TMS320C6000 DSP 32-Bit Timer Reference Guide

Literature Number: SPRU582A
March 2004

Texas Instruments

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products		**Applications**	
Amplifiers	amplifier.ti.com	Audio	www.ti.com/audio
Data Converters	dataconverter.ti.com	Automotive	www.ti.com/automotive
DSP	dsp.ti.com	Broadband	www.ti.com/broadband
Interface	interface.ti.com	Digital Control	www.ti.com/digitalcontrol
Logic	logic.ti.com	Military	www.ti.com/military
Power Mgmt	power.ti.com	Optical Networking	www.ti.com/opticalnetwork
Microcontrollers	microcontroller.ti.com	Security	www.ti.com/security
		Telephony	www.ti.com/telephony
		Video & Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless

Mailing Address: Texas Instruments
Post Office Box 655303 Dallas, Texas 75265

Preface

Read This First

About This Manual

This document describes the 32-bit timer in the digital signal processors (DSPs) of the TMS320C6000™ DSP family.

Notational Conventions

This document uses the following conventions.

- Hexadecimal numbers are shown with the suffix h. For example, the following number is 40 hexadecimal (decimal 64): 40h.
- Registers in this document are shown in figures and described in tables.
  - Each register figure shows a rectangle divided into fields that represent the fields of the register. Each field is labeled with its bit name, its beginning and ending bit numbers above, and its read/write properties below. A legend explains the notation used for the properties.
  - Reserved bits in a register figure designate a bit that is used for future device expansion.

Related Documentation From Texas Instruments

The following documents describe the C6000™ devices and related support tools. Copies of these documents are available on the Internet at www.ti.com. *Tip:* Enter the literature number in the search box provided at www.ti.com.

TMS320C6000 CPU and Instruction Set Reference Guide (literature number SPRU189) describes the TMS320C6000™ CPU architecture, instruction set, pipeline, and interrupts for these digital signal processors.

TMS320C6000 DSP Peripherals Overview Reference Guide (literature number SPRU190) describes the peripherals available on the TMS320C6000™ DSPs.

TMS320C6000 Technical Brief (literature number SPRU197) gives an introduction to the TMS320C62x™ and TMS320C67x™ DSPs, development tools, and third-party support.

TMS320C64x Technical Overview (SPRU395) gives an introduction to the TMS320C64x™ DSP and discusses the application areas that are enhanced by the TMS320C64x VelociTI™.

TMS320C6000 Programmer's Guide (literature number SPRU198) describes ways to optimize C and assembly code for the TMS320C6000™ DSPs and includes application program examples.

TMS320C6000 Code Composer Studio Tutorial (literature number SPRU301) introduces the Code Composer Studio™ integrated development environment and software tools.

Code Composer Studio Application Programming Interface Reference Guide (literature number SPRU321) describes the Code Composer Studio™ application programming interface (API), which allows you to program custom plug-ins for Code Composer.

TMS320C6x Peripheral Support Library Programmer's Reference (literature number SPRU273) describes the contents of the TMS320C6000™ peripheral support library of functions and macros. It lists functions and macros both by header file and alphabetically, provides a complete description of each, and gives code examples to show how they are used.

TMS320C6000 Chip Support Library API Reference Guide (literature number SPRU401) describes a set of application programming interfaces (APIs) used to configure and control the on-chip peripherals.

Trademarks

Code Composer Studio, C6000, C62x, C64x, C67x, TMS320C6000, TMS320C62x, TMS320C64x, TMS320C67x, and VelociTI are trademarks of Texas Instruments.

