IGLOO2 FPGA Low Standby Power - Libero SoC v11.6

DG0564 Demo Guide

October 2015

Microsemi®

Revision History

Date	Revision	Change
October 21, 2015	4	Fourth release
February 6, 2015	3	Third release
August 22, 2014	2	Second release
October, 2013	1	First release

Confidentiality Status

This document is a non-confidential.

Table of Contents

Preface

About this document

This demo is for the IGLOO®2 field programmable gate array (FPGA) devices. It provides instructions on how to use the corresponding reference design.

Intended Audience

IGLOO2 devices are used by:

- FPGA designers
- System-level designers

References

Microsemi Publications

- *SmartFusion2 and IGLOO2 Power Estimator User Guide*
- *UG0444: SmartFusion2 SoC and IGLOO2 FPGA Low Power Design User Guide*
- *UG0445: SmartFusion2 SoC and IGLOO2 FPGA Fabric User Guide*

IGLOO2 FPGA Low Standby Power - Libero SoC v11.6

Introduction

Microsemi® IGLOO2 FPGAs are designed to meet the demand of low power FPGAs. The IGLOO2 devices exhibit lower power consumption in static and dynamic modes. This demo guide describes how to implement the standby power mode on the IGLOO2 devices using SmartDesign, and measure the standby power. The design drives the light emitting diodes (LEDs) on the IGLOO2 Evaluation Kit with a pattern based on the state of the switches SW1 and SW3, as shown in Table 1.

Table 1 • **LEDs Pattern**

LED E1, F4, F3, G7 Behavior	Standby Entry (SW1)	Standby Exit (SW3)
LEDs toggle	Released	Released
LEDs on	Depressed and Released	Released
LEDs toggle	Depressed and Released	Depressed

This demo guide describes the following:

- Creating a Libero® System-on-Chip (SoC) project
- Implementing the standby power mode on the IGLOO2 devices using SmartDesign
- Importing a PDC file, running layout, and programming the IGLOO2 silicon
- Measuring the standby power using a standard digital voltmeter (DVM)/multimeter

Design Requirements

Table 2 shows the design requirements.

Table 2 • **Design Requirements**

Design Requirements	Description
Hardware Requirements	
IGLOO2 Evaluation Kit: • 12 V adapter • FlashPro4 programmer	Rev C or later
Desktop or Laptop	Windows 64-bit Operating System
Software Requirements	
Libero SoC	v11.6
FlashPro Programming Software	v11.6

Demo Design

Introduction

The demo design files are available for download from the following path in the Microsemi website:

http://soc.microsemi.com/download/rsc/?f=m2gl_dg0564_liberov11p6_df

The demo design files include:

- Libero SoC project
- Constraint file
- Programming file
- Source files
- Readme file

Refer to the Readme.txt file provided in the design files for the complete directory structure.

The design consists a 32 kHz external main crystal oscillator, fabric CCC (FCCC), standby power control logic, and fabric logic block. Figure 1 shows the block diagram of the design.

FCCC is configured to provide a 100 MHz clock to the fabric logic. It is also configured with phase-locked loop (PLL) power-down enabled. The 32 kHz external main crystal oscillator is the reference clock source for FCCC. The lock signal is used as the reset signal to the fabric logic. The standby power control logic consists a clocked S-R latch, which powers down the PLL of FCCC. The fabric logic consists 269 stages of 18-bit loadable up-counters, 13 stages of shift registers, and 11 stages of LSRAM and math blocks. It also consists an LED driver block, which is connected to a set of LEDs to monitor the state of the fabric while entering and exiting the standby power mode.

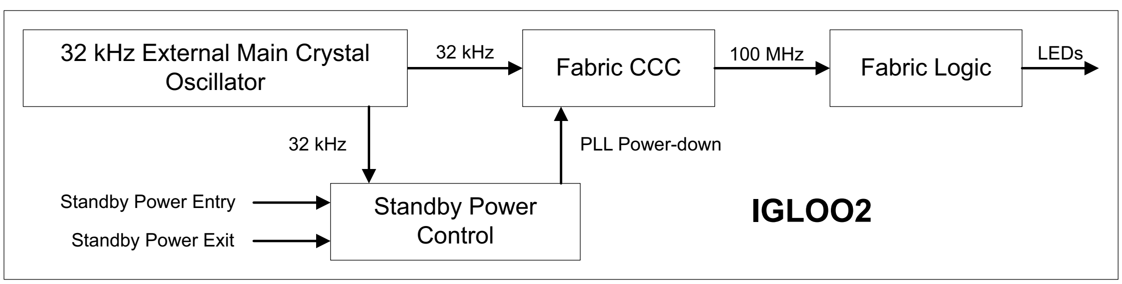

Figure 1 • **Design Block Diagram**

Extracting the Source Files

Extract the *m2gl_dg0564_liberov11p6_df.zip* file to the *<C:\ or D:\>Microsemi_prj* folder on the PC. Confirm that a folder named *IGL2_Standby_tutorial* containing sub-folders named *Source_files* and *Constraints* are extracted.

Creating the Design

This section describes how to create the standby power mode enabled design using SmartDesign. Some source files are provided in the *Source_files* folder.

Launching Libero SoC

The following steps describe how to launch Libero SoC:

1. Go to **Start > Programs > Microsemi Libero SoC v11.6 > Libero SoC v11.6,** or double-click the shortcut icon on the PC. This opens the **Libero SoC Project Manager** window, as shown in Figure 2.

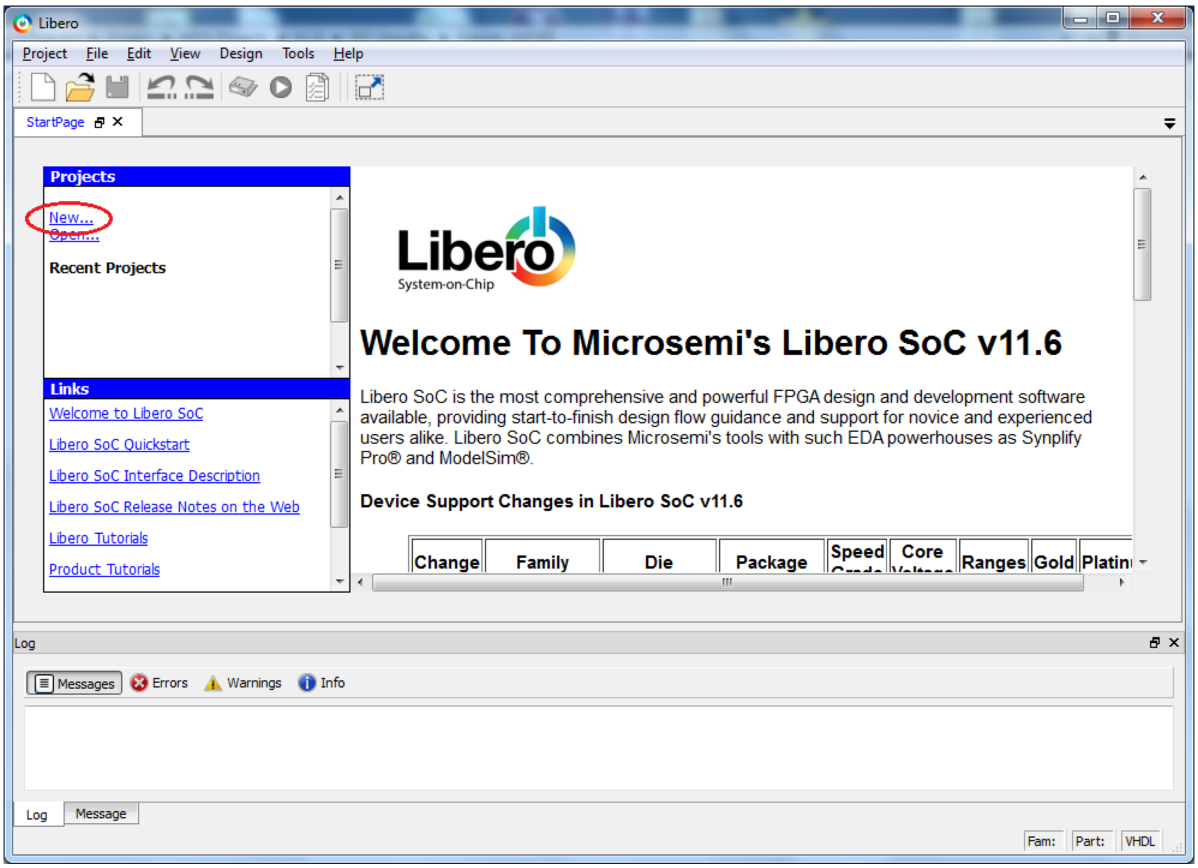

Figure 2 • Libero SoC Project Manager

2. Create a new project using one of the following options:
   – Select **New** on the **Start Page** tab, as shown in Figure 2.
   – In the Libero SoC menu, go to **Project > New Project**.

   This opens the **New Project** window, as shown in Figure 3 on page 8.
3. Enter the following information in the **New Project - Project Details** window, as shown in Figure 3 on page 8:
   – **Project Name**: IGL2_Standby
   – **Project Location**: <C:\ or D:\>Microsemi_prj\IGL2_Standby_tutorial
   – **Preferred HDL type**: VHDL
   – **Enable Block Creation**: Not selected

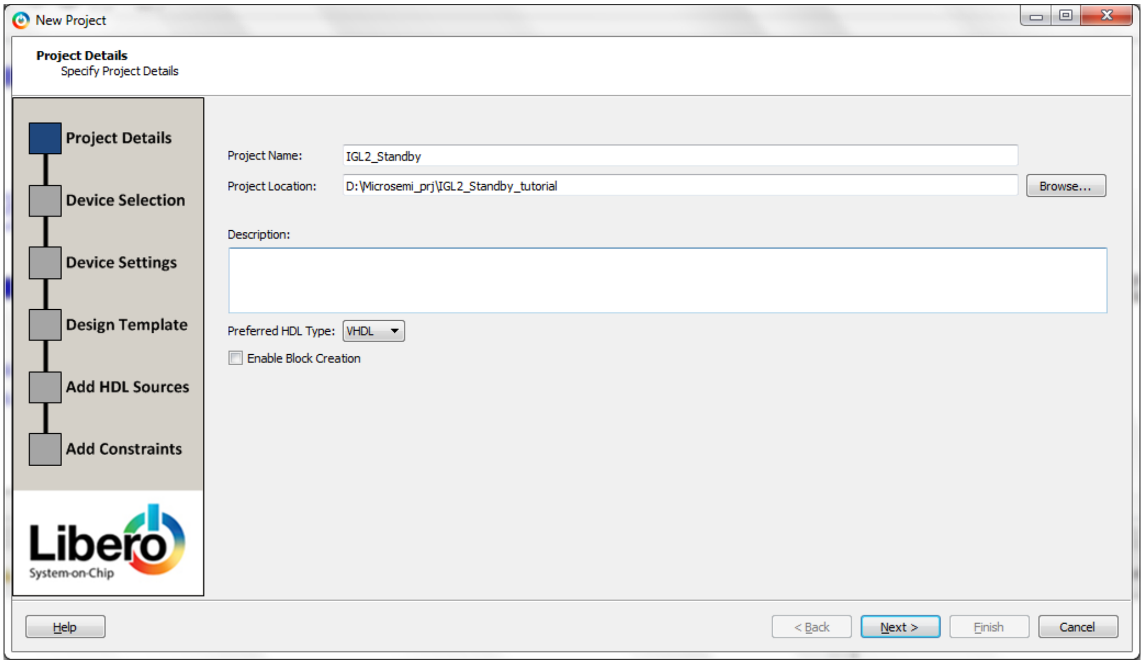

Figure 3 • **New Project - Project Details**

4. Click **Next**. This opens **New Project - Device Selection** window as shown in Figure 4 on page 9.
5. Select the following values from the drop-down list, highlighted in Figure 4 on page 9:
   – **Family**: IGLOO2
   – **Die**: M2GL010T
   – **Package**: 484 FBGA
   – **Speed**: -1
   – **Core Voltage**: 1.2
   – **Range**: COM

