LogiCORE™ IP System Monitor Wizard v2.0

Getting Started Guide

UG741 (v1.1) April 19, 2010

XILINX®

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
04/19/2010	1.1	Initial Xilinx release.

Table of Contents

Chapter 4: Detailed Example Design

Appendix A: References

Schedule of Figures

Chapter 1: Introduction

Chapter 2: Installing and Licensing the Core

Chapter 3: Running the Wizard

Chapter 4: Detailed Example Design

Schedule of Tables

Chapter 1: Introduction

Chapter 2: Installing and Licensing the Core

Chapter 3: Running the Wizard

Chapter 4: Detailed Example Design

About This Guide

The *LogiCORE IP System Monitor Wizard Getting Started Guide* describes the function and operation of the Xilinx® LogiCORE™ IP System Monitor Wizard in the Virtex®-5 LX/LXT/SXT sub-families, Virtex-6 LXT/SXT/HXT sub-families, lower-power VIrtex-6 devices, and Virtex-5Q and Virtex-6Q defense grade devices.

Guide Contents

This guide contains the following chapters:

- Preface, "About this Guide" introduces the organization and purpose of this guide, a list of additional resources, and the conventions used in this document.
- Chapter 1, "Introduction" describes the core and related information, including recommended design experience, additional resources, technical support, and submitting feedback to Xilinx.
- Chapter 2, "Installing and Licensing the Core" provides instructions for installing and licensing the System Monitor Wizard in the Xilinx® CORE Generator™ tool.
- Chapter 3, "Running the Wizard" provides a step-by-step procedure for generating the System Monitor Wizard, implementing the core in hardware using the accompanying example design, and simulating the core with the provided example test bench.
- Chapter 4, "Detailed Example Design" provides detailed information about the example design, including a description of files and the directory structure generated by the Xilinx® CORE Generator™ tool, the purpose and contents of the provided scripts, the contents of the example design, and the operation of the demonstration test bench.
- Appendix A, "References"

Additional Resources

To find additional documentation, see the Xilinx website at:

http://www.xilinx.com/support/documentation/index.htm.

To search the Answer Database of silicon, software, and IP questions and answers, or to create a technical support WebCase, see the Xilinx website at:

http://www.xilinx.com/support/mysupport.htm.

Conventions

This document uses the following conventions. An example illustrates each convention.

Typographical

The following typographical conventions are used in this document:

Convention	Meaning or Use	Example
`Courier font`	Messages, prompts, and program files that the system displays	`speed grade: - 100`
`Courier bold`	Literal commands that you enter in a syntactical statement	**`ngdbuild`** *design_name*
Helvetica bold	Commands that you select from a menu	**File → Open**
	Keyboard shortcuts	**Ctrl+C**
Italic font	Variables in a syntax statement for which you must supply values	**`ngdbuild`** *design_name*
	References to other manuals	See the *User Guide* for more information.
	Emphasis in text	If a wire is drawn so that it overlaps the pin of a symbol, the two nets are *not* connected.
Dark Shading	Items that are not supported or reserved	This feature is not supported
Square brackets [ ]	An optional entry or parameter. However, in bus specifications, such as **`bus[7:0]`**, they are required.	**`ngdbuild`** [*`option_name`*] *`design_name`*
Braces { }	A list of items from which you must choose one or more	**`lowpwr =`**{**`on`**|**`off`**}
Vertical bar |	Separates items in a list of choices	**`lowpwr =`**{**`on`**|**`off`**}
Angle brackets < >	User-defined variable or in code samples	<directory name>

Convention	Meaning or Use	Example

Vertical ellipsis
.
.
.
	Repetitive material that has been omitted	`IOB #1: Name = QOUT'`
		`IOB #2: Name = CLKIN'`
		.
		.
		.

Horizontal ellipsis `...`	Repetitive material that has been omitted	**`allow block`** *block_name loc1 loc2 ... locn;*
Notations	The prefix '0x' or the suffix 'h' indicate hexadecimal notation	A read of address 0x00112975 returned 45524943h.
	An '_n' means the signal is active low	**`usr_teof_n`** is active low.

Online Document

The following conventions are used in this document:

Convention	Meaning or Use	Example
Blue text	Cross-reference link to a location in the current document	See the section "Additional Resources" for details. Refer to "Title Formats" in Chapter 1 for details.
Blue, underlined text	Hyperlink to a website (URL)	Go to http://www.xilinx.com for the latest speed files.

XILINX®

Chapter 1

Introduction

This chapter introduces the System Monitor Wizard and provides related information, including recommended design experience, additional resources, technical support, and submitting feedback to Xilinx. The System Monitor Wizard core generates source Resistor Transistor Logic (RTL) to implement the System Monitor matched to your requirements and is designed to support both Verilog and VHDL design environments. In addition, the example design and simulation test bench delivered with the core is provided in both Verilog and VHDL.

About the Core

The System Monitor Wizard is a Xilinx® CORE Generator™ IP core, included in the latest IP Update on the Xilinx IP Center. For detailed information about the core, see the Architecture Wizards page. For information about licensing options, see Chapter 2, "Installing and Licensing the Core."

Recommended Design Experience

Although the System Monitor Wizard is a fully verified solution, the challenge associated with implementing a complete design varies depending on the configuration and functionality of the application. For best results, previous experience building high performance, pipelined FPGA designs using Xilinx implementation software and user constraints files (UCF) is recommended.

Contact your local Xilinx representative for a closer review and estimation for your specific requirements.

Related Xilinx Documents

Prior to generating the System Monitor Wizard, users should be familiar with the following:

- DS100: *Virtex-5 Family Overview* [Ref 1]
- DS150: *Virtex-6 Family Overview* [Ref 2]
- UG192: *Virtex-5 FPGA System Monitor User Guide* [Ref 3]
- UG370: *Virtex-6 FPGA System Monitor User Guide* [Ref 4]
- ISE software documentation: www.xilinx.com/ise [Ref 6]

Additional Core Resources

For detailed information and updates about the System Monitor Wizard, see the following documents, located on the Architecture Wizards page:

- *LogiCORE IP System Monitor Wizard Data Sheet*
- System Monitor Wizard Release Notes
- *LogiCORE IP System Monitor Wizard Getting Started Guide*

Technical Support

For technical support, go to www.xilinx.com/support. Questions are routed to a team with expertise using theSystem Monitor Wizard core.

Xilinx will provide technical support for use of this product as described in the *LogiCORE IP System Monitor Wizard Getting Started Guide*. Xilinx cannot guarantee timing, functionality, or support of this product for designs that do not follow these guidelines.

Feedback

Xilinx welcomes comments and suggestions about the System Monitor Wizard core and the accompanying documentation.

For comments or suggestions about the System Monitor Wizard, please submit a WebCase from www.xilinx.com/support/clearexpress/websupport.htm. Be sure to include the following information:

- Product name
- Core version number
- Explanation of your comments

Document

For comments or suggestions about the System Monitor Wizard, please submit a WebCase from www.xilinx.com/support/clearexpress/websupport.htm. Be sure to include the following information:

- Document title
- Document number
- Page number(s) to which your comments refer
- Explanation of your comments

XILINX®

Chapter 2

Installing and Licensing the Core

This chapter provides instructions for installing the LogiCORE™ IP System Monitor Wizard in the Xilinx® CORE Generator™ tool. It is not necessary to obtain a license to use the Wizard.

