AVR : Timers, Counters, Interrupts

Electronics Club, IIT Bombay

What is a timer/counter?

- All microcontrollers have clocks in them (or use one that resides outside of a microcontroller).
- The timer and counter functions in the microcontroller count in sync with the microcontroller clock.
- The microcontroller provides a very useful feature called prescaling. **Prescaling** is a way for the counter to skip a certain number of microcontroller clock ticks. The AVR microcontrollers allow prescaling numbers of: 8, 64, 256 and 1024.

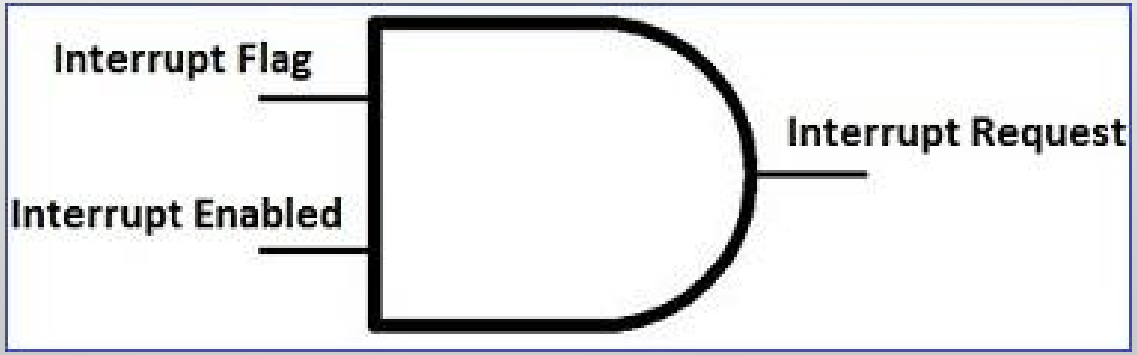
Code to make an LED blink 7 times

```
#include <avr/io.h>
int main(void) {
    DDRB = 0b00000001;
    PORTB = 0b00000000;
    TCCR1B |= 1<<CS10 | 1<<CS11;  //set prescalar
    while(1) {
        if (TCNT1 > 2232) {
            TCNT1 = 0;
            PORTB ^= 1<<PINB0;  //toggle LED
        }
    }
}
```

Interrupts

- **Interrupts** are basically events that require immediate attention by the microcontroller.
- When an interrupt event occurs the microcontroller pause its current task and attend to the interrupt by executing an **Interrupt Service Routine (ISR)**; at the end of the ISR the microcontroller returns to the task it had paused and continues its normal operations.

- An **Interrupt Service Routine (ISR)** or **Interrupt Handler** is a piece of code that should be execute when an interrupt is triggered.
- Apart from the enabled bits for the specific interrupts the global interrupt enabled bit **MUST** be enabled for interrupts to be activated in the microcontroller. This is done via the **sei()** method.



How are interrupts triggered?

- ADC
- Serial communication
- Timer matching given count
- Pin high/low

```
#include <avr/io.h>
#include <avr/interrupt.h>

int main(void) {

	sei(); //enable global interrupt

	DDRB |= 1<<PINB0;

	TCCR1B |= 1<<CS10 | 1<<CS11 | 1<<WGM12; //prescalar , enable CTC
	TIMSK1 |= 1<<OCIE1A;  //Timer1 Output Compare A Match interrupt is enabled
	OCR1A = 15624; //Value at which CTC

	while(1)  { }

}

ISR(TIMER1_COMPA_vect)   {  //CHECK arg

	PORTB ^= 1<<PINB0;

}
```