TMS320VC5501/5502 DSP Timers Reference Guide

Literature Number: SPRU618B
April 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

Copyright © 2004, Texas Instruments Incorporated

Preface

Read This First

About This Manual

This document describes three programmable 64-bit timers in the TMS320VC5501 and TMS320VC5502 digital signal processors (DSPs) of the TMS320C55x™ (C55x™) DSP generation. The first two timers, Timer0 and Timer1, are used only as general-purpose (GP) timers. The third timer is used as either a GP timer or a watchdog timer (WDT).

A fourth 64-bit timer is used as a DSP/BIOS™ counter. However, this timer is not user programmable.

Notational Conventions

This document uses the following conventions.

- The two GP timers and the WDT each have a set of registers. The registers used in the general-purpose timer are prefixed by GPT; the registers used in the watchdog timer are prefixed by WDT. This document does not use the register prefix when discussing the registers in general terms. The DSP/BIOS counter does not have a register set.
- In most cases, hexadecimal numbers are shown with the suffix h. For example, the following number is a hexadecimal 40 (decimal 64):

  40h

  Similarly, binary numbers often are shown with the suffix b. For example, the following number is the decimal number 4 shown in binary form:

  0100b
- If a signal or pin is active low, it has an overbar. For example, the $\overline{\text{RESET}}$ signal is active low.

Related Documentation From Texas Instruments

The following documents describe the C55x 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.

TMS320VC5501 Fixed-Point Digital Signal Processor Data Manual (literature number SPRS206) describes the features of the TMS320VC5501 fixed-point DSP and provides signal descriptions, pinouts, electrical specifications, and timings for the device.

TMS320VC5502 Fixed-Point Digital Signal Processor Data Manual (literature number SPRS166) describes the features of the TMS320VC5502 fixed-point DSP and provides signal descriptions, pinouts, electrical specifications, and timings for the device.

TMS320C55x Technical Overview (literature number SPRU393) introduces the TMS320C55x DSPs, the latest generation of fixed-point DSPs in the TMS320C5000™ DSP platform. Like the previous generations, this processor is optimized for high performance and low-power operation. This book describes the CPU architecture, low-power enhancements, and embedded emulation features.

TMS320C55x DSP CPU Reference Guide (literature number SPRU371) describes the architecture, registers, and operation of the CPU for the TMS320C55x DSPs.

TMS320C55x DSP Peripherals Overview Reference Guide (literature number SPRU317) introduces the peripherals, interfaces, and related hardware that are available on TMS320C55x DSPs.

TMS320C55x DSP Algebraic Instruction Set Reference Guide (literature number SPRU375) describes the TMS320C55x DSP algebraic instructions individually. Also includes a summary of the instruction set, a list of the instruction opcodes, and a cross-reference to the mnemonic instruction set.

TMS320C55x DSP Mnemonic Instruction Set Reference Guide (literature number SPRU374) describes the TMS320C55x DSP mnemonic instructions individually. Also includes a summary of the instruction set, a list of the instruction opcodes, and a cross-reference to the algebraic instruction set.

TMS320C55x Optimizing C/C++ Compiler User's Guide (literature number SPRU281) describes the TMS320C55x C/C++ Compiler. This C/C++ compiler accepts ISO standard C and C++ source code and produces assembly language source code for TMS320C55x devices.

TMS320C55x Assembly Language Tools User's Guide (literature number SPRU280) describes the assembly language tools (assembler, linker, and other tools used to develop assembly language code), assembler directives, macros, common object file format, and symbolic debugging directives for TMS320C55x devices.

TMS320C55x DSP Programmer's Guide (literature number SPRU376) describes ways to optimize C and assembly code for the TMS320C55x DSPs and explains how to write code that uses special features and instructions of the DSPs.

Trademarks

TMS320C5000, TMS320C55x, C55x, and DSP/BIOS are trademarks of Texas Instruments.

Other trademarks are the property of their respective owners.

This page is intentionally left blank.

Contents

Figures

Tables

64-Bit Timers

This document provides an overview of three 64-bit timers in the TMS320VC5501 and the TMS320VC5502 digital signal processors (DSPs) of the TMS320C55x™ DSP generation. The first two timers, Timer0 and Timer1, are used only as general-purpose (GP) timers. The third timer is used as either a GP timer or a watchdog timer (WDT).

A fourth 64-bit timer is used as a DSP/BIOS™ counter. However, this timer is not user programmable.

1 Introduction to the Timers

The DSP contains three programmable timers. Two are general-purpose (GP) timers, Timer0 (TIM0) and Timer1 (TIM1). Each of the GP timers can be configured in one of three modes using the timer mode (TIMMODE) bits in global timer control register 1 (GCTL1): 64-bit timer, dual 32-bit timers chained, or dual 32-bit timers unchained. The two GP timers do not support the watchdog timer mode. At reset, the two GP timers are configured as 64-bit timers.

The third timer can be configured as either a GP timer or a watchdog timer using the TIMMODE bits in watchdog timer global timer control register 1 (WDTGCTL1) and the watchdog timer enable (WDEN) bit in watchdog timer control register 1 (WDTWCTL1). At reset, the third timer is configured as a 64-bit GP timer.

The input/output pin of Timer0 (TIM0) and Timer1 (TIM1) can be configured as an input or an output using the timer pin mode (TIM*n*_MODE) bit in the timer signal selection register (TSSR) of the device (see Figure 1). When TIM*n*_MODE = 0, TIM*n* pin is only an input and connected to TIN1; when TIM*n*_MODE = 1, TIM*n* pin is only an output and connected to TOUT1.

The input/output pin of the watchdog timer (WDTOUT) can be configured as an input or an output using the WDT pin mode (WDT_MODE) bit in TSSR of the C5501/5502 devices (see Figure 1). When WDT_MODE = 0, the WDTOUT pin is only an input and connected to TIN1; when WDT_MODE = 1, the WDTOUT pin is only an output and connected to TOUT1. Furthermore, the TSSR bits can be used to connect the watchdog timer interrupt (WDTINT) to $\overline{\text{NMI}}$, $\overline{\text{RESET}}$, or $\overline{\text{INT3}}$.

Figure 1. Timer Pin Selection Diagram

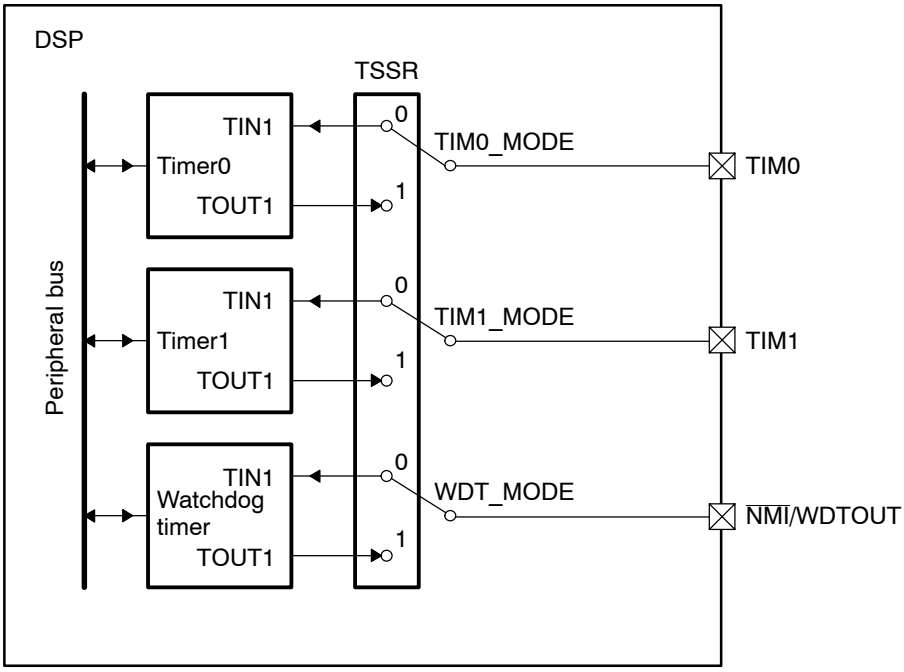
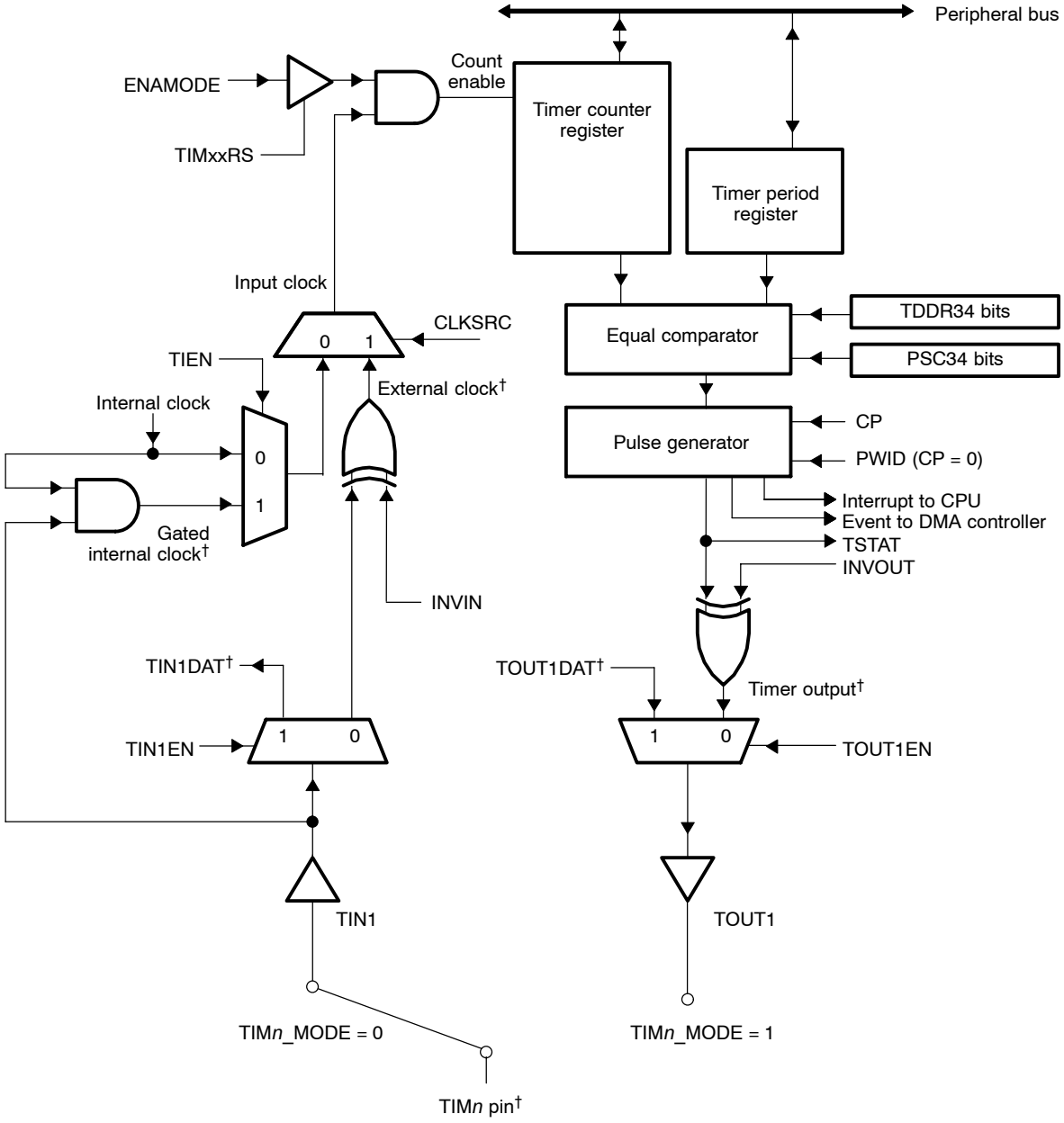


Figure 2 shows a high-level block diagram of the timer circuitry.

