



جامعة محمد بن زايد
للذكاء الاصطناعي
MOHAMED BIN ZAYED UNIVERSITY
OF ARTIFICIAL INTELLIGENCE

Introduction to Arduino Digital & Analog Input-Output

MBZUAI Robotics Club

Workshops Plan

- Workshop #1:
 - Introduction to Arduino.
 - Introduction to electric circuits.
 - Mini-projects: LED, Motor, Digital read/write, Analog read/write.
- Workshop #2:
 - Running servo motors.
 - Reading sensors: ultrasonic range sensor.
 - Using LCD displays.
 - Controlling a servo with a joystick.

Workshop Outline

- A brief about the Arduino (GPIO, Digital, Analog).
- Electric circuits (Ohm's law).
- The Arduino kit.
- *Mini-project:* Blink.
- *Mini-project:* Program a Button.
- *Mini-project:* Analog, Read Serial.
- *Mini-project:* Analog In, Out Serial.

Workshop Outcomes

By the end of this workshop, you be able to program an Arduino and use digital and analog GPIO pins.

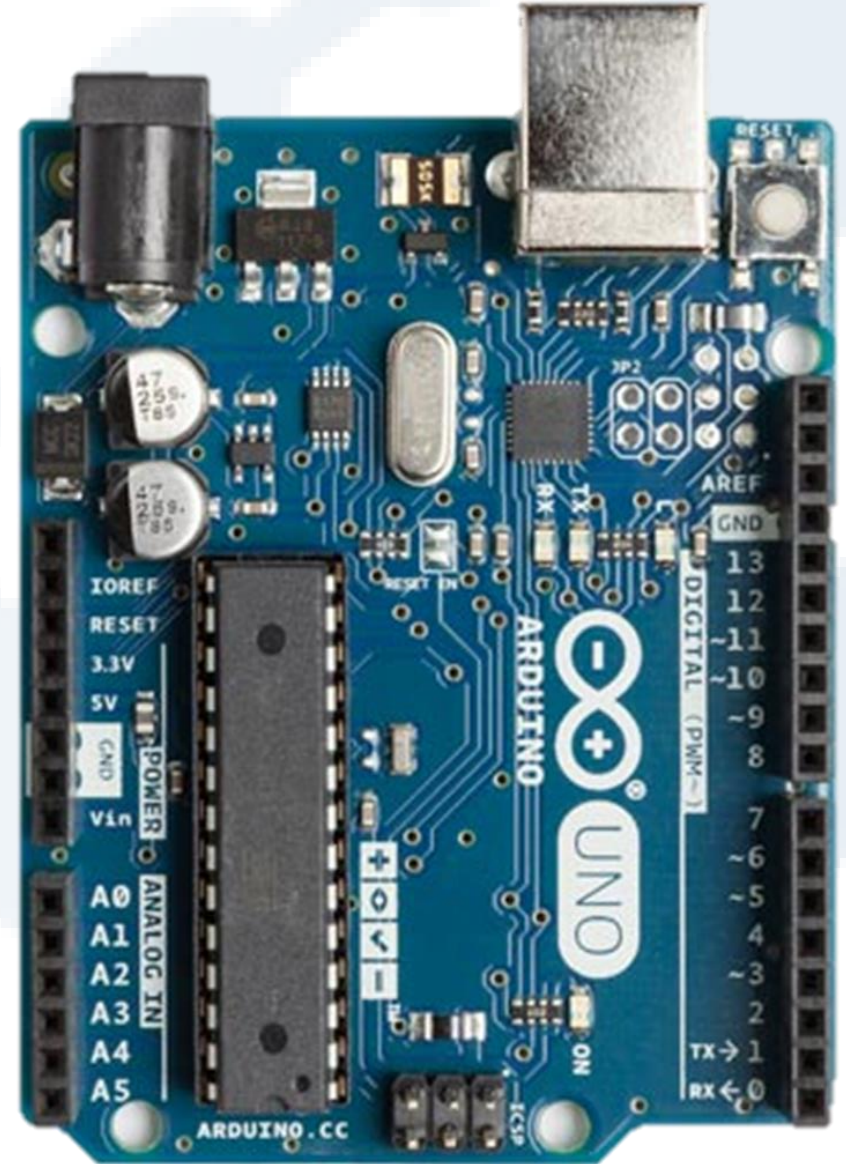
- Get familiar with the Arduino hardware.
- Learn the basics of electric circuits.
- Learn to program an Arduino using the IDE.
- Learn the basics of digital input and output.
- Learn the basics of analog input and output.

Arduino

Arduino is an open-source electronics platform

Open source and extensible software - The Arduino software is published as open-source tools, available for extension by experienced programmers.

Open source and extensible hardware - The plans of the Arduino boards are published under a Creative Commons license

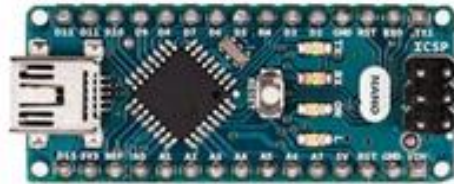


Arduino

UNO



NANO



MEGA 2560



[Other hardware](#)

PROJECT HUB

ADD PROJECT

SEARCH PROJECTS

All products ▾

All categories ▾

Trending ▾

Any difficulty ▾

Any type ▾



LED Tennis

Project showcase by Johan

914 VIEWS 0 COMMENTS 2 RESPECTS



How to monitor a beehive with Arduino Nano 33BLE (bluetooth)

Project tutorial by 5 developers

3,605 VIEWS 2 COMMENTS 15 RESPECTS



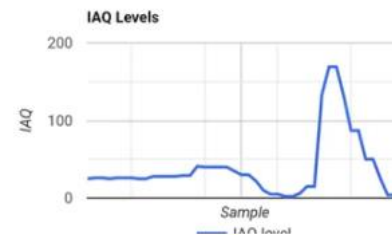
PCB Heart Necklace

Project tutorial by Arnov and Shreyas

335 VIEWS 0 COMMENTS 5 RESPECTS

Air Quality Reader

CO2 433 ppm IAQ 0 (0-500)



Arduino Nicla Air Quality app

Project tutorial by Justin Lutz

1,066 VIEWS 0 COMMENTS 1 RESPECT



Indoor Plant Watering

Project tutorial by dhruvaV

1,291 VIEWS 0 COMMENTS 6 RESPECTS



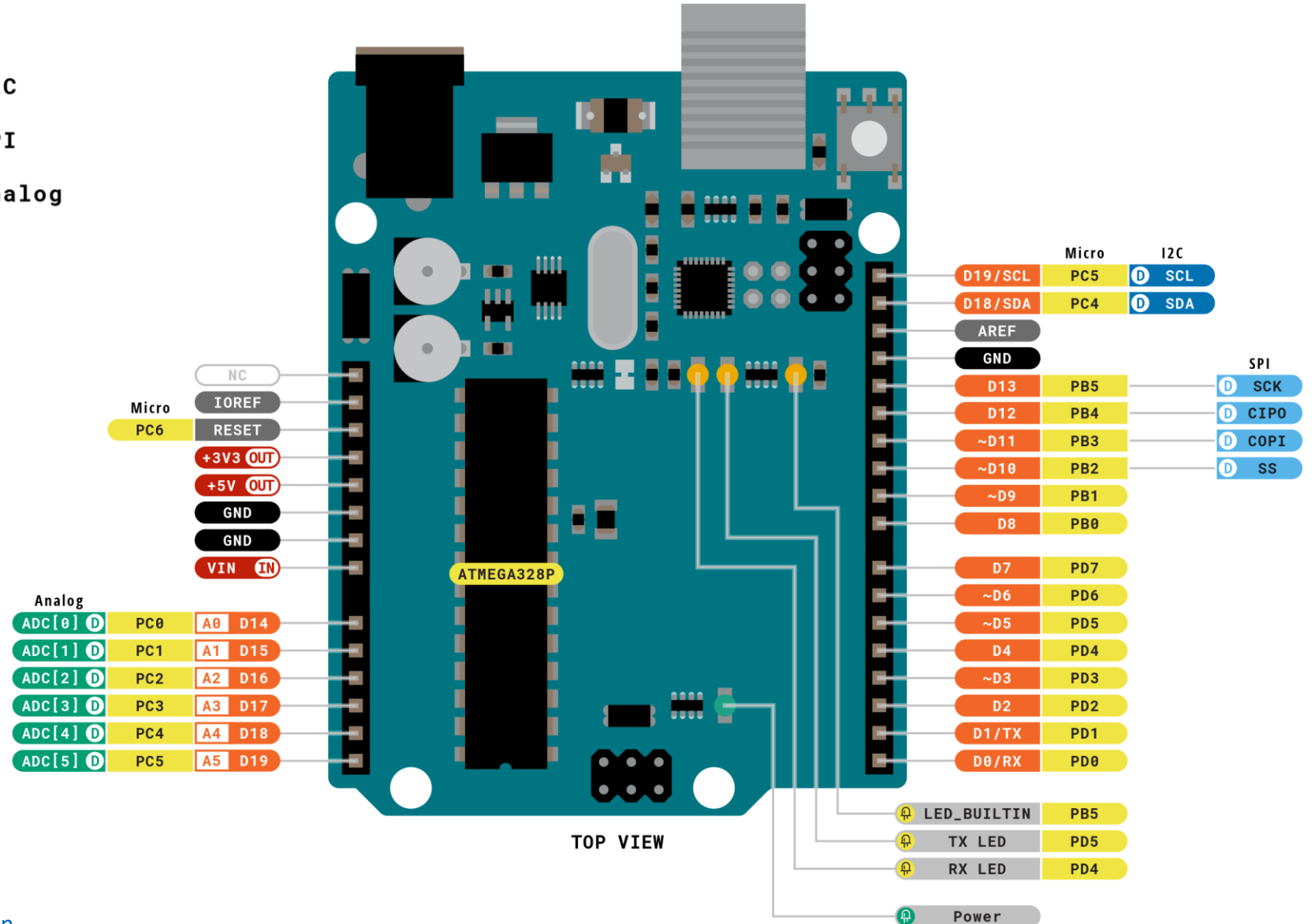
Bee-Happy - Connect

Project tutorial by 4 developers

430 VIEWS 0 COMMENTS 2 RESPECTS

General Purpose Input-Output (GPIO) Pins

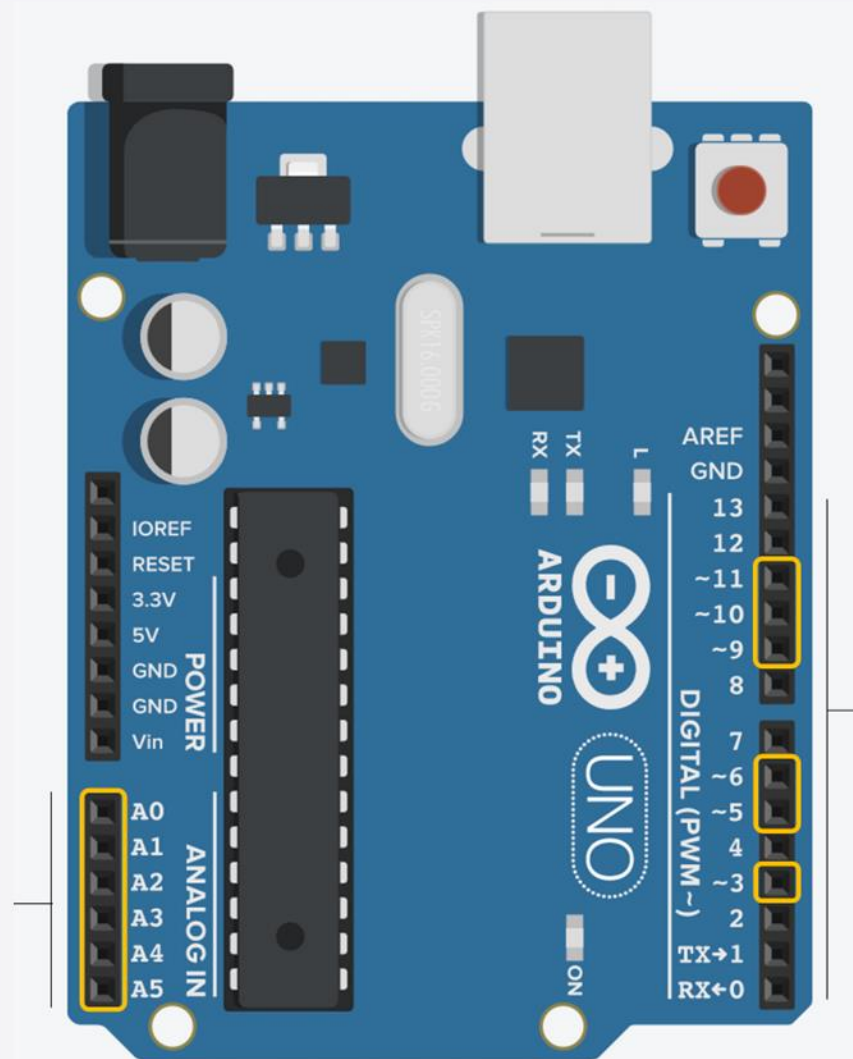
Legend:



