ALMC UBC Development Environment Setup Guide

Note: this guide is based on Xilinx EDK/ISE 11.1. At the time of this writing, version 11.5 is the newest available. In 11.1, the EDK still has integrated functionalities for software development, whereas in 11.5 these functionalities are completely removed and users are supposed to use the SDK to develop software for the FPGA. The main reason why we use 11.1 instead of 11.5 is because we had experienced some difficulties installing and running 11.5.

Table of Contents

[Establish XPS Project 2](#_Toc259051590)

[Memory Initialization 13](#_Toc259051591)

[Setup Application Projects 14](#_Toc259051592)

[Add Source Files 16](#_Toc259051593)

[Generate Linker Scripts for the Application Projects 18](#_Toc259051594)

[Build Application Projects 21](#_Toc259051595)

[Program the Upper Board 22](#_Toc259051596)

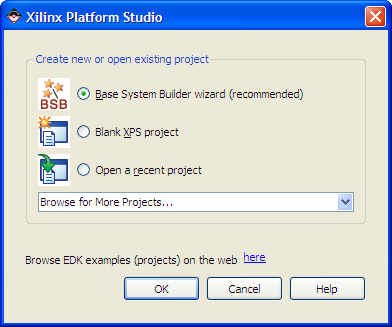
[Load ALMC\_UBC into the Flash 22](#_Toc259051597)

[Load ALMC\_Bootloader into the PROM 27](#_Toc259051598)

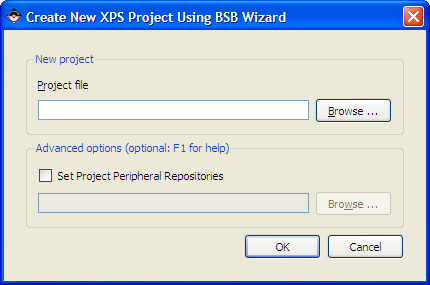
# Establish XPS Project

To build a Xilinx XPS (11.1) project for ALMC UBC

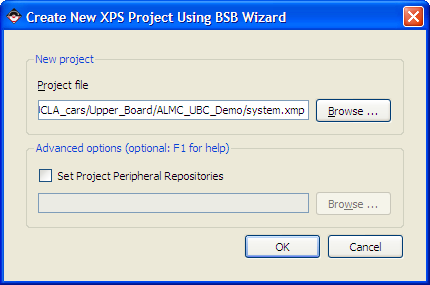
1. Place the folder “AnterosLabs” (supplied by Anteros Labs Inc) in the following directory: …\Xilinx\11.1\EDK\board\
   1. The “Xilinx\11.1” is just wherever the user happened to install the EDK 11.1 in. For example, if it was installed in the C:\ directly, then the path should read: C:\Xilinx\11.1\EDK\board\.
   2. Verify that in one of that subdirectory of “AnterosLabs”, there should be a “.xbd” file named: AnterosLabs\_EDKv11p1\_ALMC111\_Rev2\_v2\_2\_0.xbd. The resulted path for this file should be: …\Xilinx\11.1\EDK\board\AnterosLabs\boards\AnterosLabs\_EDKv11p1\_ALMC111\_Rev2\data\
2. Open Xilinx Platform Studio 11.1 (Make sure you’ve completed step 1 before carry out this step).



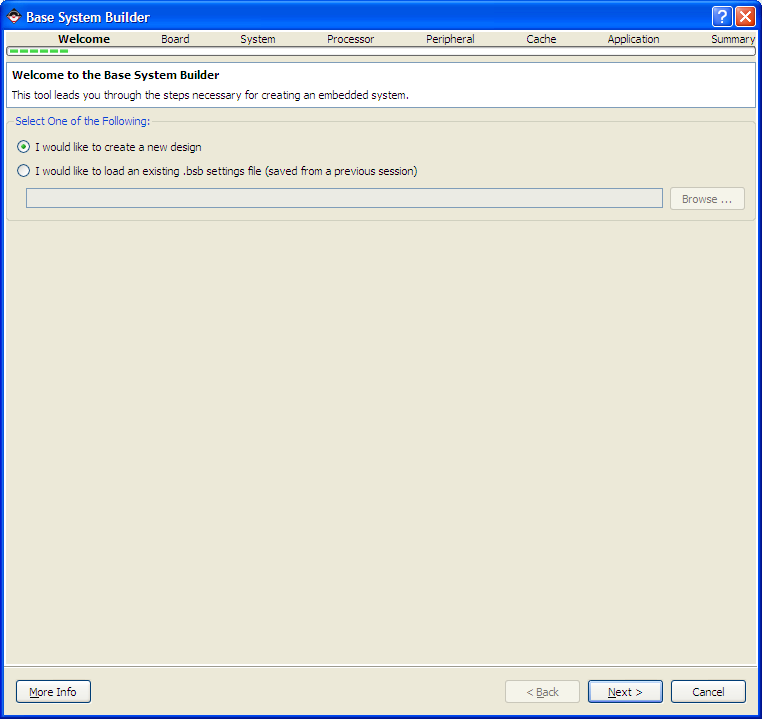
Select Base System Builder wizard, and click OK



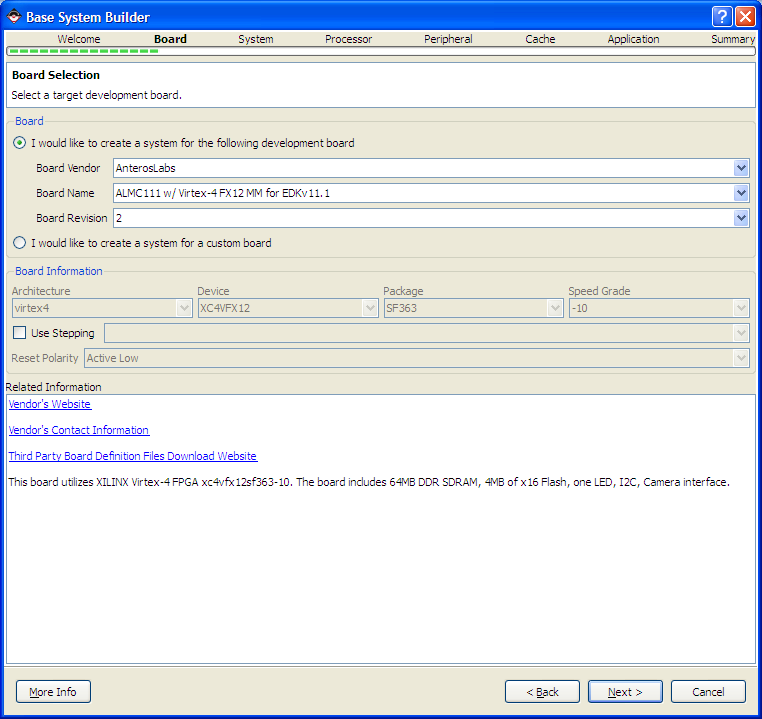
Click Browse, and then select an existing folder or create a new folder for the new XPS project.



