

Counter / Timers

WIE Weekly's Session 2

Datasheet !

<https://tinyurl.com/i-love-arduino>

<https://tinyurl.com/pinmapping>

The problem with delay

- Blocks the code
- Wastes precious computing time
- Can we assign this trivial task to something else?
- Yes! Timers!
- Especially useful for handling multiple tasks
- But what is a timer?

Timers

Timers - The Moms of a μ C

- Timers can interrupt processor to do specific tasks at high priority (like fetching groceries data)
- But we need to configure it first

Types

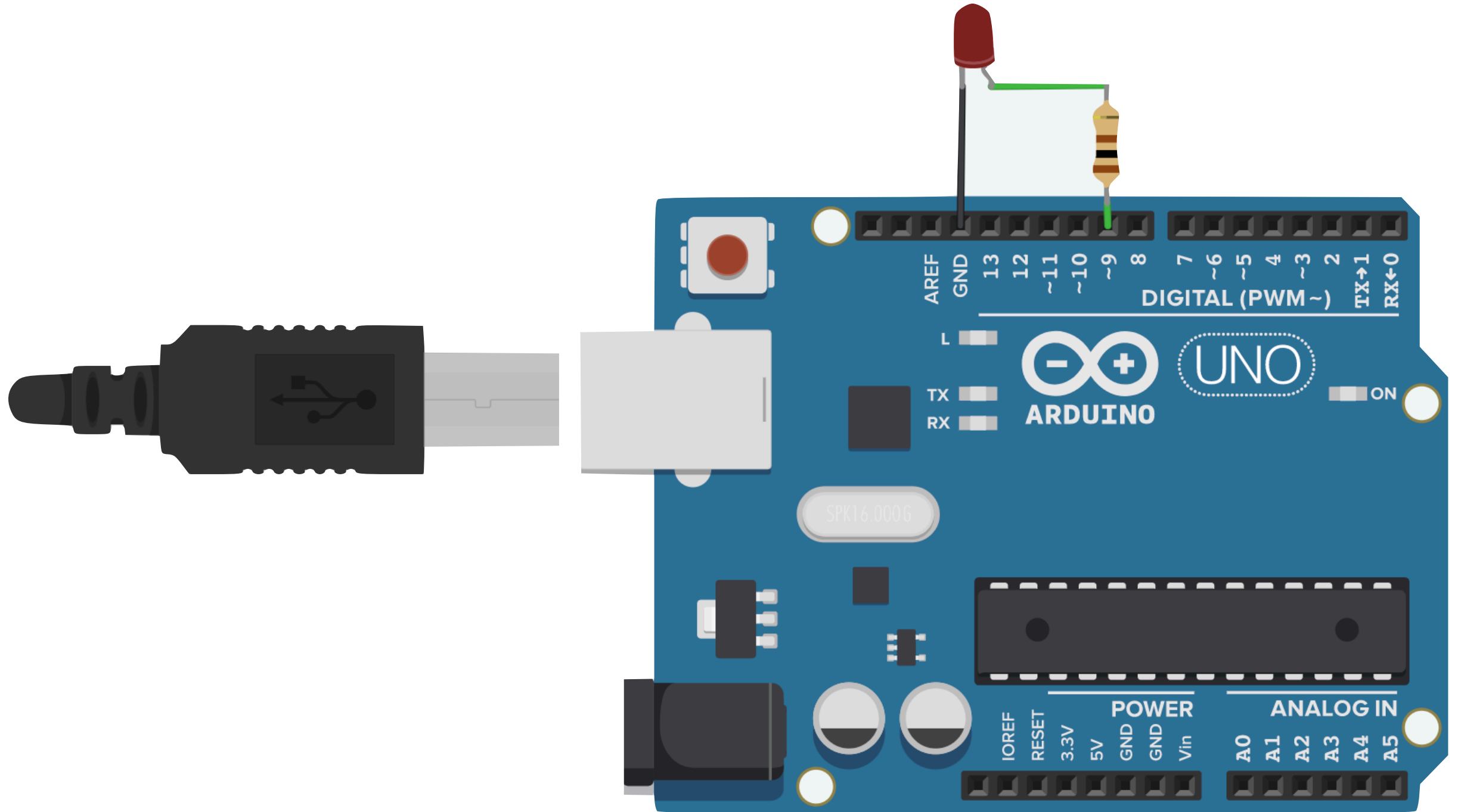
- 2 - 8bit
- 1 - 16bit

Modes

- Normal mode
- CTC mode
- PWM mode

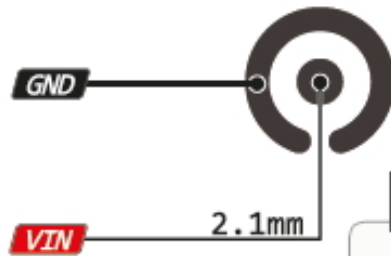