General Purpose Input-Output (GPIO) Pins

Six Analog Input Pins

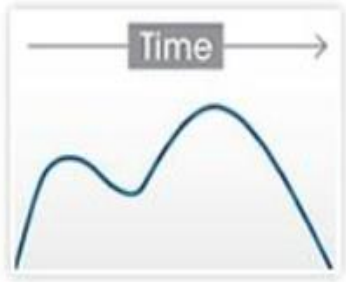
The Arduino Uno has **6** analog **input** pins, which can read a voltage signal between 0-5V using the `analogRead` command



Six Analog Output Pins

The Arduino Uno has **6** analog **output** pins indicated by the tilde ~, which can output a voltage signal between 0 and 5V using PWM via the `analogWrite` command

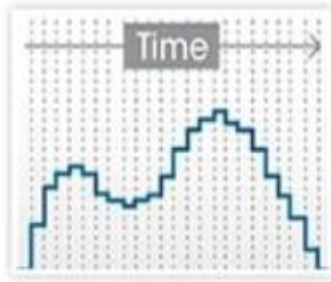
Digital vs. Analog Pins



[original]



[24bit]



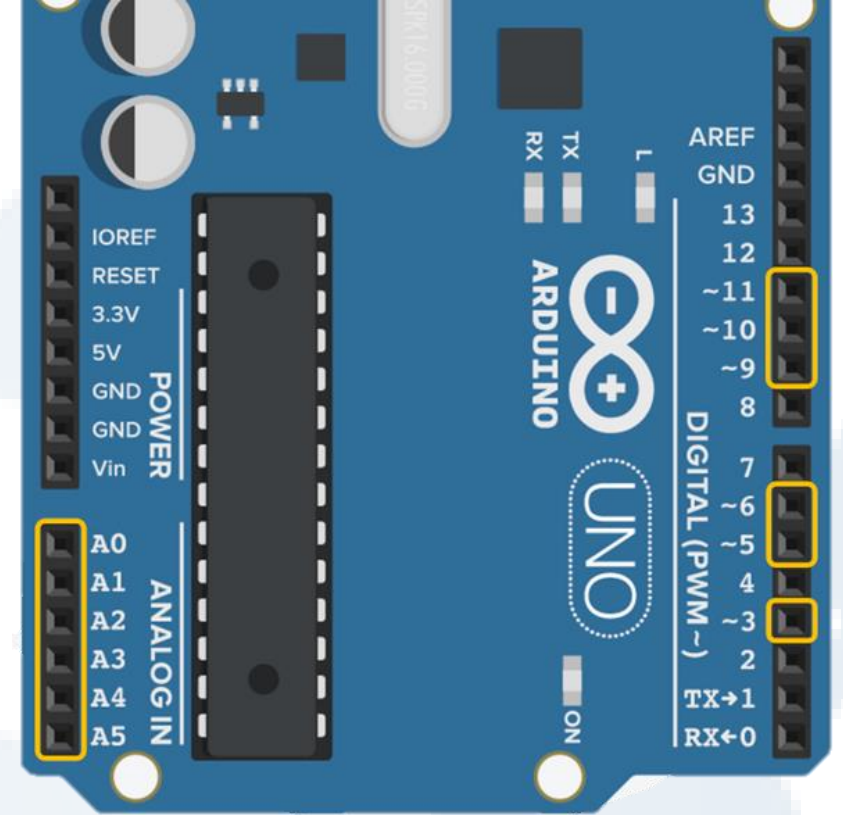
[32bit]

```
/*Analog*/
```

```
sensor_value = analogRead(in_pin_num);  
output_value = map(sensor_value, 0, 1023, 0, 255);  
analogWrite(out_pin_num, output_value); /*0 to 255*/
```

```
/*Digital*/
```

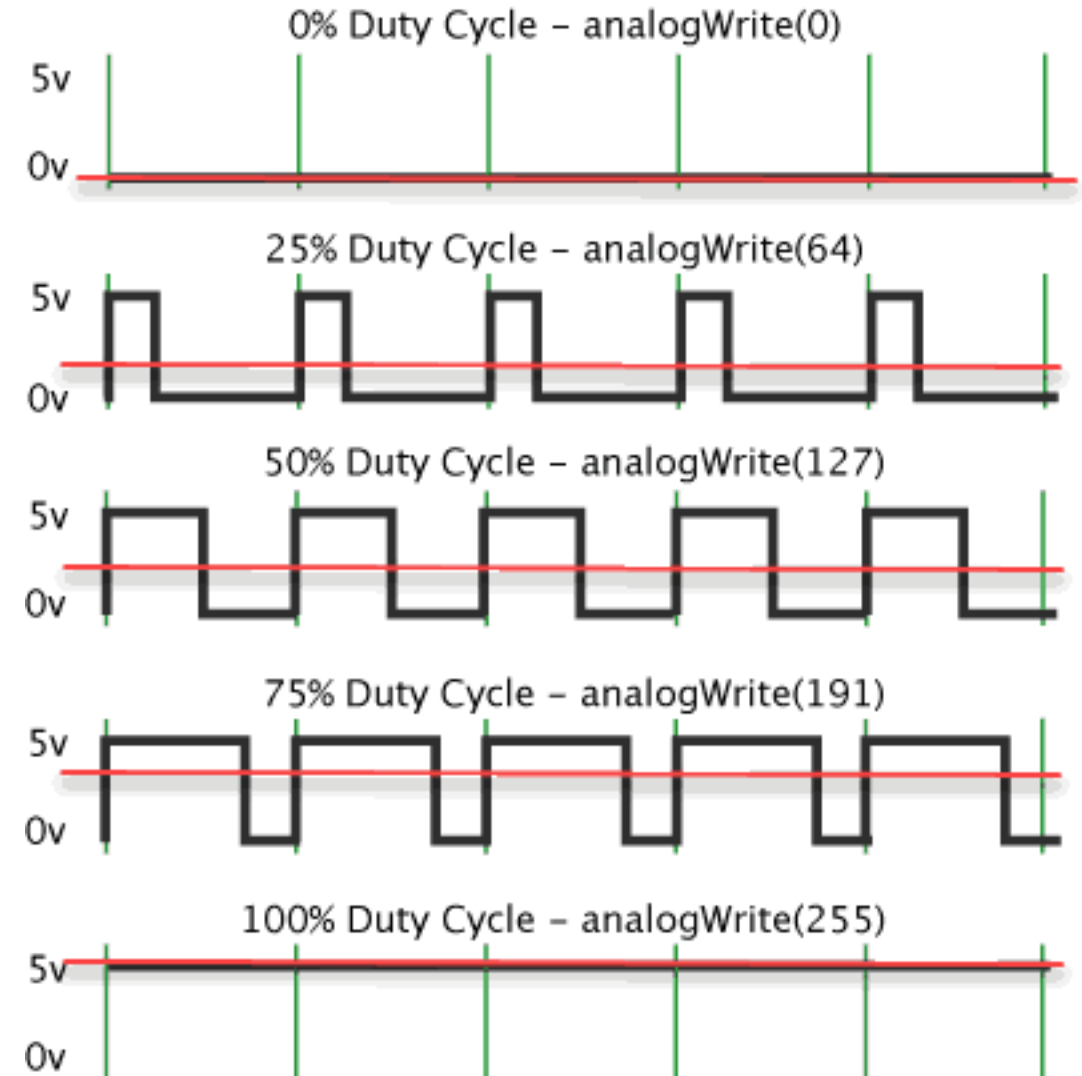
```
button_state = digitalRead(in_pin_num);  
digitalWrite(out_pin_num, 0); /*or 1*/
```



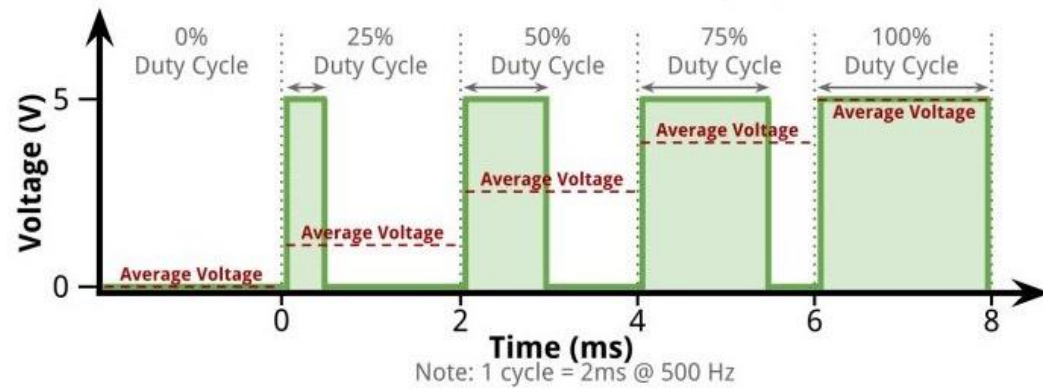
Pulse Width Modulation (PWM)

How does a **digital** device read and write (control) other **analog** devices like dimmable lights and temperature sensors.

