PolarFire FPGA 10BASE-R Ethernet SyncE Loopback

AN5102

Introduction (Ask a Question)

The PolarFire® FPGA 10G Ethernet solution is compliant to IEEE® 802.3ae standard, which supports data transfer rates up to 10.3125 Gbps. The advantages offered using PolarFire FPGAs for building 10G Ethernet solutions include: the use of low-power transceivers, low-power FPGA fabric, and an in-built SyncE-compliant jitter attenuation.

The 10G Ethernet solution is implemented using the CORE10GMAC soft IP Media Access Control (MAC) core, which can be configured either in 10GBASE-KR mode or 10GBASE-R mode.

This demo design includes a 10GBASE-R Ethernet 32-bit loopback design, which can be used as reference design for building a 10GBASE-R Ethernet SyncE loopback application. The SyncE is complaint with Jitter Attenuation Phase Locked Loop (JA PLL) enabled. This design runs on the PolaFire Evaluation board using Optical Network Tester (ONT) and can be used for simulation.

Design Requirements (Ask a Question)

The following table lists the hardware and software requirements for running the demo.

Table 1. Design Requirements

Requirements	Version
Operating system	Windows® 7 or Windows 10
Hardware	
PolarFire® Evaluation Kit (MPF300-EVAL-KIT)	Rev D or later
Optical Network Tester (ONT) for 10G Ethernet traffic generation	—
Optical fiber cable	—
Small form-factor pluggable (SFP+) module	—
Software	
Libero® SoC	**Note:** See the `readme.txt` file provided in the design files for the software versions used with this reference design.
Synplify Pro®	
ONT TestCenter	
FlashPro Express®	
ModelSim®	

Important: Libero SmartDesign and configuration figures shown in this guide are for illustration purposes only. Open the Libero design to see the latest updates.

Prerequisites (Ask a Question)

Before you begin,

1. Download the design files from this link: www.microchip.com/en-us/application-notes/AN5102

2. Download and install Libero SoC (as indicated in the website for this design) on the host PC from the following location: Libero SoC Installation link

Table of Contents

Introduction....1
Design Requirements....1
Prerequisites....1
1. Demo Design....4
1.1. Design Implementation....5
1.2. Design Blocks and IP Configuration....5
1.3. Clocking Structure....9
1.4. Reset Structure....10
2. Simulating the 10GBASE-R Ethernet SyncE Loopback Design....12
2.1. Prerequisites....12
2.2. Design Description....12
2.3. Design Implementation....13
2.4. Simulation Flow....14
3. Libero Design Flow....17
3.1. Synthesize....18
3.2. Place and Route....18
3.3. Verify Timing....20
3.4. Generate FPGA Array Data....20
3.5. Configure Design Initialization Data and Memories....20
3.6. Generate Bitstream....21
3.7. Run Program Action....21
4. Running the Demo....23
5. Appendix 1: Programming the Device Using FlashPro Express....26
6. Appendix 2: Running the TCL Script....27
7. Revision History....28
Microchip FPGA Support....29
Microchip Information....29
The Microchip Website....29
Product Change Notification Service....29
Customer Support....29
Microchip Devices Code Protection Feature....29
Legal Notice....30
Trademarks....30
Quality Management System....31
Worldwide Sales and Service....32

1. Demo Design (Ask a Question)

The 10GBASE-R Ethernet SyncE loopback hardware design loops back the Ethernet traffic generated by Optical Network Tester (ONT) through the CORE10GMAC IP. A FIFO logic is implemented in the RTL to loop the RX signals of the Core10G MAC back to the TX signals of the MAC.

This looped back data is sent through the TX interface of the transceiver that is received by ONT. Using the ONT software, the received data is analyzed for throughput rate and errors in the incoming packets and ppm difference.

The 10GBASE-R Ethernet SyncE loopback design includes the following components:

- CORE10GMAC: Serves as a 10 Gbps Ethernet MAC that transmits and receives the Ethernet packets.
- Transceiver: Acts as a 10GBASE-R physical interface for data transfers; configured for 64b/66b encoding/decoding with scrambler/descrambler enabled with a PCS interface width of 32 bits to the CORE10GMAC and JAPLL option enabled.
- CoreABC: Configures the CORE10GMAC registers.
- FIFO Interface Logic: Loops back the CORE10GMAC Rx data to Tx data.
- PF_TX_PLL: Generates the bit clock required for the transceiver and the clock option is set to Jitter cleaning mode to enable the Jitter attenuator PLL.
- PF_XCVR_REF_CLK: Generates the fabric clock and the reference clock for the transceiver and the TX_PLL.

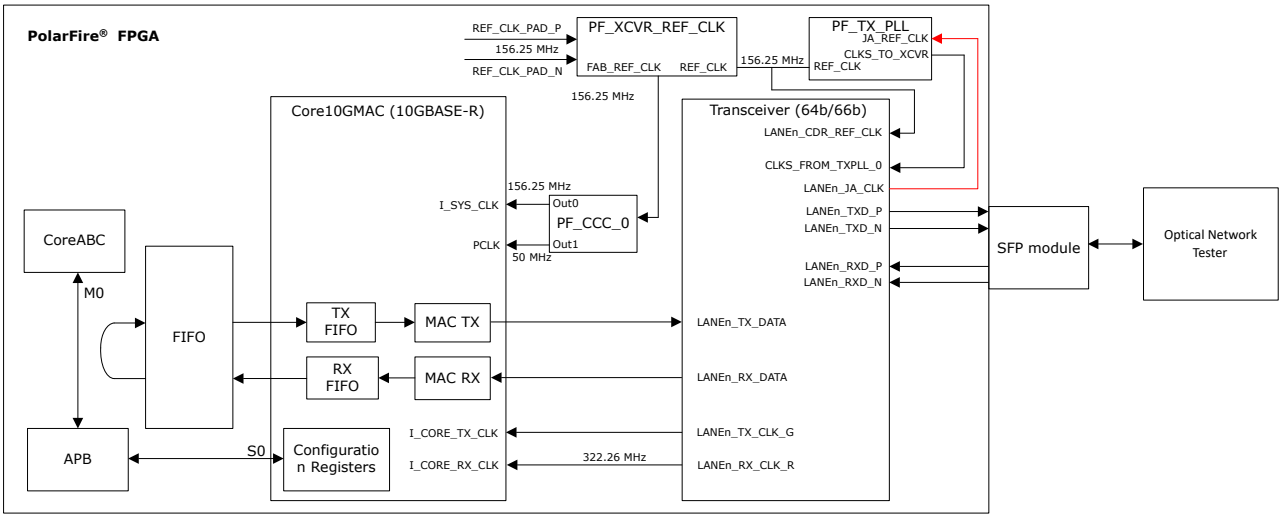
The following table lists the clock frequencies used in the design.

Table 1-1. Hardware Design Clock Frequencies

Clock	Frequency (MHz)
CDR reference clock	156.25
Transceiver bit clock	5156.25
I_SYS_CLOCK	156.25
I_CORE_TX_CLK	322.26
I_CORE_RX_CLK	322.26
JA_REF_CLK	322.26
PCLK	50

The following figure shows the top-level block diagram of the PolarFire 10G Base-R Ethernet SyncE loopback hardware implementation.

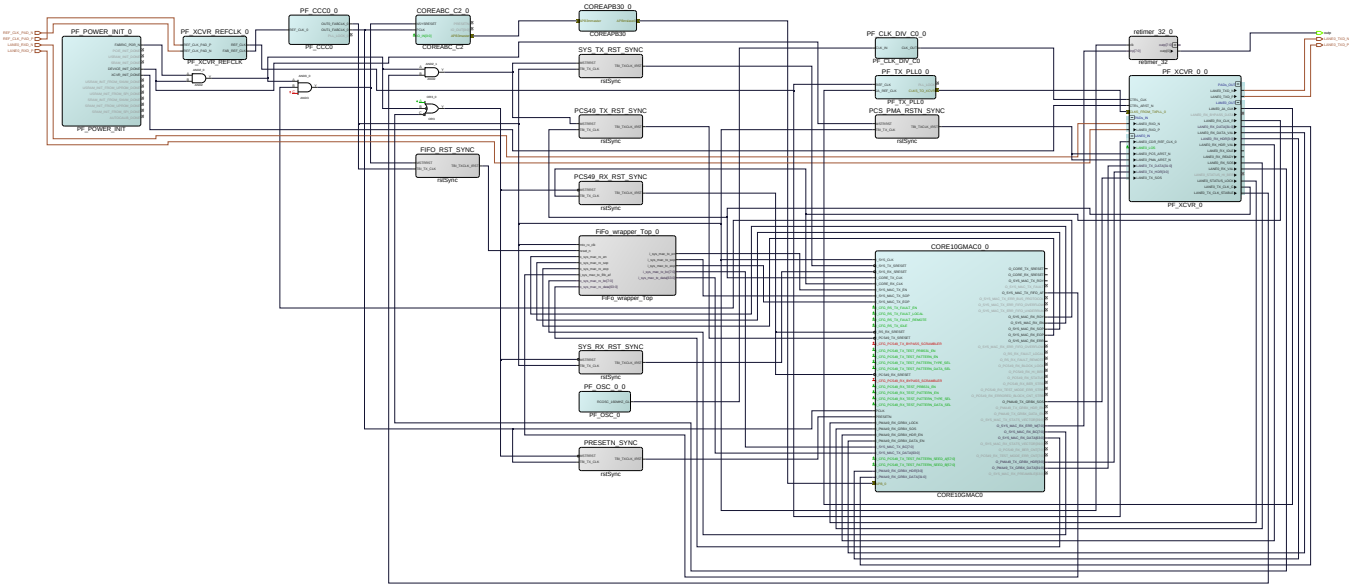
Figure 1-1. Hardware Implementation Block Diagram



1.1 Design Implementation (Ask a Question)

The following figure shows the Libero implementation of the 10G Base-R Ethernet SyncE loopback hardware design.

Figure 1-2. 10GBASE-R Loopback Hardware Design Libero Implementation—SyncE



1.2 Design Blocks and IP Configuration (Ask a Question)

The following IPs are required to be configured before simulating and implementing the demo design.

- 1.2.1. Core10GMAC
- 1.2.2. Transceiver Interface
- 1.2.3. Transmit PLL

- 1.2.4. Transceiver Reference Clock
- 1.2.5. CoreABC
- 1.2.6. CoreAPB3
- 1.2.7. PF_POWER_INIT
- 1.2.8. PF_CCC_0
- 1.2.9. FiFo_wrapper_Top

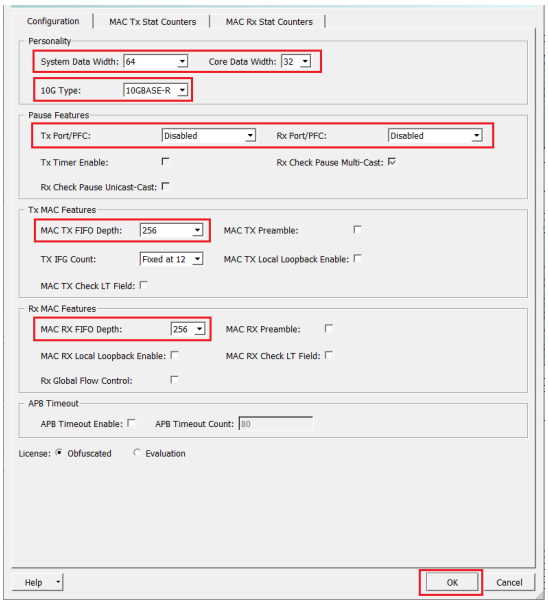
1.2.1 Core10GMAC (Ask a Question)

Core10GMAC is configured for 10GBASE-R mode with a core data width of 32 bits. Core data width is the width of the data path connected to the transceiver interface. The system data width, that is, the width of the interface to the user logic, is configured as 64 bits. (In this demo, the FiFo_wrapper_top module provides this interface.)

The Tx and Rx Pause features are disabled, and both the MAC TX FIFO depth and MAC RX FIFO depth are set to 256.

The following figure shows the settings selected in the CORE10GMAC Configurator.