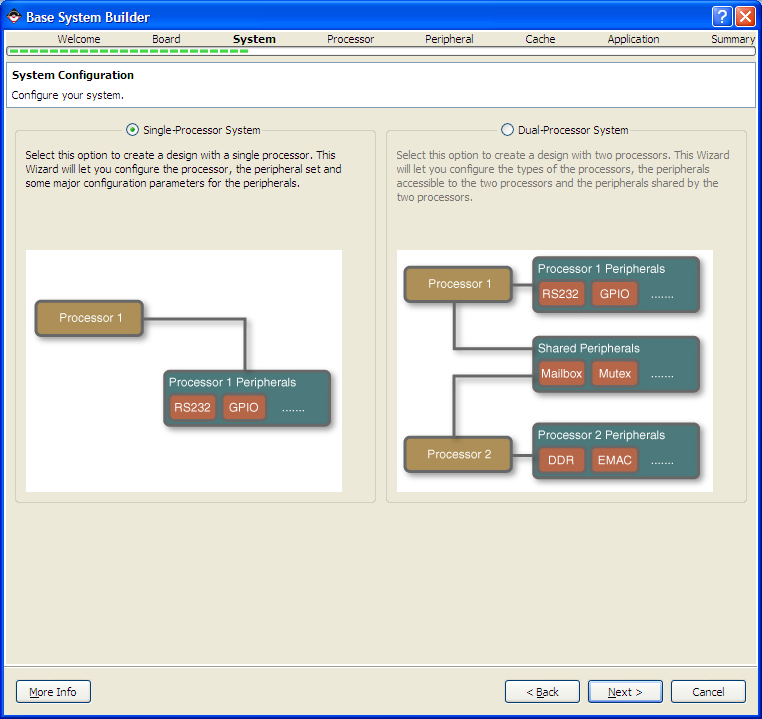
Click OK



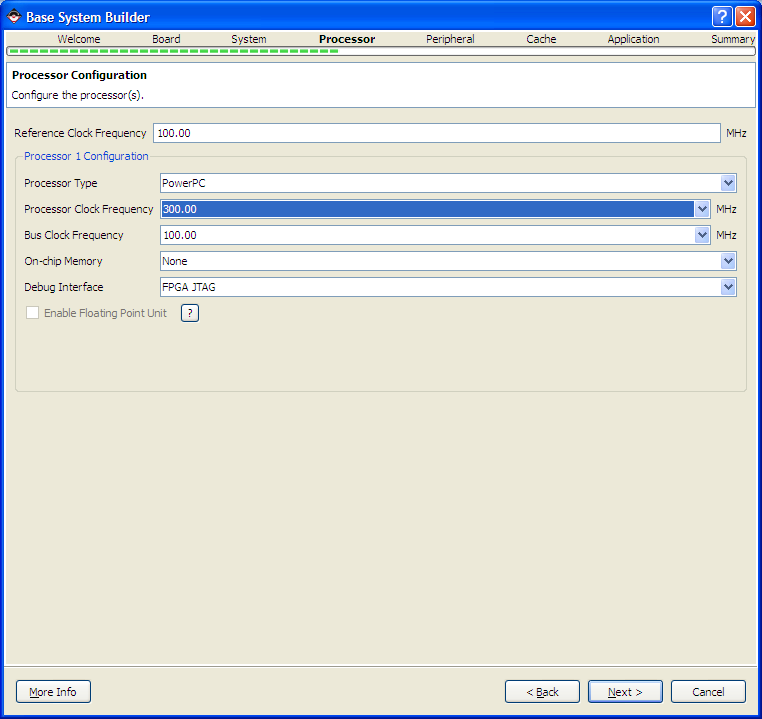
Click Next



First make sure that “AnterosLabs” is selected as Board Vendor, “ALMC111 w/ Virtex-4 FX11 MM for EDKv11.1” is selected as Board Name; then click Next.



Select Single-Processor System



In Processor Configuration, set the following:

Processor Type: PowerPC

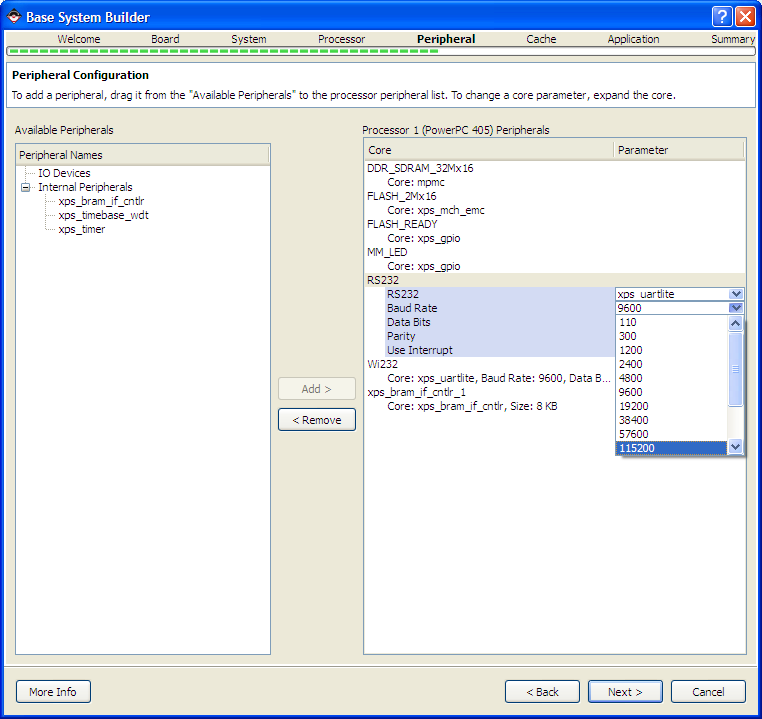
Processor Clock Frequency: 300.00MHz

Bus Clock Frequency: 100.00MHz

On-chip Memory: None

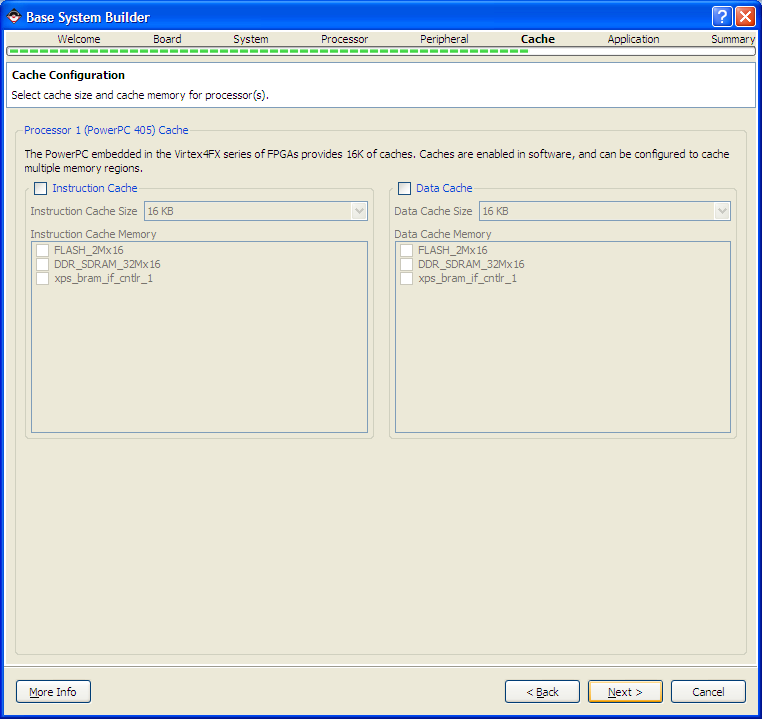
Debug Interface : FPGA JTAG

Click Next.

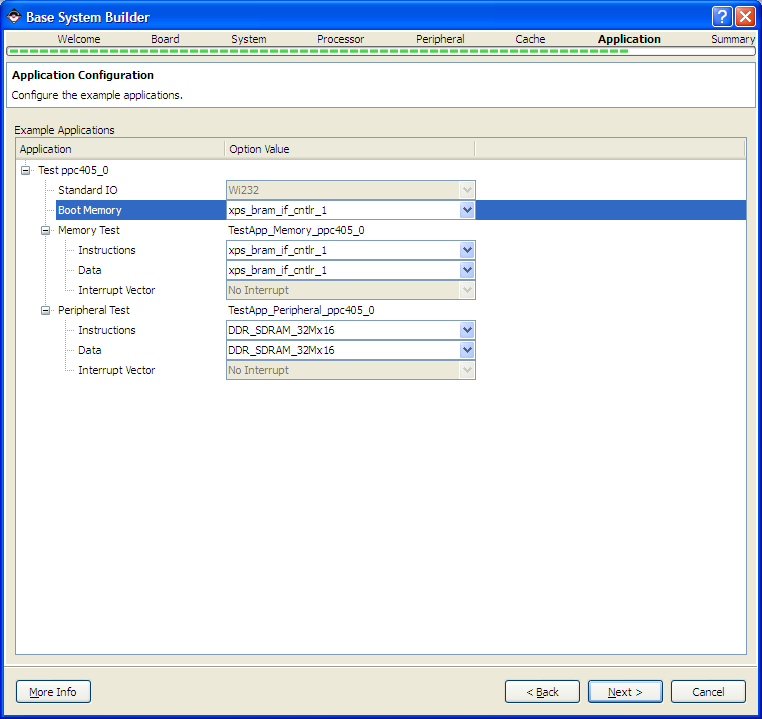


In Peripheral Configuration, make sure that the Baud Rate for both RS232 and Wi232 are set to be 115200. The size of xps\_bram\_if\_cntlr\_1 is set to be 16 KB.

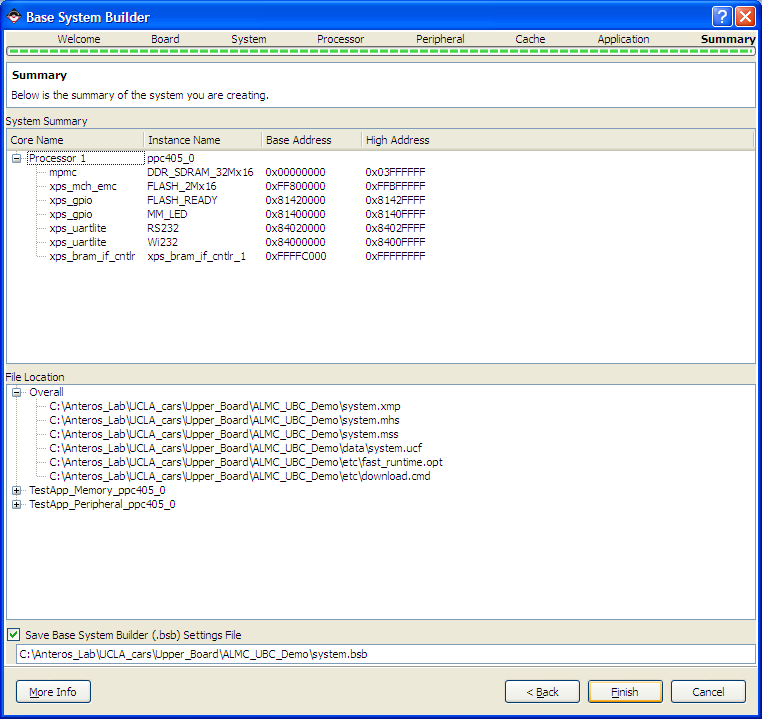
Keep all other settings as default.