Contents

1 Overview 7

2 Resetting the Timers and Enabling Counting 9

3 Timer Counting 10

4 Timer Clock Source Selection 10

5 Timer Pulse Generation 10

6 Boundary Conditions in the Control Registers 12

7 Timer Interrupts 12

8 Timer Pins as General-Purpose Input/Output 13

9 Emulation Operation 13

10 Timer Registers 13

10.1 Timer Control Register (CTL) 14

10.2 Timer Period Register (PRD) 17

10.3 Timer Count Register (CNT) 18

Revision History 19

Figures

1 Timer Block Diagram 8
2 Timer Operation in Pulse Mode (CP = 0) 11
3 Timer Operation in Clock Mode (CP = 1) 11
4 TMS320C62x/C67x Timer Control Register (CTL) 14
5 TMS320C64x Timer Control Register (CTL) 14
6 Timer Period Register (PRD) 17
7 Timer Count Register (CNT) 18

Tables

1 Differences in TMS320C6000 DSP Timers 7
2 Timer Operation Using GO and HLD Bits 9
3 TSTAT Parameters in Pulse and Clock Modes 11
4 Timer Registers 13
5 Timer Control Register (CTL) Field Descriptions 15
6 Timer Period Register (PRD) Field Descriptions 17
7 Timer Count Register (CNT) Field Descriptions 18
8 Document Revision History 19

32-Bit Timer

This document describes the 32-bit timer in the digital signal processors (DSPs) of the TMS320C6000™ DSP family.

1 Overview

The C6000™ DSP device has 32-bit general-purpose timers that can be used to:

- Time events
- Count events
- Generate pulses
- Interrupt the CPU
- Send synchronization events to the DMA

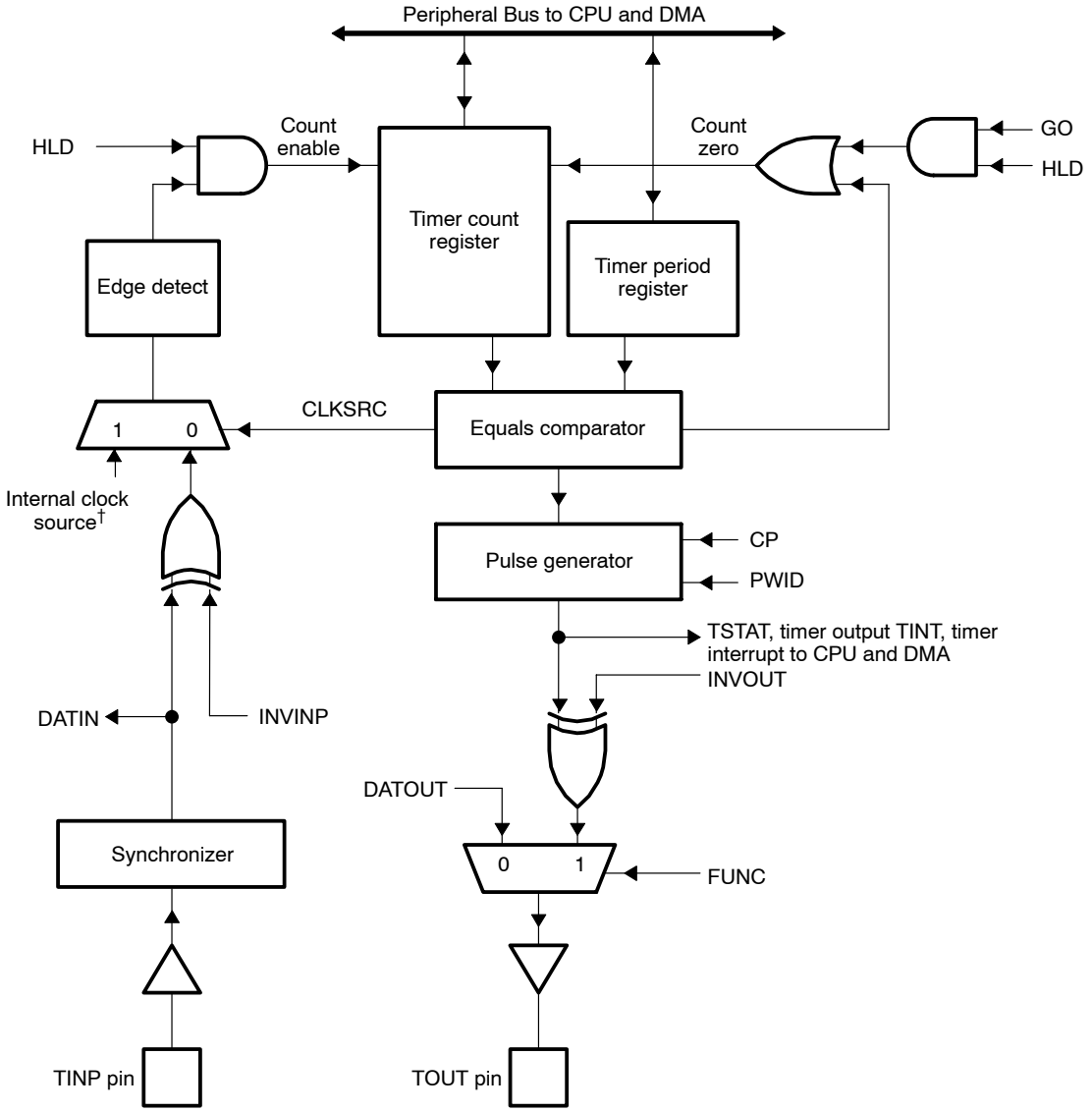
The timers have two signaling modes and can be clocked by an internal or an external source. The timers have an input pin and an output pin. The input and output pins (TINP and TOUT) can function as timer clock input and clock output. They can also be respectively configured for general-purpose input and output.