Figure 1-3. CORE10GMAC Configuration



The Core10GMAC IP is configured using the CoreABC soft processor. The Core10GMAC configuration for the demo design is as follows:

Table 1-2. Core10GMAC Configuration

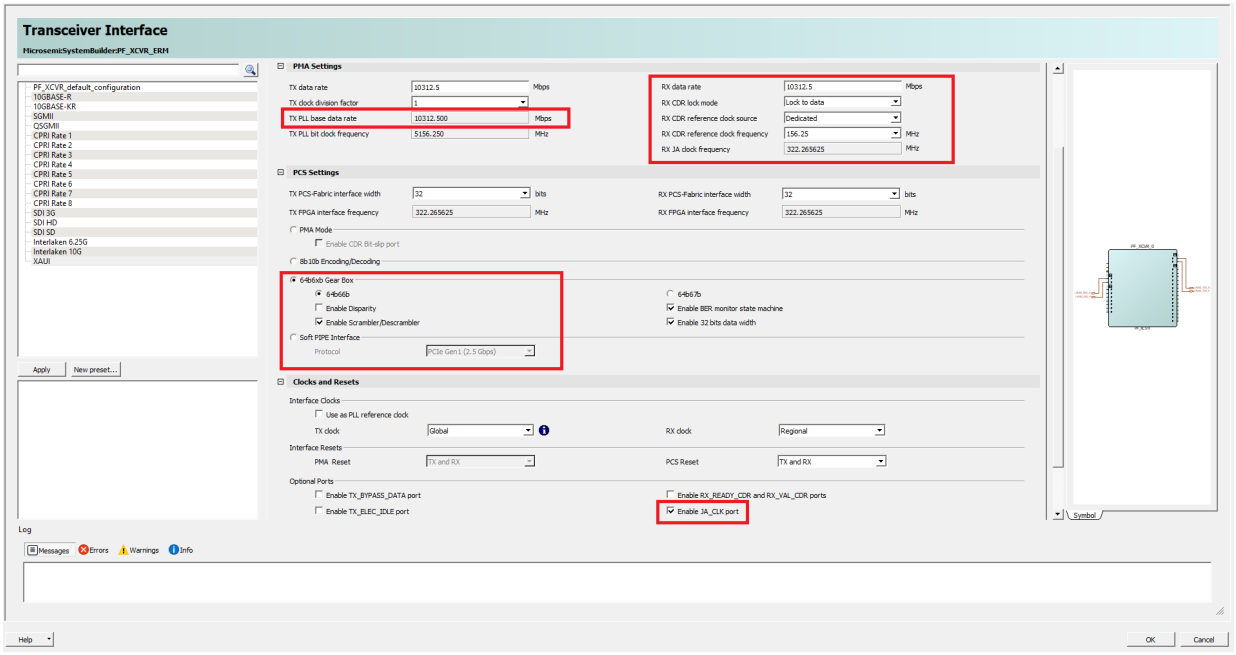
Register	Address	Offset	Bit	Binary Value
MAC Tx Config Register	(0xA)	0x3	cfg_sys_mac_tx_en	1
		0x4	sys_mac_tx_fcs_ins	1
MAC Rx Config Register	(0xB)	0x0	mac_rx_fcs_remove	1
		0x3	cfg_sys_mac_rx_en	1

For information about the features and registers of Core10GMAC, see **Libero SoC** > **Catalog** > **Core10GMAC** Handbook.

1.2.2 Transceiver Interface (Ask a Question)

The PolarFire high-speed transceiver (PF_XCVR) is a hard IP block and supports data rates from 250 Mbps to 12.5 Gbps. In this demo, PF_XCVR is configured for the data rate of 10312.5 Mbps. It is configured with a CDR reference clock of 156.25 MHz with lock to data selected as the CDR lock mode. The PCS of the transceiver is interfaced with CORE10GMAC. It is configured for the 64/66b mode with scrambler/descrambler enabled. The scrambler, which is self-synchronizing, generates sufficient transitions to aid data and clock recovery at the CDR to enable the Jitter attenuator DPLL, select the Enable JA_CLK. The following figure shows the transceiver interface configuration.

Figure 1-4. Transceiver Interface Configuration

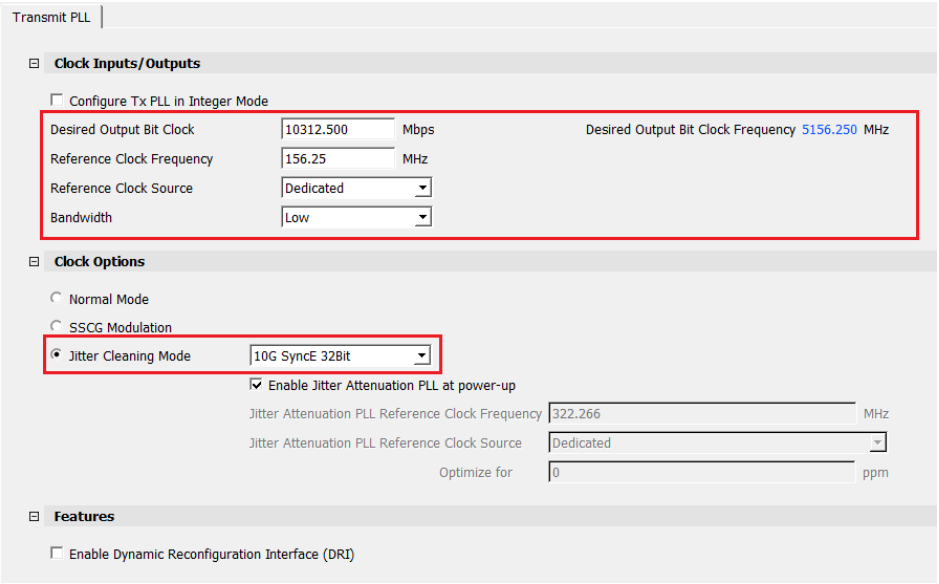


1.2.3 Transmit PLL (Ask a Question)

The PolarFire transmit PLL (PF_TX_PLL) is a hard IP block that provides a bit clock and a reference clock to the transceiver block. The transmit PLL is configured with a reference clock of 156.25 MHz and generates an output clock of 10312.5 Mbps. The clock option is set to Jitter cleaning mode to enable the Jitter attenuator PLL.

The following figure shows the transmit PLL configuration.

Figure 1-5. Transmit PLL Configuration

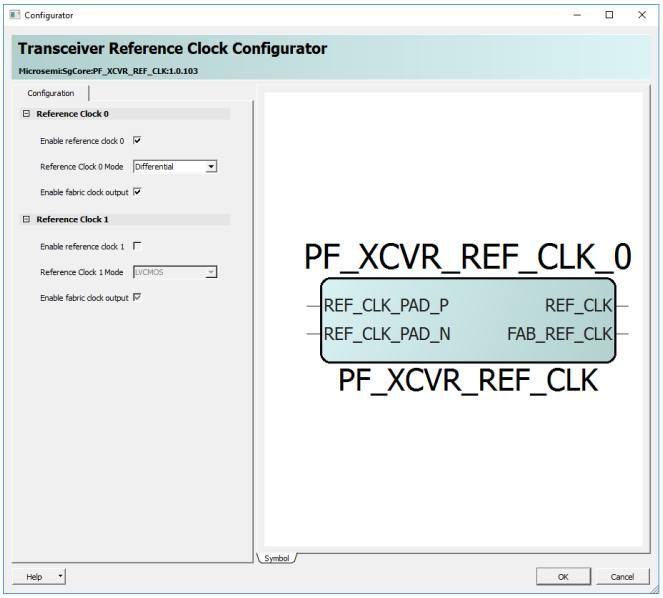


1.2.4 Transceiver Reference Clock (Ask a Question)

The transceiver reference clock (PF_XCVR_REF_CLK) is a hard IP block that provides a reference clock (REF_CLK) of 156.25 MHz to the transmit PLL. A fabric reference clock (FAB_REF_CLK) is also provided as an input to the Clock Conditioning circuit (CCC) to generate the pclk (for configuration) and I_SYS_CLK of the CORE10GMAC.

The following figure shows the transceiver reference clock configuration.

Figure 1-6. Transceiver Reference Clock Configuration



1.2.5 CoreABC (Ask a Question)

CoreABC is a configurable, low-gate count controller intended for Advanced Microcontroller Bus Architecture Advanced Peripheral Bus (AMBA APB)-based designs. Because this demo design requires only a few registers to be configured and no dynamic changes are required in the

configuration, the CoreABC processor is used in this design. Depending on the application requirements, RISC-V, Arm® Cortex®-M1, or any other soft processor might be used for configuring the registers.

1.2.6 CoreAPB3 (Ask a Question)

CoreAPB3 is a bus component that provides an AMBA AHB fabric for interconnection between an APB master and up to 16 APB slaves. CoreAPB3 supports a single APB3 master. In this design, CoreAPB3 is used to connect the CoreABC APB master interface to the CORE10GMAC APB slave interface.

1.2.7 PF_POWER_INIT (Ask a Question)

The PF_POWER_INIT block ensures the device is powered up systematically. The process of powering up the device includes the following steps:

1. Power-on reset
2. Programmed device boot
3. Design initialization

During design initialization, the transceiver configuration is initialized using the data stored in the non-volatile memory. The output of the PF_POWER_INIT block is ANDed with the resets used in the design to reset the entire logic.

1.2.8 PF_CCC_0 (Ask a Question)

The PolarFire Clock Conditioning Circuitry (CCC) block takes an input clock of 156.25 MHz from the FAB_REF_CLK signal (output of PF_XCVR_REF_CLK) and generates a 156.25 MHz clock at OUT0 and 50 MHz clock at OUT1. The OUT1 port of the CCC is used for the configuration; OUT0 is used for the user logic in the design.

1.2.9 FiFo_wrapper_Top (Ask a Question)

FiFo_wrapper_Top is a user-defined RTL module, which uses the CoreFIFO IP to loop the MAC RX PACKET INTERFACE to the MAC TX PACKET INTERFACE.

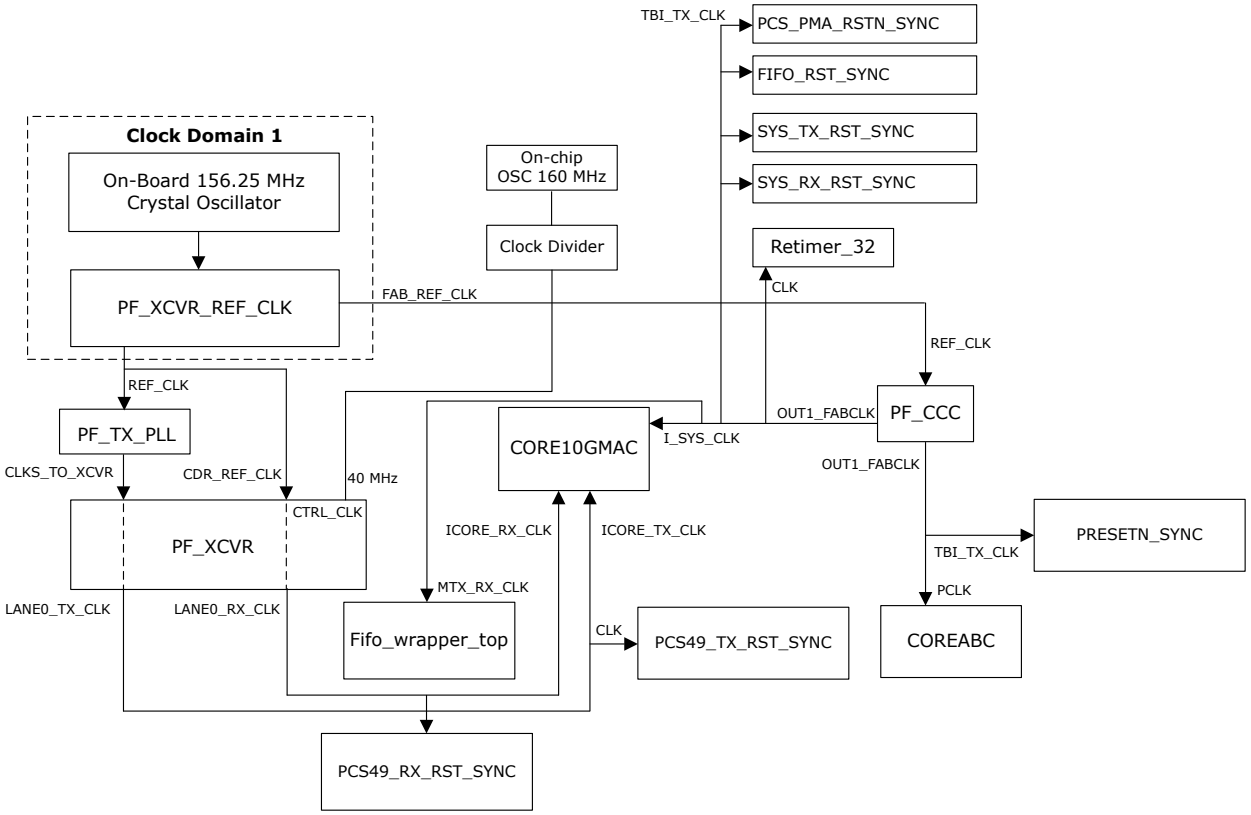
1.2.10 reset_delay (Ask a Question)

reset_delay is a user-defined RTL module, which delays the TX reset signals I_SYS_TX_SRESET and I_PCS49_TX_SRESET of CORE10GMAC by 32 clock cycles to assert the reset once the TX clocks are stable.