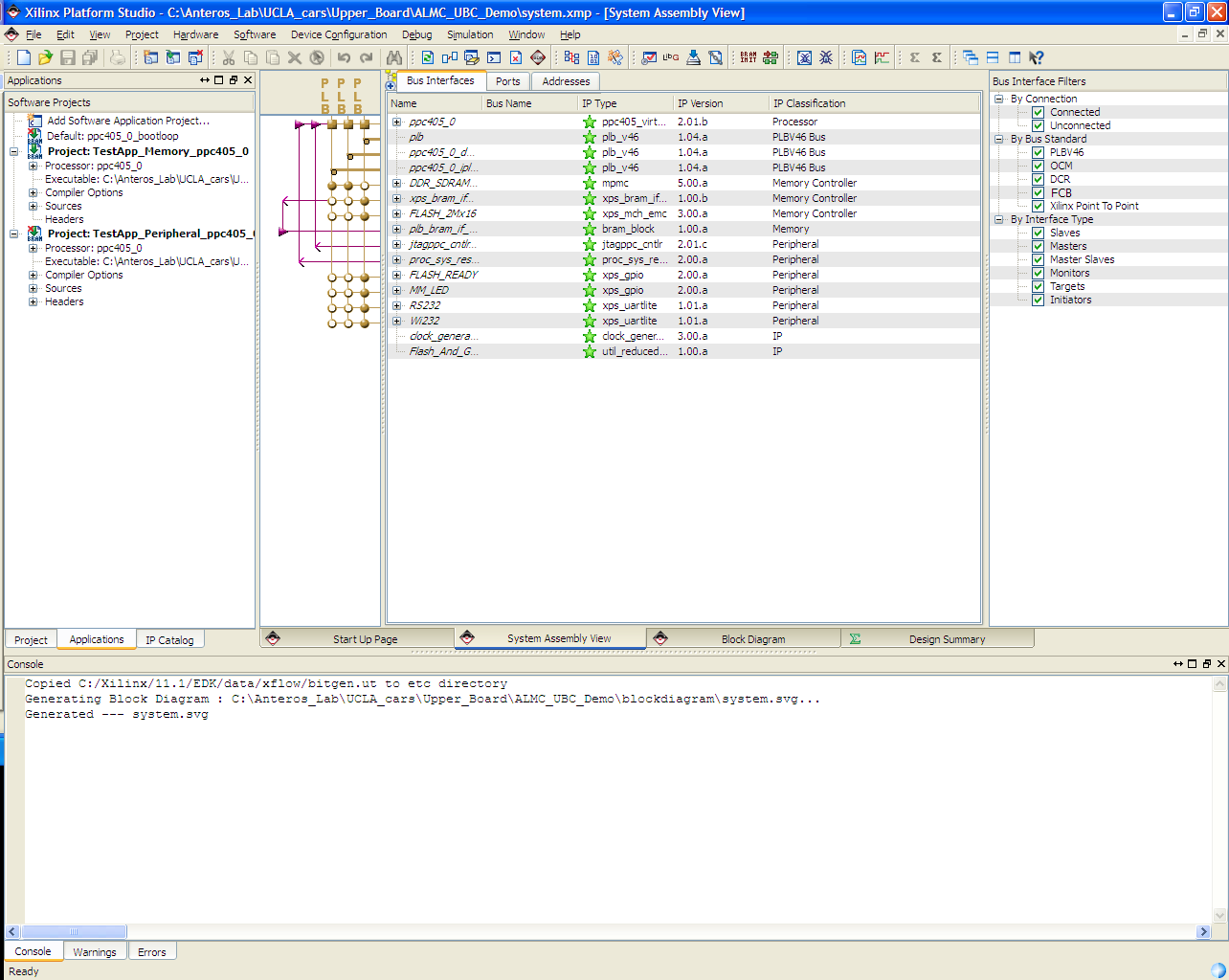
In Cache Configuration, do not select any Cache; just directly click Next.



In Application Configuration, just make sure that the Boot Memory is set to “xps\_bram\_if\_cntlr\_1” and then click Next.



Click Finish.



An initial XPS project is established.

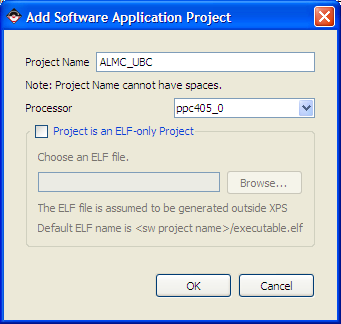
# Memory Initialization

When a XPS project is first established, we recommend perform the XST synthesis/memory initialization by selecting “Update Bitstream” under the Device Configuration menu (or the button with “BRAM INIT” icon). This is a rather time consuming step necessary for programming the FPGA, it’s better to get it out of the way first. Note that the synthesis process will induce multiple warnings, however it should not cause any errors.

# Setup Application Projects

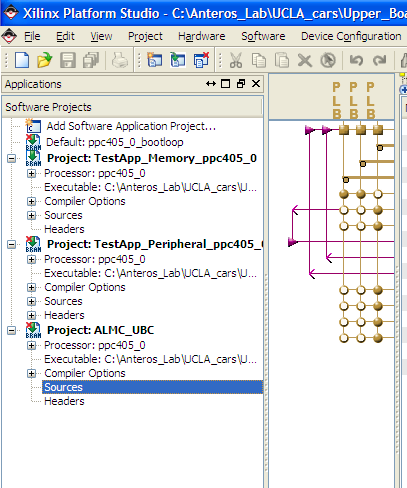
Once the synthesis/memory initialization is done, add two application projects to the XPS project.

Create an application project named ALMC\_UBC by select “Add Software Application Project…” under the Software drop-down menu.



Type in the project name “ALMC\_UBC”; then click OK.

Once the application project is added, the Software Projects panel (Application tab) should show a new branch that reflects the new project.



Create a second application project called “ALMC\_Bootloader” in a similar fashion.

# Add Source Files

Copy folders “ALMC\_UBC” and “ALMC\_Bootloader” (provided by Anteros Labs) into the project direcory (where “system.xmp” resides). Note these two folders should contain the following source/header files, respectively:

ALMC\_UBC: “ALMC\_Command.c”,

“ALMC\_Command.h”,

“ALMC\_UBC.c”,

ALMC\_Bootloader: “blconfig.h”,

“bootloader.c”,

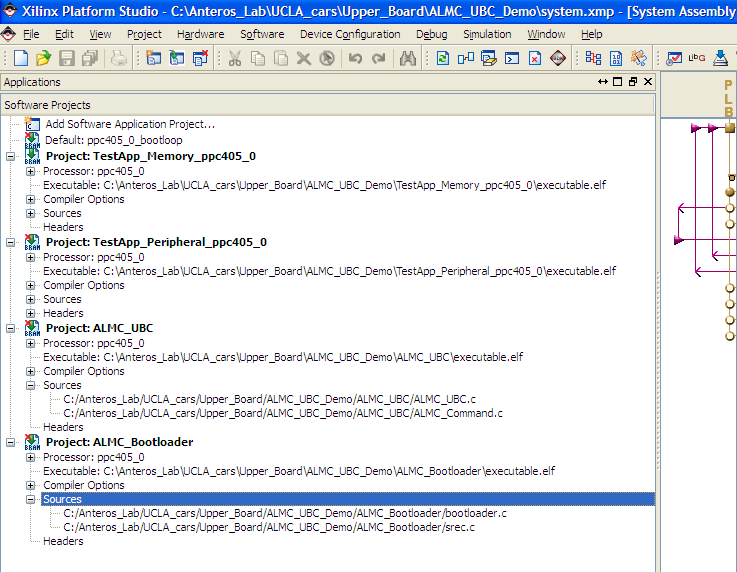
“errors.h”,

“portab.h”,

“srec.c”,

“srec.h”

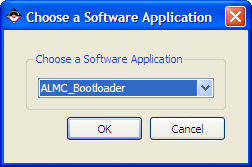
Add the above source files (.c files) to their respective application projects by right clicking the “Sources” branches and select “Add Existing Files…” (You don’t have to add the header files, although adding shouldn’t hurt).