With an internal clock, for example, the timer can signal an external A/D converter to start a conversion, or it can trigger the DMA controller to begin a data transfer. With an external clock, the timer can count external events and interrupt the CPU after a specified number of events. Table 1 summarizes the differences between the C6000 timers. Figure 1 shows a block diagram of the timers.

Table 1. Differences in TMS320C6000 DSP Timers

Features	C620x/C670x DSP	C621x/C671x DSP	C64x DSP
Emulation halt support	Yes	No	No
Internal timer input clock source frequency	CPU rate/4	CPU rate/4	CPU rate/8

Figure 1. Timer Block Diagram



† C62x/C67x DSP uses CPU/4 clock as the internal clock source to the timer.
C64x DSP uses CPU/8 clock as the internal clock source to the timer.

2 Resetting the Timers and Enabling Counting

Table 2 describes how using the GO and HLD bits in the timer control register (CTL) enable basic features of timer operation.

Table 2. Timer Operation Using GO and HLD Bits

Operation	GO	HLD	Description
Holding the timer	0	0	Counting is disabled.
Restarting the timer after hold	0	1	Timer continues from the value before hold. The timer counter is *not* reset.
Reserved	1	0	Undefined.
Starting the timer	1	1	Timer counter resets to 0 and starts counting whenever enabled. Once set, GO self-clears.

Configuring a timer requires four basic steps:

1) If the timer is not currently in the hold state, place the timer in hold (HLD = 0). Note that after device reset, the timer is already in the hold state.

2) Write the desired value to the timer period register (PRD).

3) Write the desired value to the timer control register (CTL). Do not change the GO and HLD bits in CTL.

4) Start the timer by setting the GO and HLD bits in CTL to 1.

3 Timer Counting

The timer counter runs at the CPU clock rate. However, counting is enabled on the low-to-high transition of the timer count enable source. This transition is detected by the edge-detect circuit shown in Figure 1. Each time an active transition is detected, one CPU-clock-wide clock enable pulse is generated. This makes the counter appear as if it were getting clocked by the count enable source. Thus, this count enable source is referred to as the timer input clock source.

