

Avnet FXT Evaluation Board Multi-Boot Design

**Version 1.0
September 2008**

1 Introduction

This document describes a simple PowerPC based design to demonstrate the multi-boot capability of the Virtex-5 FXT FPGA family.

2 Reference Design Requirements

This reference design will require the following software and hardware setups.

2.1 Software

The software requirements for this reference design are:

- WindowsXP
- Xilinx ISE 10.1 with Service Pack 2
- Xilinx EDK 10.1 with Service Pack 2

2.2 Hardware

The hardware setup for this reference design is:

- Computer with 1 GB RAM and 1 GB virtual memory (recommended)
- Avnet Virtex-5 FXT evaluation board
- Straight through RS232 cable
- Power supply
- JTAG programming cable (USB or PC4)

3 Multi-Boot Use Case

The Virtex-5 FPGA families have a feature called Multi-Boot that allows the FPGA to selectively load its bitstream from an attached configuration PROM that contains 2 or more bitstreams.

The FPGA application will trigger a Multi-Boot operation, causing the FPGA to reconfigure from a different bitstream. Once a Multi-Boot operation is triggered, the FPGA restarts its configuration process as usual, the FPGA clears its configuration memory and reconfigures from the configuration PROM with the new bitstream.

The following figure shows an example of the Multi-Boot use case. The initial bitstream is loaded into the FPGA to perform board-level diagnostics and verify the operation of the hardware. Once the board-level test is successfully completed, the **Re-Configure** signal is used to configure the FPGA with the second bitstream that will run an application on the board. Alternatively, a third or fourth bitstream can be added to the configuration PROM to run different applications on the board.

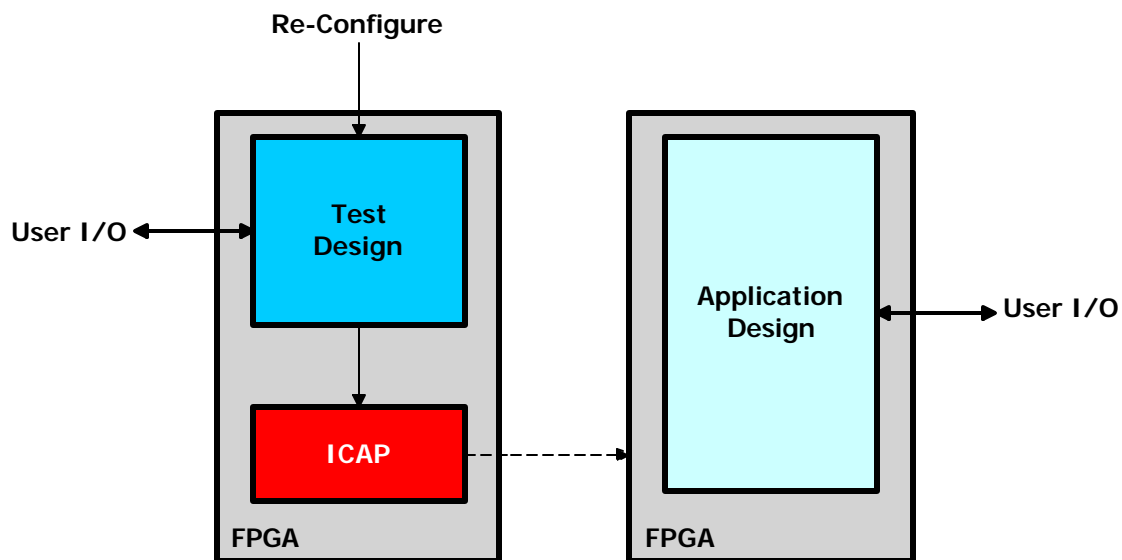


Figure 1 - Multi-Boot Use Case

All Virtex-5 devices are equipped with an Internal Configuration Access Port (ICAP) primitive. ICAP is used to send IPROG command to trigger reconfiguration. The IPROG command has similar effect as a pulsing PROGRAM_B pin; however, the dedicated reconfiguration logic is not reset.

The start address of the configuration is stored in the WBSTAR register, which is accessible via ICAP. The WBSTAR register must be programmed before sending the IPROG command. Please refer to the Virtex-5 Configuration User's Guide (ug191) for more information on using ICAP.

4 Configuration Flash Memory Map

The following figure shows the configuration Flash memory map for this reference design. As shown, the bitstream to run the diagnostics on the board is stored in the Flash, starting at address 0x00000000, while the bitstream for the application is stored in the Flash, starting at address 0x00200000 (2MB address space is dedicated to each bitstream). It should be noted the Avnet V5FXT evaluation board uses the V5FX30T device that requires 13,517,056 bits of configuration memory.

As shown in the following figure, the remaining Flash memory space can be used to store user data and/or application code.

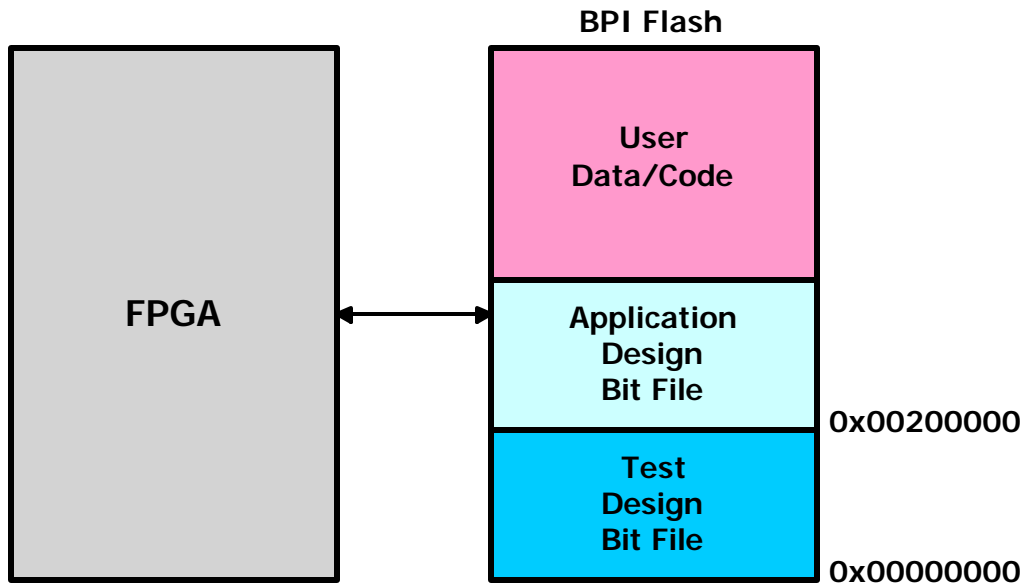


Figure 2 - Configuration Flash Memory Map

5 System Self-Test Design Block Diagram

The following figure shows a high-level block diagram of the system self-test (diagnostics) for this reference design. The design consists of:

- PowerPC processor
- 64KB of BRAM
- RS232 Port
- LEDs
- DIP Switches
- Timer
- Interrupt Controller
- Custom ICAP Controller IP

