IP CORE MANUAL

Timestamp Generator IP

px_timestamp_gen

PENTEK

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

Manual Part Number: 807.48329 Rev: 1.0 - December 09, 2016

Manual Revision History

Date	Version	Comments
12/09/16	1.0	Initial Release

Legal Notices

Copyright

Trademarks

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™ Timestamp Generator Core generates AXI4 Timestamp Streams from input Timing event streams which contain timing signals (Gate, Sync, PPS).

This core complies with the ARM® AMBA® AXI4 Specification and also provides a control/status register interface. This user manual defines the hardware interface, software interface, and parameterization options for the Timestamp Generator Core.

Features

- Register access through AXI4-Lite interface
- Software programmable input timing event data stream width
- Provides register access to control the PPS mode and source of PPS signal
- Supports a 32-bit PPS counter and 64-bit Sample clock counter
- Includes a state machine to load and reset the counters

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	VHDL
Constraints File	Not Provided[b]
Simulation Model	VHDL
Supported S/W Driver	HAL Software Support
Tested Design Flows	
Design Entry	Vivado® Design Suite 2016.3 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.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

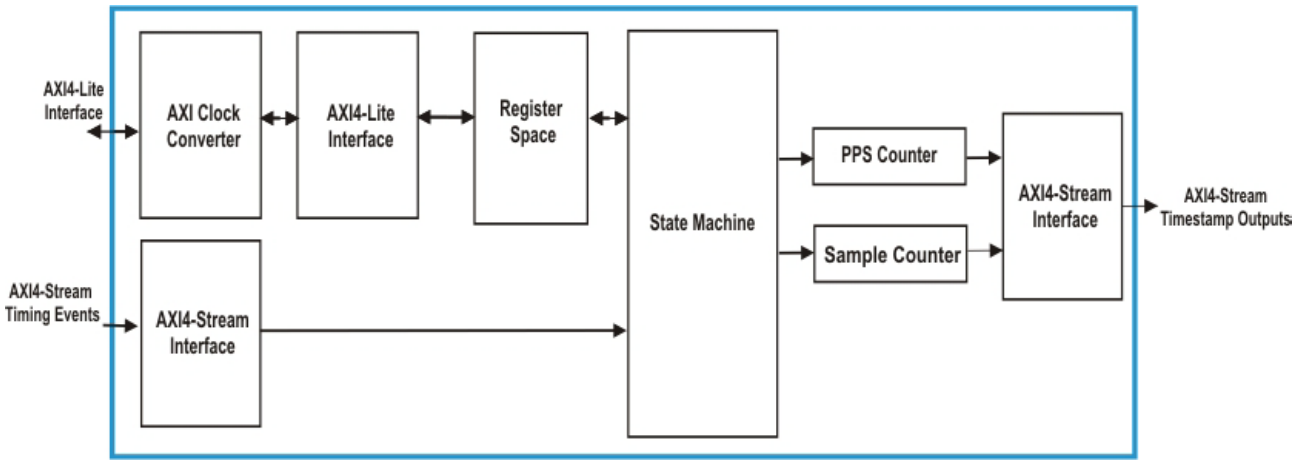
The Timestamp Generator Core accepts input Timing Event Streams (Sample Clock, Reset and Timing Signals) through an AXI4-Stream Slave Interface and delivers output Timestamp Streams, which contain timestamp and a time-aligned copy of the timing events that created them, through an AXI4-Stream Master Interface. Key components of the Timestamp Generator Core are:

- **AXI Clock Converter Core** - This is connected to the AXI4-Lite Interface, as shown in Figure 1-1, in order to operate the Register Space in the sample clock domain.
- **PPS Counter** - The 32-bit PPS counter is incremented by edge events of the PPS source selected.
- **Sample Clock Counter** - The 64-bit sample clock counter counts the number of sample clock cycles that have occurred since the last PPS event. When PPS mode is disabled, the sample counter is a free-running counter of the sample clock cycles.
- **Counter Load Control State Machine** - The state machine is used to load values to the counters and also reset the counters.

The output timestamp data stream is generated based on the PPS mode. For more details on the Timestamp output from the AXI4-Stream Master Interface, refer to Section 3.2.

Figure 1-1 is a top-level block diagram of the Pentek Timestamp Generator Core. The modules within the block diagram are explained in the later sections of this manual.

Figure 1-1: Timestamp Generator Core Block Diagram



1.1 Functional Description (continued)

- **AXI Clock Converter Core:** The AXI Clock Converter Core is included in the Xilinx AXI Interconnect Core and is used to connect one AXI memory-mapped slave to an AXI memory-mapped master which is operating in a different clock domain. In the Timestamp Generator Core, the AXI Clock Converter is used to operate the register space in the sample clock domain.

- **AXI4-Stream Interface:** The Timestamp Generator Core has two AXI4-Stream Interfaces. At the input, an AXI4-Stream Slave Interface is used to receive input timing event data streams and at the output an AXI4-Stream Master Interface is used to transfer test signal data streams through the output ports. For more details about the AXI4-Stream Interfaces please refer to **Section 3.2 AXI4-Stream Core Interfaces**.

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

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

- **Counter Load Control State Machine:** This state machine is used to control the load on the counters and has three states:

  - **Reset** - The Reset state resets the state machine based on the input reset signal (**s_axis_aresetn**) from the input AXI4-Stream Slave Interface.

  - **Wait for Arm** - When the state machine is in the Wait for Arm state, the Timestamp Generator Core waits for the arm load signal, from the Mode Control Register (see Section 4.1), to go High.

  - **Armed** - Once in the Armed State, the core waits for an edge event on the PPS signal and loads the counters with the required values when the event occurs. The state machine then goes back to the **Wait for Arm** state while the counters generate output timestamp data of all the timing events that have occurred since the last armed state. To reset the counters to start a new count, the **arm load bit** must be set to 1.

- **Counters:** These are the PPS and Sample clock counters used in the Timestamp Generator Core to generate timestamp outputs.

1.2 Applications

The Timestamp Generator Core can be used for generating AXI4 Timestamp Streams and can be incorporated into any Kintex Ultrascale FPGAs.

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

This page is intentionally blank

Chapter 2: General Product Specifications

2.1 Standards

The Timestamp Generator 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 Timestamp Generator Core is limited only by the FPGA logic speed. The values presented in this section should be used as an estimation guideline. Actual performance can vary.