Once the timer reaches a value equal to the value in the timer period register (PRD), the timer is reset to 0 on the next CPU clock. Thus, the counter counts from 0 to N. Consider the case where the period is 2 and the CPU clock/4 is selected as the timer clock source (CLKSRC = 1) for C62x/C67x DSP. Once started, the timer counts the following sequence: 0, 0, 0, 0, 1, 1, 1, 1, 2, 0, 0, 0, 1, 1, 1, 1, 2, 0, 0, 0…. Note that although the counter counts from 0 to 2, the period is 8 (2×4) CPU clock cycles rather than 12 (3×4) CPU clock cycles. Thus, the countdown period is the value of TIMER PERIOD, not TIMER PERIOD + 1.

4 Timer Clock Source Selection

Low-to-high transitions (or high-to-low transitions, if INVINP = 1) of the timer input clock allow the timer counter to increment. Two sources are available to drive the timer input clock:

- The input value on the TINP pin, selected by CLKSRC = 0. This signal is synchronized to prevent any metastability caused by asynchronous external inputs. The value present on the TINP pin is reflected by DATIN.
- Internal clock source, selected by CLKSRC = 1. The C62x/C67x DSPs use CPU clock/4 as an internal clock source. The C64x DSPs use CPU clock/8 as an internal clock source.

5 Timer Pulse Generation

The two basic pulse generation modes are pulse mode (Figure 2) and clock mode (Figure 3). You can select the mode with the CP bit in the timer control register (CTL). Note that in pulse mode, PWID in the CTL can set the pulse width to either one or two input clock periods. The purpose of this feature is to provide minimum pulse widths if TSTAT drives the TOUT output. TSTAT drives this pin when TOUT is used as a timer pin (FUNC = 1), and may be inverted by setting INVOUT = 1. Table 3 gives equations for various TSTAT timing parameters in pulse and clock modes.

Figure 2. Timer Operation in Pulse Mode (CP = 0)

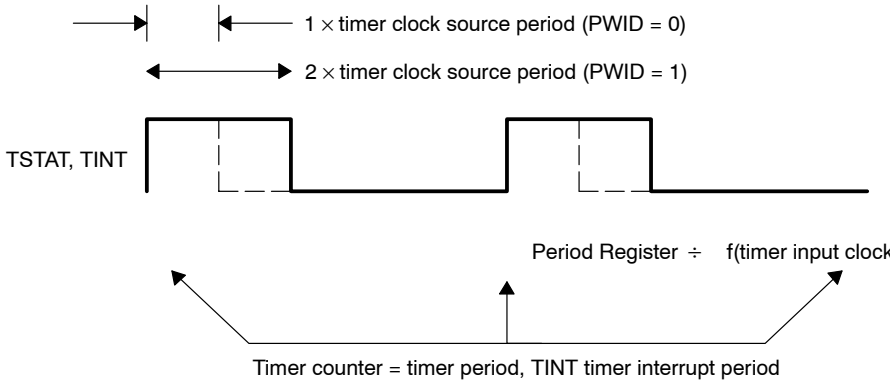


Figure 3. Timer Operation in Clock Mode (CP = 1)

