# Contents

1 Revision History ................................................................. 1  
   1.1 Revision 6.0 .............................................................. 1  
   1.2 Revision 5.0 .............................................................. 1  
   1.3 Revision 4.0 .............................................................. 1  
   1.4 Revision 3.0 .............................................................. 1  
   1.5 Revision 2.0 .............................................................. 1  
   1.6 Revision 1.0 .............................................................. 1  

2 SoftConsole v4.0 and Libero SoC v11.7 ..................................... 2  
   2.1 Introduction .............................................................. 2  
   2.2 Design Requirements .................................................... 2  
      2.2.1 Associated Project Files .......................................... 2  
   2.3 Design Overview ........................................................ 3  
      2.4 Step 1: Creating a Libero SoC Project ............................... 3  
      2.4.1 Launching Libero SoC ............................................. 3  
      2.4.2 Connecting Components in LED_Blink SmartDesign ............ 10  
   2.5 Step 2: Generating the Program File .................................. 11  
   2.6 Step 3: Programming the SmartFusion2 Board Using FlashPro .......... 14  
   2.7 Step 4: Creating Software Project using SoftConsole 4.0 ............ 15  
      2.7.1 Export Firmware .................................................... 15  
      2.7.2 Download Firmware Drivers ...................................... 16  
      2.7.3 Building Software Application using SoftConsole 4.0 ........... 18  
      2.7.4 Debugging the Application Project using SoftConsole v4.0 ...... 25  
   2.8 Conclusion ............................................................... 30  

