AC489
Application Note
Building the PolarFire SoC MSS Design

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.

51900489. 2.0 5/20

Contents

Figures

Tables

1 Building the MSS Design

Microchip's PolarFire SoC FPGAs include industry's first RISC-V based Microprocessor Subsystem (MSS) and a fabric that inherits all the features of the PolarFire family. The PolarFire SoC MSS includes 5x 64-bit RISC-V processor cores, DDR Controller, Fabric Interface Controllers (FIC), and a rich set of peripherals. For more information about the PolarFire SoC MSS and its components, see *UG0880: PolarFire SoC FPGA MSS User Guide*.

The Libero SoC design suite includes a wide range of IP cores, which enable FPGA designers to build PolarFire SoC FPGA designs for various application domains. Libero SoC includes the PFSOC_MSS SgCore IP which is used to build the MSS design in PolarFire SoC FPGAs.

This application note describes how to build the MSS design with peripherals using the PFSOC_MSS SgCore IP in Libero SoC. A reference design is included enabling designers to quickly rebuild the MSS design using the options available in the PFSOC_MSS SgCore IP Configurator.

1.1 Design Requirements

Table 1 lists hardware and software required to build the MSS design.

Table 1 • **Design Requirements**[1]

Software	**Description**
-Libero SoC	v12.4
Hardware	
-Host PC	Windows 10
License	
-Libero Gold License or higher -PolarFire SoC License	Required for Libero SoC Required for the PolarFire SoC family

1. Programming the reference design on a Hardware Kit will be supported in future releases.

1.2 Prerequisites

Before you start:

1. Download the reference design files using the following link:
   http://soc.microsemi.com/download/rsc/?f=mpfs_ac489_df
2. Download and install Libero SoC using the following link:
   https://www.microsemi.com/product-directory/design-resources/1750-libero-soc#downloads
3. Contact the local sales team to get the PolarFire SoC license along with the Libero SoC license to design for the PolarFire SoC family.
4. To work offline with Libero SoC, download and install the Libero SoC Mega Vault. Mega Vault is the repository of all the IP cores (including the PFSOC_MSS Sgcore IP) required to design for the PolarFire SoC family. To change the vault setting, see Vault Setting in Libero.

1.3 Vault Setting in Libero

To work offline with Libero SoC, the vault location must be set to the Mega Vault in Libero SoC.

Figure 1 • **Vault Settings Option**

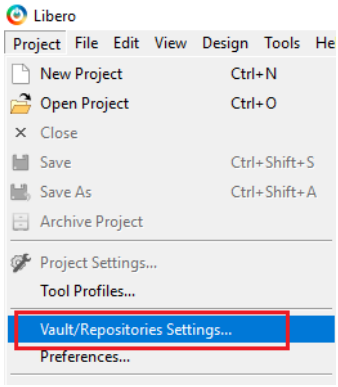
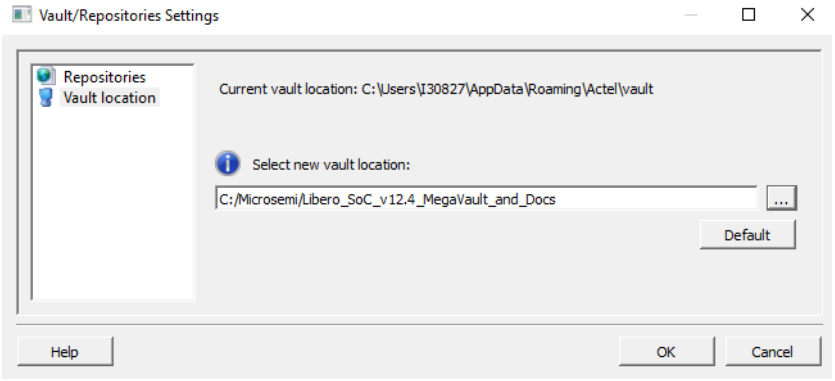


