PolarFire FPGA Low Power Application Note

AN4661

Introduction (Ask a Question)

Microchip PolarFire® FPGAs are designed to meet the demand for low-power applications. PolarFire devices exhibit lower-power consumption in static and dynamic modes. PolarFire devices offer several low-power features for the FPGA fabric, Fabric Clock Conditioning Circuitry (CCC), Transceiver, DDR memory, and other hard and soft IP blocks.

Table of Contents

1. PolarFire FPGA Low-Power (Ask a Question)

This application note provides an overview of low-power PolarFire devices and techniques for low-power design implementation using PolarFire.

1.1 Lower Power Options (Ask a Question)

The following low-power options are available in PolarFire devices:

- Transceiver Low-Power
- DDR Controller Low-Power
- PLL Low-Power
- RAM Blocks Low-Power

1.1.1 Transceiver Low-Power (Ask a Question)

The Transceiver Quad and External PLLs have several registers that power down different parts of the circuitry. This allows enabling only the circuits that are used in the design to reduce the transceiver power.

The following table lists the registers that are required to disable parts of the Transceiver quad circuitry and external PLL circuitry.

Table 1-1. Transceiver Quad Circuitry

Register	Field Name	Value for Low-Power	Description
DES_RSTPD	RXPD	1	Power down the Rx circuitry
	PDDFE	1	Power down the DFE circuitry
	PDEM	1	Power down the Eye Monitor circuitry
	RCVEN	1	Disable RX receiver
DES_PKDET	RXPKDETEN	0	Disable the RX peak detector
DES_IN_TERM	RXTEN	0	Disable RX termination resistor
DES_RXPLL_DIV	CDR_GAIN	0	Set CDR Gain to 0
DES_DFE_CAL_CTRL_0	EN_OFFSET_CAL	0	Disable offset calibration
SER_RSTPD	TXPD	1	Power down the TX circuitry
SER_TERM_CTRL	TXTEN	0	Disable the TX termination resistor
SERDES_RTT	RTT_CURRENT_PROG	0	Disable RTT trim circuitry
TXPLL_CTRL	TXPLL_AUXDIVPD	1	Disable the AUX clock output
	TXPLL_VBGREF_SEL	0	Disable the TX voltage regulator
	TXPLL_PD	1	Disable TX PLL
TXPLL_CLKBUF	TXPLL_DUALCLK1_MODE	0	Disable the refclkp input buffer
	TXPLL_DUALCLK0_MODE	0	Disable the refclkn input buffer
	TXPLL_DUALCLK1_ENTERM	0	Disable the refclkp input buffer single ended termination
	TXPLL_DUALCLK0_ENTERM	0	Disable the refclkn input buffer single ended termination
	EXTPLL_CLKBUF_EN_RDIFF	0	Disable 100Ω differential termination between refclkp and refclkn

Table 1-2. External PLL Circuitry

Register	Field Name	Value for Low-Power	Description
EXTPLL_CTRL	EXTPLL_PD	1	Power down the External PLL
	EXTPLL_VBGREF_SEL	0	Disable the TX voltage regulator
EXTPLL_CLKBUF	EXTPLL_DUALCLK1_MODE	0	Disable the refclkp input buffer
	EXTPLL_DUALCLK0_MODE	0	Disable the refclkn input buffer
	EXTPLL_DUALCLK1_ENTERM	0	Disable the refclkp input buffer single ended termination
	EXTPLL_DUALCLK0_ENTERM	0	Disable the refclkn input buffer single ended termination
	EXTPLL_CLKBUF_EN_APAD	0	Disable analog connection to refclkn pad
	EXTPLL_CLKBUF_EN_RDIFF	0	Disable 100Ω differential termination between refclkp and refclkn

Important: For more information about register configuration, see AN4592: PolarFire FPGA Dynamic Reconfiguration Interface Application Note and PolarFire Device Register Map

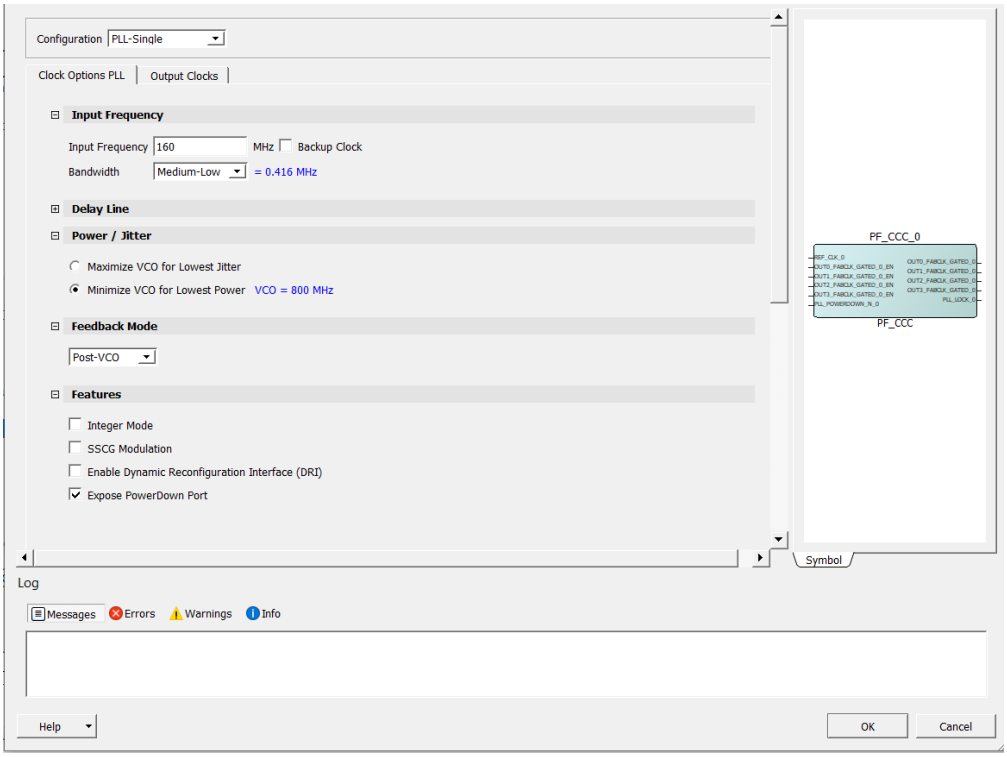
1.1.2 DDR Controller Low-Power (Ask a Question)

The PolarFire FPGA DDR subsystem supports low-power operation which puts the DDR memory in low-power mode and issues refresh commands automatically to retain data. For more information about DDR low-power options, see PolarFire Family Memory Controller User Guide.

1.1.3 PLL Low-Power (Ask a Question)

PLL can be set to Low-Power by selecting **Minimize Power**, see the following figure.

Figure 1-1. Setting to Minimize Power in CCC Configurator



1.1.4 RAM Blocks Low-Power (Ask a Question)

The dual-port LSRAM, two-port LSRAM, and micro-SRAM can also be set for minimum power consumption. The following figures show how the dual-port LSRAM, two-port LSRAM, and micro-SRAM are set to low-power.

