System Management Wizard v1.3

LogiCORE IP Product Guide

Vivado Design Suite

PG185 April 6, 2016

XILINX
ALL PROGRAMMABLE™

Table of Contents

IP Facts

Chapter 1: Overview

Applications 6
Licensing and Ordering Information 7

Chapter 2: Product Specification

SYSMON Functional Features 8
Standards 9
Performance 9
Resource Utilization 9
Port Descriptions 9
Register Space 13

Chapter 3: Designing with the Core

Clocking 46
Resets 46
Protocol Description 47
I2C Interface for SSIT Devices 47

Chapter 4: Design Flow Steps

Customizing and Generating the Core 50
Constraining the Core 69
Simulation 70
Synthesis and Implementation 70

Chapter 5: Example Design

Open Example Project Flow 72

Chapter 6: Test Bench

Appendix A: Debugging

Finding Help on Xilinx.com 74
Debug Tools 76

Simulation Debug 76
Hardware Debug 78
Interface Debug 78

Appendix B: Additional Resources and Legal Notices

Xilinx Resources 79
References 79
Revision History 79
Please Read: Important Legal Notices 80

Introduction

The LogiCORE™ IP System Management Wizard provides a complete solution for system-monitoring Xilinx UltraScale™ devices. This IP generates an HDL wrapper to configure the SYSMON for user-specified external channels, internal sensor channels, modes of operation and alarms. This IP supports monitoring of up to four user supplies. In addition, the System Management Wizard configures various interfaces for accessing SYSMON registers.

Features

- On-chip voltage and temperature measurements
- 10-bit 0.2 MSPS analog-to-digital conversion
- Access to 16 pairs of I/O pins as input channels
- Stand-alone measurement of system functionality including sequences and alarms
- Triple access (FPGA Fabric/JTAG/I2C) DRP including control and status registers
- Optional AXI4-Lite interface based on the AXI4 specification
- Optional I^2C interface
- Easy configuration of various modes and parameters
- Simple interface for channel selection and configuration
- Ability to select/deselect alarm outputs and set alarm limits
- Calculates all attributes of the Primitive based on user requirements

LogiCORE IP Facts Table	
Core Specifics	
Supported Device Family(1)	UltraScale™, UltraScale+
Supported User Interfaces	AXI4-Lite, DRP, I2C, PMBus(2)
Resources	Performance and Resource Utilization web page
Provided with Core	
Design Files	Verilog and VHDL
Example Design	Verilog
Test Bench	Verilog
Constraints File	XDC
Simulation Model	Not Provided
Supported S/W Driver	Standalone
Tested Design Flows(3)	
Design Entry	Vivado® Design Suite
Simulation	For supported simulators, see the Xilinx Design Tools: Release Notes Guide.
Synthesis	Vivado Synthesis
Support	
Provided by Xilinx at the Xilinx Support web page	

Notes:

1. For a complete list of supported devices, see the Vivado IP catalog .
2. PMBus is supported for UltraScale+ only.
3. For the supported versions of the tools, see the Xilinx Design Tools: Release Notes Guide.

Overview

The System Management Wizard guides you through configuring the SYSMON primitive through a user-friendly GUI and generates Verilog and VHDL Register Transfer Level (RTL) source files for Xilinx® UltraScale™ and UltraScale+ FPGAs.

IMPORTANT: *Throughout this Product Guide, references to SYSMON point to SYSMONE1 in UltraScale and SYSMONE4 in UltraScale+ devices.*

The System Management Wizard does not have access to the PS SYSMON in UltraScale+ devices. It instantiates PL SYSMONE4 and adds new features. An example design and simulation test bench demonstrate how to integrate the core into user designs. The top-level block diagram for the System Management Wizard is shown in Figure 1-1.

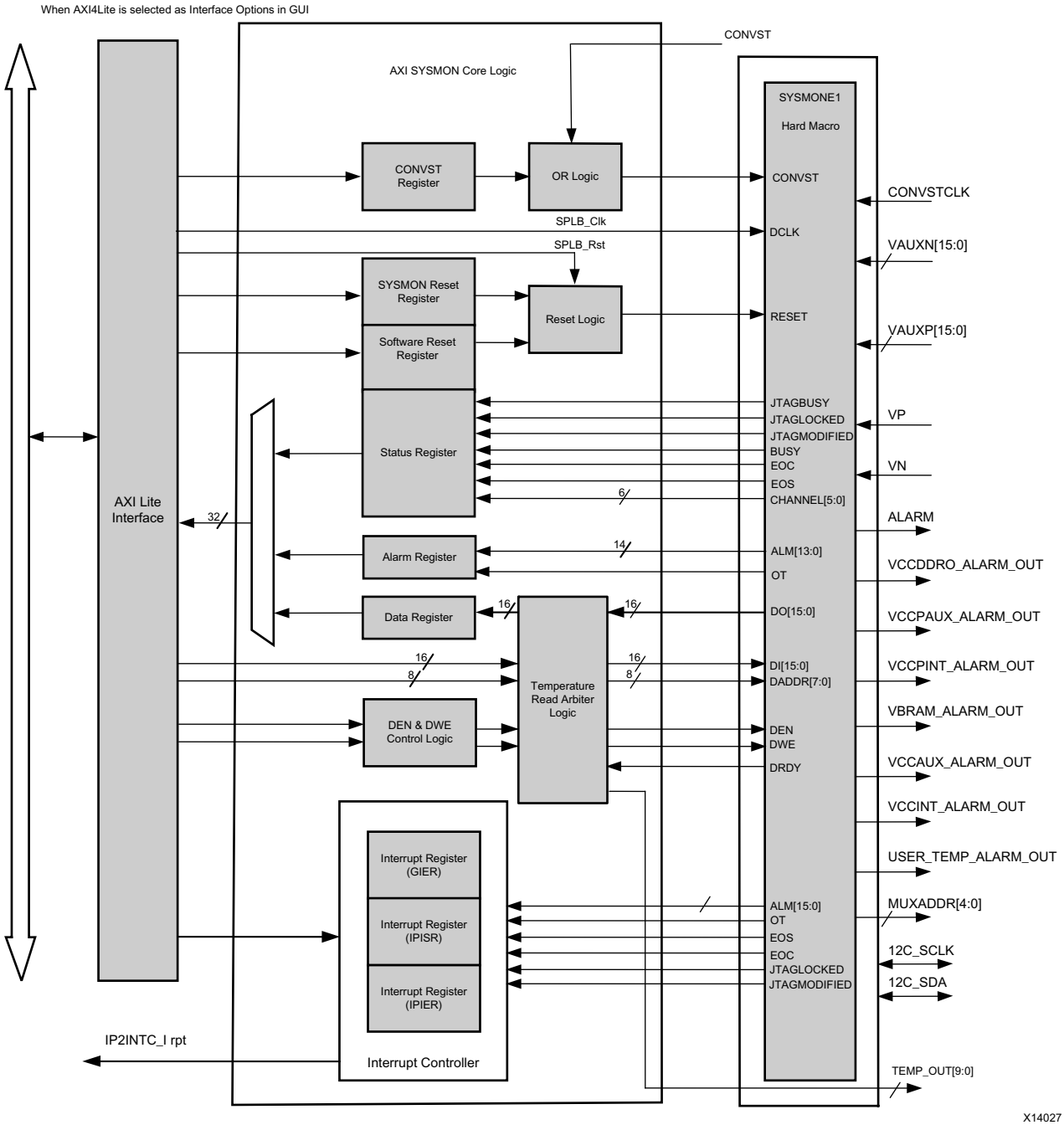

Figure 1-1: **System Management Wizard Block Diagram**

Applications

The System Management Wizard enables you to configure the integrated system management functions of the FPGA, such as monitoring user supplies and temperature.

Licensing and Ordering Information

License Checkers

If the IP requires a license key, the key must be verified. The Vivado® design tools have several license checkpoints for gating licensed IP through the flow. If the license check succeeds, the IP can continue generation. Otherwise, generation halts with error. License checkpoints are enforced by the following tools:

- Vivado design tools: Vivado Synthesis
- Vivado Implementation
- write_bitstream (Tcl command)

IMPORTANT: *IP license level is ignored at checkpoints. The test confirms a valid license exists. It does not check IP license level.*

License Type

This Xilinx LogiCORE™ IP module is provided at no additional cost with the Xilinx Vivado Design Suite under the terms of the Xilinx End User License. Information about this and other Xilinx LogiCORE IP modules is available at the Xilinx Intellectual Property page. For information about pricing and availability of other Xilinx LogiCORE IP modules and tools, contact your local Xilinx sales representative.

Product Specification

The System Management Wizard instantiates a SYSMON block for UltraScale configured to your requirements. The wizard allows you to select the channels, enable alarms, and set the alarm limits. For interfaces, you can select AXI4-Lite, DRP, or None. In addition to these interfaces, the wizard also supports I^2C and PMBus serial interfaces.

Stacked Silicon Interconnect (SSI) Kintex® UltraScale™ devices (XCKU100 and XCKU115) contain two SYSMON blocks, and Virtex® UltraScale devices (XCVU125, XCVU160, XCVU190 and XCVU440) and all Ultrascale + contain up to three SYSMON blocks. Each die in SSI device contains a SYSMON block in it.

SYSMON Functional Features

Major functional SYSMON features common to SYSMONE1 and SYSMONE4 can be used to determine an appropriate mode of operation. These features include:

- FPGA temperature and voltage monitoring
- Analog-to-digital conversion for seventeen external analog inputs
- Alarm generation based on up to 17 set parameters

Additional Features of SYSMONE4

- Direct access to measured data through ADC_DATA port
- Monitoring of PS supplies (VCCPSINTLP, VCCPSINTFP, VCCPSAUX)
- Additional system monitor within PS can operate up to 1 MSPS
- Dual sequence
- SMBALERT for power management bus (PMBus) applications
- Common-N reduces package pins for auxiliary analog inputs by sharing a single N for single ended

Standards

The System Management Wizard core contains AXI4-Lite interfaces, which is based on the AMBA® AXI4 specification.

Performance

If you enable averaging of the channel, data capture rate is reduced depending on the averaging selected. Choose the appropriate value to match your requirement. Analog input noise from the supply or board can alter the expected 10-bit digital output.

Maximum Frequencies

The maximum `s_axi_aclk/dclk` clock frequency supported is 250 MHz.

Note: In SSIT devices, using the AXI Interface does not guarantee a frequency of 250 MHz.

Resource Utilization

For details about resource utilization, visit Performance and Resource Utilization.

When only the DRP interface is selected, the System Management Wizard uses SYSMON primitive only. Therefore, no LUTs are used as resource.

The maximum clock frequency results are post-implementation using the default tool settings. The resource usage results do not include the characterization registers and represent the true logic used by the core. LUT counts include SRL16s or SRL32s.

Clock frequency does not take clock jitter into account and should be derated by an amount appropriate to the clock source jitter specification. The maximum achievable clock frequency and the resource counts might also be affected by other tool options, additional logic in the FPGA, different versions of Xilinx tools, and other factors.

Port Descriptions

Table 2-1 lists the input and output ports provided from the System Management Wizard. Availability of ports is controlled by user-selected parameters. For example, when Dynamic Reconfiguration is selected, only ports associated with Dynamic Reconfiguration are

exposed. Any port that is not exposed is tied off or connected to a signal labeled as unused in the delivered source code.

Table 2-1: **System Management Wizard I/O Signals**

Port	Direction	Description
di_in[15:0](2)	Input	Input data bus for the dynamic reconfiguration port (DRP).
sysmon_slave_sel[1:0]	Input	Selects master or slave SYSMON to access the DRP and control signals when **Interface Selection** is DRP. This port is only available for SSI devices. • 00: Master SYSMON • 01: Slave 0 SYSMON • 10: Slave 1 SYSMON
do_out[15:0]	Output	Output data bus for the dynamic reconfiguration port.
daddr_in[7:0]	Input	Address bus for the dynamic reconfiguration port.
den_in	Input	Enable signal for the dynamic reconfiguration port.
dwe_in	Input	Write enable for the dynamic reconfiguration port.
dclk_in	Input	Clock input for the dynamic reconfiguration port.
drdy_out	Output	Data ready signal for the dynamic reconfiguration port.
reset_in(2)	Input	Reset signal for the SYSMON control logic and maximum/minimum registers.
convst_in	Input	Convert start input. This input is used to control the sampling instant on the ADC input and is only used in Event Mode Timing (see *UltraScale Architecture System Monitor Advanced Specification User Guide* (UG580) [Ref 1]).
convstclk_in	Input	Convert start input. This input is connected to a global clock input on the interconnect. Like CONVST, this input is used to control the sampling instant on the ADC inputs and is only used in Event Mode Timing. The frequency of this clock should be greater than or equal to the sampling rate.
vp_in vn_in	Input	One dedicated analog-input pair. The SYSMON has one pair of dedicated analog-input pins that provide a differential analog input.
vauxp15[15:0] vauxn15[15:0]	Inputs	16 auxiliary analog-input pairs. Also, the SYSMON uses 16 differential digital-input pairs as low-bandwidth differential analog inputs. These inputs are configured as analog during FPGA configuration.
user_temp_alarm_out	Output	SYSMON temperature-sensor alarm output.
vccint_alarm_out	Output	SYSMON VCCINT-sensor alarm output.
vccaux_alarm_out	Output	SYSMON VCCAUX-sensor alarm output.
ot_out	Output	Over-Temperature alarm output.
channel_out[5:0]	Outputs	Channel selection outputs. The ADC input MUX channel selection for the current ADC conversion is placed on these outputs at the end of an ADC conversion.

Table 2-1: **System Management Wizard I/O Signals *(Cont'd)***

Port	Direction	Description
eoc_out	Output	End of Conversion signal. This signal transitions to an active-High at the end of an ADC conversion when the measurement result is written to the status registers. For detailed information, see *UltraScale Architecture System Monitor Advanced Specification User Guide* (UG580) [Ref 1].
eos_out	Output	End of Sequence. This signal transitions to an active-High when the measurement data from the last channel in the Channel Sequencer is written to the status registers. For detailed information, see *UltraScale Architecture System Monitor Advanced Specification User Guide* (UG580) [Ref 1].
busy_out	Output	ADC busy signal. This signal transitions High during an ADC conversion. This signal transitions High for an extended period during calibration.
i2c_sclk	INOUT	I^2C clock signal.
i2c_sda	INOUT	I^2C serial data signal.
jtaglocked_out(2)	Output	Used to indicate that drp port has been locked by the JTAG or I^2C interface.
jtagmodified_out(2)	Output	Used to indicate that a JTAG or I^2C write to the drp has occurred.
jtagbusy_out(2)	Output	Used to indicate that a JTAG or I^2C drp transaction is in progress.
vbram_alarm_out	Output	SYSMON VBRAM sensor alarm output.
muxaddr_out[4:0]	Output	Use in external multiplexer mode to decode external MUX channel.
alarm_out	Output	Logic OR of alarms. Can be used to flag occurrence of any alarm.
s_axi_aclk	Input	AXI Clock.
s_axi_aresetn(2)	Input	AXI Reset, Active-Low
s_axi_awaddr[12:0]	Input	AXI Write address. The write address bus gives the address of the write transaction.
s_axi_awvalid	Input	Write address valid. This signal indicates that a valid write address and control information are available.
s_axi_awready	Output	Write address ready. This signal indicates that the slave is ready to accept an address and associated control signals.
s_axi_wdata[31:0]	Input	Write data.
s_axi_wstb[3:0]	Input	Write strobes. This signal indicates which byte lanes to update in memory.
s_axi_wvalid	Input	Write valid. This signal indicates that valid write data and strobes are available.
s_axi_wready	Output	Write ready. This signal indicates that the slave can accept the write data.

Table 2-1: **System Management Wizard I/O Signals *(Cont'd)***

Port	Direction	Description
s_axi_bresp[1:0]	Output	Write response. This signal indicates the status of the write transaction: • 00 = OKAY (normal response) • 10 = SLVERR (error condition) • 11 = DECERR (not issued by core)
s_axi_bvalid	Output	Write response valid. This signal indicates that a valid write response is available.
s_axi_bready	Input	Response ready. This signal indicates that the master can accept the response information.
s_axi_araddr[12:0]	Input	Read address. The read address bus gives the address of a read transaction.
s_axi_arvalid	Input	Read address valid. This signal indicates, when High, that the read address and control information is valid and remains stable until the address acknowledgement signal, s_axi_arready, is High.
s_axi_arready	Output	Read address ready. This signal indicates that the slave is ready to accept an address and associated control signals.
s_axi_rdata[31:0]	Output	Read data.
s_axi_rresp[1:0]	Output	Read response. This signal indicates the status of the read transfer. • 00 = OKAY (normal response) • 10 = SLVERR (error condition) • 11 = DECERR (not issued by core)
s_axi_rvalid	Output	Read valid. This signal indicates that the required read data is available and the read transfer can complete.
s_axi_rready	Input	Read ready. This signal indicates that the master can accept the read data and response information.
temp_out[9:0](3)	Output	10-bit temperature output bus for MIG. This should be connected to temperature input port of MIG.
ip2intc_irpt	Output	Interrupt Control Signal. This signal indicates, when High, that one of the selected interrupt, mentioned in the Interrupt Enable Register, occurred.
adc_data_master	Output	Direct data output of analog to digital converted value of Master SYSMON.(Available for Ultrascale plus devices)
adc_data_slave0	Output	Direct data output of analog to digital converted value of Slave0 SYSMON.(Available for Ultrascale plus devices)
adc_data_slave1	Output	Direct data output of analog to digital converted value of Slave1 SYSMON.(Available for Ultrascale plus devices)
SMBALERT	Output	Optional PMBus alert. When Low indicates a system fault that must be cleared using PMBus commands. Connect to SMBALERT_TS.(Available for Ultrascale plus devices)
vccpsintlp_alarm_out	Output	PS SYSMON VCCPSINTLP sensor alarm output. This port is available only for zynq Ultrascale plus devices

Table 2-1: **System Management Wizard I/O Signals *(Cont'd)***

Port	Direction	Description
vccpsintfp_alarm_out	Output	PS SYSMON VCCPSINTFP sensor alarm output. . This port is available only for zynq Ultrascale plus devices
vccpsaux_alarm_out	Output	PS SYSMON VCCPSAUX sensor alarm output. . This port is available only for zynq Ultrascale plus devices

Notes:

1. AXI4-Lite ports are available only with the AXI4-Lite interface.
2. DRP, JTAG, and reset_in ports are not available when AXI4-Lite interface is selected.
3. The temp_out port is available only when AXI4-Lite interface is enabled.

Register Space

The SYSMON functionality is configured through control registers. For more details, see control and status register information in *UltraScale Architecture System Monitor Advanced Specification User Guide* (UG580) [Ref 1].

Table 2-2 lists the attributes associated with these control registers. Control registers can be initialized using HDL by attaching HDL attributes to the SYSMON primitive instance and configuring them according to the information provided in Table 2-2. The control registers can also be initialized through the AXI4-Lite or DRP interfaces at runtime. The System Management Wizard simplifies the initialization of these control registers in the HDL instantiation by automatically configuring them to implement the operating behavior you specify in the Vivado® Integrated Design Environment (IDE).

Table 2-2: **SYSMON Attributes**

Attribute	Name	Control Reg Address	Description
INIT_40	Configuration Register 0	40h	SYSMON configuration registers. For detailed information, see *UltraScale Architecture System Monitor Advanced Specification User Guide* (UG580) [Ref 1].
INIT_41	Configuration Register 1	41h	
INIT_42	Configuration Register 2	42h	
INIT_43	Configuration Register 3	43h	
INIT_48 to INIT_4F	Sequence Registers	48h to 4Fh	Sequence registers used to program the Channel Sequencer function in the SYSMON. For detailed information, see *UltraScale Architecture System Monitor Advanced Specification User Guide* (UG580) [Ref 1].

Table 2-2: **SYSMON Attributes *(Cont'd)***

Attribute	Name	Control Reg Address	Description
INIT_50 to INIT_6F	Alarm Limits Registers	50h to 6Fh	Alarm threshold registers for the SYSMON alarm function. For detailed information, see *UltraScale Architecture System Monitor Advanced Specification User Guide* (UG580) [Ref 1].
SIM_MONITOR_ FILE	Simulation Analog Entry File	–	This is the text file that contains the analog input stimulus. This is used for simulation.

System Management Wizard Register Descriptions for AXI4-Lite Interface

AXI4-Lite address mapping to Hard Macro Register Address:

Bits Description:

0-1: Don't Care

2-9: SYSMON Macro Register Address

10: Always set to 1

11-12: Sysmon Slave Select Signal

00 ' Master SYSMON

01 ' Slave 0 SYSMON

10 ' Slave 1 SYSMON

Sysmon Slave Select signals decides to which SYSMON on the FPGA the AXI interface is communicating to.

Table 2-3 lists the System Management Wizard IP Core registers and corresponding addresses.

Table 2-3: **IP Core Registers**

Base Address + Offset (hex)	Register Name	Access Type	Description
System Management Wizard Local Register Grouping			
C_BASEADDR + 0x00	Software Reset Register (SRR)	W[1]	Software Reset Register
C_BASEADDR + 0x04	Status Register (SR)	R[2]	Status Register
C_BASEADDR + 0x08	Alarm Output Status Register (AOSR)	R[2]	Alarm Output Status Register
C_BASEADDR + 0x0C	CONVST Register (CONVSTR)	W[1]	• Bit[0] = ADC convert start register[3] • Bit[1] = Enable temperature update logic • Bit[17:2] = Wait cycle for temperature update
C_BASEADDR + 0x10	SYSMON Reset Register (SYSMONRR)	W[1]	SYSMON Hard Macro Reset Register
System Management Wizard Interrupt Controller Register Grouping			
C_BASEADDR + 0x5C	Global Interrupt Enable Register (GIER)	R/W	Global Interrupt Enable Register
C_BASEADDR + 0x60	IP Interrupt Status Register (IPISR)	R/TOW[4]	IP Interrupt Status Register
C_BASEADDR + 0x68	IP Interrupt Enable Register (IPIER)	R/W	IP Interrupt Enable Register
System Management Wizard Hard Macro Register Grouping[5]			
C_BASEADDR + 0x400	Temperature	R[6]	10-bit Most Significant Bit (MSB) justified result of on-device temperature measurement is stored in this register.
C_BASEADDR + 0x404	VCCINT	R[6]	The 10-bit MSB justified result of on-device VCCINT supply monitor measurement is stored in this register.
C_BASEADDR + 0x408	VCCAUX	R[6]	The 10-bit MSB justified result of on-device VCCAUX Data supply monitor measurement is stored in this register.