3 Appendix: Board Setup for SmartFusion2 Security Evaluation Kit .......... 31  

4 Appendix: Board Setup for SmartFusion2 Advanced Development Kit ........ 33  

5 Appendix: Board Setup for SmartFusion2 Starter Kit ...................... 35
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Design Files Top-Level Structure</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>Libero SoC Project Manager</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>Project Details Window</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>Device Selection Window</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>Device Settings</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>Design Template Window</td>
<td>6</td>
</tr>
<tr>
<td>7</td>
<td>New Project Information Window</td>
<td>6</td>
</tr>
<tr>
<td>8</td>
<td>System Builder Dialog Box</td>
<td>7</td>
</tr>
<tr>
<td>9</td>
<td>System Builder – Device Features Page</td>
<td>7</td>
</tr>
<tr>
<td>10</td>
<td>System Builder Configurator – Select Peripherals Page</td>
<td>8</td>
</tr>
<tr>
<td>11</td>
<td>MSS_GPIO Configurator</td>
<td>9</td>
</tr>
<tr>
<td>12</td>
<td>System Builder Configurator – Clock Settings Page</td>
<td>10</td>
</tr>
<tr>
<td>13</td>
<td>System Builder Generated System</td>
<td>10</td>
</tr>
<tr>
<td>14</td>
<td>LED_Blink Design</td>
<td>11</td>
</tr>
<tr>
<td>15</td>
<td>Generate Component</td>
<td>11</td>
</tr>
<tr>
<td>16</td>
<td>Log Window</td>
<td>11</td>
</tr>
<tr>
<td>17</td>
<td>Design Flow Window</td>
<td>11</td>
</tr>
<tr>
<td>18</td>
<td>Manage Constraints</td>
<td>12</td>
</tr>
<tr>
<td>19</td>
<td>I/O Attributes</td>
<td>12</td>
</tr>
<tr>
<td>20</td>
<td>I/O Editor</td>
<td>13</td>
</tr>
<tr>
<td>21</td>
<td>Generate Bitstream</td>
<td>14</td>
</tr>
<tr>
<td>22</td>
<td>Run Programming Action</td>
<td>14</td>
</tr>
<tr>
<td>23</td>
<td>Export Firmware</td>
<td>15</td>
</tr>
<tr>
<td>24</td>
<td>Export Firmware Dialog Box</td>
<td>15</td>
</tr>
<tr>
<td>25</td>
<td>Firmware Export Successful</td>
<td>15</td>
</tr>
<tr>
<td>26</td>
<td>Firmware Catalog</td>
<td>16</td>
</tr>
<tr>
<td>27</td>
<td>Firmware Catalog</td>
<td>17</td>
</tr>
<tr>
<td>28</td>
<td>Generate Options</td>
<td>17</td>
</tr>
<tr>
<td>29</td>
<td>Warning Message</td>
<td>17</td>
</tr>
<tr>
<td>30</td>
<td>Software Tool Chain</td>
<td>17</td>
</tr>
<tr>
<td>31</td>
<td>Workspace Launcher</td>
<td>18</td>
</tr>
<tr>
<td>32</td>
<td>SoftConsole Window</td>
<td>18</td>
</tr>
<tr>
<td>33</td>
<td>Creating New C Project</td>
<td>19</td>
</tr>
<tr>
<td>34</td>
<td>C Project Window</td>
<td>19</td>
</tr>
<tr>
<td>35</td>
<td>C Project - Select Configuration</td>
<td>20</td>
</tr>
<tr>
<td>36</td>
<td>C Project - Cross GNU ARM Tool Chain</td>
<td>20</td>
</tr>
<tr>
<td>37</td>
<td>Project Explorer - Import</td>
<td>20</td>
</tr>
<tr>
<td>38</td>
<td>Import Window</td>
<td>21</td>
</tr>
<tr>
<td>39</td>
<td>Import - File System</td>
<td>21</td>
</tr>
<tr>
<td>40</td>
<td>Project Explorer</td>
<td>22</td>
</tr>
<tr>
<td>41</td>
<td>Project Explorer window - Properties</td>
<td>22</td>
</tr>
<tr>
<td>42</td>
<td>Properties for LED_Blink</td>
<td>23</td>
</tr>
<tr>
<td>43</td>
<td>Properties for LED_Blink - Miscellaneous</td>
<td>23</td>
</tr>
<tr>
<td>44</td>
<td>Properties for LED_Blink - General</td>
<td>24</td>
</tr>
<tr>
<td>45</td>
<td>Properties for LED_Blink - Miscellaneous</td>
<td>24</td>
</tr>
<tr>
<td>46</td>
<td>Project - Build All</td>
<td>25</td>
</tr>
<tr>
<td>47</td>
<td>Problems Window</td>
<td>25</td>
</tr>
<tr>
<td>48</td>
<td>Run - Debug Configurations</td>
<td>25</td>
</tr>
<tr>
<td>49</td>
<td>Debug Configurations</td>
<td>26</td>
</tr>
<tr>
<td>50</td>
<td>Debug Tab</td>
<td>27</td>
</tr>
<tr>
<td>51</td>
<td>Confirm Perspective Switch</td>
<td>27</td>
</tr>
<tr>
<td>52</td>
<td>Debugger Perspective Window</td>
<td>28</td>
</tr>
<tr>
<td>53</td>
<td>Debug Configurations Option</td>
<td>28</td>
</tr>
<tr>
<td>54</td>
<td>Values of Cortex-M3 Internal Registers</td>
<td>29</td>
</tr>
</tbody>
</table>
# Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table 1</td>
<td>Design Requirements</td>
<td>2</td>
</tr>
<tr>
<td>Table 2</td>
<td>SmartFusion2 Devices Selection</td>
<td>4</td>
</tr>
<tr>
<td>Table 3</td>
<td>SmartFusion2 GPIO Configuration</td>
<td>8</td>
</tr>
<tr>
<td>Table 4</td>
<td>Port to Pin Mapping</td>
<td>12</td>
</tr>
<tr>
<td>Table 5</td>
<td>LED Target Board</td>
<td>28</td>
</tr>
<tr>
<td>Table 6</td>
<td>SmartFusion2 Security Evaluation Kit Jumper Settings</td>
<td>31</td>
</tr>
<tr>
<td>Table 7</td>
<td>SmartFusion2 Advanced Development Kit Jumper Settings</td>
<td>33</td>
</tr>
<tr>
<td>Table 8</td>
<td>SmartFusion2 Starter Kit Jumper Settings</td>
<td>35</td>
</tr>
</tbody>
</table>
1 Revision History

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

1.1 Revision 6.0
The following changes are made in revision 6.0 of this document:
• A note about FlashPro5 Programmer is added in Debugging the Application Project using SoftConsole v4.0, page 25
• Three notes are added in Appendix: Board Setup for SmartFusion2 Security Evaluation Kit, page 31 to describe about J5 - FlashPro connector and J18 - FTDI programmer interface

1.2 Revision 5.0
In revision 5.0, this document is updated for Libero SoC v11.7 software release.

1.3 Revision 4.0
In revision 4.0, this document is updated for Libero SoC v11.6 and SoftConsole v4.0 software release.

1.4 Revision 3.0
In revision 3.0, this document is updated for Libero SoC v11.5 software release.

1.5 Revision 2.0
In revision 2.0, this document is updated for Libero SoC v11.4 software release.

1.6 Revision 1.0
Revision 1.0 was the first publication of this document.
2 SoftConsole v4.0 and Libero SoC v11.7

2.1 Introduction
This tutorial describes how to implement an ARM Cortex-M3 design using Libero® System-on-Chip (SoC) v11.7 and build a simple LED blink application using SoftConsole v4.0.

