UG0727
User Guide
PolarFire FPGA 10G Ethernet Solutions

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.

50200727. 6.0 9/23

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 6.0

The following is a summary of the changes made in revision 6.0 of this document.

- Updated sample Ethernet application block diagram to include PHY IP. See Figure 1, page 2.
- Updated information about Building Blocks for 10G Ethernet Solutions, page 4.
- Updated information about Implementing 10G Ethernet Solutions, page 6.

1.2 Revision 5.0

Updated the Libero SoC PolarFire Configurator figures to reflect the Libero SoC PolarFire v2.2 release.

1.3 Revision 4.0

The following is a summary of the changes made in revision 4.0 of this document.

- Updated the Libero SoC PolarFire Configurator figures to reflect the Libero SoC PolarFire v2.1 release.
- Updated clocking information for 10GBASE-KR and 10GBASE-R. For more information, see Clocking Requirements, page 9 and Clocking Requirements, page 12.

1.4 Revision 3.0

Updated the Libero SoC PolarFire Configurator figures to reflect the Libero SoC PolarFire v2.0 release.

1.5 Revision 2.0

The following is a summary of the changes made in revision 2.0 of this document.

- Updated Libero SoC PolarFire Configurator figures to reflect the Libero SoC PolarFire v1.1 SP1 release.
- Updated Clocking information. For more information, see 10GBASE-KR Designs, page 6 and 10GBASE-R Designs, page 10.

1.6 Revision 1.0

The first publication of this document.

2 10G Ethernet Overview

Ethernet is a family of networking interface standards used in systems and applications across multiple industries. Implementation of Ethernet solutions in FPGAs requires IP and design flows that reduce development time and utilize minimal device resources, thereby helping meet performance, power, and cost goals. Microchip PolarFire® devices support Ethernet data transfer rates ranging from 10 Mbps to 10 Gbps on a single interface.

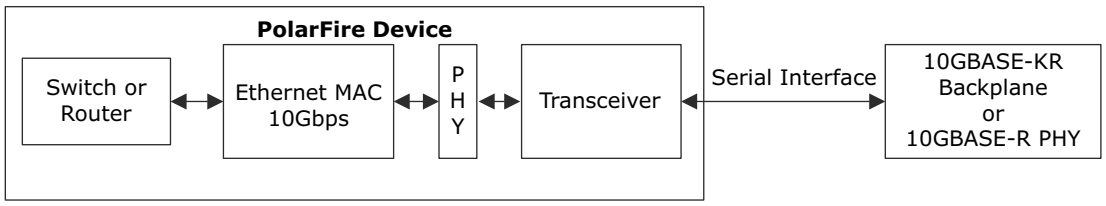
Microchip PolarFire devices provide a complete range of solutions for implementing IEEE 802.3 standard-compliant Ethernet interfaces for chip-to-chip, board-to-board, and backplane interconnects. The high-speed serial interface and soft IP blocks available in PolarFire devices enable designers to build Ethernet solutions for use in embedded systems and systems connected over copper or optical cabling.

PolarFire FPGA 10G Ethernet support is compliant with the IEEE 802.3ae standard that supports data transfer rates of up to 10.3125 Gbps. Advantages offered by PolarFire FPGAs for building 10G Ethernet solutions include the use of low-power transceivers, low-power FPGA fabric, and SyncE-compliant jitter attenuation.

In PolarFire devices, 10G Ethernet is implemented using the Core10GMAC soft IP media access control (MAC) core, which can be configured in XGMII and 10GBASE-R modes. For Ethernet backplane applications, XGMII compliant 10GBASEKR_PHY soft IP is developed. Core10GMAC supports standard Ethernet interfaces such as the 10 Gbps attachment unit interface (XAUI) and the 10 Gbps reduced attachment unit interface (RXAUI).

A typical Ethernet application, such as a switch or a router, requires an Ethernet MAC sublayer (commonly referred to as the MAC) that supports standard Ethernet interfaces, an Ethernet physical layer (PHY), and an SFP connector. The following illustration shows a sample Ethernet application.

Figure 1 • **Sample Ethernet Application**



For more information about the Ethernet MAC, including standard Ethernet interfaces, see Appendix: MAC Layers in the OSI Reference Model and Standard Ethernet Interfaces, page 16.

3 PolarFire FPGA Evaluation Kit Ethernet Support

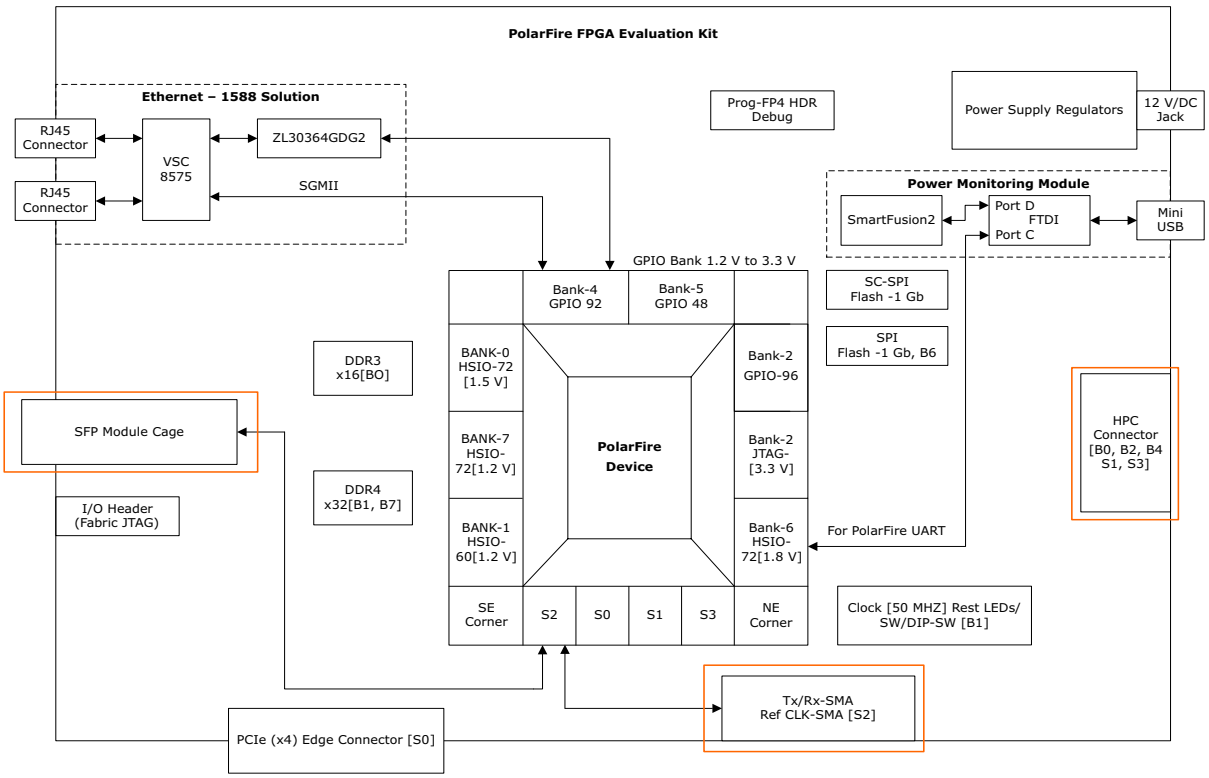
The PolarFire FPGA Evaluation Kit supports the 10GBASE-R standard (applicable to SFP applications), as well as the 10GBASE-KR standard (applicable to Ethernet backplane applications).

The PolarFire FPGA Evaluation Kit includes the following 10G Ethernet hardware components.

- SFP module supporting 1G and 10G Ethernet speeds that connects to a transceiver lane
- FPGA mezzanine card (FMC) high-pin count (HPC) connector
- SubMiniature version A (SMA) connectors

The following illustration shows the hardware of the PolarFire FPGA Evaluation Board. Highlighted in red are the hardware components used for implementing 10G Ethernet solutions.

