IP CORE MANUAL

ADS42LB69 ADC Interface Core

px_ads42lb69intrfc

PENTEK

Pentek, Inc.
One Park Way
Upper Saddle River, NJ 07458
(201) 818-5900
http://www.pentek.com/

Manual Part Number: 807.48371 Rev: 1.0 - February 07, 2017

Manual Revision History

Date	Version	Comments
02/07/17	1.0	Initial Release

Legal Notices

The information disclosed to you hereunder (the "Materials") is provided solely for the selection and use of Pentek products. To the maximum extent permitted by applicable law: (1) Materials are made available "AS IS" and with all faults, Pentek hereby DISCLAIMS ALL WARRANTIES AND CONDITIONS, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE; and (2) Pentek shall not be liable (whether in contract or tort, including negligence, or under any other theory of liability) for any loss or damage of any kind or nature related to, arising under, or in conjunction with, the Materials (including your use of Materials), including for any direct, indirect, special, incidental, or consequential loss or damage (including loss of data, profits, goodwill, or any type of loss or damage suffered as a result of any action brought by a third party) even if such damage and loss was reasonably foreseeable or Pentek had been advised of the possibility of the same. Pentek assumes no obligation to correct any error contained in the Materials or to notify you of updates to the Materials or to product specifications. You may not reproduce, modify, distribute, or publicly display the materials without prior written consent. Certain products are subject to the terms and conditions of Pentek's limited warranty, please refer to Pentek's Ordering and Warranty information which can be viewed at http://www.pentek.com/contact/customerinfo.cfm; IP cores may be subject to warranty and support terms contained in a license issued to you by Pentek. Pentek products are not designed or intended to be fail-safe or for use in any application requiring fail-safe performance; you assume sole risk and liability for the use of Pentek products in such critical applications.

Copyright

Trademarks

Pentek, Jade, and Navigator are trademarks or registered trademarks of Pentek, Inc.

ARM and AMBA are registered trademarks of ARM Limited. PCI, PCI Express, PCIe, and PCI-SIG are trademarks or registered trademarks of PCI-SIG. Texas Instruments is a trademark of Texas Instruments, Incorporated. Xilinx, Kintex UltraScale, Vivado, and Platform Cable USB are registered trademarks of Xilinx Inc., of San Jose, CA.

Printed in the United States of America.

Table of Contents

Page

Table of Contents

Page

Chapter 4: Register Space

Chapter 5: Designing with the Core

Table of Contents

Page

Chapter 6: Design Flow Steps

Table of Contents

Page

This page is intentionally blank

IP Facts

Description

Pentek's Navigator™ ADS42LB69 ADC Interface Core serves as an interface to the Texas Instruments™ Dual Channel ADS42LB69 Analog to Digital converter.

This core complies with the ARM® AMBA® AXI4 specification. This manual defines the hardware interface, software interface, and parameterization options for the ADS42LB69 ADC Interface Core.

Features

- Provides offset and gain adjustment to the ADC Data
- Register access through AXI4-Lite interface
- Output data through AXI4-Stream Interface
- Differential Termination can be applied to inputs based on the user application requirement
- Performs data rounding and saturation operation after the offset and gain stage
- Detects data overload based on user-defined overload settings
- Allows user to implement input buffers in low power or high performance mode
- Software programmable input tap delay at reset
- ADS42LB69 ADC channels power down at reset can be enabled by the user
- Input delay control with adjustable reference clock frequency
- Supports control of the ADS42LB69 ADC internal registers through a serial interface

Table 1-1: IP Facts Table

Core Specifics	
Supported Design Family[a]	Kintex® Ultrascale
Supported User Interfaces	AXI4-Lite and AXI4-Stream
Resources	See Table 2-1
Provided with the Core	
Design Files	VHDL
Example Design	Not Provided
Test Bench	Not Provided[b]
Constraints File	Not Provided[c]
Simulation Model	N/A
Supported S/W Driver	HAL Software Support
Tested Design Flows	
Design Entry	Vivado® Design Suite 2016.4 or later
Simulation	Vivado VSim
Synthesis	Vivado Synthesis
Support	
Provided by Pentek fpgasupport@pentek.com	

a.For a complete list of supported devices, see the *Vivado Design Suite Release Notes*.
b.The test bench will be available in the next revision of this core.
c.Clock constraints can be applied at the top level module of the user design.

This page is intentionally blank

Chapter 1: Overview

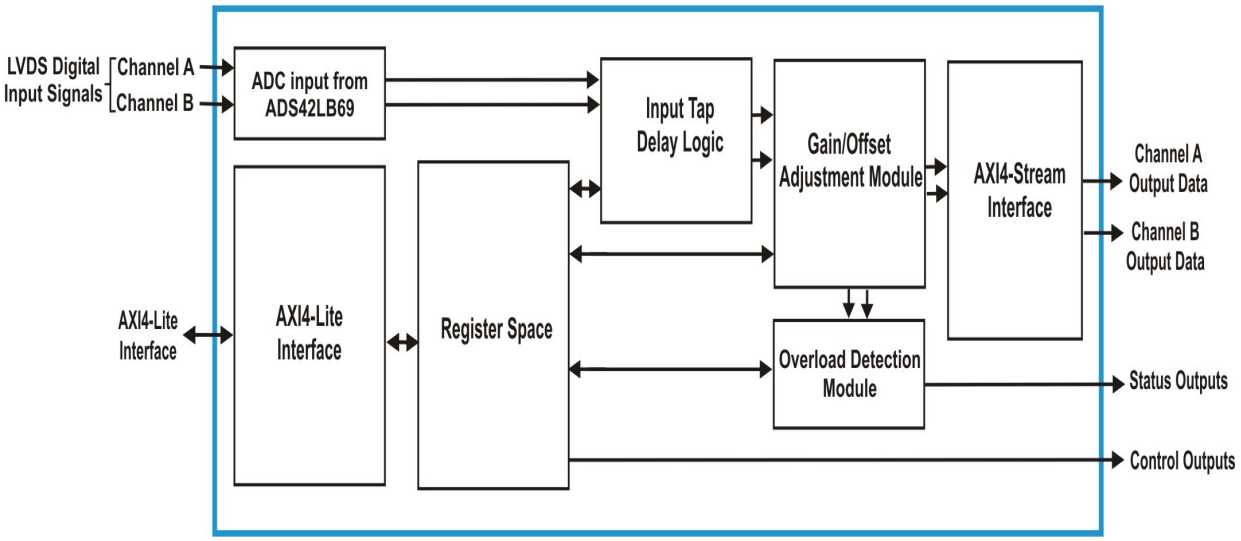
1.1 Functional Description

The ADS42LB69 ADC Interface Core provides a transaction interface to the AXI4-Lite Interface. The AXI4-Lite interface acts as a slave and is connected to the Register Space as shown in Figure 1-1, below.

The ADC Core (ADS42LB69 ADC Interface Core) also has LVDS digital inputs from the two channels of the ADS42LB69 ADC. The input data from each channel is passed through the input tap delay, gain and offset adjustment and overload detection modules before obtaining the desired output through the AXI4-Stream Interface of the core for each channel. Control outputs from the core are directed to the ADS42LB69 ADC. This core also has a serial interface to the ADS42LB69 ADC to access its internal registers.

Figure 1-1 is the top level block diagram of the ADS42LB69 ADC Interface Core. The modules within the block diagram are explained in the later sections of this manual.

