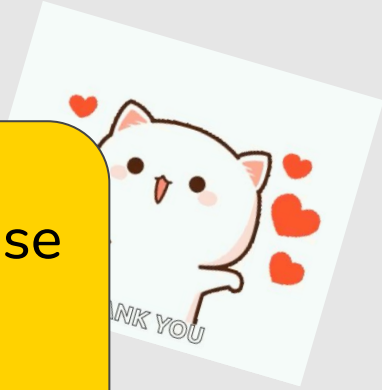
https://go.umd.edu/TWSP25

Students please sign in for the TW Workshop!



While you are waiting, please go to **tinkercad.com** and make an account!

TERRAPIN WORKS

Intro: Motors and Servos

The Plan

Motor Mechanics

DC Motor Circuit Design

Motor Speed Project

Servo Motor Circuit Design

Servo Turning Project

Fundamentals

What is a motor?

Motors convert **electrical** energy to rotational **mechanical** energy. There are two main types:

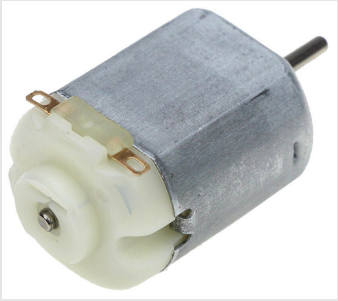
- **DC motors** for fast, uncontrolled motion
- **Servo motors** for slow, controlled motion

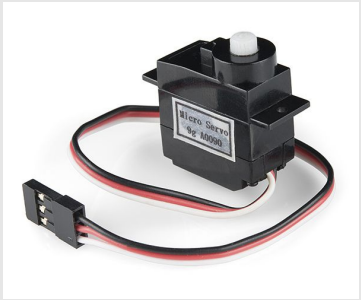
Cars

Pumps

Fans

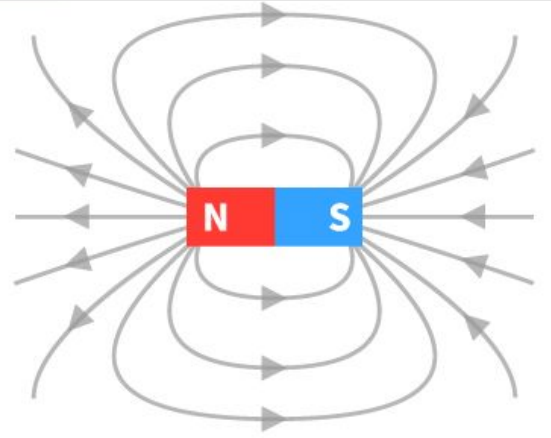
Power Plants



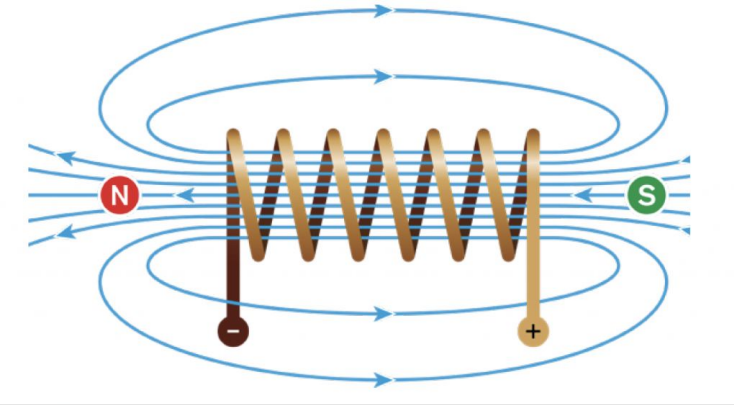


Magnets

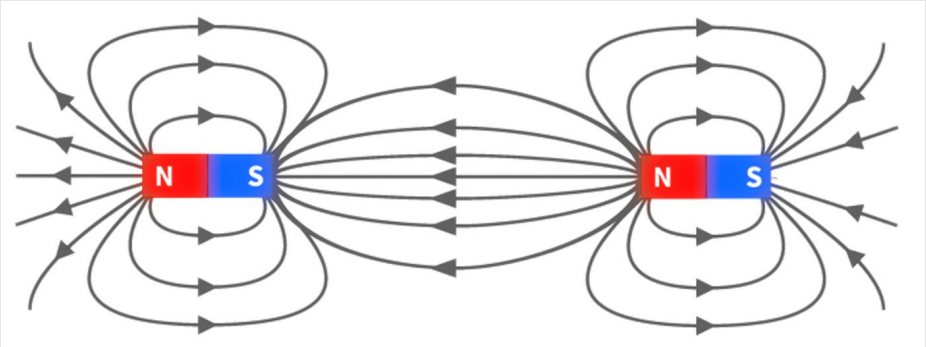
Permanent magnets are made of rare minerals that have a built-in magnetic field.