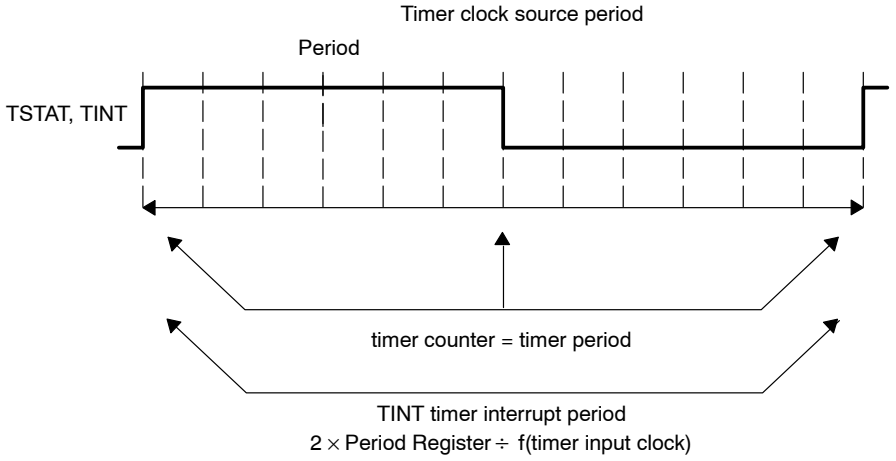


Table 3. TSTAT Parameters in Pulse and Clock Modes

Mode	Frequency	Period	Width High	Width Low
Pulse	$\frac{f\ (clock\ source)}{timer\ period\ register}$	$\frac{timer\ period\ register}{f\ (clock\ source)}$	$\frac{(PWID + 1)}{f\ (clock\ source)}$	$\frac{timer\ period\ register - (PWID + 1)}{f\ (clock\ source)}$
Clock	$\frac{f\ (clock\ source)}{2 * timer\ period\ register}$	$\frac{2 * timer\ period\ register}{f\ (clock\ source)}$	$\frac{timer\ period\ register}{f\ (clock\ source)}$	$\frac{timer\ period\ register}{f\ (clock\ source)}$

6 Boundary Conditions in the Control Registers

The following boundary conditions affect timer operation:

1) Timer period (PRD) and timer count (CNT) value is 0: After device reset and before the timer starts counting, TSTAT is held at 0. After the timer starts running by setting HLD = 1 and GO = 1 while the period and counter registers are 0, the operation of the timer depends on the CP mode selected. In pulse mode, TSTAT = 1 regardless of whether or not the timer is held. In clock mode, when the timer is held (HLD = 0), TSTAT keeps its previous value and when HLD = 1, TSTAT toggles with a frequency of 1/2 of the CPU clock frequency.

2) Counter overflow: When the timer count register (CNT) is set to a value greater than the value of the timer period register (PRD), the counter reaches its maximum value (FFFF FFFFh), rolls over to 0, and continues.

3) Writing to registers of an active timer: Writes from the peripheral bus override register updates to CNT and new status updates to the timer control register (CTL).

4) Small timer period values in pulse mode: Note that small periods in pulse mode can cause TSTAT to remain high. This condition occurs when TIMER PERIOD ≤ PWID + 1.

7 Timer Interrupts

The TSTAT signal directly drives the CPU interrupt, as well as a DMA synchronization event. The frequency of the interrupt is the same as the frequency of TSTAT.

8 Timer Pins as General-Purpose Input/Output

Upon device reset, the timer pins TINP and TOUT are general-purpose input and output (I/O) pins, respectively. By configuring the timer control register (CTL), the TINP and TOUT pins can operate as general-purpose pins even when the timer is running.

The TINP pin is always a general-purpose input pin, if the timer is not running. If the timer is running, the TINP pin is a general-purpose input pin if CLKSRC = 1 in CTL, which indicates that an internal clock source is used instead of the TINP pin. When TINP is a general-purpose input pin, the input value is readable from the DATIN bit in CTL.

The TOUT pin is a general-purpose output pin if FUNC = 0 in CTL, independent of timer operation. The FUNC bit, as shown in Figure 1, selects either the DATOUT or the TSTAT value to be driven on the TOUT pin.

9 Emulation Operation

During debug using the emulator, the CPU may be halted on an execute packet boundary for single stepping, benchmarking, profiling, or other debug uses. For C620x/C670x DSP, during an emulation halt the timer halts when the CPU clock/4 is selected as the clock source (CLKSRC = 1). Here, the counter is only enabled to count during those cycles when the CPU is not stalled due to the emulation halt. Thus, counting is reenabled during single-step operation. If CLKSRC = 0, the timer continues counting as programmed. For C621x/C671x/C64x DSP, the timer continues counting during emulation halt regardless of clock source.

10 Timer Registers

Table 4 describes the three registers that configure timer operation.