1.3 Clocking Structure (Ask a Question)

This design has one clock domain. The on-board 156.25 MHz crystal oscillator generates the clocks used in the demo design. The clock divider generates 40 MHz clock for the XCVR_ERM. The following figure shows the clocking structure of the design.

Figure 1-7. Clocking Structure



1.4 Reset Structure (Ask a Question)

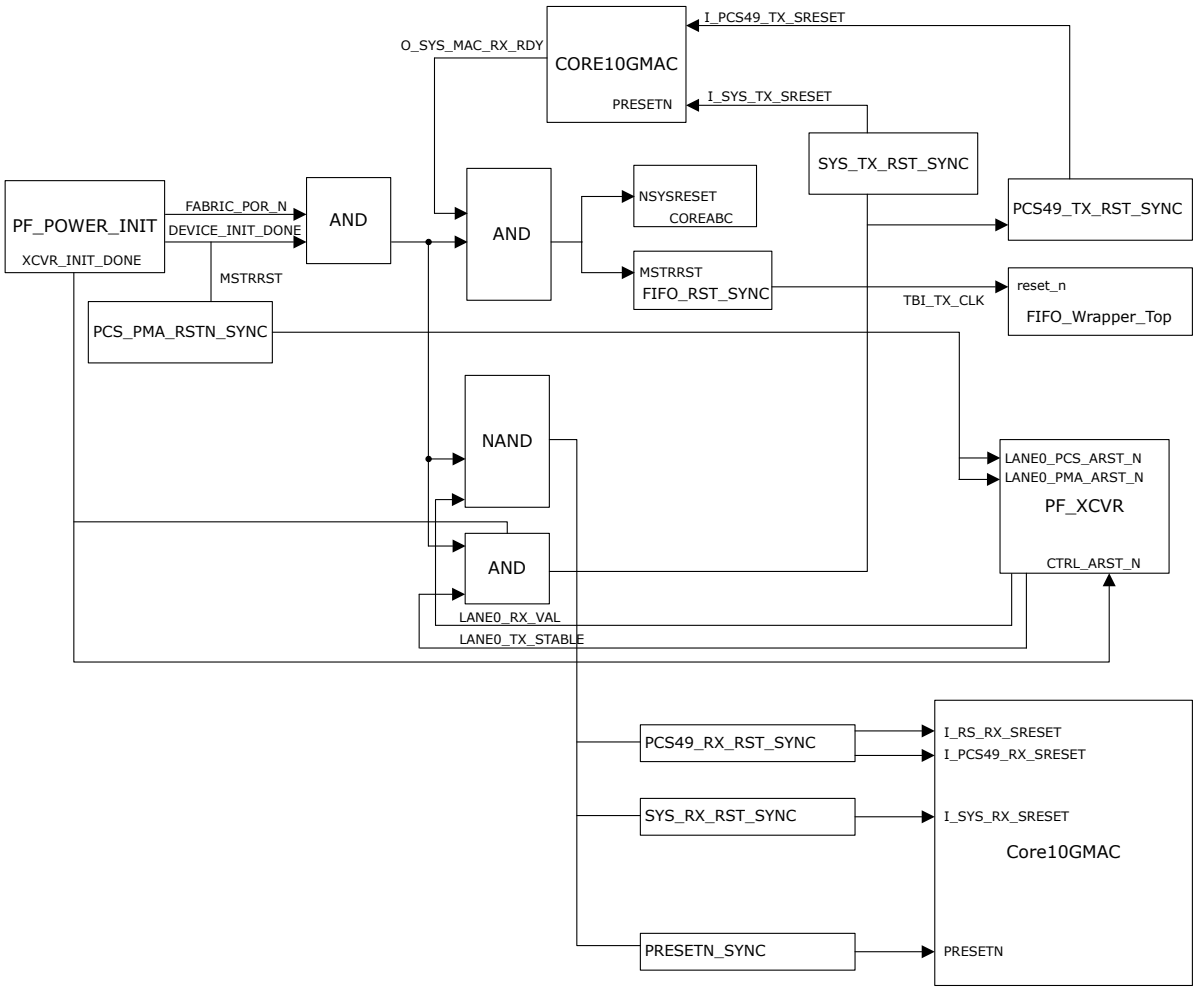
The reset structure of the design is shown in Figure 1-8. The output of the PF_POWER_INIT monitor ANDed with PLL_Lock_0 is used to reset the logic in the design. This output is combined with the following signals to reset the modules in the design:

- The output is ANDed with O_SYS_MAC_RX_RDY to reset the COREABC and FiFo_wrapper_top module. CoreABC configures the CORE10GMAC when MAC RX is ready.
- The output is NAND with the transceiver control signal, LANE0_RX_VAL, and is used to reset the RX path of the CORE10GMAC. The MAC RX path is held in reset until the transceiver, LANE0_RX_VAL is driven to "1".

The PF_POWER_INIT monitor output, ANDed with the transceiver control signal, LANE0_TX_STABLE, is issued to reset the TX path of the CORE10GMAC.

The MAC TX path is held in reset until the RESET_DELAY, OUT is driven to "1". The LANE0_PCS_ARST_N signal of PF_XCVR is reset using O_CORE_TX_SRESET of the CORE10GMAC.

Figure 1-8. Reset Structure



2. Simulating the 10GBASE-R Ethernet SyncE Loopback Design (Ask a Question)

The following sections list the prerequisites for simulation of the 10GBASE-R Ethernet SyncE loopback design, provide details about the design implementation, and describe the simulation flow.

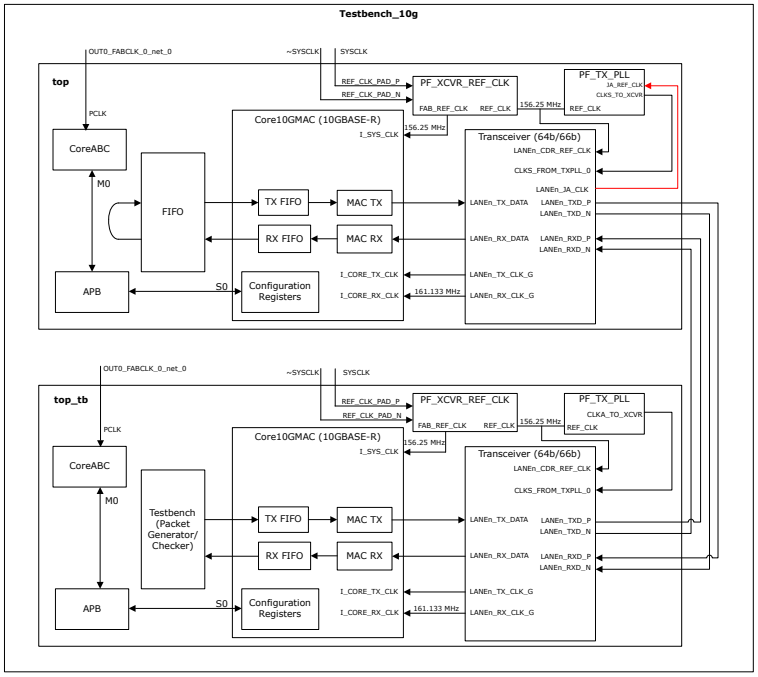
2.1 Prerequisites (Ask a Question)

Before you begin:

1. Launch **Libero SoC**, and select **Project** > **Tool Profiles**.
2. In the **Tool Profiles** window, select **Synthesis** and **Simulation** on the **Tools** panes, and select the latest active installation directory paths for these two tools.

The following figure shows the interaction between testbench and the design.

Figure 2-1. Testbench and 10GBASE-R Ethernet SyncE Loopback Design Interaction



2.2 Design Description (Ask a Question)

The 10GBASE-R Ethernet SyncE loopback simulation design includes the following components:

- **Testbench_10g**: Top testbench module that generates the clocks required for the device under test (DUT) and the testbench submodule, and interconnects the ports from DUT to the testbench submodule.
- **top_tb**: Testbench submodule, which consists of the following major blocks:
  - CoreABC: Configures 10G MAC registers.
  - packet_generator_checker: Performs the packet generator function of defining the Ethernet frame to be transmitted to CORE10GMAC (packet generator). Performs the packet checker function of receiving the looped back Ethernet frame from the CORE10GMAC and comparing it with the transmitted frame.
  - CORE10GMAC: 10 Gbps Ethernet MAC configured in Base-R mode that transmits and receives the Ethernet packets.

- **top**: DUT block of the design, which consists of the following major blocks:
  - CORE10GMAC: 10 Gbps Ethernet MAC configured in Base-R mode that transmits and receives the Ethernet packets.
  - Transceiver: 10GBASE-R physical interface for data transfers; configured in 64b/66b mode with a PCS fabric width of 64 bits.
  - CoreABC: Configures the CORE10GMAC registers.
  - FIFO interface logic: Loops back the Ethernet packets, implemented in Verilog RTL. The following table lists the simulation signals transmitted between the Testbench_10g, top, and top_tb blocks.

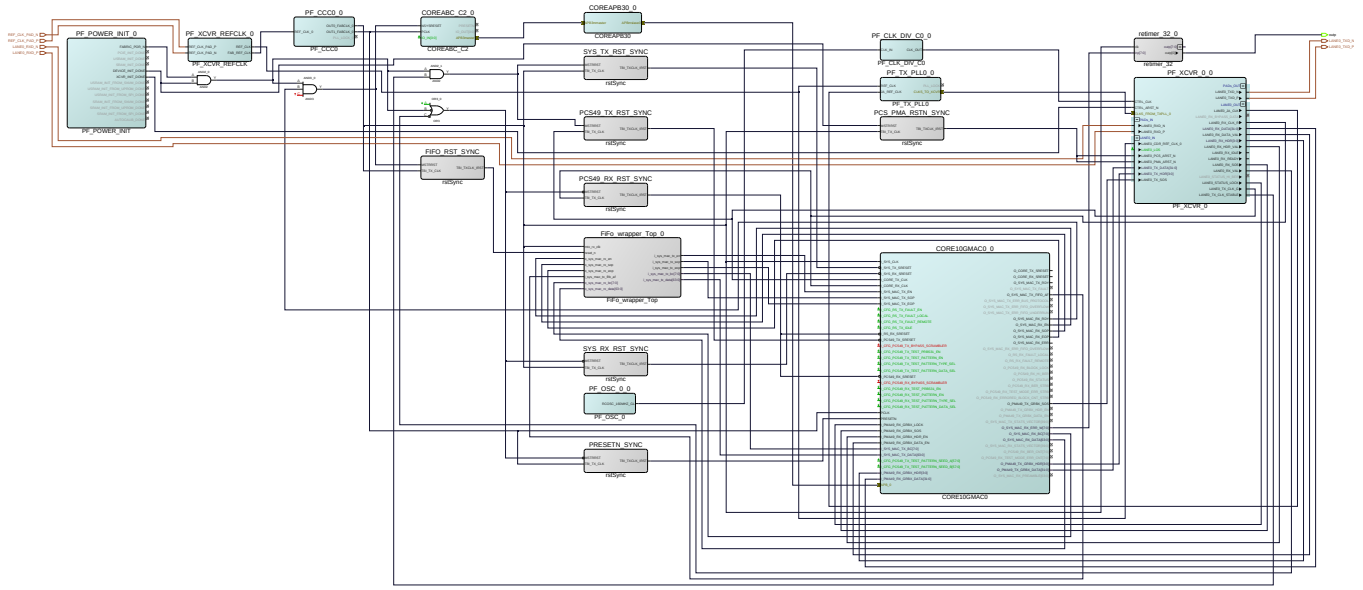
Table 2-1. Simulation Signals

Output From	Input To	Signal	Description
testbench_10g	top, top_tb	SYSCLK	156.25 MHz clock
		OUT0_FABCLK_0_net_0	50 MHz clock
top	testbench_10g	LANE0_RXD_P	Rx port signals
		LANE0_RXD_N	
		LANE0_TXD_P	Tx port signals
		LANE0_TXD_N	
top_tb	testbench_10g	LANE0_RXD_P_0	Rx port signals
		LANE0_RXD_N_0	
		LANE0_TXD_P_0	Tx port signals
		LANE0_TXD_N_0	