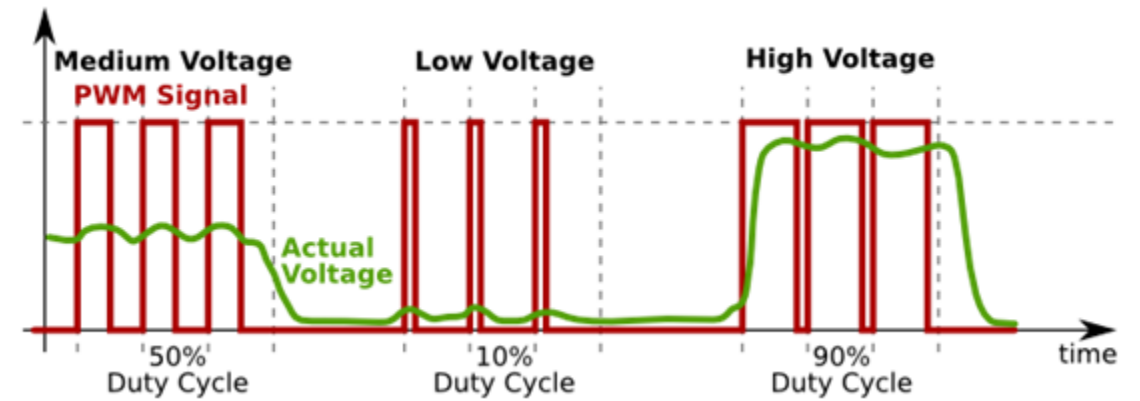
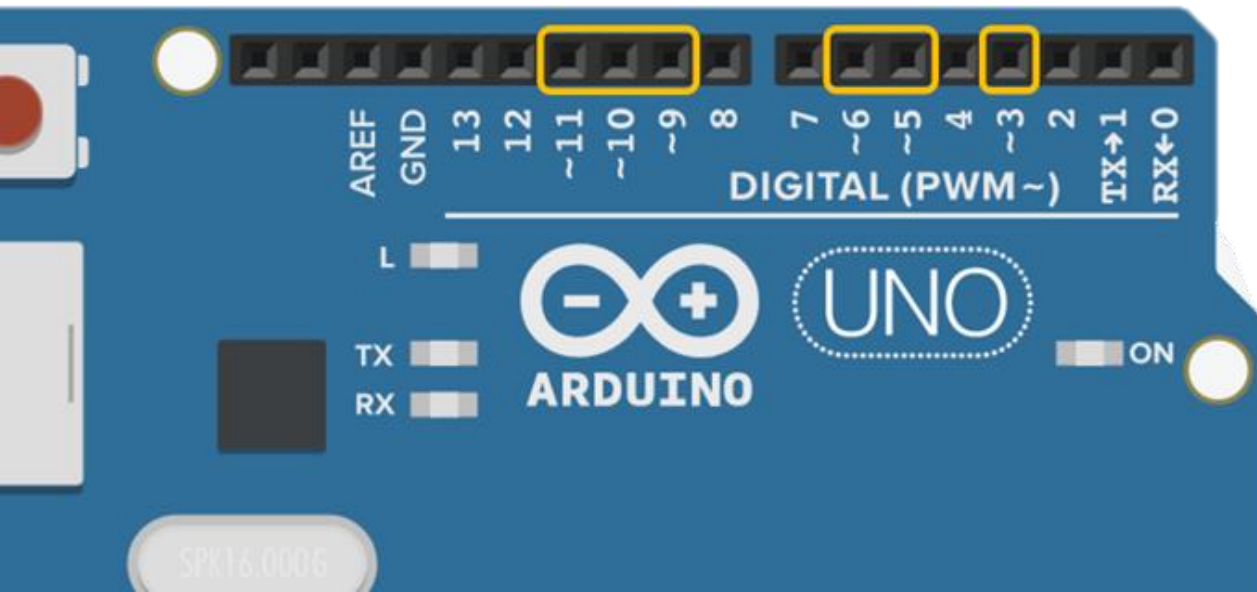
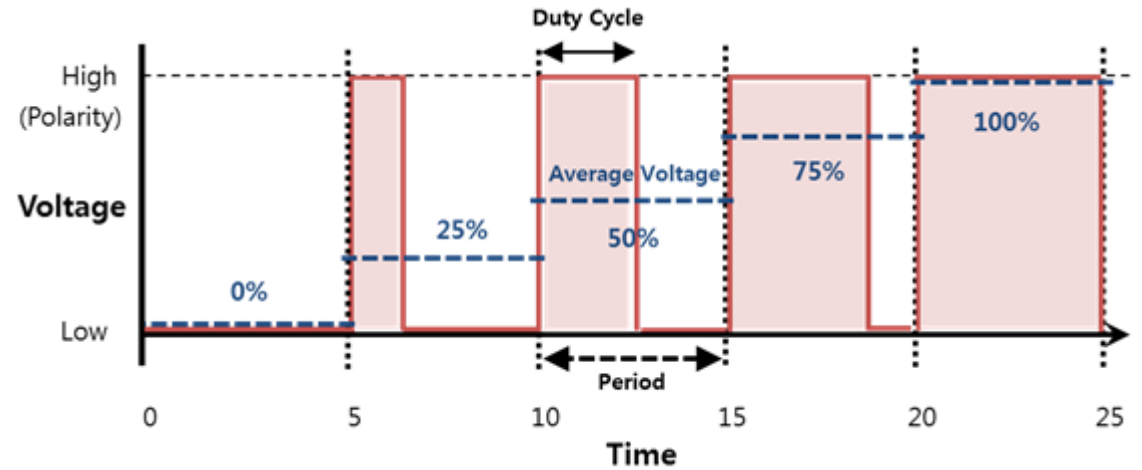
PWM is **not** true analog output, however, PWM *“fakes”* an analog-like result by applying power in pulses, or short bursts of regulated voltage.



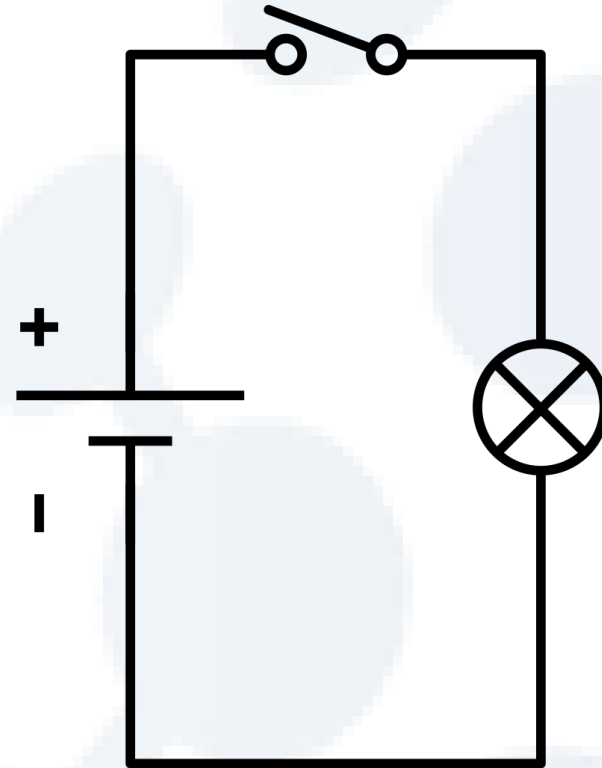
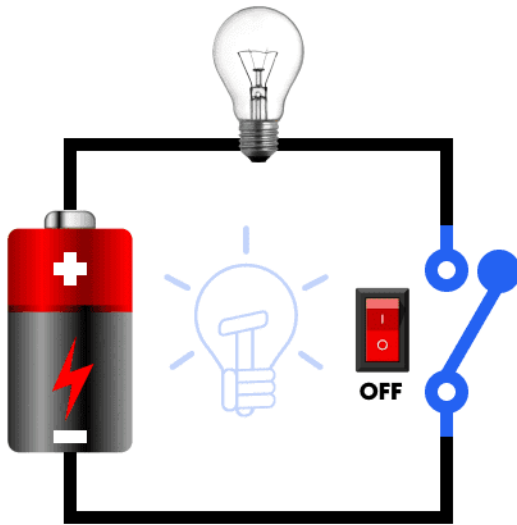
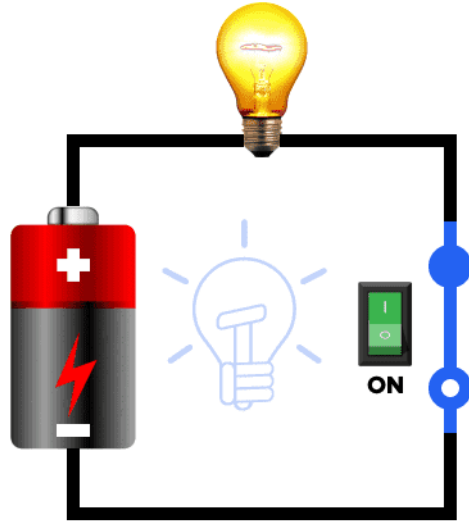
Pulse Width Modulation (PWM)





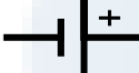



Average output voltage is proportional to duty cycle ON time.



Electric Circuits



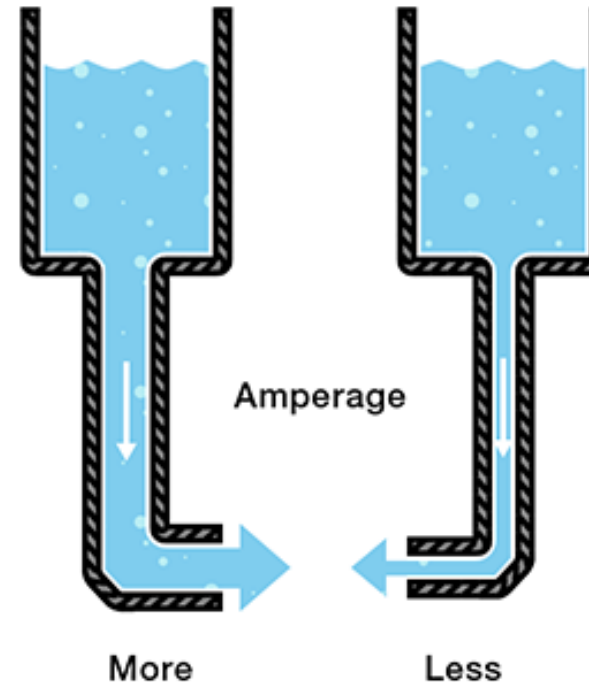
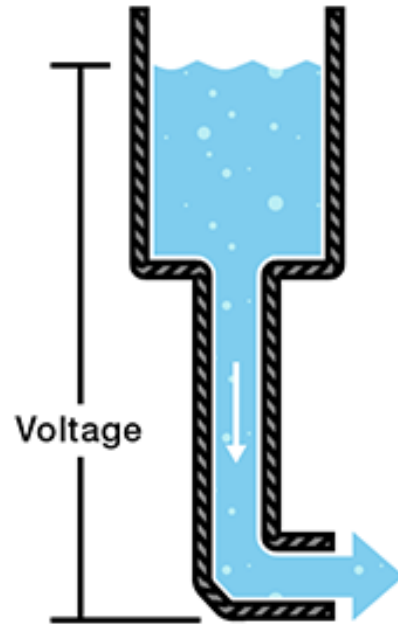
-  Diode
-  Capacitor
-  Inductor
-  Resistor
-  DC voltage source
-  AC voltage source

Ohm's Law

- **Voltage** is the difference in charge between two points.
- **Current** is the rate at which charge is flowing.
- **Resistance** is a material's tendency to resist the flow of charge (current).

$$V = I \cdot R$$

$$I = \frac{V}{R}$$

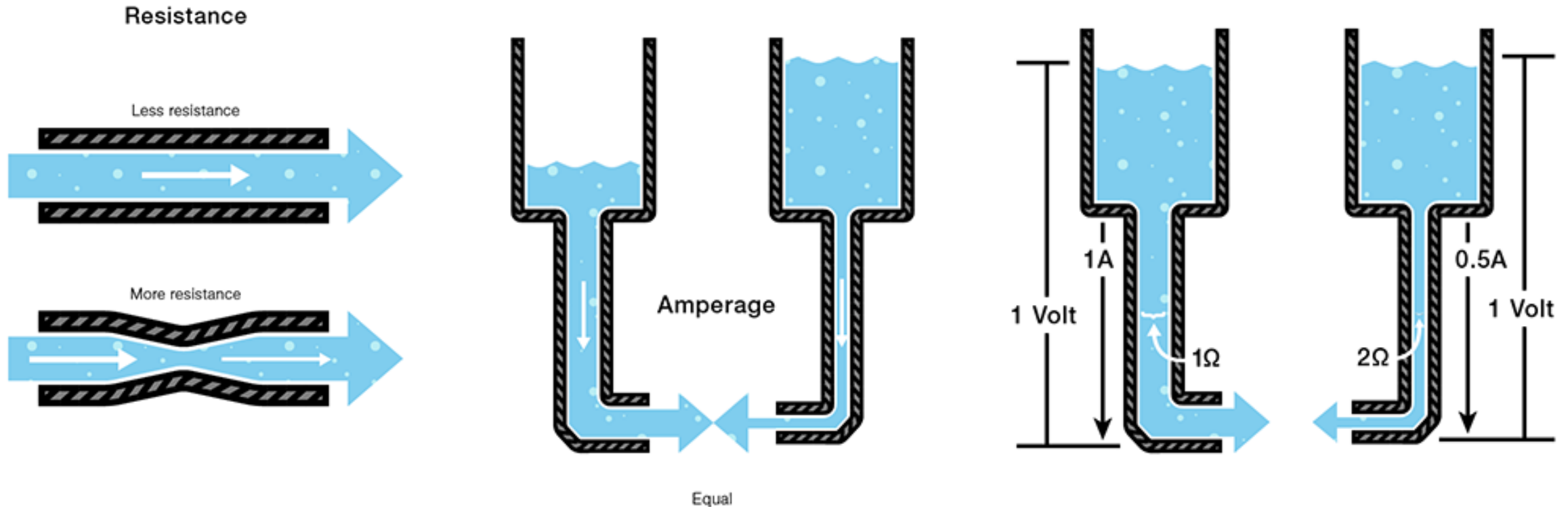


Ohm's Law

- **Voltage** is the difference in charge between two points.
- **Current** is the rate at which charge is flowing.
- **Resistance** is a material's tendency to resist the flow of charge (current).

