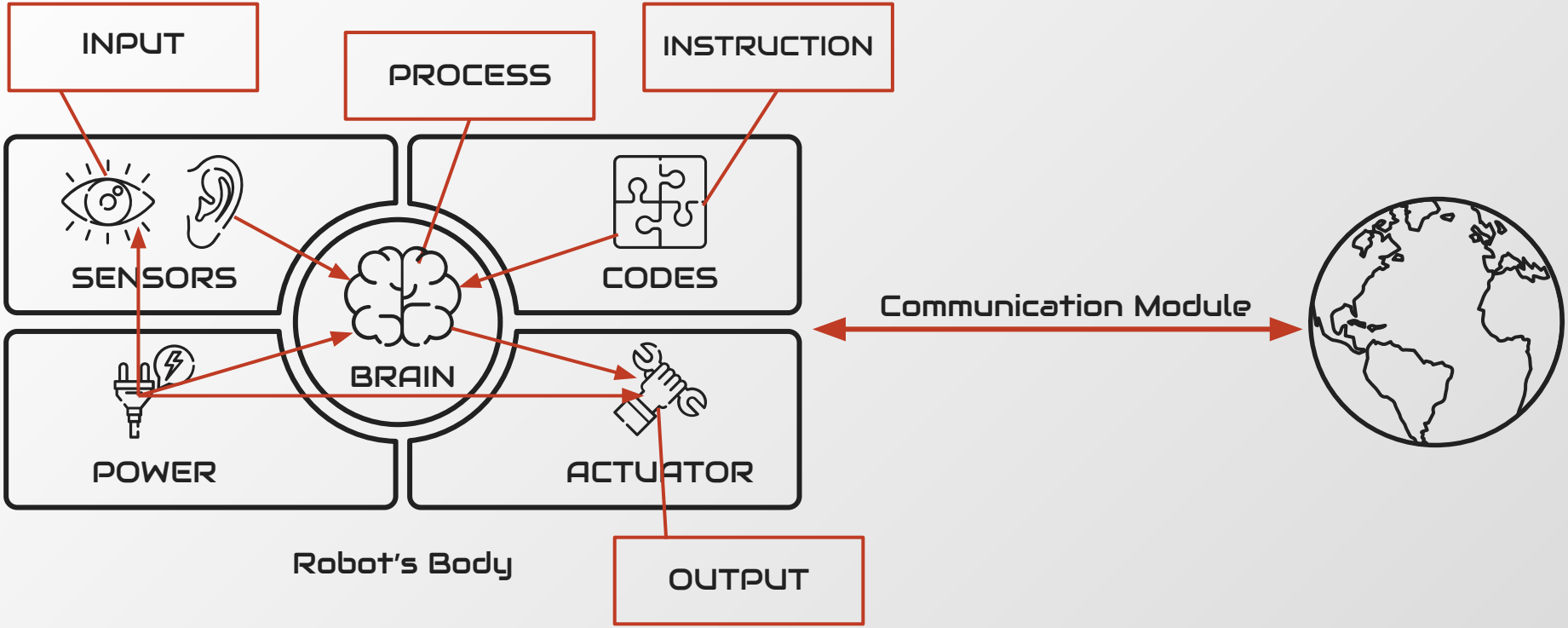
Let's Build A
RC Car

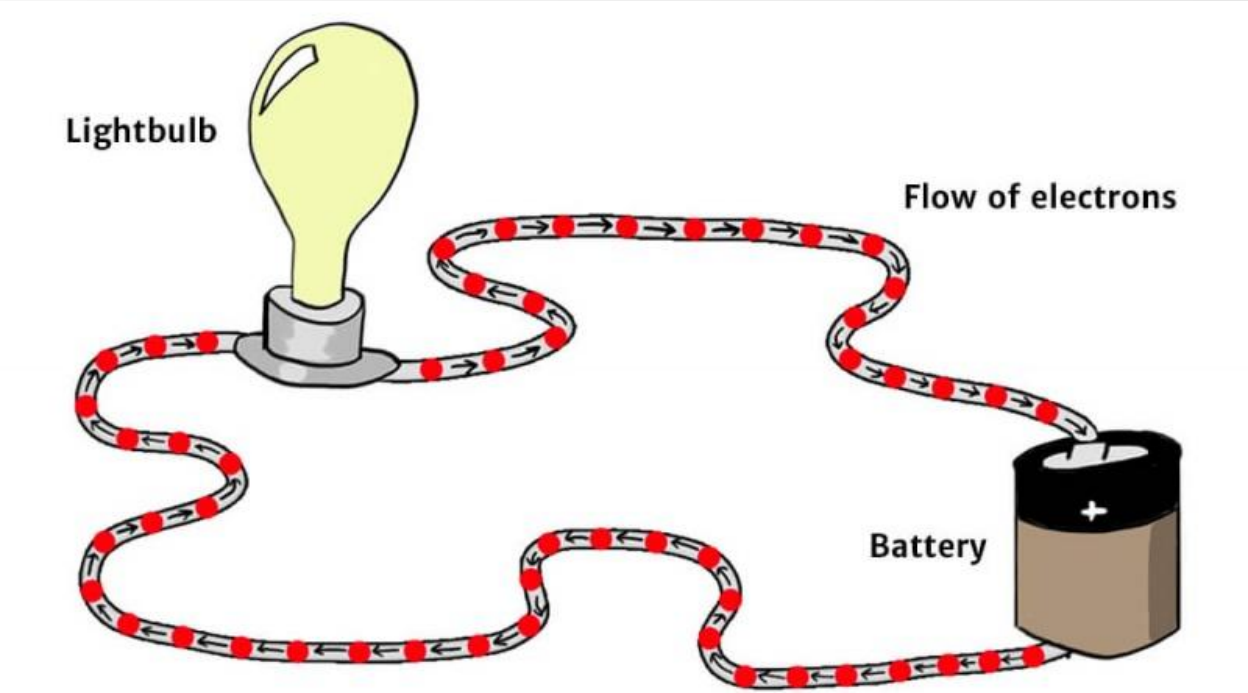
Powered by NSU Intelligent Robotics (NIRO) Lab


INPUT
PROCESS
INSTRUCTION
SENSORS
BRAIN
CODES
POWER
ACTUATOR
Communication Module
Robot's Body
OUTPUT