2.2.1 Maximum Frequencies

The Timestamp Generator core has two incoming clock signals. The Sample clock has a maximum frequency of 345MHz while the clock across the AXI4-Lite interface has a maximum frequency of 250MHz on a Kintex Ultrascale -2 speed grade FPGA. 250 MHz is typically the PCI Express (PCIe®) AXI bus clock frequency.

2.3 Resource Utilization

The resource utilization of the Timestamp Generator 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	548
Flip-Flops	1505
Memory LUTs	71
DSP	3

NOTE: Actual utilization may vary based on the user design in which the Timestamp Generator 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 Timestamp Generator 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
pts_signal_width	Integer	**Timing Signal Width:** This parameter indicates the width of the input timing event data stream at the AXI4-Stream Slave Interface. It can take the values 1, 2, 4, or 8.

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

3.1 AXI4-Lite Core Interfaces

The Timestamp Generator 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 Timestamp Generator 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 signal will reset all control registers to their initial states.
s_axi_csr_awaddr	Input	5	**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 Timestamp Generator Core.
s_axi_csr_awprot	Input	3	**Protection:** The Timestamp Generator 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 Timestamp Generator 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 Timestamp Generator 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 the **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	1	**Write Ready:** This signal is asserted by the Timestamp Generator 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 Timestamp Generator 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 Timestamp Generator 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	5	**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 Timestamp Generator Core.
s_axi_csr_arprot	Input	3	**Protection:** These bits are ignored by the Timestamp Generator 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 Timestamp Generator 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 Timestamp Generator 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 Timestamp Generator 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 Timestamp Generator Core when the read is complete and the read data is available on **s_axi_csr_rdata**. It is held until **s_axi_csr_ rready** is asserted by the user logic.
s_axi_csr_rready	Input	1	**Read Data Ready:** This signal is asserted by the user logic when it is ready to accept the Read Data.
irq	Output	1	**Interrupt:** This is an active high, edge-type interrupt output.

3.2 AXI4-Stream Core Interfaces

The Timestamp Generator Core has the following AXI4-Stream Interfaces, used to receive and transfer data streams.

- **Timing Events (PTCTL) Interface:** The interface through which Timing Events are received.
- **Timestamp Output (PTS) Interface:** The interface through which Timestamp signals are transferred through the output ports.

3.2.1 Timing Events (PTCTL) Interface

Table 3-2 defines the ports in the Timing Events Interface. This interface is an AXI4-Stream Slave Interface. See the *AMBA AXI4-Stream Specification* for more details on the operation of the AXI4-Stream Interface.

Table 3-2: Timing Events Interface Port Descriptions

Port	Direction	Width	Description
s_axis_aclk	Input	1	**Sample Clock**
s_axis_aresetn	Input		**Reset:** Active Low.
s_axis_ptctl_tdata	Input	depends on the generic parameter **pts_signal_width**	**Input Data:** This is timing event data which indicates the Gate/ Sync/ PPS signal positions. tdata[n-1:0] - Gate Positions tdata[2n-1:n] - Sync Positions tdata[3n-1:2n] - PPS Positions
s_axis_ptctl_tvalid	Input	1	**Input Data Valid:** Asserted when data is valid on **s_axis_ptctl_tdata**.

3.2 AXI4-Stream Core Interfaces (continued)

3.2.2 Timestamp Output (PTS) Interface

Table 3-3 defines the ports in the Timestamp Output Interface. This interface is an AXI4-Stream Master Interface. See the *AMBA AXI4-Stream Specification* for more details on the operation of the AXI4-Stream Interface.

Table 3-3: Timestamp Output Interface Port Descriptions

Port	Direction	Width	Description
m_axis_pts_tdata	Output	64	**Output Data:** This is the timestamp output of the timestamp generator core. When PPS mode is enabled (Set to 1): tdata[31:0] - Sample Clock Count tdata[63:32] - PPS count When PPS mode is disabled (Set to 0): tdata[63:0] - Sample Clock Count
m_axis_pts_tuser		24	**Sideband Data:** These bits indicate the Gate/Sync/PPS positions that created the timestamps. tuser[7:0] - Gate Positions tuser[15:8] - Sync Positions tuser[23:16] - PPS Positions
m_axis_pts_tvalid		1	**Output Data Valid:** Asserted when data is valid on **m_axis_pts_tdata**.

This page is intentionally blank

Chapter 4: Register Space

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

Table 4-1: Register Space Memory Map

Register Name	Address (Base Address +)	Access	Description
Mode Control	0x00	R/W	Controls mode of operation of the Timestamp Generator Core.
Sample Counter Initial Value	0x04		Controls the initial value loaded into the sample counter.
PPS Counter Initial Value	0x08		Controls the initial PPS value loaded into the sample counter.
Sample Counter Value Read-back	0x0C	R	Indicates a sample counter value that is read-back.
PPS Counter value Read-back	0x10		Indicates a PPS counter value that is read-back.
Interrupt Enable Register	0x14	R/W	Interrupt enable bits
Interrupt Status Register	0x18	R	Interrupt source status bits
Interrupt Flag Register	0x1C	R/Clr	Interrupt flag bits

4.1 Mode Control Register

This control register is used to control the Counter reset, ARM load and Clear, PPS mode, PPS active edge, PPS count enable, Latch Read-back, Counter Load Mode, and Stay Armed bits of the Timestamp Generator Core. The Mode Control Register is illustrated in Figure 4-1 and described in Table 4-2.

