Implementing Production Release Mode Programming for SmartFusion2 - Libero SoC v11.4

Application Note

December 2014

Microsemi

Revision History

Date	Revision	Change
22 December 2014	2	Second release
22 August 2014	1	First release

Confidentiality Status

This is a non-confidential document.

Table of Contents

Implementing Production Release Mode Programming for SmartFusion2 - Libero SoC v11.4

Purpose

This application note demonstrates how to build single/multiple SoftConsole application executable images in the production release mode, store them into eNVM using eNVM data storage client, and execute them on SmartFusion®2 system-on-chip (SoC) field programmable gate array (FPGA) Evaluation Kit.

Introduction

SmartFusion2 SoC FPGA devices have a maximum of two on chip 256 KB flash memories called eNVM. The eNVM is used to store the application code executable image or store data, which is used by the user application.

Refer to the eNVM chapter of the *SmartFusion2 Microcontroller Subsystem User Guide* for a detailed description about eNVM.

The user application projects are built in Release/Debug mode for execution on SmartFusion2 Evaluation kit through SoftConsole. In the Debug mode, the complete symbolic debug information is emitted to the debugging applications and the code is not optimized. In the Release mode, the complete symbolic debug information is not emitted and the code execution is optimized. The size of the final release mode executable is smaller than the debug executable. It also runs faster than the debug mode.

This application note describes the following:

- How to build a single SoftConsole application executable image in production release mode, store the image into eNVM using eNVM data storage client, and execute the image directly from eNVM of the SmartFusion2 Evaluation Kit.

  A sample SoftConsole application project fetches factorial of a number from a non-executable image (LookUpTable.hex) stored in eNVM. LookUpTable.hex file contains pre-calculated factorial values. For more information about how to create a LookUpTable.Hex file, see "Appendix B - Creating LookUpTable.Hex File" on page 48.
- How to build multiple SoftConsole application executable images in production release mode and store them into eNVM using eNVM data storage client.

  The SoftConsole application projects show how the stored executable images in eNVM directly runs from eNVM, and how to remap the images to run from eSRAM. The SoftConsole application projects designed here are to blink LED on the SmartFusion2 Evaluation Kit.

To demonstrate the above two scenarios, two design folders with the SoftConsole application projects are provided with this application note in "Appendix A - Design and Programming Files" on page 47.

References

The following list of references is used in this document:

- SmartFusion2 Microcontroller Subsystem User Guide

Design Requirements

Table 1 lists the design requirements.

Table 1 • **Design Requirements**

Design Requirements	Description
Hardware Requirements	
SmartFusion2 Evaluation Kit: • 12 V adapter (provided along with the kit) • FlashPro4 programmer (provided along with the kit) • USB A to Mini-B cable	Rev C
Host PC or Laptop	Any 64-bit Windows Operating System
Software Requirements	
Libero® System-on-Chip (SoC)	v11.4
SoftConsole	3.4SP1
Host PC Drivers (provided along with the design files)	USB to UART drivers
One of the following serial terminal emulation programs: • HyperTerminal • TeraTerm • PuTTY	–

Building Executable Image in Release Mode

The following steps describe how to build an application executable image in the release mode:

1. Open the standalone SoftConsole IDE.

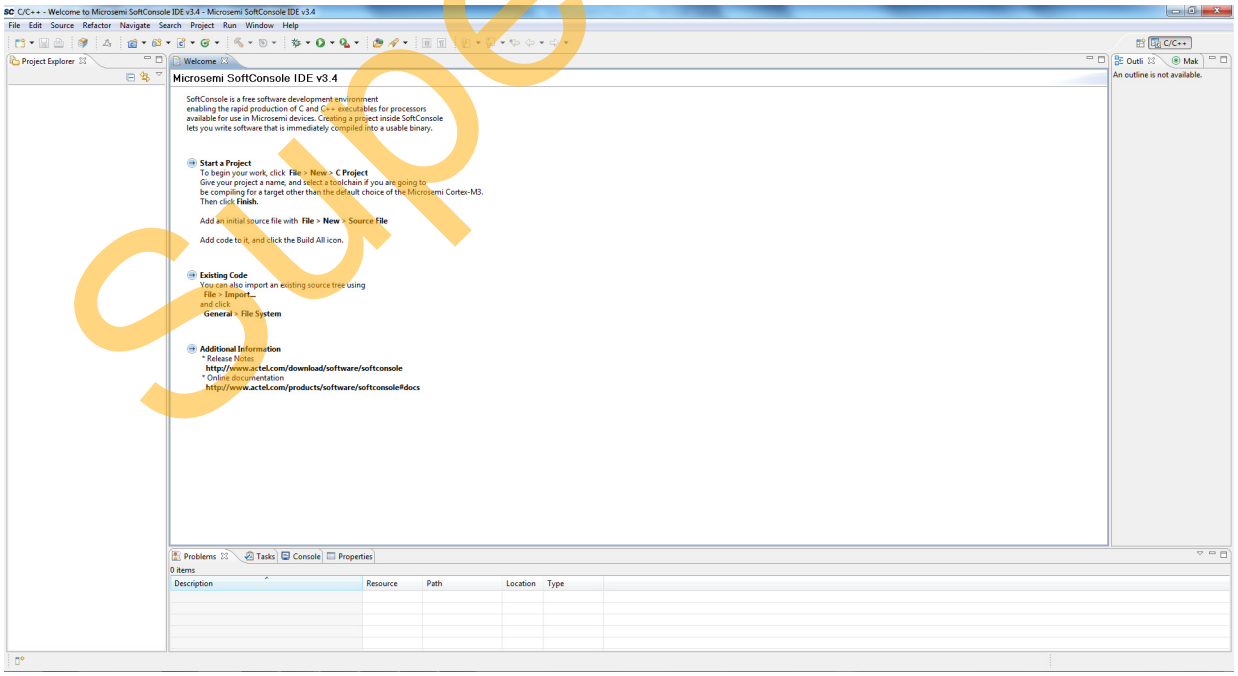

Figure 1 • **SoftConsole IDE**

2. Right-click the **Project Explorer** window and choose **Import** as shown in Figure 2.

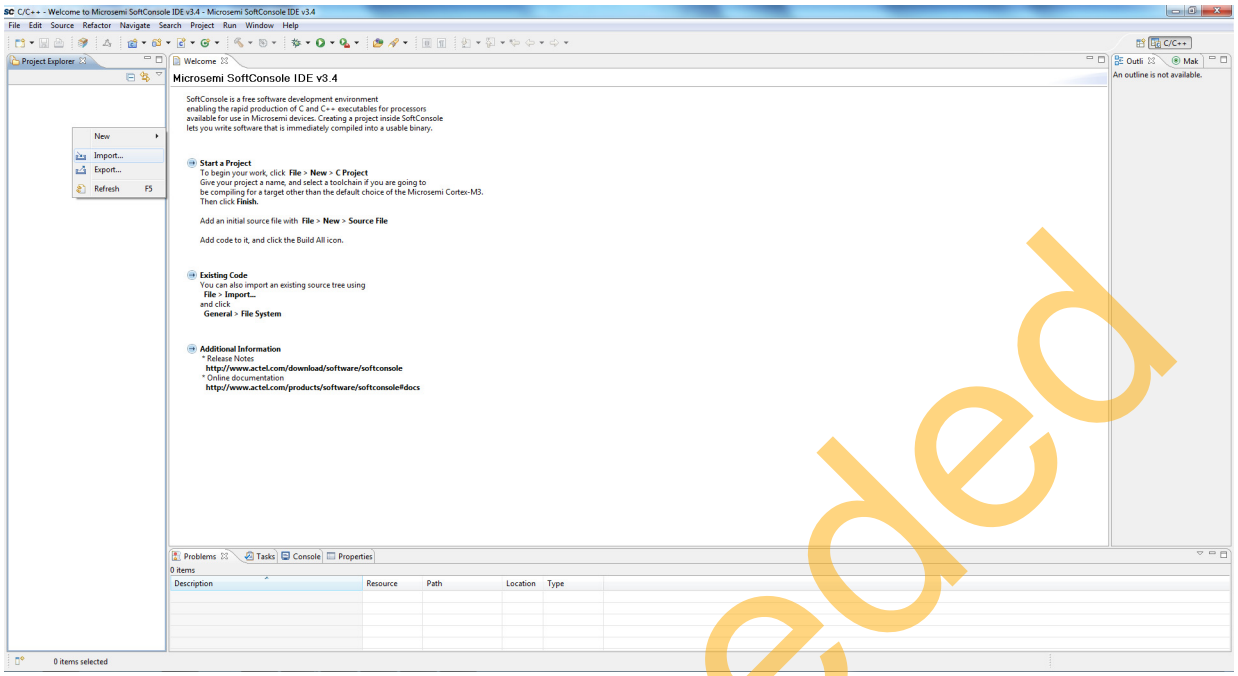

***Figure 2* • SoftConsole IDE - Import**

3. Select **General** > **Existing Projects into Workspace** as shown in Figure 3.

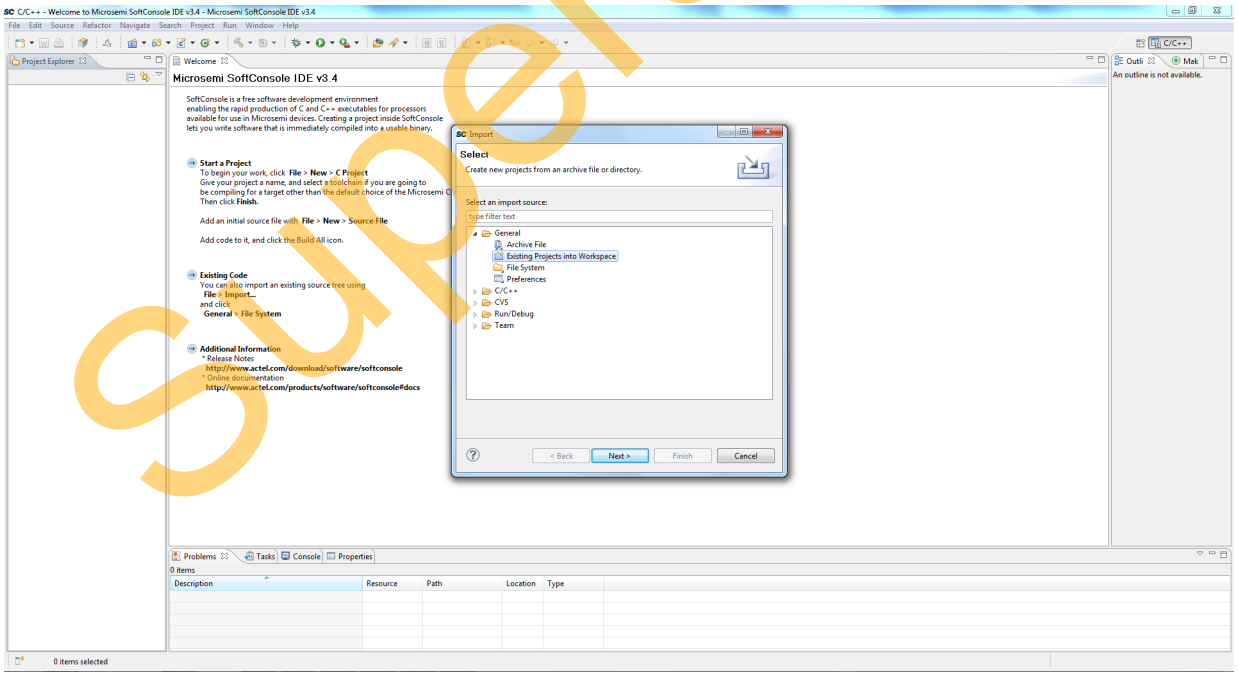

***Figure 3* • SoftConsole IDE - Importing Existing Projects**

Import window is displayed as shown in Figure 4

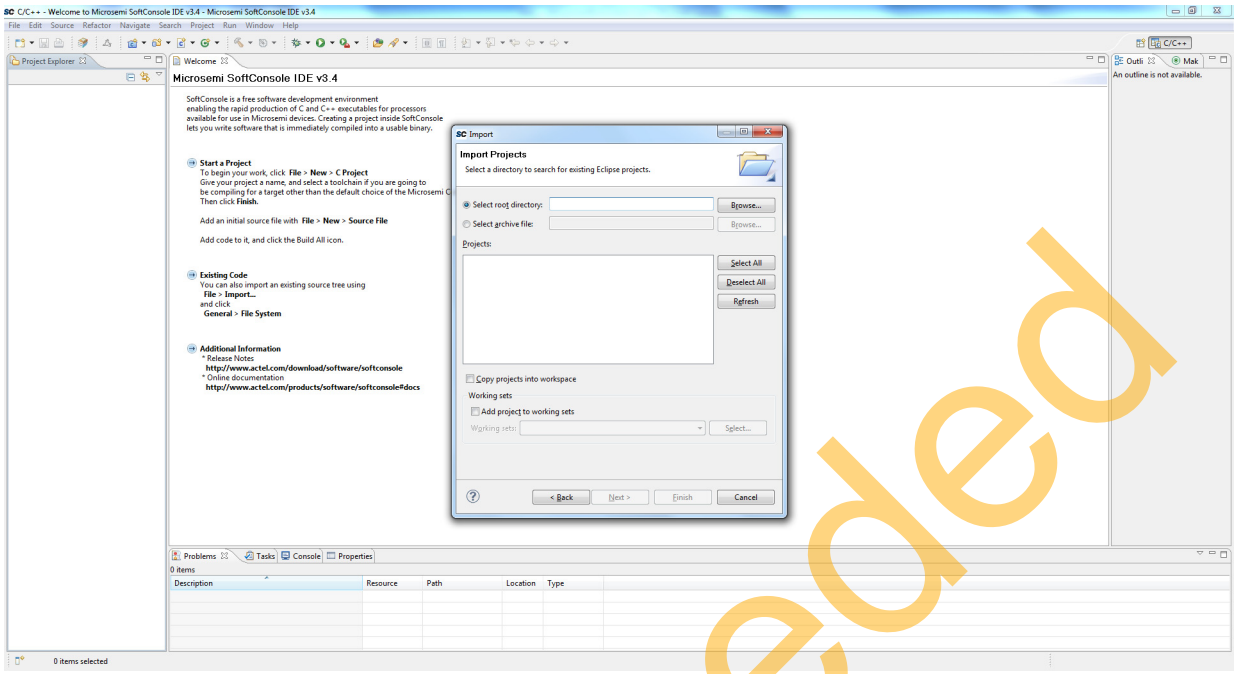

***Figure 4* • SoftConsole IDE - Browse**

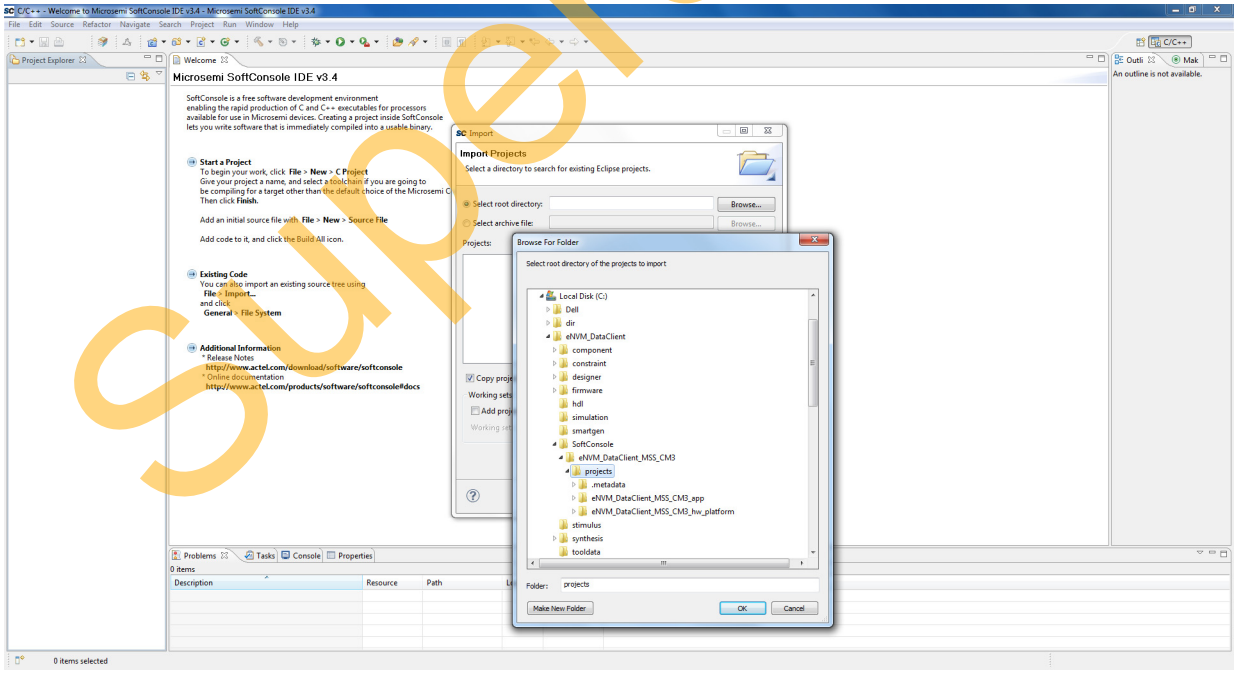

***Figure 5* • SoftConsole IDE - Browsing for Projects Folder**

4. Ensure that **Copy projects into workspace** check box is checked as shown in Figure 6. Click **Finish**.

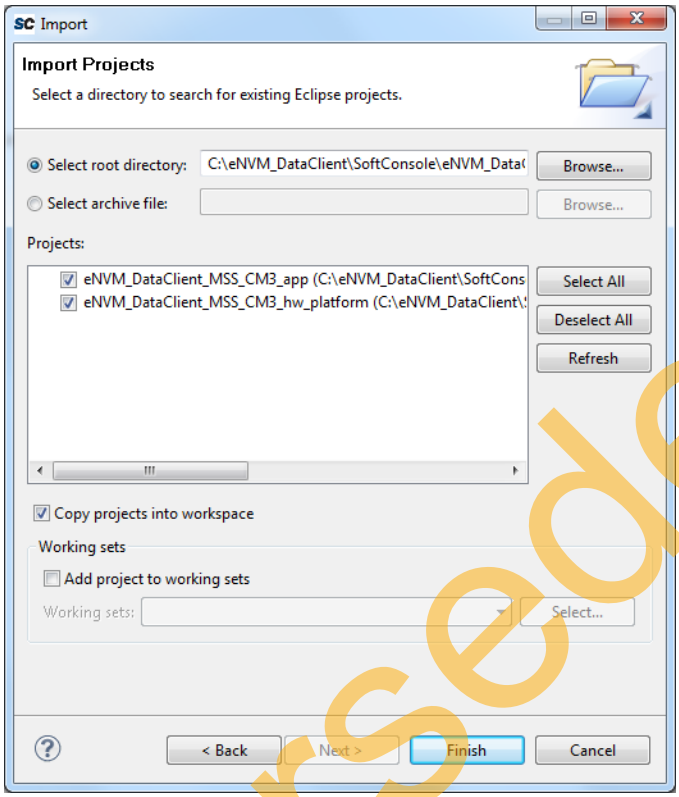

***Figure 6* • SoftConsole IDE - Creating Work Space in App Folder**

The SoftConsole perspective is shown in Figure 7.