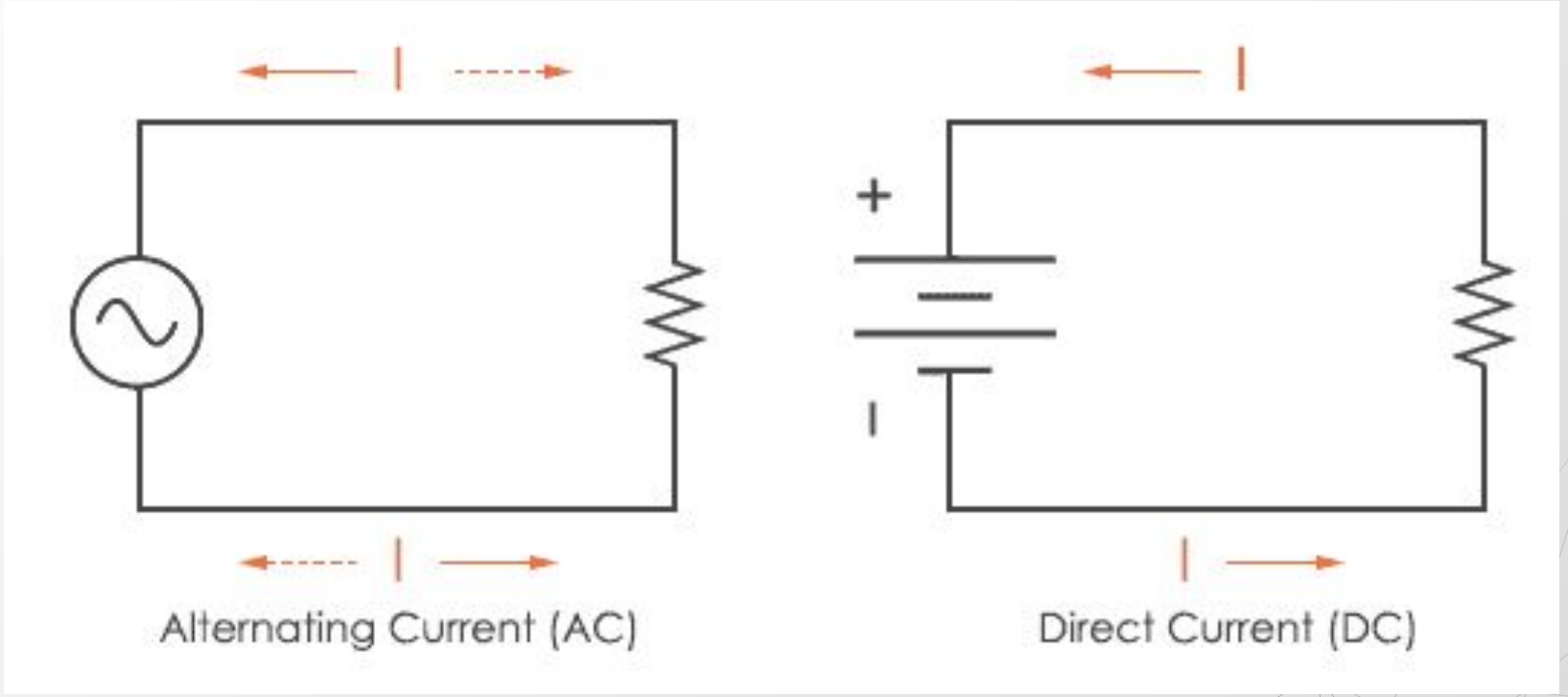
POWER

What Is Electricity?

Flow of electrons through a conductor.

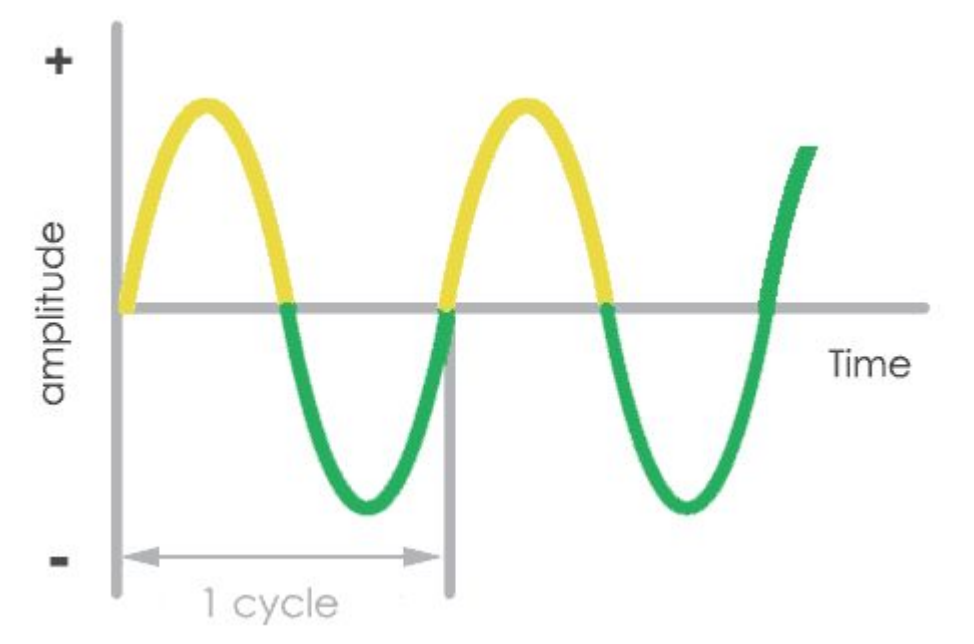


AC VS DC



Alternating Current (AC)

AC (Alternating Current) is an electric current that periodically reverses direction.



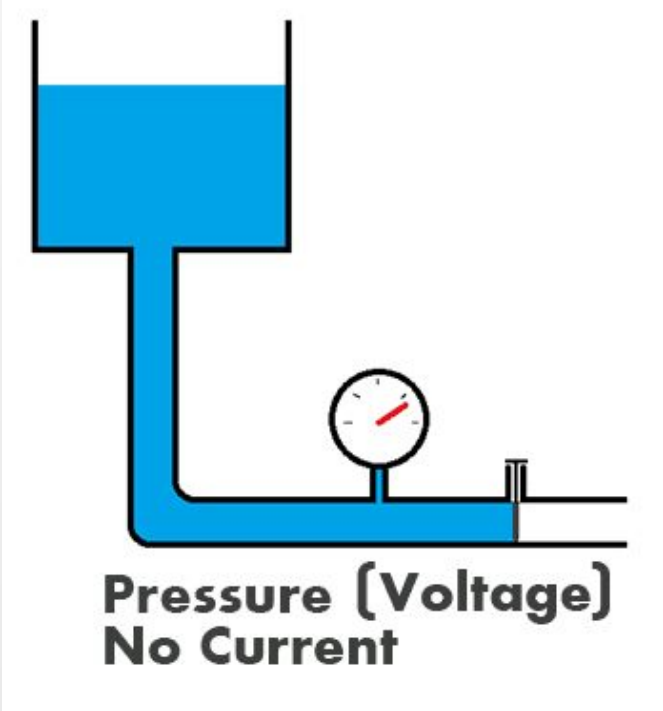
Direct Current (DC)

DC (Direct Current) is an electric current that flows in only one direction.

Voltage (V)

Voltage is the electrical pressure that pushes current through a circuit.

