OMAP5910 Dual-Core Processor
Timer
Reference Guide

Literature Number: SPRU682A
October 2003
Revised January 2004

OMAP™
TEXAS INSTRUMENTS TECHNOLOGY

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 timers, the watchdog timers, and the 32kHz clock-based timer present in the OMAP5910 device.

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.

Related Documentation From Texas Instruments

The following documents describe the OMAP5910 device and related peripherals. 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.

OMAP5910 Dual-Core Processor MPU Subsystem Reference Guide (literature number SPRU671)

OMAP5910 Dual-Core Processor DSP Subsystem Reference Guide (literature number SPRU672)

OMAP5910 Dual-Core Processor Memory Interface Traffic Controller Reference Guide (literature number SPRU673)

OMAP5910 Dual-Core Processor System DMA Controller Reference Guide (literature number SPRU674)

OMAP5910 Dual-Core Processor LCD Controller Reference Guide (literature number SPRU675)

OMAP5910 Dual-Core Processor Universal Asynchronous Receiver/Transmitter (UART) Devices Reference Guide (literature number SPRU676)

OMAP5910 Dual-Core Processor Universal Serial Bus (USB) and Frame Adjustment Counter (FAC) Reference Guide (literature number SPRU677)

OMAP5910 Dual-Core Processor Clock Generation and System Reset Management Reference Guide (literature number SPRU678)

OMAP5910 Dual-Core Processor General-Purpose Input/Output (GPIO) Reference Guide (literature number SPRU679)

OMAP5910 Dual-Core Processor MMC/SD Reference Guide (literature number SPRU680)

OMAP5910 Dual-Core Processor Inter-Integrated Circuit (I2C) Controller Reference Guide (literature number SPRU681)

OMAP5910 Dual-Core Processor Timer Reference Guide (literature number SPRU682)

OMAP5910 Dual-Core Processor Inter-Processor Communication Reference Guide (literature number SPRU683)

OMAP5910 Dual-Core Processor Camera Interface Reference Guide (literature number SPRU684)

OMAP5905 Dual-Core Processor Multichannel Serial Interface (MCSI) Reference Guide (literature number SPRU685)

OMAP5910 Dual-Core Processor Micro-Wire Interface Reference Guide (literature number SPRU686)

OMAP5910 Dual-Core Processor Real-Time Clock (RTC) Reference Guide (literature number SPRU687)

OMAP5910 Dual-Core Processor HDQ/1-Wire Interface Reference Guide (literature number SPRU688)

OMAP5910 Dual-Core Processor PWL, PWT, and LED Peripheral Reference Guide (literature number SPRU689)

OMAP5910 Dual-Core Processor Multichannel Buffered Serial Port (McBSP) Reference Guide (literature number SPRU708)

Trademarks

OMAP and the OMAP symbol are trademarks of Texas Instruments.

Contents

Figures

Tables

Timers

This document describes the 32-bit timers, the watchdog timers, and the 32kHz clock-based timer present in the OMAP5910 device.

1 MPU-32-Bit Timers

The MPU has three 32-bit timers for the operating system that provide general-purpose housekeeping functions. These timers are configured either in autoreload or one-shot mode with on-the-fly read capability. The timers generate an interrupt to the MPU when equal to zero. Figure 1 shows the 32-bit timer.

Figure 1. MPU-32-Bit Timer

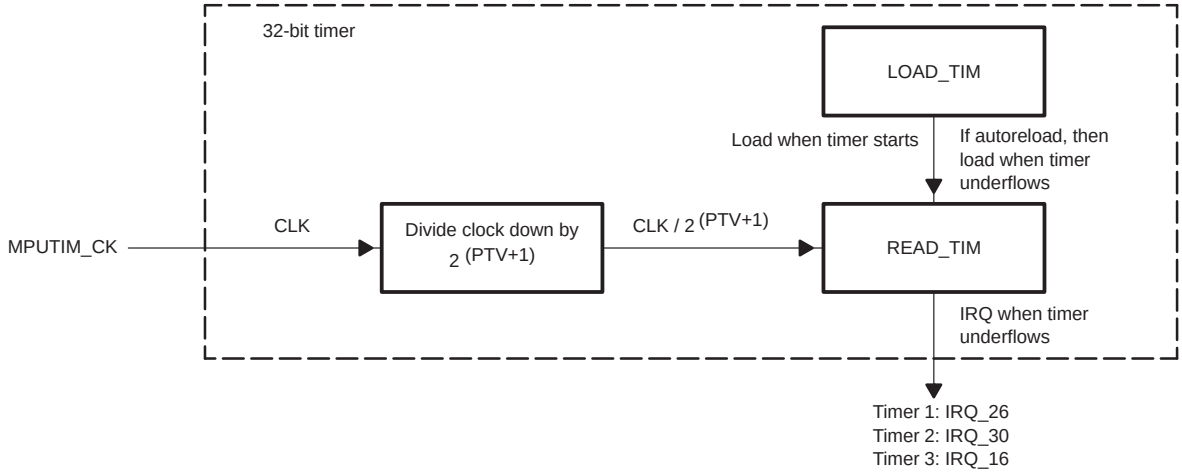


Table 1 identifies the level 1 interrupts for the three MPU 32-bit timers.

Table 1. MPU-Timer-Level-1 Interrupt

Timer	Corresponding Level 1 Interrupt
1	IRQ_26
2	IRQ_30
3	IRQ_16

The timers are 32-bit counters that receive a dedicated clock from the clock generator module 1 (either CLKIN or DPLL1 output). This clock can then be prescaled, which divides it down further. Prescaling is controlled by the PTV field of the control-timer register (CNTL_TIMER) (see Table 2).

Table 2 provides the division values for each PTV field.

Table 2. PTV Divisor: MPU Timer

PTV	Divisor
0	2
1	4
2	8
3	16
4	32
5	64
6	128
7	256

The timer-interrupt period is determined in the following manner, where t_{clk} is the clock period of the input clock, LOAD_TIM (see Figure 1) is the register that holds the value loaded when the timer counter passes through 0 or when it starts, and PTV is the prescaler field located in the control-timer register (CNTL_TIMER):

$t_{int} = t_{clk}$ X (LOAD_TIM + 1) x $2^{(PTV+1)}$

Table 3 shows the MPU timer characteristics for a variety of input frequencies.