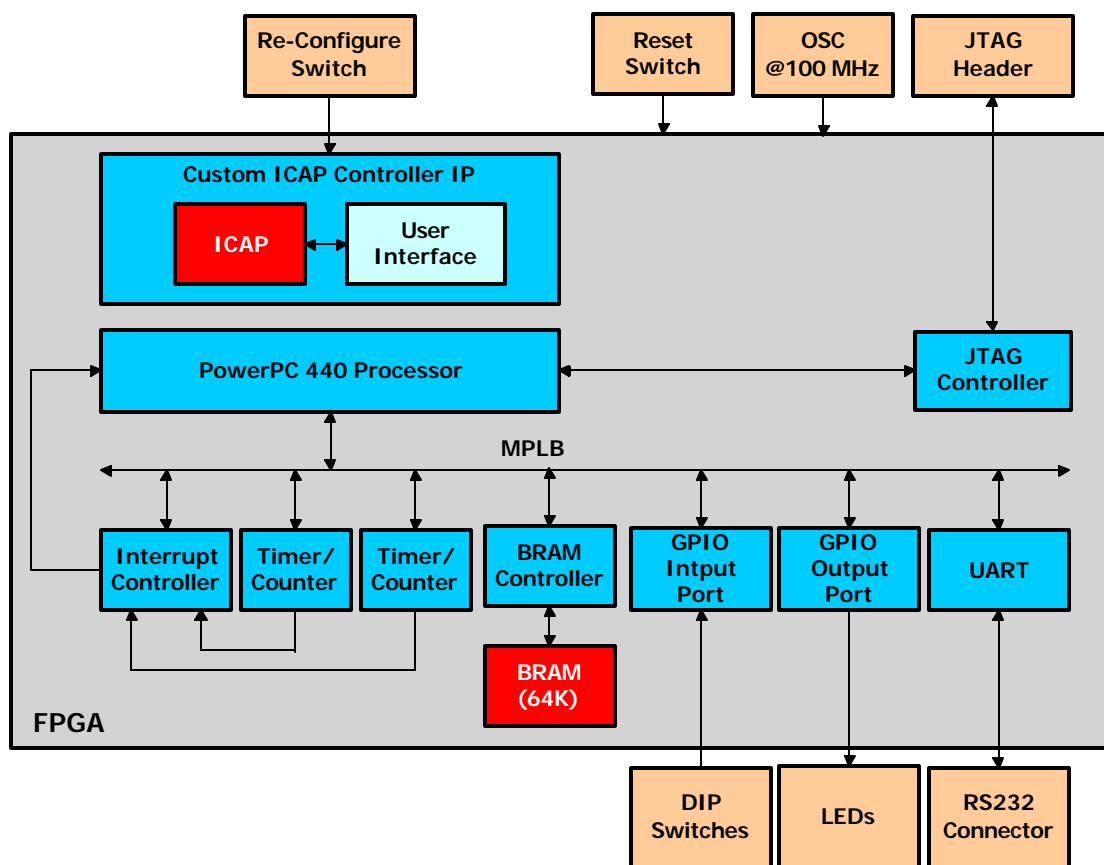
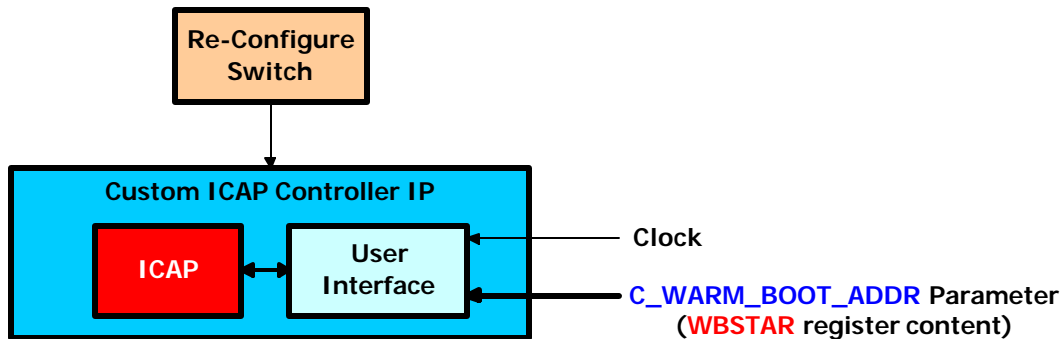


Figure 3 – System Self-Test Design Block Diagram

5.1 Custom ICAP Controller IP

The following figure shows the custom ICAP controller IP connections in the Self-Test design. As shown, the **Re-Configure** input to the ICAP IP is connected to a Push Button switch on the board. The WBSTAR content is supplied to the ICAP controller IP via a 32-bit parameter, so that the reconfiguration start address can be assigned by the user.



The following figure shows the custom ICAP controller IP instantiation in the Self-Test design MHS file. As shown, the **Re-Configure** input to the ICAP IP is connected to a Push Button switch on the board, while the **WBSTAR** register content is set to 0x00200000 via the **C_WARM_BOOT_ADDR** parameter. So, for this application, when the on-board push button switch is pressed, the 0x00200000 value is loaded into the WBSTAR register and then the IPROG command is issued by ICAP to reconfigure the FPGA.

```
18  PORT fpga_0_RS232_RX_pin = fpga_0_RS232_RX, DIR = I
19  PORT fpga_0_RS232_TX_pin = fpga_0_RS232_TX, DIR = O
20  PORT fpga_0_LEDs_8Bit_GPIO_d_out_pin = fpga_0_LEDs_8Bit_GPIO_d_out, DIR =
21  PORT fpga_0_DIP_Switches_8Bit_GPIO_in_pin = fpga_0_DIP_Switches_8Bit_GPIO
22  PORT sys_clk_pin = dcm_clk_s, DIR = I, SIGIS = CLK, CLK_FREQ = 100000000
23  PORT sys_rst_pin = sys_rst_s, DIR = I, RST_POLARITY = 1, SIGIS = RST
24  PORT multi_boot_0_re_configure_pin = multi_boot_0_re_configure, DIR = I
25
```

```
185 BEGIN multi_boot
186     PARAMETER INSTANCE = multi_boot_0
187     PARAMETER HW_VER = 1.00.a
188     PARAMETER C_WARM_BOOT_ADDR = 0x00200000
189     PORT clock_in = sys_clk_s
190     PORT re_configure = multi_boot_0_re_configure
191 END
```

6 Application Design Block Diagram

The following figure shows a high-level block diagram of the application design. The design consists of:

- PowerPC processor
- 64KB of BRAM
- RS232 Port
- LEDs
- DIP Switches
- Timer
- Interrupt Controller