6. Select the filtered device **M2GL010T-1FG484**, as shown in Figure 4.

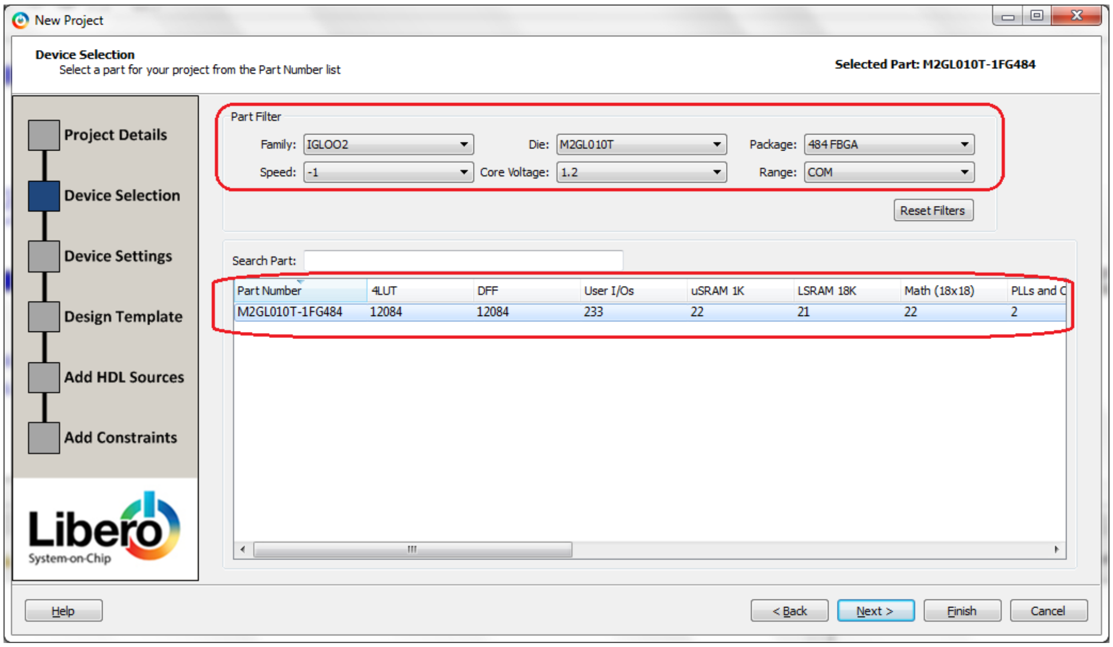

Figure 4 • **New Project - Device Selection**

7. Click **Next**. This opens **New Project - Device Settings** window as shown in Figure 5 on page 10.
8. Select the following values in the **Power supplies** section from the drop-down list, highlighted in Figure 5 on page 10:
   – **PLL Supply Voltage (V)**: 3.3 V
   – **Power on Reset Delay**: 100 ms

The PLL analog supply voltage can be either 2.5 V or 3.3 V. The voltage setting in the **New Project - Device Settings** window must match the PLL analog supply voltage on the board to ensure that the PLL works properly. The PLL analog supply voltage is connected to 3.3 V on the IGLOO2 Evaluation Kit.

9. Do not change the default selections. Click **Finish**.

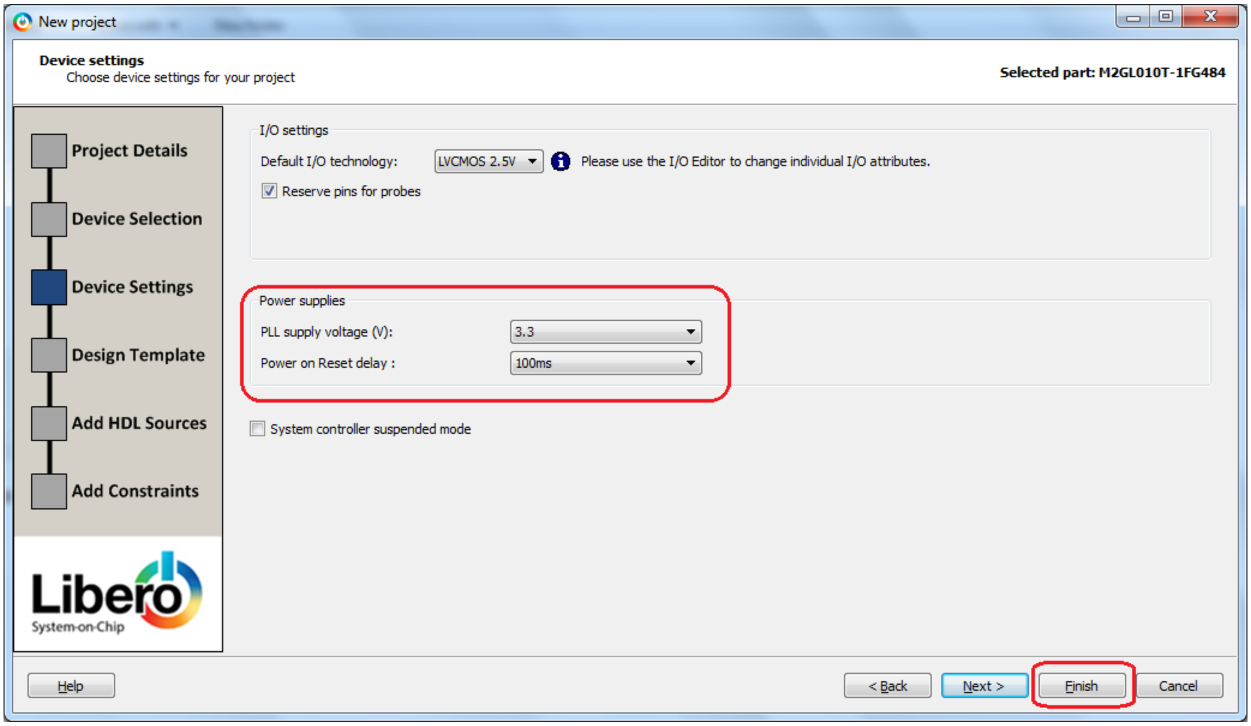

Figure 5 • **New Project - Device Settings**

10. In the **Design Flow** window, expand **Create Design**, as shown in Figure 6 on page 11.

11. Right-click **Create SmartDesign** and click **Run**.

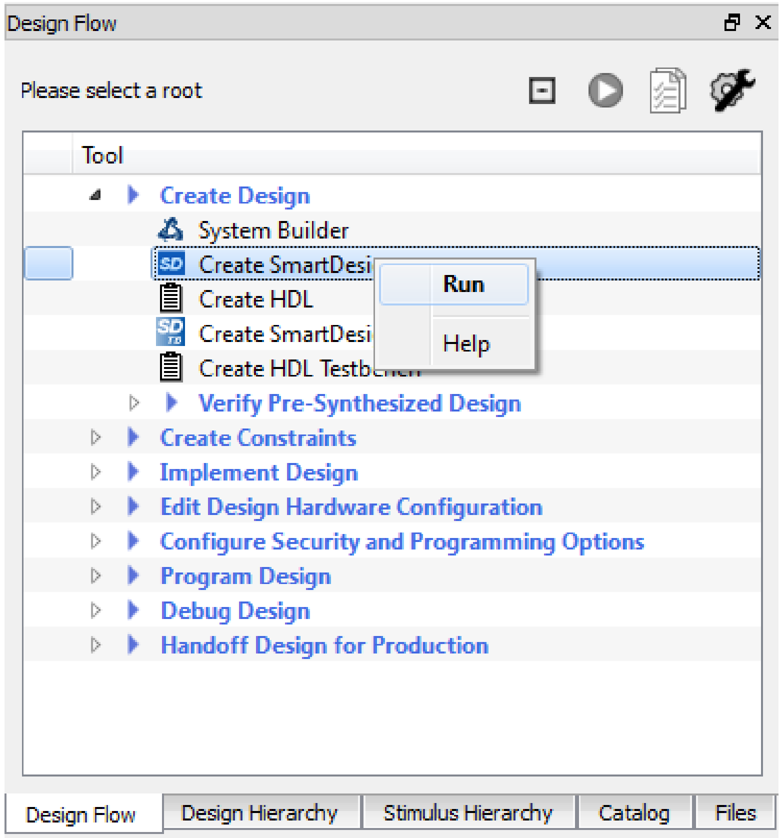

Figure 6 • **Creating SmartDesign**

12. In the **Create New SmartDesign** dialog box, enter the **Name** as IGL2_Standby and click **OK**. A new SmartDesign canvas opens.

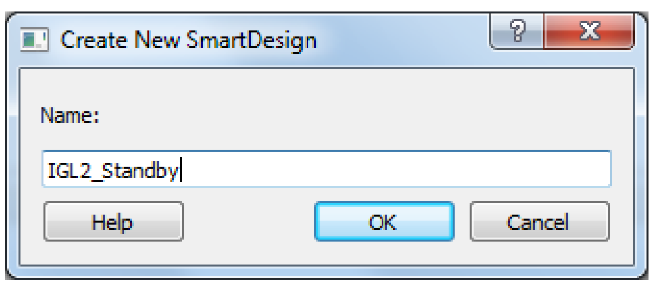

Figure 7 • **Entering SmartDesign Name**

This design uses a fabric CCC to generate a 100 MHz internal clock. The CCC reference clock is the 32 kHz external main crystal oscillator.