After completing this tutorial, you will be able to perform the following tasks:

- Create a Libero SoC project using System Builder
- Generate the programming file to program the SmartFusion®2 SoC field programmable gate array (FPGA) device
- Create a SoftConsole v4.0 project
- Compile application code
- Debug and run code using SoftConsole

2.2 Design Requirements

<table>
<thead>
<tr>
<th>Table 1 • Design Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Design Requirements</td>
</tr>
<tr>
<td><strong>Hardware Requirements</strong></td>
</tr>
<tr>
<td>SmartFusion2 Security Evaluation Kit:</td>
</tr>
<tr>
<td>• FlashPro4 or FlashPro5 programmer</td>
</tr>
<tr>
<td>• 12 V adapter</td>
</tr>
<tr>
<td>SmartFusion2 Advanced Development Kit:</td>
</tr>
<tr>
<td>• FlashPro4 or FlashPro5 programmer</td>
</tr>
<tr>
<td>• 12 V adapter</td>
</tr>
<tr>
<td>SmartFusion2 Starter Kit:</td>
</tr>
<tr>
<td>• FlashPro4 or FlashPro5 programmer</td>
</tr>
<tr>
<td>• USB A to Mini-B cable</td>
</tr>
<tr>
<td>Host PC or Laptop</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Software Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Libero SoC</td>
</tr>
<tr>
<td>SoftConsole</td>
</tr>
<tr>
<td>FlashPro programming software</td>
</tr>
</tbody>
</table>

Note: This tutorial is applicable for any one of the SmartFusion2 boards listed in the preceding table.

2.2.1 Associated Project Files

Download design files for this tutorial from the Microsemi website:
http://soc.microsemi.com/download/rsc/?f=m2s_tu0546_liberov11p7_df

The demo design files include:

- Libero Project
- Sourcefiles
- Readme file

The following figure shows the top-level structure of the design files. See the Readme.txt file provided in the design files directory for the complete directory structure.
2.3 **Design Overview**

This tutorial demonstrates a simple LED blinking application for SmartFusion2 device. Microcontroller subsystem (MSS) general-purpose input/output (GPIOs) are configured as outputs and connected to LEDs using fabric I/Os. This tutorial is applicable on one of the following SmartFusion2 boards:

- SmartFusion2 Security Evaluation Kit
- SmartFusion2 Advanced Development Kit
- SmartFusion2 Starter Kit (M2S010-FGG484)

2.4 **Step 1: Creating a Libero SoC Project**

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

2.4.1 **Launching Libero SoC**

1. Click **Start > Programs > Microsemi > Libero SoC v11.7** or double-click the shortcut on desktop to open the Libero SoC v11.7 Project Manager.
2. Create a new project by selecting **New** on the **Start Page** tab (highlighted in the following figure) or by clicking **Project > New Project** from the Libero SoC menu.

3. In the **Project Details** window, enter the following information as shown in Figure 3, page 4.
   - **Project Name**: LED_Blink
   - **Project Location**: Select an appropriate location (for example, `D:/Microsemi_prj`)
   - **Preferred HDL type**: Verilog or VHDL

---

**Figure 1** • Design Files Top-Level Structure

```
<download_folder>
  SF2_LED_Blink_SC_Tutorial_DF
    Sourcefiles
    LiberoProject
    Readme.txt
```

**Figure 2** • Libero SoC Project Manager
• Enable Block Creation: Unchecked

**Figure 3** • Project Details Window

4. Click **Next**. In the **Device Selection** window, select the information as displayed in **Figure 4**, page 5. In the Part Filter, select the values using the drop-down lists, as shown in the following table.