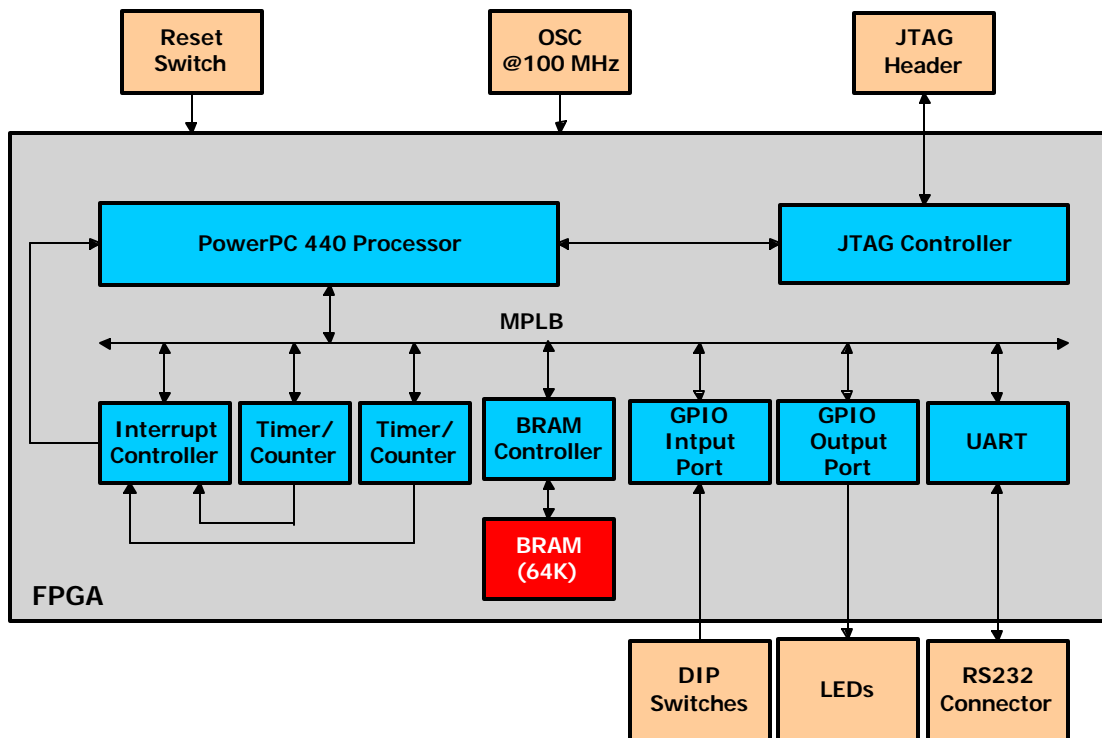
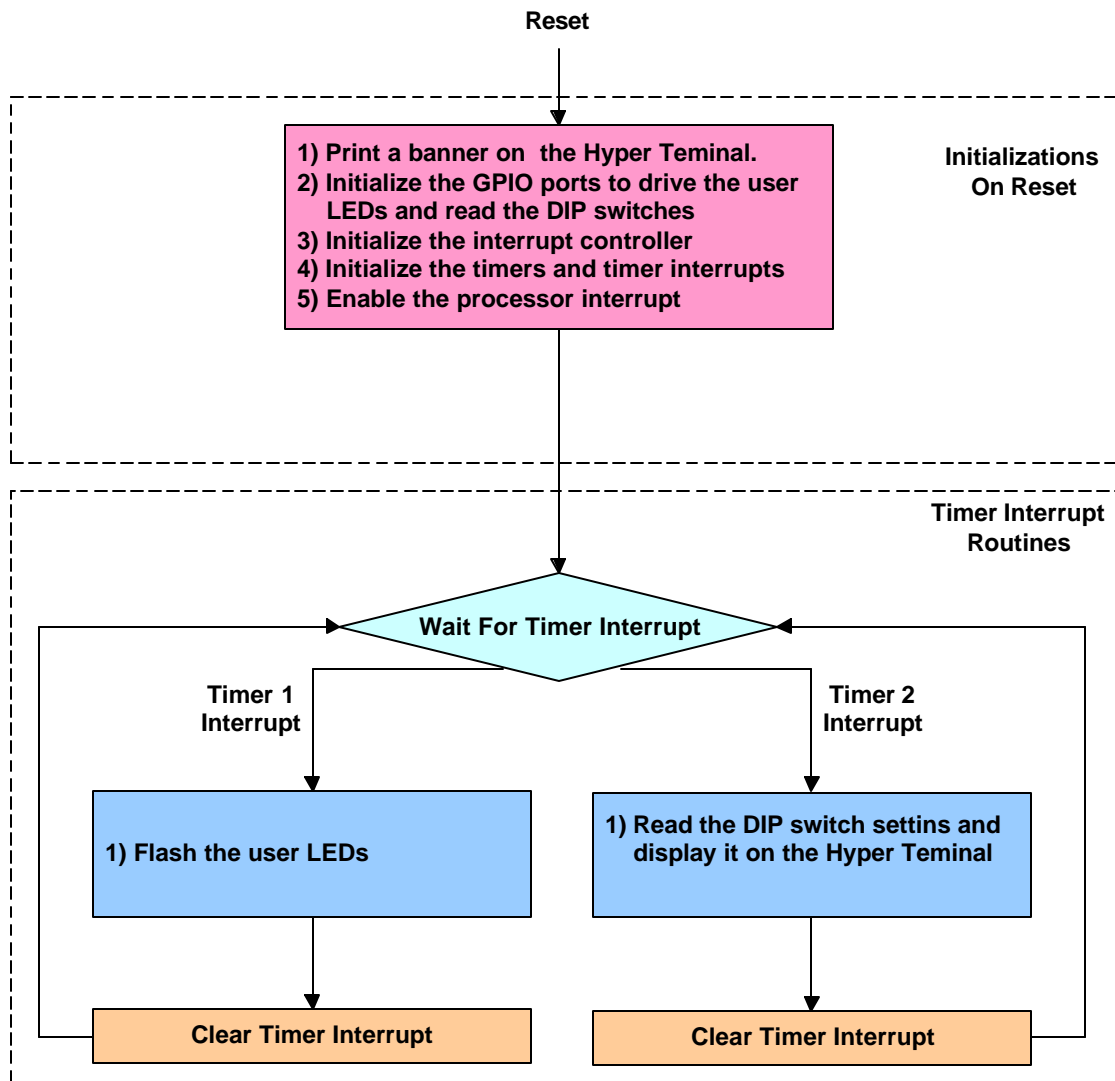


Figure 4 - Application Design Block Diagram

6.1 Application Design Software

A simple software application is used for this reference design. The following figure shows a flow chart for this program.



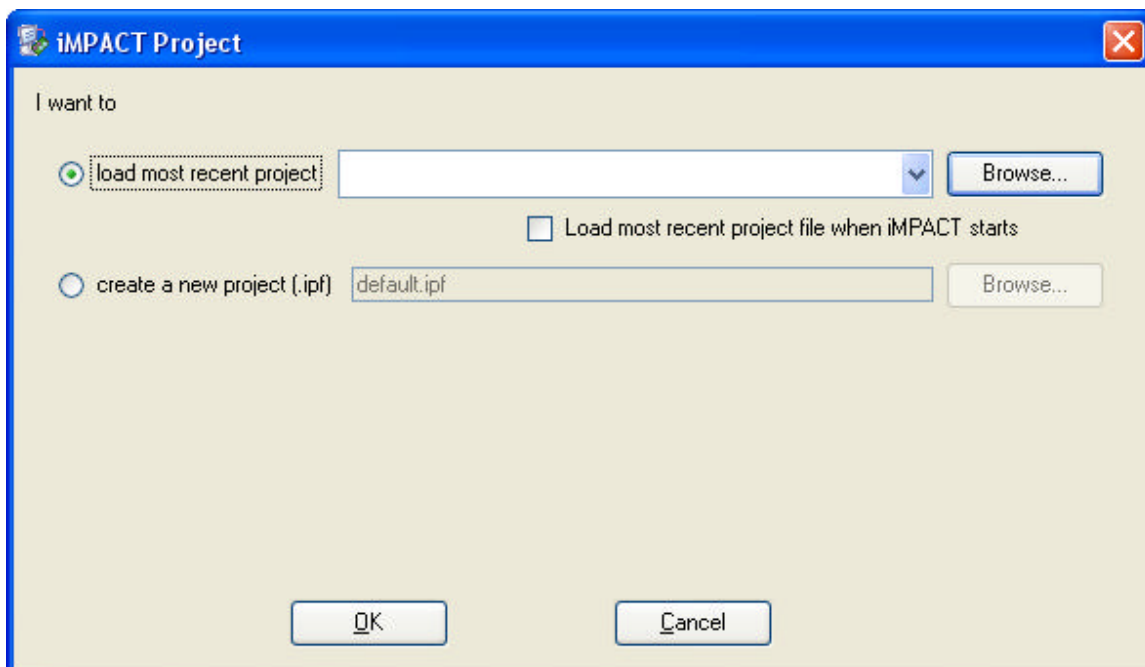
7 Setting Up the Board

Perform the following steps to setup the board for running the PPC XMK demo.

