

## Introduction

This document describes the specifications for a Advanced Microcontroller Bus Architecture (AMBA®) specification's Advanced eXtensible Interface (AXI) Timer/Counter core.

The AXI Timer/Counter is a 32-bit timer module that attaches to the AXI4-Lite interface.

## Features

- AXI interface is based on the AXI4-Lite specification
- Two programmable interval timers with interrupt, event generation, and event capture capabilities
- Configurable counter width
- One Pulse Width Modulation (PWM) output
- Freeze input for halting counters during software debug

LogiCORE IP Facts					
Core Specifics					
Supported Device Family <sup>(1)</sup>	Virtex-6, Spartan-6				
Supported User Interfaces	AXI4-Lite only				
Configuration	Resources				Frequency
	Slices	LUTs	FFs	Block RAMs	Max Freq
	See <a href="#">Table 9</a> and <a href="#">Table 10</a>				0
Provided with Core					
Documentation	Product Specification				
Design Files	VHDL				
Example Design	Not Provided				
Test Bench	Not Provided				
Constraints File	None				
Simulation Model	None				
Tested Design Tools					
Design Entry Tools	ISE 12.4 software				
Simulation	Mentor Graphics ModelSim v6.5c				
Synthesis Tools	XST 12.4				
Support					
Provided by Xilinx, Inc.					

1. For the complete list of supported devices, see the [release notes](#) for this core.

## Functional Description

The Timer/Counter is organized as two identical timer modules as shown in [Figure 2](#). Each timer module has an associated load register that is used to hold either the initial value for the counter for event generation or a capture value, depending on the mode of the timer.

The *generate value* is used to generate a single interrupt at the expiration of an interval or a continuous series of interrupts with a programmable interval. The *capture value* is the timer value that has been latched on detection of an external event. The clock rate of the timer modules is S\_AXI\_ACLK (no prescaling of the clock is performed). All of the Timer/Counter interrupts are ORed together to generate a single external interrupt signal. The interrupt service routine reads the control/status registers to determine the source of the interrupt.

## Programming Model

### Timer Modes

There are three modes that can be used with the two Timer/Counter modules:

- Generate mode
- Capture mode
- Pulse Width Modulation (PWM) mode.

The modes and their characteristics are described in the following sections.

### Generate Mode

In the Generate mode, the value in the load register is loaded into the counter. The counter, when enabled, begins to count up or down, depending on the selection of the UDT bit in the Timer Control Status Register (TCSR). See [Figure 6](#) and [Figure 7](#). On transition of the carry out of the counter, the counter stops or automatically reloads the generate value from the load register and continues counting as selected by the ARHT bit in the TCSR. The TINT bit is set in TCSR and, if enabled, the external GenerateOut signal is driven to 1 for one clock cycle. If enabled, the interrupt signal for the timer is driven to 1 for one clock cycle. This mode is useful for generating repetitive interrupts or external signals with a specified interval.

### Characteristics

The generate mode has the following characteristics:

- The value loaded into the load register is called the generate value.
- On startup, the generate value in the load register must be loaded into the counter by setting the Load bit in the Timer Control Status Register (TCSR). This applies whether the counter is set up to Auto Reload or Hold when the interval has expired. Setting the Load bit to '1' loads the counter with the value in the load register. For proper operation, the Load bit must be cleared before the counter is enabled.
- When the ARHT bit (Auto Reload/Hold) is set to '1' and the counter rolls over from all '1's to all '0's when counting up, or conversely from all '0's to all '1's when counting down, the generate value in the load register will be automatically reloaded into the counter and the counter will continue to count. If the GenerateOut signal is enabled (bit GENT in the TCSR), an output pulse will be generated (one clock period in width). This is useful for generating a repetitive pulse train with a specified period.
- When the ARHT bit (Auto Reload/Hold) is set to '0' and the counter rolls over from all '1's to all '0's, when counting up, or conversely, from all '0's to all '1's, when counting down, the counter will hold at the current value and will not reload the generate value. If the GenerateOut signal is enabled (bit GENT in the TCSR), an output pulse of one clock period in width will be generated. This is useful for a one-shot pulse that is to be generated after a specified period of time.

- The counter can be set up to count either up or down as determined by the selection of the UDT bit in the TCSR. If the counter is set up as a down counter, the generate value is the number of clocks in the timing interval. The period of the GenerateOut signal is the generate value times the clock period.
- When the counter is set to count down,  
$$\text{TIMING\_INTERVAL} = (\text{TLR}_x + 2) \times \text{AXI\_CLOCK\_PERIOD}$$
- When the counter is set to count up,  
$$\text{TIMING\_INTERVAL} = (\text{MAX\_COUNT} - \text{TLR}_x + 2) \times \text{AXI\_CLOCK\_PERIOD}$$
where MAX\_COUNT is the maximum count value of the counter, such as 0xFFFFFFFF for a 32-bit counter.
- The GenerateOut signals can be configured as high-true or low-true.

## Capture Mode

In Capture Mode, the value of the counter is stored in the load register when the external capture signal is asserted. The TINT bit is also set in the Timer Control Status Register (TCSR) on detection of the capture event. The counter can be configured as an up or down counter for this mode as determined by the selection of the UDT bit in TCSR. The ARHT bit controls whether the capture value is overwritten with a new capture value before the previous TINT flag is cleared. This mode is useful for time tagging external events while simultaneously generating an interrupt.