# Generate Linker Scripts for the Application Projects

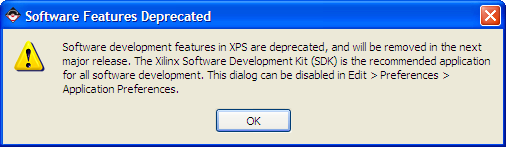
1. Generate Linker Scripts for ALMC\_Bootloader:

Select “Generate Linker Script… “ under the “Software” drop-down menu



Choose “ALMC\_Bootloader”, click OK

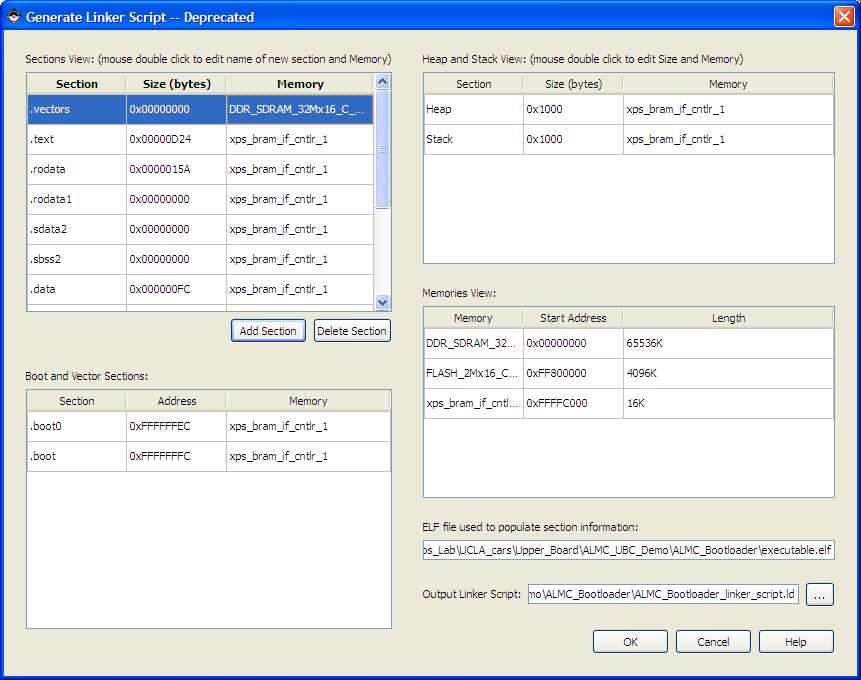
Click OK if you see the following window:



In the “Generate Linker Script” Window make the following changes:

1. Select “xps\_bram\_if\_cntlr\_1” for all the Memory. If there is a “.vector” section, set its Memory to be DDR\_SDRAM.
2. Set the sizes for both “Heap” and “Stack” to be 0x1000 (bytes). Don’t worry about any other size values, as they will be automatically generated.

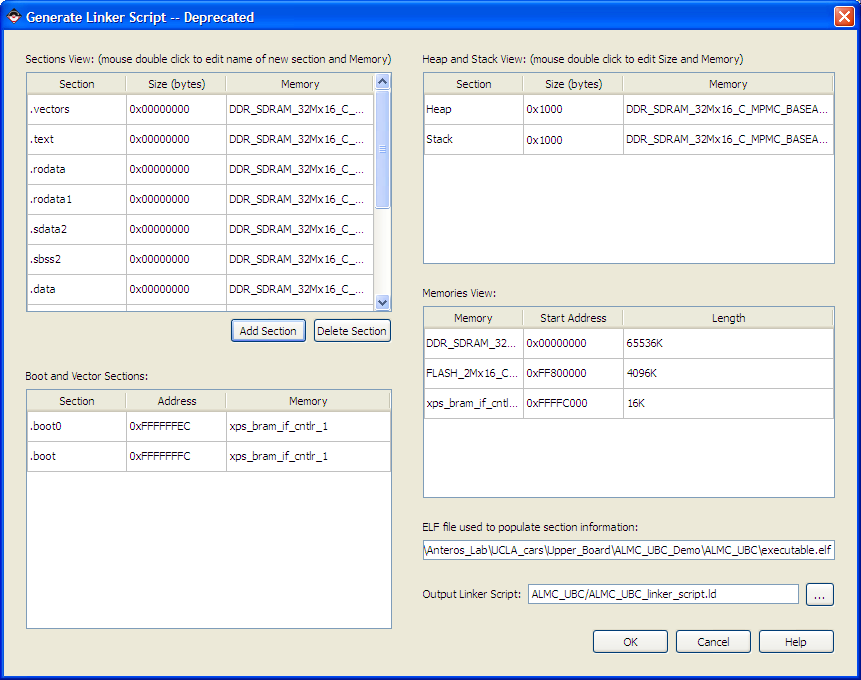
The resulted GLS window should look like the following:



Click OK.

1. Follow similar steps to generate linker script for ALMC\_UBC. In the “Generate Linker Script” Window, make the following setting/changes:
2. Set all the Memory in the “Sections View:” box and the “Heap and Stack View:” box to be DDR\_SDRAM. The Memory in the “Boot and Vector Sections:” should be left at “xps\_bram\_if\_cntlr\_1”.
3. Set the sizes for both “Heap” and “Stack” to be 0x1000 (bytes). Don’t worry about any other size values, as they will be automatically generated.

The resulted GLS window should look like the following:



# Build Application Projects

To build the ALMC\_UBC project, right click on the main branch of the project in the “Software Projects” panel: 

Select “Build Project”.

If the build is successful, you should see the following messages in the Console panel:

powerpc-eabi-size ALMC\_UBC/executable.elf

text data bss dec hex filename

23622 1356 8736 33714 83b2 ALMC\_UBC/executable.elf

Done!

Build the ALMC\_Bootloader project as well by following the same steps.

Once both projects are successfully built, we’re ready to program the upper board.