Electromagnets (aka temporary magnets) generate magnetic fields by putting electric current through a coil of wire. **They have an off switch!**



Magnetic Fields



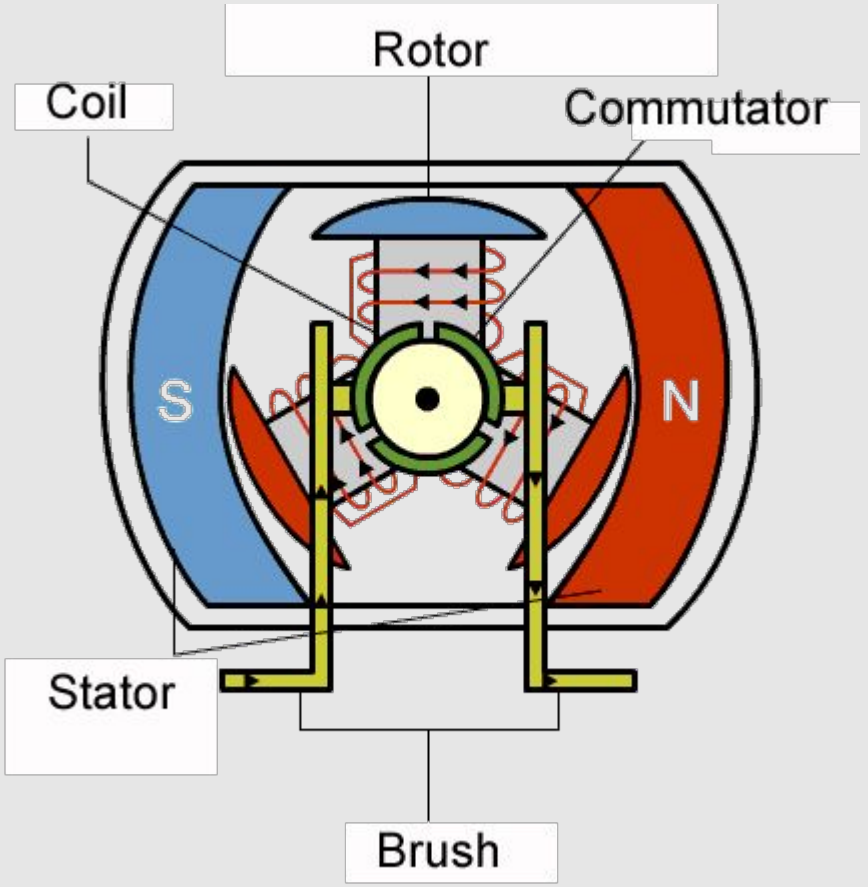
Represent **looping lines of force** through space that some objects “see”:

- Magnets (permanent **OR** temporary) can interact with other magnetic fields
- Certain materials like **iron**

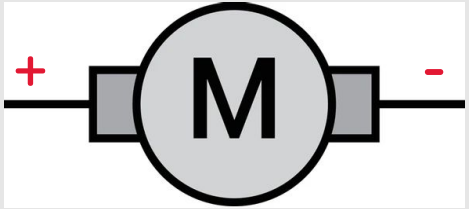
Magnets want to align their fields with each other!

Magnets in Motors

Electromagnets built into the central **rotor** get power through the **brushes**. Forces between these and permanent magnets (**stator**) arranged around the rotor spin the shaft!



DC Motors



The two terminals connect **directly** to the brushes.

Behavior is controlled by voltage across the terminals:

- **Polarity** controls direction of spin
- **Magnitude** controls speed **and** torque

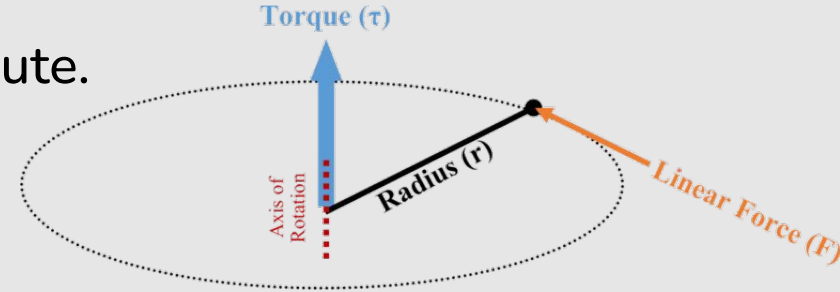
DC Motor Terminology

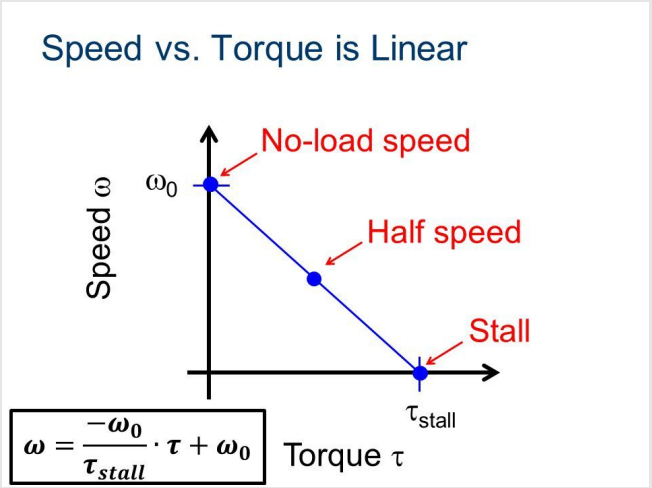
RPM: Motor speed in revolutions per minute.

Torque: Force behind the rotation.

No Load Speed: Motor speed with nothing on the shaft.

Stall Torque: Maximum force output of the motor.