Table 2-3: **IP Core Registers *(Cont'd)***

Base Address + Offset (hex)	Register Name	Access Type	Description
C_BASEADDR + 0x40C	VP/VN	R/W[7]	• When read: The 10-bit MSB justified result of A/D conversion on the dedicated analog input channel (Vp/Vn) is stored in this register. • When written: Write to this register resets the SYSMON hard macro. No specific data is required.
C_BASEADDR + 0x410	VREFP	R[6]	The 10-bit MSB justified result of A/D conversion on the reference input VREFP is stored in this register.
C_BASEADDR + 0x414	VREFN	R[6]	The 10-bit MSB justified result of A/D conversion on the reference input VREFN is stored in this register.
C_BASEADDR + 0x418	VBRAM	R[6]	The 10-bit MSB justified result of A/D conversion on the reference input VBRAM is stored in this register.
C_BASEADDR + 0x41C	Undefined	N/A	These locations are unused and contain invalid data.
C_BASEADDR + 0x420	Supply Offset	R[6]	The calibration coefficient for the supply sensor offset is stored in this register.
C_BASEADDR + 0x424	ADC Offset	R[6]	The calibration coefficient for the ADC offset calibration is stored in this register.
C_BASEADDR + 0x428	Gain Error	R[6]	The calibration coefficient for the gain error is stored in this register.
C_BASEADDR + 0x434 [10]	VCC_PSINTLP	R	The 10-bit MSB justified result of A/D conversion on the reference input VCC_PSINTLP of Master SYSMON is Stored in this register.
C_BASEADDR + 0x438 [10]	VCC_PSINFP	R	The 10-bit MSB justified result of A/D conversion on the reference input VCC_PSINFP of Master SYSMON is stored in this register
C_BASEADDR + 0x43C [10]	VCC_PSAUX	R	The 10-bit MSB justified result of A/D conversion on the reference input VCC_PSAUX of Master SYSMON is stored in this register
C_BASEADDR + 0x440	VAUXP[0]/ VAUXN[0]	R[6]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 0 is stored in this register.
C_BASEADDR + 0x444	VAUXP[1]/ VAUXN[1]	R[6]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 1 is stored in this register.

Table 2-3: **IP Core Registers *(Cont'd)***

Base Address + Offset (hex)	Register Name	Access Type	Description
C_BASEADDR + 0x448	VAUXP[2]/ VAUXN[2]	R[6]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 2 is stored in this register.
C_BASEADDR + 0x44C	VAUXP[3]/ VAUXN[3]	R[6]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 3 is stored in this register.
C_BASEADDR + 0x450	VAUXP[4]/ VAUXN[4]	R[6]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 4 is stored in this register.
C_BASEADDR + 0x454	VAUXP[5]/ VAUXN[5]	R[6]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 5 is stored in this register.
C_BASEADDR + 0x458	VAUXP[6]/ VAUXN[6]	R[6]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 6 is stored in this register.
C_BASEADDR + 0x45C	VAUXP[7]/ VAUXN[7]	R[6]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 7 is stored in this register.
C_BASEADDR + 0x460	VAUXP[8]/ VAUXN[8]	R[6]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 8 is stored in this register.
C_BASEADDR + 0x464	VAUXP[9]/ VAUXN[9]	R[6]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 9 is stored in this register.
C_BASEADDR + 0x468	VAUXP[10]/ VAUXN[10]	R[6]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 10 is stored in this register.
C_BASEADDR + 0x46C	VAUXP[11]/ VAUXN[11]	R[6]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 11 is stored in this register.
C_BASEADDR + 0x470	VAUXP[12]/ VAUXN[12]	R[6]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 12 is stored in this register.
C_BASEADDR + 0x474	VAUXP[13]/ VAUXN[13]	R[6]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 13 is stored in this register.
C_BASEADDR + 0x478	VAUXP[14]/ VAUXN[14]	R[6]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 14 is stored in this register.
C_BASEADDR + 0x47C	VAUXP[15]/ VAUXN[15]	R[6]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 15 is stored in this register.

Table 2-3: **IP Core Registers *(Cont'd)***

Base Address + Offset (hex)	Register Name	Access Type	Description
C_BASEADDR + 0x480	Max Temp	R[6]	The 10-bit MSB justified maximum temperature measurement.
C_BASEADDR + 0x484	Max VCCINT	R[6]	The 10-bit MSB justified maximum VCCINT measurement.
C_BASEADDR + 0x488	Max VCCAUX	R[6]	The 10-bit MSB justified maximum VCCAUX measurement.
C_BASEADDR + 0x48C	Max VBRAM	R[6]	The 10-bit MSB justified maximum VBRAM measurement.
C_BASEADDR + 0x490	Min Temp	R[6]	The 10-bit MSB justified minimum temperature measurement
C_BASEADDR + 0x494	Min VCCINT	R[6]	The 10-bit MSB justified minimum VCCINT measurement
C_BASEADDR + 0x498	Min VCCAUX	R[6]	The 10-bit MSB justified minimum VCCAUX measurement.
C_BASEADDR + 0x49C	Min VBRAM	R[6]	The 10-bit MSB justified minimum VBRAM measurement.
C_BASEADDR + 0x4A0 [10]	Max VCC_PSINTLP	R	The 10-bit MSB justified maximum VCC_PSINTLP of Master SYSMON measurement.
C_BASEADDR + 0x4A4 [10]	Max VCC_PSINFP	R	The 10-bit MSB justified maximum VCC_PSINFP of Master SYSMON measurement.
C_BASEADDR + 0x4A8 [10]	Max VCC_PSAUX	R	The 10-bit MSB justified maximum VCC_PSAUX of Master SYSMON measurement.
C_BASEADDR + 0x4AC [10]	Undefined	N/A	These locations are unused and contain Invalid data.
C_BASEADDR + 0x4B0 [10]	Min VCC_PSINTLP	R	The 10-bit MSB justified minimum VCC_PSINTLP of Master SYSMON measurement.
C_BASEADDR + 0x4B4 [10]	Min VCC_PSINFP	R	The 10-bit MSB justified minimum VCC_PSINFP of Master SYSMON measurement.
C_BASEADDR + 0x4B8 [10]	MIn VCC_PSAUX	R	The 10-bit MSB justified minimums VCC_PSAUX of Master SYSMON measurement.
C_BASEADDR + 0x4E0	I2C Address	R	The I^2C address captured by initial conversion on V_p/V_n channel.

Table 2-3: **IP Core Registers *(Cont'd)***

Base Address + Offset (hex)	Register Name	Access Type	Description
C_BASEADDR + 0x4FC	Flag Register	R[6]	The 16-bit register gives general status information of ALARM, Over Temperature (OT), disable information of SYSMON and information about whether the SYSMON is using internal reference voltage or external reference voltage.
C_BASEADDR + 0x500	Configuration Register 0	R/W[8]	SYSMON Configuration Register 0.
C_BASEADDR + 0x504	Configuration Register 1	R/W	SYSMON Configuration Register 1.
C_BASEADDR + 0x508	Configuration Register 2	R/W	SYSMON Configuration Register 2.
C_BASEADDR + 0x50C	Configuration Register 3	R/W	SYSMON Configuration Register 3.
C_BASEADDR + 0x510	Test Register	N/A	SYSMON Test Register (For factory test only).
C_BASEADDR + 0x514	Analog Bus Register	N/A	Configuration register for the Analog Bus.
C_BASEADDR + 0x518	Sequence Register 8	R/W	Sequencer channel selection (Vuser0-3).
C_BASEADDR + 0x51C	Sequence Register 9	R/W	Sequencer average selection (Vuser0-3).
C_BASEADDR + 0x520	Sequence Register 0	R/W	SYSMON Sequence Register 0 (ADC channel selection).
C_BASEADDR + 0x524	Sequence Register 1	R/W	SYSMON Sequence Register 1 (ADC channel selection).
C_BASEADDR + 0x528	Sequence Register 2	R/W	SYSMON Sequence Register 2 (ADC channel averaging enable).
C_BASEADDR + 0x52C	Sequence Register 3	R/W	SYSMON Sequence Register 3 (ADC channel averaging enable).
C_BASEADDR + 0x530	Sequence Register 4	R/W	SYSMON Sequence Register 4 (ADC channel analog-input mode).
C_BASEADDR + 0x534	Sequence Register 5	R/W	SYSMON Sequence Register 5 (ADC channel analog-input mode).
C_BASEADDR + 0x538	Sequence Register 6	R/W	SYSMON Sequence Register 6 (ADC channel acquisition time).
C_BASEADDR + 0x53C	Sequence Register 7	R/W	SYSMON Sequence Register 7 (ADC channel acquisition time).
C_BASEADDR + 0x540	Alarm Threshold Register 0	R/W	The 10-bit MSB justified alarm threshold register 0 (Temperature Upper).

Table 2-3: **IP Core Registers *(Cont'd)***

Base Address + Offset (hex)	Register Name	Access Type	Description
C_BASEADDR + 0x544	Alarm Threshold Register 1	R/W	The 10-bit MSB justified alarm threshold register 1 (VCCINT Upper).
C_BASEADDR + 0x548	Alarm Threshold Register 2	R/W	The 10-bit MSB justified alarm threshold register 2 (VCCAUX Upper).
C_BASEADDR + 0x54C	Alarm Threshold Register 3	R/W(9)	The 10-bit MSB justified alarm threshold register 3 (OT Upper).
C_BASEADDR + 0x550	Alarm Threshold Register 4	R/W	The 10-bit MSB justified alarm threshold register 4 (Temperature Lower).
C_BASEADDR + 0x554	Alarm Threshold Register 5	R/W	The 10-bit MSB justified alarm threshold register 5 (VCCINT Lower).
C_BASEADDR + 0x558	Alarm Threshold Register 6	R/W	The 10-bit MSB justified alarm threshold register 6 (VCCAUX Lower).
C_BASEADDR + 0x55C	Alarm Threshold Register 7	R/W	The 10-bit MSB justified alarm threshold register 7 (OT Lower)
C_BASEADDR + 0x560	Alarm Threshold Register 8	R/W	The 10-bit MSB justified alarm threshold register 8 (V_{BRAM} Upper)
C_BASEADDR + 0x570	Alarm Threshold Register 12	R/W	The 10-bit MSB justified alarm threshold register 12 (V_{BRAM} Lower)
C_BASEADDR + 0x580	Alarm Threshold Register 16	R/W	The 10-bit MSB justified alarm threshold register 16 (V_{USER0} Upper)
C_BASEADDR + 0x584	Alarm Threshold Register 17	R/W	The 10-bit MSB justified alarm threshold register 17 (V_{USER1} Upper)
C_BASEADDR + 0x588	Alarm Threshold Register 18	R/W	The 10-bit MSB justified alarm threshold register 18 (V_{USER2} Upper)
C_BASEADDR + 0x58C	Alarm Threshold Register 19	R/W	The 10-bit MSB justified alarm threshold register 19 (V_{USER3} Upper)
C_BASEADDR + 0x5A0	Alarm Threshold Register 22	R/W	The 10-bit MSB justified alarm threshold register 14 (V_{USER0} Lower).
C_BASEADDR + 0x5A4	Alarm Threshold Register 23	R/W	The 10-bit MSB justified alarm threshold register 15 (V_{USER1} Lower).
C_BASEADDR + 0x5A8	Alarm Threshold Register 24	R/W	The 10-bit MSB justified alarm threshold register 16 (V_{USER2} Lower).
C_BASEADDR + 0x5AC	Alarm Threshold Register 25	R/W	The 10-bit MSB justified alarm threshold register 17 (V_{USER3} Lower).
C_BASEADDR + 0x600	V_{USER0}	R	The 10-bit MSB justified result of the on-chip V_{USER0} supply monitor measurement is stored at this location.
C_BASEADDR + 0x604	V_{USER1}	R	The 10-bit MSB justified result of the on-chip V_{USER1} supply monitor measurement is stored at this location.

Table 2-3: **IP Core Registers *(Cont'd)***

Base Address + Offset (hex)	Register Name	Access Type	Description
C_BASEADDR + 0x608	V_{USER2}	R	The 10-bit MSB justified result of the on-chip V_{USER2} supply monitor measurement is stored at this location.
C_BASEADDR + 0x60C	V_{USER3}	R	The 10-bit MSB justified result of the on-chip V_{USER3} supply monitor measurement is stored at this location.
C_BASEADDR + 0x680	Max V_{USER0}	R	Maximum V_{USER0} measurement recorded since power-up or the last System Monitor reset.
C_BASEADDR + 0x684	Max V_{USER1}	R	Maximum V_{USER1} measurement recorded since power-up or the last System Monitor reset.
C_BASEADDR + 0x688	Max V_{USER2}	R	Maximum V_{USER2} measurement recorded since power-up or the last System Monitor reset.
C_BASEADDR + 0x68C	Max V_{USER3}	R	Maximum V_{USER3} measurement recorded since power-up or the last System Monitor reset.
C_BASEADDR + 0x6A0	Min V_{USER0}	R	Minimum V_{USER0} measurement recorded since power-up or the last System Monitor reset.
C_BASEADDR + 0x6A4	Min V_{USER1}	R	Minimum V_{USER1} measurement recorded since power-up or the last System Monitor reset.
C_BASEADDR + 0x6A8	Min V_{USER2}	R	Minimum V_{USER2} measurement recorded since power-up or the last System Monitor reset.

Table 2-3: **IP Core Registers *(Cont'd)***

Base Address + Offset (hex)	Register Name	Access Type	Description
C_BASEADDR + 0x6Ac	Min V_{USER3}	R	Minimum V_{USER3} measurement recorded since power-up or the last System Monitor reset.

Notes:

1. Reading of this register returns an undefined value.
2. Writing into this register has no effect.
3. Used in event-driven sampling mode only.
4. TOW = Toggle On Write. Writing a 1 to a bit position within the register causes the corresponding bit position in the register to toggle.
5. These are 16-bit registers internal to SYSMON. These are mapped to the lower-half word boundary on 32-bit System Management Wizard IP core registers.
6. Writing to this SYSMON hard macro register is not allowed. The SYSMON hard macro data registers are 16 bits in width. The SYSMON hard macro specification guarantees the first 10 MSB bits accuracy; so only these bits are used for reference.
7. Writing to this register resets the SYSMON hard macro. No specific data pattern is required to reset the SYSMON hard macro.
8. Read the SYSMON User Guide, for setting the different bits available in configuration registers for UltraScale devices.
9. The OT upper register is a user-configurable register for the upper threshold level of temperature. If this register is left unconfigured, then the SYSMON considers 125°C as the upper threshold value for OT. While configuring this register, the last four bits must be set to 0011, that is, Alarm Threshold Register 3[3:0] = 0011. The upper 12 bits of this register are user configurable.
10. These registers are valid for Zynq UltraScale+ devices only.

System Management Wizard Local Register Grouping for AXI4-Lite Interface

It is expected that the System Management Wizard IP core registers are accessed in their preferred-access mode only. If a write attempt is made to read-only registers, there is no affect on register contents. If the write-only registers are read, the result is undefined data. All internal registers of the core must be accessed in 32-bit format. If there is any other kind of access (half-word or byte access) for local 32-bit registers, the transaction is completed with errors for the corresponding transaction.

Software Reset Register (SRR)

The Software Reset register permits you to reset the System Management Wizard IP core including the SYSMON hard macro output ports (except JTAG-related outputs) independently of other IP cores in the systems. To activate a software reset, write 0x0000_000A to the register. Any other access, read or write, has undefined results. The bit assignment in the Software Reset register is shown in Figure 2-2 and described in Table 2-4.

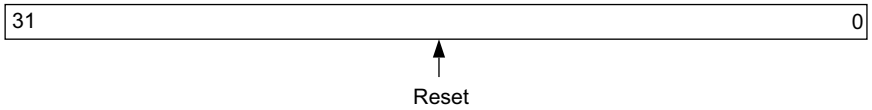

Figure 2-2: **Software Reset Register**

Table 2-4: **Software Reset Register Description (C_BASEADDR + 0x00)**

Bits	Name	Reset Value	Access Type	Description
31:0	Reset	N/A	W	The only allowed operation on this register is a write of 0x0000_000A, which resets the System Management Wizard IP Core. The reset is active only for 16 clock cycles.

Status Register (SR)

The Status register contains the System Management Wizard IP core channel status, EOC, EOS, and JTAG access signals. This register is read only. Any attempt to write the bits of the register is not able to change the bits. The Status Register bit definitions are shown in Figure 2-3 and explained in Table 2-5.

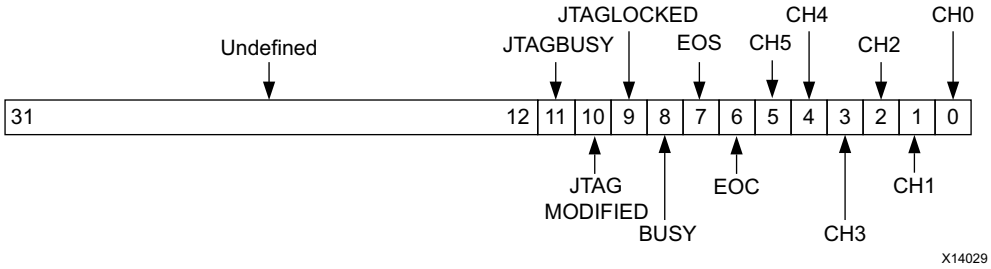

Figure 2-3: **Status Register**

Table 2-5: **Status Register (C_BASEADDR + 0x04)**

Bits	Name	Reset Value	Access Type	Description
31:12	Undefined	N/A	N/A	Undefined
11	JTAGBUSY	0	R	Used to indicate that a JTAG DRP or I^2C transaction is in progress.
10	JTAG MODIFIED	0	R	Used to indicate that a write to DRP through JTAG interface or I^2C transaction has occurred. This bit is cleared when a successful DRP read/write operation through the FPGA logic is performed. The DRP read/write through the FPGA logic fails, if JTAGLOCKED = 1
9	JTAG LOCKED	0	R	Used to indicate that a DRP port lock request has been made by the Joint Test Action Group (JTAG) interface.
8	BUSY	N/A	R	ADC busy signal. This signal transitions High during an ADC conversion.

Table 2-5: **Status Register (C_BASEADDR + 0x04) *(Cont'd)***

Bits	Name	Reset Value	Access Type	Description
7	EOS	N/A	R	End of Sequence. This signal transitions to an active-High when the measurement data from the last channel in the auto sequence is written to the status registers. This bit is cleared when a read operation is performed on status register.
6	EOC	N/A	R	End of Conversion signal. This signal transitions to an active-High at the end of an ADC conversion when the measurement is written to the SYSMON hard macro status register. This bit is cleared when a read operation is performed on status register.
5:0	CHANNEL [5:0]	N/A	R	Channel selection outputs. The ADC input MUX channel selection for the current ADC conversion is placed on these outputs at the end of an ADC conversion.

Alarm Output Status Register (AOSR)

The Alarm Output Status register contains all the alarm outputs for the System Management Wizard IP core. This register is read-only. Any attempt to write the bits of the register is not able to change the bits. The Alarm Output Status register bit definitions are shown in Figure 2-4 and explained in Table 2-6.

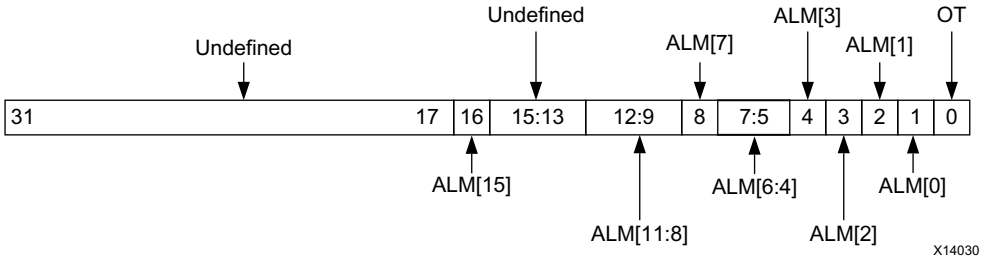

Figure 2-4: **Alarm Output Status Register**

Table 2-6: **Alarm Output Status Register (C_BASEADDR + 0x08)**

Bits	Name	Reset Value	Access Type	Description
31:17	Undefined	N/A	N/A	Undefined
16	ALM[15]	0	R	Logical ORing of ALARM bits 8 to 14. This is direct output from the SYSMON macro.
15:13	Undefined	N/A	N/A	Reserved
12:9	ALM[11:8]	0	R	Alarms for User Supplies 0-3
8	ALM[7]	0	R	Logical ORing of ALARM bits 0 to 6. This is direct output from the SYSMON macro.

Table 2-6: **Alarm Output Status Register (C_BASEADDR + 0x08) *(Cont'd)***

Bits	Name	Reset Value	Access Type	Description
7:5	ALM[6:4]	0	R	Reserved
4	ALM[3]	0	R	SYSMON VBRAM-Sensor Status. SYSMON VBRAM-sensor alarm output interrupt occurs when VBRAM exceeds user-defined threshold.
3	ALM[2]	0	R	SYSMON VCCAUX-Sensor Status. SYSMON VCCAUX-sensor alarm output interrupt occurs when VCCAUX exceeds user-defined threshold.
2	ALM[1]	0	R	SYSMON VCCINT-Sensor Status. SYSMON VCCINT-sensor alarm output interrupt occurs when VCCINT exceeds user-defined threshold.
1	ALM[0]	0	R	SYSMON Temperature-Sensor Status. SYSMON temperature-sensor alarm output interrupt occurs when device temperature exceeds user-defined threshold.
0	OT	0	R	SYSMON Over-Temperature Alarm Status. Over-Temperature alarm output interrupt occurs when the die temperature exceeds a factory set limit of 125°C.

CONVST Register (CONVSTR)

The CONVST register is used for initiating a new conversion in the event-driven sampling mode. The output of this register is logically ORed with the external CONVST input signal. This register also defines enable for the Temperature Bus update logic and the wait cycle count. The attempt to read this register results in undefined data. The CONVST Register bit definitions are shown in Figure 2-5 and explained in Table 2-7.