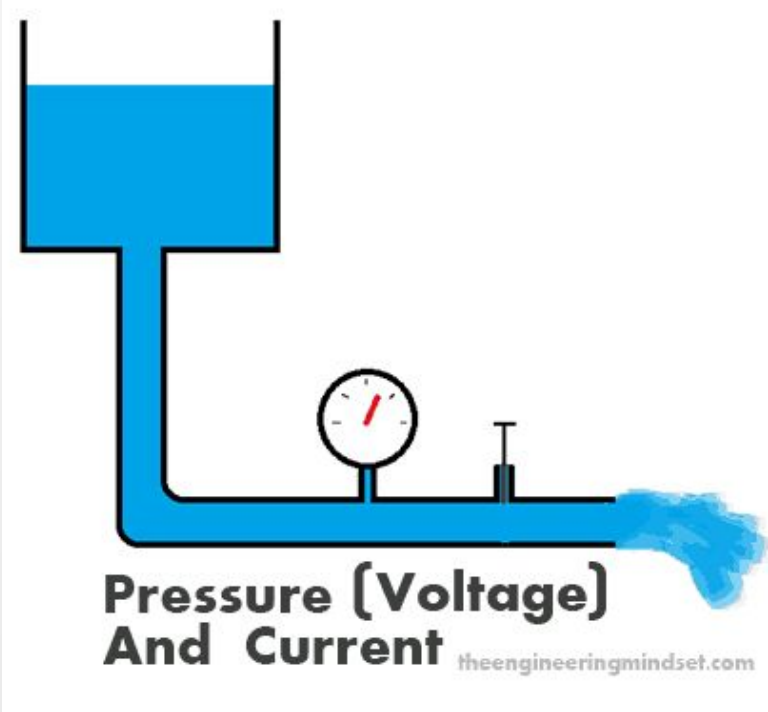
Unit: Volt (V)
Analogy: Think of it like water pressure in a pipe.
Formula: $V = I \times R$
Example: A 9V battery provides 9V of electrical pressure.



Current (I)

Current is the flow of electric charge in a circuit

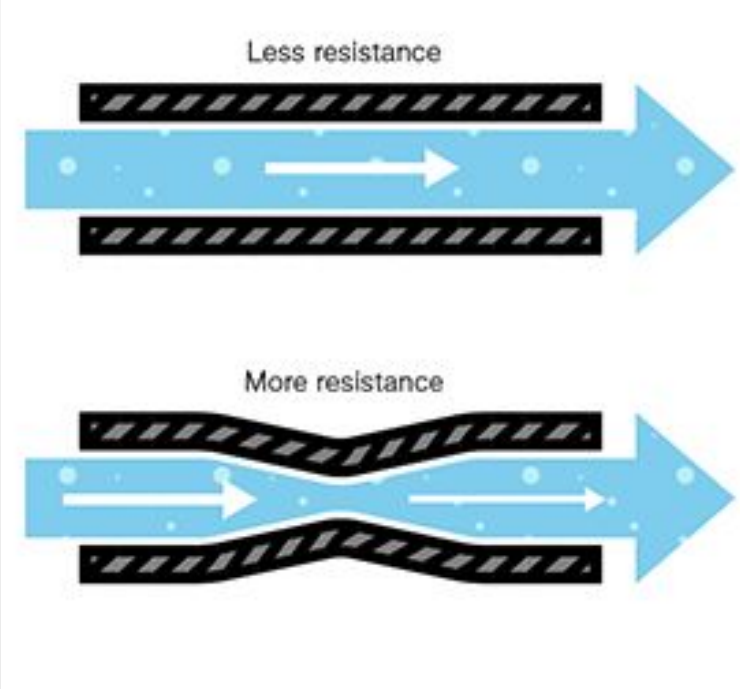
Unit: Ampere (A)
Analogy: Like water flowing through a pipe.
Formula: I = V / R
Example: A 1A current means 1 Coulomb of charge flows per second.



Resistance (R)

Resistance opposes the flow of electric current.

Unit: Ohm (Ω)
Analogy: Like a narrow pipe restricting water flow.
Formula: R = V / I
Example: A 100Ω resistor limits the current in a circuit.



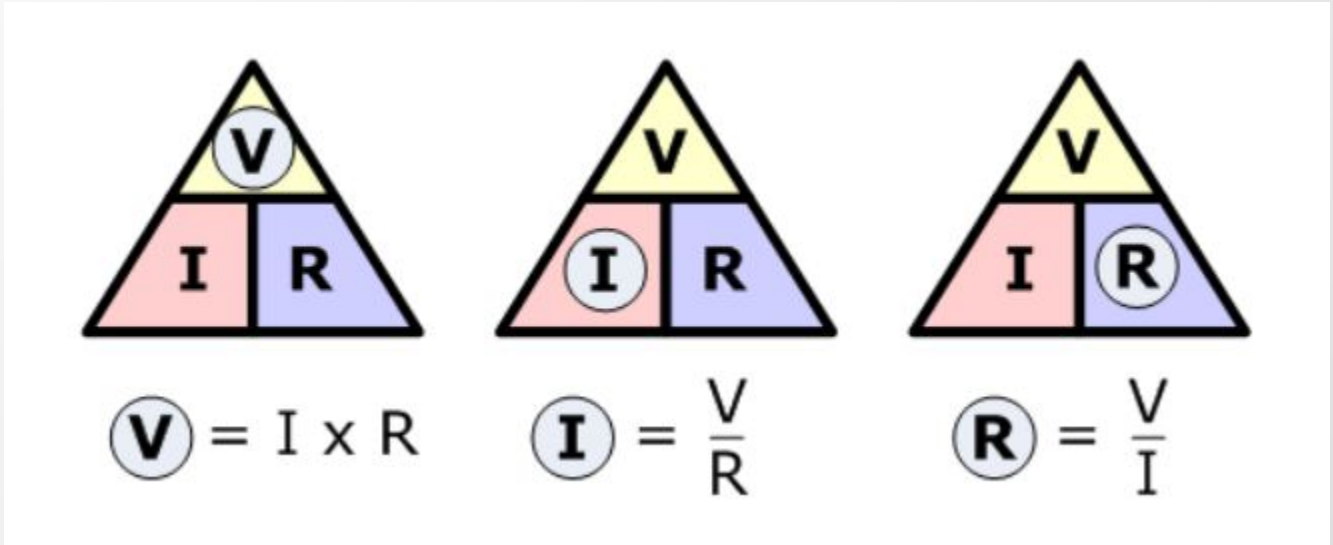
Power (P)

Power is the rate of electrical energy consumption.

Unit: Watt (W)
Formula: $P = V \times I$
Example: A 100W light bulb uses 100 watts of power.