Figure 4-1: Mode Control Register

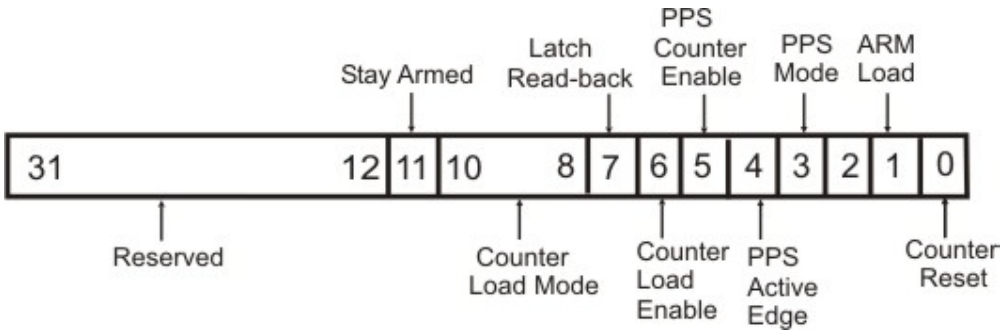


Table 4-2: Mode Control Register (Base Address + 0x00)

Bits	Field Name	Default Value	Access Type	Description
31:12	Reserved	N/A	N/A	**Reserved**
11	stay_armed	0	R/W	**Stay Armed:** When set to '1', this signal keeps the Counter Load Control State Machine in the Armed state with the **load_armed** status signal High. 0 = State machine goes to Wait for Arm state 1 = State machine remains in Armed state
10:8	count_load_mode	000	R/W	**Counter Load Mode:** These bits indicate the counter load mode based on the source of the PPS signal. 000 - Always on ARM 001 - AUX pulse rising edge 010 - PPS rising edge 011 - PPS falling edge 100 - SYNC rising edge 101 - SYNC falling edge 110 - Gate rising edge 111 - Gate falling edge

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

Bits	Field Name	Default Value	Access Type	Description
7	latch_readback	0	R/W	**Latch Read-back:** When reading from the PPS Counter Value Read-back register and the Sample Counter Value Read-back register, in order to ensure the values read from them correspond to the same clock cycle, the latch read-back bit is used. The latch read-back bit when set to '1', holds the sample counter and PPS counter values of the same clock cycle into the corresponding registers without updating them until it is set to '0' again. When the PPS Counter Value Read-back register and Sample Counter Value Read-back register are to be read, set the **latch_readback** bit to '1' and then perform the read operation. When the read is complete, set the **latch_readback** bit back to '0'.
6	count_load_en	0	R/W	**Counter Load Enable:** This bit enables the load on the counter at the rising edge or falling edge of the selected source signal when the state machine is in the Armed state. 0 = Disable 1 = Enable
5	pps_count_enable	0	R/W	**PPS Count Enable:** Enables count on the PPS counter. 0 = Disable 1 = Enable
4	pps_active_edge	0	R/W	**PPS Active Edge:** This bit indicates the active edge of the PPS signal. 0 = Rising edge is the active PPS edge 1 = Falling edge is the active PPS edge
3	pps_mode	0	R/W	**PPS Mode:** This bit controls the PPS mode of the timestamp generator core. For more details refer to Section 5.1. 0 = PPS mode disabled 1 = PPS mode enabled
2	arm_clear	0	R/W	**ARM Clear:** This is used to clear the load on the counters. 0 = Remain unchanged 1 = Clear
1	arm_load	0	R/W	**ARM Load:** When this bit is toggled '1' then '0', it changes the state of the counter load state machine to the Armed state and enables load on the counters when arm clear bit of this register is '0'.
0	cntr_rst	0	R/W	**Counter Reset:** This bit is used to reset the counters. 0 = Run 1 = Reset

4.2 Sample Counter Initial Value Register

This register controls the initial value loaded on the sample counter. These bits are the lower 32 bits of the initial value of the 64-bit sample counter. This register is illustrated in Figure 4-2 and described in Table 4-3.

Figure 4-2: Sample Counter Initial Value Register

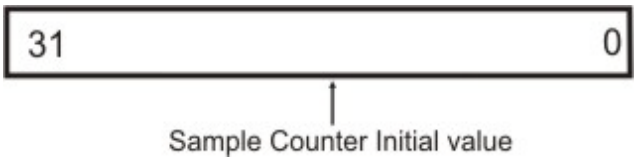


Table 4-3: Sample Counter Initial Value Register (Base Address + 0x04)

Bits	Field Name	Default Value	Access Type	Description
31:0	sample_cntr_init_val	0x00000000	R/W	**Sample Counter Initial Value:** This is the initial value on the sample counter.

4.3 PPS Counter Initial Value Register

This register controls the initial PPS counter value loaded on the sample counter. These bits are the upper 32 bits of the initial value of the 64-bit Sample counter. This register is illustrated in Figure 4-2 and described in Table 4-3.

Figure 4-3: PPS Counter Initial Value Register

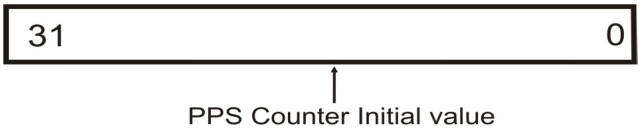


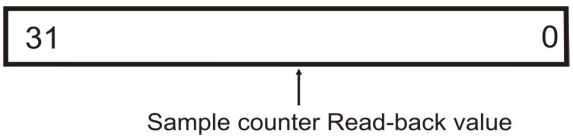
Table 4-4: PPS Counter Initial Value Register (Base Address + 0x08)

Bits	Field Name	Default Value	Access Type	Description
31:0	pps_cntr_init_val	0x00000000	R/W	**PPS Counter Initial Value:** This value is the initial PPS counter value loaded on the Sample Counter.

4.4 Sample Counter Value Read-Back Register

This is a status register which indicates the value read-back from the Sample counter. This is a read-only register. This register is illustrated in Figure 4-2 and described in Table 4-3.

Figure 4-4: Sample Counter Value Read-Back Register



.

Table 4-5: Sample Counter Value Read-Back Register (Base Address + 0x0C)

Bits	Field Name	Default Value	Access Type	Description
31:0	readback_ sample_cnt	0x00000000	R	**Sample Counter Read-back value:** These bits indicate the lower 32-bit value on the sample counter. This register must read after setting the **latch_readback** bit of the Mode Control Register to '1' in order to ensure that the Sample Counter Value Read-back register value, and the PPS Counter Value Read-back register value correspond to the same clock cycle.

4.5 PPS Counter Value Read-Back Register

This is a status register with read-only access that indicates the read-back value on the PPS counter when the PPS mode is enabled. When the PPS mode is disabled, this register indicates the value on the upper 32 bits of the Sample counter. This register is illustrated in Figure 4-2 and described in Table 4-3.