Figure 1-1: ADS42LB69 ADC Interface Core Block Diagram



- **AXI4-Lite Interface:** This module implements a 32-bit AXI4-Lite Slave interface to access the Register Space. For additional details about the AXI4-Lite Interface, refer to the Section 3.1 AXI4-Lite Core Interfaces.

- **Register Space:** This module contains the control and status registers including Interrupt Enable, Interrupt Flag, and Interrupt Status registers. Registers are accessed through the AXI4-Lite interface.

1.1 Functional Description (continued)

- **Input Tap Delay Logic:** This module introduces delay to the incoming data. The delay to be introduced is determined from the generic parameter **initial_tap_delay**, defined by the user. Tap delay is used to achieve proper setup time on the input data lines.

- **Gain and Offset Adjustment:** This module serves to allow for gain and offset adjustment of the ADC data and also performs rounding, saturation operations on the data. This module returns a 18-bit signed value.

- **Overload Detection:** This module detects an overload in the ADC data and the gain/offset data. It indicates an overload through the overload LED.

- **AXI4-Stream Interface:** This module implements a 16-bit AXI4-Stream Master interface for the output data streams from the two channels of the ADC Core. For additional details about the AXI4-Stream Interface, refer to the Section 3.2 AXI4-Stream Core Interfaces.

1.2 Applications

This core can be used as an interface to the Texas Instruments ADS42LB69 Analog to Digital converter.

1.3 System Requirements

For a list of system requirements, see the Vivado Design Suite Release Notes.

1.4 Licensing and Ordering Information

This core is included with all Pentek Navigator FPGA Design Kits for Pentek Jade series board products. Contact Pentek for Licensing and Ordering Information (www.pentek.com).

1.5 Contacting Technical Support

Technical Support for Pentek's Navigator FPGA Design Kits is available via e-mail (fpgasupport@pentek.com) or by phone (201-818-5900 ext. 238, 9 am to 5 pm EST).

1.6 Documentation

This user manual is the main document for this IP core. The following documents provide supplemental material:

1) *Vivado Design Suite User Guide: Designing with IP*

2) *Vivado Design Suite User Guide: Programming and Debugging*

3) *ARM AMBA AXI4 Protocol Version 2.0 Specification*
http://www.arm.com/products/system-ip/amba-specifications.php

4) *Texas Instruments ADS42LB69 Analog to Digital Converter Datasheet*

This page is intentionally blank

Chapter 2: General Product Specifications

2.1 Standards

The ADS42LB69 ADC Interface Core has bus interfaces that comply with the A*RM AMBA AXI4-Lite Protocol Specification* and the *AMBA AXI4-Stream Protocol Specification*.

2.2 Performance

The performance of the ADC Core is limited by the maximum operating frequency of the ADS42LB69 Analog to Digital Converter. The values presented in this section should be used as an estimation guideline. Actual performance can vary.

2.2.1 Maximum Frequencies

The ADS42LB69 ADC has a maximum operating frequency of 250 MHz. The ADC Core is therefore designed to run at a maximum clock frequency of 250 MHz.

2.3 Resource Utilization

The resource utilization of the ADS42LB69 ADC Interface Core is shown in Table 2-1. Resources have been estimated for the Kintex Ultrascale XCKU060 -2 speed grade device. These values were generated using the Vivado Design Suite.

Table 2-1: Resource Usage and Availability

Resource	# Used
LUTs	202
Flip-Flops	656

NOTE: Actual utilization may vary based on the user design in which the ADS42LB69 ADC Interface Core is incorporated.

2.4 Limitations and Unsupported Features

This section is not applicable to this IP core.

2.5 Generic Parameters

The generic parameters of the ADS42LB69 ADC Interface Core are described in Table 2-2. These parameters can be set as required by the user application while customizing the core.

Table 2-2: Generic Parameters

Port/Signal Name	Type	Description
iodelay_grp_a01	String	**IO Delay Group:** This is a string label that is used to group the IODELAY tap delay components of ADC channel A with the corresponding IDELAY control component.
iodelay_grp_a23		
iodelay_grp_a45		
iodelay_grp_a67		
iodelay_grp_a89		
iodelay_grp_a1011		
iodelay_grp_a1213		
iodelay_grp_a1415		
iodelay_grp_b01	String	**IO Delay Group:** This is a string label that is used to group the IODELAY tap delay components of ADC channel B with the corresponding IDELAY control component.
iodelay_grp_b23		
iodelay_grp_b45		
iodelay_grp_b67		
iodelay_grp_b89		
iodelay_grp_b1011		
iodelay_grp_b1213		
iodelay_grp_b1415		
initial_tap_delay	Integer	**Initial Tap Delay Value:** This parameter defines the initial tap delay (in number of taps) to be introduced to the incoming data. It can range from 0 to 511.
initial_a_gain_percent		**Initial Channel A Gain Percentage:** This is the gain percentage for Channel A of the ADC core at reset, and is used to trim the gain of the input Channel A ADC signal of the core.
initial_b_gain_percent		**Initial Channel B Gain Percentage:** This is the gain percentage for Channel B of the ADC core at reset and is used to trim the gain of the input Channel B ADC signal of the core.

Table 2-2: Generic Parameters (Continued)

Port/Signal Name	Type	Description
initial_a_offset_counts	Integer	**Initial Channel A Offset Count:** This is the initial DC offset trim for Channel A of the ADC core. This value is used to minimize ADC DC offset errors. It ranges from -32767 to 32767.
initial_b_offset_counts		**Initial Channel B Offset Count:** This is the initial DC offset trim for Channel B of the ADC core. This value is used to minimize ADC DC offset errors. It ranges from -32767 to 32767.
initial_pdwnf_a	Boolean	**ADC Channel A is powered down at Reset:** Defines the power saving mode for the ADS42LB69 ADC Channel A at reset. When set to True, Channel A of the ADC is powered down at reset.
initial_pdwnf_b		**ADC Channel B is powered down at Reset:** Defines the power saving mode for the ADS42LB69 ADC Channel B at reset. When set to True, Channel B of the ADC is powered down at reset.
initial_led_select	Integer	**Initial LED Source Select at Reset:** Selects the source of the LED at reset. 0 = ADC Input Overload 1 = Gain/Offset Trim Overload 2 = External Signal 3 = Disabled
differential_term	String	**Differential Termination:** Differential termination for the input data from both channels of the ADC. TERM_NONE = No differential termination for the inputs TERM_100 = 100 Ω differential termination
ibuf_low_pwr	Boolean	**Input Buffer Low Power:** Sets the input buffer performance. True = Input buffer implemented in low power mode; False = Input buffer implemented in high performance mode
has_ext_led_src		**Has External LED Source:** Indicates whether the ADC core has an external LED source. This can be any signal from the user logic.
idelaycntrl_refclk_freq	Integer	**IDelay Control Reference Clock:** It is the reference clock frequency in MHz for the Input Tap Delay Logic block of this core. Typically this is 200 MHz.
initial_ovld_thresh		**Initial Overload threshold:** Sets the initial overload threshold value for overload detection. It can range from 0 to 32767.
led_pulse_stretch		**LED Pulse Stretcher:** The LED pulse is stretched based on this value to make short overload events more visible on the LED. It can range from 0 to 65535.

This page is intentionally blank

Chapter 3: Port Descriptions

This chapter provides details about the port descriptions for the following interface types:

- AXI4-Lite Core Interfaces
- AXI4-Stream Core Interfaces
- I/O Signals

3.1 AXI4-Lite Core Interfaces

The ADS42LB69 ADC Interface Core uses the Control/Status Register (CSR) interface to control, and receive status from, the user design.