2.3 Design Implementation (Ask a Question)

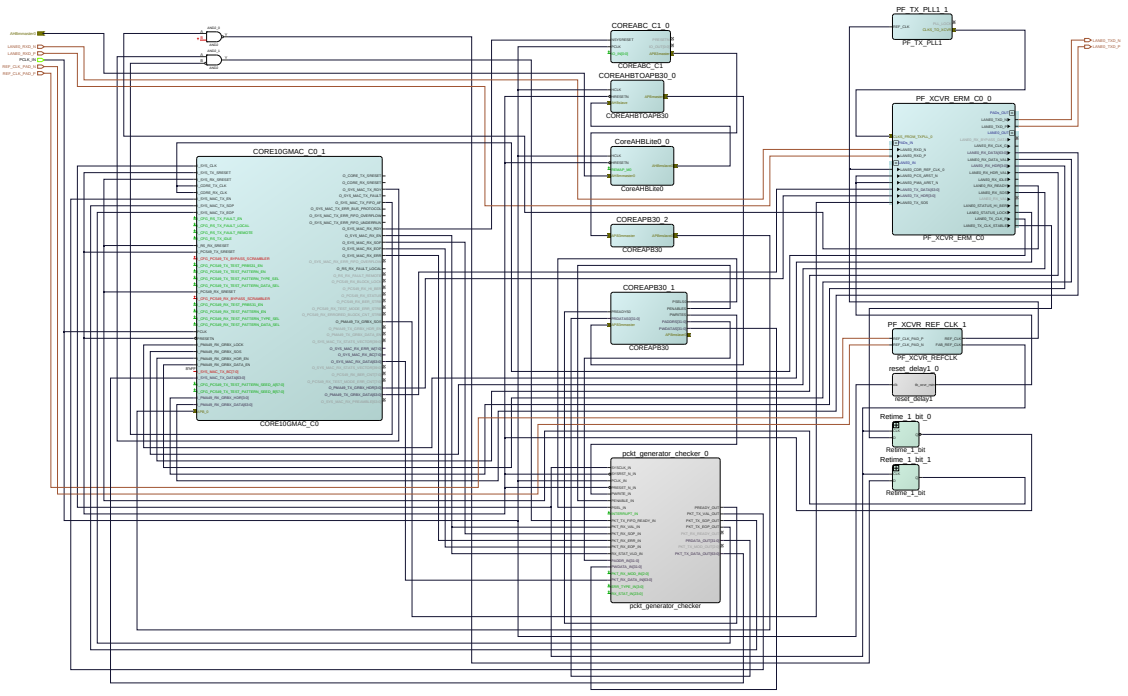
The following figure shows the Libero implementation of the top SmartDesign module.

Figure 2-2. Libero Implementation of the Top SmartDesign Module



The following figure shows the Libero implementation of the top_tb SmartDesign module.

Figure 2-3. Libero Implementation of the top-tb SmartDesign Module



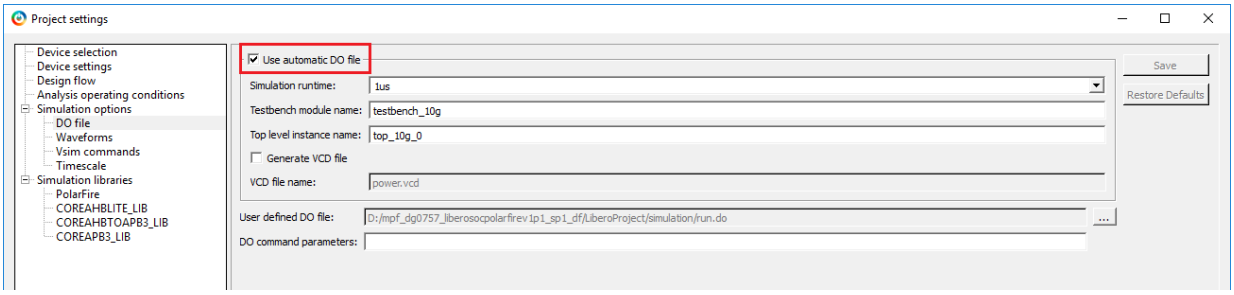
2.4 Simulation Flow (Ask a Question)

The design can be simulated using ModelSim ME Pro provided with the Libero SoC installation. The following sections describe the simulation flow for each of these ModelSim versions.

2.4.1 Initiating Simulation with ModelSim ME Pro 2022.2 (Ask a Question)

In the Libero SoC, go to **Project** > **Project Settings** > **Simulation options** > **DO** file, and ensure the **Use automatic DO** file check box is selected, as shown in the following figure.

Figure 2-4. Use Automatic DO File Option Selected

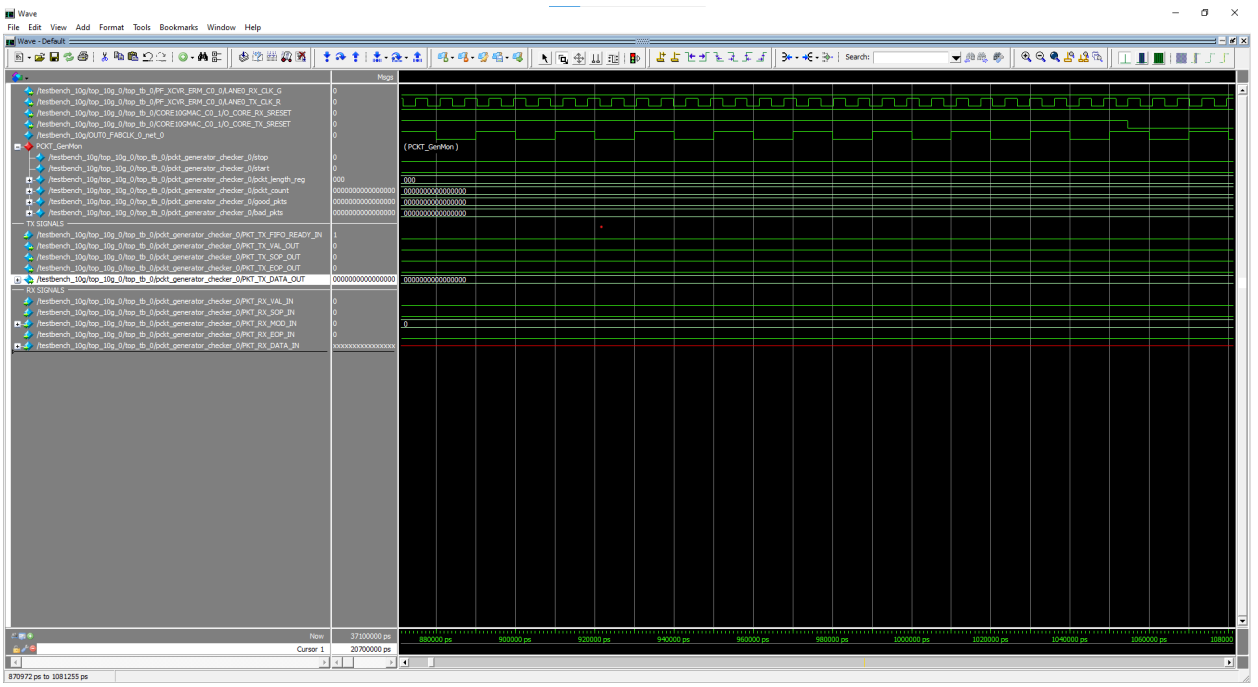


2.4.2 Simulation Results (Ask a Question)

When the simulation is initiated, ModelSim compiles all the design source files, runs the simulation, and launches the waveform viewer to show the simulation signals. The simulation results of the 10GBASE-R Ethernet SyncE loopback design are as follows:

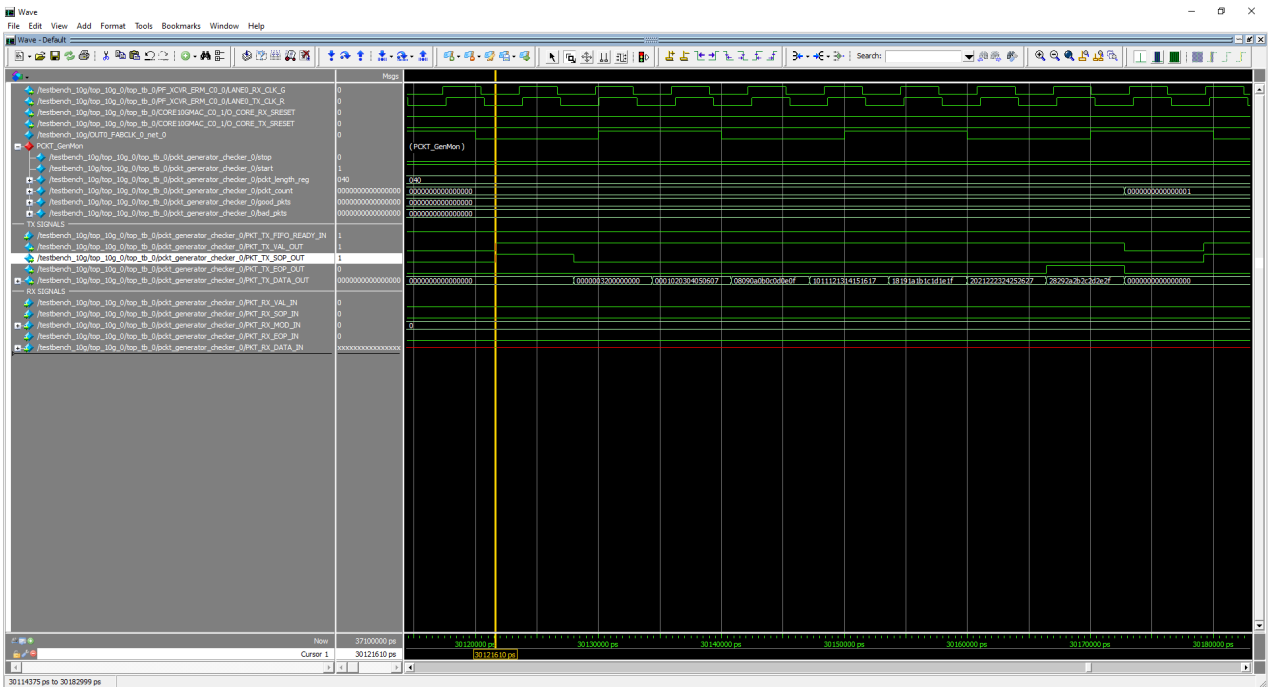
1. At 0 ns, the testbench drives the 156.25 MHz system clock to the DUT.
2. The MAC is released out of the reset. Signal O_CORE_RX_SRESET and O_CORE_RX_SRESET are at 0, as shown in the following figure.

Figure 2-5. O_CORE_RX_SRESET and O_CORE_RX_SRESET at 0



3. The packet generator starts to send the packet. The start signal triggers the packet generation. The size of the packet is set to 0x32 (80 bytes). The sent packet can be viewed on the signals under the TX SIGNALS divider in the wave window.

