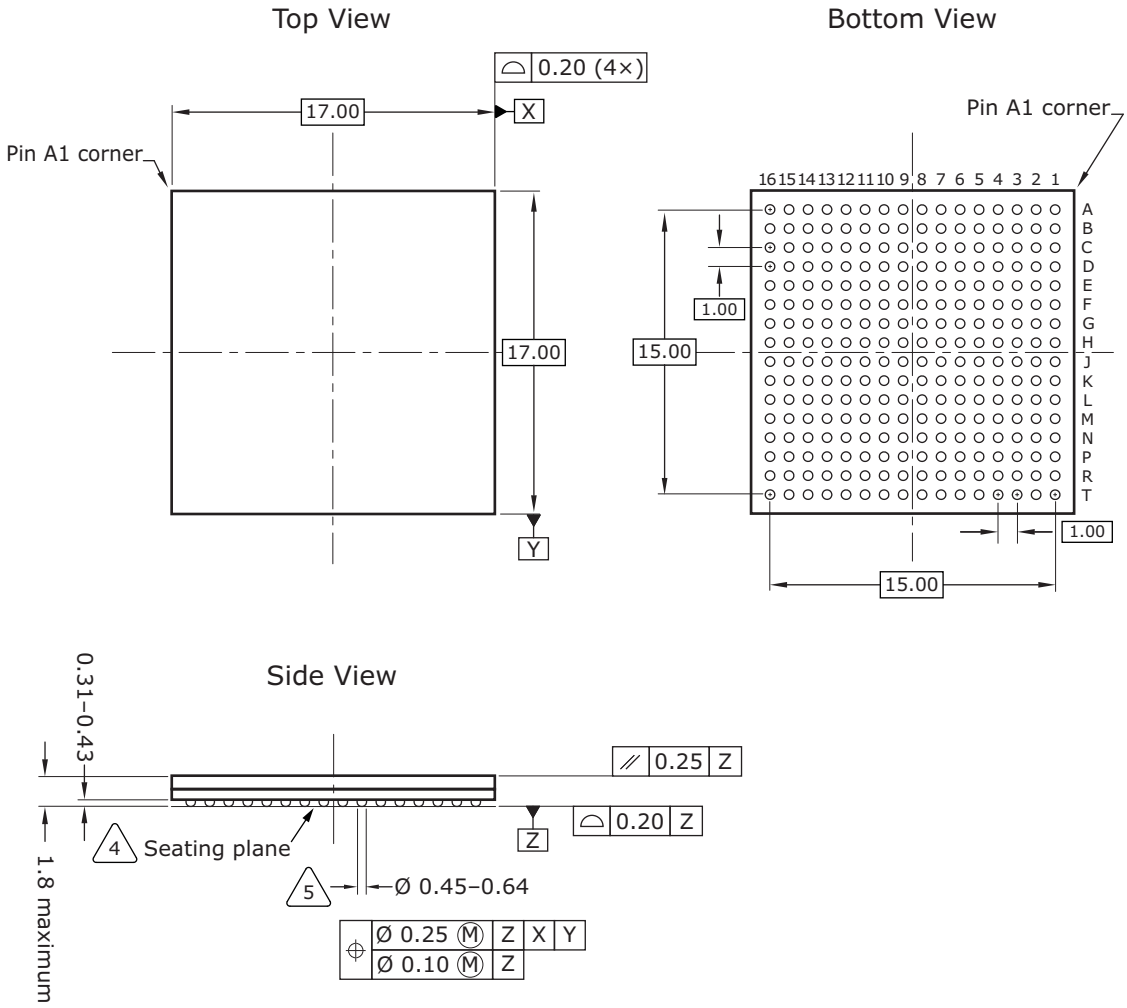
Lead(Pb)-free products from Microsemi comply with the temperatures and profiles defined in the joint IPC and JEDEC standard IPC/JEDEC J-STD-020. For more information, see the IPC and JEDEC standard.

This section provides the package drawing, thermal specifications, and moisture sensitivity rating for the VSC8572-02 device.

7.1 Package Drawing

The following illustration shows the package drawing for the VSC8572-02 device. The drawing contains the top view, bottom view, side view, dimensions, tolerances, and notes.