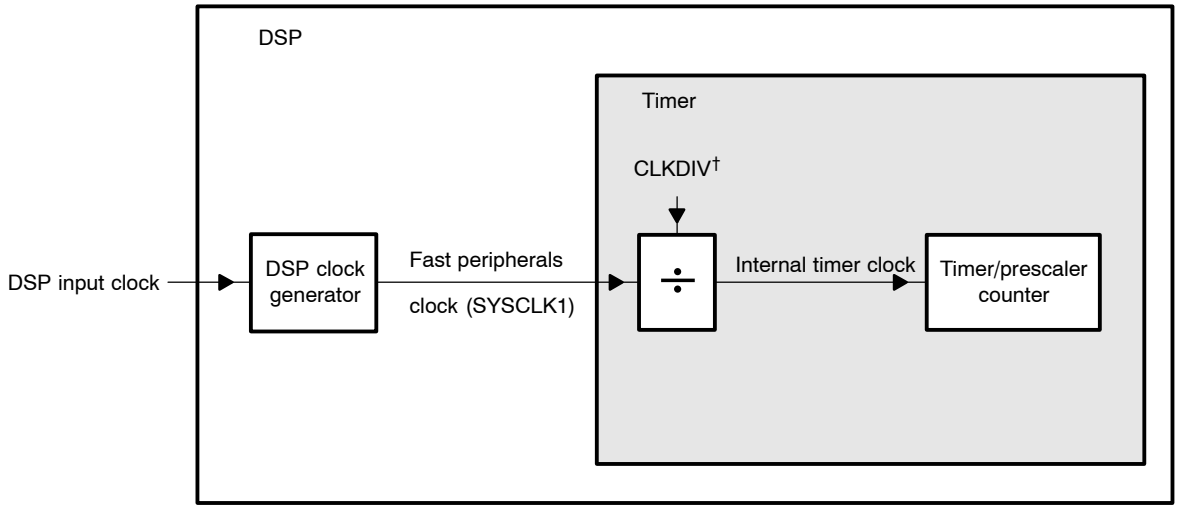
Figure 2. Timer Block Diagram



† The timer's one pin can be used for only one function at a time. The function may be one of the following: controlling a gated internal clock, supplying an external clock, acting as a general-purpose input tracked by the TIN1DAT bit, showing the timer output, or acting as a general-purpose output driven by the TOUT1DAT bit.

Each timer can be driven by an external clock at the timer pin or by an internal clock. When the internal clock is selected, the timer clock is generated as shown in Figure 3. The DSP clock generator receives a signal from a clock source as described in the device-specific data manual. One of the clocks produced by the DSP clock generator is the fast peripherals clock (SYSCLK1). SYSCLK1 is a divided-down version of the CPU clock, as described in the device-specific data manual A clock divider inside the timer divides SYSCLK1 by a preset divisor to produce the timer clock. To determine the preset divisor, read the CLKDIV bits in the timer clock speed register (CLK). On TMS320VC5501 and TMS320VC5502 devices, CLKDIV = 0, which means the timer clock has the same frequency as SYSCLK1.

Figure 3. Generation of the Internal Timer Clock



† On TMS320VC5501 and TMS320VC5502 devices, the CLKDIV field of the timer clock speed register (CLK) is fixed at 0. Therefore, the timer clock has the same frequency as the fast peripherals clock.

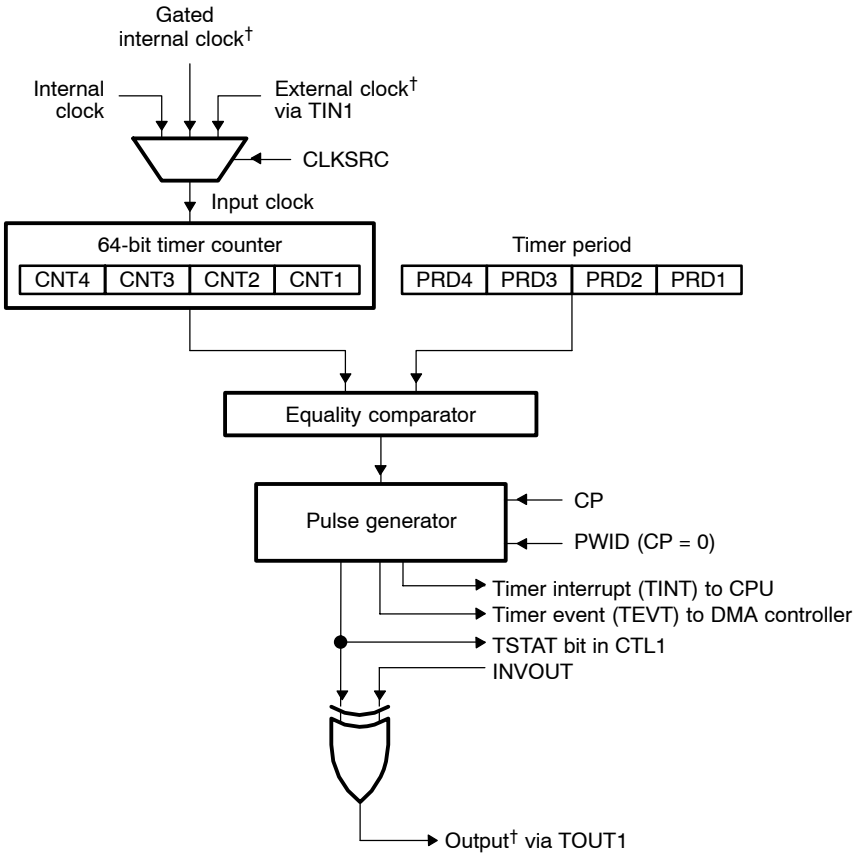
2 Timer Modes

2.1 64-Bit Timer Mode

The two GP timers and the WDT can each be configured as a 64-bit timer, using the the TIMMODE bits in GCTL1. At reset, the two GP timers and the WDT are in 64-bit timer mode.

In this mode, the timer operates as a 64-bit up-counter (Figure 4). All four counter registers (CNT1−CNT4) and all four period registers (PRD1−PRD4) form a 64-bit timer counter register and a 64-bit timer period register, respectively. When the timer is enabled (see section 3.3), the timer counter starts incrementing by 1 at every timer input clock cycle. When the timer counter matches the timer period, a maskable timer interrupt (TINT), a timer event (TEVT), and an output signal are generated. When in pulse mode (CP = 0), the timer output pin (TIM*n*) asserts a pulse that is 1, 2, 3, or 4 timer clock cycles wide, depending on the setting of the pulse width (PWID) bits in timer control register 1 (CTL1). When the timer is configured in continuous mode, the timer counter is reset to 0 on the cycle after the timer counter reaches the timer period. The timer can be stopped, restarted, reset, or disabled using CTL1 and global timer control register 1 (GCTL1).

Figure 4. 64-Bit Timer Mode Block Diagram



† Because the timer has only one pin, the gated internal clock, the external clock, and the timer output are mutually exclusive options. For example, if you use the pin for the timer output, you cannot use a gated internal clock or an external clock.

2.2 Dual 32-Bit Timer Modes

Each of the two GP timers and the WDT can be broken down into two 32-bit timers, using the the TIMMODE bits in GCTL1. In this mode, the two 32-bit timers can be operated independently (unchained mode) or in conjunction with each other (chained mode).

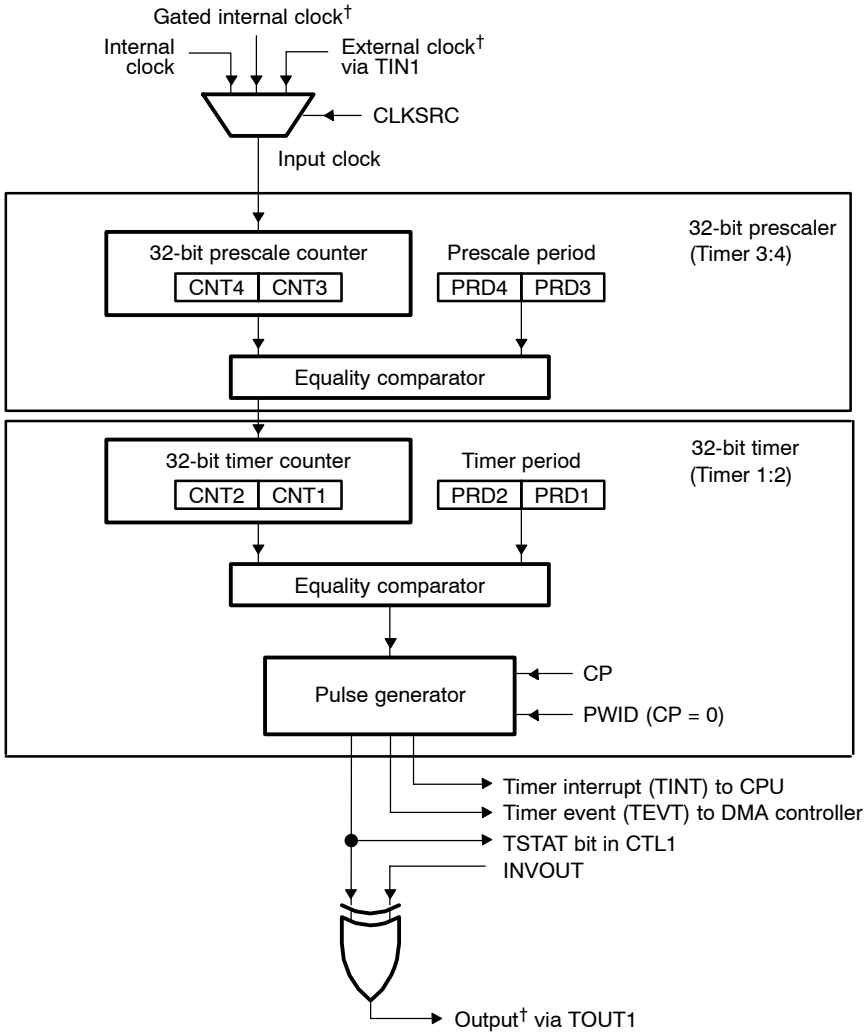
2.2.1 Chained Mode

In the chained mode (see Figure 5), one 32-bit timer (Timer 3:4) is used as a 32-bit prescaler and the other (Timer 1:2) is used as a 32-bit timer. The 32-bit prescaler is used to clock the 32-bit timer.

The 32-bit prescaler uses two counter registers (CNT4:CNT3) and two period registers (PRD4:PRD3) to form a 32-bit prescale counter register and a 32-bit prescale period register, respectively. When the timer is enabled, the prescale counter starts incrementing by 1 at every timer input clock cycle. One cycle after the prescale counter matches the prescale period, a clock signal is generated and the prescale counter register is reset to 0 (see the example in Figure 6).

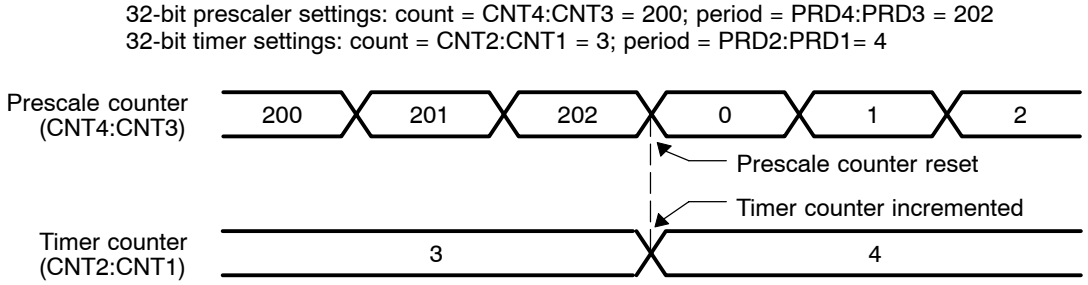
The other 32-bit timer uses two counter registers (CNT2:CNT1) and two period registers (PRD2:PRD1) to form a 32-bit timer counter register and a 32-bit timer period register, respectively. This timer is clocked by the output clock from the prescaler (see the example in Figure 6). The timer counter increments by 1 at every prescaler output clock cycle. When the timer counter matches the timer period, a maskable timer interrupt (TINT), a timer DMA event (TEVT), and an output signal are generated. When in pulse mode (CP = 0), the timer output (TOUT1) asserts a pulse that is 1, 2, 3, or 4 timer clock cycles wide, depending on the setting of the pulse width (PWID) bits in timer control register 1 (CTL1). When the timer is configured in continuous mode, the timer counter is reset to 0 on the cycle after the timer counter reaches the timer period. The timer can be stopped, restarted, reset, or disabled using CTL1 and global timer control register 1 (GCTL1). Timer control register 2 (CTL2) is not used in the chained mode.

Figure 5. Dual 32-Bit Timers Chained Mode Block Diagram



† Because the timer has only one pin, the gated internal clock, the external clock, and the timer output are mutually exclusive options. For example, if you use the pin for the timer output, you cannot use a gated internal clock or an external clock.