Figure 2 • **PolarFire FPGA Evaluation Board Hardware Block Diagram**



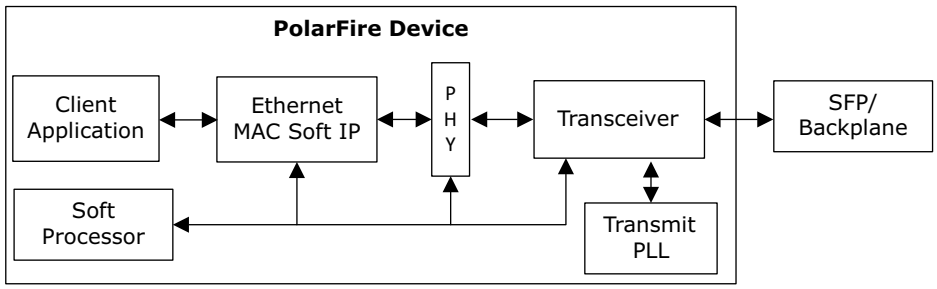
4 Building Blocks for 10G Ethernet Solutions

Microchip offers pre-designed and verified IP for all key markets and applications. A complete 10G Ethernet solution requires the following IP cores.

- Soft Processor for configuring the transceiver registers, MAC registers, PHY IP, and DRI registers.
- Ethernet MAC to process Ethernet packets
- Transceiver interface to send and receive serialized/deserialized data to and from the SFP module

The following illustration shows a sample Ethernet application developed using Microchip IP cores.

Figure 3 • **Sample Ethernet Application Using Microchip IP Cores**



For comprehensive information about all Microchip IP, see *IP Core Tool* webpage.

4.1 Soft Processor IP

CoreRISCV_AXI4: A 32-bit soft processor core such as CoreRISCV_AXI4 is used to develop processor-based Ethernet solutions. For more information about CoreRISCV_AXI4, see *CoreRISCV_AXI4 User Guide*.

CoreABC: CoreABC is a simple, configurable, low gate-count controller primarily targeted at implementing Advanced Microcontroller Bus Architecture Advanced Peripheral Bus (AMBA APB)-based designs. In an Ethernet-based application, this core is used only for configuring the Ethernet MAC. For more information about CoreABC, see the *CoreABC User Guide*.

4.2 Ethernet MAC IP (Core10GMAC)

Core10GMAC is a soft IP MAC core with built-in PCS that supports the 10GBASE-R Ethernet standards. It is compliant with the IEEE 802.3 standard, which contains PHY and MAC specifications for wired Ethernet.

Multiple Core10GMAC IP blocks can be used to develop Ethernet solutions in PolarFire devices. For more information, see *Core10GMAC User Guide*.

4.3 PHY IP

There are different IP cores available for BASE-R and BASE-KR applications:

- Core10GBaseKR_PHY: This core is designed in accordance with the IEEE® 802.3 2012 standard and supports Core10GBaseKR_PHY interface for the Backplane operations. This configurable core provides the Physical (PHY) layer when used with a transceiver interface. For more information, see *Core10GBaseKR_PHY IP* web page.
- Core10GBaseR_PHY: This core is designed to convert the XGMII compliant MAC data into transceiver gearbox signals and vice-versa. For more information, see *Core10GBaseR_PHY IP* web page. Following are the two different methods of implementing Base-R applications:
  - Configure MAC in BASE-R mode and XCVR in Gearbox mode.
  - Configure MAC in XGMII mode, and connect to BASE-R PHY IP. Configure XCVR in Gearbox mode.

4.4 Transceiver Interface IP

The PolarFire FPGA transceiver interface (PF_XCVR) provides the physical media attachment (PMA) for high-speed serial interfaces. The transceiver has a multi-lane architecture with each lane natively supporting serial data transfer rates ranging from 250 Mbps to 12.7 Gbps. For more information, see *PolarFire Family Transceiver User Guide*.

4.5 Transmit PLL

The PolarFire FPGA transmit PLL (PF_TX_PLL) provides the high-speed bit clock for the PolarFire FPGA transceiver. When used with the transceiver, the transmit PLL supports jitter attenuation for loop-timing applications where recovered clocks are used as transmit reference clocks. The jitter attenuator is compliant with SyncE G.8262 standard. For more information, see *PolarFire Family Transceiver User Guide*.

4.6 IP Licensing

The Libero® SoC PolarFire software provides free access to several Microchip IP, but some IP's require purchasing a separate license. Contact *Customer Service* for information about how to purchase licenses.

The following table lists license information for each Ethernet-based IP.

Table 1 • **License Information for Microchip 10G Ethernet-Based IP**

IP Core	License Information	Description
CoreRISCV_AXI4	Available with the Libero SoC license	
CoreABC	Available with the Libero SoC license	
PF_XCVR (transceiver interface)	Available with the Libero SoC license	
Core10GBaseR_PHY	Available with the Libero SoC license	Required for BaseR applications when MAC configured in XGMII mode
Core10GMAC	Must be purchased separately	Required for BaseR and BaseKR applications
Core10GBase_KRPHY	Must be purchased separately	Required for BaseKR applications

5 Implementing 10G Ethernet Solutions

In PolarFire devices, 10G Ethernet solutions is implemented using the Core10GMAC, BASE-R PHY and BASE-KR PHY IP cores. These IP cores are configured using:

- CoreABC soft processor in BASE-R applications
- Mi-V soft processor in BASE-KR applications

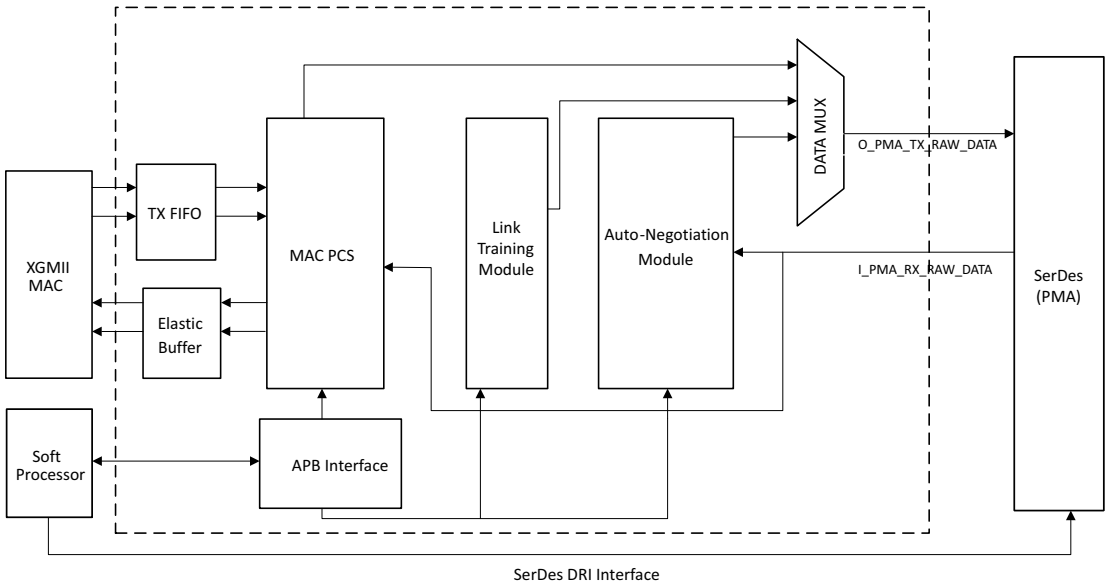
The following sections describe how the MAC interfaces with the PHY in PolarFire devices for various Ethernet interfaces. For more information about Ethernet interfaces, see Appendix: MAC Layers in the OSI Reference Model and Standard Ethernet Interfaces, page 16.

Note: BASE-KR solution is available only in -1 speed grade device.

5.1 10GBASE-KR Designs

For backplane 10GBASE-KR Ethernet designs, Core10GMAC is configured in XGMII mode and connected to the BASEKR_PHY IP. The BASE-KR IP is connected to XCVR, which is configured in PMA native mode.