Figure 98 • **Package Drawing**



Notes

1. All dimensions and tolerances are in millimeters (mm).
2. Ball diameter is 0.50 mm.
3. Radial true position is represented by typical values.
4. Primary datum Z and seating plane are defined by the spherical crowns of the solder balls.
5. Dimension is measured at the maximum solder ball diameter, parallel to primary datum Z.

7.2 Thermal Specifications

Thermal specifications for this device are based on the JEDEC JESD51 family of documents. These documents are available on the JEDEC Web site at www.jedec.org. The thermal specifications are modeled using a four-layer test board with two signal layers, a power plane, and a ground plane (2s2p

PCB). For more information about the thermal measurement method used for this device, see the JESD51-1 standard.

Table 533 • **Thermal Resistances**

Symbol	°C/W	Parameter
θ_{JCtop}	5.9	Die junction to package case top
θ_{JB}	12.7	Die junction to printed circuit board
θ_{JA}	22	Die junction to ambient
θ_{JMA} at 1 m/s	18.5	Die junction to moving air measured at an air speed of 1 m/s
θ_{JMA} at 2 m/s	16.3	Die junction to moving air measured at an air speed of 2 m/s

To achieve results similar to the modeled thermal measurements, the guidelines for board design described in the JESD51 family of publications must be applied. For information about applications using BGA packages, see the following:

- JESD51-2A, *Integrated Circuits Thermal Test Method Environmental Conditions, Natural Convection (Still Air)*
- JESD51-6, *Integrated Circuit Thermal Test Method Environmental Conditions, Forced Convection (Moving Air)*
- JESD51-8, *Integrated Circuit Thermal Test Method Environmental Conditions, Junction-to-Board*
- JESD51-9, *Test Boards for Area Array Surface Mount Package Thermal Measurements*

7.3 Moisture Sensitivity

This device is rated moisture sensitivity level 4 as specified in the joint IPC and JEDEC standard IPC/JEDEC J-STD-020. For more information, see the IPC and JEDEC standard.

8 Design Considerations

This section provides information about design considerations for the VSC8572-02 device.

8.1 Link status LED remains on while COMA_MODE pin is asserted high

When the COMA_MODE is asserted high, the link status LED may not deactivate unless the media cable is disconnected from the device.

While using COMA_MODE, link status should be verified using status registers rather than LED indicators.

8.2 LED pulse stretch enable turns off LED pins

Enabling the pulse stretch function for LED0 or LED1 by setting register 30, bits 5:6 shuts off those LED pins.

Use the default blink function setting of LED0 and LED1 rather than pulse stretching. For more information, see LED Behavior, page 111.

8.3 AMS and 100BASE-FX

When the PHY operating mode (set in register 23) is AMS and the current active media is 100BASE-FX, register 0 bit 12 will be 0. This would normally indicate that auto-negotiation is disabled and the PHY is in forced mode. But in this mode, it has other meanings.

The workaround is to ensure that bit 12 is always written as 1 when doing writes or updates to register 0 in AMS mode.

8.4 10BASE-T signal amplitude

10BASE-T signal amplitude can be lower than the minimum specified in IEEE 802.3 paragraph 14.3.1.2.1 (2.2 V) at low supply voltages.

This issue is not estimated to present any system level impact. Performance is not impaired with cables up to 130 m with various link partners.

8.5 10BASE-T link recovery failures

If the link disconnects when traffic is flowing while the device operates in a 10BASE-T mode, the PHY may not re-link.

There is a software workaround for this issue in which the device's internal microcontroller monitors link transitions in 10BASE-T mode and forces a soft power-down/power-up procedure to prevent a re-link failure.

A side effect of this software workaround is that the counts in registers 20 and 21 will be cleared (For more information, see Error Counter 2, page 105 and Error Counter 3, page 105).

8.6 SNR degradation and link drops

The link may drop after approximately 100 master/slave relationship swaps with the ring resiliency feature when using Category 5 (Cat5) cables that are longer than 75 m.

The workaround is to use a combination of an initialization script and a procedure change. Contact Microsemi for the workaround solution if the ring resiliency feature is being enabled.