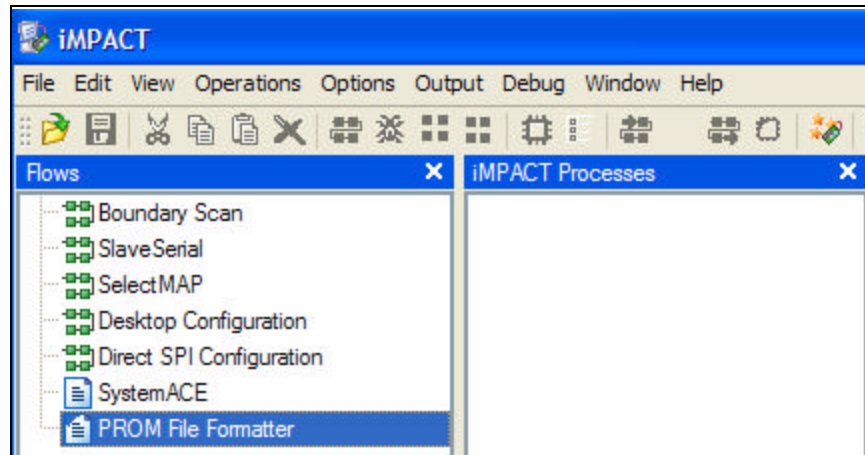
1. Verify the Power switch, **SW7**, is in the **OFF** position.
2. Install a jumper on JP3 pins 2-3
3. Install a jumper on JP2 pins 2-3
4. Install a jumper on JP1 pins 1-2
5. Install a jumper on JP5 pins 2-3 (FPGA JTAG mode)
6. Connect the power supply to the J11 connector on the FXT evaluation board and also plug it into the AC outlet.
7. Connect the USB JTAG cable to J9 and the USB port of the PC.
8. Connect a straight through RS232 cable to the board DB-9 connector (P1) and the serial port of the PC. Alternatively, you can use an RS232-USB adapter and connect this adapter to the DB-9 connector and the USB port of the PC. In this case, you must install the RS232-USB driver for the adapter.
9. Slide the power switch to the **ON** position

8 Generating the Multi-boot MCS File

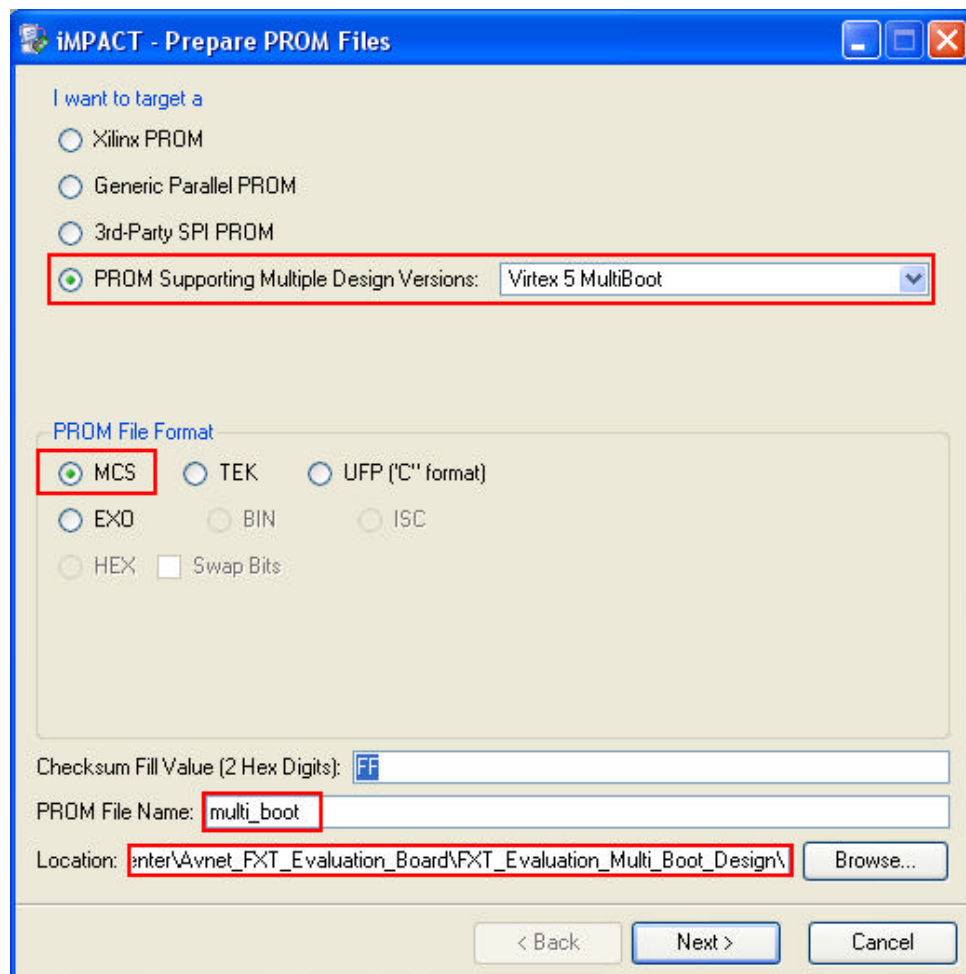
1. Start iMPACT via **Start > All Programs > Xilinx ISE Design Suite 10.1 > ISE > Accessories > iMPACT**, a dialog box similar to the one shown below will appear. Click **Cancel** to continue.



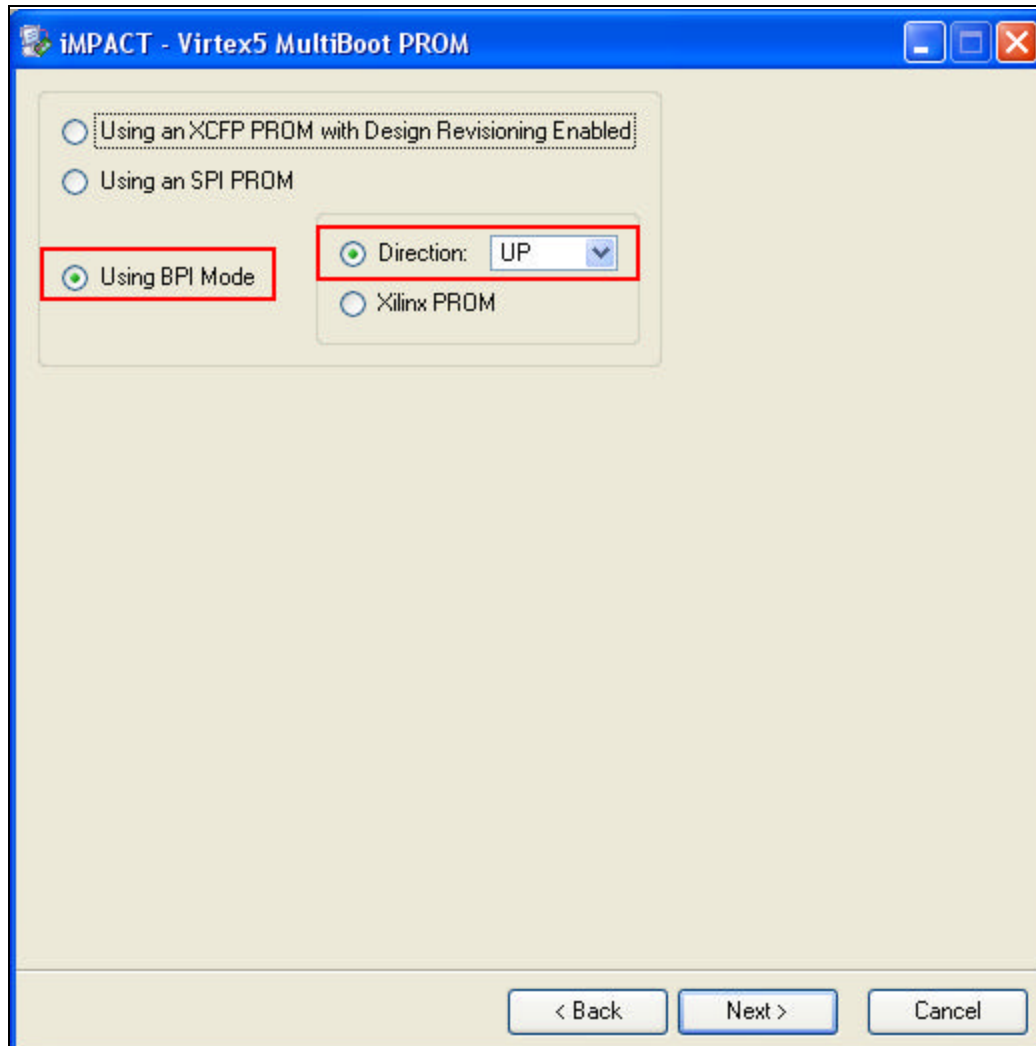
2. Double-click on **PROM File Formatter** to generate an MCS file