### Characteristics

Capture Mode has the following characteristics:

- The capture signal can be configured to be low-true or high-true.
- The capture signal is sampled within the Timer/Counter with the S\_AXI\_ACLK. The capture event is defined as the transition on the sampled signal to the asserted state. For example, if the capture signal is defined to be high-true, then the capture event is when the sampled signal, synchronized to the S\_AXI\_ACLK, transitions from '0' to '1'.
- When the capture event occurs, the counter value is written to the load register. This value is called the capture value.
- When the ARHT bit (Auto Reload/Hold) is set to '0' and the capture event occurs, the capture value is written to the Load Register which holds the capture value until the load register is read. If the load register is not read, subsequent capture events will not update the load register and will be lost.
- When the ARHT bit (Auto Reload/Hold) is set to '1', and the capture event occurs, the capture value is always written to the load register. Subsequent capture events will update the load register and will overwrite the previous value, whether it has been read or not.
- The counter can be set up to count either up or down as determined by the selection of the UDT bit in the Timer Control Status Register (TCSR).

## Pulse Width Modulation (PWM) Mode

In PWM mode, two timer/counters are used as a pair to produce an output signal (PWM0) with a specified frequency and duty factor. Timer0 sets the period and Timer1 sets the high time for the PWM0 output.

### Characteristics

PWM Mode has the following characteristics:

- The mode for both Timer0 and Timer1 must be set to Generate Mode (bit MDT in the TCSR set to '0').
- The PWMA0 bit in TCSR0 and PWMB0 bit in TCSR1 must be set to '1' to enable PWM mode.
- The GenerateOut signals must be enabled in the TCSR (bit GENT set to '1'). The PWM0 signal is generated from the GenerateOut signals of Timer0 and Timer1, so these signals must be enabled in both timer/counters.
- The assertion level of the GenerateOut signals for both timers in the pair must be set to '1'. This is done by setting C\_GEN0\_ASSERT and C\_GEN1\_ASSERT to '1'.
- The counter can be set to count up or down.

### Setting the PWM Period and Duty Factor

The PWM period is determined by the generate value in the Timer0 load register (TLR0). The PWM high time is determined by the generate value in the Timer1 load register (TLR1). The period and duty factor are calculated as follows:

When counters are configured to count up (UDT = '0'):

$$\begin{aligned} \text{PWM\_PERIOD} &= (\text{MAX\_COUNT} - \text{TLR0} + 2) \times \text{AXI\_CLOCK\_PERIOD} \\ \text{PWM\_HIGH\_TIME} &= (\text{MAX\_COUNT} - \text{TLR1} + 2) \times \text{AXI\_CLOCK\_PERIOD} \end{aligned}$$

When counters are configured to count down (UDT = '1'):

$$\begin{aligned} \text{PWM\_PERIOD} &= (\text{TLR0} + 2) \times \text{AXI\_CLOCK\_PERIOD} \\ \text{PWM\_HIGH\_TIME} &= (\text{TLR1} + 2) \times \text{AXI\_CLOCK\_PERIOD} \end{aligned}$$

where MAX\_COUNT is the maximum count value for the counter, such as 0xFFFFFFFF for a 32-bit counter.

### Interrupts

The TC interrupt signals can be enabled or disabled with the ENIT bit in the TCSR. The interrupt status bit (TINT) in the TCSR cannot be disabled and always reflects the current state of the timer interrupt. In Generate Mode, a timer interrupt is caused by the counter rolling over (the same condition used to reload the counter when ARHT is set to '1'). In Capture Mode, the interrupt event is the capture event. Characteristics of the interrupts are:

- Interrupt events can only occur when the timer is enabled. In Capture Mode, this prevents interrupts from occurring before the timer is enabled.
- The interrupt signal goes high when the interrupt condition is met and the interrupt is enabled in the TCSR. The interrupt is asserted when the interrupt signal is high.
- A single interrupt signal is provided. The interrupt signal is the OR of the interrupts from the two counters. The interrupt service routine must poll the TCSRs to determine the source or sources of the interrupt.
- The interrupt status bit (TINT in the TCSR) can only be cleared by writing a '1' to it. Writing a '0' to it has no effect on the bit. Since the interrupt condition is an edge (the counter rollover or the capture event), it can be cleared at any time and will not indicate an interrupt condition until the next interrupt event.

The top level block diagram of the AXI Timer/Counter is shown in [Figure 1](#).

The top level modules of the AXI Timer/Counter are:

- AXI Lite IPIF Module
- Timer/Counter

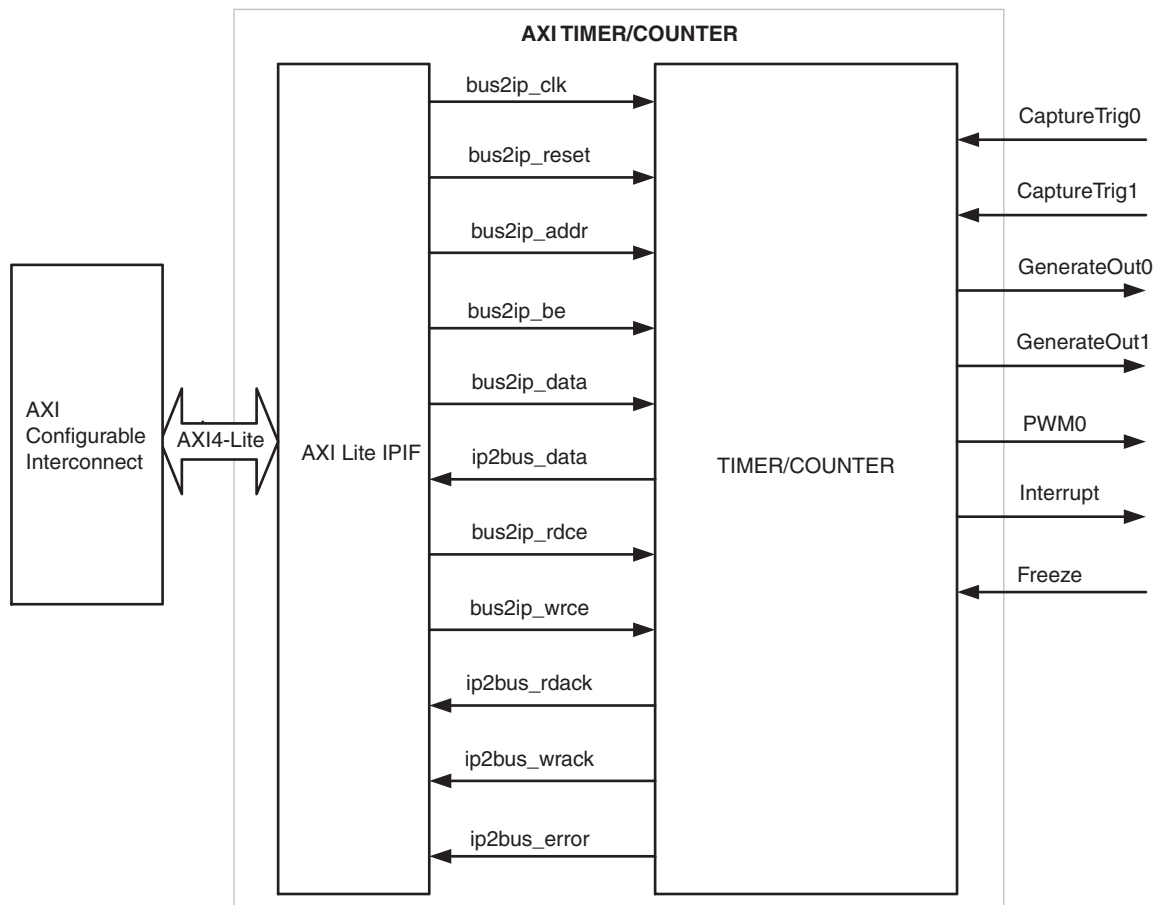


Figure 1: Block Diagram of AXI Timer

The detailed block diagram of the AXI Timer/Counter is shown in [Figure 2](#).

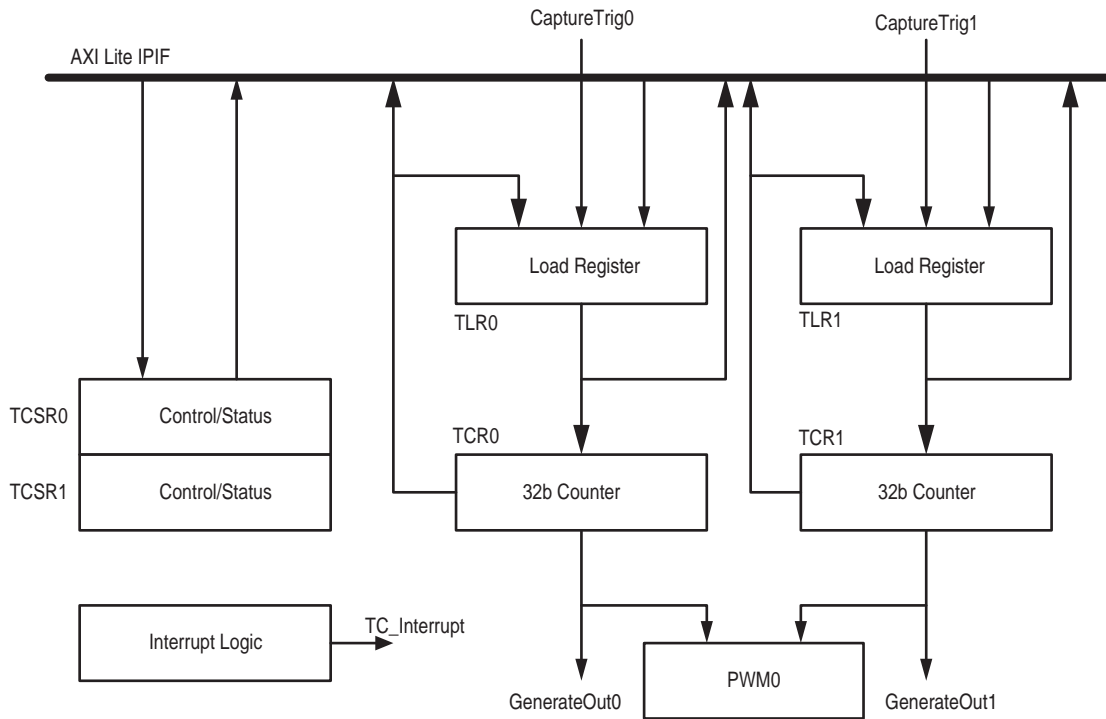


Figure 2: Detailed Block Diagram of AXI Timer

## I/O Signals

The AXI Timer I/O signals are listed and described in [Table 1](#).

Table 1: I/O Signal Description

Port	Signal Name	Interface	I/O	Initial State	Description
System Signals					
P1	S_AXI_ACLK	System	I	-	AXI Clock
P2	S_AXI_ARESETn	System	I	-	AXI Reset, active low
P3	Interrupt	System	O	0x0	AXI Timer Interrupt; Edge sensitive, rising
AXI Write Address Channel Signals					
P4	S_AXI_AWADDR[C_ S_AXI_ADDR_WIDTH -1:0]	AXI	I	-	AXI Write address. The write address bus gives the address of the write transaction.
P5	S_AXI_AWVALID	AXI	I	-	Write address valid. This signal indicates that valid write address is available.
P6	S_AXI_AWREADY	AXI	O	0x0	Write address ready. This signal indicates that the slave is ready to accept an address.

Table 1: I/O Signal Description (Cont.)

