Interfacing User Logic with the Microcontroller Subsystem - Libero SoC v11.7 Design Flow

TU0310 Tutorial
Contents

1 Preface ................................................................. 2
  1.1 Purpose ........................................................... 2
  1.2 Intended Audience .............................................. 2
  1.3 References ...................................................... 2

2 Interfacing User Logic with the Microcontroller Subsystem - Libero SoC v11.7 Design Flow ....................................................... 3
  2.1 Introduction ..................................................... 3
  2.2 Design Requirements .......................................... 4
    2.2.1 Design Files ............................................. 4
  2.3 Design Description ............................................ 4
    2.3.1 Design Steps ............................................ 5
  2.4 Step 1: Creating a New Libero SoC Project .................. 6
  2.5 Step 2: Modifying User BFM Script for Simulation ....... 21
    2.5.1 Generating Testbench .................................. 22
  2.6 Step 3: Simulating Design Using BFM Models .............. 23
  2.7 Step 4: Generating Programming File ....................... 29
  2.8 Step 5: Programming the SmartFusion2 Board Using FlashPro ............................................. 32
    2.8.1 Jumper Settings for SmartFusion2 Security Evaluation Kit Board ............................... 32
    2.8.2 Jumper Settings for SmartFusion2 Starter Kit Board .................................................. 32
    2.8.3 Programming the Device ................................ 33
  2.9 Step 6: Building the Software Application through SoftConsole ............................................... 36
  2.10 Step 7: Configuring the Serial Terminal Emulation Program .................................................. 40
  2.11 Step 8: Debugging the Application Project using SoftConsole ................................................. 42
  2.12 Step 9: Building Executable Image in Release mode ... 47
  2.13 Conclusion ....................................................... 47

3 Revision History .................................................... 50

4 Product Support ................................................... 51
  4.1 Customer Service ............................................... 51
  4.2 Customer Technical Support Center .......................... 51
  4.3 Technical Support ................................................ 51
  4.4 Website .......................................................... 51
  4.5 Contacting the Customer Technical Support Center ... 51
    4.5.1 Email ......................................................... 51
    4.5.2 My Cases ................................................... 51
    4.5.3 Outside the U.S. ........................................... 52
  4.6 ITAR Technical Support ......................................... 52
Figures

Figure 1. Block Diagram of the Design .................................................. 5
Figure 2. Libero SoC New Project Dialog Box ........................................ 6
Figure 3. New Project - Device Selection ............................................... 7
Figure 4. New Project - Device Selection ............................................... 7
Figure 5. New Project - Design Template .............................................. 8
Figure 6. Using Enhanced Constraints Flow .......................................... 8
Figure 7. Libero SoC Project Window .................................................... 9
Figure 8. SmartFusion2 System Builder Device Features .......................... 9
Figure 9. SmartFusion2 System Builder Peripherals (M2S090TS Device) .... 10
Figure 10. SmartFusion2 System Builder MSS Peripherals (M2S090TS Device) ... 11
Figure 11. SmartFusion2 System Builder MSS Peripherals (M2S010T Device) ... 12
Figure 12. CoreAHBL0SRAM Configuration ........................................ 12
Figure 13. CoreAHBL0SRAM Configuration ........................................ 13
Figure 14. CoreGPIO Configuration ................................................... 14
Figure 15. CoreGPIO Configuration ................................................... 15
Figure 16. MM_UART Configuration .................................................. 15
Figure 17. SmartFusion2 System Builder Block Clocks ............................ 16
Figure 18. SmartFusion2 System Builder CoreAHBL0t Address Map (M2S090T Device) .......................... 17
Figure 19. SmartFusion2 System Builder CoreAPB Address Map ............... 18
Figure 20. SmartFusion2 System Builder Component .............................. 18
Figure 21. SmartFusion2 System Builder Block ...................................... 19
Figure 22. SmartFusion2 System Builder Block ...................................... 20
Figure 23. Generate Component ..................................................... 20
Figure 24. Log Window ..................................................................... 21
Figure 25. Import bfm file .................................................................. 21
Figure 26. user.bfm file .................................................................... 22
Figure 27. HDL Testbench .................................................................. 22
Figure 28. Create New HDL Testbench File .......................................... 23
Figure 29. wave-do File ..................................................................... 24
Figure 30. Project Settings - Do File ................................................... 25
Figure 31. Project Settings - Waveforms ............................................. 25
Figure 32. Design Flow - Verify Pre-Synthesized Design ....................... 26
Figure 33. ModelSim Transcript Window - BFM Commands .................. 27
Figure 34. Design Simulation Waveforms ........................................... 28
Figure 35. Constraints Manager ....................................................... 29
Figure 36. Importing I/O Constraints .................................................. 29
Figure 37. Managing I/O Constraints .................................................. 30
Figure 38. Derive Constraints ........................................................... 30
Figure 39. Selecting Timing Constraint for Synthesis, Place And Route, and Timing Verification .................. 30
Figure 40. Constraint Manager .......................................................... 30
Figure 41. Verify Timing ................................................................... 31
Figure 42. Generating Bitstream ....................................................... 31
Figure 43. Run PROGRAM Action ..................................................... 33
Figure 44. SmartFusion2 Security Evaluation Kit Setup ......................... 34
Figure 45. SmartFusion2 Security Evaluation Kit Setup ......................... 35
Figure 46. Handoff Design for Firmware Development ......................... 36
Figure 47. Design Firmware ............................................................ 36
Figure 48. Select Export Firmware ..................................................... 37
Figure 49. Export Firmware ............................................................. 37
Figure 50. Export Firmware-Information ............................................ 38
Figure 51. SoftConsole Workspace Launcher ....................................... 38
Figure 52. Select Workspace Directory .............................................. 38