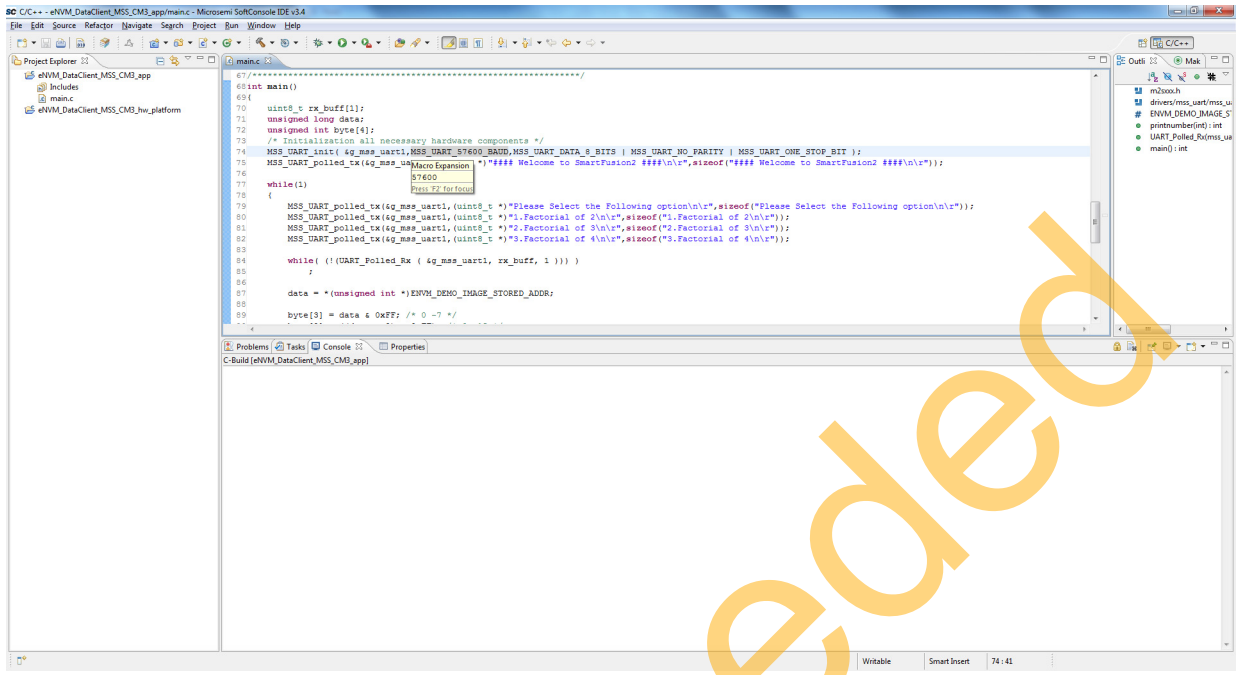

***Figure 7* • SoftConsole Workspace**

5. Right-click both the project names in the **Project Explorer** view and select **Build Configurations > Set Active > Release** as shown in Figure 8.

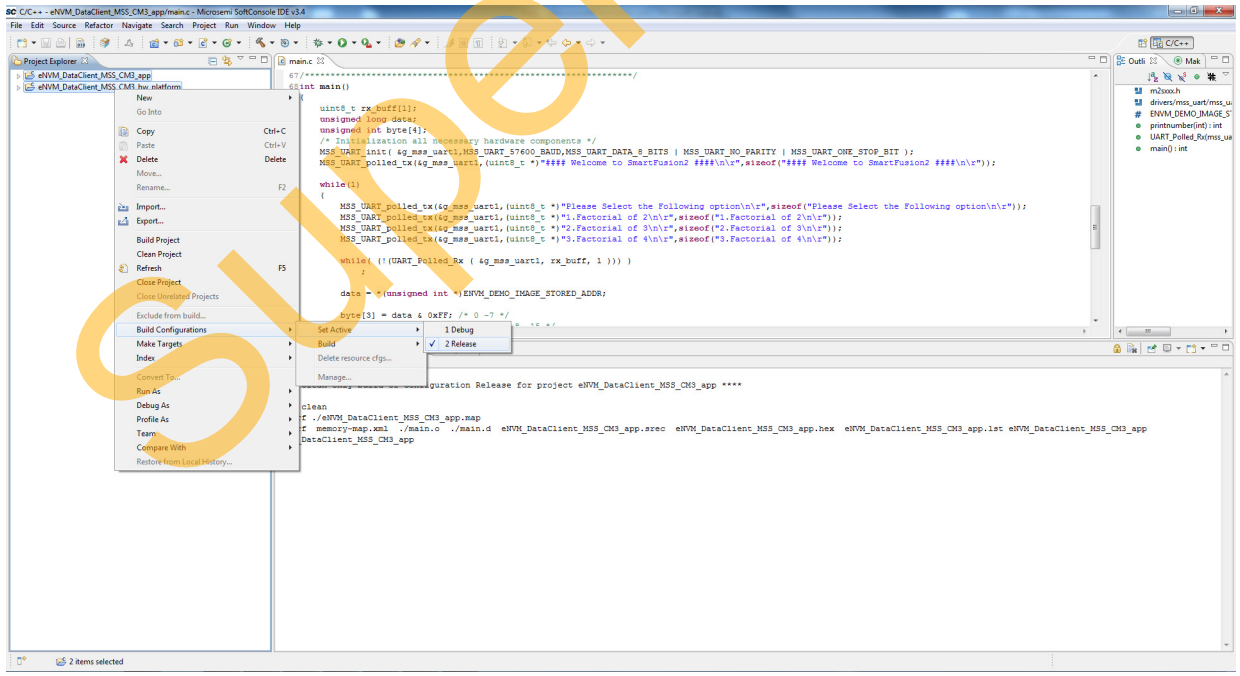

***Figure 8* • Building Configurations**

6. Go to **Project > Build All** and Build the project as shown in Figure 9. .hex file is generated in the **Release** folder (see Figure 10 and Figure 11 on page 12).

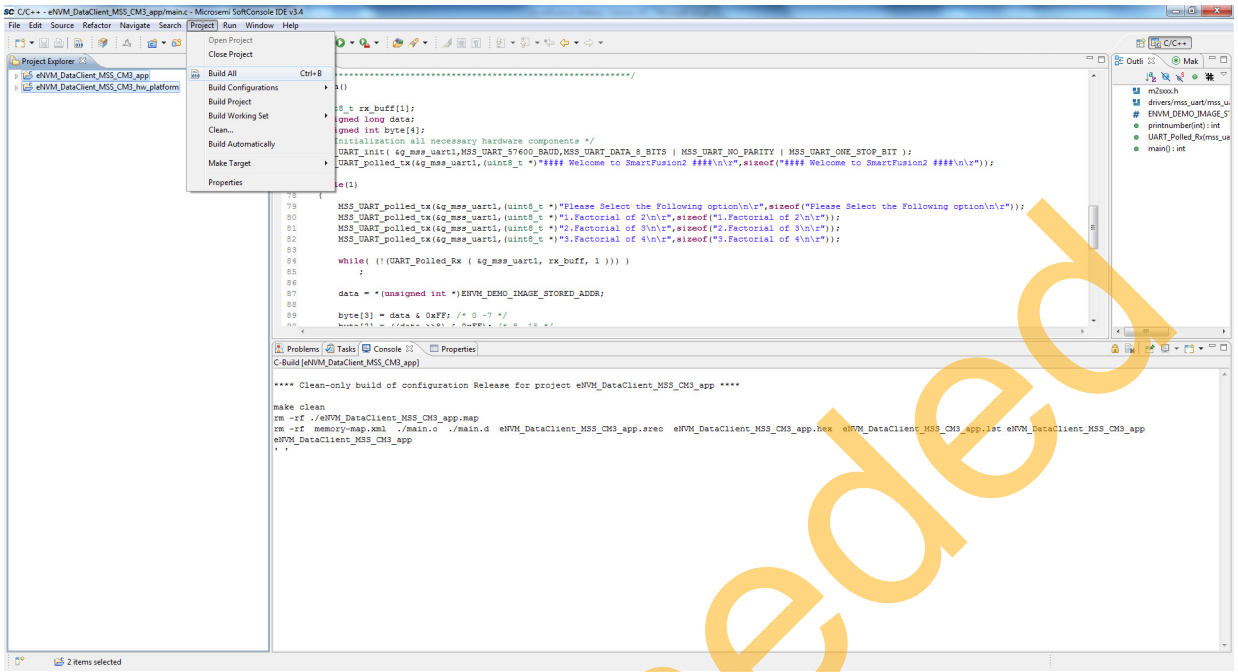

Figure 9 • **Build All**

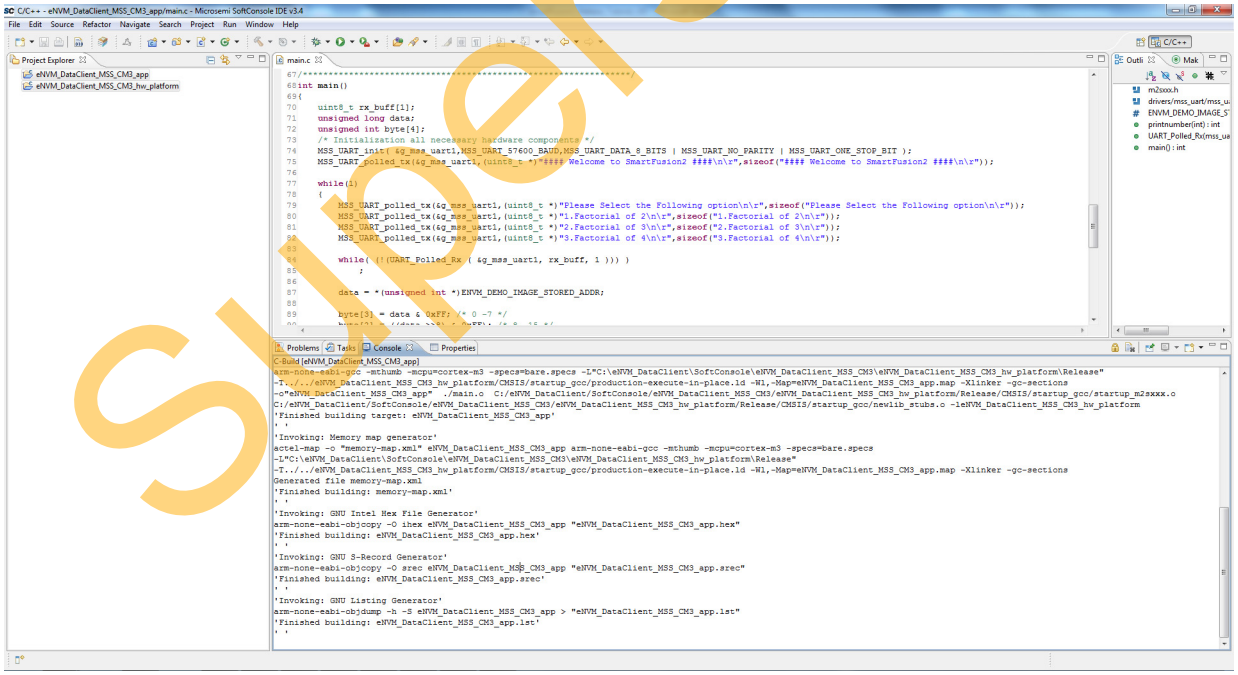

Figure 10 • **Generating Hex File**

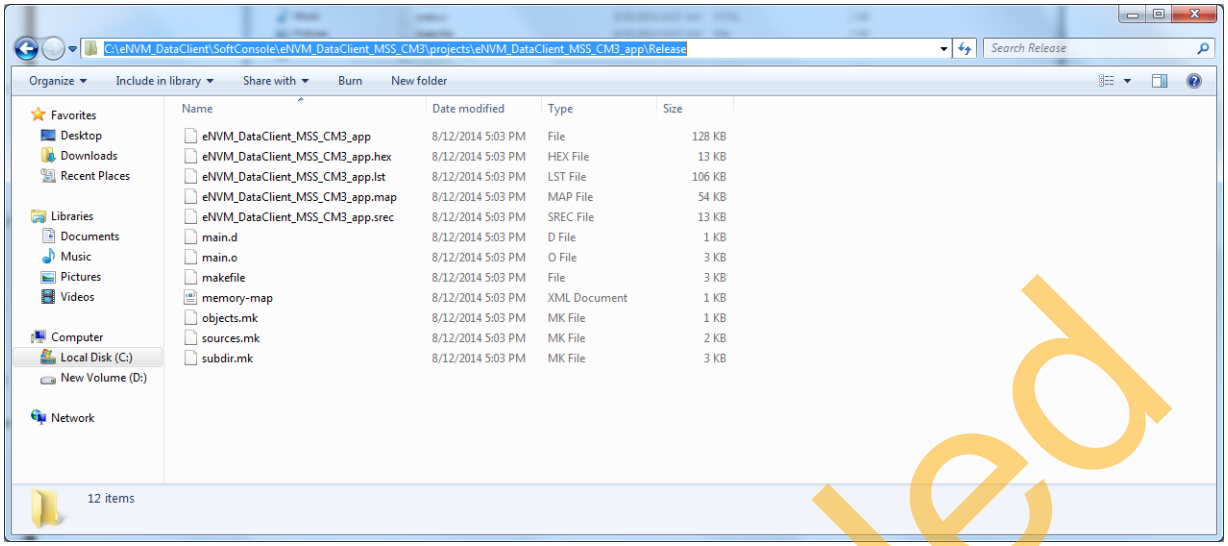

***Figure 11* • Hex File in Release Folder**

Loading the Executable Image into eNVM

The following steps describe how to load the generated executable image into eNVM of the SmartFusion2 SoC device using the System Builder eNVM Configurator:

1. Open the Libero SoC hardware project **eNVM_DataClient_top** tab (see Figure 12).

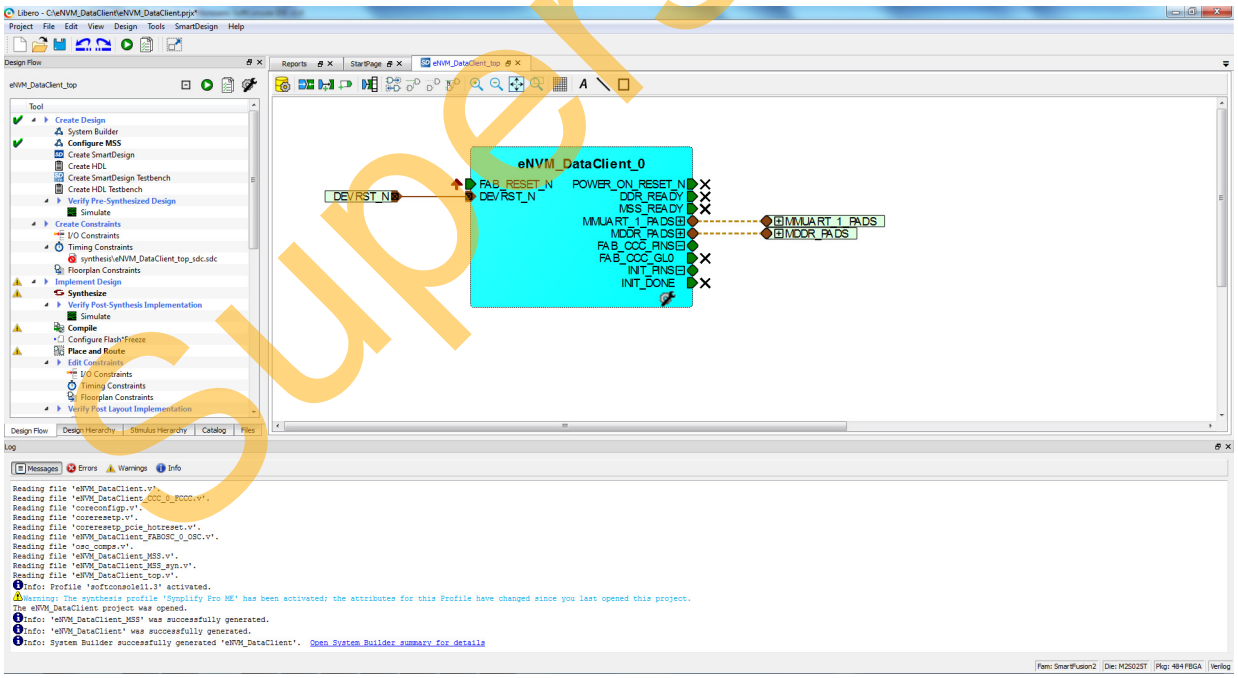

***Figure 12* • eNVM_DataClient_top Tab**

2. Double-click **eNVM_DataClient_0** (see Figure 12) and go to **System Builder - Memories** page to add the eNVM Data Storage client.

3. Select **Data Storage** under **Available Client Types** tab (see Figure 13) and click **Add to System...**. This opens **Add to Data Storage Client** window as shown in Figure 14·

