eSi-Timer

1 Contents

2 Overview

The eSi-Timer is a multi-function timer / counter. It has the following features:

- 32-bit counter.
- Timer and counter modes.
- Counter mode can either be edge or level sensitive.
- Single-shot or continuous programmable wrap.
- PWM output.
- Input capture register.
- Programmable automatic disable during debug.
- AMBA 3 APB slave interface.

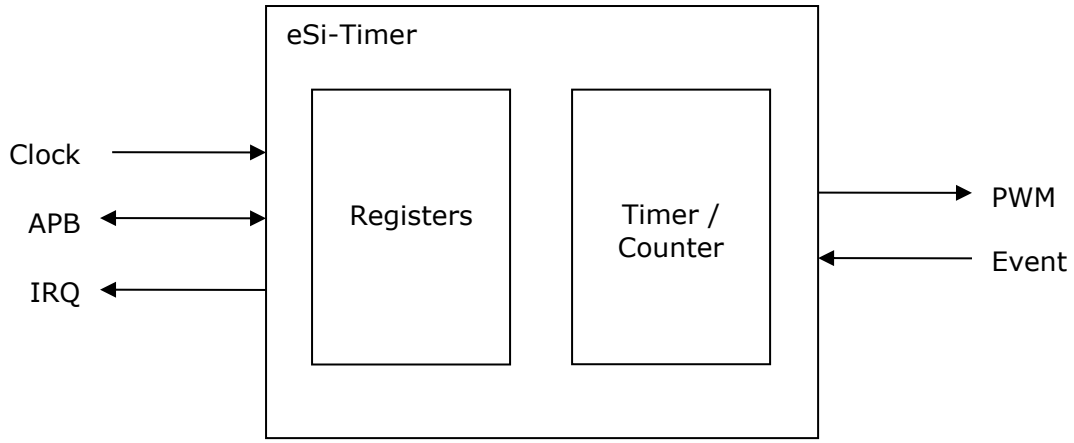

Figure 1: eSi-Timer

3 Hardware Interface

Module Name	cpu_apb_timer
HDL	Verilog
Technology	Generic
Source Files	cpu_apb_timer.v

Port	Type	Values	Description
apb_data_width	Integer	16, 32	Specifies the APB data bus width
apb_address_width	Integer	16, 32	Specifies the APB address bus width

Table 1: Parameters

Port	Direction	Width	Description
clk	Input	1	Clock used for timer. This clock must be enabled when cactive is asserted. It must be synchronous to pclk, but can be gated when not in use.
pclk	Input	1	APB clock
presetn	Input	1	APB reset, active-low
paddr	Input	apb_address_width	APB address (only 8 LSBs are used)
psel	Input	1	APB slave select
penable	Input	1	APB enable
pwrite	Input	1	APB write
pwdata	Input	apb_data_width	APB write data
evt	Input	1	Event to count
debug_active	Input	1	Indicates when debugger is active
cactive	Output	1	Clock active
pready	Output	1	APB ready
prdata	Output	apb_data_width	APB read data
pslverr	Output	1	APB slave error
pwm	Output	1	PWM output
interrupt_n	Output	1	Interrupt request, active-low

Table 2: I/O Ports

For complete details of the APB signals, please refer to the AMBA 3 APB Protocol v1.0 Specification available at:

http://www.arm.com/products/system-ip/amba/amba-open-specifications.php

The timer does not include internal synchronizing flip-flops. These should be implemented externally for the evt port if the source clock domain of this signal is asynchronous to clk.