Supported Tools and System Requirements

Operating Systems

Windows

- Windows XP Professional 32-bit/64-bit
- Windows Vista Business 32-bit/64-bit

Linux

- Red Hat Enterprise Linux WS v4.0 32-bit/64-bit
- Red Hat Enterprise Desktop v5.0 32-bit/64-bit (with Workstation Option)
- SUSE Linux Enterprise (SLE) desktop and server v10.1 32-bit/64-bit

Tools

- Xilinx Implementation
  - ISE® 12.1
- Verification/Simulation:
  - ISim
  - Mentor Graphics ModelSim: 6.5c and above
  - Synopsys VCS and VCS MX 2009.12 and above
  - Cadence Incisive Enterprise Simulator (IES) 9.2 and above
- Synthesis
  - XST
  - Synopsys Synplify Pro D-2009.12

Check the release notes for the required Service Pack; ISE Service Packs can be downloaded from www.xilinx.com/xlnx/xil_sw_updates_home.jsp?update=sp.

Before you Begin

Before installing the Wizard, you must have a MySupport account and the ISE 12.1 software installed on your system. If you already have an account and have the software installed, go to "Installing the Wizard", otherwise do the following:

1. Click **Login** at the top of the Xilinx home page then follow the onscreen instructions to create a MySupport account.

Install the ISE 12.1 software. For the software installation instructions, see the ISE Design Suite Release Notes and Installation Guide available in ISE software Documentation.

Installing the Wizard

The System Monitor Wizard is included with the ISE 12.1 software. See *ISE CORE Generator IP Updates - Installation Instructions* for details on the installation of ISE 12.1.

Verifying Your Installation

Use the following procedure to verify that you have successfully installed the System Monitor Wizard in the CORE Generator tool.

1. Start the CORE Generator tool.
2. After creating a new Virtex-5 or Virtex-6 family project or opening an existing one, the IP core functional categories appear at the left side of the window, as shown in Figure 2-1, page 17.

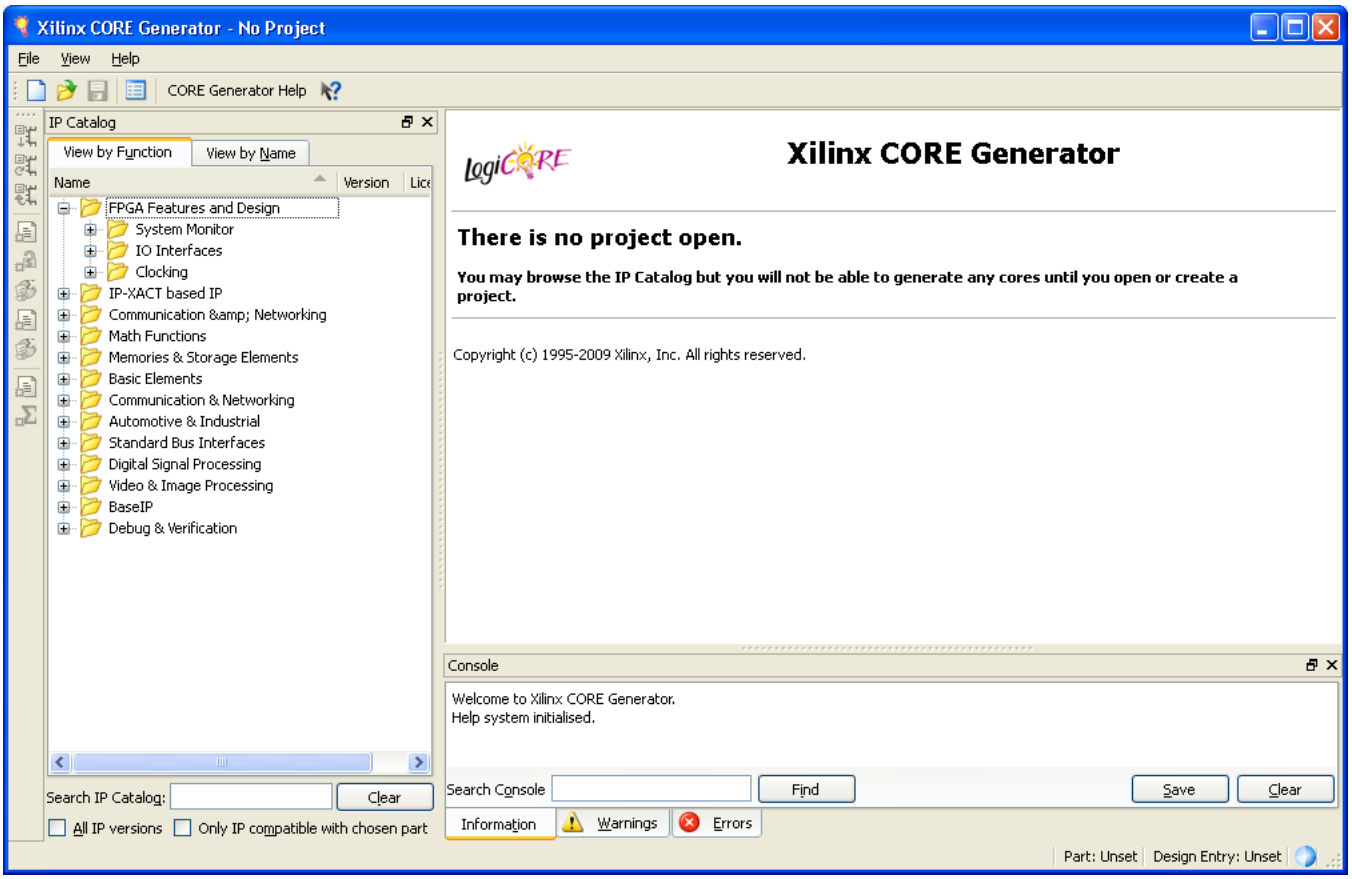

Figure 2-1: **CORE Generator Window**

3. Determine if the installation was successful by verifying that System Monitor Wizard appears at the following location in the Functional Categories list:
`/FPGA Features and Design/System Monitor`

XILINX®

Chapter 3

Running the Wizard

Overview

This chapter describes the GUI and follows the same flow required to set up the System Monitor (SYSMON). Tool tips are available in the GUI for most features; simply place your mouse over the relevant text, and additional information is provided in a pop-up dialog.

Setting Up the Project

Before generating the example design, set up the project as described in "Creating a Directory" and "Setting the Project Options," page 21 of this guide.

Creating a Directory

To set up the example project, first create a directory using the following steps:

1. Change directory to the desired location. This example uses the following location and directory name:

   `/Projects/sysmon_example`

2. Start the Xilinx CORE Generator ™ software.

   For help starting and using the CORE Generator software, see *CORE Generator Help*, available in ISE software documentation [Ref 6].

3. Choose **File > New Project** (Figure 3-1, page 20).
4. Change the name of the `.cgp` file (optional).
5. Click **Save**.