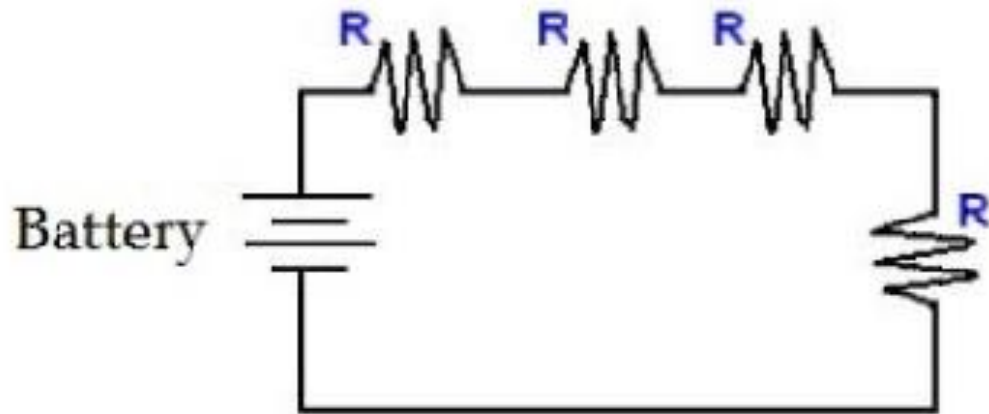
$$V = I \cdot R$$

$$I = \frac{V}{R}$$



Electric Circuits: Series

Series Connection



In **series**, the **current is the same** through each of the components, while **the voltages add up** to the circuit voltage.

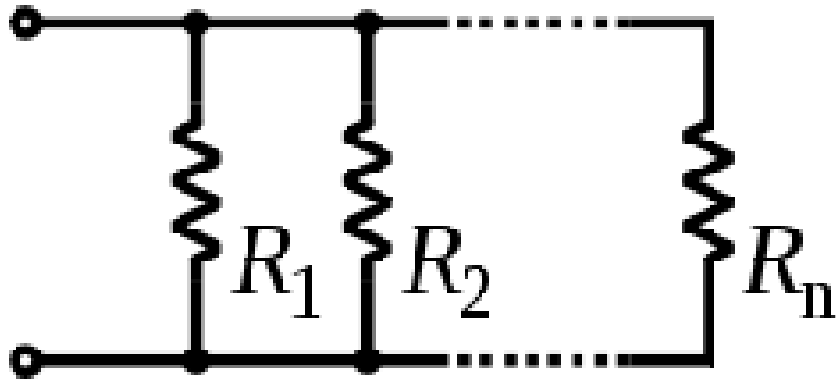
$$\text{Total Resistance (R)} = R1 + R2 + R3 \dots$$

$$\text{Inductors (L)} = L1 + L2 + L3 \dots$$

$$\frac{1}{\text{Capacitors (C)}} = \frac{1}{c1} + \frac{1}{c2} + \frac{1}{c3} \dots$$

Electric Circuits: Parallel

Parallel Series



In **parallel**, the **voltage is the same** across each of the components, while **the currents add up** to the circuit voltage.

$$\frac{1}{\text{TotalResistance}(R)} = \frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3} \dots$$

$$\frac{1}{\text{TotalInductors}(L)} = \frac{1}{L_1} + \frac{1}{L_2} + \frac{1}{L_3} \dots$$

$$\text{Capacitors}(C) = c_1 + c_2 + c_3 \dots$$

The Arduino Kit



Arduino Starter Kit Multi-language

Quickly and easily get started with learning electronics using the Arduino Starter Kit, which have a universal appeal to STEM fans at home, businesses in STEAM industries, and schools alike. No prior experience is required, as the kits introduce both coding and electronics through fun, engaging, ...

[View full details](#)

\$110.30

[CHOOSE OPTIONS](#)

[QUICK SHOP](#)



Arduino Braccio ++

The next evolution of the Tinkercat Braccio robot, Arduino Braccio ++ is a robotic arm designed solely for higher education, including engineering schools and university institutes of technology – or even advanced high school and college students studying the sciences, industrial science or techn...

[View full details](#)

\$599.00

[ADD TO CART](#)

[QUICK SHOP](#)

amazon.ae

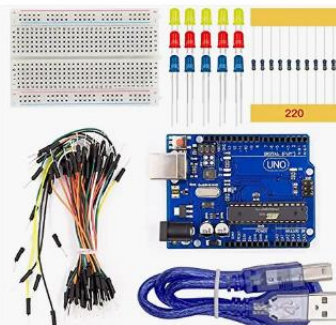


Upgraded 37 in 1 Sensors Modules Kit for Arduino Starters
DIY Raspberry Pi Mega2560 UNO R3

★★★★☆ ~ 2

AED 60⁰⁰

Shop and Win FIFA package (AED 1.00 discount)



R3 ATMEGA328P Learning Kit

★★★★★ ~ 5

AED 62⁰⁰

Shop and Win FIFA package (AED 1.00 discount)

Get it as soon as **Wednesday, October 26**

AED 10.00 shipping

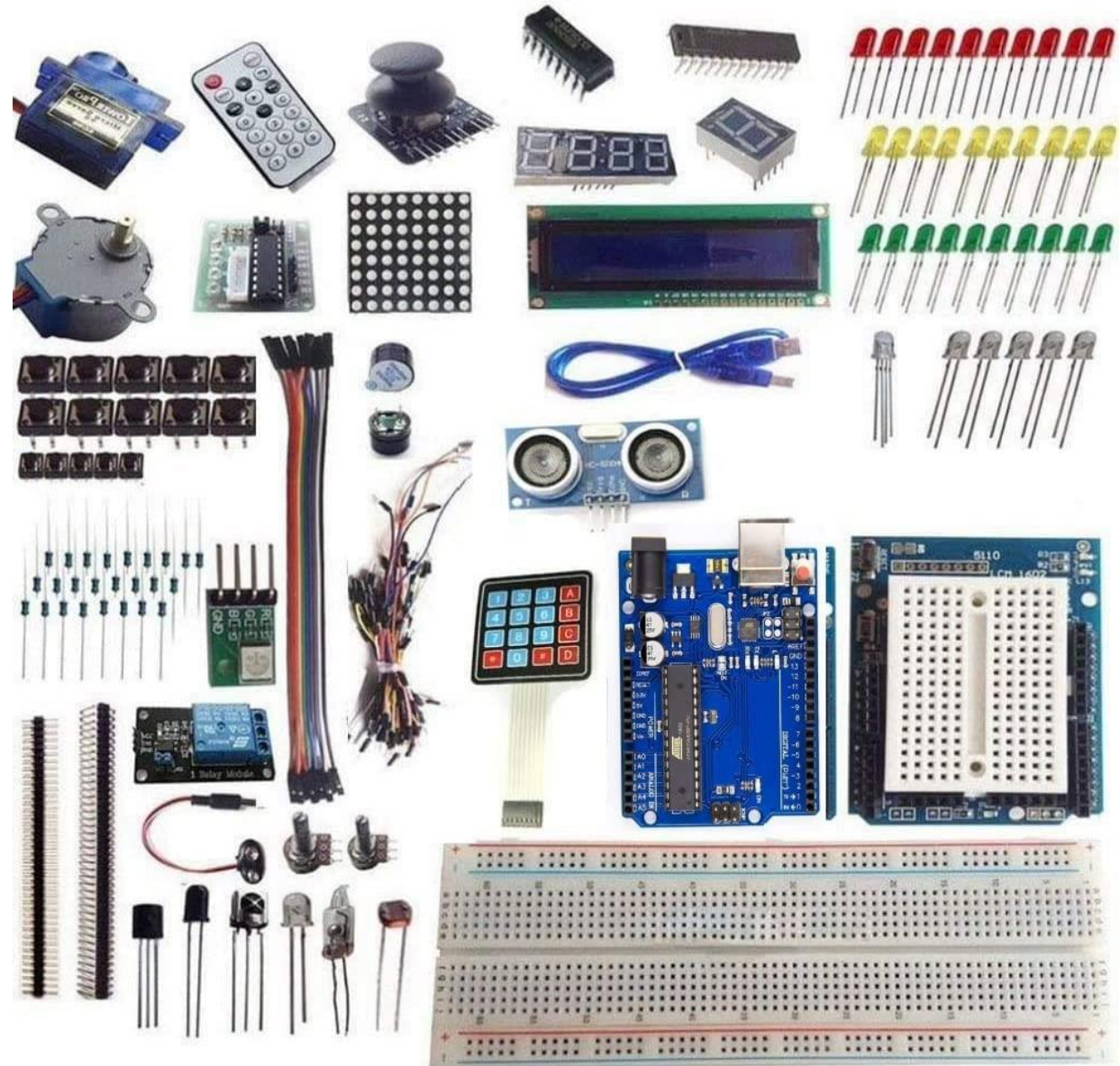


Decdeal Starter learning Kit for Arduino UNO R3 LCD1602 Servo processing

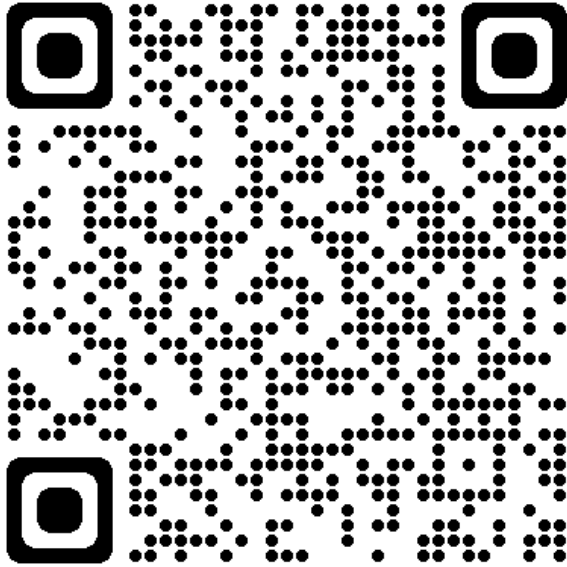
★★★★☆ ~ 1

AED 165⁰⁰

Shop and Win FIFA package (AED 1.00 discount)