3.1.1 Control/Status Register (CSR) Interface

The CSR interface is an AXI4-Lite Slave Interface that can be used to access the control and status registers in the ADC Core. Table 3-1 defines the ports in the CSR Interface. See Chapter 4 for a Control/Status Register memory map and bit definitions. See the *AMBA AXI4-Lite Specification* for more details on operation of the AXI4-Lite interfaces.

Table 3-1: Control/Status Register (CSR) Interface Port Descriptions

Port	Direction	Width	Description
s_axi_csr_aclk	Input	1	**Clock**
s_axi_csr_aresetn	Input	1	**Reset:** Active low. This value will reset all control registers to their initial states.
s_axi_csr_awaddr	Input	7	**Write Address:** Address used for write operations. It must be valid when **s_axi_csr_awvalid** is asserted and must be held until **s_axi_csr_awready** is asserted by the ADC Core.
s_axi_csr_awprot	Input	3	**Protection:** The ADC interface core ignores these bits.
s_axi_csr_awvalid	Input	1	**Write Address Valid:** This input must be asserted to indicate that a valid write address is available on **s_axi_csr_awaddr**. The ADC Core asserts **s_axi_csr_awready** when it is ready to accept the address. The **s_axi_csr_awvalid** must remain asserted until the rising clock edge after the assertion of **s_axi_csr_awready**.

Table 3-1: Control/Status Register (CSR) Interface Port Descriptions (Continued)

Port	Direction	Width	Description
s_axi_csr_awready	Output	1	**Write Address Ready:** This output is asserted by the ADC Core when it is ready to accept the write address.The address is latched when **s_axi_csr_awvalid** and **s_axi_csr_awready** are high on the same cycle.
s_axi_csr_wdata	Input	32	**Write Data:** This data will be written to the address specified by **s_axi_csr_awaddr** when **s_axi_csr_wvalid** and **s_axi_csr_ wready** are both asserted. The value must be valid when **s_axi_csr_wvalid** is asserted and held until **s_axi_csr_wready** is also asserted.
s_axi_csr_wstrb	Input	4	**Write Strobes:** This signal when asserted indicates the number of bytes of valid data on **s_axi_csr_wdata** signal. Each of these bits, when asserted indicate that the corresponding byte of **s_axi_csr_wdata** contains valid data. Bit 0 corresponds to the least significant byte, and bit 3 to the most significant.
s_axi_csr_wvalid	Input	1	**Write Valid:** This signal must be asserted to indicate that the write data is valid for a write operation. The value on **s_axi_csr_wdata** is written into the register at address **s_axi_csr_awaddr** when **s_axi_csr_wready** and **s_axi_csr_wvalid** are high on the same cycle.
s_axi_csr_wready	Output		**Write Ready:** This signal is asserted by the ADC Core when it is ready to accept data. The value on **s_axi_csr_wdata** is written into the register at address **s_axi_csr_awaddr** when **s_axi_csr_wready** and **s_axi_csr_wvalid** are high on the same cycle, assuming that the address has already or simultaneously been submitted.
s_axi_csr_bresp	Output	2	**Write Response:** The core indicates success or failure of a write transaction through this signal, which is valid when **s_axi_csr_ bvalid** is asserted; 00 = Success of normal access 01 = Success of exclusive access 10 = Slave Error 11 = Decode Error Note: For more details about this signal refer to the *AMBA AXI Specification*.
s_axi_csr_bready	Input	1	**Write Response Ready:** This signal must be asserted by the user logic when it is ready to accept the Write Response.
s_axi_csr_bvalid	Output	1	**Write Response Valid:** This signal is asserted by the ADC Core when the write operation is complete and the Write Response is valid. It is held until **s_axi_csr_bready** is asserted by the user logic.

Table 3-1: Control/Status Register (CSR) Interface Port Descriptions (Continued)

Port	Direction	Width	Description
s_axi_csr_araddr	Input	7	**Read Address:** Address used for read operations. It must be valid when **s_axi_csr_arvalid** is asserted and must be held until **s_axi_csr_arready** is asserted by the ADC Core.
s_axi_csr_arprot	Input	3	**Protection:** These bits are ignored by the ADC Core.
s_axi_csr_arvalid	Input	1	**Read Address Valid:** This input must be asserted to indicate that a valid read address is available on the **s_axi_csr_araddr**. The ADC Core asserts **s_axi_csr_arready** when it ready to accept the Read Address. This input must remain asserted until the rising clock edge after the assertion of **s_axi_csr_arready**.
s_axi_csr_arready	Output	1	**Read Address Ready:** This output is asserted by the ADC Core when it is ready to accept the read address. The address is latched when **s_axi_csr_arvalid** and **s_axi_csr_arready** are high on the same cycle.
s_axi_csr_rdata	Output	32	**Read Data:** This value is the data read from the address specified by the **s_axi_csr_araddr** when **s_axi_csr_arvalid** and **s_axi_csr_arready** are High on the same cycle.
s_axi_csr_rresp	Output	2	**Read Response:** The ADC Core indicates success or failure of a read transaction through this signal, which is valid when **s_axi_csr_rvalid** is asserted; 00 = Success of normal access 01 = Success of exclusive access 10 = Slave Error 11 = Decode Error Note: For more details about this signal refer to the *AMBA AXI Specification*.
s_axi_csr_rvalid	Output	1	**Read Data Valid:** This signal is asserted by the ADC Core when the read is complete and the read data is available on the **s_axi_csr_rdata**. It is held until **s_axi_csr_rready** is asserted by the user logic.
s_axi_csr_rready	Input		**Read Data Ready:** This signal is asserted by the user logic when it is ready to accept the Read Data.
irq	Output		**Interrupt:** This is an active High, edge-type interrupt request output.

3.2 AXI4-Stream Core Interfaces

The ADS42LB69 ADC Interface Core has the following AXI4-Stream Interface, which is used to transfer data streams.

3.2.1 Stream Data (DATAIO_PD) Interface

This interface is used to transfer ADC data streams from Channel A and Channel B of the ADC Core through the output ports. Table 3-2 defines the ports in the Stream Data Interface. This interface is an AXI4-Stream Master Interface that is used to output the ADC data. This AXI4-Stream bus is synchronous with the Sample Clock (**sample_clk**) input of the core. See the *AMBA AXI4-Stream Specification* for more details on the operation of the AXI4-Stream Interface.

Table 3-2: Stream Data Interface Port Descriptions

Port	Direction	Width	Description
m_axis_cha_pd_tdata	Output	16	**Channel A Output Data**
m_axis_cha_pd_tvalid		1	**Channel A Output Data Valid:** Asserted when data is valid on the **m_axis_cha_pd_tdata**. Since ADC data is available every clock cycle, **m_axis_cha_pd_tvalid** remains High always.
m_axis_chb_pd_tdata		16	**Channel B Output Data**
m_axis_chb_pd_tvalid		1	**Channel B Output Data Valid:** Asserted when data is valid on the **m_axis_chb_pd_tdata**. Since ADC data is available every clock cycle, **m_axis_chb_pd_tvalid** remains High always.

3.3 I/O Signals

The I/O port/signal descriptions of the top level module of the ADS42LB69 ADC Interface Core are provided in Table 3-3..