• **Family:** SmartFusion2

**Table 2** • SmartFusion2 Devices Selection

<table>
<thead>
<tr>
<th>Board</th>
<th>Die</th>
<th>Package</th>
<th>Speed</th>
<th>Core Voltage</th>
<th>Range</th>
<th>PLL Supply Voltage</th>
</tr>
</thead>
<tbody>
<tr>
<td>SmartFusion2 Security Evaluation Kit</td>
<td>M2S090TS</td>
<td>484 FBGA</td>
<td>-1</td>
<td>1.2</td>
<td>COM</td>
<td>3.3</td>
</tr>
<tr>
<td>SmartFusion2 Advanced Development Kit</td>
<td>M2S150T</td>
<td>1152 FC</td>
<td>-1</td>
<td>1.2</td>
<td>COM</td>
<td>3.3</td>
</tr>
<tr>
<td>SmartFusion2 Starter Kit</td>
<td>M2S010</td>
<td>484 FBGA</td>
<td>STD</td>
<td>1.2</td>
<td>COM</td>
<td>2.5</td>
</tr>
</tbody>
</table>
5. Click Next, the Device settings window is displayed. Select PLL Supply Voltage (V), as shown in the following figure.

**Figure 5 • Device Settings**

See Table 2, page 4 for specific board values.

6. Click Next. In the Design Template page, select Create a system builder based design check box under the Design Templates and Creators as shown in the following figure.
7. Click Finish. A **New Project Information** window is displayed, as shown in the following figure.

8. Click **Use Enhanced Constraint Flow**, as shown in the following figure.

**Figure 7 • New Project Information Window**

A System Builder dialog box is displayed, as shown in **Figure 8**, page 7.
System builder is a graphical design wizard. It creates a design based on high-level design specifications by taking the user through a set of high-level questions that define the intended system.

9. Enter a name for your system as **LED_Blink** and then click OK, as shown in the following figure.

*Figure 8* • **System Builder Dialog Box**

![System Builder Dialog Box](image)

*System Builder – Device Features* page is displayed, as shown in the following figure.

*Figure 9* • **System Builder – Device Features Page**

![System Builder – Device Features Page](image)

10. Retain the default values. Click **Next**, the **System Builder – Peripherals** page is displayed. Under the MSS Peripherals section, uncheck all the check boxes except MSS_GPIO, as shown in the following figure.
11. Double-click the wrench symbol for the MSS_GPIO peripheral to open the MSS_GPIO Configurator.

12. This design requires configuring GPIOs to drive LEDs on the board, configure the GPIOs as shown below:
   - **Set/Reset Definition** accept default settings
   - Configure GPIO as shown in the following table

### Table 3 • SmartFusion2 GPIO Configuration

<table>
<thead>
<tr>
<th>Board</th>
<th>Die</th>
<th>GPIO ID</th>
<th>Direction</th>
<th>Package Pin</th>
<th>Connectivity</th>
</tr>
</thead>
<tbody>
<tr>
<td>SmartFusion2 Security Evaluation Kit</td>
<td>M2S090TS</td>
<td>GPIO_0 to GPIO_7</td>
<td>Output</td>
<td>NA</td>
<td>FABRIC_A</td>
</tr>
<tr>
<td>SmartFusion2 Advanced Development Kit</td>
<td>M2S150T</td>
<td>GPIO_0 to GPIO_7</td>
<td>Output</td>
<td>NA</td>
<td>FABRIC_A</td>
</tr>
<tr>
<td>SmartFusion2 Starter Kit</td>
<td>M2S010</td>
<td>GPIO_0 to GPIO_1</td>
<td>Output</td>
<td>NA</td>
<td>FABRIC_A</td>
</tr>
</tbody>
</table>
13. Click Next, the System Builder – Clocks Settings page is displayed, as shown in Figure 12, page 10. Select System Clock source as On-chip 25/50 MHz RC Oscillator. The M3_CLK is configured to 100 MHz by default.

Figure 11 • MSS_GPIO Configurator
14. Click **Next**, the **System Builder – Microcontroller Options** page is displayed.
   - Retain the default values.
15. Click **Next**, the **System Builder – SECDED Options** page is displayed.
   - Retain the default values.
16. Click **Next**, the **System Builder – Security Options** page is displayed.
   - Retain the default values.
17. Click **Next**, the **System Builder – Interrupts Options** page is displayed.
   - Retain the default values.
18. Click **Next**, the **System Builder – Memory Map Options** page is displayed.
   - Retain the default values.
19. Click **Finish**.

System Builder generates the system based on the selected options.

System Builder block is created and added to the Libero SoC project, as shown in the following figure.