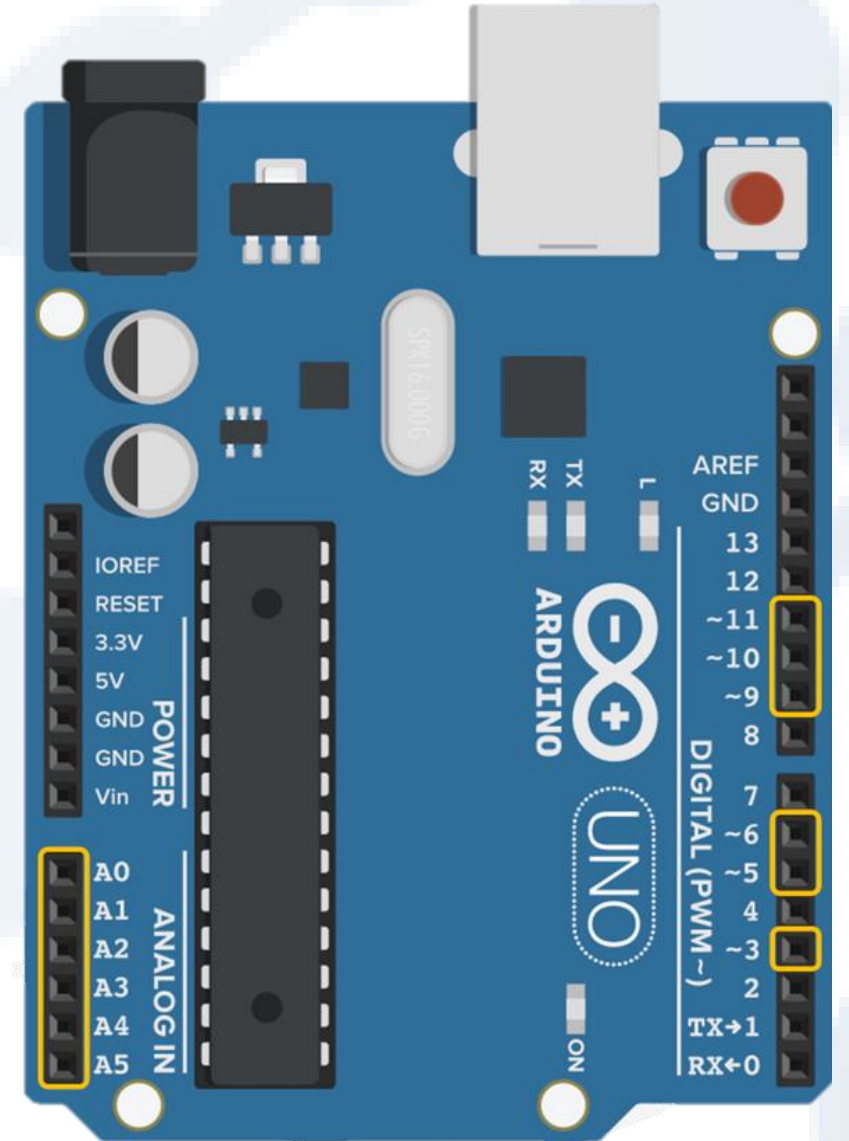


Arduino IDE



Arduino Web Editor
[Arduino Editor](#)

Arduino IDE
[Software](#) | [Arduino](#)



Mini-project

Blink

VERY EASY

Learn to define and control digital GPIO pins to turn an LED on and off.

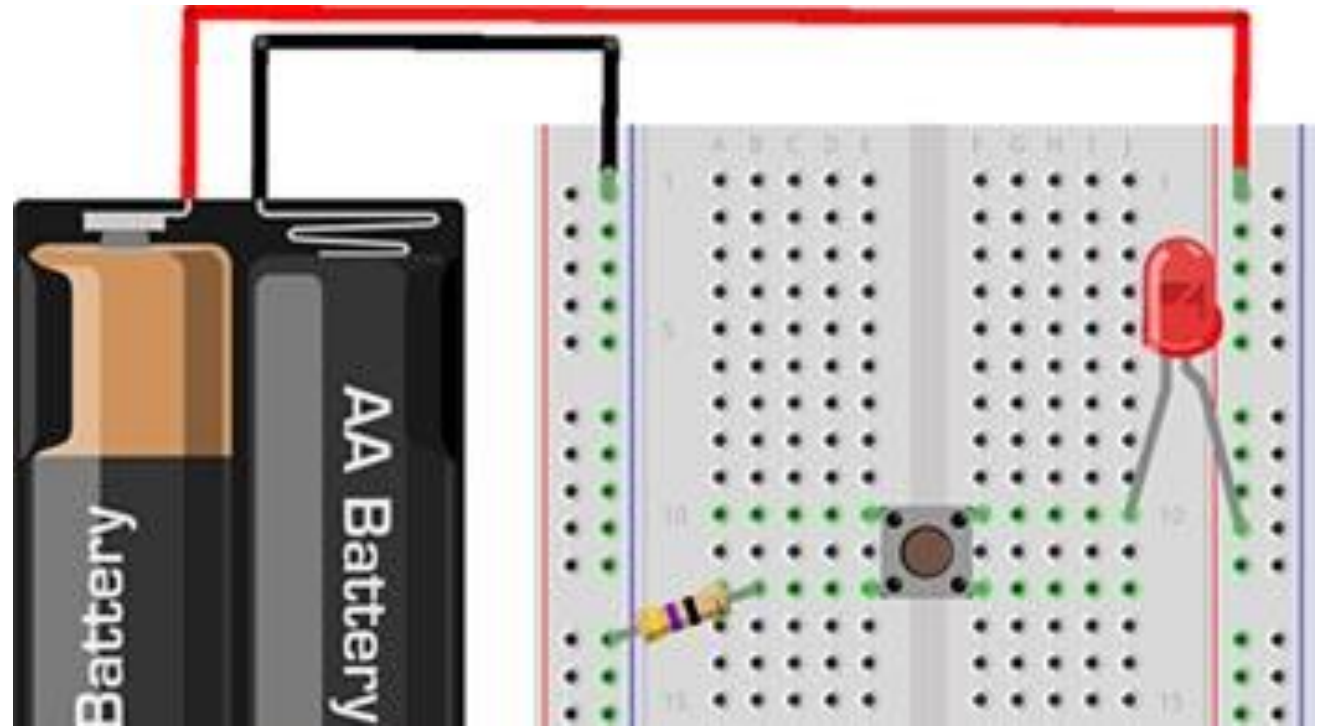
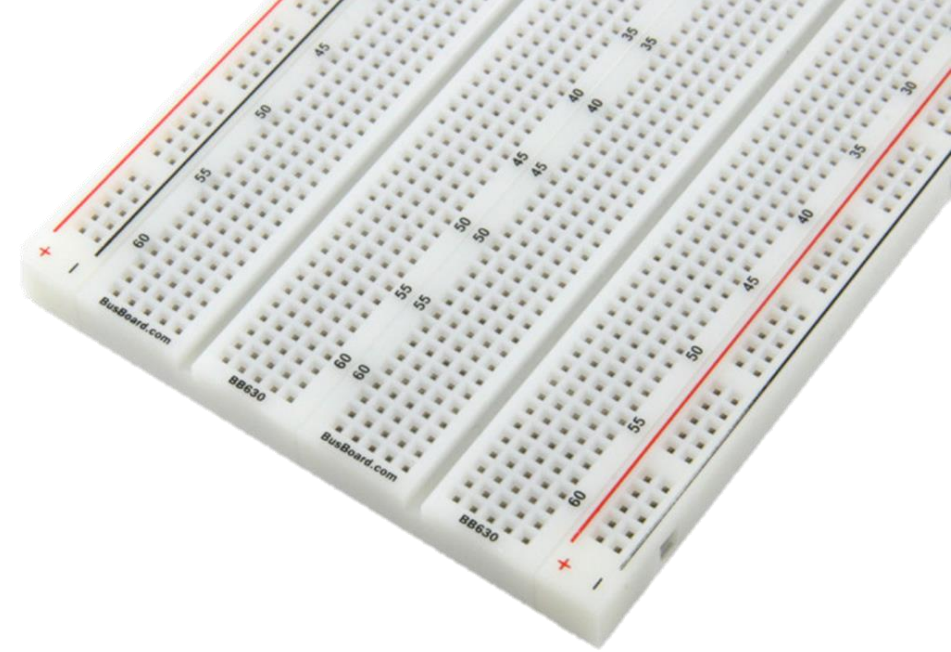
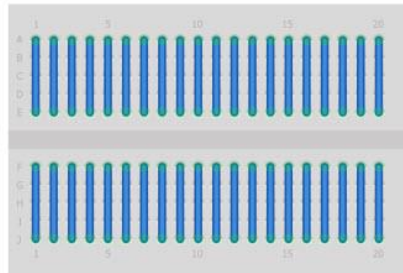
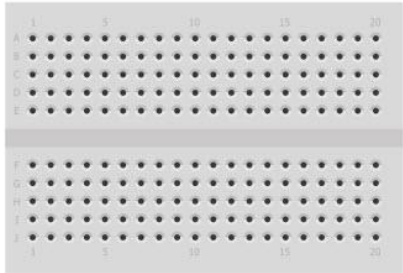
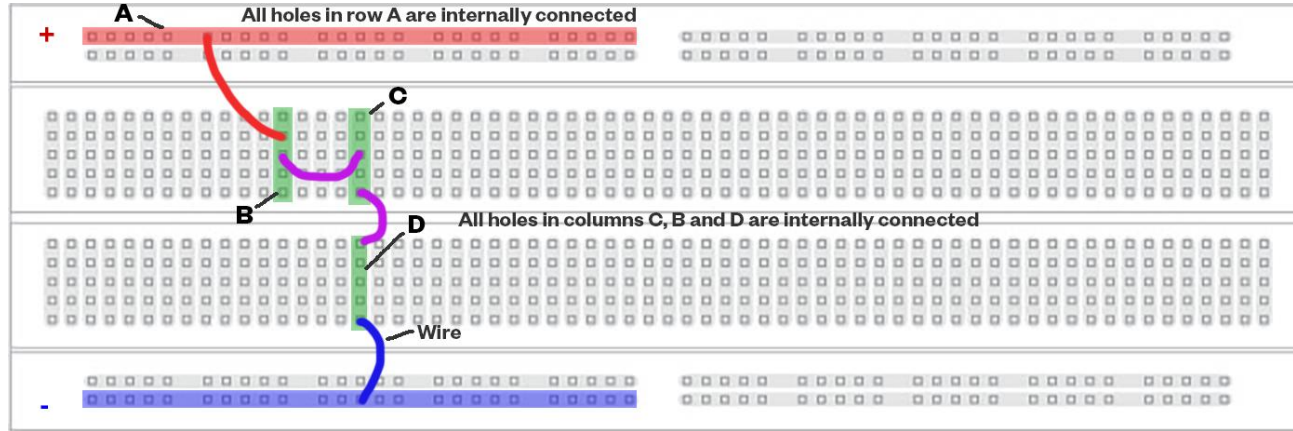
Learning outcomes:

- Get familiar with the Arduino IDE.
- Learn the Arduino script structure.
- Learn how to define IO pins.
- Learn how to write to an output pin.
- Learn how to use an LED.

