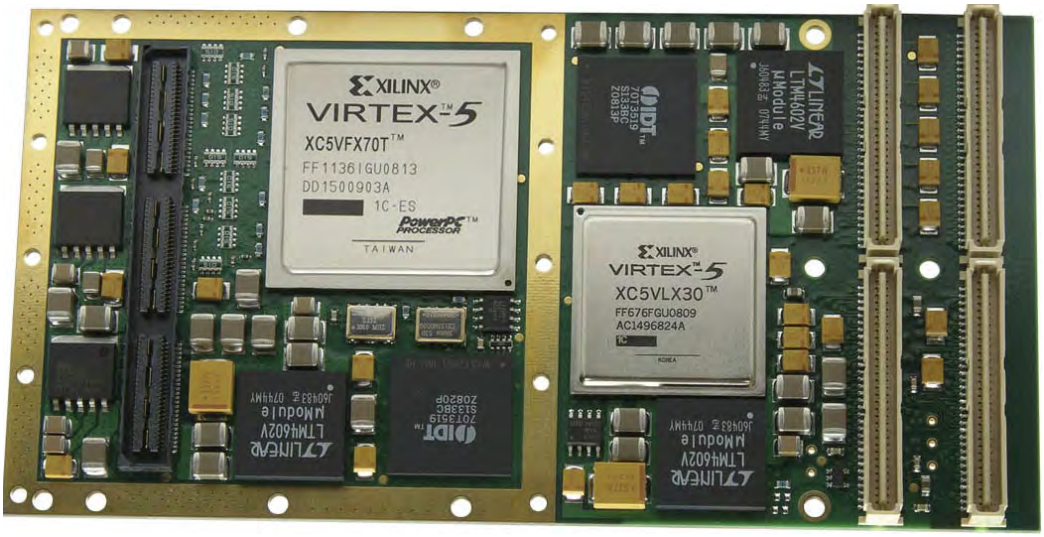
Series PMC-VFX70
Virtex-5 Based FPGA
PMC Module

Getting Started Guide with AXM-A30



ACROMAG INCORPORATED
30765 South Wixom Road
P.O. BOX 437
Wixom, MI 48393-7037 U.S.A.
solutions@acromag.com

Tel: (248) 295-0310
Fax: (248) 624-9234

8500-886-A10D000

TABLE OF CONTENTS

IMPORTANT SAFETY CONSIDERATIONS
You must consider the possible negative effects of power, wiring, component, sensor, or software failure in the design of any type of control or monitoring system. This is very important where property loss or human life is involved. It is important that you perform satisfactory overall system design and it is agreed between you and Acromag, that this is your responsibility.

1.0 General Overview

2.0 Example Design Creation

www.acromag.com

If you have problems, you should visit the Acromag worldwide web site at http://www.acromag.com. Our web site contains the most up-to-date product and software information.

An email question can be submitted from within the "Contact Us" hyperlink at the top of any web page.

Acromag's application engineers can also be contacted directly for technical assistance via telephone or FAX through the numbers listed at the bottom of this page. When needed, complete repair services are also available.

CAUTION: POWER MUST BE TURNED OFF BEFORE REMOVING OR INSERTING BOARDS

RELATED PUBLICATIONS

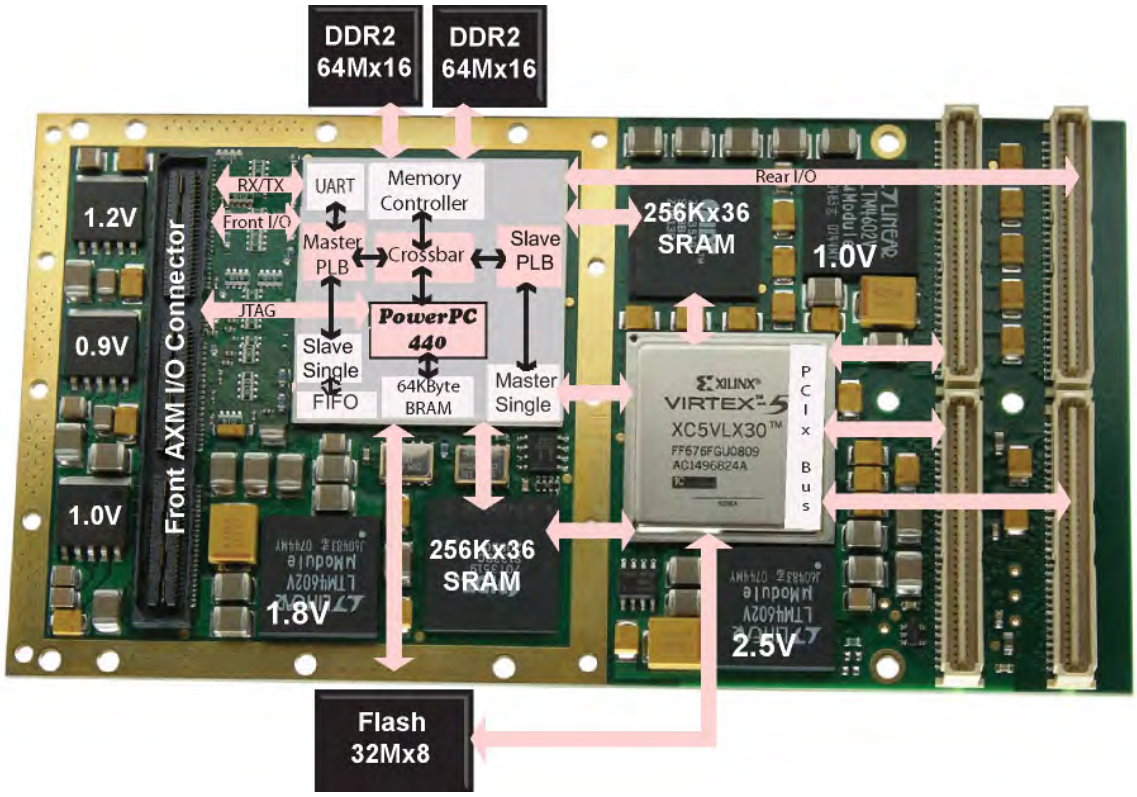
The following manuals and part specifications provide the necessary information for in depth understanding of the board.

Virtex-5 Documentation	**http://www.xilinx.com**
IDT70T3519S Spec.	**http://www.idt.com**
MT47H64M16HR Spec	**http://www.micron.com**
CY23EP05 Specification	**http://www.cypress.com**

1.0 GENERAL OVERVIEW

In this section an overview of the PMC-VFX is presented and the Acromag example design that the PMC-VFX executes is also described.

The provided PMC-VFX example design can serve as the launching point from which your custom design can be developed. The first step is to become familiar with the provided example design.