Figure 2 • **Browse Vault Location**



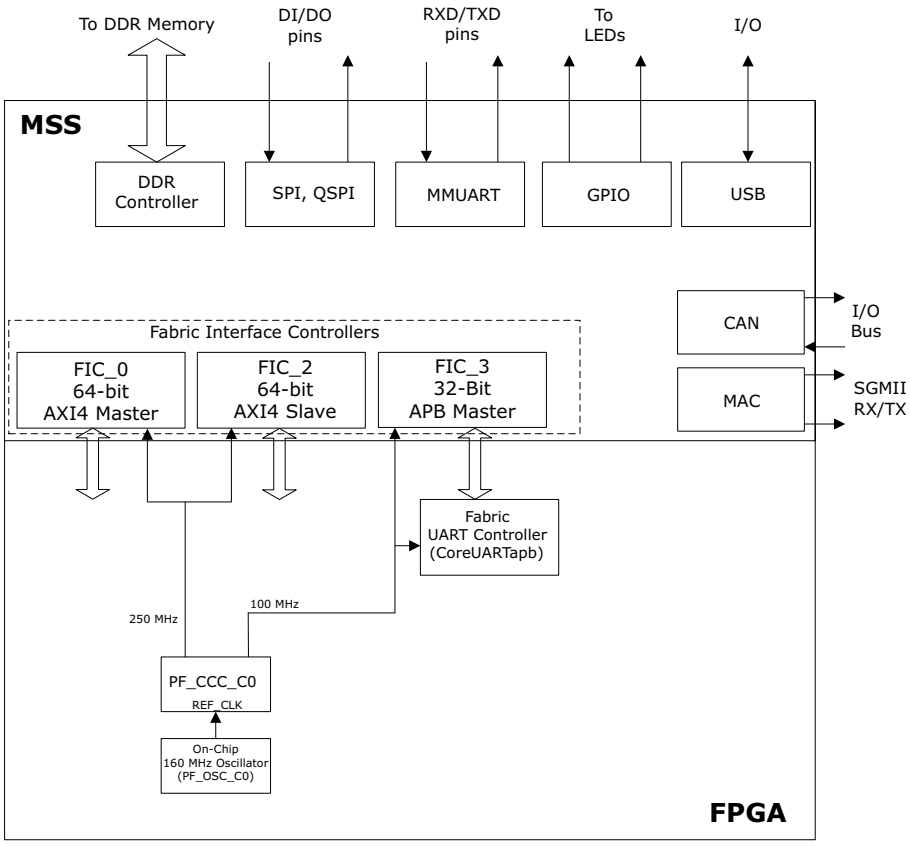
Note: Download the latest version of the MegaVault from the Libero SoC page.

1.4 Block Diagram

The MSS includes a wide range of peripherals and in the reference design, the MSS is configured by enabling DDR controller, USB, CAN, QSPI, MAC, SPI, MMUART, GPIO, and three FICs (FIC_0, FIC_2, and FIC_3). Figure 3 shows the high-level overview of the PolarFire SoC MSS design.

For more information about the PolarFire SoC MSS and its components, see *UG0880: PolarFire SoC FPGA MSS User Guide*.

Figure 3 • **Block Diagram**



1.5 Design Description

The MSS design contains the following IP blocks:

- PFSOC_MSS SgCore IP (hard IP)
- CORERESET_PF IP (soft IP)
- PF_OSC IP (hard IP)
- PF_CCC IP (hard IP)
- COREAPB3 IP (soft IP)
- CoreUARTapb IP (soft IP)

For more details regarding the IP blocks used and their configuration, open the design files provided along with the reference design.

1.5.1 I/O Ports

Table 2 lists the important I/Os in the reference design.

Table 2 • **I/O Ports**

Port Name	Direction	Description
EXT_RST_N	Input	Active-low user reset
REFCLK REFCLK_N	Input	Reference clock input received from the dedicated I/O from Bank 5 (REFCLK)
RX TX	Input Output	UART interface to the Host PC from the FPGA UART interface to the FPGA from the Host PC
SPI_1_DI SPI_1_DO	Input Output	SPI_1 Data In interface to the MSS SPI_1 Data Out interface from the MSS
MAC_0_MDC MAC_0_MDIO	Output Inout	Management interface to/from Ethernet PHY
MMUART_1_RXD MMUART_1_TXD	Input Output	MMUART_1 interface to the Host PC from the MSS MMUART_1 interface to the MSS from the Host PC
QSPI_DATA_F2M[3:0] QSPI_DATA_M2F[3:0]	Input output	QSPI Data In from fabric SPI Controller QSPI Data Out to fabric SPI Controller
CAN_1_RXBUS CAN_1_TXBUS	Input Output	CAN_1 input data CAN_1 output data
USB_CLK USB_DIR USB_NXT USB_DATA7:0	Input Input Input Inout	USB clock USB data to host PC Throttle signal from the USB PHY indicating next data USB data bus
GPIO_0_13_IN GPIO_1_22_IN GPIO_1_12_IN	Input Input Input	User input User input User input
CA[5:0] CK CKE CK_N CS DM[3:0] DQ[31:0] DQS[3:0] DQS_N[3:0] ODT	To DDR memory	SDRAM signals

