



# XUPV5-LX110T MIG Design Creation

Using ISE™ 10.1i SP3, MIG 2.3 and  
ChipScope™ Pro 10.1i



September, 2008



# Overview

- Hardware Setup
- Software Requirements
- CORE Generator™ software
  - Memory Interface Generator (MIG)
- Modify Design
  - Add ChipScope Pro Cores to Design
- Compile and Test Memory Interface

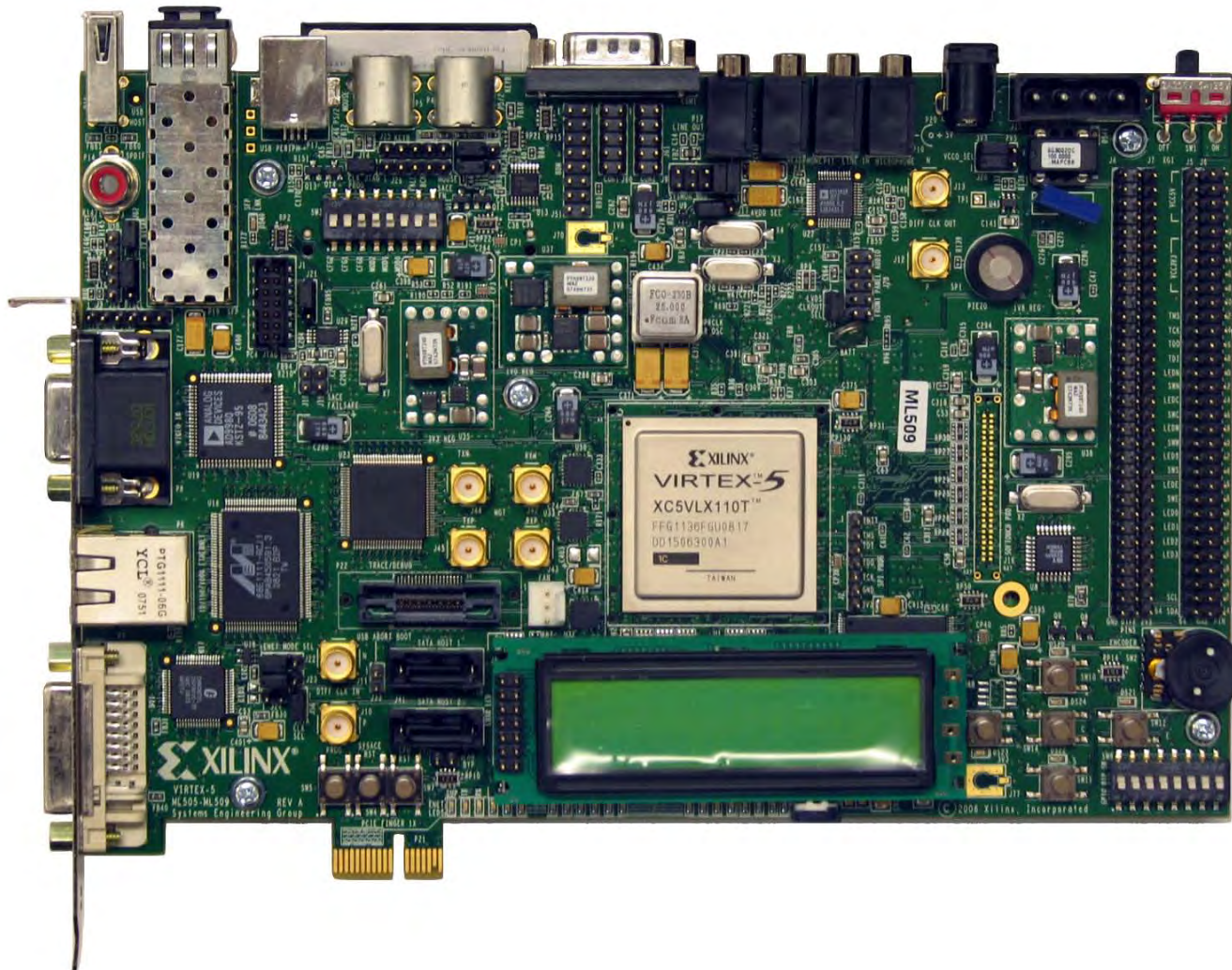


# Virtex-5 DDR2 Capabilities

- MIG DDR2 SDRAM design supports frequencies up to 333 MHz
  - The [MIG user guide](#) addresses MIG performance across device speed grades
- The XUPV5-LX110T ships with a –1 speed grade device
  - See the [Virtex-5 Data Sheet](#) for a list of Virtex-5 supported memory interface speeds



# Xilinx XUPV5-LX110T Board

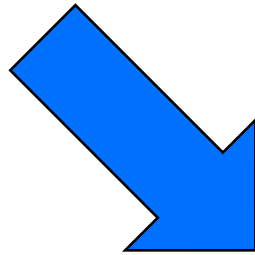


Note: The XUPV5-LX110T uses an XC5VLX110T FPGA.



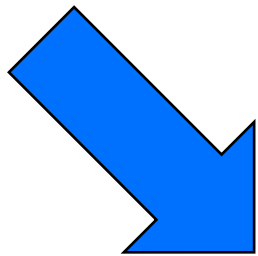
# ISE Software Requirement

- Xilinx ISE 10.1i SP3 software



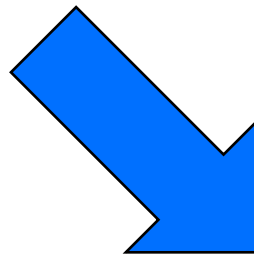
# Coregen Software Requirement

- Install Xilinx Coregen 10.1i IP Update 3



# ChipScope Software Requirement

- Xilinx ChipScope Pro 10.1i SP3



# Setting Up the Hardware

- Connect the Xilinx Parallel Cable IV (PC4)
  - [HW-PC4](#)
- SMA Cable
  - [www.flrst.com](http://www.flrst.com)
  - P/N: ASPI-024-ASPI-S402
- Optional - Pancake Fan
  - Recommended for keeping the Virtex™-5 device cool



# Setting Up the Hardware

- Connect SMA Cables from J12/13 to J10/11
  - This will serve as the 200 MHz clock source for the memory controller

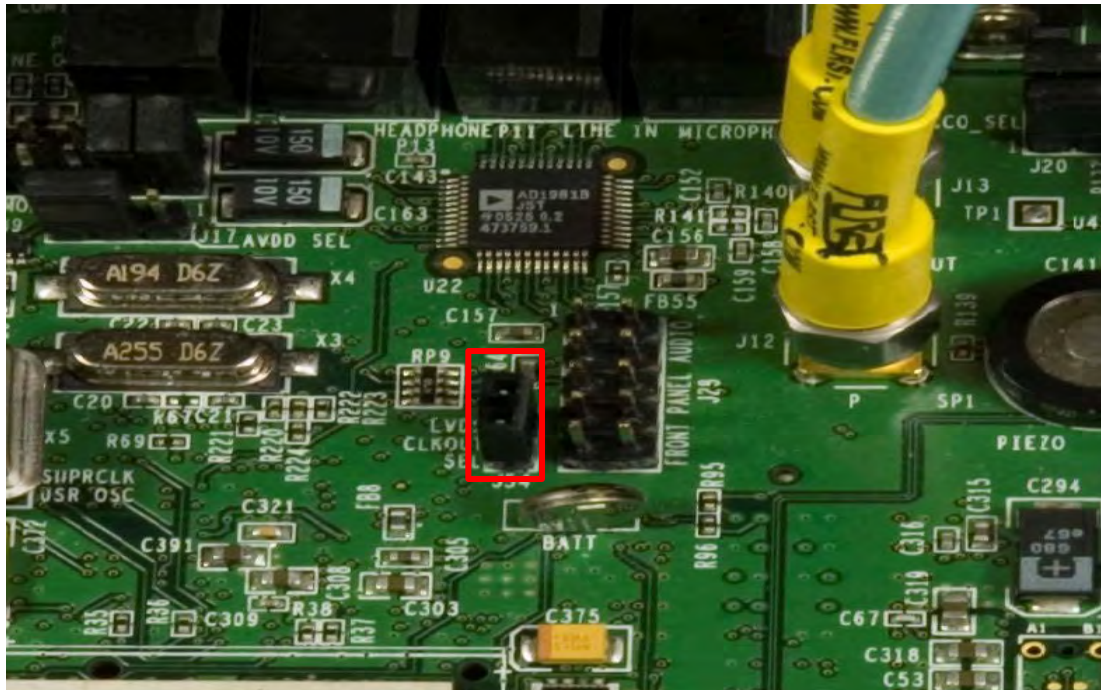


Note: Alternately, an external differential clock source can be connected to J10/11.



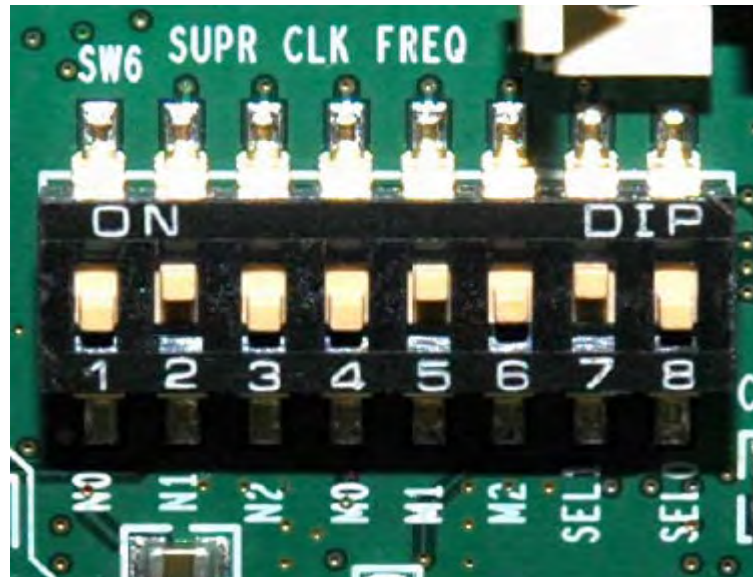
# Setting Up the Hardware

- Jumper J54 must be connected
  - This enables the output of the on-board ICS frequency source