The PMC-VFX comes preprogrammed with the example design. This program is stored in a 32Mx8 Flash memory. The 32Mx8 flash memory is a 256 sector memory with the first 128 sectors allocated for storage of the reprogrammable FPGA program code. The second 128 sectors are allocated for storage of PowerPC code and data. The PMC-VFX comes preprogrammed with both FPGA program code and PowerPC program code.

The reprogrammable FPGA code is defined by both VHDL files and PowerPC files. All VHDL and Xilinx PowerPC files used to define the example design are provided in the Engineering Design Kit (EDK).

FPGA Fabric Functions

The main functions controlled by the VHDL include:

- **Local Bus Interface** This VHDL logic provides an interface to a second smaller Virtex 5 FPGA that handles the PCI-X bus interface. The local bus interface performs with the reprogrammable FPGA acting as a slave and the PCI-X bus FPGA acting as the master. The local bus interface has a 32-bit data bus, address lines 21 to 2, four byte strobe signals, and five additional control signals.

- **Dual Port SRAM interface (DP_SRAM Component)** A 256K x 64-bit synchronous dual port SRAM memory is provided. One port interfaces to the PCI bus, and the other port is directly connected to the reprogrammable FPGA. This memory supports DMA transfers when requested by the system or the reprogrammable FPGA.
- **Front I/O Interface (AXM_A30 Component)** An interface to front of panal I/O mezzanine modules of various I/O standards is provided by way of a 150 pin high speed connector. The interface includes 31 differential signal pairs, 29 control signals, 2 clock signals, and 53 power and ground signals.
- **Rear I/O Interface (RearLVDS Component)** The reprogrammable FPGA is directly connected to 64 pins of the rear P4 connector. The reprogrammable FPGA I/O to these signals is powered by 2.5 volts and can perform any 2.5volt standard FPGA I/O.
- **PowerPC Interface (PowerPC Component and XC5VFX70T VHDL)** Access to the DDR SDRAM and Block RAM address space from the PCI bus is performed by way of the PLBV46 Master Single core. The PCI bus transfer of data is implemented using the following registers: DDR SDRAM Control/Status, DDR SDRAM Address, DDR SDRAM Read, DDR SDRAM Write, and DDR SDRAM Mask Registers. These registers are implemented in the XC5VFX70T VHDL and are described in the PMC-VFX User's Manual. Also, a user interface to the PLBV46 Slave Single core provides PowerPC read access of the Flash Program Code FIFO. The XC5VFX70T VHDL code executes the move of the PowerPC program code/data from flash memory starting at address sector 128 to the Flash Program Code FIFO.

PowerPC Functions

The main functions coded in the PowerPC files include:

- **64Mx32-bit DDR2 Memory Controller** The DDR2 SDRAM is directly accessible by the PowerPC. A DDR2 Memory Controller for the PowerPC 440 Processor Xilinx core implements this interface between the PowerPC and the DDR memory. The core is referenced as the PPC440MC core.
- **64K Byte Block RAM** The bootloop program is executed out of this memory. The bootloop program is preloaded in this Block RAM as part of the FPGA configuration file. The bootloop program will automatically execute upon power-up or reconfiguration.
- **PLBV46 Master Single Core** A user interface to the PLBV46 Master Single core provides a bi-directional interface between the PCI bus and the DDR memory. This core allows read and write of DDR SDRAM and Block RAM from the PCI bus. The PCI bus is the master of this interface.
- **PLBV46 Slave Single Core** This core allows PowerPC read access of the Flash Program Code FIFO. This core also handles PowerPC write access to the PowerPC_Read_Reg. The PowerPC_Read_Reg can be read by the PCI bus at PCI bus BAR2 address plus 0x8070.
- **UART1** The UART can not be used in conjunction with the AXM-A30.
- **UART2** The UART can not be used in conjunction with the AXM-A30.

PowerPC use and the AXM-A30

This getting started guide only introduces the basic concepts of creating and modifying the VHDL project. THIS GUIDE DOES NOT CONTAIN INSTRUCTIONS ON PROGRAMMING THE POWER PC. Please refer to the Virtex 5 Getting Started Guide located in the Engineering Design Guide for more information on programming the PowerPC.

Please note the PowerPC UART's cannot be used at the same time as the AXM-A30.

Dual Clock FIFO

The AXM-A30 contains a FIFO created using the Xilinx Core Generator. of Specifically the 5.1 FIFO Core Generator was used. The FIFO has the following properties:

Component name: ADCFIFO
Type: Independent Clocks using Block RAM
Write Width: 16 bits
Write Depth: 8192
Read Width: 16 bits

Options: No ECC, No Embedded Registers, No Flags

Reset Pin used (synchronous).
Dout Reset Value = 0

No Programmable Threshold

Using Read Data Count with Width 13 bits.

2.0 Example Design Creation

The development of the Xilinx PMC-VFX project was performed using the Xilinx ISE Design Suite 11.1 with no service pack. A great deal can be learned about how the Xilinx ISE, XPS (Xilinx Platform Studio), and SDK tools are used to develop a custom application by performing the following project creation procedures. The files provided in the EDK include the completed project executed in the example design. The steps given in the following pages will allow one to incrementally develop the project.

New Project Creation

1. Make a new directory on your computer (XC5VFX70T_REVD_AXMA30_11p1) or copy the directory structure provided on the EDK CD: Example directory: C:\Designs\PMC_V5\Programmable_FPGA\XC5VFX70T_REVD_AXMA30_11p1

2. Copy the following files
   AXM_A30.vhd,
   DP_SRAM.vhd,
   RearLVDS.vhd,
   v5_sysmon_v1_0.vhd,
   XC5VFX70T.vhd,
   XC5VFX70T.Ucf
   All of the files in the ipcore directory. ADCFIFO.* (There are a total of 14 files).

from the CD-ROM to the new directory.

3. Copy the PMC_VFX70_v2_2_0.xbd and *.ucf files from the EDK CD ROM \board\Acromag\boards\PMC_VFX70\data folder to \Xilinx\11.1\EDK\board\Acromag\boards\PMC_VFX70\data folder

4. Copy the pcores folder provided in the EDK CD ROM edk_user_repository\MyProcessorIPLib folder to \Xilinx\11.1\edk_user_repository\MyProcessorIPLib folder

5. Start the software by selecting:
   Start -> Programs -> Xilinx ISE 11.1 ->ISE-> Project Navigator

6. To create a new project the following steps can be taken. Alternatively, the project provided on the CD-ROM can serve as a starting point.