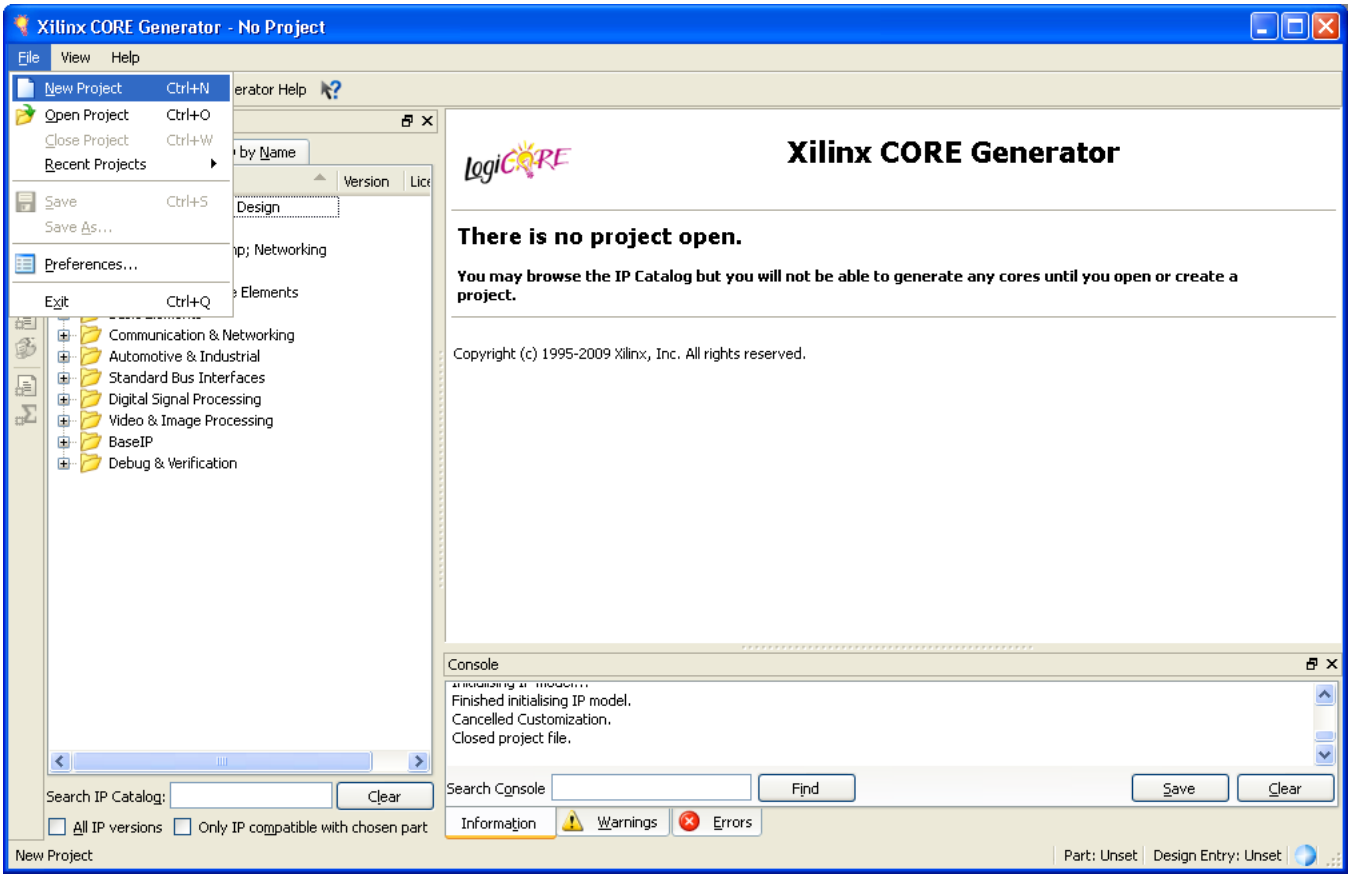

Figure 3-1: **Starting a New Project**

Setting the Project Options

Set the project options using the following steps:

1. Click **Part** in the option tree.
2. Select **Virtex5, Virtex6, Virtex-6 (lower power), Virtex-5Q defense grade, or Virtex-6Q defense grade** from the Family list.
3. Select a device from the Device list that support SYSMON.
4. Select an appropriate package from the Package list. This example uses the XC6VLX75T device (see Figure 3-2).

   Note: If an unsupported silicon family is selected, the System Monitor Wizard remains light grey in the taxonomy tree and cannot be customized. Only devices containing the System Monitor Wizard are supported by the Wizard. See the *Virtex-5 Family Overview* [Ref 1] and *Virtex-6 Family Overview* [Ref 2] for a list of devices containing SYSMON.
5. Click **Generation** in the option tree and select either Verilog or VHDL as the output language.
6. Click **OK**.

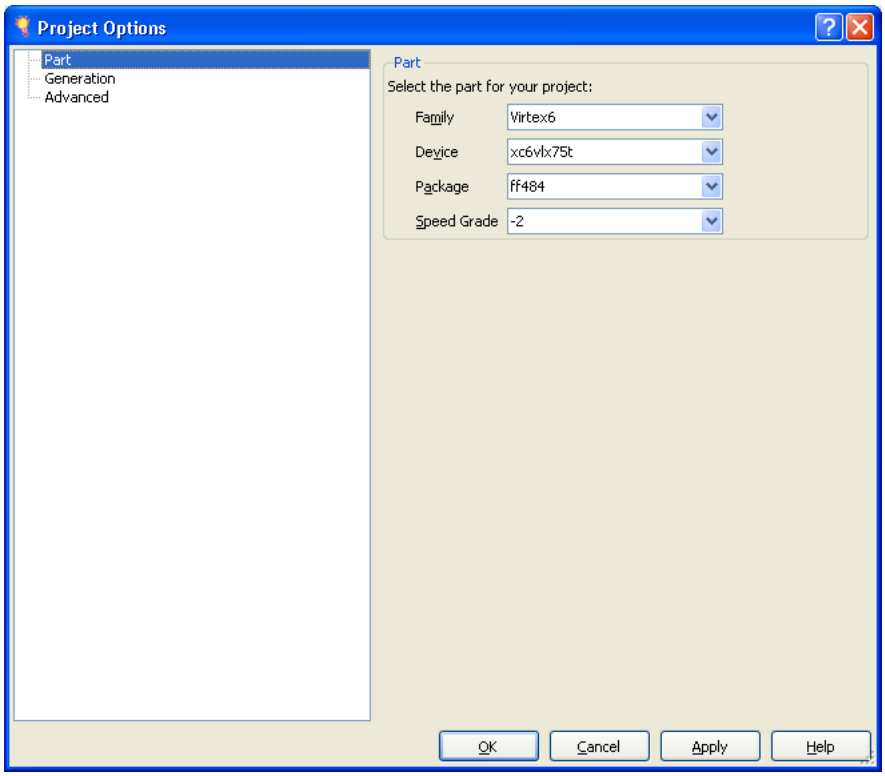

Figure 3-2: **Target Architecture Setting**

Generating the Core

This section provides instructions for generating an example SYSMON design using the default values. The wrapper and its supporting files, including the example design, are generated in the project directory. For additional details about the example design files and directories provided with the System Monitor Wizard, see Chapter 4, "Detailed Example Design."

1. Locate System Monitor Wizard in the taxonomy tree under:

   `/FPGA Features and Design/System Monitor.` (See Figure 3-3)

2. Double-click System Monitor Wizard to launch the Wizard.

After the wizard is launched, the CORE Generator tool displays a series of screens that allow you to configure the System Monitor Wizard.