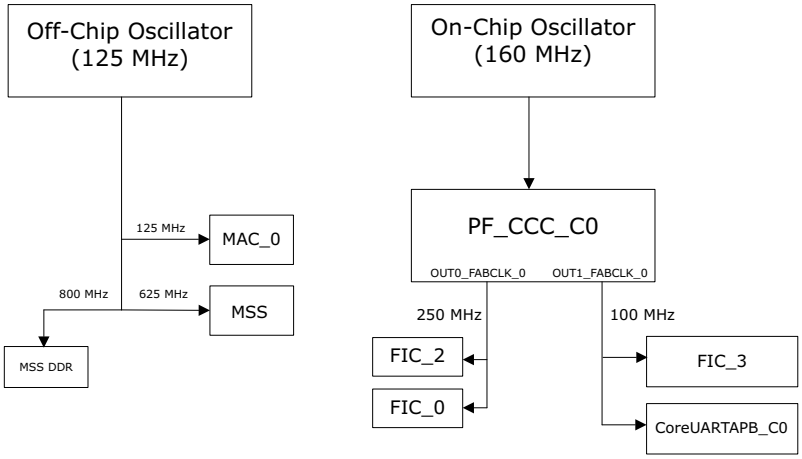
1.5.2 Clocking Structure

There are two clock sources in the design— an off-chip 125 MHz oscillator and an on-chip 160 MHz oscillator. The off-chip 125 MHz oscillator is used to source the reference clock for the MSS DDR (800 MHz), CPU Core Complex (600 MHz), and MAC_0. The on-chip 160 MHz clock source is used to source the reference clock for CoreUARTapb_C0 and FIC_3. OUT0_FABCLK0 (250 MHz) clock of the PF_CCC_C0 drives the FIC_2 clock.

The 625 MHz reference clock is further internally divided to source the AXI (312 MHz), AHB/APB (156 MHz), and eMMC clock (200 MHz) in MSS.

Figure 4 shows the clocking structure of the design.

Figure 4 • **Clocking Structure**

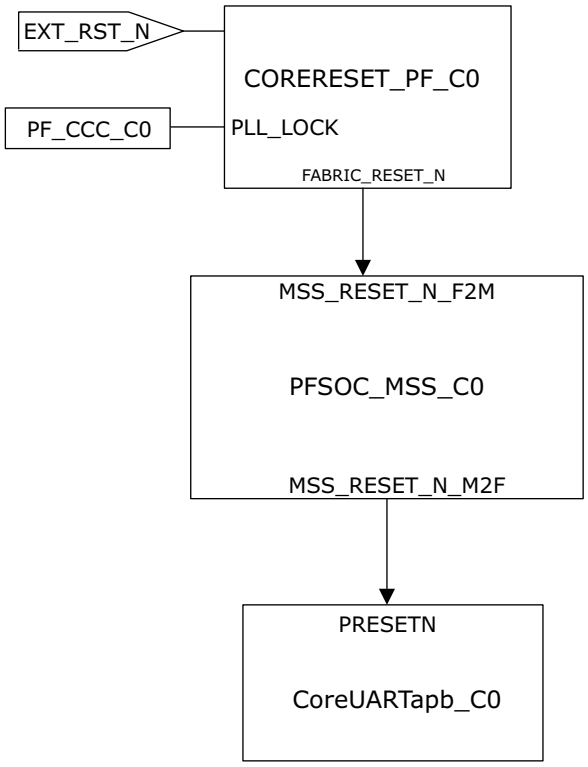


1.5.3 Reset Structure

The MSS reset is controlled by the FABRIC_RESET_N signal, which is asserted by ANDing the EXT_RST_N and the PLL_LOCK signal using the CORERESET_PF_C0 IP. After reset, MSS sends the MSS_RESET_N_M2F reset signal to the fabric to release the CoreUARTapb_C0 IP out of reset.

Figure 5 shows the reset structure of the design.

Figure 5 • **Reset Structure**



1.5.4 IP Configurations

This section describes each IP block of the MSS design.

1.5.4.1 PFSOC_MSS SgCore IP

This IP instantiates the MSS as PFSOC_MSS_C0_0. The MSS is configured with DDR Controller, FIC0, FIC2, FIC3, GPIO, SPI, and MMUART.

The MSS Configurator is used to enable the following configurations:

- Clocks Configuration
- FIC Configuration
- I/O Configuration
- DDR Configuration
- Boot Mode

1.5.4.1.1 Clocks Configuration

The MSS Configurator -> Clocks tab is used to configure the following:

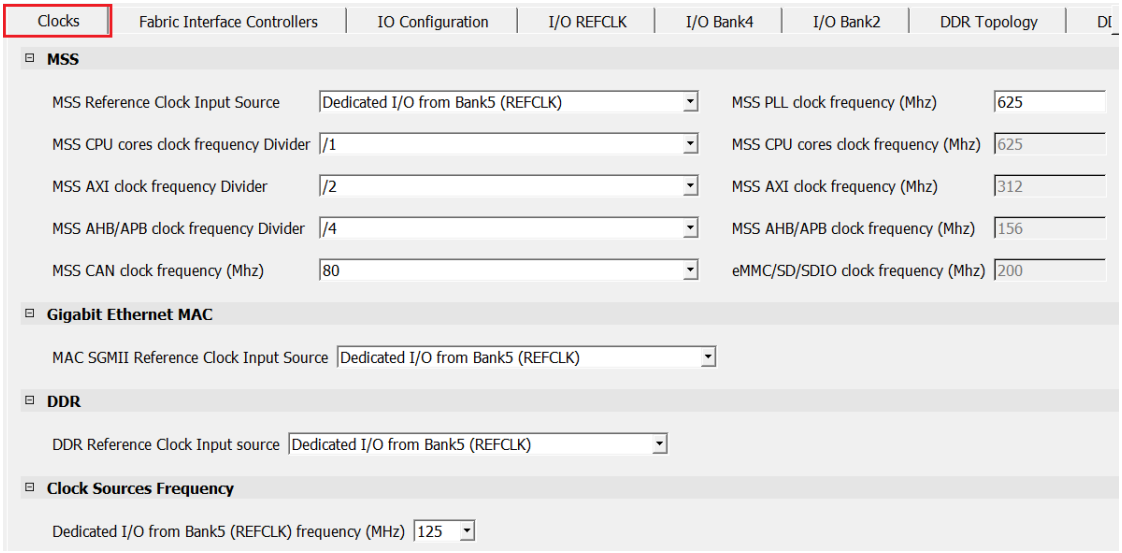
- **MSS Reference Clock Input Source**—This is set to dedicated I/Os from Bank 5 (REFCLK) to source the reference clock for the MSS PLL.
- **MSS PLL Clock Frequency**—This is set to 625 MHz. The CPU cores can operate at a maximum frequency of 625 MHz.
- **MSS CPU cores clock frequency Divider**—This is set to /1 divider.
- **MSS AXI clock frequency Divider**—This is set to /2 divider for a 312 MHz AXI clock frequency.
- **MSS AHB/APB clock frequency Divider**—This is set to /4 divider for a 156 MHz AHB/APB clock frequency.
- **MSS CAN clock frequency**—This is set to 80 MHz. This option is exposed only when CAN is selected in the Configurator.

- **eMMC/SD/SDIO clock frequency (MHz)—**This is set to 200 MHz. This option is exposed only when eMMC.SD/SDIO is selected in the Configurator.
- **DDR Reference Clock Input Source**—This is set to dedicated I/Os from Bank 5 (REFCLK) to source the DDR reference clock.
- **MAC SGMII Reference Clock Input Source**—This is set to Dedicated I/Os from BANK 5 (REFCLK) to source the SGMII reference clock. This option is exposed only when MAC is selected in the Configurator.
- **Clock Sources Frequency**—This is set to 125 MHz to source from a 125 MHz off-chip oscillator.

Note: The **DDR Reference Clock Input Source** option is displayed only when the DDR type is selected from the **DDR Topology** tab.

Figure 6 shows the clocks configuration used in the reference design.

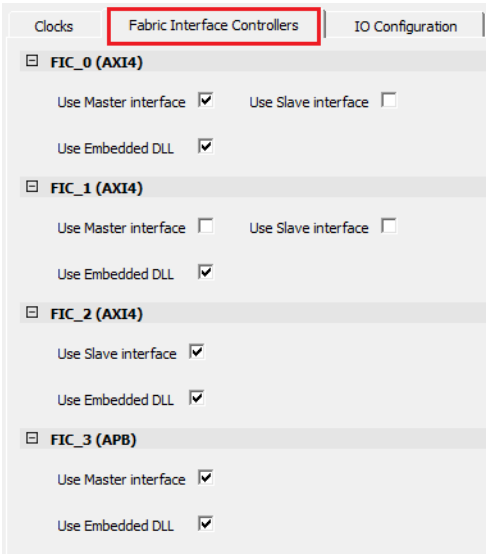
Figure 6 • **Clocks Configuration**



1.5.4.1.2 FIC Configuration

MSS includes four FICs for data transfer between MSS and the fabric. The MSS Configurator -> Fabric Interface Controllers tab is used to configure the FICs. FIC_0 and FIC_1 have both master and slave interfaces to/from the FPGA fabric. In the reference design, FIC_0 (master interface), FIC_2 (slave interface), and FIC_3 (master interface) are used as shown in Figure 7.

Figure 7 • **FIC Configuration**



Note: When a master interface is enabled for a FIC, that master interface must be connected to a slave in the fabric. When a slave interface is enabled for a FIC, that slave interface must be connected to a master in the fabric.