Figure 1-2. Low-Power Setting for Dual-Port LSRAM

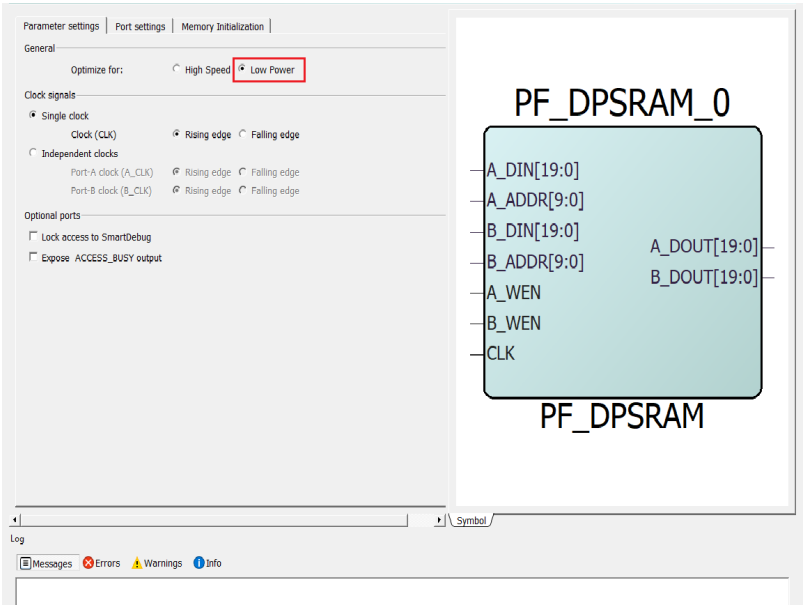


Figure 1-3. Low-Power Setting for Two-Port LSRAM

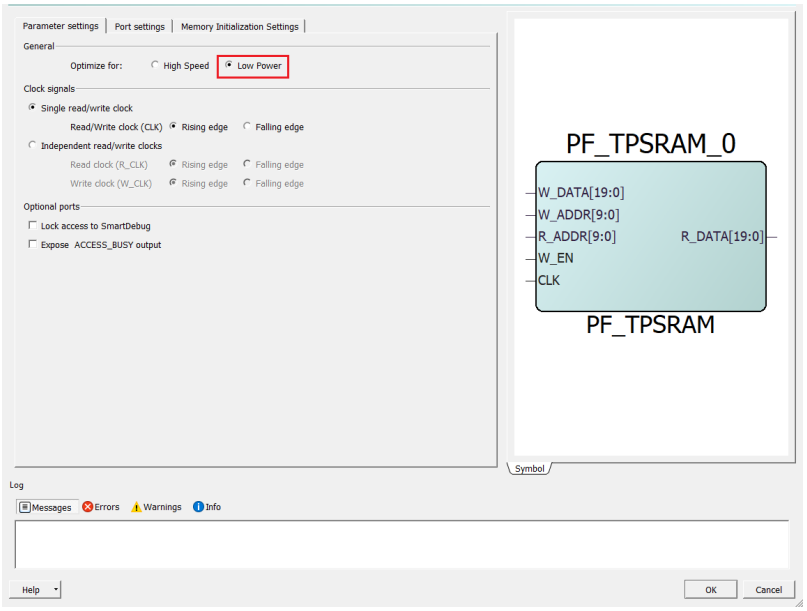
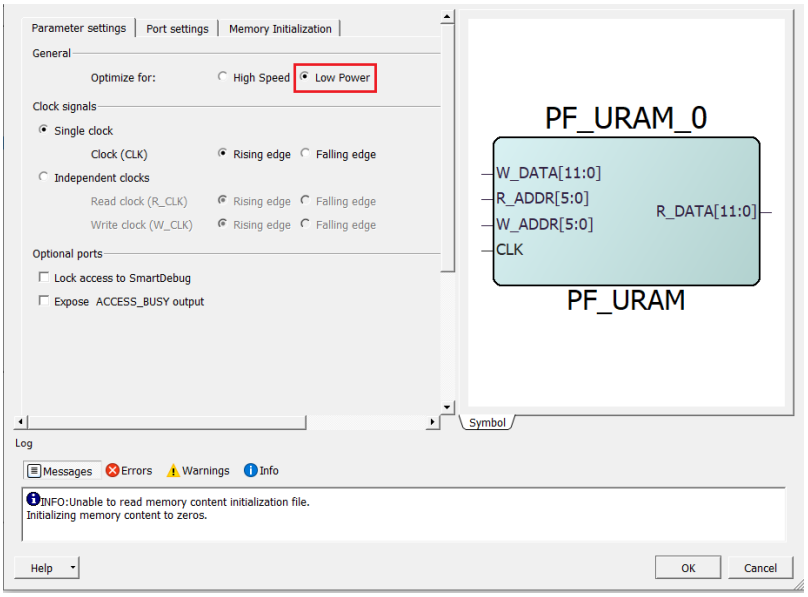


Figure 1-4. Low-Power Setting for Micro-SRAM



1.2 Design Requirements (Ask a Question)

The following table lists the hardware and software requirements to run the demo.

Table 1-3. Design Requirements

Requirement	Version
Hardware	
MPF300T-EVAL-KIT	Revision D or later
Host PC	Windows® 10
Software	
Libero® SoC	Refer to the `readme.txt` file provided in the design files for the software versions used with this reference design.
FlashPro Express	

Important: Libero® SmartDesign and configuration screen shots shown in this guide are for illustration purpose only. Open the Libero design to see the latest updates.

1.3 Prerequisites (Ask a Question)

Before you begin, perform the following steps:

1. Download the design files from www.microchip.com/en-us/application-notes/AN4661.
2. Download and install Libero SoC on the host PC from Libero SoC Documentation.

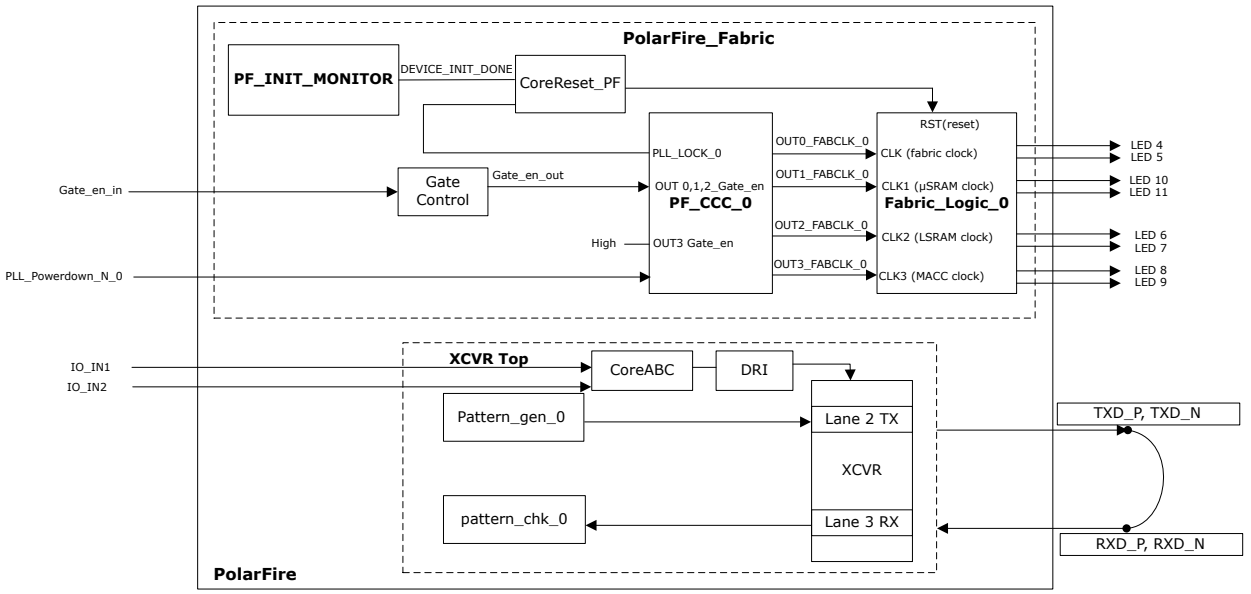
Important: The latest versions of ModelSim® and Synplify Pro® are included in the Libero SoC installation package.

3. Download the Microchip Power Monitor application from Power Monitor Application.

1.4 Demo Design (Ask a Question)

The low-power design is illustrated in the following block diagram.

Figure 1-5. Block Diagram



The PolarFire Fabric block instantiates a counter logic along with 500 µSRAM, 500 LSRAM, and 500 Mathblocks which utilizes 70% of four input LUT and DFF.

In the XCVR_Top SmartDesign, transceiver block is instantiated and looped back internally on the PolarFire Evaluation kit from Lane 2 to Lane 3. The CoreABC and DRI blocks enable the user to dynamically reconfigure the XCVR registers.