4 Software Interface

4.1 Register Map

Register	Address offset	Access	Description
counter	0x00	R/W	Counter
wrap_comparator	0x04	R/W	Wrap comparator
output_comparator	0x08	R/W	Output comparator
input_capture	0x0c	R	Input capture
output	0x10	R/W	Output value
status	0x14	R/W	Status register
control	0x18	R/W	Control register

Table 3: Register Map when BITS equals 32

Register	Address offset	Access	Description
counter_lo	0x00	R/W	Counter (low 16-bits)
counter_hi	0x02	R/W	Counter (high 16-bits)
wrap_comparator_lo	0x04	R/W	Wrap comparator (low 16-bits)
wrap_comparator_hi	0x06	R/W	Wrap comparator (high 16-bits)
output_comparator_lo	0x08	R/W	Output comparator (low 16-bits)
output_comparator_hi	0x0a	R/W	Output comparator (high 16-bits)
input_capture_lo	0x0c	R	Input capture (low 16-bits)
input_capture_hi	0x0e	R	Input capture (high 16-bits)
output	0x10	R/W	Output value
status	0x14	R/W	Status register
control	0x18	R/W	Control register

Table 4: Register Map when BITS equals 16. Little-endian.

4.1.1 Counter

The counter is a 32-bit up counter. When in timer mode (control.M equals 0), the counter will increment by 1 on every positive edge of clk. When in counter mode (control.M equals 1), the counter will increment by 1 when either the edge or level specified in the control register is detected on the input evt. When the counter contains the same value as the wrap comparator register, the next event that would cause the counter to increment will actually cause the counter to reset to 0.

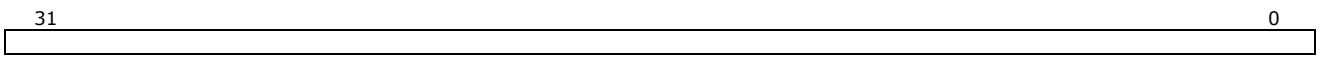

Figure 2: Format of the counter register when BITS equals 32.

When the timer is to be used with a 16-bit processor or bus, the 32-bit counter is accessed via two 16-bit registers, counter_lo and counter_hi. When counter_lo is accessed, the upper 16-bits of the counter are saved in counter_hi. This provides atomic access to the counter value, to ensure that consistent counter values are read, even if the counter updates while it is being read.

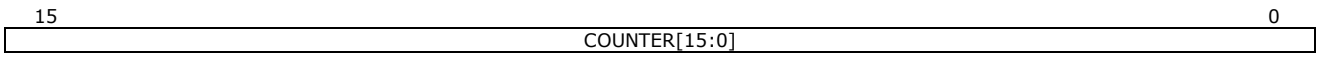

Figure 3: Format of the counter_lo register when BITS equals 16.

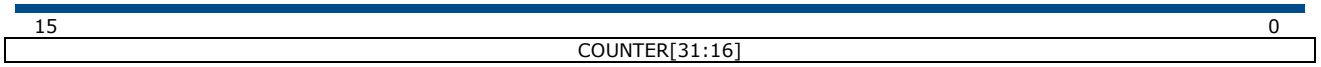

Figure 4: Format of the `counter_hi` register when BITS equals 16.

4.1.2 Wrap Comparator

The wrap comparator register contains the value after which the counter will wrap to 0. The counter therefore counts in the range [0, `wrap_comparator`].

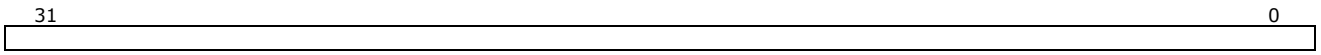

Figure 5: Format of the `wrap_comparator` register

4.1.3 Output Comparator

The output comparator register sets the value in the counter that determines when the PWM output is set to 1. The PWM output, `pwm`, is set to 0 when the counter is reset to 0.

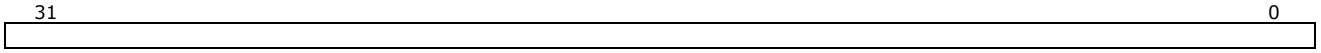

Figure 6: Format of the `output_comparator` register

4.1.4 Input Capture

The input capture register records the value in the counter register when the event described by the `control.S` and `control.M` registers on the `evt` input is detected.