1.5.4.1.3 I/O Configuration

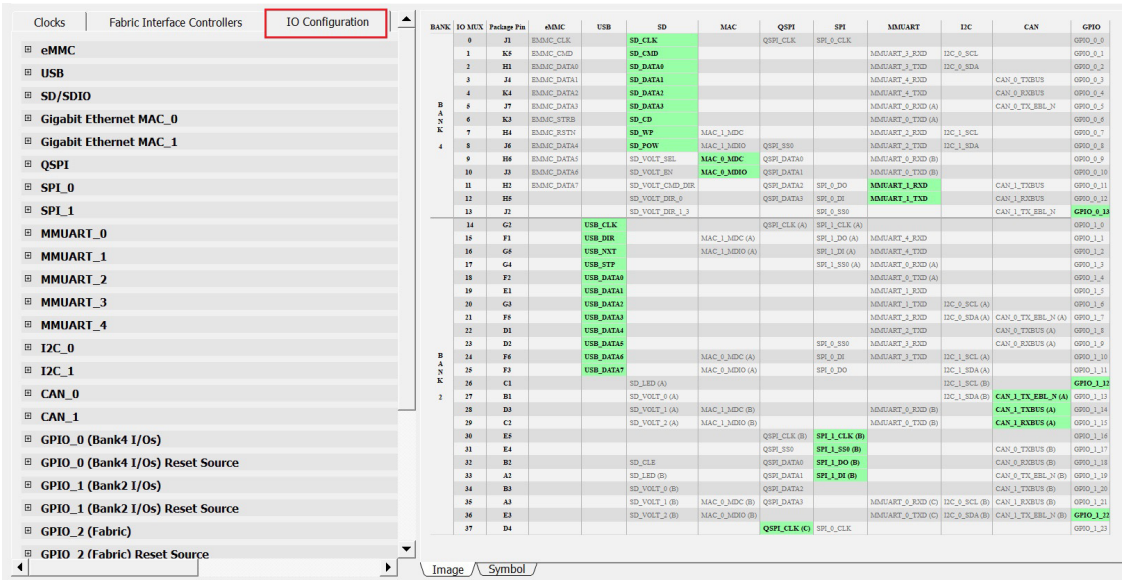
MSS supports a wide range of peripherals. The MSS Configurator -> IO Configuration tab is used to select the peripherals and assign the I/Os for those peripherals. By default, all peripherals are marked as unused. To include peripherals that are required in the design, use the drop-down corresponding to each peripheral and assign MSS I/Os or fabric I/Os for that peripheral.

The following peripherals are dedicated and cannot be assigned to fabric I/Os:

- USB peripherals are dedicated in Bank 4
- eMMC peripherals are dedicated in Bank 2
- SD/SDIO peripherals are dedicated in Bank 2

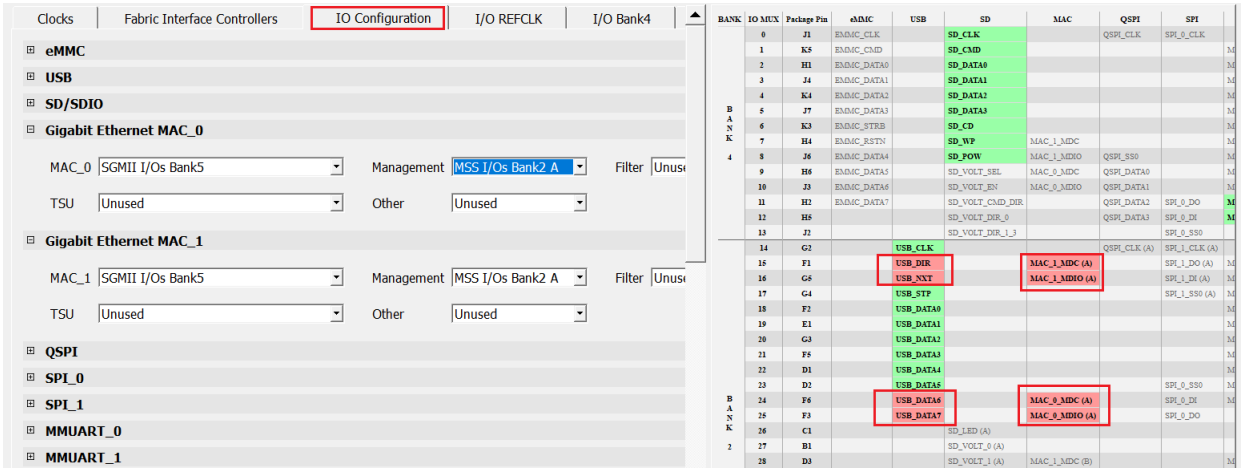
The MSS uses MMUART_1, SPI_1, CAN_1, GPIO_0_13 (Bank 4), GPIO_1_12 and GPIO_1_22 (Bank 2), QSPI, USB, and MAC_0 as shown in Figure 8.

Figure 8 • **IO Configuration**



Note: If the I/Os for a peripheral are selected in a Bank, selecting the same I/Os for another peripheral from the same Bank is not allowed. The user is warned by red highlights on overlapping I/Os as shown in Figure 9.

Figure 9 • **IO Overlap Warning**



1.5.4.1.4 DDR Configuration

The reference design configures the DDR controller for LPDDR4 memory. Figure 10, Figure 11, Figure 12, and Figure 13 show the DDR Topology, memory initialization, memory timing, and address ordering used.