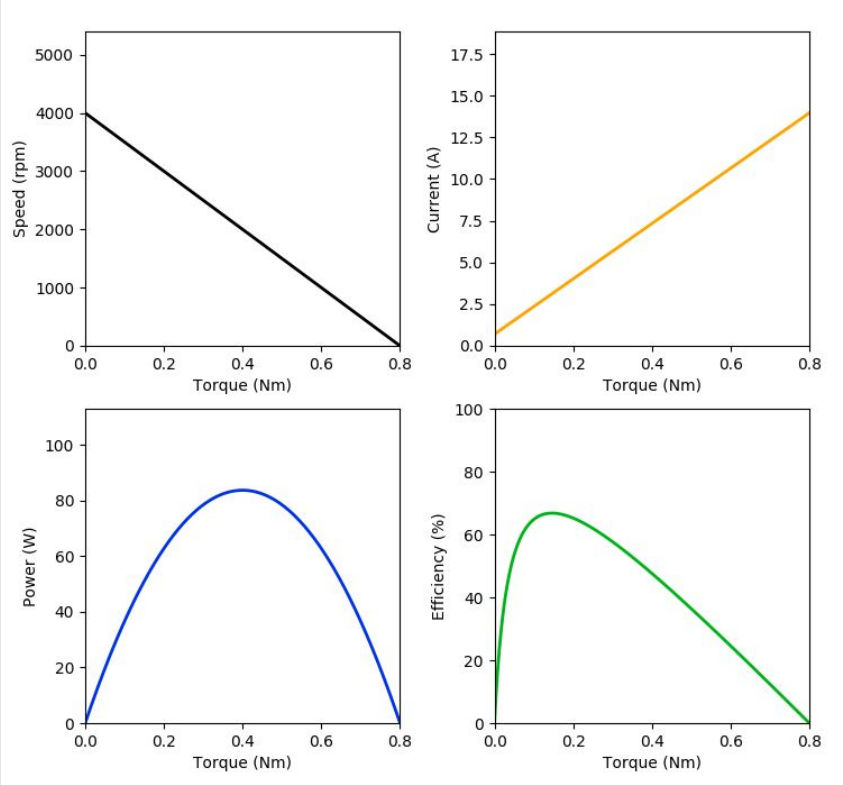


Characteristics of Motors

RPM: Motor speed decrease with an increase in torque.

Current: Current increase linearly with torque.

Efficiency: Varies per motor make sure to check the manufacturer's manual for details



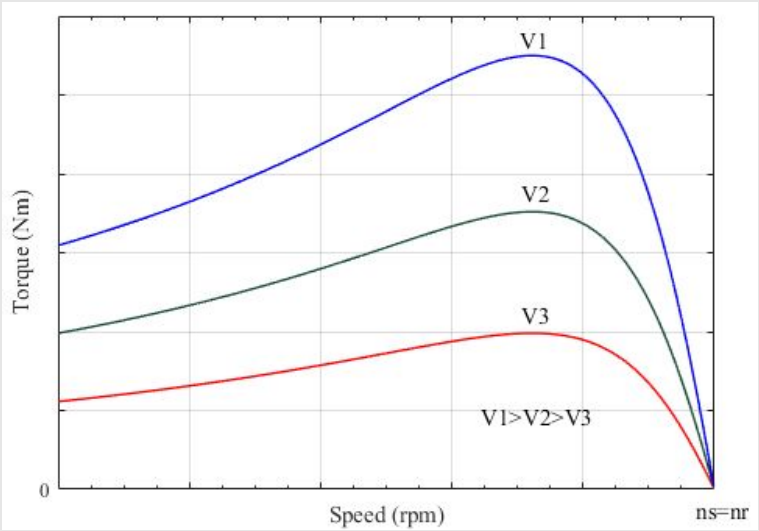
DC Motor Controls

Voltage Control

Motor output is proportional to voltage

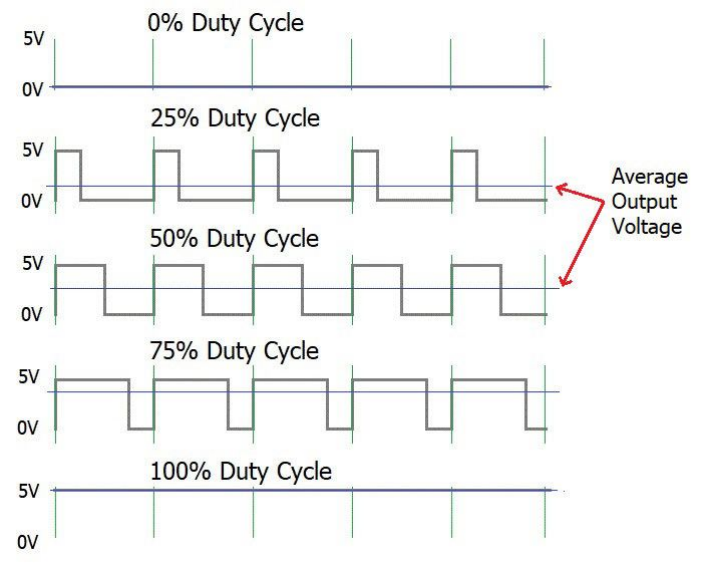
Pros : Simple and intuitive

Cons : High power consumption requires dedicated motor controllers to safely adjust voltage



Pulse Width Modulation (PWM)

PWM allows digital pins to *effectively* output voltages between **HIGH** and **LOW** by rapidly toggling the pin on and off at a set **duty cycle**.