Revision 13
Figure 53.  SoftConsole IDE ................................................................. 39
Figure 54.  Clean and Build window .............................................. 40
Figure 55.  SmartFusion2 Security Evaluation Kit USB Serial Port Drivers ........................................ 41
Figure 56.  SmartFusion2 Starter Kit USB Serial Port Drivers .................................................................. 42
Figure 57.  Debug Window ................................................................. 43
Figure 58.  Confirm Perspective Switch ............................................. 43
Figure 59.  Debug Perspective ............................................................ 44
Figure 60.  Tera Term Window ............................................................ 44
Figure 61.  Writing to LSRAM ............................................................. 45
Figure 62.  Writing to LSRAM ............................................................. 45
Figure 63.  Reading from LSRAM ....................................................... 46
Figure 64.  Selecting GPIO LED Blinking ......................................... 46
Figure 65.  Selecting GPIO LED Blinking ......................................... 47
Tables

Table 1. Design Requirements ................................................................. 4
Table 2. Jumper Settings for Security Evaluation Kit Board ......................... 32
Table 3. Jumper Settings for Starter Kit Board ......................................... 32
1 Preface

1.1 Purpose
This tutorial describes how to interface and handle communication between user logic in the field programmable gate array (FPGA) fabric and the SmartFusion®2 microcontroller subsystem (MSS). It also explains the Microsemi Libero® System-on-Chip (SoC) design software tool flow for designing applications for the SmartFusion2 SoC FPGA family of devices.

1.2 Intended Audience
This tutorial is intended for:
- FPGA designers
- System-level designers

1.3 References
The following documents are referred in this tutorial:
- UG0331: SmartFusion2 Microcontroller Subsystem User Guide
- CoreAMBA BFM User Guide
- Starter Kit Guide
- UG0594: SmartFusion2 Security Evaluation Kit User Guide
- Configuring Serial Terminal Emulation Programs Tutorial

See the following web page for a complete and up-to-date listing of SmartFusion2 device documentation:

2 Interfacing User Logic with the Microcontroller Subsystem - Libero SoC v11.7 Design Flow

2.1 Introduction

A SmartFusion2 device has two fabric interface controllers (FIC_0 and FIC_1) as a part of the MSS. These FIC blocks provide a means of interfacing from the SmartFusion2 MSS AHB-Lite (AHBL) bus to user masters or user slaves in the FPGA fabric. Each FIC block performs an AHBL to AHBL or AHBL to APB3 bridging function between the AHB Bus Matrix and AHBL or APB3 bus in the FPGA fabric. Each FIC block provides two bus interfaces between the MSS and FPGA fabric. The first one is mastered by the MSS and has slaves in the FPGA fabric; the second one has a master in the fabric and slaves in the MSS. The bus interfaces to the FPGA fabric can be either 32-bit AHBL or 32-bit APB type. The FIC block provides registered bridging between the MSS AHBL interface and the FPGA fabric AHBL/APB circuitry to run at frequency ratios of 1:1, 2:1, 4:1, 8:1, 16:1, or 32:1. In AHB-Lite configuration, a bypass mode is provided, in which signals to and from the fabric are not registered and hence requires fewer clock cycles to complete each transaction. SmartFusion2 FIC has six memory regions. You can allocate a memory region to a particular FIC that is either to FIC_0 or FIC_1. Each memory region has a predefined memory map. For more information on FIC blocks in the Fabric Interface Controller, see the UG0331: SmartFusion2 Microcontroller Subsystem User Guide.