Table 3. MPU Timer Characteristics

Input Clock	t_{clk}, Clock Period	LOAD_TIM	t_{int}, Timer Interrupt Period, for PTV = 0	t_{int}, Timer Interrupt Period, for PTV = 7
100 MHz	10 ns	0000 0001	40 ns	5.12 µs
100 MHz	10 ns	FFFF FFFF (max interrupt period)	85.9 s	10995 s (3 hr 3’ 25”)
12 MHz	83.3 ns	0000 0001	333.4 ns	42.64 µs
12 MHz	83.3 ns	FFFF FFFF (max interrupt period)	715.5 s	91589 s (25 hr, 26’29”)

1.1 Programming the MPU Timer

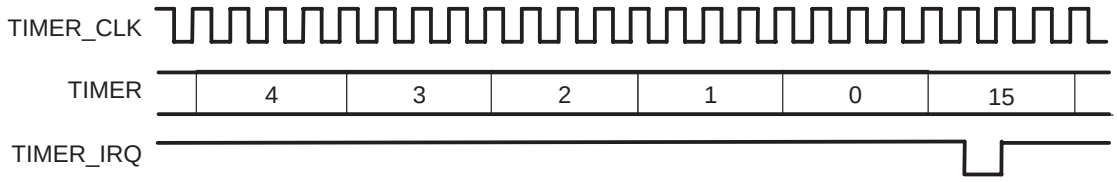
Before an MPU timer can be used, its internal clock must be enabled by setting the CLOCK_ENABLE bit of the control-timer register (CNTL_TIMER) to 1. A timer is started by setting the ST field of the control-timer register (CNTL_TIMER) to 1. It is stopped by resetting this bit to 0. When the timer is stopped, the content of the decrementer remains the same.

If the autoreload bit is disabled (the AR field of control-timer register, CNTL_TIMER, is 0), the timer decrements from the loaded value down to zero and then stops. If the autoreload bit is enabled (AR=1), the timer continues. A new value (from the load register) is loaded into the timer when it passes though zero or when it starts. An interrupt is generated when the corresponding timer counter passes through zero.

To avoid undefined results, do not change the settings of the PTV or AR fields of the control-timer register (CNTL_TIMER) or the LOAD_TIM register while the timer is running.

The timer value is held in the VALUE_TIM field of the READ_TIM register and can be read while the timer is running or stopped.

Figure 2. MPU-Timer Diagram



1.2 MPU-Timer Registers

Table 4 lists the timer registers. Table 5 through Table 7 describe the register bits.

Base address for timer 1: FFFE:C500

Base address for timer 2: FFFE:C600

Base address for timer 3: FFFE:C700

Bit width: 32 bits

Table 4. MPU Timer Registers

Timer 1, Timer 2, and Timer 3					
Register	**Descriptions**	**R/W**	**Size**	**Offset**	**Reset Value**
CNTL_TIMER	Control timer	R/W	32 bits	0x00	0x0000 0000
LOAD_TIM	Load timer	W	32 bits	0x04	Undefined
READ_TIM	Read timer	R	32 bits	0x08	Undefined

Table 5. Control-Timer Register (CNTL_TIMER)

Bits	Name	Value	Description	Reset Value
31-7	RESERVED			
6	FREE		FREE bit	0
		0	Timer stops counting in the suspend mode.	
		1	Timer continues counting in the suspend mode.	
5	CLOCK_ENABLE		External timer clock enable (not supported on 5910)	0
4-2	PTV		Prescale clock timer value (see Table 2)	0
1	AR		Autoreload/One-shot timer	0
		0	One-shot timer.	
		1	Autoreload timer.	
0	ST		Start/Stop timer	0
		0	Stop timer.	
		1	Start timer.	

Table 6. Load-Timer Register (LOAD_TIM)

Bit	Name	Description	Reset Value
31-0	LOAD_TIM	The value is loaded into VALUE_TIM when the timer passes through 0 or when it starts.	Undefined

Table 7. Read-Timer Register (READ_TIM)

Bit	Name	Description	Reset Value
31-0	VALUE_TIM	Value of the timer	Undefined

2 MPU-Watchdog Timer

The MPU-watchdog timer (see Figure 3) can be configured as either a watchdog timer or a general-purpose timer.

2.1 Introduction

The MPU-watchdog timer is power-up enabled and defaults to a watchdog timer for the OMAP5910 device. A watchdog timer requires that the user program or the OS periodically write to the count register before the counter underflows. If the counter underflows, the watchdog timer generates a reset to the MPU and the DSP. The watchdog timer detects user programs stuck in an infinite loop, loss of program control, or a runaway condition. When used as a general-purpose timer, the watchdog timer is a 16-bit timer configurable either in autoreload or one-shot mode with on-the-fly read capability. The timer generates an interrupt to the MPU when the count passes through zero (see Figure 4).

