Atmel Timers

DANIEL KOHN
TECH 3233
VER 0.1

1

Input Capture

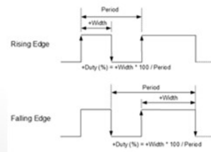
2

Timer 1

- Note that only TIMER 1 has the ability to do Input Capture.
- Timer 1 is a 16 bit timer.
- Its job is to record the timer counts (like a stopwatch) at desired edge(s) of a square wave

3

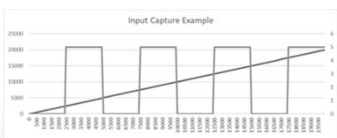
Input Capture



- Given the above, we can see that if you capture the timer count at a rising edge and capture a 2nd timer count from a falling edge, you can measure the width of the high period of the wave.
- Rising edge to rising edge gives you period of the square wave.
- Falling edge to rising edge give you the low period.

4

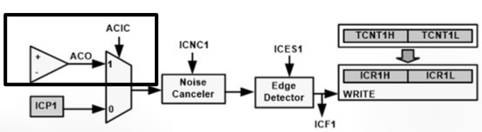
Input Capture Example



- Given the above example graph (input and TCNT and X axis values)
- We can see that the first Rising Edge happens when TNCT=2500 and the first Falling Edge is at 5000, giving us a 2500 count difference. If the prescaler is set so each count equals 8uSec, then the on time of this waveform is 2500*8uSec = 20mSec

5

Capturing

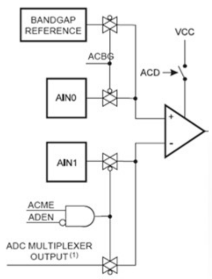


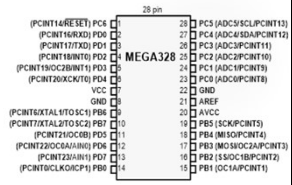
7	6	5	4	3	2	1	0	
ACD	ACBG	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0	ACSR
R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	

ACIC: Analog Comparator Input Capture Enable
0: ICP1 provides the capture signal
1: analog comparator is connected to the capturer

6

Trigger of Analog Comparator

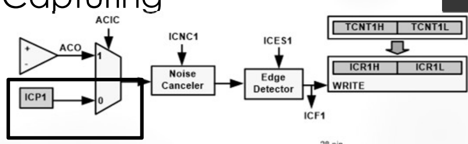


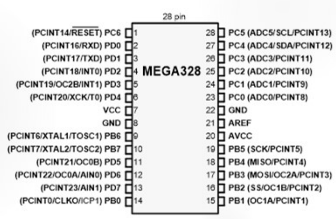


Copied from ATmega328 datasheet page 239

7

Capturing

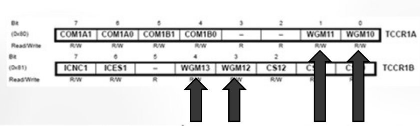




8

Input Capture Mode

We will once again use MODE 0 for the timer mode [but NOTE there are 4 WGM bits!]:



Mode	WGM13	WGM12 (CTC1)	WGM11 (PWM11)	WGM10 (PWM10)	Timer/Counter Mode of Operation	TOP	Update of OCR1x at	TOV1 Flag Set on
0	0	0	0	0	Normal	0xFFFF	Immediate	MAX

9

Timer1 Prescaler

Bit	7	6	5	4	3	2	1	0	
(0x81)	ICNC1	ICES1	–	WGM13	WGM12	CS12	CS11	CS10	TCCR1B
Read/Write	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	

Table 16-5. Clock Select Bit Description

CS12	CS11	CS10	Description
0	0	0	No clock source (Timer/Counter stopped).
0	0	1	$clk_{I/O}/1$ (No prescaling)
0	1	0	$clk_{I/O}/8$ (From prescaler)
0	1	1	$clk_{I/O}/64$ (From prescaler)
1	0	0	$clk_{I/O}/256$ (From prescaler)
1	0	1	$clk_{I/O}/1024$ (From prescaler)
1	1	0	External clock source on T1 pin. Clock on falling edge.
1	1	1	External clock source on T1 pin. Clock on rising edge.

If external pin modes are used for the Timer/Counter1, transitions on the T1 pin will clock the counter even if the pin is configured as an output. This feature allows software control of the counting.