13. In the IP **Catalog** tab, expand **Clock & Management**.

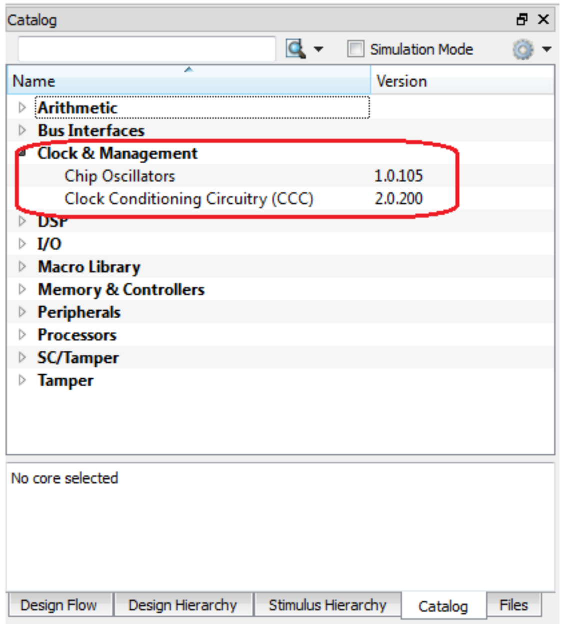

Figure 8 • **Clock & Management Category of Libero SoC IP Catalog**

14. Drag an instance of the clock conditioning circuitry (CCC) v2.0.200 component into the SmartDesign canvas.
15. Double-click the FCCC_0 component in the SmartDesign canvas and open the **FAB CCC Configurator** window, as shown in Figure 9 on page 13.
16. Click the **Basic** tab in the **FAB CCC Configurator** window, as shown in Figure 9 on page 13 and enter the following information:
    – **Reference Clock Frequency**: 0.032 MHz
    – **Reference Clock**: Select **Oscillators > Crystal Oscillator** from the drop-down list
    – **GL0: Checked; Frequency**: 100 MHz

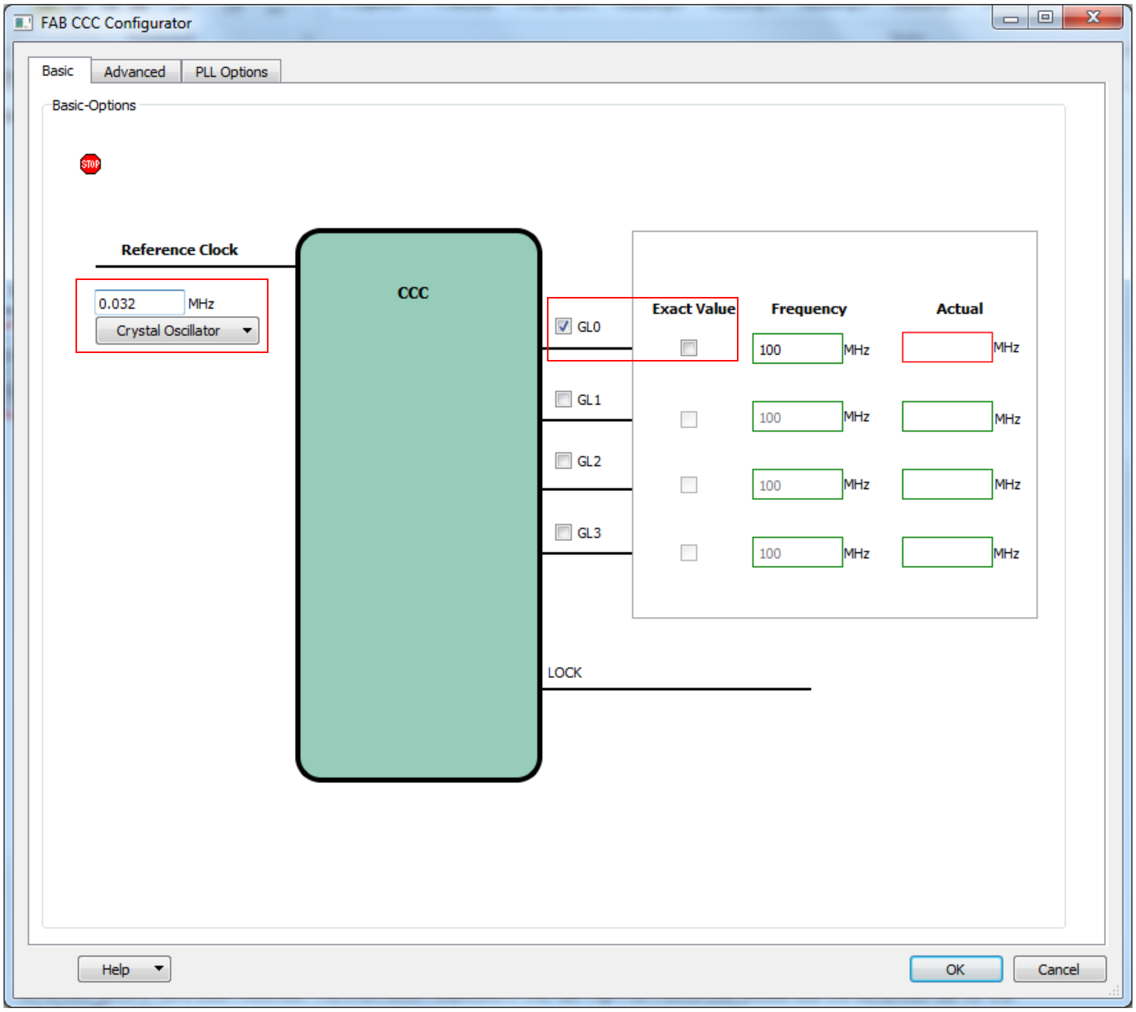

Figure 9 • **Configuring Fabric CCC**

17. Click the **Advanced** tab in the **FAB CCC Configurator** window and select **Internal > PLL Internal** from the drop-down list as PLL feedback source, as shown in Figure 10.

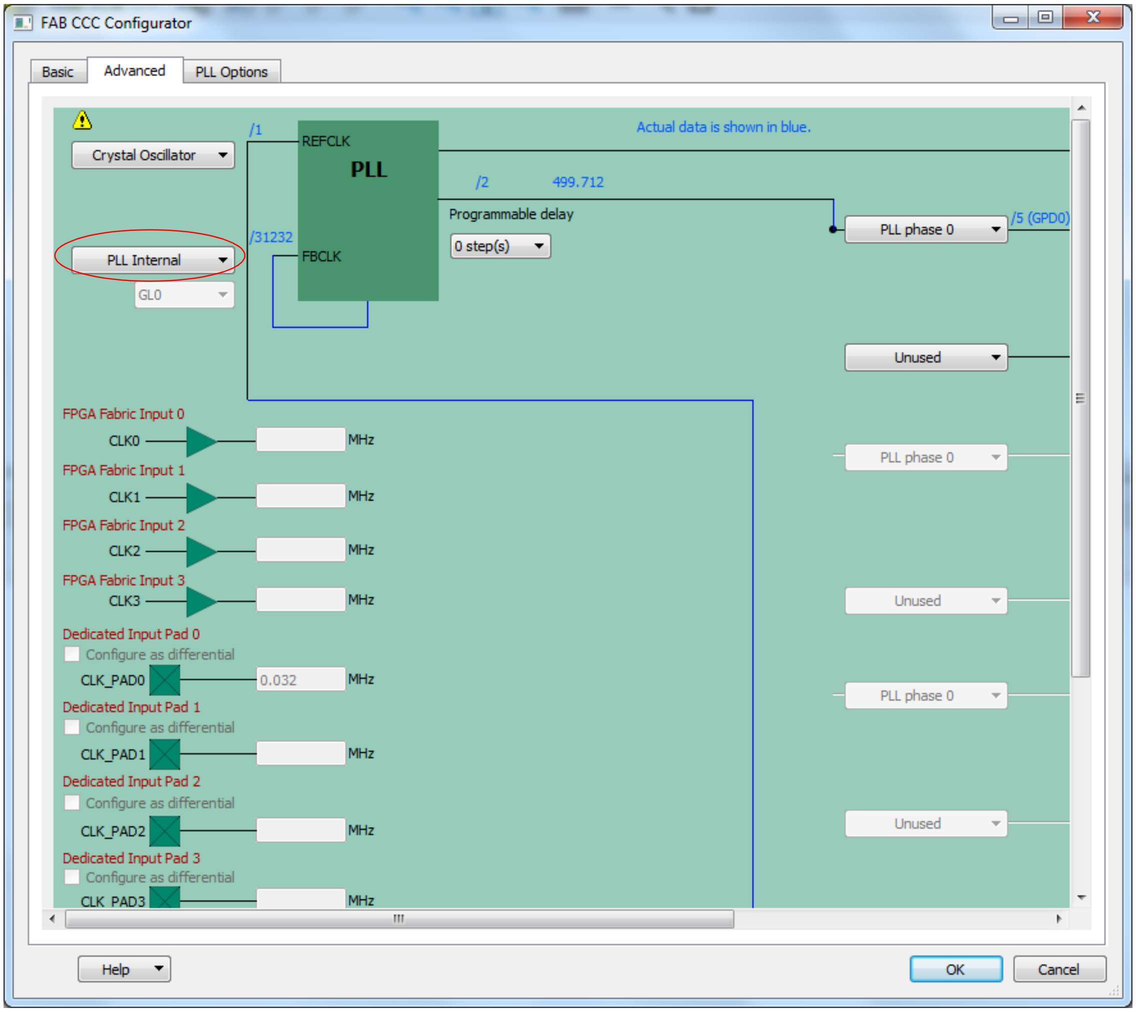

Figure 10 • **Configuring PLL Feedback Source**

18. Click the **PLL Options** tab in the **FAB CCC Configurator** window and select the **Expose PLL_ARST_N and PLL_POWERDOWN_N signals** checkbox, as shown in Figure 11.