Figure 3. MPU-Watchdog Timer

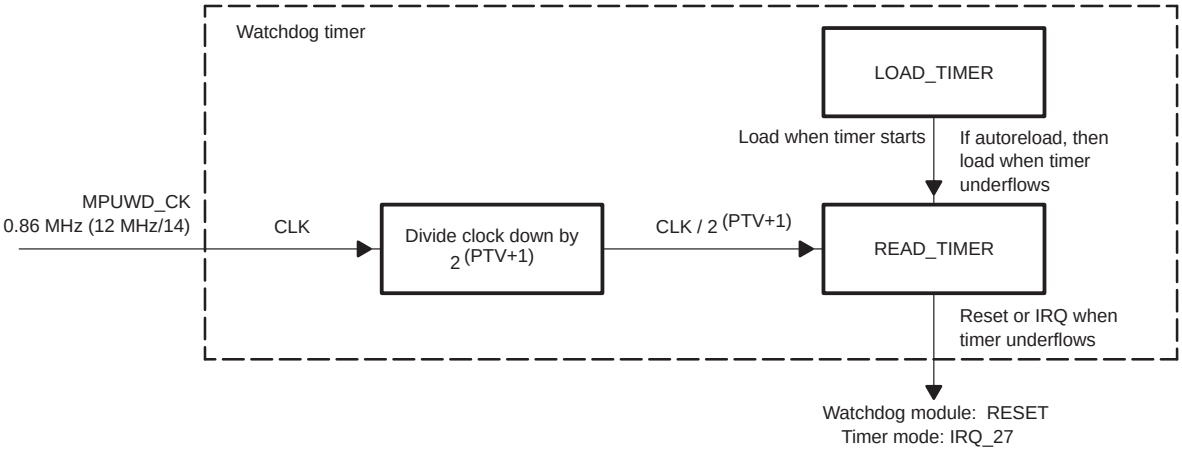


Table 8. MPU-Watchdog-Timer-Level-1 Interrupt

Timer	Corresponding Level 1 Interrupt
WD	IRQ_27

By default, this timer is configured as a watchdog timer and generates a reset of the MPU approximately every 19 seconds, unless it is disabled or updated properly. If, during system development, an unexpected reset is encountered every 19 seconds or so, this is probably the reason.

Be certain to disable the watchdog timer before placing the MPU in the deep-sleep mode. It must not be left configured as a watchdog timer. The watchdog timer underflow generates a reset to the MPU and the DSP. If CLKIN is 12 MHz and the watchdog timer values are left at their power-up state (the value loaded into LOAD_TIM is set to the maximum value of 0xFFFF at power-up), the reset occurs in approximately 19 seconds.

The watchdog timer uses a special clock from the MPU clock frequency generation module (CLKM1). This clock (MPUWD_CK) is CLKIN/14. When configured as a watchdog timer, the prescaler field (PTV of CNTL_TIMER (reference)) is fixed at 7. When configured as a general-purpose timer, the prescaler field can range from 0 to 7. The time from writing a new value to the counter underflow is between 256*Tclk to 16,777,216*Tclk, where Tclk = CLKIN/14 for a CLKIN clock frequency of 12 MHz, and the reset time is 298 μs < t > 19s.

Table 9. PTV Value and Associated Divisor Value

PTV	Divisor
0	2
1	4
2	8
3	16
4	32
5	64
6	128
7	256

The timer-interrupt period is determined in the following manner, where t_{clk} is the clock period of the input clock, LOAD_TIM is the register that holds the value loaded when the timer passes through 0 or when it starts, and PTV is the prescaler field located in the control-timer register (CNTL_TIMER). The value of the PTV field is forced to 7 if the timer is in the watchdog mode.

$$t_{int} = t_{clk} \text{ X } (\text{LOAD\_TIM} + 1) \text{ x } 2^{(\text{PTV}+1)}$$

Table 10 shows the characteristics of the MPU watchdog timer for different LOAD_TIM values.

Table 10. MPU-Watchdog-Timer Characteristics

Input clock, CLKIN	t_{clk}, Clock Period†	LOAD_TIM	t_{int}, Timer Interrupt Period, for PTV = 7
12 MHz	1167 ns	0001	597.34 µs
12 MHz	1167 ns	FFFF (max interrupt period)	19.57 s

† The 12-MHz clock is divided by 14.

CAUTION

If LOAD_TIM = 0 and AR (auto-reload mode) = 1, the timer is always 0 and can never decrement. Here the timer interrupt is asserted and stays asserted all the time.

2.2 Programming the MPU-Watchdog Timer in Watchdog Mode

On power up, the MPU-watchdog timer defaults to watchdog mode and the value loaded into the LOAD_TIM register is set to the maximum value (0xFFFF). This gives the user a time of 16,777,216 * t_{clk} (19.57 seconds) to change the timer mode or write a new value (different from 0xFFFF) into the LOAD_TIM register.

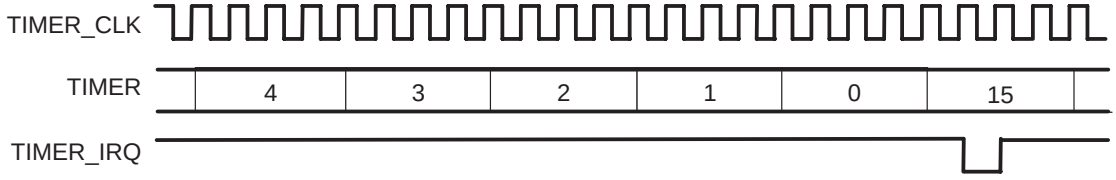
The user program or the OS must write periodically to the count register, LOAD_TIM, before the counter underflows. The newly loaded value must be different from the previous value because the write is taken into account only if the newly loaded value is different from the previous value. Due to internal sequencing, the user must wait three timer-clock periods before writing a new value into the LOAD_TIM register. If CLKIN is 12 MHz, the duration of three timer-clock periods is approximately 3.5 µs.

By writing a predefined sequence (0xF5 followed by 0xA0) to the TIMER_MODE register (see Table 15), the timer can be configured as a general-purpose timer. A sequence decode is initialized when 0xF5 is written to the TIMER_MODE register. Once in this state, if the next write is different from 0xA0, the state machine causes a reset as if the watchdog timer has underflowed. The watchdog timer cannot be disabled by simply clearing the watchdog bit of the TIMER_MODE register.

When the timer has been configured as a general-purpose timer, it can be switched back to the watchdog mode by writing a 1 to the watchdog bit of the TIMER_MODE register. In this case, the value loaded into LOAD_TIM is set to the maximum value (0xFFFF) as on power up.

In MPU-watchdog mode, the control-timer register (CNTL_TIMER) must not be used. The MPU-watchdog timer cannot be stopped by clearing the ST field. The prescale value is 7, regardless of the PTV field value. Autoreload and one-shot do not apply, because, if the counter underflows, the processor is reset and the watchdog registers are reinitialized.