Ohm's Law

$V = I \times R$

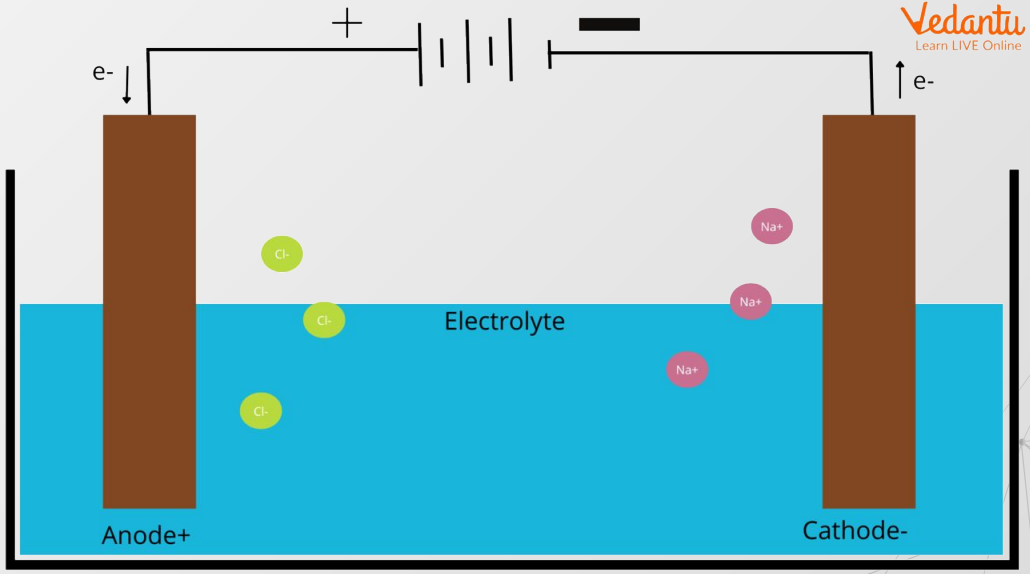


What is a Battery?

A device that stores chemical energy and converts it into electrical energy.

Basic Structure: Anode (-), Cathode (+), and Electrolyte.
Types Based on Rechargeability:

- Primary Batteries (Non-rechargeable) → Use once, then discard.
- Secondary Batteries (Rechargeable) → Can be recharged multiple times .



Primary Batteries (Non-Rechargeable)

Alkaline Batteries (1.5V)
(left to right) C, AA, AAA, N, and a 9-volt (PP3)
Common in remotes, clocks, toys

Zinc-Carbon Batteries (1.5V)
Used in low-power devices (torches, remotes)

Lithium Batteries (3V – 9V)
Found in cameras, watches, pacemakers

Secondary Batteries (Rechargeable)

Lithium-Ion (Li-ion) Batteries (3.7V)
Found in laptops, smartphones, electric vehicles

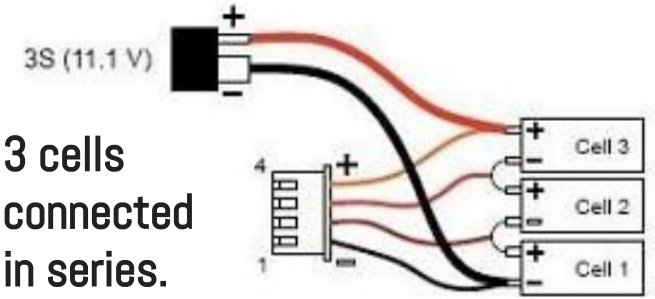
Lithium-Polymer (Li-Po) Batteries (3.7V)
Used in drones, RC cars, high-performance gadgets

Lead-Acid Batteries (12V)
Used in cars, UPS, solar systems

Nickel-Cadmium (NiCd) Batteries (1.2V)
Used in power tools, emergency lights

3S Li-Po

Cell Count (S): Defines the voltage. (e.g., 3S = 3 * 3.7V =11.1V)
Capacity (mAh): Energy storage (e.g., 2200mAh = 2.2Ah. It can provide 2.2A for an hour)
C-Rating (Discharge Rate): Determines how fast energy can be delivered.
Example: 20C rating on a 2200mAh battery
Max discharge = 2.2 × 20 = 44A.



3 cells connected in series.