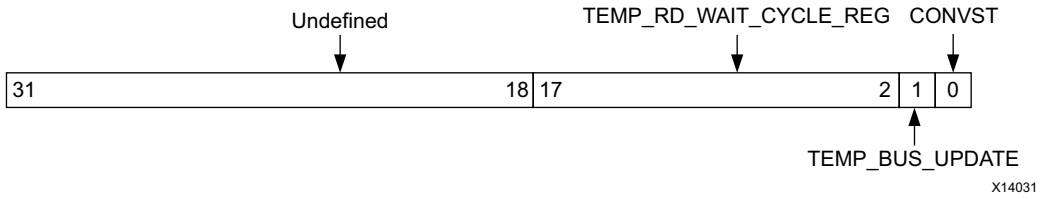

Figure 2-5: **CONVST Register**

Table 2-7: **CONVST Register (C_BASEADDR + 0x0C)**

Bits	Name	Reset Value	Access Type	Description
31:18	Undefined	N/A	N/A	Undefined
17:2	TEMP_RD_WAIT_CYCLE_REG	0x03E8	W	Wait cycle for temperature update. Temperature update logic waits for this count of the S_AXI_ACLK.

Table 2-7: **CONVST Register (C_BASEADDR + 0x0C) *(Cont'd)***

Bits	Name	Reset Value	Access Type	Description
1	TEMP_BUS_UPDATE	0	W	Enable temperature update logic enables the temperature read from SYSMON and updates of TEMP_OUT port.
0	CONVST	0	W	A rising edge on the CONVST input initiates start of ADC conversion in event-driven sampling mode. For the selected channel the CONVST bit in the register needs to be set to 1 and again reset to 0 to start a new conversion cycle. The conversion cycle ends with EOC bit going High.

SYSMON Reset Register

The SYSMON Reset register is used to reset only the SYSMON hard macro. As soon as the reset is released the ADC begins with a new conversion. If sequencing is enabled this conversion is the first in the sequence. This register resets the OT and ALM[n] output from the SYSMON hard macro. This register does not reset the interrupt registers if they are included in the design. Also any reset from the FPGA logic does not affect the RFI (Register File Interface) contents of SYSMON hard macro. The attempt to read this register results in undefined data. The SYSMON Reset register bit definitions are shown in Figure 2-6 and explained in Table 2-8.

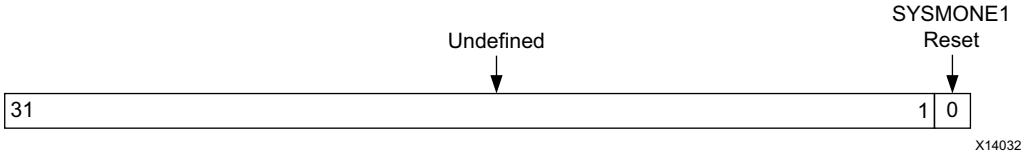

Figure 2-6: **SYSMON Reset Register**

Table 2-8: **SYSMON Reset Register (C_BASEADDR + 0x10)**

Bits	Name	Reset Value	Access Type	Description
31:1	Undefined	N/A	N/A	Undefined
0	SYSMON Reset	0	Write	Writing 1 to this bit position resets the SYSMON hard macro. The reset is released only after 0 is written to this register.

Interrupt Controller Register Grouping for AXI4-Lite Interface

The Interrupt Controller Module is included in the System Management Wizard IP core design when C_INCLUDE_INTR = 1. The System Management Wizard has several distinct interrupts that are sent to the Interrupt Controller Module, which is one of the submodules of System Management Wizard IP Core. The Interrupt Controller Module allows each

interrupt to be enabled independently (by the IP Interrupt Enable register (IPIER)). All the interrupt signals are rising-edge sensitive.

Interrupt registers are strictly 32-bit accessible. If byte/half-word or without byte enables access is made, the core behavior is not guaranteed.

The interrupt registers are in the Interrupt Controller Module. The System Management Wizard permits multiple conditions for an interrupt or an interrupt strobe which occurs only after the completion of a transfer.

Global Interrupt Enable Register (GIER)

The Global Interrupt Enable register is used to globally enable the final interrupt output from the Interrupt Controller as shown in Figure 2-7 and described in Table 2-9. This bit is a read/write bit and is cleared upon reset.

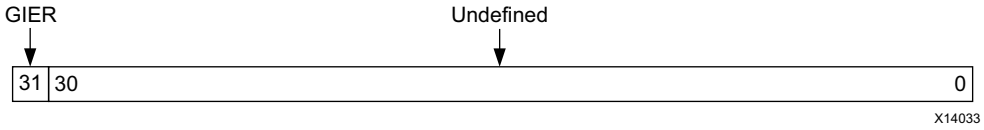

Figure 2-7: **Global Interrupt Enable Register (GIER)**

Table 2-9: **Global Interrupt Enable Register (GIER) Description (C_BASEADDR + 0x5C)**

Bits	Name	Reset Value	Access Type	Description
31	GIER	0	R/W	Global Interrupt Enable Register. It enables all individually enabled interrupts to be passed to the interrupt controller. • 0 = Disabled • 1 = Enabled
30:0	Undefine	N/A	N/A	Undefined.

IP Interrupt Status Register (IPISR)

Six unique interrupt conditions are possible in the System Management Wizard IP core. The IP Interrupt Status Register (IPISR) collects all the interrupt events. The Interrupt Controller has a register that can enable each interrupt independently. Bit assignment in the Interrupt register for a 32-bit data bus is shown in Figure 2-8 and described in Table 2-10. The interrupt register is a read/toggle on write register and by writing a 1 to a bit position within the register causes the corresponding bit position in the register to toggle. The interrupt bits in IPISR are updated soon after the interrupt is occurred. To see these

interrupts on the output pin ip2intc_irpt, respective bits in IPIER needs to be enabled. All register bits are cleared upon reset.

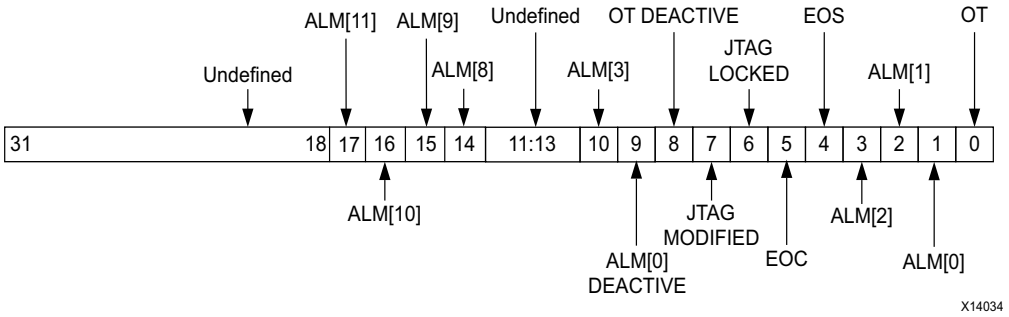

Figure 2-8: **IP Interrupt Status Register**

Table 2-10: **IP Interrupt Status Register (IPISR) Description (C_BASEADDR + 0x60)**

Bits	Name	Reset Value	Access Type	Description
31:18	Undefined	N/A	N/A	Undefined
17	ALM[11]	0	R/TOW(1)(2)	SYSMON VUSER3-Sensor Interrupt. The SYSMON VUSER3 sensor alarm output interrupt occurs when VUSER0 exceeds the user-defined threshold.
16	ALM[10]	0	R/TOW(1)(2)	SYSMON VUSER2-Sensor Interrupt. The SYSMON VUSER2-sensor alarm output interrupt occurs when VUSER2 exceeds the user-defined threshold.
15	ALM[9]	0	R/TOW(1)(2)	SYSMON VUSER1-Sensor Interrupt. The SYSMON VUSER1-sensor alarm output interrupt occurs when VUSER1 exceeds the user-defined threshold.
14	ALM[8]	0	R/TOW(1)(2)	SYSMON VUSER0-Sensor Interrupt. The SYSMON VUSER0-sensor alarm output interrupt occurs when VUSER0 exceeds the user-defined threshold.
11:13	ALM[4:6]	0	N/A	Undefined
10	ALM[3]	0	R/TOW(1)(2)	SYSMON VBRAM-Sensor Interrupt. SYSMON VBRAM-sensor alarm output interrupt occurs when VBRAM exceeds user-defined threshold.
9	ALM[0] Deactive	0	R/TOW	ALM[0] Deactive Interrupt. This signal indicates that the falling edge of the Over Temperature signal is detected. It is cleared by writing 1 to this bit position. The ALM[0] signal is generated locally from the core. This signal indicates that the SYSMON macro has deactivated the Over Temperature signal output.
8	OT Deactive	0	R/TOW(1)	OT Deactive Interrupt. This signal indicates that falling edge of the Over Temperature signal is detected. It is cleared by writing 1 to this bit position. The OT Deactive signal is generated locally from the core. This signal indicates that the SYSMON macro has deactivated the Over Temperature signal output.

Table 2-10: **IP Interrupt Status Register (IPISR) Description (C_BASEADDR + 0x60)** ***(Cont'd)***

Bits	Name	Reset Value	Access Type	Description
7	JTAG MODIFIED	0	R/TOW[(1)(2)]	JTAGMODIFIED Interrupt. This signal indicates that a write to DRP through the JTAG interface has occurred. It is cleared by writing 1 to this bit position.
6	JTAG LOCKED	0	R/TOW[(1)(2)]	JTAGLOCKED Interrupt. This signal is used to indicate that a DRP port lock request has been made by the Joint Test Action Group (JTAG) interface.
5	EOC	N/A	R/TOW[(1)(2)]	End of Conversion Signal Interrupt. This signal transitions to an active-High at the end of an ADC conversion when the measurement is written to the SYSMON hard macro status register.
4	EOS	N/A	R/TOW[(1)(2)]	End of Sequence Interrupt. This signal transitions to an active-High when the measurement data from the last channel in the auto sequence is written to the status registers.
3	ALM[2]	0	R/TOW[(1)(2)]	SYSMON V_{CCAUX}-Sensor Interrupt. SYSMON V_{CCAUX}-sensor alarm output interrupt occurs when V_{CCAUX} exceeds the user-defined threshold.
2	ALM[1]	0	R/TOW[(1)(2)]	SYSMON V_{CCINT}-Sensor Interrupt. SYSMON V_{CCINT}-sensor alarm output interrupt occurs when V_{CCINT} exceeds the user-defined threshold.
1	ALM[0]	0	R/TOW[(1)(2)]	SYSMON Temperature-Sensor Interrupt. SYSMON temperature-sensor alarm output interrupt occurs when device temperature exceeds the user-defined threshold.
0	OT	0	R/TOW[(1)(2)]	Over-Temperature Alarm Interrupt. Over-Temperature alarm output interrupt occurs when the die temperature exceeds a factory set limit of 125 °C.

Notes:

1. TOW = Toggle On Write. Writing a 1 to a bit position within the register causes the corresponding bit position in the register to toggle.
2. This interrupt signal is directly generated from the SYSMON hard macro.

IP Interrupt Enable Register (IPIER)

The Interrupt Enable Register (IPIER) register allows the system interrupt output (ip2intc_irpt) to be active. This interrupt is generated if an active bit in the IPISR register corresponds to an enabled bit in the IPIER register. The IPIER register has an enable bit for each defined bit of the IPISR as shown in Figure 2-9 and described in Table 2-11. IPIER acts as a gate between IPISR and the output interrupt port (ip2intc_irpt). All bits are cleared upon reset.

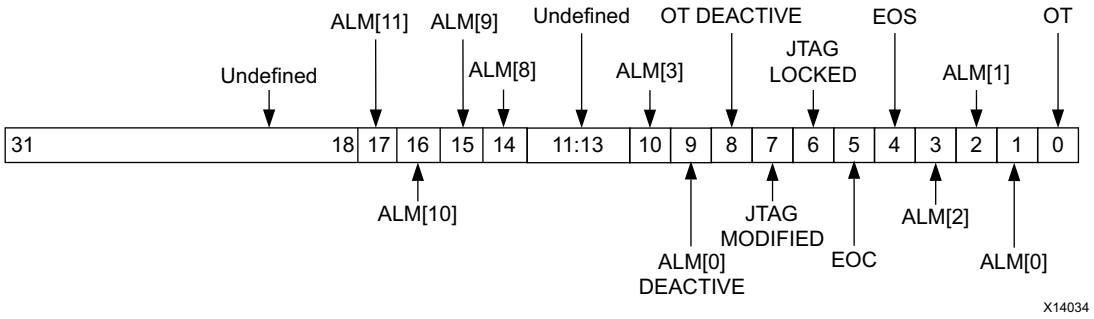

Figure 2-9: **IP Interrupt Enable Register (IPIER)**

Table 2-11: **IP Interrupt Enable Register (IPIER) Description (C_BASEADDR + 0x68)**

Bits	Name	Reset Value	Access Type	Description
31:18	Undefined	N/A	N/A	Undefined
17	ALM[11]	0	R/W	SYSMON VUSER3-Sensor Interrupt • 0 = Disabled • 1 = Enabled
16	ALM[10]	0	R/W	SYSMON VUSER2-Sensor Interrupt • 0 = Disabled • 1 = Enabled
15	ALM[9]	0	R/W	SYSMON VUSER1-Sensor Interrupt • 0 = Disabled • 1 = Enabled
14	ALM[8]	0	R/W	SYSMON VUSER0-Sensor Interrupt • 0 = Disabled • 1 = Enabled
11:13	ALM[4:6]	0	N/A	Undefined.
10	ALM[3]	0	R/W	SYSMON VBRAM-Sensor Interrupt • 0 = Disabled • 1 = Enabled
9	ALM[0] Deactive	0	R/W	ALM[0] Deactive Interrupt • 0 = Disabled • 1 = Enabled
8	OT Deactive	0	R/W	OT Deactive Interrupt • 0 = Disabled • 1 = Enabled
7	JTAG MODIFIED	0	R/W	JTAGMODIFIED Interrupt • 0 = Disabled • 1 = Enabled
6	JTAG LOCKED	0	R/W	JTAGLOCKED Interrupt • 0 = Disabled • 1 = Enabled

Table 2-11: **IP Interrupt Enable Register (IPIER) Description (C_BASEADDR + 0x68) *(Cont'd)***

Bits	Name	Reset Value	Access Type	Description
5	EOC	0	R/W	End of Conversion Signal Interrupt • 0 = Disabled • 1 = Enabled
4	EOS	0	R/W	End of Sequence Interrupt • 0 = Disabled • 1 = Enabled
3	ALM[2]	0	R/W	SYSMON V_{CCAUX}-Sensor Interrupt • 0 = Disabled • 1 = Enabled
2	ALM[1]	0	R/W	SYSMON V_{CCINT}-Sensor Interrupt • 0 = Disabled • 1 = Enabled
1	ALM[0]	0	R/W	SYSMON Temperature-Sensor Interrupt • 0 = Disabled • 1 = Enabled
0	OT	0	R/W	Over-Temperature Alarm Interrupt • 0 = Disabled • 1 = Enabled

Locally Generated Interrupt Bits in IPIER and IPISR

The interrupt bits ranging from Bit[16] to Bit[0] in IPISR as well as IPIER are direct output signals of the SYSMON hard macro. The signals like OT Deactive (Bit[8]), ALM[0] Deactive (Bit[9]), are locally generated in the core. These interrupts are generated on the falling edge of the Over Temperature and AML[0] signals. The falling edge of these signals can be used in controlling external things like controlling the fan or air-conditioning of the system.

Hard Macro Register (DRP Register) Grouping for AXI4-Lite Interface

The SYSMON hard macro register set consists of all the registers present in the SYSMON hard macro on 7 series FPGAs. The addresses of these registers are shown in Table 2-3. Because these registers are 16 bits wide but the processor data bus is 32 bits wide, the hard macro register data resides on the lower 16 bits of the 32-bit data bus. See Figure 2-10.

The 10-bit MSB aligned A/D converted value of different channels from SYSMON hard macro are left-shifted and reside from bit position 15 to 6 of the processor data bus. The remaining bit positions from 5 to 0 should be ignored while considering the ADC data for different channels. Along with 16-bit data, the JTAGMODIFIED and JTAGLOCKED bits are passed that can be used by the software driver application for determining the validity of the DRP read data.

The JTAGMODIFIED bit is cleared when a DRP read/write operation through the FPGA logic is successful. If JTAGLOCKED = 1, a DRP read/write through the FPGA logic fails. The JTAGLOCKED signal is independently controlled through JTAG TAP. These SYSMON hard macro registers should be accessed in their preferred access-mode only. The System Management Wizard IP core is not able to differentiate any non-preferred access to the SYSMON hard macro registers.

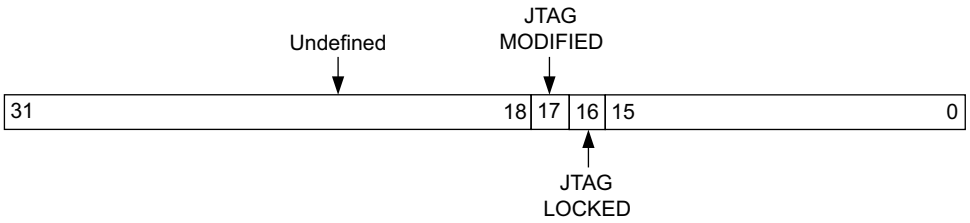

Figure 2-10: **DRP Register**

DRP registers are accessed as part of the core local registers.

IMPORTANT: *These registers must be accessed through the core local registers. Any attempt to access these registers in byte or half-word method returns an error response from core.*

Stacked Silicon Interconnect (SSI) Slave Address Map

When you access these addresses, the control is switched to slave SYSMON, and all the control and status signals are mapped to this primitive. It is recommended to switch between master and slave SYSMON address map after getting an EOC/EOS interrupt (for AXI4-Lite) or EOC/EOS pulse (for DRP).

Figure 2-11 shows the logic to switch between two SYSMON primitives.

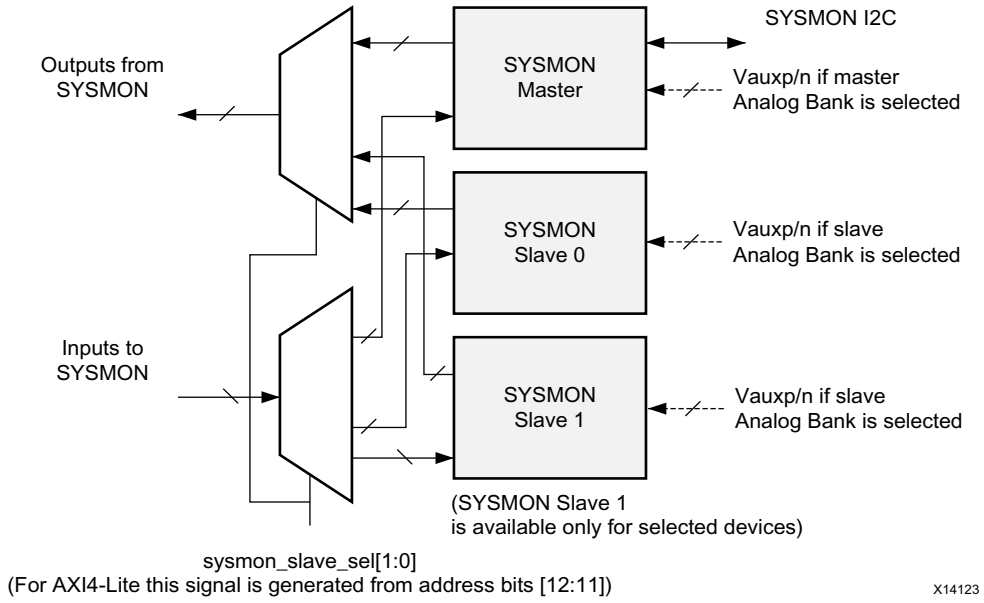

Figure 2-11: **Switching Between Two/Three SYSMONE**

Table 2-12: **IP Core Registers for Slave 0 SYSMON**

Base Address + Offset (hex)	Register Name	Access Type	Description
System Management Wizard Hard Macro Register Grouping[1]			
C_BASEADDR + 0xC00	Temperature	R[2]	10-bit Most Significant Bit (MSB) justified result of on-device temperature measurement is stored in this register.
C_BASEADDR + 0xC04	VCCINT	R[2]	The 10-bit MSB justified result of on-device VCCINT supply monitor measurement is stored in this register.
C_BASEADDR + 0xC08	VCCAUX	R[2]	The 10-bit MSB justified result of on-device VCCAUX Data supply monitor measurement is stored in this register.
C_BASEADDR + 0xC0C	VP/VN	R/W[3]	• When read: The 10-bit MSB justified result of A/D conversion on the dedicated analog input channel (Vp/Vn) is stored in this register. • When written: Write to this register resets the SYSMON hard macro. No specific data is required.
C_BASEADDR + 0xC10	VREFP	R[2]	The 10-bit MSB justified result of A/D conversion on the reference input VREFP is stored in this register.
C_BASEADDR + 0xC14	VREFN	R[2]	The 10-bit MSB justified result of A/D conversion on the reference input VREFN is stored in this register.
C_BASEADDR + 0xC18	VBRAM	R[2]	The 10-bit MSB justified result of A/D conversion on the reference input VBRAM is stored in this register.

Table 2-12: **IP Core Registers for Slave 0 SYSMON *(Cont'd)***

Base Address + Offset (hex)	Register Name	Access Type	Description
C_BASEADDR + 0xC1C	Undefined	N/A	These locations are unused and contain invalid data.
C_BASEADDR + 0xC20	Supply Offset	R[2]	The calibration coefficient for the supply sensor offset is stored in this register.
C_BASEADDR + 0xC24	ADC Offset	R[2]	The calibration coefficient for the ADC offset calibration is stored in this register.
C_BASEADDR + 0xC28	Gain Error	R[2]	The calibration coefficient for the gain error is stored in this register.
C_BASEADDR + 0xC34 [6]	VCC_PSINTLP	R	The 10-bit MSB justified result of A/D conversion on the reference input VCC_PSINTLP of Slave0 SYSMON is Stored in this register.
C_BASEADDR + 0xC38 [6]	VCC_PSINFP	R	The 10-bit MSB justified result of A/D conversion on the reference input VCC_PSINFP of Slave0 SYSMON is stored in this register
C_BASEADDR + 0xC3C [6]	VCC_PSAUX	R	The 10-bit MSB justified result of A/D conversion on the reference input VCC_PSAUX of Slave0 SYSMON is stored in this register
C_BASEADDR + 0xC40	VAUXP[0]/ VAUXN[0]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 0 is stored in this register.
C_BASEADDR + 0xC44	VAUXP[1]/ VAUXN[1]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 1 is stored in this register.
C_BASEADDR + 0xC48	VAUXP[2]/ VAUXN[2]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 2 is stored in this register.
C_BASEADDR + 0xC4C	VAUXP[3]/ VAUXN[3]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 3 is stored in this register.
C_BASEADDR + 0xC50	VAUXP[4]/ VAUXN[4]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 4 is stored in this register.
C_BASEADDR + 0xC54	VAUXP[5]/ VAUXN[5]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 5 is stored in this register.
C_BASEADDR + 0xC58	VAUXP[6]/ VAUXN[6]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 6 is stored in this register.
C_BASEADDR + 0xC5C	VAUXP[7]/ VAUXN[7]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 7 is stored in this register.
C_BASEADDR + 0xC60	VAUXP[8]/ VAUXN[8]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 8 is stored in this register.
C_BASEADDR + 0xC64	VAUXP[9]/ VAUXN[9]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 9 is stored in this register.
C_BASEADDR + 0xC68	VAUXP[10]/ VAUXN[10]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 10 is stored in this register.