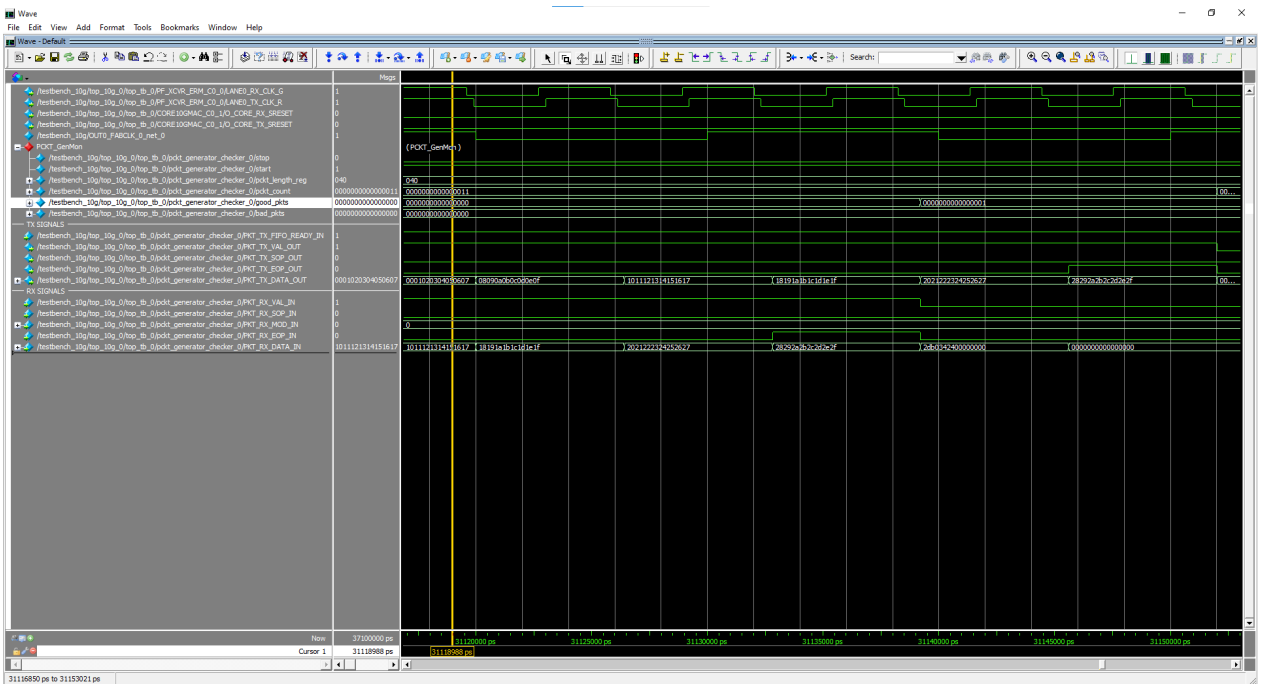
Figure 2-6. Ethernet Packet Sent



4. The packet checker receives the sent packet. The signals can be viewed under the RX SIGNALS divider in the wave window. The packet sent matches with the packet received.

5. The packet checker compares the incoming packet with the sent packet and increments the good packets (good_pckts) count by 1, as shown in the following figure.

Figure 2-7. Good Packets Count Incremented by 1



The sent packet is looped back, and no errors are observed in the received packet, showing successful completion of 10GBASE-R Ethernet SyncE loopback.

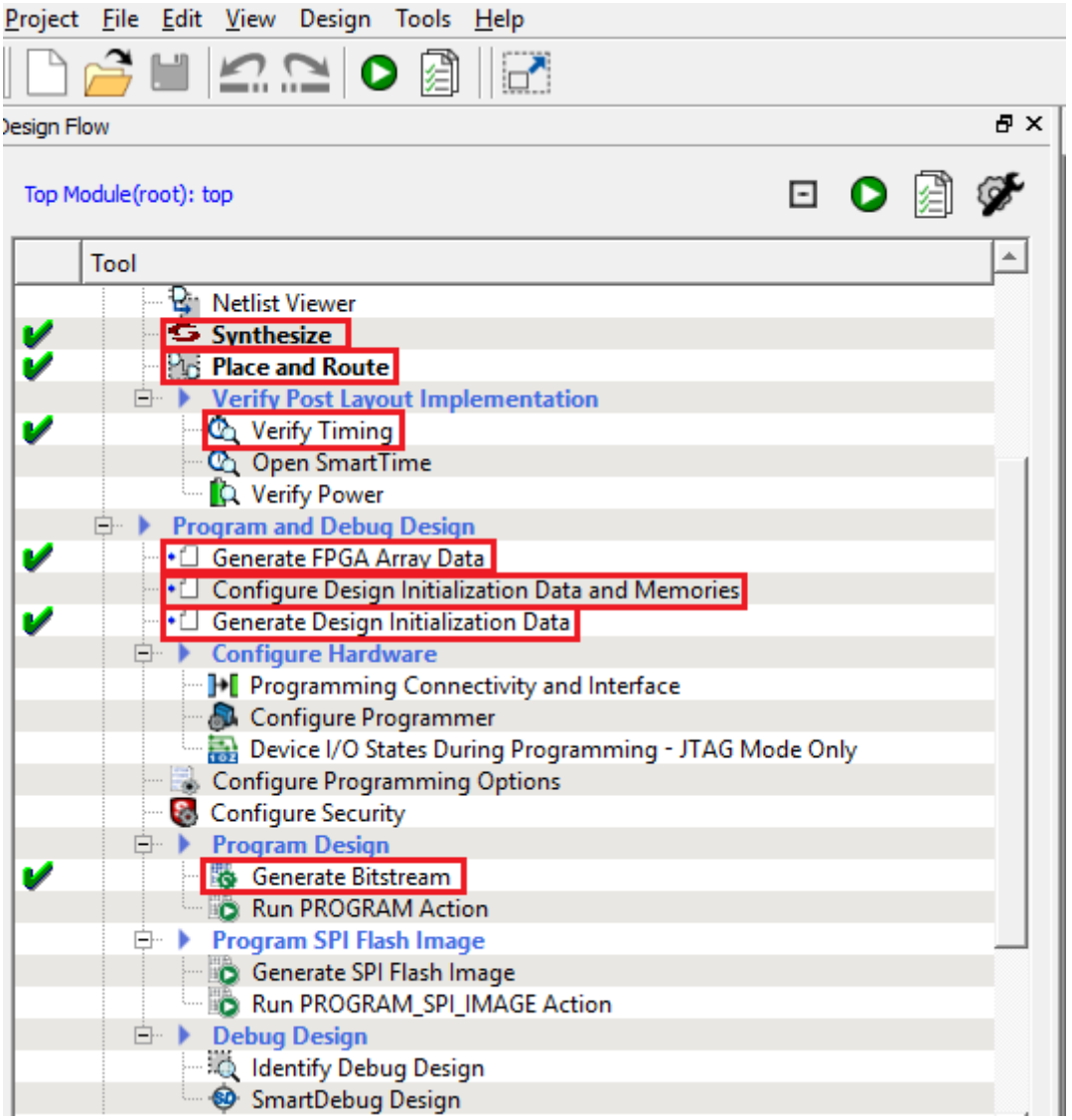
3. Libero Design Flow (Ask a Question)

This section describes the Libero design flow, which involves the following processes:

- 3.1. Synthesize
- 3.2. Place and Route
- 3.3. Verify Timing
- 3.4. Generate FPGA Array Data
- 3.5. Configure Design Initialization Data and Memories
- 3.6. Generate Bitstream
- 3.7. Run Program Action

The following figure shows these options in the **Design Flow** tab.

Figure 3-1. Libero Design Flow Options



3.1 Synthesize (Ask a Question)

To synthesize the design, perform the following steps:

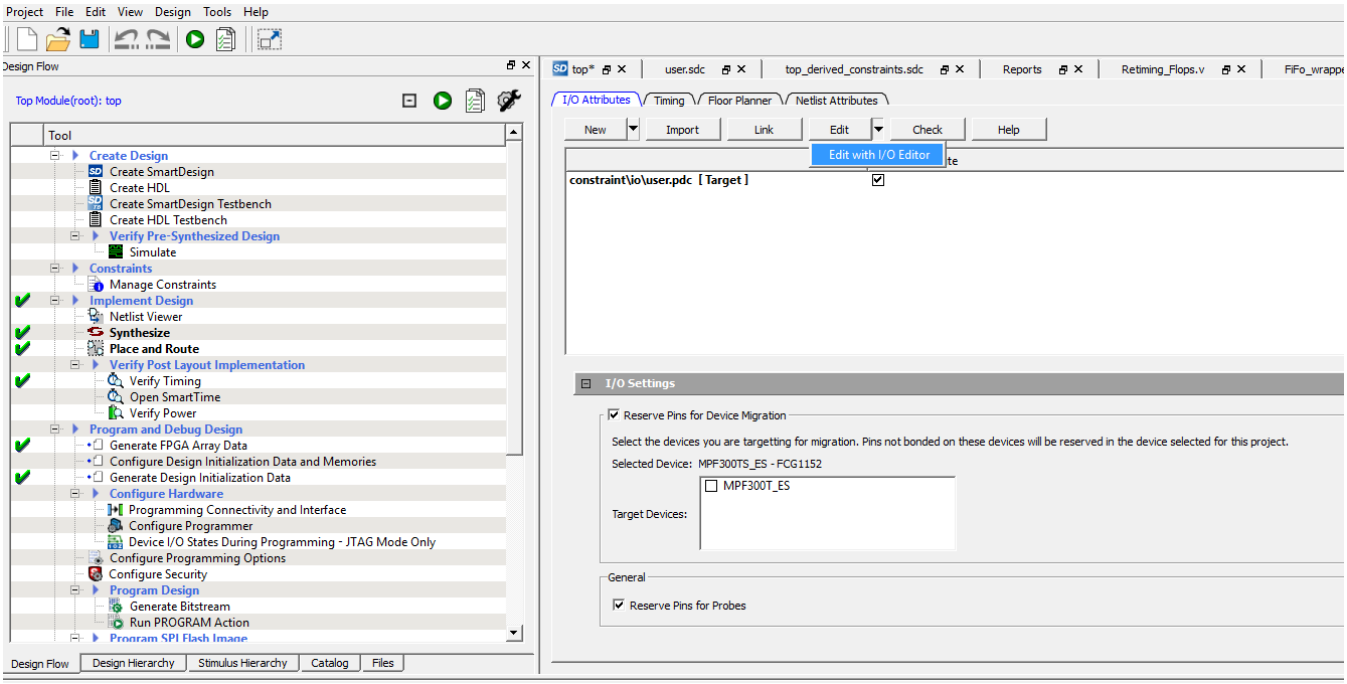
1. On the **Design Flow** window, double-click **Synthesize**. When the synthesis is successful, a green tick mark appears next to **Synthesize**, as shown in Figure 3-1.
2. Right-click **Synthesize** and select **View Report** to view the synthesis report and log files in the **Reports** tab. View the `top_SD.srr` and `top_SD_compile_netlist.log` files to debug synthesis and compile errors.

3.2 Place and Route (Ask a Question)

To place and route the design, TX_PLL, XCVR_REF_CLK and PF_XCVR must be configured using the I/O Editor. Follow these steps to configure the components and place and route the design:

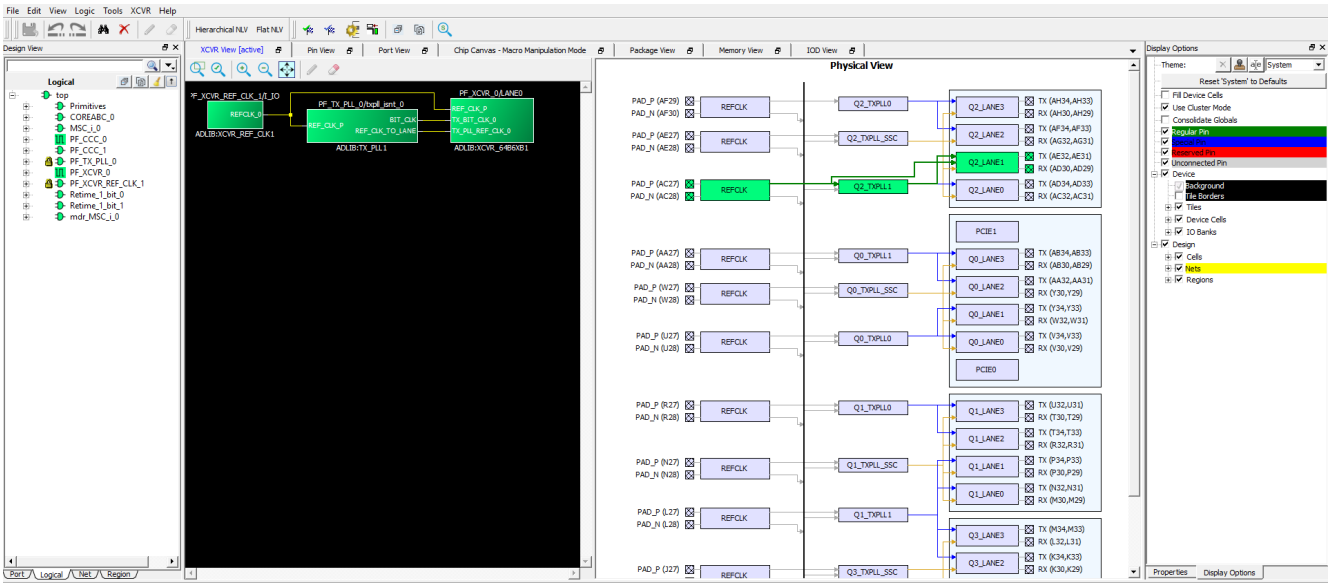
1. On the **Design Flow** window, double-click **Manage Constraints**.
2. On the **I/O Attributes** tab, click **Edit with I/O Editor**, as shown in the following figure.