Port	Signal Name	Interface	I/O	Initial State	Description
AXI Write Channel Signals					
P7	S_AXI_WDATA[C_S_AXI_DATA_WIDTH - 1:0]	AXI	I	-	Write data
P8	S_AXI_WSTB[C_S_AXI_DATA_WIDTH/8-1:0]	AXI	I	-	Write strobes. This signal indicates which byte lanes to update in memory. <sup>(1)</sup>
P9	S_AXI_WVALID	AXI	I	-	Write valid. This signal indicates that valid write data and strobes are available.
P10	S_AXI_WREADY	AXI	O	0x0	Write ready. This signal indicates that the slave can accept the write data.
AXI Write Response Channel Signals					
P11	S_AXI_BRESP[1:0]	AXI	O	0x0	Write response. This signal indicates the status of the write transaction. "00" - OKAY (normal response) "10" - SLVERR (error condition)
P12	S_AXI_BVALID	AXI	O	0x0	Write response valid. This signal indicates that a valid write response is available.
P13	S_AXI_BREADY	AXI	I	-	Response ready. This signal indicates that the master can accept the response information.
AXI Read Address Channel Signals					
P14	S_AXI_ARADDR[C_S_AXI_ADDR_WIDTH - 1:0]	AXI	I	-	Read address. The read address bus gives the address of a read transaction.
P15	S_AXI_ARVALID	AXI	I	-	Read address valid. This signal indicates, when HIGH, that the read address is valid and will remain stable until the address acknowledgement signal, S_AXI_ARREADY, is high.
P16	S_AXI_ARREADY	AXI	O	0x1	Read address ready. This signal indicates that the slave is ready to accept an address.
AXI Read Data Channel Signals					
P17	S_AXI_RDATA[C_S_AXI_DATA_WIDTH - 1:0]	AXI	O	0x0	Read data
P18	S_AXI_RRESP[1:0]	AXI	O	0x0	Read response. This signal indicates the status of the read transfer. "00" - OKAY (normal response) "10" - SLVERR (error condition)
P19	S_AXI_RVALID	AXI	O	0x0	Read valid. This signal indicates that the required read data is available and the read transfer can complete.

Table 1: I/O Signal Description (Cont.)

Port	Signal Name	Interface	I/O	Initial State	Description
P20	S_AXI_RREADY	AXI	I	-	Read ready. This signal indicates that the master can accept the read data and response information.
AXI Timer/Counter Signals					
P21	CaptureTrig0	Timer	I	-	Capture Trigger 0
P22	CaptureTrig1	Timer	I	-	Capture Trigger 1
P23	Freeze	Timer	I	-	Freeze Count Value
P24	GenerateOut0	Timer	O	0x0	Generate Output 0
P25	GenerateOut1	Timer	O	0x0	Generate Output 1
P26	PWM0	Timer	O	0x0	Pulse Width Modulation Output 0

**Notes:**

1. This signal is not used. The AXI Timer assumes that all byte lanes are active.

## Design Parameters

To allow users to create the AXI Timer that is uniquely tailored for their systems, certain features can be parameterized in the AXI Timer design. This allows users to have a design that only utilizes the resources required by the system and operates at the best possible performance. The AXI Timer design parameters are shown in [Table 2](#).

In addition to the parameters listed in this table, there are also parameters that are inferred for each AXI interface in the EDK tools. Through the design, these EDK-inferred parameters control the behavior of the AXI Interconnect. For a complete list of the interconnect settings related to the AXI interface, see [DS768 AXI Interconnect IP Data Sheet](#).

Table 2: Design Parameters

Generic	Feature/Description	Parameter Name	Allowable Values	Default Value	VHDL Type
System Parameter					
G1	Target FPGA family	C_FAMILY	virtex6, spartan6	virtex6	string
AXI Parameters					
G2	AXI Protocol type	C_S_AXI_PROTOCOL	AXI4LITE	AXI4LITE	string
G3	AXI Base Address	C_BASEADDR	Valid Address <sup>(1)</sup>	0xFFFFFFFF <sup>(3)</sup>	std_logic_vector
G4	AXI High Address	C_HIGHADDR	Valid Address <sup>(2)</sup>	0x00000000 <sup>(3)</sup>	std_logic_vector
G5	AXI address bus width	C_S_AXI_ADDR_WIDTH	32	32	integer
G6	AXI data bus width	C_S_AXI_DATA_WIDTH	32	32	integer



Table 2: Design Parameters (Cont.)

Generic	Feature/Description	Parameter Name	Allowable Values	Default Value	VHDL Type
Timer Parameters					
G7	The width in bits of the counters in the AXI Timer/Counter	C_COUNT_WIDTH	8 - 32	32	integer
G8	Number of Timer modules	C_ONE_TIMER_ONLY	0 = Two timers are present 1 = One timer is present (No PWM mode)	0	integer
G9	Assertion level for CaptureTrig0	C_TRIG0_ASSERT	'0' = CaptureTrig0 input is low-true '1' = CaptureTrig0 input is high-true	1	std_logic
G10	Assertion level for CaptureTrig1	C_TRIG1_ASSERT	'0' = CaptureTrig1 input is low-true '1' = CaptureTrig1 input is high-true	1	std_logic
G11	Assertion level for GenerateOut0	C_GEN0_ASSERT	'0' = GenerateOut0 output is low-true '1' = GenerateOut0 output is high-true	1	std_logic
G12	Assertion level for GenerateOut1	C_GEN1_ASSERT	'0' = GenerateOut1 output is low-true '1' = GenerateOut1 output is high-true	1	std_logic

**Notes:**

1. The user must set the values. The C\_BASEADDR must be a multiple of the range, where the range is C\_HIGHADDR - C\_BASEADDR + 1.
2. The range specified by C\_HIGHADDR - C\_BASEADDR must be a power of 2 and greater than or equal to 0xFFF.
3. An invalid default value is used to require the user to set this parameter correctly.

## Allowable Parameter Combinations

The address range specified by C\_BASEADDR and C\_HIGHADDR must be a power of 2 and must be at least 0xFFF. For example, if C\_BASEADDR = 0xE0000000, C\_HIGHADDR must be at least = 0x0xE000FFF.