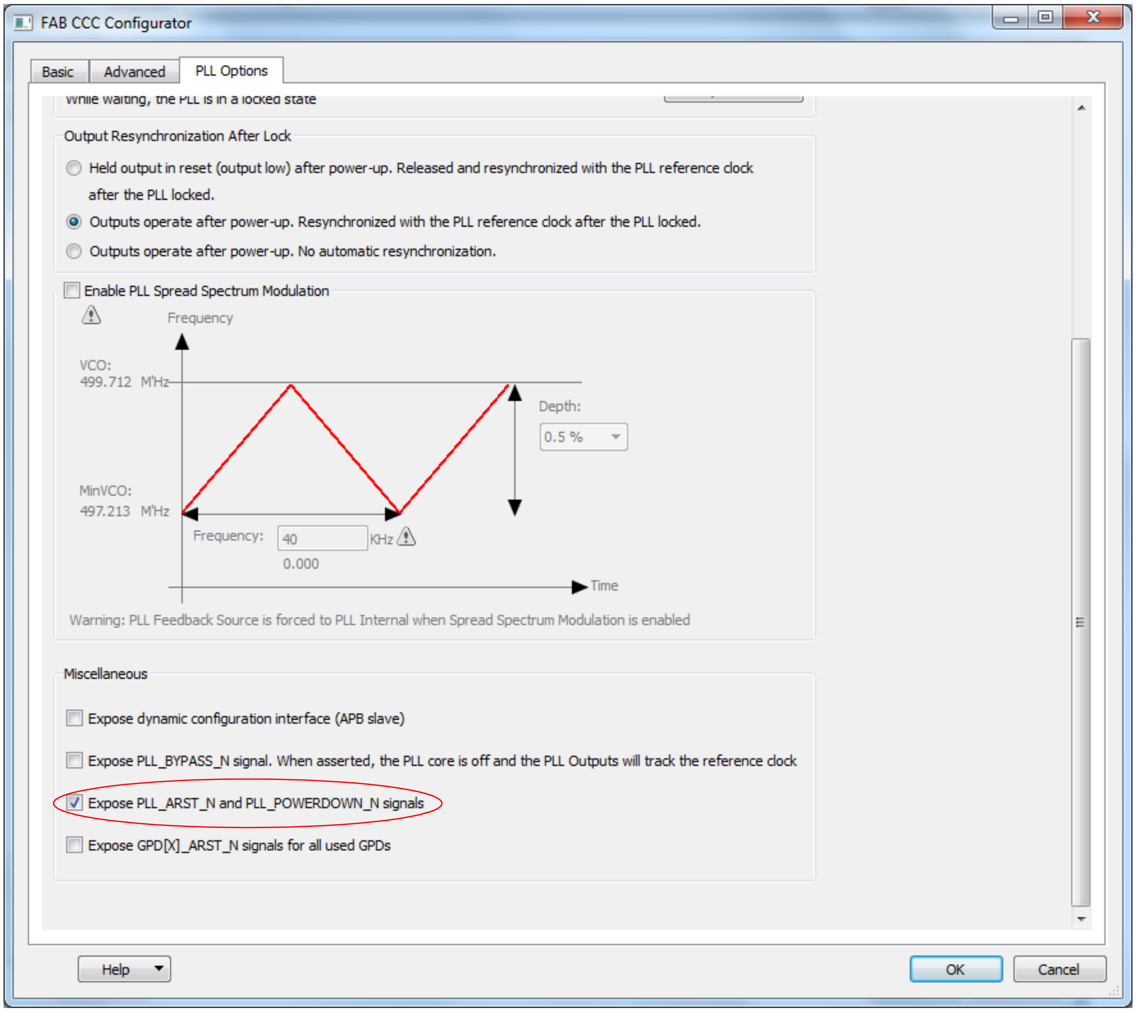

Figure 11 • **Configuring PLL Power-down Signal**

19. Click **OK**.
20. Drag an instance of the Chip Oscillators v1.0.105 component from the IP catalog into the SmartDesign canvas.
21. Double-click the OSC_0 component in the SmartDesign canvas and open the **Chip Oscillators Configurator** window, as shown in Figure 12 on page 16.
22. Configure the external main crystal oscillator to drive FCCC and fabric logic.
    Enter the following information, as shown in Figure 12 on page 16:
    – **External Main Crystal Oscillator**: Select
    – **Source**: Select **Crystal (32 KHz - 20 MHz)** from the drop-down list
    – **Frequency**: 0.032 MHz
    – **Drives Fabric CCC(s)**: Select
    – **Drives Fabric Logic**: Select

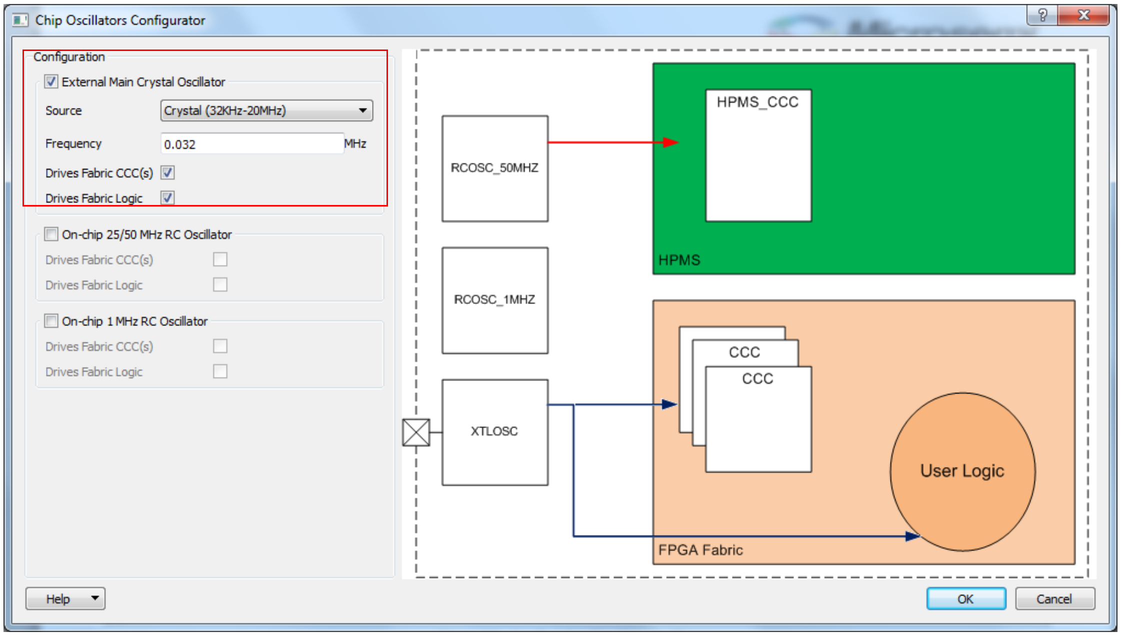

Figure 12 • **Configuring Chip Oscillators**

23. Click **OK**.

24. Import the VHDL source files into the project by selecting **Create HDL** under **Create Design** in the **Design Flow** tab. Right-click and select **Import Files…,** as shown in Figure 13.

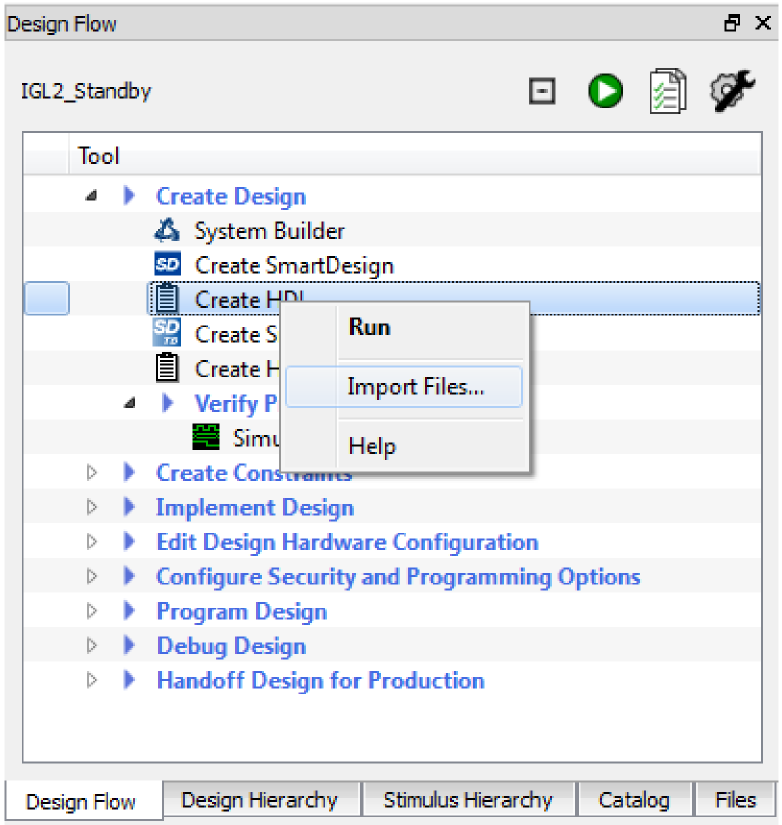

Figure 13 • **Importing HDL Source Files**

25. Browse to *<C:\ or D:\>Microsemi_prj\IGL2_Standby_tutorial\Source_files,* select all `.vhd`, `.v`, and `.h` files, and click **Open**.

The files are visible in the **Design Hierarchy** tab.

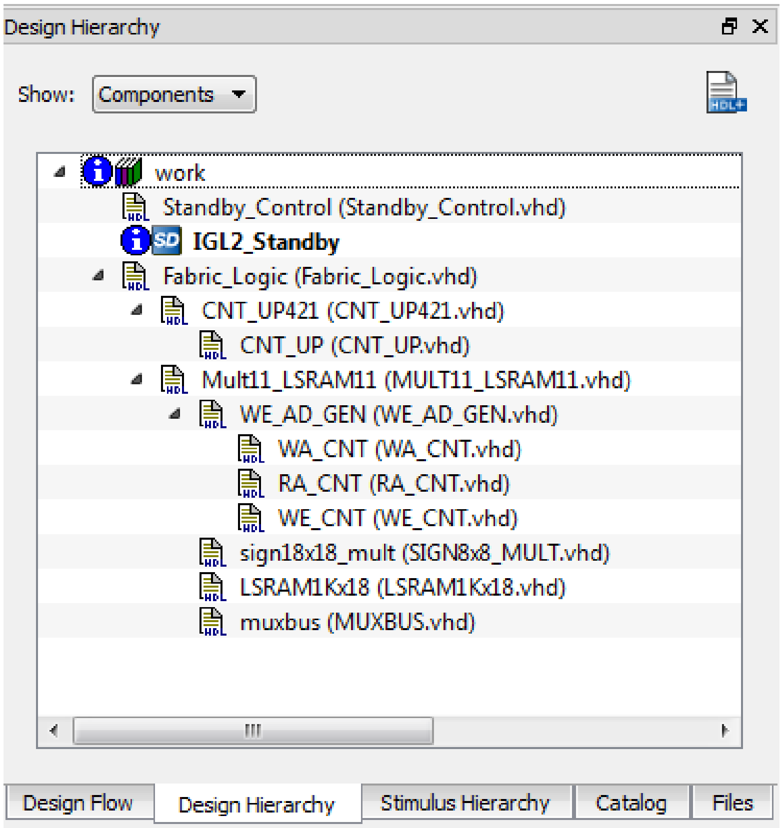

Figure 14 • **Design Hierarchy Tab with Imported Files**

26. Drag the Standby_Control and Fabric_Logic components into the SmartDesign canvas. The SmartDesign resembles Figure 15 on page 19.
27. Align the components to improve the appearance of the canvas.

    Expand the canvas area by selecting **View > Maximize Work Area,** or click the  icon on the tool bar.