---

### 2.4.2 Connecting Components in LED_Blink SmartDesign

The following steps describe how to connect the components in LED_Blink SmartDesign:

1. Connect the pins as follows:
   - Tie the **FAB_RESET_N** to high by right-clicking and selecting **Tie High**.
   - Mark the output port **POWER_ON_RESET_N** as unused by right-clicking and selecting **Mark Unused**.
• Mark the output port MSS_READY as unused by right-clicking and selecting **Mark Unused**.
• Expand INIT_PINS, right-click INIT_DONE and select **Mark Unused**.
• Expand FAB_CCC_PINS, right-click FAB_CCC_GL0 and select **Mark Unused**.
• Right-click FAB_CCC_LOCK and select **Mark Unused**.
• Right-click GPIO_FABRIC and select **Promote to Top Level**.

2. Click **File > Save**. The LED_Blink design is displayed, as shown in the following figure.

**Figure 14 • LED_Blink Design**

![LED_Blink Design](image)

3. Generate the LED_Blink SmartDesign by clicking **SmartDesign > Generate Component** or by clicking **Generate Component** on the SmartDesign toolbar.

**Figure 15 • Generate Component**

![Generate Component](image)

After successful generation of the system, the message **info: LED_Blink' was successfully generated** is displayed on the Libero SoC **Log** window, as shown in the following figure.

**Figure 16 • Log Window**

![Log Window](image)

### 2.5 Step 2: Generating the Program File

1. Double-click **Synthesize** in the **Design Flow** window, as shown in the following figure to complete the synthesis.

**Figure 17 • Design Flow Window**

![Design Flow Window](image)
2. Double-click **Manage Constraints** in the Design Flow window, as shown in the following figure.

*Figure 18 • Manage Constraints*

![Manage Constraints](image)

3. Click **Edit with I/O Editor** under I/O Attributes, as shown in the following figure. The I/O Editor window is displayed, as shown in Figure 20, page 13.

*Figure 19 • I/O Attributes*

![I/O Attributes](image)

4. Make the pin assignments, as shown in the following table. After the pins are assigned, the I/O Editor is displayed, as shown in Figure 20, page 13.

*Table 4 • Port to Pin Mapping*

<table>
<thead>
<tr>
<th>Port Name</th>
<th>Pin Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>SmartFusion2 Security Evaluation Kit</td>
<td></td>
</tr>
<tr>
<td>GPIO_0_M2F</td>
<td>H5</td>
</tr>
<tr>
<td>GPIO_1_M2F</td>
<td>H6</td>
</tr>
<tr>
<td>GPIO_2_M2F</td>
<td>J6</td>
</tr>
<tr>
<td>GPIO_3_M2F</td>
<td>H7</td>
</tr>
<tr>
<td>GPIO_4_M2F</td>
<td>G7</td>
</tr>
<tr>
<td>GPIO_5_M2F</td>
<td>F3</td>
</tr>
<tr>
<td>GPIO_6_M2F</td>
<td>F4</td>
</tr>
</tbody>
</table>
5. After updating the I/O editor, click Commit and Check.
6. Close the I/O Editor.
7. Click Generate Bitstream in Design flow window, as shown in the following figure, to generate the programming file.
2.6 Step 3: Programming the SmartFusion2 Board Using FlashPro

Jumper settings for the supported target boards and board setup for running the tutorial are given in the following chapters:

- Appendix: Board Setup for SmartFusion2 Advanced Development Kit, page 33.
- Appendix: Board Setup for SmartFusion2 Starter Kit, page 35.

1. To program the SmartFusion2 device, double-click Run PROGRAM Action in the Design Flow window, as shown in the following figure.
2.7 Step 4: Creating Software Project using SoftConsole 4.0

The following steps show how to create a software project using SoftConsole 4.0.

2.7.1 Export Firmware

1. Double-click Export Firmware in Handoff design for Production in the Design Flow window, as shown in the following figure.

Figure 23 • Export Firmware

Export Firmware dialog box is displayed as shown in the following figure.

2. Select Export hardware configuration (Software IDE independent), as shown in the following figure.

Figure 24 • Export Firmware Dialog Box

3. Click OK, a notification window appears saying Firmware project was successfully exported to <drive:>/Microsemi_prj/LED_Blink, as shown in the following figure.

Figure 25 • Firmware Export Successful

4. Click OK.
2.7.2 Download Firmware Drivers

The following drivers are used in this tutorial:

- CMSIS
- GPIO

To generate the required drivers:

1. Open the Microsemi SoC Firmware catalog from: Start > Programs > Microsemi > Libero SoC v11.7 > Firmware Catalog > Firmware Catalog.

