

# 1 Overview

This document describes an hardware event timer component suitable for use in real-time applications. The component provides a single free-running counter and some deadline value registers which are compared to this absolute time base.

Depending on implementations the main counter and deadline values can be up to 64 bits wide while memory mapped register accesses are 32 bits wide.

Some operations are available to atomically update deadline values relative to the main counter using a single write operation from the host processor. The component has been designed to avoid race conditions checking in software by using *less or equal* comparators for deadlines.

## 2 Operation

The component contains a 16 bits prescaler stage. The *prescaler counter register* is decreased by one on rising edges of the input clock if the *counter enabled* bit of the *control register* is set. Each time the prescaler counter reaches zero, it is reloaded with the value of the *prescaler reload register* and the *main counter register* is increased by one.

The 64 bits *main counter register* value is compared on to all *deadline register* values on all cycles where no other operations are requested. When a deadline value is less or equal than the main counter value, the deadline is reached and appropriate action is taken depending on associated configuration:

- When the associated bit in the *period enable* register is set, the value of the deadline *period register* is added to the current deadline value. An interrupt is raised if the associated bit in the *interrupts enable* register is set.
- When the associated bit in the *period enable* register is cleared, An interrupt is raised if the associated bit in the *interrupts enable* register is set and the interrupt is then disabled by clearing this bit.

Several operations can be performed to update deadlines:

- The 64 bits *deadline registers* can be written using 32 bits accesses. Use of absolute deadline values enables timer reprogramming to handle earlier events without timing inaccuracy.
- The *deadline set registers* can be used to atomically set a deadline with specified 32 bits delay. This allows scheduling a delayed interrupt with a single 32 bits write from host the processor without even disabling interrupts.
- The *deadline add registers* can be used to atomically add a 32 bits value to a deadline. This allows modifying the delay of an active deadline without entering a critical code section or conditionally rescheduling an event from the interrupt handler with the exact same delay for instances.
- The *deadline copy register* can be used to copy or shift deadline values between deadline registers.

Whether deadline interrupts are enabled automatically when one of these operations is performed is globally controlled by bits in the *control register*.

## 3 Timer registers map

Real-time timer registers table

Name	Address	Direction	Long name	Description
sccnt	0x0	rw		

Name	Address	Direction	Long name	Description
			Prescaler counter register	Prescaler counter value is decremented on rising edge of the input clock and reloaded with a configurable value when it reaches zero.
scrld	0x4	rw	Prescaler reload register	This register holds the reload value for the prescaler counter. The maximum value is 0xffff.
cfg	0x8	r	Device configuration register	This read only register holds timer component informations
ctrl	0xc	rw	Control register	This register contains bits to configure various device behaviors
rtcl	0x10	r	Main counter lower 32 bits value	This register contains the lower 32 bits of the current value of the timer and latches its higher 32 bits in the rtctmp register on read.
rtch	0x14	r	Main counter higher 32 bits value	This register contains the higher 32 bits of the current value of the timer and latches its lower 32 bits in the rtctmp register on read.
rtctmp	0x18	rw	32 bits values scratch register	This register is used to store lower or higher 32 bits of 64 bits values when accessing main counter or deadlines values.
pe	0x20	rw	Period enable register	If a bit is set in this register, the corresponding deadline register is updated with addition of the associated period register each time the deadline is reached.
ie	0x24	rw	Interrupts enable register	If a bit is set in this register, an interrupt will be generated when the corresponding deadline is reached.
ip	0x28	rw	Interrupts pending register	A bit is set in this register when an interrupt has been generated by the corresponding deadline. Setting a bit to one in this register clears the interrupt pending flag.
copy	0x2c	w	Deadline copy command register	When bit N is written to one in this register, the associated deadline register value is overwritten with the value of the N+1 deadline register. The last deadline register can not be overwritten.
cancel	0x30	w	Deadline cancel register	When bit N is written to one in this register, the corresponding irq enable and pending flags are both cleared.

Real-time timer registers table (deadline registers)

dln1	0x80	rw	Deadline1 value register	This register contains lower 32 bits of the deadline value. When read, the higher 32 bits of the deadline value is latched in the rtctmp register. When written to, the deadline register higher 32 bits are set to the value of the rtctmp register.
dln1s	0x84	w	Deadline1 set register	When writting to this register, the associated 64 bits deadline register is set to the sum of the timer current value and the written unsigned value.
dln1a	0x88	w	Deadline1 add register	When writting to this register, the written value is sign-extended to 64 bits and added to the associated deadline value.
dln1p	0x8c	rw	Deadline1 period register	The value of this register is added to the deadline value when the associated bit in the pe register is set and the deadline is reached.
dln2	0x90	rw	Deadline2 value register	
dln2s	0x94	w	Deadline2 set register	
dln2a	0x98	w	Deadline2 add register	

dln2p	0x9c	rw	Deadline2 period register
dln3	0xa0	rw	Deadline3 value register
dln3s	0xa4	w	Deadline3 set register
dln3a	0xa8	w	Deadline3 add register
dln3p	0xac	rw	Deadline3 period register
dln4	0xb0	rw	Deadline4 value register
dln4s	0xb4	w	Deadline4 set register
dln4a	0xb8	w	Deadline4 add register
dln4p	0xbc	rw	Deadline4 period register
dln5	0xc0	rw	Deadline5 value register
dln5s	0xc4	w	Deadline5 set register
dln5a	0xc8	w	Deadline5 add register
dln5p	0xcc	rw	Deadline5 period register
dln6	0xd0	rw	Deadline6 value register
dln6s	0xd4	w	Deadline6 set register
dln6a	0xd8	w	Deadline6 add register
dln6p	0xdc	rw	Deadline6 period register
dln7	0xe0	rw	Deadline7 value register
dln7s	0xe4	w	Deadline7 set register
dln7a	0xe8	w	Deadline7 add register
dln7p	0xec	rw	Deadline7 period register

Real-time timer cfg register fields

Name	Width	Direction	Long name	Description
dl_cnt	3 bits	r	Deadline registers count	
irq	5 bits	r	Irq vector number	
si	1 bits	r	Separate interrupts	

### 3 Timer registers map

## Real-time timer ctrl register fields

<b>Name</b>	<b>Width</b>	<b>Direction</b>	<b>Long name</b>	<b>Description</b>
ce	1 bits	rw	Chip enabled	When cleared, this bit prevents the timer from running
iew	1 bits	rw	Interrupt enable on deadline write	When set, deadline interrupt is automatically enabled when its value is updated by writting to the associated dln register
ies	1 bits	rw	Interrupt enable on deadline set	When set, deadline interrupt is automatically enabled when its value is updated by writting to the associated dln set register
iea	1 bits	rw	Interrupt enable on deadline add	When set, deadline interrupt is automatically enabled when its value is updated by writting to the associated dln add register
iec	1 bits	rw	Interrupt enable on deadline copy	When set, deadline interrupt is automatically enabled when its value is updated by writting to copy register