7. File -> New Project
   Project Name VFX70
   Project Location Give path to VFX70 directory created C:\Designs\PMC_V5\Programmable_FPGA\XC5VFX70T_REVD_AXMA30_11p
1 **Top-Level source type** HDL

Select-> Next

Enter the following Device Properties
Product Category All
Family Virtex5
Device XC5VFX70T
Package FF1136
Speed Grade -1
Top-Level Source Type HDL
Synthesis Tool XST (VHDL/Verilog)

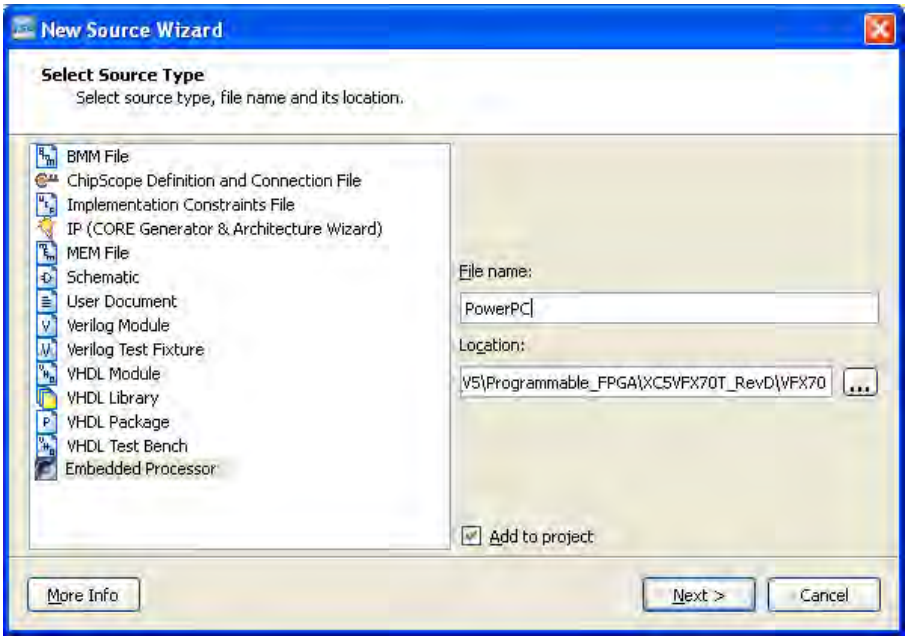
Simulator Modelsim-XE VHDL
Preferred Language VHDL
Select ->**Enable Enhanced Design Summary**
Select-> Next

Create New Source Dialog Opens
Select-> Next (no new source are needed)

Add Existing Sources Dialog
Select **Add Source** button
Browse to the directory location of the .vhd and ucf files you moved in step 2 above.
-> Add Source -> XC5VFX70T.vhd -> Open
-> Add Source -> AXM_A30.vhd -> Open
-> Add Source -> ADCFIFO.xco -> Open
-> Add Source -> DP_SRAM.vhd -> Open
-> Add Source -> RearLVDS.vhd -> Open
-> Add Source -> v5_sysmon_v1_0.vhd -> Open
-> Add Source -> XC5VFX70T.Ucf -> Open
They should not be copied to project since they already are stored in the project directory. De-Select the Copy to Project check box.
Select-> Next
Select-> Finish
-> Adding source files -> OK

8. In the Source in Project Dialog Window select "XC5VFX70T-XC5VFX70T_arch(XC5VFX70T.vhd)" so that it is hi-lighted

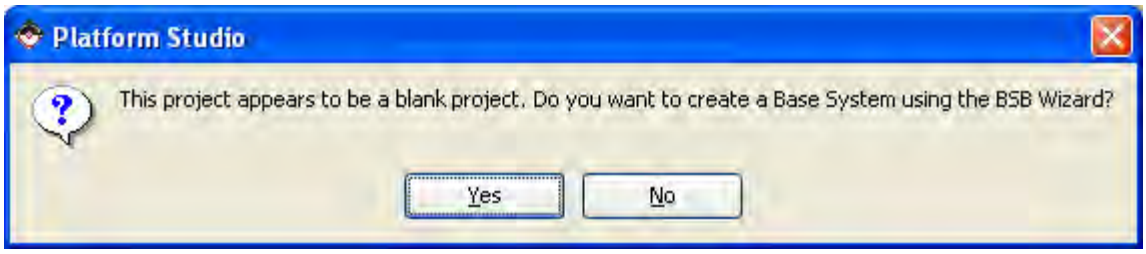
9. Add the processor subsystem as a module in the ISE tool.
Select **Project -> New Source** and then “Embedded Processor” from the resulting list.
Enter the file name **PowerPC**.



Select ->Next
Select ->Finish

This will add an xmp file to the design and launch XPS. In the future **clicking on it will launch XPS (Xilinx Platform Studio)**. In XPS you can build the processor system. Exiting XPS leaves you back in ISE, and you can add your other existing modules to the ISE project just as you would have previously when there was no embedded processor.

