UG0904
User Guide
PolarFire SoC FPGA Gigabit Ethernet MAC

Microsemi
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.

50200904. 1.0 4/20

Contents

Figures

Tables

1 Revision History

The revision history describes the changes that were implemented in the document. The changes are listed by revision, starting with the current publication.

1.1 Revision 1.0

The first publication of this document.

2 Acronyms

The following acronyms are used in this document.

Table 1 • **List of Acronyms**

Acronym	Expanded
APB	Advanced Peripheral Bus
AXI	Advanced eXtensible Interface
DMA	Direct Memory Access
FCS	Frame Check Sequence
GEM	Gigabit Ethernet MAC
GMII	Gigabit Media Independent Interface
IEEE	Institute of Electrical and Electronics Engineers
IPG	Inter Packet Gap
MAC	Media Access Control
MDIO	Management Data Input/Output
MII	Media Independent Interface
MMSL	MAC Merge Sublayer
MSS	Microprocessor Subsystem
PCS	Physical Coding Sublayer
PTP	Precision Time Protocol
SGMII	Serial Gigabit Media Independent Interface
TCP/IP	Transmission Control Protocol/Internet Protocol
TSN	Time Sensitive Networking
TSU	Time Stamping Unit
UDP	User Datagram Protocol

3 PolarFire SoC Gigabit Ethernet MAC

PolarFire SoC devices contain industry's first RISC-V multi-core Microprocessor Subsystem (MSS) with 5x 64-bit RISC-V processor cores. The PolarFire SoC MSS contains two hardened Gigabit Ethernet MAC IP blocks—GEM_0 and GEM_1— to enable Ethernet solutions over copper or optical cabling. As a prerequisite, see *UG0880: PolarFire SoC FPGA MSS User Guide* to know MSS buses/blocks. GEM_0 and GEM_1 are functionally identical, hence, GEM_0 and GEM_1 are referred as GEM throughout the document.

GEM supports 10 Mb/s, 100 Mb/s, and 1000 Mb/s (1 Gb/s) speeds. GEM provides a complete range of solutions for implementing IEEE 802.3 standard-compliant Ethernet interfaces for chip-to-chip, board-to-board, and backplane interconnects. The IP is configured using the PFSOC_MSS IP Configurator in Libero SoC.

3.1 Features

GEM supports the following features.

- IEEE 802.3 compliant
- IEEE 802.1Q TSN features:
  - IEEE 802.1AS
  - IEEE 802.1Qav
  - IEEE 802.1Qbv
  - IEEE 802.1CB frame redundancy and elimination
  - IEEE 802.1Qci receive (ingress) traffic policing
  - IEEE 802.3br frame preemption (or interspersing express traffic)
  - IEEE 802.1Qbb priority-based flow control
  - IEEE 802.1Q VLAN tagging with recognition of incoming VLAN and priority tagged frames
- DMA support
- TCP/IP offloading capability
- Integrated 1000 BASE-X PCS for SGMII-based applications
- Programmable jumbo frames up to 10,240 bytes
- Frame Filtering
- Full and half duplex modes at 10/100M and full duplex at 1Gbps interface speeds for MII, GMII, and SGMII.
- Wake-on LAN support

3.2 Overview

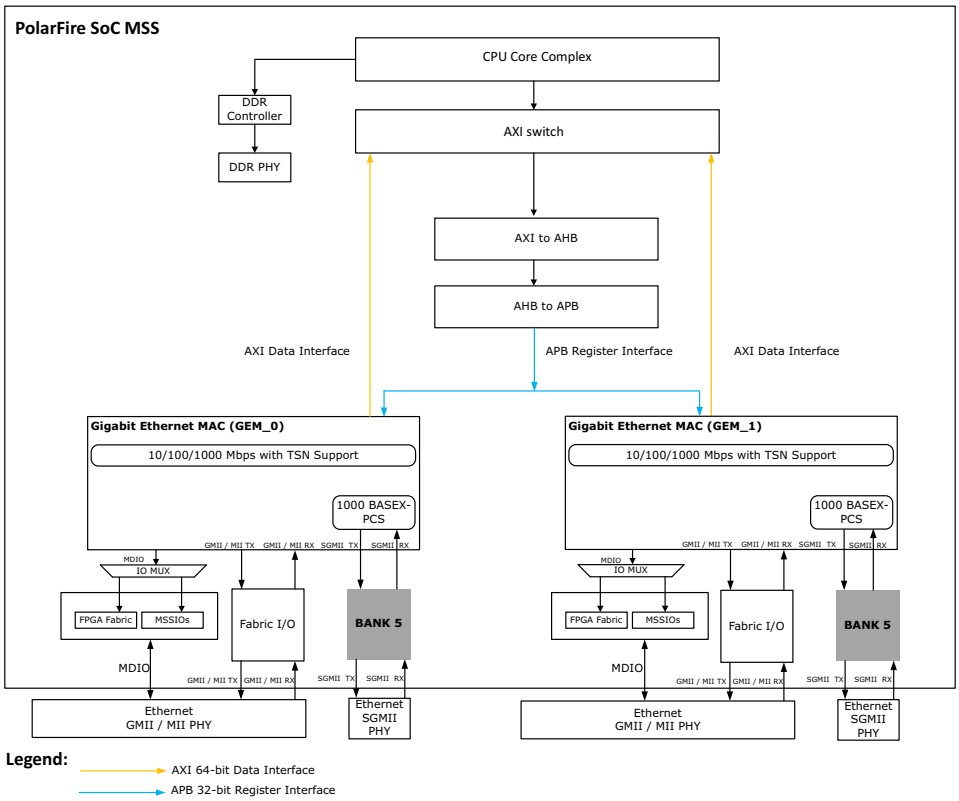
GEM is accessed by the CPU Core Complex via the AXI switch using the following interfaces:

- AXI interface—used for data transfers
- APB interface—used for configuration purpose

GEM can be configured for SGMII or MII/GMII.The MII/GMII is only connected to the FPGA fabric. The PCS sub-block performs the 8b/10b operation for SGMII. SGMII is connected to the I/O BANK 5. Management Data Input/Output (MDIO) interface signals can be routed either from the FPGA fabric or from a dedicated MSSIO. The external PHY registers are configured using management interface (MDIO) of the GEM.

Figure 1 shows a high-level block diagram of GEM blocks.

Figure 1 • **High-Level Block Diagram**



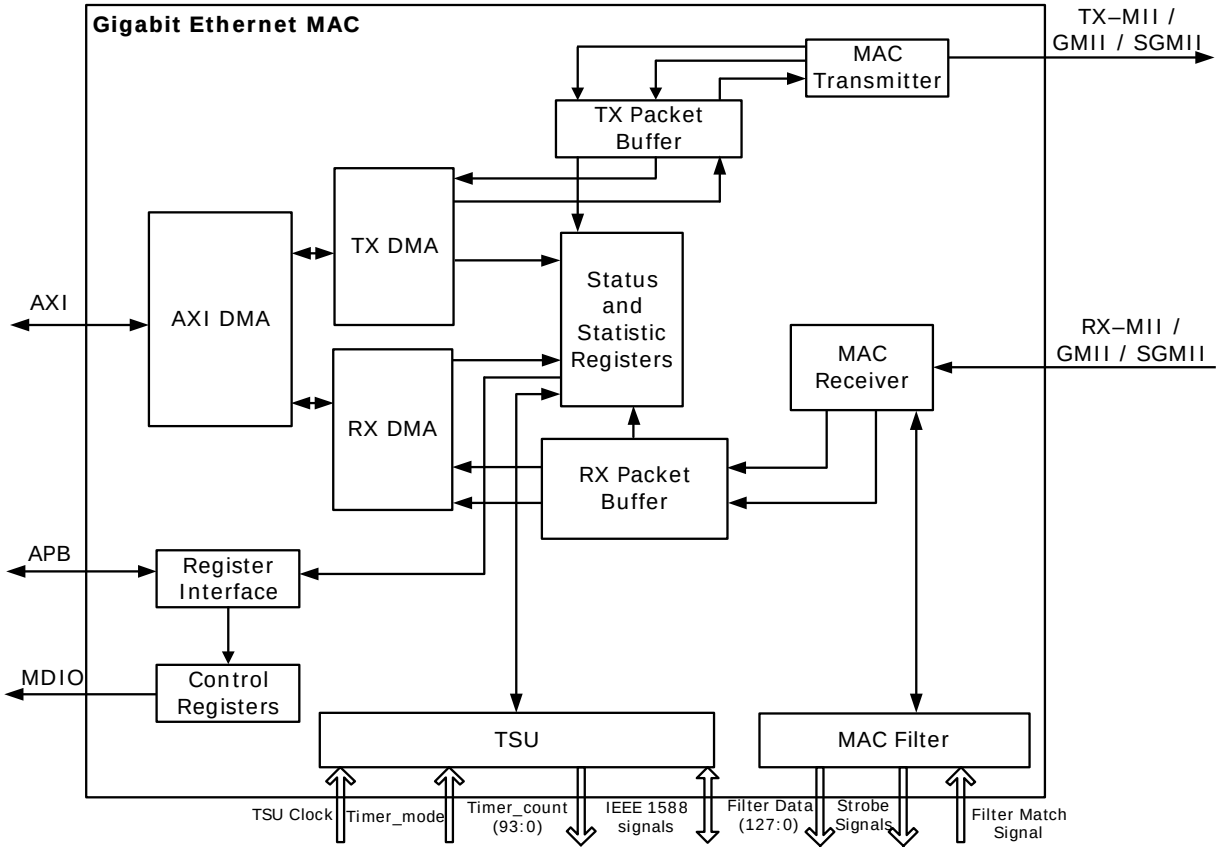
4 Functional Description

GEM includes the following functional blocks:

- Integrated 1000BASE-X Physical Coding Sublayer (PCS) for encoding and decoding the data and for Auto Negotiation (AN)
- Time Stamping Unit (TSU) for timer operations
- TSN block to support Timing Sensitive Networking (TSN) features
- High-speed AXI DMA block to transfer data to and from the processor
- Filter block filters out the received frames

Figure 2 shows the detailed functional block diagram of GEM.

Figure 2 • **Functional Block Diagram**



4.1 MAC Transmitter

The MAC transmitter block retrieves data from the memory using DMA controller, which is connected through the AXI interface. DMA reads the data from memory using the AXI master interface and stores it to the TX packet buffer. Then the MAC transmitter block retrieves the data from the TX packet buffer and adds preamble and, if necessary, pad and Frame Check Sequence (FCS). The data is transmitted out using configured data interface such as MII, GMII, or SGMII.