Figure 4. MPU-Watchdog Timer Diagram



2.3 Programming the MPU-Watchdog Timer in Timer Mode

The MPU-watchdog timer is started by setting the ST field of the control-timer register (CNTL_TIMER) to 1. It is stopped by resetting this bit to 0. When the timer is stopped, the content of the decrementer is frozen.

If the autoreload bit is disabled (the AR field of the control-timer register, CNTL_TIMER, is 0), the timer decrements from the loaded value down to zero and then stops. If the autoreload bit is enabled (AR = 1), the timer continues. A new value (from the load register) is loaded into the timer when it passes though zero or when it starts. An interrupt is produced when the corresponding timer is equal to zero.

To avoid an undefined value, do not change the setting of the PTV field or the AR field of the control-timer register (CNTL_TIMER) or the LOAD_TIM register while the timer is running. The PTV can be set to values other than 7 when the watchdog timer is in the timer mode.

The timer value is held in the VALUE_TIM field of the READ_TIM register and can be read while the timer is running or stopped.

2.4 MPU-Watchdog Timer Registers

Table 11 lists the MPU-watchdog timer registers. Table 12 through Table 15 describe the register bits.

Base address for watchdog timer: FFFE:C800

Bit width: 32 bits

Table 11. MPU-Watchdog-Timer Registers

Name	Description	R/W	Size	Offset	Reset Value
CNTL_TIMER	Control timer	R/W	16 bits	x00	0x0002
LOAD_TIM	Load timer	W	16 bits	X04	0xFFFF
READ_TIM	Read timer	R	16 bits	X04	0xFFFF
TIMER_MODE	Timer mode	R/W	16 bits	X08	0x8000

Table 12. Control-Timer Register (CNTL_TIMER)

Bits	Name	Value	Description	Reset Value
15-12	RESERVED			
11-9	PTV		Prescale clock timer value	0
8	AR		Autoreload/One-shot timer	0
		0	One-shot timer.	
		1	Autoreload timer.	
7	ST		Start/Stop timer	0
		0	Stop timer.	
		1	Start timer.	
6-2	RESERVED			
1	FREE		FREE bit	1
		0	Enables suspend functionality.	
		1	Timer runs free, regardless of the suspend value.	
0	RESERVED		Reserved	

Table 13. Load-Timer Register (LOAD_TIM)

Bit	Name	Description	Reset Value
15-0	LOAD_TIM	General-purpose timer: This value is loaded when the timer passes through 0 or when it starts. Watchdog timer: Reload the timer with this value.	FFFF

Table 14. Read-Timer Register (READ_TIM)

Bit	Name	Description	Reset Value
15-0	VALUE_TIM	Read the timer value	FFFF

Table 15. Timer-Mode Register (TIMER_MODE)

Bit	Name	Value	Description	Reset Value
15	WATCHDOG		Write access	1
		1	Switch the timer mode back to the watchdog mode. Writing a 0 in this bit has no effect.	
14-8	RESERVED			
7-0	WATCHDOG_DIS		Write access only	NA
			Writing a predefined sequence (0xF5) followed by 0xA0 in this field disables the watchdog and configures the timer as a general-purpose timer.	
			Functionality:	
			After receiving 0xF5, if the second write access is different from 0xA0, the MPU core is reset.	

3 32-kHz Timer

The MPU subsystem operating system (OS) requires interrupts at regular time intervals for OS scheduling purposes. The OS time intervals can be from 1 ms to 30 ms. These time intervals can be generated using the three MPU 32-bit OS/general-purpose timers, which use CLKIN or DPLL1; however, they can not be used when the system clock is not operating. Therefore, a 32-kHz clock-based timer is needed to provide the required OS timing interval. The clock period of 32 kHz is 30.60 µs. (Note: 32 kHz refers to 32678, not 32000.)

3.1 Operating System Scalable Clock-Tick Interrupt Function

A programmable interval timer is required to generate a periodic interrupt, also called the system clock tick, to the OS. This is used to keep track of the current time to control the operation of the device drivers.

For example, Microsoft Windows CE OS scheduling requires the following:

- The periodic interrupt occurs every 1-25 ms.
- The timer is expected to run in all modes except when suspended.

The 32-kHz timer is a 24-bit down-counter that generates CPU interrupts for the MPU. The following capabilities are available:

- Timer reset
- Timer current value reading
- Timer start and stop
- Interrupt generation as timer down-counts to zero
- Timer autorestart after it counts to zero
- On-the-fly register read and write
- Interrupt disabling by writing a 1 to the interrupt bit in the control register

Timer	Corresponding Level 2 Interrupt
32-kHz timer	IRQ_22

The tick-value register (TVR) contains the desired value for the timer to count down. The tick-counter register (TCR) is loaded with this value, then starts to count down to zero and generates a negative edge sensitive interrupt (low-level pulse duration = 15 µs) to the interrupt handler. Once the interrupt is back to the high level, the counter is reloaded from its register and then starts to count down again.

3.1.1 Overriding Normal Counting

Normal operation can be overridden by using two bits in the timer-control register (TCR):

- The timer-reload bit (TRB) causes the counter to be reloaded on the next clk32-kHz cycle (whether or not the timer is counting).
- The timer-start-stop bit (TSS) causes the counter to be stopped on the next clk32-kHz cycle. When the timer is stopped, the content of the counter is frozen.

3.1.2 Loading/Autorestart of the Timer

Loading the counter in the timer can be done in two fashions:

- Write a 1 to the TRB bit in the timer-control register (TCR).
- Wait until the counter reaches zero and is reloaded from its register, if the autorestart bit (ARL) in the timer-control register (TCR) is set to 1. If not, then the timer is stopped.

3.1.3 Timer-Interrupt Period

The timer-interrupt period is defined by the value loaded into the tick-value register (TVR).

The timer-interrupt rate is as follows:

IRQ rate =(TVR+1) / 32768

Table 16. Timer-Interrupt Period

TVR Value	Interrupt Period
0x000000	30.5 µs
0x00028F	19.9 ms
0xFFFFFF (Value at reset)	512 s (8 min 32 s)

3.2 32-kHz-Timer Registers

Base address for 32-kHz timer: FFFB:9000

Table 17 lists the 32-kHz-timer registers. Table 19 through Table 21 describe the individual registers.