Xilinx Platform Studio Base System Builder

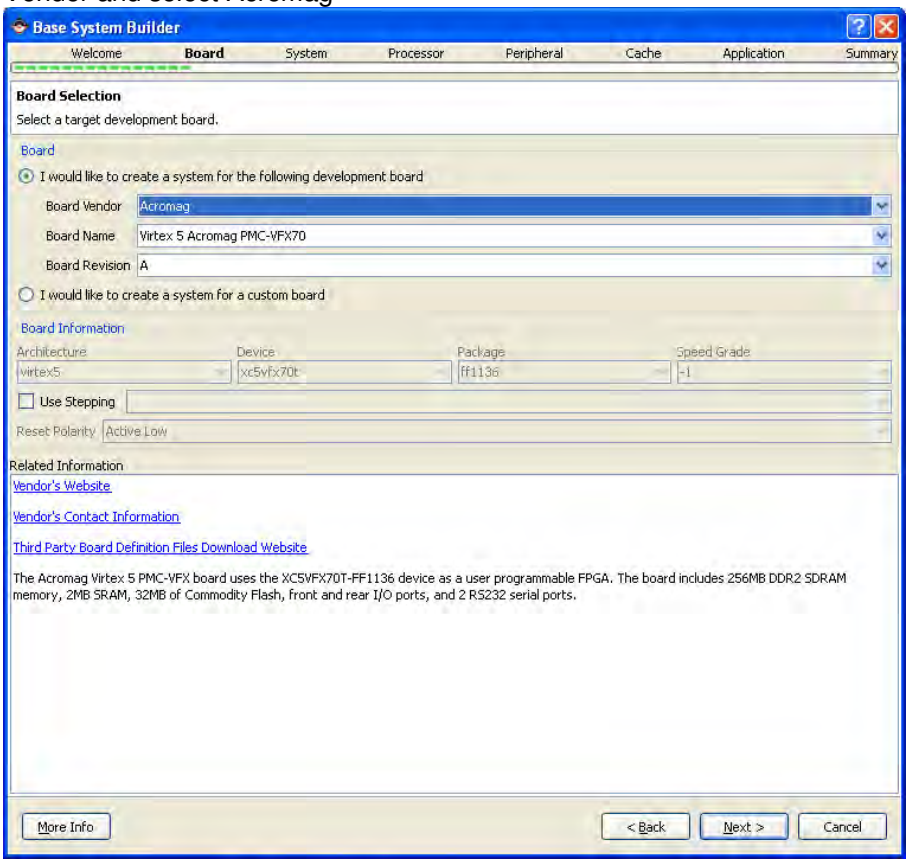


Select ->Yes

Welcome to the Base System Builder! Dialog opens

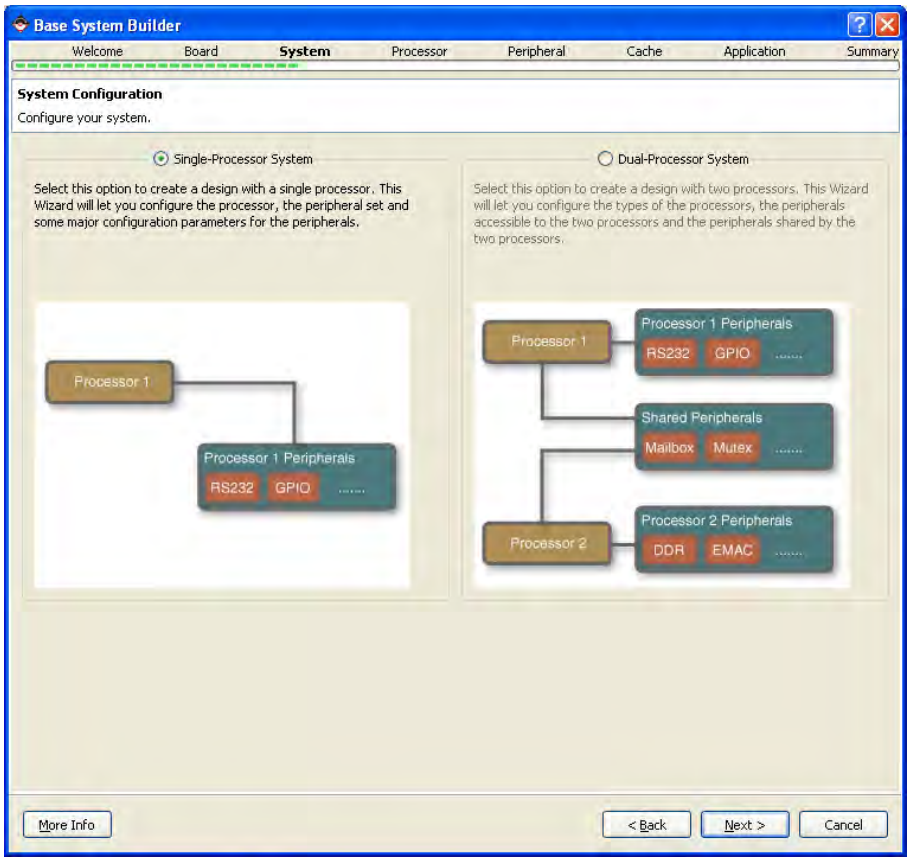
Select "**I would like to create a new design**"
Select -> Next

The Board Selection Dialog opens.
Select the down arrow of the Board Vendor and select Acromag