A message, New cores are available for download is displayed at the bottom of the Firmware Catalog, as shown in the following figure.

**Figure 26 • Firmware Catalog**

2. Click Download them now, to download most recent drivers for peripherals.

3. In Firmware catalog window, right-click SmartFusion2 CMSIS Hardware Abstraction Layer and then click Generate, as shown in the following figure.
4. In Generate Options window, browse Project folder at `<drive:>`Microsemi_prj\LED_Blink\firmware and then click OK.

5. A warning message is displayed, as shown in the following figure. Click Yes.

6. Configuring SmartFusion2_CMSIS_TOP_0 window is displayed, as shown in the following figure. Select SoftConsole from the Software Tool Chain drop-down list and click OK.

7. Repeat steps 3 to 7 to download SmartFusion2 MSS GPIO driver.
2.7.3 Building Software Application using SoftConsole 4.0

1. Click Start > Programs > Microsemi SoftConsole v4.0 > Microsemi SoftConsole v4.0 to open the SoftConsole IDE. The SoftConsole Workspace Launcher window is displayed.

2. Browse to the location to select D:\Microsemi_prj\LED_Blink\SoftConsole, as shown in the following figure.

   Figure 31 • Workspace Launcher

   ![Workspace Launcher](image)

   The SoftConsole workspace is displayed, as shown in the following figure.

   Figure 32 • SoftConsole Window

   ![SoftConsole Window](image)

3. Click File > New > C project as shown in the following figure.
4. Enter Project name as LED_Blink, as shown in the following figure.

5. Click **Next**, **Select Configurations** window is displayed, as shown in the following figure.
6. Do not change default settings. Click **Next**.
7. **Cross GNU ARM Tool chain** window is displayed, as shown in the following figure.

**Figure 36 • C Project - Cross GNU ARM Tool Chain**

8. Click **Finish**.
9. Right-click **LED_Blink** and click **Import** as shown in the following figure.

**Figure 37 • Project Explorer - Import**

10. **Import** window is displayed, as shown the following figure.
11. Click **File System** and then click **Next**.

*Figure 38 • Import Window*

12. Browse to `D:\Microsemi_prj\LED_Blink\firmware` and check **firmware** check box, as shown in the following figure.

*Figure 39 • Import - File System*

13. Click **Finish**.

**Note:** If any changes are made to the Libero SoC project, firmware needs to be exported from Libero and new firmware must be imported to **LED_Blink**.

14. Using Windows explorer, browse to the `main.c` file location in the respective design files folder as follows:

- For SmartFusion2 Security Evaluation Kit:
  `<download_folder>\SF2_LED_Blink_SC_Tutorial_DF\Sourcefiles\SF2_Security_Kit`.
• For SmartFusion2 Advanced Development Kit:
  <download_folder>\SF2_LED_Blink_SC_Tutorial_DF\Sourcefiles\SF2_Adv_Dev_Kit.
• For SmartFusion2 Starter Kit:
  <download_folder>\SF2_LED_Blink_SC_Tutorial_DF\Sourcefiles\SF2_Starter_Kit.

15. Copy the main.c file to the LED_Blink project in the SoftConsole workspace, as shown in the following figure.

![Project Explorer](image)

**Figure 40**  Project Explorer

16. Right-click LED_Blink and click Properties, as shown in the following figure.

![Project Explorer window - Properties](image)

**Figure 41**  Project Explorer window - Properties

17. Click Settings under the C/C++ Build tab, as shown in the following figure.
18. Under **Cross ARM C compiler**, click **Miscellaneous** and enter
   `--specs=cmsis.specs`, in **Other compiler flags** text box as shown in the following figure.

**Figure 42 • Properties for LED_Blink**

![Properties for LED_Blink](image)

19. Under **Cross ARM C Linker**, click **General** as shown in **Figure 44**, page 24.
20. Click **add button** and add following linker Script path:
   ```
   "$\{workspace\_loc/$\{ProjName\}/CMSIS/startup\_gcc/debug-in-microsemi-smartfusion2-esram.ld\}"
   ```
   21. After adding Linker Script, **Properties for LED_Blink**, window is displayed, as shown in the following figure.
22. Under **Cross ARM C Linker**, click **Miscellaneous**.
23. Check **Use newlib-nano(--specs=nano.specs)** option, as shown in the following figure.