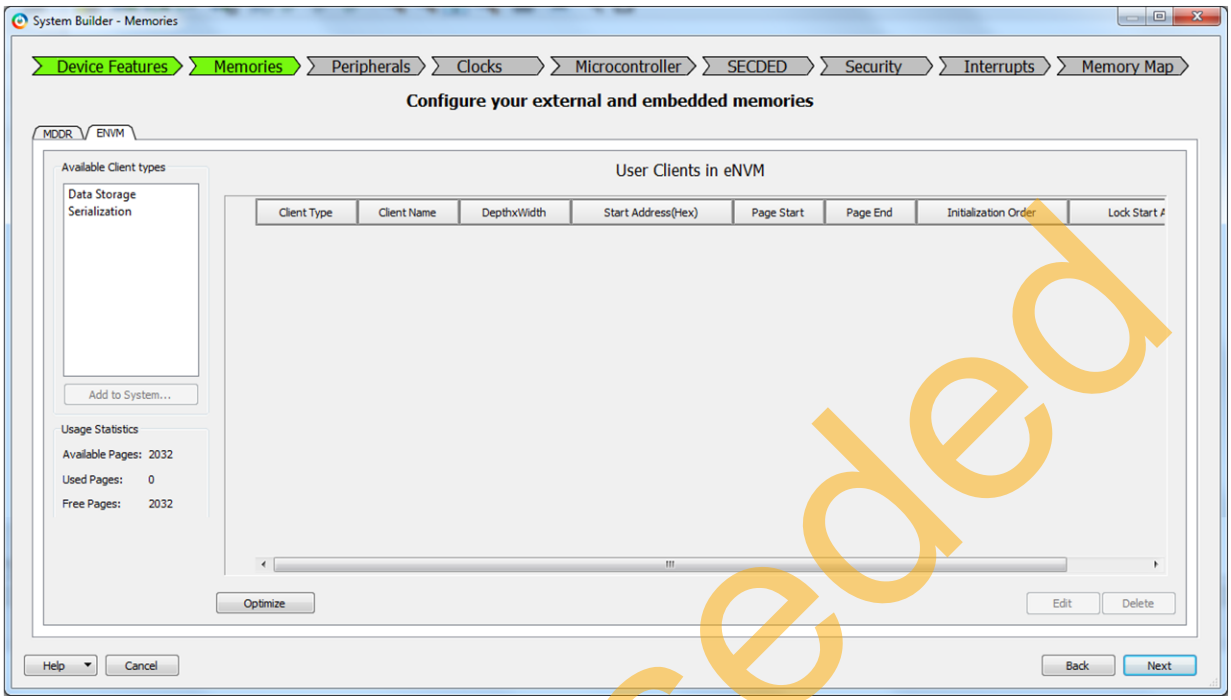

***Figure 13* • System Builder - Memories Page**

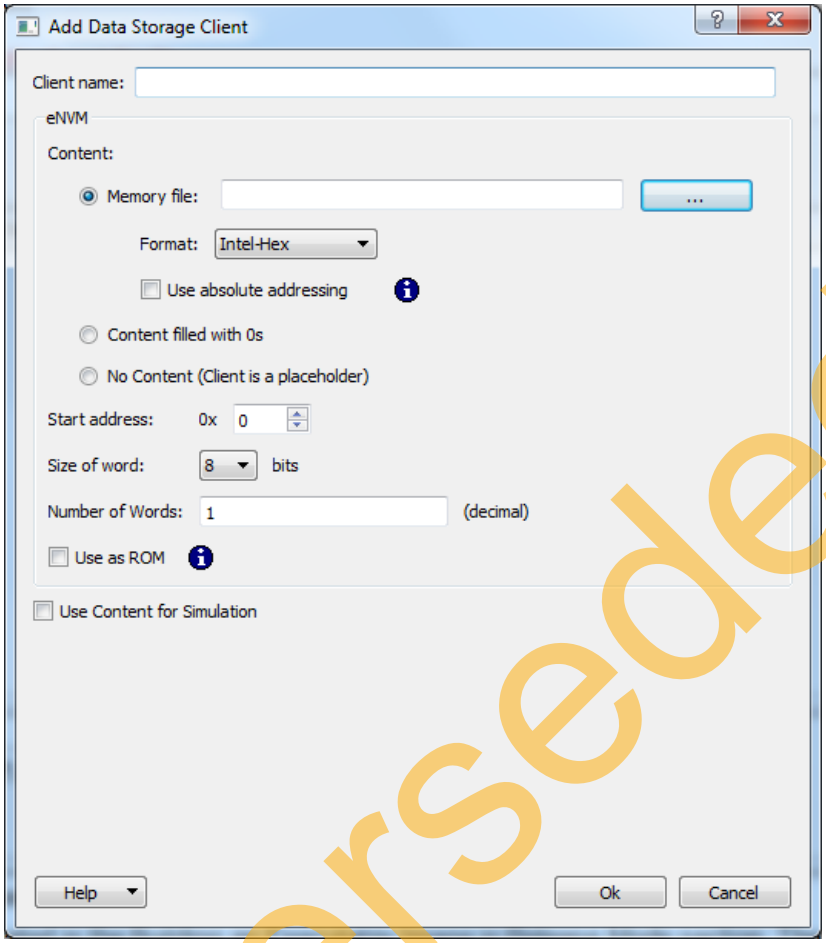

Figure 14 • **Add Data Storage Client Window**

4. Enter a data storage client name in the **Add to Data Storage Client** window.

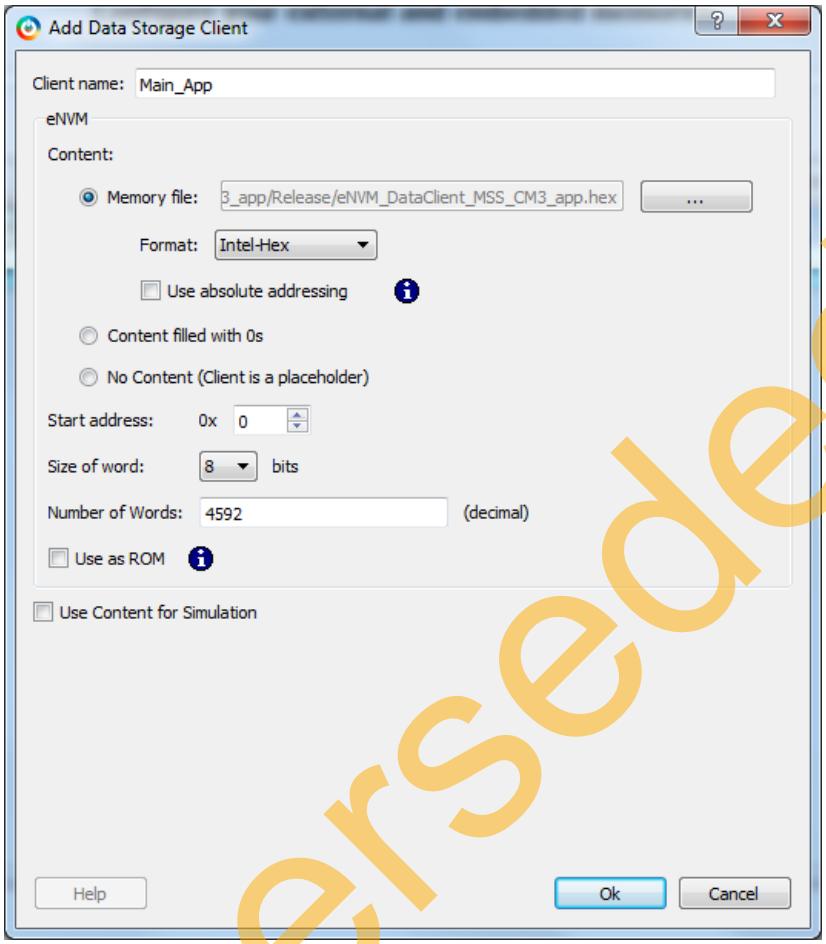

***Figure 15* • Add Data Storage Client Window**

5. Browse for the .hex file generated in the SoftConsole project after building an executable image in the Release Mode. The generated executable image is in the **Release** folder under SoftConsole project workspace (see Figure 16 on page 16 and Figure 17 on page 17).

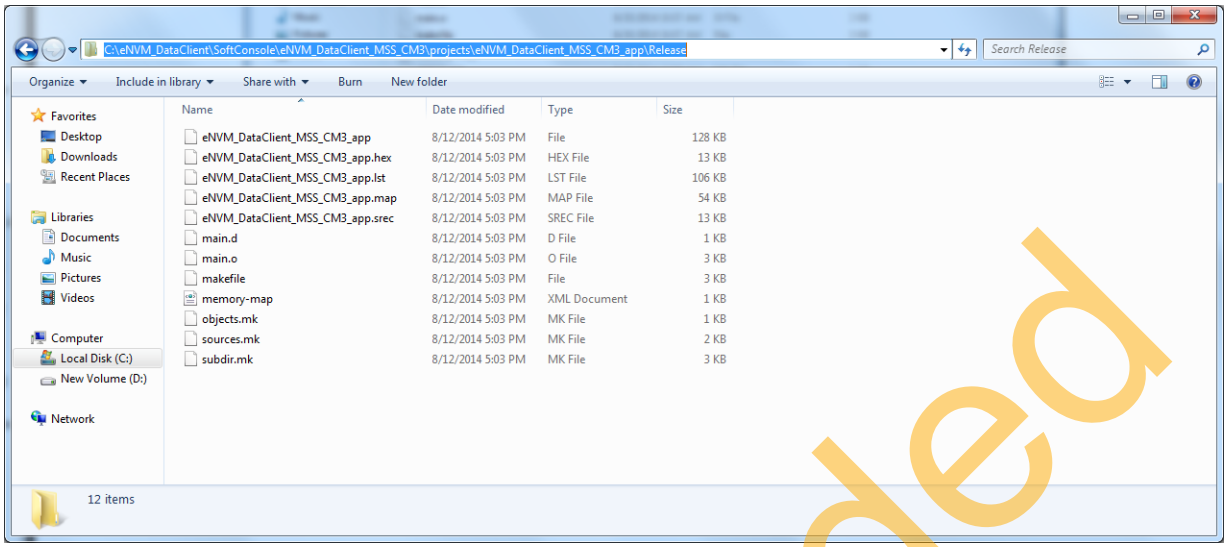

Figure 16 • **Executable Image in Release Folder**

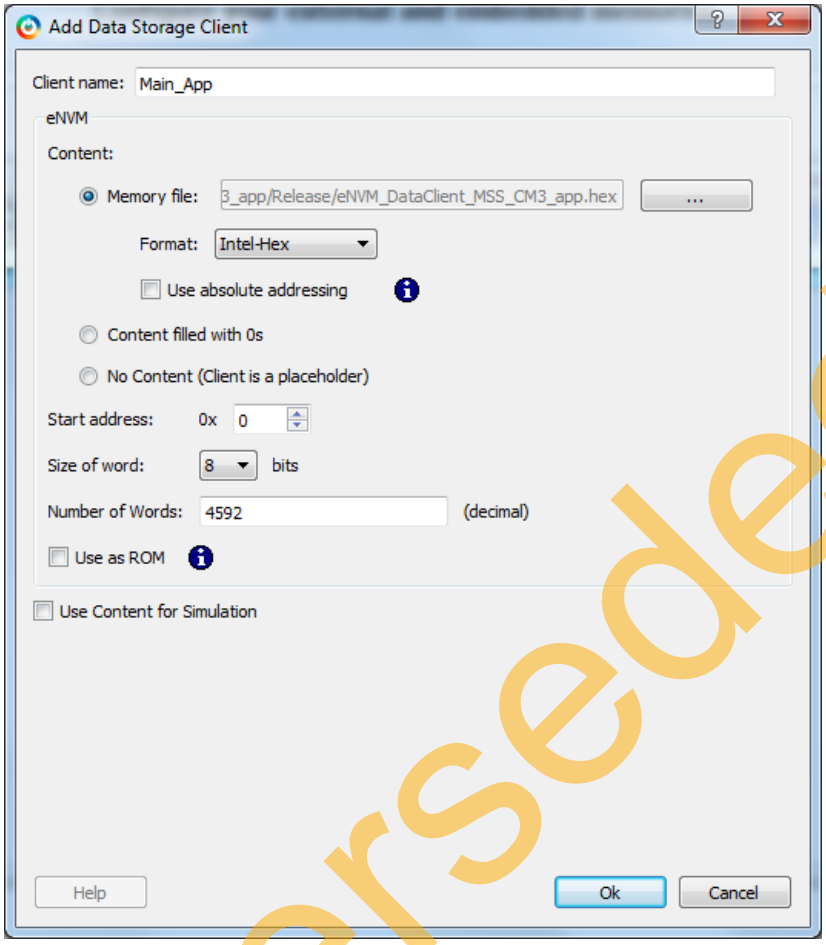

Figure 17 • **Add Data Storage Client Window**

6. Click **Ok** in the **Add Data Storage Client** window.
7. Add **LookUpTable.hex** file to eNVM Data Storage Client as shown in Figure 18 on page 18 and Figure 19 on page 19.

Note: LookUpTable.hex file contains the pre-calculated data of the factorial of numbers 2, 3, and 4 (2, 6, and 24 in Hexa decimal format). This is a non-executable image and is stored in eNVM. The factorial data is fetched by the Main application.

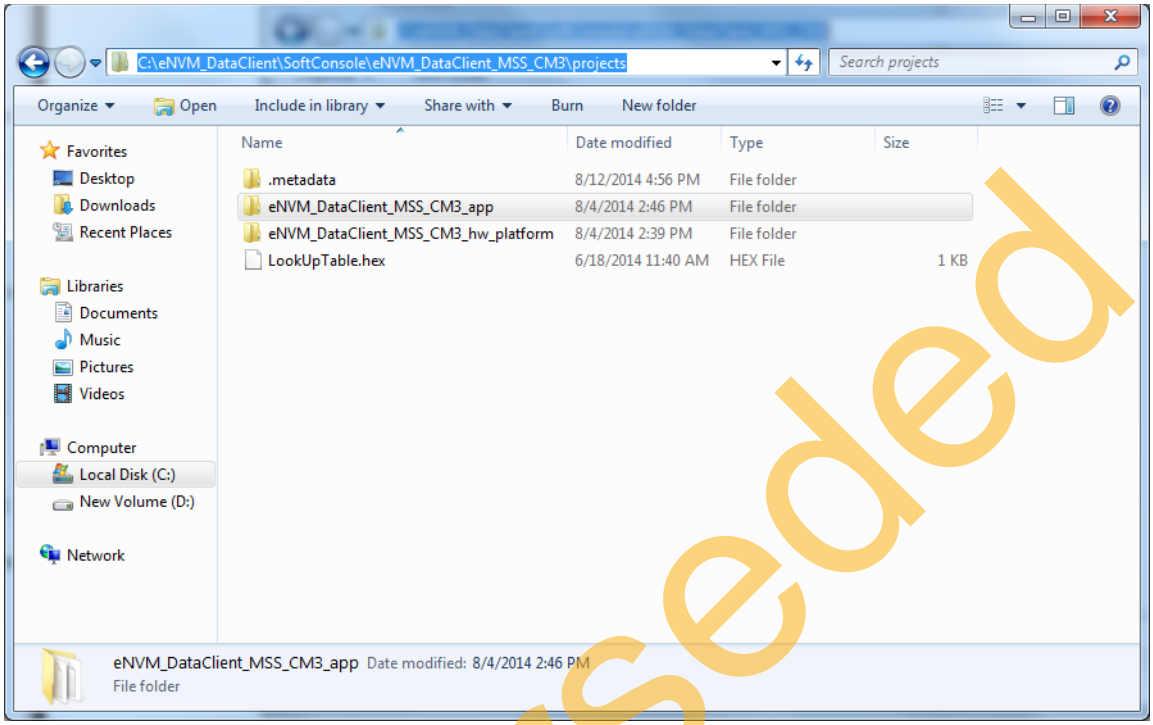

Figure 18 • **LookUpTable.hex File**

8. Select the Start address as 20000 in the **Add Data Storage Client** window (see Figure 19). This is the address in eNVM, where LookUpTable.hex file is stored. Click **Ok**.

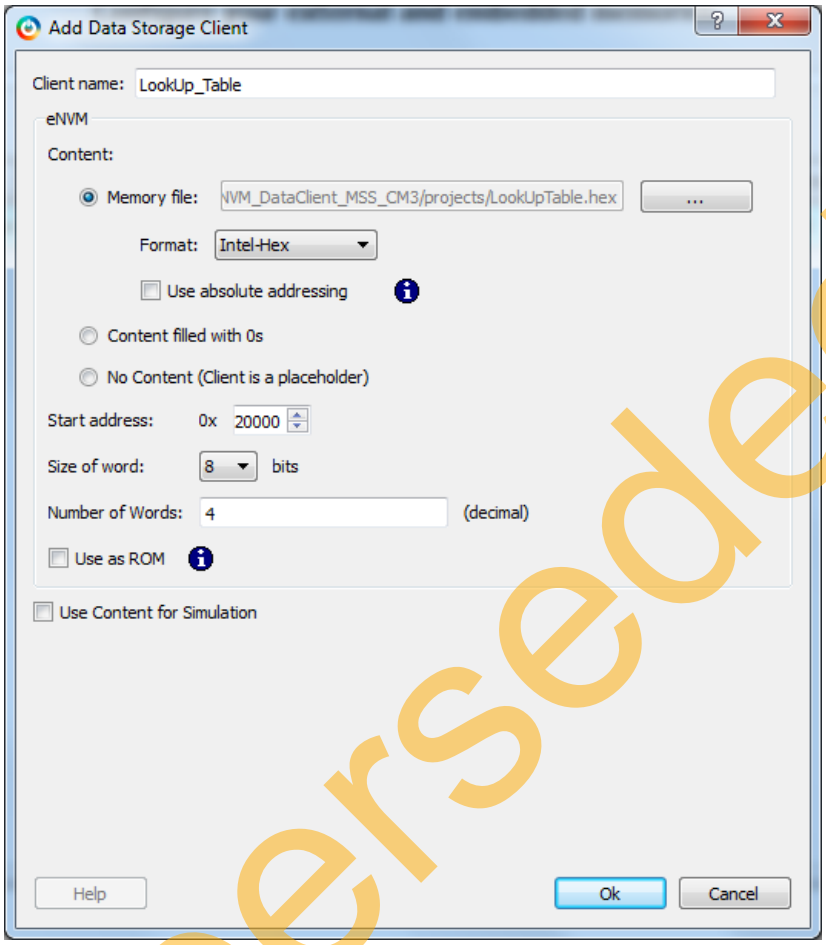

***Figure 19* • Add Data Storage Client Window**

9. In the **System Builder - Memories** page (see Figure 20), keep the other settings to default and click **Next**.

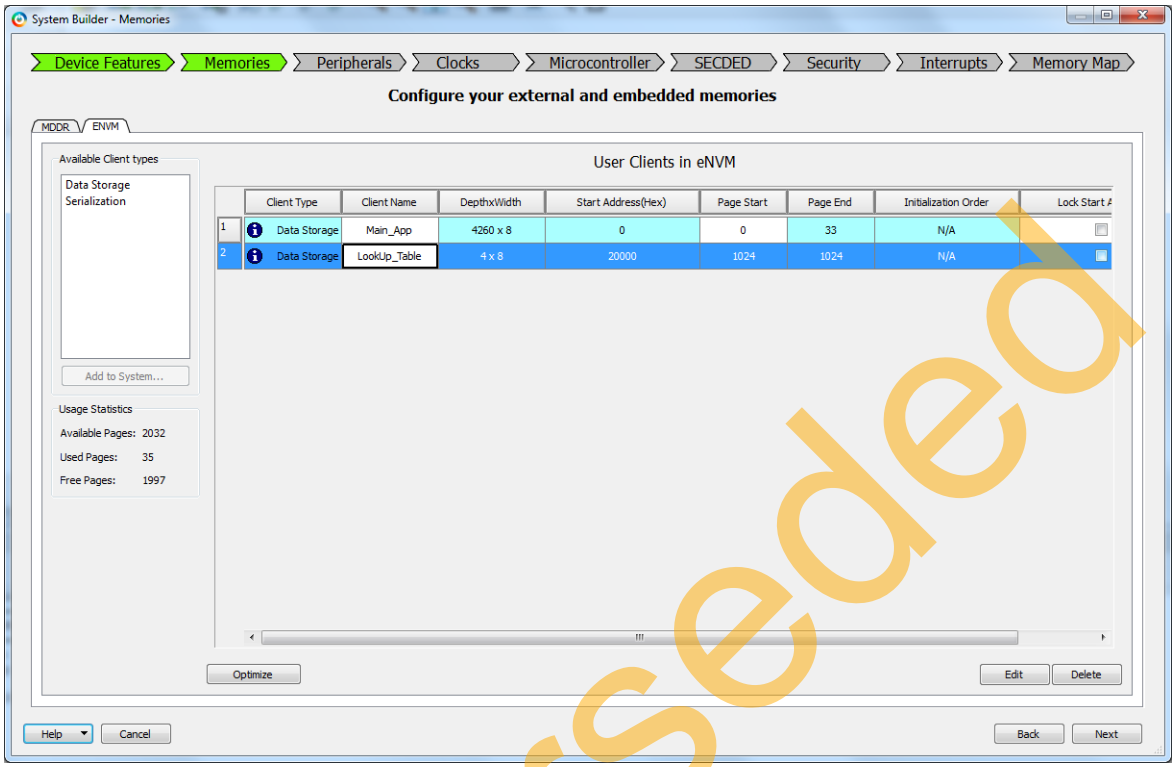

***Figure 20* • System Builders - Memories Page**

10. Save **eNVM_DataClient_top** and regenerate the **eNVM_DataClient_top** component by clicking **Generate Component** in SmartDesign.
11. Click **Generate Bitstream** to complete the remaining steps to generate fdb file (synthesis, place-and-route).