How to Configure?

- TCNT (CURRENT VALUE)
- OC (O/P COMAPRE)
- OCR (O/P COMPARE REG)
- TCCR (CONFIGURATION REG)
 - WG (WAVEFORM GEN)
 - CS (CLOCK SELECT)
 - COM (COMPARE OUTPUT MODE)



UNO PINOUT

7-12V Depending on current drawn

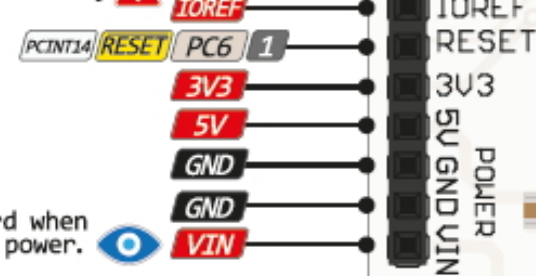


⚠ Absolute MAX per pin 40mA recommended 20mA

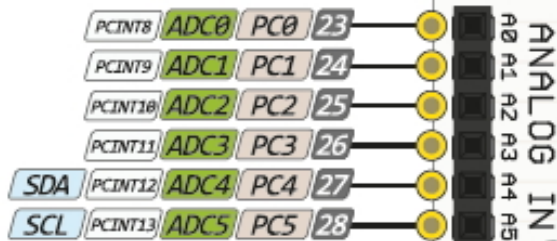
⚠ Absolute MAX 200mA for entire package

IOREF provides a logic reference voltage for shields that use it. It is connected to the 5V bus.

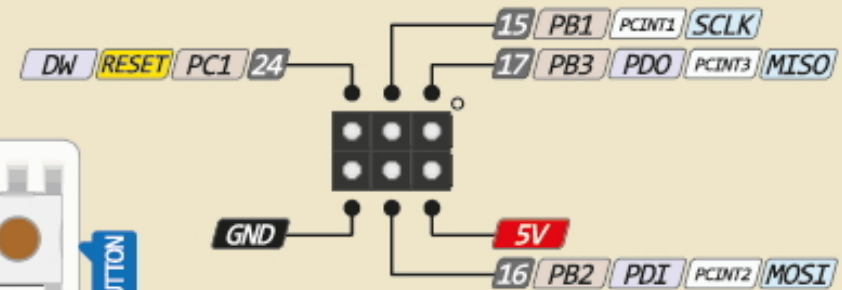
R3 Only ⚠



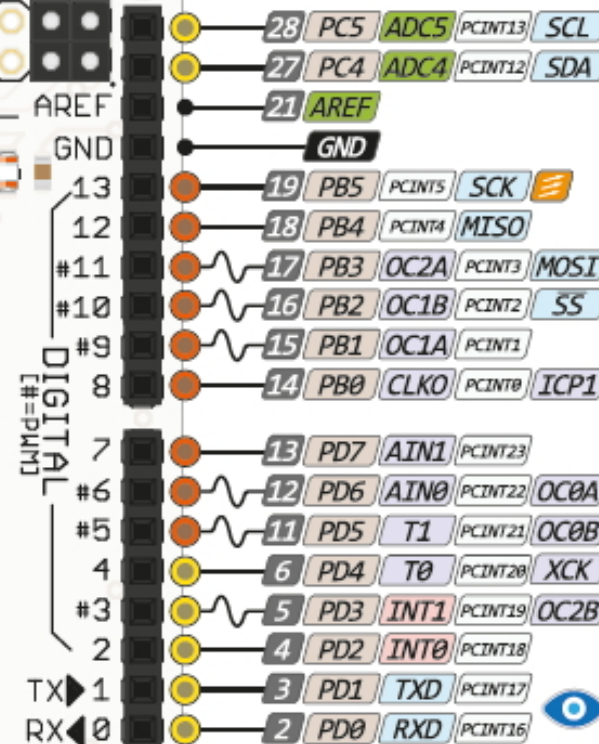
The input voltage to the board when it is running from external power. Not USB bus power.