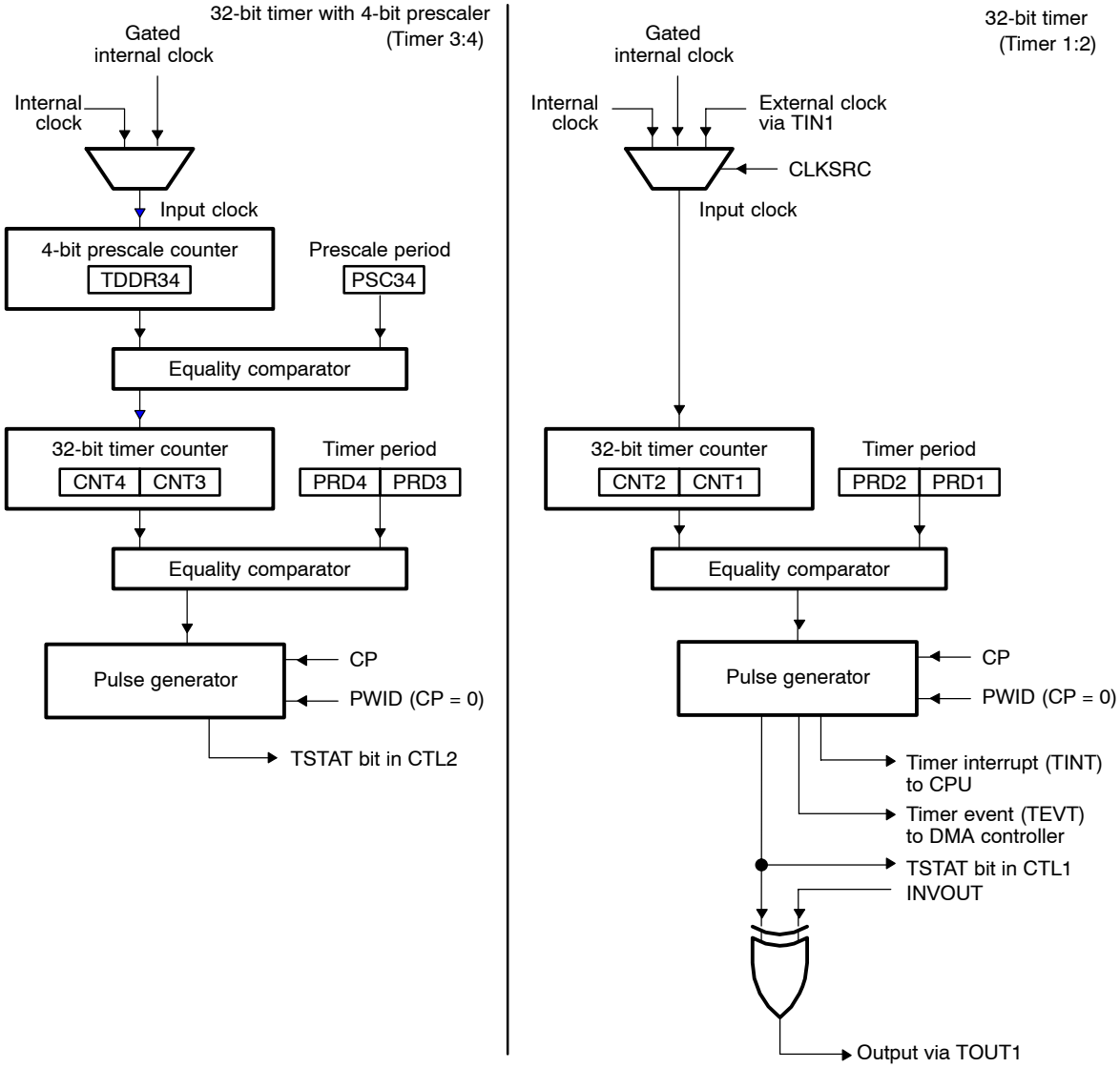
Figure 6. Dual 32-Bit Timers Chained Mode Example



2.2.2 Unchained Mode

In the unchained mode (Figure 7), the timer can operate as two independent 32-bit timers. One 32-bit timer (Timer 3:4) can be configured as a 32-bit timer being clocked by a 4-bit prescaler. The other (Timer 1:2) can be used as a 32-bit timer.

Figure 7. Dual 32-Bit Timers Unchained Mode Block Diagram

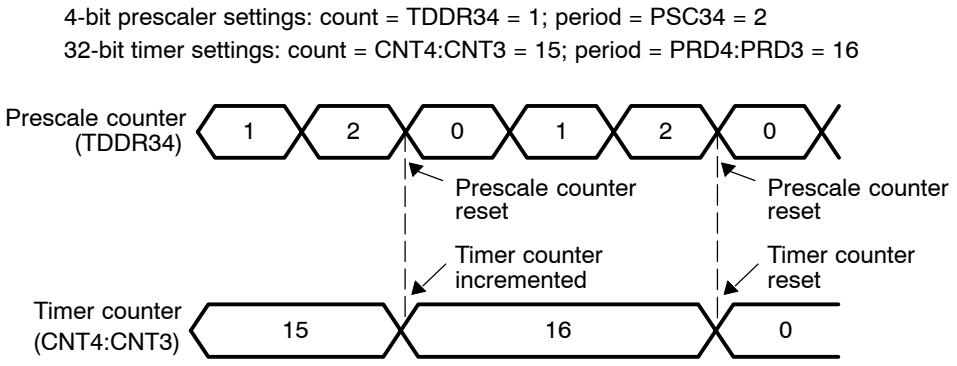


2.2.2.1 32-Bit Timer With a 4-Bit Prescaler

In the unchained mode, the 4-bit prescaler must be clocked by the internal clock; an external clock source cannot be used for timer 3:4. The 4-bit prescaler uses the timer divide-down ratio bits (TDDR34) and the prescale counter bits (PSC34) in timer control register 1 (CTL1) to form a 4-bit prescale counter register and a 4-bit prescale period register, respectively. When the timer is enabled, the prescale counter starts incrementing by 1 at every timer input clock cycle. One cycle after the prescale counter matches the prescale period, a clock signal is generated for the 32-bit timer.

The 32-bit timer uses two counter registers (CNT4:CNT3) and two period registers (PRD4:PRD3) to form a 32-bit timer counter register and a 32-bit timer period register, respectively. The 32-bit timer is clocked by the output clock from the 4-bit prescaler (see the example in Figure 8). The timer counter increments by 1 at every prescaler output clock cycle. When the timer counter matches the timer period, an output signal is generated; the state of the output signal is read in the timer status (TSTAT) bit of timer control register 2 (CTL2). When in pulse mode (CP = 0), TSTAT stays high or low for 1, 2, 3, or 4 timer clock cycles. The pulse width depends on the setting of the pulse width (PWID) bits in CTL2. When in clock mode (CP = 1), the TSTAT bit changes state (high-to-low or low-to-high) every time the timer counter matches the timer period. When the timer is configured in continuous mode, the timer counter is reset to 0 on the cycle after the timer counter reaches the timer period (see the example in Figure 8). The timer can be stopped, restarted, reset, or disabled using CTL2 and global timer control register 1 (GCTL1). Timer control register 1 (CTL1) has no control of Timer 3:4.

Figure 8. Dual 32-Bit Timers Unchained Mode Example



2.2.2.2 32-Bit Timer

The other 32-bit timer (Timer 1:2) uses two counter registers (CNT2:CNT1) and two period registers (PRD2:PRD1) to form a 32-bit timer counter register and a 32-bit timer period register, respectively. When the timer is enabled, the timer counter increments by 1 at every timer input clock cycle. When the timer counter matches the timer period, a maskable timer interrupt (TINT), a timer DMA event (TEVT), and an output signal are generated; the state of the output signal is also read in the timer status (TSTAT) bit of timer control register 1 (CTL1). When in pulse mode (CP = 0) and depending on the timer output inverter control (INVOUT) bit in CTL1, the timer output pin (TIM*n*) stays high or low for 1, 2, 3, or 4 timer clock cycles. The pulse width depends on the setting of the pulse width (PWID) bits in CTL1. When in clock mode (CP = 1), the timer output and the TSTAT bit change state (high-to-low or low-to-high) every time the timer counter matches the timer period. When the timer is configured in continuous mode, the timer counter is reset to 0 on the cycle after the timer counter reaches the timer period. The timer can be stopped, restarted, reset, or disabled using CTL1 and global timer control register 1 (GCTL1). Timer control register 2 (CTL2) has no control of Timer 1:2.

2.3 Counter and Period Registers Used in GP Timer Modes

Table 1 summarizes the counter registers (CNT*n*) and period registers (PRD*n*) used in each GP timer mode.

Table 1. Counter and Period Registers Used in GP Timer Modes

Timer Mode	Counter Registers	Period Registers
64-bit general-purpose	CNT4:CNT3:CNT2:CNT1	PRD4:PRD3:PRD2:PRD1
Dual 32-bit chained		
Prescaler (Timer 3:4)	CNT4:CNT3	PRD4:PRD3
Timer (Timer 1:2)	CNT2:CNT1	PRD2:PRD1
Dual 32-bit unchained		
Timer (Timer 1:2)	CNT2:CNT1	PRD2:PRD1
Timer with prescaler (Timer 3:4)	PSC34 bits and CNT4:CNT3	TDDR34 bits and PRD4:PRD3

3 Timer Operation

The following paragraphs describe the overall timer operation. For specific details on the watchdog timer operation, see section 4.

3.1 Timer Pin

As shown in Figure 1 on page 12, each timer has one pin that can be configured as an input or output using the corresponding pin mode bit (TIM*n*_MODE) in the timer signal selection register (TSSR) of the C5501/5502 devices. When the mode bit is 0, the pin is an input and is connected to TIN1; when the mode bit is 1, the pin is an output and is connected to TOUT1. As described in Table 2, the input clock source and the output signal on the pin is affected by other bits in timer control register 1 (CTL1).

Table 2. Configuring the Timer Pin

TIM*n*_MODE Bit	Timer Pin Function
0	Timer input. Clock source can be internal or external as selected by the CLKSRC bit. Internal clock source can be gated and enabled by the TIEN bit.
1	Timer output. The signal on the pin changes each time the timer counter value matches the value in the timer period register. The signal polarity is selected by the INVOUT bit, and the signal toggles or pulses, depending on the CP bit. If pulsing is selected, the pulse width is defined by the PWID bits.

3.1.1 Timer Pin Used for Clock Source

When TIM*n*_MODE = 0, the timer pin is an input and can be used for gating the internal clock (Figure 9) or to provide an external clock source (Figure 10). To enable clock gating, make CLKSRC = 0 and TIEN = 1 in the appropriate timer control register (CTL*n*). To enable the external clock, make CLKSRC = 1 in CTL*n*.

When TIM*n*_MODE = 0, the timer pin cannot be used for the timer output. The only outputs are the timer interrupt to the CPU and the timer event to the DMA controller.

Figure 9. Timer Pin Used for Clock Gating

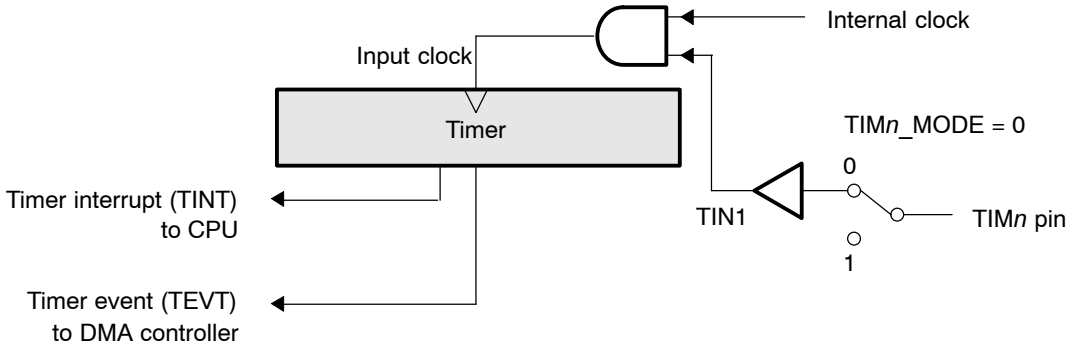
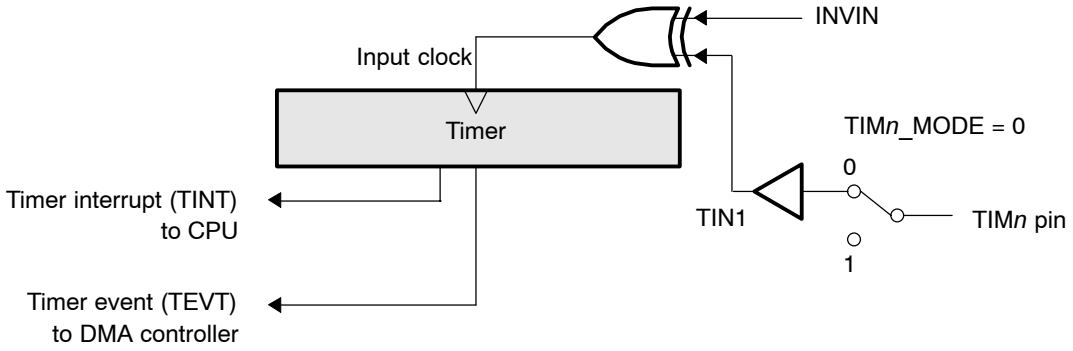


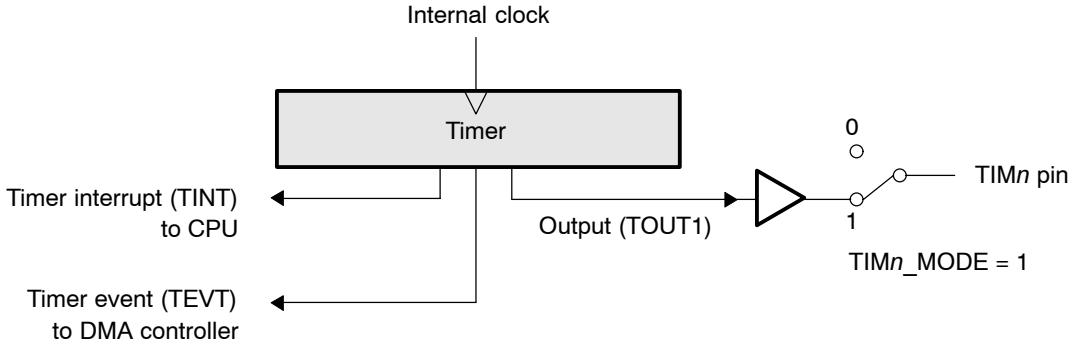
Figure 10. Timer Pin Used for External Clock Source



3.1.2 Timer Pin Used for Timer Output (TOUT1)

When TIM*n*_MODE = 1, the timer pin is used for the timer output (Figure 11); therefore, the timer pin cannot be used for clock gating or for an external clock source. The input clock must be the internal clock.