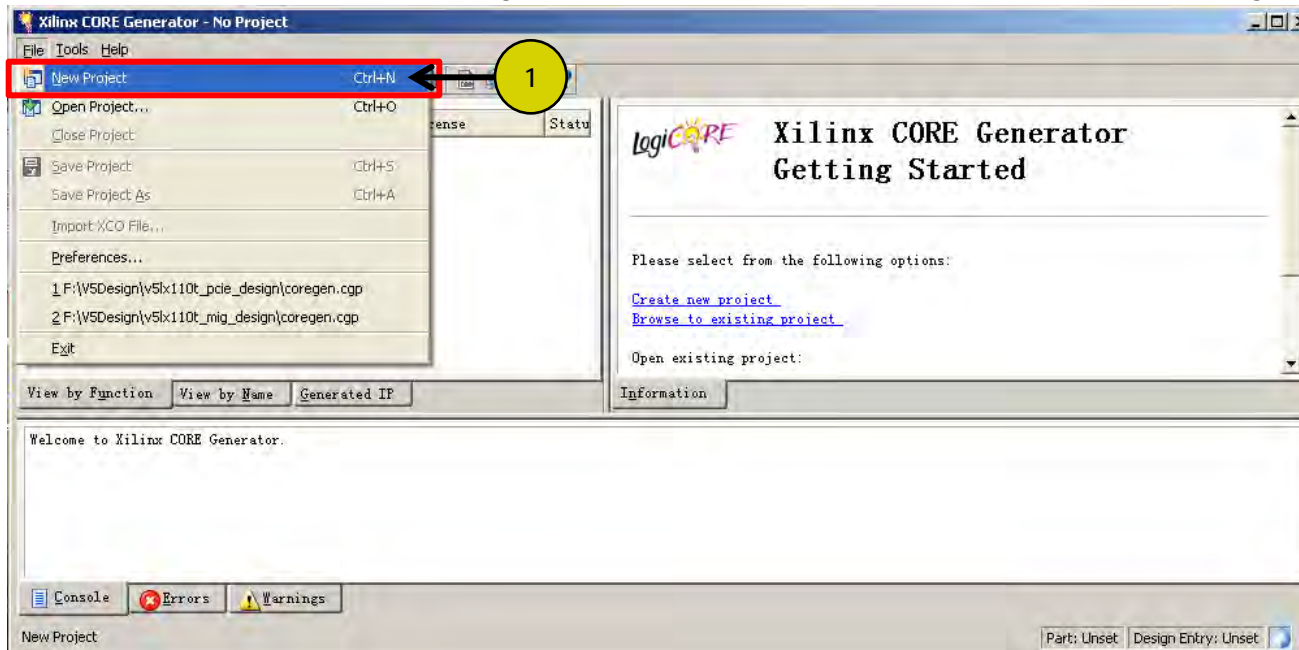
# Setting Up the Hardware

- Use SW6 to set the memory clock frequency
- Set SW6 to 200 MHz (1)
  - 200 MHz = 010 010 10 (Max memory clock frequency for a -1 speed grade)



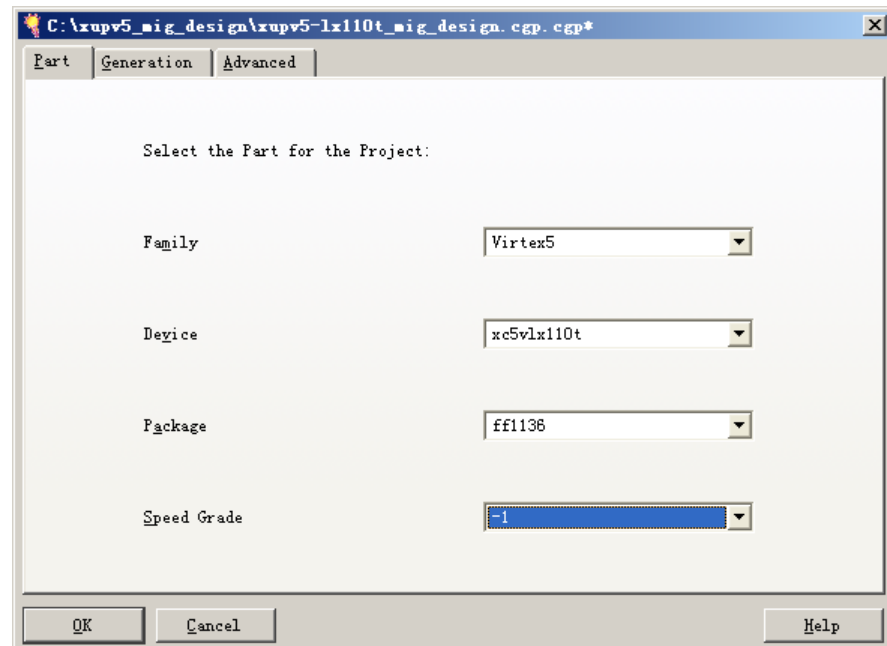
# CORE Generator

- Open the CORE Generator
  - Start → All Programs → Xilinx ISE Design Suite 10.1i → ISE → Accessories → CORE Generator
- Create a new project; select File → New Project (1)



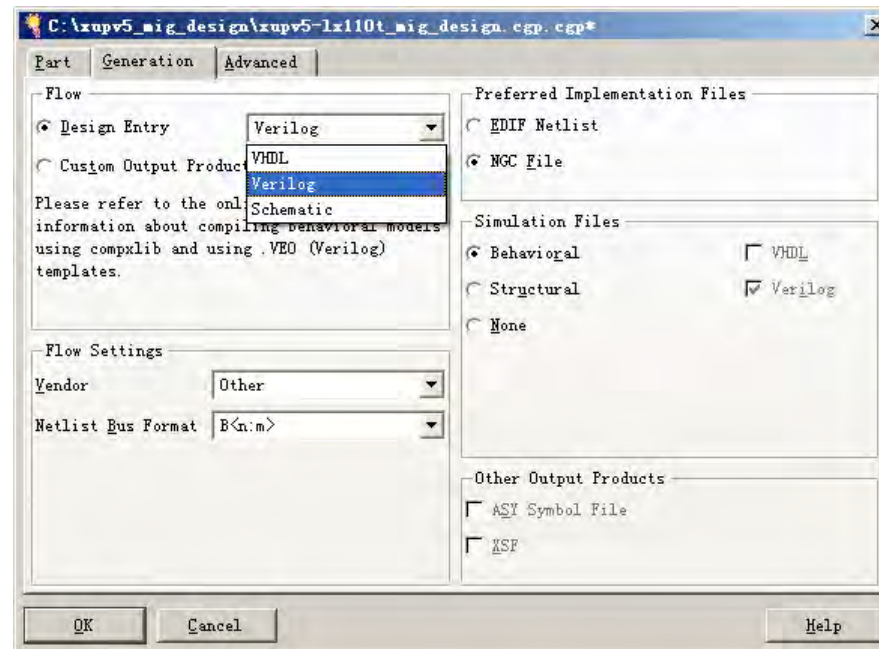
# MIG Core Generation

- Create a project directory: xupv5\_mig\_design
- Name the project: xupv5-lx110t\_mig\_design.cgp
- Set the Part (as shipped on the XUPV5-LX110T):
  - Family: Virtex5
  - Device: xc5vlx110t
  - Package: ff1136
  - Speed Grade: -1