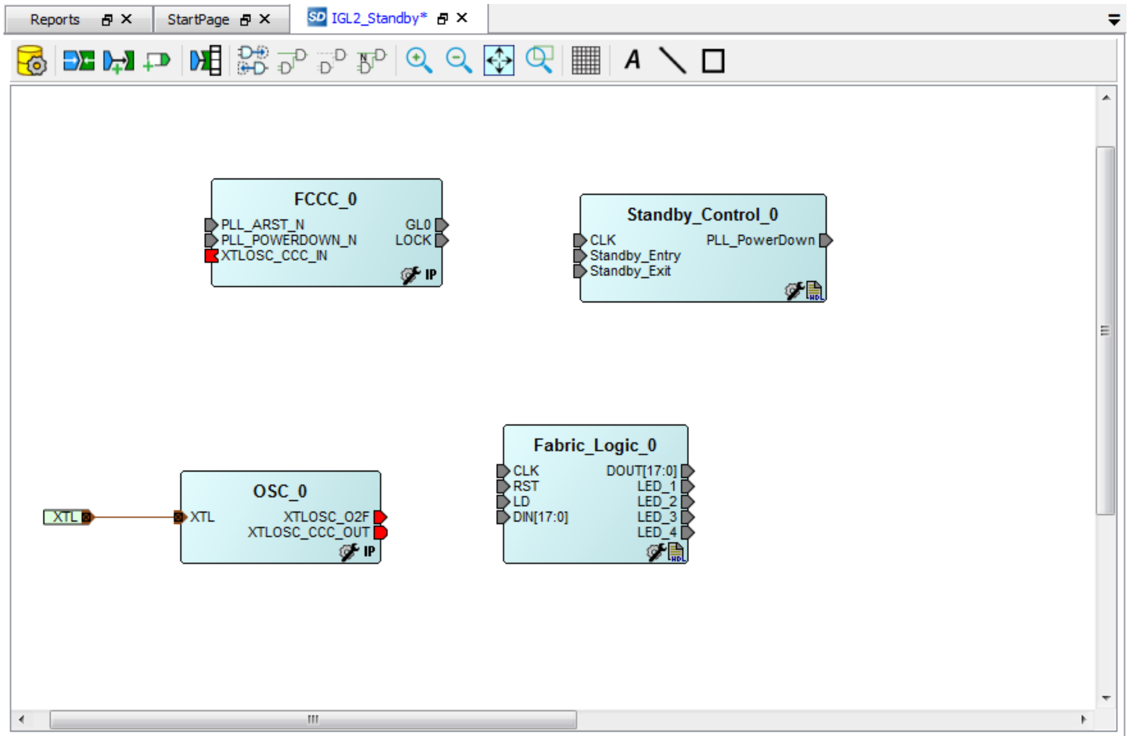

Figure 15 • **SmartDesign Canvas after Adding Components**

Connecting Components in the Canvas

SmartDesign in Libero SoC has a connection mode that supports click, drag, and release to connect the components.

Connect the components in the SmartDesign canvas using the following procedure:

1. Select **SmartDesign > Connection Mode** from the Libero SoC menu.
2. Connect the XTLOSC_CCC_OUT port of the OSC_0 component to the XTLOSC_CCC_IN port of the FCCC_0 component as follows:

   a. Click and hold the XTLOSC_CCC_OUT port of the OSC_0 component.

   b. Drag the XTLOSC_CCC_IN port of the FCCC_0 component and release the mouse button to connect.

   Note: You can also connect the ports by selecting them using **CTRL** (Ctrl + Click to select a port), right-clicking any of the selected ports, and selecting **Connect**.
3. Connect the other components in the SmartDesign canvas as per Table 3.

Table 3 • **Connections in Canvas**

From	To
OSC_0: XTLOSC_O2F	Standby_Control_0: CLK
Standby_Control_0: PLL_PowerDown	FCCC_0: PLL_ARST_N
	FCCC_0: PLL_POWERDOWN_N
FCCC_0: GL0	Fabric_Logic_0: CLK
FCCC_0: LOCK	Fabric_Logic_0: RST

4. Select **SmartDesign > Connection Mode** from the Libero SoC menu to exit connection mode.
5. Promote the ports shown in Table 4 to the top level. Right-click the port and select **Promote to Top Level**.

Table 4 • **Promote to Top Level**

Ports
Standby_Control_0: Standby_Entry
Standby_Control_0: Standby_Exit
Fabric_Logic_0: LD
Fabric_Logic_0: DIN[17:0]
Fabric_Logic_0: DOUT[17:0]
Fabric_Logic_0: LED_1
Fabric_Logic_0: LED_2
Fabric_Logic_0: LED_3
Fabric_Logic_0: LED_4

The SmartDesign canvas appears as shown in Figure 16. Drag the components or use the SmartDesign Auto Arrange feature to improve the appearance of the canvas.

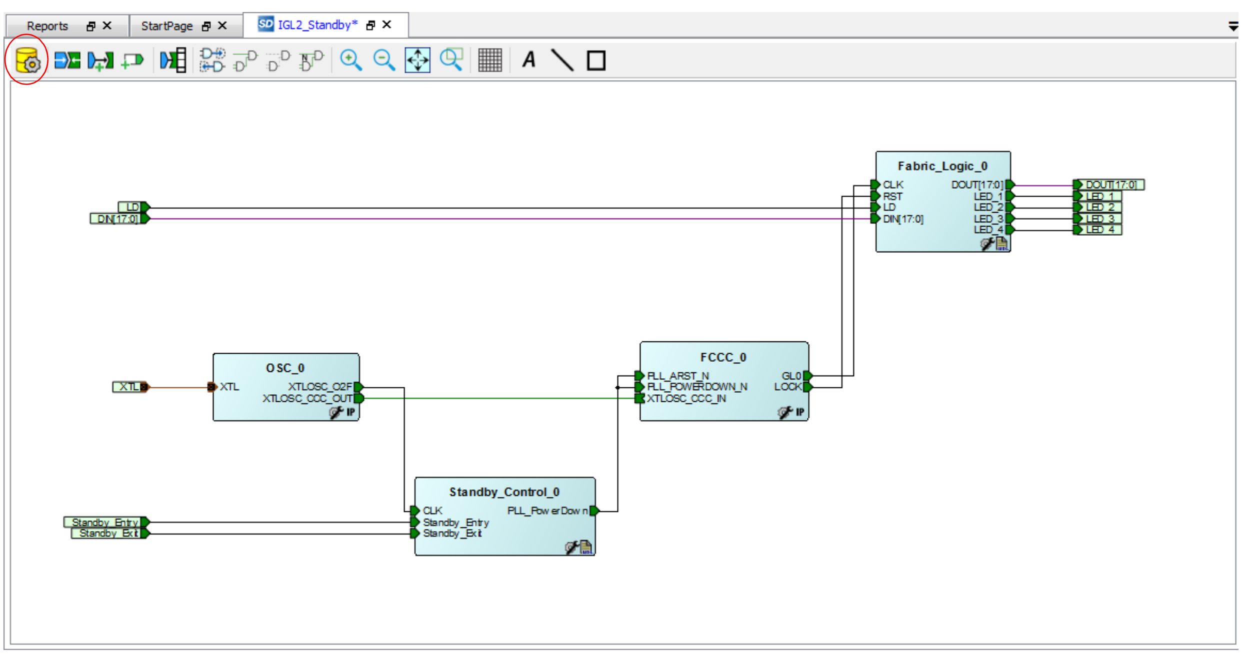

Figure 16 • **SmartDesign Canvas after Connections**

6. Go to **File > Save IGL2_Standby**, to save the design.
7. Generate the design by selecting **SmartDesign > Generate Component,** or by clicking the **Generate Component** icon on the SmartDesign toolbar (highlighted in Figure 16).
8. Go to **View > Restore Work Area** to restore the work area, if you expanded the work area earlier.
9. Confirm that the message **IGL2_Standby was generated** appears in the Libero Log window.
10. Go to **File > Close IGL2_Standby** to close the design.

Importing Physical Constraint files

This section describes how to import a physical design constraint (PDC) file to make I/O attribute and pin assignments for the layout.

The following steps describe how to make I/O assignments:

1. Expand **Create Constraints** in the **Design Flow** tab. Right-click **I/O Constraints** and select **Import Files...**..

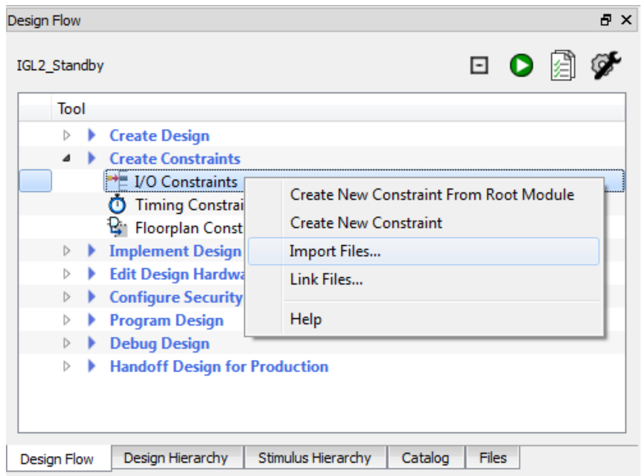

Figure 17 • **Importing I/O PDC Constraint File**

2. Browse to *<C:\ or D:\>Microsemi_prj\IGL2_Standby_tutorial\Constraints,* select the `IGL2_Standby.pdc` file, and click **Open**.
3. Select **No** in the **Information** dialog box.

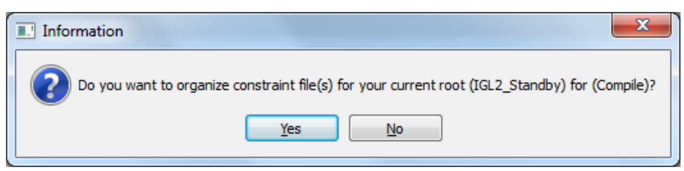

Figure 18 • **Information Dialog Box after Importing PDC Constraint File**

The file is visible in the Libero SoC **Files** tab under **constraint > io**.

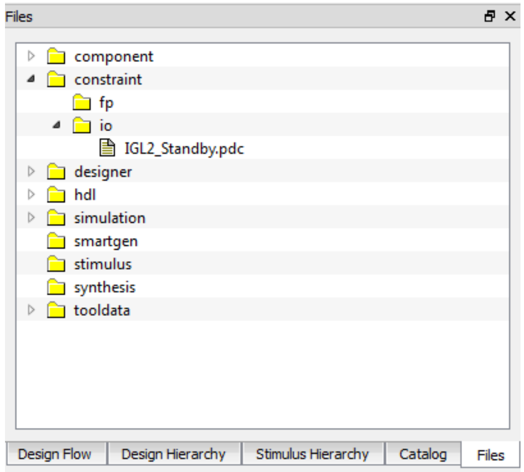

Figure 19 • **I/O PDC Constraint File in Libero SoC Project**

A description of the designer PDC constraints is available in the Libero Help (Go to **Help > Help Topics > Implement Design > Constrain Place and Route > Assigning Design Constraints > Design Constraints Guide > Reference > Constraints by File Format > PDC Command Reference**).