Figure 4-5: PPS Counter Value Read-Back Register

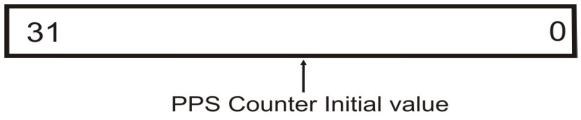


Table 4-6: PPS Counter Value Read-Back Register (Base Address + 0x10)

Bits	Field Name	Default Value	Access Type	Description
31:0	readback_pps_count	0x00000000	R	**PPS Counter Read-back Value:** These bits indicate the value on the PPS counter when the PPS mode is enabled. When the PPS mode is disabled, these bits indicate the upper 32 bits of the Sample counter. This register must be read after setting the **latch_readback** bit of the Mode Control Register to '1', in order to ensure that the PPS Counter Value Read-back register value, and the Sample Counter Value Read-back register value correspond to the same clock cycle.

4.6 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 4.7). This register is illustrated in Figure 4-6 and described in Table 4-7.

Figure 4-6: Interrupt Enable Register

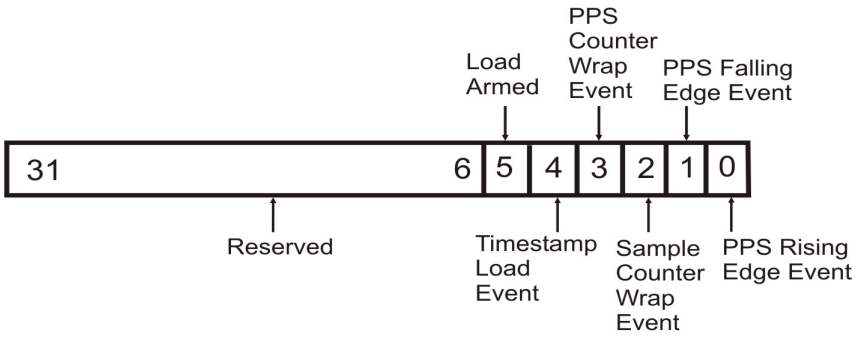


Table 4-7: Interrupt Enable Register (Base Address + 0x14)

Bits	Field Name	Default Value	Access Type	Description
31:6	Reserved	N/A	N/A	**Reserved**
5	load_armed	0	R/W	**Load Armed:** This bit enables/disables the load armed interrupt source. The load armed interrupt source indicates that the Counter Load Control State Machine is in the Armed state. 0 = Disable interrupt 1 = Enable interrupt
4	timestamp_load_ event	0	R/W	**Timestamp Load Event:** This bit enables/disables the timestamp load event interrupt source. The timestamp load event interrupt source indicates that the counters are loaded with initial values and other data to generate the timestamp output. 0 = Disable interrupt 1 = Enable interrupt

Table 4-7: Interrupt Enable Register (Base Address + 0x14) (Continued)

Bits	Field Name	Default Value	Access Type	Description
3	pps_cntr_wrap _event	0	R/W	**PPS Counter Wrap Event:** This bit enables/disables the PPS counter wrap event interrupt source. The PPS counter wrap event interrupt source indicates that the PPS counter has reached the maximum value and wrapped around to count from zero (all bits in counter set to 0). 0 = Disable interrupt 1 = Enable interrupt
2	sample_cntr_wrap _event	0	R/W	**Sample Counter Wrap Event:** This bit enables/disables the sample counter wrap event interrupt source. The sample counter wrap event interrupt source indicates that the sample counter has reached the maximum value and wrapped around to count from zero (all bits in counter set to 0). 0 = Disable interrupt 1 = Enable interrupt
1	pps_falling_edge _event	0	R/W	**PPS Signal Falling Edge Event:** This bit enables/disables the PPS signal falling edge event interrupt source. 0 = Disable interrupt 1 = Enable interrupt
0	pps_rising_edge _event	0	R/W	**PPS Signal Rising Edge Event:** This bit enables/disables the PPS signal rising edge event interrupt source. 0 = Disable interrupt 1 = Enable interrupt

4.7 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 4-7 and described in Table 4-8.

Figure 4-7: Interrupt Status Register

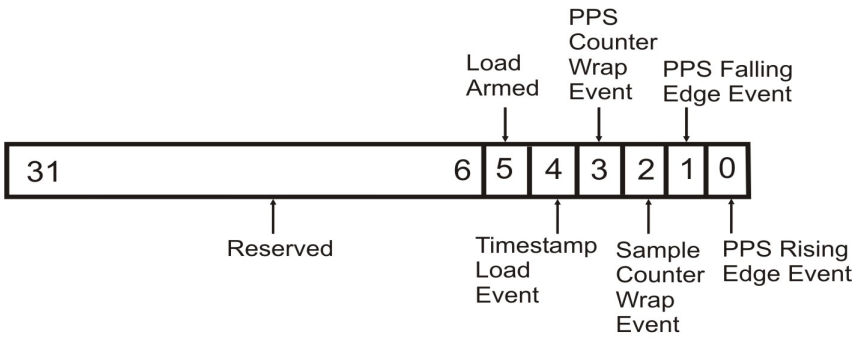


Table 4-8: Interrupt Status Register (Base Address + 0x18)

Bits	Field Name	Default Value	Access Type	Description
31:6	Reserved	N/A	N/A	**Reserved**
5	load_armed	0	R	**Load Armed:** This bit indicates the status of the load armed interrupt source. The load armed interrupt source indicates that the Counter Load Control State Machine is in the Armed state. 0 = No interrupt 1 = Interrupt condition asserted
4	timestamp_load_event	0	R	**Timestamp Load Event:** This bit indicates the status of the timestamp load event interrupt source. The timestamp load event interrupt source indicates that the counters are loaded with initial values and other data to generate the timestamp output. 0 = No interrupt 1 = Interrupt condition asserted

Table 4-8: Interrupt Status Register (Base Address + 0x18) (Continued)				
Bits	**Field Name**	**Default Value**	**Access Type**	**Description**
3	pps_cntr_wrap _event	0	R	**PPS Counter Wrap Event:** This bit indicates the status of the PPS counter wrap event interrupt source. The PPS counter wrap event interrupt source indicates that the PPS counter has reached the maximum value and wrapped around to count from zero (all bits in counter set to 0). 0 = No interrupt 1 = Interrupt condition asserted
2	sample_cntr_wrap _event	0	R	**Sample Counter Wrap Event:** This bit indicates the status of the sample counter wrap event interrupt source. The sample counter wrap event interrupt source indicates that the sample counter has reached the maximum value and wrapped around to count from zero (all bits in counter set to 0). 0 = No interrupt 1 = Interrupt condition asserted
1	pps_falling_edge _event	0	R	**PPS Signal Falling Edge Event:** This bit indicates the status of the PPS signal falling edge event interrupt source. 0 = No interrupt 1 = Interrupt condition asserted
0	pps_rising_edge _event	0	R	**PPS Signal Rising Edge Event:** This bit indicates the status of the PPS signal rising edge event interrupt source. 0 = No interrupt 1 = Interrupt condition asserted

4.8 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 enable register. The Interrupt Flag Register is illustrated in Figure 4-8 and described in Table 4-9.