Blink: Hardware Required

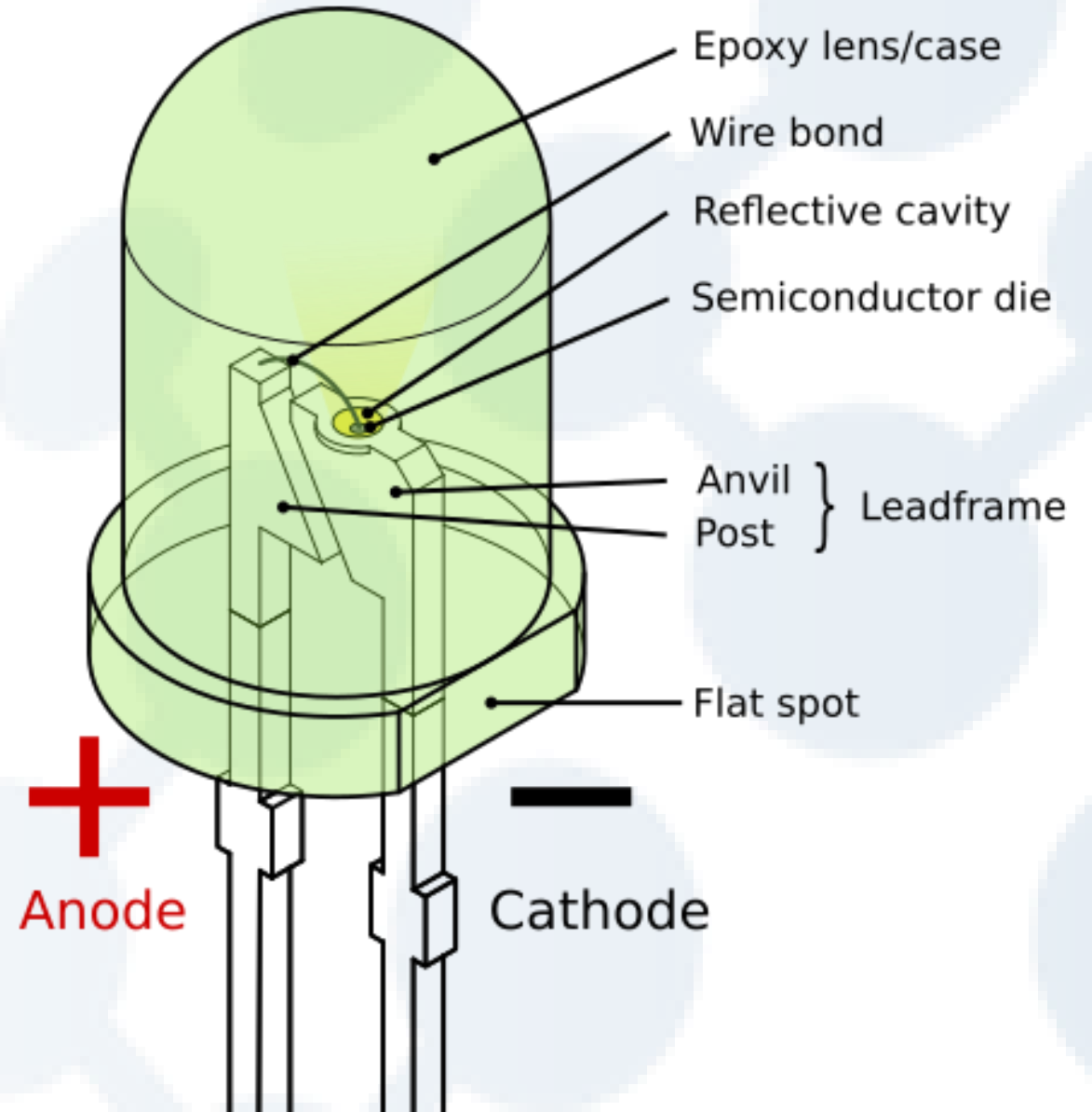
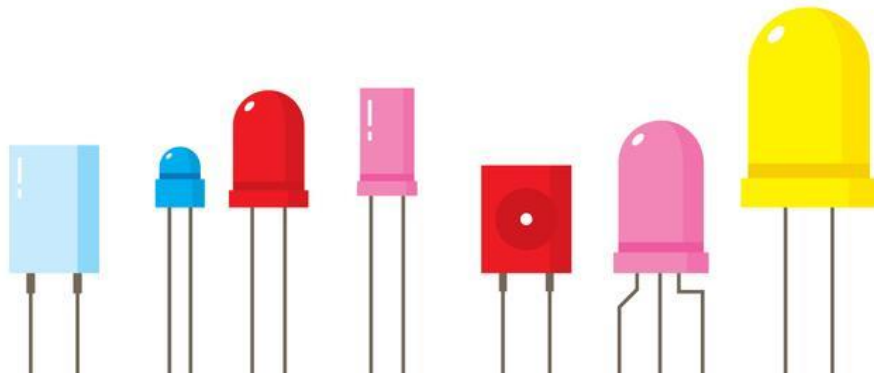
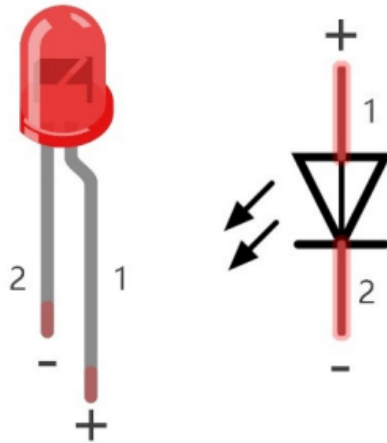
- LED
- 220 Ohm resistor
- Breadboard
- Jumper wires

Breadboard

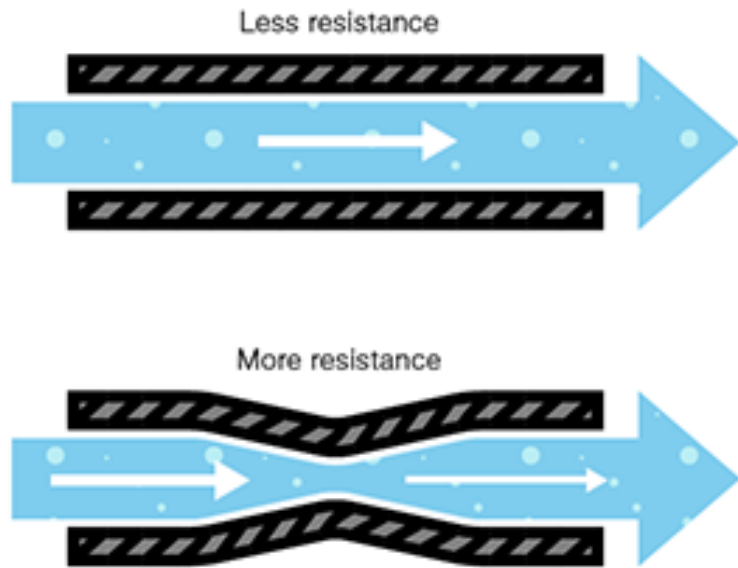


Light Emitting Diodes (LEDs)

It is a semiconductor device that emits light when an electric current is passed through it.



Resistors



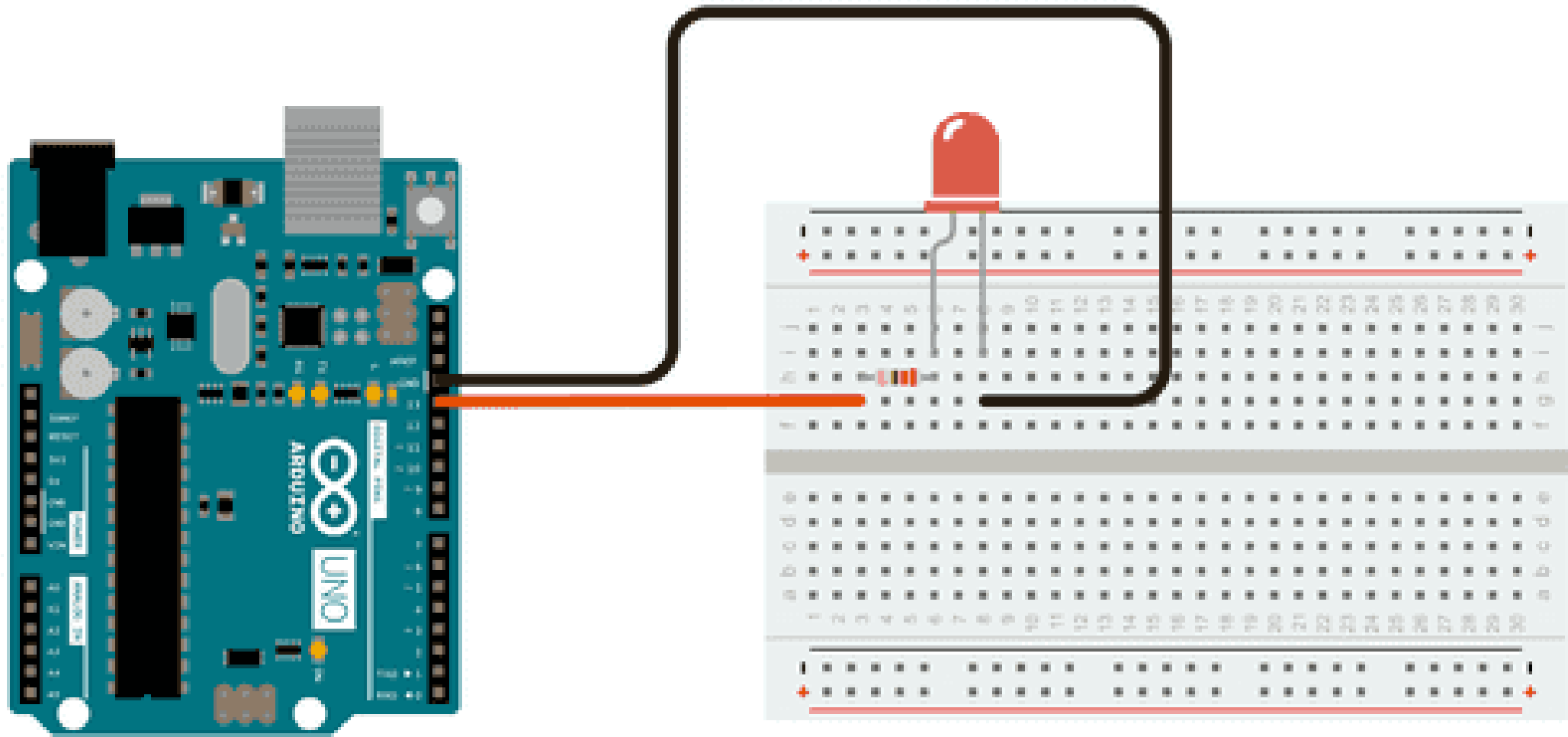
120 ohm 5%

	1st	2nd	3rd	Multiplier	Tolerance	
0	Black	Black	Black	Black	1	
1	Brown	Brown	Brown	Brown	10^1	Brown 1%
2	Red	Red	Red	Red	10^2	Red 2%
3	Orange	Orange	Orange	Orange	10^3	
4	Yellow	Yellow	Yellow	Yellow	10^4	
5	Green	Green	Green	Green	10^5	
6	Blue	Blue	Blue	Blue	10^6	
7	Violet	Violet	Violet	Violet	10^7	
8	Grey	Grey	Grey	Grey	10^8	
9	White	White	White	White	10^9	
				Gold	0.1	Gold 5%
				Silver	0.01	Silver 10%

1st 2nd 3rd Multiplier Tolerance

4.7k ohm 1%

Blink: Circuit Diagram



Blink: The Script

```
// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);                     // wait for a second
  digitalWrite(LED_BUILTIN, LOW);  // turn the LED off by making the voltage LOW
  delay(1000);                     // wait for a second
}
```

Mini-project

Program a Button

VERY EASY

Learn to read a digital signal and use it to control an LED.

Learning outcomes:

- Learn how to read digital signals.
- Get familiar with switches.

Program a Button: Hardware Required

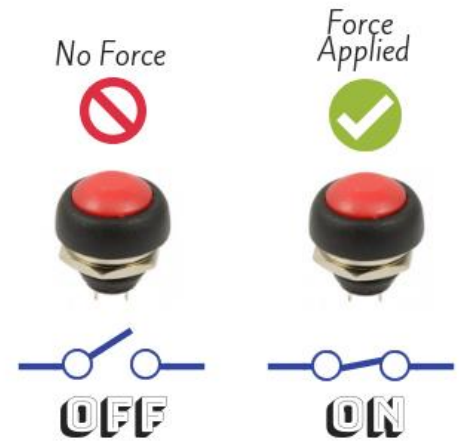
- Momentary button or Switch
- 10K ohm resistor
- Breadboard
- Jumper wires

Switches

A Switch is a simple device that is used to break a circuit and disturb the flow of electrons.