Figure 11. Timer Pin Used for Timer Output (TOUT1)



3.2 Timer Mode Selection

The timer can be configured as a 64-bit general-purpose timer or dual 32-bit timers (chained or unchained) using the timer mode (TIMMODE) bits in global timer control register 1 (GCTL1). See Table 3. At reset, the timer is configured as a 64-bit GP timer. Note that the two GP timers, Timer0 and Timer1, do not support the watchdog timer mode.

Table 3. Timer Mode Selection

TIMMODE bits		
Bit 3	**Bit 2**	**Timer mode**
0	0	64-bit general-purpose timer (default)
0	1	Dual 32-bit timers (unchained)
1	0	64-bit watchdog timer
1	1	Dual 32-bit timers (chained)

3.3 Timer Enabling

In the 64-bit timer mode or the dual 32-bit timers chained mode, the timer can be enabled by setting TIM12RS and TIM34RS bits in global timer control register 1 (GCTL1) to 1 and setting ENAMODE bits in timer control register 1 (CTL1) to 01b or 10b.

In the dual 32-bit timers unchained mode, the 32-bit timer can be enabled by setting TIM12RS bit in GCTL1 to 1 and ENAMODE bits in CTL1 to 01b or 10b. The 32-bit timer with prescaler can be enabled by setting TIM34RS bit in GCTL1 to 1 and ENAMODE bits in timer control register 2 (CTL2) to 01b or 10b.

Table 4 is a summary of timer enabling.

Table 4. Timer Enabling

	ENAMODE Bits						
	CTL1		CTL2		GCTL1		
Timer Mode	**Bit 7**	**Bit 6**	**Bit 7**	**Bit 6**	**TIM34RS**	**TIM12RS**	**Timer Status**
64-bit general-purpose	0	0	X	X	X	X	Disabled (default)
	0	1	X	X	1	1	Enabled one time
	1	0	X	X	1	1	Enabled continuously
Dual 32-bit chained	0	0	X	X	X	X	Disabled (default)
	0	1	X	X	1	1	Enabled one time
	1	0	X	X	1	1	Enabled continuously
Dual 32-bit unchained	0	0	0	0	X	X	Both timers disabled (default)
32-bit timer	0	1	X	X	X	1	32-bit timer enabled one time
	1	0	X	X	X	1	32-bit timer enabled continuously
32-bit timer with prescaler	X	X	0	1	1	X	32-bit timer enabled one time
	X	X	1	0	1	X	32-bit timer enabled continuously

3.4 Timer Clock Source Selection

As shown in Table 5 and Figure 12, the timer clock source is selected using the clock source (CLKSRC) bit and timer input enable (TIEN) bit in timer control register *n* (CTL*n*). Three clock sources are available to drive the timer clock:

- ❏ The internal clock, by setting CLKSRC = 0 and TIEN = 0.
- ❏ The internal clock gated by the timer input signal, by setting CLKSRC = 0 and TIEN = 1.
- ❏ The external clock on the timer input pin (TIM*n*), by setting CLKSRC = 1. This input signal is synchronized internally and can be inverted by setting the timer inverter control (INVIN) bit in CTL*n* to 1.

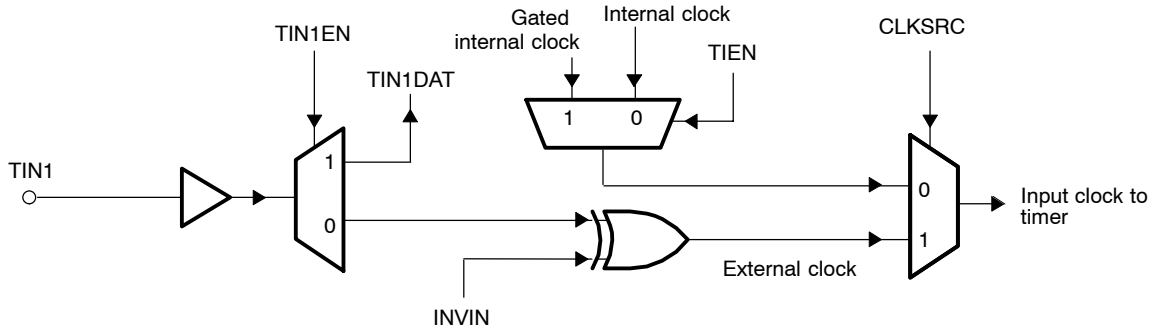
At reset, the clock source is the internal clock. The internal clock is derived from the DSP clock generator as shown in Figure 3 on page 14.

When the clock source is the gated internal clock, the timer starts counting when the timer input transitions from low to high and the timer stops counting when the timer input transitions from high to low.

Table 5. Timer Clock Source Selection

CLKSRC	TIEN	Input Clock
0	0	Internal clock (default)
0	1	Gated internal clock
1	X	External clock on timer input (TIN1)

Figure 12. Timer Clock Source Block Diagram



3.5 Timer Output Mode Selection

The two basic timer output modes are pulse mode and clock mode. The timer output mode is selected using the clock/pulse mode (CP) bit in timer control register *n* (CTL*n*). When in the pulse mode (CP = 0), the pulse width (PWID) bits set the pulse width to 1, 2, 3, or 4 timer clock cycles. This pulse can be inverted by setting the timer output inverter control (INVOUT) bit to 1.

When in the clock mode (CP = 1), the signal on the timer output pin has a 50% duty cycle. The signal toggles (from high to low or from low to high) each time the timer counter reaches the timer period.

3.6 Timer Counting

The timer counter runs at the timer clock rate specified by the clock source (CLKSRC) bits in timer control register *n* (CTL*n*). Counting is enabled by setting the enabling mode (ENAMODE) bits in CTL*n* to 01b or 10b. When enabled, the timer counter starts incrementing until the counter reaches a value equal to the value in the timer period register. Once the timer counter matches the timer period:

- ❑ if the timer is set to enable one time (ENAMODE = 01b), the timer counter is reset to 0 then stops.
- ❑ if the timer is set to enable continuously (ENAMODE = 10b), the timer counter is reset to 0 then continues counting.

Once the timer stops, if an external clock is used as the timer clock, the disable period must last at least one external clock period or the timer will not start counting again. When using the external clock, the count value is synchronized to the internal clock.

Note that when both the timer counter and timer period are cleared to 0, the timer can be enabled but the timer counter does not increment because the timer period is 0.

3.7 Timer Reset Sources

The timer has two reset sources: hardware reset and the timer reset bits (TIM12RS and TIM34RS) in global timer control register 1 (GCTL1).

- ❑ When a hardware reset is asserted, all the registers are set to their default values and the TIM*n* pins are configured as inputs.
- ❑ When TIM12RS is cleared to 0, TSTAT in CTL1 is reset to 0 and TOUT1 is in the high-impedance state.
- ❑ When TIM34RS is cleared to 0, TSTAT in CTL2 is reset to 0.

3.8 Timer Interrupt Rate

To receive periodic interrupts, configure the timer to run in the continuous mode (ENAMODE = 10b). Each time the timer finishes counting, it can generate a timer interrupt for the CPU and a timer event for the DMA controller. The rate at which this occurs (the timer interrupt rate) depends on whether the timer has a prescaler.

If the timer does not have a prescaler, there is only one counter. When the timer counter reaches the programmed timer period, the timer generates an interrupt and a DMA event. Because the timer is in the continuous mode, one cycle after the timer counter matches the timer period, the timer counter is reset to 0 and starts counting again. The timer interrupt rate is:

$$TINTrate = \frac{Timer\ input\ clock\ rate}{Programmed\ timer\ period + 1}$$

If a timer has prescaler, there are two counters. One cycle after the prescale counter reaches the programmed prescale period, the timer counter is incremented by 1, and the prescale counter is reset to start counting again. If the prescaler continues long enough, it increments the timer counter to the programmed timer period. At that time, the timer generates an interrupt and a DMA event. One cycle later (assuming the continuous mode), the timer counter is reset to 0 and starts counting again. The timer interrupt rate in this case is:

$$TINTrate = \frac{Timer\ input\ clock\ rate}{(Programmed\ prescale\ period + 1)\ (Programmed\ timer\ period + 1)}$$

3.9 Timer Operation in Idle Mode

When a timer is not in the watchdog timer mode, the timer can be placed in the idle mode by:

1) Setting the corresponding (TIM0, TIM1, or WDT) bit in the peripheral idle control register (PICR) of the C5501/5502 device to 1.

2) Setting the PERI bit in the idle configuration register (ICR) of the C5501/5502 device to 1.

3) Executing an IDLE instruction.

Once the idle mode is enabled, the timer immediately enters the idle mode and stops counting when the CPU executes an IDLE instruction.

In the watchdog timer mode, the requirements are different. See section 4.6 for the watchdog idle procedure.

3.10 Timer Emulation Modes

Each timer has an emulation management register (EMU). As shown in Table 6, the FREE and SOFT bits of EMU determine how the timer responds to an *emulation suspend event*. An emulation suspend event corresponds to any type of emulator access to the DSP, such as a hardware or software breakpoint, a probepoint, or a printf instruction.

Table 6. Timer Emulation Modes Selection

FREE	SOFT	Emulation Mode
0	0	The timer stops immediately.
0	1	The timer stops when the timer counter value increments and reaches the value in the timer period register.
1	X	The timer runs free regardless of SOFT bit status.

When using an internal clock as the timer clock source, the timer counter increments properly when single stepping. For example, the timer increments by one for each single step if the timer clock is equal to the CPU clock; or increments by one for every four single steps if the timer clock is equal to one-fourth of the CPU clock.

3.11 Timer Operation in the GPIO Mode

Upon a hardware reset, the timer input/output pin (TIM*n*) is not configured for general-purpose I/O (GPIO). The timer input/output can operate as GPIO if configured for GPIO. When TIM*n*_MODE = 0 in the TSSR, the TIM*n* pin is only a general-purpose input and connected to TIN1; when TIM*n*_MODE = 1, the TIM*n* pin is only a general-purpose output and connected to TOUT1 (see Figure 1 on page 12).

GPIO functionality and timer functionality cannot be used at the same time. GPIO functionality can be enabled in GPEN only when the timer is reset (TIM12RS = 0 and TIM34RS = 0 in GCTL1). While GPIO functionality is enabled, the timer cannot be taken out of reset.

The TSTAT bit in CTL1 is not modified during GPIO activity.

3.11.1 Timer Pin Used as a General-Purpose Output

To use the timer pin (TIM*n*) configured as a general-purpose output:

1) Set TIM*n*_MODE = 1 in TSSR of the C5501/5502 device (TIM*n* connected to TOUT1).
2) Set TOUT1EN = 1 in GPEN (TOUT1 enabled for GPIO).
3) Set TOUT1DIR = 1 in GPDIR (TOUT1 functions as a general-purpose output).
4) Write to TOUT1DAT. The general-purpose output (TOUT1) is driven to the logic level that is written to and stored in TOUT1DAT.