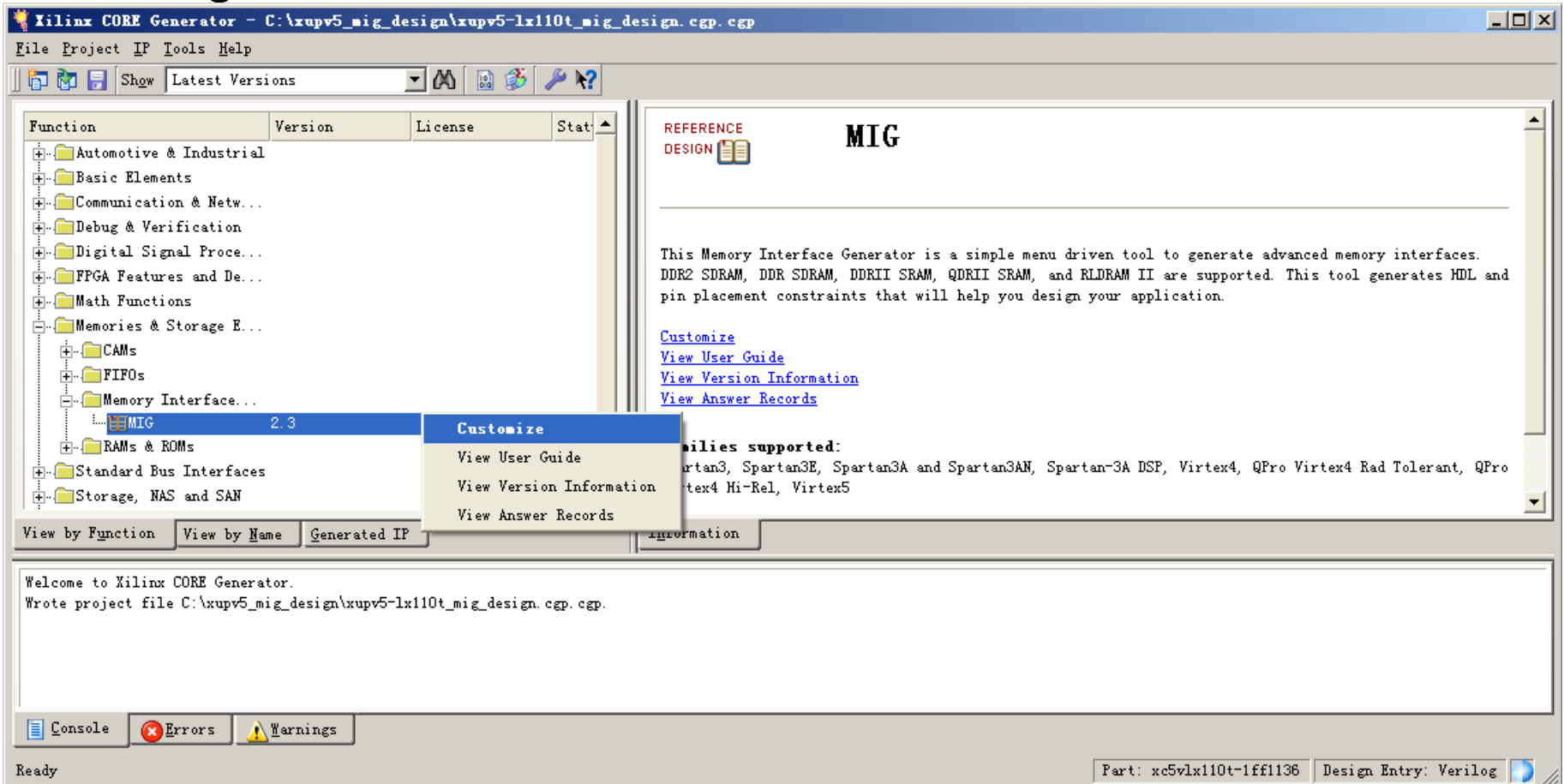
# MIG Core Generation

- Set the Design Entry to Verilog
- Click OK



# MIG Core Generation

- Right click on the MIG 2.3 and select Customize

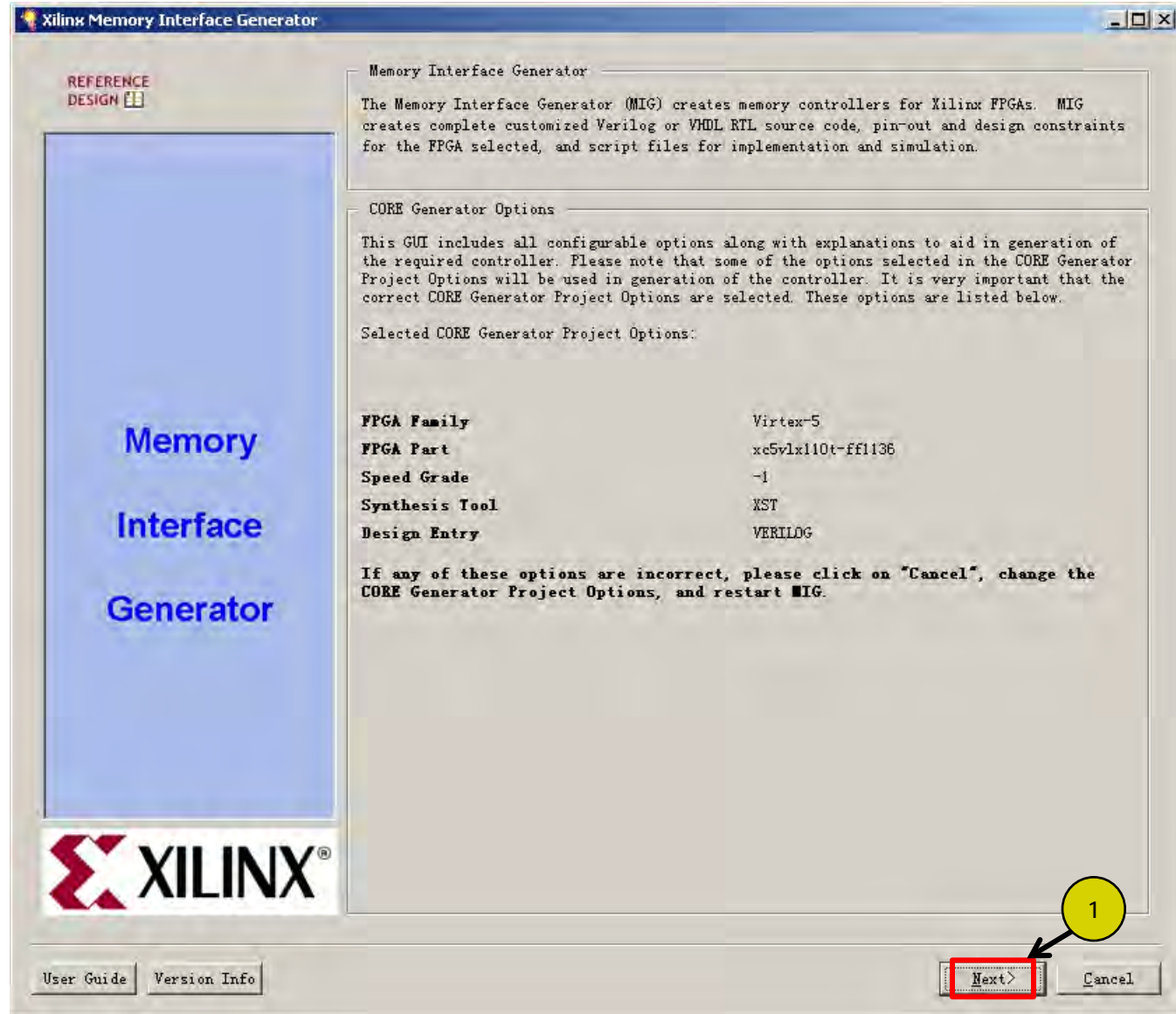


Note: See [Demos On Demand](#) for an online tutorial on the use of MIG.



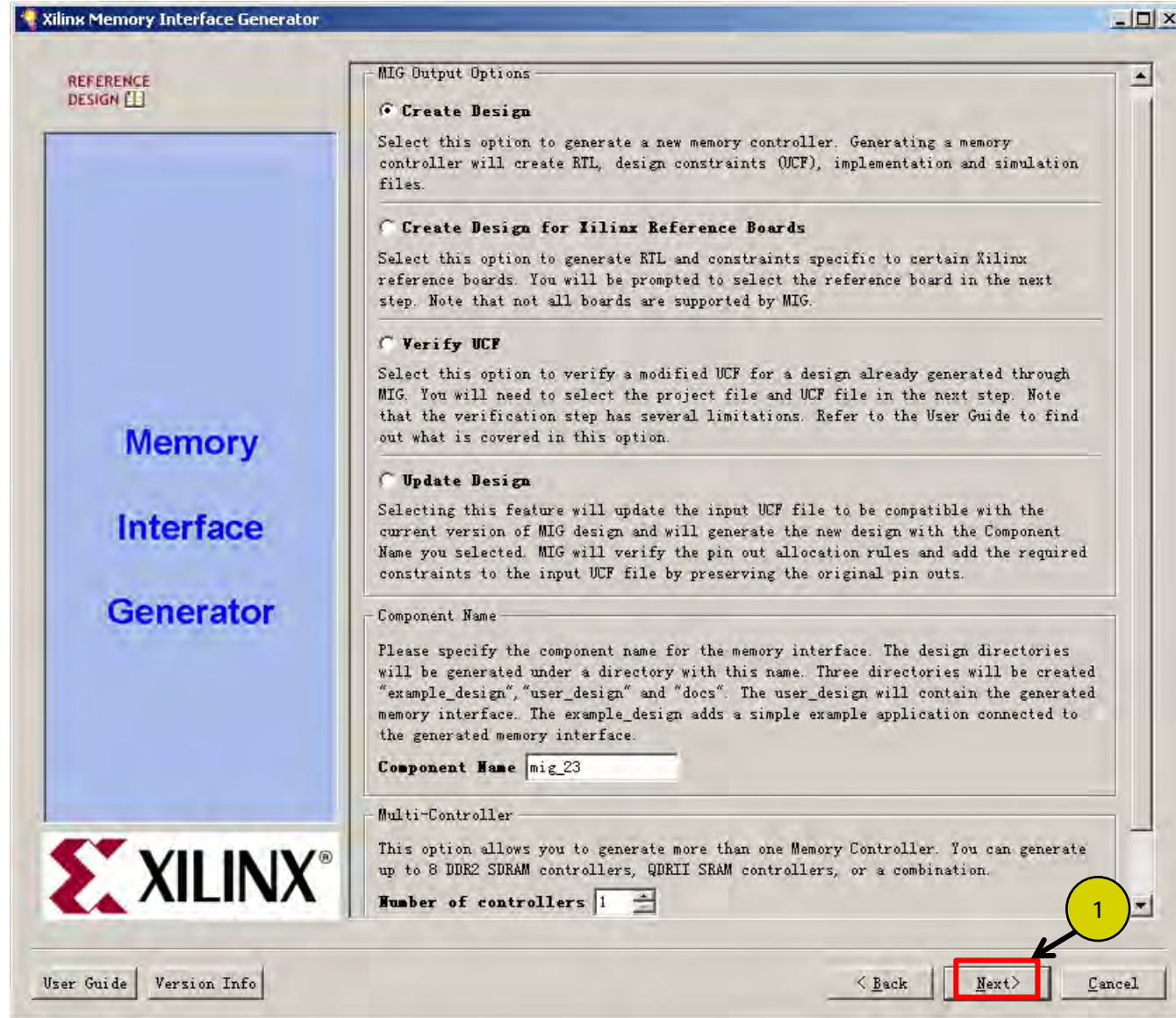
# MIG

- Click Next(1)



# MIG

- Click Next(1)



# MIG

- Click Next(1)

REFERENCE DESIGN

Pin Compatible FPGAs ✓

Memory Selection ✓

Controller Options ✓

Mode Register ✓

Extended Mode Register ✓

FPGA Options ✓

Reserve Pins ✓

Bank Selection

Summary

Memory Model License

PCB Information

Design Notes

Finish

XILINX®

Pin Compatible FPGAs

Pin Compatible FPGAs include all devices with the same package as the target device. Different FPGA devices with the same package do not have the same bonded pins. By selecting Pin Compatible FPGAs, MIG will only select pins that are common between the target device and all selected devices. Use the default UCF in the par folder for the target part. If you do not choose a Pin Compatible FPGA now and need to use a different FPGA later, the generated UCF may not work for the new device and a board spin may be required.

Target FPGA

Pin Compatible FPGAs

- virtex5
  - lx
    - xc5vlx50t-ff1136
    - xc5vlx85t-ff1136
    - xc5vlx155t-ff1136
  - fx
    - xc5vfx70t-ff1136
    - xc5vfx100t-ff1136
  - sx
    - xc5vsx50t-ff1136
    - xc5vsx95t-ff1136

PowerPC440 Based Memory Controller

To achieve maximum DDR2 performance, specific pin-outs may be required for memory controller designs that will be used with the PowerPC440 processor block in the FXT device. These pin outs are provided through MIG on checking the PPC440 check box. In dual processor devices the desired PowerPC block can be selected with the PowerPC440 Block Selection drop box. If the target and compatible devices do not have the same number of PowerPC440 blocks then the output pin out will be optimal for the single processor device. This is supported for single controller only.

PPC440

PowerPC440 Block Selection

< Back **Next >** Cancel

User Guide Version Info



# MIG

- Click Next(1)

**REFERENCE DESIGN**

**Pin Compatible FPGAs** ✓

**Memory Selection** ✓

**Controller Options** ✓

**Mode Register** ✓

**Extended Mode Register** ✓

**FPGA Options** ✓

**Reserve Pins** ✓

**Bank Selection**

**Summary**

**Memory Model License**

**PCB Information**

**Design Notes**

**Finish**

**XILINX**

Memory Selection

Select the type of controller. DDR2 PPC440 only supports a DDR2 SDRAM controller. If you require multiple controllers, your choices are limited to DDR2 SDRAM or QDRII SRAM for each controller. If you don't see the type of controller you require, it may not be supported yet. Please refer to the User Guide for a detailed list of supported controllers for each FPGA family.

Memory Selection for Controller0 **DDR2 SDRAM**

User Guide Version Info < Back **Next >** Cancel

1



# MIG

- MIG defaults to a maximum frequency of 266 for a Virtex-5 -1 speed grade selection
  - See UG086
- Set the Memory Type to SODIMMs
- Set the Memory part to MT4HTF3264HY-53E
- Click Next

**Options for Controller 0 - DDR2 SDRAM**

**Frequency**  
The allowed frequency range is a function of the selected FPGA part, FPGA speed grade, memory controller type, and clocking type. Note that the available memory parts will be limited based on this selection. The first controller of a particular type will determine the frequency for all subsequent controllers of that type.

**Memory Type**  
If DDR2 or DDR SDRAM is chosen, select either Component or DIMM. DDR2 PPC440 supports single-rank registered DIMMs and components. All other designs support Component only. Deep designs are supported only with dual rank DIMMs.

**Memory Part**  
Select the memory part for the interface. If the exact part that you will be using is not available here, you may be able to find an equivalent part. Alternately, you can create a part using the "Create Custom Part" selection at the bottom of this drop box. Refer to Userguide for

**Data Width**  
MIG supports multiples of 64 for components up to 128 bits. DDR2 PPC440 supports 16-bit, 32-bit, 64-bit and 72-bit data widths. Note that the selection is dependent upon the previously selected parameters.

**Memory Depth**  
DDR2 SDRAM designs can support depth expansion for greater density. Choose the depth of design for this configuration. Note that deep memories require additional signal integrity considerations. MIG does not limit the frequency range based on the loading. IBIS simulations are recommended to determine the maximum frequency of operation. This DDR2 SDRAM interface will calibrate

**Memory Details:** 256MB, x16, row:13, col:10, bank:2, unbuffered, data bits per strobe:8, with data mask, single rank



# MIG

- Click Next on this screen

**REFERENCE DESIGN**

- Pin Compatible FPGAs ✓
- Memory Selection ✓
- Controller Options ✓
- Mode Register** ✓
- Extended Mode Register ✓
- FPGA Options ✓
- Reserve Pins ✓
- Bank Selection
- Summary
- Memory Model License
- PCB Information
- Design Notes
- Finish

### Mode Register Data for Controller 0 - DDR2 SDRAM

Choose the Mode Register settings for the memory device. This register is not applicable for QDRII, DDRII SRAM memory types. This GUI restricts changes to those supported by the controller. For more information consult the memory vendor's data sheet. Note that CAS Latency is dependent on the frequency specified in the prior Controller Options page.