This tutorial covers the following:

1. Creating a project for a SmartFusion2 SoC FPGA using the Microsemi Libero SoC tool set.
2. Using SmartFusion2 System Builder to Configure MSS and generate System Builder Component.
3. Configuring fabric interface controllers (FIC_0) to interface user logic in the fabric with the MSS.
4. Using on-chip oscillators and fabric CCC (FAB_CCC) for generating system clocks.
5. Writing a simple bus functional model (BFM) script for simulating the design.
6. Verifying the design by running BFM commands.
7. Generating the programming file to program the SmartFusion2 device.
8. Opening the project in SoftConsole from Libero SoC and writing the application code.
9. Validating the application design on SmartFusion2 board.
2.2 Design Requirements

Table 1 • Design Requirements

<table>
<thead>
<tr>
<th>Design Requirements</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Hardware Requirements</strong></td>
<td></td>
</tr>
<tr>
<td>• SmartFusion2 Security Evaluation Kit Board or</td>
<td>Rev D or later</td>
</tr>
<tr>
<td>• SmartFusion2 Starter Kit Board (SF2-STARTER-KIT)</td>
<td></td>
</tr>
<tr>
<td>• FlashPro4 programmer</td>
<td></td>
</tr>
<tr>
<td><strong>USB Cables</strong></td>
<td>–</td>
</tr>
<tr>
<td><strong>Desktop Computer or Laptop</strong></td>
<td>Any 64-bit Windows Operating System</td>
</tr>
<tr>
<td><strong>Software Requirements</strong></td>
<td></td>
</tr>
<tr>
<td>Libero SoC</td>
<td>v11.7</td>
</tr>
<tr>
<td>SoftConsole</td>
<td>v3.4 SP1*</td>
</tr>
<tr>
<td>Host PC Drivers</td>
<td>USB Drivers</td>
</tr>
</tbody>
</table>

**Note:** *For this application note, SoftConsole v3.4 SP1 is used. For using SoftConsole v4.0, see the TU0546 SoftConsole v4.0 and Libero SoC v11.7 Tutorial.*

2.2.1 Design Files

You can download the associated project files for this tutorial from the Microsemi website: http://soc.microsemi.com/download/rsc/?f=m2s_tu0310_liberov11p7_df

The project files include the following:

• Source
• Solution
• Programming File
• Readme file

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

2.3 Design Description

The design uses the SmartFusion2 MSS block, one CCC block, on-chip 25/50 MHz RC oscillator and two different slaves in the FPGA fabric. The MSS FIC_0 is configured for the AHBL master interface is connected to the slaves CoreAHBLSRAM and CoreGPIO using the CoreAHBLite, CoreAHBTOAPB, and CoreAPB3 bus interfaces. Figure 1 shows the block diagram of the design. The ARM® Cortex®-M3 processor or any other MSS master can access these slaves through the FIC blocks. In this design, you can:

• Verify the bus read and write to the fabric peripherals from the MSS side using BFM models.
• Perform read and write to the CoreAHBLSRAM memory, configure the CoreGPIO block, and set GPIO outputs using a BFM script.
• Validate the bus read and write to the CoreAHBLSRAM, set the GPIO to blink the LEDs on the SmartFusion2 Security Evaluation Kit board and SmartFusion2 Starter Kit board.
2.3.1 Design Steps

The major steps to run this tutorial are as follows:

1. Creating a new Libero SoC project for the SmartFusion2 device.
2. Using SmartFusion2 System Builder to configure the FIC blocks and clock.
3. Writing user BFM script to simulate a design.
4. Simulating the design using BFM Models and ModelSim.
5. Generate a programming file to program the SmartFusion2 SoC FPGA device.
6. Open the software project in SoftConsole and write the application program.
7. Run the design on the SmartFusion2 Security Evaluation Kit board or SmartFusion2 Starter Kit board.
2.4 **Step 1: Creating a New Libero SoC Project**

The following steps describe how to create a new Libero SoC project:

1. Open Libero SoC design software (**Start > Programs > Microsemi Libero SoC 11.7 > Libero SoC 11.7**) or click the Libero SoC shortcut available on your desktop. The version number of the Libero SoC design software depends on the version that is installed on your PC. You can use either v11.7 or latest.
2. Select **New Project** from the **Project** menu.
3. Enter the following **New Project** information as shown in and click **Next**.
   - **Project Name**: SmartFusion2_FIC_Tutorial
   - **Project Location**: Select an appropriate location (for example, D:/Microsemi_prj)
   - **Preferred HDL Type**: Verilog

*Figure 2 • Libero SoC New Project Dialog Box*

4. Select the following values from the drop-down lists for **Device Selection**, as shown in Figure 3.
   - **Family**: SmartFusion2
   - **Die**: M2S090TS
   - **Package**: 484 FBGA
   - **Speed**: -1
   - **Core Voltage**: 1.2
   - **Range**: COM
5. Click Next.
6. Select the information for Device Settings as shown in Figure 4 and click Next.