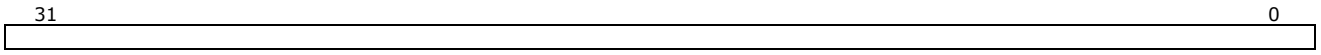

Figure 7: Format of the `input_comparator` register

4.1.5 Output Register

The output register holds the value that is output to the `pwm` port.

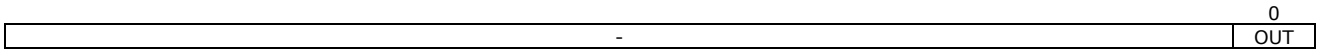

Figure 8: Format of the `output` register

4.1.6 Status Register

The status register contains a selection of flags that indicate the current status of the timer. To clear a bit in the status register, write a 1 to it. Writing 0 will leave it unchanged.

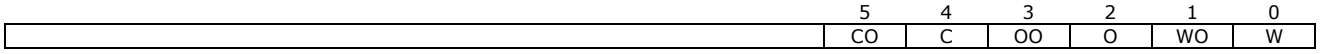

Figure 9: Format of the `status` register

Register	Values	Description
W	0 - No wrap 1 - Wrapped	Wrapped flag. Indicates whether the counter has wrapped
WO	0 - No wrap overflow 1 - Wrap overflow	Wrapped overflow flag. Indicates if the W flag was set when the counter wrapped
O	0 - Output comparator not reached 1 - Output comparator reached	Output flag. Indicates if the counter has exceeded the value in the output comparator register
OO	0 - No output overflow 1 - Output overflow	Output overflow flag. Indicates if the O flag was already set when the counter exceeded the value in the output comparator
C	0 - No capture 1 - Capture	Input capture. Indicates if an input capture event has occurred
CO	0 - No capture overflow 1 - Capture overflow	Input capture overflow. Indicates if the input capture flag, C, was already set when the last input capture event occurred

Table 5: Fields of the `status` register

4.1.7 Control Register

The control register contains a selection of flags that control the operation of the timer.

8	7	6	5	4	3	2	1	0
DD	CIE	OIE	WIE	SS	T	S	M	E

Figure 10: Format of the `control` register

Register	Values	Description
E	0 - Disabled 1 - Enabled	Enables the timer
M	0 - Timer 1 - Counter	Mode
S	0 - Negative 1 - Positive	Counter sense
T	0 - Level 1 - Edge	Counter trigger
SS	0 - Continuous 1 - Single-shot	Single-shot mode
WIE	0 - Disabled 1 - Enabled	Wrap interrupt enable
OIE	0 - Disabled 1 - Enabled	Output compare interrupt enable
CIE	0 - Disabled 1 - Enabled	Input capture interrupt enable
DD	0 – Enable during debug 1 – Disable during debug	Disable timer when debugger is active

Table 6: Fields of the `control` register

4.2 Interrupts

The timer supports the following interrupts.

- Wrap interrupt
- Output compare interrupt
- Input capture interrupt

The wrap interrupt will be raised when the counter wraps to 0 and the `WIE` flag in the `control` register is set to 1. The wrap interrupt can be acknowledged by writing a 1 to the `status.W` flag.

The output compare interrupt will be raised when the PWM output, `pwm`, is set to 1 and the `OIE` flag in the control register is set to 1. The output compare interrupt can be acknowledged by writing a 1 to the `status.O` flag.

The input capture interrupt will be raised when the event described by the `control.S` and `control.M` registers is detected on the `evt` input. The input capture interrupt can be acknowledged by writing a 1 to the `status.C` flag.

5 Revision History

Hardware Revision	Software Release	Description
1	1.0.0	Initial release.
2	3.0.2	Renamed `out` to `pwm`.
3	6.0.2	Added `debug_active` input. Added `control.DD` field.

Table 7: Revision History