**Figure 45 • Properties for LED_Blink - Miscellaneous**

24. Click **OK**.
25. Click **Project** and click **Build All**, as shown in the following figure.
Figure 46 • Project - Build All

26. Ensure that the Problems tab in the displayed window must not have any errors, as shown in the following figure.

Figure 47 • Problems Window

2.7.4 Debugging the Application Project using SoftConsole v4.0

The following steps describe how to debug the application project using SoftConsole v4.0:

1. Click **Debug Configurations** in the Run menu of the SoftConsole, as shown in the following figure. The Debug Configurations window is displayed, as shown in Figure 49, page 26.

Figure 48 • Run - Debug Configurations

2. Double-click **GDB OpenOCD Debugging** to view the configurations, as shown in the following figure.
3. Ensure that the following values are filled in the corresponding fields:
   - **Name**: LED_Blink Debug
   - **Project**: LED_Blink
   - **C/C++ Application**: Debug\LED_Blink.elf

4. Select the **Debugger** tab in the **Debug Configurations** dialog box. The command `set DEVICE M2S090` specifies the target device, as shown in Figure 50, page 27. This command needs to be modified based on the target silicon.
   - SmartFusion2 Security Evaluation Kit - set DEVICE M2S090
   - SmartFusion2 Advanced Development Kit - set DEVICE M2S150
   - SmartFusion2 Starter Kit - set DEVICE M2S010
5. Click **Debug**.
6. On the **Confirm Perspective Switch** window, click **Yes** as shown in the following figure.

**Figure 51 • Confirm Perspective Switch**

The **SoftConsole Debugger Perspective** window is displayed, as shown in the following figure.
7. Click Run > Resume to run the application. LEDs start blinking on the SmartFusion2 target boards. The following table shows which LEDs blink for the different SmartFusion2 target boards.

Table 5 • LED Target Board

<table>
<thead>
<tr>
<th>Target Board</th>
<th>LEDs</th>
</tr>
</thead>
<tbody>
<tr>
<td>SmartFusion2 Security Evaluation Kit</td>
<td>H5, H6, J6, H7, G7, F3, F4, and E1</td>
</tr>
<tr>
<td>SmartFusion2 Advanced Development Kit</td>
<td>DS0, DS1, DS2, DS3, DS4, DS5, DS6, and DS7</td>
</tr>
<tr>
<td>SmartFusion2 Starter Kit</td>
<td>DS4, DS3</td>
</tr>
</tbody>
</table>

8. Launch the debug session:
   • By selecting Debug Configurations from the Run menu of SoftConsole.
   or
   • By selecting the Debug Configurations using the Debug button as shown in the following figure.

Figure 53 • Debug Configurations Option

9. Click the Registers tab to view the values of the Cortex-M3 processor internal registers, as shown in the following figure.
10. Click the Variables tab to view the values of variables in the source code, as shown in the following figure.

**Figure 55 • Values of the Variables in the Source Code**

11. In the Debug window, click Window > Show View > Disassembly to display the assembly level instructions. The Disassembly window with assembly instructions is displayed on the right-side of the Debug perspective window, as shown the following figure.

**Figure 56 • Assembly Level Instructions**

12. Source code can be single-stepped by choosing Run > Step Into or Run > Step Over. Observe the changes in the source code window and disassembly view. Performing a Step Over provides an option for stepping over functions. The entire function is run but there is no need to single-step through each instruction contained in the function.

13. Click Instruction Stepping ( ) and perform Step Into operations. Observe that Step Into executes a single line of assembly code.

14. Click Instruction Stepping to exit the instruction stepping mode. Single-step through the application and observe the instruction sequence in the source code window of the Debug perspective, and the values of the variables and registers.

15. Add breakpoints in the application to force the code to halt, single-step, and observe the instruction sequence.
16. When debug process is finished, terminate execution of the code by choosing Run > Terminate.
17. Close Debug Perspective by selecting Close Perspective from the Window menu.

**Note:** By default SoftConsole debugs using the first FlashPro5 programmer that it detects. If there is no FlashPro5 connected then it will use the first FlashPro3/4 that it detects.

### 2.8 Conclusion

This tutorial provides steps to create a Libero SoC design using System Builder. It describes the procedure to build, debug, and run a SoftConsole application. It also provides a simple design to blink LEDs.
3 Appendix: Board Setup for SmartFusion2 Security Evaluation Kit

The following figure shows the board setup for running the tutorial on the SmartFusion2 Security Evaluation kit board.