Table 17. 32-kHz-Timer Registers

Name	Description	R/W	Size	Address	Offset
CR	Timer control	R/W	32 bits	FFFB:9000	0x08
TVR	Tick value	R/W	32 bits	FFFB:9000	0x00
TCR	Tick counter	R	32 bits	FFFB:9000	0x04

3.2.1 Synchronization Issues

Synchronization of reads and writes to the 32-kHz clock is done in different ways for each register. This leads to slight restrictions concerning register access (see Table 18).

Table 18. Read/Write Synchronization

Register Name	Read	Write
CR	Can be read anytime. The value read is the last value written.	Two consecutive writes must be separated by at least 1 clk32 period (31 µs). If this is not the case, the value written is not guaranteed.
TCR	Reads are resynchronized on the MPUXOR_CK clock to prevent the peripheral bus from timing out. Can be read anytime, providing the MPUXOR_CK is running. If not, the value is not guaranteed. Software must wait one 32-kHz period after turning on the MPUXOR_CK clock before the TCR register can be read.	Writing to this has no effect.
TVR	Can be read anytime. The value read is the last value written.	Two consecutive writes must be separated by at least 1 clk32 period (31 µs). If this is not the case, the value written is not guaranteed.

Table 19. Timer-Control Register (CR)

Bit	Name	Value	Function	Reset Value
31-4	Reserved			
3	ARL		Autoreload/start	1
		0	One-shot mode. When the counter reaches zero, an interrupt is generated and the timer is stopped.	
		1	Sets the timer to autorestart mode.	
2	IT_ENA		Interrupt enable	0
		0	Interrupt disabled.	
		1	Interrupt enabled.	
1	TRB		Timer reload bit	0
			TRB = 1 reloads the counter. Once the counter is reloaded, TRB is set to 0.	
0	TSS		Timer start/stop	0
		0	Stop timer.	
		1	Start timer.	
			If the one-shot mode is selected (ARL = 0), this bit is automatically reset by internal logic when the timer is equal to 0.	

Table 20. Tick-Value Register (TVR)

Bit	Name	Function	Reset Value
31-24	Reserved		
23-0	TICK_VALUE_REG	This value is loaded when the timer passes through 0 or when it starts.	0xFFFFFF

Table 21. Tick-Counter Register (TCR)

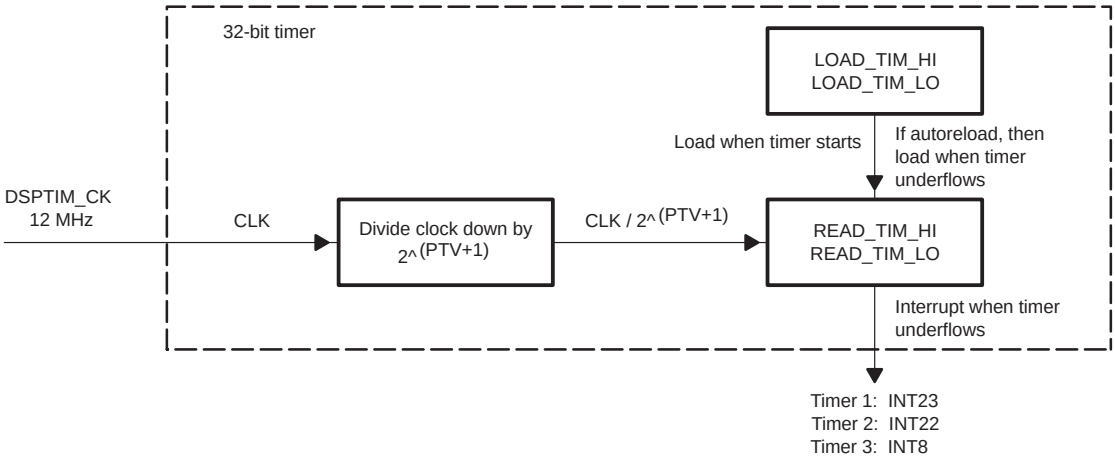
Bit	Name	Function	Reset Value
31-24	Reserved		
23-0	TICK_COUNTER_REG	Value of the timer	0xFFFFFF

4 DSP-32-Bit Timers

The DSP has three 32-bit timers that are available for general-purpose housekeeping functions. The counters/timers are configurable either in autoreload or in one-shot mode with on-the-fly read capability. Each timer generates a corresponding level 1 interrupt to the DSP when equal to zero, as shown in Table 22, *Timer Interrupt Levels*.

Figure 5 shows the DSP timers in detail.

Figure 5. DSP Timers



4.1 DSP-Timer-Interrupt Levels

Table 22. DSP-Timer-Interrupt Levels

Timer	Corresponding Level 1 Interrupt	Required Sensitivity Setup
1	INT23	Edge
2	INT22	Edge
3	INT8	Edge

The timers are counters that receive a dedicated clock from the clock generator module 2 (either CK_REF or CK_GEN2 output divided by 2). This clock can be prescaled (divided down) as controlled by the prescale clock-timer value (PTV) field of the control-timer register (shown in Table 23, *PTV Divisors: 32-Bit Timers*).

Table 23. PTV Divisors: 32-Bit Timers

PTV	Divisor
0	2
1	4
2	8
3	16
4	32
5	64
6	128
7	256

The timer-interrupt period is calculated as follows:

$t_{int} = t_{clk} \times (LOAD\_TIM + 1) \times 2^{(PTV+1)}$

where:

t_{clk} is the clock period of the input clock.

The load-timer register (LOAD_TIM) holds the value loaded when the timer passes through 0 or when it starts.

PTV is the prescaler field located in the control-timer register.

Table 24 shows the characteristics for all three timers for different input frequencies.