Figure 4 • **Backplane 10GBASE-KR Ethernet Designs**



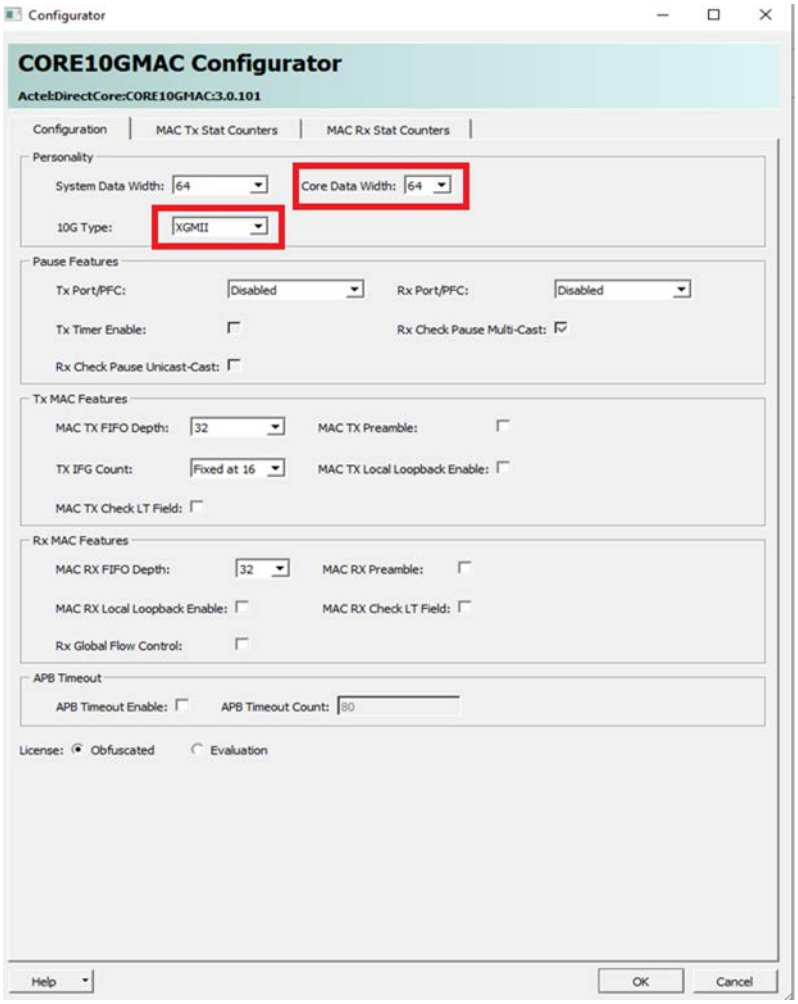
Note: Core10GBaseKR_PHY can be used with Microchip Core10GMAC or third party 10G MAC with XGMII.

5.1.1 Configuring IP and Transceiver Interface Using Libero SoC PolarFire

To implement backplane 10GBASE-KR Ethernet designs, use the following settings in Libero SoC PolarFire.

- Configure Core10GMAC in XGMII mode, as shown in the following figure.

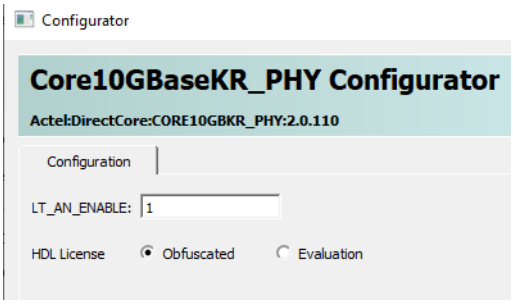
Figure 5 • **Core10GMAC Configuration**



Note: For 10GBASE-KR, the CORE10GMAC IP supports the core data width of 64 bits.

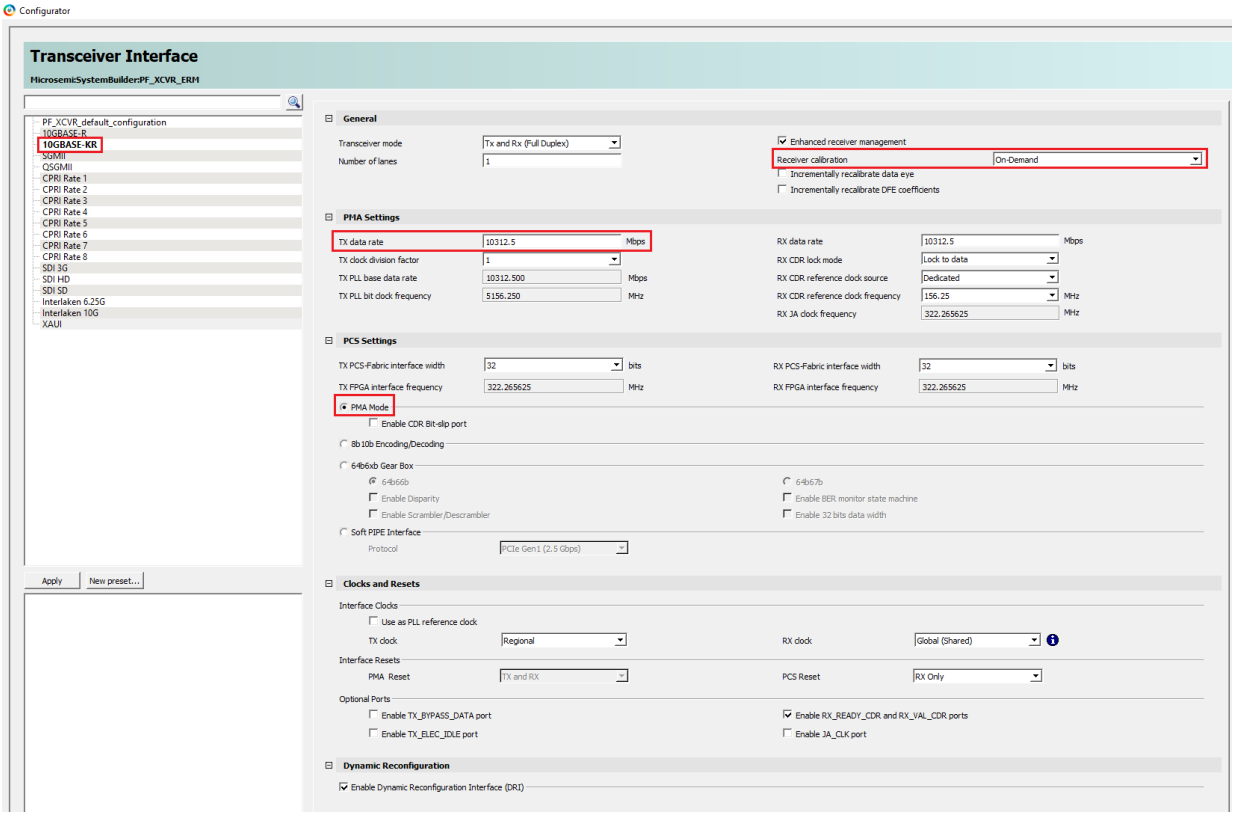
- Configure the BASE-KR IP as shown in the following figure. LT_AN_ENABLE option must be configured as 1 to perform auto negotiation and link training.

Figure 6 • **Core10GBaseKR_PHY Configurator**



- Configure the transceiver in 10GBASE-KR mode, as shown in the following figure.

Figure 7 • **Transceiver Interface Configuration for 10GBASE-KR Designs**



When configuring the transceiver, the transceiver must be set to 10312.5 Mbps to match the 10GBASE-KR data transfer rate, as shown in the preceding figure.

The **PMA Mode** option under the PCS settings needs to be selected for the 10G Base-KR applications. The 10G BaseKR application, uses the built-in PCS functionality of the BASE-KR PHY IP. Therefore, the transceiver is configured in PMA mode.

5.1.2 Clocking Requirements

The following clocks are required for 10GBASE-KR designs.