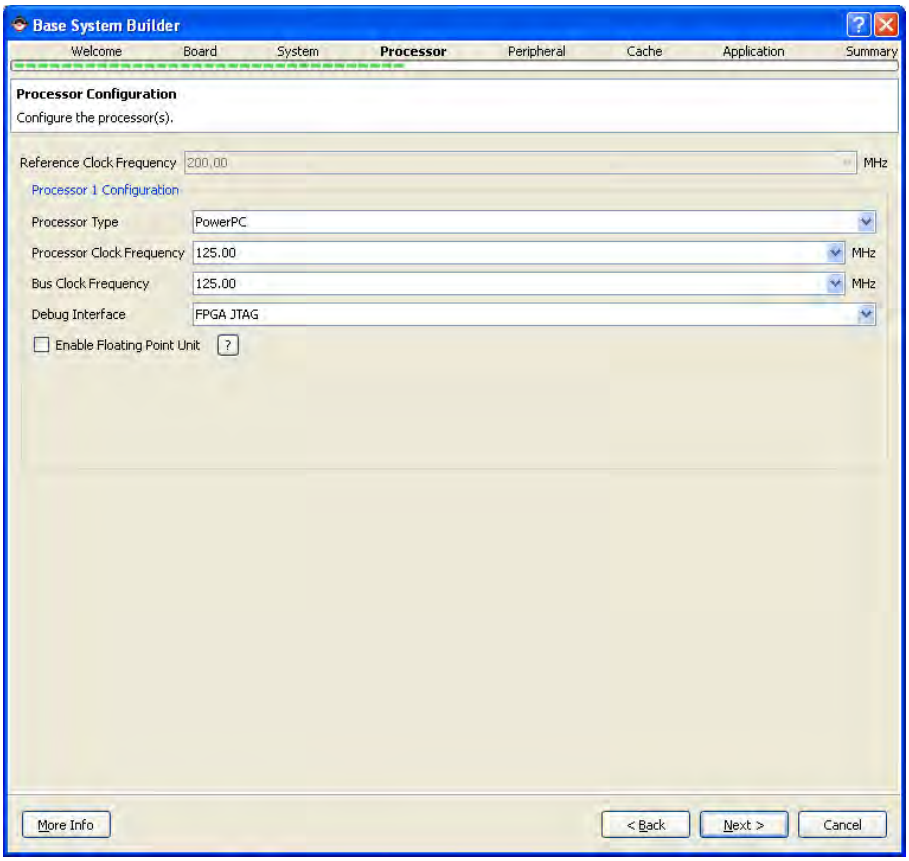
Select-> Next

The System Configuration Dialog opens.
Select **Single Processor System**



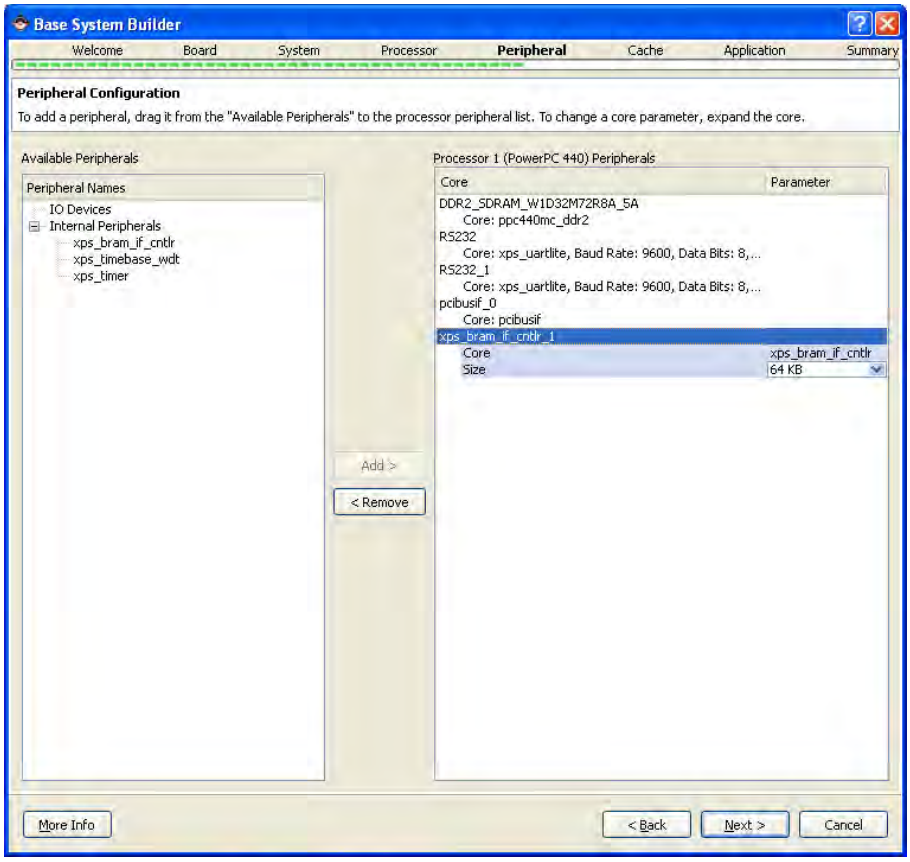
Select -> Next

The Processor Configuration Dialog opens.
Keep all the setting shown.



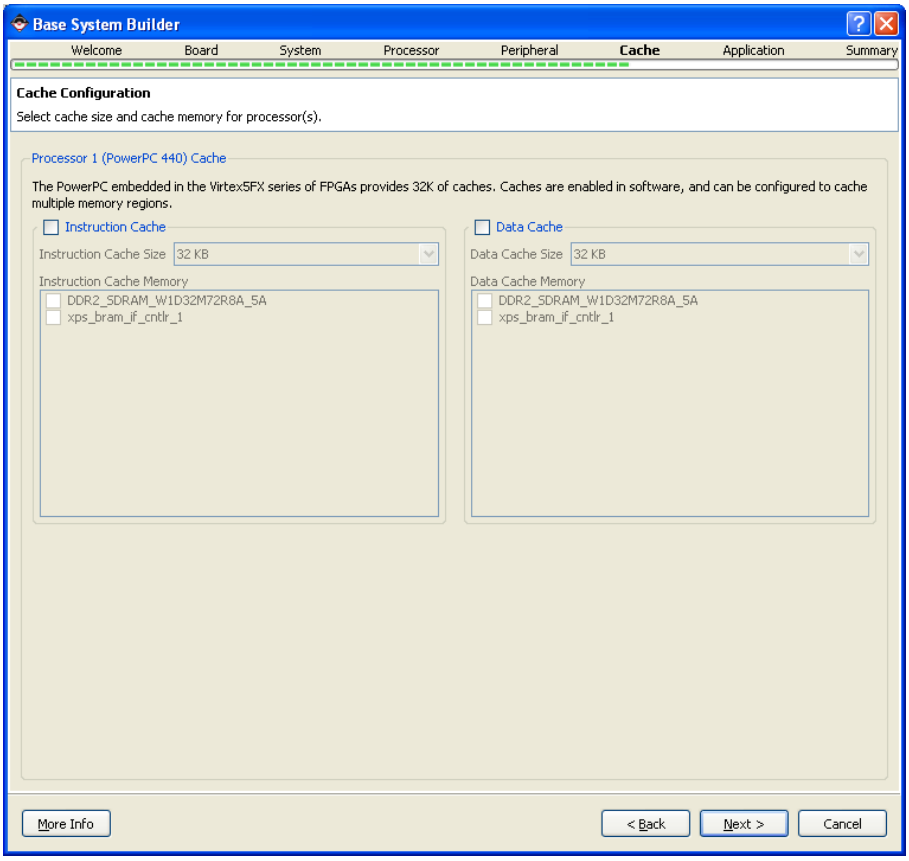
Select -> Next

The Peripheral Configuration Dialog opens.
Change the Memory Size for the Block RAM to 64K as seen below.



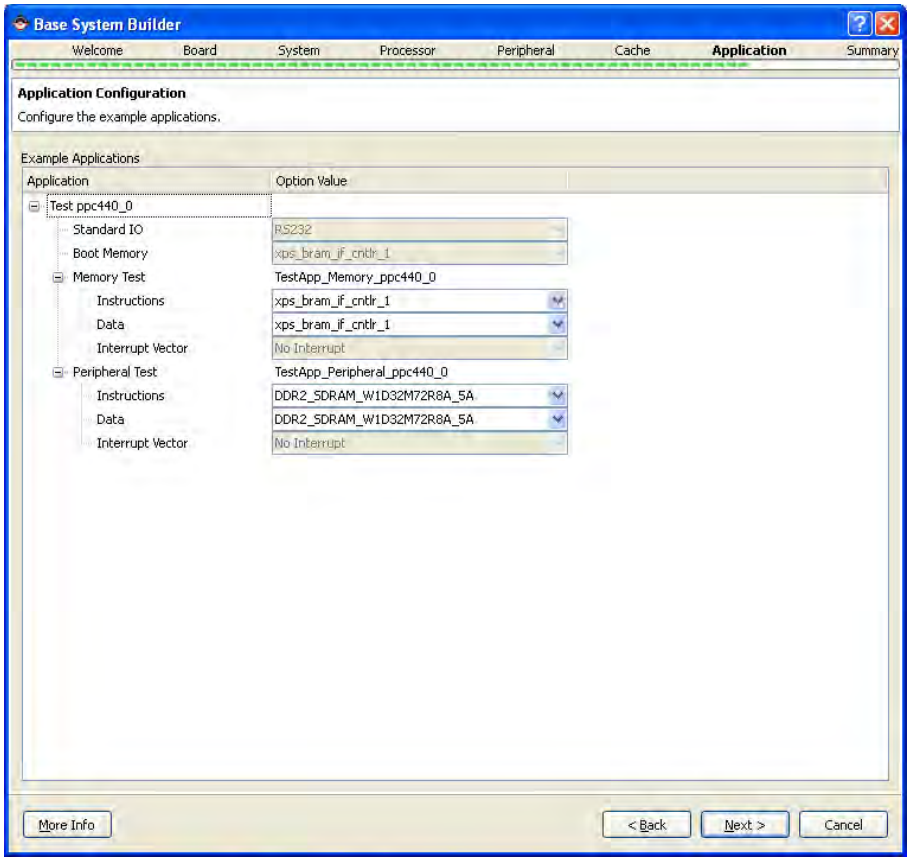
Select -> Next

The Cache Configuration Dialog opens.
Keep all the setting shown.