4.2 DSP-Timer Characteristics

Table 24. DSP-Timer Characteristics

Input Clock	t_{clk}, Clock Period	LOAD_TIM	t_{int}, Timer Interrupt Period for PTV = 0	t_{int}, Timer Interrupt Period for PTV = 7
12 MHz	83.3 ns	0001	333 ns	42.67 µs
12 MHz	83.3 ns	FFFF (max interrupt period)	10.92 ms	1398.1 ms

If LOAD_TIM = 0 and AR (auto-reload mode) = 1, the timer is always 0 and can never decrement. Here the timer interrupt is asserted and stays asserted all the time. Since the timer interrupts are edge-sensitive, only one interrupt is recognized because there is one initial edge, and then the interrupt is asserted constantly.

4.3 Programming the DSP Timer

To start a DSP timer, set the start-timer bit (ST) of the control-timer register to 1. Reset the bit to 0 to stop the timer. When the timer stops, the decrementer content is frozen.

Set the autoreload bit (AR) of the control-timer register to 0 to have the timer decrement from the loaded value down to zero and then stop. Set the AR to 1 to have the timer continue.

- A new value from the load register is loaded into the timer when it passes though zero or when it starts.
- An interrupt is produced when the corresponding timer is equal to zero.

To avoid undefined results, do not program the PTV, AR, or load-timer register while the timer is running.

The timer value is held in the value-timer register (VALUE_TIM) and can be read while the timer is running or stopped.

The load-timer and read-timer registers are actually split into two 16-bit portions; therefore, two 16-bit accesses from the DSP are required. To properly read the read-timer register, access the upper 16 bits first, then the lower 16 bits. The DSP can access them through a single 32-bit access.

4.4 DSP-Timer Registers

Table 25 lists the timer registers. Table 26 through Table 33 describe the register bits.

Table 25. DSP-Timer Registers

Register Name	Description	R/W	Size (Bits)	Offset	Reset Value
CNTL_TIMER	Control timer	R/W	16	0x00	0x0002
LOAD_TIM_HI	Load timer—high	W	16	0x02	0xFFFF
LOAD_TIM_LO	Load timer—low	W	16	0x03	0xFFFF
READ_TIM	Read timer	R	16	0x02	0xFFFF
TIMER_MODE	Timer mode	R/W	16	0x04	0x8000

Table 26. Control-Timer Register (CNTL_TIMER)

Bit	Name	Value	Descriptions	Reset Value
15-8	Unused			
7	SOFT		This bit is used with the FREE bit to determine the peripheral state when a breakpoint is encountered. Used in the emulation mode.	0
		0	Peripheral halts immediately, either retaining or discarding the current state.	
		1	Peripheral stops after completion of the current task.	
6	FREE		This bit is used with the SOFT bit to determine the peripheral state when a breakpoint is encountered. Used in the emulation mode.	0
		0	SOFT bit selects the emulation mode.	
		1	Peripheral clock runs free regardless of the SOFT bit.	
5	CLOCK_ENABLE		External timer clock enable (Not supported on 5910)	0
4-2	PTV		Prescale clock timer value	0
1	AR		Autoreload/One-shot timer	0
		0	One-shot timer.	
		1	Autoreload timer.	
0	ST		Start/Stop timer	0
		0	Stop timer.	
		1	Start timer.	
			With the one-shot mode selected (AR = 0), the bit is automatically reset by internal logic when the timer equals 0.	

The load-timer register (LOAD_TIM) is a 32-bit register (see Table 27 and Table 28). The data width of the TIPB connected to this peripheral is only 16 bits. Therefore, two 16-bit TIPB write transactions are needed to load the load-timer register (LOAD_TIM).

If the DSP is ready to load the load-timer register (LOAD_TIM), it can send a 32-bit write request (with offset address of 04) to the DSPI. The DSPI has the capability of converting this 32-bit request into two 16-bit TIPB writes on the DSP TIPB.

Table 27. Load-Timer-High Register (LOAD_TIM_HI)

Bit	Name	Description	Reset Value
15-0	LOAD_TIM_HI	This value is loaded when the timer passes through 0 or when it starts. LOAD_TIM_HI is the same as LOAD_TIM[31:16].	Undefined

Table 28. Load-Timer-Low Register (LOAD_TIM_LO)

Bit	Name	Description	Reset Value
15-0	LOAD_TIM_LO	This value is loaded when the timer passes through 0 or when it starts. LOAD_TIM_LO is the same as LOAD_TIM[15:0].	Undefined

The read-timer register (READ_TIM) is a 32-bit register (see Table 29 and Table 30). The data width of the TIPB connected to this peripheral is only 16 bits. Therefore, two 16-bit TIPB read transactions are required to read the value of the read-timer register (READ_TIM). Also, note that since the TIPB strobe is completely asynchronous with the timer_clk, synchronization is done to make sure that the read-timer register (READ_TIM) value is not read while it is being incremented.

Table 29. Read-Timer-High Register (VALUE_TIM_HI)

Bit	Name	Description	Reset Value
15-0	VALUE_TIM_HI	Value of the timer. This is the same as READ_TIM[31:16].	Undefined

Table 30. Read-Timer-Low Register (VALUE_TIM_LO)

Bit	Name	Description	Reset Value
15-0	VALUE_TIM_LO	Value of the timer. This is the same as READ_TIM[15:0] at the time of the last TIPB read to READ_TIM_HI.	Undefined

The following sequence must be followed to read the READ_TIM register properly:

1) Perform a TIPB-read transaction to read the upper 16 bits of the read-timer register (READ_TIM) (offset = 8). When the read-timer register (READ_TIM) is read and synchronized, the upper 16 bits are driven onto the data bus of the TIPB and the lower 16 bits of the read-timer register (READ_TIM) are stored in a temporary register.

2) Perform a TIPB-read transaction to read the lower 16 bits of the read-timer register (READ_TIM) (offset = 10). During this read, the value of the temporary register is forwarded onto the TIPB bus instead of reading the read-timer register (READ_TIM) again. This is done because the TIMER can change value between the two TIPB read transactions.

Therefore, to read the value of the read-timer register (READ_TIM) correctly, the first TIPB-read access must be to the upper 16 bits (that is, offset = 8), followed by the TIPB-read access to the lower 16 bits (that is, offset = 10 (decimal)).