- **I_CORE_TX_CLOCK:** 156.25 MHz transmit clock for the 64-bit MAC configuration. This clock can be driven from a CCC.
- **I_CORE_RX_CLOCK:** 156.25 MHz receive clock for the 64-bit MAC configuration. This clock can be driven from a CCC.
- **I_SYS_CLOCK:** 156.25 MHz system clock decouples the user clock and the core clock domains. This clock can be driven by I_CORE_TX_CLOCK and I_CORE_RX_CLOCK
- **BASE-KR IP Clocking**
  - I_TX_CLK: 322.2 MHz transmit clock for BASE-KR IP configuration. This must be driven by the transceiver transmit clock.
  - I_RX_CLK: 322.2 MHz receive clock for BASE-KR IP configuration. This must be driven by the transceiver receive clock.
- **LANEn_CDR_REF_CLK:** 156.25-MHz reference clock to lane CDR, driven by the transceiver reference clock or a fabric clock conditioning circuit (CCC).
- **CLKS_TO_XCVR:** Clocks from the transmit PLL bus interface port with the following underlying signals common to all lanes instantiated in the transceiver interface IP core:
  - LOCK
  - BIT_CLK
  - REF_CLK_TO_LANE

The following table lists the recommended CTLE settings for different channel loss.

Table 2 • **Recommended CTLE Settings**

Insertion Loss	RX_CTLE_values
Short (6.5 dB)	5 GHz +7.3dB
Medium (17.0dB)	5 GHz +7.3dB
Long (25.0dB)	5 GHz +7.3dB

Note: You can change the CTLE settings depending upon the channel loss and the observations made while testing. These XCVR changes can be configured to generate a pdc file using Libero SoC. For more information, see *AC483: PolarFire FPGA Transceiver Signal Integrity Application Note*.

5.1.3 Firmware Support for BaseKR_PHY

The Core10GBaseKR_PHY driver is responsible for configuring the transceiver registers, DRI registers, implementation of auto-negotiation, and link training algorithms. The GitHub repository provides access to the documentation for the driver and generates sample projects that illustrate how to use the driver.

The Mi-V Soft RISC-V firmware driver is available on *Mi-V-Soft-RISC-V GitHub* page and the Mi-V Soft RISC-V SoftConsole example project is available under *miv-rv32-bare-metal-examples* in the GitHub page.

5.2 10GBASE-R Designs

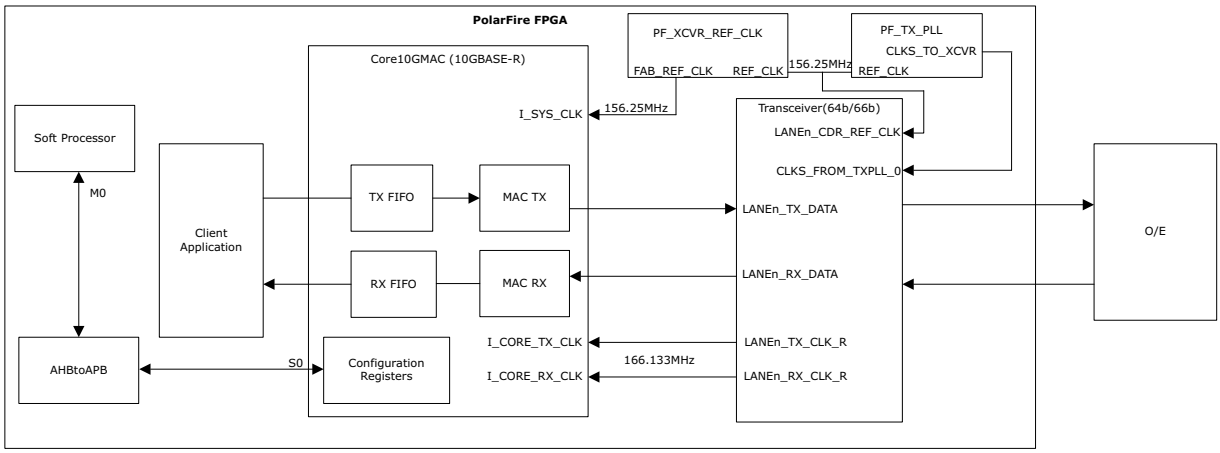
10GBASE-R designs can be configured in two different ways:

- Configuring MAC in BASE-R mode
- Configuring MAC in XGMII mode and connecting to BaseR_PHY

5.2.1 Configuring MAC in BASE-R Mode

Core10GMAC is configured in 10GBASE-R mode and connected to the transceiver, as shown in the following illustration. The designs illustrated in this section use the 64-bit data width for the MAC and the transceiver. If required, the data width can be changed to 32 bits in the Core10GMAC IP and the transceiver interface configurator. Updating the data width automatically updates the port names and the FPGA interface frequency.

Figure 8 • **10G MAC configured in BASE-R Mode**



5.2.2 Configuring IP and Transceiver Lane Using Libero SoC PolarFire

To implement 10GBASE-R designs, use the following settings in Libero SOC PolarFire.

- Configure Core10GMAC in 10GBASE-R mode, as shown in the following figure.

Figure 9 • **Core10GMAC Configuration in BASE-R Mode**

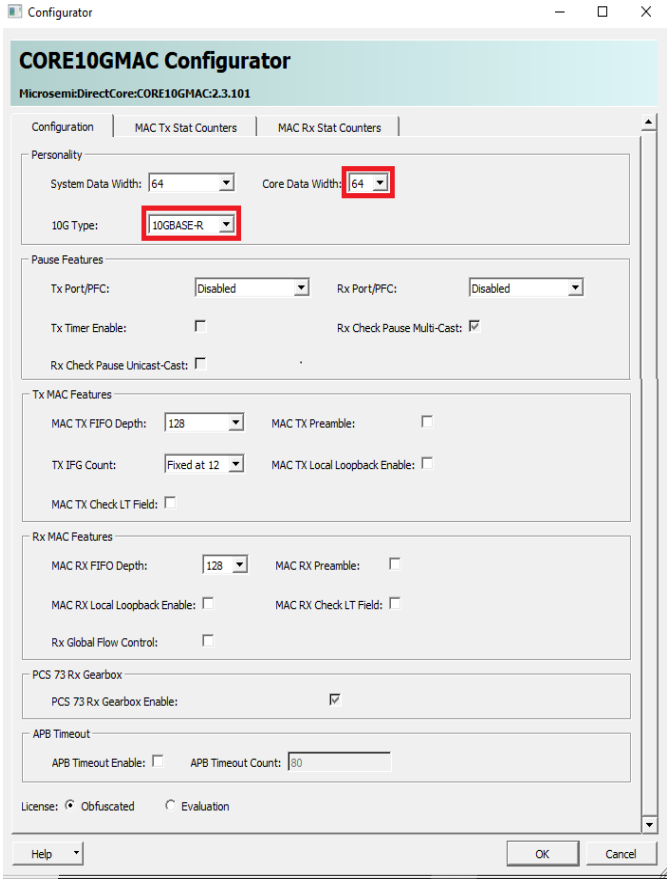
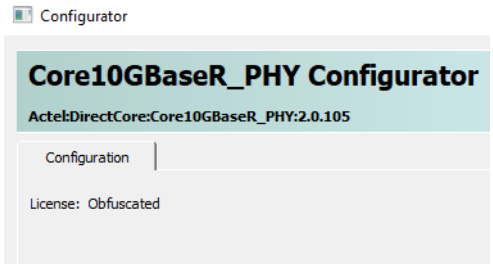
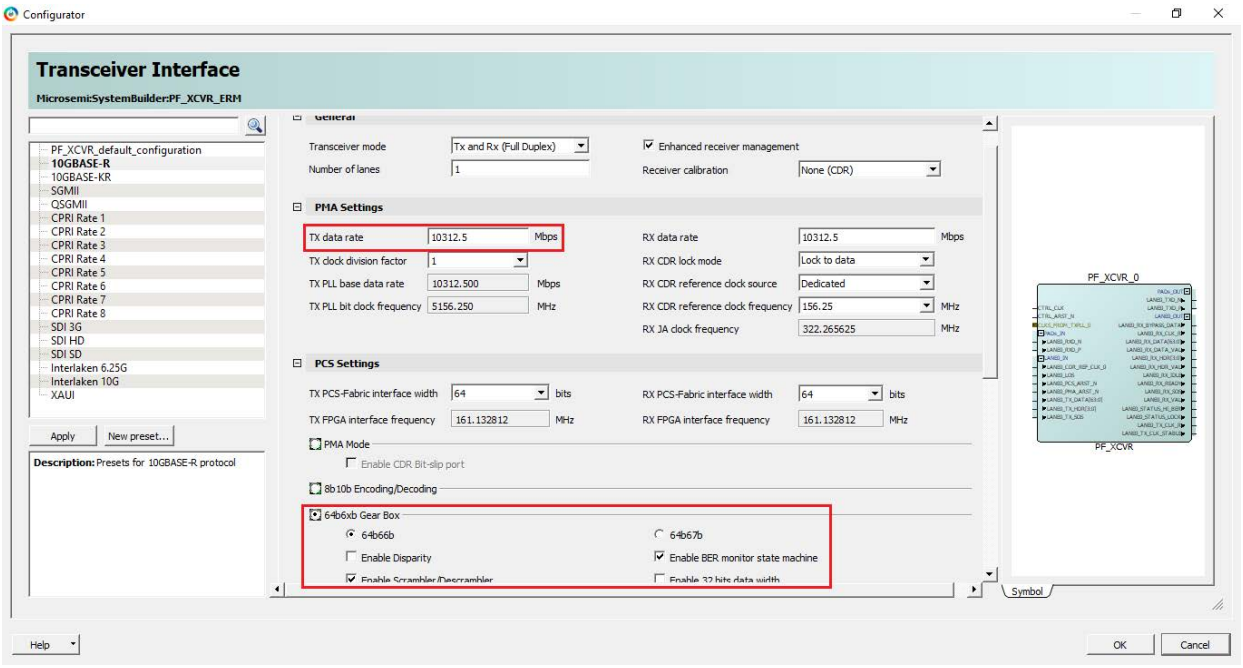