Both half-duplex and full-duplex Ethernet modes of operation are supported. When operating in half-duplex mode, the MAC transmitter block generates data according to the Carrier Sense Multiple Access with Collision Detect (CSMA/CD) protocol. The start of transmission is deferred if Carrier Sense (CS) is active. If collision (col) becomes active during transmission, a jam sequence is asserted, and the transmission is re-tried after a random back off. The CS and col signals have no effect in full-duplex mode.

According to the IEEE 802.3 standards, an Ethernet MAC must allow a minimum amount of time before another packet is sent. This pause time between packets is known as Inter Packet Gap (IPG). The purpose of the inter-packet gap is to allow enough time for the receiver to recover the clock and to perform cleanup operations. During this period IDLE packets will be transmitted. The standard minimum inter packet gap for transmission is 96 bit times. Using GEM, the IPG may be stretched beyond 96 bits depending on the length of the previously transmitted frame. The IPG stretch only works in the full-duplex mode.

4.2 MAC Receiver

MAC receiver block receives data using MII, GMII, or SGMII interface and stores the data in the RX packet buffer. Using RX DMA controller, data from the RX packet buffer is read and transferred to memory using AXI interface.

The MAC receive block checks for valid preamble, FCS, alignment and length, and presents received frames to the MAC address checking block. Firmware can configure GEM to receive jumbo frames up to 10,240 bytes.

The address checker identifies:

- Four source or destination specific 48-bit addresses
- Four different types of ID values
- A 64-bit hash register for matching multi-cast and unicast addresses as required.
- Broadcast address of all ones, copy all frames and act on external address matching signals.

Supports offloading of IP, TCP, and UDP checksum calculations (both IPv4 and IPv6 packet types are supported) and can automatically discard frames with a bad checksum. As the MAC supports TSN features, it identifies 802.1CB streams and automatically eliminates duplicate frames. Statistics are provided to report counts of rogue and out-of-order frames, latent errors, and the timer reset events.

During frame reception, if the frame is found to be too long, a bad frame indication is sent to the DMA controller and the receiver logic does not send the frame memory. At the end of frame reception, the receive block indicates to the DMA block whether the frame is good or bad. The DMA block recovers the current receive buffer if the frame is bad.

4.3 Register Interface

Control registers drive the MDIO interface, set up DMA activity, start frame transmission and select modes of operation such as full-duplex, half-duplex, and 10/100/1000 Mbps operation. The register interface is through AMBA APB interface, which connects to the core complex subsystem.

The statistics register block contains registers for counting various types of an event associated with transmit and receive operation. These registers, along with the status words stored in the receive buffer list, enable the software to generate network management statistics registers.

4.4 AXI DMA

Built-in DMA controller supports Direct Memory Access (DMA). DMA is attached to the MAC buffer memories to provide a scatter gather type capability for packet data storage in an embedded processor system or System on Chip (SoC).

DMA uses the AXI interface for data transfer and uses the APB interface for configuration and monitoring DMA descriptors. DMA uses separate transmit and receive buffers as memories to store the frames to be transmitted or received. It uses separate transmit and receive lists of buffer descriptors, with each descriptor describing a buffer area in the memory. This allows the Ethernet packets to be broken up and scattered around the system memory.

TX DMA is responsible for transmit operations and RX DMA is responsible for receive operations. TX DMA reads the data from memory, which is connected through the AXI interface and stores data to the transmit packet buffers. RX DMA fetches the data from the receive packet buffers and transfers it to the application memory.

Receive buffer depth is programmable from the range of 64 bytes to 16320 bytes. The start location for each receive buffer is stored in the memory in a list of receive buffer descriptors, at an address location pointed by the receive buffer queue pointer. The base address for the receive buffer queue pointer is configured using the DMA registers.

Transmit frames can be in the range of 14 bytes to 10,240 bytes long. As a result, it is possible to transmit jumbo frames. The start location for each transmit buffer is stored in a list of transmit buffer descriptors at a location pointed by the transmit buffer queue pointer. The base address for this queue pointer is configured using the DMA registers.

Following are the features of DMA Controller:

- 64-bit data bus width support
- 64-bit address bus width support
- Support up to 16 outstanding AXI transactions. These transactions can cross multiple frame transfers.
- Ability to store multiple frames in the packet buffer resulting in the maximum line rate
- Supports priority queuing
- Supports TCP/IP advanced offloads to reduce CPU overhead

AXI read operations are routed to the AXI read channel and all write operations to the write channel. Both read and write channels may operate simultaneously. Arbitration logic is implemented when multiple requests are active on the same channel. For example, when the transmit DMA requests for transmission of data, at the same time the receive DMA requests for the reception of data. In these cases, the receive DMA is granted the bus before the transmit DMA. However, most requests are either receive data writes or transmit data reads both of which can operate in parallel and can execute simultaneously.

4.5 MAC Filter

The filter block determines, which frames are written to the DMA interface. Filtering is performed on received frames based on the state of the external matching pins, the contents of the specific address, type and hash registers, and the frame's destination address, and the field type.

GEM supports the following filtering mechanisms:

- Internal Filter
- External Filter and Receive Priority Queuing

4.5.1 Internal Filter

GEM is configured to have four specific address filters. Each filter is configured to contain a MAC address, which is specified to be compared against the Source Address (SA) or Destination Address (DA) of each received frame. There is also a mask field to allow certain bytes of the address that are not to be included in the comparison. If the filtering matches for a specific frame, then it is passed on to the DMA memory. Otherwise, the frame is dropped.