# Program the Upper Board

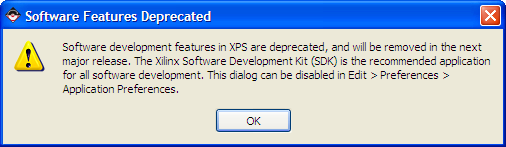
There are two programs we need to load into the ALMC upper board:

1. Load ALMC\_UBC into the Flash
2. Load ALMC\_Bootloader into the PROM

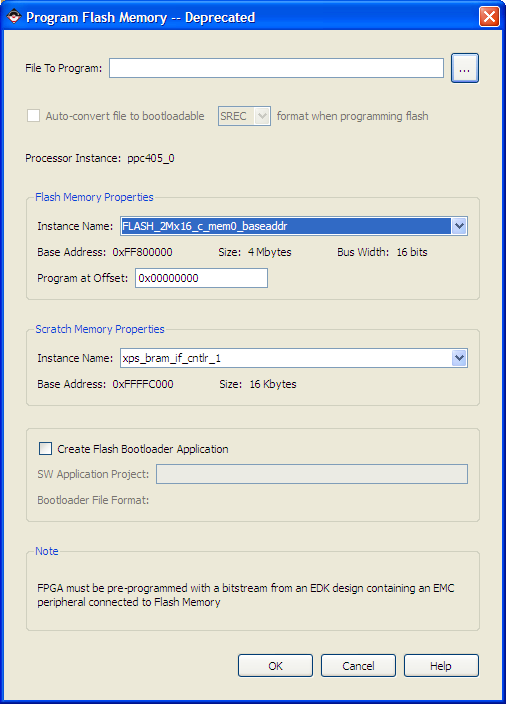
## Load ALMC\_UBC into the Flash

To load ALMC\_UBC into the Flash memory, follow the steps below:

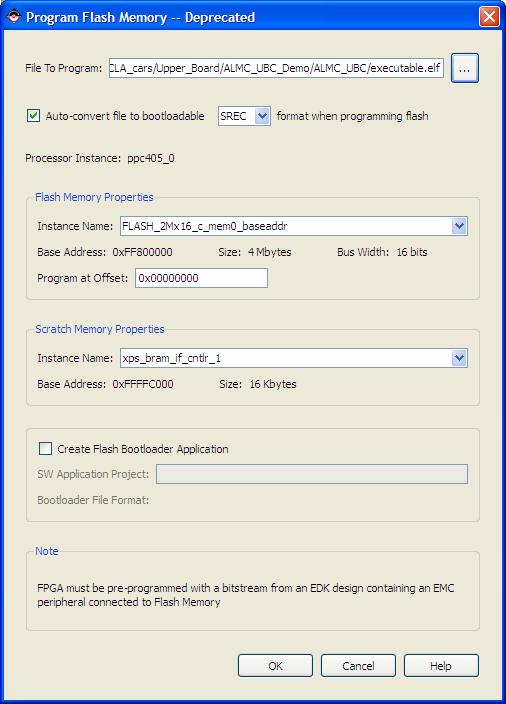
1. Mark the ALMC\_Bootloader project to Initialize BRAMs. This is done by right click on “Project: ALMC\_Bootloader” ; then select “Mark to Initialize BRAMs”. Once this is done, the red cross on the BRAM icon should disappear.
2. Select “Update Bitstream” (under the “Device Configuration” drop-down menu). If this is successful, it should say “Memory Initialization completed successfully.” in the Console panel.
3. Connect the **P8** port on the upper board to the PC via a Xilinx Platform Cable USB.
4. Power the car on. (The Status light on the Platform Cable should change from orange to green).
5. Connect a wireless module to the PC and open a Hyper Terminal for monitoring the upper board.
6. Go back to the XPS and select “Download Bitstream” under the “Device Configuration” drop-down menu. If this is successful, you should see ALMC\_Bootloader messages showing on the Hyper Terminal.
7. Select “Program Flash Memory” under the “Device Configuration” drop-down menu. Click OK if you see the following window:



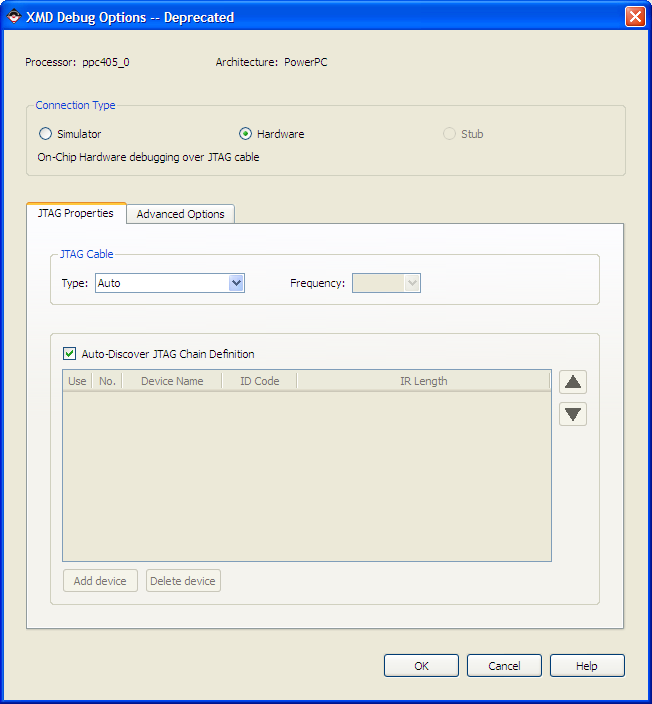
Then you should see the following window:



1. Select the “File To Program”: browse to and select the “executable.elf” file in the ALMC\_UBC directory.
2. Check the box to indicate “Auto-convert file to bootable”, and make sure the rest of the settings are as displayed below (Scratch Memory should be set to xps\_bram\_if\_cntlr\_1). Then click OK.



1. The first time you “Program Flash Memory”, you will be prompted to set XMD Debug Options (see the screen cap below). Just make sure the settings are as shown in the screen cap and then click OK.

****

1. Once the programming is done, you should see the following message on the Console Panel: “Flashwriter completed successfully!

Done!”

1. To verify that the ALMC\_UBC program is successfully loaded into the Flash Memory, just select “Download Bitstream” again.

If loading is indeed successful, you should be able to see ALMC\_UBC demo program running after the bootloader messages (in the Hyper Terminal), which will look like the following:

-------# 0000 --------

Select a command:

a) STOP

b) MOVE\_XY c) MOVE\_SV

f) WRIT\_LED

i) CMEP j) RSEP

k) DRDS l) DREN

m) STDS n) STEN

o) READ\_ID

p) READ\_XY q) READ\_SV

v) READ\_SF w) UPD\_SF

x) READ\_PAR y) UPD\_PAR

z) Stream Test

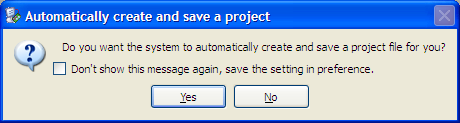
0) Exit

Note: before ALMC\_Bootloader is loaded into the PROM, the only way to restart the upper board program (in the Flash Memory) is to download the bootloader bitstream again.