Figure 10 • **Core10GBaseR_PHY Configurator**



- Configure the transceiver interface in 10GBASE-R mode, as shown in the following figure.

Figure 11 • **Transceiver Interface Configuration for 10GBASE-R Designs**



When configuring the transceiver, the transceiver date rate must be set to 10312.5 Mbps to match the 10GBASE-R data transfer rate, as shown in the preceding figure.

The **TX clock division factor** option allows the transceiver lane high-speed clock from the TX PLL to be divided. Thus, it allows the user to share a higher rate TX PLL and locally divide the clock (for data transfer rates up to 6.4 Gbps only). The default value for this field is 1.

The 64b6xb gear box option must be selected under PCS settings to allow encoding/decoding and scrambling of Ethernet data.

5.2.3 Clocking Requirements

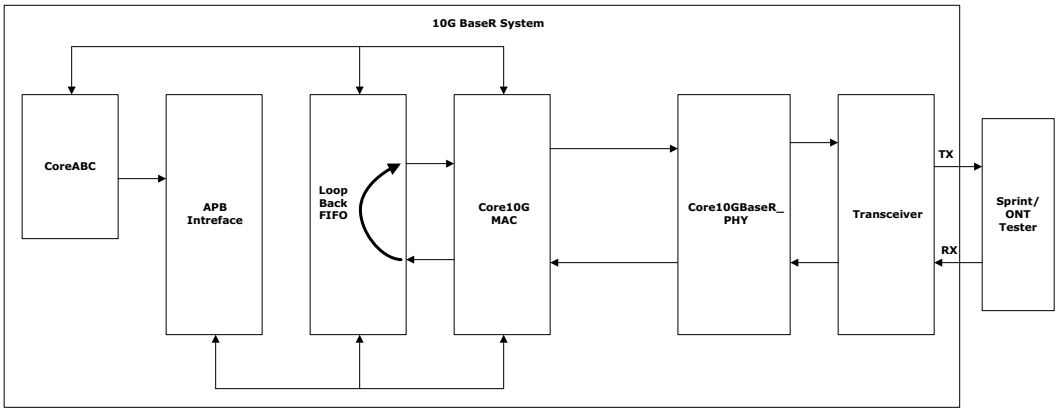
The following clocks are required for 10GBASE-R design:.

- **I_CORE_TX_CLOCK**: 322.266-MHz or 161.133-MHz transmit clock for the 32-bit and 64-bit MAC configuration respectively. This clock must be driven by the transceiver transmit clock.
- **I_CORE_RX_CLOCK**: 322.266-MHz or 161.133-MHz receive clock for the 32-bit and 64-bit MAC configuration respectively. This clock must be driven by the transceiver receive clock.
- **I_SYS_CLOCK**: This is the 156.25 MHz system clock decouples the user clock and the core clock domains. This clock can be driven by:
  - Fabric reference clock generated by PF_XCVR_REF_CLK
  - User-generated fabric clock by an independent clock conditioning circuit (CCC)
- **LANEn_CDR_REF_CLK**: 156.25-MHz reference clock to lane CDR, driven by the transceiver reference clock or a fabric clock conditioning circuit (CCC).
- **CLKS_TO_XCVR**: Clocks from the transmit PLL bus interface port with the following underlying signals common to all lanes instantiated in the transceiver interface IP core:
  - LOCK
  - BIT_CLK
  - REF_CLK_TO_LANE

5.2.4 Configuring MAC in XGMII Mode

Core10GMAC is configured in XGMII mode and connected to BaseR_PHY, as shown in the following illustration.

Figure 12 • **10GBASE-R MAC in XGMII Mode**



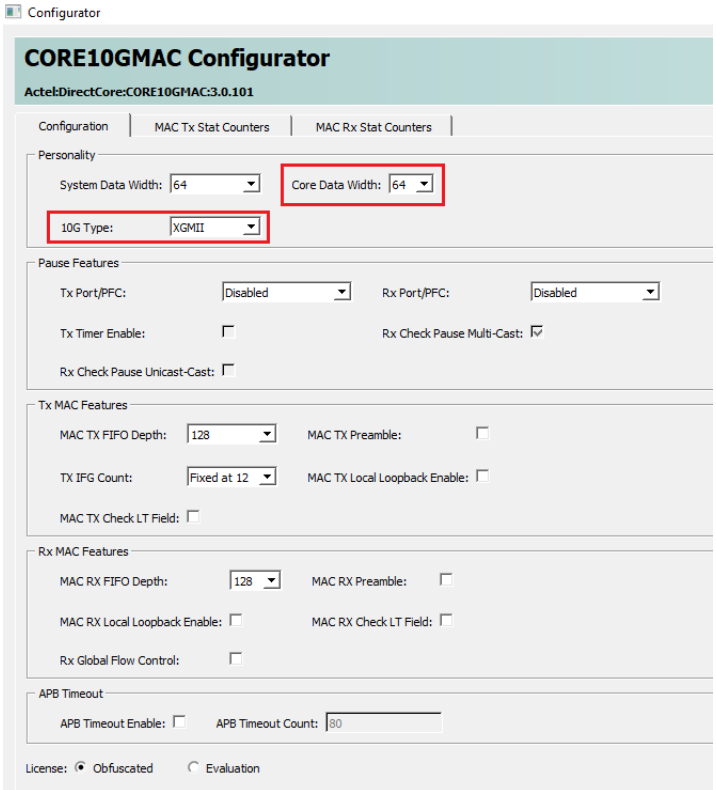
Note: Any third party MAC in XGMII mode can be used with BASE-R IP

5.2.5 Configuring MAC, BaseR_PHY, and Transceiver Lane Using Libero SoC PolarFire

To implement 10GBASE-R designs, use the following settings in Libero SoC.