7. Select Create a System Builder based design under Design Templates and Creators as shown in Figure 5
8. Click Finish. The New Project Information window is displayed as shown in Figure 6. This tutorial uses the enhanced constraints flow of Libero v11.7, which simplifies the management of all constraints (I/O, timing, floor planning and Netlist optimization constraints).

9. Select Use Enhanced Constraints Flow as shown in Figure 6.

10. The System Builder dialog box is displayed. Enter a name for your system as shown in Figure 7.
11. Enter **SmartFusion2_FIC_Tutorial** as the name of the system and click **OK**. The **System Builder** window is displayed, as shown in **Figure 8**.

**Figure 8 • SmartFusion2 System Builder Device Features**

12. Select **Next**, **System Builder- Peripherals** page is displayed, as shown in **Figure 9** and **Figure 10** on page 11. This tutorial uses the MSS MMUART peripherals.
13. Select **MM_UART_1** for SmartFusion2 Security Evaluation Kit (**M2S090TS** device) or **MM_UART_0** for SmartFusion2 Starter Kit (**M2S010** device) and clear the check boxes for all the other peripherals, as shown in Figure 10 on page 11 and Figure 11 on page 12.

14. Drag the **CoreAHBLSRAM** and **CoreGPIO** IPs to **MSS FIC_0 - MSS Master Subsystem** for **M2S090TS** device as shown in Figure 10. This tutorial uses the **CoreAHBLSRAM** and **CoreGPIO** IPs.
15. Drag the CoreAHBLRAM and CoreGPIO IPs to MSS FIC_0 - MSS Master Subsystem for M2S010T device, as shown in Figure 11.
16. Configure COREAHBLSRAM_0 by clicking the Configure icon, as shown in Figure 12.

*Figure 12 • CoreAHBLSRAM Configuration*

Use the settings as shown in Figure 13.
17. Click **OK** after completion of COREAHBLSRAM configuration.
18. Click **CoreGPIO Configure** icon and use the following settings for SmartFusion2 Security Evaluation Kit board as shown in **Figure 14** on page 14, and keep the rest at default states
   - **Number of I/Os**: 8 - For SmartFusion2 Security Evaluation Kit board
   - **Output enable**: Internal
   - **Fixed Config**: Select the check box
   - **I/O Type**: Output

**Note**: For SmartFusion2 Starter Kit, select Number of I/Os: 2.
Note: For SmartFusion2 Starter Kit board, the CoreGPIO Configuration window is displayed, as shown in Figure 15.
19. Click **OK** after completion of CoreGPIO configuration.
20. Double-click the **MM_UART_1** configure icon for M2S090TS device and **MM_UART_0** configure icon for M2S010T device.
21. Select **IO** from the **Connect To** drop-down list and retain the default settings, as shown in Figure 16.

**Figure 16 • MM_UART Configuration**

22. Click **OK**.
23. Select **Next**. **System Builder- Clock Settings** page is displayed, as shown in Figure 17. Select the following options:
   - **System Clock**: Set it to On-chip 25/50 MHz RC Oscillator from the drop-down list.
   - **M3_CLK**: 100 MHz
   - **MSS APB_0/1 Clocks**: 100 MHz
   - **Fabric Interface Clocks**: 100 MHz
24. Click **Next**, the System Builder - **Microcontroller Options** page is displayed.
   - Leave all the Default Selections.
25. Click **Next**, the System Builder - **SECDED Options** page is displayed.
   - Leave all the Default Selections.
26. Click **Next**, the System Builder - **Interrupts Options** page is displayed.
   - Leave all the Default Selections.
27. Click **Next**, the System Builder - **Memory Map Options** page is displayed.
   - Leave all the Default Selections.

*Figure 18* shows the address map for AHBL peripherals.
Figure 18 • SmartFusion2 System Builder CoreAHBLite Address Map (M2S090T Device)
Figure 19 shows the address map for APB3 peripheral.

**Figure 19** • SmartFusion2 System Builder CoreAPB Address Map

![SmartFusion2 System Builder CoreAPB Address Map](image)

28. Click **Finish**.

The System Builder generates the system based on the selected options.

The System Builder block is created and added to Libero SoC project, as shown in Figure 20.

**Figure 20** • SmartFusion2 System Builder Component

![SmartFusion2 System Builder Component](image)

To initialize a user design in the SmartFusion2 devices, Microsemi provides a CoreResetP soft Reset Controller IP. The CoreResetP IP handles a sequence of reset signals in the SmartFusion2 devices. The CoreResetP does automatically be instantiated and connected by the System Builder.