12. Click **Run PROGRAM Action** to program the SmartFusion2 SoC Evaluation Kit (see Figure 21 and Figure 22).

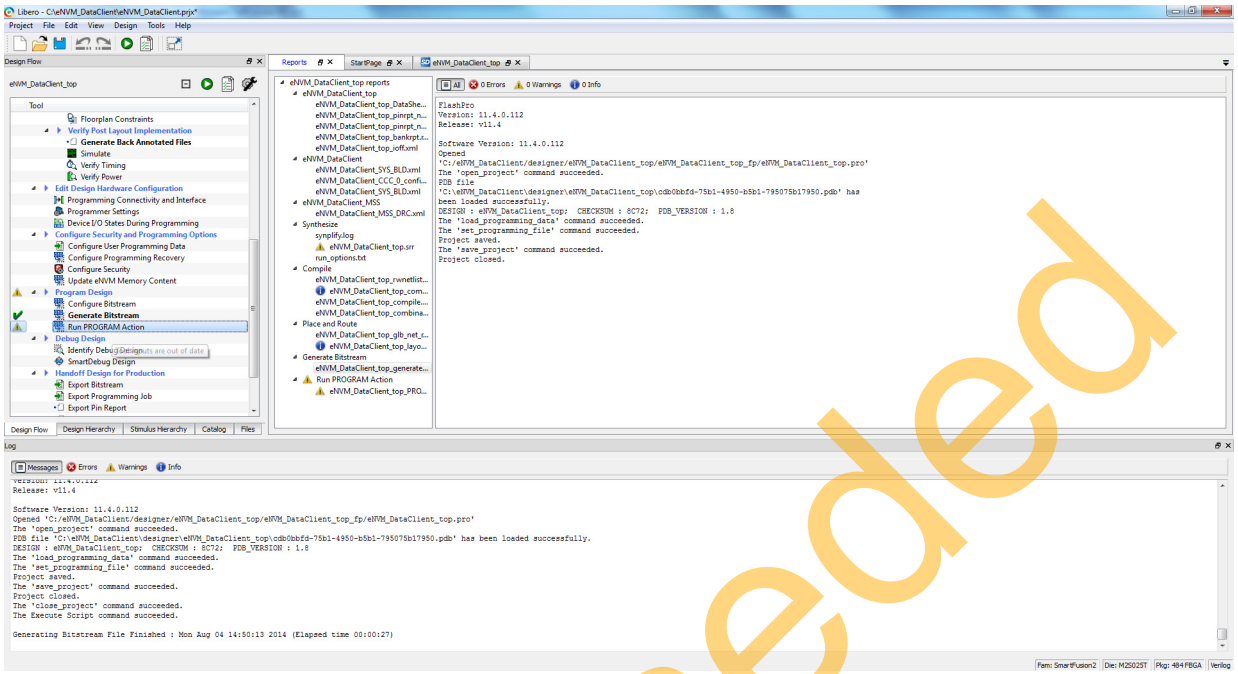

***Figure 21* • Selecting Run PROGRAM Action**

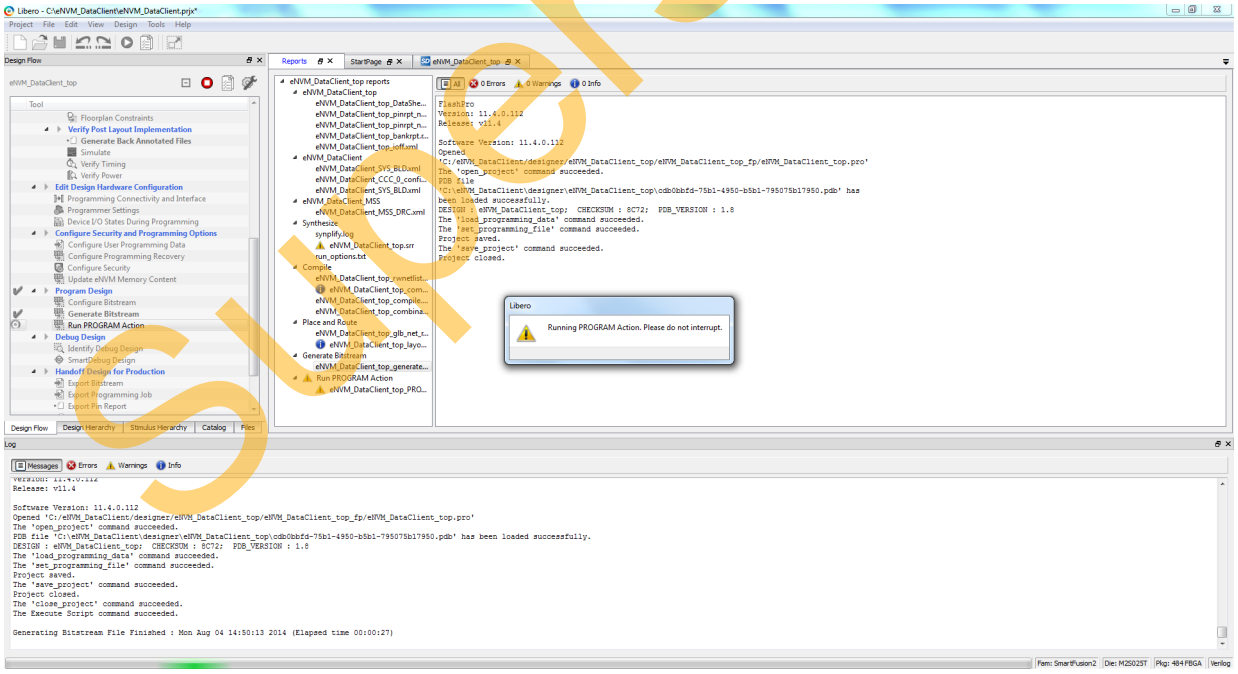

***Figure 22* • Programming SmartFusion2 SoC Evaluation Kit**

Updating eNVM Memory Content

This option is available in Libero design flow to load the updated .hex files after compiling the application code in SoftConsole. Without this option, every time the SoftConsole application is built, System Builder eNVM Configurator is required for updating the eNVM client with the new .hex file generated, which is time consuming.

If the new .hex file generated from the SoftConsole application project is larger than the previous .hex file, **Update eNVM Memory Content** throws an error. Go back to the System Builder and manually update the Hex file.

The following steps describe how to use **Update eNVM Memory Content** option:

1. Choose **Update eNVM Memory Content** option from Libero design flow.

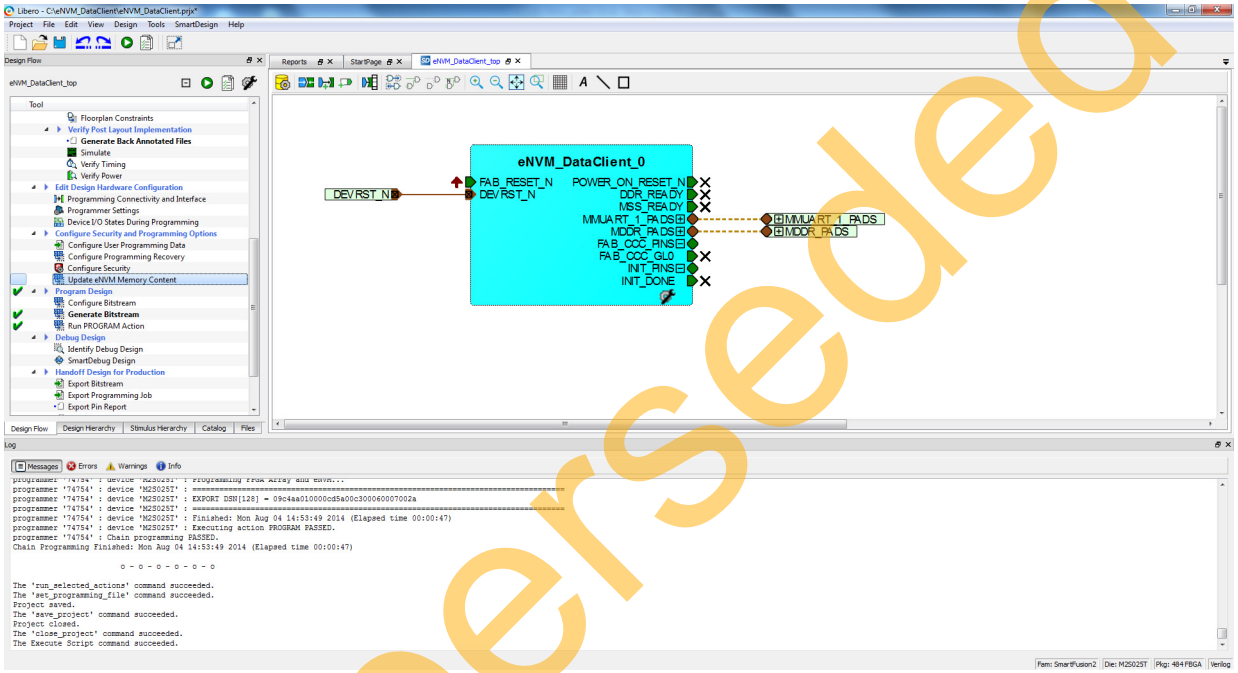

***Figure 23* • Choosing Update eNVM Memory Content Option from Libero Design Flow**

2. Update the **eNVM Memory Content** window as shown in Figure 24.

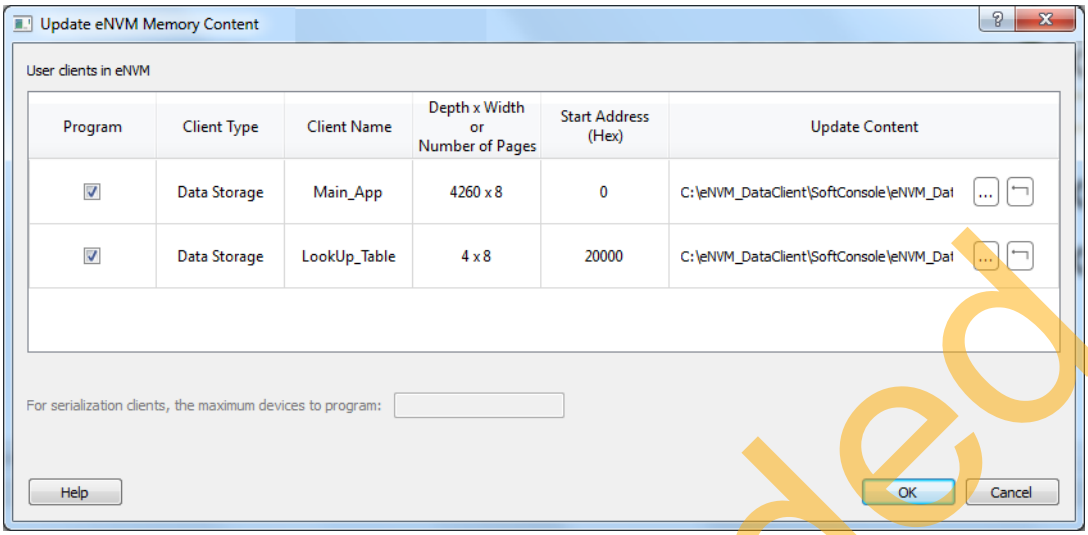

Figure 24 • **eNVM Memory Content Window**

3. Click **Update Content** (see Figure 24) and browse through the updated .hex file generated in the Release Mode from the SoftConsole application project (see Figure 25). Click **Ok.**

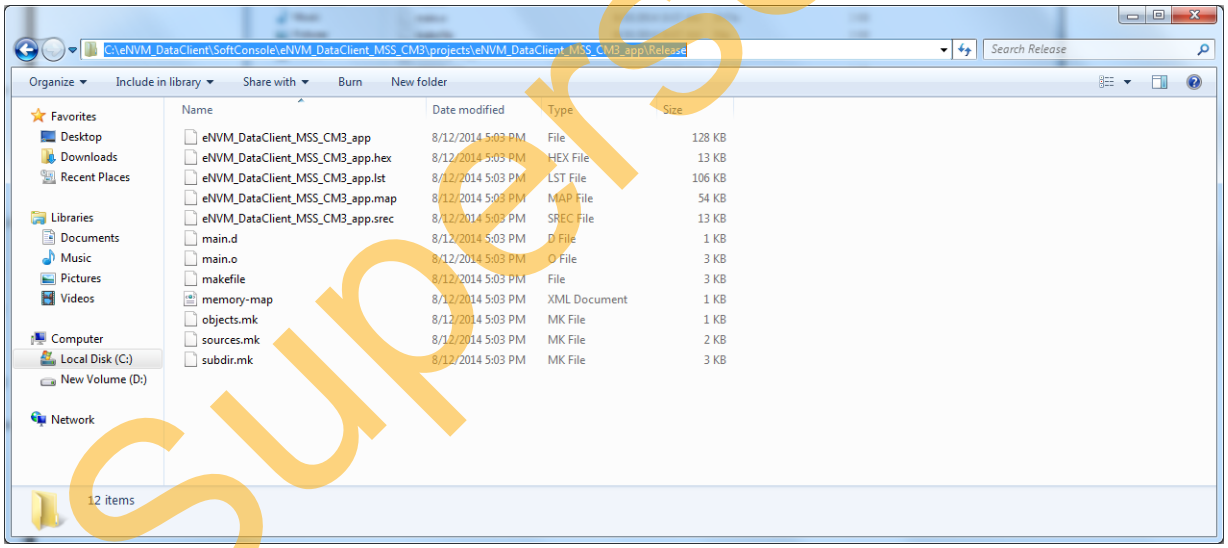

Figure 25 • **Updated Hex File in Release Folder**

4. Click **Generate Bitstream** to complete the remaining steps to generate fdb file (synthesis, place-and-route).
5. Click **Run PROGRAM Action** to program the SmartFusion2 SoC Evaluation Kit.

Running the Design on SmartFusion2 SoC FPGA Evaluation Kit

The following steps describe how to run the design:

1. Connect the FlashPro4 programmer to the J59 connector of SmartFusion2 SoC FPGA Evaluation Kit.
2. Connect one end of the USB mini-B cable to the J24 connector provided on the SmartFusion2 SoC FPGA Evaluation Kit.and the other end to the host PC. Ensure that the USB to UART bridge drivers are automatically detected (can be verified in the Device Manager as shown in Figure 26).

Note: Copy the COM port number for serial port configuration. Ensure that the COM port location is specified as **on USB Serial Converter D**, as shown in Figure 26.

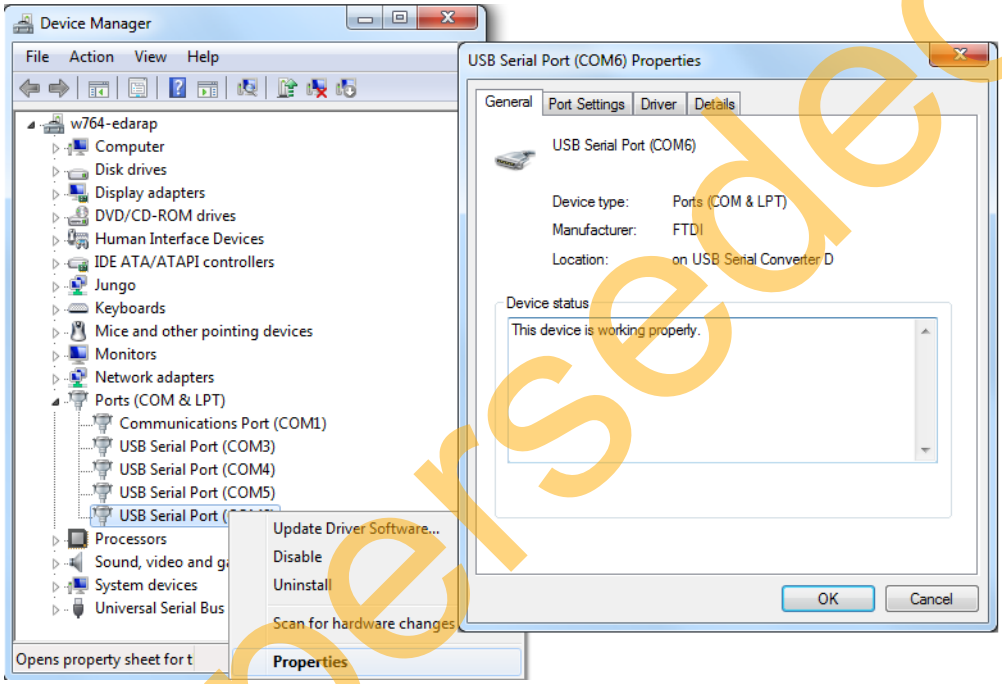

***Figure 26* • Device Manager Window**

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. Connect the power supply to the J18 connector and change the power supply switch SW7 to ON.
5. Start the HyperTerminal program with a baud rate of 57600, 8 data bits, 1 stop bit, no parity, and no flow control. If the PC does not have HyperTerminal program, use any free serial terminal emulation program such as PuTTY or Tera Term. Refer to the Configuring Serial Terminal Emulation Programs tutorial for configuring the HyperTerminal, Tera Term, and PuTTY.
6. Program the SmartFusion2 SoC FPGA Evaluation Kit Board with the provided programming file using the FlashPro software (Give the *.stp file path. Refer to "Appendix A - Design and Programming Files" on page 47).

Note: This step is required if *.stp file in the design folders is used. In Libero design, **Run PROGRAM Action** programs the *.stp file to the Board.

7. After successful programming, press SW6 switch to reset the board.
8. The serial terminal program displays the user options as shown in Figure 27.

   Select option 1 to get Factorial of number 2.

   Select option 2 to get Factorial of number 3.

Select option 3 to get Factorial of number 4.

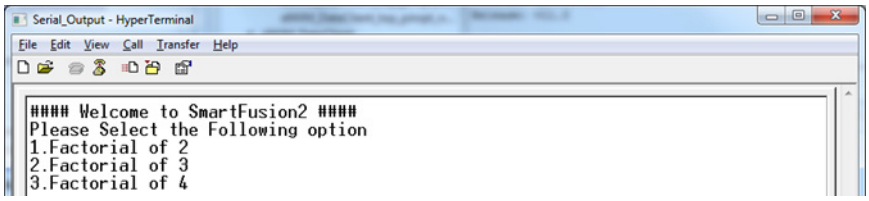

Figure 27 • **Selecting Factorial of a Number - User Options**

The application image fetches the Factorial Value from the Look up table (Non-executable image) and displays it as shown in Figure 28.

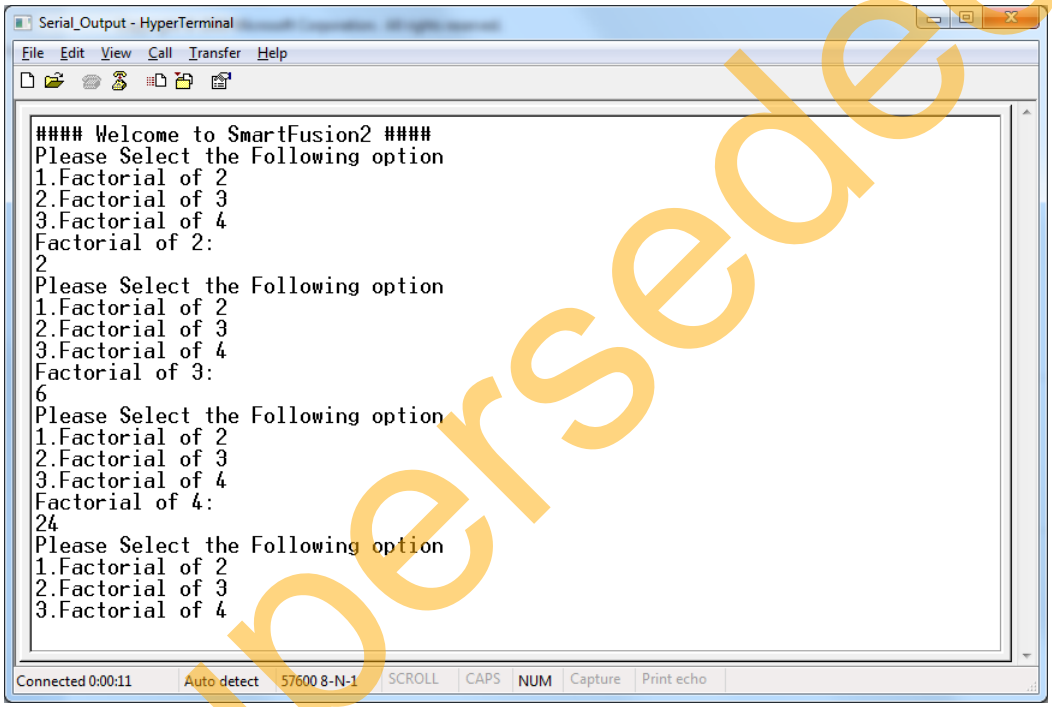

Figure 28 • **Displaying Factorial Value Based on User Option**

Building Multiple Executable Images in Release Mode

The design files are available for download from the following path in the Microsemi website:
http://soc.microsemi.com/download/rsc/?f=sf2_release_mode_programming_11p4_DF

This design contains the eNVM data storage clients for Multiple Hex images. It runs the images in eNVM and eSRAM, and the LEDs blink in various patterns. This design demonstrates how to build multiple images in the release mode and store them in eNVM.