Frames may also be filtered using the Type ID field for matching. There are four types of ID registers in the internal register space, and these may be enabled individually. GEM supports the recognition of specific source or destination addresses. The number of a specific source or destination address filters are configurable and can range from 0 (zero) to 4.

4.5.2 External Filter and Receive Priority Queuing

In internal filter, filtering is done based on either source or destination address. To allow for more sophisticated matching of incoming frames a circuitry can be designed in the FPGA fabric. As a frame is received, the MAC parses the frame and determines what field is currently present. A strobe signal is provided to allow the latching of each field in the fabric. Customized matching circuitry can then analyze whether it is interested in that specific frame (that is, filtering use case) or what is the receive priority queue to put the frame in (that is, prioritizing use case).

In order to allow this customization of filtering or priority queuing to be done by the user in the FPGA fabric, the receive data along with strobe signals are fed to the FPGA fabric. The user logic must assert one of four match signals back to the MAC, within a specified time, if it determines that the frame is to be stored in the DMA memory. The incoming match signal must be gated off by Ethernet MAC until the specified time budget has elapsed, in order to ensure that no glitch is fed through the combinational logic cloud. For the filtering use case, the match must be asserted before the destination address reaches the end of the receive pipeline.

4.6 Time Stamping Unit

TSU implements a timer, which counts the time in seconds and nanoseconds format. This block is supplied with tsu_clk, which ranges from 5 MHz to 400 MHz. The timer is implemented as a 94-bit register as follows.

- The upper 48 bits counts seconds
- The next 30 bits counts nanoseconds
- The lower 16 bits counts sub nanoseconds

The timer increments at each tsu_clk period and an interrupt is generated the seconds increment. The timer value can be read, written, and adjusted through the APB interface.

There are two modes of operation:

- Timer Adjust Mode
- Increment Mode

4.6.1 Timer Adjust Mode

In timer adjust mode, the tsu_clk is supplied from the FPGA fabric. The maximum clock frequency is 125 MHz. There are several signals, synchronous to tsu_clk output by the MAC.

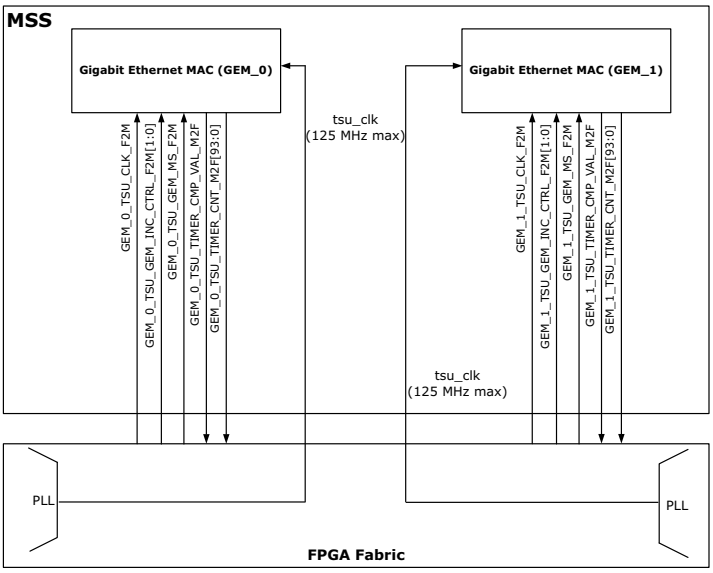
In this mode, the timer operation is also controlled from the fabric by input signals called gem_tsu_inc_ctrl [1:0] along with gem_tsu_ms.

When the gem_tsu_inc_ctrl [1:0] is set to:

- 2b'11 – Timer register increments as normal
- 2b'01 – Timer register increments by an additional nanosecond
- 2b'10 – Timer increments by a nanosecond less
- 2b'00:
  - When the gem_tsu_ms is set to: logic 1, the nanoseconds timer register is cleared and the seconds timer register is incremented with each clock cycle.
  - When the gem_tsu_ms is set to: logic 0, the timer register increments as normal, but the timer value is copied to the sync strobe register.

The TSU timer count value can be compared to a programmable comparison value. For the comparison, the 48 bits of the seconds value and the upper 22 bits of the nanoseconds value are used. The timer_cmp_val signal is output from the core to indicate when the TSU timer value is equal to the comparison value stored in the timer comparison value registers.

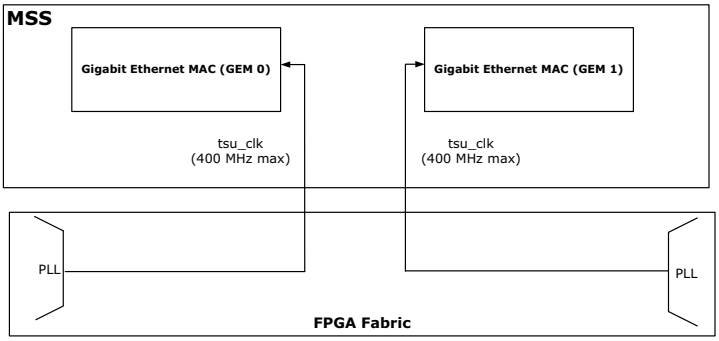
Figure 3 • **TSU from Fabric (Timer Adjust Mode)**



4.6.2 Increment Mode

In the increment mode, the tsu_clk is supplied either from an external reference clock or from the FPGA fabric. The maximum clock frequency is 400 MHz. In this mode, the timer signals interfacing the FPGA fabric are gated off.