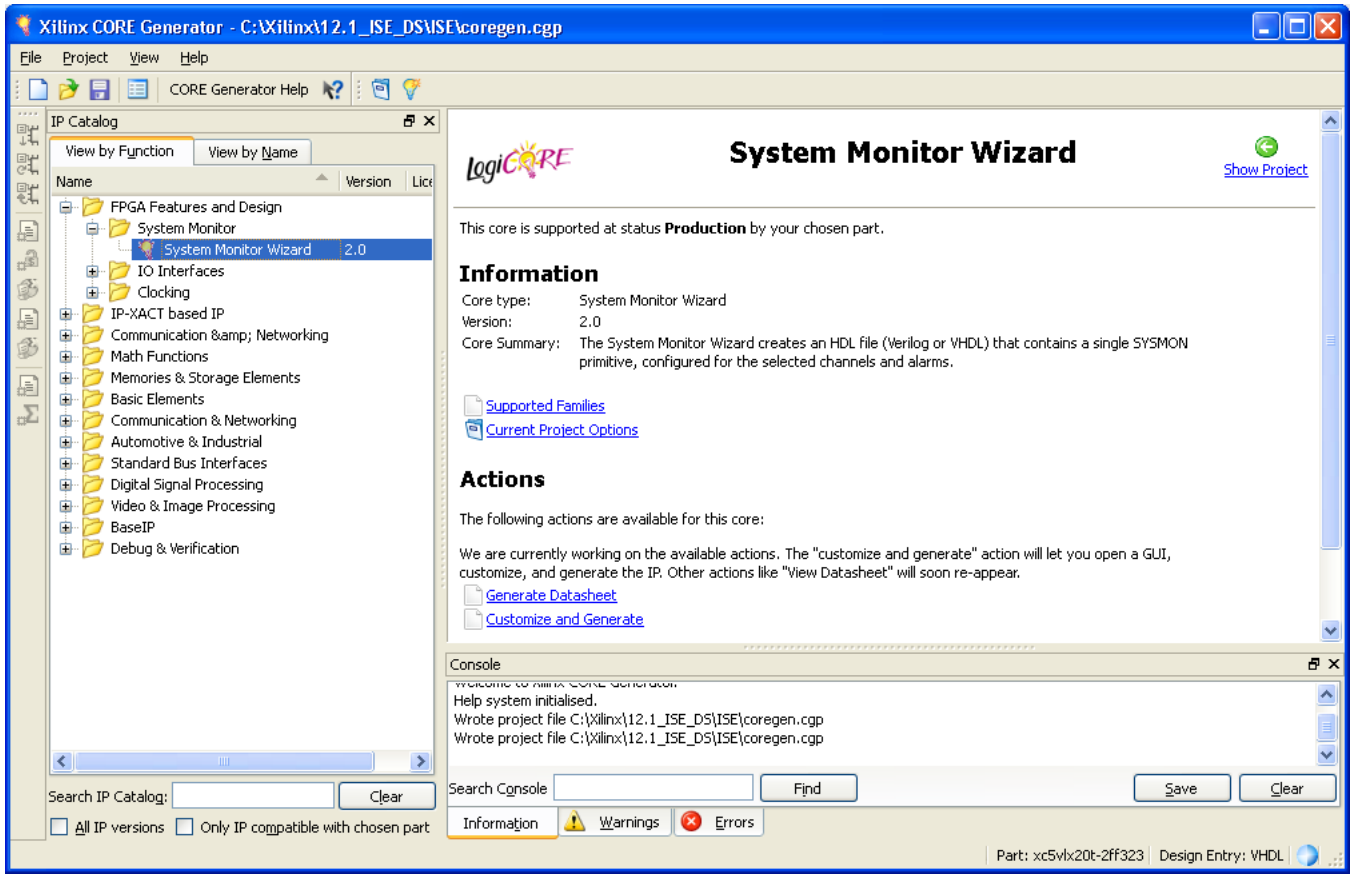

Figure 3-3: **Locating the System Monitor Wizard**

System Monitor Setup

The System Monitor Wizard screen (Page 1) of the Wizard (Figure 3-4) allows you to select the component name, analog stimulus filename, startup channel mode, timing mode, and DRP timing options.

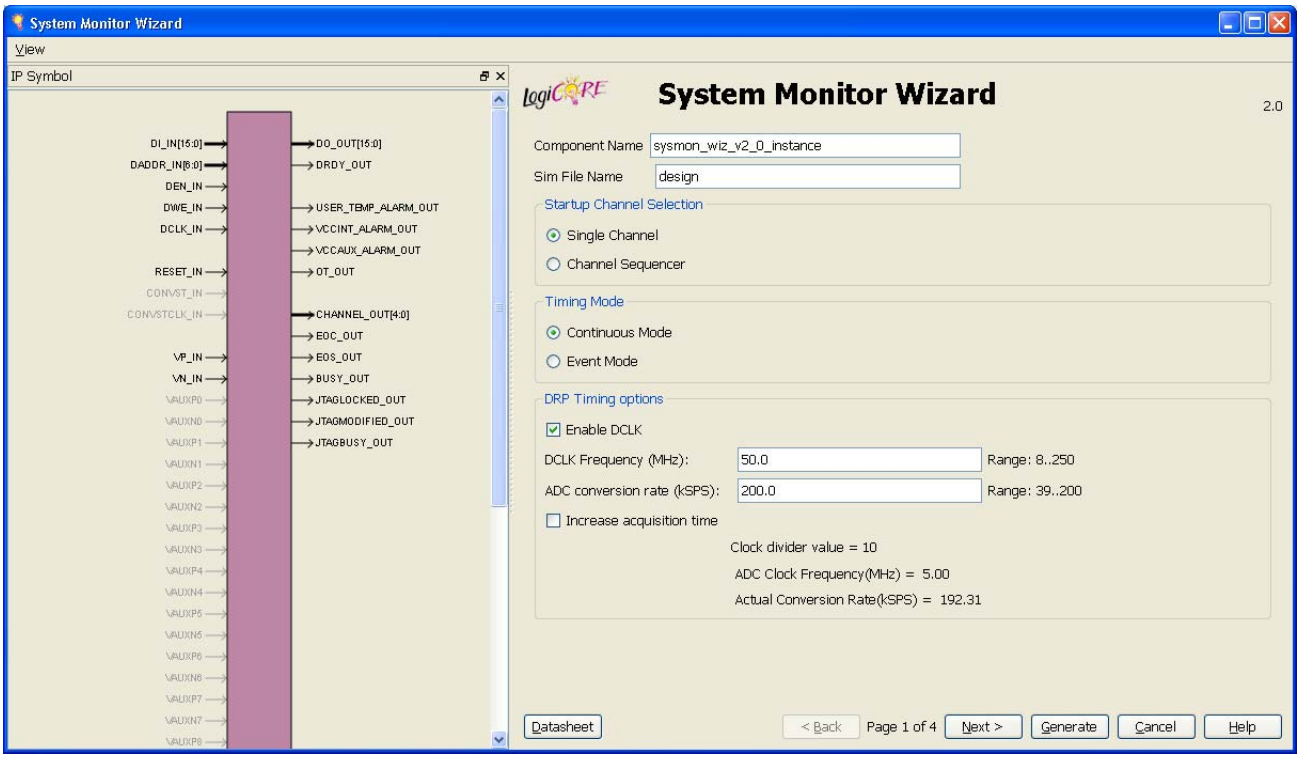

Figure 3-4: **System Monitor Setup - Page 1**

Component Name

User selectable component name is available. Component names must not contain any reserved words in Verilog or VHDL.

SIM File Name

Use this field to customize the name of the System Monitor analog stimulus file.

Startup Channel Selection

The System Monitor can be configured in one of the two modes listed below:

- Single Channel: In this mode, the user can select only one channel to monitor.
- Channel Sequencer: Choosing this mode, allows the user to select any number of channels to monitor. The channels to be used for this mode can be selected on Page 4 (Figure 3-7, page 29) and Page 5 (Figure 3-8, page 30) of the Wizard.