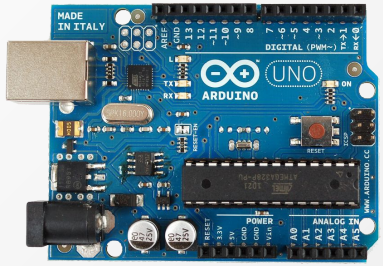
Li-Po Charger

iMAX B6
Multi-Function LiPo
Balance Charger

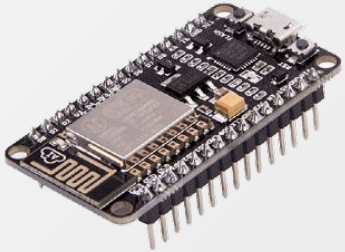
Imax B3 Ac Compact
Balance Charger

BRAIN

Microcontrollers

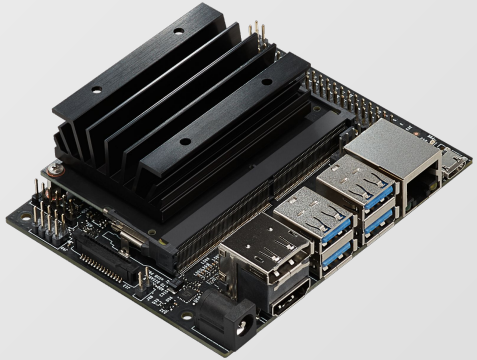


ARDUINO UNO



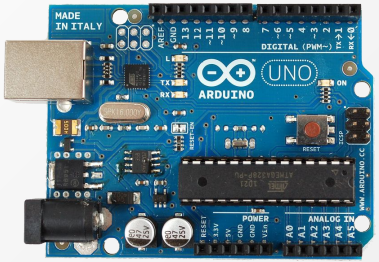
ESP 32

RASPBERRY PI

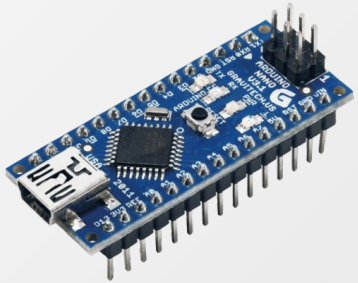


JETSON NANO

Different Types Of Arduino

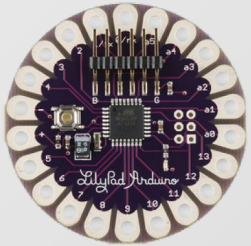


ARDUINO UNO



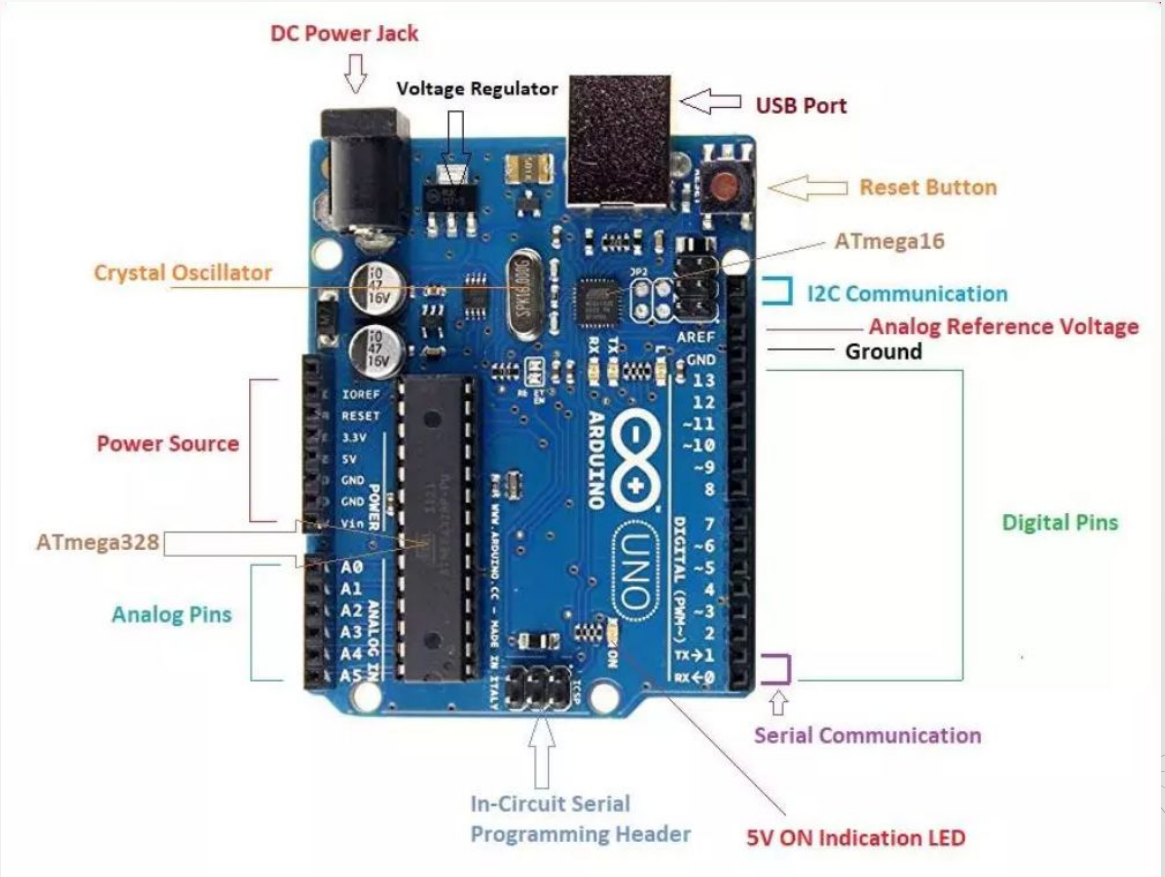
ARDUINO NANO

ARDUINO MEGA

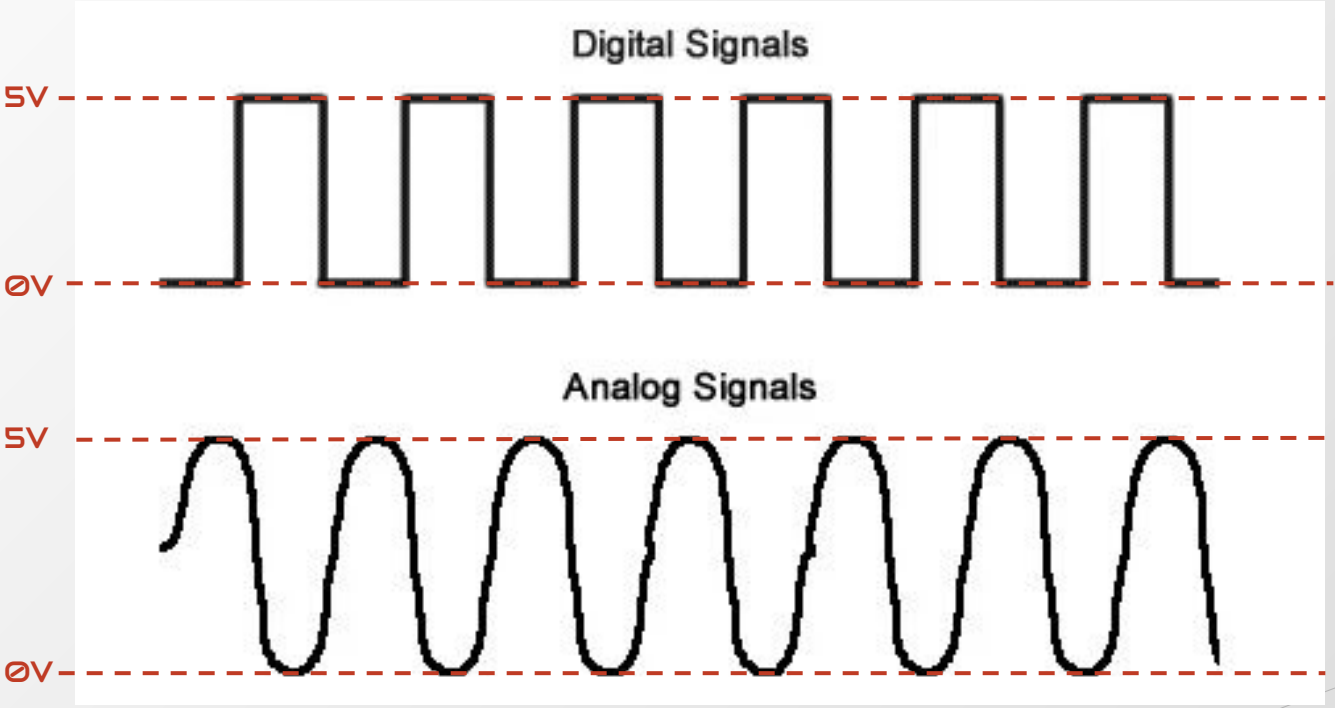


ARDUINO LILY

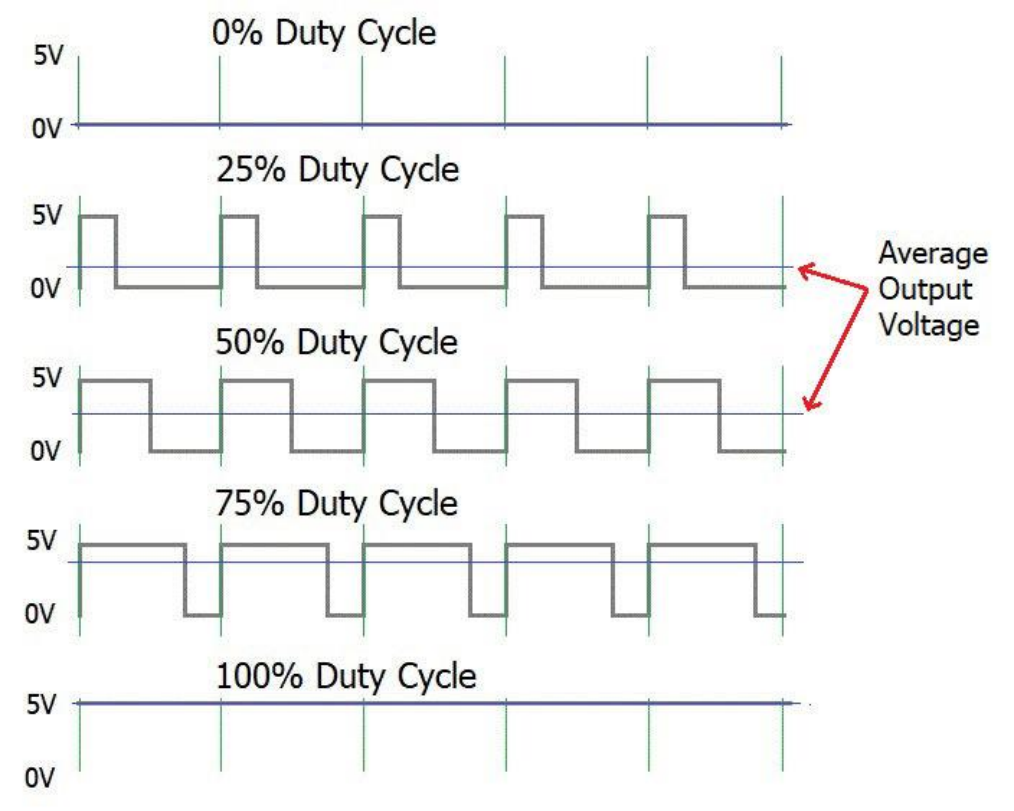
Arduino Uno


DC Power Jack
Voltage Regulator
USB Port
Reset Button