Duty Cycle is the fraction of time for which the pin is on.

$$D = \frac{T_{on}}{T_{total}}$$

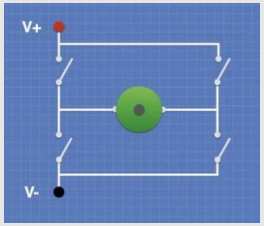
$$V_{eq} = DV_{max}$$

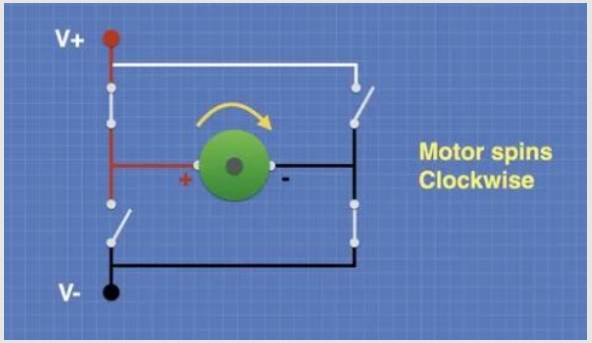
DC Motor Controllers

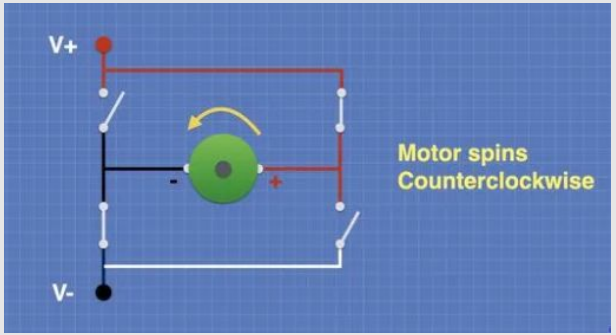
H-Bridge: Controls direction of current that controls direction of rotation.

Clockwise Motion: Current flows into + terminal and out of - terminal.

Counterclockwise Motion: Current flows into - terminal and out of + terminal.







Application

How to Choose a Motor

- How fast do you need to go ?
- Where is the motor used ?
- Size constraints ?
- What controller is available ?
- Power Constraints ?

DC Motor Types

Brushed DC

Advantages

- Cheapest and simplest motor
- Speed linear to applied voltage
- Simple motor control

Disadvantages

- High maintenance
- Low life-span (due to physical wear on brushes)

Brushless DC

Advantages

- High efficiency
- Little to no maintenance
- Long life span
- High output power per frame size

Disadvantages

- More complicated motor control
- Large initial costs

Stepper

Advantages

- Accurate position control
- Excellent low speed torque
- Long life

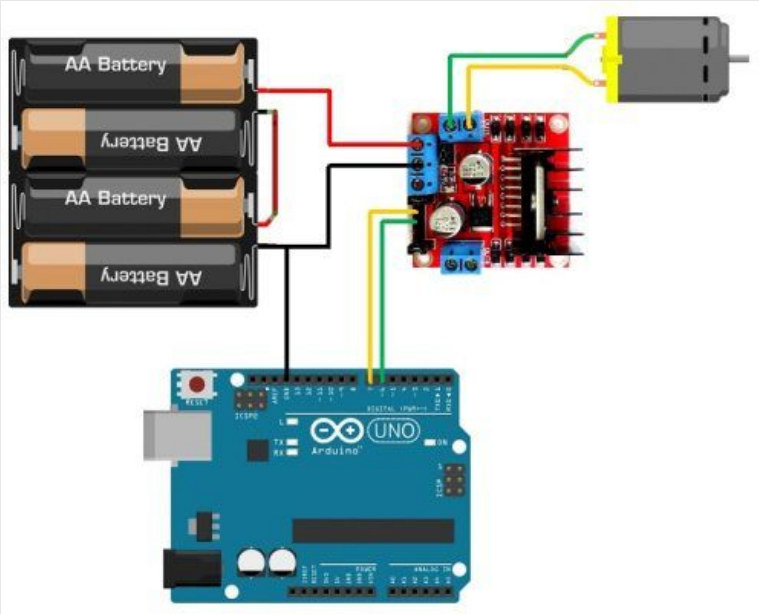
Disadvantages

- Low efficiency
- Prone to resonances, noise, and torque ripple
- Cannot accelerate loads rapidly

Example H-Bridge Used in ENES 100

External Power: Most motors need more than 5V.

Green and Yellow Input: Input PWM into one of them. Direction depends on which one



Project: Motor Control

Let's Review:

Power : Separate power required for big components

Transistor : Can be used to regulate power flow

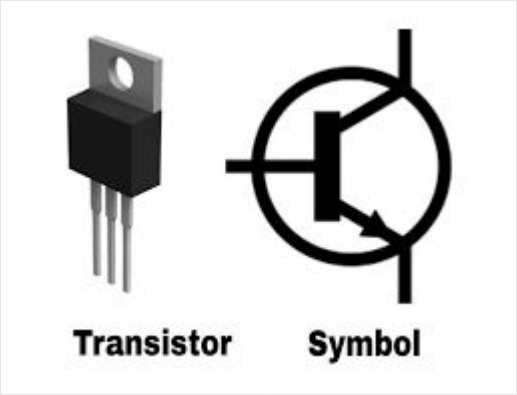
Potentiometer : Variable resistor that is adjusted by a knob

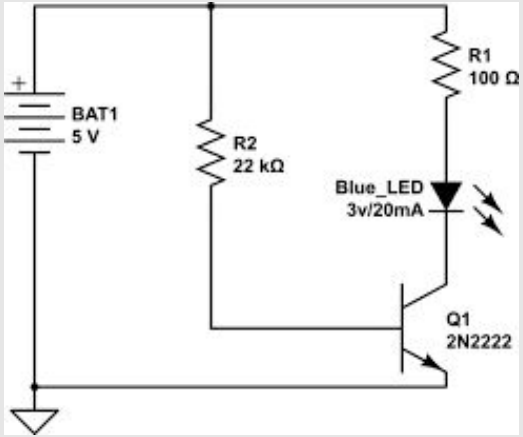
Analog : Arduino can read continuous input