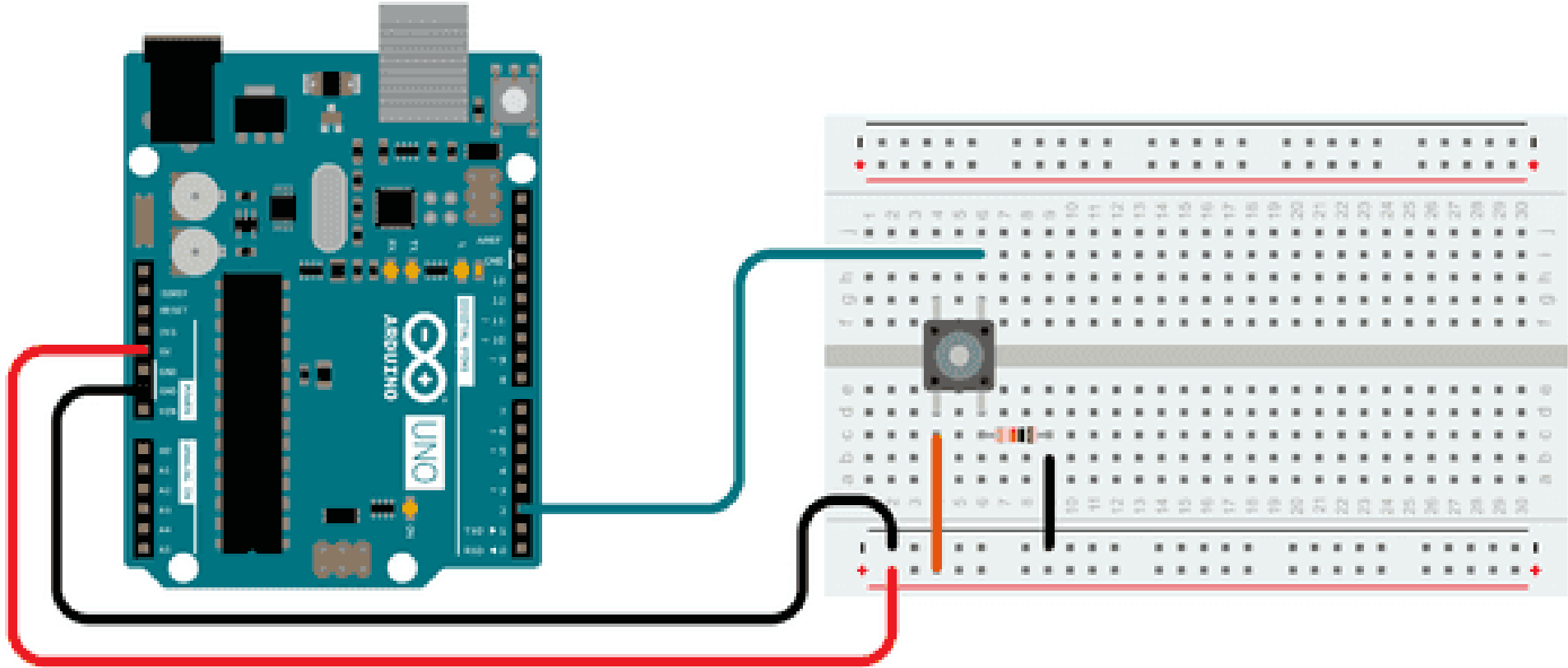


N/C Normally Closed



N/O Normally Open

Program a Button: Circuit Diagram



Program a Button: The Script

```
// constants won't change. They're used here to set pin numbers:
const int buttonPin = 2;  // the number of the pushbutton pin
const int ledPin = 13;    // the number of the LED pin

// variables will change:
int buttonState = 0;  // variable for reading the pushbutton status

void setup() {
  // initialize the LED pin as an output:
  pinMode(ledPin, OUTPUT);
  // initialize the pushbutton pin as an input:
  pinMode(buttonPin, INPUT);
}

void loop() {
  // read the state of the pushbutton value:
  buttonState = digitalRead(buttonPin);

  // check if the pushbutton is pressed. If it is, the buttonState is HIGH:
  if (buttonState == HIGH) { // turn LED on:
    digitalWrite(ledPin, HIGH);
  } else { // turn LED off:
    digitalWrite(ledPin, LOW);
  }
}
```

Mini-project

Analog, Read Serial

EASY

Learn to read an analog signal and print the value to the Arduino Serial Monitor.

Learning outcomes:

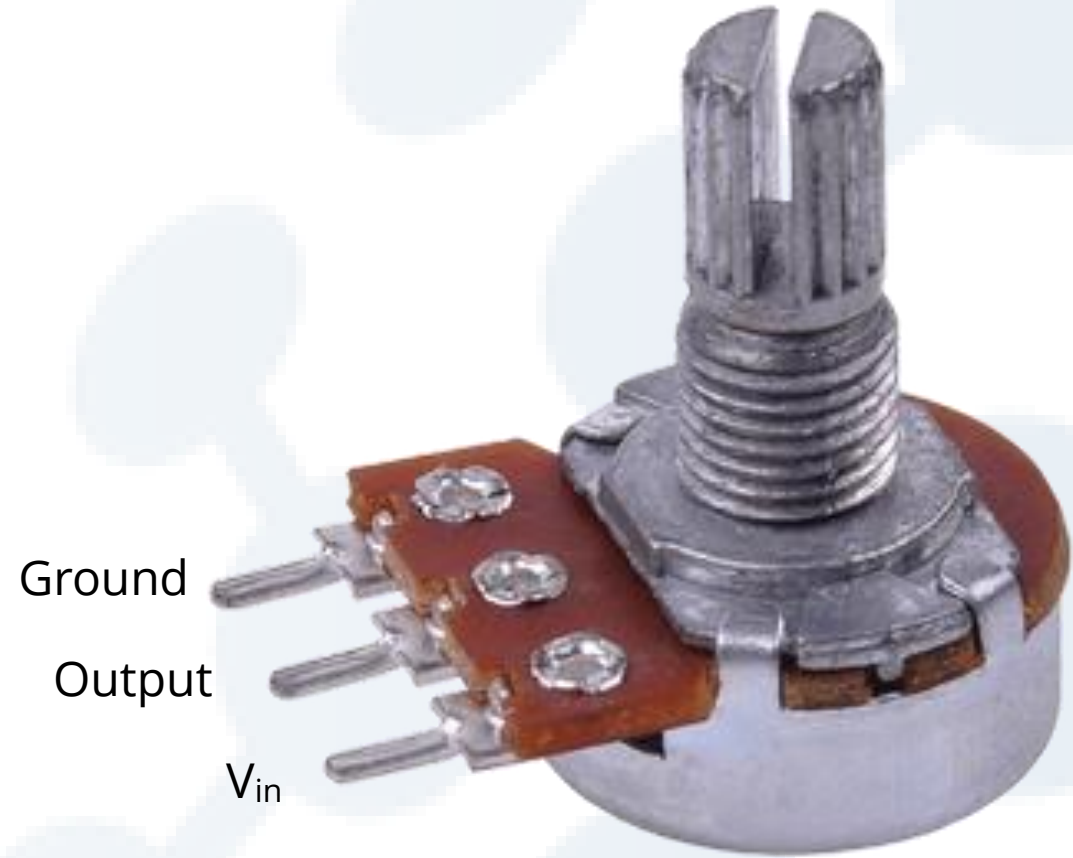
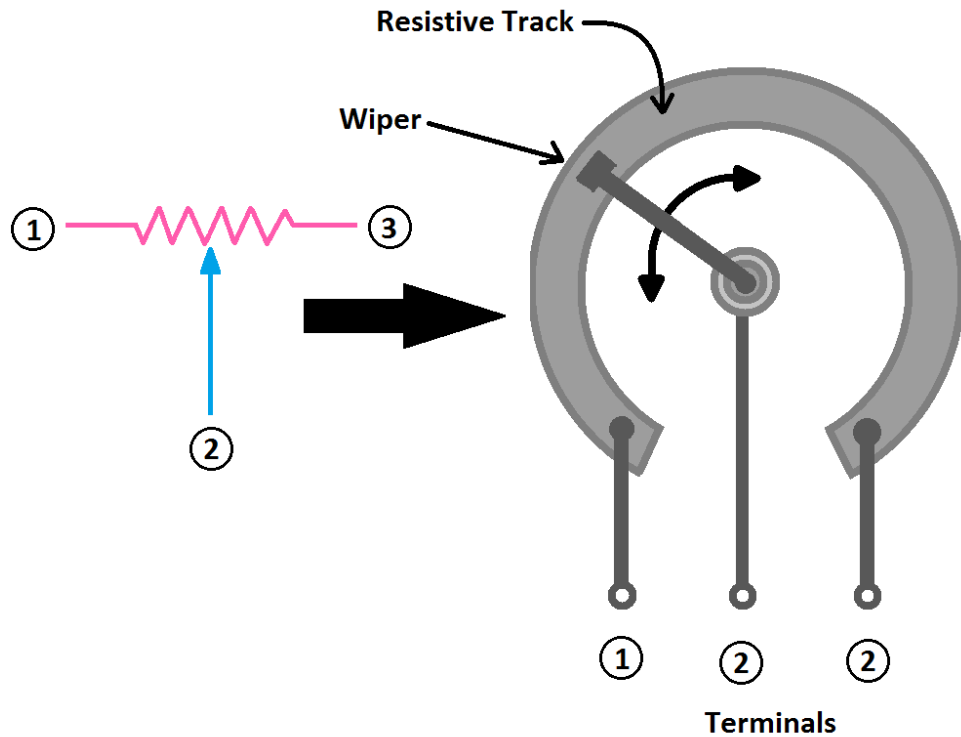
- Learn how to read analog signals.
- Learn how to set up the serial baud rate.
- Learn to use the Arduino Serial Monitor.
- Get familiar with potentiometers.

Analog, Read Serial: Hardware Required

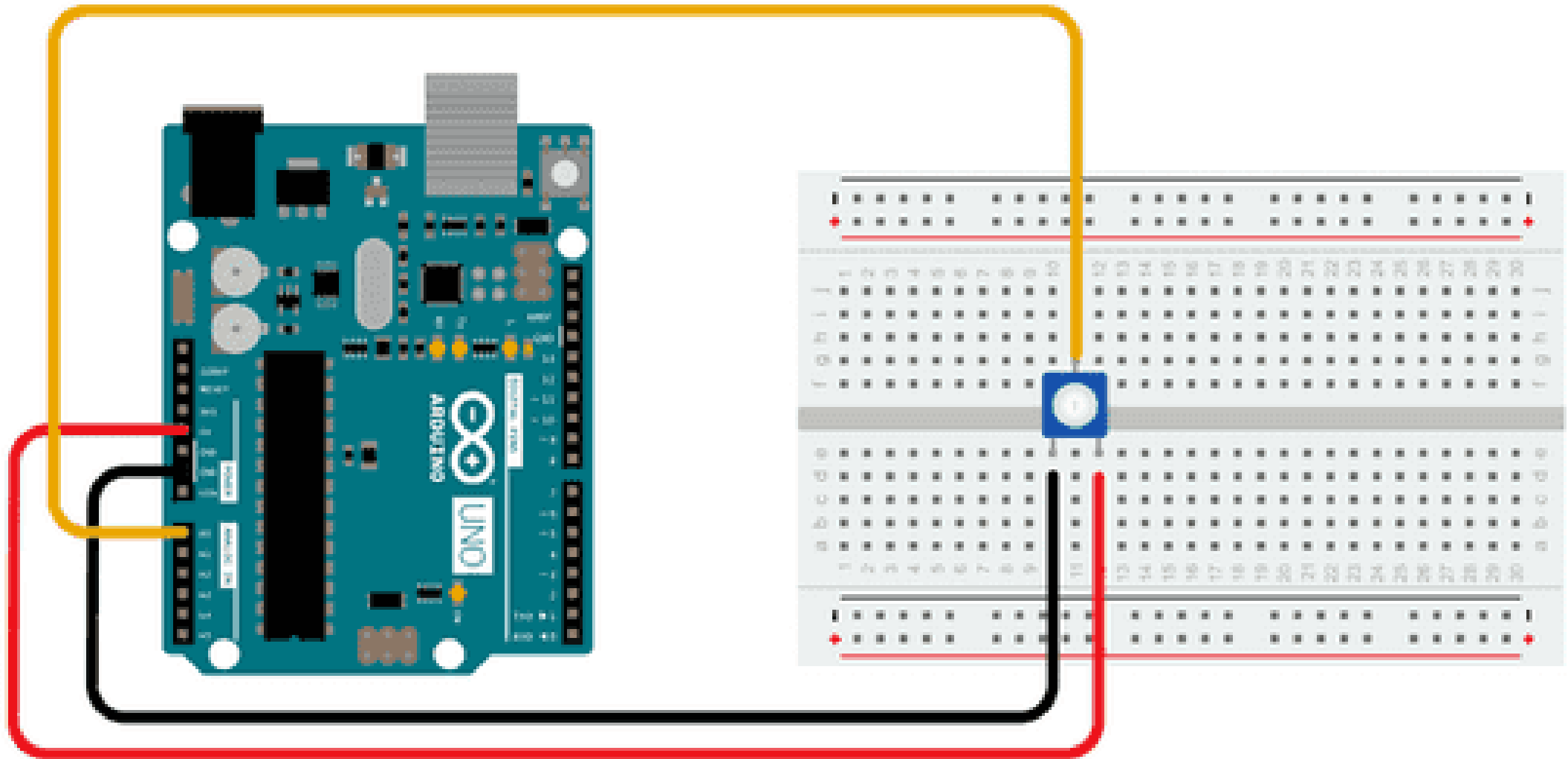
- 10k ohm Potentiometer
- Breadboard
- Jumper wires

Potentiometers

A 3-terminal variable resistor in which the resistance is manually varied to control the flow of electric current.