Table 3-3: I/O Signals

Port/Signal Name	Type	Direction	Description
ADS42LB69 ADC Input			
da0_p	std_logic	Input	**LVDS Digital Data Input from the Channel A of the ADS42LB69 ADC**
da0_n			
da2_p			
da2_n			
da4_p			
da4_n			
da6_p			
da6_n			
da8_p			
da8_n			
da10_p			
da10_n			
da12_p			
da12_n			
da14_p			
da14_n			

Table 3-3: I/O Signals (Continued)			
Port/Signal Name	**Type**	**Direction**	**Description**
db0_p	std_logic	Input	**LVDS Digital Data Input from the Channel B of the ADS42LB69 ADC**
db0_n			
db2_p			
db2_n			
db4_p			
db4_n			
db6_p			
db6_n			
db8_p			
db8_n			
db10_p			
db10_n			
db12_p			
db12_n			
db14_p			
db14_n			
sample_clk	std_logic	Input	**Sample Clock Input:** This is the sample clock input from the ADS42LB69 ADC.
Control Outputs			
a_power_down	std_logic	Output	**Channel A Power Down:** Controls the power down mode of Channel A of the ADS42LB69 ADC. Active High. This signal is to be connected externally to the ADS42LB69 ADC.
b_power_down			**Channel B Power Down:** Controls the power down mode of Channel B of the ADS42LB69 ADC. Active High. This signal is to be connected externally to the ADS42LB69 ADC.
ovld_led_n			**Overload LED Drive:** This indicates an overload of the source of the LED selected by the user at reset. Active Low. The LED source is defined by the generic parameter **initial_led_select**.

Table 3-3: I/O Signals (Continued)

Port/Signal Name	Type	Direction	Description
Control Outputs (continued)			
ext_led_src	std_logic	Input	**External LED Source:** This is the external source, from the user design, to the Overload LED Drive.
adc_reset		Output	**ADC Reset:** Active High. This is a hardware reset output to the ADS42LB69 ADC which is used to reset the device after power-up. This signal must be connected externally to the ADC. For more details about the ADC reset refer to the ADS42LB69 ADC Datasheet.
Status Outputs			
adc_ovld_a	std_logic	Output	**ADC Channel A Data Overload:** Active High. This status bit indicates an overload of the incoming data from Channel A of the ADC.
adc_ovld_b			**ADC Channel B Data Overload:** Active High. This status bit indicates an overload of the incoming data from Channel B of the ADC.
go_trim_ovld_a			**Channel A Gain/Offset Trim Overload:** Active High. This bit indicates an overload of the Channel A data from the gain/offset adjustment module.
go_trim_ovld_b			**Channel B Gain/Offset Trim Overload:** Active High. This bit indicates an overload of the Channel B data from the gain/offset adjustment module.
ADC Serial Interface			
spi_miso	std_logic	Input	**Serial Data Bus from the ADC:** This is the input data from the ADC for a register read operation.
spi_sclk		Output	**ADC Serial Interface Clock:** This is the clock signal for the serial interface of the ADC. This signal registers the serial data into the ADC on the rising edge and sources the serial data from the ADC on the falling edge.
spi_mosi			**Serial Data Bus to the ADC:** This is the serial data bus to the ADC. Each register access of the ADC requires a 16-bit pattern on this output consisting of the address field, and data field. For additional information refer to the ADS42LB69 ADC Datasheet.
spi_sen			**Serial Data Enable to the ADC:** Active Low. This bit enables the data output on the Serial Data Bus to the ADC.

This page is intentionally blank

Chapter 4: Register Space

This chapter provides the memory map and register descriptions for the register space of the ADS42LB69 ADC Interface Core. The memory map is provided in Table 4-1.

Table 4-1: Register Space Memory Map

Register Name	Address (Base Address +)	Access	Description
Control Register 1	0x00	R/W	Controls the input tap delay.
Control Register 2	0x04		Controls the integer clock delay, power down, ADC reset, LED source select, and overload threshold count.
Channel A Gain/Offset Trim Control Register	0x08		This register holds the gain and offset trim values for Channel A data.
Channel B Gain/Offset Trim Control Register	0x0C		This register holds the gain and offset trim values for Channel B data.
ADC Serial Control	0x10		Controls the Serial Interface of the ADC.
ADC Serial Status	0x14	R	Indicates the status of the ADC Serial Interface.
Reserved	0x18	N/A	Reserved
Reserved	0x1C		Reserved
Reserved	0x20		Reserved
Interrupt Enable Register	0x24	R/W	Interrupt enable bits
Interrupt Status Register	0x28	RO	Interrupt source status bits
Interrupt Flag Register	0x2C	R/Clr	Interrupt flag bits

4.1 Control Register 1

This register is used to control the input tap delay of the input data for both channels. This register is illustrated in Figure 4-1 and described in Table 4-2.

NOTE: The bits [8:0] are set based on the generic parameter **initial_tap_delay** defined by the user, which is used as the default value. The tap size may vary over temperature.

Figure 4-1: Control Register 1

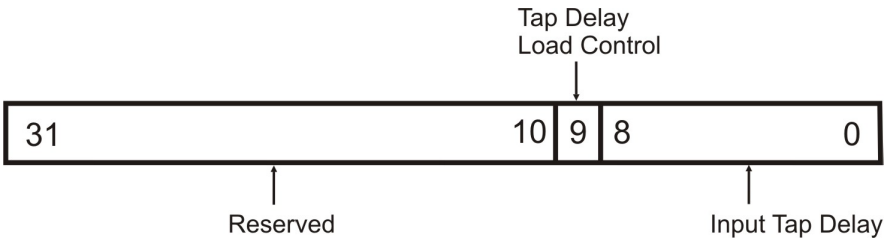


Table 4-2: Control Register 1 (Base Address + 0x00)

Bits	Field Name	Default Value	Access Type	Description
31:10	Reserved	N/A	N/A	**Reserved**
9	dly_ld_ctl	0	R/W	**Tap Delay Load Control:** This bit controls loading of the tap delay value (bits [8:0]) into the Xilinx IDELAY component of the core. The IDELAY component is used to introduce tap delay to the incoming data. Toggling this bit '1' then '0' will enable the tap delay value to be loaded into the IDELAY component.
8:0	tap_delay	-		**Input Tap Delay:** These bits are used to control the number of taps required to introduce the desired delay to the input data of the core. The Kintex Ultrascale FPGA has a 512 tap delay line with a maximum delay value of 7600 ps and minimum delay value of 1250 ps.

4.2 Control Register 2

This register controls the integer clock delay, ADC reset, Channel A power down, Channel B power down, LED source select, and overload threshold count of this core. This register is illustrated in Figure 4-2 and described in Table 4-3.

NOTE: The bits [31:16] and [6:3] are set based on the generic parameters defined by the user, which are used as default values.