## Parameter - I/O Signal Dependencies

The dependencies between the AXI timer core design parameters and I/O signals are described in [Table 3](#). In addition, when certain features are parameterized out of the design, the related logic will no longer be a part of the design. The unused input signals and related output signals are set to a specified value.

Table 3: Parameter-I/O Signal Dependencies

Generic or Port	Name	Affects	Depends	Relationship Description
Design Parameters				
G5	C_S_AXI_ADDR_WIDTH	P4, P14	-	Defines the width of the ports
G6	C_S_AXI_DATA_WIDTH	P7, P8, P17	-	Defines the width of the ports
I/O Signals				
P4	S_AXI_AWADDR[C_S_AXI_ADDR_WIDTH-1:0]	-	G5	Port width depends on the generic C_S_AXI_ADDR_WIDTH.
P7	S_AXI_WDATA[C_S_AXI_DATA_WIDTH-1:0]	-	G6	Port width depends on the generic C_S_AXI_DATA_WIDTH.
P8	S_AXI_WSTB[C_S_AXI_DATA_WIDTH/8-1:0]	-	G6	Port width depends on the generic C_S_AXI_DATA_WIDTH.
P14	S_AXI_ARADDR[C_S_AXI_ADDR_WIDTH -1:0]	-	G5	Port width depends on the generic C_S_AXI_ADDR_WIDTH.
P17	S_AXI_RDATA[C_S_AXI_DATA_WIDTH -1:0]	-	G6	Port width depends on the generic C_S_AXI_DATA_WIDTH.

## Register Data Types and Organization

Timer Counter registers are accessed as one of the following types:

- Byte (8 bits)
- Half word (2 bytes)
- Word (4 bytes)

The AXI Timer/Counter registers are organized as little-endian data. The bit and byte labeling for the little-endian data types is shown in the [Figure 3](#).

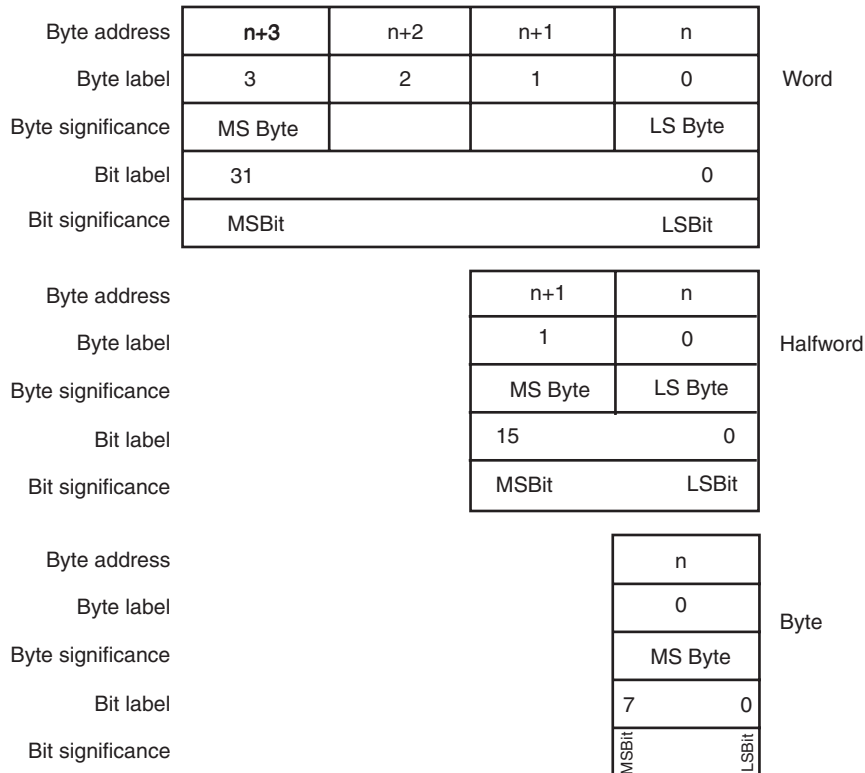


Figure 3: Little Endian Data Types

## Register Descriptions

Table 4 shows all the AXI Timer registers and their addresses. Accesses to addresses inside the core address range other than these registers will return an OKAY response, with reads returning 0 data values and writes having no effect.

Table 4: Register Overview

Register Name	Base Address + offset (hex)	Access Type	Default Value (hex)	Description
TCSR0	C_BASEADDR + 0x00	Read/Write	0x0	Control/Status Register 0
TLR0	C_BASEADDR + 0x04	Read/Write	0x0	Load Register 0
TCR0	C_BASEADDR + 0x08	Read	0x0	Timer/Counter Register 0
TCSR1	C_BASEADDR + 0x10	Read/Write	0x0	Control/Status Register 1
TLR1	C_BASEADDR + 0x14	Read/Write	0x0	Load Register 1
TCR1	C_BASEADDR + 0x18	Read	0x0	Timer/Counter Register 1

### Load Register (TLR0 and TLR1)

When the counter width has been configured as less than 32 bits, the load register value is right-justified in TLR0 and TLR1. The least-significant counter bit is always mapped to load register bit 0.

Figure 4 and Table 5 show the load register.

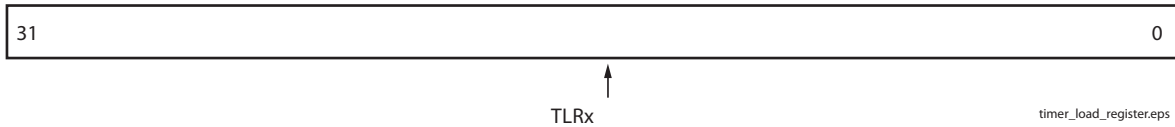


Figure 4: Timer/Counter Load Register

Table 5: Load Register Bit Definitions

Bit(s)	Name	Core Access	Reset Value	Description
31-0	Timer/Counter Load Register	Read/Write	0x0	Timer/Counter Load register

### Timer/Counter Register (TCR0 and TCR1)

When the counter width has been configured as less than 32 bits, the count value is right-justified in TCR0 and TCR1. The least-significant counter bit is always mapped to Timer/Counter Register bit 0. Figure 5 and Table 6 show the Timer/counter register.