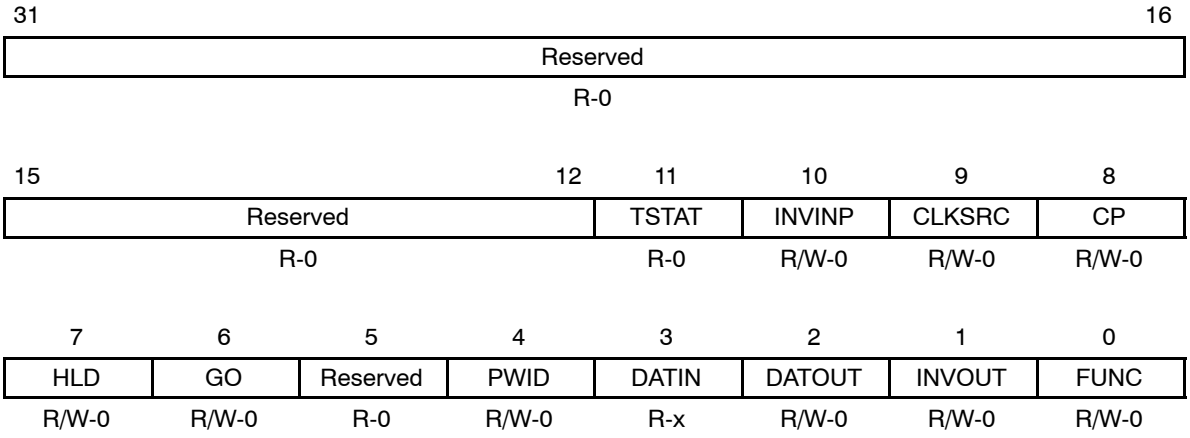
Table 4. Timer Registers

Acronym	Register Name	Section
CTL	Timer Control Register	10.1
PRD	Timer Period Register	10.2
CNT	Timer Count Register	10.3

10.1 Timer Control Register (CTL)

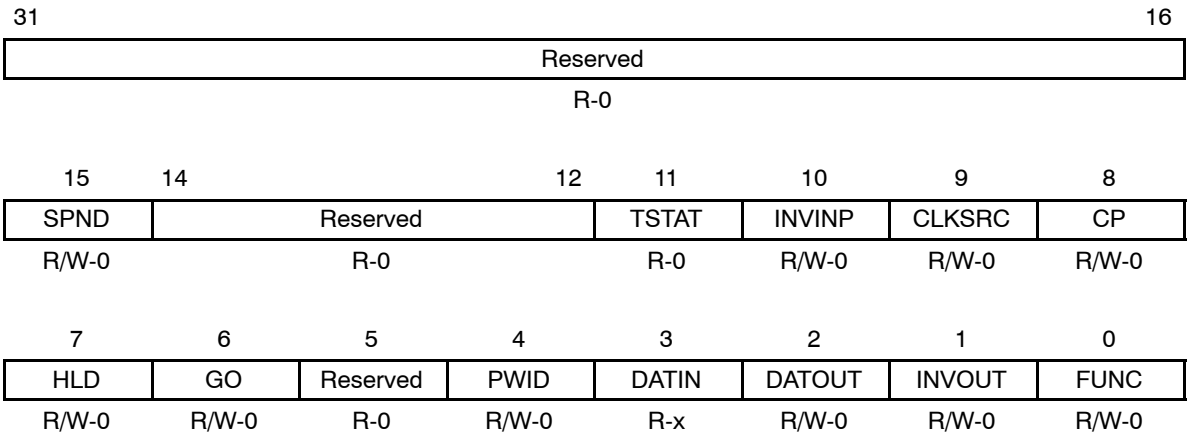
The timer control register (CTL) determines the operating mode of the timer, monitors the timer status, and controls the function of the TOUT pin. The timer control register is shown in Figure 4, for the C62x/C67x DSP, and in Figure 5, for the C64x DSP, and described in Table 5.