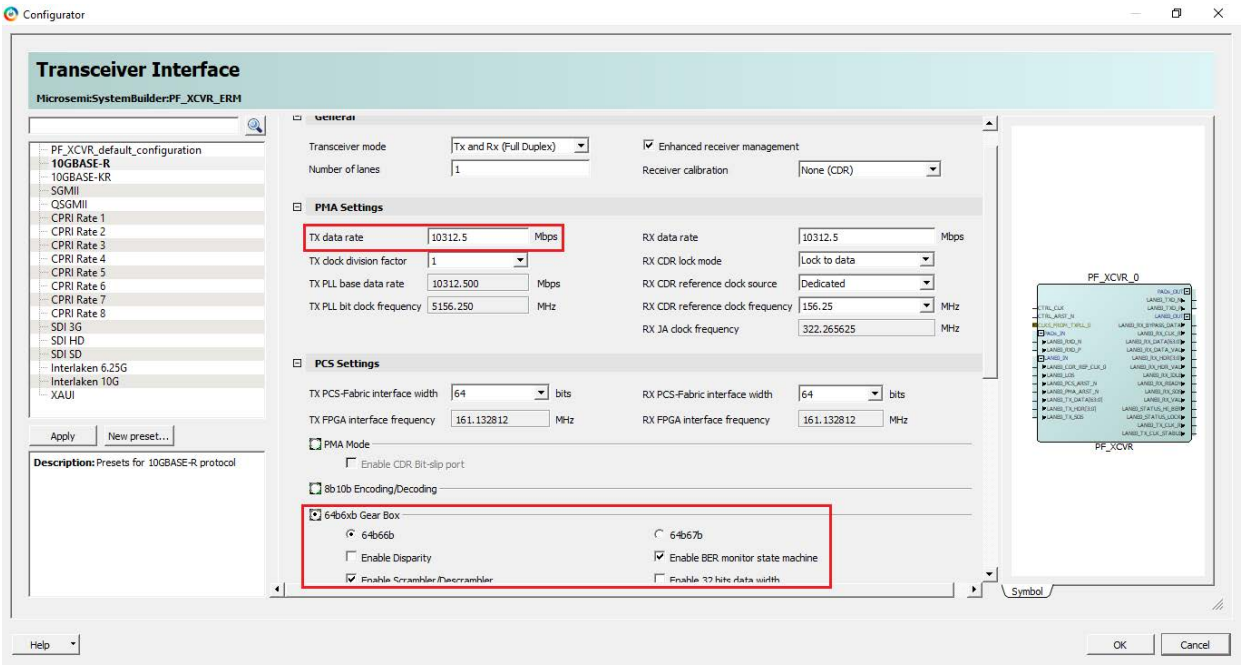
- Configure Core10GMAC in XGMII mode, as shown in the following figure.

Figure 13 • **Core10GMAC Configuration in XGMII Mode**



- Configure the transceiver interface in 10GBASE-R mode, as shown in the following figure.

Figure 14 • **Transceiver Interface Configuration for 10GBASE-R Designs**



When configuring the transceiver, the transceiver date rate must be set to 10312.5 Mbps to match the 10GBASE-R data transfer rate, as shown in the preceding figure.

The 64b6xb gear box option must be selected under PCS settings to allow encoding/decoding and scrambling of Ethernet data.

5.2.6 Clocking Requirements

The following clocks are required for MAC IP designs.

- **I_CORE_TX_CLOCK:** 156.25 MHz transmit clock for the 64-bit MAC configuration. This clock can be driven from a CCC.
- **I_CORE_RX_CLOCK:** 156.25 MHz receive clock for the 64-bit MAC configuration. This clock can be driven from a CCC.
- **I_SYS_CLOCK:** 156.25 MHz system clock decouples the user clock and the core clock domains. This clock can be driven by I_CORE_TX_CLOCK and I_CORE_RX_CLOCK

The following clocks are required for BASER_PHY IP designs.

- I_TX_CLK: 322.2 MHz transmit clock. This must be driven by the transceiver transmit clock.
- I_RX_CLK: 322.2 MHz receive clock. This must be driven by the transceiver receive clock.
- I_XGMII_TX_CLK: 156.25 MHz Transmit XGMII clock.
- I_XGMII_RX_CLK: 156.25 MHz Receive XGMII clock.
- **LANEn_CDR_REF_CLK:** 156.25-MHz reference clock to lane CDR, driven by the transceiver reference clock or a fabric clock conditioning circuit (CCC).
- **CLKS_TO_XCVR:** Clocks from the transmit PLL bus interface port with the following underlying signals common to all lanes instantiated in the transceiver interface IP core:
  - LOCK
  - BIT_CLK
  - REF_CLK_TO_LANE

5.3 Firmware Support for BaseR_PHY

For information about MAC registers configuration, see *AN5102: PolarFire FPGA 10BASE-R Ethernet SyncE Loopback Application Note (Earlier DG0757)*.

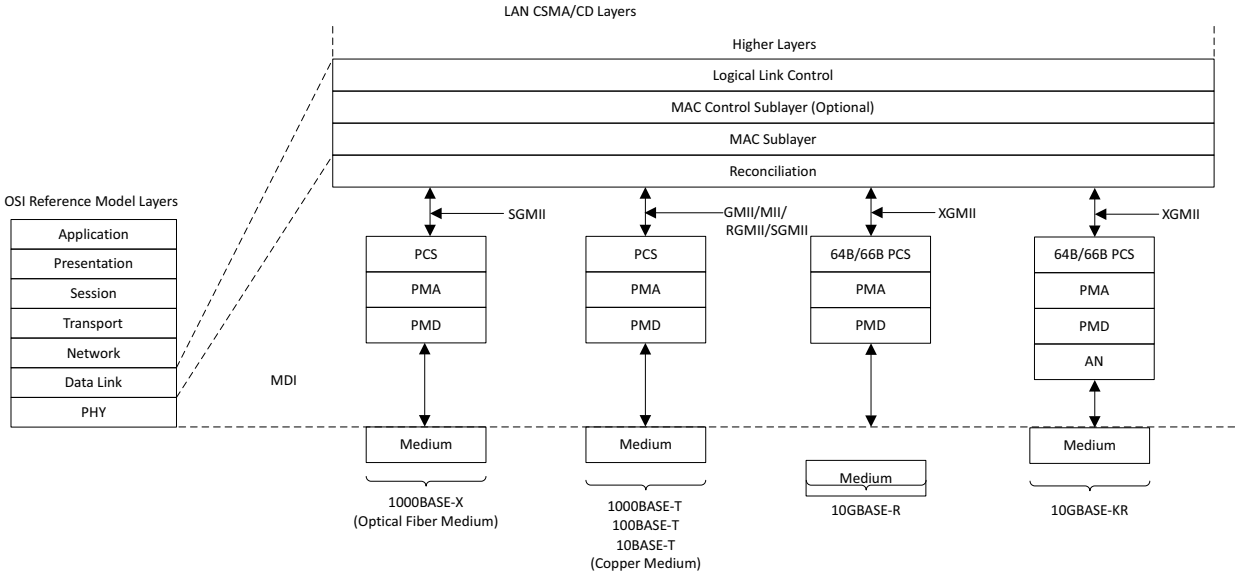
6 Appendix: MAC Layers in the OSI Reference Model and Standard Ethernet Interfaces

This section discusses how Ethernet MAC layers fit into the open systems interconnection (OSI) reference model and provides information about the standard interfaces used for Ethernet connections.

6.1 MAC Layers

The OSI reference model is a standard framework for data communication between networked systems. The following illustration shows the relationship between the OSI reference model and the Ethernet MAC as defined in the IEEE 802.3-2012 standard. It also illustrates where the supported physical interfaces (PCS, PMA, and PMD) fit into the architecture. The MAC and MAC control sublayers shown are handled by the Ethernet MAC.

Figure 15 • **IEEE Standard 802.3-2012 Ethernet Model**



The data link and physical layers in the OSI model are explained below.

6.1.1 LLC

The logical layer control (LLC) sublayer acts as an interface between the MAC and the network layer. It provides frame synchronization, flow control, and error checking mechanisms. It also offers multiplexing mechanisms to allow several network protocols to exist simultaneously in a multi-point network and share the same network medium for transmitting and receiving Ethernet packets.

6.1.2 MAC Sublayer

The MAC sublayer, referred to as the MAC, is defined in IEEE 802.3-2012, clauses 2, 3, and 4. The MAC is the second sublayer of the data link layer in the seven-layer OSI model. It provides addressing and channel access control mechanisms that allow terminals and network nodes in a multiple access network using a shared medium, such as an Ethernet network, to communicate with each other. The MAC is responsible for the transmission of data packets to and from the network-interface card. It is independent of the physical layer and can connect to any type of physical layer device.

6.1.3 MAC Control Sublayer

The MAC control sublayer, defined in IEEE 802.3-2012, clause 31 provides real-time flow control manipulation for the MAC. MAC and MAC control sublayer functions are performed by the Ethernet MAC in all modes of operation.