Synthesis and Layout

Use the push-button flow to synthesize the design with Synplify Pro, run layout, and generate the programming file as mentioned below:

1. Expand **Create Constraints > I/O Constraints** in the Libero SoC **Design Flow** tab. Right-click **IGL2_Standby.pdc** under **Constraints**.
2. Right-click and select **Use for Compile,** as shown in Figure 20. A green tick mark appears on the constraint file indicating that the file will be used.

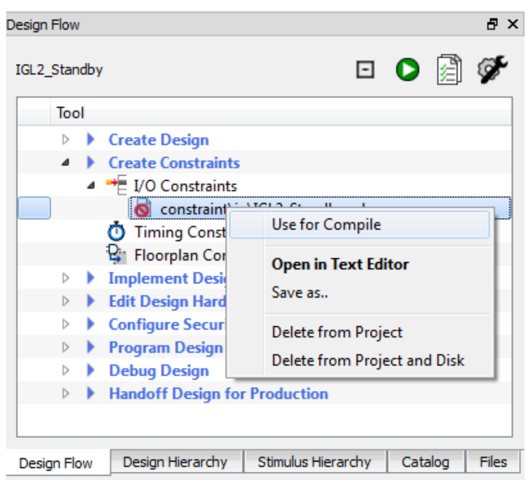

Figure 20 • **Selecting I/O PDC Constraint File in Design Flow Tab**

3. Click the G**enerate Bitstream** icon in the **Design Flow** tab (highlighted in Figure 21), or select **Design > Generate Bitstream** to synthesize the design, run layout using the I/O constraints that are created, and generate the programming file.

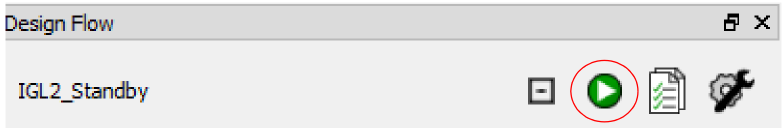

Figure 21 • **Generate Programming Data Icon**

The design implementation tools run in the batch mode. Successful completion of a design step is indicated by a green tick mark next to I**mplement Design** in the **Design Flow** tab, as shown in Figure 22.

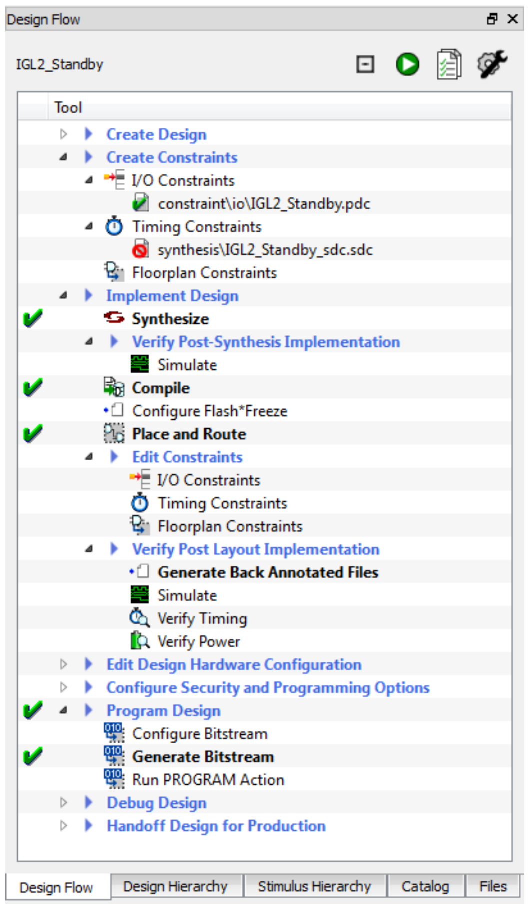

Figure 22 • **Successful Design Implementation**

4. Generate a power report by right-clicking **Verify Power** under **Verify Post Layout Implementation** in the **Design Flow** tab and selecting **Run**.

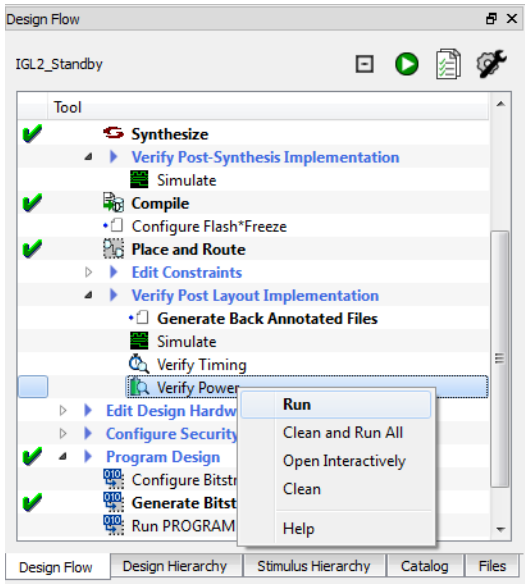

Figure 23 • **Generating Post Layout Power Report**

5. The **Reports** tab displays reports for the tools used to implement the design. Select **IGL2_Standby_power_report.xml** under **Verify Power** in the **Reports** tab to view the power consumption.

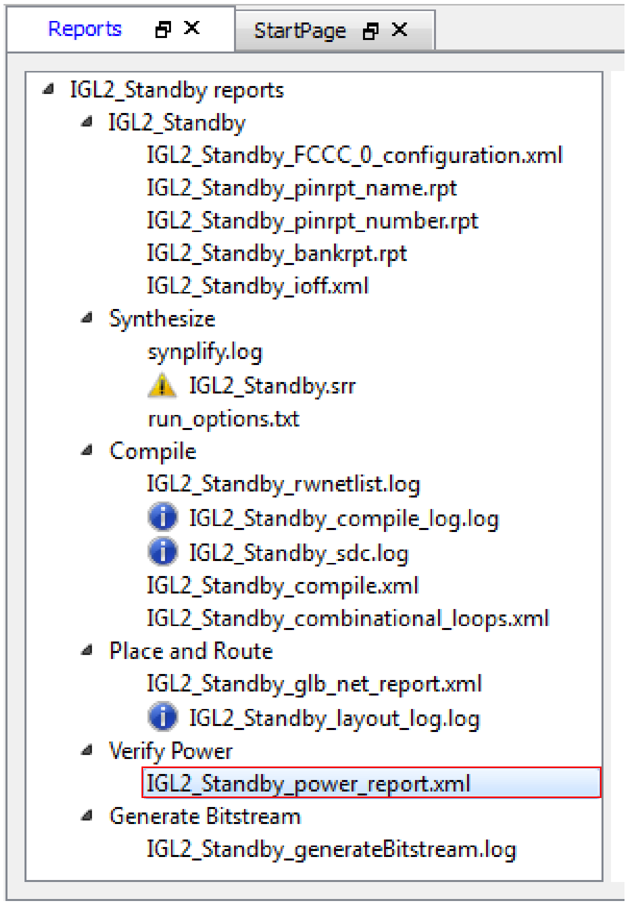

Figure 24 • **Reports Tab after Implementing Design**

The **Reports** tab displays the power report, as shown in Figure 25.

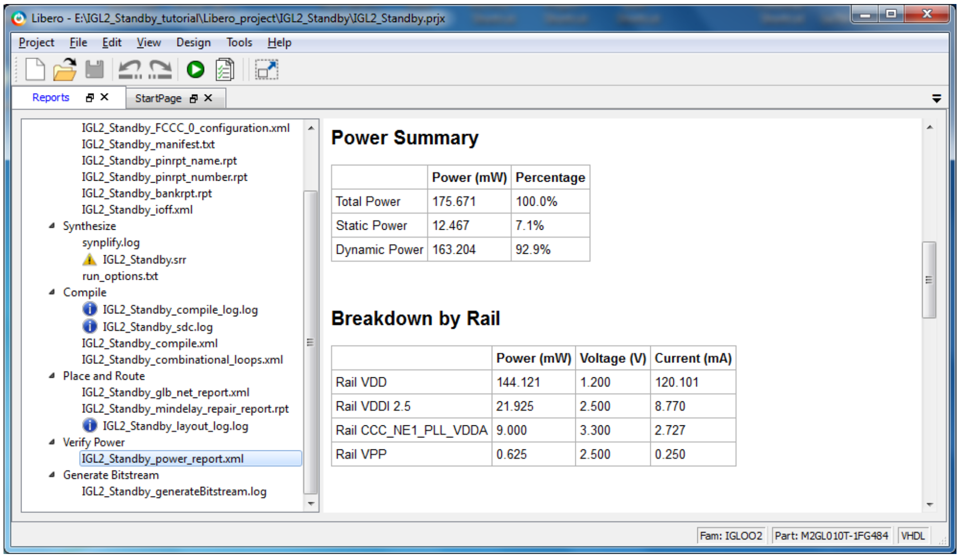

Figure 25 • **Power Report**

Programming

The following steps describe how to run FlashPro in the batch mode and program IGLOO2 M2GL010T on the IGLOO2 Evaluation Kit board:

1. Prior to programming and powering up the IGLOO2 Evaluation Kit board, ensure that the jumpers are positioned as shown in Table 5.