Analog, Read Serial: Circuit Diagram



Analog, Read Serial: The Script

```
// the setup routine runs once when you press reset:
void setup() {
  // initialize serial communication at 9600 bits per second:
  Serial.begin(9600);
}

// the loop routine runs over and over again forever:
void loop() {
  // read the input on analog pin 0:
  int sensorValue = analogRead(A0);
  // print out the value you read:
  Serial.println(sensorValue);
  delay(1); // delay in between reads for stability
}
```

Mini-project

Analog In, Out Serial

EASY

Learn to read an analog signal and use it to control the speed of a motor.

Learning outcomes:

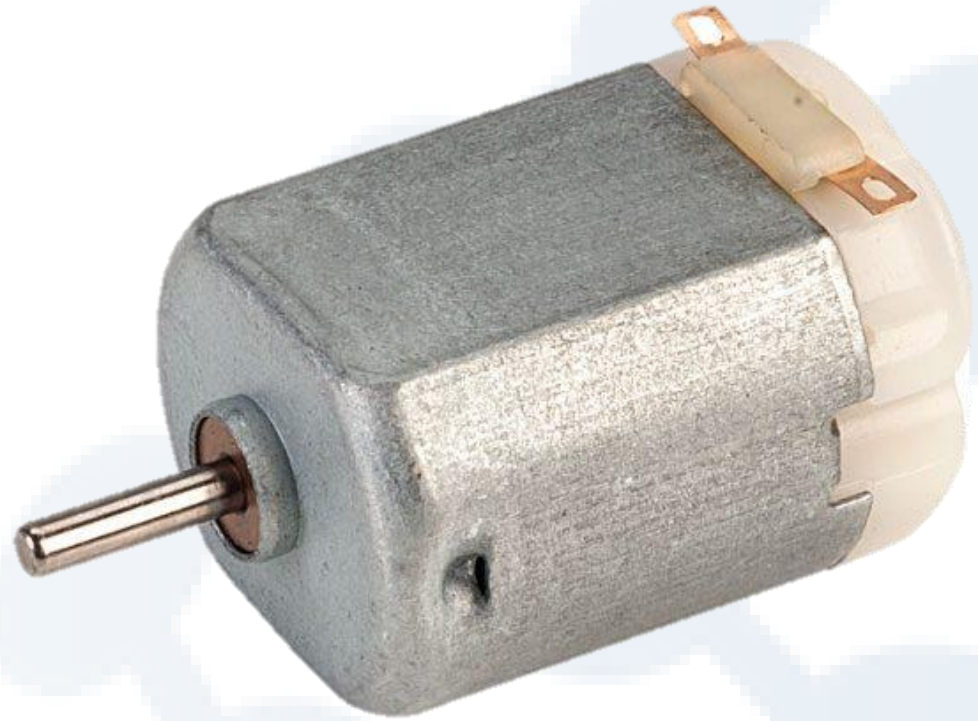
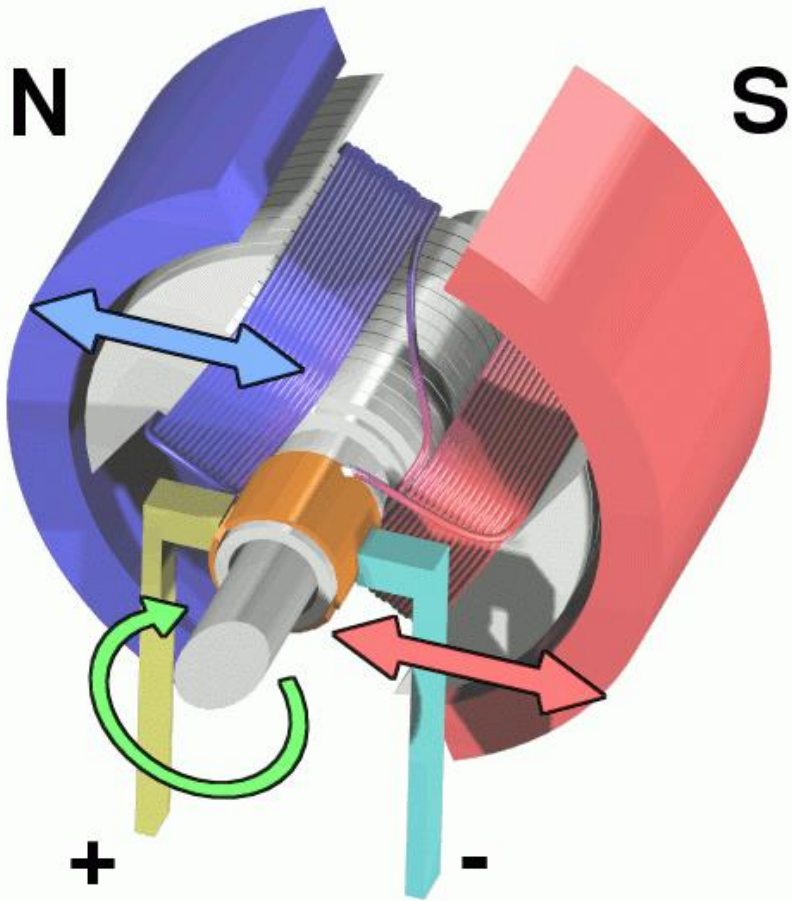
- Get familiar with Pulse Width Modulation (PWM).
- Learn how to use an electric motor.

Analog In, Out Serial: Hardware Required

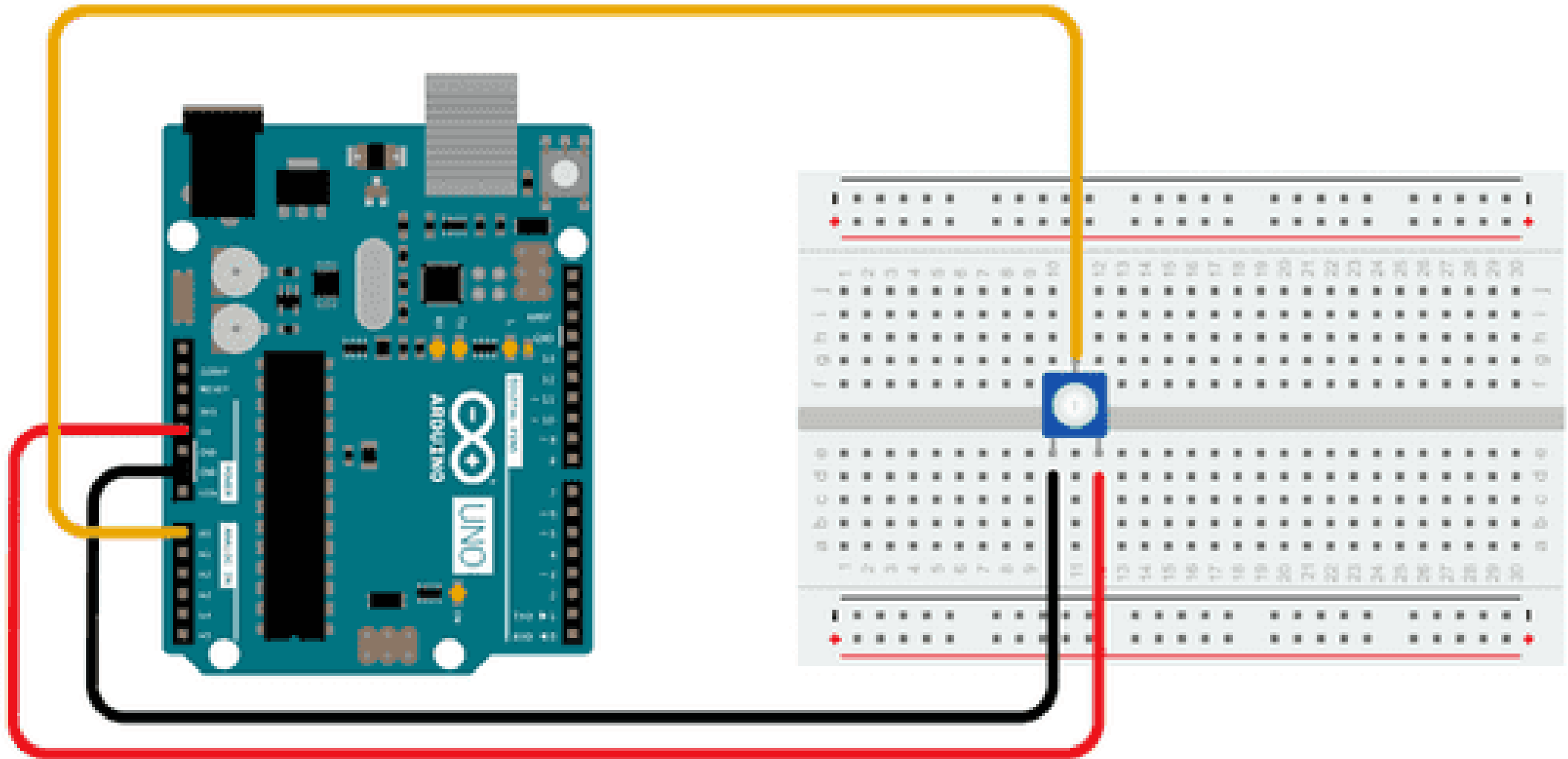
- 10k ohm Potentiometer
- Electric motor
- Breadboard
- Jumper wires

Electric Motors

A DC electrical motor is any rotary motor that converts direct current (DC) electrical energy into mechanical energy.



Analog In, Out Serial: Circuit Diagram



Analog In, Out Serial: The Script

```
// These constants won't change. They're used to give names to the pins used:
const int analogInPin = A0; // Analog input pin that the potentiometer is attached to
const int analogOutPin = 9; // Analog output pin that the LED is attached to

int sensorValue = 0;          // value read from the pot
int outputValue = 0;          // value output to the PWM (analog out)

void setup() {
  // initialize serial communications at 9600 bps:
  Serial.begin(9600);
}
```


Analog In, Out Serial: The Script

```
void loop() {  
  // read the analog in value:  
  sensorValue = analogRead(analogInPin);  
  
  // map it to the range of the analog out:  
  outputValue = map(sensorValue, 0, 1023, 0, 255);  
  
  // change the analog out value:  
  analogWrite(analogOutPin, outputValue);  
  
  // print the results to the Serial Monitor:  
  Serial.print("sensor = ");  
  Serial.print(sensorValue);  
  Serial.print("\t output = ");  
  Serial.println(outputValue);  
  
  // wait 2 milliseconds before the next loop for the analog-to-digital  
  // converter to settle after the last reading:  
  delay(2);  
}
```



Thank you