The following steps describe how to build multiple application executable image in the release mode:

1. Open the standalone SoftConsole IDE.

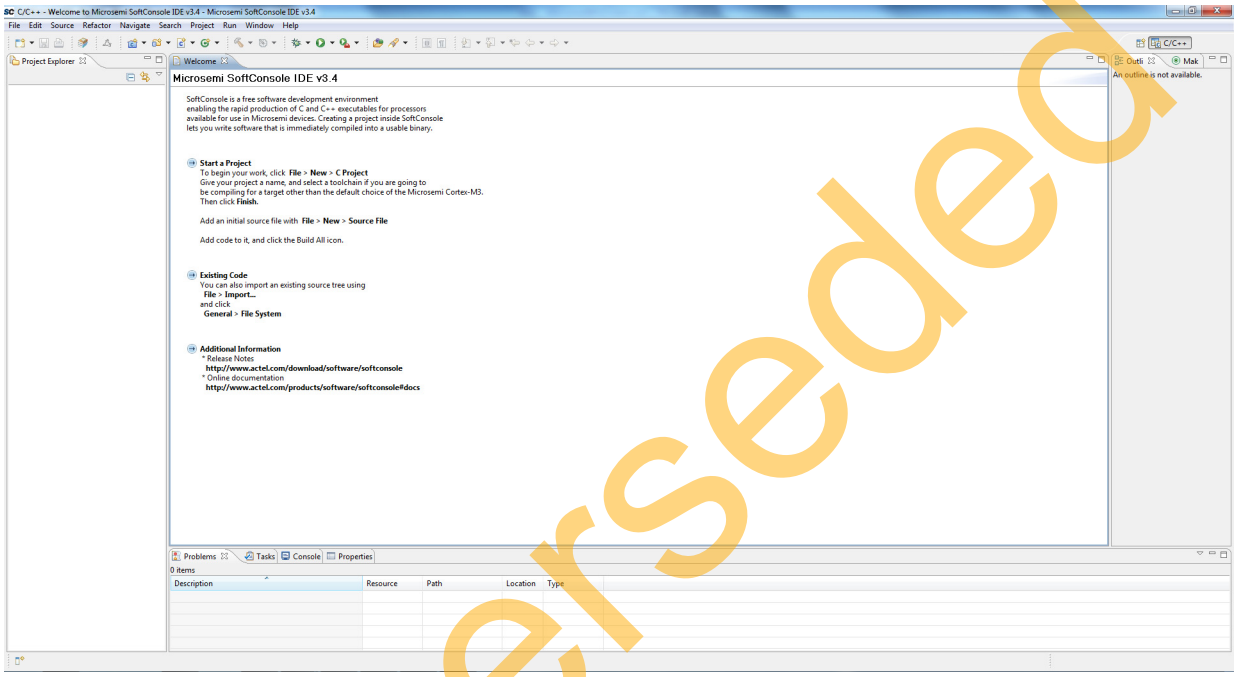

Figure 29 • **SoftConsole IDE**

2. Right-click the **Project Explorer** window and choose **Import** as shown in Figure 30.

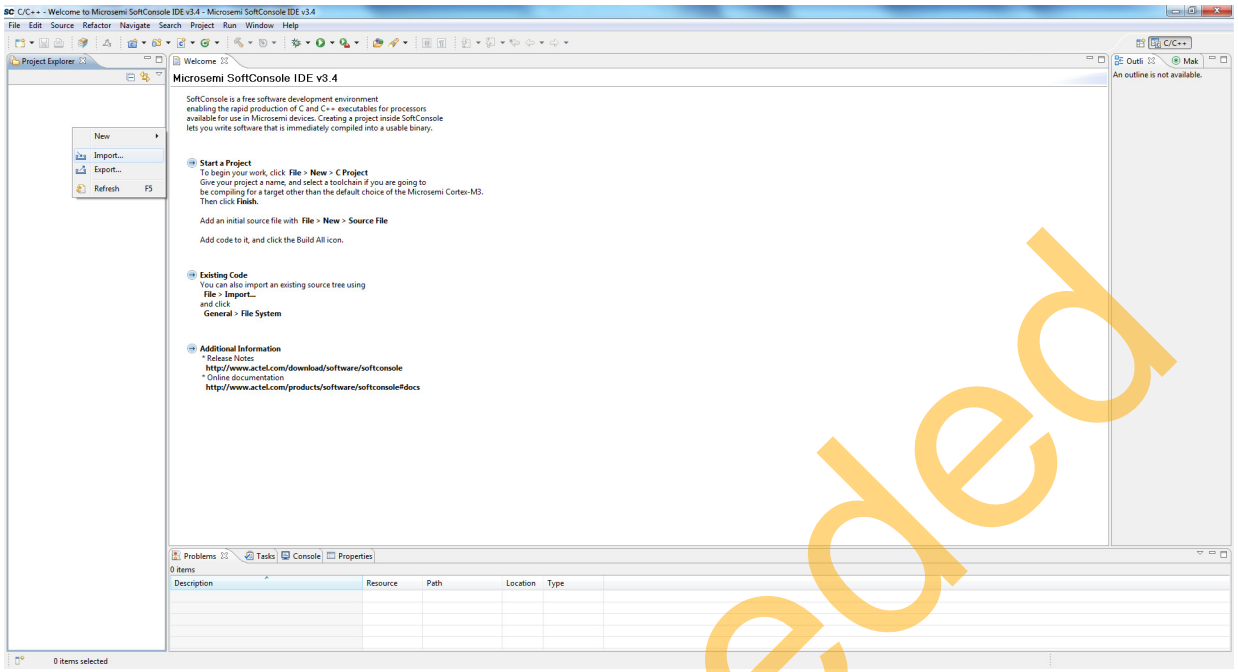

Figure 30 • **SoftConsole IDE - Import**

3. Select **General > Existing Projects into Workspace** as shown in Figure 31.

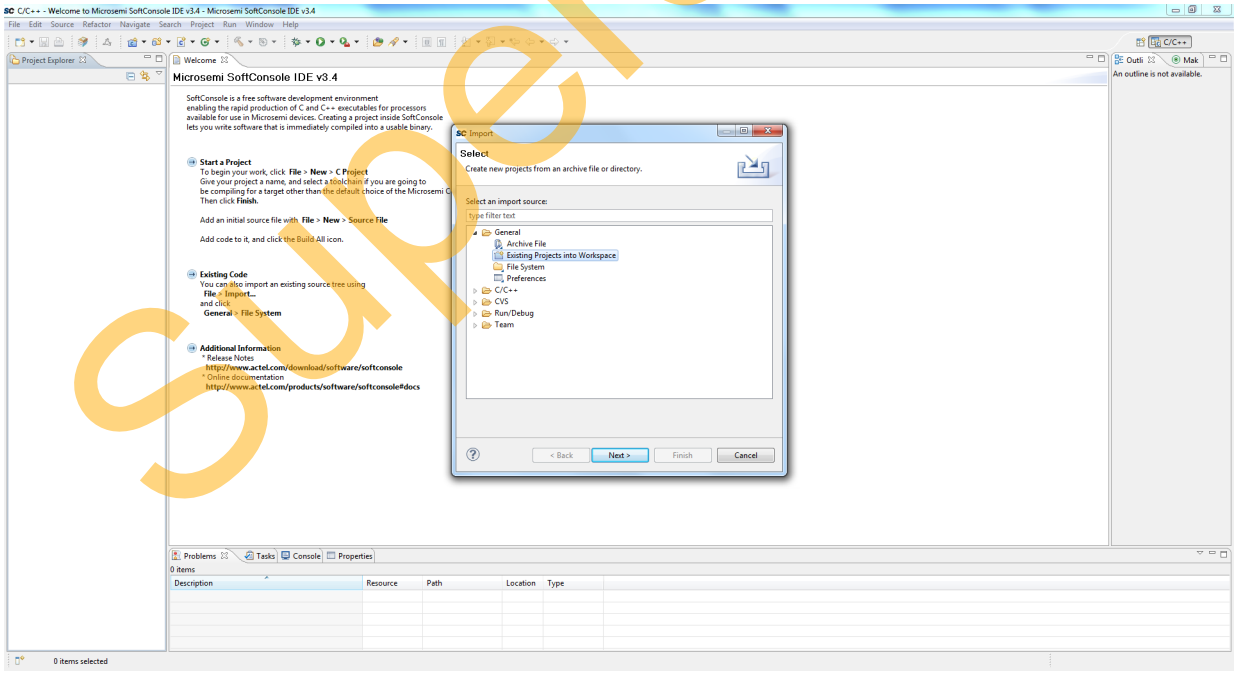

Figure 31 • **SoftConsole IDE - General**

Import window is shown in Figure 32.

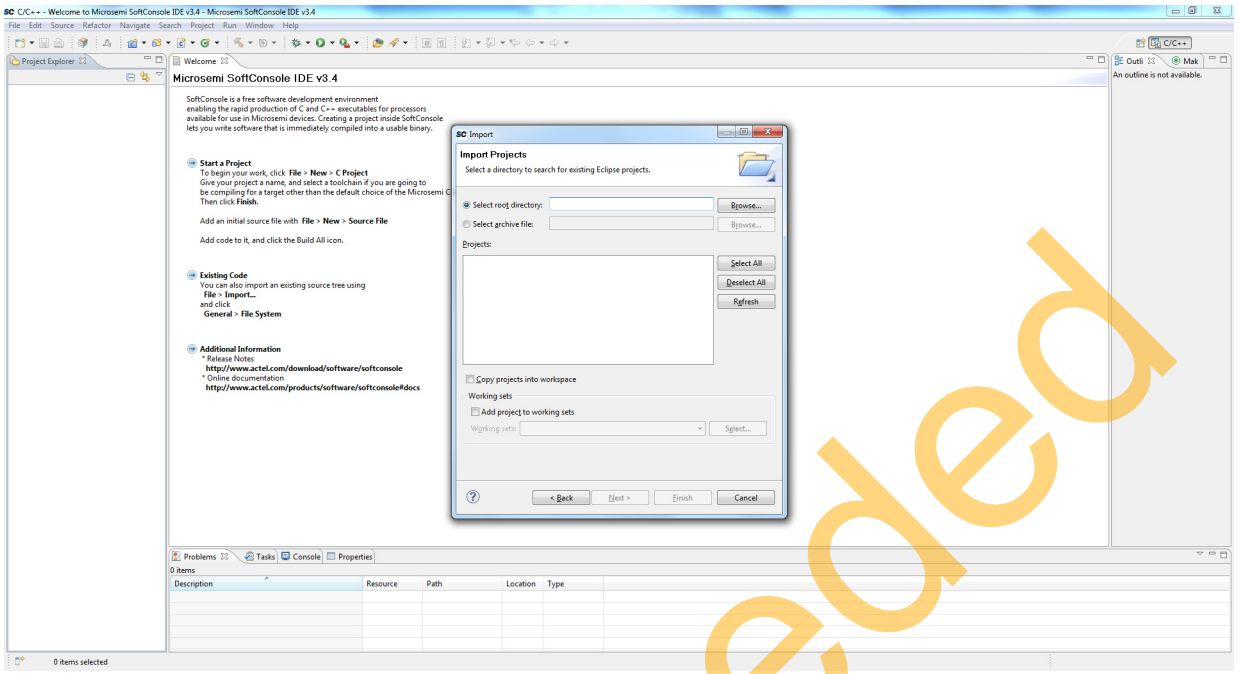

***Figure 32* • SoftConsole IDE - Browse**

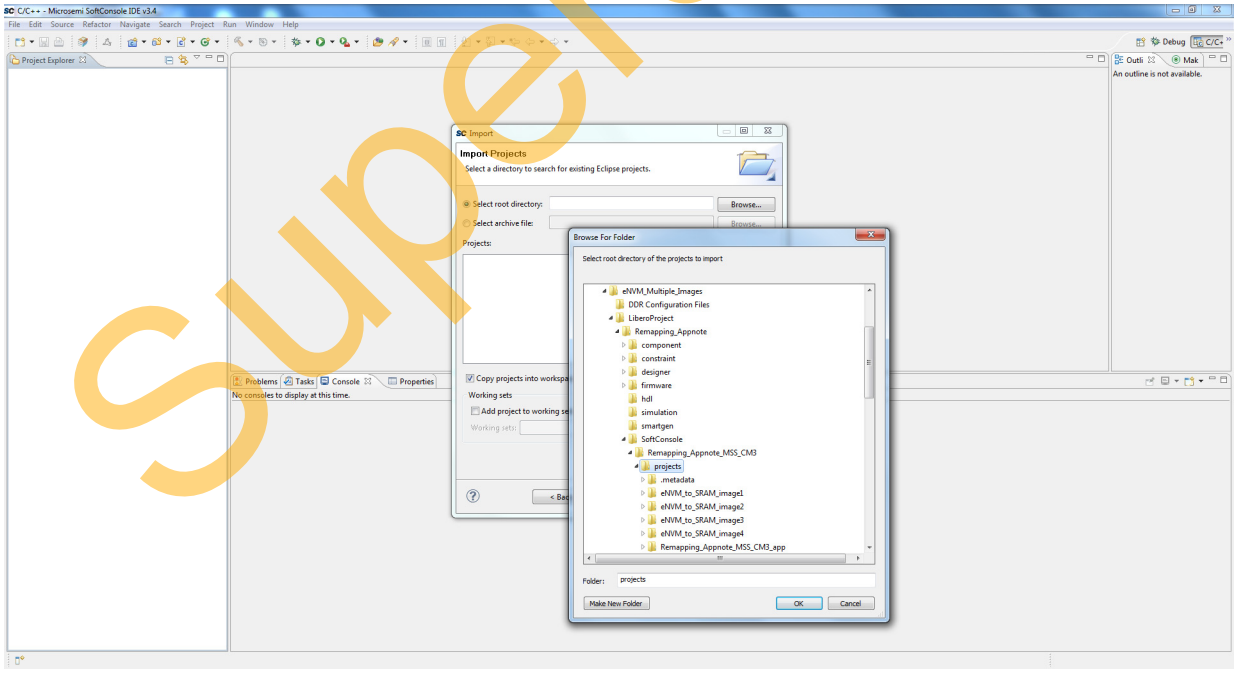

***Figure 33* • SoftConsole IDE - Browsing for App Folder**

4. Ensure that **Copy projects into workspace** check box is checked as shown in Figure 34. Click **Finish.**

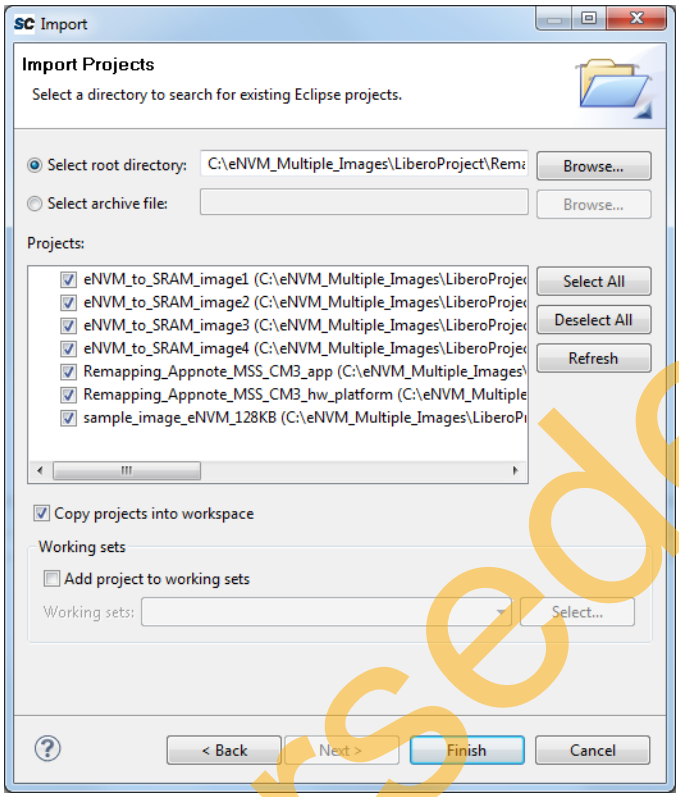

***Figure 34* • SoftConsole IDE - Creating Work Space in App Folder**

The SoftConsole perspective is shown in Figure 35.

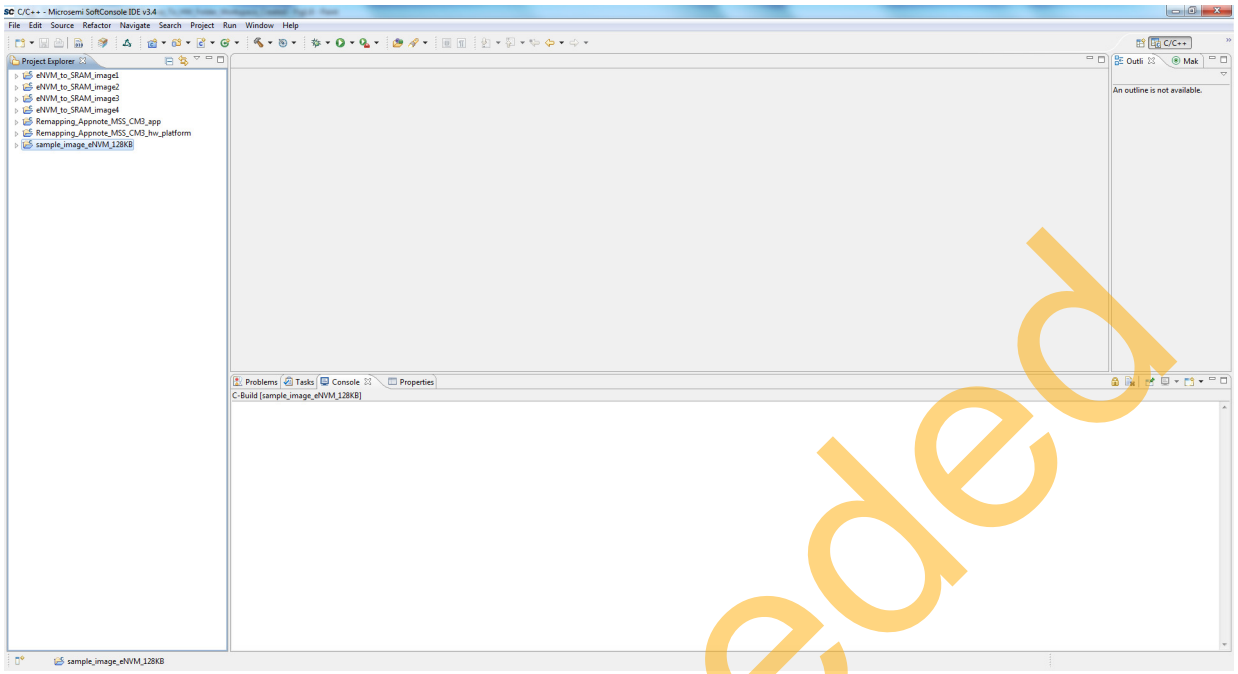

Figure 35 • **SoftConsole Workspace**

5. Right-click both project names in the **Project Explorer** view and select **Build Configurations > Set Active > Release** as shown in Figure 36.