Analog to Digital : Map function converts analog to digital

Transistor

- **Electrical Switch**
- One terminal impacts current flow between the other two terminals

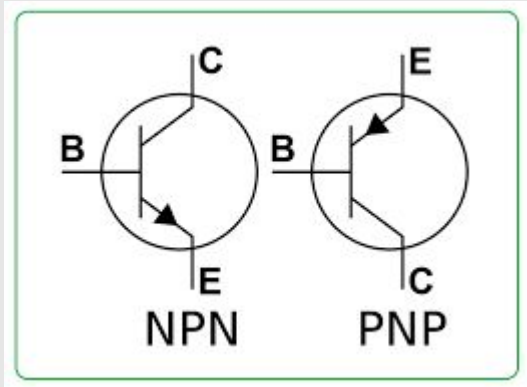


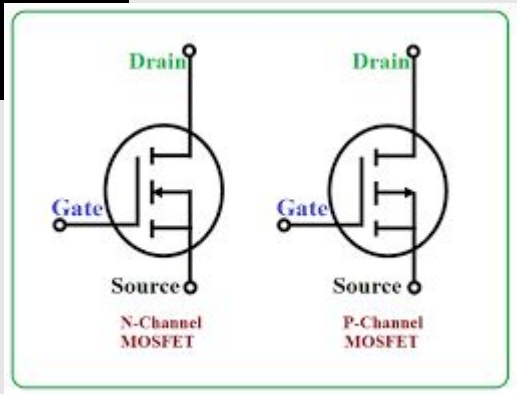


MOSFET vs. BJT

MOSFET

- Voltage Dependent
- Gate control pin





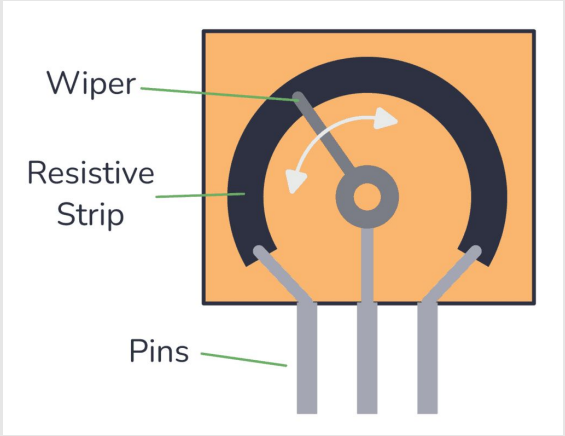
BJT

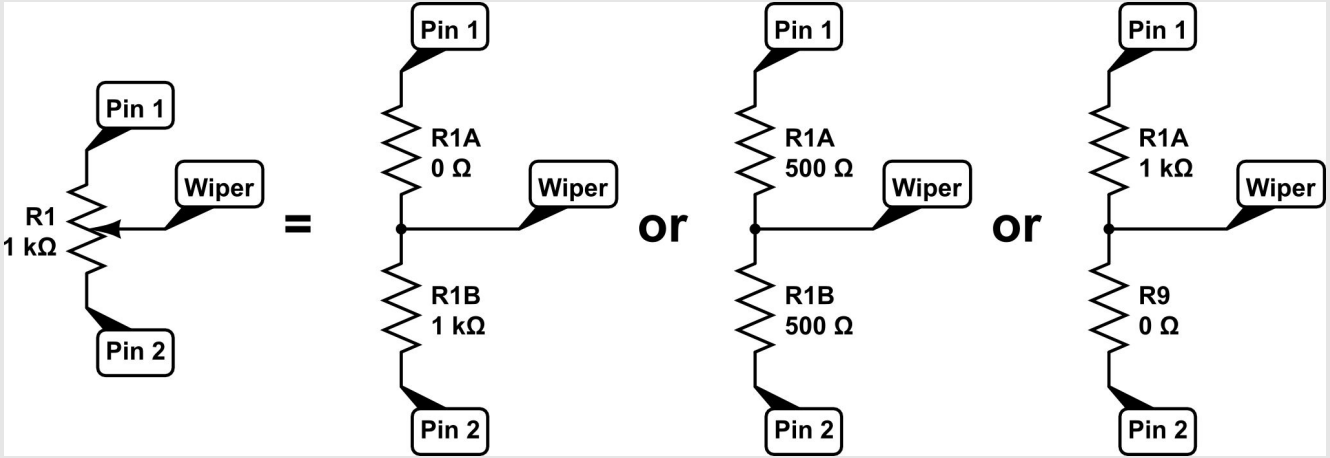
- Current Dependent
- Base control pin

Potentiometers

These **variable resistor dividers** can change the center pin (wiper) voltage as you turn the knob!

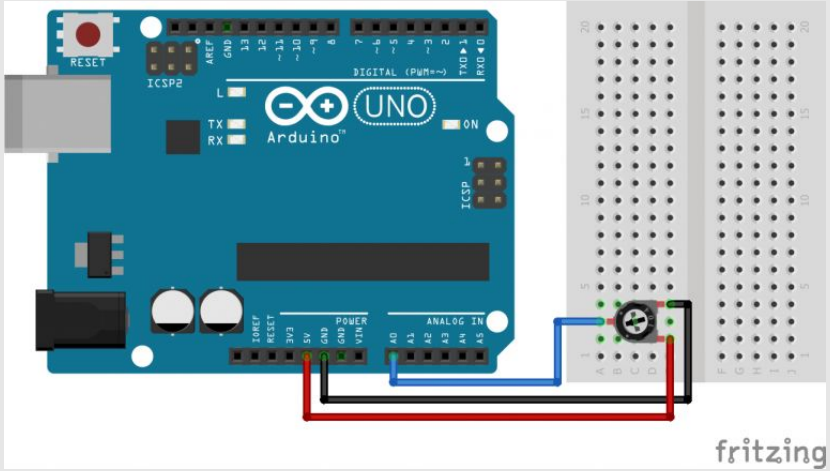
A pot's written **value** is the resistance between the outer pins.