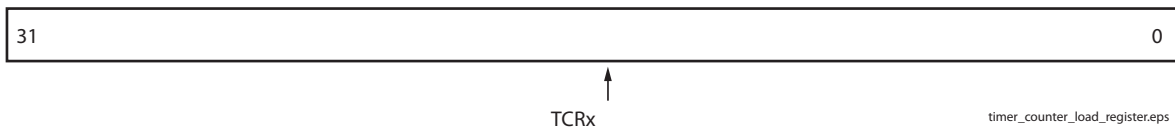


Figure 5: Timer/Counter Register

Table 6: Timer/Counter Register Bit Definitions

Bit(s)	Name	Core Access	Reset Value	Description
31-0	Timer/Counter Register	Read	0x0	Timer/Counter register

### Control/Status Register 0 (TCSR0)

Figure 6 and Table 7 show the Control/Status register 0. Control/Status Register 0 contains the control and status bits for timer module 0.

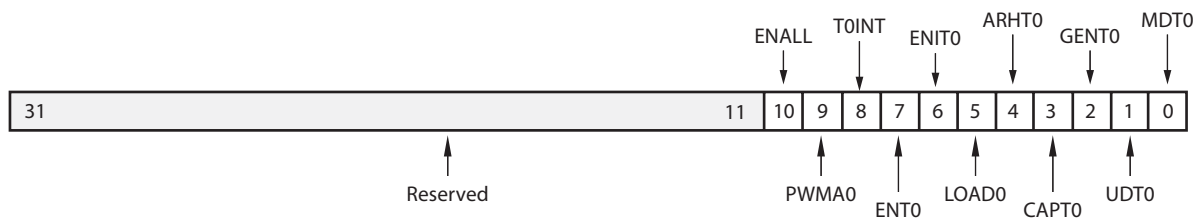


Figure 6: Control/Status Register 0

Table 7: Control/Status Register 0 Bit Definitions

Bit(s)	Name	Core Access	Reset Value	Description
31 - 11	Reserved	N/A	-	Reserved
10	ENALL	Read/Write	0	<b>Enable All Timers</b> 0 = No effect on timers 1 = Enable all timers (counters run) This bit is mirrored in all control/status registers and is used to enable all counters simultaneously. Writing a '1' to this bit sets ENALL, ENT0, and ENT1. Writing a '0' to this register clears ENALL but has no effect on ENT0 and ENT1.
9	PWMA0	Read/Write	0	<b>Enable Pulse Width Modulation for Timer0</b> 0 = Disable pulse width modulation 1 = Enable pulse width modulation PWM requires using Timer0 and Timer1 together as a pair. Timer0 sets the period of the PWM output, and Timer1 sets the high time for the PWM output. For PWM Mode, MDT0 and MDT1 must be '0' and C_GEN0_ASSERT and C_GEN1_ASSERT must be '1'.
8	TOINT	Read/Write	0	<b>Timer0 Interrupt</b> Indicates that the condition for an interrupt on this timer has occurred. If the timer mode is capture and the timer is enabled, this bit indicates a capture has occurred. If the mode is generate, this bit indicates the counter has rolled over. Must be cleared by writing a '1'. <i>Read:</i> 0 = No interrupt has occurred 1 = Interrupt has occurred <i>Write:</i> 0 = No change in state of TOINT 1 = Clear TOINT (clear to '0')
7	ENT0	Read/Write	0	<b>Enable Timer0</b> 0 = Disable timer (counter halts) 1 = Enable timer (counter runs)
6	ENIT0	Read/Write	0	<b>Enable Interrupt for Timer0</b> Enables the assertion of the interrupt signal for this timer. Has no effect on the interrupt flag in TCSR0. 0 = Disable interrupt signal 1 = Enable interrupt signal
5	LOAD0	Read/Write	0	<b>Load Timer0</b> 0 = No load 1 = Loads timer with value in TLR0

Table 7: Control/Status Register 0 Bit Definitions (Cont.)

Bit(s)	Name	Core Access	Reset Value	Description
4	ARHT0	Read/Write	0	<b>Auto Reload/Hold Timer0</b> When the timer is in Generate Mode, this bit determines whether the counter reloads the generate value and continues running or holds at the termination value. In Capture Mode, this bit determines whether a new capture trigger overwrites the previous captured value or if the previous value is held. 0 = Hold counter or capture value 1 = Reload generate value or overwrite capture value
3	CAPT0	Read/Write	0	Enable External Capture Trigger Timer0 0 = Disables external capture trigger 1 = Enables external capture trigger
2	GENT0	Read/Write	0	Enable External Generate Signal Timer0 0 = Disables external generate signal 1 = Enables external generate signal
1	UDT0	Read/Write	0	Up/Down Count Timer0 0 = Timer functions as up counter 1 = Timer functions as down counter
0	MDT0	Read/Write	0	Timer0 Mode See the <a href="#">Timer Modes</a> section. 0 = Timer mode is generate 1 = Timer mode is capture

### Control/Status Register 1 (TCSR1)

Figure 7 and Table 8 show the Control/Status register 1. Control/Status Register 1 contains the control and status bits for timer module 1.