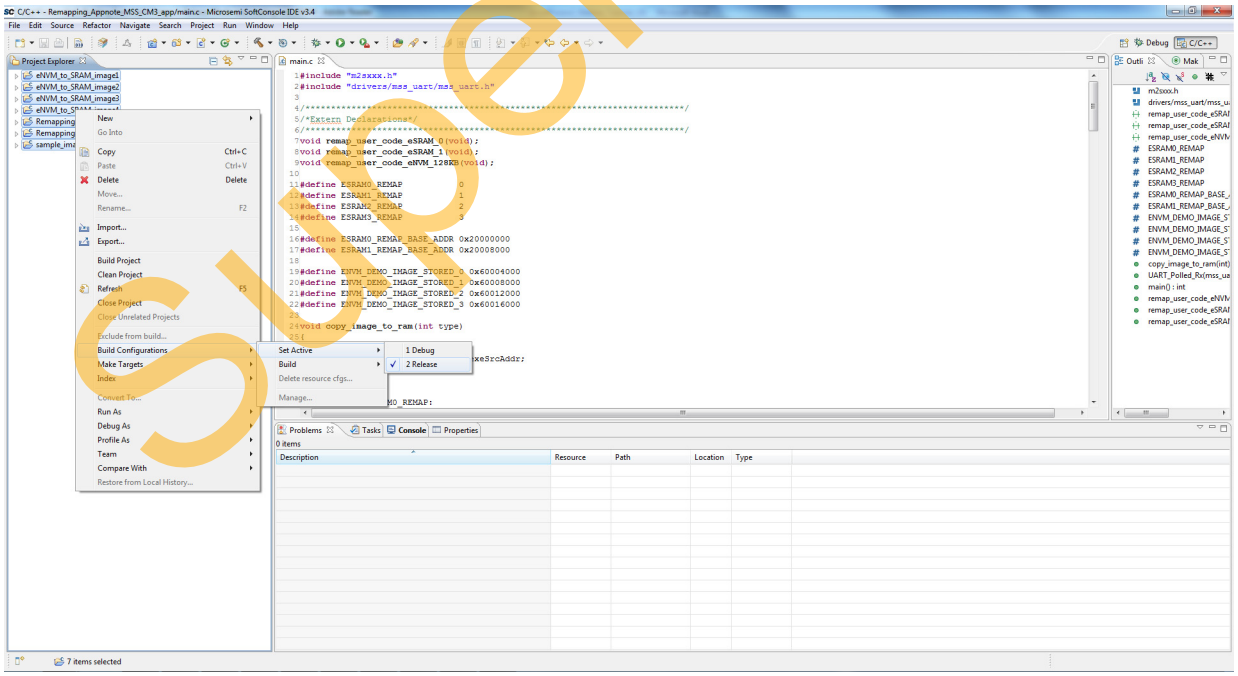

Figure 36 • **Building Configuration**

6. Go to **Project > Build All** and build the project as shown in Figure 37.
   .hex file is generated in the **Release** folder created in the project folder.

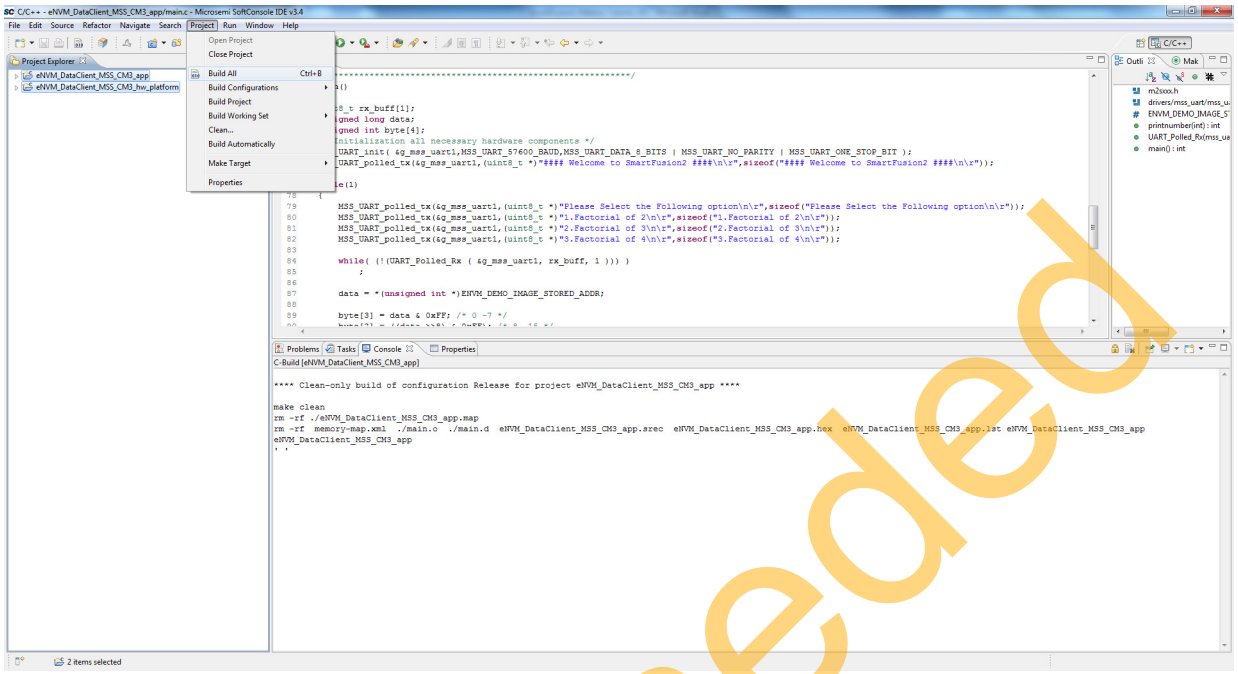

Figure 37 • **Build All**

Loading Multiple Executable Images into eNVM

Multiple SoftConsole application projects generate multiple executable images. These executable images are stored in eNVM.

The following steps describe how to load the generated application executable images into eNVM of the SmartFusion2 SoC device using the System Builder eNVM Configurator:

1. Open the Libero SoC hardware project **Remapping_Appnote_top** tab.

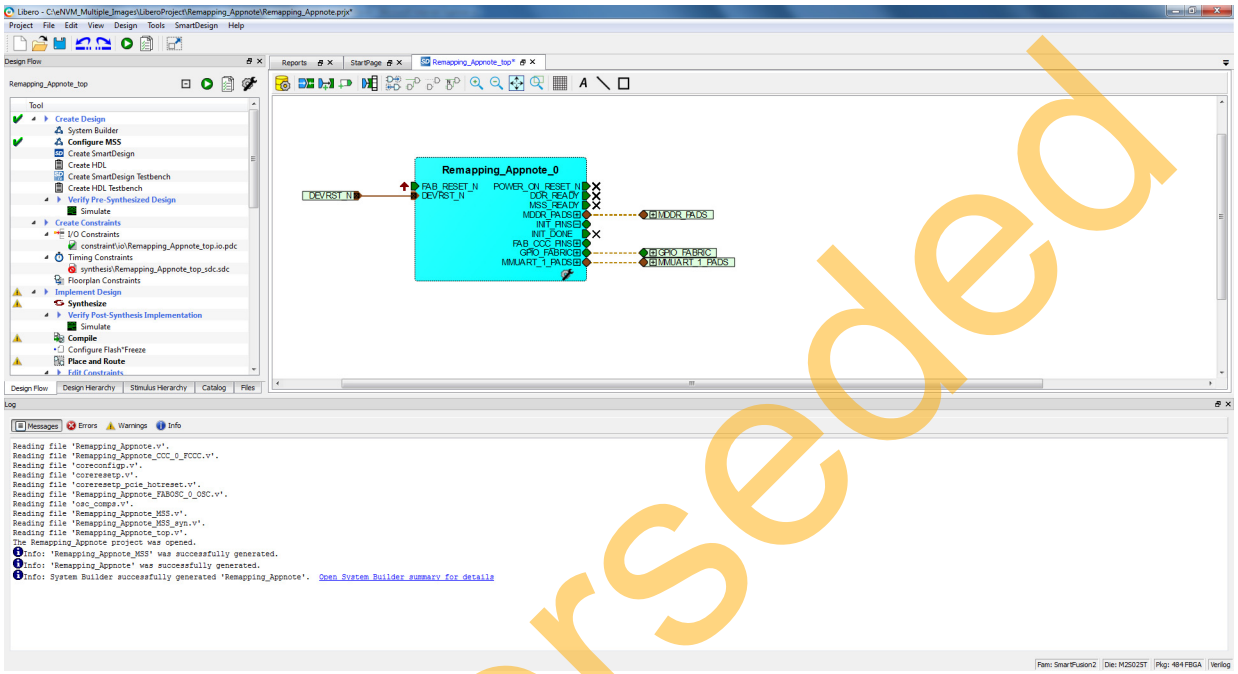

***Figure 38* • Remapping_Appnote_top Tab**

2. Double-click **Remapping_Appnote _0** (see Figure 38) and go to **System Builder - Memories** page (see Figure 39) to add the eNVM Data Storage client.

3. Select **Data Storage** under the **Available Client types** (see Figure 39) and click **Add to System...**. This opens **Add to Data Storage Client** window as shown in Figure 40 on page 34·

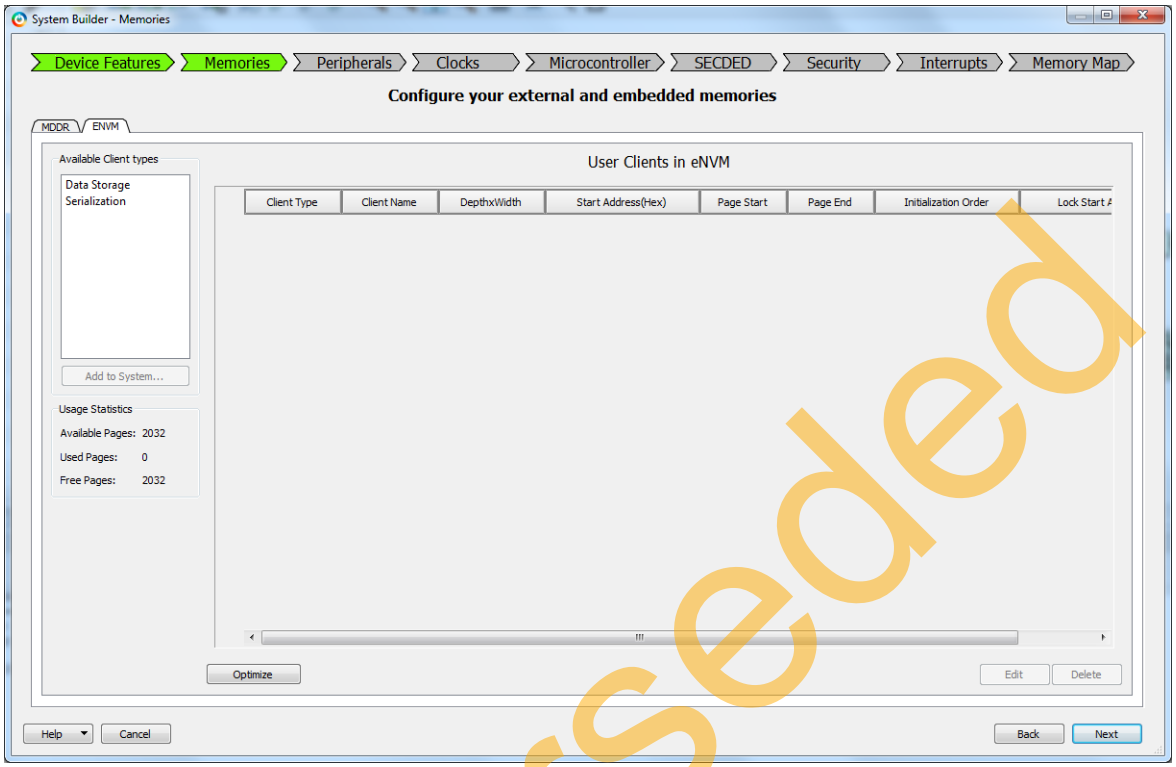

***Figure 39* • System Builder - Memories Page**

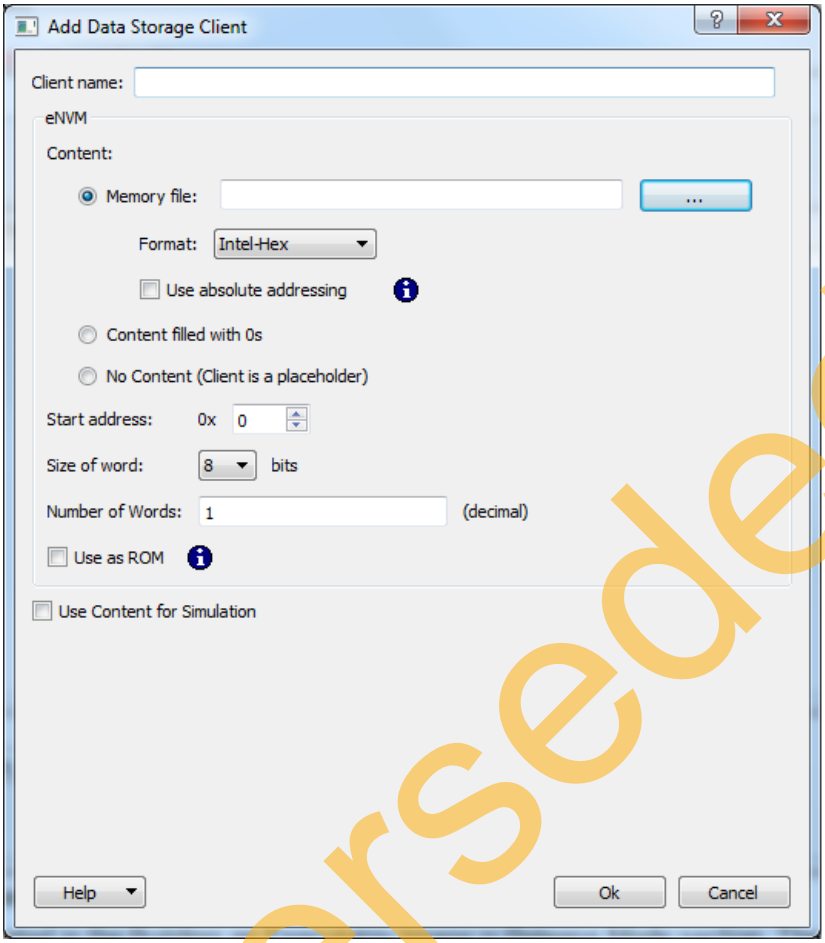

Figure 40 • **Add to Data Storage Client Window**

4. Enter a data storage client name in the **Add to Data Storage Client** window.

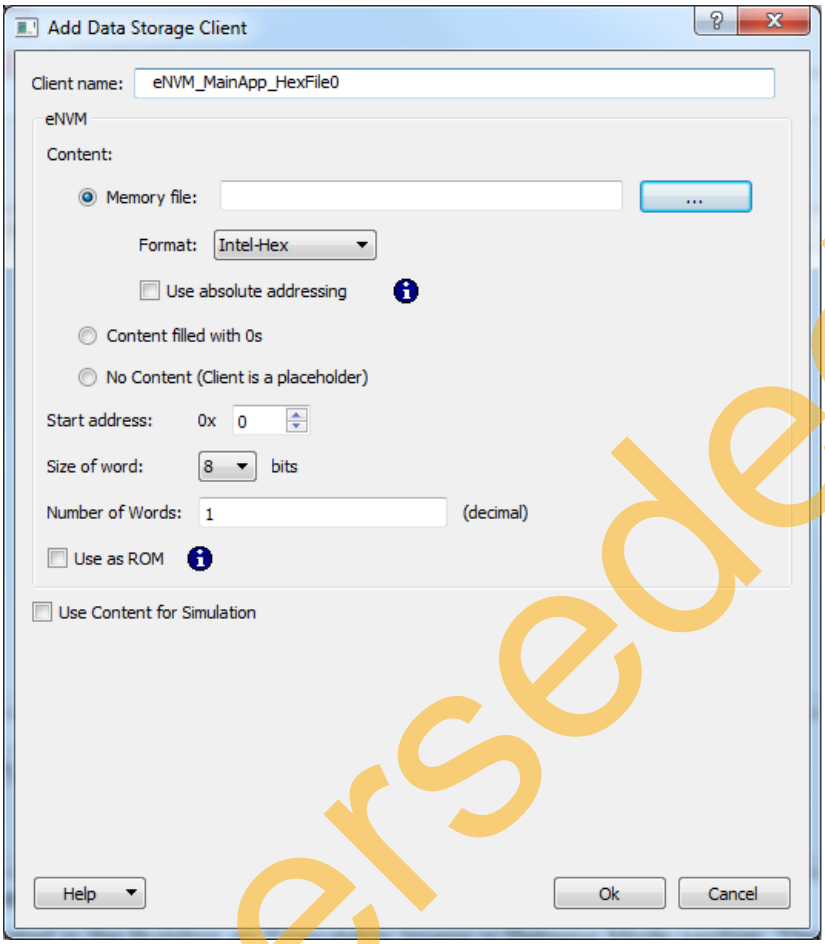

Figure 41 • **Add to Data Storage Client Window**

5. Browse for the .hex file generated after compiling the SoftConsole project in the Release Modes (see Figure 43 on page 37). The generated executable image is in the **Release** folder under SoftConsole project workspace (see Figure 42).

