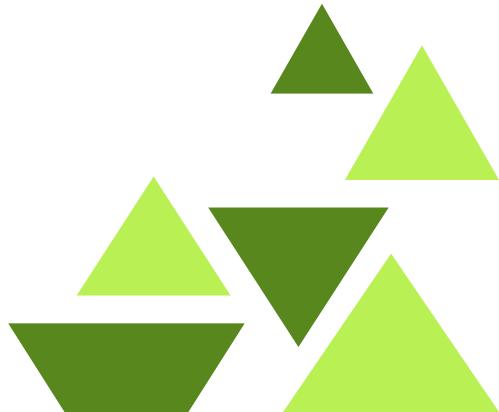




SMART TECHNOLOGY



LECTURE

2

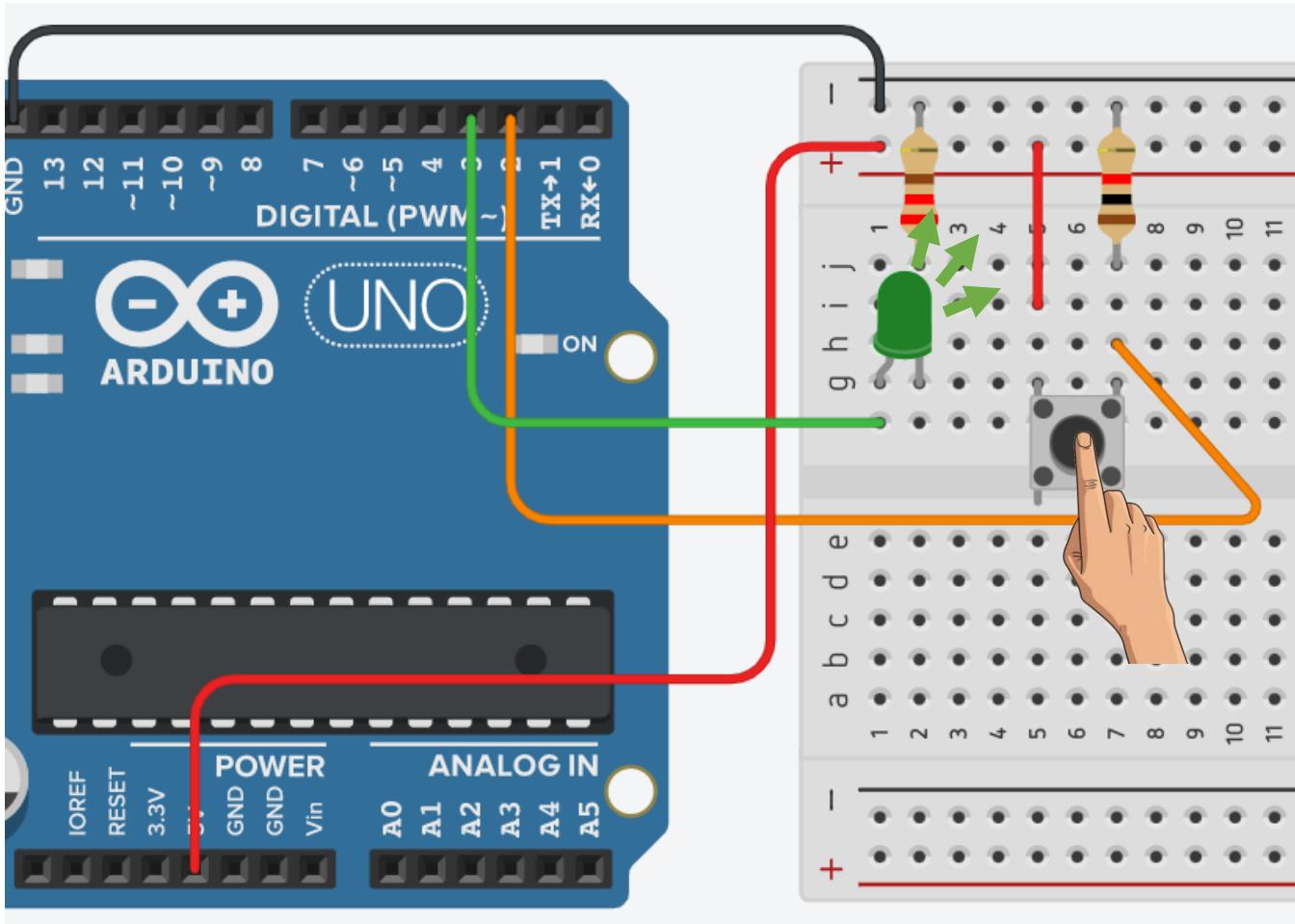
```

int led = 3;
int button = 2;
int Reading = 0;

void setup()
{
    pinMode(led, OUTPUT);
    pinMode(button, INPUT);
}

void loop()
{
    Reading = digitalRead(button);
    if (Reading == HIGH) { digitalWrite(led, HIGH); }
    else { digitalWrite(led, LOW); }
}