Figure 7: Control/Status Register 1

Table 8: Control/Status Register 1 Bit Definitions

Bit(s)	Name	Core Access	Reset Value	Description
31 - 11	Reserved	N/A	-	Reserved
10	ENALL	Read/Write	0	<b>Enable All Timers</b> 0 = No effect on timers 1 = Enable all timers (counters run) This bit is mirrored in all control/status registers and is used to enable all counters simultaneously. Writing a '1' to this bit sets ENALL, ENT0, and ENT1. Writing a '0' to this register clears ENALL but has no effect on ENT0 and ENT1.
9	PWMA0	Read/Write	0	<b>Enable Pulse Width Modulation for Timer1</b> 0 = Disable pulse width modulation 1 = Enable pulse width modulation PWM requires using Timer0 and Timer1 together as a pair. Timer0 sets the period of the PWM output, and Timer1 sets the high time for the PWM output. For PWM Mode, MDT0 and MDT1 must be '0' and C_GEN0_ASSERT and C_GEN1_ASSERT must be '1'.
8	T1INT	Read/Write	0	<b>Timer1 Interrupt</b> Indicates that the condition for an interrupt on this timer has occurred. If the timer mode is capture and the timer is enabled, this bit indicates a capture has occurred. If the mode is generate, this bit indicates the counter has rolled over. Must be cleared by writing a '1'. <i>Read:</i> 0 = No interrupt has occurred 1 = Interrupt has occurred <i>Write:</i> 0 = No change in state of T1INT 1 = Clear T1INT (clear to '0')
7	ENT1	Read/Write	0	<b>Enable Timer1</b> 0 = Disable timer (counter halts) 1 = Enable timer (counter runs)
6	ENIT1	Read/Write	0	<b>Enable Interrupt for Timer1</b> Enables the assertion of the interrupt signal for this timer. Has no effect on the interrupt flag in TCSR1. 0 = Disable interrupt signal 1 = Enable interrupt signal
5	LOAD1	Read/Write	0	<b>Load Timer1</b> 0 = No load 1 = Loads timer with value in TLR1

Table 8: Control/Status Register 1 Bit Definitions (Cont.)

Bit(s)	Name	Core Access	Reset Value	Description
4	ARHT1	Read/Write	0	<b>Auto Reload/Hold Timer1</b> When the timer is in generate mode, this bit determines whether the counter reloads the generate value and continues running or holds at the termination value. In capture mode, this bit determines whether a new capture trigger overwrites the previous captured value or if the previous value is held until it is read. 0 = Hold counter or capture value 1 = Reload generate value or overwrite capture value
3	CAPT1	Read/Write	0	Enable External Capture Trigger Timer1 0 = Disables external capture trigger 1 = Enables external capture trigger
2	GENT1	Read/Write	0	Enable External Generate Signal Timer1 0 = Disables external generate signal 1 = Enables external generate signal
1	UDT1	Read/Write	0	Up/Down Count Timer1 0 = Timer functions as up counter 1 = Timer functions as down counter
0	MDT1	Read/Write	0	Timer1 Mode See the <a href="#">Timer Modes</a> section. 0 = Timer mode is generate 1 = Timer mode is capture



## Design Implementation

### Target Technology

The intended target technologies are Virtex<sup>®</sup>-6 and Spartan<sup>®</sup>-6 FPGAs.

### Device Utilization and Performance Benchmarks

#### Core Performance

Since the AXI Timer core will be used with other design modules in the FPGA, the utilization and timing numbers reported in this section are estimates only. When the AXI Timer core is combined with other designs in the system, the utilization of FPGA resources and timing of the AXI Timer design will vary from the results reported here.

The AXI Timer resource utilization for various parameter combinations measured with the Virtex-6 FPGA as the target device are detailed in [Table 9](#).

**Table 9: Performance and Resource Utilization Benchmarks on a Virtex-6 FPGA (xc6vlx130t-1-ff1156)**

Parameter Values		Device Resources			Performance
C_ONE_TIMER_ONLY	C_COUNT_WIDTH	Slice Flip-Flops	Slices	LUTs	f <sub>MAX</sub> (MHz)
1	8	76	54	105	224
1	16	97	57	129	232
1	32	147	75	196	241
0	8	97	61	161	210
0	16	137	82	209	209
0	32	214	117	332	204

The AXI Timer resource utilization for various parameter combinations measured with a Spartan-6 FPGA as the target device are detailed in [Table 10](#).

**Table 10: Performance and Resource Utilization Benchmarks on a Spartan-6 FPGA (xc6slx45t-2-fgg484)**

Parameter Values		Device Resources			Performance
C_ONE_TIMER_ONLY	C_COUNT_WIDTH	Slice Flip-Flops	Slices	LUTs	f <sub>MAX</sub> (MHz)
1	8	76	43	103	171
1	16	97	54	128	167
1	32	147	74	195	155
0	8	97	73	161	157
0	16	137	82	208	132
0	32	219	131	342	132

### System Performance

To measure the system performance ( $F_{max}$ ) of this core, this core was added to a Virtex-6 FPGA system and a Spartan-6 FPGA system as the Device Under Test (DUT).