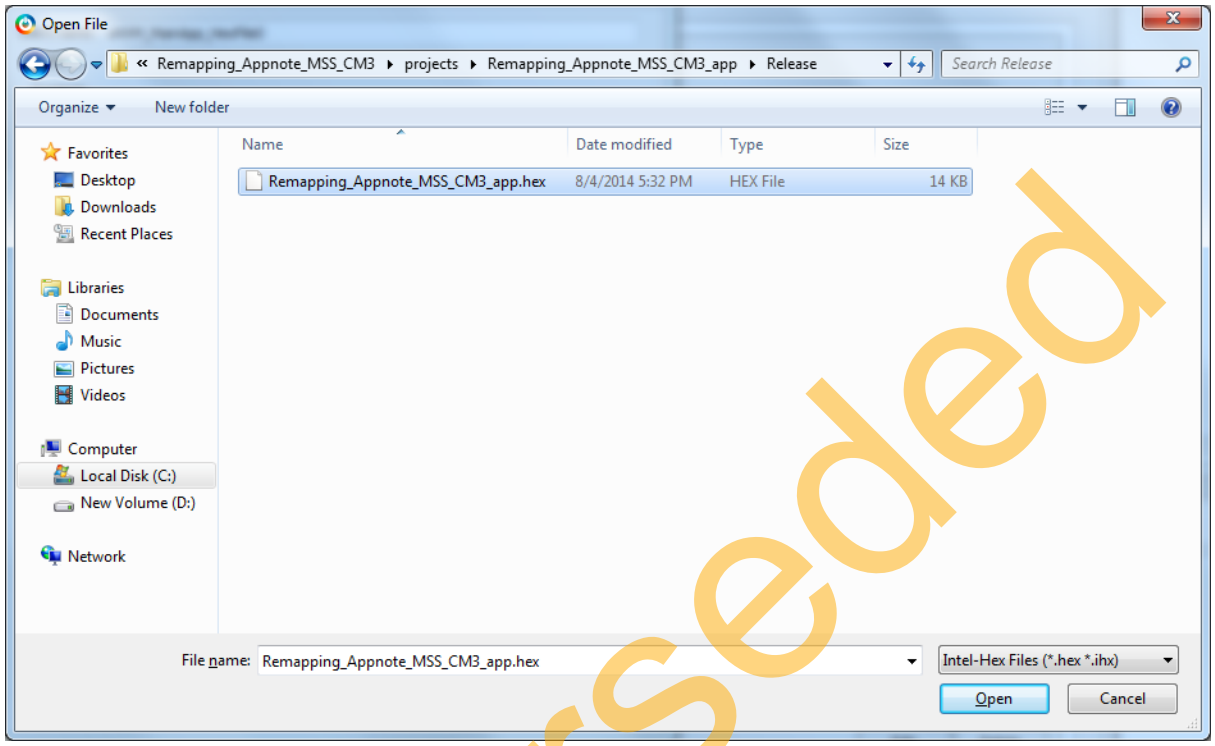

***Figure 42* • Executable Image in Release Folder**

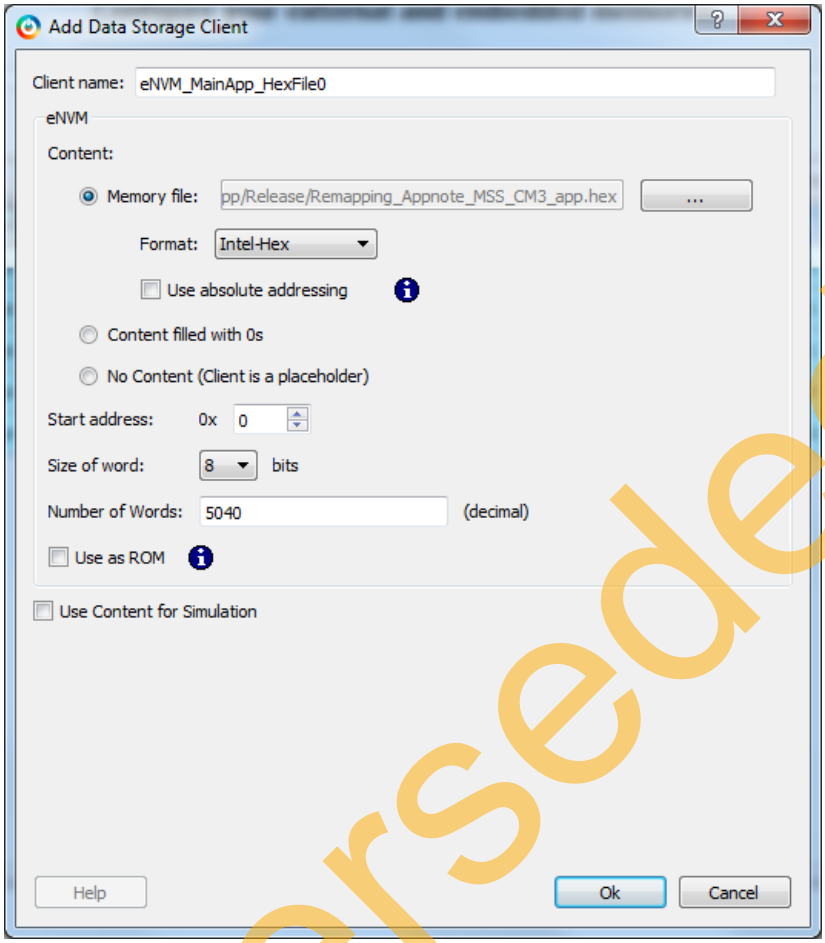

Figure 43 • **Add Data Storage Client Window**

6. Click **Ok** in the **Add Data Storage Client** window.
7. Add the remaining Hex Files with their corresponding start addresses as shown in Figure 45.

Table 2 lists the SoftConsole application projects in the Release mode, their .hex files, and their mapping to the eNVM Client Name.

Table 2 • **SoftConsole Application Projects and their Mapping to eNVM Client**

ENVM Client Name	SoftConsole Release Mode Path	Hex File Name	Start Address
eNVM_MainApp_HexFile0	Remapping_Appnote_MSS_CM3_app\Release	Remapping_Appnote_MSS_CM3_app.hex	0x0
eNVM_to_SRAM_HexFile1	eNVM_to_SRAM_image1\Release	eNVM_to_SRAM_image1.hex	0x4000
eNVM_to_SRAM_HexFile2	eNVM_to_SRAM_image2\Release	eNVM_to_SRAM_image2.hex	0x8000
eNVM_to_SRAM_HexFile3	eNVM_to_SRAM_image3\Release	eNVM_to_SRAM_image3.hex	0x12000
eNVM_to_SRAM_HexFile4	eNVM_to_SRAM_image4\Release	eNVM_to_SRAM_image4.hex	0x16000

Table 2 • **SoftConsole Application Projects and their Mapping to eNVM Client (continued)**

eNVM_128KB_HexFile5	sample_image_eNVM_128KB\Release	sample_image_eNVM_128KB.hex	0x20000

Note: Start address is the address in which the executable image is stored in eNVM. Provide the required memory size based on the total size of the executable image. Otherwise, the images overlap and displays error as shown Figure 44.

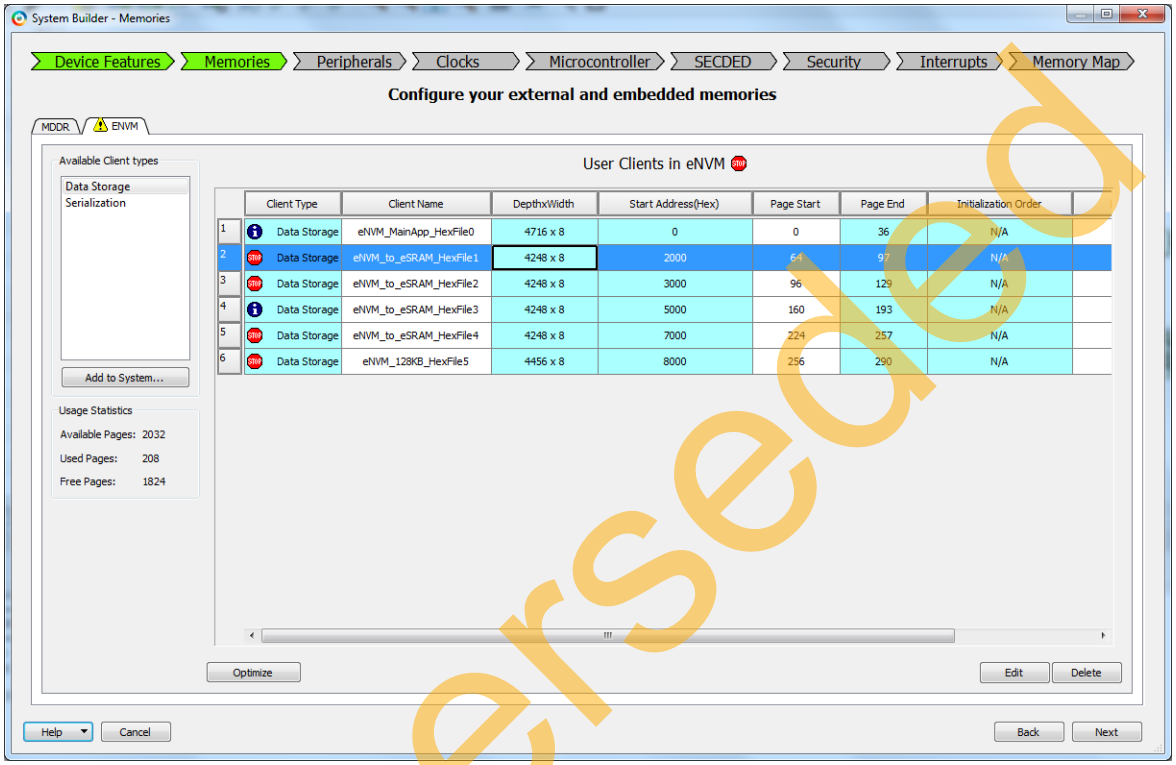

Figure 44 • **Error due to Image Overlap**

Figure 45 shows the correct memory locations chosen. It does not display any error.

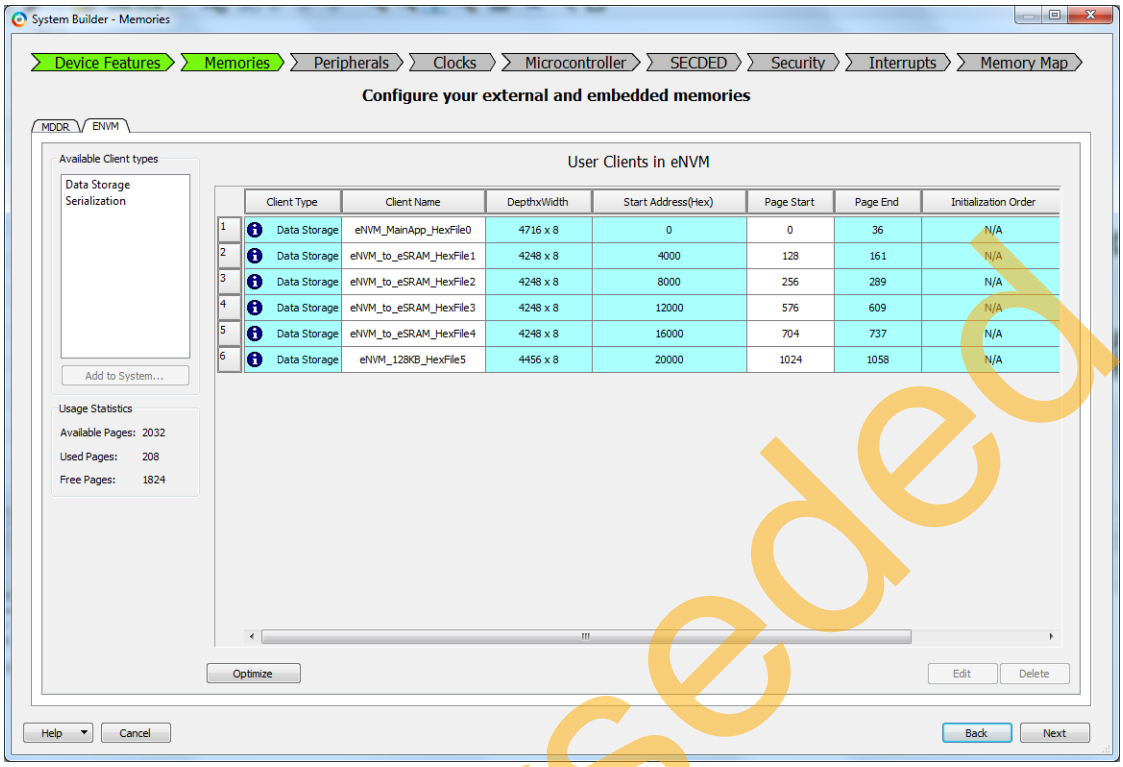

***Figure 45* • Hex Files with Correct Start Addresses**

8. In the **System Builder - Memories** page (see Figure 45), keep the other settings to default and click **Next**.
9. Save **Remapping_Appnote_top** and regenerate the **Remapping_Appnote_0** component by clicking Generate Component in SmartDesign.
10. Click **Generate Bitstream** to complete the remaining steps to generate fdb file (synthesis, place-and-route).

11. Click **Run PROGRAM Action** to program the SmartFusion2 SoC FPGA to initialize the eNVM with the memory file as shown in Figure 46 and Figure 47.

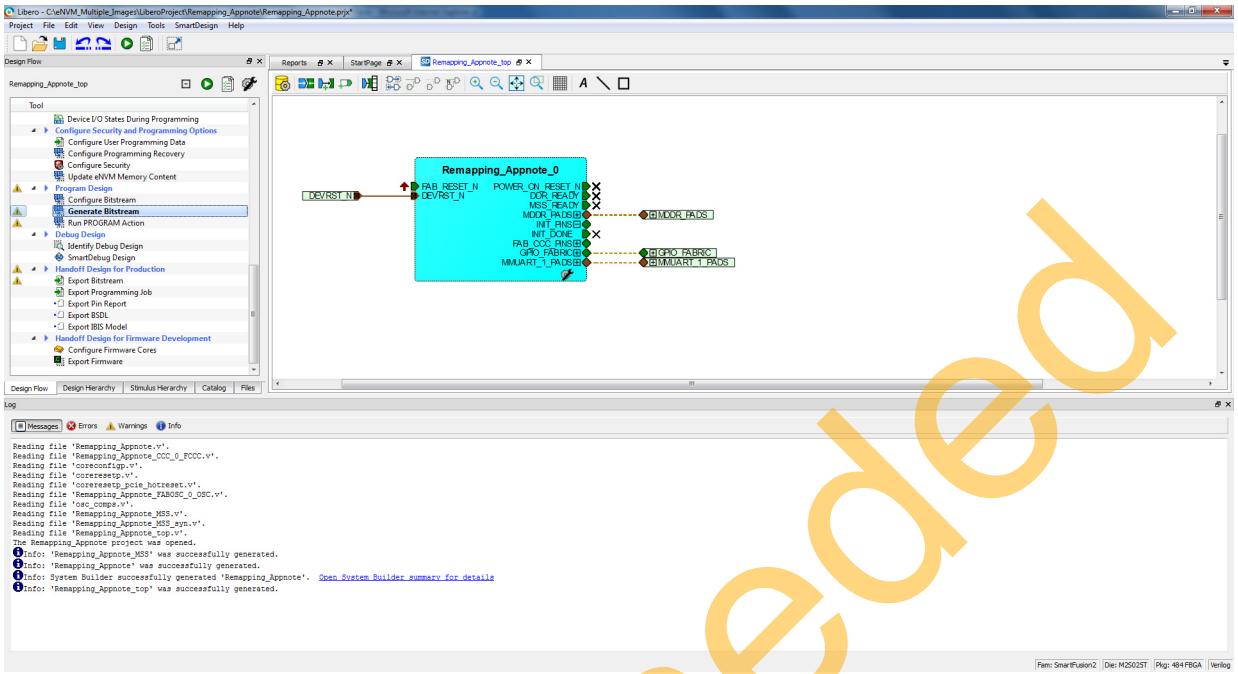

***Figure 46* • Regenerating Remapping_Appnote_0 Component**

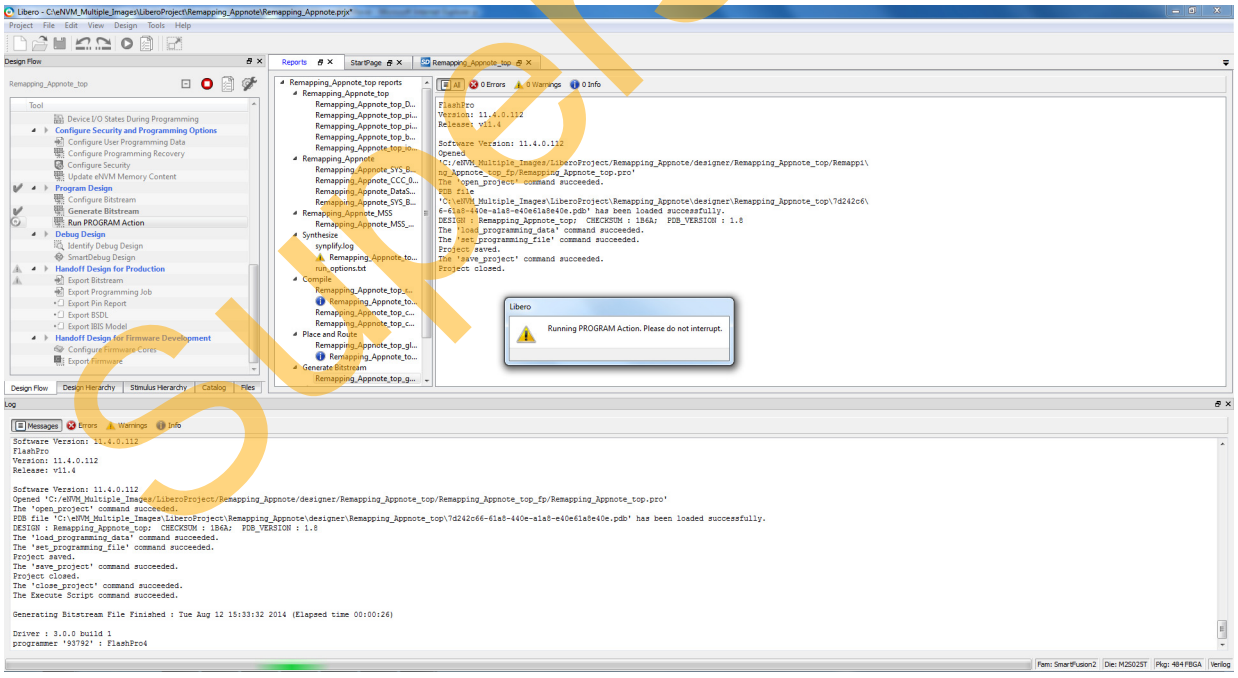

***Figure 47* • Programming SmartFusion2 SoC Evaluation Kit**

Updating eNVM Memory Content

This option is available in Libero design flow to load the updated .hex files after compiling the application code in SoftConsole. Without this option, every time the SoftConsole application is built, System Builder eNVM Configurator is required to update the eNVM client with the new .hex file generated, which may be time consuming.

If the new .hex file generated from the SoftConsole application project is larger than the previous .hex file, **Update eNVM Memory Content** throws an error. Go back to the System Builder and manually update the Hex file.