Figure 4-8: Interrupt Flag Register

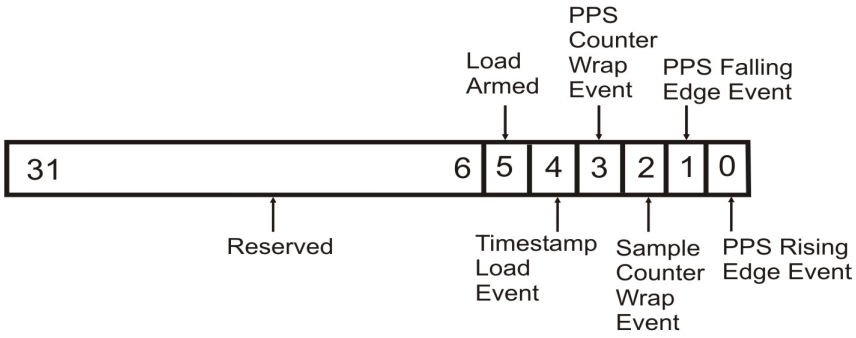


Table 4-9: Interrupt Flag Register (Base Address + 0x1C)

Bits	Field Name	Default Value	Access Type	Description
31:6	Reserved	N/A	N/A	**Reserved**
5	load_armed	0	R/Clr	**Load Armed:** This bit indicates the load armed interrupt flag. The load armed interrupt source indicates that the Counter Load Control State Machine is in the Armed state. **Read:** 0 = No interrupt 1 = Interrupt condition asserted **Clear:** 1 = Clear latch

Table 4-9: Interrupt Flag Register (Base Address + 0x1C) (Continued)

Bits	Field Name	Default Value	Access Type	Description
4	timestamp_load_ event	0	R/Clr	**Timestamp Load Event:** This bit indicates the timestamp load event interrupt flag. The timestamp load event interrupt source indicates that the counters are loaded with initial values and other data to generate the timestamp output. **Read:** 0 = No interrupt 1 = Interrupt condition asserted **Clear:** 1 = Clear latch
3	pps_cntr_wrap _event	0	R/Clr	**PPS Counter Wrap Event:** This bit indicates the PPS counter wrap event interrupt flag. The PPS counter wrap event interrupt source indicates that the PPS counter has reached the maximum value and wrapped around to count from zero (all bits in counter set to 0). **Read:** 0 = No interrupt 1 = Interrupt condition asserted **Clear:** 1 = Clear latch
2	sample_cntr_wrap _event	0	R/Clr	**Sample Counter Wrap Event:** This bit indicates the sample counter wrap event interrupt flag. The sample counter wrap event interrupt source indicates that the sample counter has reached the maximum value and wrapped around to count from zero (all bits in counter set to 0). **Read:** 0 = No interrupt 1 = Interrupt condition asserted **Clear:** 1 = Clear latch
1	pps_falling_edge _event	0	R/Clr	**PPS Signal Falling Edge Event:** This bit indicates the PPS signal falling edge event interrupt flag. **Read:** 0 = No interrupt 1 = Interrupt condition asserted **Clear:** 1 = Clear latch
0	pps_rising_edge _event	0	R/Clr	**PPS Signal Rising Edge Event:** This bit indicates the PPS signal rising edge event interrupt source. **Read:** 0 = No interrupt 1 = Interrupt condition asserted **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 Timestamp Generator Core.

5.1 General Design Guidelines

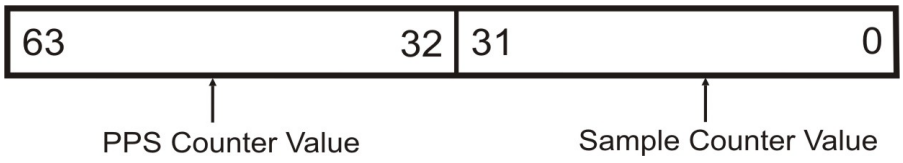
The Timestamp Generator Core provides the required logic to generate timestamp output data from incoming timing events data. This IP core supports AXI4-Lite and AXI4-Stream user interfaces. The user can customize the core based on the input timing signal signal width by setting the generic parameter as described in Section 2.5.

PPS Mode

The Timestamp Generator Core has a 32-bit PPS counter incremented by edge events of the PPS source selected. It also has a 64-bit Sample clock counter to count the number of sample clock cycles that have occurred since the last PPS event.

When PPS mode is enabled, the output AXI4-Stream timestamp data of the core has a 32-bit PPS counter value and 32-bit sample clock counter value as shown in Figures 5-2. When PPS mode is disabled, the sample counter is a free-running counter of the sample clock cycles and the output data is the 64-bit sample counter value.

Figure 5-1: AXI4-Stream Timestamp Output Data with PPS



5.2 Clocking

Sample Clock: **s_axis_aclk**

This clock is used to clock all ports of the core.

CSR Clock: **s_axi_csr_aclk**

This clock is the input AXI4-Lite interface clock to the core which is converted using the AXI Clock converter core to operate the other modules within the Timestamp Generator Core in the Sample Clock domain.

5.3 Resets

Main reset: **s_axis_aresetn**

This is an active low synchronous reset associated with **s_axis_aclk**. When asserted, the state machine in the Timestamp Generator Core is reset.

CSR Reset: **s_axi_csr_aresetn**

This is an active low reset synchronous with **s_axi_csr_clk**. When asserted, the control/status registers and the interrupt registers are reset.

5.4 Interrupts

This core has an edge type (rising edge-triggered) interrupt output. It is synchronous with the **s_axis_aclk**. On the rising edge of any interrupt signal, a one clock cycle wide pulse is output from the core on its **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.

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

CSR Interface: This is the Control/Status Register Interface and is associated with **s_axis_aclk**. It is a standard AXI4-Lite Slave interface. See Chapter 4 for the control register memory map, which provides more details on the registers that can be accessed through this interface.