Select **->** Next

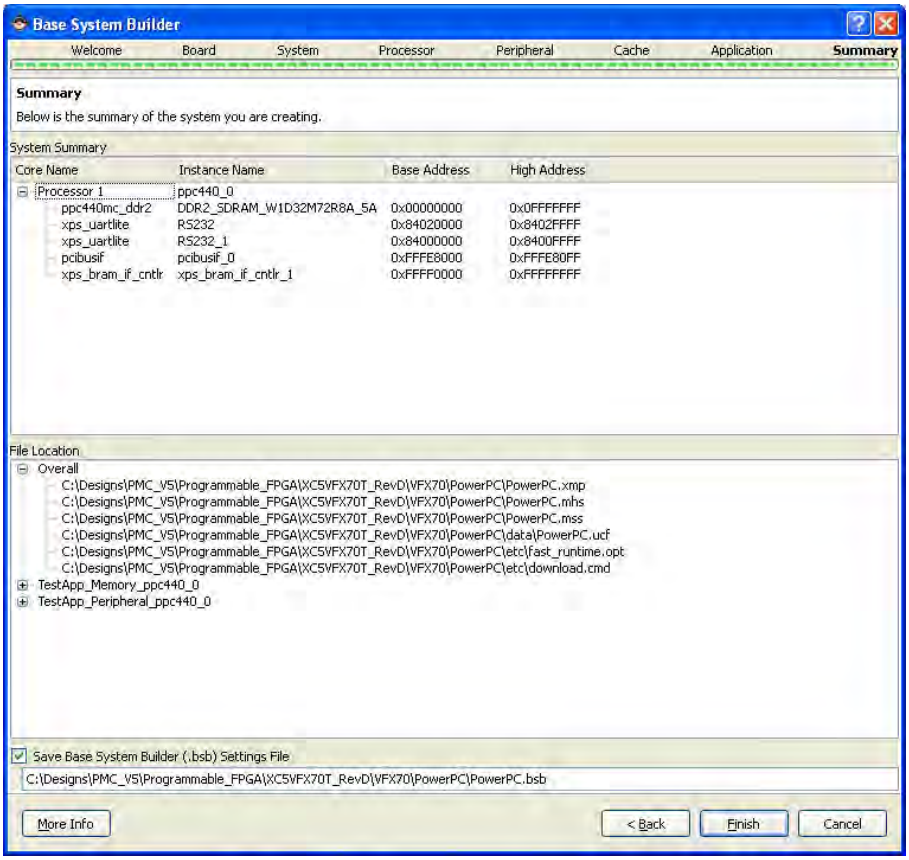
The Application Configuration Dialog opens.
Keep all the setting shown.



Select ->Next

The Summary Dialog opens.

Review the summary of project setting shown in this dialog. Notice the PowerPC base address assigned to the Block RAM, UARTs, and DDR2 memory, and pcibusif



Select -> Finish

In Xilinx Platform Studio window (XPS)

PowerPC MHS File

Make the following changes to the PowerPC.mhs file.
Double select MHS File: PowerPC.mhs under the Projects Files to open it.

At line 218 of the PowerPC.mhs file replace clk_200_0000MHz with dcm_clk_s.
The new line 218 should look like the following.
PORT mi_mcclk_200 = dcm_clk_s

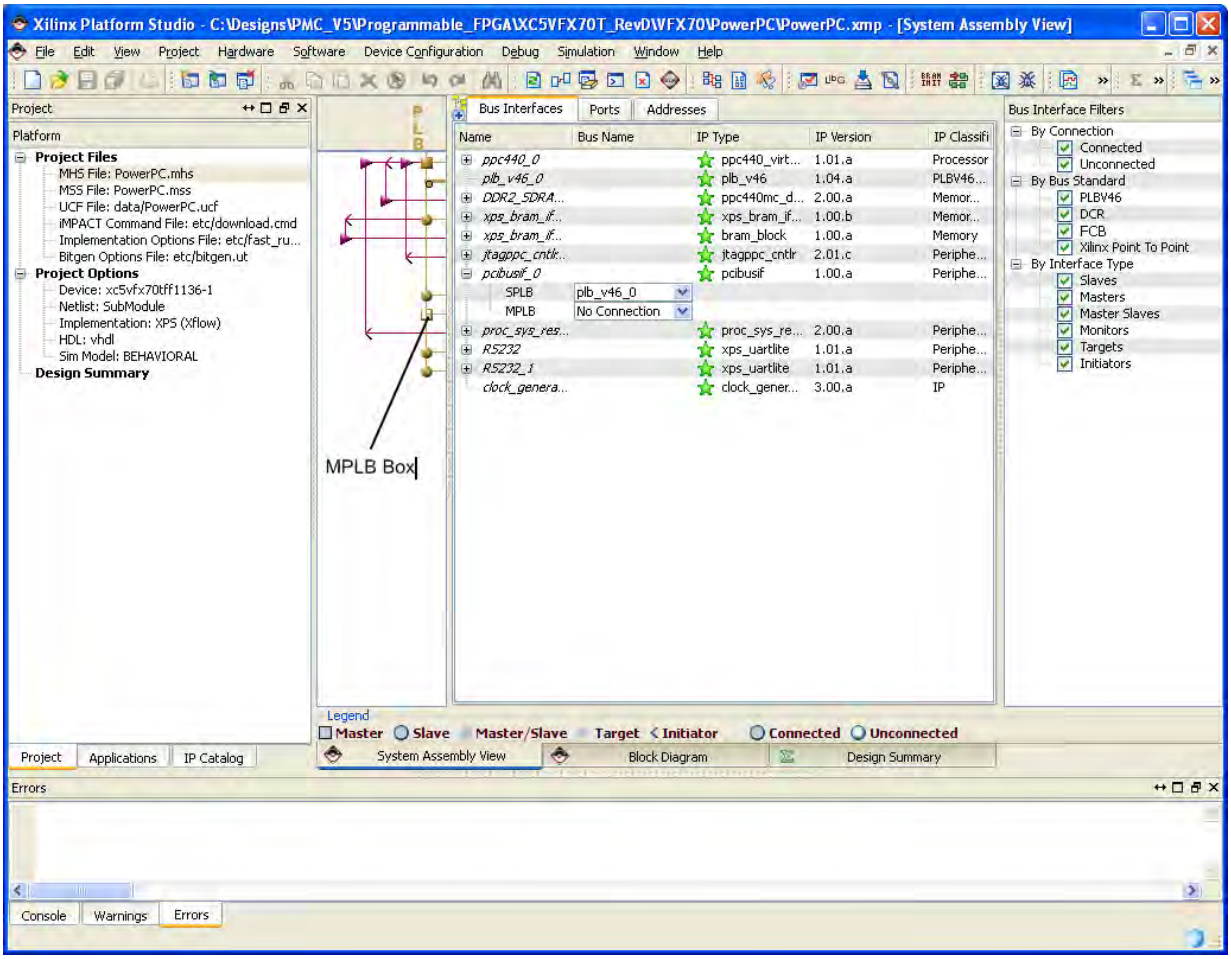
Find the BEGIN clock_generator section and comment out lines250 to 253 and also line 263. A line is commented out by placing the # sign at the beginning of the line. Lines 250 to 253 and 263 should look like the following.

```
# PARAMETER C_CLKOUT3_FREQ = 200000000
# PARAMETER C_CLKOUT3_BUF = TRUE
# PARAMETER C_CLKOUT3_PHASE = 0
# PARAMETER C_CLKOUT3_GROUP = PLL0_ADJUST

# PORT CLKOUT3 = clk_200_0000MHz
```

The PCIBUSIF needs to be added to the bus interface as seen in the Bus Interface tab

Notice in the PLB bus interface view the box is not filled in. This means that the pcibusif_0 has not yet been connected to the PLB bus. Select the box next to MPLB and it will be connected to the PLB bus.