Table 5 • **Jumper Settings**

Jumper	Location	Setting
J3	Above the On/Off Switch in Figure 26	1-2 installed
J8	Below the JTAG Programming Header (J5) in Figure 26	1-2 installed

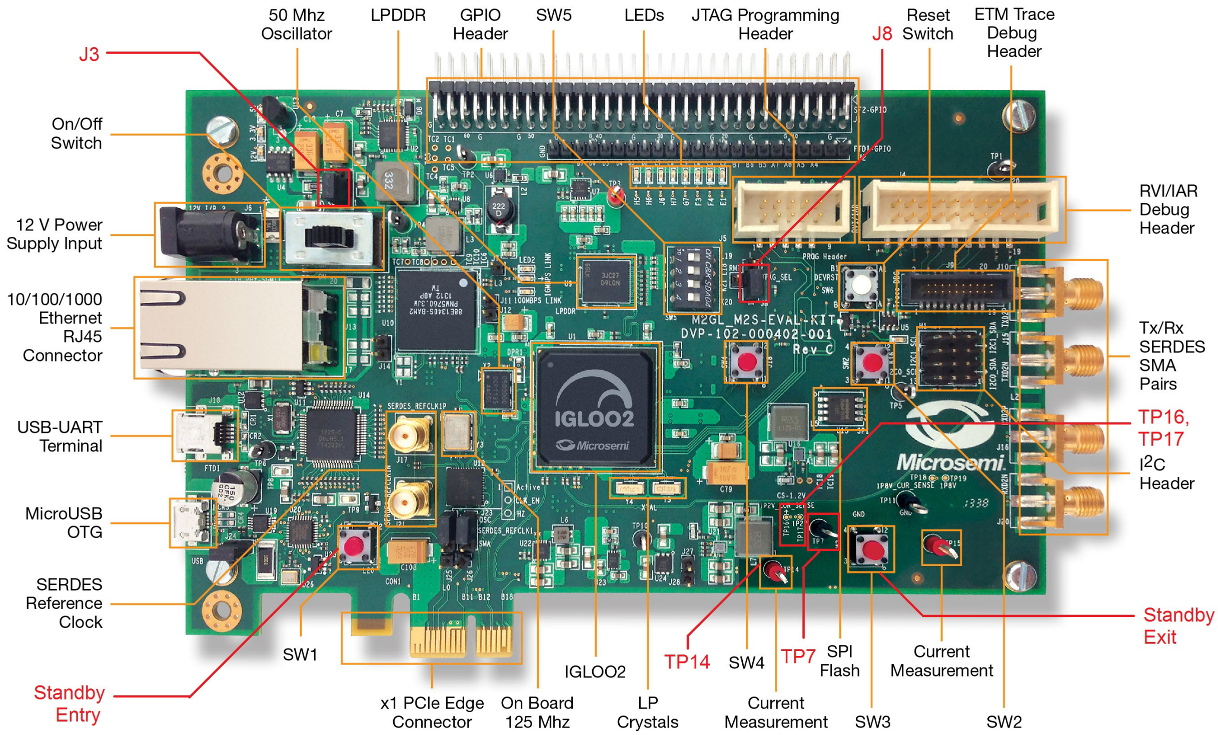

Figure 26 • **IGLOO2 Evaluation Kit**

2. Plug the FlashPro4 ribbon cable into connector J5 (JTAG Programming Header) on the IGLOO2 Evaluation Kit board.
3. Connect FlashPro4 to the USB port of the PC using the mini USB cable.
4. Install the FlashPro4 drivers if prompted. The drivers are located at: *<FlashPro Installation Directory>\Drivers*.
5. Power on the board by plugging in the power cable and switching on the power switch. Three green LEDs on the top left of the board are powered on.

6. In the **Design Flow** tab, expand **Program Design**. Right-click **Run PROGRAM Action** and select **Run** to begin programming.

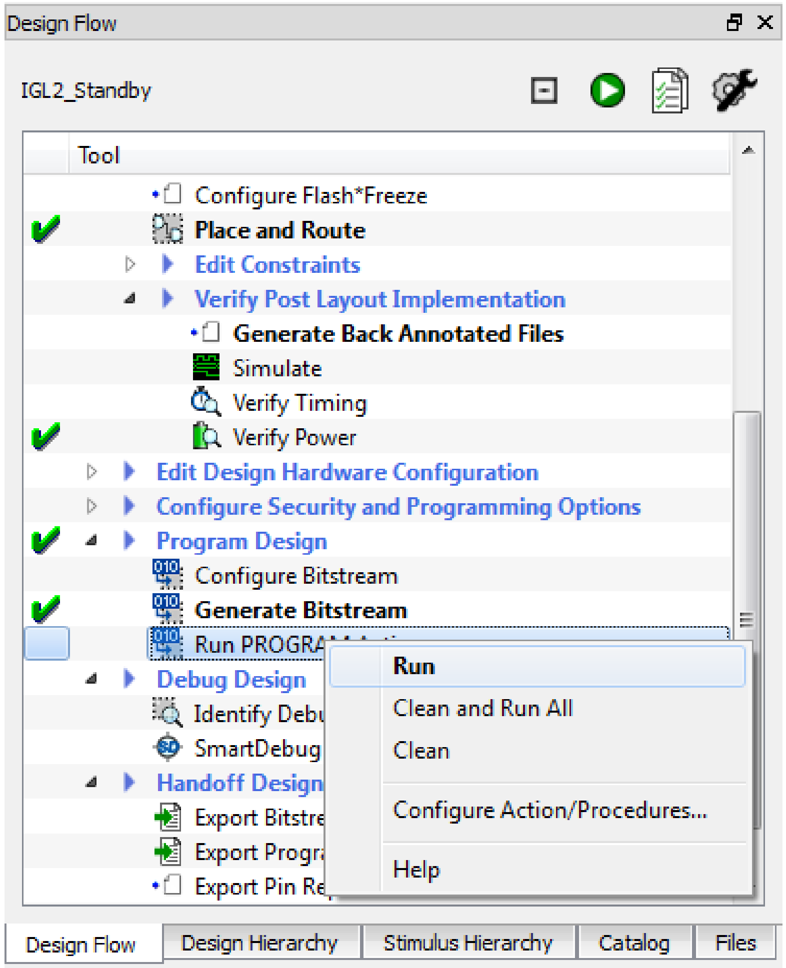

Figure 27 • **Launching Programming Software from Design Flow Tab**

FlashPro runs in the batch mode and programs the device. Programming messages are visible in the Libero SoC log window. Programmer number differs.

Note: Do not interrupt the programming sequence. It may damage the device or programmer.

The following message is displayed in the Reports view under Program Device when the device is programmed successfully as shown in Figure 28 on page 29. Programmer number differs:

```
programmer '92327' : device 'M2GL010T' : Executing action PROGRAM PASSED.
```

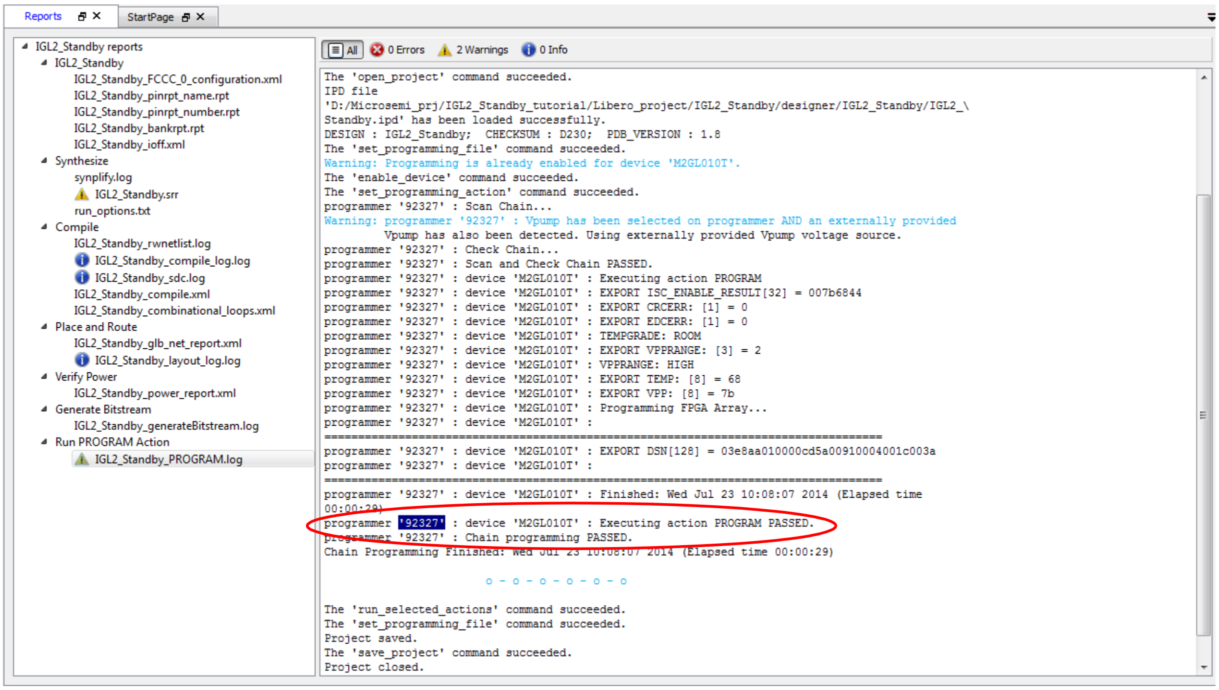

Figure 28 • **Programming Messages in Libero SoC Log Window**

7. A green tick mark appears next to **Program Design and Program Device** in the **Design Flow** tab indicating that programming is completed successfully.

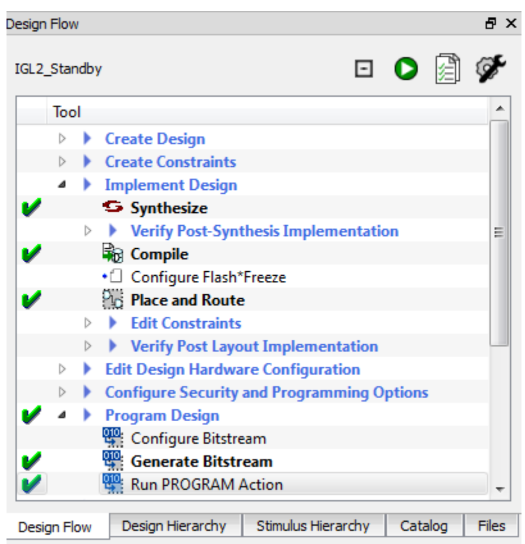

Figure 29 • **Design Flow Tab after Programming**

8. Select **Project > Exit** to close Libero SoC. Select **Yes**, if prompted about saving the changes.

Running the Demo Design

Power Measurement (Normal Operation and Standby)

The IGLOO2 Evaluation Kit board has a voltage measuring circuit that measures the voltage across the VDD (1.2 V) current sense resistor.

The core power can be calculated using the following equations:

$$Core\ \ Current\ \ (mA)\ =\ Measured\ \ Voltage\ \ (mV) \div 5(Scaling\ \ Factor)$$

EQ 1

$$Core\ \ Power\ \ (mW)\ =\ 1.2 \times Core\ \ Current$$

EQ 2

Connect the positive terminal of a standard digital voltmeter (DVM)/multimeter to TP14 and negative terminal to TP7.

Note the digital voltmeter/multimeter reading and calculate the power using the above equations.

Precise Standby Power Measurement

Precise and accurate power measurements can be obtained by measuring voltage across the 1.2 V, 0.05 Ω sense resistor. Test points TP16 and TP17 can be used to directly measure voltage across the 1.2 V sense resistor. Since the current drawn by the device in standby mode is expected to be around or less than 10 mA, the voltage measured across the 0.05 Ω sense resistor is expected to be less than 0.5 mV. A precise digital voltmeter such as Fluke-287 that can measure sub-millivolt readings must be used to read voltage measured across the sense resistor.