Figure 4-2: Control Register 2

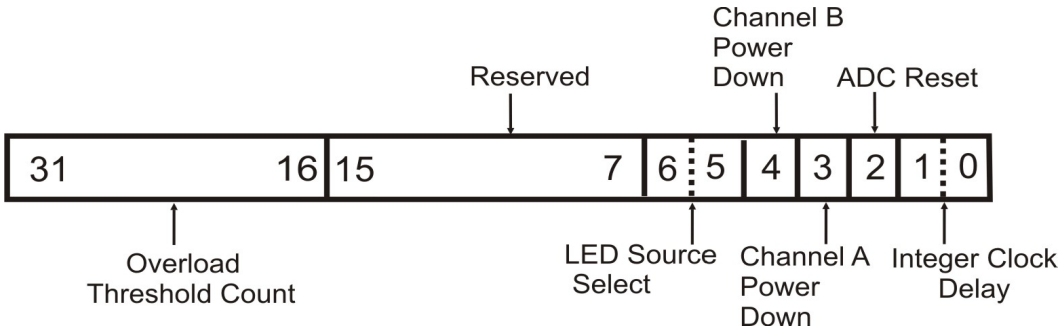


Table 4-3: Control Register 2 (Base Address + 0x04)

Bits	Field Name	Default Value	Access Type	Description
31:16	ovld_thresh_cnt	-	R/W	**Overload Threshold Count:** These bits set the overload threshold value for overload detection for data from both channels.
15:7	Reserved	N/A	N/A	**Reserved**
6:5	led_sel	-	R/W	**LED Source Select:** These bits select the source of the overload LED drive. 00 = ADC Input Overload 01 = Gain/Offset Trim Overload 10 = External Signal 11 = Disabled
4	pwdnf_b	-	R/W	**Channel B Power Down:** Controls the power down of Channel B of the ADS42LB69 ADC. For addition details about the power down modes of the ADS42LB69, refer to the *Texas Instruments ADS42LB69 Analog to Digital Converter Datasheet*. 0 = ADC Channel B default operation 1 = ADC Channel B power down

Table 4-3: Control Register 2 (Base Address + 0x04) (Continued)				
Bits	**Field Name**	**Default Value**	**Access Type**	**Description**
3	pwdnf_a	-	R/W	**Channel A Power Down:** Controls the power down of Channel A of the ADS42LB69 ADC. For addition details about the power down modes of the ADS42LB69, refer to the *Texas Instruments ADS42LB69 Analog to Digital Converter Datasheet*. 0 = ADC Channel A default operation 1 = ADC Channel A power down
2	adc_reset	0	R/W	**ADC Reset:** This bit controls the hardware reset output to the ADS42LB69 ADC which is used to reset the device after power-up. 0 = Run 1 = Reset
1:0	int_delay	00	R/W	**Integer Clock Delay:** These bits are used to introduce an additional variable delay to the input data from both channels with respect to the sample clock frequency.

4.3 Channel A Gain/Offset Trim Control Register

This register holds the values of the gain and offset trim for Channel A data. This register is illustrated in Figure 4-3 and described in Table 4-4.

NOTE: The bits [31:0] are set based on the generic parameters defined by the user, which are used as default values.

Figure 4-3: Channel A Gain/Offset Trim Control Register

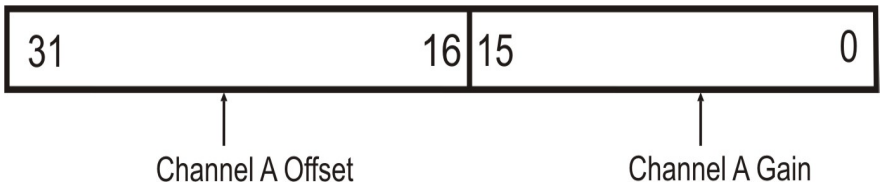


Table 4-4: Channel A Gain/Offset Trim Control Register (Base Address + 0x08)

Bits	Field Name	Default Value	Access Type	Description
31:16	offset_a	-	R/W	**Channel A Offset:** DC offset trim for Channel A data from the ADC. It is a 16-bit two's complement value.
15:0	gain_a	-		**Channel A Gain:** Unsigned binary gain trim value for the Channel A ADC data input. 0x0000 - zero gain 0x8000 - unity gain 0x8FFF - almost gain of 2

4.4 Channel B Gain/Offset Trim Control Register

This register holds the values of the gain and offset trim for Channel B data. This register is illustrated in Figure 4-4 and described in Table 4-5.

NOTE: The bits [31:0] are set based on the generic parameters defined by the user, which are used as default values.

Figure 4-4: Channel B Gain/Offset Trim Control Register

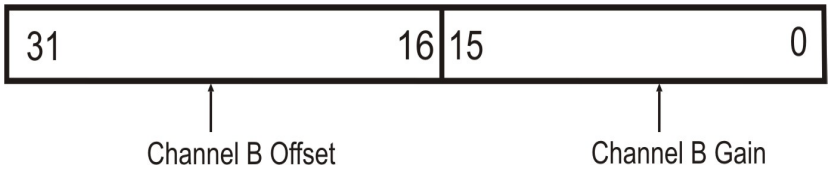


Table 4-5: Channel B Gain/Offset Trim Control Register (Base Address + 0x0C)

Bits	Field Name	Default Value	Access Type	Description
31:16	offset_b	-	R/W	**Channel B Offset:** DC offset trim for Channel B data from the ADC. It is a 16-bit two's complement value.
15:0	gain_b	-		**Channel B Gain:** Unsigned binary gain trim value for the Channel B ADC data input. 0x0000 - zero gain 0x8000 - unity gain 0x8FFF - almost gain of 2

4.5 ADC Serial Control Register

This register controls the Serial Interface of the ADS42LB69 ADC. This register is illustrated in Figure 4-5 and described in Table 4-6.

Figure 4-5: ADC Serial Control Register

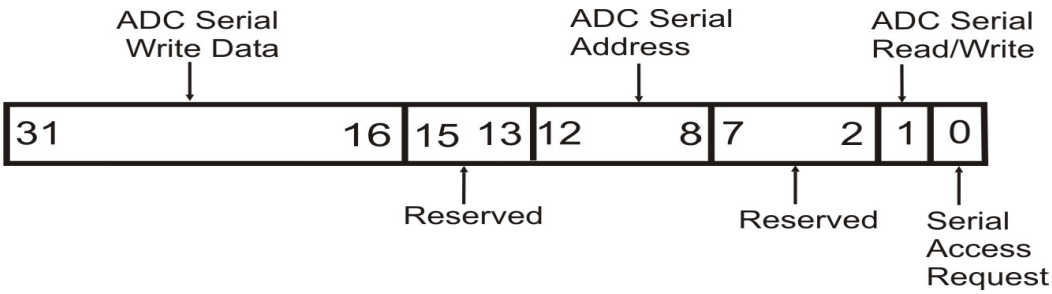


Table 4-6: ADC Serial Control Register (Base Address + 0x10)

Bits	Field Name	Default Value	Access Type	Description
31:24	Reserved	N/A	N/A	**Reserved**
23:16	ser_wr_data	0x00	R/W	**ADC Serial Write Data:** These bits hold the data to be written to the register accessed across the SPI of the ADS42LB69 ADC.
15:14	Reserved	N/A	N/A	**Reserved**
13:8	ser_addr	000000	R/W	**ADC Serial Address:** These bits control the address of the register within the ADS42LB69 ADC to be accessed through the Serial Interface.
7:2	Reserved	N/A	N/A	**Reserved**
1	ser_rd_wr	0	R/W	**ADC Serial Read/ Write:** This bit controls the type of operation (read/write) to be performed across the Serial Interface of the ADC. 0 = Write 1 = Read
0	ser_access			**Serial Access Request:** This bit must be toggled '1' then '0' to request access to the Serial Interface of the ADC for a read/write operation.

4.6 ADC Serial Status Register

This register indicates the status of the Serial Interface of the ADS42LB69 ADC. This register is illustrated in Figure 4-6 and described in Table 4-7.