Timing Events (PTCTL) Interface: This is the interface through which Timing Events are received. It is an AXI4-Stream Slave interface and is associated with **s_axis_aclk**. For more details about this interface, refer to Section 3.2.1.

Timestamp Output (PTS) Interface: This is the interface through which Timestamp outputs are transferred through the output ports. It is an AXI4-Stream Master interface. For more details about this interface, refer to Section 3.2.2.

5.6 Programming Sequence

This section briefly describes the programming sequence of registers in the Timestamp Generator Core.

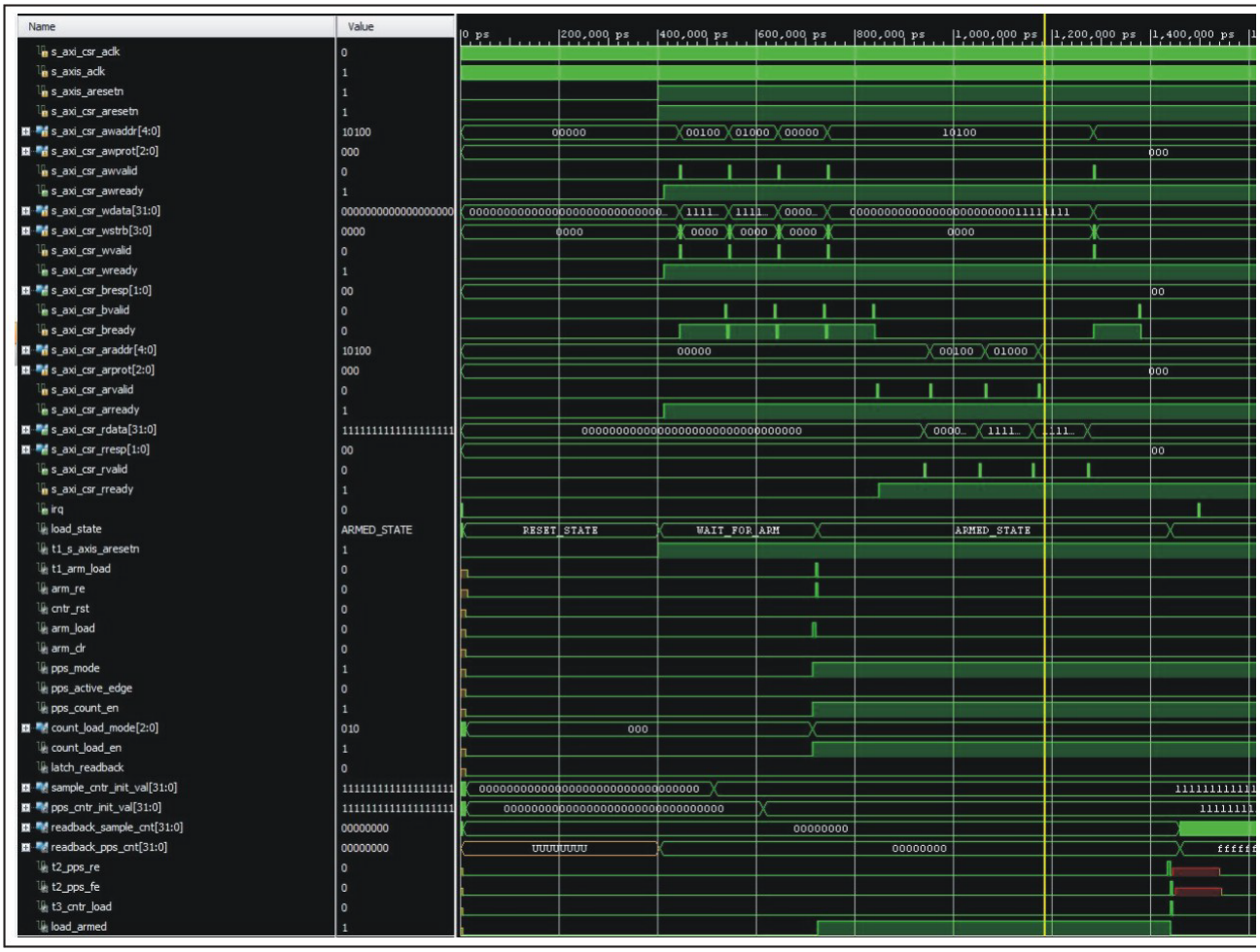
1) Ensure that the interrupt flag register is cleared.
2) Enable the interrupt enable bits based on the user design requirement.
3) Assign the desired value to the generic parameter.
4) Set the control registers with the required values.
5) Observe the outputs across the output ports.
6) When done, check the interrupt flag register and clear the interrupts.

5.7 Timing Diagrams

The timing diagram for the Timestamp Generator Core is shown in Figures 5-2. This timing diagram is obtained by running the simulation of the test bench of the core in Vivado VSim environment. For more details about the test bench, refer to Section 6.5.

The timing diagram depicts the functionality of the core for a rising edge PPS signal. The read-back sample counter and PPS counter values can be read from Figures 5-2. This timing diagram shows three rising edges on the PPS signal for which the PPS counter and the sample clock counter operate.