Figure 3-2. Edit with I/O Editor Option



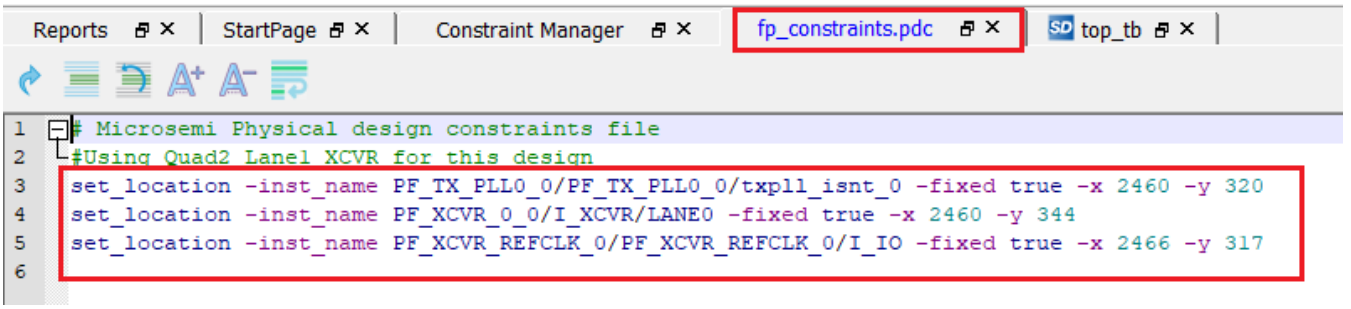
3. Using the XCVR View in I/O Editor, place TX_PLL, XCVR_REF_CLK, and PF_XCVR TX as shown in the following figure.

Figure 3-3. I/O Editor Transceiver View



When all the components are placed, the location of the components is updated in the `user_fp.pdc` file (located in **Constraint Manager** > **Floor planner** tab), as shown in the following figure.

Figure 3-4. Component Locations Updated in user_fp.pdc File



4. On the **Design Flow** window, double-click **Place and Route**.
5. When place and route is successful, a green tick mark appears next to **Place and Route**, as shown in Figure 3-1.
6. Right-click **Place and Route** and select **View Report** to view the place and route report and log files in the **Reports** tab. View the `top_place_and_route_constraint_coverage.xml` file for place and route constraint coverage.

3.2.1 Resource Utilization (Ask a Question)

The resource utilization report is written to the `TOP_SD_layout_log.log` file. To view this file, go to the **Reports tab** > **top reports** > **Place and Route**. The following table lists the resource utilization of the design after place and route. These values may vary slightly for different Libero runs, settings, and seed values.

Table 3-1. Resource Utilization

Type	Used	Total	Percentage
4LUT	5271	299544	1.76

.....continued			
Type	Used	Total	Percentage
DFF	5112	299544	1.71
I/O register	0	510	0.00
Logic element	6698	299544	2.24

3.3 Verify Timing (Ask a Question)

To verify timing, perform the following steps:

1. On the **Design Flow** window, double-click **Verify Timing**. When the design meets the timing requirements, a green tick mark appears next to **Verify Timing**, as shown in Figure 3-1.
2. Right-click **Verify Timing** and select **View Report** to view the verify timing report and log files in the **Reports** tab.

3.4 Generate FPGA Array Data (Ask a Question)

On the **Design Flow** tab, double-click **Generate FPGA Array Data**.

When the FPGA array data is generated, a green tick mark appears next to **Generate FPGA Array Data**, as shown in Figure 3-1.

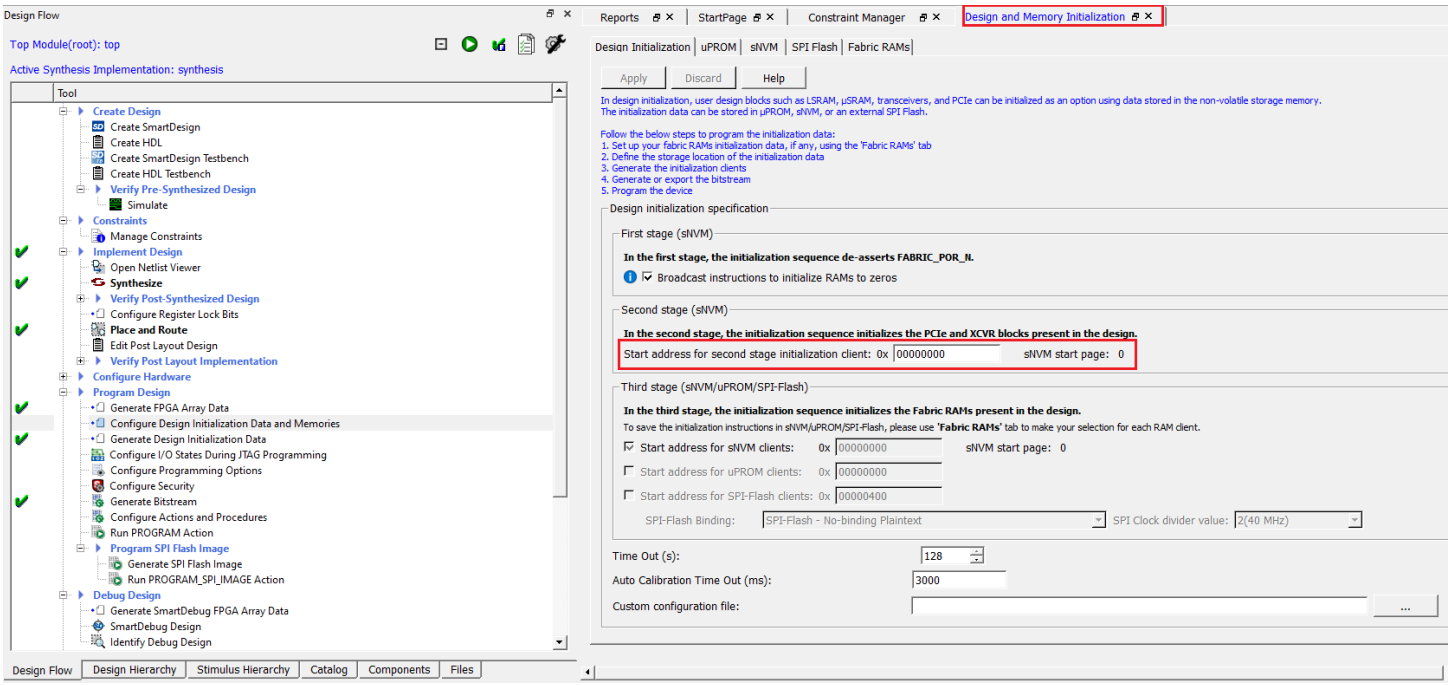
3.5 Configure Design Initialization Data and Memories (Ask a Question)

The **Configure Design Initialization Data and Memories** option creates the non-PCIe transceiver initialization client, which initializes the transceiver block when the PolarFire device powers up.

To create the transceiver initialization client, perform the following steps:

1. On the **Design Flow** window, double-click **Configure Design Initialization Data and Memories**. The Design and Memory Initialization window opens, as shown in the following figure.

Figure 3-5. Design and Memory Initialization Window



2. Under **Second stage pane (sNVM)** enter the start address where the transceiver initialization client must be created in the sNVM, as shown in the preceding figure.

3. On the **Design Flow** window, double-click the **Generate Design Initialization Data** to generate the initialization client.
4. When the initialization client is generated, a green tick mark appears next to **Generate Design Initialization Data**, as shown in Figure 3-1.

3.6 Generate Bitstream (Ask a Question)

To generate the bitstream, perform the following steps:

1. On the **Design Flow** tab, double-click **Generate Bitstream**. When the bitstream is successfully generated, a green tick mark appears next to **Generate Bitstream**, as shown in Figure 3-1.
2. Right-click **Generate Bitstream** and select **View Report** to view the corresponding log file in the **Reports** tab.

3.7 Run Program Action (Ask a Question)

After generating the bitstream file, the PolarFire device must be programmed. Follow these steps to program the PolarFire device.

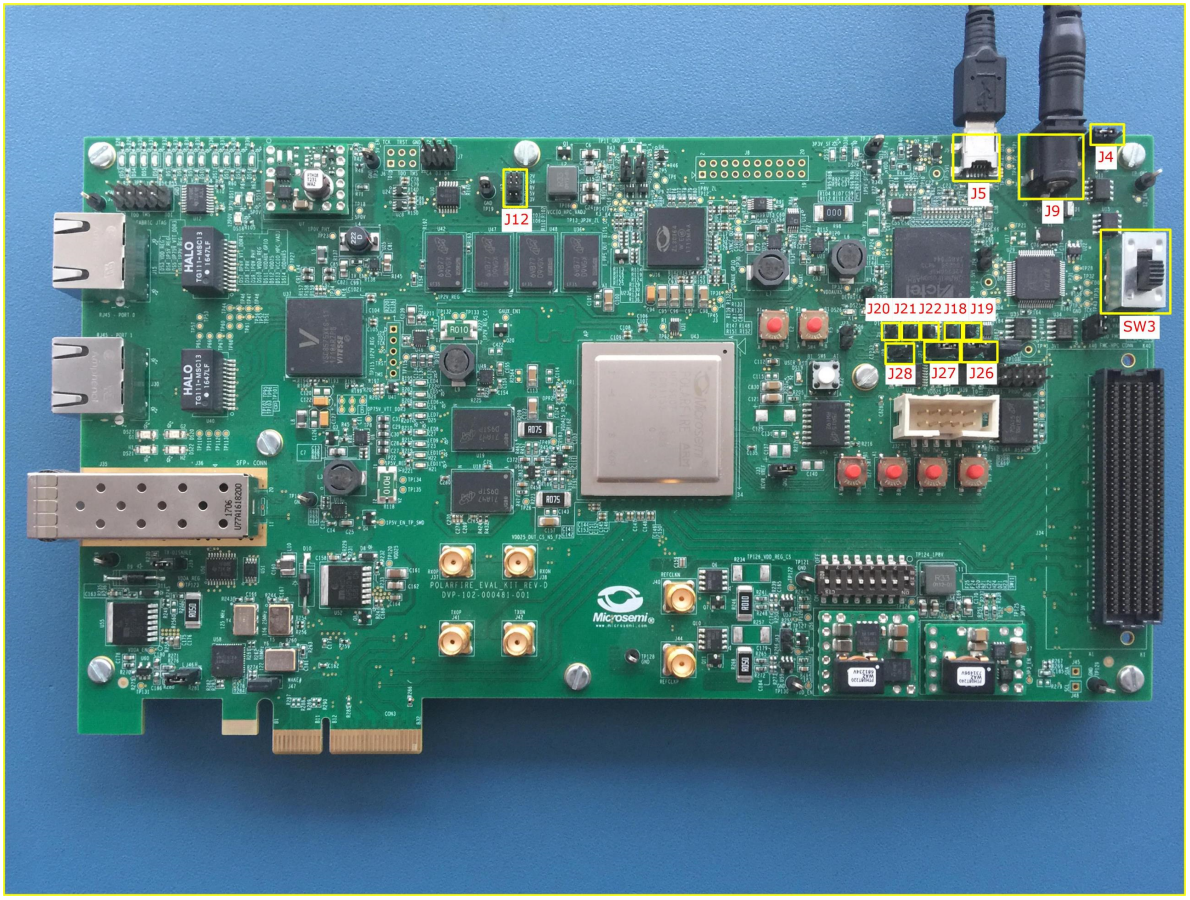
1. Ensure that the jumper settings on the board are as listed in the following table.