Figure 4 • **TSU from MSS (Increment Mode)**



4.7 IEEE 1588 Implementation

IEEE 1588 is a standard for precision time synchronization in local area networks. It works with the exchange of special PTP frames. The PTP messages can be transported over IEEE 802.3/Ethernet, over Internet Protocol Version 4 or over Internet Protocol Version 6. GEM detects when the PTP event messages: sync, delay_req, pdelay_req, and pdelay_resp are transmitted and received. GEM asserts various strobe signals for different PTP event messages.

GEM supports the following functionalities:

- Identifying PTP frames
- Extracting time stamp information out of received PTP frames
- Inserting time stamp information into received data frames, before passing to buffer memory
- Inserting time stamp information into transmitted data frames
- Allowing control of TSU either through MSS or FPGA fabric

GEM samples the TSU timer value when the TX or RX SOF event of the frame passes the MII/GMII boundary. This event is an existing signal synchronous to MAC TX/RX clock domains. The MAC uses the sampled time stamp to insert the time stamp into transmitted PTP sync frames (if one step sync feature is enabled) or to pass to the register block to capture the time stamp in APB accessible registers, or to pass to the DMA to insert into TX or RX descriptors. For each of these, the SOF event, which is captured in the tx_clk and rx_clk domains respectively, is synchronized to the tsu_clk domain and the resulting signal is used to sample the TSU count value.

There is a difference between IEEE 802.1 AS and IEEE 1588. The difference is IEEE 802.1AS uses the Ethernet multi cast address 0180C200000E for sync frame recognition whereas IEEE 1588 does not. GEM is designed to recognize sync frames with both 802.1AS and 1588 addresses and so can support both 1588 and 802.1AS frame recognition simultaneously.

4.7.1 PTP Strobes

There are number of strobe signals from the GEM to the FPGA fabric. These signals indicate the transmission/reception of various PTP frames. Table 2 lists these signals.

Table 2 • **PTP Strobe Signals**

Signal Name	Description
DELAY_REQ_RX	Asserted when the PTP RX delay request is detected.
DELAY_REQ_TX	Asserted when the PTP TX delay request is detected.
PDELAY_REQ_RX	Asserted when the PTP PDELAY RX request is detected.

Table 2 • **PTP Strobe Signals *(continued)***

Signal Name	Description
PDELAY_REQ_TX	Asserted when the PTP PDELAY TX request is detected.
PDELAY_RESP_RX	Asserted when the PTP PDELAY RX response request is detected.
PDELAY_RESP_TX	Asserted when the PTP PDELAY TX response request is detected.
SOF_RX	Asserted on SFD, de-asserted at EOF.
SOF_TX	Asserted on SFD, de-asserted at EOF.
SYNC_FRAME_RX	Asserted when the SYNC_FRAME RX response request is detected.
SYNC_FRAME_TX	Asserted when the SYNC_FRAME TX response request is detected.

4.7.2 PTP Strobe Usage (GMII)

When GEM is configured in the GMII/MII mode, transmit PTP strobes are synchronous to mac_tx_clk and receive PTP strobes are synchronous to mac_rx_clk. GEM sources these clocks from the fabric.

4.7.3 PTP Strobe Usage (SGMII)

When GEM is configured in the SGMII mode, the PTP strobes must be considered asynchronous because the TX and RX clocks are not available in the FPGA fabric. Hence, the strobe signals must be synchronized with a local clock in the fabric before being used.

4.8 Time Sensitive Networking

GEM includes the following key TSN functionalities among others:

- IEEE 802.1 Qav Support – Credit based Shaping
- IEEE 802.1 Qbv – Enhancement for Scheduled Traffic
- IEEE 802.1 CB Support
- IEEE 802.1 Qci Receive Traffic Policing
- IEEE 802.3br Support

4.8.1 IEEE 802.1 Qav Support – Credit based Shaping

A credit-based shaping algorithm is available on the two highest priority active queues and is defined in IEEE 802.1Qav Forwarding and Queuing Enhancements for Time-Sensitive Streams. Traffic shaping is enabled through the register configuration. Queuing can be handled using any of the following methods.

- Fixed priority
- Deficit Weighted Round Robin (DWRR)
- Enhanced transmission selection

Selection of the queuing method is done through register configuration. The internal registers of the GEM are described in Register Address Map.

4.8.2 IEEE 802.1 Qbv – Enhancement for Scheduled Traffic

IEEE 802.1 Qbv is a TSN standard for enhancement for scheduled traffic and specifies time aware queue-draining procedures based on the timing derived from IEEE 802.1 AS. It adds transmission gates to the eight priority queues, which allow low priority queues to be shut down at specific times to allow higher priority queues immediate access to the network at specific times.

GEM supports IEEE 802.1Qbv by allowing time-aware control of individual transmit queues. GEM has the ability to enable and disable transmission on a particular queue on a periodic basis with the on or off cycling starting at a specified TSU clock time.

4.8.3 IEEE 802.1 CB Support

IEEE 802.1CB “Frame Replication and Elimination for Reliability” is one of the Time Sensitive Networking (TSN) standards. Using Frame Replication and Elimination for Reliability (FRER) within a network increases the probability that a given packet is delivered using multi-path paths through the network.