The ADC

- It converts voltage measurements into a 10-bit digital value between 0 and 1023
- The Arduino Uno has six ADCs at pins A0-A5
- The Arduino Uno does **not** have any DACs



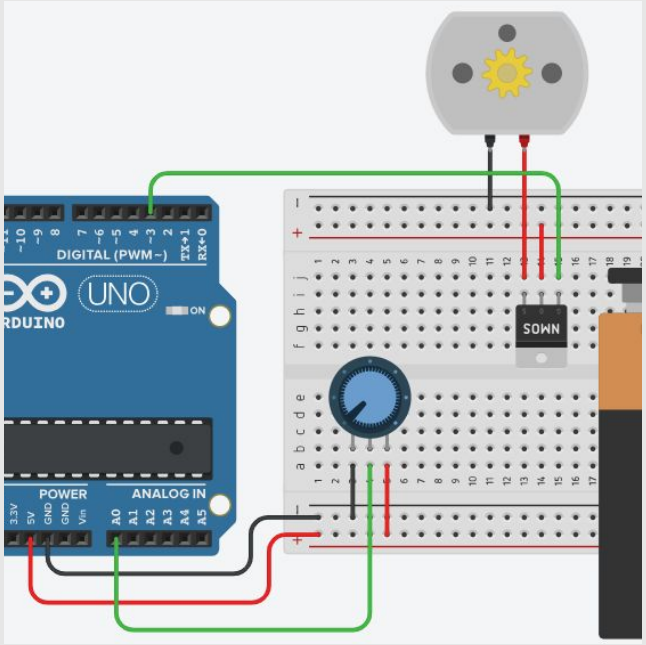

```
void setup() {
  int ADCvalue = 0;
  // initialize serial communication at 9600 baud rate
  Serial.begin(9600);
}
void loop() {
  // Read the analog value from pin A0
  int ADCvalue = analogRead(A0);
  // print the value at serial monitor
  Serial.println(ADCvalue);
  delay(100);        // delay in between reads for stability
}
```

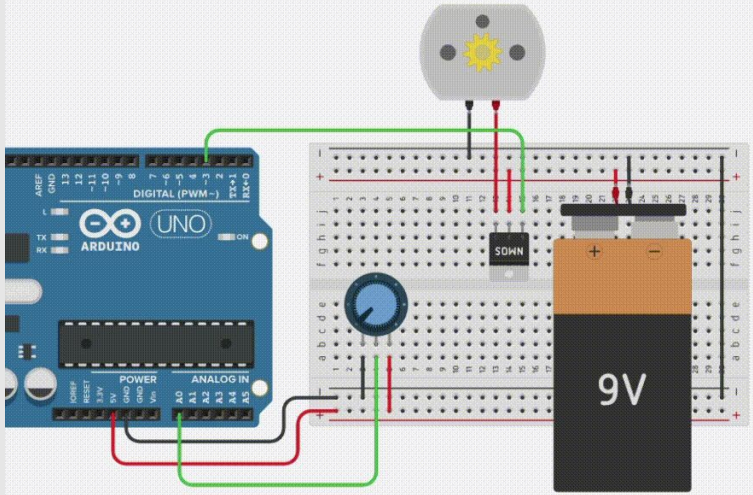
Project: DC Motor Control

```
int motorCtl = 3;
int pot = A0;

void setup() {
  pinMode(motorCtl,OUTPUT);
  pinMode(pot,INPUT);
}

void loop() {
  int potVal = analogRead(pot);
  int motorVal = map(potVal,0,1023,0,255);
  analogWrite(motorCtl,motorVal);
  delay(10);
}
```





Code | Initializing... | Send To

1 (Arduino Uno R3)

```
int motorCtl = 3;
int pot = A0;

void setup() {
  pinMode(motorCtl,OUTPUT);
  pinMode(pot,INPUT);
}

void loop() {
  int potVal = analogRead(pot);
  int motorVal = map(potVal,0,1023,0,255);
  analogWrite(motorCtl,motorVal);
  delay(10);
}
```

Serial Monitor

Send | Clear

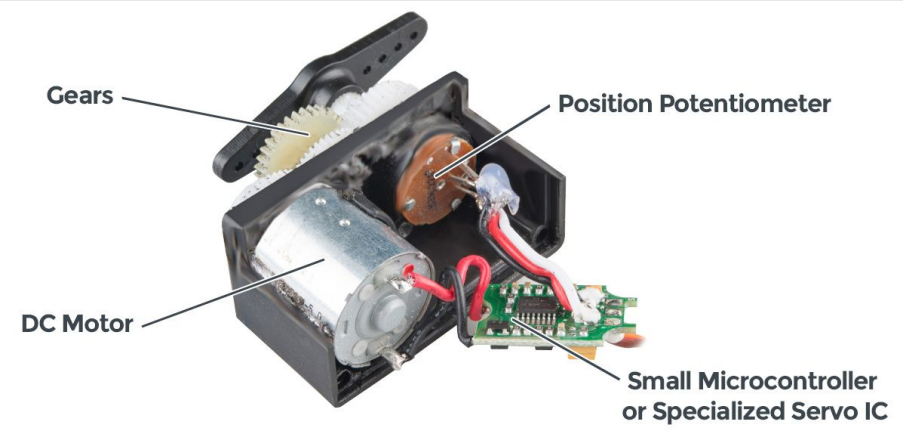
Servo Motor Controls

Intro: Servo Motors

Servos : Special motors for fine control (usually with limited range

Rotation Control : Done via position sensors

Uses : Printers, Robotics, Cameras, etc.