Figure 10 • **DDR Topology**

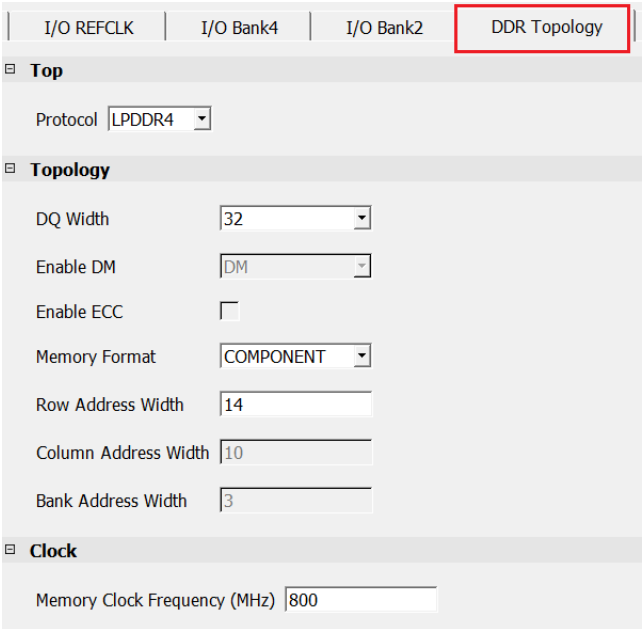


Figure 11 • **DDR Memory Initialization**

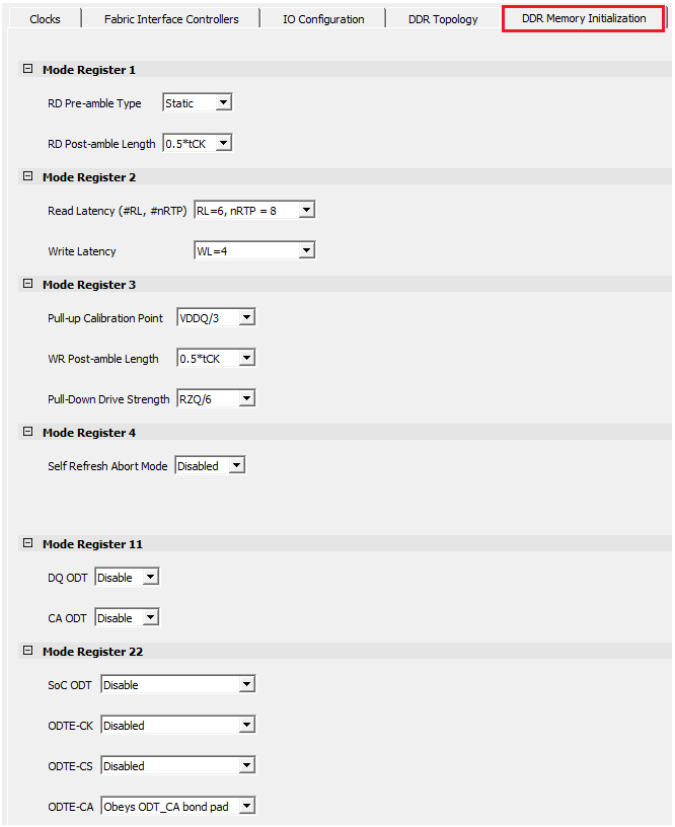


Figure 12 • **DDR Memory Timing**

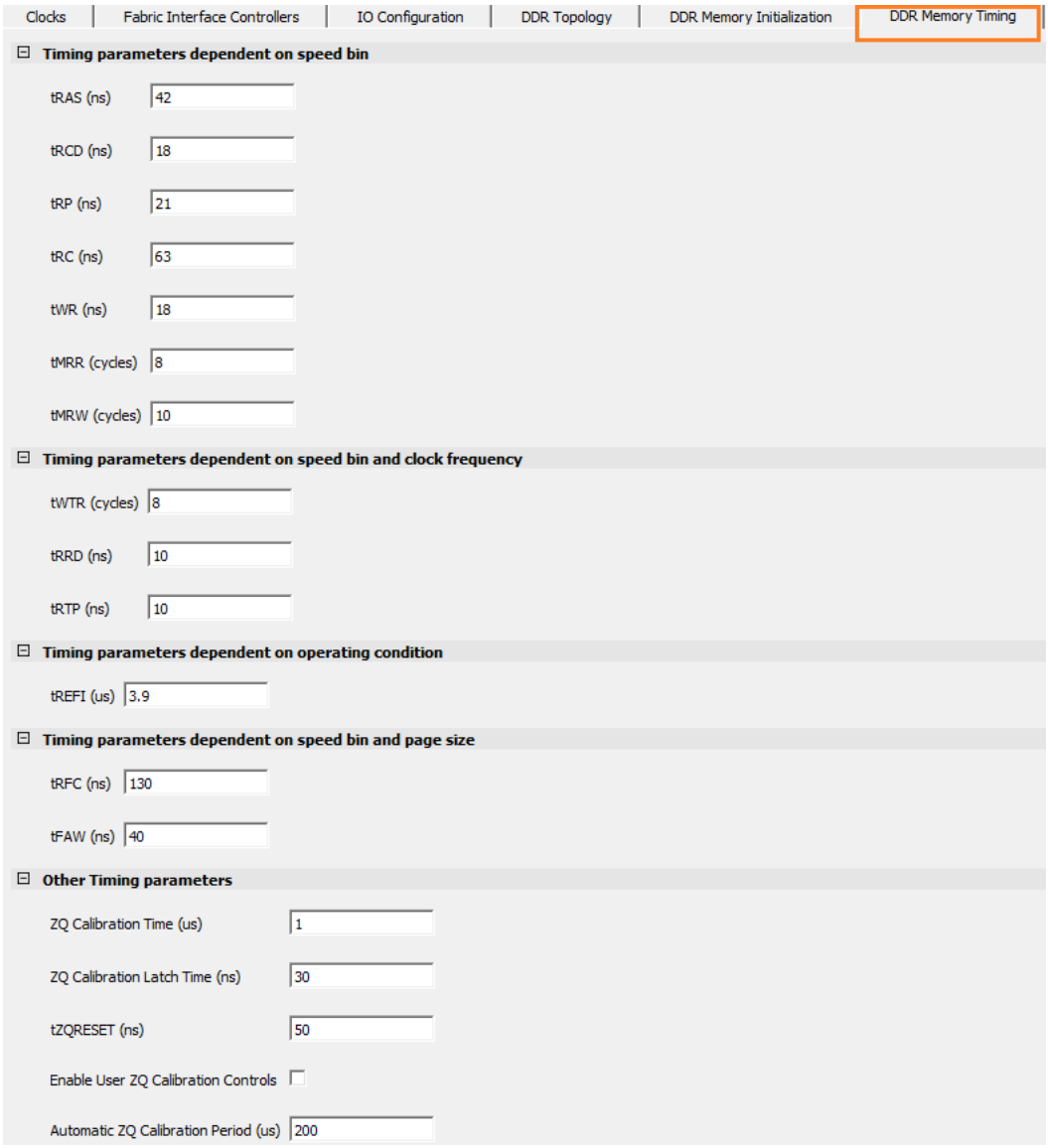
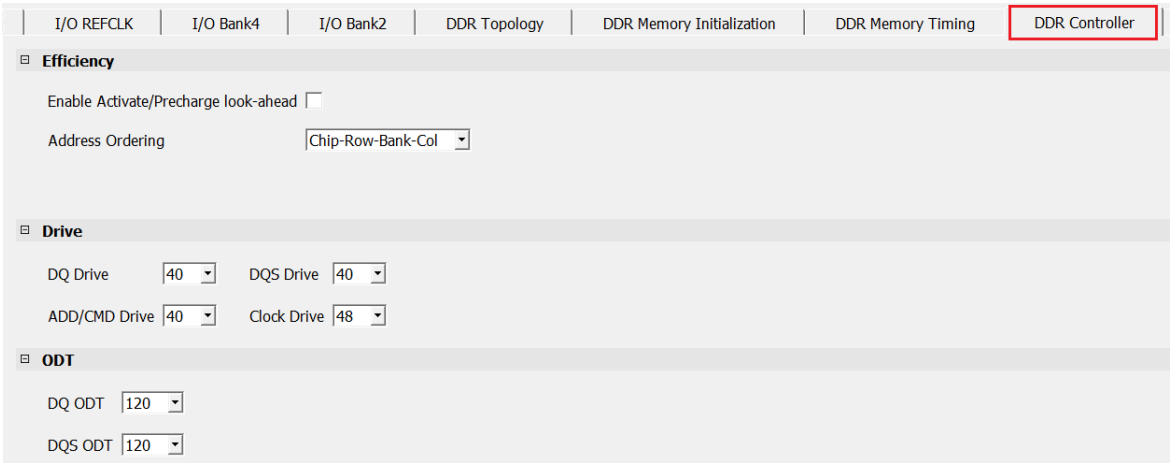


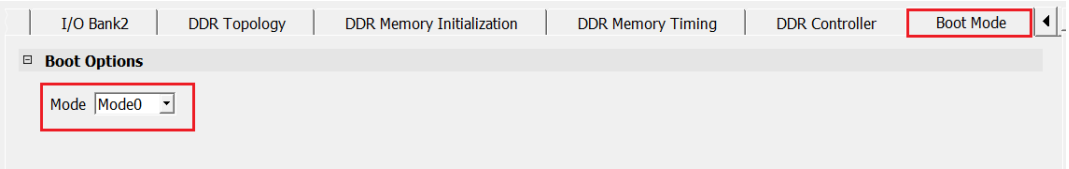
Figure 13 • **DDR Controller**



1.5.4.1.5 Boot Mode

Boot Mode option is set to **Mode0**. **Mode0** is non-secure boot mode. In this mode, the MSS Processor E51 monitor core boots from the boot code stored in the eNVM (Boot Flash).

Figure 14 • **Boot Mode**



For more information about MSS boot modes, see *UG0881: PolarFire SoC FPGA Booting And Configuration User Guide*.

