Atmel Timers

DANIEL KOHN
TECH 3233
VER 0.1

1

Timers

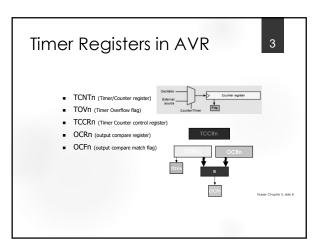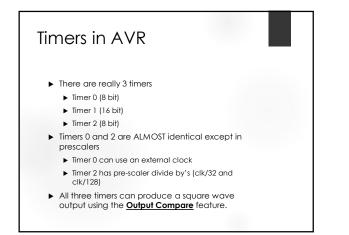- Most uCont have timer functions.
- Timers are based on a free running timer count (0-255 for 8 bit or 0 – 65535 for 16 bit) that automatically updates the Timer Counter Register (Usually called TCNTx) at a know interval.
- When a counter reaches its max value it OVERFLOWS and resets back to zero (this just shows the last few counts and the roll over):

2 5 2
2 5 3
2 5 4

2

Timer Registers in AVR 3

- TCNTn (Timer/Counter register)
- TOVn (Timer Overflow flag)
- TCCRn (Timer Counter control register)
- OCRn (output compare register)
- OCFn (output compare match flag)

Nasser Chapter 5, slide 8

3

Timers in AVR

- There are really 3 timers
  - Timer 0 (8 bit)
  - Timer 1 (16 bit)
  - Timer 2 (8 bit)
- Timers 0 and 2 are ALMOST identical except in prescalers
  - Timer 0 can use an external clock
  - Timer 2 has pre-scaler divide by's (clk/32 and clk/128)
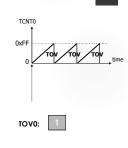- All three timers can produce a square wave output using the **Output Compare** feature.

4

Timer 1

- In addition to the fact that Timer 1 is a 16 bit timer, it also has the ability to measure On Time, Off Time and Total time of an INPUT square wave using the **Input Capture** feature.
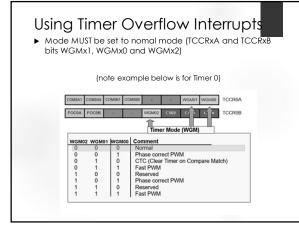
5

Timer Overflow
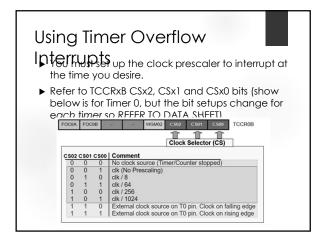
6

4/9/2020

Timer Overflow

7

- A "free running" timer counts from 0-255 (for 8 bit) or 0-65535 (for 16 bit).
- Once it reaches its highest number, the next count will reset the timer count back to zero and trigger the OVERFLOW FLAG (and interrupt if set)
- The graphic shows an 8 bit overflow

TCNT0

0xFF

TOV TOV TOV

0   time

TOV0: 1

Graphic from Nasser Chapter 5, slide 12

7

Using Timer Overflow Interrupts

- Mode MUST be set to nomal mode (TCCRxA and TCCRxB bits WGMx1, WGMx0 and WGMx2)

(note example below is for Timer 0)

COM0A1	COM0A0	COM0B1	COM0B0	-	-	WGM01	WGM00	TCCR0A
FOC0A	FOC0B	-	-	WGM02	CS02	CS01	CS00	TCCR0B

Timer Mode (WGM)

WGM02	WGM01	WGM00	Comment
0	0	0	Normal
0	0	1	Phase correct PWM
0	1	0	CTC (Clear Timer on Compare Match)
0	1	1	Fast PWM
1	0	0	Reserved
1	0	1	Phase correct PWM
1	1	0	Reserved
1	1	1	Fast PWM

8

Using Timer Overflow Interrupts

- You must set up the clock prescaler to interrupt at the time you desire.
- Refer to TCCRxB CSx2, CSx1 and CSx0 bits (show below is for Timer 0, but the bit setups change for each timer so REFER TO DATA SHEET)

FOC0A	FOC0B	-	-	WGM02	CS02	CS01	CS00	TCCR0B

Clock Selector (CS)

CS02	CS01	CS00	Comment
0	0	0	No clock source (Timer/Counter stopped)
0	0	1	clk (No Prescaling)
0	1	0	clk / 8
0	1	1	clk / 64
1	0	0	clk / 256
1	0	1	clk / 1024
1	1	0	External clock source on T0 pin. Clock on falling edge
1	1	1	External clock source on T0 pin. Clock on rising edge

9

Calculations

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

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

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

10

Example

Calculate the time for one overflow given an 8 bit timer and osc clock of 16MHz (Arduino uno clock) and a prescaler of clk/1024

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

$$= \frac{16MHz}{1024} = 15625 \ Hz$$

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

$$= \frac{1}{15625Hz} = 64uSec$$

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

$$= 2^8 * 64uSec = 16.384mSec$$

11

Using Timer Overflow Interrupts

▶ You must turn on the Interrupt Enable Bit

(note example below is for Timer 0)

TIMSK0: | - | - | - | - | - | OCIE0B | OCIE0A | TOI0E |

Interrupt	Vector Name in Atmel Studio
Timer/Counter2 Overflow	TIMER2_OVF_vect
Timer/Counter1 Overflow	TIMER1_OVF_vect
Timer/Counter0 Overflow	TIMER0_OVF_vect

• Don't forget you will still need to turn on all maskable interrupts with the sei() function.

12

TOF Polling

- You can also poll the TOVx bit to determine if a Timer Overflow has occurred

TIFR0 – Timer/Counter 0 Interrupt Flag Register

Bit	7	6	5	4	3	2	1	0	
0x15 (0x35)	–	–	–	–	–	OCF0B	OCF0A	TOV0	TIFR0
Read/Write	R	R	R	R	R	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- If you are polling you must write a ONE to TOVx to clear the bit.

13

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

14