8.7 Clause 45 register 3.22

The clause 45, register 3.22 is cleared upon read only when the extended page access register (register 31) is set to 0.

This register cannot be read when the page access register is set to a value other than 0.

The workaround is to set extended page access register to 0 before accessing clause 45, register 3.22.

8.8 Clause 45 register 3.1

Clause 45, register 3.1, Rx and Tx LPI received bits are cleared upon read only when the extended page access register (register 31) is set to 0.

This has a minor implication for software that needs to ensure that the extended page access register is set to 0 before reading clause 45, register 3.1.

The workaround is to set extended page access register to 0 before accessing clause 45, register 3.1.

8.9 Clause 45 register address post-increment

Clause 45 register address post-increment only works when reading registers and only when extended page access register (register 31) is set to 0.

The workaround is to access the registers individually.

8.10 Fast link failure indication

The fast link failure indication for all the ports is enabled using port 0, register 19E.4.

The workaround is to set register 19E.4 = 1 in PHY 0 to enable Fast Link Fail indication.

8.11 Timestamp accuracy in 10BASE-T mode

Timestamp accuracy in 10BASE-T mode is ±400 ns.

Timing accuracy is reduced on networks running in 10BASE-T mode. There is currently no workaround for this issue.

8.12 Near-end loopback with AMS enabled

Near-end loopback does not work when AMS is enabled. Near-end loopback is controlled by setting bit 14 of register 0.

The workaround is to disable AMS when enabling loopback. This is a debug feature and does not have any real life implications.

8.13 Carrier detect assertion

Carrier detect assertion is set to false incorrectly when 9 out of 10 bits in the K28.1 word are in error.

No real life implication is expected, because the event that can trigger this error is extremely unlikely. If it does occur, the link may drop momentarily and come back up.

8.14 Link status not correct in register 24E3.2 for 100BASE-FX operation

The link status in register 24E3.2 only reflects the status of 1000BASE-X links. It does not reflect the status of 100BASE-FX links.

The workaround is to check register 28.4:3 for media operating mode (10 for fiber), 28.4:3 for speed status (100 for 100 Mbps), and then check 16.12 for current link status.

8.15 Register 28.14 does not reflect autonegotiation disabled in 100BASE-FX mode

Register 28.14 does not reflect autonegotiation status in 100BASE-FX mode. It works correctly in all copper and 1000BASE-X media modes.

The workaround is to use register 0.12 for autonegotiation status in 100BASE-FX mode when AMS is disabled. For more information about limitations when AMS is enabled, see AMS and 100BASE-FX, page 327.

8.16 Near-end loopback non-functional in protocol transfer mode

Near-end loopback does not work correctly when the device is configured in protocol transfer mode.

This is a debug feature and does not have any effect on the normal operation of the device.

8.17 Fiber-media CRC counters non-functional in protocol transfer mode at 10 Mbps and 100 Mbps

Packets received on the media SerDes interface will not be counted correctly in registers 28E3 and 29E3 when the device is configured in protocol transfer mode and operating at 10 Mbps or 100 Mbps speeds.

These counters are used for debugging and there is no effect on the normal operation of the device.

8.18 Fiber-media recovered clock does not squelch based on link status

To squelch the clock in fiber media mode, code sync status is used instead of link status. This causes the clock to not be squelched if the device is configured in 1000BASE-X mode with autonegotiation enabled when the transmit fiber is unplugged.

There is a software workaround for this issue where the device's internal microcontroller monitors link status and forces the clock off when no link is present.

8.19 1000BASE-X parallel detect mode with Clause 37 autonegotiation enabled

When connected to a forced-mode link partner and attempting autonegotiation, the PHY in 1000BASEX parallel detect mode requires a minimum 250 ms IDLE stream in order to establish a link. If the PHY port is programmed with 1000BASE-X parallel detect-enabled (MAC-side register 16E3 bit 13, or media-side register 23E3 bit 13), then a forced-mode link partner sending traffic with an inter-packet gap less than 250 ms will not allow the local device’s PCS to transition from a link-down to link-up state.