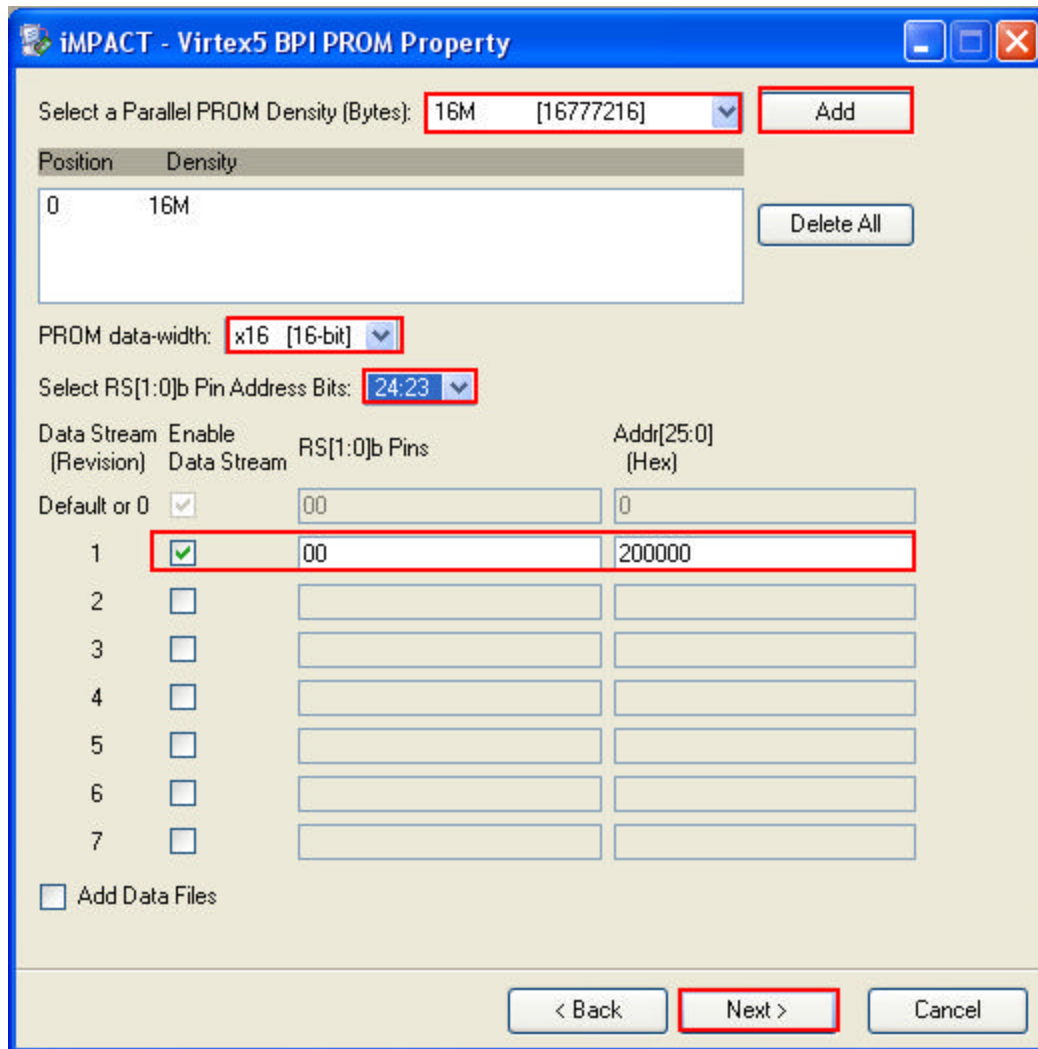
3. In the following dialog box, set the options as shown. The **Location** is where you would like to store the generated MCS file (**FXT_Evaluation_Multi_Boot_Design** folder for this example) and the **PROM File Name** is user selectable (for this lab the file name will be **multi_boot**).



4. In the following dialog box, set the options are shown and click **Next** to continue.



5. In the following dialog box
- Select the **16M** parallel PROM from the drop-down dialog box.
 - Click on **Add** to add the PROM.
 - Set the **PROM data-width** to **x16 (16-bits)**.
 - Set the **Select RS[1:0]b Pin Address Bits** to **24:23**.
 - Check the configuration location '1' and set the RS and Address values as shown (RS = 00 and Address = 200000)
 - Click **Next** to continue