Timing Mode

The SYSMON can operate in two timing modes:

- Continuous Mode: In this mode, the SYSMON continues to sample and convert the selected channel/channels.
- Event Mode: This mode requires an external trigger event, CONVST or CONVSTCLK, to start a conversion on the selected channel. Event Mode should only be used with external channels. SYSMON does not support Event Timing Mode when Channel Sequencer is enabled.

DRP Timing Options

The System Monitor clock (ADCCLK) is derived from the dynamic reconfiguration port (DRP) clock DCLK. The SYSMON supports a DRP clock frequency of up to 250MHz. The SYSMON can also operate in absence of DCLK. For more information on the DRP see the *Virtex-5 FPGA System Monitor User Guide* [Ref 3] or *Virtex-6 FPGA System Monitor User Guide* [Ref 4].

The ADCCLK clock, should be in the range of 1-5.2 Mhz. To support this lower frequency clock the System Monitor has an internal clock divider. The GUI allows an external DCLK frequency and required ADC conversion rate (maximum 200 ksps) to be specified. Based on the value of DCLK clock, the wizard then calculates the appropriate clock divider value based on the values of DCLK clock and ADC conversion rate, the wizard calculates the appropriate clock divider value.

The wizard also displays the ADC Clock frequency value and the actual conversion rate of the ADC.

The System Monitor defaults to Continuous Timing Mode if no external clock (DCLK) is detected. For Single Channel Mode of operation, the GUI provides the option of increasing the acquisition time associated with external analog input channels. Enabling Increase Acquisition Time would restrict the selection of the channel to external channels. See the *Virtex-5 FPGA System Monitor User Guide* [Ref 3] and the *Virtex-6 FPGA System Monitor User Guide* [Ref 4] for more information.

I/O Ports

The I/O Port Selection screen (Page 2) of the Wizard (Figure 3-5) allows the user to select the I/O ports on the SYSMON primitive.

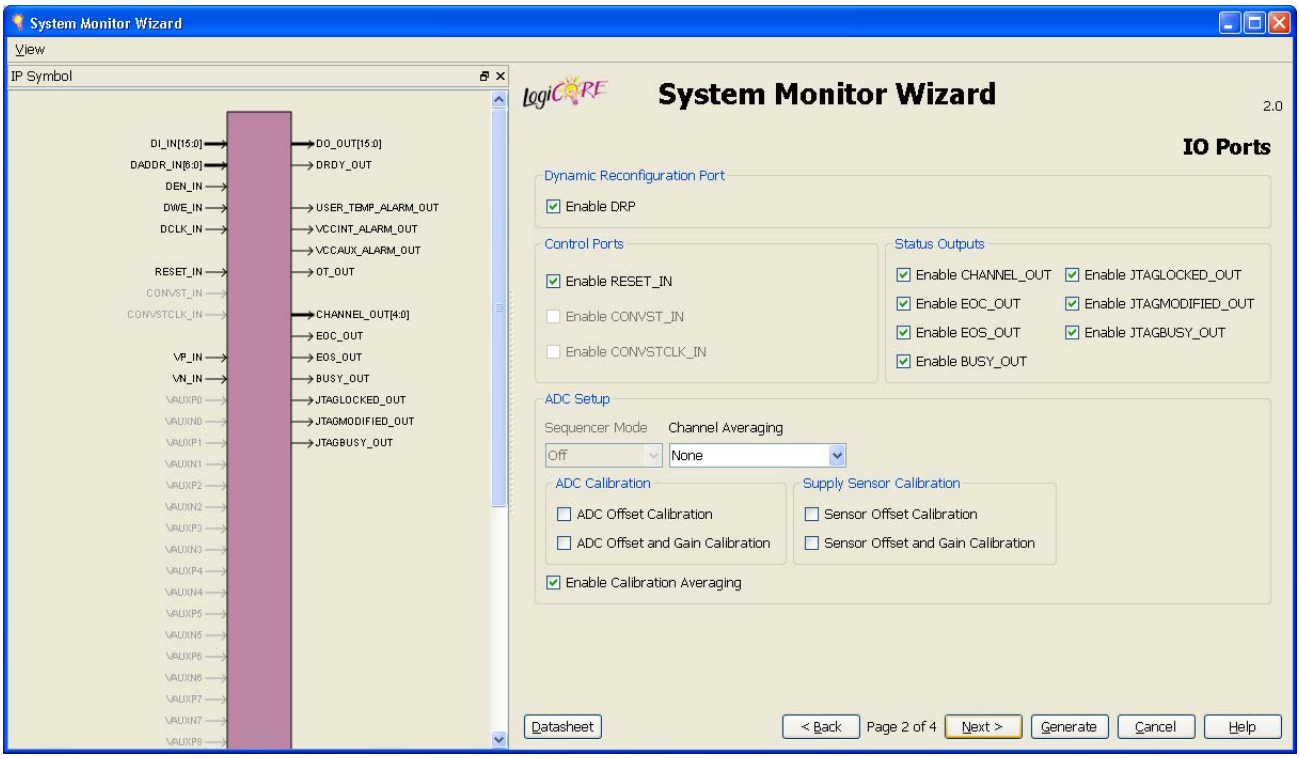

Figure 3-5: **I/O Ports - Page 2**

Dynamic Reconfiguration Port

This port is the FPGA fabric interface for System Monitor. It facilitates access to the register file interface of the System Monitor. The System Monitor control registers can be read or written using this port. This port can be enabled only when DCLK clock is present.

Control Ports

This section allows the user to select control input ports:

- RESET_IN allows an external input reset signal to be connected to the System Monitor
- CONVST_IN and/or CONVSTCLK_IN as trigger sources for Event Mode Timing

Status Outputs

A number of output status signals are also provided to facilitate interfacing of the System Monitor to a user design. See the *Virtex-5 FPGA System Monitor User Guide* [Ref 3] and the *Virtex-6 FPGA System Monitor User Guide* [Ref 4] for more information.

ADC Setup

If the SYSMON is configured for Channel Sequencer Mode, the user can choose the required sequencer mode. The available options are Continuous, One-pass or Default mode.

The Channel Averaging drop-down menu allows the user to select the required averaging value. The available options are None, 16, 64 and 256.

The user can also select the type of ADC Calibration and/or Supply Sensor Calibration. This can be done by checking the respective checkboxes

Calibration Averaging, by default, is enabled in SYSMON. The user can disable this by unchecking the box.

Alarm Setup

The Alarm Setup screen (Page 3) of the Wizard (Figure 3-6) allows alarm outputs to be enabled for the on-chip sensors. If a measurement of an on-chip sensor lies outside the specified limits, then a logic output will go active if enabled. For a detailed description of the alarm functionality see the *Virtex-5 FPGA System Monitor User Guide* [Ref 3] and the *Virtex-6 FPGA System Monitor User Guide* [Ref 4].