Figure 5-2: Timestamp Generator IP Core Timing Diagram

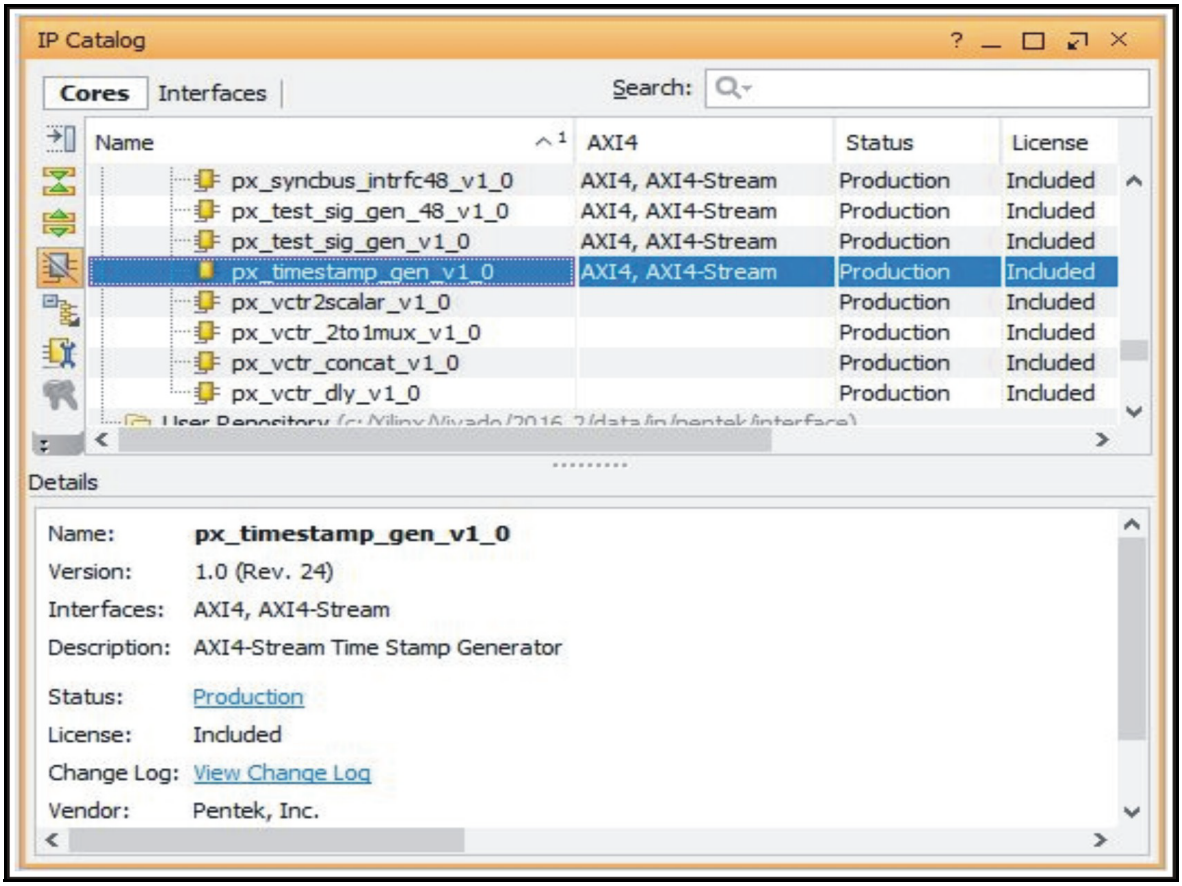


Chapter 6: Design Flow Steps

6.1 Pentek IP Catalog

This chapter describes customization and generation of the Pentek Timestamp Generator Core. It also includes simulation, synthesis, and implementation steps that are specific to this IP core. This 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_timestamp_gen_v1_0** as shown in Figure 6-1.

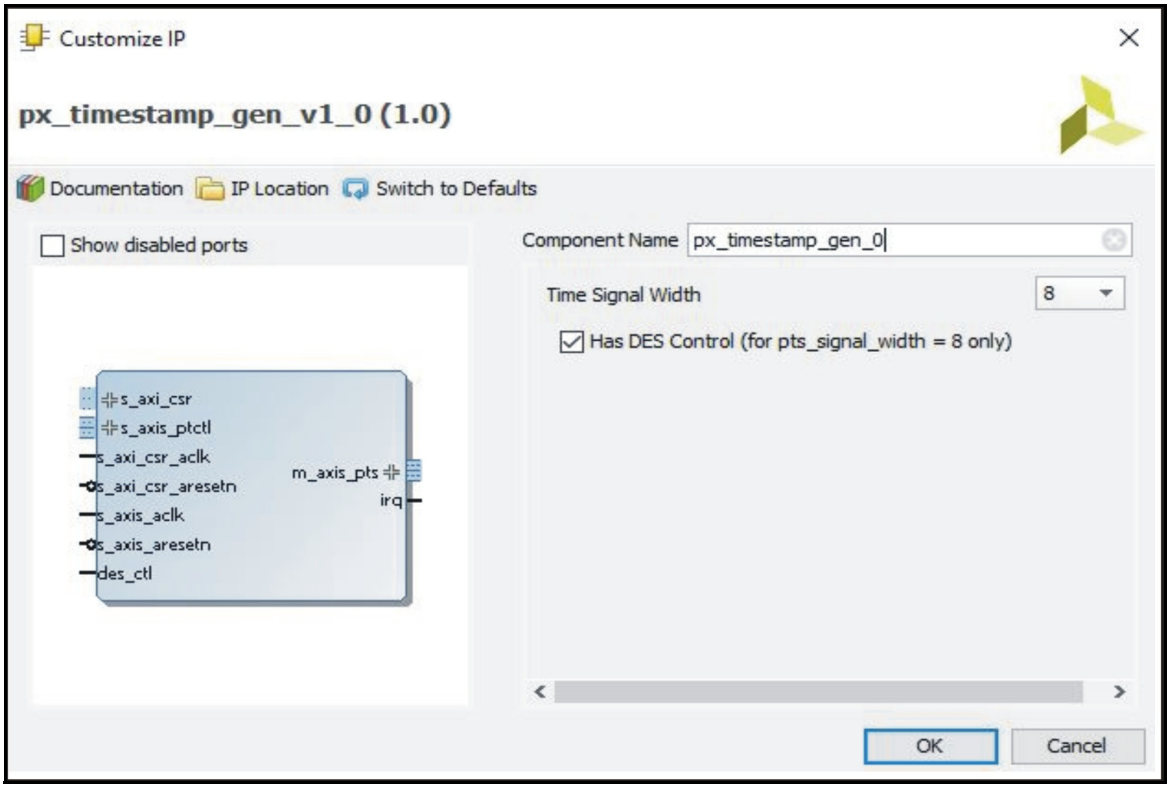
Figure 6-1: Timestamp Generator Core in Pentek IP Catalog



6.1 Pentek IP Catalog (continued)

When you select the **px_timestamp_gen_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: Timestamp Generator Core IP Symbol



6.2 User Parameters

Timing Signal Width: This gives the value for the input timing event data stream width. This parameter can be defined by the user to any of the values 1, 2, 4, or 8.

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 Timestamp Generator Core in Vivado Design Suite.

Required Constraints

The XDC constraints are not provided with the Timestamp Generator Core. Clock 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 clock (**`s_axi_csr_aclk`**) can take frequencies up to 250 MHz. The sample clock (**`s_axis_aclk`**) has a maximum frequency of 345 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 Timestamp Generator Core has a test bench which generates the output waveforms using the Vivado VSim environment. The test bench is designed to run at 250 MHz input clock frequency and Sample clock frequency of 345 MHz. The test bench uses a timing signal width of 1 and provides parameter values through the **parameters.txt** file. The contents of the **test_parameters.txt** file along with descriptions of the parameters are provided in Table 6-1.