```



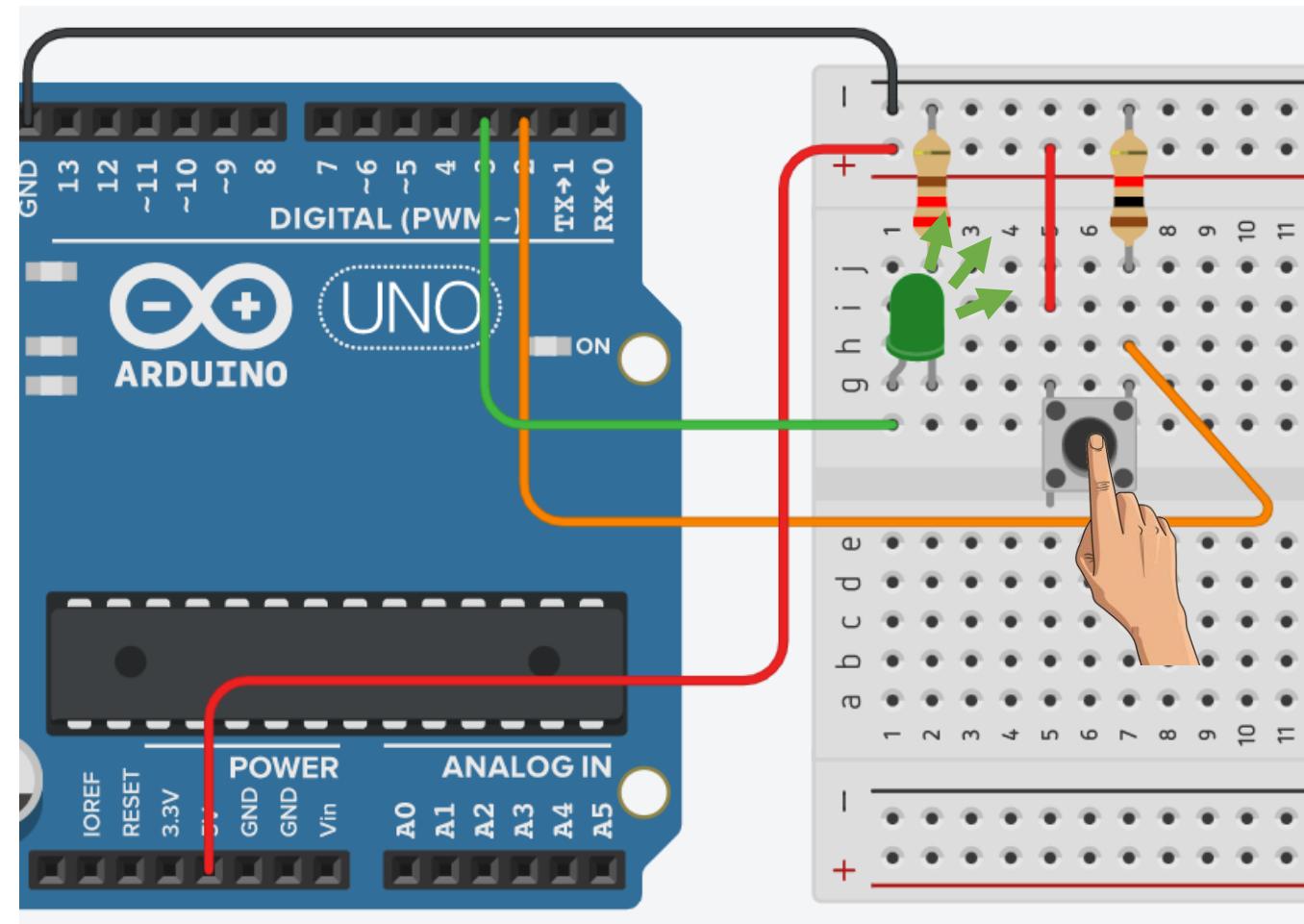
**Reading
LOW**

```

int led = 3;
int button = 2;
int Reading = 0;
int counter = 0;
void setup()
{
    pinMode(led, OUTPUT);
    pinMode(button, INPUT);
}

void loop()
{
    Reading = digitalRead(button);
    if (Reading == HIGH)
    {
        counter++;
        delay(250);
        if(counter==1)
        { digitalWrite(led, HIGH);}
        else
        { digitalWrite(led, LOW); counter=0; }
    }
}