Open the System Builder component in the Smart Design canvas to view how these blocks are connected.
29. Connect the pins as follows:
   - Right-click FAB_RESET_N and select Tie High.
   - To select POWER_ON_RESET_N and MSS_READY pins, hold CTRL key, select pins, right-click and select Mark Unused.
   - Expand INIT_PINS, right-click INIT_DONE and select Mark Unused.
   - Expand FAB_CCC_PINS, right-click FAB_CCC_GL0 and FAB_CCC_LOCK and select Mark Unused.
   - Expand CoreGPIO_0_0_PINS,
     - Mark the INT[7:0] PINS as unused by right-clicking and selecting Mark Unused.
     - Tie the GPIO_IN[7:0] to high by right-clicking and selecting Tie High.
     - Promote the GPIO_OUT[7:0] to top by right-clicking and selecting Promote to Top Level.

After connecting the pins, the System Builder block is displayed, as shown in Figure 21.

Figure 21 • SmartFusion2 System Builder Block

Note: The System Builder block for SmartFusion2 Starter Kit is shown in Figure 22.
30. Click **Generate Component** icon on the SmartDesign toolbar or right-click on the canvas and select **Generate Component**.

After successful generation of the system, the message **SmartFusion2_FIC_Tutorial was successfully generated** is displayed in the Libero SoC log window if the design is generated without any errors. The log window is displayed, as shown in **Figure 24**.
2.5 Step 2: Modifying User BFM Script for Simulation

Verify the design by using the BFM master or slave model and a BFM script to drive the AHBL/APB input of the DUT. This setup allows the BFM to write or read to the AHBL/APB register set and to verify that the DUT is behaving as expected.

This step explains adding BFM commands to the user.bfm file to perform design simulation. For more information on BFM commands, see the CoreAMBA BFM User Guide. The user.bfm file is created by Libero SoC Design software and is available in the simulation folder of the project files.

**Note:** Download the project files. Refer to the "Design Files" section on page 4.

1. For SmartFusion2 Security Evaluation Kit board, right-click the simulation under project files and select import files to import the user.bfm file which is located in downloaded design files (\SF2_FIC_Tutorial\Source\For_SF2_Eval_Kit_Board\user.bfm) as shown in Figure 25 or select Files > Import > Others to import the user.bfm file.
2. Click Yes to all to replace the existing user.bfm file.

**Note:** For SmartFusion2 Starter Kit, import the user.bfm to simulation files from design files. (\SF2_FIC_Tutorial\Source\For_SF2_Starter_Kit_Board\user.bfm)

3. After importing, double-click the user.bfm file under simulation folder. This opens the user.bfm file as a new tab in the project window, as shown in Figure 26.
2.5.1 Generating Testbench

1. From the File menu, select New > HDL Testbench as shown in Figure 27.

Figure 27 • HDL Testbench

The Create New HDL Testbench File dialog box is displayed
2. Select **HDL Type** as **Verilog** or **VHDL**.
3. Enter **Name** as testbench in the text box and retain the default settings.
4. Enter **Clock Period (ns)** as 10.
5. Click **OK**.

### 2.6 Step 3: Simulating Design Using BFM Models

This section describes how to use the testbench and BFM script file to simulate the design.

1. Add the **wave.do** file to the SmartFusion2_FIC_Tutorial design simulation folder by clicking **File > Import > Others**.
2. Browse to the **wave.do** file location in the design files folder:
   `SF2_FIC_Tutorial\Source\For_SF2_Eval_Kit_Board`. **Figure 29** shows the **wave.do** file under simulation folder in the Files window.
Note: For SmartFusion2 Starter Kit board, browse to the `wave.do` file located in the downloaded design files folder: `SF2_FIC_Tutorial\Source\For_SF2_Starter_Kit_Board`.

3. Set up the simulation environment as follows:
   Select Project > Project Settings. On the Project Settings window, under Simulation Options, select DO File to change the simulation run time, enter 50us in the Simulation runtime field, as shown in Figure 30.
4. Save the Do File configuration, this can be done by clicking the Save.
5. Select Waveforms under Simulation Options as shown in Figure 31:
   a. Select Include Do file.
   b. Select Log all signals in the design check box.
   c. Click Close to close the Project settings dialog box.
   d. Select Save when prompted to save the changes.

Note: You can also add ports or signals of interest in the ModelSim software.
6. Select the **Design Flow** tab in the project window.

7. Expand the **Verify Pre-Synthesized Design**, as shown in Figure 32. Double-click **Simulate** to invoke ModelSim. After invoking ModelSim, the design is loaded. Alternatively, right-click the **Simulate** and select **Open Interactively**

*Figure 32 • Design Flow – Verify Pre-Synthesized Design*
8. Maximize the **ModelSim Transcript** window to see the BFM commands execution. Ensure that there are no errors. **Figure 33** shows the ModelSim Transcript window.