Figure 4-6: ADC Serial Status Register

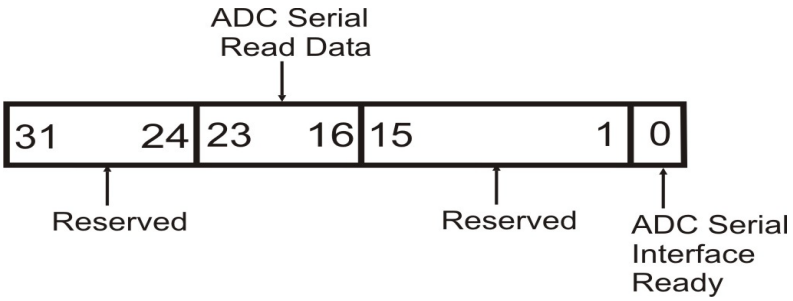


Table 4-7: ADC Serial Status Register (Base Address + 0x14)

Bits	Field Name	Default Value	Access Type	Description
31:24	Reserved	N/A	N/A	**Reserved**
23:16	ser_rd_data	0x00	R	**ADC Serial Read Data:** These bits hold the data input to the core from the ADC Serial Interface when a read request to a register within the ADC was made.
15:1	Reserved	N/A	N/A	**Reserved**
0	ser_sm_rdy	0	R	**ADC Serial Interface Ready:** This bit indicates whether the Serial Interface of the ADS42LB69 ADC is ready for a new transaction or busy with an existing read/write operation. 0 = Serial Interface Busy 1 = Serial Interface ready for a new R/W operation

4.7 Interrupt Enable Register

The bits in the interrupt enable register are used to enable (or disable) the generation of interrupts based on the condition of certain circuit elements, known as interrupt sources. When a bit in this register associated with a given interrupt source is High, an interrupt will be generated by the rising edge of that source's Interrupt Status Register bit (see Section 4.8). This register is illustrated in Figure 4-7 and the bits are described in Table 4-8.

Figure 4-7: Interrupt Enable Registers

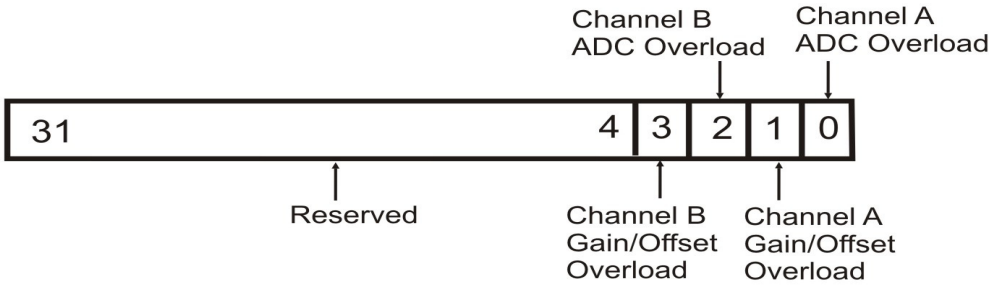


Table 4-8: Interrupt Enable Register (Base Address + 0x24)

Bits	Field Name	Default Value	Access Type	Description
31:4	Reserved	N/A	N/A	**Reserved**
3	go_ovld_b	0	R/W	**Channel B Gain/Offset Data Overload:** This bit enables/ disables the overload interrupt source of Channel B ADC data after a gain/ offset adjustment. 0 = Disable interrupt 1 = Enable interrupt
2	adc_ovld_b			**Channel B ADC Data Overload:** This bit enables/ disables the Channel B ADC data overload interrupt source. 0 = Disable interrupt 1 = Enable interrupt
1	go_ovld_a			**Channel A Gain/Offset Data Overload:** This bit enables/ disables the overload interrupt source of Channel A ADC data after a gain/ offset adjustment. 0 = Disable interrupt 1 = Enable interrupt
0	adc_ovld_a			**Channel A ADC Data Overload:** This bit enables/ disables the Channel A ADC data overload interrupt source. 0 = Disable interrupt 1 = Enable interrupt

4.8 Interrupt Status Register

The Interrupt Status Register has read-only access associated with each interrupt condition. A status bit changes to '1' when the source interrupt occurs. When a status bit in this register changes to '1' the corresponding flag bit in the Interrupt Flag Register is set to '1'. A status bit in this register clears to '0' when that interrupt condition clears, whereas the associated flag bit in the Interrupt Flag Register remains at logic '1' until it is explicitly cleared by the user.

Some of the interrupt sources are transient and so may not appear in the Interrupt Status Register at the time it is read. In such cases use the Interrupt Flag Register to see the interrupt conditions that have occurred. The Interrupt Status Register is illustrated in Figure 4-8 and the bits are described in Table 4-9.

Figure 4-8: Interrupt Status Registers

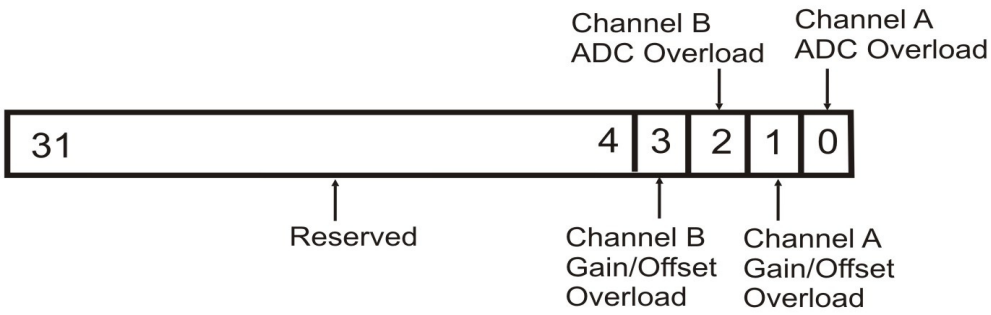


Table 4-9: Interrupt Status Register (Base Address + 0x28)

Bits	Field Name	Default Value	Access Type	Description
31:4	Reserved	N/A	N/A	**Reserved**
3	go_ovld_b	0	R	**Channel B Gain/Offset Data Overload:** This bit indicates the status of the overload interrupt source of Channel B ADC data after a gain/ offset adjustment. 0 = No interrupt 1 = Interrupt condition asserted
2	adc_ovld_b			**Channel B ADC Data Overload:** This bit indicates the status of the Channel B ADC data overload interrupt source. 0 = No interrupt 1 = Interrupt condition asserted

Table 4-9: Interrupt Status Register (Base Address + 0x28) (Continued)

Bits	Field Name	Default Value	Access Type	Description
1	go_ovld_a	0	R	**Channel A Gain/Offset Data Overload:** This bit indicates the status of the overload interrupt source of Channel A ADC data after a gain/ offset adjustment. 0 = No interrupt 1 = Interrupt condition asserted
0	adc_ovld_a			**Channel A ADC Data Overload:** This bit indicates the status of the Channel A ADC data overload interrupt source. 0 = No interrupt 1 = Interrupt condition asserted

4.9 Interrupt Flag Register

The Interrupt Flag Register has read/clear access associated with each interrupt condition. When reset, this register has all bits set to '0' (cleared). Each flag bit in this register latches an interrupt occurrence. A '1' in any flag bit in this register indicates that an interrupt has occurred.

Note that when any status bit in the Interrupt Status Register, changes to '1' the corresponding flag bit in this register will also be set to '1'. However, when a status bit in the Interrupt Status Register clears from '1' to '0', the corresponding latched flag bit in this register does not clear, but remains at '1'. To clear the flag bits, write '1's to the desired bits. The flags are not affected by the Interrupt Enable Register. The Interrupt Flag Register is illustrated in Figure 4-8 and the bits are described in Table 4-9.