ATmega16
Crystal Oscillator
I2C Communication
Analog Reference Voltage
Ground
Power Source
Digital Pins
ATmega328
Analog Pins
Serial Communication
In-Circuit Serial Programming Header
5V ON Indication LED

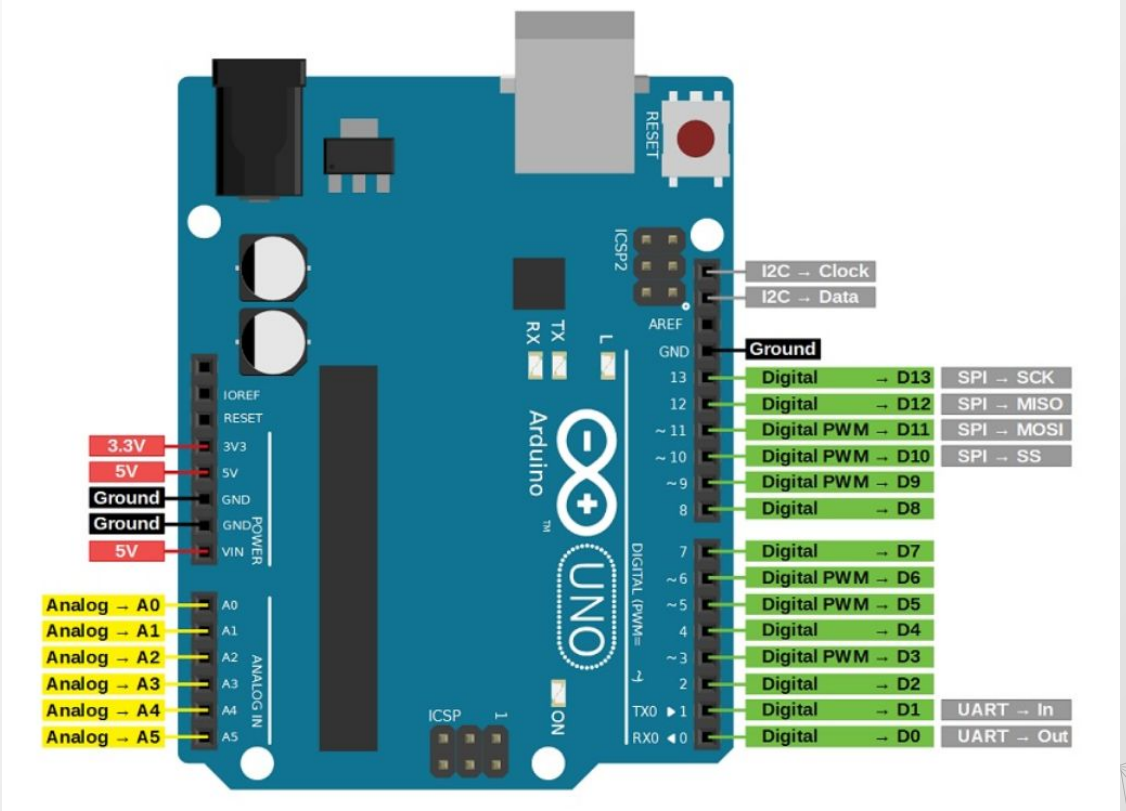
Digital VS Analog Signals



PWM: Pulse Width Modulation



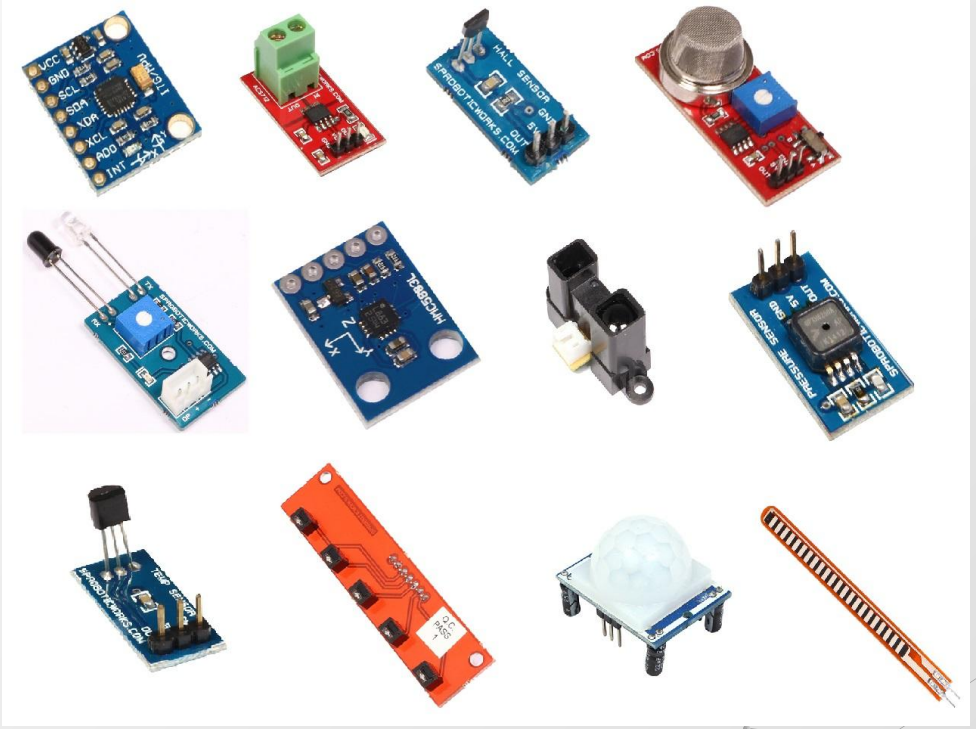
Arduino Uno Pin Diagram



SENSORS

What are Sensors?

Devices that detect and respond to physical inputs like temperature, light, sound, and motion etc.



Classification of Sensors

Active vs. Passive Sensors

- ★ **Active:** Require external power (e.g., IR sensor, ultrasonic sensor).
- ★ **Passive:** Work without external power (e.g., thermocouples).