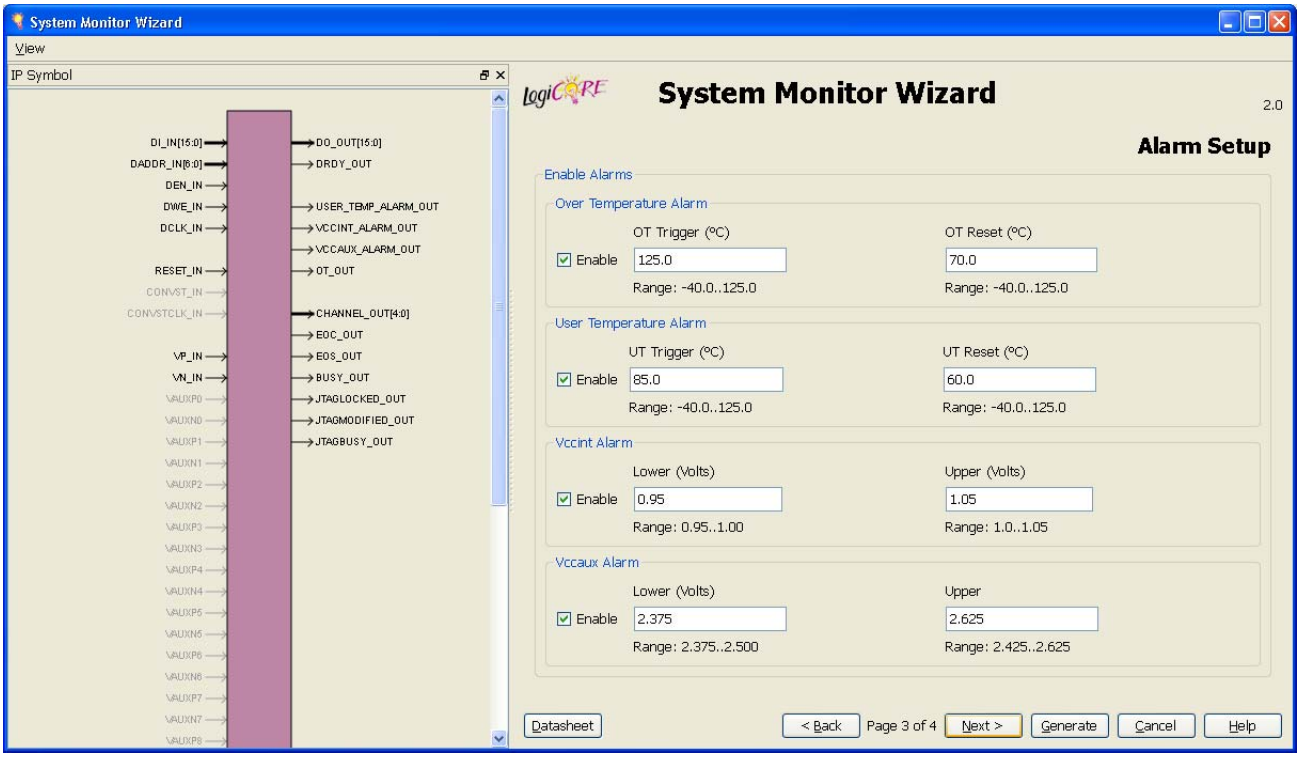

Figure 3-6: **Alarm Setup - Page 3**

Enable Alarms

Use the checkboxes to enable alarm logic outputs. The four options are:

- Over temperature alarm
- User temperature alarm
- V_{CCNT} alarm
- V_{CCAUX} alarm

Temperature Alarm Limits

Trigger and Reset levels for the user temperature alarm output can be entered using these fields. The trigger level for OT (over temperature) alarm is factory set to 125°C for a Virtex-5 device; only the lower reset value can be specified by the user. For a Virtex-6 device, the user can set both; the trigger as well as reset levels for the OT alarm.

VCCINT and VCCAUX Limits

Both upper and lower alarm thresholds can be specified for the on-chip power supplies. If the measured value moves outside these limits the alarm logic output will go active. The alarm output is reset once a measurement inside these limits is generated. The default limits in the GUI represent ± 5% on the nominal supply value.

Channel Sequencer Setup P1 and Setup P2

Channel Sequencer Setup P1 (Figure 3-7) and Setup P2 (Figure 3-8) screens of the Wizard are used to configure the Channel Sequencer. All the possible channels that can be included in the sequence are listed in the table spread across screens 4 and 5 (Figure 3-7 and Figure 3-8) of the Wizard:

- Use the Channel Sequencer Setup P1 and P2 screen to select Channels for monitoring, enable Averaging for selected channels, enable Bipolar mode for external channels and increase the Acquisition time for the selected channels.