Figure 4-9: Interrupt Flag Registers

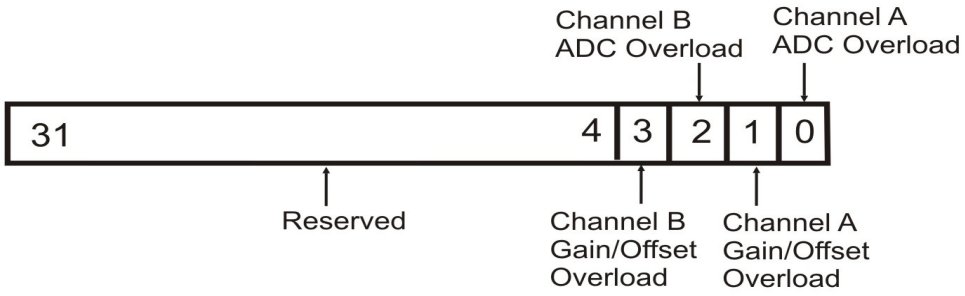


Table 4-10: Interrupt Flag Register (Base Address + 0x2C)

Bits	Field Name	Default Value	Access Type	Description
31:2	Reserved	N/A	N/A	**Reserved**
3	go_ovld_b	0	R/Clr	**Channel B Gain/Offset Data Overload:** This bit indicates the Channel B gain/ offset ADC overload interrupt flag. **Read:** 0 = No interrupt 1 = Interrupt latched **Clear:** 1 = Clear latch
2	adc_ovld_b			**Channel B ADC Data Overload:** This bit indicates the Channel B ADC data overload interrupt flag. **Read:** 0 = No interrupt 1 = Interrupt latched **Clear:** 1 = Clear latch

Table 4-10: Interrupt Flag Register (Base Address + 0x2C) (Continued)

Bits	Field Name	Default Value	Access Type	Description
1	go_ovld_a	0	R/Clr	**Channel A Gain/Offset Data Overload:** This bit indicates the Channel A gain/ offset ADC overload interrupt flag. **Read:** 0 = No interrupt 1 = Interrupt latched **Clear:** 1 = Clear latch
0	adc_ovld_a			**Channel A ADC Data Overload:** This bit indicates the Channel A ADC data overload interrupt flag. **Read:** 0 = No interrupt 1 = Interrupt latched **Clear:** 1 = Clear latch

This page is intentionally blank

Chapter 5: Designing with the Core

This chapter includes guidelines and additional information to facilitate designing with the ADS42LB69 ADC Interface Core.

5.1 General Design Guidelines

The ADS42LB69 ADC Interface Core is used as an interface to the Texas Instruments Dual Channel ADS42LB69 Analog to Digital Converter. It provides gain/offset adjustment and overload detection of the ADS42LB69 ADC output data from both channels. This core also provides a Serial Interface to access the registers within the ADC.

5.2 Clocking

AXI4-Lite Clock: **s_axi_csr_aclk**

The **s_axi_csr_aclk** is used to clock the AXI4-Lite Control/Status Register (**s_axi_csr**) interface of the core.

Sample Clock: **sample_clk**

This clock is used to clock the ADS42LB69 inputs, and the AXI4-Stream outputs of the core.

5.3 Resets

Main reset: **s_axi_csr_aresetn**

This is an active low synchronous reset associated with the **s_axi_csr_aclk**. When asserted, all state machines in the core are reset, all FIFOs are flushed and all the control registers are cleared back to their initial default states.

5.4 Interrupts

This core has an edge-type (rising edge-triggered) interrupt output. It is synchronous with the **s_axi_csr_aclk**. On the rising edge of any interrupt signal, a one clock cycle wide pulse is output from the core on it's **irq** output. Each interrupt event is stored in two registers accessible on the **s_axi_csr** bus. The Interrupt Status Register always reflects the current state of the interrupt condition, which may have changed since the generation of the interrupt. The Interrupt Flag Register latches the occurrence of each interrupt, in a bit that retains its state until explicitly cleared.

The Interrupt flags can be cleared by writing '1' to the associated bit's location. All interrupt sources that are enabled (via the Interrupt Enable Register) are "OR ed" onto the **irq** output.

5.4 Interrupts (continued)

Note: All interrupt sources are latched in the interrupt flag register, even when an interrupt source is not enabled to create an interrupt.

Note: Because this core uses edge-triggered interrupts, the fact that an interrupt condition may remain active after servicing will not cause the generation of a new interrupt. A new interrupt will only be generated by another rising edge on an interrupt source.

5.5 Interface Operation

Control/Status Register Interface: This is the control/status register Interface. It is associated with the **s_axi_csr_aclk**. It is a standard AXI4-Lite type interface. See Chapter 5 for the control register memory map, for more details on the registers that can be accessed through this interface.

Stream Data (DATAIO_PD) Interface: This interface is used to transfer output ADC data streams from both the channels. It is a standard AXI4-Stream Master Interface. For more details about this interface refer to Section 3.2.1.

5.6 Programming Sequence

This section briefly describes the programming sequence of registers to initiate and complete a transaction on the ADS42LB69 ADC Interface Core. The programming sequence for this core is as follows:

1) Ensure that the Interrupt Flag Register is cleared.

2) The input data passes through the tap delay logic and the gain/offset adjustment modules for the input delay, gain/offset adjustment defined by the control register bits.

3) The data passes through the overload detection block and indicates any overload through the Overload LED Drive.

4) The data is transferred across the output ports.

5.7 Timing Diagrams

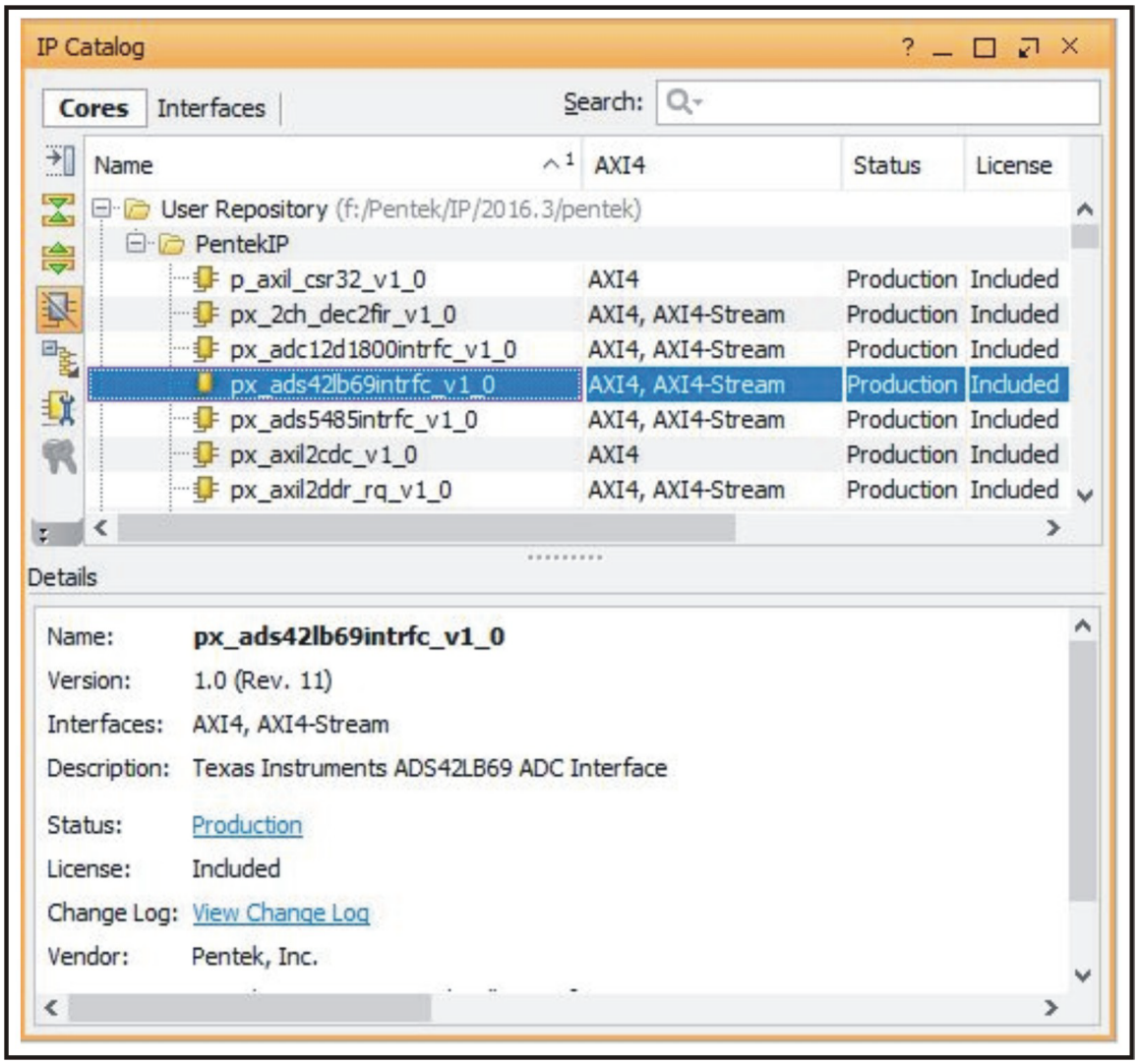
This section is not applicable to this IP core.