8.20 Anomalous PCS error indications in Energy Efficient Ethernet mode

When a port is processing traffic with Energy Efficient Ethernet enabled on the link, certain PCS errors (such as false carriers, spurious start-of-stream detection, and idle errors) and EEE wake errors may occur. There is no effect on traffic bit error rate for cable lengths up to 75 meters, and minor packet loss may occur on links longer than 75 meters. Regardless of cable length, some error indications should not be used while EEE is enabled. These error indications include false carrier interrupts (Interrupt Status register 26 bit 3), receive error interrupts (Interrupt Status register 26 bit 0), and EEE wake error interrupts.

Contact Microsemi for a script that needs to be applied during system initialization if EEE will be enabled.

8.21 Long link-up times while in forced 100BASE-TX mode of operation

While in forced 100BASE-TX operation and attempting to link up, the device may experience abnormally long link-up times.

This issue can only occur if the Unified API is not used with the device. In those circumstances, the workaround for this issue is to clear all speed advertisements in the autonegotiation advertisement

registers (register 4, bits 9:5 and register 9, bits 9:8), then toggle the auto-negotiation enable bit of the mode control register (register 0, bit 12) for a port upon detecting its link is down. Any advertisements temporarily cleared can then be restored once register 0, bit 12 is cleared.

Contact Microsemi for the latest code sequence included in the Unified API.

8.22 Timestamp errors due to IEEE 1588 Reference Clock interruption

Interruption of the IEEE 1588 reference clock after release of device hardware reset will corrupt the local time counter (LTC) value. After clock interruption, an LTC reload is required using the Unified API.

8.23 1588 bypass shall be enabled during engine reconfiguration

When the 1588 datapath is enabled, the 1588 bypass feature shall be enabled before reprogramming 1588 configuration registers. It is recommended to disable 1588 bypass before live traffic begins flowing through the re-provisioned port.

8.24 Missing clock pulses on serial timestamp output interface

The serial timestamp output interface may not generate the final 1588_SPI_CLK cycle for certain timestamp push-out transactions. This issue can be worked around by programming the SI_CLK_LO_CYCS to value 0x1.

Use the latest PHY API for a workaround to this issue.

8.25 Station managers cannot use MDIO address offsets 0x2 and 0x3 with the PHY

In addition to responding to the two lowest MDIO addresses that can be resolved with device serial management interface address bits 4:2, the device will unexpectedly respond to offsets that have bit 1 set (for a detailed addressing diagram, see SMI Frames, page 74). However, PHY2 and PHY3 are unusable targets on this device, so their corresponding MDIO addresses must not be used by the SMI station manager that controls this device. It is essential to avoid assigning addresses to other devices on the bus that would overlap the 0x2 and 0x3 offsets.

The workaround for this issue is to ensure the station manager connected to this device avoids using the two MDIO addresses immediately following the PHY1 target.

9 Ordering Information

The VSC8572 device is offered with two operating temperature ranges. The range for VSC8572-02 is 0 °C ambient to 125 °C junction, and the range for VSC8572-05 is -40 °C ambient to 125 °C junction.

VSC8572XKS-02 and VSC8572XKS-05 are packaged in a lead(Pb)-free, 256-pin, plastic ball grid array (BGA) with a 17 mm × 17 mm body size, 1 mm pin pitch, and 1.8 mm maximum height.

Lead(Pb)-free products from Microsemi comply with the temperatures and profiles defined in the joint IPC and JEDEC standard IPC/JEDEC J-STD-020. For more information, see the IPC and JEDEC standard.

The following table lists the ordering information for the VSC8572-02 device.

Table 534 • **Ordering Information**

Part Order Number	Description
VSC8572XKS-02	Lead-free, 256-pin, plastic BGA with a 17 mm × 17 mm body size, 1 mm pin pitch, and 1.8 mm maximum height. The operating temperature is 0 °C ambient to 125 °C junction[1].
VSC8572XKS-05	Lead-free, 256-pin, plastic BGA with a 17 mm × 17 mm body size, 1 mm pin pitch, and 1.8 mm maximum height. The operating temperature is –40 °C ambient to 125 °C junction[1].

1. For carrier class applications, the maximum operating temperature is 110 °C junction.