The MAC supports a subset of this standard and provides the capability for stream identification and frame elimination but does not provide support for the replication of frames.

4.8.4 IEEE 802.1 Qci Receive Traffic Policing

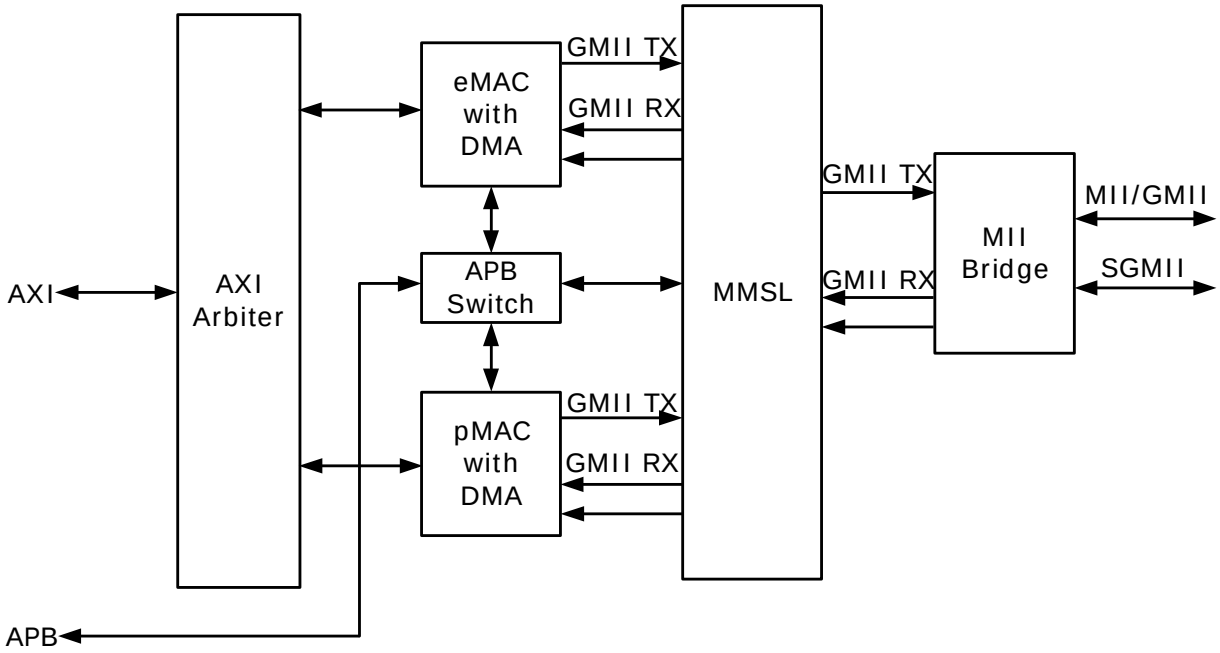
IEEE 802.11 Qci is a policing mechanism that discards frames in receive (ingress) if they exceed their allocated frame length or flow rate. TSN standards enable provisioning the resources in a network in such a way that high priority traffic is guaranteed to get through as long as it does not exceed its frame length and flow rate allocation.

4.8.5 IEEE 802.3br Support

All default operations of MAC are done by using PMAC. One more MAC, which is identical to PMAC is used termed as EMAC, which is used when IEEE 802.3br is configured. IEEE 802.3br Interspering Express Traffic is one of the TSN standards, which defines a mechanism to allow an express frame to be transmitted with minimum delay at the expense of delaying completion of normal priority frames.

This standard has been implemented by instantiating two separate MAC modules with related DMA, an MAC Merge Sub Layer (MMSL) and an AXI arbiter. One MAC is termed the express or eMAC and the other is a pre-emptable or pMAC. The eMAC is designed to carry time sensitive traffic, which must be delivered within a known time.

Figure 5 • **IEEE 802.3br Support**



4.9 PHY Interface

GEM can be configured to support the SGMII or the GMII/MII PHY. When using SGMII, the PCS block of that GEM is used.

4.9.1 Physical Coding Sublayer

A PCS is incorporated for 1000BASE-X operation which includes 8b/10b encoder, decoder, and the Auto Negotiation module. This interface is connected to I/O BANK 5.

4.9.2 GMII / MII Interface

A GMII/MII is interfaced between each MAC and the FPGA fabric, to provide flexibility to the user. It allows:

- Performing customized manipulation of data on-the-fly
- 8-bit parallel data lines are used for data transfer.
- In 10/100 Mbps mode txd[3:0] is used, txd[7:4] tied to logic 0 while transmission. rxd[3:0] is used, rxd[7:4] is tied to logic 0 during reception of data.
- In 1000 Mbps mode all txd[7:0] and rxd[7:0] bits are used.

4.9.3 SGMII

GEM includes the SGMII functional block, which provides the SGMII interface between GEM and Ethernet PHY. The SGMII block provides the following functionalities:

- Clock Domain Recovery (CDR) of received 125 MHz clock
- Serializing or De-serializing
- PLL for synthesis of a 125 MHz transmit clock

The SGMII block routes the data to the PHY via the dedicated IO BANK 5.