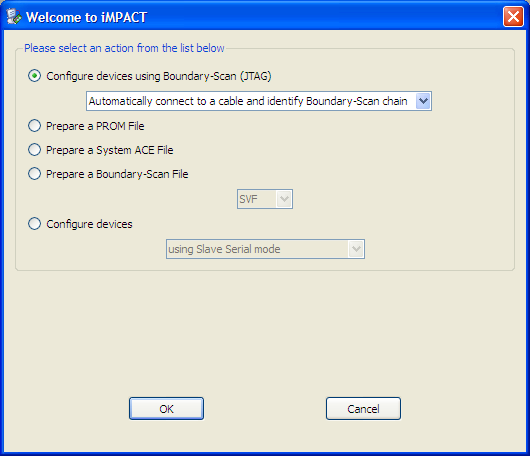
## Load ALMC\_Bootloader into the PROM

To avoid downloading the bootloader bitstream each time we want to restart the upper board program, we have to load a bit file of the bootloader into the PROM. This can be done by following the steps below:

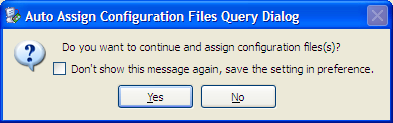
1. Notice that a file named “download.bit” is generated/regenerated in the “\implementation” directory when you update the bitstream. Make sure the “download.bit” is generated when the “ALMC\_Bootloader” project is selected as “Mark to Initialize BRAMs”; this will make certain that “download.bit” contains ALMC\_Bootloader.
2. Close XPS. Open iMPACT 11.1 (Start Menu 🡪 All Programs 🡪 Xilinx ISE Design Suite 11 🡪 ISE 🡪 Accessories 🡪 iMPACT).
3. You might be prompted with the following window:



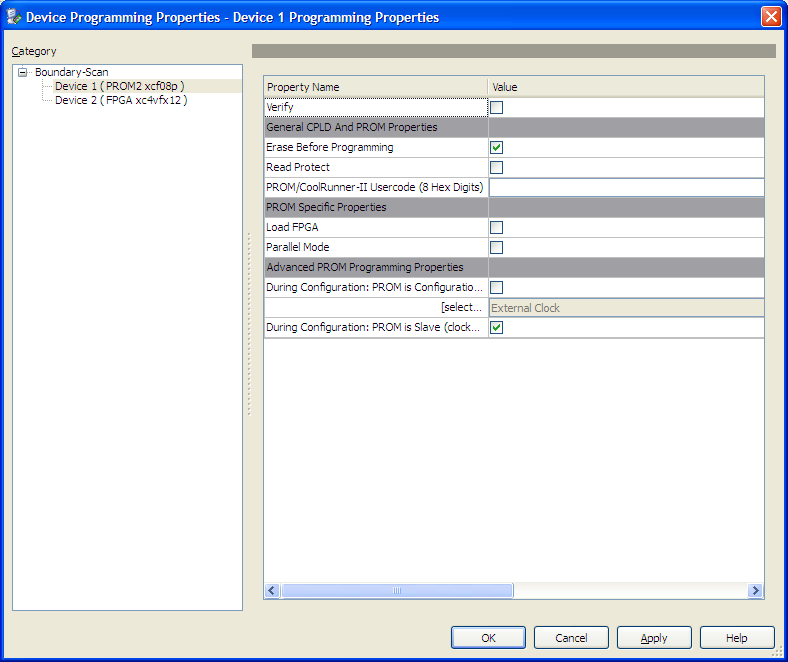
1. Make sure the Xilinx Platform Cable is still connected. Power on the car if it’s not. Again the Status LED on the Platform Cable should be green before carrying on to the next step.
2. Go back to the “Automatically create and save a project” window and click Yes.



Select “Configure devices using Boundary-Scan (JTAG)” and click OK.

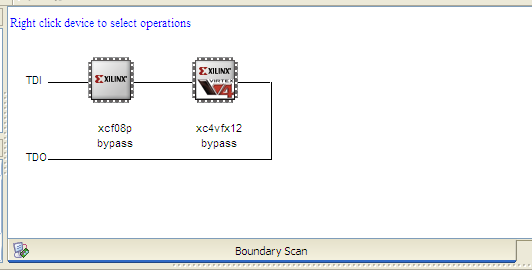


Click No.



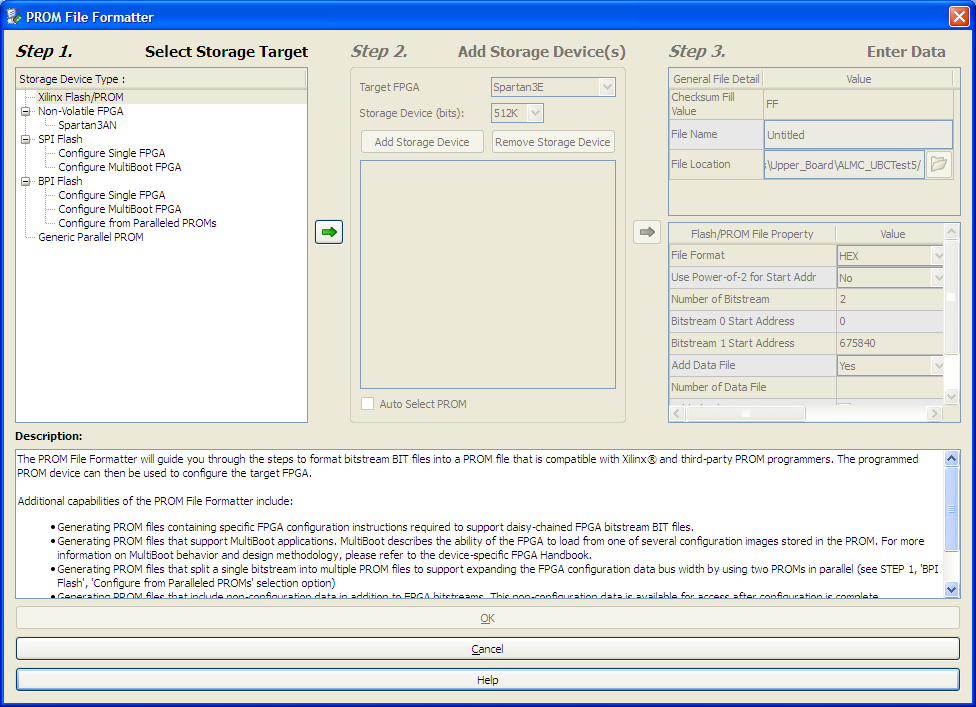
Click OK.

1. In the “Boundary Scan” panel, you should be able to see two devices (“xcf08p” and “xc4vfx12”) in a chain between “TDI” and “TDO”.

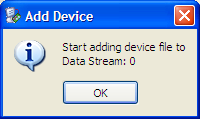


Among them, “xcf08p” is the PROM, “xc4vfx12” is the FPGA itself.