3.11.2 Timer Pin Used as a General-Purpose Input

To use the timer pin (TIM*n*) configured as a general-purpose input:

1) Set TIM*n*_MODE = 0 in TSSR of the C5501/5502 device (TIM*n* connected to TIN1).
2) Set TIN1EN = 1 in GPEN (TIN1 enabled for GPIO).
3) Set TIN1DIR = 0 in GPDIR (TIN1 functions as general-purpose input).
4) Read TIN1DAT. The logic value read in TIN1DAT is equal to the logic level present at the general-purpose input (TIN1). The general-purpose input is sampled with the internal clock.

3.11.3 Timer Pin Used as Input to Source the Interrupt

To use the timer pin (TIM*n*) configured as a general-purpose input to source the interrupt:

1) Set TIM*n*_MODE = 0 in TSSR of the C5501/5502 device (TIM*n* connected to TIN1).
2) Set TIN1EN = 1 in GPEN (TIN1 enabled for GPIO).
3) Set TIN1INT = 1 in GPINT (TIN1 sources timer interrupt or event).
4) Set TIN1INV in GPINT accordingly (invert enable bit).
5) Set TIN1DIR = 0 in GPDIR (TIN1 functions as general-purpose input).

3.12 Timer Operation Boundary Conditions

The following boundary conditions affect the timer operation.

3.12.1 Writing to and Reading From the Reserved Registers

Write the reset value to the reserved registers. Reading from the reserved registers returns zeros.

3.12.2 Timer Count = 0 and Timer Period = 0 (No Prescaler)

Consider a timer that has no prescaler:

- A 64-bit timer, or
- Timer(1:2) in the 32-bit dual timers unchained mode.

In the special case when timer count = 0 and timer period = 0:

- After a hardware reset and before the timer starts counting (ENAMODE = 00b), the timer output signal is held low.
- Once the timer is enabled, its behavior depends on the selected enabling mode (ENAMODE = 01b or 10b in the timer control register) and the selected timer output mode (CP = 0 or 1 in the timer control register). The options are summarized in Table 7.
- A timer interrupt (TINT) is not generated.

Table 7. Timer Operation When Timer Count = 0 and Timer Period = 0

Timer Enabling Mode	Timer Output Mode	Timer Operation When Timer Count = 0 and Timer Period = 0 (No Prescaler)
One-time mode (ENAMODE = 01b)	Pulse mode (CP = 0)	The timer output pulses once at the first timer clock cycle, and the timer stops counting at the next timer clock cycle. The pulse width is defined by the PWID bits of the timer control register.
	Clock mode (CP = 1)	The timer output toggles once at the first timer clock cycle. The timer stops counting at the next timer clock cycle.
Continuous mode (ENAMODE = 10b)	Pulse mode (CP = 0)	The timer output pulses once at the first timer clock cycle, and the timer continues to count up. Whenever the timer counter reaches its maximum value, it rolls around to 0 (see section 3.12.4), generating another pulse. The pulse width is defined by the PWID bits of the timer control register.
	Clock mode (CP = 1)	The timer output toggles once at the first timer clock cycle and then toggles with a frequency of half the timer clock frequency as the timer continues to count.

3.12.3 Timer Count = 0, Timer Period = 0, Prescale Count = 0, and Prescale Period = 0

Consider a timer that has a prescaler:

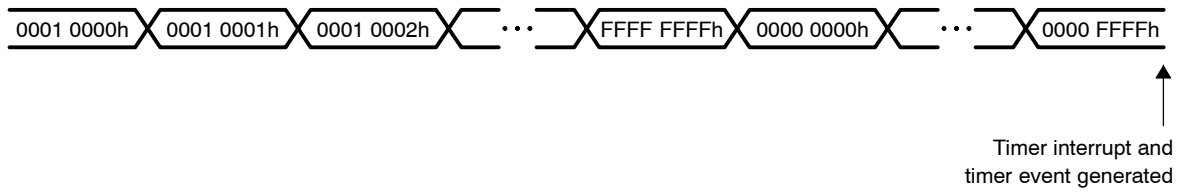
- The combination timer in the 32-bit dual timers chained mode.
- Timer(3:4) in the 32-bit dual timers unchained mode.

In the special case when timer count = 0, timer period = 0, prescale count = 0, and prescale period = 0, the timer operates in the same manner as a non-prescaled timer with timer count = 0 and timer period = 0 (see section 3.12.2).

3.12.4 Timer Counter Overflow

Timer counter overflow can happen when the timer counter register is set to a value greater than the value in the timer period register. The counter reaches its maximum value (FFFF FFFFh or FFFF FFFF FFFF FFFFh), rolls over to 0, and continues counting until it reaches the timer period. An example is in Figure 13.

Figure 13. 32-Bit Timer Counter Overflow Example



3.12.5 Writing to Registers of an Active Timer

Writes from the peripheral bus to the timer registers are not allowed when the timer is active, except for stopping or resetting the timers. In the 64-bit and dual 32-bit timer modes, registers that are protected by hardware are: CNT1, CNT2, CNT3, CNT4, PRD1, PRD2, PRD3, PRD4, GCTL1 (except the TIM12RS and TIM34RS bits), and CTL1 (except the ENAMODE bits).

3.12.6 Small Timer Period Value in Pulse Mode

Small timer periods in pulse mode (CP = 0) can cause TSTAT to remain high when ENAMODE is not 0. This condition can occur when PRD ≤ PWID + 1.

3.12.7 Reading the Counter Registers

Table 8 summarizes how to read the counter registers. When reading the timer count in 64-bit timer mode, the CPU must read the first 16-bit word in CNT1. When this occurs, the timer copies the other counter registers (CNT2, CNT3, and CNT4) into a set of shadow registers (the copying to the shadow registers starts when byte 0 is accessed). When reading the counter registers CNT2, CNT3, and CNT4, hardware logic forces the reads to read from the shadow registers CNT2S, CNT3S, and CNT4S.

When reading the timer count in a dual 32-bit timer mode, the CPU must read the first 16-bit word in CNT1 or CNT3. When this occurs, the timer copies CNT2 or CNT4 into a shadow register. CNT2 or CNT4 is read from the shadow register, CNT2S or CNT4S.

Table 8. Reading Counter Registers

Timer Mode	CPU	
64-bit timer	Read CNT1 –>	Read CNT1
		Copy CNT2 to CNT2S
		Copy CNT3 to CNT3S
		Copy CNT4 to CNT4S
	Read CNT2 –>	Read from CNT2S
	Read CNT3 –>	Read from CNT3S
	Read CNT4 –>	Read from CNT4S
32-bit timer	Read CNT1 –>	Read CNT1
		Copy CNT2 to CNT2S
	Read CNT2 –>	Read from CNT2S
	Read CNT3 –>	Read CNT3
		Copy CNT4 to CNT4S
	Read CNT4 –>	Read from CNT4S

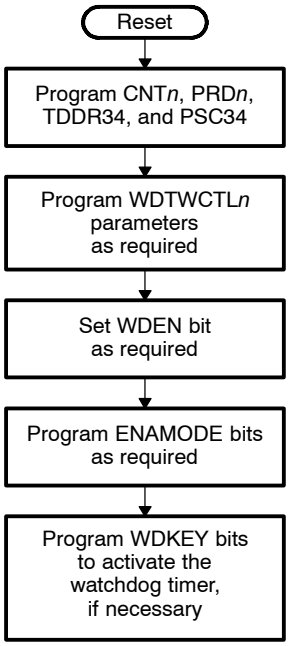
3.13 Initializing the Timer

After a hardware reset, the enabling mode (ENAMODE) bits in the timer control register (CTL*n*) are cleared to 0 and the timer is disabled. The timer counter and period registers are cleared to 0. The timer can be configured to the desired mode by programming the control registers, CTL*n* and (in the case of the watchdog timer mode) WDTWCTL*n*.

A typical way to initialize the timer is shown in Figure 14:

1) Write the timer counter and period values to CNT*n* and PRD*n*.
2) If the 4-bit prescaler is used, write the values to the TDDR34 and PSC34 bits.
3) Set the remaining control bits to the required state.
4) Set WDEN = 1 to use the timer as watchdog timer, if necessary.
5) To start the timer, set the ENAMODE bits to use the timer as a continuous interrupt generator (ENAMODE = 10b) or as a one-time counter (ENAMODE = 01b).
6) Program the WDKEY bits, if the watchdog timer mode is selected.

Figure 14. Timer Initialization



4 Watchdog Timer

The third timer can be configured as either a 64-bit watchdog timer or as a GP timer (64-bit mode, dual 32-bit chained mode, or dual 32-bit unchained mode). As a watchdog timer, it can be used to prevent system lockup when the software becomes trapped in loops with no controlled exit.

After a hardware reset, the timer is configured as a 64-bit GP timer and the watchdog mode is disabled. The timer then can be reconfigured as a watchdog timer using the timer mode (TIMMODE) bits in watchdog timer global timer control register 1 (WDTGCTL1) and the watchdog timer enable (WDEN) bit in watchdog timer control register 1 (WDTWCTL1). In the watchdog timer mode, the timer requires a special service sequence to be executed periodically. Without this periodic servicing, the timer counter increments until it matches the timer period and causes a watchdog timeout event.

Once the timer is configured as a watchdog timer, it cannot be reconfigured as a GP timer until a software reset or hardware reset. Whether this timer is used as a GP timer or as a watchdog timer, when the timer counter matches the timer period, the timer generates two signals: an output signal (described in section 4.1) and an interrupt signal (described in section 4.2). Typically, one or the other is used, depending on whether an external or internal trigger is desired.

4.1 Watchdog Timer Output Signal

The timer output signal can be ported to the timer output pin (WDTOUT), which in turn, can be externally connected to an interrupt pin. For example, in the watchdog timer mode, the timer output pin could be connected to the $\overline{\text{RESET}}$ pin, so that a watchdog timeout event resets the DSP. If the timer is acting as a GP timer, the timer output pin could be connected to the $\overline{\text{INT3}}$ pin or another maskable interrupt pin. In either case, the timer must be configured to generate an active-low pulse of the required width for the CPU to properly recognize the interrupt. The required pulse width depends on the minimum timing requirements described in the device-specific data manual.

The watchdog timer output pin is multiplexed with the $\overline{\text{NMI}}$ input pin; its function is controlled using the NMI/WDTOUT_CFG bit of the timer signal selection register (TSSR) in the C5501/5502 device. If the $\overline{\text{NMI}}$/WDTOUT pin is configured as a watchdog timer pin, it can be configured as an input pin or an output pin using the WDT_MODE bit of TSSR.

4.2 Watchdog Timer Interrupt Signal (WDTINT)

The watchdog timer interrupt signal (WDTINT) can be programmed to have no effect or to internally trigger one of three interrupts: $\overline{\text{INT3}}$ (maskable interrupt 3), $\overline{\text{RESET}}$ (hardware reset), or $\overline{\text{NMI}}$ (nonmaskable interrupt). This programming is done with the the IWCON bits, which are in the timer signal selection register (TSSR) of the C5501/5502 device. In the timer's default state, TINT is not connected to $\overline{\text{INT3}}$, $\overline{\text{RESET}}$, or $\overline{\text{NMI}}$ and, therefore, does not interrupt the CPU.

In the watchdog timer mode, WDTINT would likely be connected to $\overline{\text{RESET}}$ or $\overline{\text{NMI}}$, so that the CPU is interrupted immediately. If the timer is configured as a GP timer, WDTINT would likely be connected to $\overline{\text{INT3}}$.

Note:

If the $\overline{\text{NMI}}$/WDTOUT pin is configured for use as WDTOUT and the watchdog timer interrupt (WDTINT) is tied internally to $\overline{\text{NMI}}$, only WDTINT drives $\overline{\text{NMI}}$; any external source driving the $\overline{\text{NMI}}$/WDTOUT pin is ignored.