4.9.4 PHY Management Interface

GEM includes an MDIO interface, which can be routed through the MSSIO or the FPGA I/Os. the MDIO interface is provided to allow GEM to access the PHY's management registers. This interface is controlled by the PHY management register. Writing to this register causes a PHY management frame to send the PHY over the MDIO interface. PHY management frames are used to either write or read from PHY's control and status registers.

If desired, however, the user can just bring out one management interface (and not use the second) as it is possible to control multiple PHYs through one interface. Management Data Clock (MDC) should not toggle faster than 2.5 MHz (minimum period of 400ns), as defined by the IEEE 802.3 standard. MDC is generated by dividing down processor clock (pclk). A register configuration determines by how much pclk should be divided to produce MDC.

5 Clocking

GEM requires the following clocks:

- **tsu_clk:** Clock frequency ranges from 5 MHz to 400 MHz for the time stamp unit. Timestamp accuracy improves with higher frequencies. To support single step time stamping, tsu_clk frequency must be greater than 1/8th the frequency of tx_clk or rx_clk.
- **tx_clk:** Clock frequency ranges are: 1.25 MHz, 2.5 MHz,12.5 MHz, 25 MHz, and 125 MHz for MAC transmit clock, used by the MAC transmit block. In the 10/100 GMII mode, tx_clk runs at either 2.5 MHz or 25 MHz as determined by the external PHY MII clock input. When using gigabit mode, the transmit clock must be sourced from a 125 MHz reference clock. Depending on the system architecture, this reference clock may be sourced from an on-chip clock multiplier, generated directly from an off-chip oscillator, or taken from the PHY rx_clk. In the SGMII mode, this clock is sourced from the gtx_clk.
- **gtx_clk:** 125 MHz PCS transmit clock. In SGMII application, this is recovered from input data and needs to be balanced with the tx_clk.
- **rx_clk:** Clock frequency ranges are: 1.25 MHz, 2.5 MHz, 12.5 MHz, 25 MHz, 62.5 MHz, and 125 MHz. This clock is used by the MAC receive synchronization blocks. In the 10/100 and gigabit mode using the GMII/MII interface, this clock is sourced from the rx_clk input of the external PHY and can be either 2.5 MHz, 25 MHz, or 125 MHz.

The following table lists the required frequencies of the transmit clock.

Table 3 • **Transmit Clock Frequencies**

	gtx_clk (MHz)		tx_clk (MHz)	
MAC Speed Mode (Mbps)	**SGMII**	**GMII**	**SGMII**	**MII**
10	125	N/A	1.25	2.5
100	125	N/A	12.5	25
1000	125	125	125	125

The following table lists the required frequencies of the receive clock.

Table 4 • **Receive Clock Frequencies**

	pcs_rx_clk (MHz)		rx_clk (MHz)	
MAC Speed Mode (Mbps)	**SGMII**	**GMII/MII**	**SGMII**	**GMII/MII**
10	125	N/A	1.25	2.5
100	125	N/A	12.5	25
1000	125	N/A	62.5	125

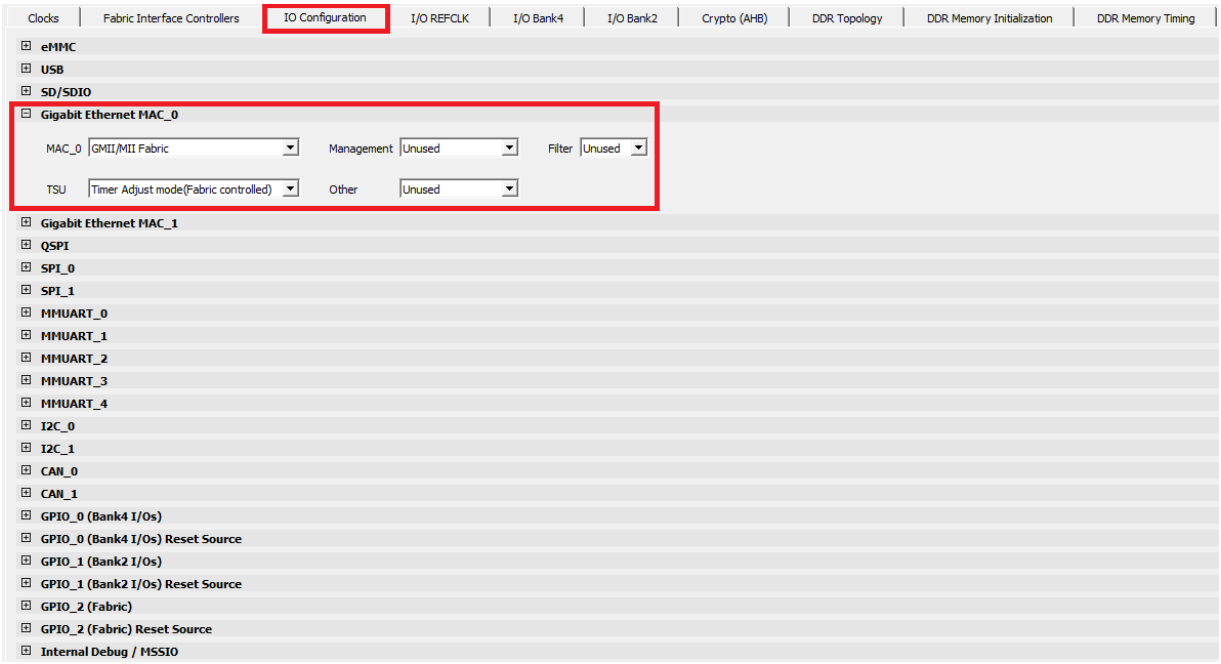
For more information about GEM Clocking, see *UG0913: PolarFire SoC FPGA Clocking Resources User Guide*.