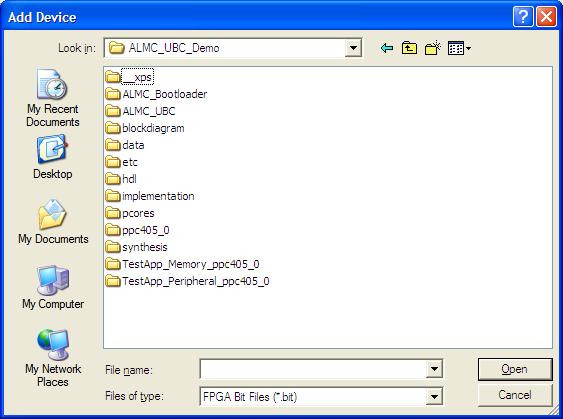
1. Right click the “xcf08p” icon. Select “Get Device ID”. If it says “ReadIdcode Succeeded”, then the PROM is functioning properly. We can now go on the program the PROM.
2. Generate .mcs file:
   1. In the “iMPACT Flows” panel, double-click the “PROM File Formatter” branch.



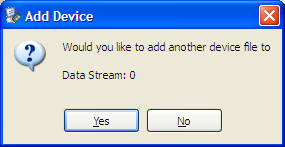
* 1. In Step 1., select “Xilinx Flash/PROM” and then click the green arrow.
  2. Select “Platform Flash” and “xcf08 [8M]”. Click “Add Storage Device”.
  3. Select the “xcf08p [8M]” text just shown below and click the second green arrow.
  4. Adjust the settings in Step 3. as follows:
     1. Checksum Fill Value: FF
     2. File Name: ALMC\_Bootloader
     3. File Location: the main directory of your XPS project.
     4. File Format: MCS
     5. Enable Revisioning: No
     6. Enable Compression: No
  5. Click OK.



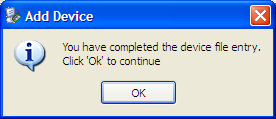
Click OK.



Browse into the “\implementation” directory, and open “download.bit”.

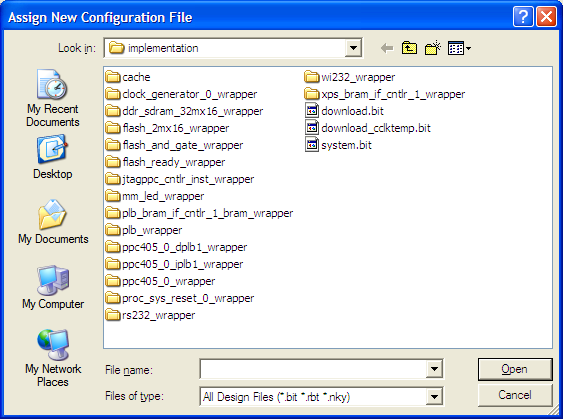


Click No.



Click OK.

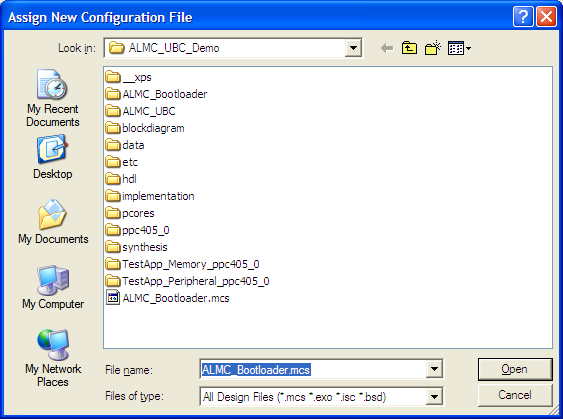
* 1. Double-click the “Generate File…” arrow in the “iMPACT Processes” panel.
  2. It should say “Generate Succeeded” in the “PROM File Formatter” panel. Go to the “\implementation” directory, you will see that a file named “download\_cclktemp.bit” was just generated.
  3. Go back to the PROM File Formatter panel, right click on the “xc4vfx12” icon and select “Assign New Configuration File…”.



This time, open “download\_cclktemp.bit” instead.

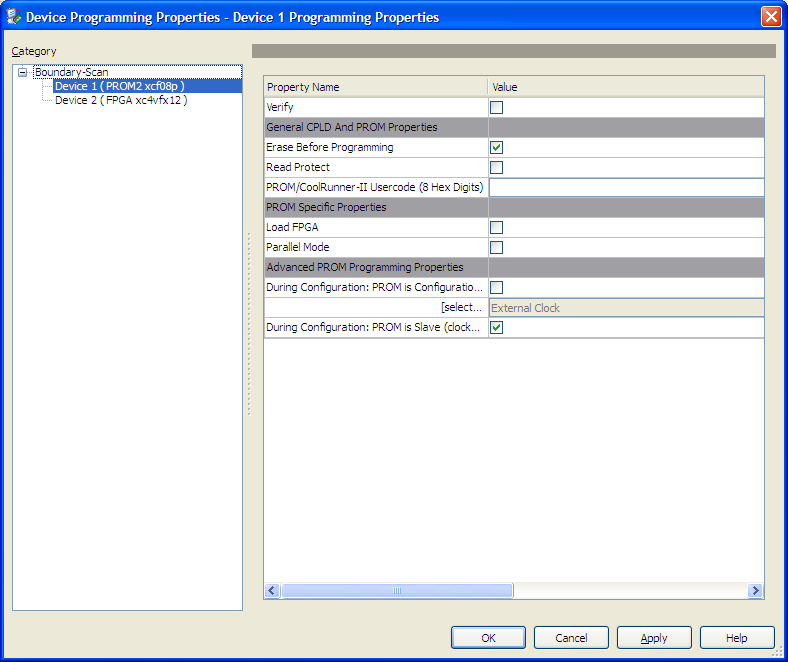
* 1. Double-click the “Generate File…” again. Once this is successful, notice that a new “ALMC\_Bootloader.mcs” file was just generated in the main directory of our XPS project. This is the file we want to load into the PROM.
  2. Note that as early as step h, a .mcs file of the same name has already been generated. However that was not the correct file to load. Step i and j must be completed before the correct .mcs file can be generated.

1. Go back the “Boundary Scan” panel. Right click on the “xcf08p” icon and select “Assign New Configuration File…”

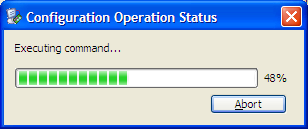


Open “ALMC\_Bootloader.mcs”.

1. Right click on the “xcf08p” icon again and select “Program”.



Check property “Verify”, “Erase Before Programming”, and “During Configuration: PROM is Slave…”, and then click OK.



If it says “Program Succeeded”, the ALMC\_Bootloader is successfully loaded into the PROM!

1. Power cycle the car. You should be able to see the upper board rebooting without using the “Download Bitstream” function in the XPS. Note that this process does NOT need to be repeated when a new program is loaded into the Flash Memory.