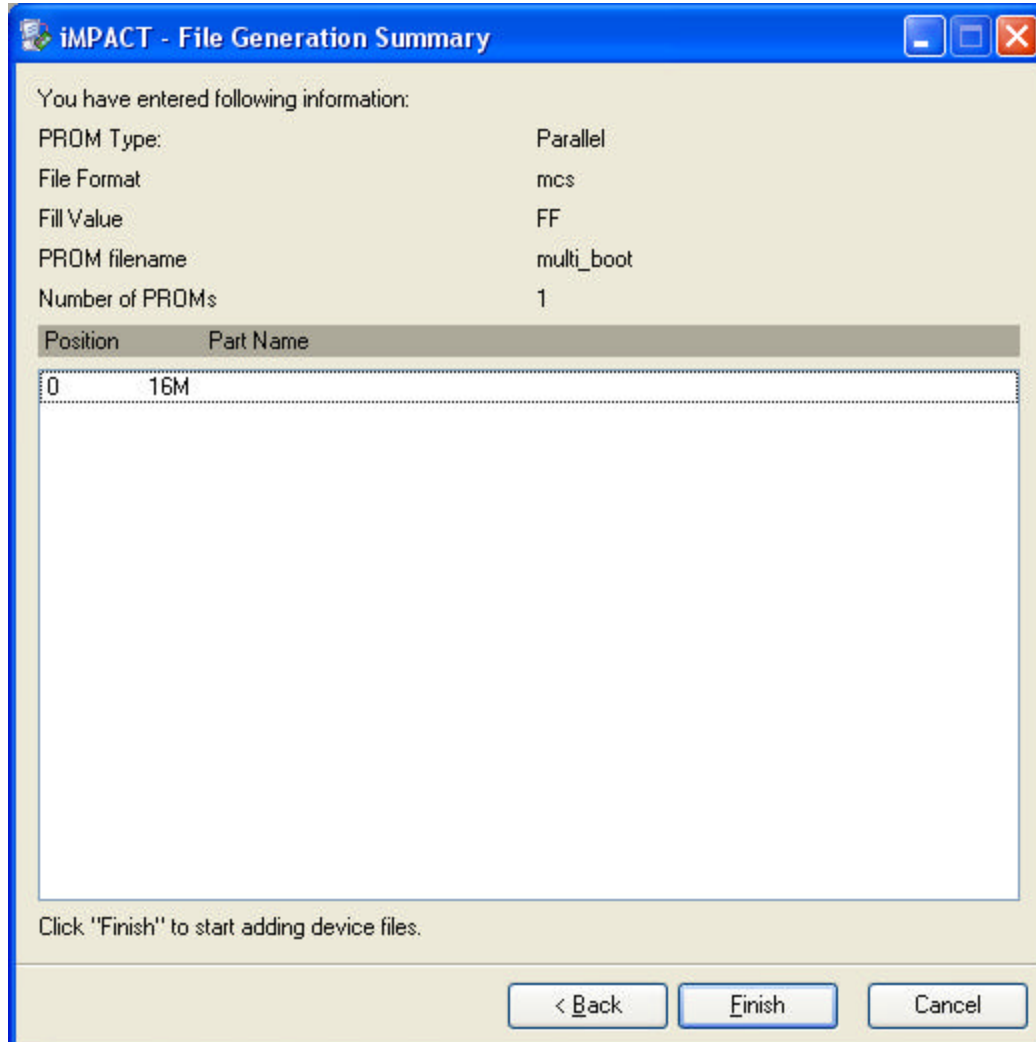


The dialog box is titled "IMPACT - Virtex5 BPI PROM Property". It contains the following elements:

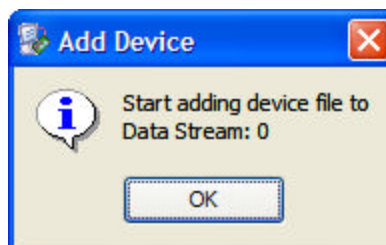
- Select a Parallel PROM Density (Bytes):** A dropdown menu showing "16M" and "[16777216]". An "Add" button is to the right.
- Table:** A table with two columns: "Position" and "Density". It contains one row with "0" and "16M". A "Delete All" button is to the right.
- PROM data-width:** A dropdown menu showing "x16 [16-bit]".
- Select RS[1:0]b Pin Address Bits:** A dropdown menu showing "24:23".
- Table:** A table with four columns: "Data Stream (Revision)", "Enable Data Stream", "RS[1:0]b Pins", and "Addr[25:0] (Hex)". It contains eight rows, numbered 1 to 7. Row 1 is highlighted with a red box. The "Enable Data Stream" column has checkboxes. The "RS[1:0]b Pins" column has text boxes. The "Addr[25:0] (Hex)" column has text boxes.
- Buttons:** "< Back", "Next >", and "Cancel".

	Data Stream (Revision)	Enable Data Stream	RS[1:0]b Pins	Addr[25:0] (Hex)
Default or 0		<input checked="" type="checkbox"/>	00	0
1		<input checked="" type="checkbox"/>	00	200000
2		<input type="checkbox"/>		
3		<input type="checkbox"/>		
4		<input type="checkbox"/>		
5		<input type="checkbox"/>		
6		<input type="checkbox"/>		
7		<input type="checkbox"/>		

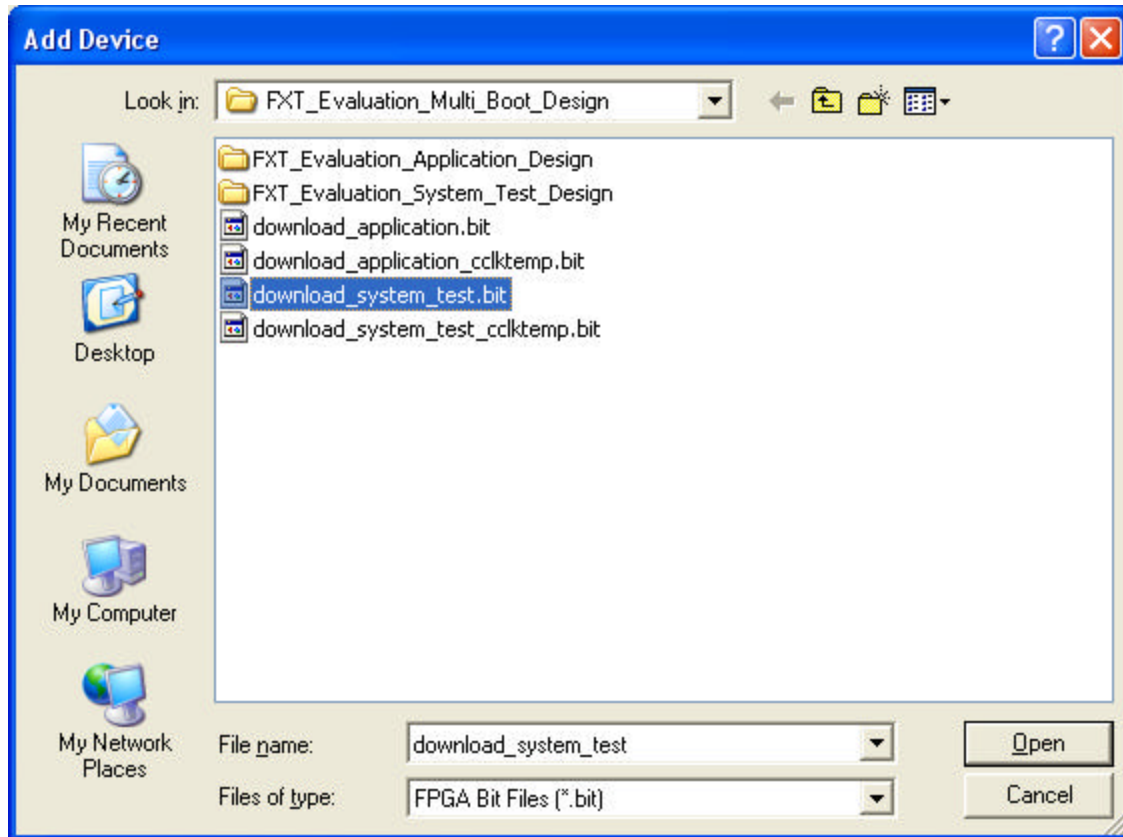
6. In the following dialog box, click **Finish** to continue.



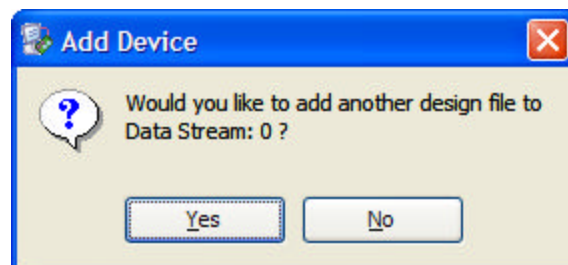
7. When the following dialog box appears, click **OK** to continue.



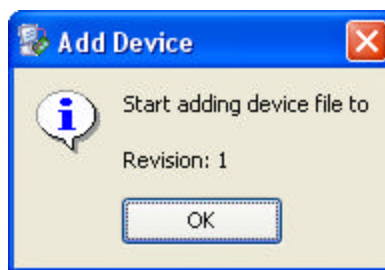
8. Select the **download_system_test.bit** file and click **Open** (this file is located in the **FXT_Evaluation_Multi_Boot_Design** folder of the reference design).