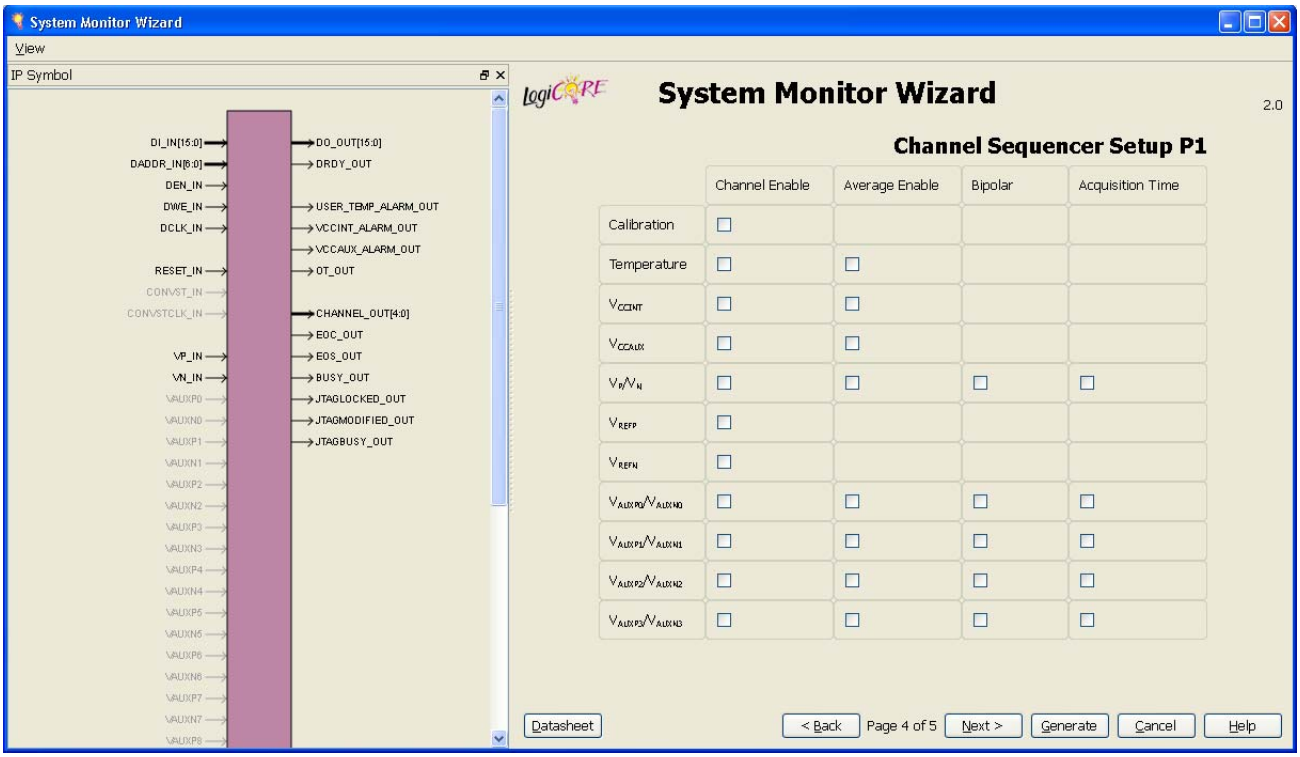

Figure 3-7: **Channel Sequencer Setup P1 - Page 4**

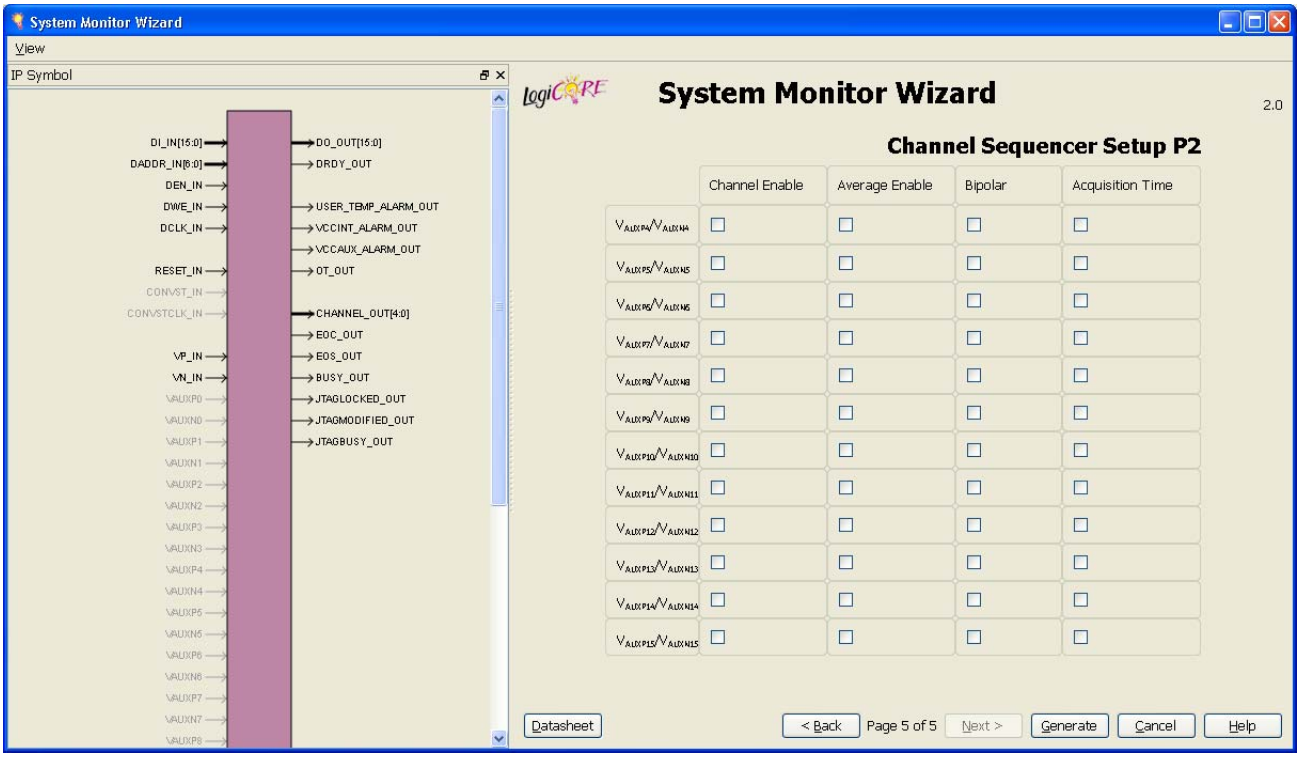

Figure 3-8: **Channel Sequencer Setup P2 - Page 5**

Single Channel Mode

If the Single Channel Mode operation (see "Startup Channel Selection," page 23) is selected on Page 1 of the Wizard (Figure 3-4, page 23), the Single Channel Setup is displayed on Page 4 of the Wizard (Figure 3-9). Page 4 allows the user to select the channel for measurement and the analog input mode if the channel is an external analog input (that is, unipolar or bipolar).

The columns Channel Enable, Average Enable and Increase Acquisition Time are disabled and are shown only for user information and ease.

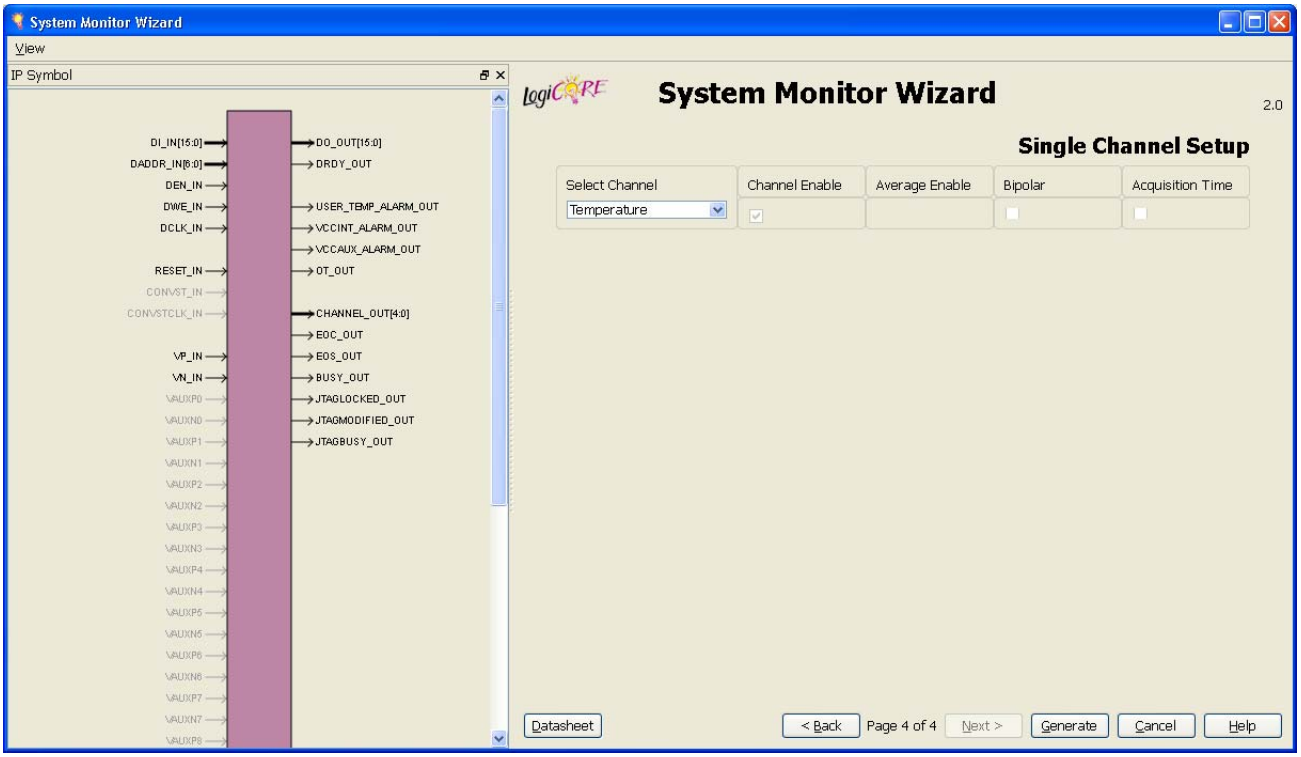

Figure 3-9: **Single Channel Setup - Page 4**

Generating the HDL Wrapper

After selecting the configuration options, click Generate on the final Wizard screen to generate the HDL wrapper and other Wizard outputs.

The output files are placed in the project directory you selected or created when setting up a new CORE Generator project.

XILINX®

Chapter 4

Detailed Example Design

This chapter provides detailed information about the example design, including a description of files and the directory structure generated by the Xilinx® CORE Generator™ software, the purpose and contents of the provided scripts, the contents of the example HDL wrappers, and the operation of the demonstration test bench.