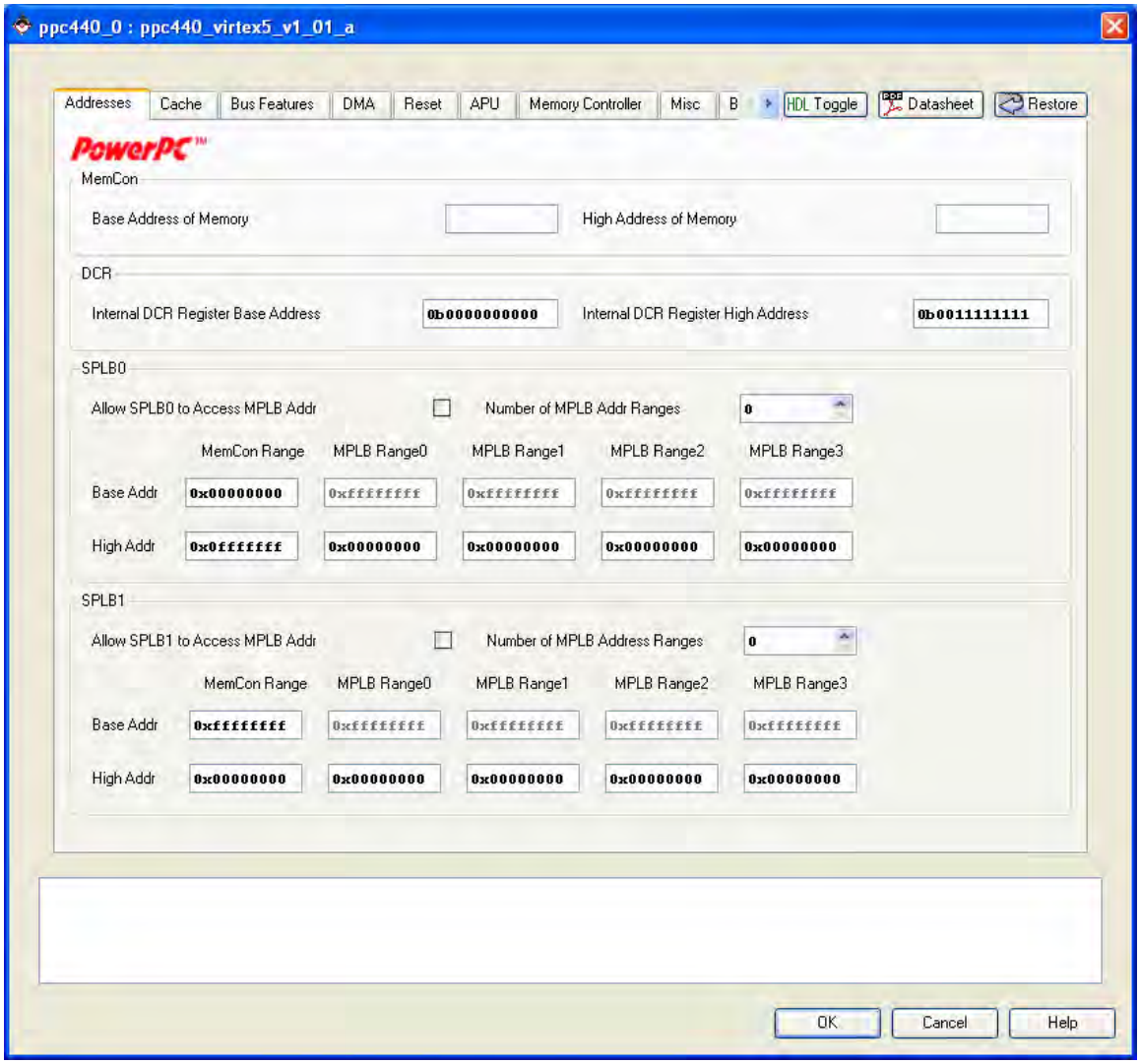


A PLBV46 Master Single custom IP is used to implement an External Master of the PCI bus. The use of the PLBV46 Master Single IP connects to the crossbar using a SPLB0/1 interface.

Currently only the MPLB port, where the PPC440 is the master, is aware of the External Memory connections to DDR. The SPLB0/1 ports require the MEMCON address ranges to be set to allow PLBV46 Master Single IP to have access to DDR Memory.

In XPS with Project tab and the Bus Interface tab selected right mouse select on ppc440_0 and select Configure IP.
The ppc440_0:ppc440_virtex5_v1_01_a dialog opens.

Set the base address to 0x00000000 and High Addr to 0FFFFFFF for the SPLB0 MemCon Range.