Figure 4. TMS320C62x/C67x Timer Control Register (CTL)



Legend: R = Read only; R/W = Read/Write; -*n* = value after reset; -x = value is indeterminate after reset

Figure 5. TMS320C64x Timer Control Register (CTL)



Legend: R = Read only; R/W = Read/Write; -*n* = value after reset; -x = value is indeterminate after reset

Table 5. Timer Control Register (CTL) Field Descriptions

Bit	*field*†	*symval*†	Value	Description
31–16	Reserved	–	0	Reserved. The reserved bit location is always read as 0. A value written to this field has no effect.
15	SPND‡			Suspend mode bit. Stops timer from counting during an emulation halt. Only affects operation if the clock source is internal, CLKSRC = 1. Reads always return a 0.
			0	Timer continues counting during an emulation halt.
			1	Timer stops counting during an emulation halt.
14–12	Reserved	–	0	Reserved. The reserved bit location is always read as 0. A value written to this field has no effect.
11	TSTAT			Timer status bit. Value of timer output.
		0	0	
		1	1	
10	INVINP			TINP inverter control bit. Only affects operation if CLKSRC = 0.
		NO	0	Noninverted TINP drives timer.
		YES	1	Inverted TINP drives timer.
9	CLKSRC			Timer input clock source bit.
		EXTERNAL	0	External clock source drives the TINP pin.
		CUPOVR4	1	Internal clock source. For C62x/C67x DSP: CPU clock/4 For C64x DSP: CPU clock/8
8	CP			Clock/pulse mode enable bit.
		PULSE	0	Pulse mode. TSTAT is active one CPU clock after the timer reaches the timer period. PWID determines when it goes inactive.
		CLOCK	1	Clock mode. TSTAT has a 50% duty cycle with each high and low period one countdown period wide.

† For CSL implementation, use the notation TIMER_CTL_*field_symval*
‡ For C64x DSP only; for C621x/C671x DSP, this bit is reserved.

Table 5. Timer Control Register (CTL) Field Descriptions (Continued)

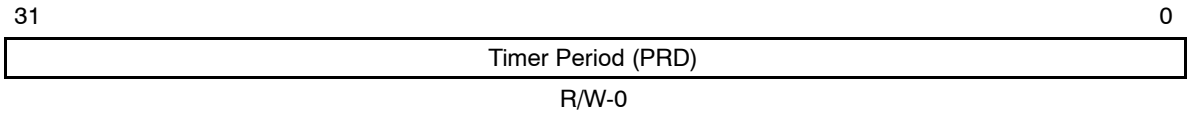
Bit	*field*†	*symval*†	Value	Description
7	HLD			Hold bit. Counter may be read or written regardless of HLD value.
		YES	0	Counter is disabled and held in the current state.
		NO	1	Counter is allowed to count.
6	GO			GO bit. Resets and starts the timer counter.
		NO	0	No effect on the timers.
		YES	1	If HLD = 1, the counter register is zeroed and begins counting on the next clock.
5	Reserved	–	0	Reserved. The reserved bit location is always read as 0. A value written to this field has no effect.
4	PWID			Pulse width bit. Only used in pulse mode (CP = 0).
		ONE	0	TSTAT goes inactive one timer input clock cycle after the timer counter value equals the timer period value.
		TWO	1	TSTAT goes inactive two timer input clock cycles after the timer counter value equals the timer period value.
3	DATIN			Data in bit. Value on TINP pin.
		0	0	Value on TINP pin is logic low.
		1	1	Value on TINP pin is logic high.
2	DATOUT			Data output bit.
		0	0	DATOUT is driven on TOUT.
		1	1	TSTAT is driven on TOUT after inversion by INVOUT.
1	INVOUT			TOUT inverter control bit (used only if FUNC = 1).
		NO	0	Noninverted TSTAT drives TOUT.
		YES	1	Inverted TSTAT drives TOUT.
0	FUNC			Function of TOUT pin.
		GPIO	0	TOUT is a general-purpose output pin.
		TOUT	1	TOUT is a timer output pin.