Convert the voltage measured across sense resistor to power using the following equation:

$$Power\ \ (mW)\ =\ (Voltage(mV)/0.05) \times 1.2$$

EQ 3

Total Power (Dynamic and Static)

The following steps describe how to calculate total power:

1. Reset the board by pressing and releasing the Reset button (SW6 DEVRST).
2. Observe the pattern of the LEDs E1, F4, F3, and G7 after resetting the board.
3. Measure the power.

Note: If the LEDs are not toggling after reset, the device is in the Standby mode. Press and release the standby exit push button (SW3) and observe the LEDs lighting pattern. When the LEDs start toggling, measure the power.

Standby Power

The following steps describe how to calculate standby power:

1. Press and release the standby entry push button (SW1) and observe the LEDs lighting pattern. The LEDs stop toggling.
2. Measure the power.
3. Press and release the standby exit push button (SW3).
4. When finished, remove power from the board.

Appendix: Power Estimator

Power Estimator

The following steps describe how to use Power Estimator and calculate the total power:

1. Download the Power Estimator, *SmartFusion2 and IGLOO2 Power Calculator.*
2. Double-click and open the power estimator spreadsheet.
3. Click the Summary worksheet. The Summary worksheet provides the device settings and the power summary.
4. Change the device settings by entering the following information:
   – **Family**: Select **IGLOO2** from the drop-down list
   – **Device**: Select **M2GL010T** from the drop-down list
   – **Package**: Select **484 FBGA** from the drop-down list

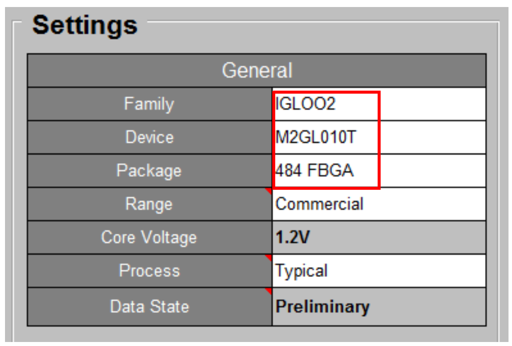

Figure 1 • **Settings Section in the Device Settings and Summary Worksheet**

The Summary worksheet has an integrated initialize power estimator wizard. This wizard provides an option to select design specific information. Upon running the wizard, it populates the power calculator spreadsheet with information about the design and performs power estimation for the design.

5. Click **Initialize Power Estimator** as shown in Figure 2. The **Initialize Power Estimator** dialog box opens as shown in Figure 3 on page 32.

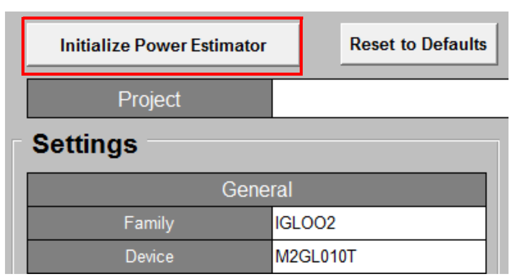

Figure 2 • **Initialize Power Estimator**

6. Enter the following information in the **Initialize Power Estimator** dialog box:

- **Set all FPGA fabric resources to**: 50%
- **uSRAM**: Move the slider to zero, 0/22 (0%)
- **IO**:
  - **Technology**: LVCMOS25
  - **#Inputs**: 21
  - **#Outputs**: 22
- **Default RAM Enable Rate**: 100%

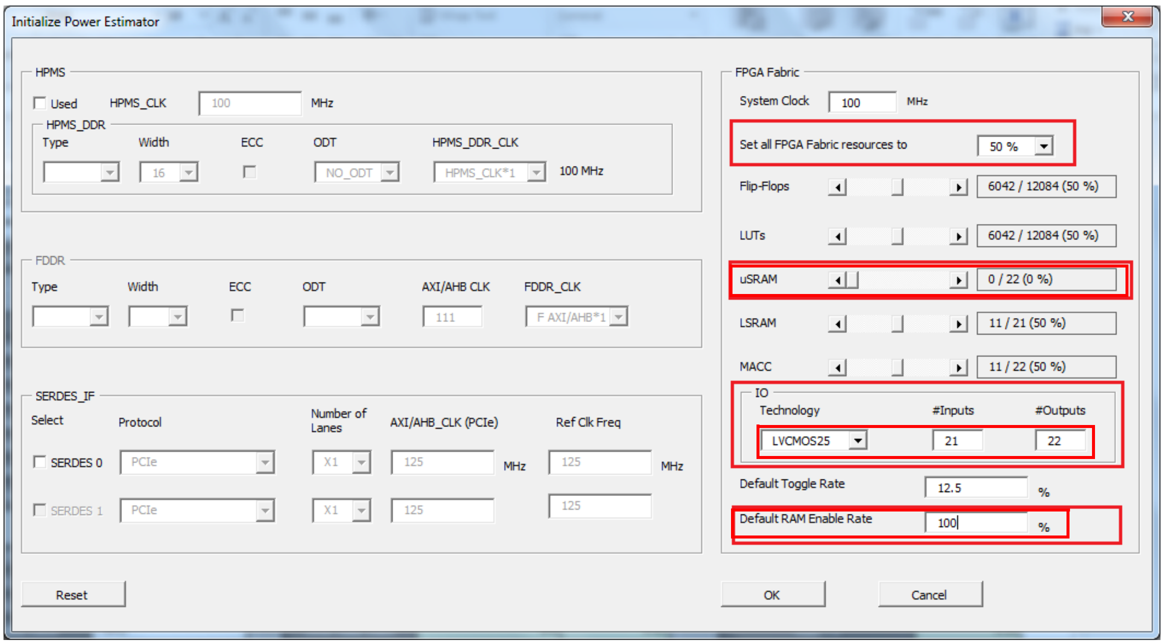

Figure 3 • **Initialize Power Estimator Wizard**

7. Click **OK**. Click **Yes** in the **Reset and set to the values specified** dialog box.
8. Click the **CCC & Oscillator** worksheet and scroll down to the **FAB_CCC Power** section. Enter the following information in the **FAB_CCC Power** table:
   - **Name**: FCC_0
   - **Reference clock frequency (MHz)**: 0.032
   - **PLL output frequency (MHz)**: 500 MHz
   - **Output1 frequency (MHz)**:100 MHz

FAB_CCC Power								
Name	Reference Clock Frequency (MHz)	PLL Output Clock Frequency (MHz)	Output1 Frequency (MHz)	Output2 Frequency (MHz)	Output3 Frequency (MHz)	Output4 Frequency (MHz)	VDD Power (mW)	PLL_VDDA Power (mW)
FCCC_0	0.032	500	100				2.59	5.00
							0.00	0.00
							0.00	0.00
							0.00	0.00
							0.00	0.00
							0.00	0.00
							0.00	0.00
							0.00	0.00

Figure 4 • **FAB_CCC Section**

9. Click the **Summary** worksheet to get the total power. The **Power Summary** section is populated with the Total Active Mode power.

Power Summary

Active Mode: Summary		
Total Power (mW)		179.86
Thermal Power (mW)		179.86
Junction Temperature Tj (°C)		26.08
Effective Theta JA (°C/W)		6.03
Thermal Margin	Maximum Ta (°C)	83.92
	Maximum Power (mW)	9949.04

Figure 5 • **Power Summary**

10. The **Modes and Scenarios** section is populated with the total power in the Active, Standby, and Flash*Freeze modes.

Modes and Scenarios

Low Power Mode Scenario			
Mode	% Time in Mode	Power in Mode (mW)	Power in scenario (mW)
Active	50.00%	179.86	89.93
Standby	0.00%	9.10	0.00
Flash*Freeze	50.00%	4.30	2.15
		Scenario Power	92.08

Figure 6 • **Modes and Scenarios**

11. Close Power Estimator.

A – List of Changes

The following table shows the important changes made in this document for each revision.

Date	Changes	Page
Revision 4 (October 2015)	Updated the document for Libero SoC v11.6 (SAR 71653).	NA
Revision 3 (February 2015)	Updated the document for Libero SoC v11.5 (SAR 64364).	NA
Revision 2 (August 2014)	Updated the document for Libero SoC v11.4.	NA
Revision 1 (October 2013)	Initial release.	NA

B – Product Support

Microsemi SoC Products Group backs its products with various support services, including Customer Service, Customer Technical Support Center, a website, electronic mail, and worldwide sales offices. This appendix contains information about contacting Microsemi SoC Products Group and using these support services.

Customer Service

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, 408.643.6913

Customer Technical Support Center

Microsemi SoC Products Group staffs its Customer Technical Support Center with highly skilled engineers who can help answer your hardware, software, and design questions about Microsemi SoC Products. The Customer Technical Support Center spends a great deal of time creating application notes, answers to common design cycle questions, documentation of known issues, and various FAQs. So, before you contact us, please visit our online resources. It is very likely we have already answered your questions.

Technical Support

For Microsemi SoC Products Support, visit http://www.microsemi.com/products/fpga-soc/design-support/fpga-soc-support.

Website

You can browse a variety of technical and non-technical information on the Microsemi SoC Products Group home page, at http://www.microsemi.com/products/fpga-soc/fpga-and-soc.

Contacting the Customer Technical Support Center

Highly skilled engineers staff the Technical Support Center. The Technical Support Center can be contacted by email or through the Microsemi SoC Products Group website.

Email

You can communicate your technical questions to our email address and receive answers back by email, fax, or phone. Also, if you have design problems, you can email your design files to receive assistance. We constantly monitor the email account throughout the day. When sending your request to us, please be sure to include your full name, company name, and your contact information for efficient processing of your request.

The technical support email address is soc_tech@microsemi.com.

My Cases

Microsemi SoC Products Group customers may submit and track technical cases online by going to My Cases.

Outside the U.S.

Customers needing assistance outside the US time zones can either contact technical support via email (soc_tech@microsemi.com) or contact a local sales office. Visit About Us for sales office listings and corporate contacts.

ITAR Technical Support

For technical support on RH and RT FPGAs that are regulated by International Traffic in Arms Regulations (ITAR), contact us via soc_tech@microsemi.com. Alternatively, within My Cases, select **Yes** in the ITAR drop-down list. For a complete list of ITAR-regulated Microsemi FPGAs, visit the ITAR web page.

Microsemi Corporation (Nasdaq: MSCC) offers a comprehensive portfolio of semiconductor and system solutions for communications, defense & security, aerospace 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; security technologies and scalable anti-tamper products; Ethernet Solutions; Power-over-Ethernet ICs and midspans; as well as custom design capabilities and services. Microsemi is headquartered in Aliso Viejo, Calif., and has approximately 3,600 employees globally. Learn more at **www.microsemi.com**.

Microsemi Corporate 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

E-mail: sales.support@microsemi.com

50200564-4/10.15