Burst Length	4 (010)
Burst Type	sequential (0)
CAS Latency	4 (100)
Mode	normal (0)
DLL Reset	no (0)
Write Recovery	4 (011)
PD Mode	fast exit (0)

Mode Register: 0\_0110\_0100\_0010

User Guide | Version Info | < Back | Next > | Cancel



# MIG

- Click Next on this screen

**REFERENCE DESIGN**

- Pin Compatible FPGAs ✓
- Memory Selection ✓
- Controller Options ✓
- Mode Register ✓
- Extended Mode Register ✓**
- FPGA Options ✓
- Reserve Pins ✓
- Bank Selection
- Summary
- Memory Model License
- PCB Information
- Design Notes
- Finish

### Extended Mode Register Data for Controller 0 - DDR2 SDRAM

Choose the Extended Mode Register settings for the memory device. This register is not applicable for QDR1I, DDR1I SRAM, RLD1RAM1I memory types. Settings are restricted to those supported by the controller. For more information consult the memory vendor's data sheet.

**DLL Enable**

**Output Drive Strength**

The Virtex-5 DDR2 interface requires that if parallel termination is used at the memory I/Os, then it must be ODT rather than external termination resistor(s). This is a requirement of the read capture scheme in use.

**RTT (nominal) - ODT**

**Additive Latency (AL)**

**OCD Operation**

**DQS# Enable**

**RDQS Enable**

**Outputs**

Extended Mode Register

User Guide | Version Info | < Back | Next > | Cancel

# MIG

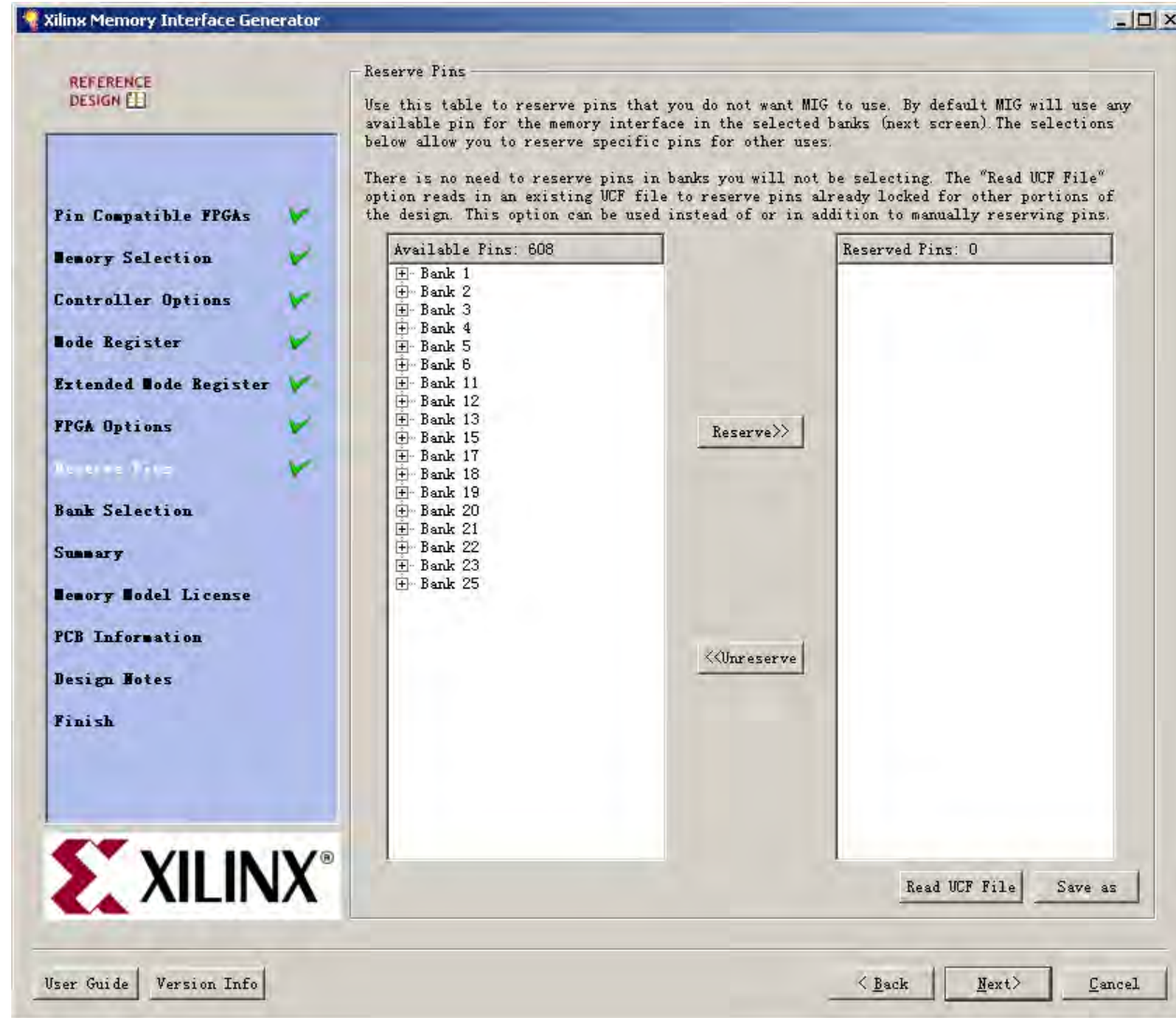
•Click Next on this screen

<b>DCM Option</b>
If the design is generated with the DCM option enabled, the required clocks are generated from a DCM internal to the design. If this option is disabled, required clocks must be provided as inputs with the appropriate phase relationships. Disabling this option can result in lower DCM usage if the appropriate clock signals are already available in the overall design.
<b>Use DCM</b> <input checked="" type="checkbox"/>
<b>Digitally Controlled Impedance (DCI)</b>
Select the DCI (Digitally Controlled Impedance) I/O standard to enable the FPGA internal termination for the DQ and DQS signals. Consult the User Guide for more information and use IBIS simulation to determine the best termination strategy. This can also be changed after generation by altering the UCF manually.
<b>DCI for DQ/DQS</b> <input checked="" type="checkbox"/>
Select the DCI (Digitally Controlled Impedance) I/O standard to activate the internal 25 ohm series termination in the SSTL driver for address and control output signals to the memory. Consult the User Guide for more information and use IBIS simulation to determine the best termination strategy. This can also be changed after generation by altering the UCF manually.
<b>DCI for Address/Control</b> <input type="checkbox"/>
<b>SSTL Class Option</b>
Class II is recommended for all SSTL signals in memory interfaces. However, better signal integrity may sometimes be achieved with Class I for Address & Control. If IBIS simulations indicate that Class I is superior for your application, select Class I below. This can be changed after generation by modifying the UCF. This option changes the drive strength for Address & Control. The drive strength remains Class II for data.
<b>Class for Address and Control</b> <input type="text" value="Class II"/>
<b>Debug Signals Control</b>
This allows the debug signals (calibration status signals) to be monitored on the ChipScope tool. Selecting this option will port map the debug signals to the ChipScope modules in the design top module. The debug signals width is calibrated based upon the selected design data width. If the design data width is greater than or equal to 32 bits then the debug signals width is calibrated only for 32-bit data.
<b>Debug Signals for Memory Controller</b> <input type="text" value="Disable"/>
<b>Limit to 2 Bytes per Bank</b>
DDR2 SDRAM DQ signals transition simultaneously. Concentrating many of them in an I/O bank will increase the amount of simultaneous switching noise the I/O bank will experience. Choose this option to spread the DQ bytes across more I/O banks. Check the timing of the resultant design before committing to a PCB layout.
<b>Limit to 2 Bytes per Bank</b> <input type="checkbox"/>
<b>System Clock</b>
Choose the desired input clock configuration.
<b>System Clock</b> <input type="text" value="Differential"/>
<b>IODELAY Power Versus Performance</b>
Choose <b>High</b> Performance Mode for lowest IODELAY jitter and maximum interface performance. Choose <b>Normal</b> Performance Mode to reduce power by approximately 2.8 mW per pin when interface performance requirements are less stringent. The timing spreadsheet generated by MIG indicates the change.
<b>Performance Mode</b> <input type="text" value="HIGH"/>



# MIG

- Click Next on this screen



# MIG

- Click Next on this screen

**Bank Selection For Controller 0 - DDR2 SDRAM**

The FPGA bank diagram below is an architectural view physical representation of the selected part. Choose the banks you would like to use for your memory interface. You do not get to select the actual pins. By default MIG will use any available pins for the memory interface in the selected banks. If you have custom pin-out use Update Design option in the MIG Output Options page to generate the design and updated UCF file. Generally banks in a single contiguous column provide the best overall internal timing. Predefined fixed bank selection is used for PPC440 depending on the selected device. Enter the WASSO (Weighted Average Simultaneous Switching Outputs) limits for your PCB as described in the selected FPGA User Guide. MIG will limit the number of pins per bank based on these WASSO limits.

For each signal grouping, the numbers indicate the number of pins currently allocated and the total needed. Priority is given for Data, then Address, then System Control in each bank.

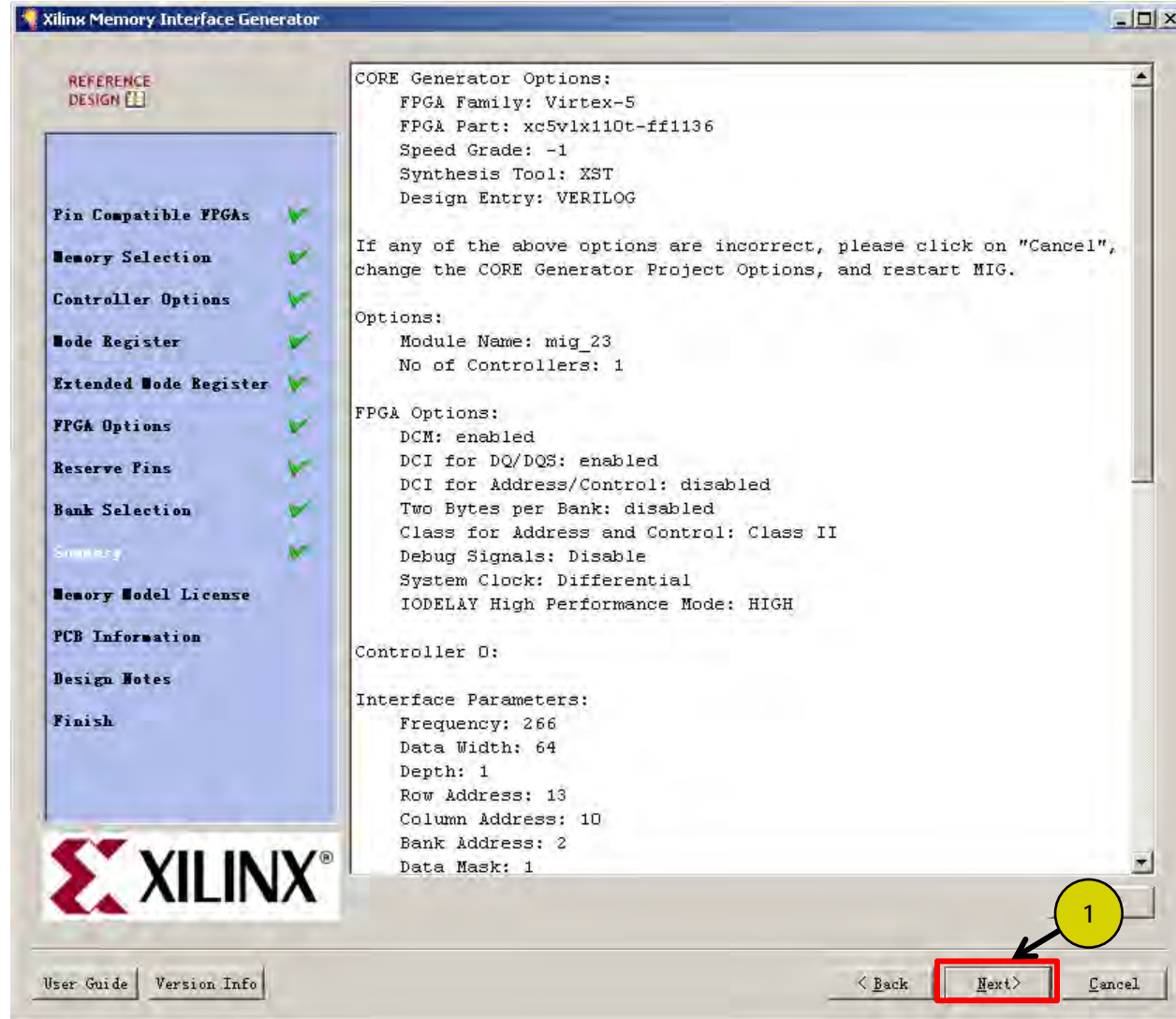
Address: 25/21 ✓ Data: 88/8 ✓ System Control: 3, ✓ System Clock: 4/4 ✓