6.5 Simulation (continued)

Table 6-1: Test Parameters File Contents and Parameter Descriptions

Parameter	Type	Value	Description
load_mode	std_logic_vector	0x2	**Counter Load Mode:** This parameter indicates the ramp counter and DDS offset load mode based on the source of the PPS signal. 000 - Always on ARM 001 - AUX pulse rising edge 010 - PPS rising edge 011 - PPS falling edge 100 - SYNC rising edge 101 - SYNC falling edge 110 - Gate rising edge 111 - Gate falling edge
load_enable	Boolean	True	**Counter Load Enable:** When True, this parameter enables the load on the counter at the rising edge or falling edge of the selected source signal when the state machine is in the Armed state.
pps_mode			**PPS Mode:** When True, the PPS mode of the timestamp generator core is enabled.
pps_rising_active_edge			**PPS Active Edge:** This parameter defines the active edge of the PPS signal. When True, rising edge is the active PPS edge. When False, falling edge is the active PPS edge
sample_cntr_init_val	std_logic_vector	0xFFFFFFFE	**Sample Counter Initial Value:** This is the initial value loaded into the lower 32 bits of the Sample counter.
pps_cntr_init_val		0xFFFFFFF0	**PPS Counter Initial Value:** This is the initial PPS counter vale loaded into the upper 32 bits of the Sample counter.
ticks_per_cycle	Integer	1	**Timestamp Ticks per Cycle:**
cycles_per_pps		32	**Clock cycles between PPS pulses in simulation**

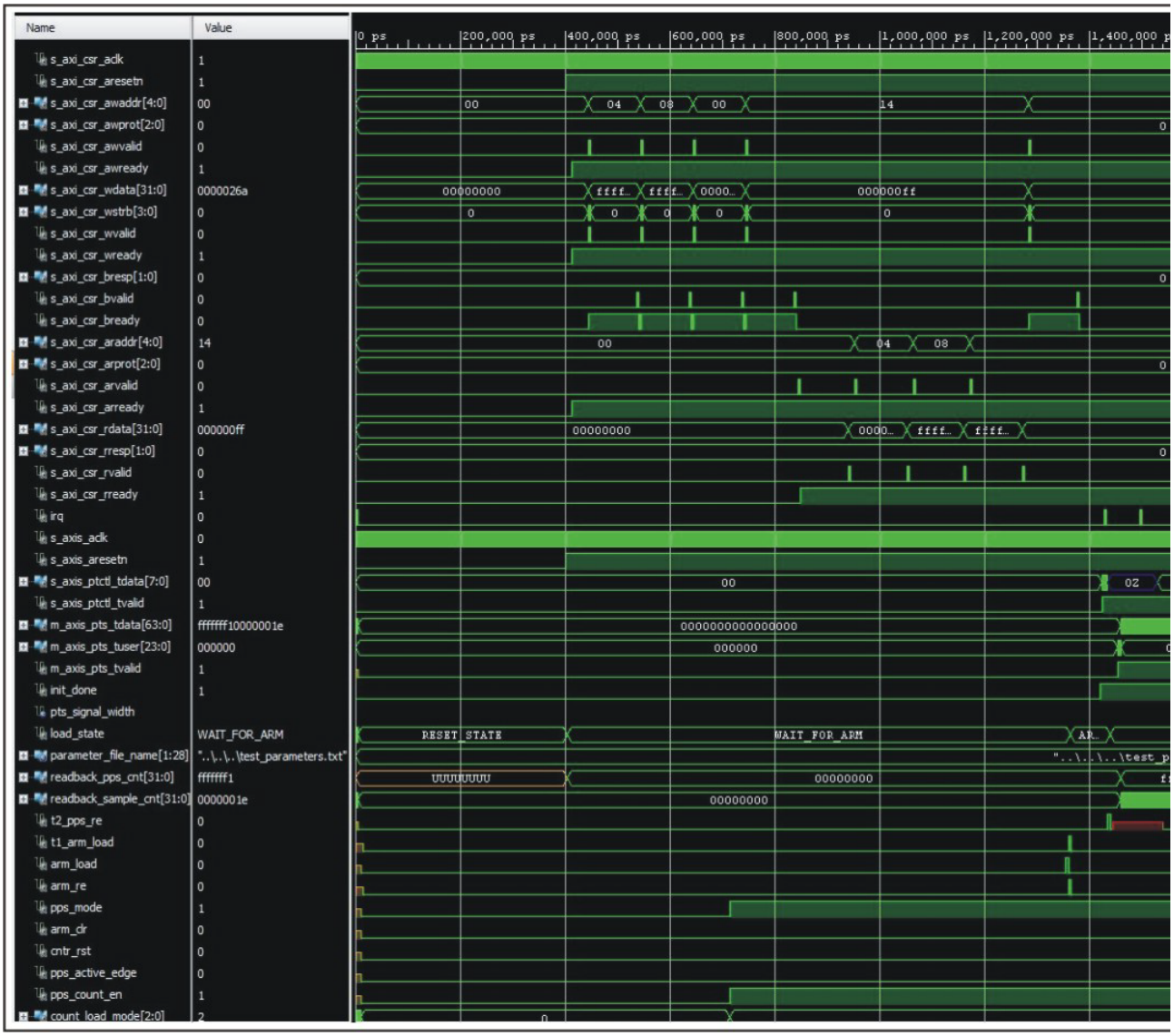
The test bench also assigns the required values to the control registers and Interrupt Enable Register. The test bench runs for a rising edge PPS signal with the PPS mode enabled. The **Arm Load** bit of the Mode Control Register is set High for one clock cycle.

During this clock cycle, the **Counter** load control state machine moves from the **Wait for Arm** state to the **Armed** state. In this state, it waits for the rising edge of the PPS signal. Once the PPS signal has a rising edge event, the state machine loads the counters and then moves to the **Wait for Arm** state.

6.5 Simulation (continued)

The counters increment for each event on the PPS signal until they are reset. The programming procedure is the same as described in Section 5.6. When run, the simulation produces the results shown in Figure 6-3.

Figure 6-3: Timestamp Generator Core Test Bench Simulation Output



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