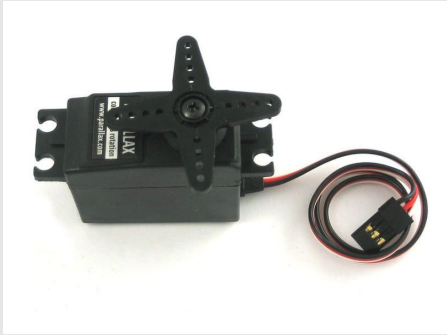


More on Servos

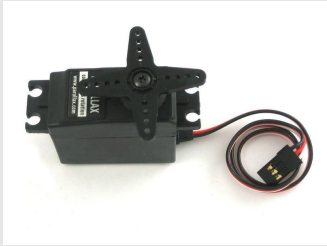
Linear : Can rotate to specific angel and easier to code.

Continuous Rotation : Can rotate continuously but harder to code

3 Terminals : Power, GND, and signal.



Control of Servos




```arduino
// Continuous Servo
#include <Servo.h>
Servo myservo;

int pos = 0;

void setup() {
  // The servo control wire is connected to Arduino D2 pin.
  myservo.attach(2);
  // Servo is stationary.
  myservo.write(90);
}

void loop() {
  // Servo spins forward at full speed for 1 second.
  myservo.write(180);
  delay(1000);
  // Servo is stationary for 1 second.
  myservo.write(90);
  delay(1000);
  // Servo spins in reverse at full speed for 1 second.
  myservo.write(0);
  delay(1000);
  // Servo is stationary for 1 second.
  myservo.write(90);
  delay(1000);
}
```

```arduino
// Linear Servo
#include <Servo.h>
Servo servo;

void setup()
{
  servo.attach(2);
}

void loop()
{
  // This variable is holding the target angle
  int target = 90
  // The read function has a margin of error which is accounted
  // for by the following if statement
  if(!(servo.read() > target-5 && servo.read() < target+5)){
    // This if and else statement rotate the motor in the proper
    // direction depending if the targe angle is higher or lower
    // than the current.
    if(target > servo.read()){
        for(int i =  servo.read(); i < temp; i+= 2){
            delay(50);
            servo.write(i);
        }
    }else{
        for(int i = servo.read(); i >= temp; i-= 2){
           delay(50);
           servo.write(i);
        }
    }
  }
}
```

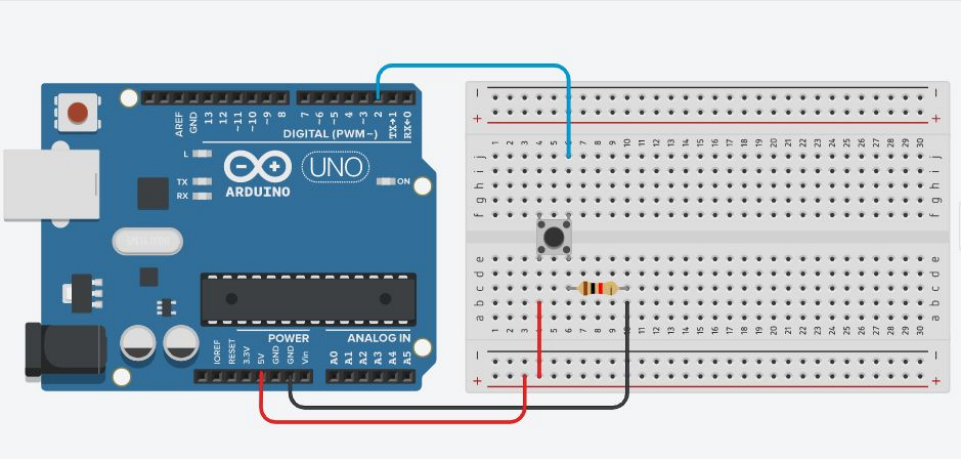
Application

Project: Servo Turn Control

Power : Low power required, arduino can provide it.