Table 2-12: **IP Core Registers for Slave 0 SYSMON *(Cont'd)***

Base Address + Offset (hex)	Register Name	Access Type	Description
C_BASEADDR + 0xC6C	VAUXP[11]/ VAUXN[11]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 11 is stored in this register.
C_BASEADDR + 0xC70	VAUXP[12]/ VAUXN[12]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 12 is stored in this register.
C_BASEADDR + 0xC74	VAUXP[13]/ VAUXN[13]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 13 is stored in this register.
C_BASEADDR + 0xC78	VAUXP[14]/ VAUXN[14]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 14 is stored in this register.
C_BASEADDR + 0xC7C	VAUXP[15]/ VAUXN[15]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 15 is stored in this register.
C_BASEADDR + 0xC80	Max Temp	R[2]	The 10-bit MSB justified maximum temperature measurement.
C_BASEADDR + 0xC84	Max VCCINT	R[2]	The 10-bit MSB justified maximum VCCINT measurement.
C_BASEADDR + 0xC88	Max VCCAUX	R[2]	The 10-bit MSB justified maximum VCCAUX measurement.
C_BASEADDR + 0xC8C	Max VBRAM	R[2]	The 10-bit MSB justified maximum VBRAM measurement.
C_BASEADDR + 0xC90	Min Temp	R[2]	The 10-bit MSB justified minimum temperature measurement
C_BASEADDR + 0xC94	Min VCCINT	R[2]	The 10-bit MSB justified minimum VCCINT measurement
C_BASEADDR + 0xC98	Min VCCAUX	R[2]	The 10-bit MSB justified minimum VCCAUX measurement.
C_BASEADDR + 0xC9C	Min VBRAM	R[2]	The 10-bit MSB justified minimum VBRAM measurement.
C_BASEADDR + 0xCA0 [6]	Max VCC_PSINTLP	R	The 10-bit MSB justified maximum VCC_PSINTLP of Slave0 SYSMON measurement.
C_BASEADDR + 0xCA4 [6]	Max VCC_PSINFP	R	The 10-bit MSB justified maximum VCC_PSINFP of Slave0 SYSMON measurement.
C_BASEADDR + 0xCA8 [6]	Max VCC_PSAUX	R	The 10-bit MSB justified maximum VCC_PSAUX of Slave0 SYSMON measurement.
C_BASEADDR + 0xCAC[6]	Undefined	N/A	These locations are unused and contain Invalid data.
C_BASEADDR + 0xCB0 [6]	Min VCC_PSINTLP	R	The 10-bit MSB justified minimum VCC_PSINTLP of Slave0 SYSMON measurement.
sC_BASEADDR + 0xCB4 [6]	Min VCC_PSINFP	R	The 10-bit MSB justified minimum VCC_PSINFP of Slave0 SYSMON measurement.

Table 2-12: **IP Core Registers for Slave 0 SYSMON *(Cont'd)***

Base Address + Offset (hex)	Register Name	Access Type	Description
C_BASEADDR + 0xCB8 (6)	MIn VCC_PSAUX	R	The 10-bit MSB justified minimums VCC_PSAUX of Master SYSMON measurement.
C_BASEADDR + 0xCFC	Flag Register	R(2)	The 16-bit register gives general status information of ALARM, Over Temperature (OT), disable information of SYSMON and information about whether the SYSMON is using internal reference voltage or external reference voltage.
C_BASEADDR + 0xD00	Configuration Register 0	R/W(4)	SYSMON Configuration Register 0.
C_BASEADDR + 0xD04	Configuration Register 1	R/W	SYSMON Configuration Register 1.
C_BASEADDR + 0xD08	Configuration Register 2	R/W	SYSMON Configuration Register 2.
C_BASEADDR + 0xD0C	Configuration Register 3	R/W	SYSMON Configuration Register 3.
C_BASEADDR + 0xD10	Test Register	N/A	SYSMON Test Register (For factory test only).
C_BASEADDR + 0xD14	Analog Bus Register	N/A	Configuration register for the Analog Bus.
C_BASEADDR + 0xD18	Sequence Register 8	R/W	Sequencer channel selection (Vuser0-3).
C_BASEADDR + 0xD1C	Sequence Register 9	R/W	Sequencer average selection (Vuser0-3).
C_BASEADDR + 0xD20	Sequence Register 0	R/W	SYSMON Sequence Register 0 (ADC channel selection).
C_BASEADDR + 0xD24	Sequence Register 1	R/W	SYSMON Sequence Register 1 (ADC channel selection).
C_BASEADDR + 0xD28	Sequence Register 2	R/W	SYSMON Sequence Register 2 (ADC channel averaging enable).
C_BASEADDR + 0xD2C	Sequence Register 3	R/W	SYSMON Sequence Register 3 (ADC channel averaging enable).
C_BASEADDR + 0xD30	Sequence Register 4	R/W	SYSMON Sequence Register 4 (ADC channel analog-input mode).
C_BASEADDR + 0xD34	Sequence Register 5	R/W	SYSMON Sequence Register 5 (ADC channel analog-input mode).
C_BASEADDR + 0xD38	Sequence Register 6	R/W	SYSMON Sequence Register 6 (ADC channel acquisition time).
C_BASEADDR + 0xD3C	Sequence Register 7	R/W	SYSMON Sequence Register 7 (ADC channel acquisition time).

Table 2-12: **IP Core Registers for Slave 0 SYSMON *(Cont'd)***

Base Address + Offset (hex)	Register Name	Access Type	Description
C_BASEADDR + 0xD40	Alarm Threshold Register 0	R/W	The 10-bit MSB justified alarm threshold register 0 (Temperature Upper).
C_BASEADDR + 0xD44	Alarm Threshold Register 1	R/W	The 10-bit MSB justified alarm threshold register 1 (V_{CCINT} Upper).
C_BASEADDR + 0xD48	Alarm Threshold Register 2	R/W	The 10-bit MSB justified alarm threshold register 2 (V_{CCAUX} Upper).
C_BASEADDR + 0xD4C	Alarm Threshold Register 3	R/W(5)	The 10-bit MSB justified alarm threshold register 3 (OT Upper).
C_BASEADDR + 0xD50	Alarm Threshold Register 4	R/W	The 10-bit MSB justified alarm threshold register 4 (Temperature Lower).
C_BASEADDR + 0xD54	Alarm Threshold Register 5	R/W	The 10-bit MSB justified alarm threshold register 5 (V_{CCINT} Lower).
C_BASEADDR + 0xD58	Alarm Threshold Register 6	R/W	The 10-bit MSB justified alarm threshold register 6 (V_{CCAUX} Lower).
C_BASEADDR + 0xD5C	Alarm Threshold Register 7	R/W	The 10-bit MSB justified alarm threshold register 7 (OT Lower)
C_BASEADDR + 0xD60	Alarm Threshold Register 8	R/W	The 10-bit MSB justified alarm threshold register 8 (V_{BRAM} Upper)
C_BASEADDR + 0xD70	Alarm Threshold Register 12	R/W	The 10-bit MSB justified alarm threshold register 12 (V_{BRAM} Lower)
C_BASEADDR + 0xD80	Alarm Threshold Register 16	R/W	The 10-bit MSB justified alarm threshold register 16 (V_{USER0} Upper)
C_BASEADDR + 0xD84	Alarm Threshold Register 17	R/W	The 10-bit MSB justified alarm threshold register 17 (V_{USER1} Upper)
C_BASEADDR + 0xD88	Alarm Threshold Register 18	R/W	The 10-bit MSB justified alarm threshold register 18 (V_{USER2} Upper)
C_BASEADDR + 0xD8C	Alarm Threshold Register 19	R/W	The 10-bit MSB justified alarm threshold register 19 (V_{USER3} Upper)

Table 2-12: **IP Core Registers for Slave 0 SYSMON *(Cont'd)***

Base Address + Offset (hex)	Register Name	Access Type	Description
C_BASEADDR + 0xDA0	Alarm Threshold Register 22	R/W	The 10-bit MSB justified alarm threshold register 14 (V_{USER0} Lower).
C_BASEADDR + 0xDA4	Alarm Threshold Register 23	R/W	The 10-bit MSB justified alarm threshold register 15 (V_{USER1} Lower).
C_BASEADDR + 0xDA8	Alarm Threshold Register 24	R/W	The 10-bit MSB justified alarm threshold register 16 (V_{USER2} Lower).
C_BASEADDR + 0xDAC	Alarm Threshold Register 25	R/W	The 10-bit MSB justified alarm threshold register 17 (V_{USER3} Lower).
C_BASEADDR + 0xE00	V_{USER0}	R	The 10-bit MSB justified result of the on-chip V_{USER0} supply monitor measurement is stored at this location.
C_BASEADDR + 0xE04	V_{USER1}	R	The 10-bit MSB justified result of the on-chip V_{USER1} supply monitor measurement is stored at this location.
C_BASEADDR + 0xE08	V_{USER2}	R	The 10-bit MSB justified result of the on-chip V_{USER2} supply monitor measurement is stored at this location.
C_BASEADDR + 0xE0C	V_{USER3}	R	The 10-bit MSB justified result of the on-chip V_{USER3} supply monitor measurement is stored at this location.
C_BASEADDR + 0xE80	Max V_{USER0}	R	Maximum V_{USER0} measurement recorded since power-up or the last System Monitor reset.
C_BASEADDR + 0xE84	Max V_{USER1}	R	Maximum V_{USER1} measurement recorded since power-up or the last System Monitor reset.
C_BASEADDR + 0xE88	Max V_{USER2}	R	Maximum V_{USER2} measurement recorded since power-up or the last System Monitor reset.
C_BASEADDR + 0xE8C	Max V_{USER3}	R	Maximum V_{USER3} measurement recorded since power-up or the last System Monitor reset.
C_BASEADDR + 0xEA0	Min V_{USER0}	R	Minimum V_{USER0} measurement recorded since power-up or the last System Monitor reset.
C_BASEADDR + 0xEA4	Min V_{USER1}	R	Minimum V_{USER1} measurement recorded since power-up or the last System Monitor reset.
C_BASEADDR + 0xEA8	Min V_{USER2}	R	Minimum V_{USER2} measurement recorded since power-up or the last System Monitor reset.

Table 2-12: **IP Core Registers for Slave 0 SYSMON *(Cont'd)***

Base Address + Offset (hex)	Register Name	Access Type	Description
C_BASEADDR + 0xEAC	Min V_{USER3}	R	Minimum V_{USER3} measurement recorded since power-up or the last System Monitor reset.

Notes:

1. These are 16-bit registers internal to SYSMON. These are mapped to the lower-half word boundary on 32-bit System Management Wizard IP core registers.
2. Writing to this SYSMON hard macro register is not allowed. The SYSMON hard macro data registers are 16 bits in width. The SYSMON hard macro specification guarantees the first 10 MSB bits accuracy; so only these bits are used for reference.
3. Writing to this register resets the SYSMON hard macro. No specific data pattern is required to reset the SYSMON hard macro.
4. Read the SYSMON User Guide, for setting the different bits available in configuration registers for UltraScale devices.
5. The OT upper register is a user-configurable register for the upper threshold level of temperature. If this register is left unconfigured, then the SYSMON considers 125°C as the upper threshold value for OT. While configuring this register, the last four bits must be set to 0011, that is, Alarm Threshold Register 3[3:0] = 0011. The upper 12 bits of this register are user configurable.
6. These registers are valid for Zynq UltraScale+ devices only.

Table 2-13: **IP Core Registers for Slave 1 SYSMON**

Base Address + Offset (hex)	Register Name	Access Type	Description
System Management Wizard Hard Macro Register Grouping(1)			
C_BASEADDR + 0x1400	Temperature	R(2)	10-bit Most Significant Bit (MSB) justified result of on-device temperature measurement is stored in this register.
C_BASEADDR + 0x1404	VCCINT	R(2)	The 10-bit MSB justified result of on-device VCCINT supply monitor measurement is stored in this register.
C_BASEADDR + 0x1408	VCCAUX	R(2)	The 10-bit MSB justified result of on-device VCCAUX Data supply monitor measurement is stored in this register.
C_BASEADDR + 0x140C	VP/VN	R/W(3)	• When read: The 10-bit MSB justified result of A/D conversion on the dedicated analog input channel (Vp/Vn) is stored in this register. • When written: Write to this register resets the SYSMON hard macro. No specific data is required.
C_BASEADDR + 0x1410	VREFP	R(2)	The 10-bit MSB justified result of A/D conversion on the reference input VREFP is stored in this register.
C_BASEADDR + 0x1414	VREFN	R(2)	The 10-bit MSB justified result of A/D conversion on the reference input VREFN is stored in this register.
C_BASEADDR + 0x1418	VBRAM	R(2)	The 10-bit MSB justified result of A/D conversion on the reference input VBRAM is stored in this register.
C_BASEADDR + 0x141C	Undefined	N/A	These locations are unused and contain invalid data.

Table 2-13: **IP Core Registers for Slave 1 SYSMON *(Cont'd)***

Base Address + Offset (hex)	Register Name	Access Type	Description
C_BASEADDR + 0x1420	Supply Offset	R[2]	The calibration coefficient for the supply sensor offset is stored in this register.
C_BASEADDR + 0x1424	ADC Offset	R[2]	The calibration coefficient for the ADC offset calibration is stored in this register.
C_BASEADDR + 0x1428	Gain Error	R[2]	The calibration coefficient for the gain error is stored in this register.
C_BASEADDR + 0x1434	VCC_PSINTLP	R	The 10-bit MSB justified result of A/D conversion on the reference input VCC_PSINTLP of Slave1 SYSMON is stored in this register.
C_BASEADDR + 0x1438[6]	VCC_PSINFP	R	The 10-bit MSB justified result of A/D conversion on the reference input VCC_PSINFP of Slave1 SYSMON is stored in this register.
C_BASEADDR + 0x143C[6]	VCC_PSAUX	R	The 10-bit MSB justified result of A/D conversion on the reference input VCC_PSAUX of Slave1 SYSMON is stored in this register.
C_BASEADDR + 0x1440[6]	VAUXP[0]/ VAUXN[0]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 0 is stored in this register.
C_BASEADDR + 0x1444	VAUXP[1]/ VAUXN[1]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 1 is stored in this register.
C_BASEADDR + 0x1448	VAUXP[2]/ VAUXN[2]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 2 is stored in this register.
C_BASEADDR + 0x144C	VAUXP[3]/ VAUXN[3]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 3 is stored in this register.
C_BASEADDR + 0x1450	VAUXP[4]/ VAUXN[4]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 4 is stored in this register.
C_BASEADDR + 0x1454	VAUXP[5]/ VAUXN[5]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 5 is stored in this register.
C_BASEADDR + 0x1458	VAUXP[6]/ VAUXN[6]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 6 is stored in this register.
C_BASEADDR + 0x145C	VAUXP[7]/ VAUXN[7]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 7 is stored in this register.
C_BASEADDR + 0x1460	VAUXP[8]/ VAUXN[8]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 8 is stored in this register.
C_BASEADDR + 0x1464	VAUXP[9]/ VAUXN[9]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 9 is stored in this register.
C_BASEADDR + 0x1468	VAUXP[10]/ VAUXN[10]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 10 is stored in this register.
C_BASEADDR + 0x146C	VAUXP[11]/ VAUXN[11]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 11 is stored in this register.
C_BASEADDR + 0x1470	VAUXP[12]/ VAUXN[12]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 12 is stored in this register.

Table 2-13: **IP Core Registers for Slave 1 SYSMON *(Cont'd)***

Base Address + Offset (hex)	Register Name	Access Type	Description
C_BASEADDR + 0x1474	VAUXP[13]/ VAUXN[13]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 13 is stored in this register.
C_BASEADDR + 0x1478	VAUXP[14]/ VAUXN[14]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 14 is stored in this register.
C_BASEADDR + 0x147C	VAUXP[15]/ VAUXN[15]	R[2]	The 10-bit MSB justified result of A/D conversion on the auxiliary analog input 15 is stored in this register.
C_BASEADDR + 0x1480	Max Temp	R[2]	The 10-bit MSB justified maximum temperature measurement.
C_BASEADDR + 0x1484	Max VCCINT	R[2]	The 10-bit MSB justified maximum VCCINT measurement.
C_BASEADDR + 0x1488	Max VCCAUX	R[2]	The 10-bit MSB justified maximum VCCAUX measurement.
C_BASEADDR + 0x148C	Max VBRAM	R[2]	The 10-bit MSB justified maximum VBRAM measurement.
C_BASEADDR + 0x1490	Min Temp	R[2]	The 10-bit MSB justified minimum temperature measurement
C_BASEADDR + 0x1494	Min VCCINT	R[2]	The 10-bit MSB justified minimum VCCINT measurement
C_BASEADDR + 0x1498	Min VCCAUX	R[2]	The 10-bit MSB justified minimum VCCAUX measurement.
C_BASEADDR + 0x149C	Min VBRAM	R[2]	The 10-bit MSB justified minimum VBRAM measurement.
C_BASEADDR + 0x14A0[6]	Max VCC_PSINTLP	R	The 10-bit MSB justified maximum VCC_PSINTLP of Slave1SYSMON measurement.
C_BASEADDR + 0x14A4[6]	Max VCC_PSINFP	R	The 10-bit MSB justified maximum VCC_PSINFP of Slave1SYSMON measurement.
C_BASEADDR + 0x14A8[6]	Max VCC_PSAUX	R	The 10-bit MSB justified maximum VCC_PSAUX of Slave1SYSMON measurement.
C_BASEADDR + 0x14AC[6]	Undefined	N/A	These locations are unused and contain Invalid data.
C_BASEADDR + 0x14B0[6]	Min VCC_PSINTLP	R	The 10-bit MSB justified minimum VCC_PSINTLP of Slave1SYSMON measurement.
C_BASEADDR + 0x14B4[6]	Min VCC_PSINFP	R	The 10-bit MSB justified minimum VCC_PSINFP of Slave1SYSMON measurement.
C_BASEADDR + 0x14B8[6]	MIn VCC_PSAUX	R	The 10-bit MSB justified minimums VCC_PSAUX of Slave1 SYSMON measurement.

Table 2-13: **IP Core Registers for Slave 1 SYSMON *(Cont'd)***

Base Address + Offset (hex)	Register Name	Access Type	Description
C_BASEADDR + 0x14FC	Flag Register	R[2]	The 16-bit register gives general status information of ALARM, Over Temperature (OT), disable information of SYSMON and information about whether the SYSMON is using internal reference voltage or external reference voltage.
C_BASEADDR + 0x1500	Configuration Register 0	R/W[4]	SYSMON Configuration Register 0.
C_BASEADDR + 0x1504	Configuration Register 1	R/W	SYSMON Configuration Register 1.
C_BASEADDR + 0x1508	Configuration Register 2	R/W	SYSMON Configuration Register 2.
C_BASEADDR + 0x150C	Configuration Register 3	R/W	SYSMON Configuration Register 3.
C_BASEADDR + 0x1510	Test Register	N/A	SYSMON Test Register (For factory test only).
C_BASEADDR + 0x1514	Analog Bus Register	N/A	Configuration register for the Analog Bus.
C_BASEADDR + 0x1518	Sequence Register 8	R/W	Sequencer channel selection (Vuser0-3).
C_BASEADDR + 0x151C	Sequence Register 9	R/W	Sequencer average selection (Vuser0-3).
C_BASEADDR + 0x1520	Sequence Register 0	R/W	SYSMON Sequence Register 0 (ADC channel selection).
C_BASEADDR + 0x1524	Sequence Register 1	R/W	SYSMON Sequence Register 1 (ADC channel selection).
C_BASEADDR + 0x1528	Sequence Register 2	R/W	SYSMON Sequence Register 2 (ADC channel averaging enable).
C_BASEADDR + 0x152C	Sequence Register 3	R/W	SYSMON Sequence Register 3 (ADC channel averaging enable).
C_BASEADDR + 0x1530	Sequence Register 4	R/W	SYSMON Sequence Register 4 (ADC channel analog-input mode).
C_BASEADDR + 0x1534	Sequence Register 5	R/W	SYSMON Sequence Register 5 (ADC channel analog-input mode).
C_BASEADDR + 0x1538	Sequence Register 6	R/W	SYSMON Sequence Register 6 (ADC channel acquisition time).
C_BASEADDR + 0x153C	Sequence Register 7	R/W	SYSMON Sequence Register 7 (ADC channel acquisition time).
C_BASEADDR + 0x1540	Alarm Threshold Register 0	R/W	The 10-bit MSB justified alarm threshold register 0 (Temperature Upper).