4.3 Watchdog Timer Mode Restrictions

The watchdog timer mode is selected and enabled when TIMMODE = 10b in WDTGCTL1 and WDEN = 1 in WDTWCTL1. This mode has the following restrictions:

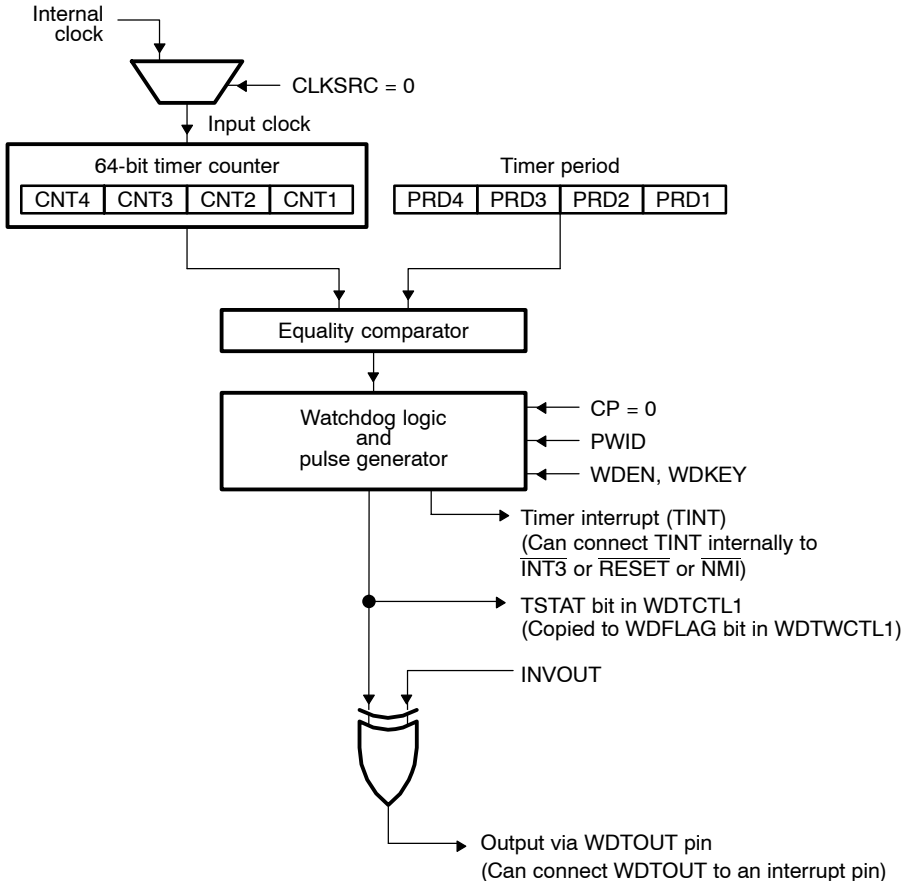
- ❏ No dual 32-bit timers mode
- ❏ No gated clock
- ❏ No external clock
- ❏ No one-time enabling
- ❏ No clock mode (only pulse mode)

4.4 Watchdog Timer Mode Operation

Figure 15 shows the timer when it is used in the watchdog timer mode. Note that in this mode, the timer clock must be set to internal clock (CLKSRC = 0). The CP bit is forced to 0 because the pulse mode is required for the watchdog timer output. All four counter registers (WDTCNT1–WDTCNT4) form a 64-bit timer counter register and all four period registers (WDTPRD1–WDTPRD4) form a 64-bit period register. When the timer counter matches the timer period, the timer generates a watchdog timeout event. This event:

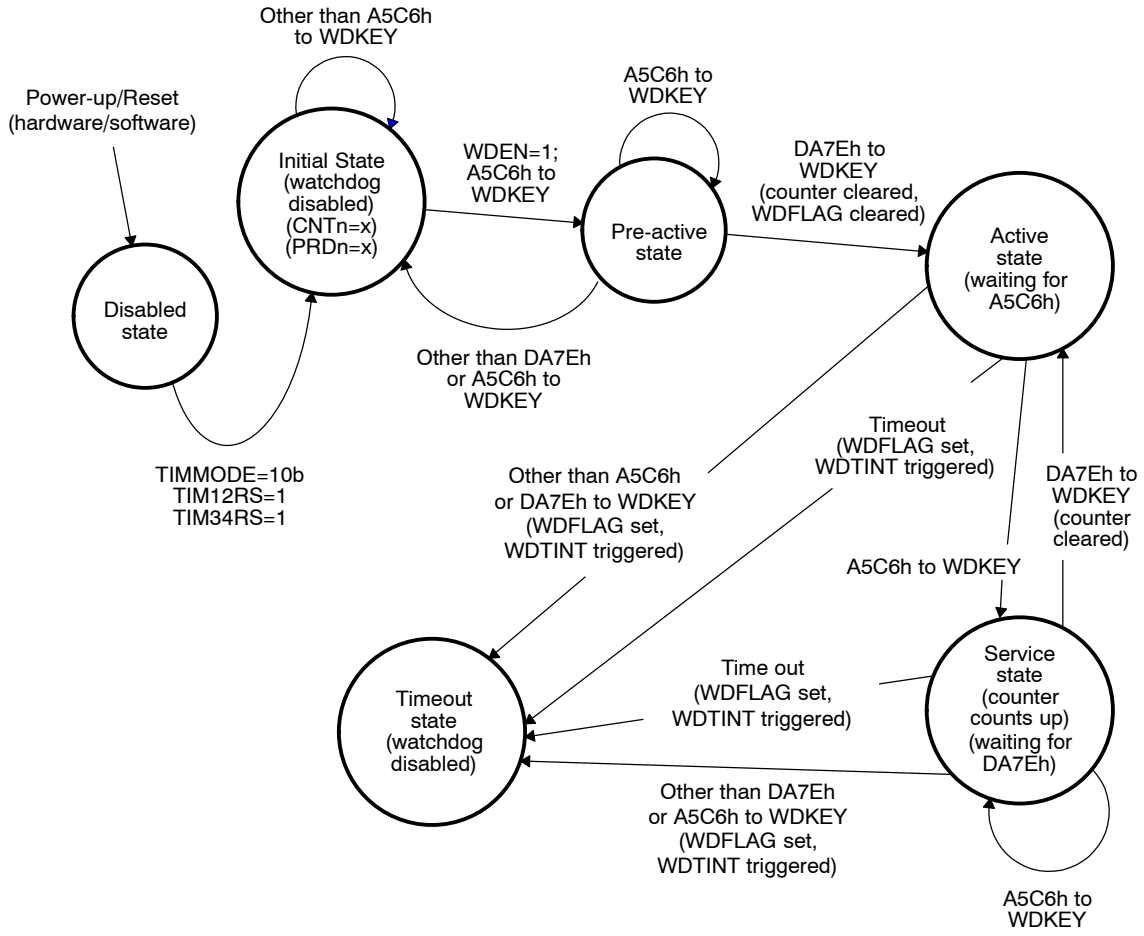
- ❏ Drives the watchdog timer output signal (WDTOUT) and/or the watchdog timer interrupt signal (WDTINT).
- ❏ Resets the timer counter to 0.
- ❏ Sets the TSTAT bit, which is copied to the WDFLAG bit of WDTWCTL1.

Figure 15. Watchdog Timer in the Watchdog Timer Mode



To activate the watchdog timer, a certain sequence of events must be followed, as shown in the state diagram of Figure 16.

Figure 16. Watchdog Timer Operation State Diagram



Once the watchdog timer is activated, it can be disabled only by a watchdog timeout event or by a hardware reset. A special key sequence is required to prevent the watchdog timer from being accidentally serviced while the software is trapped in a dead loop or by some other software failure.

To prevent a watchdog timeout event, the timer has to be serviced periodically (you could use another on-chip timer or an off-chip timer) by writing A5C6h followed by DA7Eh to the watchdog timer service key (WDKEY) bits of watchdog timer control register 2 (WDTWCTL2) before the timer finishes counting up. Both A5C6h and DA7Eh are allowed to be written to the WDKEY bits, but only the correct sequence of A5C6h followed by DA7Eh to the WDKEY bits services the watchdog timer. Any other writes to the WDKEY bits triggers the watchdog timeout event immediately.

A write of any value to WDTWCTL1 while the watchdog timer is active causes a watchdog timeout event. A read from WDTWCTL1 does not cause the event.

When the watchdog timer is in the Timeout state, the watchdog timer is disabled, the WDEN bit is cleared to 0, and the timer is reset. After entering the Timeout state, the watchdog timer cannot be enabled again until a hardware reset occurs.

After a hardware reset, the watchdog timer is disabled; however, reads or writes to the watchdog timer registers are allowed. Once the WDEN bit is set and A5C6h is written to the WDKEY bits, the watchdog timer enters the Pre-active state. In the Pre-active state:

- A write to WDTWCTL1 is allowed only when the write comes with the correct key (A5C6h or DA7Eh) to the WDKEY bits.
- A write of DA7Eh to the WDKEY bits when the WDEN bit set to 1 resets the counters and activates the watchdog timer.

The WDTWCTL*n*, WDTPRD*n*, and WDTCTL*n* registers must be configured before the watchdog timer enters the Active state. The WDEN bit must be set to 1 before writing DA7Eh to the WDKEY bits in the Pre-active state. Every time the watchdog timer is serviced by the correct WDKEY sequence, the watchdog timer counter is automatically reset.

Note:

Before the watchdog timer enters the Active state, the timer output signal is never asserted. Only the timer interrupt is asserted, when the timer finishes counting up. In this case, the timer interrupt can be used to:

- Indicate that the watchdog timer is not in the Active state
- Generate a periodic general-purpose interrupt, if the watchdog behavior is not desired at the time

4.5 Watchdog Timer Register Write Protection

Once the watchdog timer enters the Pre-active state (see Figure 16 on page 39), writes to WDTCNT1, WDTCNT2, WDTCNT3, WDTCNT4, WDTPRD1, WDTPRD2, WDTPRD3, and WDTPRD4, and WDTCTL1 are write protected. While the watchdog timer is in the Timeout state, writing to the WDEN bit has no effect.

However, any write to WDTWCTL1 while the watchdog timer is active causes a watchdog timeout event. Writing an incorrect value to the WDKEY bits of WDWTCTL2 also results in a watchdog timeout event.

Once the watchdog timer enters its Initial state (see Figure 16), do not write to WDTGCTL1.

4.6 Watchdog Timer Operation in Idle Mode

Once the watchdog timer is enabled, the timer cannot go into the idle mode unless the proper key sequence (01b followed by 10b) is written into the WDIKEY bits in WDTWCTL1.

To idle the timer in the watchdog timer mode:

1) Make WDT = 1 in the peripheral idle control register (PICR) of the C5501/5502 device.

2) Make PERI = 1 in the idle configuration register (ICR) of the C5501/5502 device.

3) Write the sequence of 01b followed by 10b to the watchdog idle enable key (WDIKEY) bits in WDTWCTL1. (A write of 00b to the WDIKEY bits disallows the watchdog timer from entering the idle mode.)

4) Execute an IDLE instruction. The timer immediately enters the idle mode and stops counting.

5 Timer Registers

Each of the three timers (TIM0, TIM1, and WDT) contains the set of registers in the shaded rows of Table 9. For a general-purpose timer (TIM0 or TIM1), these register names are prefixed by GPT and suffixed by the timer number (for example, EMU is GPTEMU_0 for TIM0). For the watchdog timer (WDT), the names are prefixed by WDT (for example, EMU is WDTEMU). The watchdog timer requires two additional control registers, WDTWCTL1 and WDTWCTL2, which are listed at the end of Table 9. Refer to the device-specific data manual for the addresses of all the timer registers.

Table 9. Timer Registers

Acronym	Name	Section
EMU	Emulation management register	5.1
CLK	Timer clock speed register	5.2
GPINT	GPIO interrupt control register	5.3
GPEN	GPIO enable register	5.4
GPDAT	GPIO data register	5.5
GPDIR	GPIO direction register	5.6
CNT1	Counter register 1	5.7
CNT2	Counter register 2	5.7
CNT3	Counter register 3	5.7
CNT4	Counter register 4	5.7
PRD1	Period register 1	5.8
PRD2	Period register 2	5.8
PRD3	Period register 3	5.8
PRD4	Period register 4	5.8
CTL1	Timer control register 1	5.9
CTL2	Timer control register 2	5.9
GCTL1	Global timer control register 1	5.10
WDTWCTL1	Watchdog timer control register 1	5.11
WDTWCTL2	Watchdog timer control register 2	5.12

5.1 Emulation Management Register (EMU)

The FREE and SOFT bits of EMU (see Figure 17 and Table 10) determine how the timer responds to an emulation suspend event. An emulation suspend event corresponds to any type of emulator access to the DSP, such as a hardware or software breakpoint, a probepoint, or a printf instruction. For additional emulation information, see section 3.10.