The demo design flow is described as follows:

- The DEVICE_INIT_DONE signal of the PF_INIT_MONITOR block is asserted after the device is initialized.
- CoreReset_PF IP core is used to control reset signal of the Fabric_Logic_0 and XCVR_Top blocks.
- The PF_CCC_0 block provides the following fabric clocks:
  - CLK: 100 MHz clock for the fabric
  - CLK1: 100 MHz clock for the µSRAM blocks
  - CLK2: 100 MHz clock for the LSRAM blocks
  - CLK3: 100 MHz clock for Mathblocks
- These separate clocks are provided in the design to gate clocks to each fabric block, if required. The transceiver (PF_XCVR) block instantiates the transceiver in 8b/10b mode. This block receives clock from the REF_CLK signal of PF_XCVR_REF_CLK_0. The PF_TX_PLL_0 block also derives its reference clock from REF_CLK of PF_XCVR_REF_CLK_0.
- PLL_Powerdown port is used to enable the PLL Powerdown option.
- Gate_en_in signal is fed to Gate Control block and output of the Gate Control Block is connected to OUT0, 1, and 2_FABCLK_GATED_0_EN.
- OUT3_FABCLK_GATED_0_EN is connected High.
- The TX and RX lanes of the transceiver are looped back using on board PCB loopback.
- The pattern_gen_0 block is implemented to send data to the transceiver block. The pattern_chk_0 block is implemented to check errors in the data received by the transceiver block.

- DRI interface is used to configure the XCVR/TX_PLL in ON and OFF mode.

Two programming job files are provided with this demo.

- With low-power options (`PF_Demo_Low_power.job`).
- Without using low-power options (`PF_Demo_Normal.job`)

1.4.1 Low-Power Option (Ask a Question)

This section describes the different low-power options used in the demo design.

1.4.1.1 PLL Power Down (Ask a Question)

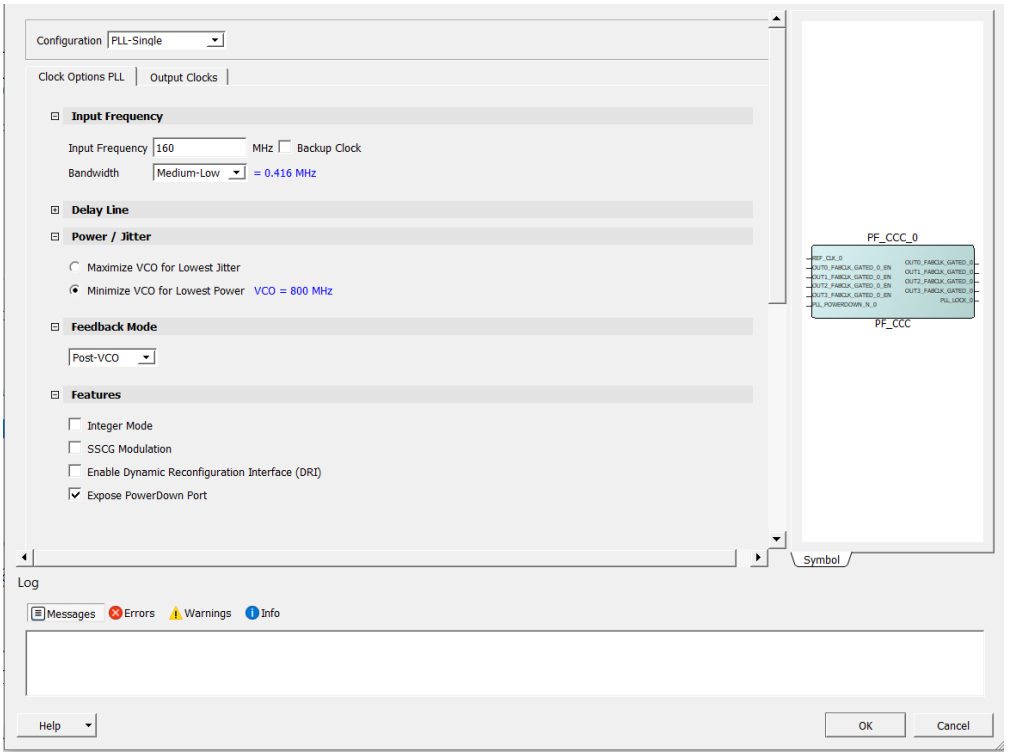
The Active low-power down input (PLL_POWERDOWN_N) can be exposed using CCC configurator. The PLL_POWERDOWN_N is an asynchronous signal, which can be used to reset the PLL from the FPGA fabric, which forces the PLL to its lowest power state and the clock outputs are driven Low.

In the design, this port is exposed and connected to DIP1 switch.

- DIP1-0: Power Down Mode
- DIP1-1: Normal Mode

The following figure shows the PF_CCC configurator settings for the demo design.

Figure 1-6. PLL Power Down—CCC Configurator Settings



Important: For Normal demo design, select **Maximize VCO for Lowest Jitter** in the CCC configurator.

1.4.1.2 Clock Gating (Ask a Question)

Clock gating is a popular technique used in many synchronous circuits for reducing dynamic power dissipation. Clock gating saves power by adding more logic to a circuit to prune the clock tree.

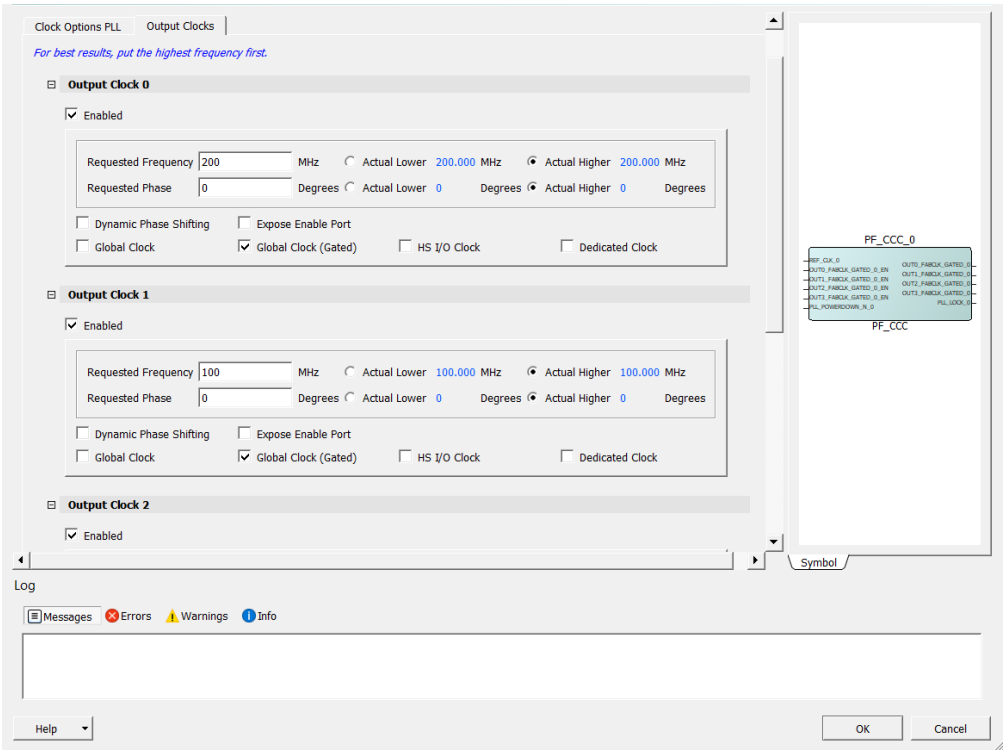
Pruning the clock disables portions of the circuitry, hence the flip-flops in them do not have to switch states. Gate enable/disable pins can be exposed by using the CCC configurator.

The design has a clock gating enable/disable switch, which is connected to DIP2 for OUTPUT0, 1, and 2.

- DIP2-1: Clock Gating Enabled (Clock is available)
- DIP2-0: Clock Gating Disabled (Clock is not available)

The following figure shows the PF_CCC configurator settings for the demo design.