Directory and File Structure

- **<project directory>**
  Top-level project directory; name is user-defined
  - <project directory>/<component name>
    Core release notes file
    - <component name>/doc
      Product documentation
    - <component name>/example design
      Verilog and VHDL (or whichever, if it's only one) design files
    - <component name>/implement
      Implementation script files
      - implement/results
        Results directory, created after implementation scripts are run, and contains implement script results
    - <component name>/simulation
      Simulation scripts
      - simulation/functional
        Functional simulation files

Directory and File Contents

The System Monitor Wizard directories and their associated files are defined in the following sections.

<project directory>

The <project directory> contains all the CORE Generator project files.

Table 4-1: **Project Directory**

Name	Description
<project_dir>	
<component_name>.v[hd]	Verilog or VHDL simulation model.
<component_name>.xco	CORE Generator project-specific option file; can be used as an input to the CORE Generator.
<component_name>_flist.txt	List of files delivered with the core.
<component_name>.{veo|vho}	VHDL or Verilog instantiation template.

Back to Top

<project directory>/<component name>

The <component name> directory contains the readme file provided with the core, which may include last-minute changes and updates.

Table 4-2: **Component Name Directory**

Name	Description
<project_dir>/<component_name>	
sysmon_wiz_readme.txt	Core readme file.

Back to Top

<component name>/doc

The doc directory contains the PDF documentation provided with the core.

Table 4-3: **Doc Directory**

Name	Description
<project_dir>/<component_name>/doc	
sysmon_wiz_ds608.pdf	*LogiCORE IP System Monitor Wizard Data Sheet*
sysmon_wiz_gsg741.pdf	*LogiCORE IP System Monitor Wizard Getting Started Guide*

Back to Top

<component name>/example design

The example design directory contains the example design files provided with the core.

Table 4-4: **Example Design Directory**

Name	Description
<project_dir>/<component_name>/example_design	
<component_name>_exdes.v(hd)	Verilog and VHDL top-level example design file.

Back to Top

<component name>/implement

The implement directory contains the core implementation script files.

Table 4-5: **Implement Directory**

Name	Description
<project_dir>/<component_name>/implement	
implement.bat implement.sh	Windows and Linux based implementation scripts.
xst.prj	The XST project file for the example design; it lists all of the source files to be synthesized.
xst.scr	The XST script file for the example design that is used to synthesize the core, called from the implement script described above.

Back to Top

implement/results

The results directory is created by the implement script, after which the implement script results are placed in the results directory.

Table 4-6: **Results Directory**

Name	Description
<project_dir>/<component_name>/implement/results	
Implement script result files.	

Back to Top

<component name>/simulation

The simulation directory contains the simulation scripts provided with the core.

Table 4-7: **Simulation Directory**

Name	Description
<project_dir>/<component_name>/simulation	
<component_name>_tb.v[hd]	Demonstration test bench.

Back to Top

simulation/functional

The functional directory contains functional simulation scripts provided with the core.

Table 4-8: **Functional Directory**

Name	Description
<project_dir>/<component_name>/simulation/functional	
simulate_isim.sh simulate_isim.bat	Linux and Windows simulation scripts for ISIM simulator.
simulate_mti.do	Modelsim simulation script.
simulate_ncsim.sh	Linux script for running simulation using Cadence Incisive Enterprise Simulator (IES).
simulate_vcs.sh	Linux script for running simulation using VCS MX.

Back to Top

Implementation Scripts

The implementation script is either a shell script or batch file that processes the example design through the Xilinx tool flow. It is located at:

LINUX and UNIX

```
<project_dir>/<component_name>/implement/implement.sh
```

Windows

```
<project_dir>/<component_name>/implement/implement.bat
```

The implement script performs the following steps:

- Synthesizes the HDL example design files using XST
- Runs Ngdbuild to consolidate the core netlist and the example design netlist into the NGD file containing the entire design
- Maps the design to the target technology
- Place-and-routes the design on the target device
- Performs static timing analysis on the routed design using Timing Analyzer (TRCE)
- Generates a bitstream
- Enables Netgen to run on the routed design to generate a VHDL or Verilog netlist (as appropriate for the Design Entry project setting) and timing information in the form of SDF files

The Xilinx tool flow generates several output and report files. These are saved in the following directory which is created by the implement script:

```
<project_dir>/<component_name>/implement/results
```

Simulation Scripts

Functional Simulation

The test scripts are a ModelSim, Cadence IES, VCS, VCS MX, or ISim macro that automate the simulation of the test bench. They are available from the following location:

```
<project_dir>/<component_name>/simulation/functional/
```

The test script performs the following tasks:

- Compiles the structural UniSim simulation model
- Compiles HDL Example Design source code
- Compiles the demonstration test bench
- Starts a simulation of the test bench
- Opens a Wave window and adds signals of interest
- Runs the simulation to completion

Example Design

Top Level Example Design

The following files describe the top-level example design for the System Monitor Wizard core.

VHDL

```
project_dir>/<component_name>/example_design/<component_name>_exdes.vhd
```

Verilog

```
project_dir>/<component_name>/example_design/<component_name>_exdes.v
```

The example design, instantiates the SYSMON core that is generated by the wizard.

Demonstration Test Bench

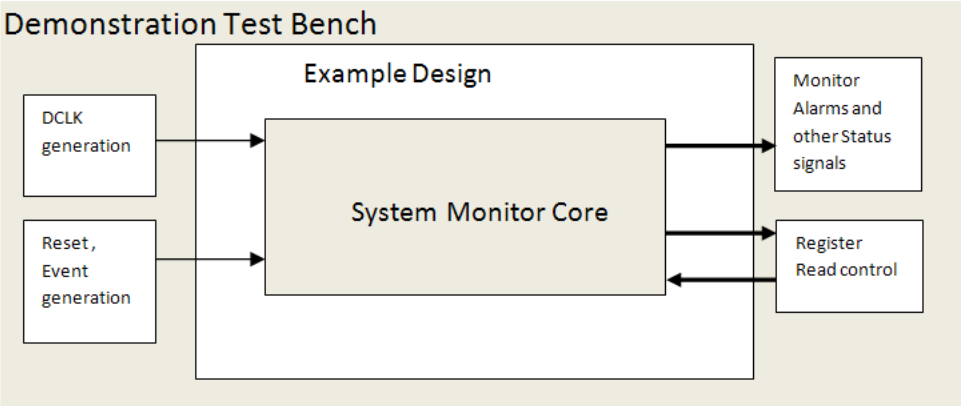

Figure 4-1: **Demonstration Test Bench for the System Monitor Wizard and Example Design**

The following files describe the demonstration test bench.

VHDL

```
project_dir>/<component_name>/simulation/<component_name>_tb.vhd
```

Verilog

```
project_dir>/<component_name>/simulation/<component_name>_tb.v
```

The demonstration test bench is a simple VHDL or Verilog program to exercise the example design and the core.

The demonstration test bench performs the following tasks:

- Generates the input DCLK clock signal
- Applies a reset to the example design
- Monitors the alarms and other status outputs
- Reads the respective registers when a conversion is complete

XILINX®

Appendix A

References

Documents Specific to Virtex® FPGAs

1. DS100: *Virtex-5 Family Overview*
2. DS150: *Virtex-6 Family Overview*

Documents Specific to the System Monitor

3. UG192: *Virtex-5 FPGA System Monitor User Guide*
4. UG370: *Virtex-6 FPGA System Monitor User Guide*
5. DS608: *LogiCORE IP System Monitor Wizard Data Sheet*

Xilinx® ISE® Tools and Solutions

6. ISE software documentation: www.xilinx.com/ise