The following steps describe how to use **Update eNVM Memory Content** option:

1. Choose **Update eNVM Memory Content** option from Libero design flow.

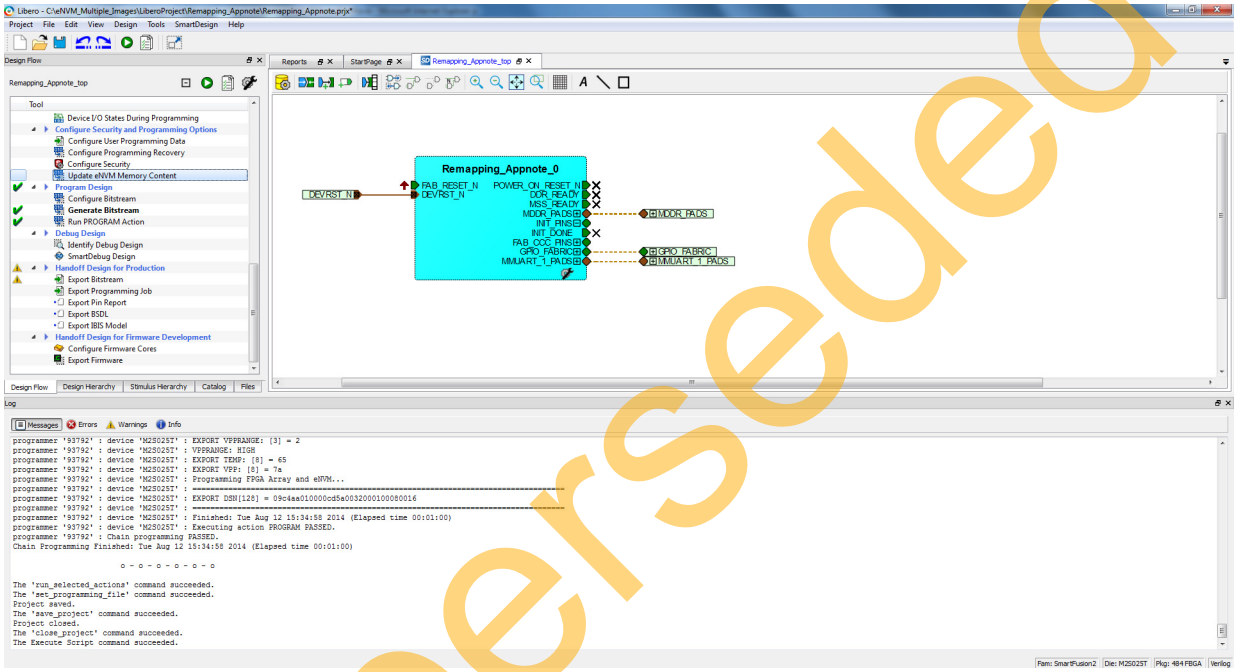

Figure 48 • **Update eNVM Memory Content Option**

2. Update **eNVM Memory Content** window as shown in Figure 49.

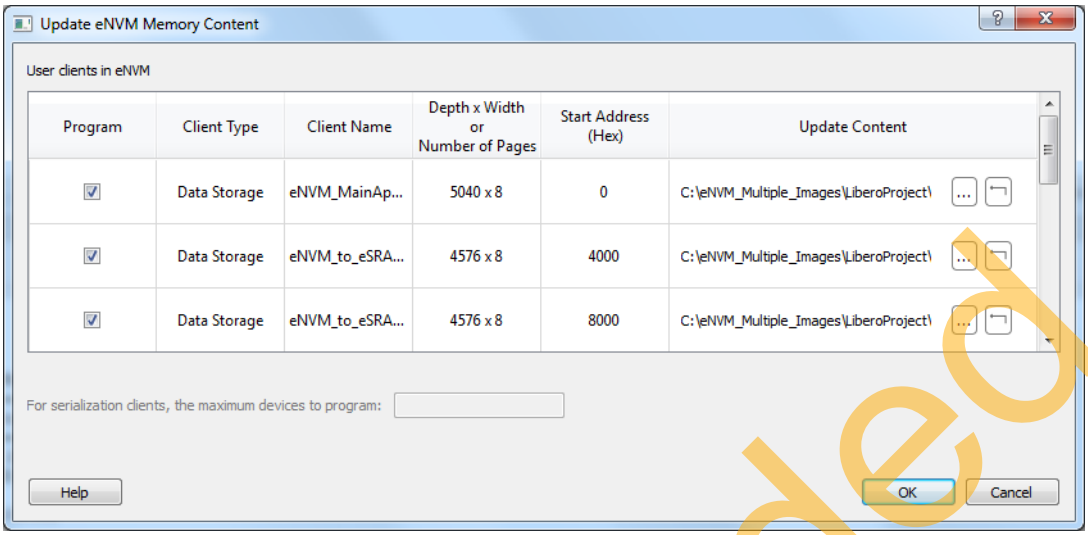

***Figure 49* • Update eNVM Memory Content Window**

3. Click **Update Content** (see Figure 49) and browse through the updated .hex file (see Figure 50 on page 42). Click **Ok**.

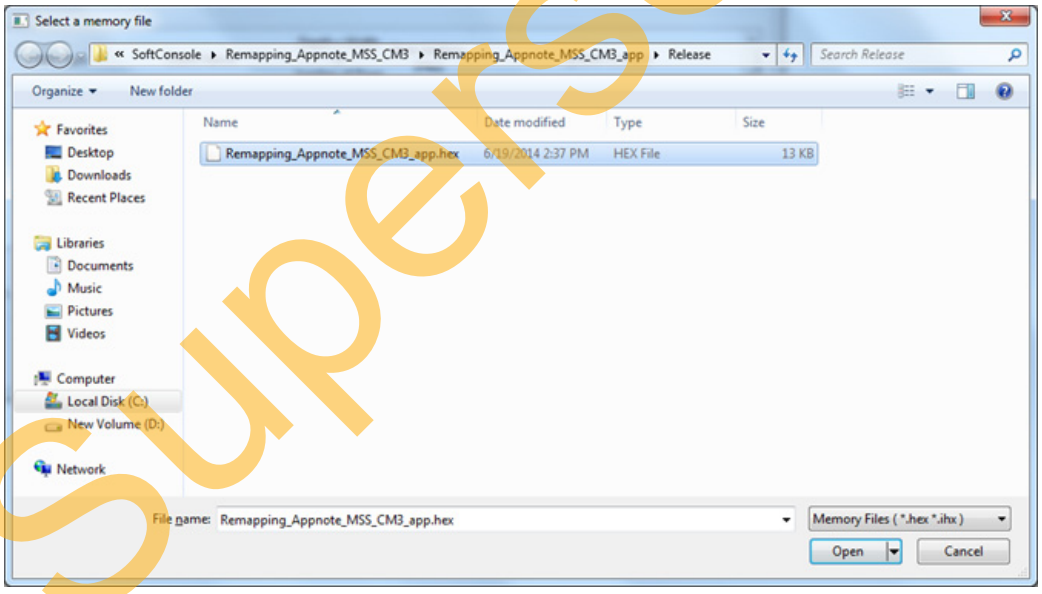

***Figure 50* • Browsing through Updated Hex File**

4. Update the .hex files for the remaining data storage clients.
5. Perform **Run PROGRAM Action.**

Running the Design on SmartFusion2 SoC FPGA Evaluation Kit

The following steps describe how to run the design:

1. Connect the FlashPro4 programmer to the J59 connector of SmartFusion2 SoC FPGA Evaluation Kit.
2. Connect one end of the USB mini-B cable to the J24 connector provided on the SmartFusion2 SoC FPGA Evaluation Kit and the other end to the host PC. Ensure that the USB to UART bridge drivers are automatically detected (can be verified in the Device Manager), as shown in Figure 51.

Note: Copy the COM port number for serial port configuration. Ensure that the COM port location is specified as **on USB Serial Converter D**, as shown in Figure 51.

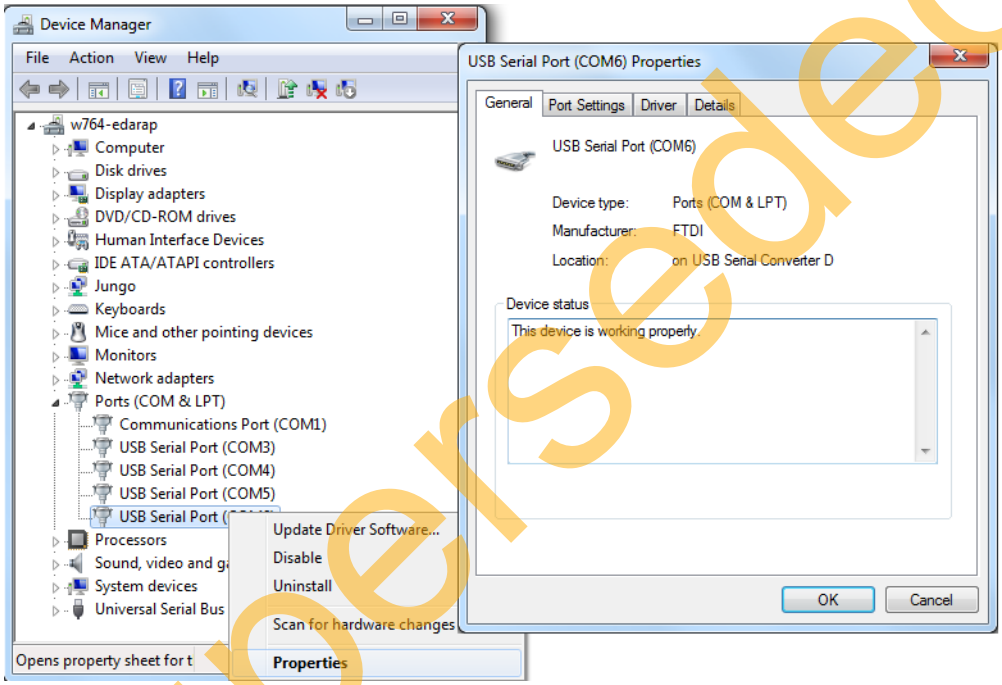

***Figure 51* • Device Manager Window**

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. Connect the power supply to the J18 connector and change the power supply switch SW7 to ON.
5. Start the HyperTerminal program with a baud rate of 57600, 8 data bits, 1 stop bit, no parity, and no flow control. If the PC does not have HyperTerminal program, use any free serial terminal emulation program such as PuTTY or Tera Term. Refer to the Configuring Serial Terminal Emulation Programs tutorial for configuring the HyperTerminal, Tera Term, and PuTTY.
6. Program the SmartFusion2 SoC FPGA Evaluation Kit Board with the provided programming file using the FlashPro software (Give the ***.**stp file path. Refer to "Appendix A - Design and Programming Files" on page 47).
7. This step is required if *.stp file in the design folders is used. In Libero design, **Run PROGRAM Action** programs the *.stp file to the Board.After successful programming, press SW6 switch to reset the board.

8. The serial terminal program displays the user options as shown in Figure 52.

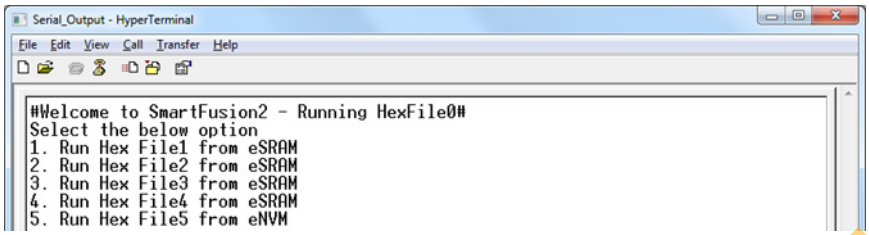

***Figure 52* • Running Hex File - User Options**

9. Select option 1 as shown in Figure 53. The application image runs from eSRAM. E1, F4 LEDs blink on the Board.
10. Reset the SW6 pin on SmartFusion2 Evaluation Kit, which brings the application to the Main Menu.

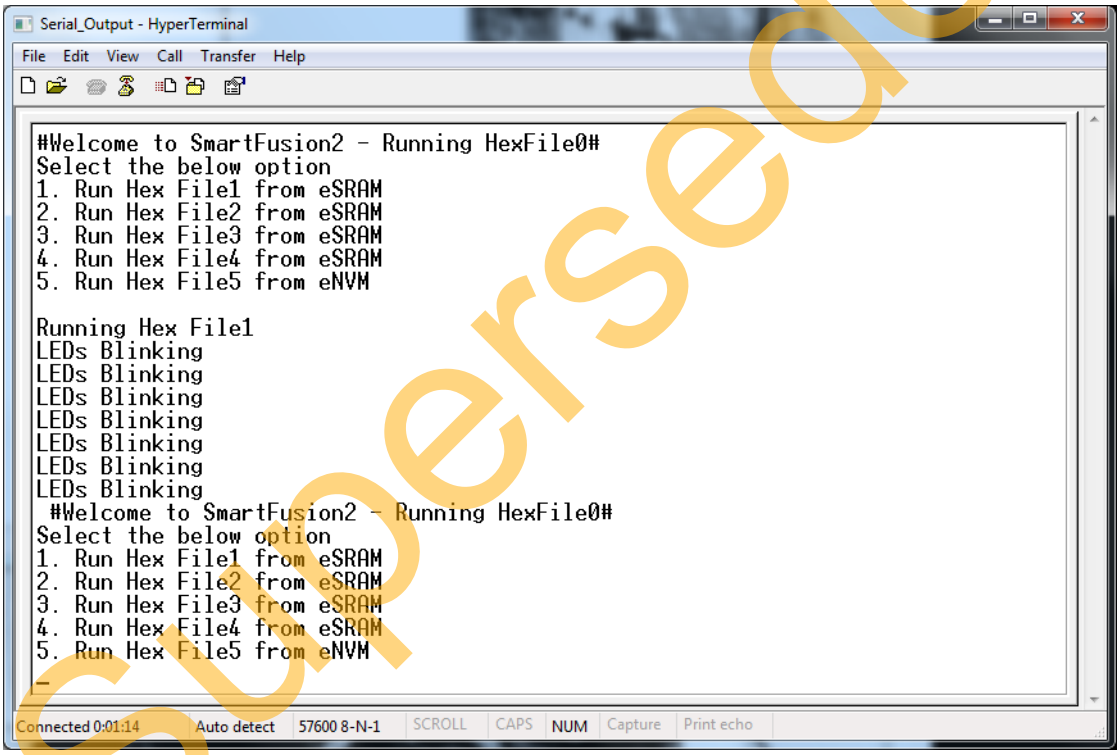

***Figure 53* • LEDs Blink Based on User Option**

11. Select option 2. The application image runs from eSRAM. F3, G7 LEDs blink on the Board.
12. Reset the SW6 pin on SmartFusion2 Evaluation Kit, which brings the application to the Main Menu.
13. Select option 3. The application image runs from eSRAM. H7, J6 LEDs blink on the Board.
14. Reset the SW6 pin on SmartFusion2 Evaluation Kit, which brings the application to the Main Menu.
15. Select option 4. The application image runs from eSRAM. H6, H5 LEDs blink on the Board.
16. Reset the SW6 pin on SmartFusion2 Evaluation Kit, which brings the application to the Main Menu.

17. Select option5. The application image runs from eNVM. E1, F4, F3, G7, H7, J6, H6, H5 LEDs blink on the Board.
18. Reset the SW6 pin on SmartFusion2 Evaluation Kit, which brings the application to the Main Menu.

Conclusion

This application note describes the step-by-step approach to build single/multiple SoftConsole application executable images in production release mode, store the executable images into eNVM using eNVM data storage client, and execute them on SmartFusion2 Evaluation Kit.

Appendix A - Design and Programming Files

You can download the design files from the Microsemi SoC Products Group website:
http://soc.microsemi.com/download/rsc/?f=sf2_release_mode_programming_11p4_DF

The design file consists of Libero Verilog, SoftConsole software project, programming files (*.stp) for SmartFusion2 SoC FPGA Evaluation Kit. Refer to the Readme.txt file included in the design file for the directory structure and description.

Appendix B - Creating LookUpTable.Hex File

The LookUpTable.Hex file contains the pre-calculated factorial data and this is a non-executable image. The following steps describe how to create the LookUpTable.Hex file:

1. Open the notepad and enter the factorial of number 2,3,4 as 2,6,24 as shown in Figure 1-1.

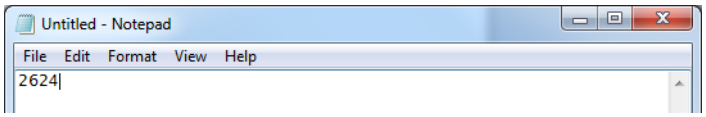

***Figure 1-1* • Entering Factorial Values in Notepad**

2. Save the notepad as **LookUpTable.bin** as shown in Figure 1-2.

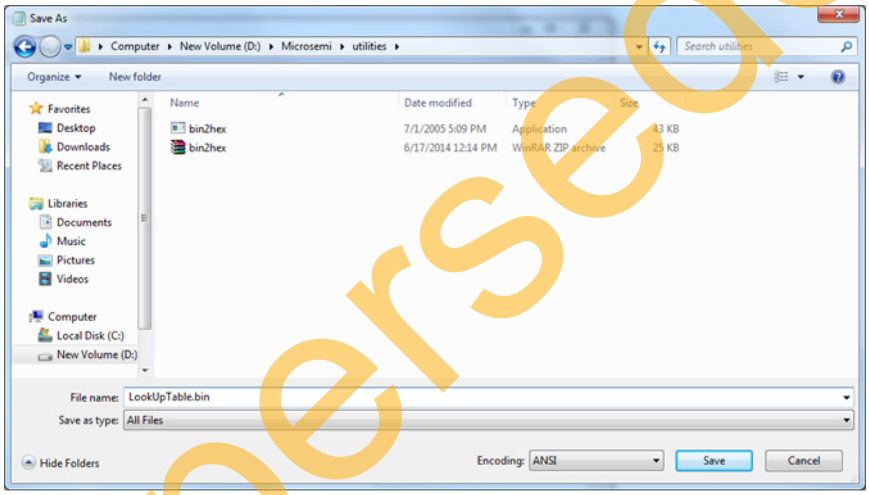

***Figure 1-2* • Saving Notepad**

3. The executable file, `bin2hex.exe` converts the `LookUpTable.bin`, which is in binary format to `LookUpTable.Hex` in Hex format. This executable file is available in the following path of the design files provided along with this application note:
http://soc.microsemi.com/download/rsc/?f=sf2_release_mode_programming_11p4_DF

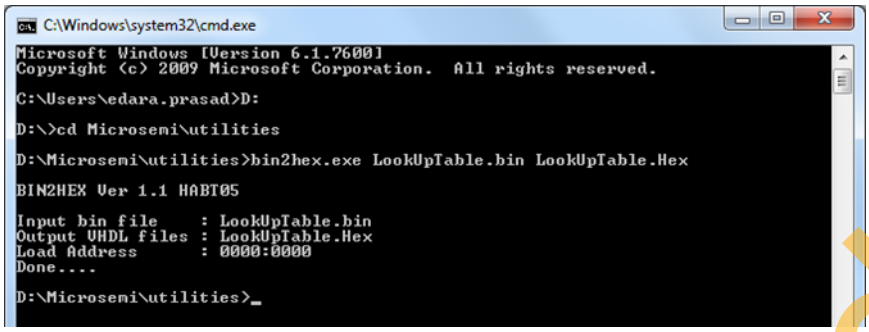

Figure 1-3 • **Converting Binary File to Hex File**

4. **LookUpTable.Hex** file is generated as shown in Figure 1-4. This file is copied to SoftConsole project folder of the eNVM Data client SoftConsole project.

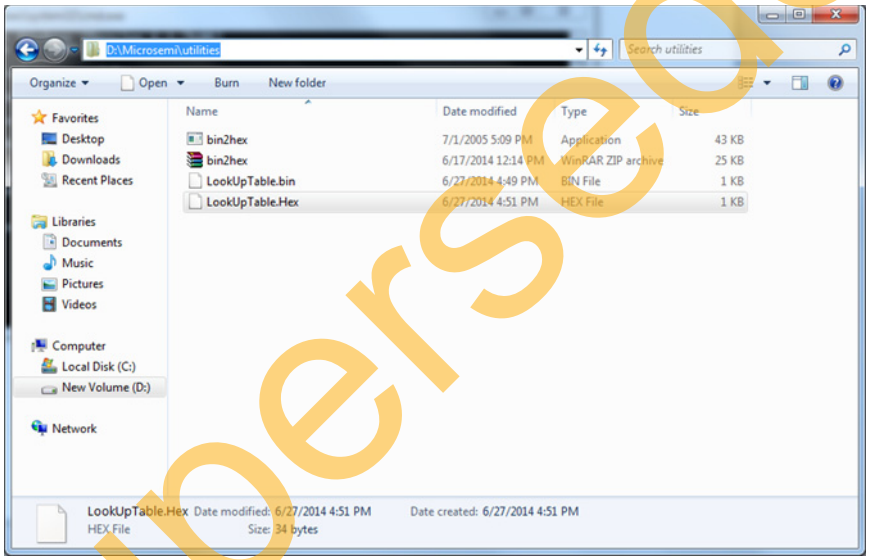

Figure 1-4 • **LookUpTable.Hex File**

A – List of Changes

The following table lists critical changes that were made in each revision of the chapter in the application note.

Date	Changes	Page
Revision 2 (December 2014)	Added the `bin2hex.exe` file to the design files and added a link to design files in the "Appendix B - Creating LookUpTable.Hex File" chapter (SAR 63426).	48
Revision 1 (August 2014)	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

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

Website

You can browse a variety of technical and non-technical information on the SoC home page, at 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 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

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; Power-over-Ethernet ICs and midspans; as well as custom design capabilities and services. Microsemi is headquartered in Aliso Viejo, Calif., and has approximately 3,400 employees globally. Learn more at **www.microsemi.com**.

51900293-2/12.14