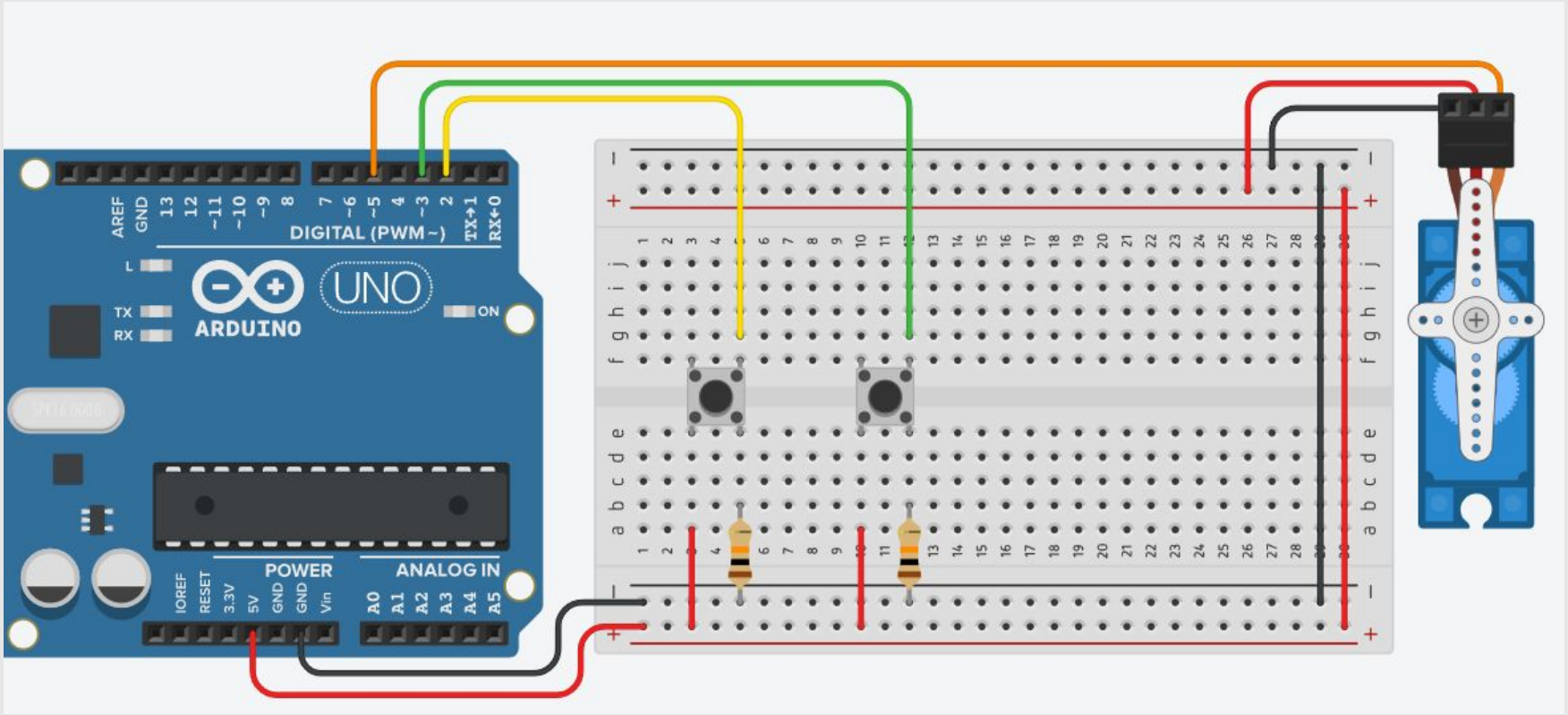
Servo Library : Has a variety of useful functions to control servo

Buttons : Arduino can read voltage from the digital pin that can detect if the button is pressed.



Reference: https://shorturl.at/dw8UG

Circuit


AREF
GND
13
12
~11
~10
~9
8
7
~6
~5
4
~3
2
TX→1
RX←0
DIGITAL (PWM~)
L
TX
RX
ARDUINO
UNO
ON
POWER
ANALOG IN
IOREF
RESET
3.3V
5V
GND
GND
Vin
A0
A1
A2
A3
A4
A5

Code

```
#include <Servo.h>

int signalPin = 3;
int right = 4;
int left = 5;

Servo myServo;
int angle = 0;

void setup() {
  pinMode(right,INPUT);
  pinMode(left,INPUT);
  pinMode(signalPin,OUTPUT);
  myServo.attach(signalPin);
}

```

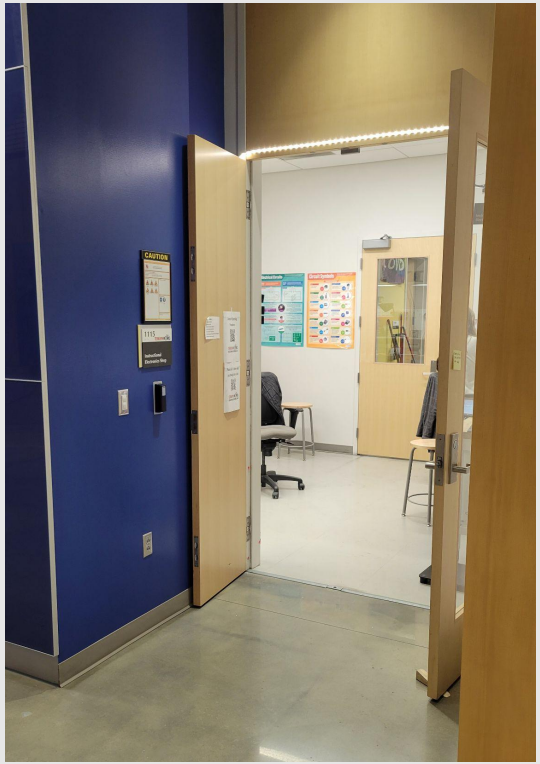
```
void loop() {
  #getting button inputs
  if(digitalRead(right) == HIGH)
    angle++;
  else if(digitalRead(left) == HIGH)
    angle--;

  #keeping angle within valid bounds
  angle = constrain(angle,0,180);

  #writing angle to servo motor
  myServo.write(angle);

  delay(10);
}
```

Come visit the IES!



1115 AJC

Open Lab 2:00- 7:00 PM Weekdays

Pleaes give us your feedback!
https://tinyurl.com/6eayw8r8

TERRAPIN WORKS