6.1.4 Reconciliation Sublayer

The reconciliation sublayer maps the signals between the physical medium interface and the MAC layer.

6.1.5 Physical Sublayers (PCS, PMA, and PMD)

The Ethernet physical layer consists of the following three sublayers.

- Physical coding sublayer (PCS)—performs auto-negotiation and coding operations such as 8b/10b encoding
- Physical medium attachment sublayer (PMA)—performs data serialization and clock recovery necessary to move serial data in and out of the device
- Physical medium-dependent sublayer (PMD)—hosts the transceiver that receives and transmits data through the physical medium

6.1.6 10GBASE-R

10GBASE-R is a serial-encoded PCS that supports 32- and 64-bit data transmission over fiber-optic media. It allows Ethernet framing at 10.3125 Gbps.

Figure 16 • **10GBASE-R System Diagram**

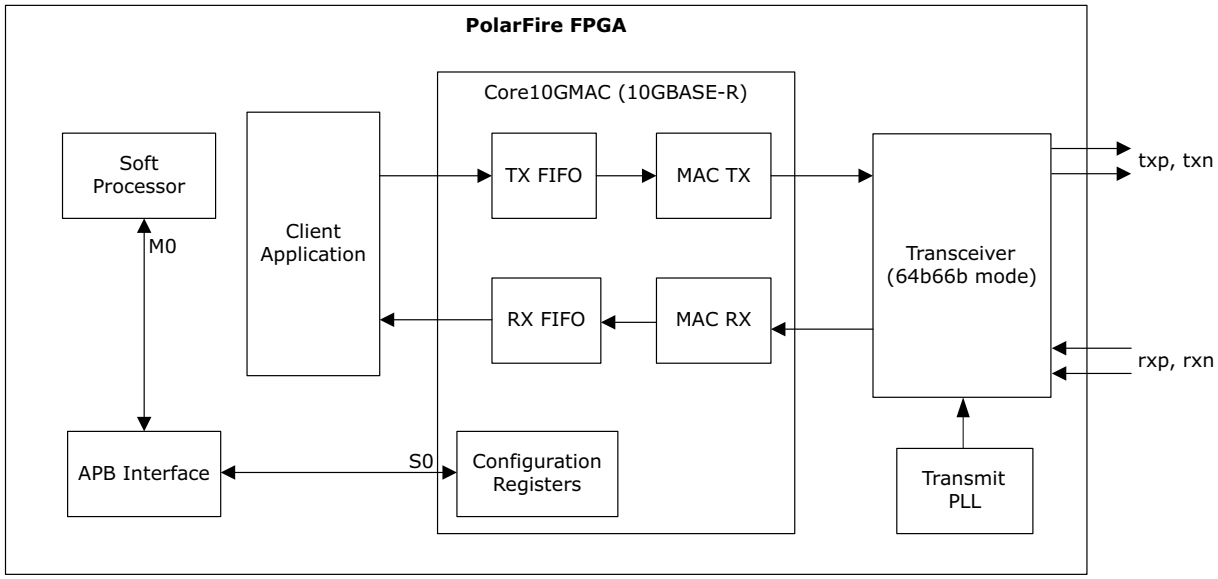
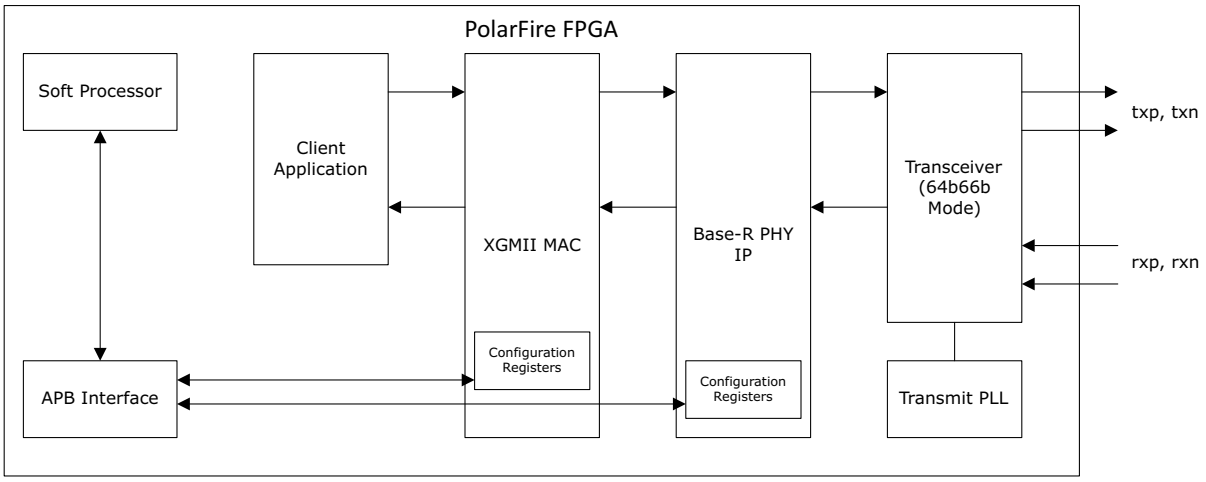


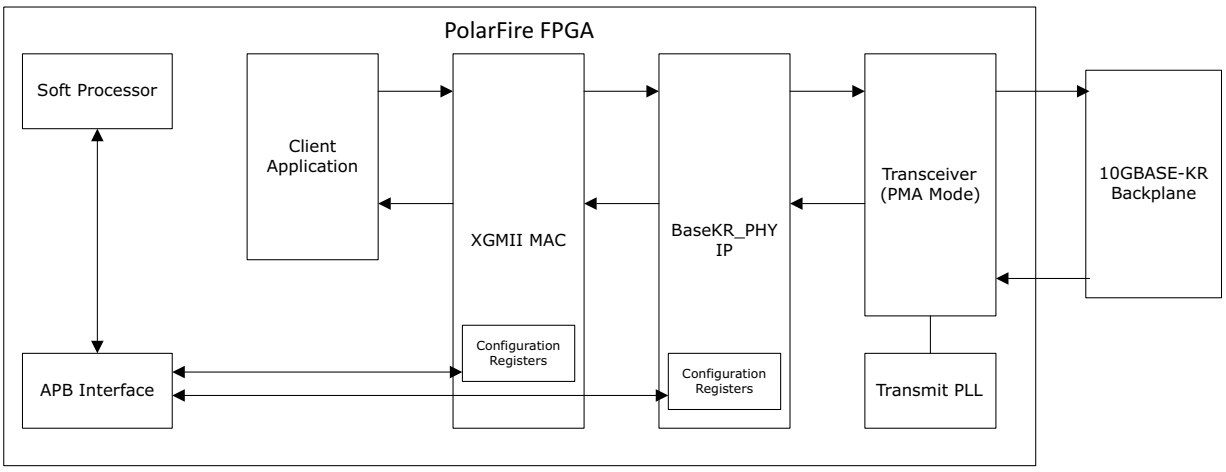
Figure 17 • **10GBASE-R System Diagram using MAC XGMII Mode**



6.1.7 10GBASE–KR

10GBASE-KR is a serial-encoded PCS that supports 32-bit data transmission over copper media. It allows Ethernet framing at 10.3125 Gbps (similar to 10GBASE-R). The link training block and the optional auto-negotiation (AN) feature available in the 10GBASE-KR standard distinguishes it from the 10GBASE-R standard.