Figure 1-7. Clock Gating—CCC Configurator Settings

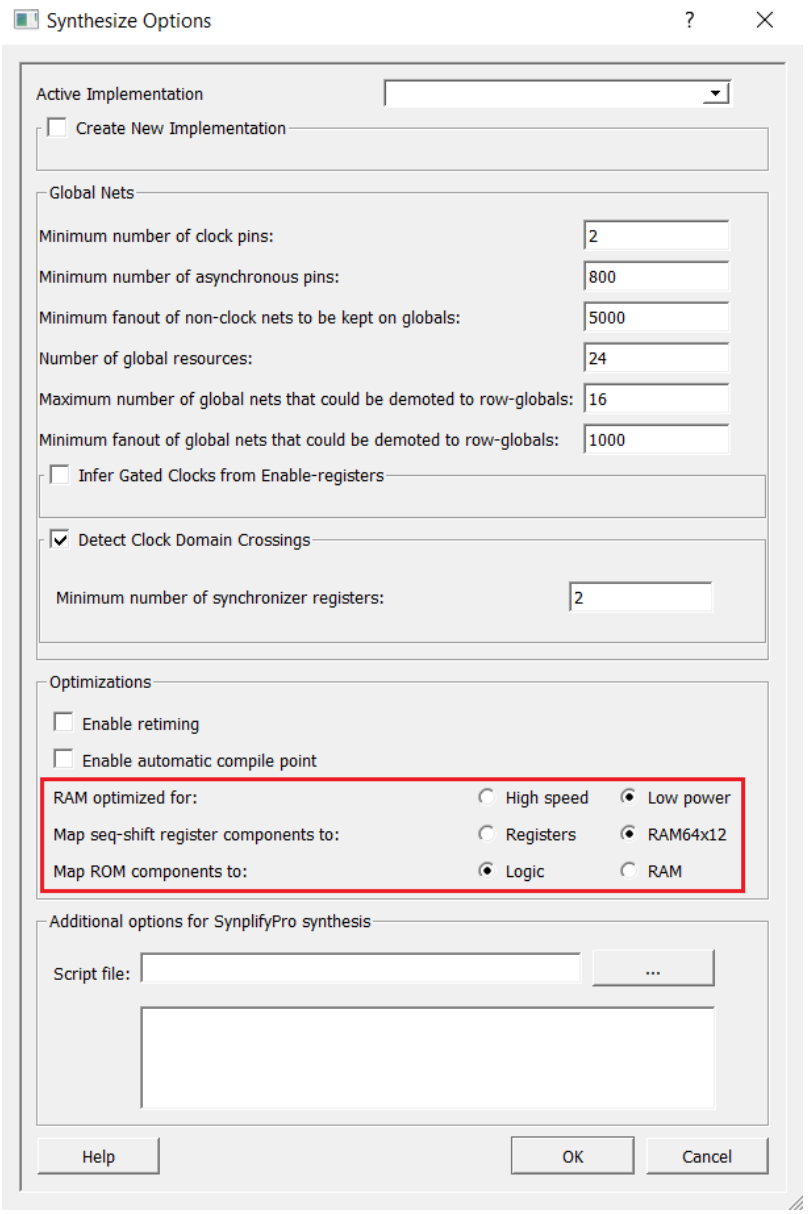


1.4.1.3 Synthesize and Place and Route (Ask a Question)

In Libero, Synthesize has RAM optimization option. RAMs can be optimized for the following two modes.

- **High speed** - RAM optimization is geared towards speed. The resulting synthesized design achieves better performance (higher speed) at the expense of more FPGA resources.
- **Low power** - RAM optimization is geared towards low-power. RAMs are inferred and configured to ensure the lowest power consumption.

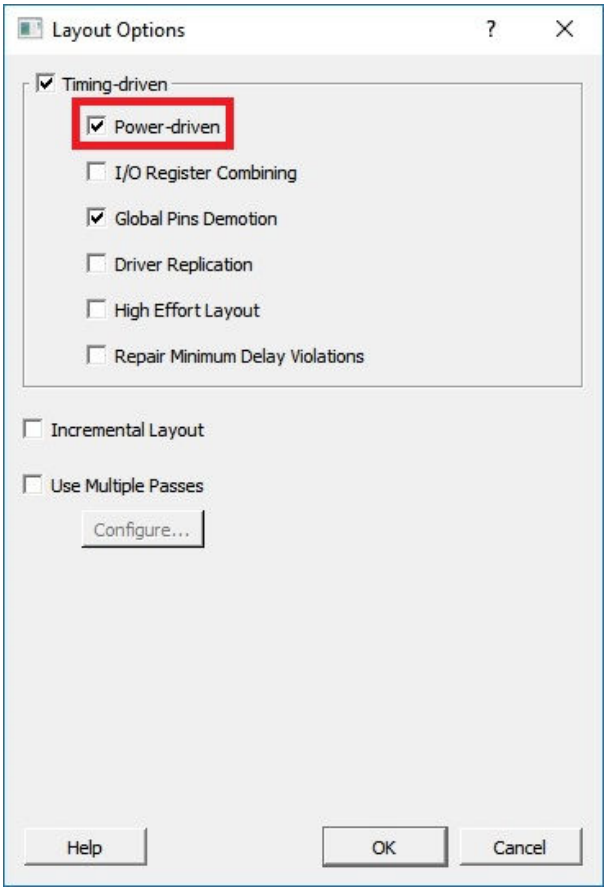
Figure 1-8. Synthesize Options



In Libero, the Place and Route has Power-driven option. Enable this option to run Power-Driven layout. The primary goal of Power-driven layout is to reduce dynamic power while maintaining timing constraints.

Important: For Normal demo design, in Synthesis option RAM is optimized for High speed, and in Place and route Power-driven option is disabled.

Figure 1-9. Layout Options



1.4.1.4 Transceiver (Ask a Question)

In this design, Transceiver can be dynamically switched ON and OFF using DRI interface. DIP-3 and DIP-4 switches are connected to inputs of CoreABC (IN_IN_0 and I0_IN_1). Based on selection, XCVR related register dynamically reconfigures.

Table 1-4. Power Down/Up XCVR

Mode	DIP4	DIP3
XCVR OFF	0	1
XCVR ON	1	0

The following figure shows the XCVR register settings for XCVR OFF mode.

Figure 1-10. Transceiver OFF—Register Settings

```
//-----------------------------------------------xcvr off and txpll off
$xcvr_off_txpll_off
// Assert PMA  Reset
    IOWRT 0x011
//SER_RSTPD
    APBWRT DAT 0 0x4078 0x00000007
    APBWRT DAT 0 0x8078 0x00000007
//DES_RSTPD
    APBWRT DAT 0 0x404C 0x0000002F
    APBWRT DAT 0 0x804C 0x0000002F

//LSRT Assert and deassert on Q2_lane2 and Q2_lane3
    IOWRT 0x001
    APBWRT DAT 0 0x4068 0x00000D0D
    APBWRT DAT 0 0x8068 0x00000D0D

//TX PLL Powerdown
    IOWRT 0x011
    APBWRT DAT 1 0x0008 0x0820010
    JUMP $done
```

The following figure shows the XCVR register settings for XCVR ON mode.