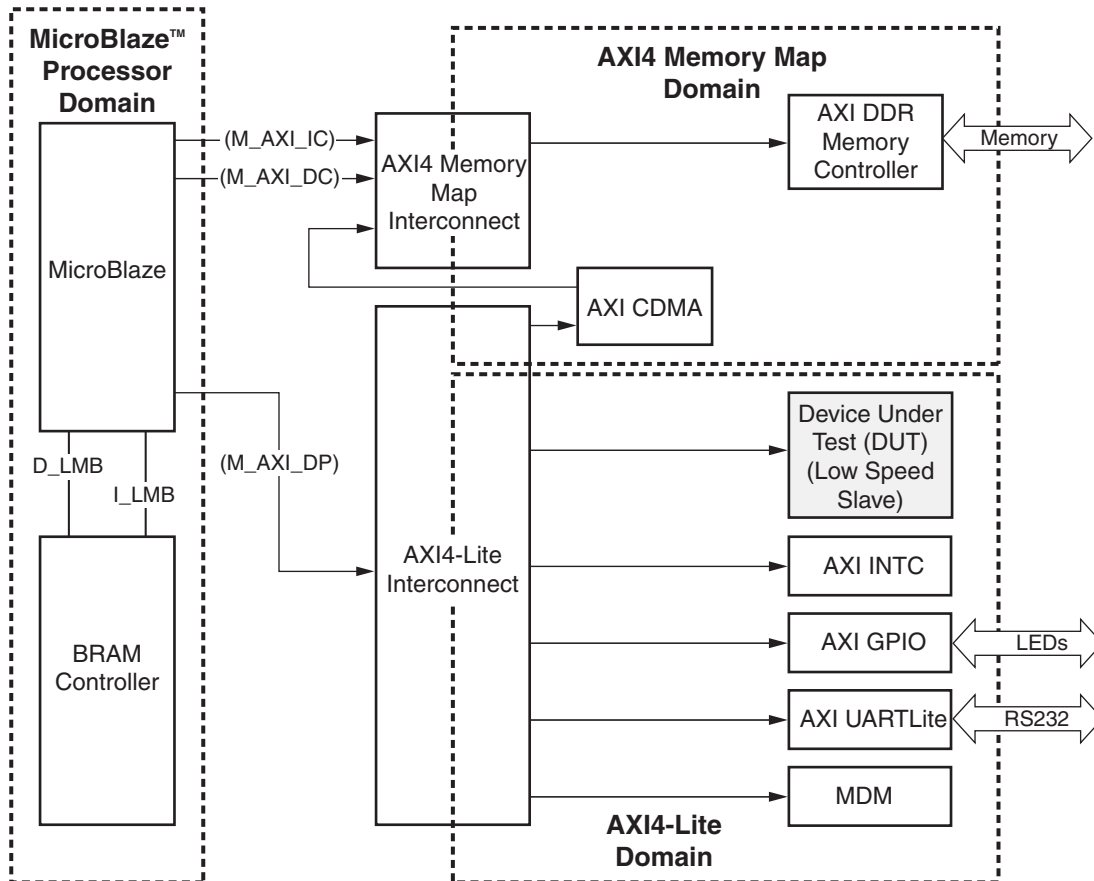


Figure 8: Virtex-6 and Spartan-6 Devices  $F_{MAX}$  Margin System

Because the AXI Timer core will be used with other design modules in the FPGA, the utilization and timing numbers reported in this section are estimates only. When this core is combined with other designs in the system, the utilization of FPGA resources and timing of the design will vary from the results reported here.

The target FPGA was then filled with logic to drive the LUT and block RAM utilization to approximately 70% and the I/O utilization to approximately 80%. Using the default tool options and the slowest speed grade for the target FPGA, the resulting target  $F_{MAX}$  numbers are shown in Table 11.

Table 11: AXI Timer System Performance

Target FPGA	Target $F_{MAX}$ (MHz)
Spartan-6	110
Virtex-6	180

The target  $F_{MAX}$  is influenced by the exact system and is provided for guidance. It is not a guaranteed value across all systems.

## Support

Xilinx provides technical support for this LogiCORE™ product when used as described in the product documentation. Xilinx cannot guarantee timing, functionality, or support of product if implemented in devices that are not defined in the documentation, if customized beyond that allowed in the product documentation, or if changes are made to any section of the design labeled *DO NOT MODIFY*.

## Ordering Information

This Xilinx LogiCORE IP module is provided at no additional cost with the Xilinx ISE® Design Suite Embedded Edition software under the terms of the [Xilinx End User License](#). The core is generated using the Xilinx ISE Embedded Edition software (EDK).

Information about this and other Xilinx LogiCORE IP modules is available at the [Xilinx Intellectual Property](#) page. For information on pricing and availability of other Xilinx LogiCORE modules and software, please contact your [local Xilinx sales representative](#).

## Reference Documents

- AXI4 AMBA AXI Protocol Version: 2.0 Specification
- [DS765](#) LogiCORE IP AXI Lite IPIF Data Sheet
- [DS768](#) AXI Interconnect IP Data Sheet

## List of Acronyms

Acronym	Spelled Out
ARHT	Auto Reload/Hold
AXI	Advanced eXtensible Interface
DUT	Device Under Test
FF	Flip-Flop
FPGA	Field Programmable Gate Array
IP	Intellectual Property
ISE	Integrated Software Environment
LUT	Lookup Table
PWM	Pulse Width Modulation
RAM	Random Access Memory
TCSR	Timer Control Status Register
TLRO	Timer0 load register
TLR1	Timer1 load register
VHDL	VHSIC Hardware Description Language (VHSIC an acronym for Very High-Speed Integrated Circuits)
XST	Xilinx Synthesis Technology

## Revision History

The following table shows the revision history for this document:

Date	Version	Description of Revisions
09/21/10	1.0	First release of the core with AXI interface support.
09/21/10	1.0.1	Documentation only. Added inferred parameters text on page 8.
12/14/10	1.1	Updated to v1.01.a version; updated tools to 12.4.

## Notice of Disclaimer

Xilinx is providing this product documentation, hereinafter "Information," to you "AS IS" with no warranty of any kind, express or implied. Xilinx makes no representation that the Information, or any particular implementation thereof, is free from any claims of infringement. You are responsible for obtaining any rights you may require for any implementation based on the Information. All specifications are subject to change without notice. XILINX EXPRESSLY DISCLAIMS ANY WARRANTY WHATSOEVER WITH RESPECT TO THE ADEQUACY OF THE INFORMATION OR ANY IMPLEMENTATION BASED THEREON, INCLUDING BUT NOT LIMITED TO ANY WARRANTIES OR REPRESENTATIONS THAT THIS IMPLEMENTATION IS FREE FROM CLAIMS OF INFRINGEMENT AND ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Except as stated herein, none of the Information may be copied, reproduced, distributed, republished, downloaded, displayed, posted, or transmitted in any form or by any means including, but not limited to, electronic, mechanical, photocopying, recording, or otherwise, without the prior written consent of Xilinx.