Figure 17. Emulation Management Register (EMU)

15	2	1	0
Reserved		SOFT	FREE
R/W-0		R-0	R-0

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

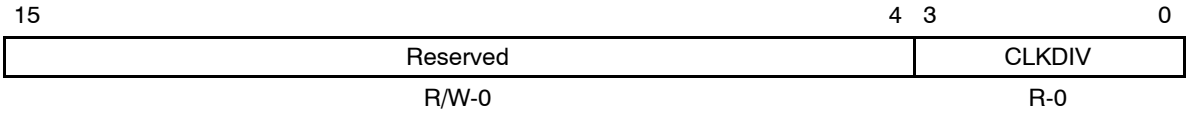
Table 10. Emulation Management Register (EMU) Bit Field Descriptions

Bit	Field	Value	Description
15–2	Reserved	0	Reserved. The reserved bit location is always read as 0. A value written to this field has no effect.
1	SOFT		Used in conjunction with FREE bit to determine how the timer responds to an emulation suspend event. When the FREE bit is 0, the SOFT bit selects the timer response.
		0	The timer stops immediately.
		1	The timer stops when the timer counter register increments and reaches the value in the timer period register.
0	FREE		Used in conjunction with SOFT bit to determine how the timer responds to an emulation suspend event. When the FREE bit is 0, the SOFT bit selects the timer response.
		0	The SOFT bit selects the timer response.
		1	The timer runs free, regardless of the value of the SOFT bit.

5.2 Timer Clock Speed Register (CLK)

The timer clock speed register (CLK) can be read to identify the internal clock frequency divider ratio for the timer. On the C5501/5502 device, the ratio is 1/1, meaning the internal timer clock (if used) is equal to the fast peripherals clock (SYSCLK1). Therefore, the CLKDIV field of CLK is always read as 0 on the C5501/5502 device.

Figure 18. Timer Clock Speed Register (CLK)



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

Table 11. Timer Clock Speed Register (CLK) Bit Field Descriptions

Bit	Field	Value	Description
15–4	Reserved	0	Reserved. The reserved bit location is always read as 0. A value written to this field has no effect.
3–0	CLKDIV		Clock divide-down ratio bits. Defines the internal clock frequency divider ratio for the timer. The CLKDIV bits are read-only bits.
		0h	Internal clock source for the timer is the fast peripherals clock (SYSCLK1).
		1h	Internal clock source for the timer is SYSCLK1 divided by1.
		2h	Internal clock source for the timer is SYSCLK1 divided by 2.
		3h	Reserved
		4h	Internal clock source for the timer is SYSCLK1 divided by 4.
		5–7h	Reserved
		8h	Internal clock source for the timer is SYSCLK1 divided by 8.
		9–15h	Reserved

5.3 GPIO Interrupt Control Register (GPINT)

The GPIO interrupt control register (GPINT) determines the interrupt mode of the timer. Use of this register's bits is described in section 3.11, *Timer Operation When Configured in the GPIO Mode*.

Figure 19. GPIO Interrupt Control Register (GPINT)

15	5	4	3	1	0
Reserved		TIN1INV	Reserved		TIN1INT
R/W-0		R/W-0	R/W-0		R/W-0

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

Table 12. GPIO Interrupt Control Register (GPINT) Bit Field Descriptions

Bit	Field	Value	Description
15–5	Reserved	0	Reserved. The reserved bit location is always read as 0. A value written to this field has no effect.
4	TIN1INV		Timer input 1 invert enable bit. Inverts the timer input 1 signal when it is used to source a timer interrupt or event (TIN1INT = 1). TIN1INV has no effect when TIN1INT = 0.
		0	Noninverted (rising) timer input 1 signal generates a timer interrupt or event.
		1	Inverted (falling) timer input 1 signal generates a timer interrupt or event.
3–1	Reserved	0	Reserved. The reserved bit location is always read as 0. A value written to this field has no effect.
0	TIN1INT		Timer input 1 interrupt enable bit. Enables the timer input 1 to source the timer interrupt or event in the GPIO mode.
		0	Timer sources the timer interrupt or event.
		1	Timer input 1 sources the timer interrupt or event.

5.4 GPIO Enable Register (GPEN)

The GPIO enable register (GPEN) determines the operating mode of the timer pin in the GPIO mode. Section 3.11, *Timer Operation When Configured in the GPIO Mode*, describes how to use this register's bits.

GPIO functionality and timer functionality cannot be used at the same time. GPIO functionality can be enabled in GPEN only when the timer is reset (TIM12RS = 0 and TIM34RS = 0 in GCTL1). While GPIO functionality is enabled, the timer cannot be taken out of reset.

Figure 20. GPIO Enable Register (GPEN)

15	2	1	0
Reserved		TOUT1EN	TIN1EN
R/W-0		R/W-0	R/W-0

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

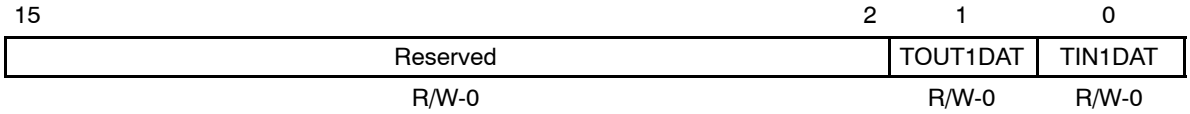
Table 13. GPIO Enable Register (GPEN) Bit Field Descriptions

Bit	Field	Value	Description
15–2	Reserved	0	Reserved. The reserved bit location is always read as 0. A value written to this field has no effect.
1	TOUT1EN		Timer output 1 enable bit in the GPIO mode.
		0	Timer output 1 (TOUT1) functions as a timer output.
		1	Timer output 1 (TOUT1) functions as GPIO. The TIM12RS and TIM34RS bits must both be 0 before 1 is written to TOUT1EN. The timer cannot be taken out of reset while TOUT1EN is 1.
0	TIN1EN		Timer input 1 enable bit in the GPIO mode.
		0	Timer input 1 (TIN1) functions as a timer input.
		1	Timer input 1 (TIN1) functions as GPIO. The TIM12RS and TIM34RS bits must both be 0 (timer reset) before 1 is written to TIN1EN. The timer cannot be taken out of reset while TIN1EN is 1.

5.5 GPIO Data Register (GPDAT)

The GPIO data register (GPDAT) determines the signal level on the timer pins in the GPIO mode. Use of this register's bits is described in section 3.11, *Timer Operation When Configured in the GPIO Mode*.

Figure 21. GPIO Data Register (GPDAT)



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

Table 14. GPIO Data Register (GPDAT) Bit Field Descriptions

Bit	Field	Value	Description
15–2	Reserved	0	Reserved. The reserved bit location is always read as 0. A value written to this field has no effect.
1	TOUT1DAT		Timer output 1 data bit in the GPIO mode. When timer output 1 is configured for GPIO (TOUT1EN = 1), use TOUT1DAT to control the signal level on the timer pin. When TOUT1DIR = 0, the timer pin functions as a general-purpose input. When TOUT1DIR = 1, timer output 1 functions as a general-purpose output. The general-purpose output is driven to the logic level that is written to and stored in TOUT1DAT.
		0	Drive the signal on the timer pin low.
		1	Drive the signal on the timer pin high.
0	TIN1DAT		Timer input 1 data bit in the GPIO mode. When timer input 1 is configured for GPIO (TIN1EN = 1), use TIN1DAT to control the signal level on the timer pin. When TIN1DIR = 0, timer pin input 1 functions as a general-purpose input. The logic value read in this bit is equal to the logic level present at the timer pin. Writes to this bit have no effect. The general-purpose input is sampled with the internal clock. When TIN1DIR = 1, timer pin functions as a general-purpose output.
		0	Drive the signal on the timer pin low.
		1	Drive the signal on the timer pin high.

5.6 GPIO Direction Register (GPDIR)

The GPIO direction register (GPDIR) determines the direction, input or output, of the timer pin in the GPIO mode. See section 3.11, *Timer Operation When Configured in the GPIO Mode*, for the details of how to use this register's bits.

Figure 22. GPIO Direction Register (GPDIR)

15	2	1	0
Reserved		TOUT1DIR	TIN1DIR
R/W-0		R/W-0	R/W-0

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

Table 15. GPIO Direction Register (GPDIR) Bit Field Descriptions

Bit	Field	Value	Description
15–2	Reserved	0	Reserved. The reserved bit location is always read as 0. A value written to this field has no effect.
1	TOUT1DIR		Timer output 1 direction bit in the GPIO mode. When timer output 1 is configured for GPIO (TOUT1EN = 1), use TOUT1DIR to control the direction of the timer pin. TOUT1DIR must be cleared to 0 for timer pin to be used as an input in the GPIO interrupt mode.
		0	The timer pin functions as a general-purpose input.
		1	The timer pin functions as a general-purpose output.
0	TIN1DIR		Timer input 1 direction bit in the GPIO mode. When timer input 1 is configured for GPIO (TIN1EN = 1), use TIN1DIR to control the direction of the timer pin. TIN1DIR must be cleared to 0 for the timer pin to be used as an input in the GPIO interrupt mode.
		0	The timer pin functions as a general-purpose input.
		1	The timer pin functions as a general-purpose output.

5.7 Counter Registers (CNT1–CNT4)

The timer counter register (Figure 23) is a 64-bit wide register consisting of four 16-bit counter registers: CNT1, CNT2, CNT3, and CNT4. Each of the four registers has the field shown in Figure 24 and is accessed via a separate address.

- In the general-purpose timer mode, the 64-bit timer counter increments when it is enabled to count. The timer counter register is cleared to 0 at reset.
- In the enabled continuous mode (ENAMODE = 10b), the 64-bit timer counter increments until the value in the timer counter matches the programmed timer period. The timer counter is reset to 0 and continues to increment on the next clock after the value in the timer counter matches the timer period.
- In a dual 32-bit timer mode, the 64-bit timer counter register is divided into two pairs of 16-bit registers: CNT2:CNT1 and CNT4:CNT3. These two register pairs can be configured as chained or unchained.

A hardware reset clears all four counter registers, but software resets do not affect them. When the TIM12RS bit is cleared, CNT2 and CNT1 keep their current values, and when the TIM34RS bit is cleared, CNT4 and CNT3 keep their current values.

Figure 23. 64-bit Timer Counter Register

63–48	47–32	31–16	15–0
CNT4	CNT3	CNT2	CNT1
R/W-0	R/W-0	R/W-0	R/W-0

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

Figure 24. Counter Register (CNTn)

15–0
CNT*n*
R/W-0

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

Table 16. Counter Register (CNTn) Bit Field Descriptions

Bit	Field	Value	Description
15–0	CNT*n*	0000h–FFFFh	Counter register. This register is one half of a 32-bit prescale counter or 32-bit timer counter, or one fourth of a 64-bit timer counter.

5.8 Period Registers (PRD1–PRD4)

The timer period register (Figure 25) is a 64-bit wide register consisting of four 16-bit period registers: PRD1, PRD2, PRD3, and PRD4. Each of the four registers has the field shown in Figure 26 and is accessed via a separate address.

In a 64-bit timer mode, all 64 period bits contain the number of timer input clock cycles to count. This number controls the frequency of the timer output.

In a 32-bit dual timer mode, the 64-bit timer period register is divided into two pairs of 16-bit registers: PRD2:PRD1 and PRD4:PRD3. These two register pairs are used in conjunction with the two counter-register pairs, CNT2:CNT1 and CNT4:CNT3.

Figure 25. 64-bit Timer Period Register

63 48	47 32	31 16	15 0
PRD4	PRD3	PRD2	PRD1
R/W-0	R/W-0	R/W-0	R/W-0

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

Figure 26. Period Register (PRDn)

15 0
PRD*n*
R/W-0

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

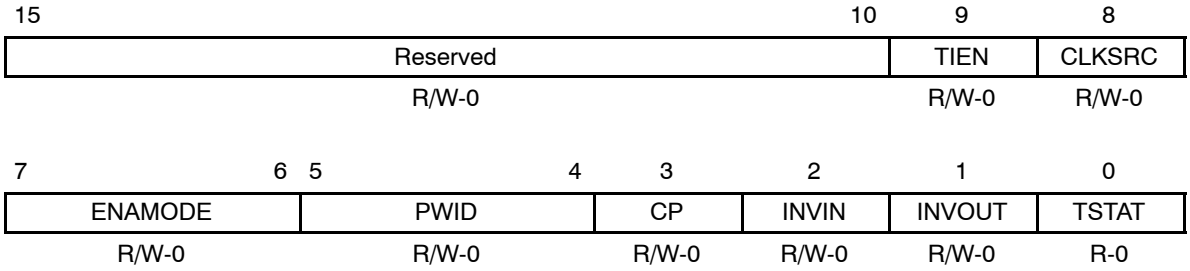
Table 17. Period Register (PRDn) Bit Field Descriptions

Bit	Field	Value	Description
15–0	PRD*n*	0000h–FFFFh	Period bits. This register contains one half or one fourth of the timer period.

5.9 Timer Control Registers (CTL1 and CTL2)

Each of the two timer control registers (CTL1 and CTL2) has the fields shown in Figure 27 and described in Table 18. The fields determine the operation of the timer, monitor the timer status (TSTAT), and control the function of the timer input/output. CTL2 is used only when the timer is configured in dual 32-bit timers unchained mode (TIMMODE = 01b in GCTL1).