Figure 1-11. Transceiver ON—Mode Register Settings

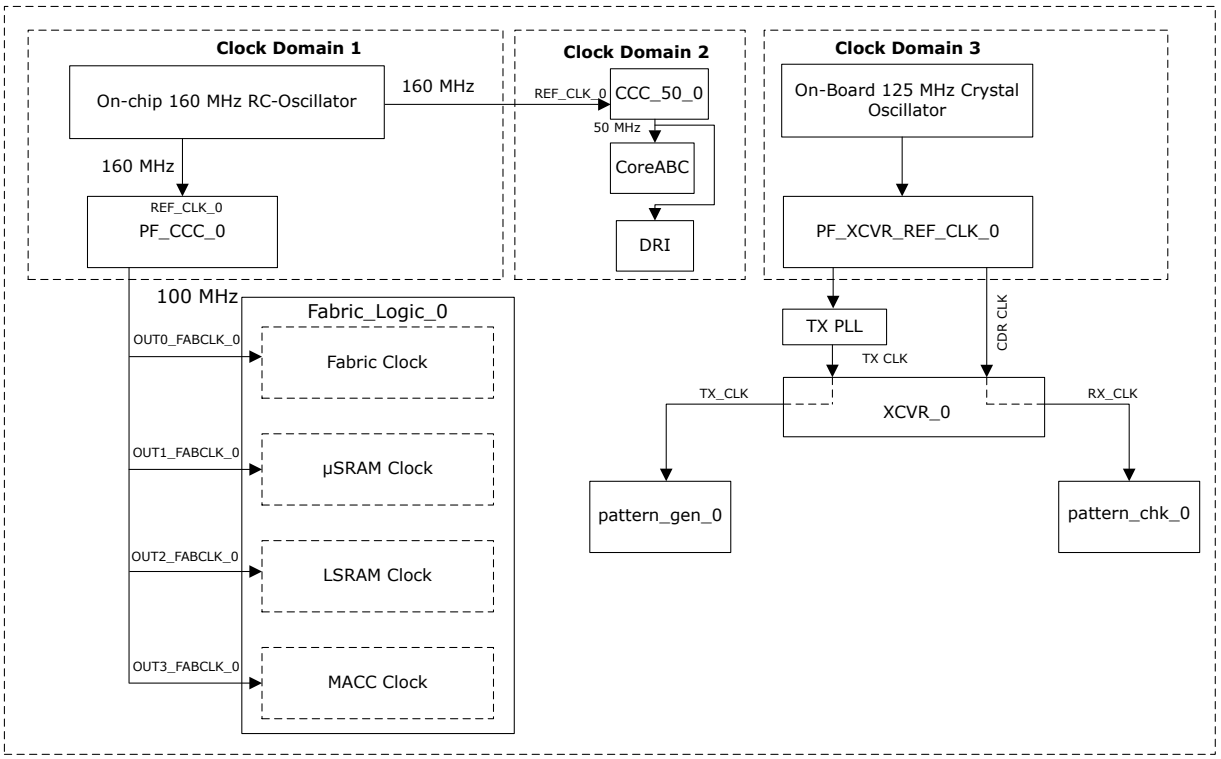
```
//-----------------------------------------------xcvr on and txpll on
$xcvr_on_txpll_on
// Assert PMA  Reset
    IOWRT 0x011
//SER_RSTPD
    APBWRT DAT 0 0x4078 0x00000003
    APBWRT DAT 0 0x8078 0x00000003
//DES_RSTPD
    APBWRT DAT 0 0x404C 0x00000032
    APBWRT DAT 0 0x804C 0x00000032
//  Power up XCVR and deassert PMA reset
//SER_RSTPD
    APBWRT DAT 0 0x4078 0x00000001
    APBWRT DAT 0 0x8078 0x00000001
//DES_RSTPD
    APBWRT DAT 0 0x404C 0x0000003C
    APBWRT DAT 0 0x804C 0x0000003C
//LSRT Assert and deassert on Q2_lane2 and Q2_lane3
    IOWRT 0x001
    APBWRT DAT 0 0x4068 0x00000D0D
    APBWRT DAT 0 0x8068 0x00000D0D
    APBWRT DAT 0 0x4068 0x00000404
    APBWRT DAT 0 0x8068 0x00000404
//TX PLL Power on
    IOWRT 0x011
    APBWRT DAT 1 0x0008 0x1800010
    JUMP $done
```

Important: For more information about register configuration, see AN4592: PolarFire FPGA Dynamic Reconfiguration Interface Application Note and PolarFire Device Register Map

1.5 Clocking Structure (Ask a Question)

The following figure shows the clocking structure implemented in the demo design.

Figure 1-12. Clocking Structure



1.6 Resource Utilization (Ask a Question)

The following tables list the resource utilization of the low-power and Normal demo designs after Synthesis and Place and Route. These values may vary for different Libero runs, settings, and seed values.

Table 1-5. Resource Utilization for Low-Power Demo

Type	Used	Total	Percentage
4LUT	186382	299544	62.22
DFF	180140	299544	60.14
I/O Register	0	510	0.00
Logic Element	189750	299544	63.35

Table 1-6. Resource Utilization for Normal Demo

Type	Used	Total	Percentage
4LUT	186182	299544	62.16
DFF	198484	299544	66.26
I/O Register	0	510	0.00
Logic Element	208219	299544	69.51

2. Running the Demo (Ask a Question)

Prerequisites for the procedure:

1. On the host PC, download and install the Microchip Power Monitor application from Power Monitor Application.
2. Ensure that the jumper settings on the board are same as listed in the following table:

Table 2-1. Jumper Settings

Jumper	Description
J18, J19, J20, J21, and J22	Close Pins 2 and 3 for Programming PolarFire® FPGA through FTDI.
J28	Close Pins 1 and 2 for programming through the on board FlashPro5.
J4	Close Pins 1 and 2 for manual power switching using switch SW3.
J12	Close Pins 3 and 4 for 2.5V

3. Connect the power supply cable to the J9 connector on the board.
4. Connect the USB cable from the Host PC to the J5 (FTDI port) on the board.
5. Power on the board using the SW3 slide switch.

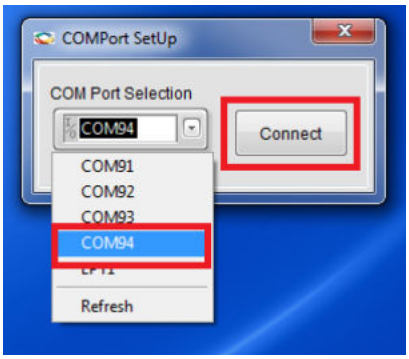
To run the demo, perform the following steps:

1. Ensure **DIP-3** and **DIP-4** are **ON** and **DIP-1** and **DIP-2** are **OFF**.

> **Important:** In Evaluation Kit, DIP switches are active-low.

2. To program the design without low-power options (`PF_Demo_Normal.job`) using FlashPro Express, see Appendix 1: Programming the Device Using FlashPro Express.
3. The LEDs {4, 5}, {6, 7}, {8, 9} and {10,11} blink at different rates. This indicates that the fabric components are in Active mode.
4. On the host PC desktop, click **Start** and then select **PowerMonitor**.
5. In the **COMPort SetUp** dialog box, select the highest COM port from the drop-down and click **Connect**, as shown in the following figure.

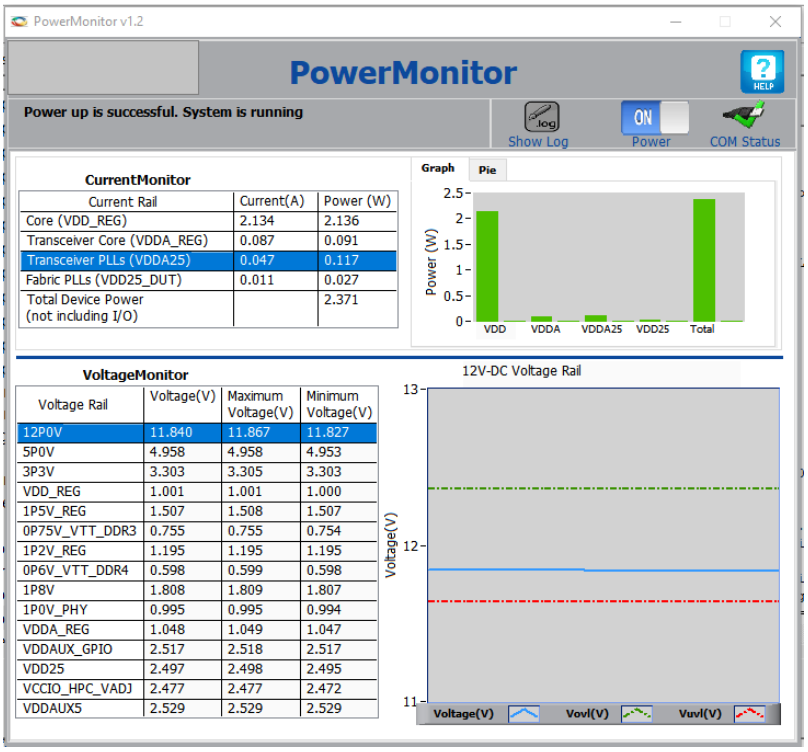
Figure 2-1. COMPort SetUp



The PowerMonitor application successfully connects to the board and starts displaying the Core Fabric (VDD) power, Fabric PLL (VDD25) power, Transceiver Core (VDDA) power and Transceiver PLL (VDDA25) power.