6 Configuring GEM Using Libero

The Libero SoC design suite includes the PFSOC_MSS core in Catalog for configuring and instantiating the MSS in the PolarFire SoC FPGA.

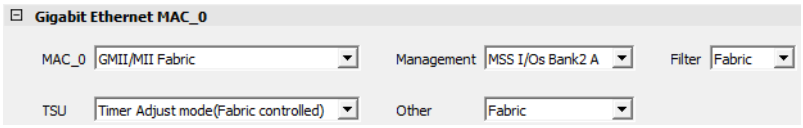
The following figure shows GEM_0 and GEM_1 in the **IO Configuration** tab of the PolarFire SoC MSS System Configurator. In this example, GEM_0 is shown.

Figure 6 • **PolarFire SoC MSS I/Os—GEM_0 Configured**



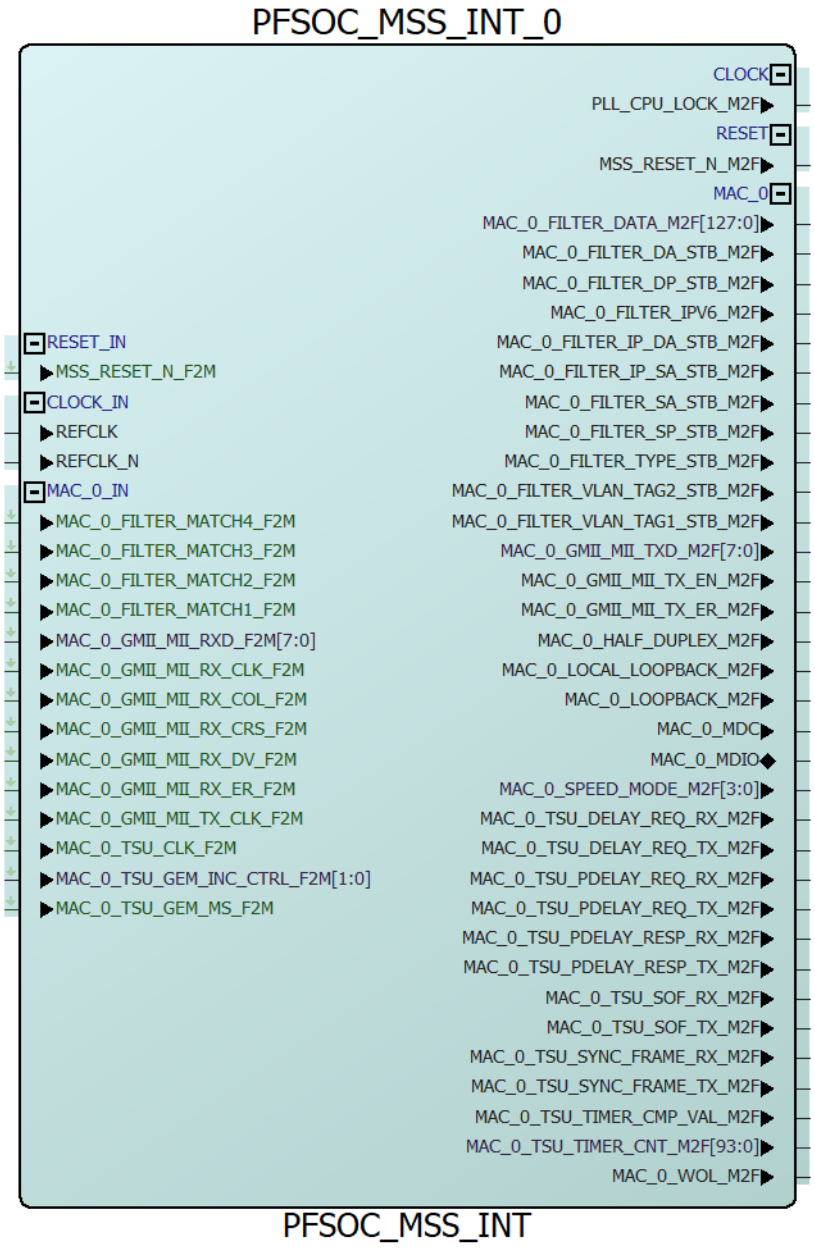
By default, GEM is unused. To configure different modes in the GEM, use the drop-down list and make the required selection.

Figure 7 • **GEM_0**



Based on the options that are enabled, the corresponding ports are exposed on the MSS block as shown in Figure 8.

Figure 8 • **MSS GEM Ports**



7 Register Address Map

GEM is configured using the following internal registers.

Table 5 • **Register Address Map**

Address Offset (Hex)	Register Type	Width
MAC Registers or Pre-emptable MAC Registers		
0x0000	Control and status	32
0x0100	Statistics	32
0x01BC	Time Stamp Unit	32
0x0200	Physical Coding Sublayer	32
0x0260	Miscellaneous	32
0x0300	Extended Filter	32
0x0400	Priority Queue and screening	32
0x0800	Time Sensitive Networking	32
0x0F00	MAC Merge Sublayer	32
eMAC Registers		
0x1000 to 0x1FFF	eMAC	32

For more information about registers, refer to *PolarFire SoC Device Register Map*.