Analog vs. Digital Sensors

- ★ **Analog:** Continuous signals (e.g., temperature sensor).
- ★ **Digital:** Discrete signals (e.g., push button, PIR sensor).

Camera Sensors

Applications: Robotics, Surveillance, Object Detection, Computer Vision.

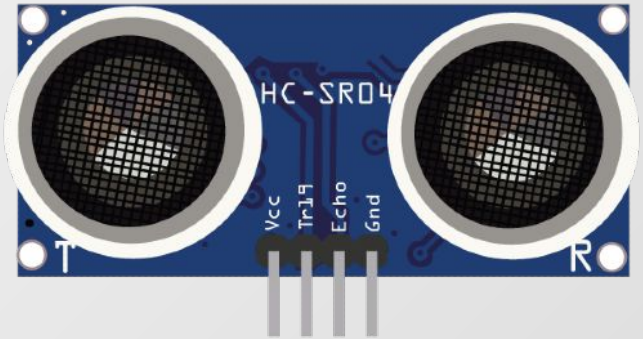
Demo Idea: Use an Raspberry Pi camera for object tracking.

Ultrasonic Sensors(HC-SR04)

Emits ultrasound waves (40kHz), calculates distance based on time taken for echo to return.

Applications: Obstacle detection in robots, water level monitoring, car reverse parking sensors etc..

Demo Idea: Use an Arduino + HC-SR04 to measure distance and display it on an LCD.



Infrared (IR) Sensors

Emit & detect IR light

Applications: Security systems, gesture control, line-following robots.

Demo Idea: Show an IR remote working with an Arduino receiver module.

IR Array Sensors

Uses multiple IR sensors arranged in an array to detect line position.

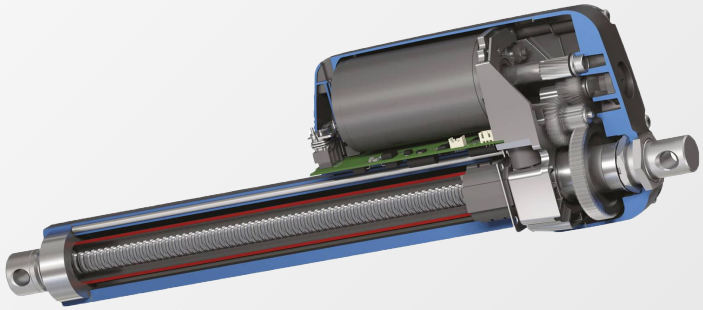
Applications:
Line-following robots, object tracking.

Demo Idea: Show a line-following robot using an IR array sensor.

ACTUATOR

What are Sensors?

An actuator is a machine that moves or controls components in a system by converting energy into physical motion



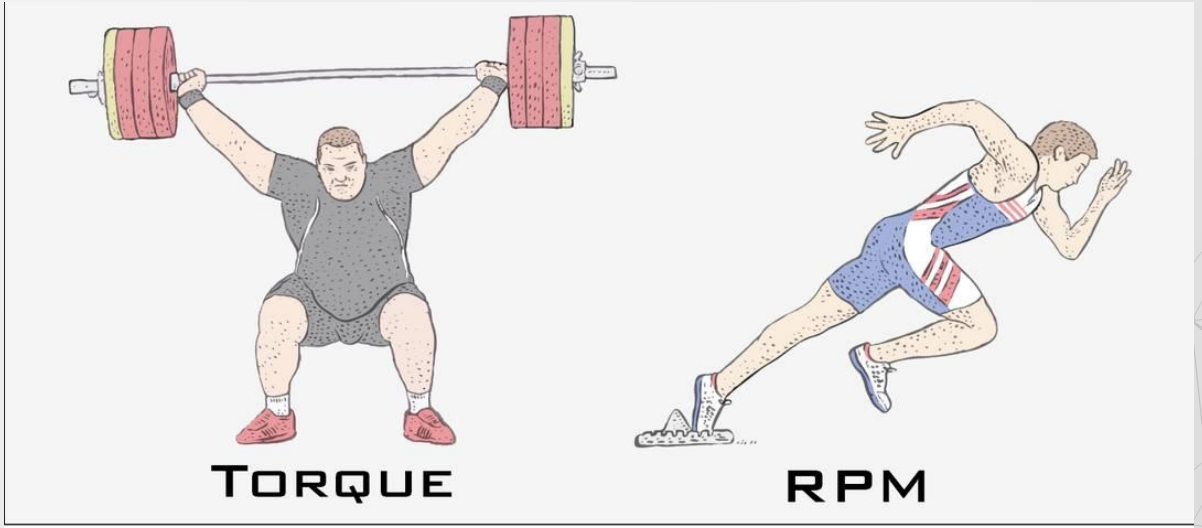
ACTUATOR

DC Motor

Torque vs RPM (Speed)

Torque (T): The rotational force a motor produces. Measured in Newton-meters (Nm).
RPM (Revolutions Per Minute): The speed at which the motor rotates.

Relationship:
Higher torque > Lower RPM
Higher RPM > Lower torque



Basic DC Motor

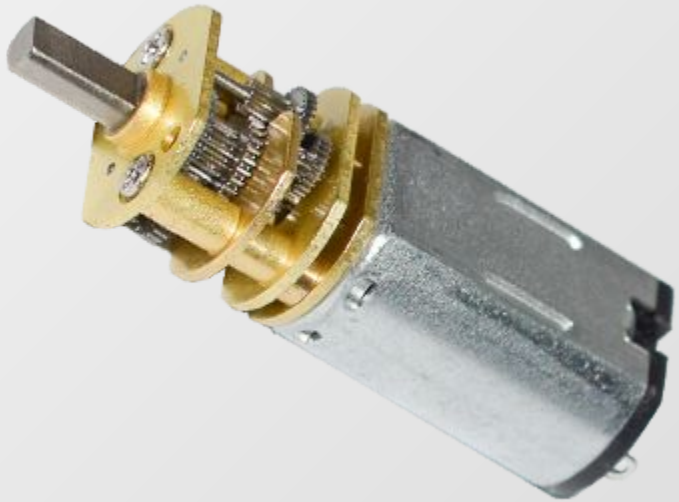
Simple two-wire motor
High RPM, Low Torque
Works when connected to a battery
Example: Small toy motors

Yellow Gear Motor

Geared DC motor with low RPM and high torque
Common in robotics (used in small robot wheels)
Operates at 3V-12V