Table 3-2. Jumper Settings for PolarFire Device Programming

Jumper	Description
J18, J19, J20, J21, and J22	Short pin 2 and 3 for programming the PolarFire FPGA through FTDI
J28	Short pin 1 and 2 for programming through the on-board FlashPro Express
J26	Short pin 1 and 2 for programming through the FTDI SPI
J27	Short pin 1 and 2 for programming through the FTDI SPI
J39	Short pin 1 and 2 for enabling the TX
J4	Short pin 1 and 2 for manual power switching using SW3

2. Connect the power supply cable to the **J9** connector on the board.
3. Connect the host PC to the **J5** connector (FTDI port) on the board using a USB cable.
4. Power on the board using the **SW3** slide switch. The following figure shows the PolarFire Evaluation Board setup for programming the device and running the reference design.

Figure 3-6. PolarFire Evaluation Board Setup



5. Double-click **Run PROGRAM Action** from the **Libero** > **Design Flow** tab. When the device is successfully programmed, a green tick mark appears next to **Run PROGRAM Action**, as shown in Figure 3-1.

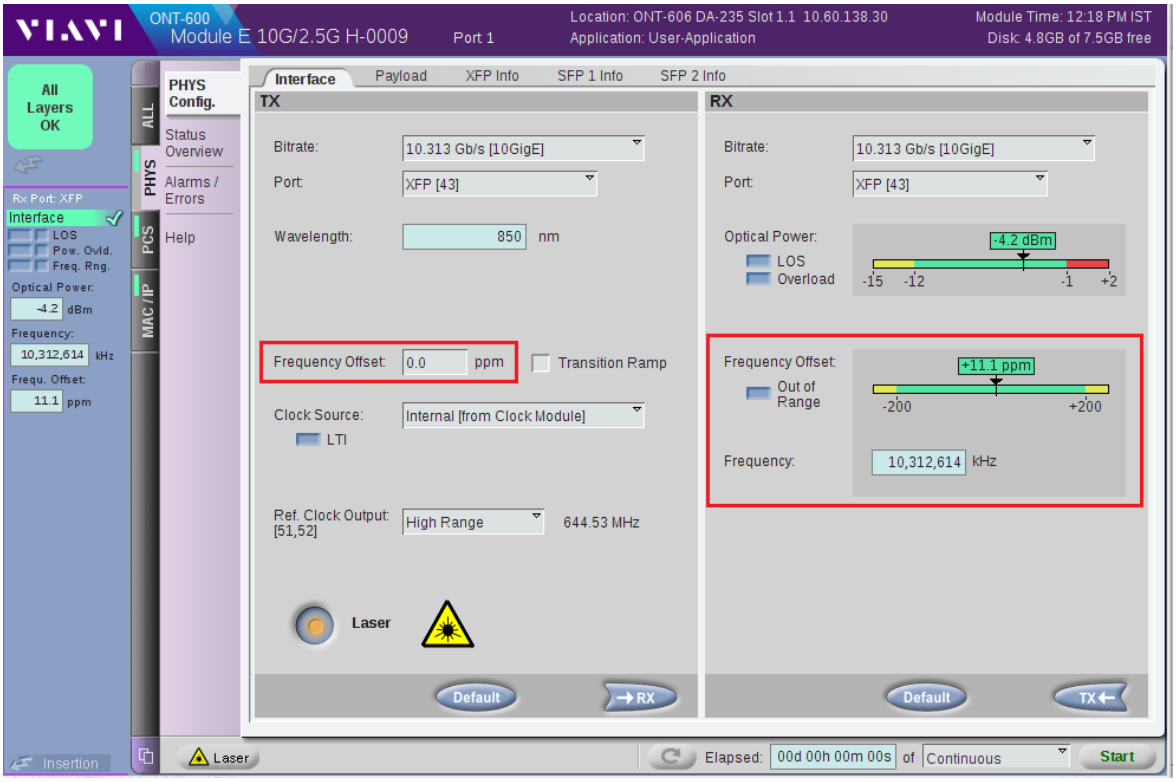
For information about running the demo, see 4. Running the Demo.

4. Running the Demo (Ask a Question)

Follow these steps to run the PolarFire 10GBASE-R Ethernet SyncE loopback hardware demo design on the PolarFire Evaluation Board.

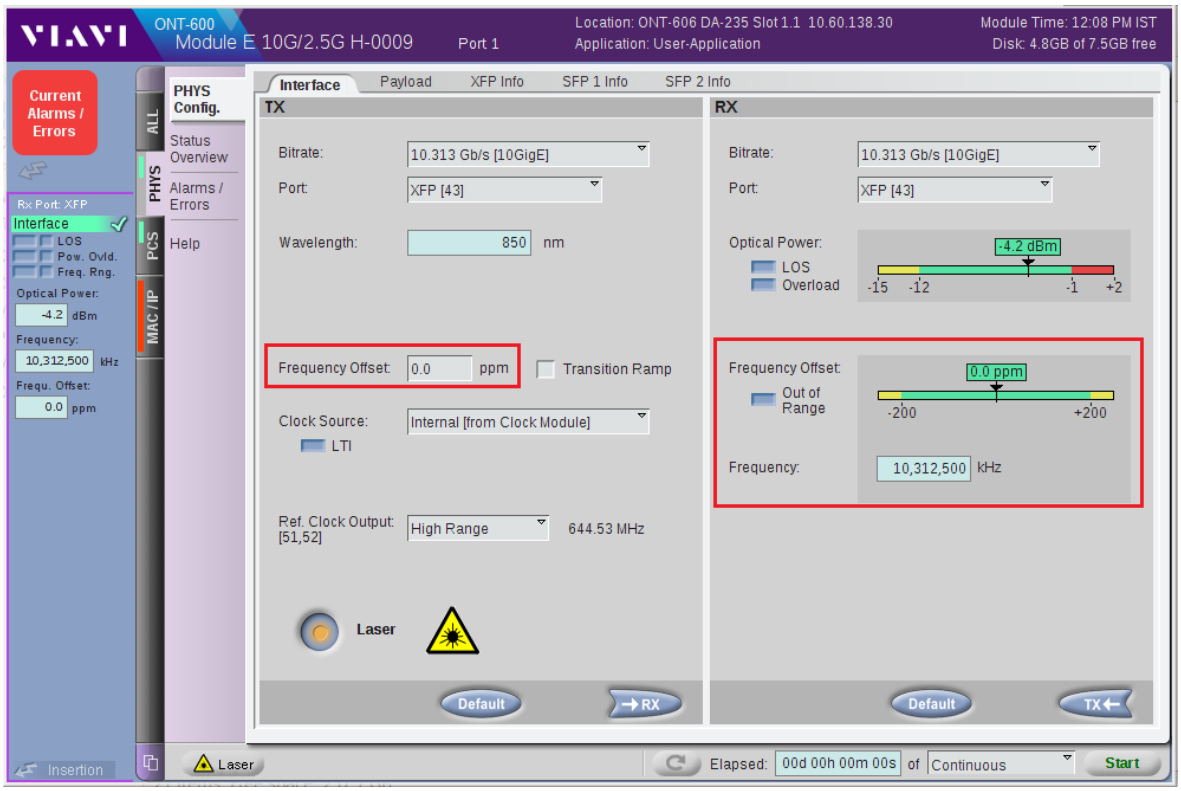
1. The reference design is validated using ONT. There exists the ppm offset in the TX and RX frequencies when the Jitter Cleaning Mode is not enabled in the TX PLL as shown in the following figure.

Figure 4-1. Frequency Offset—SyncE Disabled



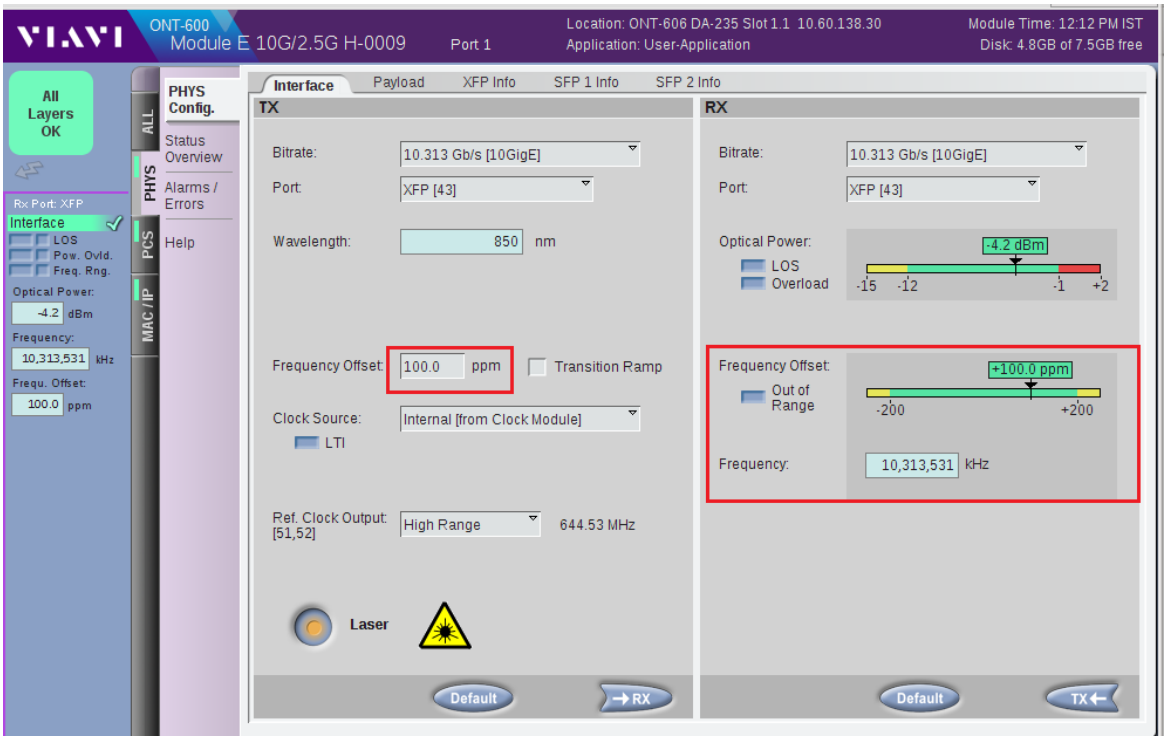
2. When the design is built with Jitter Cleaning Mode of TX PLL enabled the Frequency offset between TX and RX is 0 ppm.

Figure 4-2. Frequency Offset Between TX and RX



3. When the TX clock frequency is offset by 100 ppm, the RX clock frequency also gets adjusted by 100 ppm, which shows that JA PLL is tracking and adjusting the clock as per the offset in the received clock.
4. The data generated by the ONT tester is looped back at the FIFO and received at the ONT tester. The data throughput is verified and displayed on the ONT screen by the All Layers OK indicator in green color, as shown in the following figure.

Figure 4-3. Frequency Offset—SyncE Enabled



5. Appendix 1: Programming the Device Using FlashPro Express (Ask a Question)

This section describes how to program the PolarFire device with the Job programming file using a FlashPro programmer. The default location of the Job file is located at the following location: `mpf_an5102_v2023p1_df\Programming_Files\top.job`

To program the PolarFire device using FlashPro Express, complete the following steps:

1. Ensure that the jumper settings on the board are the same as listed in Table 3-2.
   Note: The power supply switch must be switched off while making the jumper connections.
2. Connect the power supply cable to the **J9** connector on the board.
3. Connect the USB cable from the Host PC to the **J5** (FTDI port) on the board.
4. Power on the board using the **SW3** slide switch.
5. On the host PC, launch the **FlashPro Express** software.
6. To create a new job project, click **New** or on the **Project** menu, select **New Job Project from FlashPro Express Job**.
7. Enter the following in the New Job Project from FlashPro Express Job dialog box:
   - Programming job file: Click **Browse**, and navigate to the location where the `.job` file is located and select the file. The default location is: `<download_folder>\mpf_an5102_v2023p1_df\Programming_Files`
   - FlashPro Express job project location: Click **Browse** and navigate to the location where you want to save the project.
8. Click **OK**. The required programming file is selected and ready to be programmed in the device.
9. The FlashPro Express window appears. Confirm that a programmer number appears in the Programmer field. If it does not, confirm the board connections and click **Refresh/Rescan Programmers**.
10. Click **RUN**. When the device is programmed successfully, a RUN PASSED status is displayed as shown in the following figure.
11. Close **FlashPro Express** or in the **Project** tab, click **Exit**.

6. Appendix 2: Running the TCL Script (Ask a Question)

TCL scripts are provided in the design files folder under directory TCL_Scripts. If required, the design flow can be reproduced from Design Implementation till generation of job file.