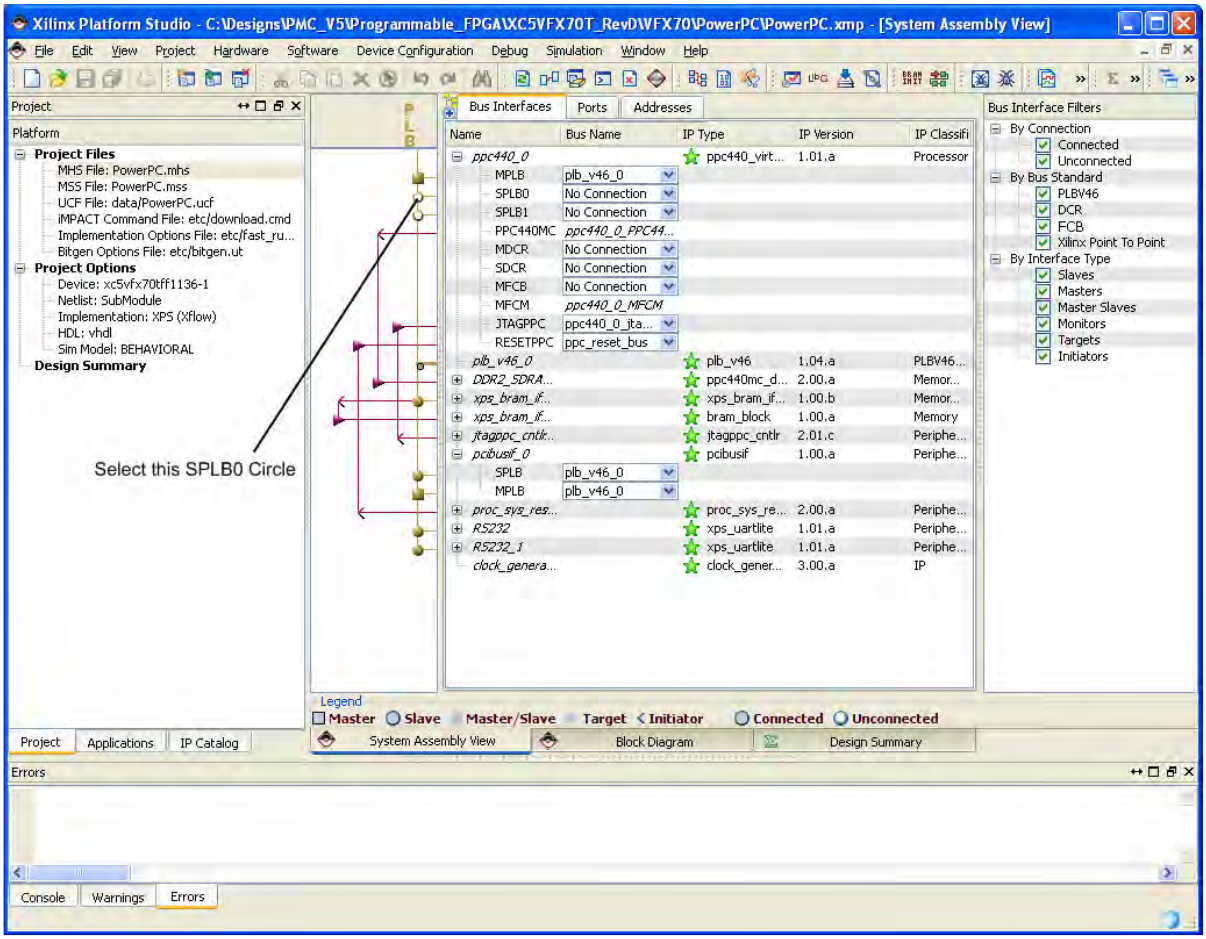


Select -> OK

In the MHS file the parameters C_SPLB0_RNG_MC_BASEADDR and C_SPLB0_RNG_MC_HIGHADDR have been set to the address range as set above. Lines 72 and 73 of the .mhs file should read as follow:

PARAMETER C_SPLB0_RNG_MC_BASEADDR = 0x00000000
PARAMETER C_SPLB0_RNG_MC_HIGHADDR = 0x0FFFFFFF

The SPLB0 still needs to be connected to the PLB bus. In XPS with the Project tab selected and the Bus Interface tab selected, select the circle next to the SPLB0 to make the connection.



File -> Save Project
File -> Exit

Compile Design In ISE

The Xilinx Platform Studio (XPS) project has already been instantiated in the xc5vfx70t.vhd file.
This was done as follows:

- Select PowerPC in the ISE source tab;
- In the processes tab, double select “View HDL Instantiation Template”
- Copy and paste the provided text into the xc5vfx70t.vhd and connect the ports correctly

The XPS will become a submodule in the ISE project.

The xc5vfx70t.vhd file as provided in the EDK already contains the PowerPC component Declaration. The component declaration can be found at lines 477 to 523.

The PowerPC instantiation can be found in the xc5vfx70t.vhd file at lines 1056 to 1100.

In the Processes For Source Dialog Window select:
"Synthesize-XST" so that it is highlighted

Run the compiler by selecting:
Process -> run

In the Processes For Source Dialog Window select:
"Implement Design" so that it is highlighted

Run the compiler by selecting:
Process -> run

In the Processes For Source Dialog Window select:
"Generate Programming File" so that it is highlighted

Process -> Properties
Then select Startup Options tab

FPGA Start-Up Clock	CCLK
Enable Internal Done Pipe	Do not Check
Done (Output Events)	6
Enable Outputs (Output Events)	5
Release Write Enable	5
Release DLL	Default
Match Cycle	Auto
Drive Done Pin High	Check

-> OK

Process -> Properties
Then select Configurations Options tab

Configuration Rate	2
Configuration Pin M0	Pull Down

All other settings remain unchanged
-> OK

Run the Generate Programming File by selecting:
Process -> run

After the design compiles without errors a new program file can be generated.
In the Processes For Source Dialog Window select:
Under Configure Target Device select:
"Generate Target PROM/ACE File" so that it is highlighted

Process -> Run

In the ISE iMPACT dialog select PROM File Formatter from under the iMPACT Flows section

Step 1. Select Generic Parallel PROM and then select the Green Arrow Parallel PROM (Bytes) set to 4M and then select Add Storage Device Then Select the second Green Arrow

Leave the Checksum Fill Value => FF
File Nave => VFX70
File Location => XC5VFX70T_RevD/VFX70

File Format => MCS
Loading Direction => UP
Number of Data Streams => 1
Data Stream 0 Start Address => 0
Add Data Files => No

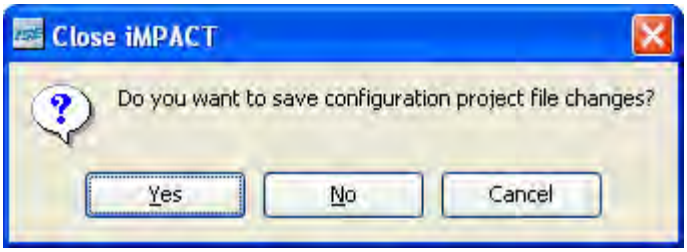
Select->OK

“Start adding device file to Data Stream 0” Select-> OK

“Add Device” -> Select XC5VFX70T.bit Select-> Open

“Would you like to add another design file to Data Stream 0?”
Select->No
Select->OK

Select->Operations->Generate File
The file VFX70.mcs is generated, and this file will next be downloaded over the PCI bus to configure the Virtex-5 device.
Exit without saving.



Select-> No

To configure and run the Virtex 5 PMC-VFX70 board it is recommended that the Acromag software drivers be used. The Acromag software will simplify download of the .mcs file and access of the system monitor registers for reading the temperature of the device. This is a great way to verify initial functionality of the board. For example, the windows application PCIVFXDemo.exe (available with the PCISW-API-WIN software package) can be used to download the new .mcs file and test read of the board temperature.

Further Reading

For further information on PowerPC programming please refer to the VFX Getting Started Guide (file PMC_VFX_Getting_Started_Guide_ISE11.1.pdf) in the Manuals & References folder in the VFX Engineering Design Kit.