Chapter 6: Design Flow Steps

6.1 Pentek IP Catalog

This chapter describes customization and generation of the Pentek ADS42LB69 ADC Interface Core. It also includes simulation, synthesis and implementation steps that are specific to this core. This IP core can be generated from the Vivado IP Catalog when the Pentek IP Repository has been installed. It will appear in the IP Catalog list as **px_ads42lb69intrfc_v1_0** as shown in Figure 6-1.

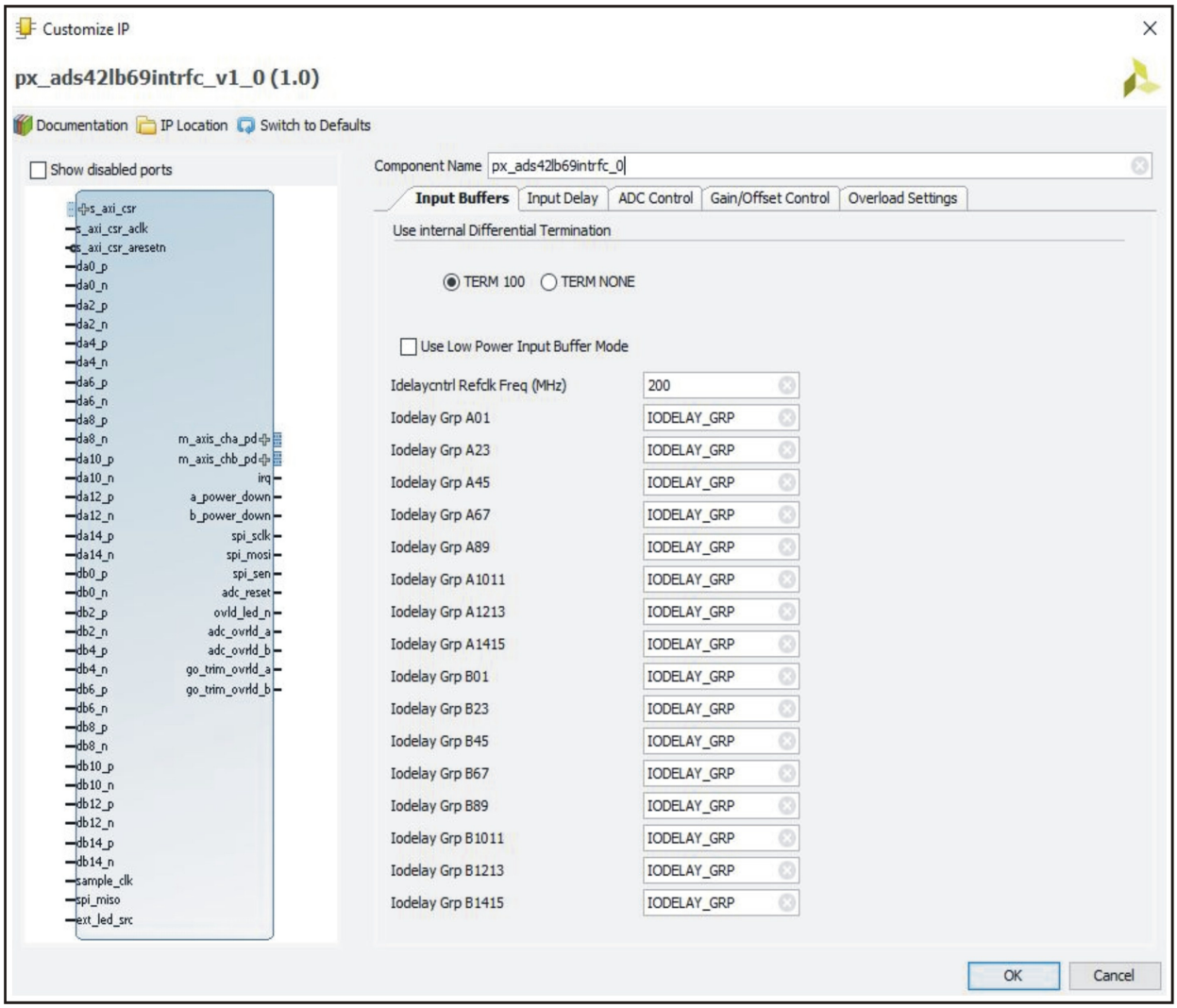
Figure 6-1: ADS42LB69 ADC Interface Core in Pentek IP Catalog



6.1 Pentek IP Catalog (continued)

When you select the **px_ads42lb69intrfc_v1_0** core, a screen appears that shows the core's symbol and the core's parameters (see Figure 6-2). The core's symbol is the box on the left side.

Figure 6-2: ADS42LB69 ADC Interface Core IP Symbol



6.2 User Parameters

The user parameters of this IP core are described in Section 2.5 of this user manual.

6.3 Generating Output

For more details about generating and using IP in the Vivado Design Suite, refer to the *Vivado Design Suite User Guide - Designing with IP*.

6.4 Constraining the Core

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

Required Constraints

The XDC constraints are not provided with this core. The necessary constraints can be applied in the top level module of the user design.

Device, Package, and Speed Grade Selections

This IP works for the Kintex Ultrascale FPGAs.

Clock Frequencies

The maximum Sample clock (**sample_clk**) frequency and AXI4-Lite Interface clock (**s_axi_csr_aclk**) frequency for this IP core is 250 MHz.

Clock Management

This section is not applicable for this IP core.

Clock Placement

This section is not applicable for this IP core.

Banking and Placement

This section is not applicable for this IP core.

Transceiver Placement

This section is not applicable for this IP core.

I/O Standard and Placement

This section is not applicable for this IP core.

6.5 Simulation

The test bench and the simulation results for this IP core will be available in the next revision of this manual.

6.6 Synthesis and Implementation

For details about synthesis and implementation see the *Vivado Design Suite User Guide - Designing with IP*.

This page is intentionally blank