```

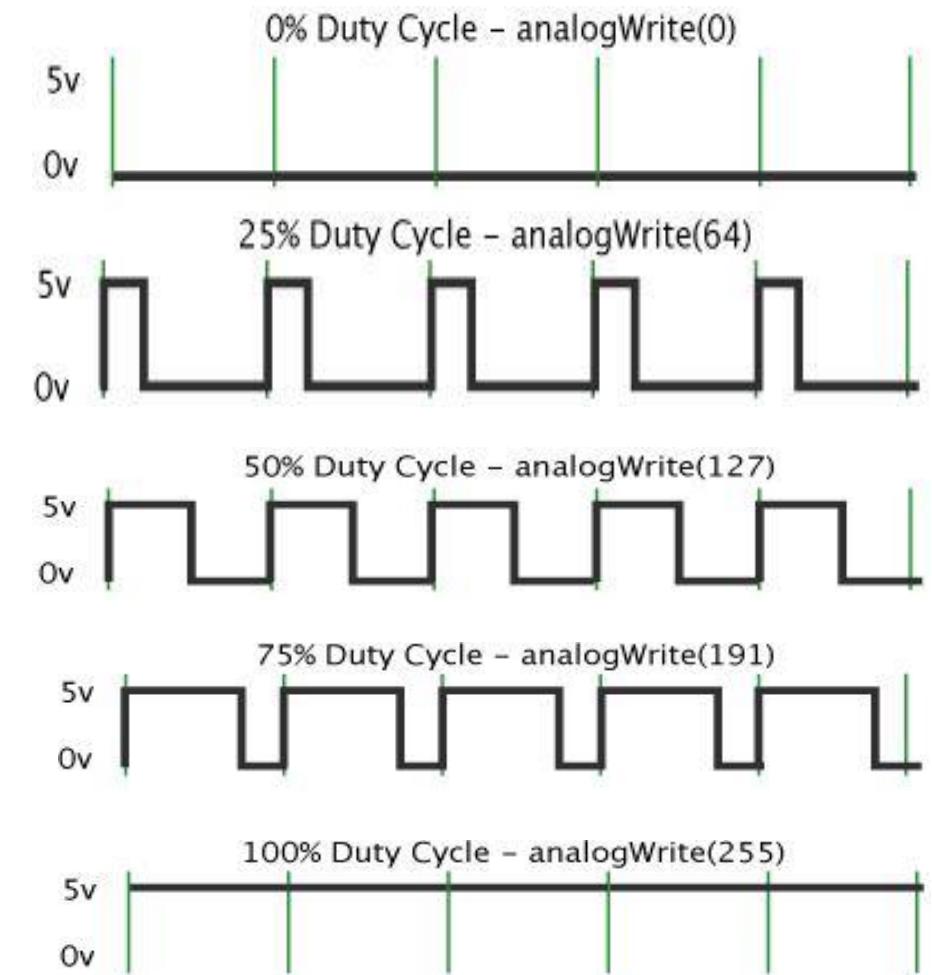


counter



• Pulse Width Modulation PWM

- PWM frequency at about 500Hz, the green lines would measure 2 milliseconds each.
- A call to `analogWrite()` is on a scale of 0 – 255 → MAX Value(255)
- `analogWrite(255)` requests a 100% duty cycle (always on)
- and `analogWrite(127)` is a 50% duty cycle (on half the time) for example.



