SmartFusion cSoC: Building Executable Image in Release Mode and Loading into eNVM Tutorial
# Table of Contents

**SmartFusion cSoC: Building Executable Image in Release Mode and Loading into eNVM** ...........................................................................................................3

Introduction ..................................................................................................................3

Building an Executable Image in Release Mode .............................................................4

Loading the Executable Image into eNVM ....................................................................6

**List of Changes** ..........................................................................................................14

**Product Support** ........................................................................................................15

Customer Service ..........................................................................................................15

Customer Technical Support Center ..............................................................................15

Technical Support ..........................................................................................................15

Website .........................................................................................................................15

Contacting the Customer Technical Support Center ....................................................15

ITAR Technical Support .................................................................................................16
SmartFusion cSoC: Building Executable Image in Release Mode and Loading into eNVM

Introduction

SmartFusion® customized system-on-chip (cSoC) devices contain a hard embedded microcontroller subsystem (MSS), programmable analog circuitry, and FPGA fabric consisting of logic tiles, SRAM, and PLLs. The MSS consists of 100 MHz ARM® Cortex™-M3 processor, advanced high-performance bus (AHB) matrix, system registers, Ethernet MAC, DMA engine, real-time counter (RTC), embedded nonvolatile memory (eNVM), embedded SRAM (eSRAM), fabric interface controller (FIC), I²C, SPI, and EMC. Figure 1 shows the SmartFusion cSoC block diagram.

This tutorial explains how to build an application executable image in ‘release mode’ and load it into eNVM for executing code in place in eNVM of the SmartFusion cSoC device. You can load the application executable image into eNVM with the help of eNVM data storage client from SmartDesign MSS Configurator and IAP or FlashPro programming software. In the release mode, you cannot use SoftConsole debugger to load the executable image into eNVM.

This tutorial is in continuation of the SmartFusion tutorials and explains you how to build the applications in release mode after completion of application debugging. Refer to the Using UART with SmartFusion - Microsemi Libero SoC and SoftConsole Flow Tutorial to understand the flow for debugging mode.

![SmartFusion cSoC Block Diagram](image)
Building an Executable Image in Release Mode

This section outlines the steps required to build an executable image in release mode for executing the code in place in eNVM of the SmartFusion cSoC device.

1. Open the SoftConsole project workspace provided in the design files of this tutorial.

2. Right click on both project names in the Project Explorer view and select Build Configurations > Set Active > Release as shown in Figure 2.

3. Select Project > Properties from the SoftConsole menu. Expand the C/C++ Build and select 'Settings'.

4. Provide the release mode linker script file to the linker by changing the 'Linker flags' field to "-T../../OLED_MSS_MSS_CM3_0_hw_platform/CMSIS/startup_gcc/production-execute-in-place.ld" as shown in the Figure 3.
5. Click **Apply** and then click **OK**.

6. Build the project (**Project > Build All**) and observe that `.hex` file is generated in ‘Release’ folder created in project folder as shown in Figure 4 · and Figure 5 ·.
Loading the Executable Image into eNVM

This section outlines the steps required to load the generated executable image into eNVM of the SmartFusion cSoC device using the SmartDesign MSS Configurator.

1. Open the Libero SoC hardware project provided in the design files of this tutorial.
2. Open the MSS component created using the SmartDesign MSS Configurator by double clicking on the MSS component present in the ‘Design Hierarchy’ window. The SmartDesign MSS Configurator tab is displayed as shown in Figure 6.

If you are using the SoftConsole standalone flow then the created MSS component can be launched from the SoftConsole project workspace (Run > External Tools > MSS_Configurator).

Figure 5 · Hex File Location in Release Folder
3. Double click on the ENVM module available in the SmartDesign MSS Configurator to add eNVM data storage client. The MSS_ENVM_0 configurator window is displayed as shown in the Figure 7 ·.
4. Select **Data Storage** under the **Available client types** tab and click **Add to System**. The **Add to Data Storage Client** window opens as shown in Figure 8.

5. Enter a data storage client name in the **Add to Data Storage Client** window.
6. Browse for .hex file generated in the Building an Executable Image in Release Mode section. The generated executable image can be found in the Release folder under SoftConsole project workspace as shown in the Figure 9.

---

**Figure 8 - Adding eNVM Data Storage Client**

**Figure 9 - Browsing for .hex File**
7. Change the size of word to 32 bits as shown in the Figure 10.

Figure 10 - Loading *.hex File

8. Click OK in the Add Data Storage Client window and the MSS_ENVM_0 Configurator window.

9. Regenerate the MSS component by clicking SmartDesign > Generate Component. Note: If you are following SoftConsole Standalone flow then skip step 11.

10. Click Generate Programming Data to complete the remaining steps to generate fdb file. (synthesis, place-and-route).

11. Open the FlashPro project by right-clicking on the Program Device icon in the Libero SoC Design Flow window and select Open Interactively. If you are following the SoftConsole Standalone flow then open the created FlashPro project and modify the Programming file by Import the updated Embedded Flash Memory Configuration file (.efc file) from the generated MSS component folder. Refer to the SoftConsole Standalone Flow Tutorial for more information on importing .efc file.
12. The FlashPro tool will be displayed as shown in the Figure 11.

![Figure 11 · PDB File Loaded by FlashPro](image)

13. Click **File > Export > Export Single Programming File** for generating the STAPL file as shown in Figure 12.

![Figure 12 · STAPL File](image)
14. Clear all the **Output formats** except the STAPL File as shown in Figure 13. Click **Export** and then **Close**.

**Figure 12 · Exporting The Programming File**

**Figure 13 · Generating STAPL File**
15. Click **Configure Device** and then **Browse**.

![Figure 14 · Configure the Device](image)

16. Select the .stp file and click **Open** as shown in **Figure 15 ·**.

![Figure 15 · Selecting .stp File](image)
17. Click **PROGRAM** to program the device. Observe the FlashPro log window showing ‘Program Embedded Flash Memory Module ALL...’ command execution as shown in the Figure 16.

![FlashPro Programming](image)

**Figure 16 - FlashPro Programming**

18. Once programming is done, save the project and close FlashPro project.

19. Remove the USB cable for the programmer and restart the board to run the application.

**Note:** You need not run the debugger using SoftConsole since the code is already programmed into eNVM using FlashPro.
# List of Changes

<table>
<thead>
<tr>
<th>Revision</th>
<th>Changes</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Revision 2</td>
<td>Updated screenshots and content for Libero SoC SP1 (SAR 38221).</td>
<td>NA</td>
</tr>
<tr>
<td>(April 2012)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Revision 1</td>
<td>Updated screenshot for Libero SoC and support for STAPL File generation (SAR 34981).</td>
<td>7, 11</td>
</tr>
<tr>
<td>(November 2011)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Note:** The revision number is located in the part number after the hyphen. The part number is displayed at the bottom of the last page of the document. The digits following the slash indicate the month and year of publication.
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

Visit the Microsemi SoC Products Group Customer Support website for more information and support (http://www.microsemi.com/soc/support/search/default.aspx). Many answers available on the searchable web resource include diagrams, illustrations, and links to other resources on website.

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/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. Sales office listings can be found at www.microsemi.com/soc/company/contact/default.aspx.

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_itar@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 solutions for: aerospace, defense and security; enterprise and communications; and industrial and alternative energy markets. Products include high-performance, high-reliability analog and RF devices, mixed signal and RF integrated circuits, customizable SoCs, FPGAs, and complete subsystems. Microsemi is headquartered in Aliso Viejo, Calif. Learn more at [www.microsemi.com](http://www.microsemi.com).