Bank	Available IO's	WASSO Limit	Address	Data	System Control	System Clock
Bank:23	2	38	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Bank:5	38	38	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Bank:19	2	38	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
Bank:3	15	19				<input checked="" type="checkbox"/>
Bank:20	38	38	<input type="checkbox"/>	<input type="checkbox"/>		



# MIG

- Click Next(1)



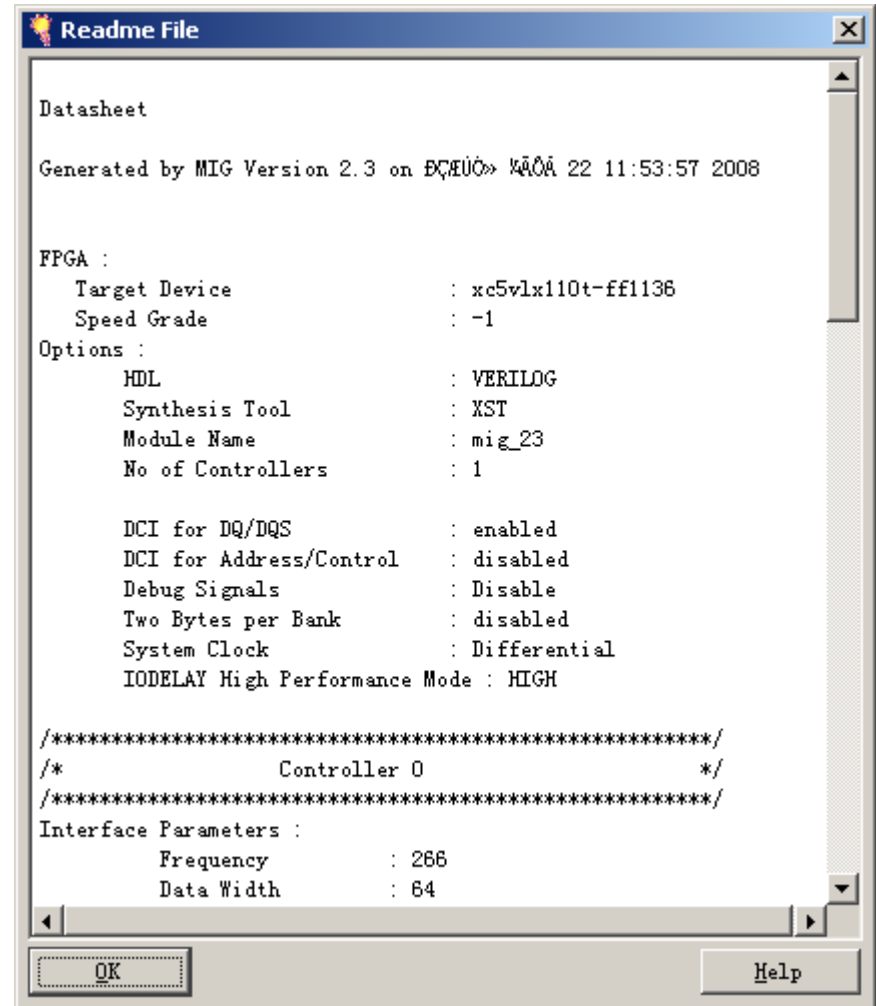
# MIG

- Click Generate (1)
- Click Finish and select YES after generation



# MIG

- After the MIG core finishes generating, click OK on the Datasheet window



The screenshot shows a 'Readme File' window with the following content:

```
Datasheet

Generated by MIG Version 2.3 on 2008-11-22 11:53:57

FPGA :
  Target Device      : xc5vlx110t-ff1136
  Speed Grade       : -1
Options :
  HDL                : VERILOG
  Synthesis Tool     : XST
  Module Name        : mig_23
  No of Controllers  : 1

  DCI for DQ/DQS     : enabled
  DCI for Address/Control : disabled
  Debug Signals      : Disable
  Two Bytes per Bank : disabled
  System Clock       : Differential
  IODELAY High Performance Mode : HIGH

/*****/
/*          Controller 0          */
/*****/

Interface Parameters :
  Frequency          : 266
  Data Width         : 64
```

At the bottom of the window, there are 'OK' and 'Help' buttons.

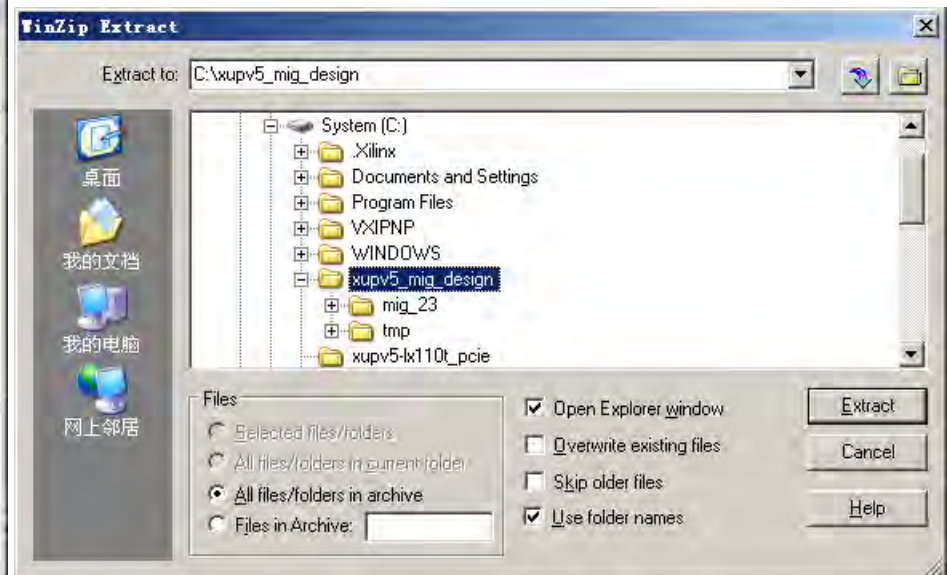
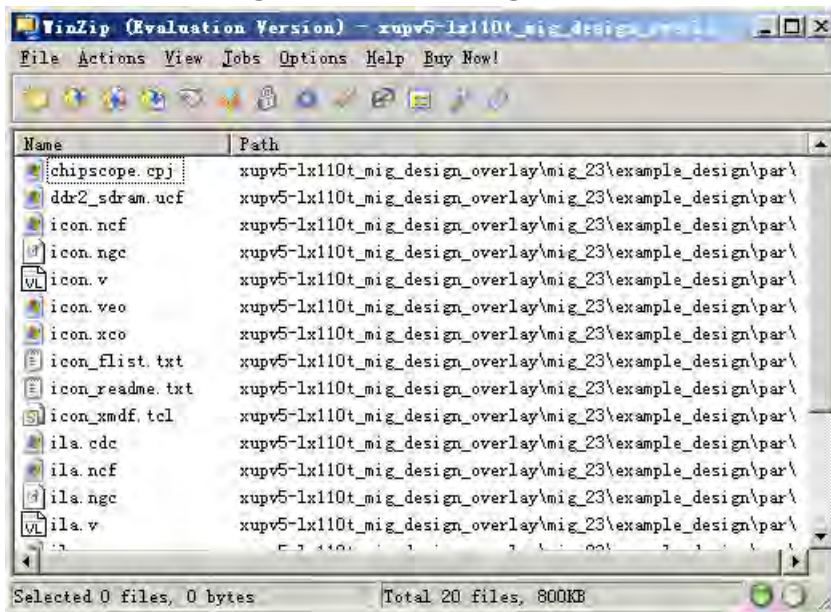
# Design Modifications

- Add overlay files
  - XUPV5-LX110T specific UCF file as per [Answer Record 29313](#)
  - Pre-compiled ChipScope Pro design files used to validate the design
- Modify top level Verilog file
  - Support for XUPV5-LX110T Board as per [AR 29313](#)
  - Add ChipScope Pro to design



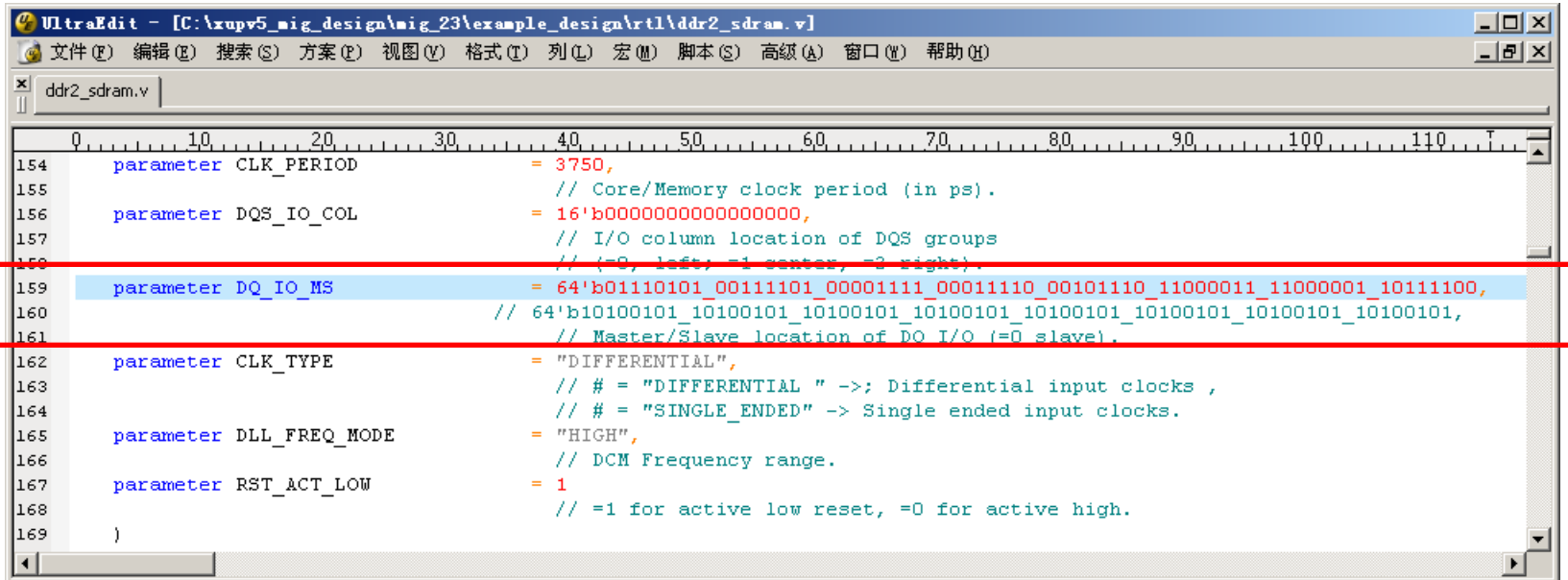
# Add XUPV5-LX110T Files

- Unzip [xupv5-lx110t mig design overlay.zip](#) file
  - Unzip to the xupv5\_mig\_design directory
  - See [ChipScope Pro documentation](#) for details on generating/instantiating the ICON and ILA cores