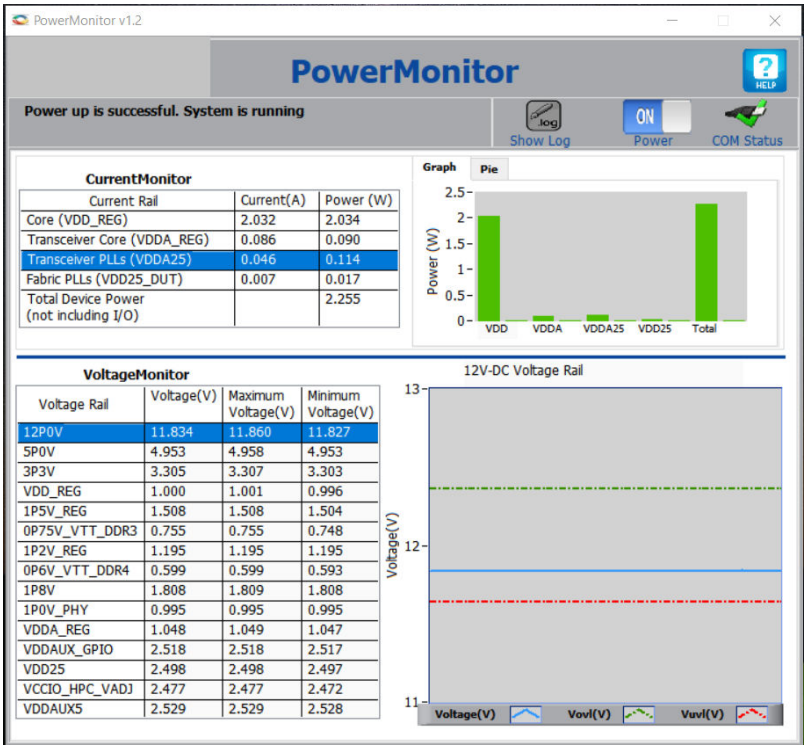
The total power consumed by the device is displayed in the PowerMonitor GUI, as shown in the following figure.

Figure 2-2. Total Power—PolarFire Normal Demo



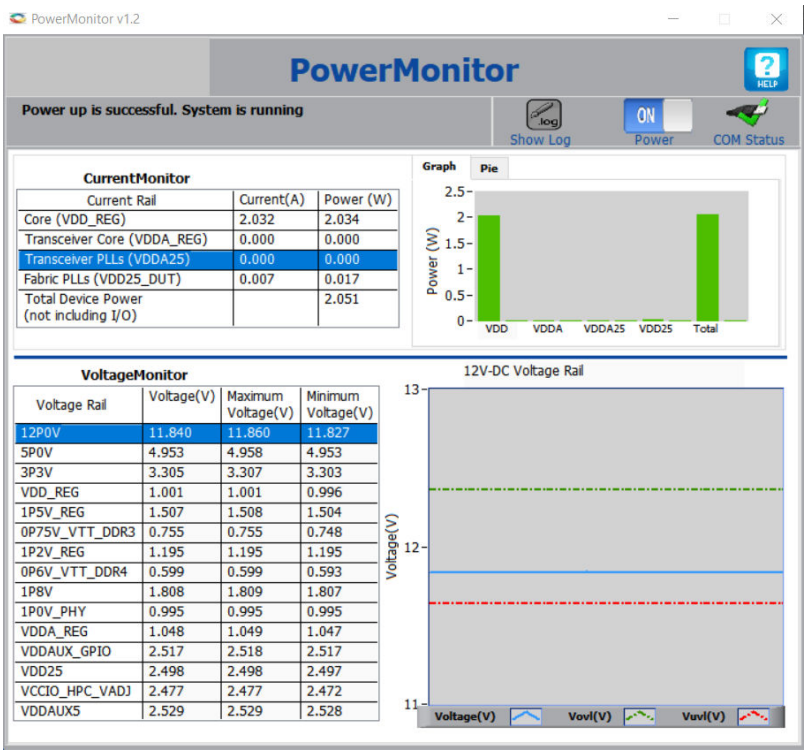
6. To program the design with Low-power options (`PF_Demo_Low_power.job`) using FlashPro Express, see Appendix 1: Programming the Device Using FlashPro Express section.
   The total power consumed by the device is displayed in the PowerMonitor GUI.

Figure 2-3. Total Power—PolarFire Low-Power Demo



7. Turn off **XCVR** and **TX PLL** by changing **DIP-3 OFF** and measure power.
   The total power consumed by the device is displayed in the PowerMonitor GUI.

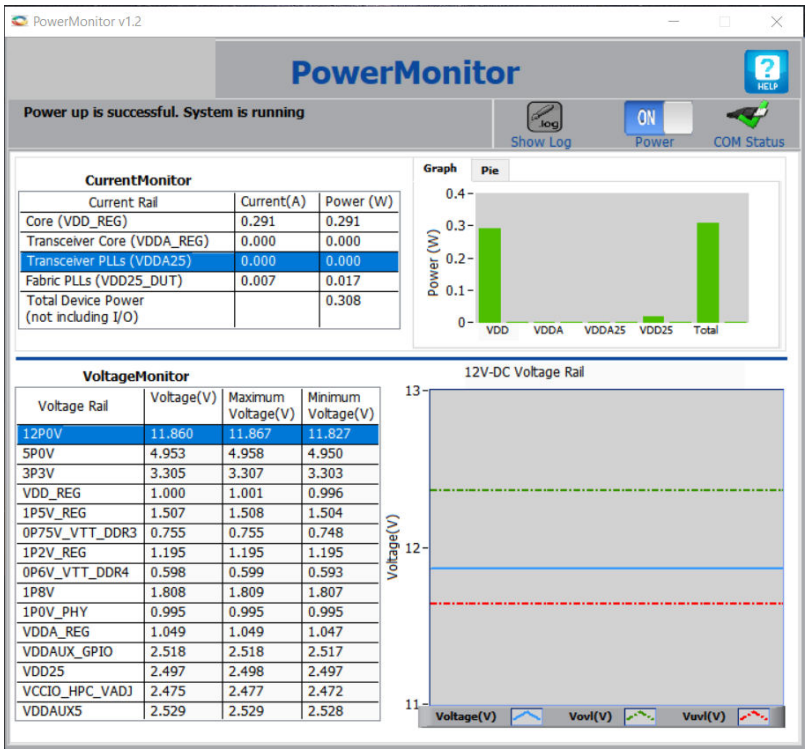
Figure 2-4. Total Power—XCVR and TX PLL



8. Turn on clock gating by changing **DIP-2 ON** and measure power.

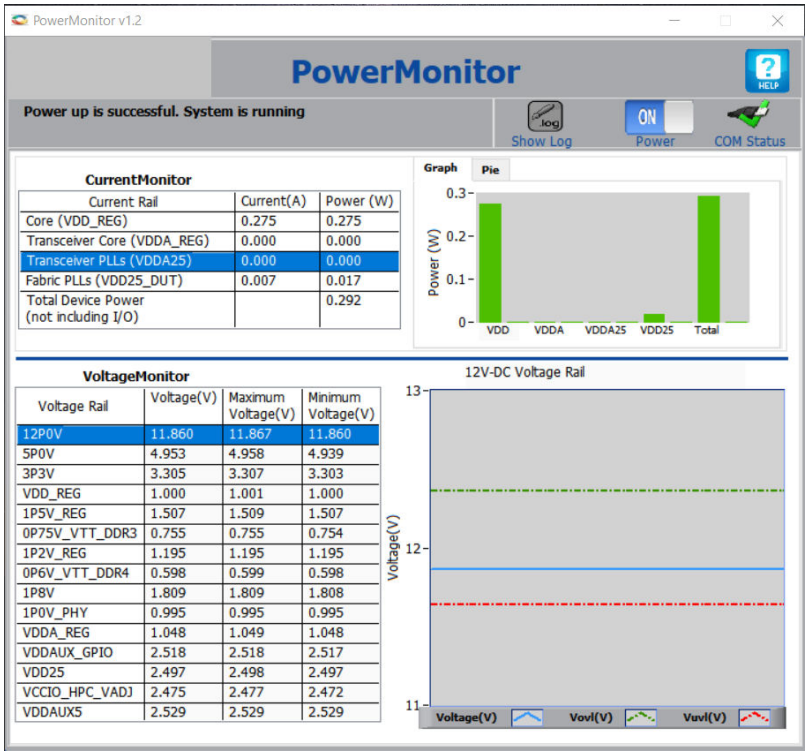
9. The LEDs {8, 9}, blink at different rates as the clock is available. The LEDs {4, 5}, {6, 7} and {10,11} maintain previous state as the clocks are not available.
   The total power consumed by the device is displayed in the PowerMonitor GUI.