To run the TCL, follow the steps:

1. Launch the Libero software.
2. Select **Project** > **Execute Script....**
3. Click **Browse** and select `script.tcl` from the downloaded TCL_Scripts directory.
4. Click **Run**.

After successful execution of TCL script, Libero project is created within TCL_Scripts directory.

For more information about TCL scripts, refer to `mpf_an5102_v2023p1_df/TCL_Scripts/readme.txt`.

See the Libero® SoC TCL Command Reference Guide for more details on TCL commands. Contact Technical Support for any queries encountered while running the TCL script.

7. Revision History (Ask a Question)

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

Revision	Date	Description
A	08/2023	The following is the list of changes in revision A of the document: • The document was migrated to the Microchip template. • The document number was updated to DS00005102 from 50200757. • The document was updated for Libero SoC v2023.1 release.
8.0	—	The following is a summary of the changes made in this revision. • Updated the document for Libero SoC v12.2. • Removed the references to Libero version numbers.
7.0	—	The document was updated for Libero SoC v12.0 release.
6.0	—	The document was updated for Libero SoC PolarFire v2.2 release.
5.0	—	The document was updated for Libero SoC PolarFire v2.1 release.
4.0	—	The following is a summary of the changes made in this revision. • The document was updated to include features and enhancements introduced in the Libero SoC PolarFire v2.0 release. • The design requirements were updated. For more information, see 1. Design Requirements. • Details about the demo design, including the hardware implementation block diagram, were updated. For more information, see 1. Demo Design. • Clocking Structure diagram was added. For more information, see Figure 1-7. • A new section which details the reset structure of the design is added. For more information, see 1.4. Reset Structure. • Information about simulating the design was updated. For more information, see 2. Simulating the 10GBASE-R Ethernet SyncE Loopback Design.
3.0	—	The following is a summary of the changes made in this revision. • The document was updated to include features and enhancements introduced in the Libero SoC PolarFire v1.1 SP1 release. • Hardware requirements were added, and software requirements were updated to include Spirent TestCenter and FlashPro. For more information, see 1. Design Requirements. • Information about how to program the device was added. For more information, see 5. Appendix 1: Programming the Device Using FlashPro Express. • Information about how to run the hardware reference design was added. For more information, see 4. Running the Demo.
2.0	—	The following is a summary of the changes made in this revision. • The document was updated for Libero SoC PolarFire v1.1 release. • Information about resource utilization was added. For more information, see3.2.1. Resource Utilization.
1.0	—	The first publication of this document.

Microchip FPGA Support

Microchip FPGA products group backs its products with various support services, including Customer Service, Customer Technical Support Center, a website, and worldwide sales offices. Customers are suggested to visit Microchip online resources prior to contacting support as it is very likely that their queries have been already answered.

Contact Technical Support Center through the website at www.microchip.com/support. Mention the FPGA Device Part number, select appropriate case category, and upload design files while creating a technical support case.

Contact Customer Service for non-technical product support, such as product pricing, product upgrades, update information, order status, and authorization.

- From North America, call **800.262.1060**
- From the rest of the world, call **650.318.4460**
- Fax, from anywhere in the world, **650.318.8044**

Microchip Information

The Microchip Website

Microchip provides online support via our website at www.microchip.com/. This website is used to make files and information easily available to customers. Some of the content available includes:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQs), technical support requests, online discussion groups, Microchip design partner program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

Product Change Notification Service

Microchip's product change notification service helps keep customers current on Microchip products. Subscribers will receive email notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, go to www.microchip.com/pcn and follow the registration instructions.

Customer Support

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Embedded Solutions Engineer (ESE)
- Technical Support

Customers should contact their distributor, representative or ESE for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in this document.

Technical support is available through the website at: www.microchip.com/support

Microchip Devices Code Protection Feature

Note the following details of the code protection feature on Microchip products:

- Microchip products meet the specifications contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is secure when used in the intended manner, within operating specifications, and under normal conditions.
- Microchip values and aggressively protects its intellectual property rights. Attempts to breach the code protection features of Microchip product is strictly prohibited and may violate the Digital Millennium Copyright Act.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of its code. Code protection does not mean that we are guaranteeing the product is “unbreakable”. Code protection is constantly evolving. Microchip is committed to continuously improving the code protection features of our products.

Legal Notice

This publication and the information herein may be used only with Microchip products, including to design, test, and integrate Microchip products with your application. Use of this information in any other manner violates these terms. Information regarding device applications is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. Contact your local Microchip sales office for additional support or, obtain additional support at www.microchip.com/en-us/support/design-help/client-support-services.

THIS INFORMATION IS PROVIDED BY MICROCHIP "AS IS". MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE, OR WARRANTIES RELATED TO ITS CONDITION, QUALITY, OR PERFORMANCE.

IN NO EVENT WILL MICROCHIP BE LIABLE FOR ANY INDIRECT, SPECIAL, PUNITIVE, INCIDENTAL, OR CONSEQUENTIAL LOSS, DAMAGE, COST, OR EXPENSE OF ANY KIND WHATSOEVER RELATED TO THE INFORMATION OR ITS USE, HOWEVER CAUSED, EVEN IF MICROCHIP HAS BEEN ADVISED OF THE POSSIBILITY OR THE DAMAGES ARE FORESEEABLE. TO THE FULLEST EXTENT ALLOWED BY LAW, MICROCHIP'S TOTAL LIABILITY ON ALL CLAIMS IN ANY WAY RELATED TO THE INFORMATION OR ITS USE WILL NOT EXCEED THE AMOUNT OF FEES, IF ANY, THAT YOU HAVE PAID DIRECTLY TO MICROCHIP FOR THE INFORMATION.

Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Trademarks

The Microchip name and logo, the Microchip logo, Adaptec, AVR, AVR logo, AVR Freaks, BesTime, BitCloud, CryptoMemory, CryptoRF, dsPIC, flexPWR, HELDO, IGLOO, JukeBlox, KeeLoq, Kleer, LANCheck, LinkMD, maXStylus, maXTouch, MediaLB, megaAVR, Microsemi, Microsemi logo, MOST, MOST logo, MPLAB, OptoLyzer, PIC, picoPower, PICSTART, PIC32 logo, PolarFire, Prochip Designer, QTouch, SAM-BA, SenGenuity, SpyNIC, SST, SST Logo, SuperFlash, Symmetricom, SyncServer, Tachyon, TimeSource, tinyAVR, UNI/O, Vectron, and XMEGA are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

AgileSwitch, APT, ClockWorks, The Embedded Control Solutions Company, EtherSynch, Flashtec, Hyper Speed Control, HyperLight Load, Libero, motorBench, mTouch, Powermite 3, Precision Edge, ProASIC, ProASIC Plus, ProASIC Plus logo, Quiet- Wire, SmartFusion, SyncWorld, Temux, TimeCesium, TimeHub, TimePictra, TimeProvider, TrueTime, and ZL are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Adjacent Key Suppression, AKS, Analog-for-the-Digital Age, Any Capacitor, AnyIn, AnyOut, Augmented Switching, BlueSky, BodyCom, Clockstudio, CodeGuard, CryptoAuthentication, CryptoAutomotive, CryptoCompanion, CryptoController, dsPICDEM, dsPICDEM.net, Dynamic

ISBN: 978-1-6683-3030-2

Quality Management System

For information regarding Microchip's Quality Management Systems, please visit www.microchip.com/quality.

Worldwide Sales and Service

AMERICAS	ASIA/PACIFIC	ASIA/PACIFIC	EUROPE
Corporate Office	**Australia - Sydney**	**India - Bangalore**	**Austria - Wels**
2355 West Chandler Blvd.	Tel: 61-2-9868-6733	Tel: 91-80-3090-4444	Tel: 43-7242-2244-39
Chandler, AZ 85224-6199	**China - Beijing**	**India - New Delhi**	Fax: 43-7242-2244-393
Tel: 480-792-7200	Tel: 86-10-8569-7000	Tel: 91-11-4160-8631	**Denmark - Copenhagen**
Fax: 480-792-7277	**China - Chengdu**	**India - Pune**	Tel: 45-4485-5910
Technical Support:	Tel: 86-28-8665-5511	Tel: 91-20-4121-0141	Fax: 45-4485-2829
www.microchip.com/support	**China - Chongqing**	**Japan - Osaka**	**Finland - Espoo**
Web Address:	Tel: 86-23-8980-9588	Tel: 81-6-6152-7160	Tel: 358-9-4520-820
www.microchip.com	**China - Dongguan**	**Japan - Tokyo**	**France - Paris**
Atlanta	Tel: 86-769-8702-9880	Tel: 81-3-6880- 3770	Tel: 33-1-69-53-63-20
Duluth, GA	**China - Guangzhou**	**Korea - Daegu**	Fax: 33-1-69-30-90-79
Tel: 678-957-9614	Tel: 86-20-8755-8029	Tel: 82-53-744-4301	**Germany - Garching**
Fax: 678-957-1455	**China - Hangzhou**	**Korea - Seoul**	Tel: 49-8931-9700
Austin, TX	Tel: 86-571-8792-8115	Tel: 82-2-554-7200	**Germany - Haan**
Tel: 512-257-3370	**China - Hong Kong SAR**	**Malaysia - Kuala Lumpur**	Tel: 49-2129-3766400
Boston	Tel: 852-2943-5100	Tel: 60-3-7651-7906	**Germany - Heilbronn**
Westborough, MA	**China - Nanjing**	**Malaysia - Penang**	Tel: 49-7131-72400
Tel: 774-760-0087	Tel: 86-25-8473-2460	Tel: 60-4-227-8870	**Germany - Karlsruhe**
Fax: 774-760-0088	**China - Qingdao**	**Philippines - Manila**	Tel: 49-721-625370
Chicago	Tel: 86-532-8502-7355	Tel: 63-2-634-9065	**Germany - Munich**
Itasca, IL	**China - Shanghai**	**Singapore**	Tel: 49-89-627-144-0
Tel: 630-285-0071	Tel: 86-21-3326-8000	Tel: 65-6334-8870	Fax: 49-89-627-144-44
Fax: 630-285-0075	**China - Shenyang**	**Taiwan - Hsin Chu**	**Germany - Rosenheim**
Dallas	Tel: 86-24-2334-2829	Tel: 886-3-577-8366	Tel: 49-8031-354-560
Addison, TX	**China - Shenzhen**	**Taiwan - Kaohsiung**	**Israel - Ra'anana**
Tel: 972-818-7423	Tel: 86-755-8864-2200	Tel: 886-7-213-7830	Tel: 972-9-744-7705
Fax: 972-818-2924	**China - Suzhou**	**Taiwan - Taipei**	**Italy - Milan**
Detroit	Tel: 86-186-6233-1526	Tel: 886-2-2508-8600	Tel: 39-0331-742611
Novi, MI	**China - Wuhan**	**Thailand - Bangkok**	Fax: 39-0331-466781
Tel: 248-848-4000	Tel: 86-27-5980-5300	Tel: 66-2-694-1351	**Italy - Padova**
Houston, TX	**China - Xian**	**Vietnam - Ho Chi Minh**	Tel: 39-049-7625286
Tel: 281-894-5983	Tel: 86-29-8833-7252	Tel: 84-28-5448-2100	**Netherlands - Drunen**
Indianapolis	**China - Xiamen**		Tel: 31-416-690399
Noblesville, IN	Tel: 86-592-2388138		Fax: 31-416-690340
Tel: 317-773-8323	**China - Zhuhai**		**Norway - Trondheim**
Fax: 317-773-5453	Tel: 86-756-3210040		Tel: 47-72884388
Tel: 317-536-2380			**Poland - Warsaw**
Los Angeles			Tel: 48-22-3325737
Mission Viejo, CA			**Romania - Bucharest**
Tel: 949-462-9523			Tel: 40-21-407-87-50
Fax: 949-462-9608			**Spain - Madrid**
Tel: 951-273-7800			Tel: 34-91-708-08-90
Raleigh, NC			Fax: 34-91-708-08-91
Tel: 919-844-7510			**Sweden - Gothenberg**
New York, NY			Tel: 46-31-704-60-40
Tel: 631-435-6000			**Sweden - Stockholm**
San Jose, CA			Tel: 46-8-5090-4654
Tel: 408-735-9110			**UK - Wokingham**
Tel: 408-436-4270			Tel: 44-118-921-5800
Canada - Toronto			Fax: 44-118-921-5820
Tel: 905-695-1980			
Fax: 905-695-2078			