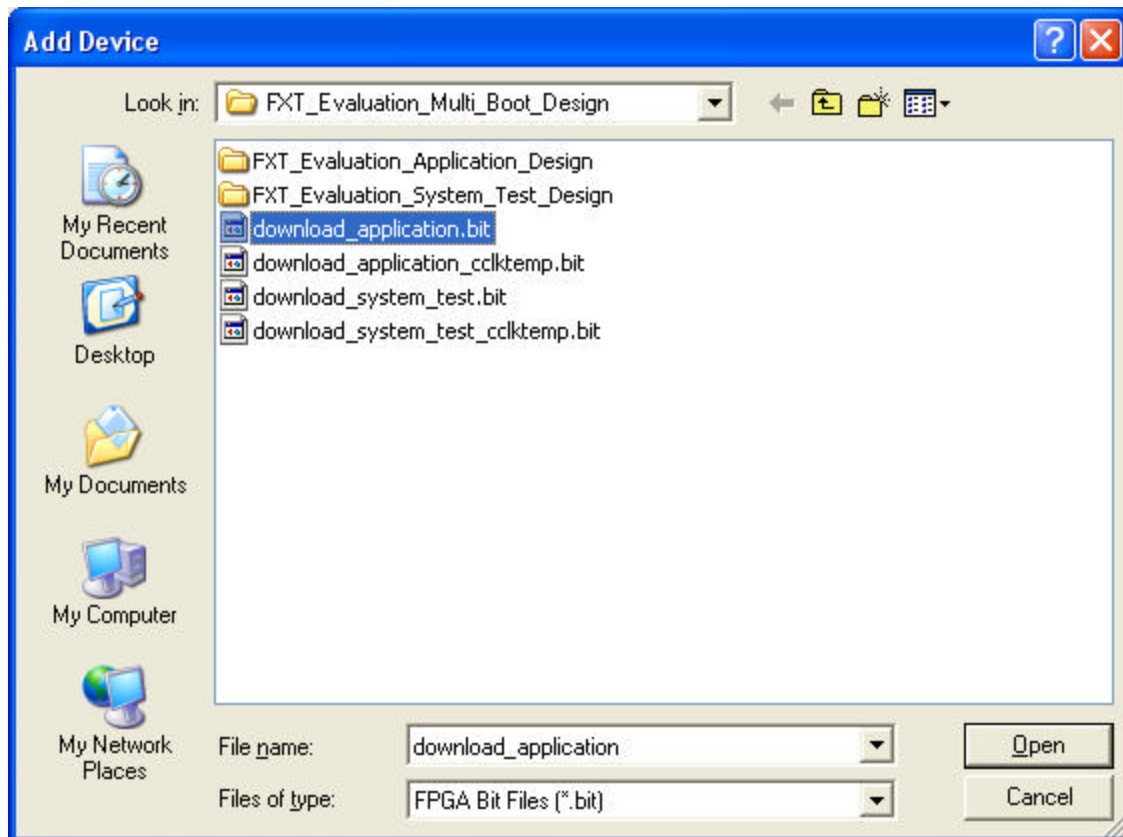
9. When the following dialog box appears, click **No** to continue.



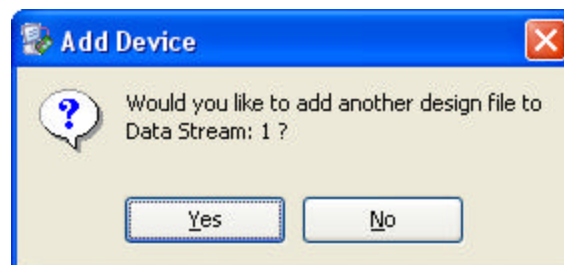
10. Click **OK** to continue.



11. Select the **download_application.bit** file and click **Open** (this file is located in the **FXT_Evaluation_Multi_Boot_Design** folder of the reference design).



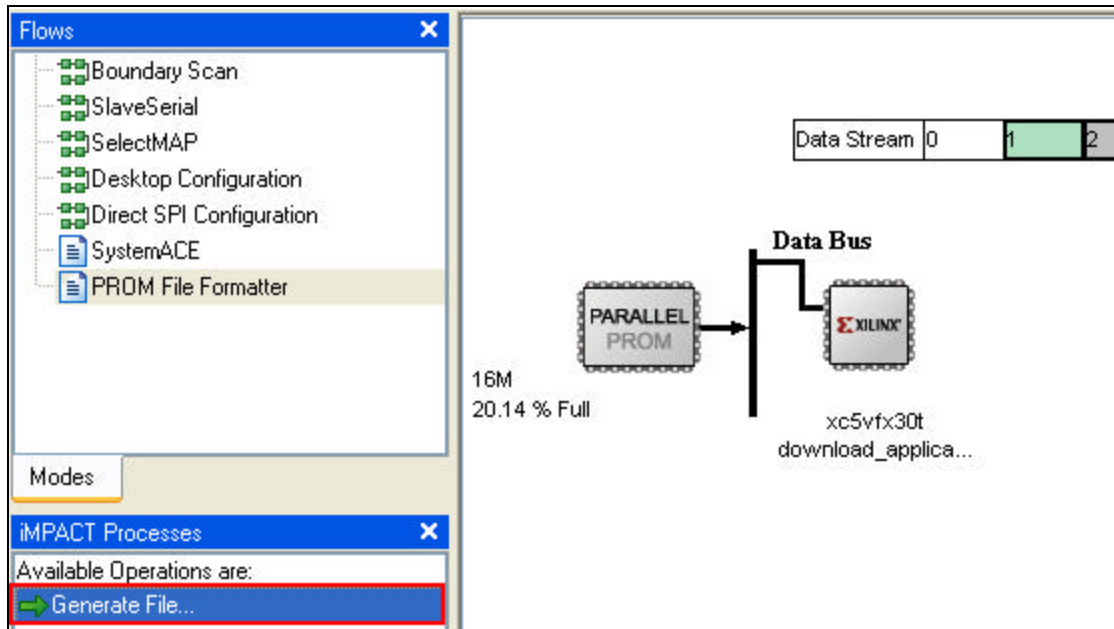
12. When the following dialog box appears, click **No** to continue.