**Figure 33** • **ModelSim Transcript Window – BFM Commands**
After successful BFM simulation, observe the ModelSim waveform window for the read and write bus transactions to the fabric peripherals, as shown in Figure 34.

9. Notice the result of GPIO configuration BFM commands in GPIO states.

**Figure 34 • Design Simulation Waveforms**

10. Quit the ModelSim simulator by selecting **File > Quit**.
2.7 **Step 4: Generating Programming File**

The following steps describe how to generate a program file:

1. Expand **Constraints**, as shown in Figure 35, and double-click **Manage Constraints** to add the I/O and Timing constraints.

*Figure 35 • Constraints Manager*

2. Click **Import** as shown in Figure 36, to add the I/O constraints.

*Figure 36 • Importing I/O Constraints*
3. Browse the SmartFusion2_FIC_Tutorial.io.pdc file in the design files folder, and select the check box as shown in Figure 37 to select the constraint for place-and-route.

![Figure 37 • Managing I/O Constraints](image)

4. Select the **Timing** tab in **Constraints Manager**, and click **Derive constraints** as shown in Figure 38, to generate the timing constraints for the design.

![Figure 38 • Derive Constraints](image)

5. Select **Yes** in the **Message** dialog box as shown in Figure 39.

![Figure 39 • Selecting Timing Constraint for Synthesis, Place And Route, and Timing Verification](image)

6. The **Constraint Manager** is displayed as shown in Figure 40.

![Figure 40 • Constraint Manager](image)

7. In the **Design Flow** window, under **Verify Post Layout Implementation** double-click **Verify Timing** as shown in Figure 41, to verify the design timing.
8. Under Program Design click Generate Bitstream to generate the programming file, as shown in Figure 42.

Figure 41 • Verify Timing

Figure 42 • Generating Bitstream
2.8 Step 5: Programming the SmartFusion2 Board Using FlashPro

Before programming the device, ensure that FlashPro4 programmer is properly connected to the FlashPro Header of board. Use the following details to ensure the correct jumper settings. For more information, see the Starter Kit Guide and UG0594: SmartFusion2 Security Evaluation Kit User Guide.

2.8.1 Jumper Settings for SmartFusion2 Security Evaluation Kit Board

Connect the jumpers on the SmartFusion2 Security Evaluation Kit, as shown in Table 2. Switch OFF the power supply switch while connecting the jumper.

Table 2 • Jumper Settings for Security Evaluation Kit Board

<table>
<thead>
<tr>
<th>Jumper</th>
<th>Pin (From)</th>
<th>Pin (To)</th>
</tr>
</thead>
<tbody>
<tr>
<td>J3, J8</td>
<td>1 (default)</td>
<td>2</td>
</tr>
</tbody>
</table>

2.8.2 Jumper Settings for SmartFusion2 Starter Kit Board

Connect the jumpers on the SmartFusion2 Starter Kit, as shown in Table 3.

Table 3 • Jumper Settings for Starter Kit Board

<table>
<thead>
<tr>
<th>Designation</th>
<th>Name</th>
<th>Settings</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>JP1</td>
<td>VCC3</td>
<td>1-2 Closed</td>
<td>The +3.3 V voltage from the output of the U2 LDO regulator is applied to the SOM and to the SOM-BSB-EXT.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3-4 Open</td>
<td>The +3.3 V voltages from the output of the U2 LDO regulator are not applied to the D1 double diode ORing scheme.</td>
</tr>
<tr>
<td>JP2</td>
<td>JTAG Mode Selection</td>
<td>1-2 Open</td>
<td>The SmartFusion2 JTAG controller is in the FPGA programming mode.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3-4 Closed</td>
<td>The settings of jumpers 3-4 do not affect M2S-SOM.</td>
</tr>
<tr>
<td>JP3</td>
<td>VCC5</td>
<td>1-3 Open</td>
<td>The +3.3 V LDO regulator is powered from the +5 V USB power through the P1 mini USB connector.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2-4 Closed</td>
<td></td>
</tr>
</tbody>
</table>
2.8.3 Programming the Device

Double-click the Run PROGRAM Action under Program Design in the Design Flow window as shown in Figure 43 to program the SmartFusion2 SoC FPGA device.