Table 2-13: **IP Core Registers for Slave 1 SYSMON *(Cont'd)***

Base Address + Offset (hex)	Register Name	Access Type	Description
C_BASEADDR + 0x1544	Alarm Threshold Register 1	R/W	The 10-bit MSB justified alarm threshold register 1 (V_{CCINT} Upper).
C_BASEADDR + 0x1548	Alarm Threshold Register 2	R/W	The 10-bit MSB justified alarm threshold register 2 (V_{CCAUX} Upper).
C_BASEADDR + 0x154C	Alarm Threshold Register 3	R/W(5)	The 10-bit MSB justified alarm threshold register 3 (OT Upper).
C_BASEADDR + 0x1550	Alarm Threshold Register 4	R/W	The 10-bit MSB justified alarm threshold register 4 (Temperature Lower).
C_BASEADDR + 0x1554	Alarm Threshold Register 5	R/W	The 10-bit MSB justified alarm threshold register 5 (V_{CCINT} Lower).
C_BASEADDR + 0x1558	Alarm Threshold Register 6	R/W	The 10-bit MSB justified alarm threshold register 6 (V_{CCAUX} Lower).
C_BASEADDR + 0x155C	Alarm Threshold Register 7	R/W	The 10-bit MSB justified alarm threshold register 7 (OT Lower)
C_BASEADDR + 0x1560	Alarm Threshold Register 8	R/W	The 10-bit MSB justified alarm threshold register 8 (V_{BRAM} Upper)
C_BASEADDR + 0x1570	Alarm Threshold Register 12	R/W	The 10-bit MSB justified alarm threshold register 12 (V_{BRAM} Lower)
C_BASEADDR + 0x1580	Alarm Threshold Register 16	R/W	The 10-bit MSB justified alarm threshold register 16 (V_{USER0} Upper)
C_BASEADDR + 0x1584	Alarm Threshold Register 17	R/W	The 10-bit MSB justified alarm threshold register 17 (V_{USER1} Upper)
C_BASEADDR + 0x1588	Alarm Threshold Register 18	R/W	The 10-bit MSB justified alarm threshold register 18 (V_{USER2} Upper)
C_BASEADDR + 0x158C	Alarm Threshold Register 19	R/W	The 10-bit MSB justified alarm threshold register 19 (V_{USER3} Upper)
C_BASEADDR + 0x15A0	Alarm Threshold Register 22	R/W	The 10-bit MSB justified alarm threshold register 14 (V_{USER0} Lower).

Table 2-13: **IP Core Registers for Slave 1 SYSMON *(Cont'd)***

Base Address + Offset (hex)	Register Name	Access Type	Description
C_BASEADDR + 0x15A4	Alarm Threshold Register 23	R/W	The 10-bit MSB justified alarm threshold register 15 (V_{USER1} Lower).
C_BASEADDR + 0x15A8	Alarm Threshold Register 24	R/W	The 10-bit MSB justified alarm threshold register 16 (V_{USER2} Lower).
C_BASEADDR + 0x15AC	Alarm Threshold Register 25	R/W	The 10-bit MSB justified alarm threshold register 17 (V_{USER3} Lower).
C_BASEADDR + 0x1600	V_{USER0}	R	The 10-bit MSB justified result of the on-chip V_{USER0} supply monitor measurement is stored at this location.
C_BASEADDR + 0x1604	V_{USER1}	R	The 10-bit MSB justified result of the on-chip V_{USER1} supply monitor measurement is stored at this location.
C_BASEADDR + 0x1608	V_{USER2}	R	The 10-bit MSB justified result of the on-chip V_{USER2} supply monitor measurement is stored at this location.
C_BASEADDR + 0x160C	V_{USER3}	R	The 10-bit MSB justified result of the on-chip V_{USER3} supply monitor measurement is stored at this location.
C_BASEADDR + 0x1680	Max V_{USER0}	R	Maximum V_{USER0} measurement recorded since power-up or the last System Monitor reset.
C_BASEADDR + 0x1684	Max V_{USER1}	R	Maximum V_{USER1} measurement recorded since power-up or the last System Monitor reset.
C_BASEADDR + 0x1688	Max V_{USER2}	R	Maximum V_{USER2} measurement recorded since power-up or the last System Monitor reset.
C_BASEADDR + 0x168C	Max V_{USER3}	R	Maximum V_{USER3} measurement recorded since power-up or the last System Monitor reset.
C_BASEADDR + 0x16A0	Min V_{USER0}	R	Minimum V_{USER0} measurement recorded since power-up or the last System Monitor reset.
C_BASEADDR + 0x16A4	Min V_{USER1}	R	Minimum V_{USER1} measurement recorded since power-up or the last System Monitor reset.
C_BASEADDR + 0x16A8	Min V_{USER2}	R	Minimum V_{USER2} measurement recorded since power-up or the last System Monitor reset.

Table 2-13: **IP Core Registers for Slave 1 SYSMON *(Cont'd)***

Base Address + Offset (hex)	Register Name	Access Type	Description
C_BASEADDR + 0x16AC	Min V_{USER3}	R	Minimum V_{USER3} measurement recorded since power-up or the last System Monitor reset.

Notes:

1. These are 16-bit registers internal to SYSMON. These are mapped to the lower-half word boundary on 32-bit System Management Wizard IP core registers.
2. Writing to this SYSMON hard macro register is not allowed. The SYSMON hard macro data registers are 16 bits in width. The SYSMON hard macro specification guarantees the first 10 MSB bits accuracy; so only these bits are used for reference.
3. Writing to this register resets the SYSMON hard macro. No specific data pattern is required to reset the SYSMON hard macro.
4. See the *SYSMON User Guide* [Ref 1] for setting the different bits available in configuration registers for UltraScale devices.
5. The OT upper register is a user-configurable register for the upper threshold level of temperature. If this register is left unconfigured, then the SYSMON considers 125°C as the upper threshold value for OT. While configuring this register, the last four bits must be set to 0011, that is, Alarm Threshold Register 3[3:0] = 0011. The upper 12 bits of this register are user configurable.
6. These registers are valid for Zynq UltraScale+ devices only.

Designing with the Core

This chapter includes guidelines and additional information to facilitate designing with the core.

Clocking

The clock to SYSMON primitive is `dclk`. When AXI4-Lite is selected as the bus interface, `dclk` is connected to the `s_axi_aclk` clock . So, the `adcclk` division factor must be programmed in correlation with the `s_axi_aclk` frequency.

When DRP or None interface is selected, `dclk` clock is at the top-level of the IP, and `adcclk` division factor must be programmed in correlation with the `dclk` frequency.

When Streaming is enabled for DRP or None interface selection, `m_axis_aclk` is connected to `dclk`.

Resets

When AXI4-Lite is selected as the bus interface, certain registers of the IP can be reset by writing a value 0xA to register 0x00. The AXI4-Lite and AXI4-Stream interfaces also have individual reset pins.

When DRP or None interface is selected, `reset_in` is the input port at the top-level of the IP.

Protocol Description

For more detailed information, see the AXI4-Lite protocol specifications. Figure 3-1 shows the simulation snapshots for Temperature value read from SYSMON register.

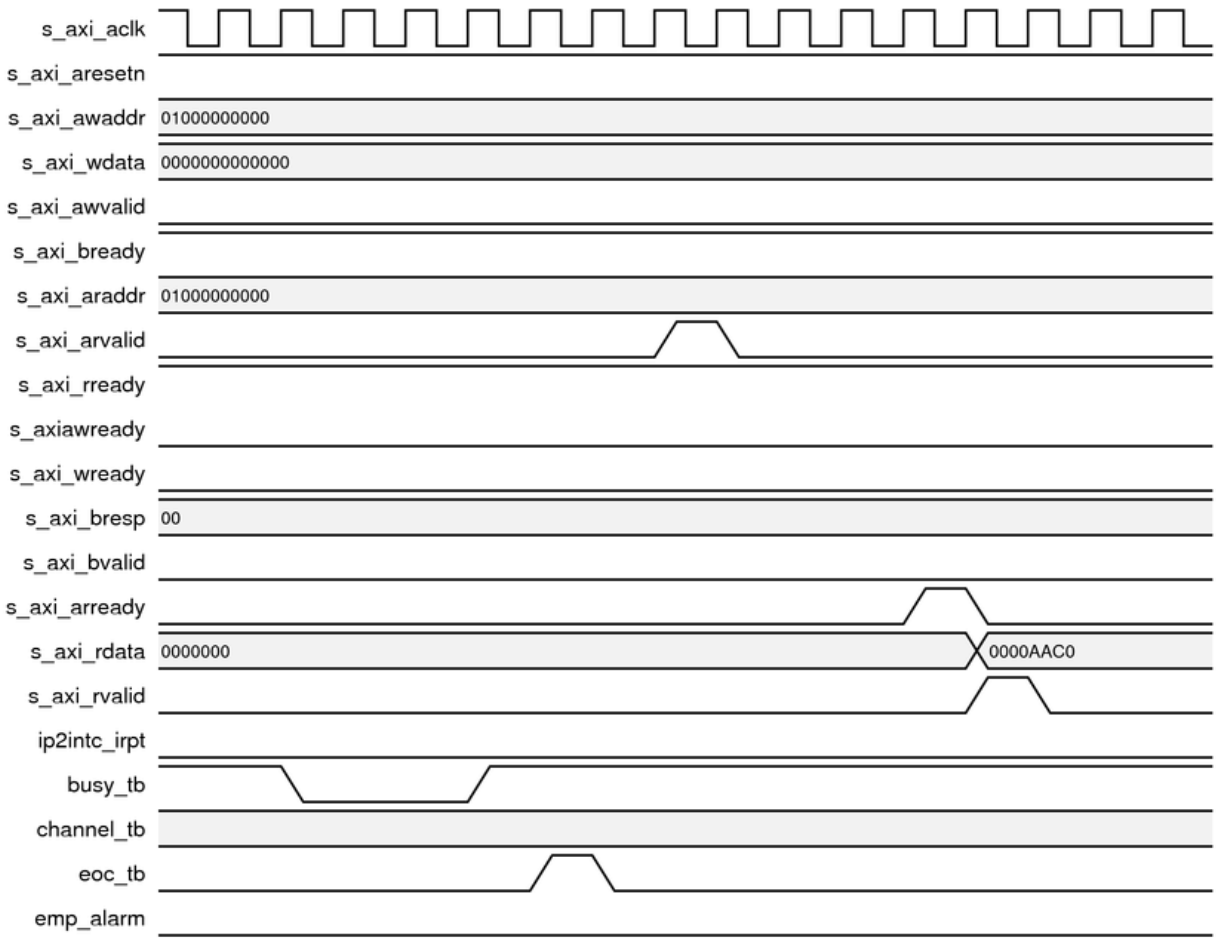

Figure 3-1: **AXI4-Lite Interface Reading Temperature Values in Simulation**

I2C Interface for SSIT Devices

The System Management Wizard core enables conversion of the I^2C data to DRP data for making the I^2C interface available to slave devices of SSIT devices. Select **Enable I2C for all SLRs** to enable this feature. By selecting this feature:

- address override option is always enabled

- I^2C can be addressed only through address override feature
- direct address decoding from Vp/Vn is not supported
- I2C address mentioned in the Override Address would be the assigned to the Master SYSMON
- Slave 0 I2C address would be Master I2C address incremented by 1
- Slave 1 I2C address would be Master I2C address incremented by 2

Figure 3-2 illustrates the implementation of Enable I^2C for all SLRs logic in the System Management Wizard. For details on I^2C interface in SYSMON, see the *UltraScale Architecture System Monitor Advanced Specification User Guide* (UG580) [Ref 1].

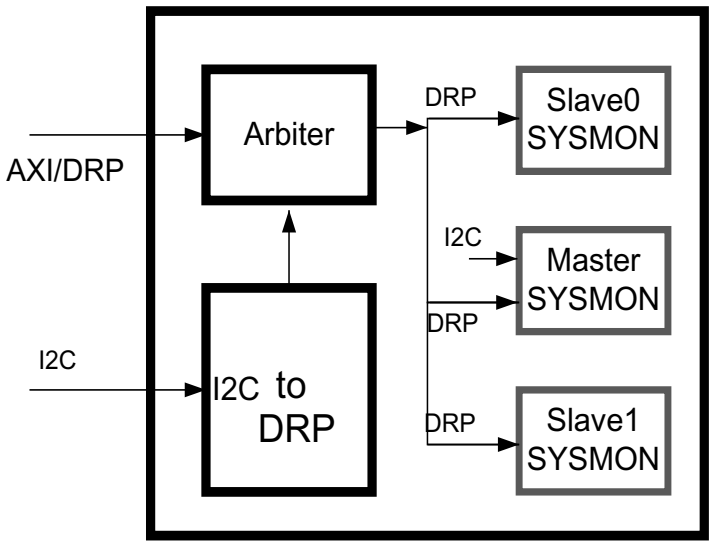

Figure 3-2: **Block Diagram - Enable I2C for all SLRs**

The data shifted into the 32-bit SYSMON DR instructs the arbitrator to carry out a write, read, or no operation on the SYSMON DRP. The arbitration follows the same rules as the primitive SYSMON. For details, see the *UltraScale Architecture System Monitor Advanced Specification User Guide* (UG580) [Ref 1]. Figure 3-3 shows the data format of the DRP command loaded into the SYSMON DR.

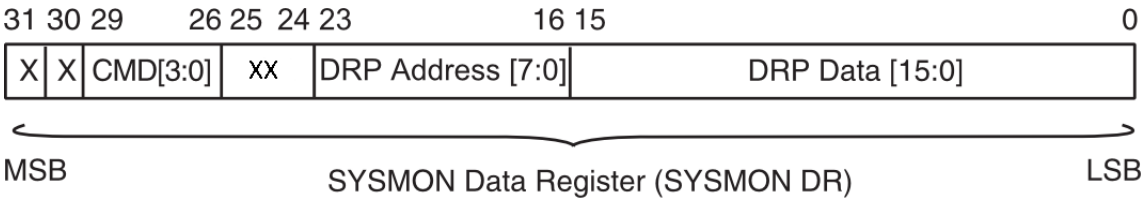

Figure 3-3: **SYSMON DRP Command**

The first 16 LSBs of SYSMON DR[15:0] contain the DRP register data. For both read and write operations, the address bits SYSMON DR[23:16] hold the DRP target register address.

The command bits SYSMON DR[29:26] specify a read, write, or no operation as shown in Table 3-1.

Table 3-1: **DRP Commands**

CMD[3:0]				Operation
0	0	0	0	No operation
0	0	0	1	DRP read
0	0	1	0	DRP write
-	-	-	-	Not defined

Clock and reset must be enabled when the I^2C interface in SYSMON is enabled.

Design Flow Steps

This chapter describes customizing and generating the core, constraining the core, and the simulation, synthesis and implementation steps that are specific to this IP core. More detailed information about the standard Vivado® design flows and IP integrator can be found in the following Vivado Design Suite user guides:

- *Vivado Design Suite User Guide: Logic Simulation* (UG900) [Ref 2]
- *Vivado Design Suite User Guide: Designing IP Subsystems using IP Integrator* (UG994) [Ref 3]
- *Vivado Design Suite User Guide: Designing with IP* (UG896) [Ref 5]
- *Vivado Design Suite User Guide: Getting Started* (UG910) [Ref 7]

Customizing and Generating the Core

This section includes information about using Xilinx tools to customize and generate the core in the Vivado Design Suite.

If you are customizing and generating the core in the Vivado IP integrator, see the *Vivado Design Suite User Guide: Designing IP Subsystems using IP Integrator* (UG994) [Ref 3] for detailed information. IP integrator might auto-compute certain configuration values when validating or generating the design. To check whether the values change, see the description of the parameter in this chapter. To view the parameter value, run the `validate_bd_design` command in the Tcl console.

This chapter describes the use of system Management Wizard v1.0 in Vivado® Integrated Design Environment (IDE).

TIP: *Tool tips are available in the Vivado IDE for most features. Place your mouse over the relevant text, and additional information is provided in a dialog box.*

You can customize the IP for use in your design by specifying values for the various parameters associated with the IP core using the following steps:

1. Select the IP from the IP catalog under **FPGA Features and Design**.

2. **Double-click System Management Wizard** or select the Customize IP command from the toolbar or right-click menu.

For details, see the *Vivado Design Suite User Guide: Designing with IP* (UG896) [Ref 5] and the *Vivado Design Suite User Guide: Getting Started* (UG910) [Ref 7].

Note: Figures in this chapter are illustrations of the Vivado IDE. This layout might vary from the current version.

TIP: *This section describes how to set up a project in the Vivado Design Suite flow. Before generating the example design, set up the project as described in Creating a Directory and Setting the Project Options of this guide.*

The Component Name is a user selectable component name. Reserved words in Verilog or VHDL must not be used as component name.

Basic Tab

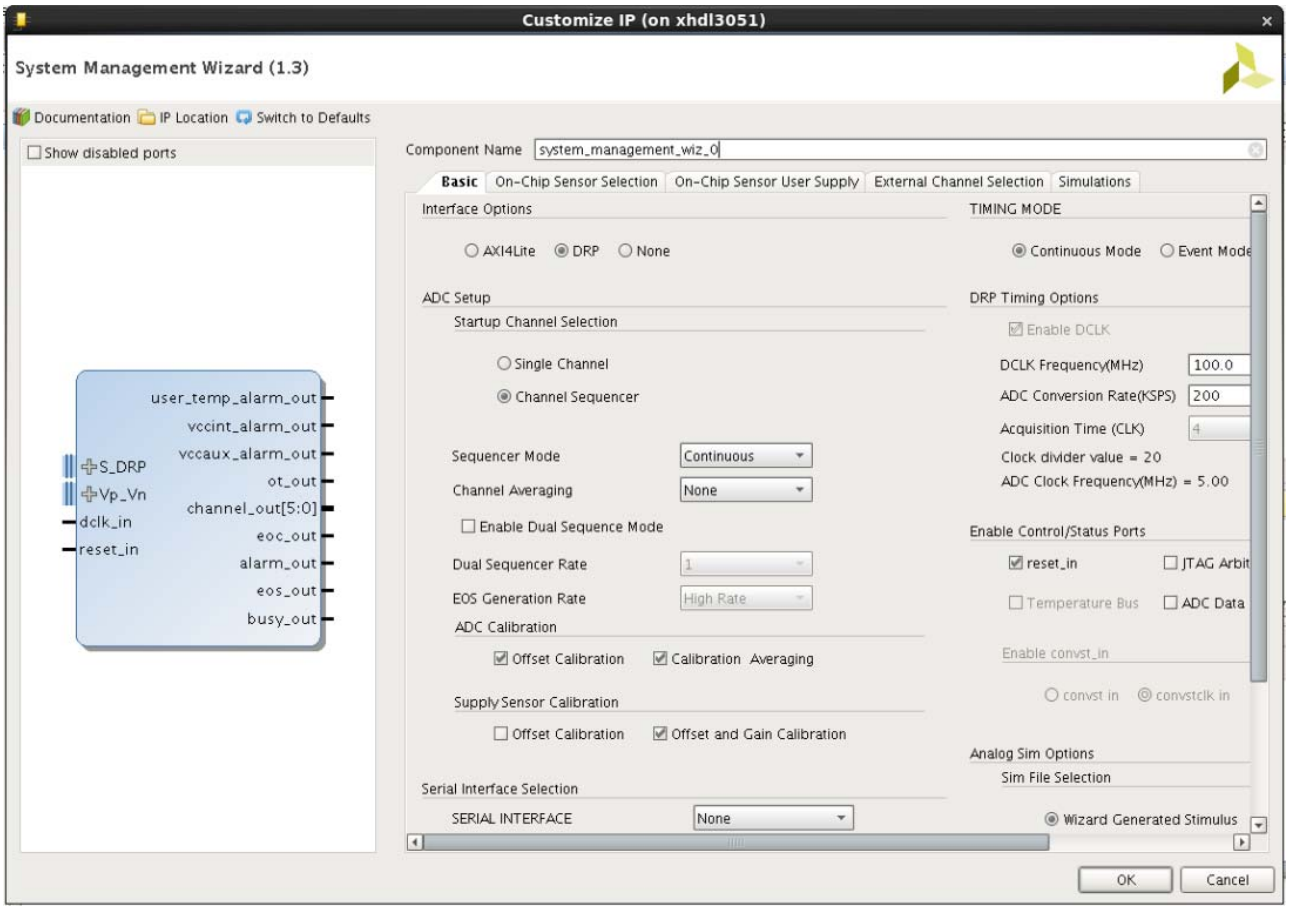

Figure 4-1: **Basic Tab**

The parameters on the Basic tab are as follows:

- **Interface Options**: Selects the interface for the System Management Wizard. DRP is the default option. You can select AXI4-Lite, DRP, or None. The DRP port is the FPGA logic interface for SYSMON. It facilitates access to the register file interface of the SYSMON. The SYSMON control registers can be read or written using this port. This port can only be enabled when DCLK clock is present.
- **ADC Setup**
  - **Startup Channel Selection**: SYSMON can be configured in one of the following modes:
    - Single Channel: In this mode, you can select only one channel to monitor. All channels and alarms shown in the GUI for Sequencer mode are also available for single-channel mode in the drop-down list, allowing only one channel on any tab.
    - Channel Sequencer: Allows you to select any number of channels to monitor. The channels to be used for this mode can be selected on all other tabs.

      For more information about the Channel Sequencer mode, see the *UltraScale Architecture System Monitor Advanced Specification User Guide* (UG580) [Ref 1].
  - **Sequencer Mode**: If the SYSMON is configured for Channel Sequencer mode, you can choose the required sequencer mode. The available options are Continuous, One-pass or Default mode.
  - **Channel Averaging**: Select the required averaging value. The available options are None, 16, 64, and 256.
  - **ADC Calibration/Supply Sensor Calibration**: You can select ADC offset calibration and the type of supply sensor calibration by checking the respective check boxes. Calibration averaging is enabled by default in SYSMON. You can disable this by deselecting the box.
- **Enable Slave SYSMON**: This section contains the parameters to enable/disable the Slave SYSMONs for SSIT devices. These parameters are visible only if the device selected is SSIT.
  - **Enable Slave 0 SYSMON**: If this parameter is enabled, SYSMON in Slave 0 SLR would be instantiated in the Wizard and the SSIT Slave 0 Sensors tab would be appeared in the GUI. The default value is off.
  - **Enable Slave 1 SYSMON**: If this parameter is enabled, SYSMON in Slave 1 SLR would be instantiated in the Wizard and the SSIT Slave1 Sensors tab would be appeared in the GUI. This parameter is disabled by default.
- **SYSMON I^2C Options**: Enables SYSMON I^2C pins and configuration to get access to the DRP registers pre and post configuration. SYSMON calculates it's own I^2C address (at power-up the voltage on the dedicated analog input channel VP/VN is measured and the four MSBs of the measured value are used to decode the I^2C slave address), but

you can override this by enabling SYSMON I^{2}C Address Override option in the Vivado IDE.