Note:

If the DSP is ready to read the read-timer register (READ_TIM), it can send a 32-bit read request (with offset address of 08) to the DSPI. The DSPI can convert this 32-bit request into two 16-bit TIPB writes on the DSP TIPB, thus resolving all the above sequencing issues.

Table 31. DSP-Timer-1 Registers

Register Name	Description	R/W	Size (Bits)	Word Address	Reset Value
CNTL_TIMER1	Timer control register	R/W	16	0x2800	0x0000
LOAD_TIM1	Value that must be loaded into the timer when the timer passes through 0	W	32	0x2802	Undefined
READ_TIM1	Timer counter	R	32	0x2804	Undefined

Table 32. DSP-Timer-2 Registers

Register Name	Description	R/W	Size (Bits)	Word Address	Reset Value
CNTL_TIMER2	Timer control register	R/W	16	0x02C00	0x0000
LOAD_TIM2	Value that must be loaded into the timer when it passes through 0	W	32	0x02C02	Undefined
READ_TIM2	Timer counter	R	32	0x02C04	Undefined

Table 33. DSP-Timer-3 Registers

Register Name	Description	R/W	Size (Bits)	Word Address	Reset Value
CNTL_TIMER3	Timer control register	R/W	16	0x03000	0x0000
LOAD_TIM3	Value that must be loaded into the timer when it passes through 0	W	32	0x03002	Undefined
READ_TIM3	Timer counter	R	32	0x03004	Undefined

5 DSP-Watchdog Timer

When powered up, the timer defaults to the watchdog timer for the DSP. This configuration requires that the user program or the OS periodically write to the count register before the counter underflows, to prevent the timer from generating a reset to the DSP. This function detects user programs stuck in infinite loops, which can result in program control loss or runaway programs.

Figure 6. Watchdog Timer

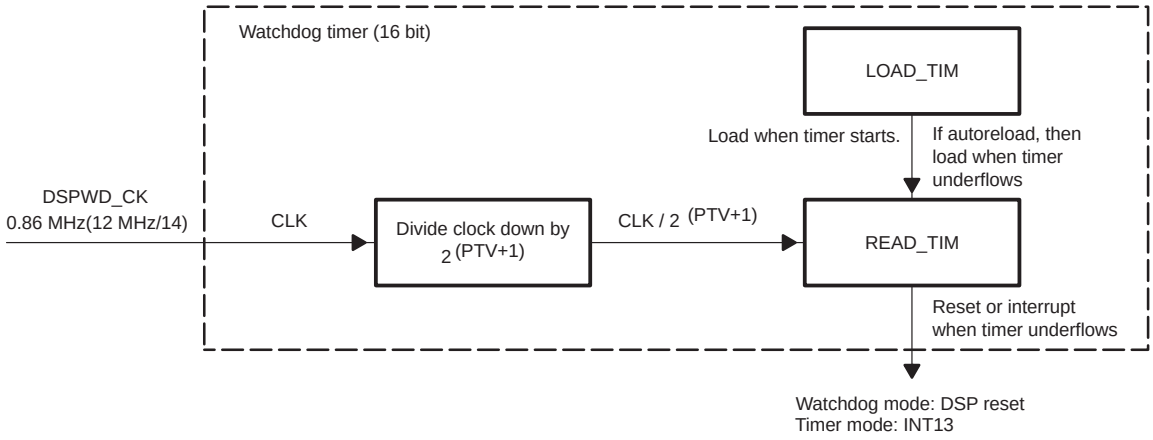


Table 34. DSP-Watchdog-Timer Interrupt

Timer	Corresponding Level 1 Interrupt
WD	INT13

Note:

By default, this timer is configured as a watchdog timer and, unless disabled or updated properly, generates a reset of the DSP approximately every 19 seconds. If, during system development, an unexpected reset is encountered every 19 seconds or so, this is probably the reason.

CAUTION

Be certain to disable the watchdog timer before placing the DSP processor in the deep-sleep mode. It must not be left configured as a watchdog timer.

The watchdog-timer underflow resets the DSP. If the input clock is 12 MHz and the watchdog timer values are left at their power-up state (the value loaded into the load-timer register (LOAD_TIM) is set to the maximum value of 0xFFFF), reset occurs in approximately 19 seconds.

The watchdog timer uses the clock derived from the clock frequency generation module for synchronization. This clock (DSPWD_CK) is CLKIN/14. When configured as a watchdog timer, the prescaler field (PTV) of the control-timer register (CNTL_TIMER) is fixed to 7. When configured as a general-purpose timer, the prescaler field can range from 0 to 7. The time from writing a new value to counter underflow is:

between $256*t_{clk}$ to $16{,}777{,}216*t_{clk}$

where t_{clk} = [input clock]/14 for a clock frequency of 12 MHz

and the reset time is: 298 µs < t > 19 s).

Table 35. PTV Divisors: Watchdog Timer

PTV	Divisor
0	2
1	4
2	8
3	16
4	32
5	64
6	128
7	256

The timer period is defined by:

- The value of the PTV, which is forced to 7 if the timer is in the watchdog mode
- The value of the load-timer register

The timer interrupt period is:

$t_{int} = t_{clk} \times (LOAD\_TIM + 1) \times 2^{(PTV+1)}$

where t_{clk} is the clock period of the input clock.

Table 36 shows the characteristics of the watchdog timer for different input frequencies:

Table 36. Watchdog-Timer Characteristics

Input Clock	t_{clk}, Clock Period†	LOAD_TIM	t_{int}, Timer Interrupt Period, PTV = 7
12 MHz	1167 ns	0001	597.34 µs
12 MHz	1167 ns	FFFF (max interrupt period)	19.57 s

† The 12-MHz clock is divided by 14.

If LOAD_TIM = 0 and AR (auto-reload mode) = 1, the timer is always 0 and can never decrement. Here the timer interrupt is asserted and stays asserted all the time. As the timer interrupts are edge-sensitive, only one interrupt is recognized because there is one initial edge, and then the interrupt is asserted constantly.