Figure 2-5. Total Power—Clock Gating



10. Make CCC in Power-Down mode by changing **DIP-1 ON** and measure power.
    The total power consumed by the device is displayed in the PowerMonitor GUI.

Figure 2-6. Total Power—Power Down Mode



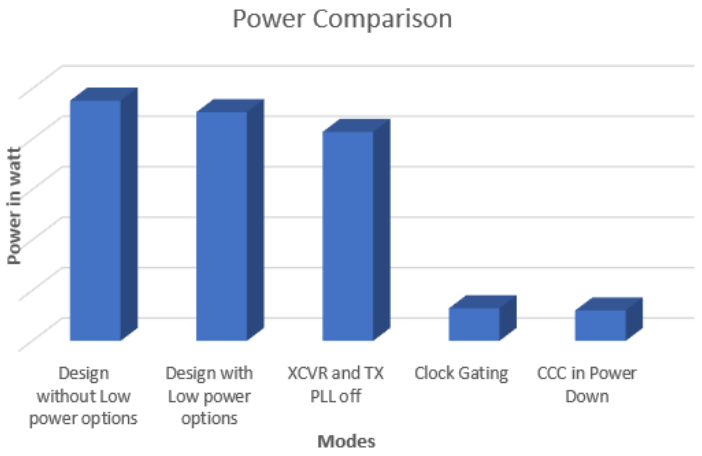
The following table lists the percentage of change after performing the preceding steps.

Table 2-2. Change in Power

Modes	Percentage of Power Saved After Each Step (Approximately)
Design without Low-power options	—
Design with Low-power options	5.0
XCVR and TX PLL off	9.0
Clock Gating	84.0
CCC in Power Down	6.0

The following figure shows the graphical comparison of the power in various modes.

Figure 2-7. Power Comparison



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

This chapter describes how to program the PolarFire device with the Job programming file using a FlashPro programmer. The default location of the Job file is: `mpf_AN4661_v2024p1_eval_df\Programming_Job`

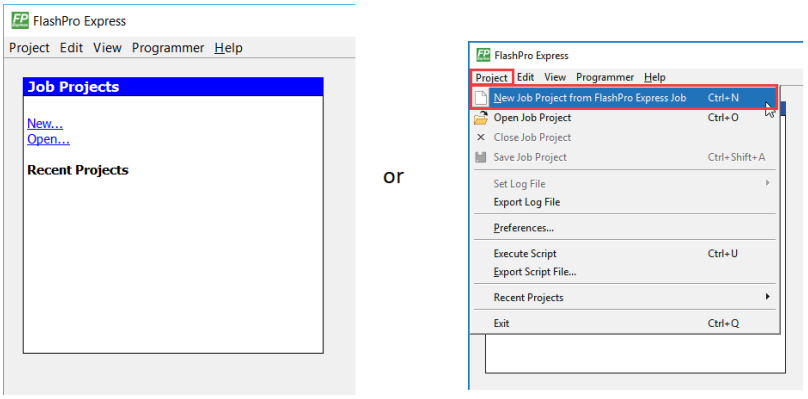
To program the PolarFire device using FlashPro Express, perform the following steps:

1. Ensure that the jumper settings on the board are the same as listed in Table 2-1.

> **Important:** The power supply switch must be switched OFF while making the jumper connections.

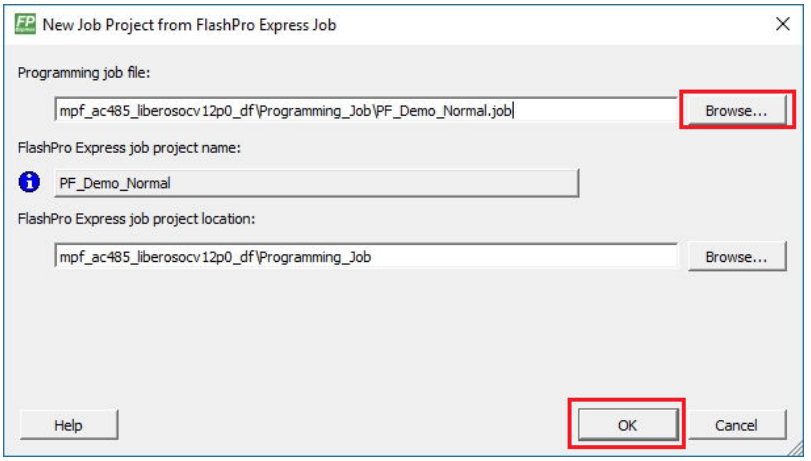
2. Connect the power supply cable to the J9 connector on the board.
3. Connect the USB cable from the Host PC to the J5 (FTDI port) on the board.
4. Power on the board using the SW3 slide switch.
5. On the host PC, launch the FlashPro Express software.
6. Click **New** or select **New Job Project** from **FlashPro Express Job from Project menu** to create a new job project, as shown in the following figure.

Figure 3-1. FlashPro Express Job Project



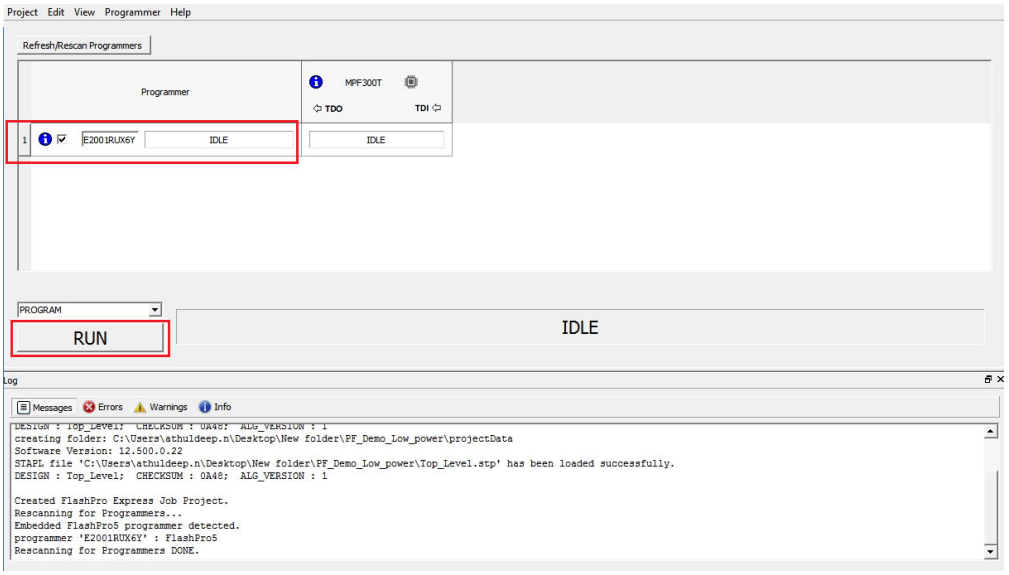
7. Enter the following in the New Job Project from FlashPro Express Job dialog box:
   - **Programming job file**: Click **Browse**, navigate to the location where the .job file is located, and select the file. The default location is: `<download_folder>\mpf_AN4661_v2024p1_eval_df\Programming_Job`.
   - **FlashPro Express job project location**: Click **Browse** and navigate to the location where user want to save the project.

Figure 3-2. New Job Project from FlashPro Express Job



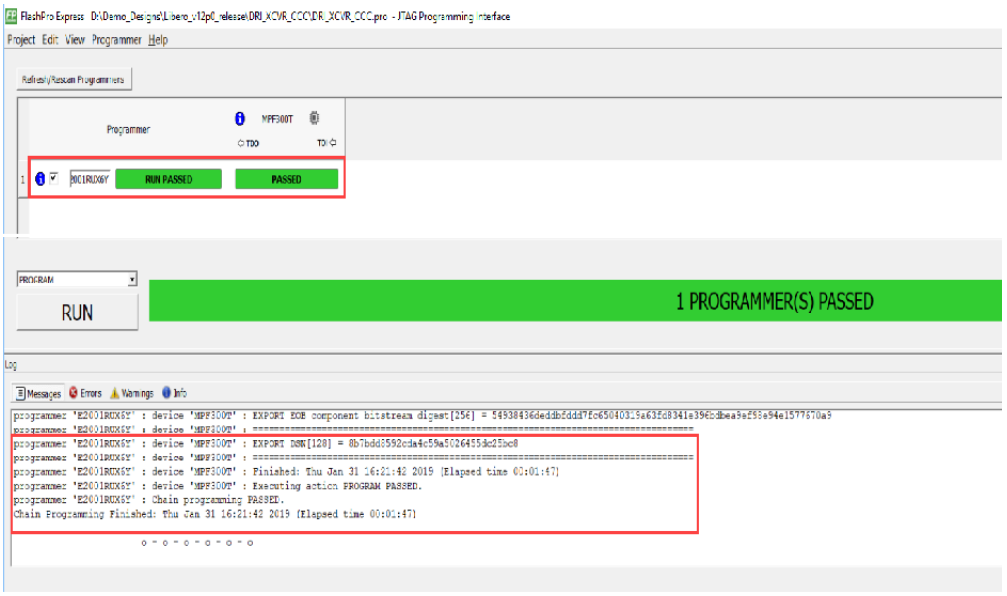
8. Click **OK**. The required programming file is selected and ready to be programmed in the device.
9. The FlashPro Express window appears, as shown in the following figure. Confirm that a programmer number appears in the Programmer field. If it does not, confirm the board connections and click **Refresh/Rescan** programmers.