Table 4-1: **I^{2}C Slave Address Decoding**

MSBs of Measured Voltage	I^{2}C Slave Address
0000	0110010
0001	0001011
0010	0010011
0011	0011011
0100	0100011
0101	0101011
0110	0110011
0111	0111011
1000	1000011
1001	1001011
1010	1010011
1011	1011011
1100	1100011
1101	1101011
1110	1110011
1111	0111010

- **Enable I2C on SLR0 (only Master)**: Select to enable the direct I^{2}C interface on the Master SLR device.
- **Enable I2C on all SLRs**: Select to enable conversion of the I^{2}C data to DRP data for making I2C interface available for all SLRs that do not have a direct I^{2}C interface.
- **SYSMON I2C Address Override**: Select to enable the address override option. Enabling this option uses the address given in the override option.
- **SYSMON I2C Slave Address (in Hex)**: Specify the address for the I^{2}C. This address overrides the address encoded from Vp/Vn. When **Enable I2C on all SLRs** is enabled, the address given is the I^{2}C address for the Master SYSMON. Slave 0 I^{2}C address equals the Master I^{2}C address incremented by 1. Slave 1 I^{2}C address equals the Master I^{2}C address incremented by 2.

- **Timing Mode**:
  - **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.

- **DRP Timing Options**: The SYSMON clock (ADCCLK) is derived from the dynamic reconfiguration port (DRP) clock DCLK. The SYSMON supports a DRP clock frequency of up to 250 MHz. The SYSMONE can also operate in the absence of DCLK. For more information on the DRP see *UltraScale Architecture System Monitor Advanced Specification User Guide* (UG580) [Ref 1].

  The ADC Clock Frequency should be 1-5.2 MHz. To support this lower frequency clock, the SYSMON has an internal clock divider. Specify the external DCLK frequency and required ADC conversion rate (maximum 0.2 Msps) in the Vivado® IDE. Based on the value of DCLK clock, the wizard calculates the appropriate clock divider value based on the values of DCLK clock and ADC conversion. The wizard also displays the ADC Clock frequency value and the actual conversion rate of the ADC.

- **Enable Control/Status Ports**: The Control/Status Port Selection allows you to select the I/O ports on the SYSMON primitive.
  - **reset_in**: Allows an external input reset signal to be connected to the SYSMON.
  - **Enable convst_in**: Sets `convst_in` or `convstclk_in` as trigger sources for Event Mode Timing.
  - **Temperature Bus**: There is only one SYSMON primitive available in a FPGA for use. If the System Management Wizard core is used in a system using MIG, the TEMP_OUT bus should be connected to the device_temp_i input port of the MIG block. This disables inference of the SYSMON hard block in MIG. Enabling temperature bus provides a 10-bit TEMP_OUT port with the temperature update logic. This checkbox is available when the interface option is AXI4-Lite.
- **JTAG Arbiter**: Enables JTAG status ports to check the status of JTAG access to SYSMON registers. Other output status signals are also provided to facilitate interfacing of the SYSMON to a user design. For more information, see *UltraScale Architecture System Monitor Advanced Specification User Guide* (UG580) [Ref 1].
- **Analog Sim File Options**: You can provide the relative or absolute path and update the name of the Analog Stimulus File in this section.
  - **Sim File Selection**: The default name and path for the analog stimulus is design.txt. It is generated in the core simulation area. By changing the default option to **relative path csv**, you can set a custom path for the analog stimulus TXT file.

    To provide a CSV file instead of a TXT file, you can set **Sim File Selection** to **Relative path csv** and set the **Analog Stimulus File** name and the file location. The CSV to TXT conversion is performed automatically when the example design is opened for this IP. If the example project is not required, run the CSV to TXT Tcl script generated after the output products of the IP are generated. The conversion script (`<Component name>_csv_to_txt.tcl`) is created in `tcl` folder located in IP

path. This script should be sourced in the Vivado Tcl console for conversion. The relative path of the script is:

```
project_1/system_management_wiz_0_example/system_management_wiz_0_example.srcs/ttcl
directory
```

Figure 4-2 shows an example XLS input.

	A	B	C	D	E	F	G	H	I	J	K	L
1	TIME	TEMP	VCCINT	VCCBRAM	VCCAUX	VP	VN	VAUXP[0]	VAUXN[0]	VAUXP[1]	VAUXN[1]	VAUXP
2	0	63	0.91	0.91	1.8	0.5	0	0.5	0	0.5	0	
3	6250	88	0.97	0.97	1.94	0.3	0	0.2	0	0.2	0	
4	11450	53	0.81	0.81	1.7	0.9	0	0.9	0	0.9	0	
5	22900	63	0.91	0.91	1.8	0.5	0	0.5	0	0.5	0	

Figure 4-2: **Example XLS Input**

Figure 4-3 shows the same file as a CSV input.

```
TIME,TEMP,VCCINT,VCCBRAM,VCCAUX,VP,VN,VAUXP[0],VAUXN[0],VAUXP[1],VAUXN[1],VAUXP
0,63,0.91,0.91,1.8,0.5,0,0.5,0,0.5,0,0.5,0,0.5,0,0.5,0,0.5,0,0.5,0,0.5,0,0.5,0,
6250,88,0.97,0.97,1.94,0.3,0,0.2,0,0.2,0,0.2,0,0.2,0,0.2,0,0.2,0,0.2,0,0.2,0,0.
11450,53,0.81,0.81,1.7,0.9,0,0.9,0,0.9,0,0.9,0,0.9,0,0.9,0,0.9,0,0.9,0,0.9,0,0.
22900,63,0.91,0.91,1.8,0.5,0,0.5,0,0.5,0,0.5,0,0.5,0,0.5,0,0.5,0,0.5,0,0.5,0,0.
```

Figure 4-3: **Example CSV Input**

Figure 4-4 shows the same file as a converted TXT file.

```
TIME TEMP VCCINT VCCBRAM VCCAUX VP VN VAUXP[0] VAUXN[0] VAUXP[1] VAUXN[1] VAUXP
0 63 0.91 0.91 1.8 0.5 0 0.5 0 0.5 0 0.5 0 0.5 0 0.5 0 0.5 0 0.5 0 0.5 0 0.5 0
6250 88 0.97 0.97 1.94 0.3 0 0.2 0 0.2 0 0.2 0 0.2 0 0.2 0 0.2 0 0.2 0 0.2 0 0.
11450 53 0.81 0.81 1.7 0.9 0 0.9 0 0.9 0 0.9 0 0.9 0 0.9 0 0.9 0 0.9 0 0.9 0 0.
22900 63 0.91 0.91 1.8 0.5 0 0.5 0 0.5 0 0.5 0 0.5 0 0.5 0 0.5 0 0.5 0 0.5 0 0.
```

Figure 4-4: **Example Converted TXT File**

- **Analog Stimulus File**: Customizes the name of the SYSMON analog stimulus file.
- **Sim File Location**: Enabled when **Sim File Selection** is not default. Relative or absolute path of the analog stimulus can be provided in this box. Relative path is with respect to the simulation directory. If the example design behavioral simulation is run, the relative path is with respect to `project_1/system_management_wiz_0_example/system_management_wiz_0_example.sim/sim_1/behav` directory.

- **Individual Simulations**: Allows you to feed each input a different stimulus. A unique wave can be fed as input to each analog input. When this parameter is enabled, a table appears in the GUI for taking inputs for each analog input separately.

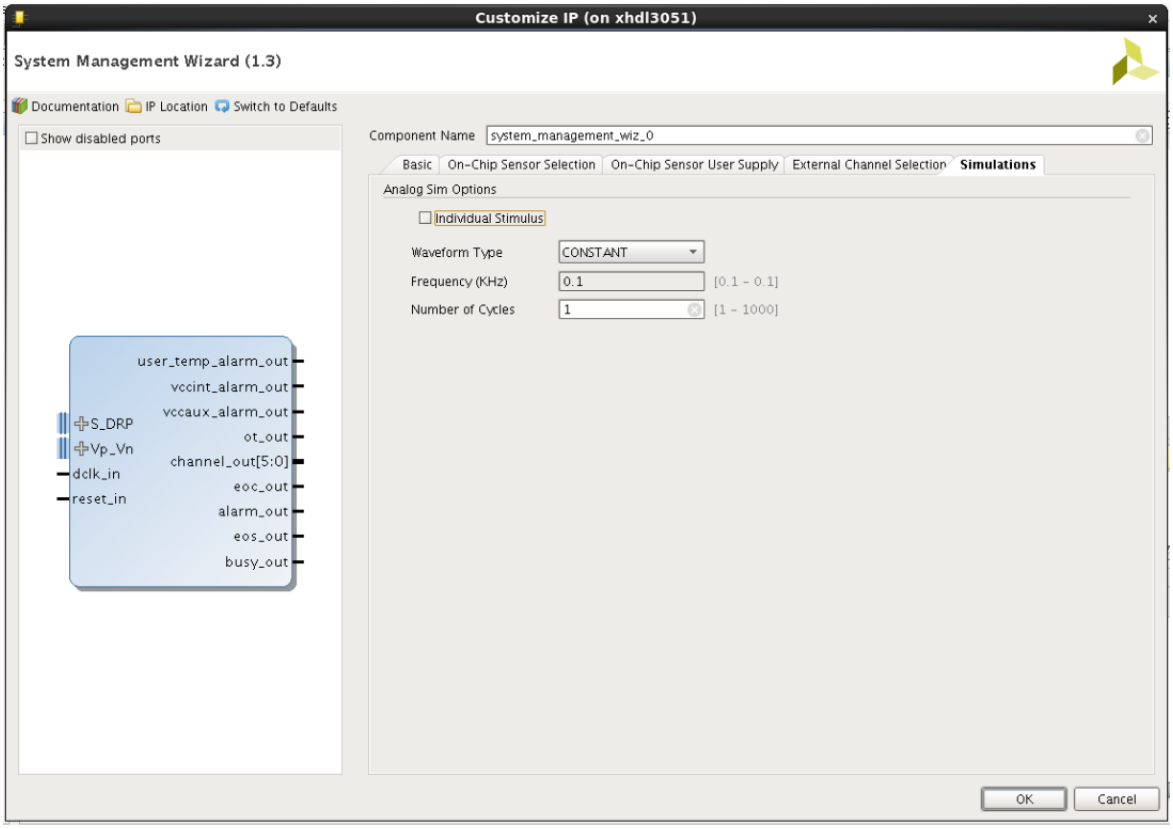

Figure 4-5: **Stimulations Tab**

- **Waveform Type**: Choose CONSTANT, SINE, TRIANGLE, or SQUARE wave as stimulus on external channel analog inputs. See Figure 4-6, Figure 4-7, and Figure 4-8 for examples of the waveform types.

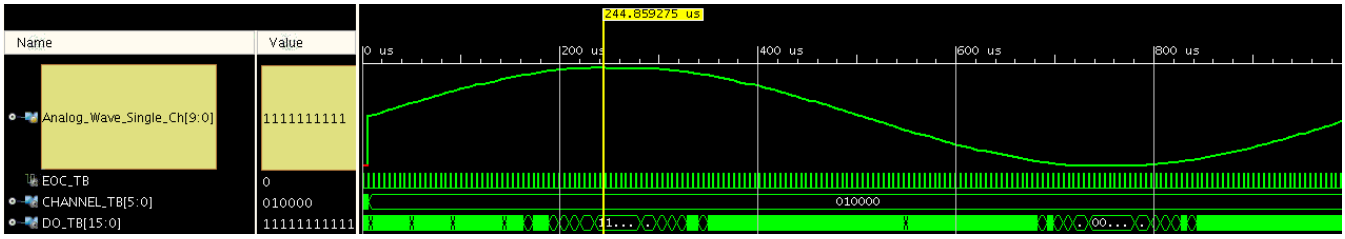

Figure 4-6: **Sine Wave**

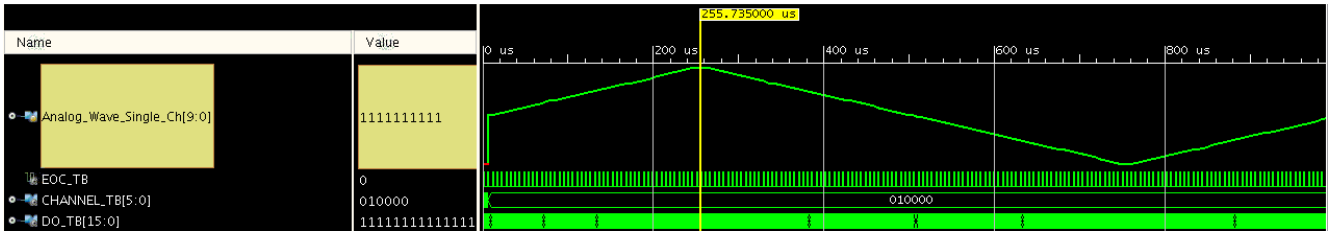

Figure 4-7: **Triangle Wave**

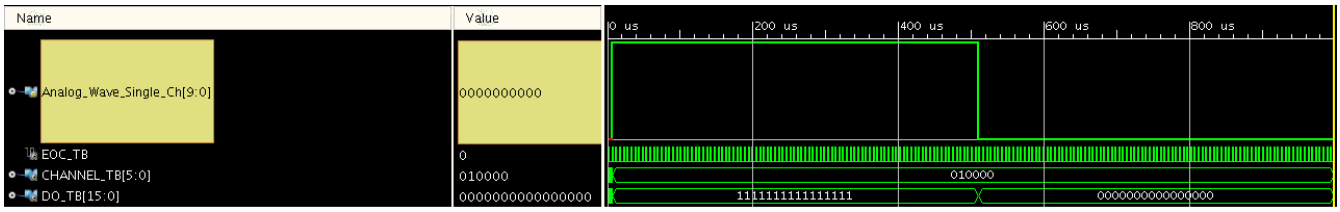

Figure 4-8: **Triangle Wave**

- **Frequency**: The analog waveform frequency can be configured from 0.1 KHz to half of the ADC sampling rate. In sequencer mode, the range depends on the number of channels selected.
- **Number of Wave**: 1 to 1,000 periods of the selected waveform type can be generated in the analog stimulus file.

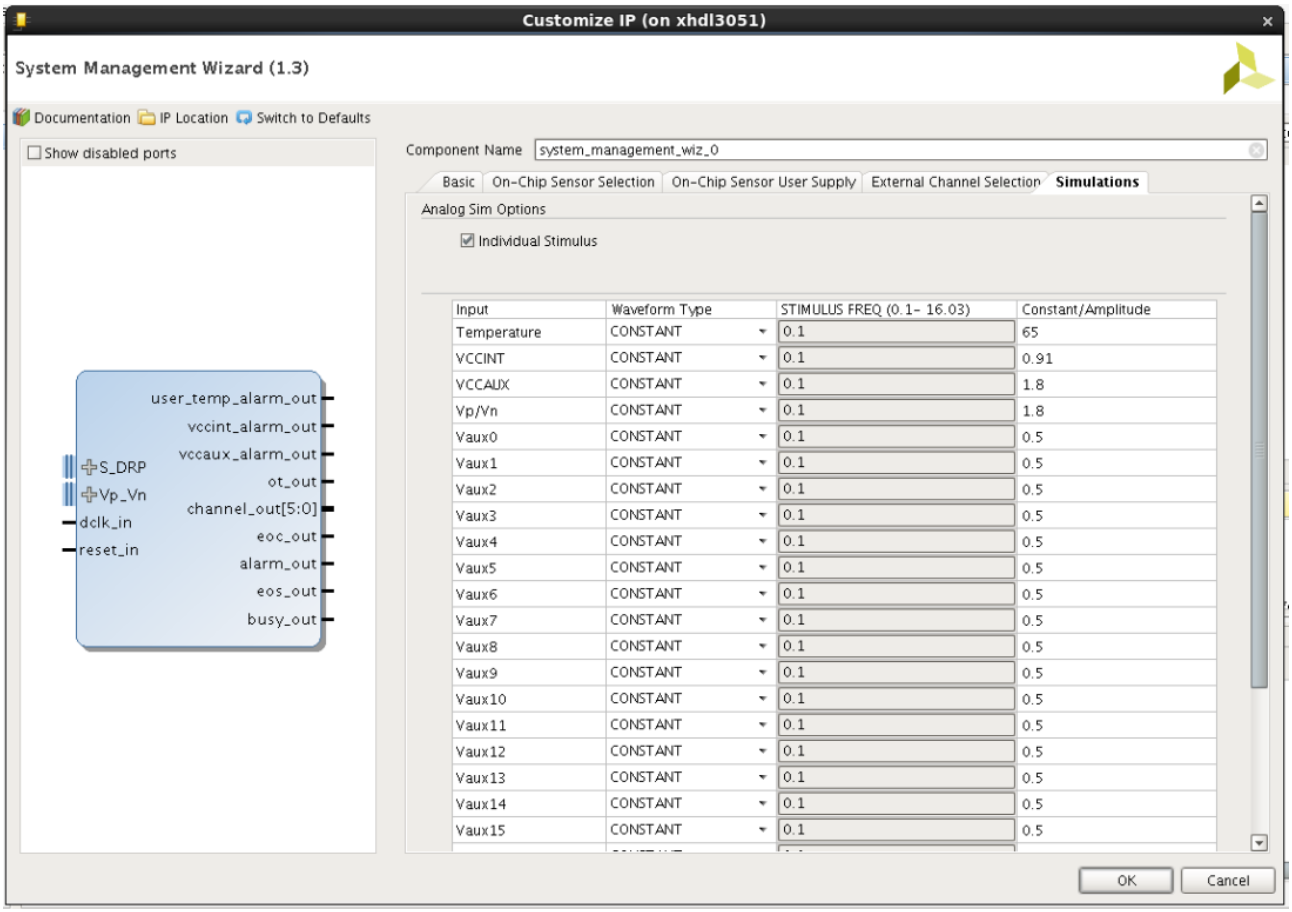

Figure 4-9: **Number of Wave**

On-Chip Sensor Selection tab

All on-chip sensor channels (Temperature, VCCINT, VCCAUX, and others) are available for selection in this tab.

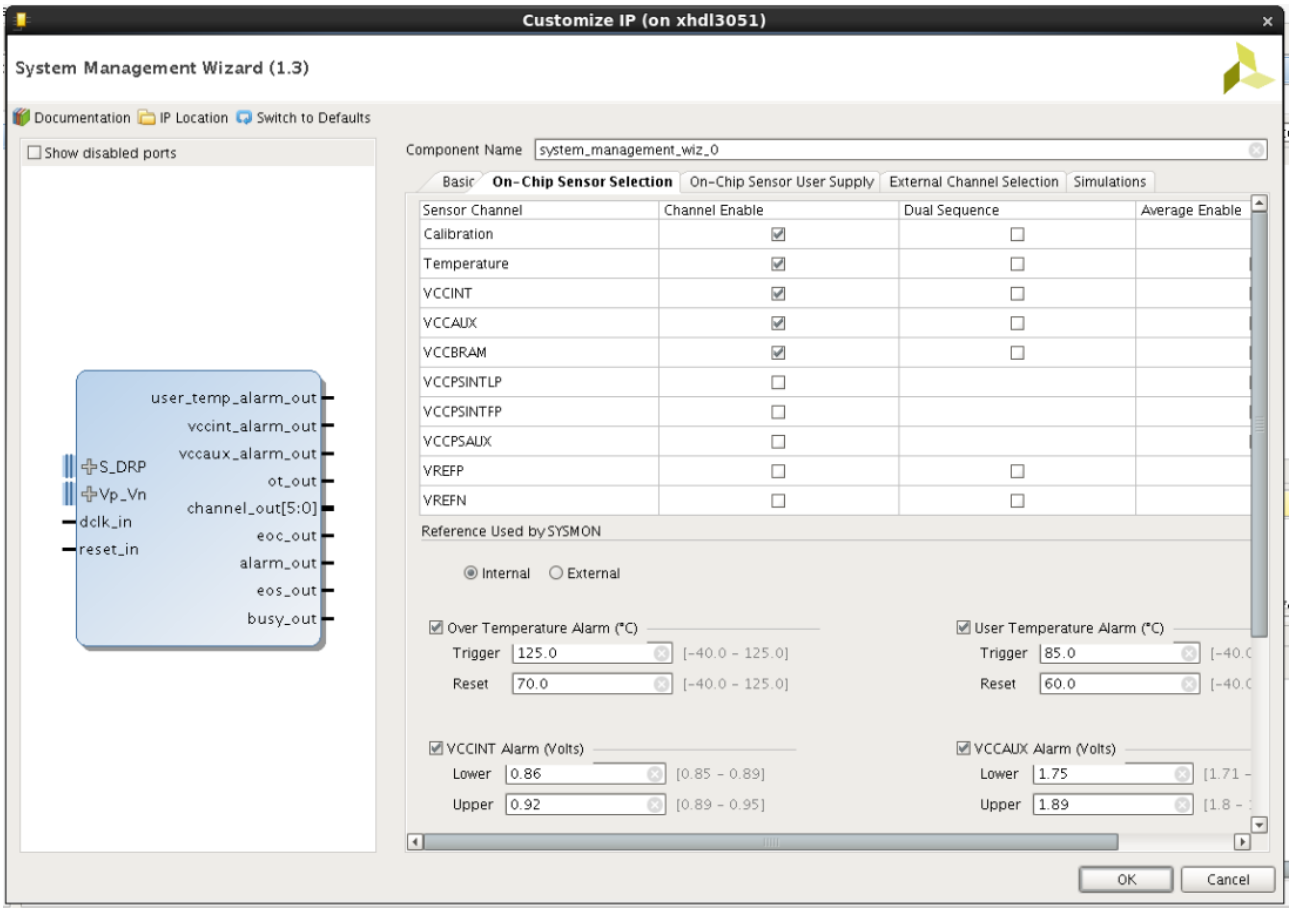

Figure 4-10: **On-Chip Sensor Selection and Alarms for Channel Sequencer**

The alarms listed in this tab (Figure 4-10) allows the 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 goes active if enabled. For a detailed description of the alarm functionality see *UltraScale Architecture System Monitor Advanced Specification User Guide* (UG580) [Ref 1].

Use the checkboxes to enable alarm logic outputs.

- **Reference Used by SYSMON:** Temperature calculations use different set of calculations for on-chip reference and external reference. Select an option, either internal or external reference, for appropriate calculations.
- **Over Temperature Alarm and User Temperature Alarm**: Trigger and Reset levels for temperature alarm output can be entered using these fields. Both fields can be set.

- **VCCINT Alarm, VCCAUX Alarm, and VBRAM Alarm**: 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 goes active. The alarm output is reset when a measurement inside these limits is generated. The default limits in the Vivado IDE represent ±5% on the nominal supply value.