Figure 43 • Run PROGRAM Action

Figure 44 shows the board setup for running the application design on the SmartFusion2 Security Evaluation Kit board.
Figure 45 shows the board setup for running the application design on the SmartFusion2 Starter Kit board.
Note: Do not interrupt the programming sequence; it may damage the device or the programmer. If you face any problems, contact Microsemi Tech Support at soc_tech@microsemi.com.
2.9 Step 6: Building the Software Application through SoftConsole

The following steps describe how to build the software application:

1. In the Design Flow window, double-click Configure Firmware Cores under Handoff Design For Firmware Development as shown in Figure 46, to select the firmware drivers for software application.

![Figure 46 • Handoff Design for Firmware Development](image)

2. The Design Firmware window displays compatible firmware drivers based on the peripherals configured in the design. The following drivers are used in this tutorial:
   - CMSIS
   - CoreGPIO
   - MMUART

3. In the DESIGN_FIRMWARE tab, clear all the driver check boxes, except CMSIS, CoreGPIO and MMUART as shown in Figure 47.
   
   **Note:** Select the latest version of the drivers.

![Figure 47 • Design Firmware](image)

4. Click the Download all firmware button highlighted in red to download the latest version of drivers for peripherals as shown in Figure 47. Close the DESIGN_FIRMWARE tab.

6. Right-click and select Export Firmware… as shown in Figure 48.

*Figure 48 • Select Export Firmware*

7. The Export Firmware dialog box is displayed as shown in Figure 49. Enter the following information in the Export Firmware dialog box:
   - Browse to the Location such as `<C:\ or D:\Microsemi_prj\SmartFusion2_FIC_Tutorial>.
   - Select the Create project check box and select SoftConsole3.4 from the drop-down list.

*Figure 49 • Export Firmware*

8. Click OK. The Information dialog box is displayed as shown in Figure 50.
9. Click **OK**.
10. Click **Start > Programs > Microsemi SoftConsole v3.4 > Microsemi SoftConsole IDE v3.4** or double-click the shortcut icon on your desktop. The **SoftConsole Workspace Launcher** is displayed as shown in Figure 50.

**Figure 50 • Export Firmware-Information**

![Figure 50](image)

9. Click **OK**.
10. Click **Start > Programs > Microsemi SoftConsole v3.4 > Microsemi SoftConsole IDE v3.4** or double-click the shortcut icon on your desktop. The **SoftConsole Workspace Launcher** is displayed as shown in Figure 50.

**Figure 51 • SoftConsole Workspace Launcher**

![Figure 51](image)

11. Navigate to the **SoftConsole folder** and select **projects** folder as shown in Figure 52.

**Figure 52 • Select Workspace Directory**

![Figure 52](image)

12. Click **OK**.
   The **SoftConsole IDE** window is displayed as shown in Figure 53.
13. Go to the source folder in the downloaded design files folder, copy the code from the `Source_eval.c` file. In SoftConsole editor under `SmartFusion2_FIC_Tutorial_MSS_CM3_app` project, place the copied code in the `main.c` file and delete the existing code.

**Note:** For the SmartFusion2 Starter Kit board, the Code provided in `source_starter.c` in source files.

14. Select **Project > Clean** to perform a clean build. Accept the default settings in the **clean** dialog box and click **OK**, as shown in Figure 54.
15. Ensure that there are no errors.

2.10 Step 7: Configuring the Serial Terminal Emulation Program

Prior to running the application program, configure the terminal emulator program on your PC. Perform the following steps to use the SmartFusion2 Security Evaluation Kit board or SmartFusion2 Starter Kit board:

1. Connect one end of the USB mini-B cable to the respective USB connector provided on the SmartFusion2 board.
2. Connect the other end of the USB cable to the host PC. Ensure that the USB to UART bridge drivers are automatically detected, as shown in Figure 55 and Figure 56 on page 42.
**Figure 55**  •  SmartFusion2 Security Evaluation Kit USB Serial Port Drivers
3. If USB to UART bridge drivers are not installed, download and install the drivers from www.microsemi.com/soc/documents/CDM_2.08.24_WHQL_Certified.zip.

4. Start a terminal emulator program with the baud rate set to 57600, 8 data bits, 1 stop bit, no parity, and no flow control. Refer to the Configuring Serial Terminal Emulation Programs Tutorial for configuring HyperTerminal, TeraTerm, and PuTTY.

2.11 Step 8: Debugging the Application Project using SoftConsole

Use the following steps to debug the application project using SoftConsole:

1. Select SmartFusion2_FIC_Tutorial_sb_MSS_CM3_app in Project Explorer.
2. Select the Debug Configurations from the Run menu of the SoftConsole. The Debug dialog is displayed.
3. Double-click on Microsemi Cortex-M3 Target to display an image similar to Figure 57.
4. Confirm that the following appear on the Main tab in the Debug window:
   - **Name**: SmartFusion2_FIC_Tutorial_sb_MSS_CM3_app Debug
   - **Project**: SmartFusion2_FIC_Tutorial_sb_MSS_CM3_app
   - **C/C++ Application**: Debug\SmartFusion2_FIC_Tutorial_sb_CM3_app

5. Click **Apply** and **Debug**.

6. Click **Yes**, when prompted for **Confirm Perspective Switch**. This displays the debug view mode as shown in Figure 58.