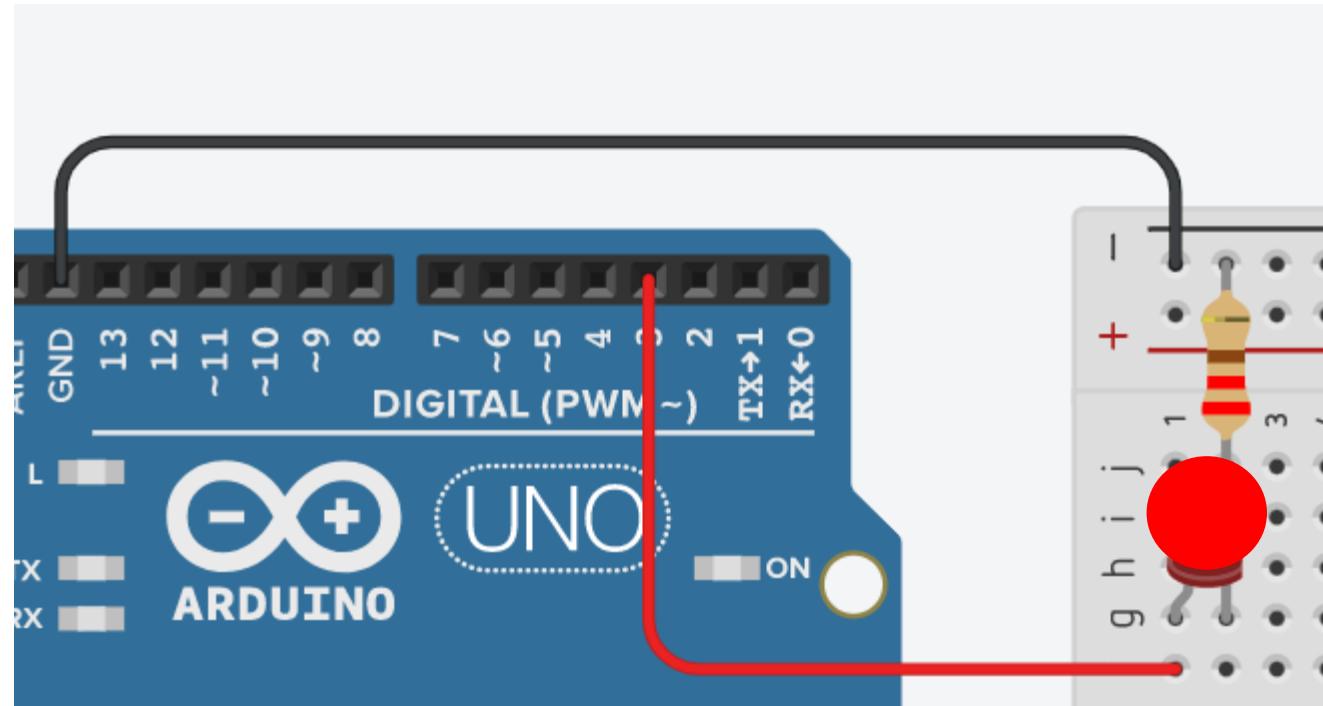
• Example

```

int led = 3;

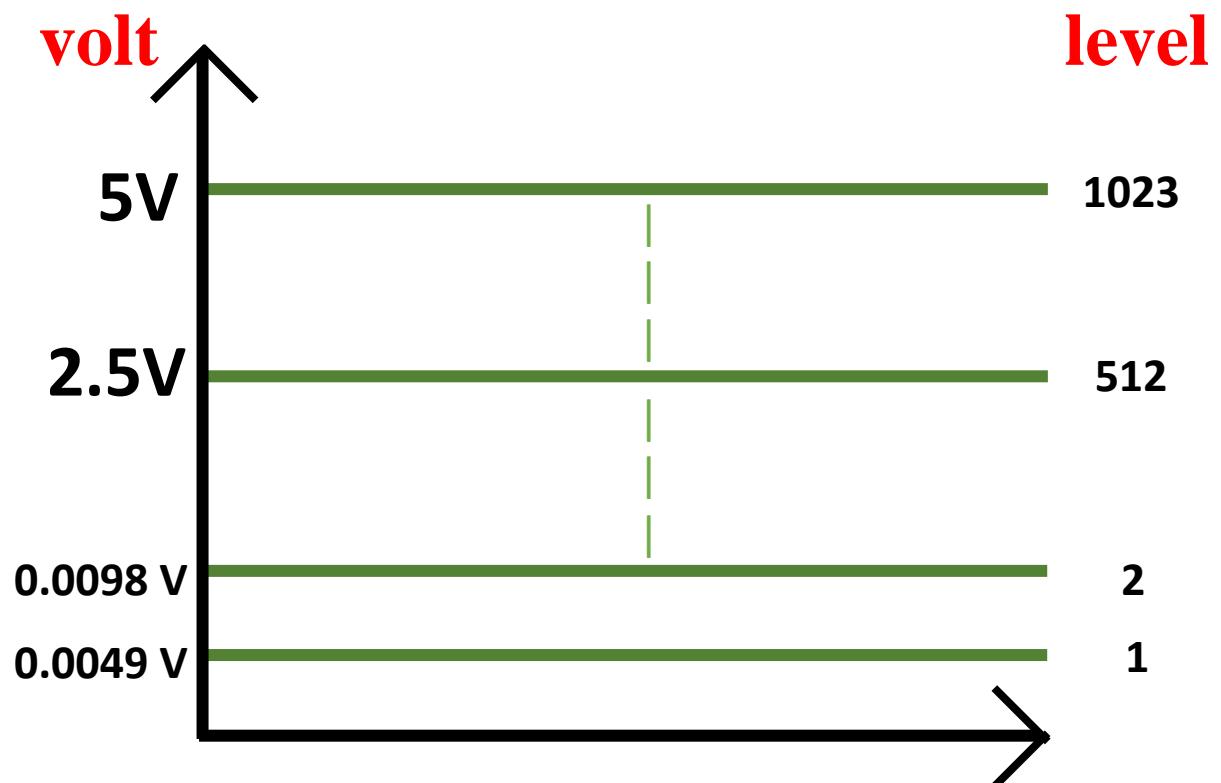
void setup()
{
    pinMode(led, OUTPUT);
}

void loop()
{
    analogWrite(led, 0);
    delay(1000);
    analogWrite(led, 65);
    delay(1000);
    analogWrite(led, 128);
    delay(1000);
    analogWrite(led, 255);
    delay(1000);
}
    
```