13. Click **OK** to continue.

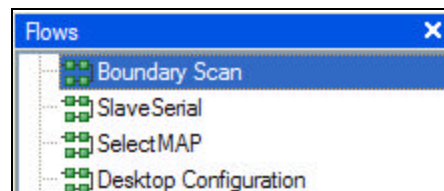


14. Double-click on **Generate File ...** to create the MCS file.

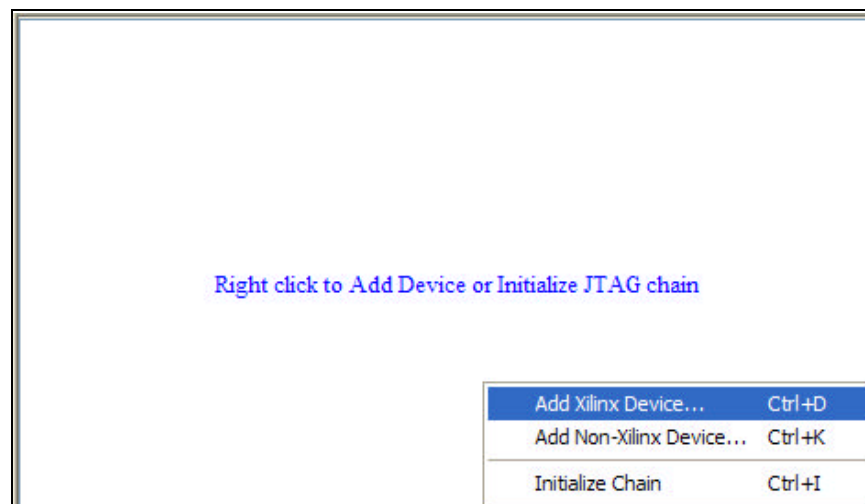


8.1 Programming the BPI Flash

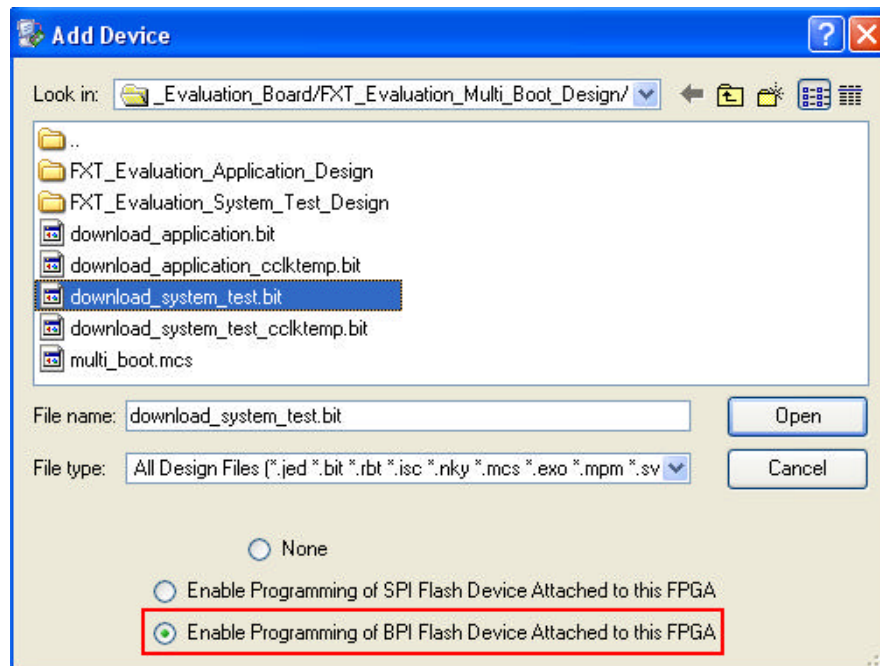
1. Double-click on the **Boundary Scan**.



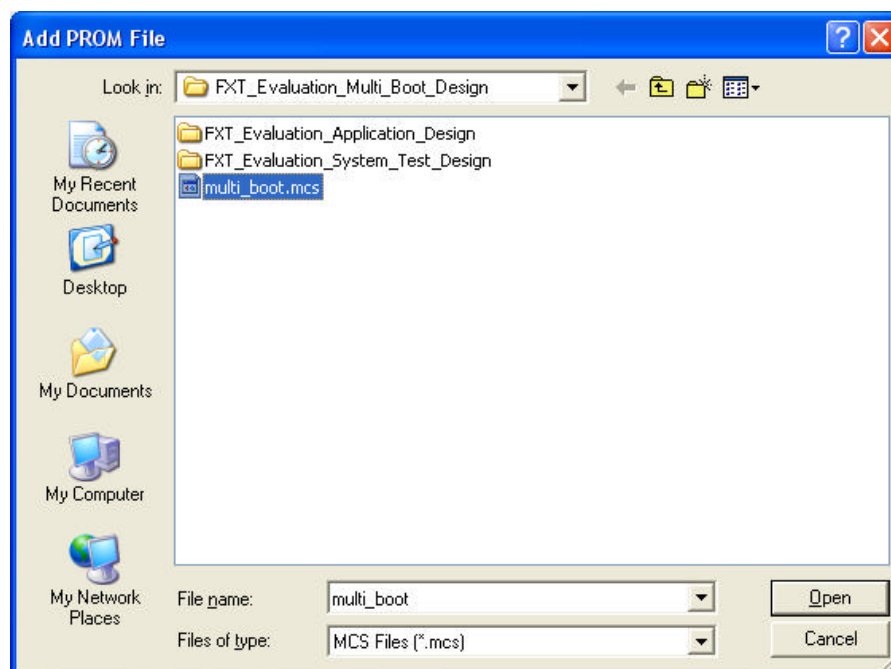
2. Right-click in the white area and select **Add Xilinx Device...**



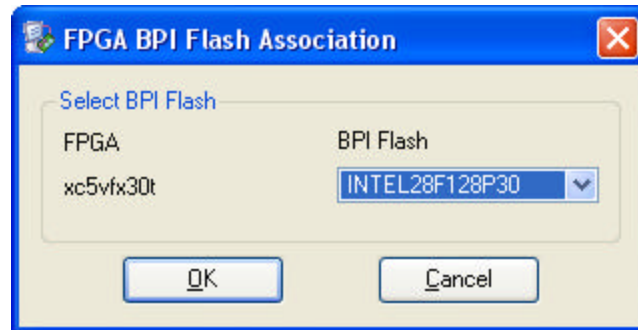
3. In the **Add Device** dialog box:
 - a) Click on **download_system_test.bit** file to select it
 - b) Click on the **Enable Programming of BPI Flash Device Attached to this FPGA** radio button at the bottom of this dialog box.
 - c) Click on **Open** to continue.



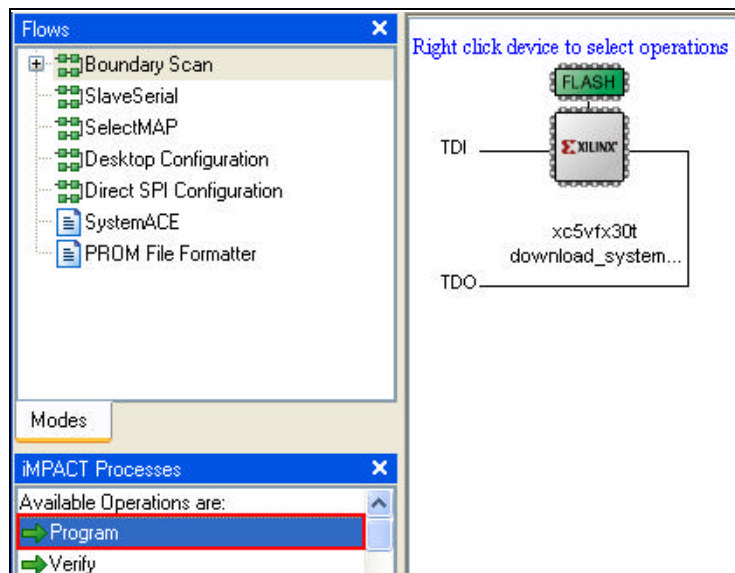
4. In the **Add PROM File** dialog box, select the **multi_boot.mcs** file and click **Open** to continue.



5. Select the Flash device (**INTEL28F128P30**) as shown in the following figure and click **OK** to continue.



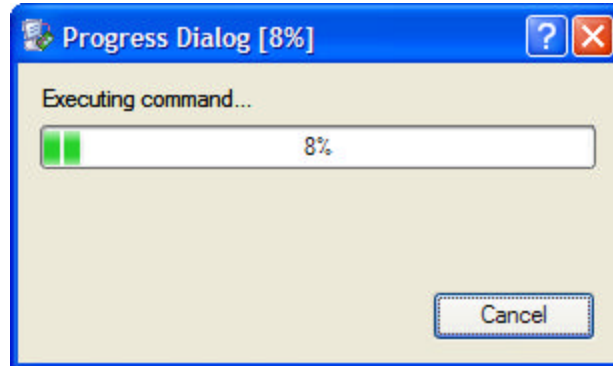
6. Click on the **FLASH** to highlight it (it should turn Green) and double-click in **Program** to program the Flash (alternatively, you can right-click on the Flash device and select **Program**).



7. In the following dialog box, make sure the **Verify** and **Erase Before Programming** are checked and click **OK** to continue.



Flash erase and programming will begin (this will take a couple of minutes).



8. After programming is completed, make sure the FPGA mode jumper is set to BPI (**install a jumper on JP5 pins 1-2**) and power-cycle the board (or press and release the PROGRAM button, SW6). You should see the following on the Hyper Terminal.