On-Chip Sensor User Supply Tab

IMPORTANT: *This tab is for non-Stacked Silicon Interconnect (SSI) devices. If using an SSI device, this tab configures the master SYSMON. For SSI devices, also see SSIT Slave0/Slave1 Sensors Tab.*

On-Chip Sensor User supply (VUSER 0 - 3) can be enabled for monitoring up to three supplies for HP banks and up to four supplies for HR banks. User supplies are hooked to four SYSMON analog bus in each quadrant of UltraScale™ devices. A range of banks depending on the position with respect to SYSMON block in FPGA are placed into four quadrants NE, SW, SE, NW respectively. Select the bank and supply for each enabled VUSER supply or alarm. The System Management Wizard runs DRC to check the valid configuration of the VUSER bank and supply.

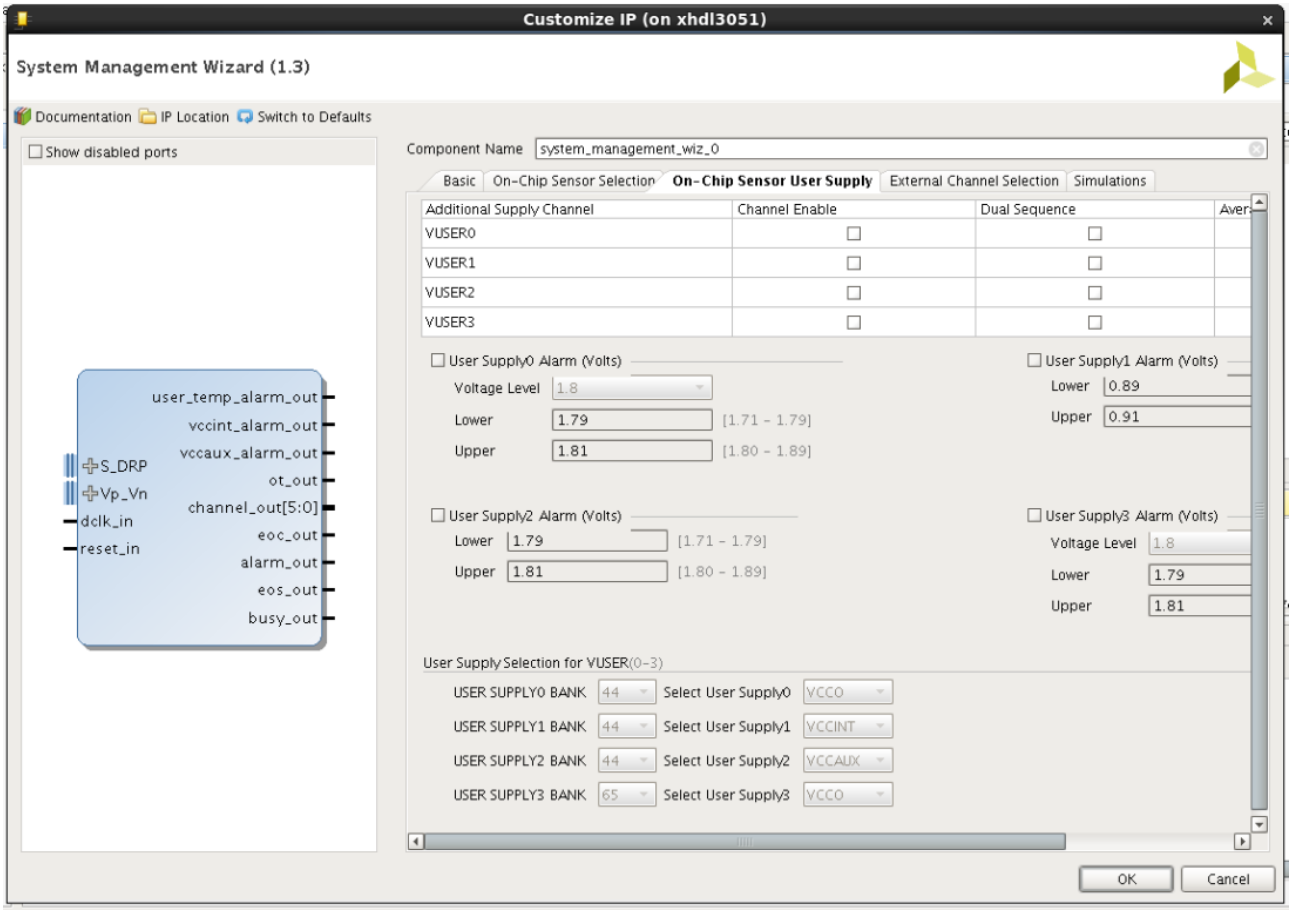

Figure 4-11: **On-Chip Sensor User Supply and Alarms Tab for Channel Sequencer**

- **Channel Enable and Average Enable**: Use the checkboxes to enable alarm logic outputs.
  - VUSER0
  - VUSER1
  - VUSER2
  - VUSER3
- **User Supply0, User Supply1, User Supply2, User Supply3 Alarms**: Both upper and lower alarm thresholds can be specified for the selected user supplies. The range varies with the type of the supply. If the measured value moves outside these limits, the alarm logic output goes active. The alarm output is reset when a measurement inside these limits is generated. The default limit is ±5% on the nominal supply value.

SSIT Slave0/Slave1 Sensors Tab

IMPORTANT: *This tab applies to Stacked Silicon Interconnect (SSI) devices only.*

For SSI devices, the SSIT Slave0/Slave1 Sensors tab configures temperature and VUSER supplies. Please see Figure 4-12 and details in On-Chip Sensor User Supply Tab to configure these.

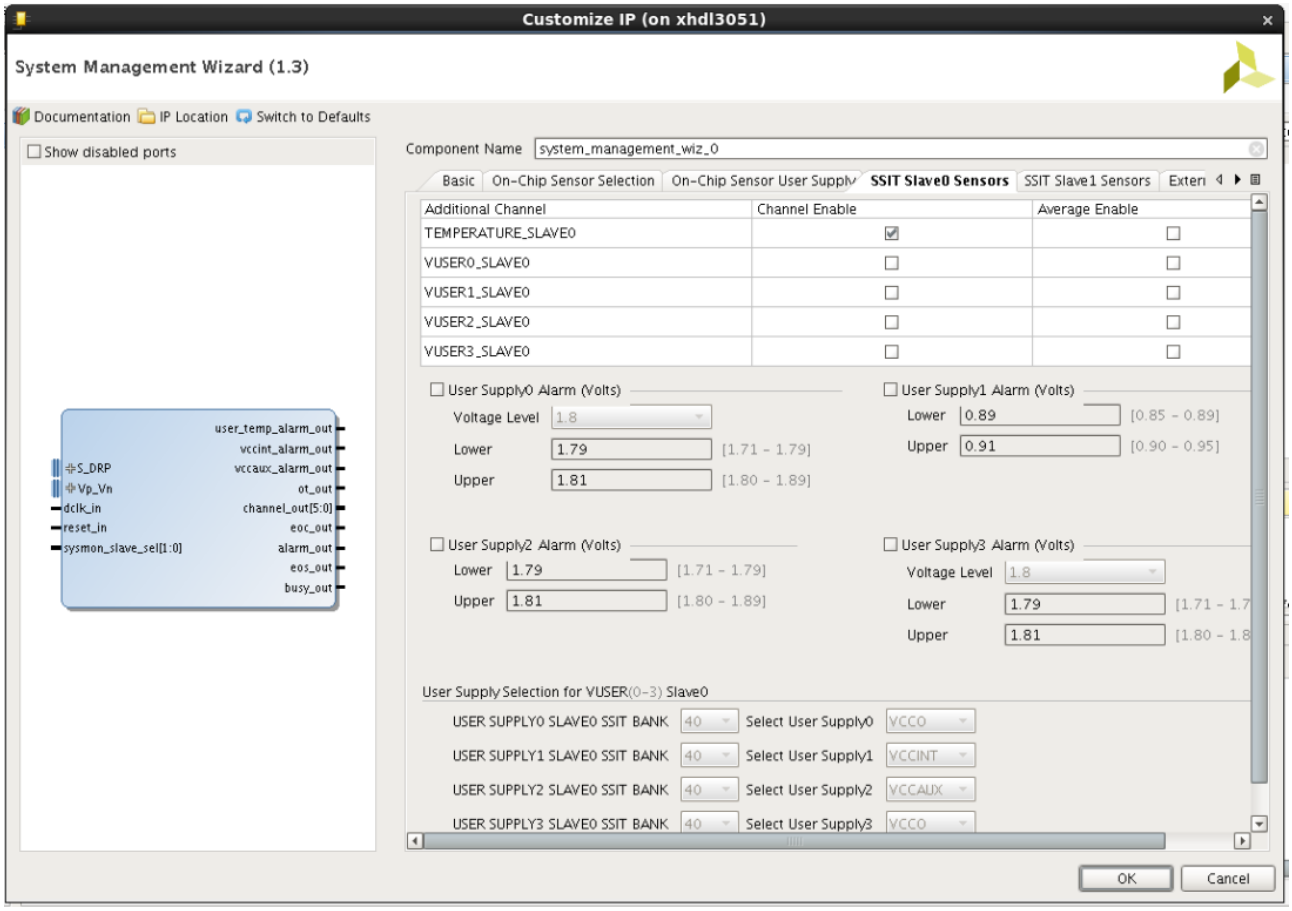

Figure 4-12: **SSIT Slave0/Slave1 Sensors Tab**

External Channel Selection Tab

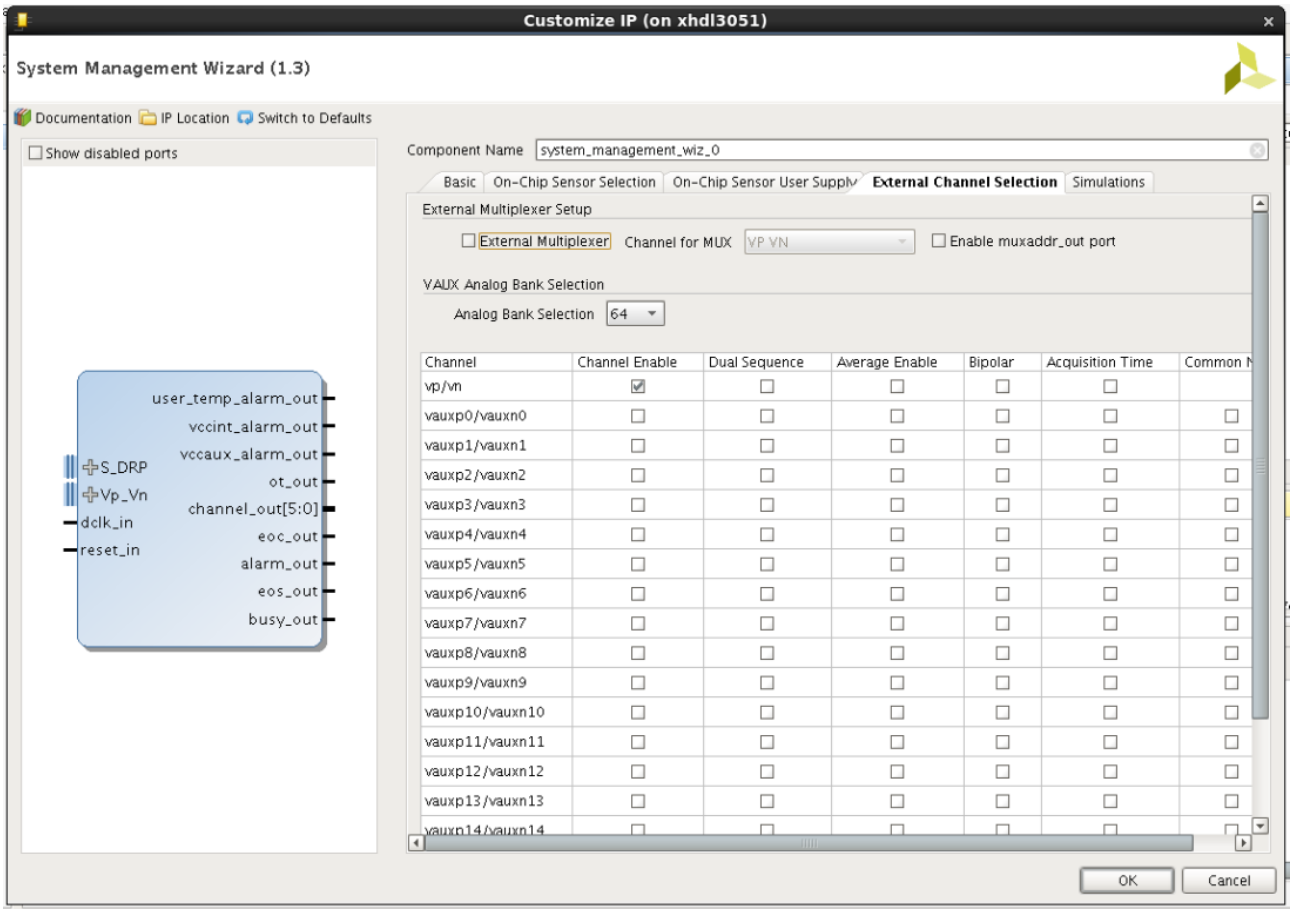

Figure 4-13: **External Channel Selection Tab for Channel Sequencer**

- **External Multiplexer Setup**: SYSMON supports a timing mode that uses an external analog multiplexer when device I/O resources might be limited, or when auxiliary analog I/O are more valuable when used to implement another interface.
  - **External Multiplexer**: Enables the external multiplexer.
  - **Channel for MUX**: Specifies the external channel to which the MUX connects.
  - **External Channel Configuration**: All external channels Vp/Vn and VAUXP/N [0-15] are available in this tab. Use this to select channels for monitoring, enable averaging, enable bipolar mode and increase the acquisition time for selected channels.
- **VAUX Analog Bank Selection**: Available banks of selected part that contains the VAUX pin pairs. Constraints (XDC) for the enabled VAUX channels are written by the System Management Wizard for the selected bank.

IMPORTANT: *If an SSI device is selected, the analog bank selection of the SSI slave bank connects the VAUX pin to the slave SYSMONE.*

Generating the HDL Wrapper

After selecting the configuration options, click **OK** on the System Management Wizard screen to generate the HDL wrapper and other System Management Wizard outputs.

The output files are placed in the `<project_name>/<project_name>.srcs/sources_1/ip/<component_name>/` directory you selected or created when setting up a new Vivado Design Environment project.

User Parameters

Table 4-2 shows the relationship between the GUI fields in the Vivado IDE and the User Parameters (which can be viewed in the Tcl console).

Table 4-2: **GUI Parameter to User Parameter Relationship**

Vivado IDE Parameter(1)	User Parameter/Value(1)	Default Value
Basic Tab		
Component Name	Component_Name	system_management_wiz_0
Interface Options	INTERFACE_SELECTION	DRP
ADC Setup		
Startup Channel Selection	SYSMON_STARTUP_SELECTION	Channel Sequencer
Sequencer Mode	SEQUENCER_MODE	Continuous
Channel Averaging	CHANNEL_AVERAGING	None
ADC Calibration		
Offset Calibration	ADC_OFFSET_CALIBRATION	FALSE
Calibration Averaging	ENABLE_CALIBRATION_AVERAGING	TRUE
Supply Sensor Calibration		
Offset Calibration	SENSOR_OFFSET_CALIBRATION	FALSE
Offset and Gain Calibration	SENSOR_OFFSET_AND_GAIN_CALIBRATION	TRUE
SYSMON I2C Options		
Enable I2C on SLR0 (only Master)	ENABLE_I2C	FALSE
Enable I2C on all SLRs	ENABLE_I2C_SLAVE	FALSE
SYSMON I2C Address Override	I2C_ADDRESS_OVERRIDE	FALSE
SYSMON I2C SLAVE Address (in Hex)	I2C_SLAVE_ADDRESS	00
Timing Mode	TIMING_MODE	Continuous Mode
DRP Timing Options		

Table 4-2: **GUI Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter[1]	User Parameter/Value[1]	Default Value
Enable DCLK	ENABLE_DCLK	TRUE
DCLK Frequency	DCLK_FREQUENCY	100
ADC Conversion Rate	ADC_CONVERSION_RATE	200
Acquisition Time (CLK)	INCREASE_ACQUISITION_TIME	FALSE
Enable Control/Status Ports		
reset_in	ENABLE_RESET	TRUE
JTAG Arbiter	ENABLE_JTAG_ARBITER	FALSE
Temperature Bus	ENABLE_TEMP_BUS	FALSE
Enable Convst_in	ENABLE_CONVST	convst in
Analog Sim Options		
Sim File Selection	SIM_FILE_SEL	Default
Stimulus File	SIM_FILE_NAME	design
Sim File location	SIM_FILE_REL_PATH	"./"
Waveform Type	WAVEFORM_TYPE	CONSTANT
Frequency(KHz)	STIMULUS_FREQ	0.1
Number of Wave	NUM_WAVE	1
On Chip Sensor Selection Tab		
Calibration		
Channel Enable	CHANNEL_ENABLE_CALIBRATION	TRUE
Average Enable	-	-
Temperature	-	-
Channel Enable	CHANNEL_ENABLE_TEMPERATURE	TRUE
Average Enable	AVERAGE_ENABLE_TEMPERATURE	FALSE
VCCINT		
Channel Enable	CHANNEL_ENABLE_VCCINT	TRUE
Average Enable	AVERAGE_ENABLE_VCCINT	FALSE
VCCAUX		
Channel Enable	CHANNEL_ENABLE_VCCAUX	TRUE
Average Enable	AVERAGE_ENABLE_VCCAUX	FALSE
VCCBRAM		
Channel Enable	CHANNEL_ENABLE_VCCBRAM	TRUE
Average Enable	AVERAGE_ENABLE_VCCBRAM	FALSE
VREFP		
Channel Enable	CHANNEL_ENABLE_VREFP	FALSE
Average Enable	-	-

Table 4-2: **GUI Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter[1]	User Parameter/Value[1]	Default Value
VREFN		
Channel Enable	CHANNEL_ENABLE_VREFN	FALSE
Average Enable	-	-
Reference Used by SYSMON	REFERENCE	Internal
Over Temperature Alarm	OT_ALARM	TRUE
Trigger	TEMPERATURE_ALARM_OT_TRIGGER	125
Reset	TEMPERATURE_ALARM_OT_RESET	70
User Temperature Alarm	USER_TEMP_ALARM	TRUE
Trigger	TEMPERATURE_ALARM_TRIGGER	85
Reset	TEMPERATURE_ALARM_RESET	60
VCCINT Alarm (Volts)	VCCINT_ALARM	TRUE
Trigger	VCCINT_ALARM_LOWER	0.86
Reset	VCCINT_ALARM_UPPER	0.92
VCCAUX Alarm (Volts)	VCCAUX_ALARM	TRUE
Trigger	VCCAUX_ALARM_LOWER	1.75
Reset	VCCAUX_ALARM_UPPER	1.89
VCCBRAM Alarm (Volts)	ENABLE_VBRAM_ALARM	FALSE
Trigger	VBRAM_ALARM_LOWER	0.86
Reset	VBRAM_ALARM_UPPER	0.92
On-Chip Sensor User Supply Tab		
VUSER0		
Channel Enable	CHANNEL_ENABLE_VUSER0	FALSE
Average Enable	AVERAGE_ENABLE_VUSER0	FALSE
VUSER1		
Channel Enable	CHANNEL_ENABLE_VUSER1	FALSE
Average Enable	AVERAGE_ENABLE_VUSER1	FALSE
VUSER2		
Channel Enable	CHANNEL_ENABLE_VUSER2	FALSE
Average Enable	AVERAGE_ENABLE_VUSER2	FALSE
VUSER3		
Channel Enable	CHANNEL_ENABLE_VUSER3	FALSE
Average Enable	AVERAGE_ENABLE_VUSER3	FALSE
User Supply0 Alarm (Volts)	USER_SUPPLY0_ALARM	FALSE
Voltage Level	SELECT_USER_SUPPLY0_LEVEL	1.8
Lower	USER_SUPPLY0_ALARM_LOWER	1.79

Table 4-2: **GUI Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter[1]	User Parameter/Value[1]	Default Value
Upper	USER_SUPPLY0_ALARM_UPPER	1.81
User Supply1 Alarm (Volts)	USER_SUPPLY1_ALARM	FALSE
Lower	USER_SUPPLY1_ALARM_LOWER	0.89
Upper	USER_SUPPLY1_ALARM_UPPER	0.91
User Supply2 Alarm (Volts)	USER_SUPPLY2_ALARM	FALSE
Lower	USER_SUPPLY2_ALARM_LOWER	1.75
Upper	USER_SUPPLY2_ALARM_UPPER	1.81
User Supply3 Alarm (Volts)	USER_SUPPLY3_ALARM	FALSE
Voltage Level	SELECT_USER_SUPPLY3_LEVEL	1.8
Lower	USER_SUPPLY3_ALARM_LOWER	1.79
Upper	USER_SUPPLY3_ALARM_UPPER	1.81
User Supply Selection for VUSER		
USER SUPPLY0 BANK	USER_SUPPLY0_BANK	44
Select User Supply0	SELECT_USER_SUPPLY0	VCCO
USER SUPPLY1 BANK	USER_SUPPLY1_BANK	44
Select User Supply1	SELECT_USER_SUPPLY1	VCCINT
USER SUPPLY2 BANK	USER_SUPPLY2_BANK	44
Select User Supply2	SELECT_USER_SUPPLY2	VCCAUX
USER SUPPLY3 BANK	USER_SUPPLY3_BANK	65
Select User Supply3	SELECT_USER_SUPPLY3	VCCO BOT
SSIT Slave0 Sensors (Enabled only for SSIT devices)		
Temperature_Slave0		
Channel Enable	CHANNEL_ENABLE_TEMPERATURE_SLAVE0_SSIT	TRUE
Average Enable	AVERAGE_ENABLE_TEMPERATURE_SLAVE0_SSIT	FALSE
VUSER0_SLAVE0		
Channel Enable	CHANNEL_ENABLE_VUSER0_SLAVE0_SSIT	FALSE
Average Enable	AVERAGE_ENABLE_VUSER0_SLAVE0_SSIT	FALSE
VUSER1_SLAVE0		
Channel Enable	CHANNEL_ENABLE_VUSER1_SLAVE0_SSIT	FALSE
Average Enable	AVERAGE_ENABLE_VUSER1_SLAVE0_SSIT	FALSE
VUSER2_SLAVE0		
Channel Enable	CHANNEL_ENABLE_VUSER2_SLAVE0_SSIT	FALSE
Average Enable	AVERAGE_ENABLE_VUSER2_SLAVE0_SSIT	FALSE
VUSER3_SLAVE0		
Channel Enable	CHANNEL_ENABLE_VUSER3_SLAVE0_SSIT	FALSE