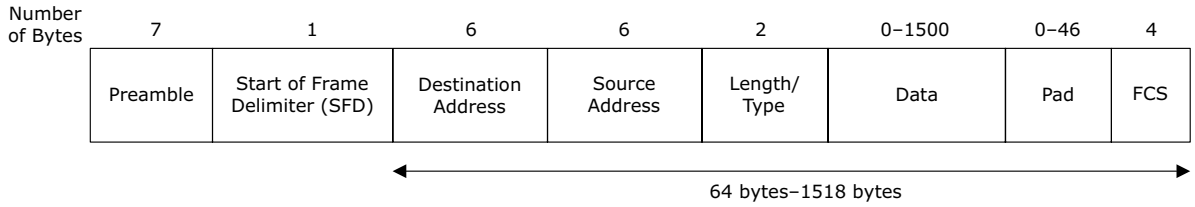
Figure 18 • **10GBASE-KR System Diagram**



7 Appendix: Ethernet Frame Format

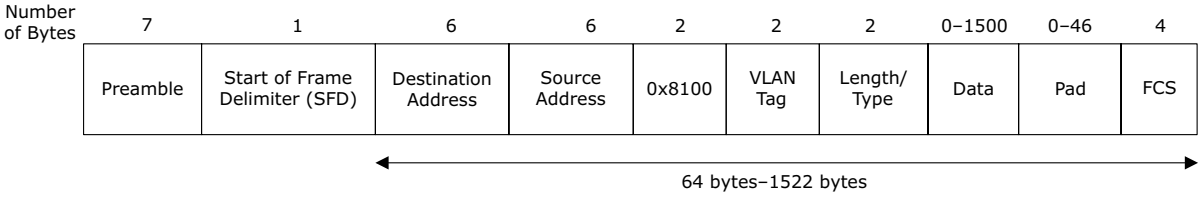
Ethernet data is encapsulated in frames consisting of a preamble, a start-of-frame delimiter (SFD), and the actual frame starting from the destination address and ending with the frame check sequence (FCS) field. The bytes within each field in an Ethernet frame are transmitted from left to right, going from the least significant bit to the most significant bit. A typical Ethernet frame's data length is between 0 bytes to 1500 bytes. Frames with data lengths greater than 1500 bytes are called jumbo Ethernet frames. The actual Ethernet frame begins after the SFD. The following illustration shows the format of a standard Ethernet frame.

Figure 19 • **Standard Ethernet Frame Format**



The Ethernet MAC also accepts virtual local area network (VLAN) frames. VLANs are specified in IEEE 802.1Q. A virtual, bridged LAN logically groups the network devices that share the same physical network. This way, the network traffic in a VLAN group is only visible to those devices that are members of that network group. For VLAN-type frames, the Ethernet MAC accepts four bytes more than a standard Ethernet frame. The following illustration shows the format of an Ethernet VLAN frame.

Figure 20 • **Ethernet VLAN Frame Format**



The following sections describe the individual fields in an Ethernet frame.

7.1 Preamble

The preamble field synchronizes receiver clocks within a network and contains seven bytes, with the pattern 0x55, transmitted from left to right. During transmission on the physical interface, this field is automatically inserted by the Ethernet MAC. On reception, it is stripped from the incoming frame before the data is passed to the MAC client. The MAC can receive Ethernet frames even if the preamble does not exist, as long as a valid SFD is available.

7.2 SFD

The SFD field marks the start of the Ethernet frame and must follow the pattern 0xD5. For transmission, this field is automatically inserted by the Ethernet MAC. On reception, it is stripped from the incoming frame before the data is passed to the MAC client.

7.3 MAC Address Fields

The MAC address is a unique identifier assigned to PHY interfaces to allow devices to communicate over the network. A MAC address can either be a source address or a destination address, depending on whether it is transmitting the frame or receiving it.

- Destination address—the MAC address of the frame's intended recipient on the network. It is the first field in an Ethernet frame that is transmitted and received between stations.
- Source address—the MAC address of the frame's initiator on the network. It is the second field in an Ethernet frame.

The least significant bit of a MAC address determines if a MAC address is an individual (unicast) address, a group (multicast) address, or a broadcast address.

- An individual address, also known as a unicast address, is specific to a station (device) on the network. For this address type, the destination address ends with 0.
- A group address, also known as a multicast address, is used to group logically-related stations. For this address type, the destination address ends with 1.
- A broadcast address is a multicast address used to group all stations on the LAN. For this address type, the destination address field has all 1s.

The Ethernet MAC supports transmission and reception of unicast, multicast, and broadcast packets. During transmission, the bit representing the individual or group (multicast) address is the first bit to appear in the address field of an Ethernet frame.

7.4 VLAN Tag (for VLAN Frames Only)

A VLAN tag field consists of the VLAN ID inserted into a packet header to identify the VLAN the packet belongs to. Based on the VLAN ID, switches determine the port or interface to which the broadcast packet needs to be sent.

7.5 Length/Type

The value of this field determines if it is interpreted as a length field or a type field as defined by IEEE 802.3-2012. The MAC interprets a value of 1500 bytes or less is interpreted by the MAC as a length field and a value of 1536 bytes or more as a type field. A length field represents the number of bytes in the following data field. This value excludes any bytes inserted in the pad field following the data field. A length/type field value of 0x8100 indicates a VLAN frame, and a value of 0x8808 indicates a PAUSE MAC control frame.

During transmission, the Ethernet MAC does not process the length/type field. On reception, if this field is a length field, the Ethernet MAC's receive engine interprets this value and removes any padding that may be displayed in the pad field.

If the field is a length field and length/type checking is enabled in the Ethernet MAC, the MAC compares the length against the actual data field length and flags an error if a mismatch occurs. If the field is a type field, the Ethernet MAC ignores the value and simply passes it along with the packet data with no further processing. The length/type field is retained in the receive packet data.

7.6 Data

For a normal frame, the data field can vary from 0 to 1,500 bytes in length for a normal frame. The Ethernet MAC can handle jumbo frames of any length. This field is provided in the packet data for transmissions and is retained in the receive packet data.

7.7 Pad

The pad field ensures that the Ethernet frame is at least 64 bytes in length. This is the minimum length required for successful CSMA/CD operation. The field can vary from 0 to 46 bytes in length.

7.8 FCS

A frame check sequence (FCS) is an error-detecting code that can optionally be added to an Ethernet frame. The value of the FCS field is calculated using the data in the destination address, source address, length/type, data, and pad fields through a 32-bit cyclic redundancy check (CRC) algorithm. For transmission, this field can be either inserted automatically by the Ethernet MAC or supplied by the client. On reception, the incoming FCS value is verified for every frame. If an incorrect FCS value is received, the Ethernet MAC indicates to the client that it has received a bad frame. The FCS field can either be passed on to the client or dropped by the Ethernet MAC based on whether the FCS feature is enabled in the MAC.

8 Appendix: Glossary

This section defines common terms associated with Ethernet architecture used in this document.

Note: This section does not define MAC sublayers, standard Ethernet interfaces, Ethernet frame fields, and 10GBASE-R/10GBASE-KR standards, which are described in previous sections.

Table 3 • **Common Ethernet Terms**

Term	Definition
Physical layer (PHY)	The PHY is the physical layer of the MAC, which, when instantiated, connects a link layer device (often called a MAC) to a physical medium such as an optical fiber or a copper cable.
SFP	A small form factor pluggable (SFP) connector, commonly known as an SFP, is a compact, hot-pluggable transceiver (transmitter and receiver in a single package) used for carrying data over optical or copper wires.
Transceiver	A transceiver is a pair of functional blocks that converts serial data to parallel data and vice versa. The primary use of a transceiver is to provide data transmission over a single/differential line, thereby minimizing the number of I/O pins and interconnects required for the design.
Txp, Txn/Rxp, Rxn	Txp, txn and rxp, rxn are pairs of differential signals used to connect the transceiver to the SFP connectors.
Auto-negotiation	Auto-negotiation is an Ethernet procedure that allows two link partners to exchange capability parameters (speed, duplex mode, and flow control) and then choose the highest data transmission speed supported by both the devices. In the OSI model, the auto-negotiation capability resides in the PHY.
Link training	Link training is a process that enables communication between a transmitter and a receiver by controlling the transmit and receive signal integrity settings. Electrical characteristics and bit rate of the link are established during link training.
Differential Manchester encoding Scheme	Differential Manchester encoding Scheme is a mechanism used in auto-negotiation. It is a line code where data and clock signals are combined to form a two-level self-synchronizing data stream.
O/E	An optical-to-electrical converter (commonly known as an O/E) is a device that converts optical signals to electrical signals. The SFP module included in the PolarFire FPGA Evaluation Kit functions as an O/E in 10GBASE-R designs.