**Figure 58 • Confirm Perspective Switch**

7. Debug Perspective is similar as shown Figure 59.
8. Run the application by clicking Run > Resume or click the Run icon on the SoftConsole toolbar. The Application options along with the greeting message are displayed in the terminal program window as shown in Figure 60.

Figure 60 • Tera Term Window

9. Select Writing to LSRAM, it prompts for Enter the offset address between 0 to 8188 to write and press Enter as shown in Figure 61.

Note: In PuTTY, press CTRL+J instead of Enter.
10. After Entering the offset address, it prompts for **Enter data to write** as shown in Figure 62.

11. Select **Reading from the LSRAM**, it prompts for **Enter the offset address to read and press Enter** as shown in Figure 63.
12. Select **GPIO LED Blinking**, it prompts for **Enter the number between 0 to 255 and press Enter** as shown in **Figure 64 on page 46**.

**Note:** For SmartFusion2 Starter Kit, Tera Term window is displayed, as shown in **Figure 65**.

**Figure 64 • Selecting GPIO LED Blinking**
2.12 Step 9: Building Executable Image in Release mode

You can build an application executable image in-release mode and load it into eNVM for executing code in eNVM of the SmartFusion2 SoC FPGA device. You can load the application executable image into eNVM with the help of eNVM data storage client from System Builder eNVM Configurator. In release mode, you cannot use SoftConsole debugger to load the executable image into eNVM.

2.13 Conclusion

This tutorial outlined the design flow for creating a SmartFusion2 project using Libero SoC design software, configuring the SmartFusion2 MSS, interfacing fabric peripherals to the SmartFusion2 MSS using fabric interface controllers (FIC_0), simulation of the design using BFM commands and running the application design on board.
# Revision History

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

<table>
<thead>
<tr>
<th>Revision</th>
<th>Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Revision 13 (March 2016)</td>
<td>Updated the document for Libero v11.7 software release changes (SAR 76666).</td>
</tr>
<tr>
<td>Revision 12 (December 2015)</td>
<td>Updated Figure 1 on page 5 and Figure 44 on page 34 (SAR 73892).</td>
</tr>
<tr>
<td>Revision 11 (October 2015)</td>
<td>Updated the document for Libero v11.6 software release changes (SAR 72067).</td>
</tr>
<tr>
<td>Revision 10 (February 2015)</td>
<td>Updated the document for Libero v11.5 software release (SAR 64506).</td>
</tr>
<tr>
<td>Revision 9 (August 2014)</td>
<td>Updated the document for Libero v11.4 software release (SAR 59820).</td>
</tr>
<tr>
<td>Revision 8 (May 2014)</td>
<td>Updated the document for Libero v11.3 software release (SAR 56454).</td>
</tr>
<tr>
<td>Revision 7 (February 2014)</td>
<td>Updated the document (SAR 54212).</td>
</tr>
<tr>
<td>Revision 6 (November 2013)</td>
<td>Updated the document for Libero version 11.2 (SAR 52904).</td>
</tr>
<tr>
<td>Revision 5 (April 2013)</td>
<td>Updated the document for 11.0 production SW release (SAR 47302).</td>
</tr>
<tr>
<td>Revision 4 (February 2013)</td>
<td>Updated the document for Libero 11.0 Beta SP1 software release (SAR 44868).</td>
</tr>
<tr>
<td>Revision 3 (November 2012)</td>
<td>Updated the document for Libero 11.0 Beta SPA software release (SAR 42904).</td>
</tr>
<tr>
<td>Revision 2 (October 2012)</td>
<td>Updated the document for Libero 11.0 Beta launch (SAR 41696).</td>
</tr>
<tr>
<td>Revision 1 (May 2012)</td>
<td>Updated the document for LCP2 software release (SAR 38954).</td>
</tr>
<tr>
<td>Revision 1 (December 2015)</td>
<td>Initial release.</td>
</tr>
</tbody>
</table>
4 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.

4.1 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

4.2 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.

4.3 Technical Support


4.4 Website


4.5 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.

4.5.1 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.

4.5.2 My Cases

Microsemi SoC Products Group customers may submit and track technical cases online by going to My Cases.
4.5.3 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.

4.6 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; 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. Microsemi is headquartered in Aliso Viejo, Calif, and has approximately 4,800 employees globally. Learn more at www.microsemi.com.