† For CSL implementation, use the notation TIMER_CTL_*field_symval*
‡ For C64x DSP only; for C621x/C671x DSP, this bit is reserved.

10.2 Timer Period Register (PRD)

The timer period register (PRD) contains the number of timer input clock cycles to count. This number controls the TSTAT signal frequency. The timer period register is shown in Figure 6 and described in Table 6.

Figure 6. Timer Period Register (PRD)



Legend: R/W = Read/Write; *-n* = value after reset

Table 6. Timer Period Register (PRD) Field Descriptions

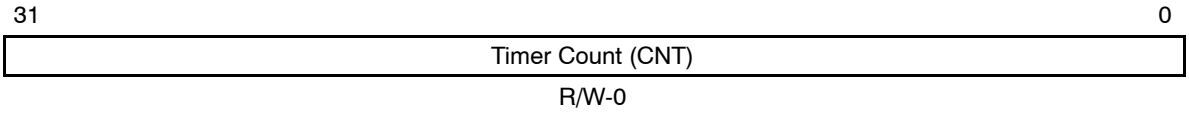
Bit	Field	*symval*†	Value	Description
31–0	PRD	OF(*value*)	0–FFFF FFFFh	Period bits. This 32-bit value is the number of timer input clock cycles to count and is used to reload the timer count register (CNT). This number controls the frequency of the timer output status bit (TSTAT).

† For CSL implementation, use the notation TIMER_PRD_PRD_*symval*

10.3 Timer Count Register (CNT)

The timer count register (CNT) contains the current value of the incrementing counter The timer count register increments by 1 when it is enabled to count and resets to 0 on the next CPU clock after the value in the timer period register (PRD) is reached. The timer count register is shown in Figure 7 and described in Table 7.

Figure 7. Timer Count Register (CNT)



Legend: R/W = Read/Write; -*n* = value after reset

Table 7. Timer Count Register (CNT) Field Descriptions

Bit	Field	*symval*†	Value	Description
31–0	CNT	OF(*value*)	0–FFFF FFFFh	Main count bits. This 32-bit value is the current count of the main counter. This value is incremented by 1 every input clock cycle.

† For CSL implementation, use the notation TIMER_CNT_CNT_*symval*

Revision History

Table 8 lists the changes made since the previous version of this document.

Table 8. Document Revision History

Page	Additions/Modifications/Deletions
	This document has been reviewed for accuracy and there are no changes since the previous version (July 2003) of this document.

This page is intentionally left blank.

Index

B

block diagram 8

C

CLKSRC bit 14
clock source selection 10
CNT 18
CNT bits 18
control register boundary conditions 12
count events 7
counting 10
CP bit 14
CTL 14

D

DATIN bit 14
DATOUT bit 14

E

emulation operation 13
enabling counting 9

F

FUNC bit 14

G

general-purpose input/output 13
generate pulses 7
GO bit 14

H

HLD bit 14

I

interrupt the CPU 7
interrupts 12
INVINP bit 14
INVOUT bit 14

M

modes, pulse and clock 11

N

notational conventions 3

O

overview 7

P

PRD 17
PRD bits 17
pulse generation 10
PWID bit 14

R

registers
  boundary conditions 12
  timer 13
  timer control register (CTL) 14
  timer count register (CNT) 18
  timer period register (PRD) 17
related documentation from Texas Instruments 3
resetting the timer 9
revision history 19

S

selection of clock sources 10
send synchronization events to the DMA 7
SPND bit 14

T

time events 7
timer
  block diagram 8
  clock source selection 10
  counting 10
  emulation operation 13
  enabling counting 9
  general-purpose input/output 13
  interrupts 12
  overview 7
  pulse generation 10
  register boundary conditions 12
  registers 13
  resetting 9
timer control register (CTL) 14
timer count register (CNT) 18
timer period register (PRD) 17
trademarks 4
TSTAT bit 14
TSTAT parameters 11