# Modify Design for XUPV5-LX110T

- Open the top-level Verilog file, mig\_23\example\_design\rtl\ddr2\_sdram.v
  - Change the DQ\_IO\_MS parameter:  
64'b10100101\_10100101\_10100101\_10100101\_10100101\_10100101\_10100101\_10100101 To:  
64'b01110101\_00111101\_00001111\_00011110\_00101110\_11000011\_11000001\_10111100



The screenshot shows the UltraEdit text editor with the file 'ddr2\_sdram.v' open. The code is as follows:

```
154 parameter CLK_PERIOD          = 3750,  
155       // Core/Memory clock period (in ps).  
156 parameter DQS_IO_COL        = 16'b0000000000000000,  
157       // I/O column location of DQS groups  
158       // (-8, left; -1 center; +3 right).  
159 parameter DQ_IO_MS          = 64'b01110101_00111101_00001111_00011110_00101110_11000011_11000001_10111100,  
160       // 64'b10100101_10100101_10100101_10100101_10100101_10100101_10100101_10100101,  
161       // Master/Slave location of DQ I/O (=0 slave).  
162 parameter CLK_TYPE          = "DIFFERENTIAL",  
163       // # = "DIFFERENTIAL" -> Differential input clocks ,  
164       // # = "SINGLE_ENDED" -> Single ended input clocks.  
165 parameter DLL_FREQ_MODE     = "HIGH",  
166       // DCM Frequency range.  
167 parameter RST_ACT_LOW      = 1  
168       // =1 for active low reset, =0 for active high.  
169 )
```

The line for the `DQ_IO_MS` parameter (lines 159-161) is highlighted in blue and enclosed in a red rectangular box, indicating the modification.

# Add ChipScope Pro

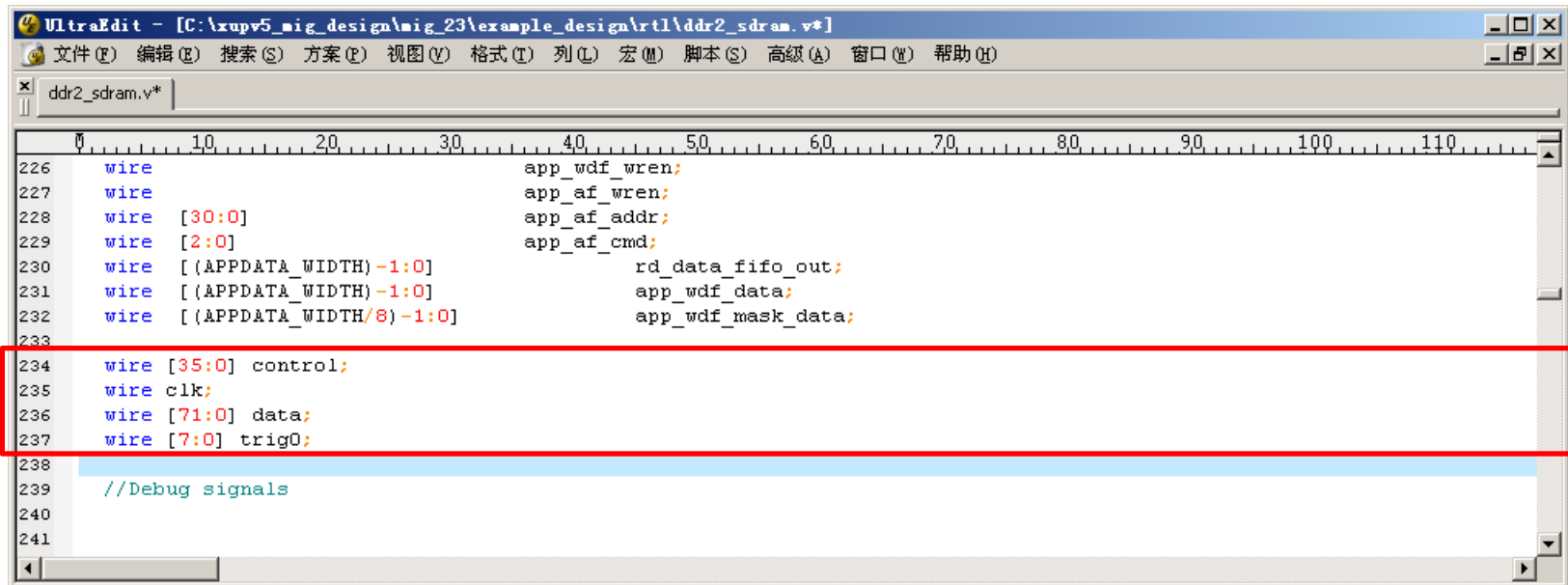
- Add these lines to the top-level Verilog file:

```
wire [35:0] control;
```

```
wire clk;
```

```
wire [71:0] data;
```

```
wire [7:0] trig0;
```



The screenshot shows the UltraEdit text editor with a Verilog file named 'ddr2\_sdram.v\*' open. The code is as follows:

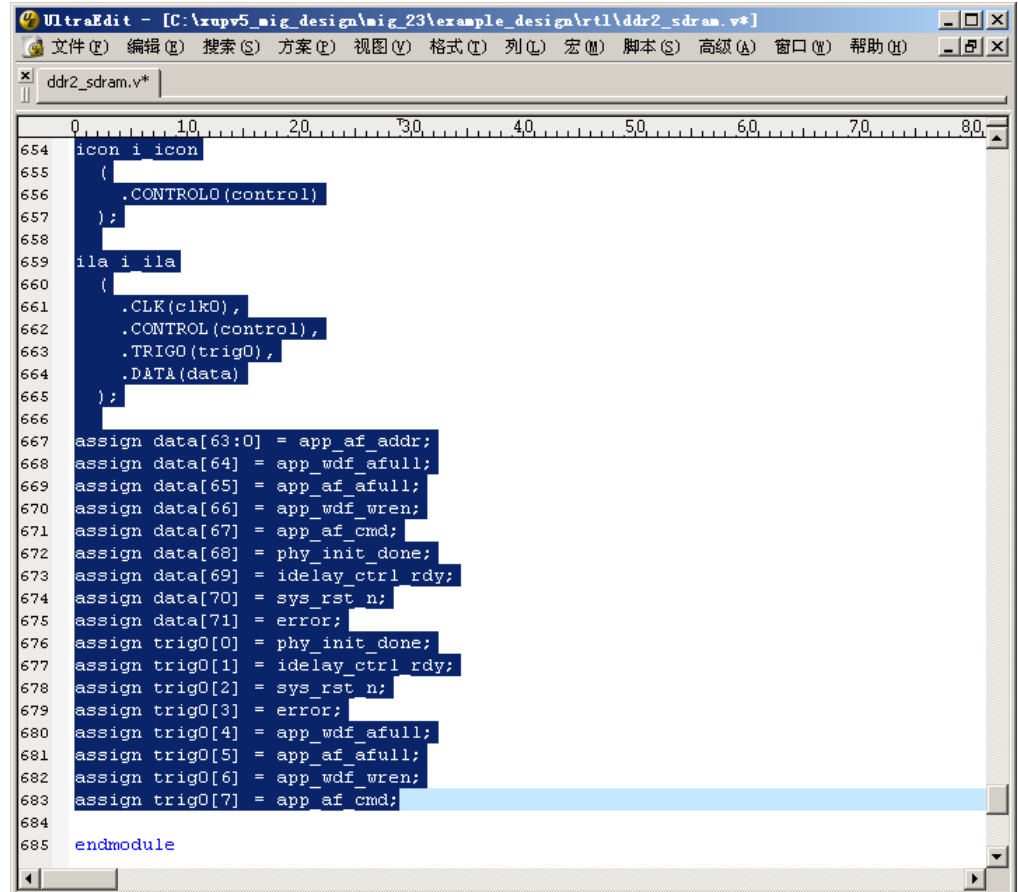
```
226 wire          app_wdf_wren;
227 wire          app_af_wren;
228 wire [30:0]    app_af_addr;
229 wire [2:0]     app_af_cmd;
230 wire [ (APPDATA_WIDTH)-1:0] rd_data_fifo_out;
231 wire [ (APPDATA_WIDTH)-1:0] app_wdf_data;
232 wire [ (APPDATA_WIDTH/8)-1:0] app_wdf_mask_data;
233
234 wire [35:0] control;
235 wire clk;
236 wire [71:0] data;
237 wire [7:0] trig0;
238
239 //Debug signals
240
241
```

A red rectangular box highlights lines 234 through 237, which correspond to the Verilog code provided in the previous blocks.

# Add ChipScope Pro

- Add these lines to the top-level Verilog file before endmodule:

```
icon i_icon(.CONTROL0(control));
ila i_ila(.CLK(clk0),.CONTROL(control),
        .TRIG0(trig0),.DATA(data));
assign data[63:0] = app_af_addr;
assign data[64] = app_wdf_afull;
assign data[65] = app_af_afull;
assign data[66] = app_wdf_wren;
assign data[67] = app_af_cmd;
assign data[68] = phy_init_done;
assign data[69] = idelay_ctrl_rdy;
assign data[70] = sys_rst_n;
assign data[71] = error;
assign trig0[0] = phy_init_done;
assign trig0[1] = idelay_ctrl_rdy;
assign trig0[2] = sys_rst_n;
assign trig0[3] = error;
assign trig0[4] = app_wdf_afull;
assign trig0[5] = app_af_afull;
assign trig0[6] = app_wdf_wren;
assign trig0[7] = app_af_cmd;
```