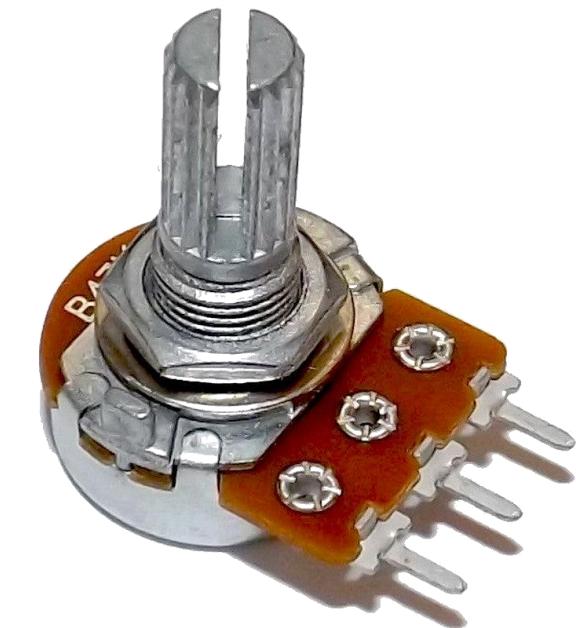
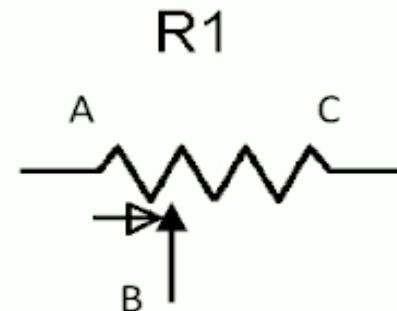
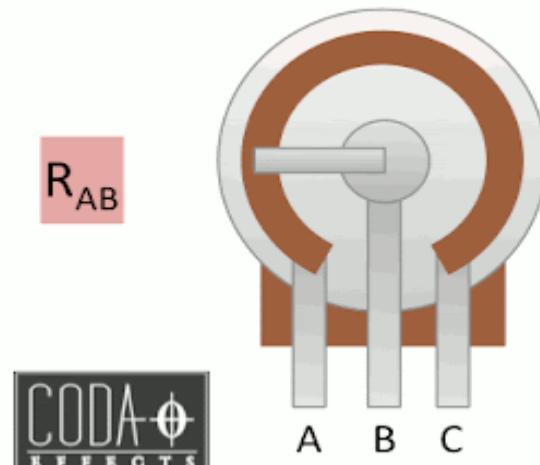
- ADC (analog to digital converter)

- 5V divided to 1024 level between 0 and 1023
- (4.9 mV) per unit

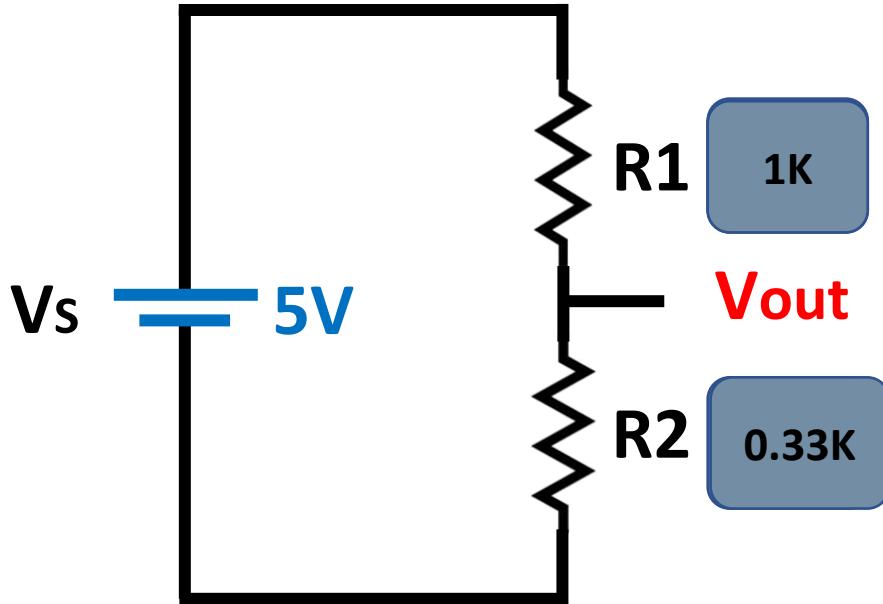


• What is potentiometer

- Potentiometer or “POT” is a two variable resistance that can change its resistance value by knob