Figure 3-3. Programming the Device



10. Click **RUN** to program the device. When the device is programmed successfully, a RUN PASSED status is displayed as shown in the following figure.

Figure 3-4. FlashPro Express—RUN PASSED



11. Close **FlashPro Express**, or in the **Project** tab click **Exit**.

4. Appendix 2: Measuring Power (Ask a Question)

The following tools are available for power measurement.

- Power Calculator
- SmartPower
- Power Monitor

4.1 Power Calculator (Ask a Question)

Microchip Power Estimator (MPE) PolarFire is a spreadsheet-based tool that enables designers to estimate the power consumption of PolarFire FPGAs from design concept to design implementation. It provides thermal analysis, as well as information about the contribution of various factors in the total power consumption of FPGA. Operating frequencies, device resources, clock resources, toggle rates, and other parameters are first entered into the Power Estimator. These parameters are then combined with pre-determined power models based on simulation and characterized device data to estimate power consumption. For more information about the Power Calculator, see PolarFire FPGA and PolarFire SoC FPGA Power Estimator User Guide.

4.2 SmartPower (Ask a Question)

SmartPower provides a detailed and accurate way to analyze designs for Microchip SoC FPGAs from top-level summaries to deep down specific functions within the design, such as gates, nets, I/Os, memories, clock domains, blocks, and power supply rails. You can analyze the hierarchy of block instances and specific instances within a hierarchy, and each can be broken down in different ways to show the respective power consumption of the component pieces. For more information about the SmartPower, see SmartPower User Guide

4.3 Power Monitor (Ask a Question)

The SmartFusion® A2F 200 device on the PolarFire board monitors the voltage and current on different PolarFire power rails. It measures the current for different components and displays the power on the Microchip PowerMonitor application. PowerMonitor is a Graphical User Interface (GUI) application that runs on the host PC. The power monitoring program on the SmartFusion device measures the total device power without any manual measurements. For more information about the Power Monitor, see UG0747: PolarFire FPGA Evaluation Kit User Guide

5. Appendix 3: Recommendations (Ask a Question)

This section provides information about the recommended settings to achieve lower power.

- Transceiver Power Reduction Recommendations
- LSRAM/µSRAM Power Reduction Recommendations
- Mathblock Power Reduction Recommendations

5.1 Transceiver Power Reduction Recommendations (Ask a Question)

The PolarFire device has options to reduce the power of the Transceiver PMA and the associated DFE calibration block. The following options are available to reduce the power consumption of the transceiver PMA as well as the DFE calibration block within the PCIESS and transceiver PCS blocks.

- Disabling DFE and EM blocks, when CDR mode is used during normal operation
- Disabling EM block (PDEM = 1'b0), when DFE mode is used during normal operation
- Disabling the Calibration clock (DFE_CAL_CEN = 1'b0), when either PDDFE = 1'b1 or PDEM = 1'b1

> **Important:** These settings must be restored before any DFE/CDR calibration or eye monitor functions can be performed.

- Modifying the CTLE Drive settings from the default of 0x2:
  - For setting 0x1, the estimated power reduction is by 1.5 mW.
  - For setting 0x3 (only for Revision F), the estimate power increases by 3.75 mW (3.93 mW when used with VDDA = 1.05V).
  - For PDDFE=1'b1 and PDEM = 1'b1, the Transceiver PMA power can be reduced further by setting CSENT[3:1]_DFEEM = 0x0.
- Reduce the Tx amplitude:
  - The Tx amplitude must be large enough to transmit the required data and withstand cross-talk from other lanes, as much as possible. Additional amplitude beyond this optimal limit only increases power, noise, and cross-talk in the system.
  - De-emphasis must be used to improve the performance of the system by removing high frequency content that must be transferred across the backplane. Modifying de-emphasis parameters has no effect on the overall Transceiver power.
  - Examples of power reduction from the base of 88 mW at 6.875 Gbps for 1000 mV peak-peak amplitude settings are:

Table 5-1. Example Settings

Serial Number	TX Amplitude (mV)	pk-pk (mW)
1	1000	88
2	800	81
3	600	74
4	400	67
5	200	60

- Disable the TxPLL auxiliary clock:
  - If the auxiliary clock from the PLL is not needed, it must be disabled by setting TXPLL_AUXDIVPD/EXTPLL_AUXDIVPD = 1'b1. This setting saves the significant power on VDD rail.

- Some functions within the Serial subsystem require the auxiliary clock output to be enabled. The main function known to require this function to be enabled is the jitter attenuator function.

5.2 LSRAM/µSRAM Power Reduction Recommendations (Ask a Question)

Following are the recommended power reductions settings for LSRAM/µSRAM:

- Disabling the LSRAM read enable signal retains their previous output value and there will be no dynamic read power consumed.
- Use the Block enable signal for read and write address enable logic to avoid the continuous toggling and thereby consumes the power only during read/write operation. During IDLE, disabling the block enable signal saves the power.
- Cascading memory blocks in deep saves the power. For example, two blocks of 1024 × 20 combined to create 2048 × 20.

5.3 Mathblock Power Reduction Recommendations (Ask a Question)

Enabling the input and output pipeline registers in Mathblock avoids the glitches in combinational logic between I/O ports, this reduces the sudden power fluctuations. Pipelining the I/O ports increases the high performance, but it also increases the total power consumption due to additional pipeline registers. If the design needs moderate performance with not many glitches, it is recommended to use the non-pipelining method to reduce the power consumption in Mathblocks.

6. Appendix 4: 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 window till generation of job file.

To run the TCL, perform the following 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, see `mpf_AN4661_v2024p1_eval_df/TCL_Scripts/TCL_Script_readme.txt`.

For more details on TCL commands, see Tcl Commands Reference Guide. 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 current publication.

Table 7-1. Revision History

Revision	Date	Description
C	01/2025	The following is a summary of the changes made in the C revision. • Updated the Microchip Power Monitor application weblink in the Prerequisites and Running the Demo sections.
B	06/2024	The following is a summary of the changes made in B revision. • Updated the document for Libero® SoC v2024.1. • Updated the design files and TCL script's location throughout the document.
A	08/2022	The following is a summary of the changes made in A revision. • The document was migrated to the Microchip template. • The document number was updated to DS00004661 from 51900485. • The document ID was updated to AN4661 from AC485. • Removed section "PCIe Low Power Option" from the document. • Updated Figure 1-1. • Updated Figure 1-6. • Updated Figure 1-7. • Updated Figure 1-8 and added a note. • Updated the design files and TCL script's location throughout the document.
3.0	—	Added Appendix 4: Running the TCL Script.
2.0	—	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.
1.0	—	The first publication of this document. This document replaces PolarFire® Low Power User Guide and PolarFire Low Power Demo Guide documents.

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

Trademarks

The “Microchip” name and logo, the “M” logo, and other names, logos, and brands are registered and unregistered trademarks of Microchip Technology Incorporated or its affiliates and/or subsidiaries in the United States and/or other countries (“Microchip Trademarks”). Information regarding Microchip Trademarks can be found at https://www.microchip.com/en-us/about/legal-information/microchip-trademarks.

ISBN: 979-8-3371-0469-0

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.

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 products are 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.