1.5.4.2 CORERESET_PF IP

This soft IP is instantiated as CORERESET_PF_C0_0 and is used to reset the MSS block. CORERESET_PF_C0_0 IP resets the MSS using the FABRIC_RESET_N signal. The FABRIC_RESET_N signal is asserted when the EXT_RST_N and PLL_LOCK signals are asserted.

1.5.4.3 PF_OSC IP

This hard IP is instantiated as PF_OSC_C0_0 (on-chip 160 MHz RC oscillator) and feeds the reference clock to the PF_CCC_C0_0. This clock source from the fabric is used to generate two clocks to drive the FIC DLLs.

1.5.4.4 PF_CCC IP

This hard IP is instantiated as PF_CCC_C0_0 and generates the following clocks using the 160 MHz input reference from PF_OSC_C0_0:

- OUT0_FABCLK_0 (250 MHz): Drives the FIC_2_ACLK clock signal. This clock is used by a master in the fabric for DDR operations, and the FIC_0_ACLK clock signal used by a slave in the fabric for general data transfers.
- OUT1_FABCLK_0 (100 MHz): Drives the FIC_3_PCLK clock signal. This clock is used by a APB slave in the fabric.

1.5.4.5 COREAPB3 IP

This soft IP is used as a bridge to connect to the fabric UART controller (CoreUARTapb_C0_0).

1.5.4.5.1 CoreUARTapb IP

This soft IP is instantiated as CoreUARTapb_C0_0 and controls the UART interface between the SoC FPGA and a host.

2 Libero Design Flow

This chapter describes the Libero design flow for running this demo design, which includes:

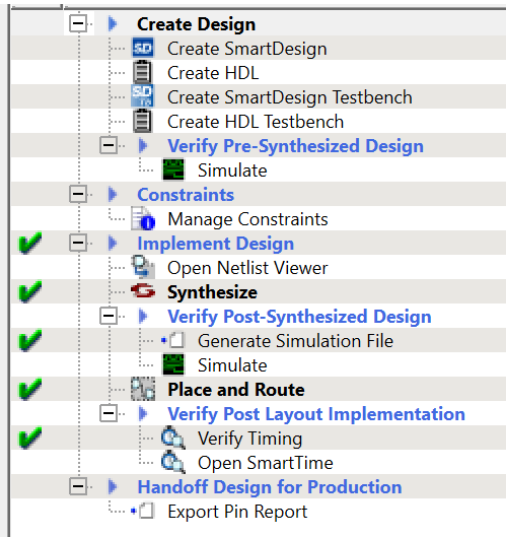
- Synthesize
- Place and Route
- Verify Timing

Note: The reference design will be targeted for hardware in future releases. This release of Libero SoC supports up to timing verification. To view the reference design, launch Libero SoC, use the Open Project option, and select the `hw_project.prjx` file from the following design flies location:

```
<$Download_Directory>\mpfs_ac489_df\hw_project
```

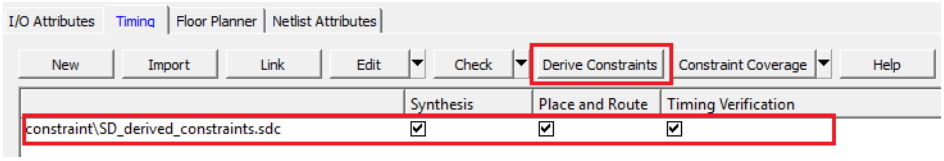
The following figure shows these options in the Design Flow tab.

Figure 11 • **Libero Design Flow**



Note: The timing constraints of the reference design are generated using the Derive Constraints option from **Design Flow** -> **Manage Constraints** -> **Timing** tab. The timing constraints are generated in the SD_derived_constraints.sdc file and it is selected for Synthesis, Place and Route, and Timing Verification as shown in Figure 12.

Figure 12 • **Derived Constraints**



2.1 Synthesize

To synthesize the design:

1. On the Design Flow tab, double-click Synthesize.
   When the synthesis is successful, a green tick mark appears next to Synthesize, as shown in Figure 11.
2. Right-click Synthesize and select View Report to view the synthesis report and log files in the Reports tab.

2.2 Place and Route

To place and route the design:

1. On the Design Flow tab, double-click Place and Route.
   When place and route is successful, a green tick mark appears next to Place and Route, as shown Figure 11.
2. Right-click Place and Route and select View Report to view the place and route report and the log files in the Reports tab.

2.3 Verify Timing

On the Design Flow tab, double-click Verify Timing.

1. When the design successfully meets the timing requirements, a green tick mark appears next to Verify Timing, as shown in Figure 11.
2. Right-click Verify Timing and select View Report to view the verify timing report and log files in the Reports tab.

3 Revision History

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

3.1 Revision 2.0

Updated for Libero SoC 12.4.

3.2 Revision 1.0

The first publication of this document.