5.1 Programming the DSP-Watchdog Timer in Watchdog Mode

On power up, the watchdog timer is enabled in the watchdog mode and the value loaded into the load-timer register is set to the maximum value (0xFFFF). This gives the user a duration of 16,777,216 * t_{clk} to change the timer mode or write a new value (different from 0xFFFF) into the load timer register.

The user program or the OS must write periodically to the load-timer register (LOAD_TIM) before the counter underflows. The newly loaded value must be different from the previous because the write is taken into account only if the newly loaded value is different from the previous one. Due to internal sequencing, the user must wait for three timer-clock periods before writing a new value into the load-timer register. If the input clock is 12 MHz, three timer-clock periods are approximately 3.5 µs.

By writing a predefined sequence (0xF5 followed by 0xA0) to the TIMER_MODE register (see Table 41), the timer can be configured as a general-purpose timer. A sequence decode is initialized when 0xF5 is written to the TIMER_MODE register. Once in this state, if the next write is different from 0xA0, the state machine causes a reset as if the watchdog timer has underflowed. The watchdog timer cannot be disabled by simply clearing the watchdog bit of the TIMER_MODE register.

Once the timer has been configured as a general-purpose timer, it can be switched back to the watchdog mode by writing a 1 to bit 15 (WATCHDOG) of the timer-mode register (TIMER_MODE). In this case, the value loaded into the load-timer register (LOAD_TIM) is set to the maximum value (0xFFFF) as on power-up.

In the watchdog mode, the control-timer register (CNTL_TIMER) must not be used. The watchdog timer can not be stopped by clearing bit 7 (ST), and the prescale value is 7, regardless of the PTV field. Autoreload and one-shot do not apply, because if the counter underflows, the processor is reset and the watchdog registers are reinitialized.

5.2 Programming the DSP-Watchdog Timer in Timer Mode

To start a timer, set the start timer (ST) bit of the control-timer register to 1. Reset the bit to 0 to stop the timer. When the timer stops, the decrementer content is frozen.

Set the autoreload (AR) bit of the control-timer register to 0 to decrement from the loaded value down to zero and then stop. Set the AR bit to 1 to continue. A new value from the load-timer register is loaded into the timer when it passes though zero or when it starts. An interrupt is produced when the corresponding timer is equal to zero.

To avoid undefined results, do not program the PTV and AR bits or the load register while the timer is running. The PTV bits can be set to values other than 7 when the watchdog timer is in timer mode.

The timer value is held in the read-timer register and can be read while the timer is running or stopped.

The base word address for watchdog timer is 0x003400.

5.3 DSP-Watchdog-Timer Registers

Table 37 shows the DSP-watchdog-timer registers. Table 38 through Table 41 describe the register bits.

Table 37. DSP-Watchdog-Timer Registers

Register Name	Description	R/W	Size (Bits)	Address	Reset Value
CNTL_TIMER	Control timer	R/W	16	x003400	0x0002
LOAD_TIM	Load timer	W	16	x003402	0xFFFF
READ_TIM	Read timer	R	16	x003402	0xFFFF
TIMER_MODE	Timer mode	R/W	16	x003404	0x8000

Table 38. Control-Timer-Register (CNTL_TIMER)

Bit	Name	Value	Description	Reset Value
15-12	Reserved			
11-9	PTV		Prescale clock timer value	0
8	AR		Autoreload/One-shot timer	0
		0	One-shot timer.	
		1	Autoreload timer.	
			If the one-shot mode is selected (AR = 0), this bit is automatically reset by internal logic when the timer is equal to 0.	
7	ST		Start/Stop the timer	0
		0	Stop the timer.	
		1	Start the timer.	
6-2			Reserved	
1	FREE		FREE bit	1
		0	Enables the emulation suspend function; the timer can be frozen during the emulation halt on the DSP.	
		1	Timer runs free, regardless of the emulation halt condition.	
0			Reserved	

Table 39. Load-Timer Register (LOAD_TIM)

Bit	Name	Description	Reset Value
15-0	LOAD_TIM	General-purpose timer. This value is loaded when the timer passes through 0 or when it starts.	FFFF
		Watchdog timer. Reload the timer with this value.	

Table 40. Read-Timer Register (READ_TIM)

Bit	Name	Description	Reset Value
15-0	VALUE_TIM	Value of the timer	FFFF

Table 41. Timer Mode (TIMER_MODE)

Bit	Name	Value	Description	Reset Value
15	WATCHDOG		Write access: 1: Switch back from the timer mode to the watchdog mode. Writing a 0 in this bit has no effect. Read access: Status of timer mode:	1
		0	Timer is used as a general-purpose counter.	
		1	Timer is used as a watchdog timer.	
14-8			Reserved	
7-0	WATCHDOG_DIS		Write access only: Writing a predefined sequence (0xF5 followed by 0xA0) in this field disables watchdog functionality. After having received 0xF5, if the second write access is different from 0xA0, the DSP core is reset.	1

Revision History

This document was revised to SPRU682A from SPRU682, which was released in October 2003.

Table 42 lists the changes made since the previous version of this document.

Table 42. Document Revision History

Page	Additions/Modifications/Deletions
Global	Fixed spacing issues.
Global	Changed U to Undefined in the Reset Value column for certain tables.
9	Updated Figure 1, *MPU-32-Bit Timer*.
10	Changed Table 2 to *PTV Divisors: MPU Timer*.
10	Updated paragraphs above Table 3 in section 1, *MPU-32-Bit Timers*.
11	Updated the first two paragraphs of section 1.1, *Programming the MPU Timer.*
12	Added 0 in front of all Offset column values in Table 4, *MPU Timer Registers*.
12	Added note to bit 5 description emphasizing that it is not supported on 5910 to Table 5, *Control-Timer Register.*
18	Described additional actions for bits 7-0 on Table 15, *Timer-Mode Register*.
26	Added note to bit 5 description emphasizing that it is not supported on 5910 toTable 26, *Control-Timer Register*.
32	Added paragraph to section 5.1, *Programming the DSP-Watchdog Timer in Watchdog Mode*.

Index

P

R

S

T

W