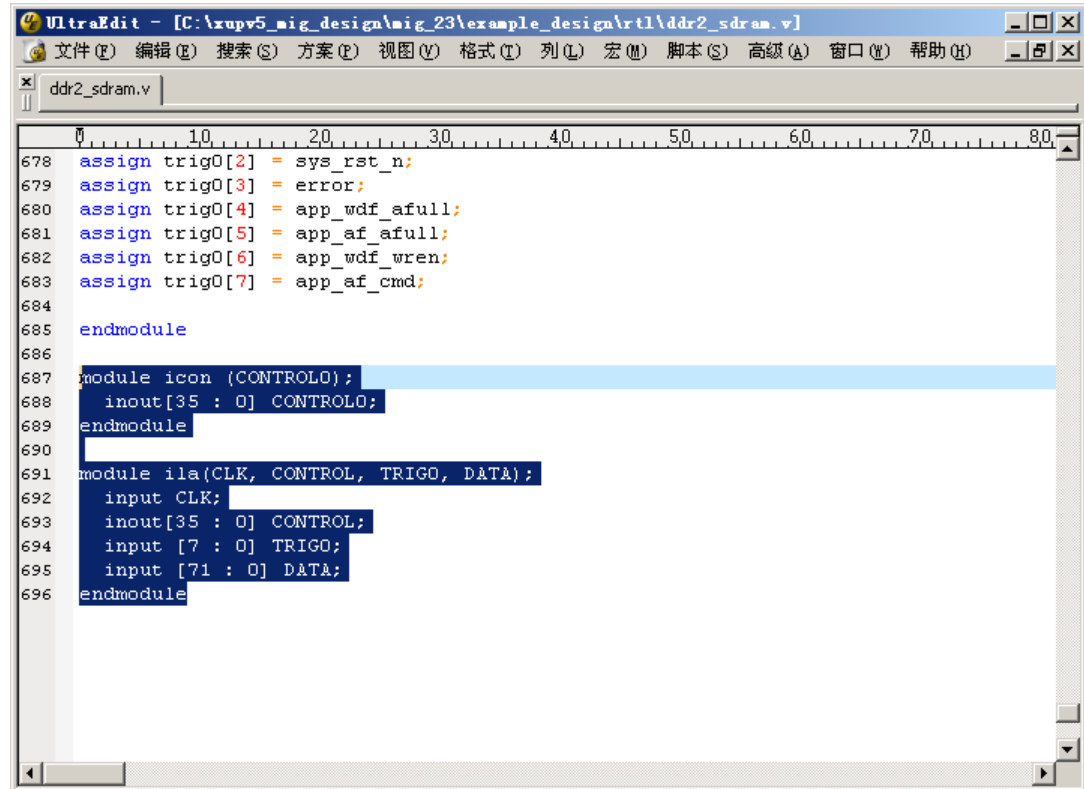


```
UltraEdit - [C:\xup5_mig_design\mig_23\example_design\rtl\ddr2_sdram.v*]
文件(F) 编辑(E) 搜索(S) 方案(O) 视图(V) 格式(T) 列(L) 宏(M) 脚本(S) 高级(A) 窗口(W) 帮助(H)
ddr2_sdram.v*
0 10 20 30 40 50 60 70 80
654 icon i_icon
655 (
656     .CONTROL0(control)
657 );
658
659 ila i_ila
660 (
661     .CLK(clk0),
662     .CONTROL(control),
663     .TRIG0(trig0),
664     .DATA(data)
665 );
666
667 assign data[63:0] = app_af_addr;
668 assign data[64] = app_wdf_afull;
669 assign data[65] = app_af_afull;
670 assign data[66] = app_wdf_wren;
671 assign data[67] = app_af_cmd;
672 assign data[68] = phy_init_done;
673 assign data[69] = idelay_ctrl_rdy;
674 assign data[70] = sys_rst_n;
675 assign data[71] = error;
676 assign trig0[0] = phy_init_done;
677 assign trig0[1] = idelay_ctrl_rdy;
678 assign trig0[2] = sys_rst_n;
679 assign trig0[3] = error;
680 assign trig0[4] = app_wdf_afull;
681 assign trig0[5] = app_af_afull;
682 assign trig0[6] = app_wdf_wren;
683 assign trig0[7] = app_af_cmd;
684
685 endmodule
```

# Add ChipScope Pro

- Add these lines to the top-level Verilog file after endmodule:

```
module icon (CONTROL0);  
inout[35 : 0] CONTROL0;  
endmodule  
  
module ila(CLK, CONTROL,  
          TRIG0, DATA);  
  
input CLK;  
inout[35 : 0] CONTROL;  
input [7 : 0] TRIG0;  
input [71 : 0] DATA;  
endmodule
```

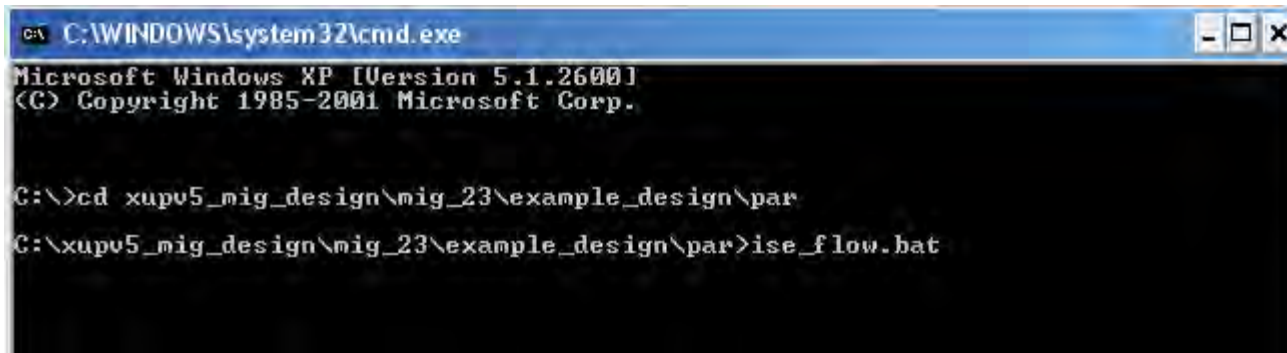


```
UltraEdit - [C:\xupv5_mig_design\mig_23\example_design\rtl\ddr2_sdram.v]  
文件(F) 编辑(E) 搜索(S) 方案(O) 视图(V) 格式(T) 列(L) 宏(M) 脚本(S) 高级(A) 窗口(W) 帮助(H)  
ddr2_sdram.v  
0 10 20 30 40 50 60 70 80  
678 assign trig0[2] = sys_rst_n;  
679 assign trig0[3] = error;  
680 assign trig0[4] = app_wdf_afull;  
681 assign trig0[5] = app_af_afull;  
682 assign trig0[6] = app_wdf_wren;  
683 assign trig0[7] = app_af_cmd;  
684  
685 endmodule  
686  
687 module icon (CONTROL0);  
688     inout[35 : 0] CONTROL0;  
689 endmodule  
690  
691 module ila(CLK, CONTROL, TRIG0, DATA);  
692     input CLK;  
693     inout[35 : 0] CONTROL;  
694     input [7 : 0] TRIG0;  
695     input [71 : 0] DATA;  
696 endmodule
```

# Generate Bitstream

- Start a windows command shell and enter these commands:

```
cd xupv5_mig_design\mig_23\example_design\par  
ise_flow.bat
```



```
C:\WINDOWS\system32\cmd.exe  
Microsoft Windows XP [Version 5.1.2600]  
(C) Copyright 1985-2001 Microsoft Corp.  
  
C:\>cd xupv5_mig_design\mig_23\example_design\par  
C:\xupv5_mig_design\mig_23\example_design\par>ise_flow.bat
```



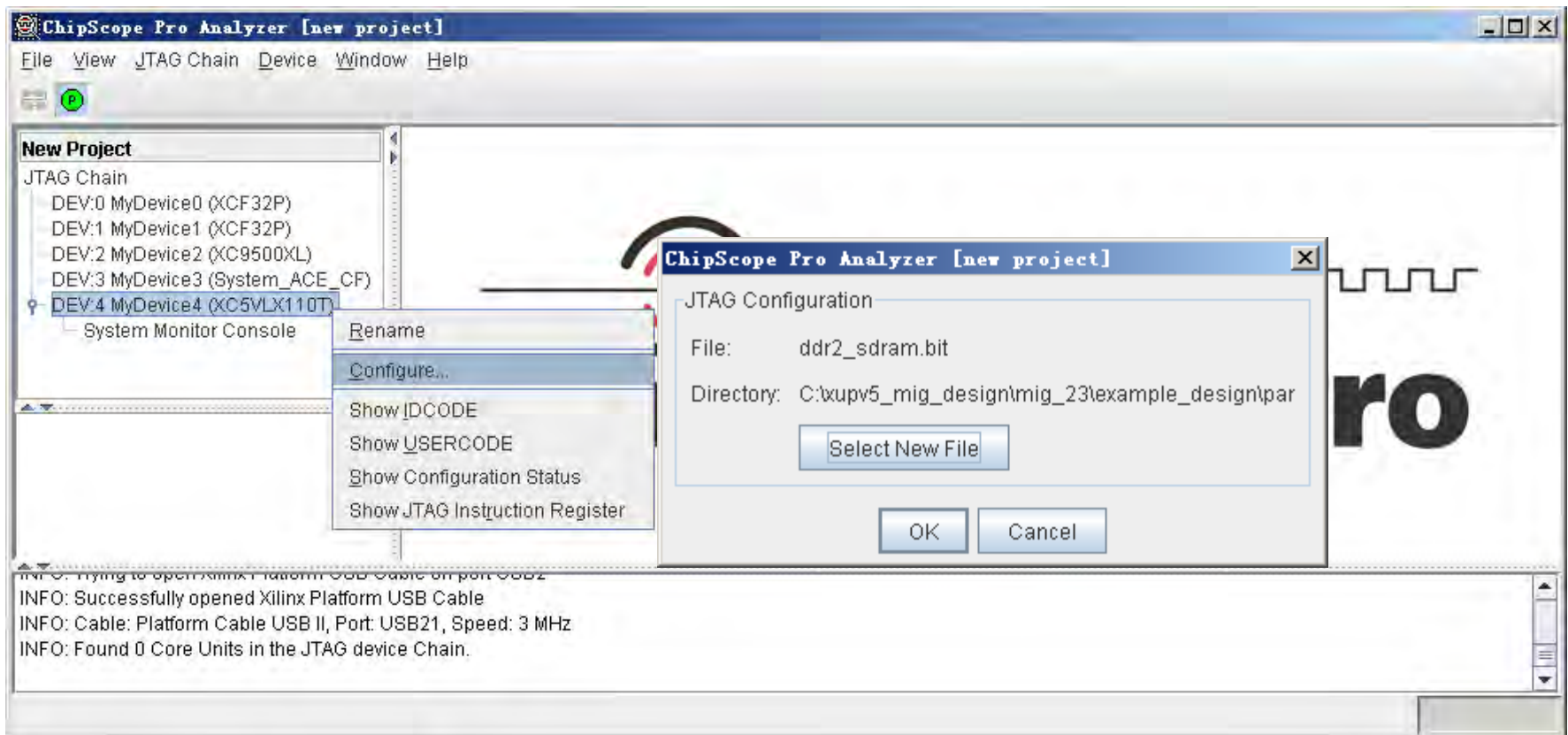
# ChipScope Pro Setup

- After the design compiles, open ChipScope Pro Analyzer
- Click on the Open Cable Button (1)