Table 4-2: **GUI Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter[1]	User Parameter/Value[1]	Default Value
Average Enable	AVERAGE_ENABLE_VUSER3_SLAVE0_SSIT	FALSE
User Supply0 Alarm (Volts)	USER_SUPPLY0_SLAVE0_SSIT_ALARM	FALSE
Voltage Level	SELECT_USER_SUPPLY0_SLAVE0_SSIT_LEVEL	1.8
Lower	USER_SUPPLY0_SLAVE0_SSIT_ALARM_LOWER	1.79
Upper	USER_SUPPLY0_SLAVE0_SSIT_ALARM_UPPER	1.81
User Supply1 Alarm (Volts)	USER_SUPPLY1_SLAVE0_SSIT_ALARM	FALSE
Lower	USER_SUPPLY1_SLAVE0_SSIT_ALARM_LOWER	0.89
Upper	USER_SUPPLY1_SLAVE0_SSIT_ALARM_UPPER	0.91
User Supply2 Alarm (Volts)	USER_SUPPLY2_SLAVE0_SSIT_ALARM	FALSE
Lower	USER_SUPPLY2_SLAVE0_SSIT_ALARM_LOWER	1.79
Upper	USER_SUPPLY2_SLAVE0_SSIT_ALARM_UPPER	1.81
User Supply3 Alarm (Volts)	USER_SUPPLY3_SLAVE0_SSIT_ALARM	FALSE
Voltage Level	SELECT_USER_SUPPLY3_SLAVE0_SSIT_LEVEL	1.8
Lower	USER_SUPPLY3_SLAVE0_SSIT_ALARM_LOWER	1.79
Upper	USER_SUPPLY3_SLAVE0_SSIT_ALARM_UPPER	1.81
User Supply Selection for VUSER Slave0		
USER SUPPLY0 SLAVE0 SSIT BANK	USER_SUPPLY0_SLAVE0_SSIT_BANK	51
Select User Supply0	SELECT_USER_SUPPLY0_SLAVE0_SSIT	VCCO
USER SUPPLY1 SLAVE0 SSIT BANK	USER_SUPPLY1_SLAVE0_SSIT_BANK	51
Select User Supply1	SELECT_USER_SUPPLY1_SLAVE0_SSIT	VCCINT
USER SUPPLY2 SLAVE0 SSIT BANK	USER_SUPPLY2_SLAVE0_SSIT_BANK	51
Select User Supply2	SELECT_USER_SUPPLY2_SLAVE0_SSIT	VCCAUX
USER SUPPLY3 SLAVE0 SSIT BANK	USER_SUPPLY3_SLAVE0_SSIT_BANK	51
Select User Supply3	SELECT_USER_SUPPLY3_SLAVE0_SSIT	VCCO
SSIT Slave1 Sensors (Enabled only for SSIT devices)		
Temperature_Slave0		
Channel Enable	CHANNEL_ENABLE_TEMPERATURE_SLAVE1_SSIT	TRUE
Average Enable	AVERAGE_ENABLE_TEMPERATURE_SLAVE1_SSIT	FALSE
VUSER0_SLAVE1		
Channel Enable	CHANNEL_ENABLE_VUSER0_SLAVE1_SSIT	FALSE
Average Enable	AVERAGE_ENABLE_VUSER0_SLAVE1_SSIT	FALSE
VUSER1_SLAVE1		

Table 4-2: **GUI Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter[1]	User Parameter/Value[1]	Default Value
Channel Enable	CHANNEL_ENABLE_VUSER1_SLAVE1_SSIT	FALSE
Average Enable	AVERAGE_ENABLE_VUSER1_SLAVE1_SSIT	FALSE
VUSER2_SLAVE1		
Channel Enable	CHANNEL_ENABLE_VUSER2_SLAVE1_SSIT	FALSE
Average Enable	AVERAGE_ENABLE_VUSER2_SLAVE1_SSIT	FALSE
VUSER3_SLAVE1		
Channel Enable	CHANNEL_ENABLE_VUSER3_SLAVE1_SSIT	FALSE
Average Enable	AVERAGE_ENABLE_VUSER3_SLAVE1_SSIT	FALSE
User Supply0 Alarm (Volts)	USER_SUPPLY0_SLAVE1_SSIT_ALARM	FALSE
Voltage Level	SELECT_USER_SUPPLY0_SLAVE1_SSIT_LEVEL	1.8
Lower	USER_SUPPLY0_SLAVE1_SSIT_ALARM_LOWER	1.79
Upper	USER_SUPPLY0_SLAVE1_SSIT_ALARM_UPPER	1.81
User Supply1 Alarm (Volts)	USER_SUPPLY1_SLAVE1_SSIT_ALARM	FALSE
Lower	USER_SUPPLY1_SLAVE1_SSIT_ALARM_LOWER	0.89
Upper	USER_SUPPLY1_SLAVE1_SSIT_ALARM_UPPER	0.91
User Supply2 Alarm (Volts)	USER_SUPPLY2_SLAVE1_SSIT_ALARM	FALSE
Lower	USER_SUPPLY2_SLAVE1_SSIT_ALARM_LOWER	1.79
Upper	USER_SUPPLY2_SLAVE1_SSIT_ALARM_UPPER	1.81
User Supply3 Alarm (Volts)	USER_SUPPLY3_SLAVE1_SSIT_ALARM	FALSE
Voltage Level	SELECT_USER_SUPPLY3_SLAVE1_SSIT_LEVEL	1.8
Lower	USER_SUPPLY3_SLAVE1_SSIT_ALARM_LOWER	1.79
Upper	USER_SUPPLY3_SLAVE1_SSIT_ALARM_UPPER	1.81
User Supply Selection for VUSER Slave1		
USER SUPPLY0 SLAVE1 SSIT BANK	USER_SUPPLY0_SLAVE1_SSIT_BANK	51
Select User Supply0	SELECT_USER_SUPPLY0_SLAVE1_SSIT	VCCO
USER SUPPLY1 SLAVE1 SSIT BANK	USER_SUPPLY1_SLAVE1_SSIT_BANK	51
Select User Supply1	SELECT_USER_SUPPLY1_SLAVE1_SSIT	VCCINT
USER SUPPLY2 SLAVE1 SSIT BANK	USER_SUPPLY2_SLAVE1_SSIT_BANK	51
Select User Supply2	SELECT_USER_SUPPLY2_SLAVE1_SSIT	VCCAUX
USER SUPPLY3 SLAVE1 SSIT BANK	USER_SUPPLY3_SLAVE1_SSIT_BANK	51
Select User Supply3	SELECT_USER_SUPPLY3_SLAVE1_SSIT	VCCO
External Channel Selection Tab		

Table 4-2: **GUI Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter[1]	User Parameter/Value[1]	Default Value
External Multiplexer	ENABLE_EXTERNAL_MUX	FALSE
Channel for MUX	EXTERNAL_MUX_CHANNEL	FALSE
Enable muxaddr_out port	EXTERNAL_MUXADDR_ENABLE	
vp/vn		
Channel Enable	CHANNEL_ENABLE_VP_VN	TRUE
Average Enable	AVERAGE_ENABLE_VP_VN	FALSE
Bipolar	BIPOLAR_VP_VN	FALSE
Acquisition Time	ACQUISITION_TIME_VP_VN	FALSE
VAUX Analog Bank Selection		
Analog Bank Selection	ANALOG_BANK_SELECTION	44
vausxp<0-15>/vauxn<0-15>		
Channel Enable	CHANNEL_ENABLE_VAUXP<0-15>_VAUXN <0-15>	FALSE
Average Enable	AVERAGE_ENABLE_VAUXP<0-15>_VAUXN <0-15>	FALSE
Bipolar	BIPOLAR_VAUXP<0-15>_VAUXN<0-15>	FALSE
Acquisition Time	ACQUISITION_TIME_VAUXP<0-15>_VAUXN <0-15>	FALSE

1. Parameter values are listed in the table where the GUI parameter value differs from the user parameter value. Such values are shown in this table as indented below the associated parameter.

Output Generation

For details, see the *Vivado Design Suite User Guide: Designing with IP* (UG896) [Ref 5].

Constraining the Core

This section contains information about constraining the core in the Vivado® Design Suite.

Required Constraints

For the AXI4-Lite interface, the required constraint is:

```
create_clock -period <period-in-ns> [get_ports s_axi_aclk]
```

For the DRP interface, required constraint is:

```
create_clock -period <period-in-ns> [get_ports dclk_in]
```

The System Management Wizard writes the required ANALOG IOSTANDARD constraint on VP/VN ports. Setting the **Analog Bank Selection** for vaux pins (shown in Figure 4-13) writes the pin LOC and IOSTANDARD constraint for vaux ports.

Clock Frequencies

The System Management Wizard supports clock frequencies 8 to 250 MHz.

Clock Management

Depending on the configuration, the ADC clock is internally divided by the SYSMON primitive to achieve the desired sampling rate.

Simulation

This section contains information about simulating IP in the Vivado® Design Suite.

Analog waveform simulation is performed using the `design.txt` file which contains the time reference and the analog values for a selected channel. This file is generated by default. The analog and its digital equivalence comparison is in the example design test bench to verify the SYSMON behavior.

You can provide your own waveform in a file using the relative path option in the Vivado IDE. In this case, the comparison values should be updated with respect to the analog stimulus to complete the example design simulation without error.

Simulation of channel averaging is not supported in the System Management Wizard example design test bench.

For details, see the *Vivado Design Suite User Guide: Designing with IP* (UG896) [Ref 5].

Synthesis and Implementation

This section contains information about synthesis and implementation in the Vivado® Design Suite.

UltraScale™ devices need LOC constraints for VAUXP/VAUXN pin pairs to be specified in XDC. VP/VN is a dedicated input and does not need any pin LOC constraint, but ANALOG IOSTANDARD is required for implementation. Each UltraScale device bank contains 16 dual IO pin pairs to support analog IO functionality. The System Management Wizard generates these constraints depending on the user selected bank for VAUX pin pairs.

To support the I^2C interface, System Management Wizard generates pin LOC constraints on the dual purpose I^2C pin for the implementation.

For details about synthesis and implementation, see the *Vivado Design Suite User Guide: Designing with IP* (UG896) [Ref 5].

Example Design

This chapter contains information about the example design provided in the Vivado® Design Suite.

Note: Example design is not supported when I^2c is enabled and SYSMON I^2C Address Override option is not selected in the Vivado IDE.

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

Verilog

```
<project_name>/<project_name>.srcs/sources_1/ip/<component_name>/example_design/
<component_name>_exdes.v
```

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

Open Example Project Flow

In the Vivado Design Environment, use the following command to create an example project flow:

```
open_example_project [get_ips <component_name>]
```

Use of this command in the Tcl Console invokes a separate example design project that creates `<component_name>_exdes` as the top module for synthesis and `<component_name>_tb` as the top module for simulation. Implementation or simulation of the example design can be run from the example project.

Note: When I^2C is enabled in the wizard and you want to see the example design, address override option needs to be enabled.

Test Bench

This chapter contains information about the test bench provided in the Vivado® Design Suite.

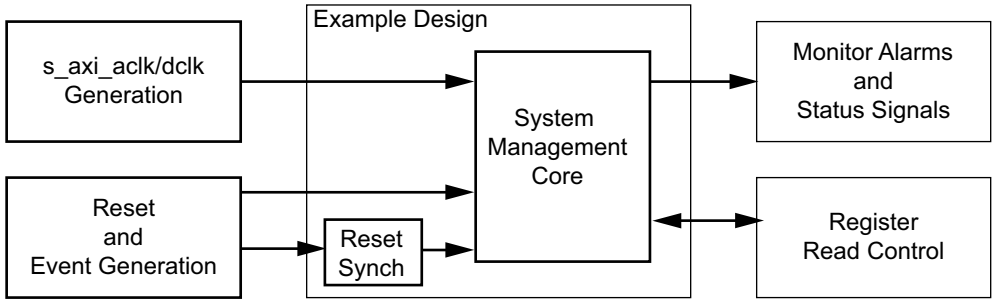

Figure 6-1: **System Management Wizard Test Bench**

The following files describe the demonstration test bench.

Verilog

```
<project_name>/<project_name>.srcs/sources_1/ip/<component_name>/simulation/
<component_name>_tb.v
```

The demonstration test bench is a simple Verilog program to exercise the example design and the core. The demonstration test bench performs the following tasks:

- Generates the input `s_axi_aclk/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.

Debugging

This appendix includes details about resources available on the Xilinx Support website and debugging tools.

TIP: *If the IP generation halts with an error, there might be a license issue. See License Checkers in Chapter 1 for more details.*

Finding Help on Xilinx.com

To help in the design and debug process when using the System Management Wizard, the Xilinx Support web page contains key resources such as product documentation, release notes, answer records, information about known issues, and links for obtaining further product support.

Documentation

This product guide is the main document associated with the System Management Wizard. This guide, along with documentation related to all products that aid in the design process, can be found on the Xilinx Support web page or by using the Xilinx Documentation Navigator.

Download the Xilinx Documentation Navigator from the Design Tools tab on the Downloads page. For more information about this tool and the features available, open the online help after installation.

Answer Records

Answer Records include information about commonly encountered problems, helpful information on how to resolve these problems, and any known issues with a Xilinx product. Answer Records are created and maintained daily ensuring that users have access to the most accurate information available.

Answer Records for this core can be located by using the Search Support box on the main Xilinx support web page. To maximize your search results, use proper keywords such as

- Product name

- Tool message(s)
- Summary of the issue encountered

A filter search is available after results are returned to further target the results.

Master Answer Record for the System Management Wizard

AR: 58763

Technical Support

Xilinx provides technical support in the Xilinx Support web page for this LogiCORE™ IP product when used as described in the product documentation. Xilinx cannot guarantee timing, functionality, or support if you do any of the following:

- Implement the solution in devices that are not defined in the documentation.
- Customize the solution beyond that allowed in the product documentation.
- Change any section of the design labeled DO NOT MODIFY.

To contact Xilinx Technical Support, navigate to the Xilinx Support web page.

Debug Tools

There are many tools available to address System Management Wizard design issues. It is important to know which tools are useful for debugging various situations.

Vivado Design Suite Debug Feature

The Vivado® Design Suite debug feature inserts logic analyzer and virtual I/O cores directly into your design. The debug feature also allows you to set trigger conditions to capture application and integrated block port signals in hardware. Captured signals can then be analyzed. This feature in the Vivado IDE is used for logic debugging and validation of a design running in Xilinx.

The Vivado Lab Edition logic analyzer is used to interact with the logic debug LogiCORE IP cores, including:

- ILA 2.0 (and later versions)
- VIO 2.0 (and later versions)

See *Vivado Design Suite User Guide: Programming and Debugging* (UG908) [Ref 6].

Reference Boards

Various Xilinx development boards support the System Management Wizard. These boards can be used to prototype designs and establish that the core can communicate with the system.

Simulation Debug

The simulation debug flow for Questa® SIM is illustrated below. A similar approach can be used with other simulators.

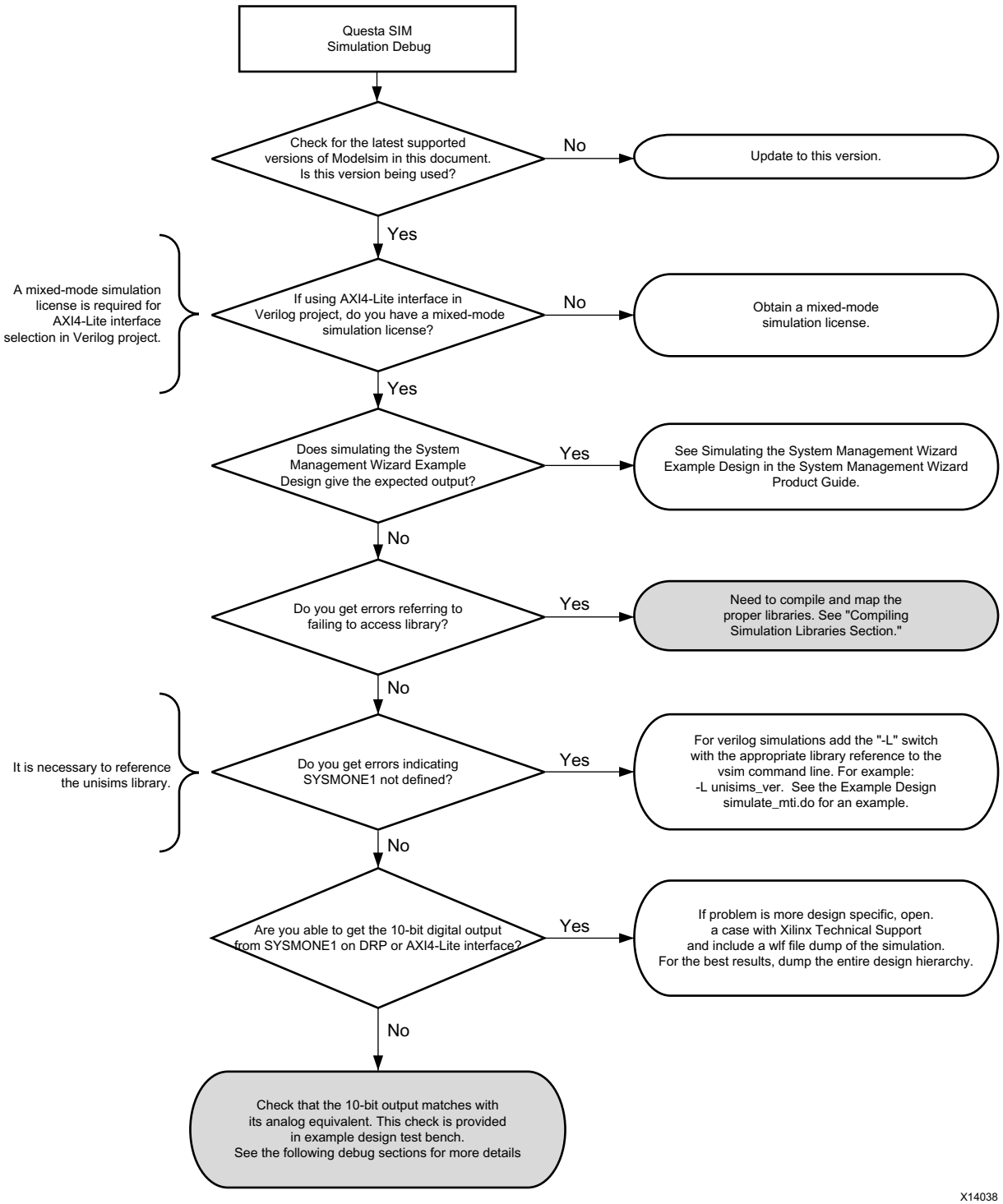

Figure A-1: **Questa SIM Debug Flow Diagram**

Hardware Debug

Hardware issues can range from link bring-up to problems seen after hours of testing. This section provides debug steps for common issues. The Vivado Lab Edition are a valuable resource to use in hardware debug. The signal names mentioned in the following individual sections can be probed using the Vivado Lab Edition for debugging the specific problems.

General Checks

Ensure that all the timing constraints for the core were properly incorporated from the example design and that all constraints were met during implementation.

- Does it work in post-place and route timing simulation? If problems are seen in hardware but not in timing simulation, this could indicate a PCB issue. Ensure that all clock sources are active and clean.
- If using MMCMs in the design, ensure that all MMCMs have obtained lock by monitoring the `locked` port.
- If your outputs go to 0, check your licensing.

Interface Debug

AXI4-Lite Interfaces

Read from a register that does not have all 0s as a default to verify that the interface is functional. Output `s_axi_arready` asserts when the read address is valid, and output `s_axi_rvalid` asserts when the read data/response is valid. If the interface is unresponsive, ensure that the following conditions are met:

- The `s_axi_aclk and aclk` inputs are connected and toggling.
- The interface is not being held in reset, and `s_axi_areset` is an active-Low reset.
- The interface is enabled, and `s_axi_aclken` is active-High (if used).
- The main core clocks are toggling and that the enables are also asserted.
- If the simulation has been run, verify in simulation and/or a Vivado Lab Edition capture that the waveform is correct for accessing the AXI4-Lite interface.

Appendix B

Additional Resources and Legal Notices

Xilinx Resources

For support resources such as Answers, Documentation, Downloads, and Forums, see Xilinx Support.

References

These documents provide supplemental material useful with this product guide:

1. *UltraScale Architecture System Monitor Advanced Specification User Guide* (UG580)
2. *Vivado Design Suite User Guide: Logic Simulation* (UG900)
3. *Vivado Design Suite User Guide: Designing IP Subsystems using IP Integrator* (UG994)
4. *Vivado Design Suite User Guide: Implementation* (UG904)
5. *Vivado Design Suite User Guide: Designing with IP* (UG896)
6. *Vivado Design Suite User Guide: Programming and Debugging* (UG908)
7. *Vivado Design Suite User Guide: Getting Started* (UG910)

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
04/06/2016	1.3	• Added UltraScale+ support. • Added PMBus interface. • Added SYSMON4 features. • Added register description for AXI-Lite interface.
09/30/2015	1.2	• Changed control register from SYSMONE1 to SYSMON. • Updated I2C addressing for SSIT devices.

Date	Version	Revision
04/01/2015	1.2	• Added I2C Interface for SSIT Devices section. • Added User Parameter section. • Updated Customizing and Generating the Core section.
10/01/2014	1.1	• Added ANALOG constraints on VP/VN ports. • Added SIN, TRIANGLE and SQUARE stimulus generation in the range 0.1 KHz to 96 KHz. • Added CSV file format to TXT file conversion for simulation.
06/04/2014	1.1	• Added details for designs using SSI devices. • Updated width of AXI Write Address and AXI Read Address signals to 12:0. • Added sysmon_slave_sel[1:0] signal.
04/02/2014	1.0	Updated IP core registers.
12/18/2013	1.0	Initial Xilinx release.

Please Read: Important Legal Notices