• Voltage Divider



$$V_{out} = V_s \times \frac{R_2}{R_1 + R_2}$$

$$V_{out} = 5 \times \frac{1}{1+1} = 2.5 \text{ V}$$

$$V_{out} = 5 \times \frac{3}{1+3} = 3.75 \text{ V}$$

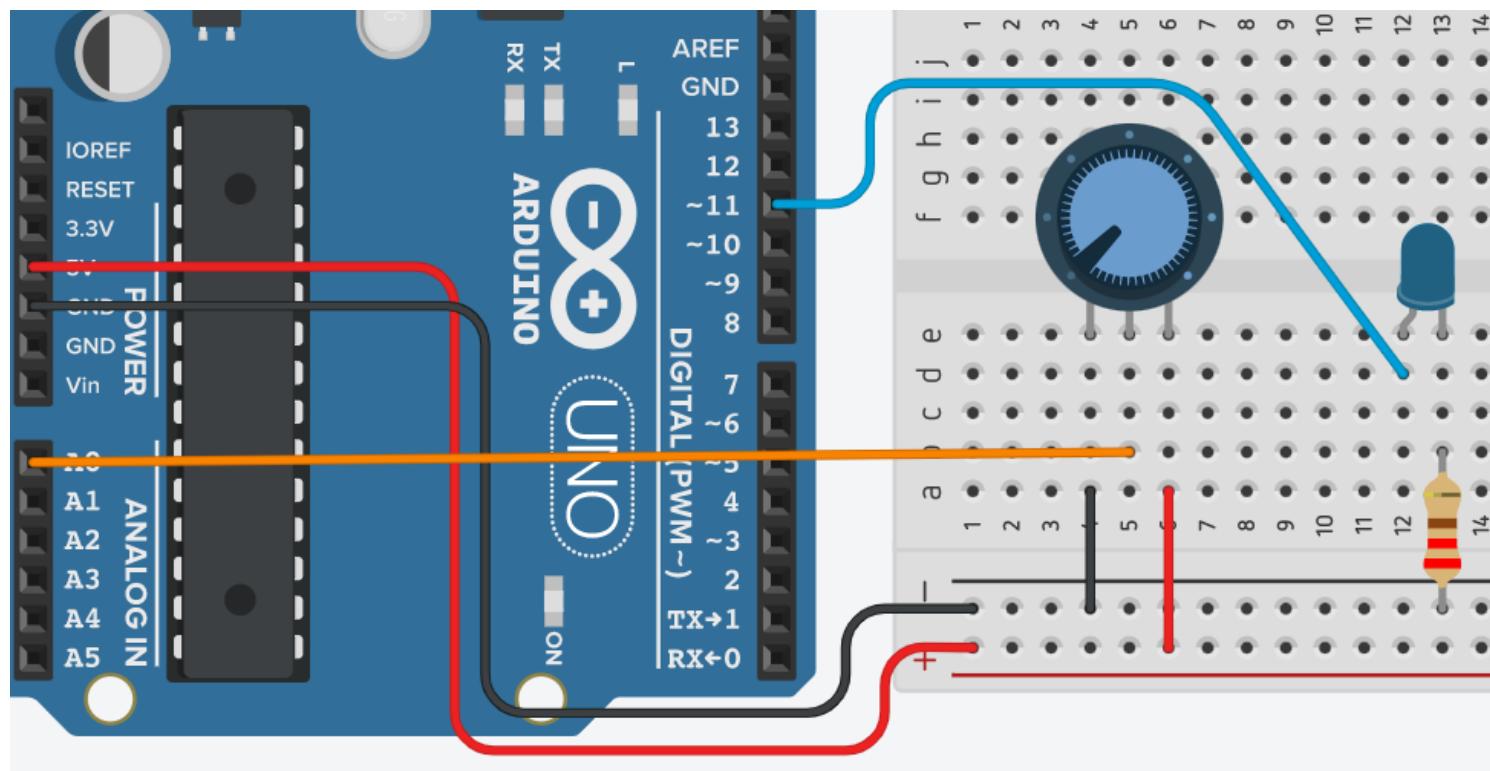
$$V_{out} = 5 \times \frac{0.33}{1+0.33} = 1.24 \text{ V}$$

• Flasher Control

```

int Pot = A0;
int Led = 11;
int sensorValue=0;
void setup()
{
    pinMode(Led, OUTPUT);
    pinMode(Pot , INPUT);
}

void loop()
{
    sensorValue = analogRead(Pot);
    digitalWrite(Led, HIGH);
    delay(sensorValue);
    digitalWrite(Led, LOW);
    delay(sensorValue);
}
    
```



• Brightness Control

```

int Pot = A0;
int Led = 11;
int POT_Reading = 0;

void setup()
{
    pinMode(Pot, INPUT);
    pinMode(Led, OUTPUT);
}

void loop()
{
    POT_Reading = analogRead(Pot);
    analogWrite(Led, POT_Reading);
}
    
```

analogWrite(Led, 0→255);



POT_Reading = 0→255 

POT_Reading = 256→1023 

Over Flow

256 → 0

257 → 1

258 → 2

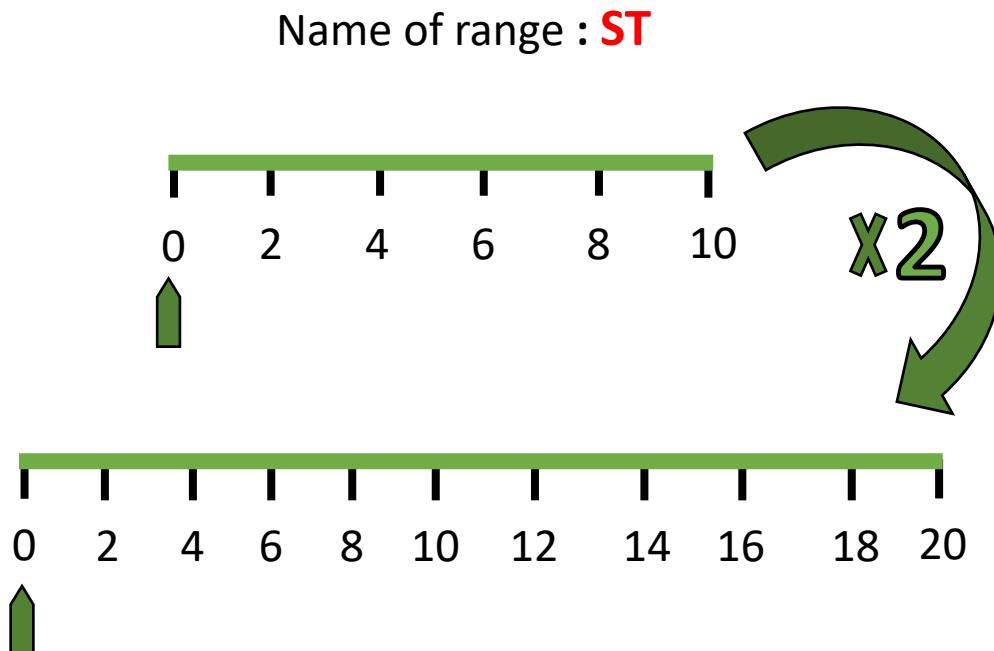
• Over Flow

```
byte x;  
void setup()  
{  
    Serial.begin(9600);  
    x=0;    Serial.println(x);  
    x=100;  Serial.println(x);  
    x=200;  Serial.println(x);  
    x=255;  Serial.println(x);  
    x=256;  Serial.println(x);  
    x=257;  Serial.println(x);  
    x=1000; Serial.println(x);  
}  
  
void loop()  
{  
}
```