Figure 27. Format of a Timer Control Register (CTLn)



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

Table 18. Timer Control Register (CTLn) Bit Field Descriptions

Bit	Field	Value	Description
15–10	Reserved		Reserved. The reserved bit location is always read as 0. A value written to this field has no effect.
9	TIEN		Timer input enable bit determines if the timer clock is gated by the timer input. Applicable only when CLKSRC = 0.
		0	Timer clock is not gated by the timer input.
		1	Timer clock is gated by a high state of the timer input synchronized with the internal clock. Timer starts counting when timer input transitions from low to high. Timer stops counting when timer input transitions from high to low.
8	CLKSRC		Clock source bit determines the clock source for the timer.
		0	The clock source is the internal clock.
		1	The clock source is the signal on the timer pin.

Table 18. Timer Control Register (CTLn) Bit Field Descriptions (Continued)

Bit	Field	Value	Description
7–6	ENAMODE		Enabling mode bits determine the timer mode.
		00b	The timer is disabled (not counting) and maintains the current value.
		01b	The timer is enabled one time. The timer stops after the timer counter reaches the timer period.
		10b	The timer is enabled continuously. The timer counter increments until it reaches the timer period. One timer clock cycle later, the timer counter is reset to 0 and continues counting.
		11b	Reserved
5–4	PWID		Pulse width bits. PWID is only used in pulse mode (CP = 0). PWID controls the width of the timer output signal. The polarity of the pulse is controlled by the INVOUT bit. The timer output signal is recorded in the TSTAT bit and can be made visible on the timer output pin.
		00b	The pulse width is 1 timer clock cycle.
		01b	The pulse width is 2 timer clock cycles.
		10b	The pulse width is 3 timer clock cycles.
		11b	The pulse width is 4 timer clock cycles.
3	CP		Clock/pulse mode bit for timer output. In the watchdog timer mode (TIMMODE = 10b), the pulse mode is selected automatically and the CP bit is a don't care.
		0	Pulse mode. When the timer counter reaches the timer period, the timer output appears as a pulse with the width defined by the PWID bits and the polarity defined by the INVOUT bits.
		1	Clock mode. The timer output signal has a 50% duty cycle signal. When the timer counter reaches the timer period, the level of the timer output signal is toggled (from high to low or from low to high).
2	INVIN		Timer input inverter control bit. Only affects operation if CLKSRC = 1.
		0	A noninverted timer input drives the timer.
		1	An inverted timer input drives the timer.
1	INVOUT		Timer output inverter control bit.
		0	The timer output is not inverted.
		1	The timer output is inverted.

Table 18. Timer Control Register (CTLn) Bit Field Descriptions (Continued)

Bit	Field	Value	Description
0	TSTAT		Timer status bit. This is a read-only bit that shows the value of the timer output. TSTAT drives the timer pin (TIM*n* or WDTOUT) when the pin is used as a timer output pin (not a GPIO pin) and may be inverted by setting INVOUT = 1. TSTAT is not modified when the timer is in the GPIO mode; instead, TSTAT continues the hold the value it held before the GPIO mode was entered.
		0	Timer output is low.
		1	Timer output is high.

5.10 Global Timer Control Register 1 (GCTL1)

The global timer control register (GCTL1) contains a field for selecting the operating mode of the timer (TIMMODE), timer reset bits (TIM34RS and TIM12RS), and counters for Timer 3:4 in the dual 32-bit timers unchained mode (TDDR34 and PSC34).

Figure 28. Global Timer Control Register 1 (GCTL1)

15 – 12	11 – 8
TDDR34	PSC34
R/W-0	R/W-0

7 – 4	3 – 2	1	0
Reserved	TIMMODE	TIM34RS	TIM12RS
R/W-0	R/W-0	R/W-0	R/W-0

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

Table 19. Global Timer Control Register 1 (GCTL1) Bit Field Descriptions

Bit	Field	Value	Description
15–12	TDDR34		Timer divide-down ratio bits. This field is the prescale counter for Timer 3:4 in the dual 32-bit timers unchained mode (TIMMODE = 01b). When the timer is enabled, TDDR34 increments every timer clock cycle. The timer counter (CNT4:CNT3) increments on the cycle after the TDDR34 matches the value of PSC34. TDDR34 resets to 0 and continues. If enabled one time, when CNT4:CNT3 matches the PRD4:PRD3, the timer stops; if the timer is enabled continuously, CNT4:CNT3 resets to 0 on the cycle after matching the PRD4:PRD3 and the timer continues counting. The default value is 0000b.

Table 19. Global Timer Control Register 1 (GCTL1) Bit Field Descriptions (Continued)

Bit	Field	Value	Description
11–8	PSC34		Prescale period bits. This field specifies the prescale period for Timer 3:4 in the dual 32-bit timers unchained mode (TIMMODE = 01b). The default value is 0000b.
7–4	Reserved		Reserved. The reserved bit location is always read as 0. A value written to this field has no effect.
3–2	TIMMODE		Timer mode bits determine the timer operating mode.
		00b	The timer is in the 64-bit general-purpose timer mode.
		01b	The timer is in the dual 32-bit timers unchained mode.
		10b	The timer is in the 64-bit watchdog timer mode.
		11b	The timer is in the dual 32-bit timers chained mode.
1	TIM34RS		Timer 3:4 reset bit. Note that in order for the timer to function properly in 64-bit general-purpose timer mode both the TIM34RS and TIM12RS bits must be set to 1. If the timer is in the watchdog timer Active state, changing this bit does not affect the timer.
		0	Timer 3:4 is in reset. The TSTAT bit of CTL2 is reset to 0. However, the counter registers CNT4 and CNT3 keep their current values.
		1	Timer 3:4 is not in reset. Timer 3:4 can be used as a 32-bit timer.
0	TIM12RS		Timer 1:2 reset bit. Note that in order for the timer to function properly in 64-bit general-purpose timer mode both the TIM34RS and TIM12RS bits must be set to 1. If the timer is in the watchdog timer Active state, changing this bit does not affect the timer.
		0	Timer 1:2 is in reset. The TSTAT bit of CTL1 is reset to 0, and the timer output signal is in the high-impedance state. However, the counter registers CNT2 and CNT1 keep their current values.
		1	Timer 1:2 is not in reset. Timer 1:2 can be used as a 32-bit timer.

5.11 Watchdog Timer Control Register 1 (WDTWCTL1)

The watchdog timer control register 1 (WDTWCTL1) determines the state of the watchdog timer and monitors the watchdog timer status.

Figure 29. Watchdog Timer Control Register 1 (WDTWCTL1)

15	14	13 12	11 0
WDFLAG	WDEN	WDIKEY	Reserved
R/W-0	R/W-0	R/W-0	R/W-0

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

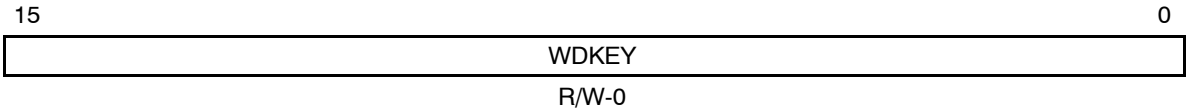
Table 20. Watchdog Timer Control Register 1 (WDTWCTL1) Bit Field Descriptions

Bit	Field	Value	Description
15	WDFLAG		Watchdog timer flag bit. WDFLAG is cleared when the watchdog timer is moved to the Active state, when a hardware reset occurs, and when 1 is written to WDFLAG.
		0	No watchdog timer timeout event occurred.
		1	Watchdog timer timeout event occurred.
14	WDEN		Watchdog timer enable bit. WDEN must be set to move the watchdog timer to the Pre-active state (see Figure 16 on page 39).
		0	Watchdog timer is disabled. Watchdog timer output pin is disconnected from the watchdog timer timeout event, and the timer counter starts to count. The timer is in the general-purpose timer mode.
		1	Watchdog timer is enabled. The watchdog timer output pin is connected to the watchdog timeout event. The watchdog timer can be disabled by a watchdog timeout event or by a hardware reset.
13–12	WDIKEY	00b–11b	Watchdog idle enable key bits. As described in section 4.6, a write sequence of 01b followed by 10b is required before the watchdog timer can enter the idle mode. Writing 00b to WDIKEY prevents the watchdog timer from entering the idle mode. WDIKEY is not applicable in the general-purpose timer mode.
11–0	Reserved	0	Reserved. The reserved bit location is always read as 0. A value written to this field has no effect.

5.12 Watchdog Timer Control Register 2 (WDTWCTL2)

The watchdog timer control register 2 (WDTWCTL2) provides the 16-bit WDKEY for watchdog service.

Figure 30. Watchdog Timer Control Register 2 (WDTWCTL2)



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

Table 21. Watchdog Timer Control Register 2 (WDTWCTL2) Bit Field Descriptions

Bit	Field	Value	Description
15–0	WDKEY	0000h–FFFFh	Watchdog timer service key bits. Before a watchdog timeout event occurs, only a write sequence of a A5C6h followed by a DA7Eh services the watchdog timer. Any other write triggers a watchdog timeout event immediately. The default value of WDKEY is 0000h. WDKEY is not applicable in the general-purpose timer mode.

Revision History

This document was revised to SPRU618B from SPRU618A, which was dated February 2003. Notable changes that were made since the last revision are listed in the following table.

Table 22. Revision History

Page	Additions/Modifications/Deletions
Global	Applied consistent terminology for counter and period values. Each timer has a *timer counter* that counts up to the *timer period*. If the timer has a prescaler, it also contains a *prescale counter* that counts up to the *prescale period*.
Global	Made revisions to clarify that: ❑ The timer output/interrupt is generated when the timer counter counts up to the timer period. If the timer is set to run continuously (ENAMODE = 10b), one timer clock cycle after the timer counter matches the timer period, the timer counter is reset to 0 and starts counting again. ❑ If the timer counter is clocked by a prescaler, one timer clock cycle after the prescale counter matches the prescale period, the timer counter is incremented by 1.
Global	Made revisions to indicate that if the internal clock is selected to drive the timer, the internal clock is the fast peripherals clock (SYSCLK1) of the DSP.
Global	Made revisions to indicate that the third, or watchdog, timer may be configured in any one of the four timer modes: 64-bit watchdog timer mode, 64-bit general-purpose timer mode, 32-bit dual timers chained mode, or 32-bit dual timers unchained mode.
3	Revised the preface.
11	Changed the title of section 1 to *Introduction to the Timers*.
13	Modified Figure 2, *Timer Block Diagram*.
14	Added Figure 3, *Generation of the Internal Timer Clock*, and the paragraph that describes it.
16	Modified Figure 4, *64-Bit Timer Mode Block Diagram*.
18, 19	Modified Figure 5, *Dual 32-Bit Timers Chained Mode Block Diagram*, and Figure 6, *Dual 32-Bit Timers Chained Mode Example.*
20, 21	Modified Figure 7, *Dual 32-Bit Timers Unchained Mode Block Diagram*, and Figure 8, *Dual 32-Bit Timers Unchained Mode Example*.
23	Modified the first paragraph of section 3.1, *Timer Pin*.
23	Modified section 3.1.1, *Timer Pin Used for Clock Source*.

Table 22. Revision History (Continued)

Page	Additions/Modifications/Deletions
27	Modified section 3.4, *Timer Clock Source Selection*.
29	Modified section 3.8, *Timer Interrupt Rate*.
30	Modified section 3.10, *Timer Emulation Modes*.
30	Added the following information to section 3.11, *Timer Operation in the GPIO Mode*: GPIO functionality and timer functionality cannot be used at the same time. GPIO functionality can be enabled in GPEN only when the timer is reset (TIM12RS = 0 and TIM34RS = 0 in GCTL1). While GPIO functionality is enabled, the timer cannot be taken out of reset. The TSTAT bit in CTL1 is not modified during GPIO activity.
31	Removed the figure from section 3.11.1, *Timer Pin Used as a General-Purpose Output*.
31	Removed the figure from section 3.11.2, *Timer Pin Used as a General-Purpose Input*.
32	Modified section 3.12.2, *Timer Count = 0 and Timer Period = 0 (No Prescaler)*.
33	Replaced section 3.12.3 with a section titled *Timer Count = 0, Timer Period = 0, Prescale Count = 0, and Prescale Period = 0*.
33	Corrected the third data value in Figure 13, *32-Bit Timer Counter Overflow Example.*
34	Corrected the 32-bit timer information in Table 8, *Reading Timer Counter Registers*.
36	Modified section 4, *Watchdog Timer*.
42	Modified section 5, *Timer Registers*.
59	Revised the index.

Index

H

I

N

O

P

R

S

T

U

W