**Figure 57 • SmartFusion2 Security Evaluation Kit Setup**

1. Connect the jumpers on the SmartFusion2 Security Evaluation kit board as listed in the following table. For more information on jumper locations, see Figure 58, page 32 for SmartFusion2 Security Evaluation kit board jumper locations.

   **CAUTION:** While making the jumper connections, the SW7 power supply switch on the board must be in OFF position.

   **Table 6 • SmartFusion2 Security Evaluation Kit Jumper Settings**

<table>
<thead>
<tr>
<th>Jumper</th>
<th>Pin (From)</th>
<th>Pin (To)</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>J22, J23, J8, and J3</td>
<td>1</td>
<td>2</td>
<td>These are the default jumper settings of the SmartFusion2 Security Evaluation Kit board. Ensure that these jumpers are set accordingly.</td>
</tr>
</tbody>
</table>

2. Connect the FlashPro4 or FlashPro5 programmer to the J5 connector of the SmartFusion2 Security Evaluation kit.

   **Note:** J5 - FlashPro connector is normally used for FlashPro programming of the FPGA and SoftConsole debugging.
Note: J18 - FTDI programmer interface, used to program the external serial peripheral interface (SPI) flash cannot be used for SoftConsole debugging.

Note: If both J5 and J18 are connected to the host computer on which SoftConsole is running then SoftConsole must be configured to use J5 for debugging.

3. Connect the power supply to the J6 connector.
4. Switch ON the SW7 power supply switch.

The following figure shows the jumper locations on the SmartFusion2 Security Evaluation kit board.

**Figure 58 • SmartFusion2 Security Evaluation Kit Board Jumper Locations**

Note: Jumpers highlighted in red are set by default.

Note: The locations of the jumpers in the preceding figure are searchable.
The following figure shows the board setup for running the demo on the SmartFusion2 Advanced Development kit board.

Figure 59 • SmartFusion2 Advanced Development Kit Setup

1. Connect the jumpers on the SmartFusion2 Advanced Development kit board as listed in the following table. For more information on jumper locations, see Figure 60, page 34 of SmartFusion2 Advanced Development kit board Jumper Locations.

**CAUTION:** While making the jumper connections, the SW7 power supply switch on the board must be in OFF position.

<table>
<thead>
<tr>
<th>Jumper</th>
<th>Pin (from)</th>
<th>Pin (to)</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>J116, J353, J354, and J54</td>
<td>1</td>
<td>2</td>
<td>These are the default jumper settings of the SmartFusion2 advanced development kit board. Ensure that these jumpers are set accordingly.</td>
</tr>
<tr>
<td>J123</td>
<td>2</td>
<td>3</td>
<td>JTAG programming via FTDI</td>
</tr>
<tr>
<td>J124, J121, and J32</td>
<td>2</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>

2. Connect the FlashPro4 or FlashPro5 programmer to the J37 connector of the SmartFusion2 Advanced Development kit.
3. Connect the power supply to the J42 connector.
4. Switch **ON** the SW7 power supply switch.
The following figure shows the jumper locations on the SmartFusion2 Advanced Development kit board.

*Figure 60 • SmartFusion2 Advanced Development Kit Board Jumper Locations*

**Notes:**
- Jumpers highlighted in red are set by default.
- Jumpers highlighted in green must be set manually.
- The locations of the jumpers in the preceding figure are searchable.
Appendix: Board Setup for SmartFusion2 Starter Kit

The following figure shows the board setup for running the demo on the SmartFusion2 starter kit board.

Figure 61 • SmartFusion2 Starter Kit Setup

1. Connect the jumpers on the SmartFusion2 Starter kit board as listed in Table 8, page 35.

<table>
<thead>
<tr>
<th>Jumper</th>
<th>Pin (From)</th>
<th>Pin (To)</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>JP1</td>
<td>1</td>
<td>2</td>
<td>These are the default jumper settings of SmartFusion2 Starter kit board. Ensure that these jumpers are set accordingly.</td>
</tr>
<tr>
<td>JP2</td>
<td>3</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>JP3</td>
<td>2</td>
<td>4</td>
<td></td>
</tr>
</tbody>
</table>

2. Connect the FlashPro4 or FlashPro5 programmer to the P5 connector of the SmartFusion2 Starter kit.
3. Connect the host PC USB port to the P1 Mini USB connector on the SmartFusion2 Starter kit board using the USB Mini-B cable. As soon as the connection to the PC is made, the on-board LED DS2 will illuminate, indicating that the board has power.