COM7

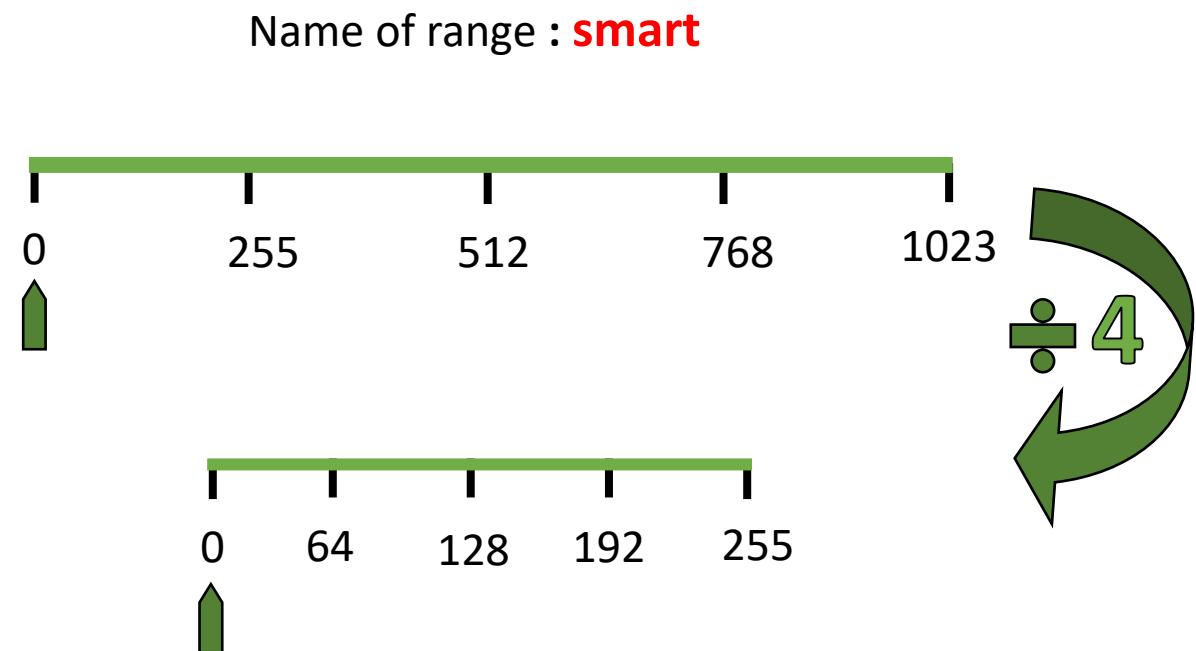
0
100
200
255
0
1
232

• Map Function



Map (value , fromLow, fromHigh , toLow , toHigh)

Map (ST , 0 , 10 , 0 , 20)



Map (value , fromLow, fromHigh , toLow , toHigh)

Map (Smart , 0 , 1023 , 0 , 255)

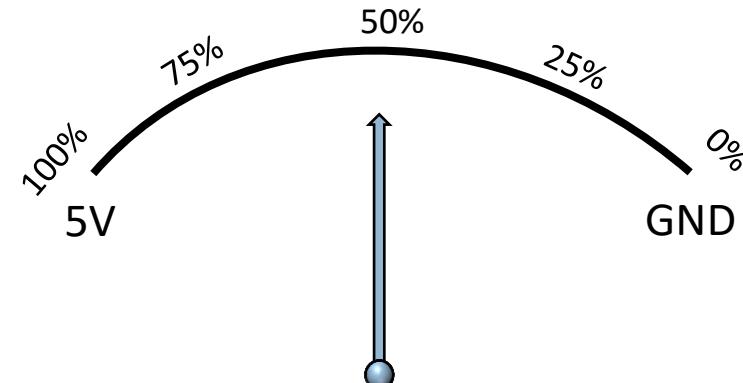
Brightness Control

```

int Pot = A0;
int Led = 3;
int ledBrightness = 0;
int sensorValue = 0;

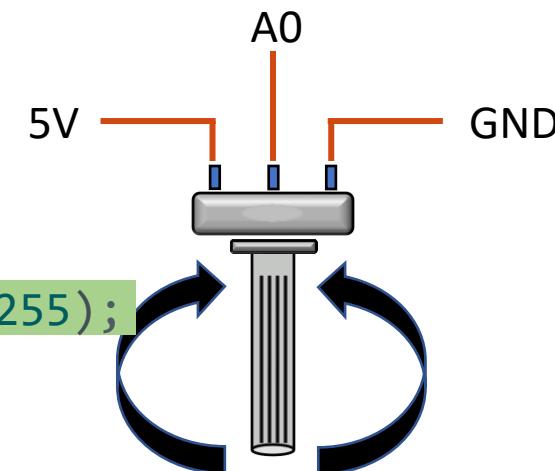
void setup()
{
    pinMode(Pot, INPUT);
    pinMode(Led, OUTPUT);
}

void loop()
{
    sensorValue = analogRead(Pot);
    ledBrightness = map(sensorValue, 0, 1023, 0, 255);
    analogWrite(Led, ledBrightness);
}
    
```