Calculations use the same formulas as previously shown, but now You want to ensure that the timer is set for the frequency/period of Waveform to be measured

10

Calculations

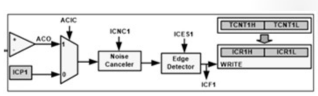
$$f_{scaled} = \frac{F_{osc}}{Prescaler}$$

$$t_{percount} = \frac{1}{f_{scaled}}$$

$$t_{ovf} = 2^n * t_{percount}$$

$t_{percount}$ will be less than the min time that can be measured. You would want the min time to be measured to be a multiple counts.
t_{ovf} will be the max time that can be measured (without having to count the number of overflows on timer1

11



ICNC1	ICES1	-	WGM13	WGM12	CS12	CS11	CS10	TCCR1B

ICNC1: Input Capture Noise Canceller
0:disabled
1:Enabled (captures after 4 successive equal valued samples)
ICSES1: Input Capture Edge Select
0: Falling edge
1: Rising edge

Bit	7	6	5	4	3	2	1	0	
(0x6F)	–	–	ICIE1	–	–	OCIE1B	OCIE1A	TOIE1	TIMSK1
Read/Write	R	R	R/W	R	R	R/W	R/W	R/W	

- **Bit 5 – ICIE1: Timer/Counter1, Input Capture Interrupt Enable**

When this bit is written to one, and the I-flag in the Status Register is set (interrupts globally enabled), the Timer/Counter1 Input Capture interrupt is enabled. The corresponding Interrupt Vector (see "Interrupts" on page 66) is executed when the ICF1 Flag, located in TIFR1, is set.

Note: It is NOT always necessary to use the interrupt for Input Capture, since it will still store TCNT1 to ICR1 Automatically on the Edge selected.

12

Bit	7	6	5	4	3	2	1	0	
0x16 (0x36)	–	–	ICF1	–	–	OCF1B	OCF1A	TOV1	TIFR1
Read/Write	R	R	R/W	R	R	R/W	R/W	R/W	

- **Bit 5 – ICF1: Timer/Counter1, Input Capture Flag**

This flag is set when a capture event occurs on the ICP1 pin. When the Input Capture Register (ICR1) is set by the WGM13:0 to be used as the TOP value, the ICF1 Flag is set when the counter reaches the TOP value.

ICF1 is automatically cleared when the Input Capture Interrupt Vector is executed. Alternatively, ICF1 can be cleared by writing a logic one to its bit location.

You can use Polling to check this bit to see if the input capture Edge has been detected.

Remember to clear the flag, you write a LOGIC 1 to the bit.

13

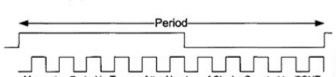
Steps to program the Input Capture Function

We use the following steps to measure the edge arrival time for the Input Capture function.

1. Initialize the TCCR1A and TCCR1B for a proper timer mode (any mode other than modes 8, 10, 12, and 14), enable or disable the noise canceller, and select the edge (positive or negative) we want to measure the arrival time for.
2. Initialize the ACSR to select the desired event source.
3. Monitor the ICF1 flag in TIFR to see if the edge has arrived. Upon the arrival of the edge, the TCNT1 value is loaded into the ICR1 register automatically by the AVR. Example 15-22 shows how the Input Capture function works. The Input Capture function is widely used to measure the period or the pulse width of an incoming signal.

14

Measuring Period



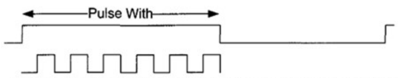
Measuring Period in Terms of the Number of Clocks Counted by TCNT

We can use the following steps to measure the period of a wave.

1. Initialize the TCCR1A and TCCR1B.
2. Initialize the ACSR to select the desired event source.
3. Monitor the ICF1 flag in TIFR to see if the edge has arrived. Upon the arrival of the edge, the TCNT1 is loaded into the ICR1 register automatically by the AVR.
4. Save the ICR1.
5. Monitor the ICF1 flag in TIFR to see if the second edge has arrived. Upon the arrival of the edge, the TCNT is loaded into the ICR1 register automatically by the AVR.
6. Save the ICR1 for the second edge. By subtracting the second edge value from the first edge value we get the time. See Examples 15-23 and 15-24. Also see Figure 15-20.

15

Measuring Pulse Width



Measuring Pulse Width in Terms of the Number of Clocks Counted by TCNT

We can use the following steps to measure the pulse width of a wave.

1. Initialize TCCR1A and TCCR1B, and select capturing on rising edge.
2. Initialize ACSR to select the desired event source.
3. Monitor the ICF1 flag in TIFR to see if the edge has arrived. Upon the arrival of the edge, the TCNT1 value is loaded into the ICR1 register automatically by the AVR.
4. Save the ICR1 and change the capturing edge to the falling edge.
5. Monitor the ICF1 flag in TIFR to see if the second edge has arrived. Upon the arrival of the edge, the TCNT value is loaded into the ICR1 register automatically by the AVR.
6. Save the ICR1 for the second edge. Subtract the second edge value from the first edge value to get the time.

16

Note on IC and OVF

- Note that if you are measuring a very slow square wave, Timer Overflows might occur. If this is the case, you will need to take into account the number of times the OVF occurs when calculating the time.

17

References

- Textbook
  - http://nicerland.com/avr/
- Atmel 328P Datasheet
  - http://ww1.microchip.com/downloads/en/DeviceDoc/ATmega48A-PA-88A-PA-168A-PA-328-P-DS-DS40002061A.pdf

18