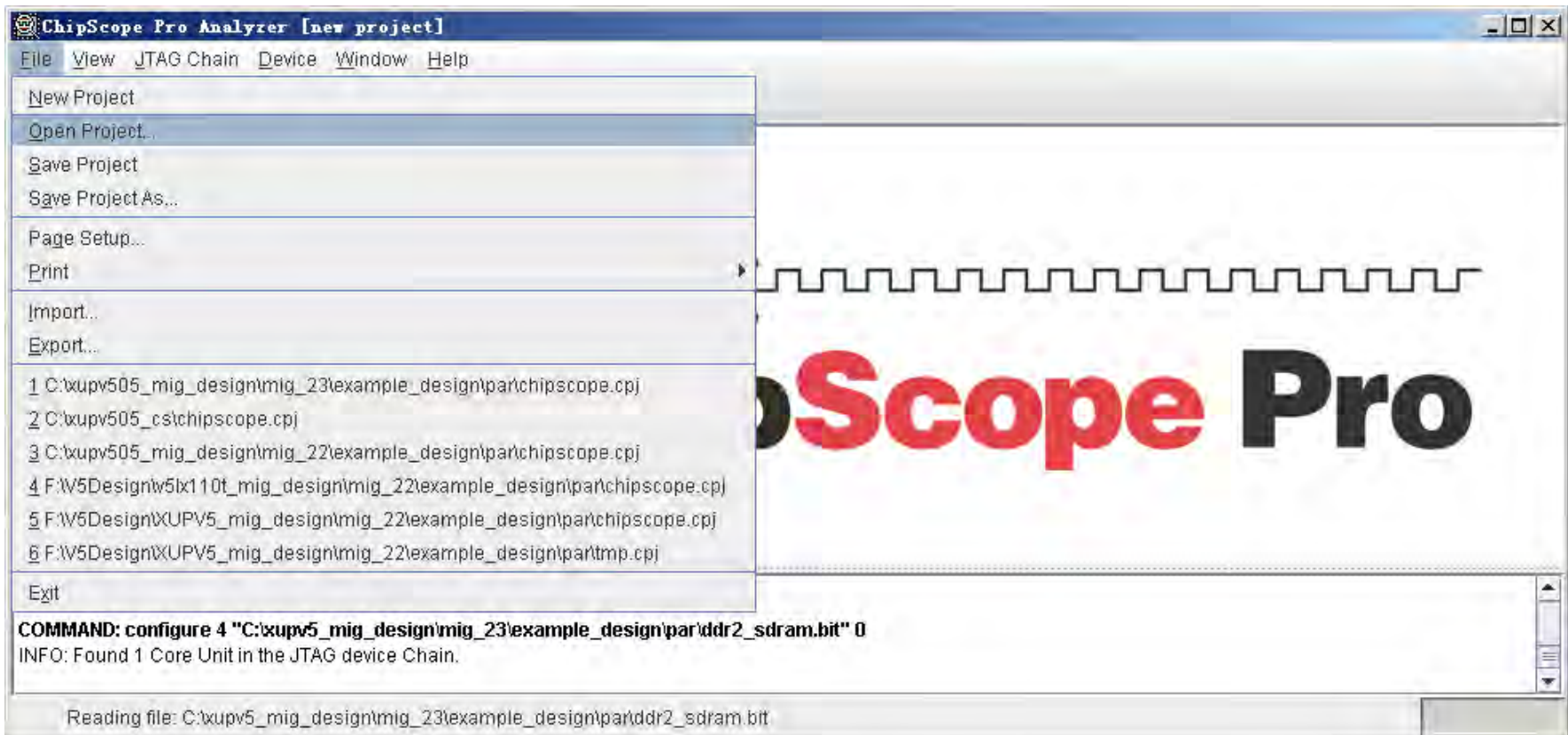
# ChipScope Pro Setup

- Select Device →DEV:4 MyDevice4 (XC5VLX110T) →Configure...
- Select <Design Path>\mig\_23\example\_design\par\ddr2\_sdram.bit



# ChipScope Pro Setup

- Select File → Open Project...
- Select <Design Path>\mig\_23\example\_design\par\chipscope.cpj



# ChipScope Pro Setup

- Click on Trigger Setup to view trigger settings
- The error bit value should be set to 1

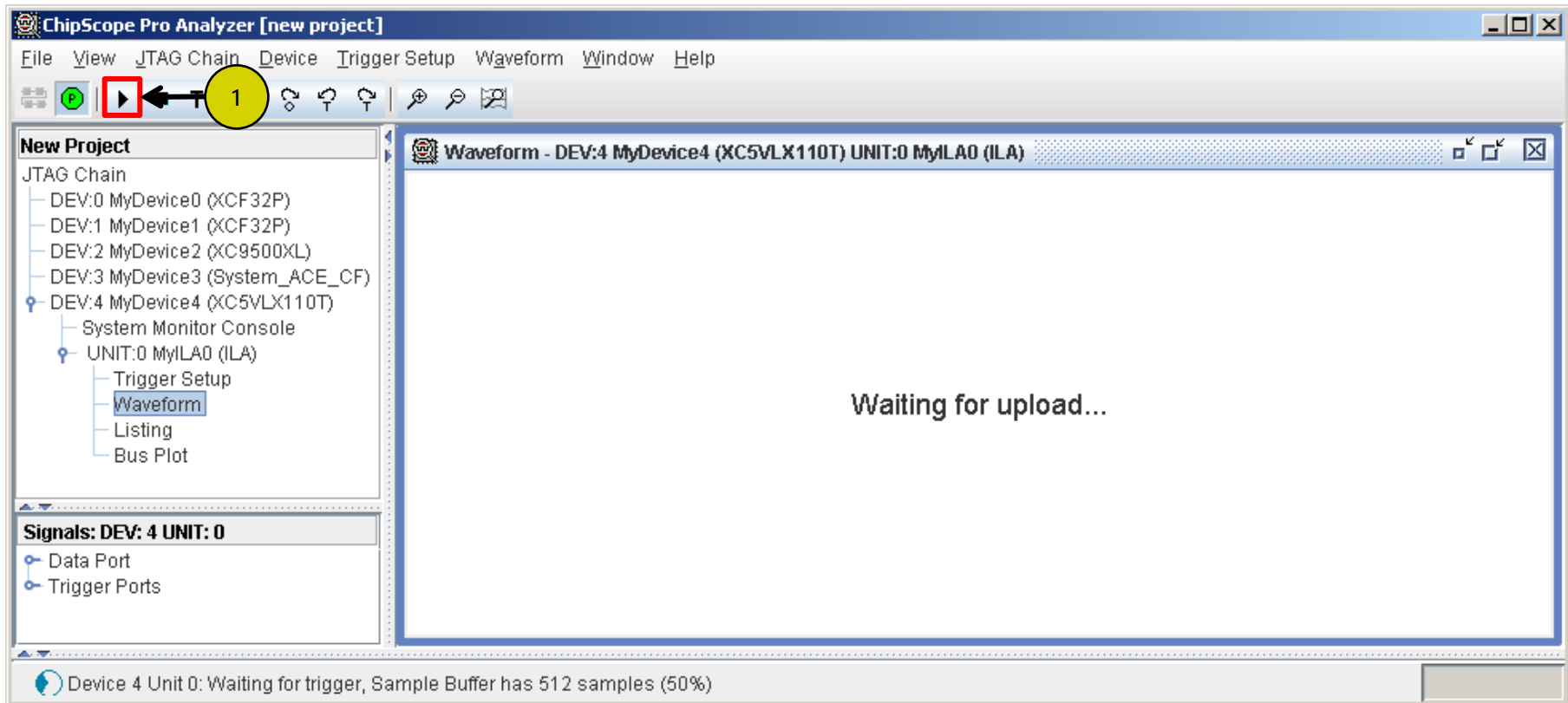
The screenshot shows the ChipScope Pro Analyzer interface. The main window is titled "Trigger Setup - DEV:4 MyDevice4 (XC5VLX110T) UNIT:0 MyILA0 (ILA)". The interface is divided into several sections:

- New Project:** A tree view showing the JTAG Chain with devices DEV:0 to DEV:4. The selected device is DEV:4 MyDevice4 (XC5VLX110T), and the selected unit is UNIT:0 MyILA0 (ILA). The "Trigger Setup" option is highlighted.
- Signals: DEV: 4 UNIT: 0:** A list of signals including "Data Port" and "Trigger Ports".
- Match Table:** A table with columns: Match Unit, Function, Value, Radix, and Counter. The "error" bit is set to 1.
- Trig Table:** A table with columns: Add, Active, Trigger Condition Name, and Trigger Condition Equation. The "error" bit is set to 1.
- Capture:** Settings for Type (Window), Windows (1), Depth (1024), and Position (512). Storage Qualification is set to "All Data".

At the bottom of the window, the command "COMMAND: run 4 0" is displayed, along with "Upload" and "DONE" buttons.

# ChipScope Pro Setup

- Click on Waveform; click the Arm Trigger button (1)
- Detection of an error will cause ChipScope Pro to trigger



# ChipScope Pro Setup

- To force a trigger, in order to view the waveform, click the T! button (1)

The screenshot shows the ChipScope Pro Analyzer interface. The toolbar at the top contains several icons, with the 'T!' button (a square with 'T!' inside) highlighted by a red box and a yellow circle containing the number '1'. The main window displays a waveform capture for 'DEV:4 MyDevice4 (XC5VLX110T) UNIT:0 MyILA0 (ILA)'. The waveform table lists the following signals:

Bus/Signal	X	O	0	80	160	240	320	400	480	560	640	720	800	880	960
app_wdf_afull	1	1													
app_af_afull	0	0													
app_wdf_wren	0	0													
app_af_cmd	0	0													
phy_init_done	1	1													
idelay_ctrl_rdy	1	1													
sys_rst_n	1	1													
error	0	0													
app_af_addr	0000	0000													

The waveform shows a sequence of signals over time. A cursor is positioned at sample 136, where a tooltip indicates 'sys\_rst\_n=1 Window=0 Sample=136'. The bottom of the interface shows the 'Upload' button and the 'DONE' button.

# XUPV5-LX110T DDR2 Verified at 200 MHz

- Virtex-5 –1 speed grade device supports 200 MHz DDR2 operation as stated in the [Virtex-5 datasheet](#)
- The XUPV5-LX110T SODIMM supports 200 MHz
  - Supplied DDR2 SODIMM is rated for 266 MHz operation
- The MIG test design and ChipScope Pro Analyzer verify operation of the XUPV5-LX110T with a Virtex-5 –1 speed grade device, at the specified clock rate of 200 MHz



# Appendix



# XUPV5-LX110T DDR2 at 266 MHz

- An external signal generator was used to clock the MIG test design on the XUPV5-LX110T at the maximum clock rate selectable in the MIG tool, 266 MHz
- Supplied DDR2 SODIMM is rated for 266 MHz operation
- The MIG test design and ChipScope Pro Analyzer verify operation of the XUPV5-LX110T at the maximum selectable MIG clock rate of 266 MHz for a -1 speed grade device

– See [UG086](#)



# Documentation

- Virtex-5 FPGA User Guide  
[http://www.xilinx.com/support/documentation/user\\_guides/ug190.pdf](http://www.xilinx.com/support/documentation/user_guides/ug190.pdf)
- Virtex-5 Packaging and Pinout Specification  
[http://www.xilinx.com/support/documentation/user\\_guides/ug195.pdf](http://www.xilinx.com/support/documentation/user_guides/ug195.pdf)
- Demos on Demand–Memory Interface Solutions with Xilinx FPGAs  
[http://www.demosondemand.com/clients/xilinx/001/page\\_new2/index.asp#35](http://www.demosondemand.com/clients/xilinx/001/page_new2/index.asp#35)
- Xilinx Memory Interface Generator (MIG) 2.1 User Guide  
<http://www.xilinx.com/support/software/memory/protected/ug086.pdf>
- Memory Interfaces Made Easy with Xilinx FPGAs and the Memory Interface Generator  
[http://www.xilinx.com/support/documentation/white\\_papers/wp260.pdf](http://www.xilinx.com/support/documentation/white_papers/wp260.pdf)