ATMEGA 82U/16U2 ICSP



Cut to disable autoreset

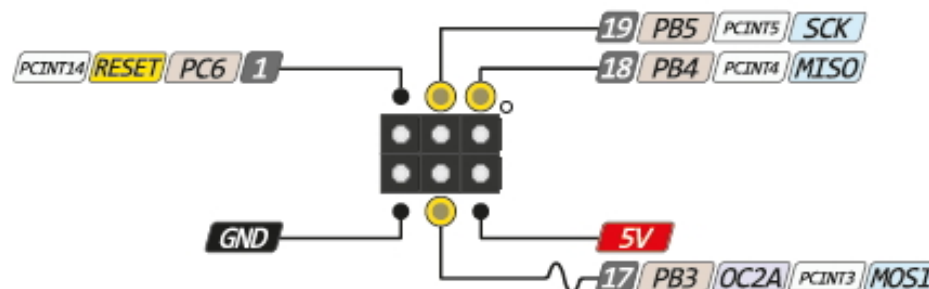


⚠ R3 Only

- Power
- GND
- Serial Pin
- Analog Pin
- Control
- INT
- Physical Pin
- Port Pin
- Pin function
- Interrupt Pin
- PWM Pin
- Port Power ⚠

👁 Connected to the ATmega and used for USB program and communicating with it

⚠ The power sum for each pin's group should not exceed 100mA



1. Normal Mode 😊

```
int main() {  
    cli();  
    DDRB |= 1 << PB1;  
    TCCR1B |= (1 << CS12);  
    TIMSK1 |= 1 << TOIE1;  
    sei();  
    while (1);  
    return 0;  
}
```

```
ISR(TIMER1_OVF_vect) {  
    PORTB ^= (1 << PB1);  
}
```

1. CTC Mode 😊

```
int main(void)  
{  
    DDRB |= (1 << PINB1);  
    TCCR1A |= (0 << WGM11);  
  
    TCCR1B |= (1 << WGM12); //for CTC OCR  
    // TCCR1B |= (1 << WGM13); //for CTC ICR  
    TCCR1B |= (1 << CS12) | (0 << CS10); //256 ps  
    //ICR1 = 50000;  
    TCCR1A |= (1 << COM1A0);  
    OCR1A = 20000;  
  
    while (1);  
    return (0);  
}
```

Can I make a 16-Bit Timer with just
8-Bit Timer ???

```

union reg {
    struct bitsy {
        unsigned int b0: 1;
        unsigned int b1: 1;
        unsigned int b2: 1;
        unsigned int b3: 1;
        unsigned int b4: 1;
        unsigned int b5: 1;
        unsigned int b6: 1;
        unsigned int b7: 1;
    } bits;
    int bytes: 8;
} x;

void init(int ocr) {
    DDRB |= (1 << PB1);
    cli();

    OCR0A = ocr;
    TCCR0A |= (1 << WGM01) | (1 << COM0A0);
    TCCR0B |= (1 << CS02) | (1 << CS00);
    TIMSK0 |= (1 << OCIE0A);
    sei();
}

```

```

void start(int millies) {
    int temp = 0;

    if (millies < 256) {
        init(millies);
    } else {
        temp = ceil(millies / 255.0);
        x.bytes = temp;

        while (x.bytes > 0) {
            init(255);
        }
    }
}

int main() {
    while (1) {
        start(17000);
        PORTB ^= (1 << PB1);
    }
}

ISR(TIMER0_COMPA_vect) {
    x.bytes -= 1;
}

```


**What will be covered in this
workshop**

AVR Coding in its entirety