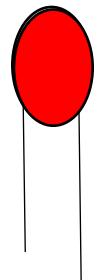
sensorValue

1023



ledBrightness

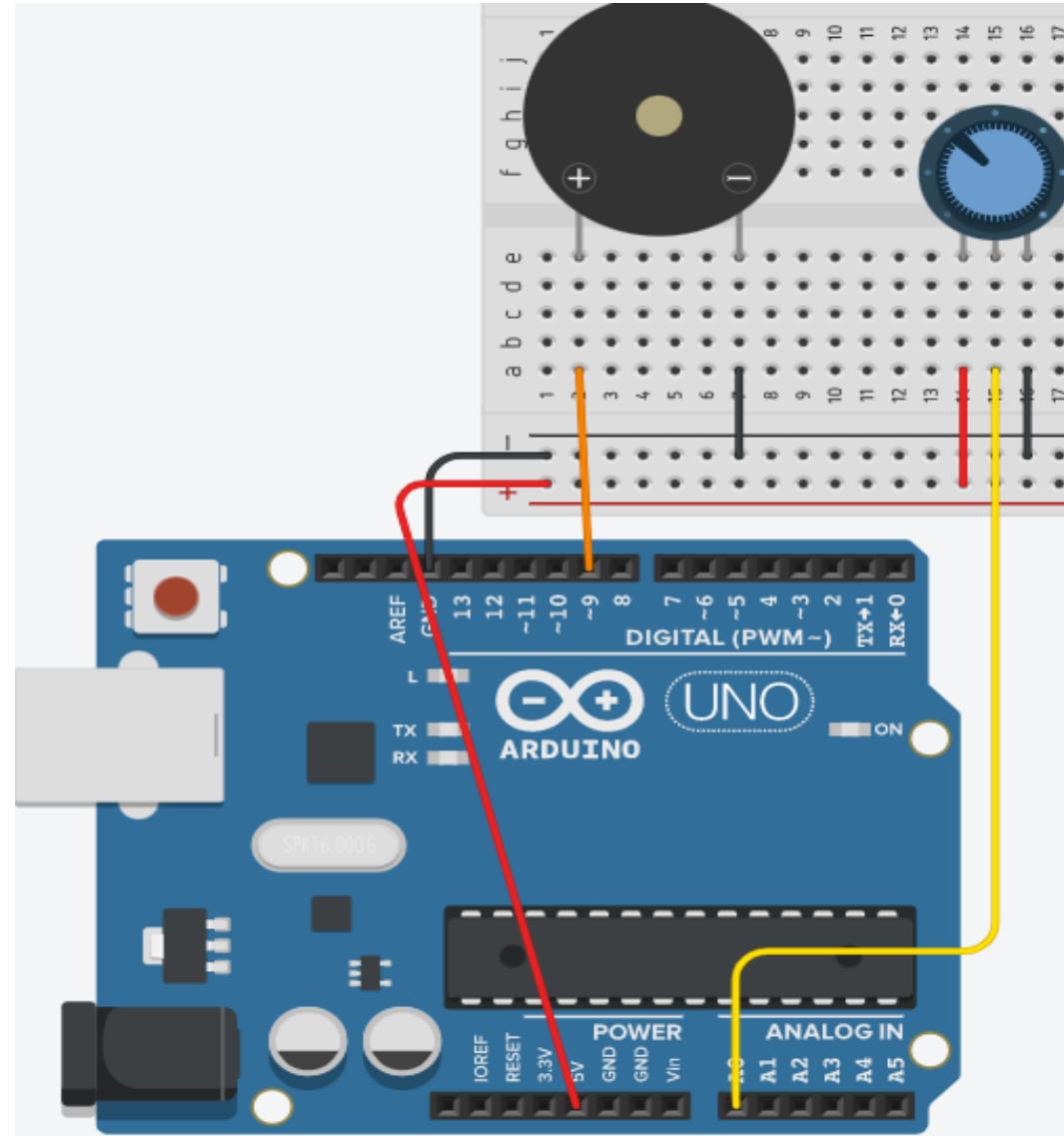
255



- Task

- Control buzzer tones with POT

Hint you can use **Tone()** function google it



**THANKS
FOR
COMING**