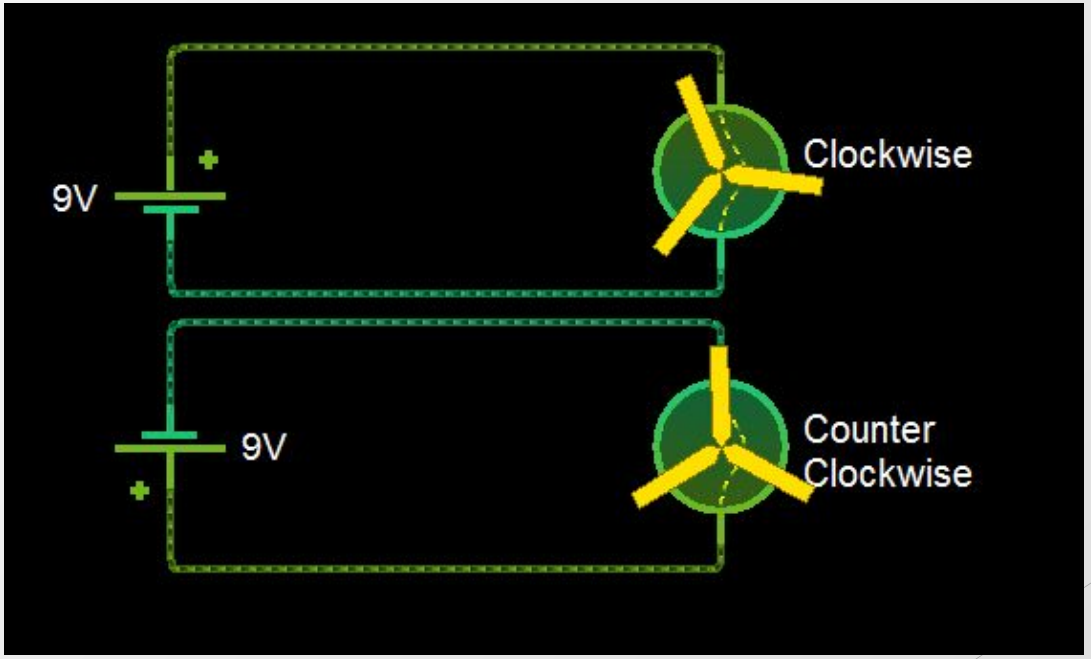
Geared DC Motor

A DC motor with a gearbox
Increases torque and reduces RPM
Used in robot arms, electric cars, conveyor belts



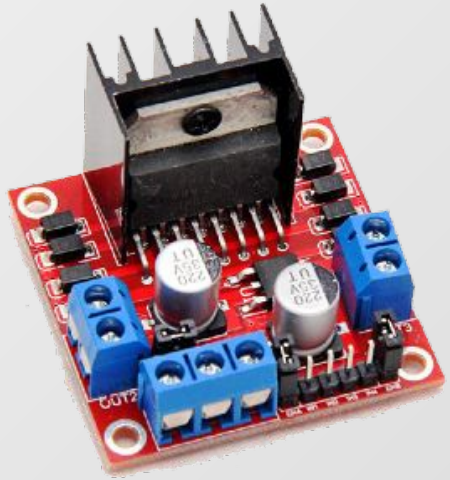
Motor Rotation Direction

DC motors can rotate in either direction (clockwise or counter-clockwise) and you can easily change the direction by inverting the polarity of the applied voltage.



L298N Motor Driver (H-Bridge)

A motor driver is an electronic circuit that allows a microcontroller (low power) to control a motor (high power) safely.



Can drive 2 DC motors
Works with 5V-35V motors
Can supply 2A per channel
Has built-in heat sink
Control Pins:

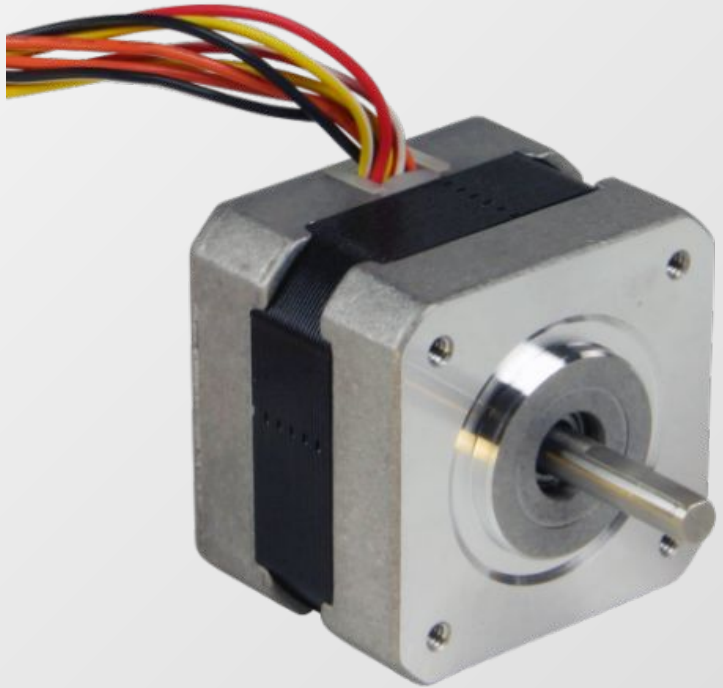
- IN1, IN2 → Motor 1
- IN3, IN4 → Motor 2
- ENA, ENB → Speed control (PWM)

Servo Motor

Precise position control (e.g., 0° to 180° rotation)
Uses PWM signals for control
Example: SG90, MG995
Used in robotic arms, RC cars, mechanical systems

Stepper Motor

Moves in small steps (e.g., 1.8° per step)
Precise angular control
High torque at low speeds
Used in CNC machines, 3D printers



Brushless DC (BLDC) Motor

No physical brushes → More efficient & long-lasting
Used in drones, electric vehicles, industrial machines
Requires an ESC (Electronic Speed Controller) for operation

Ball Castor / Castor Wheel

A castor wheel, also known as a caster wheel or simply a castor, is a small, undriven wheel designed to be attached to the bottom of a larger object to facilitate easy movement

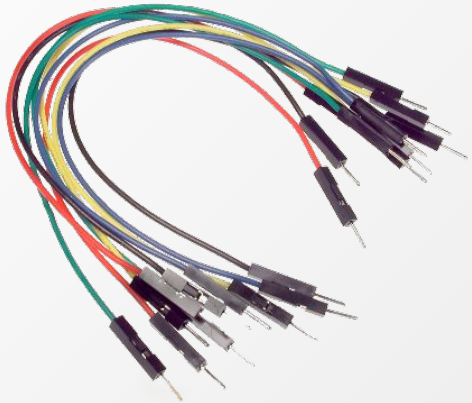
DC MOTOR SET

WIRES & CONNECTORS

Wires

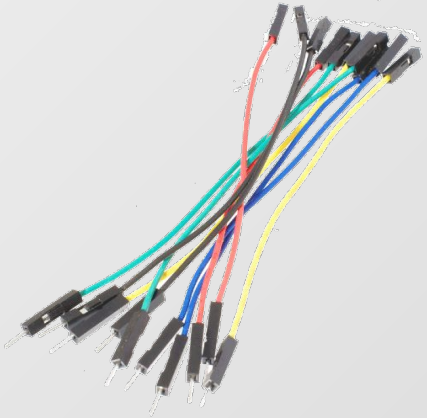
Why are wires important?
Wires transmit electrical signals or power between components. Choosing the right wire prevents overheating, voltage drops, and circuit failures.

Jumper Wires



Male to Male
Used in breadboards

Female to Female
Used for sensor modules



Male to Female
Used for connecting microcontrollers to sensors

Rocket Switch

Used to turn ON/OFF power circuits.

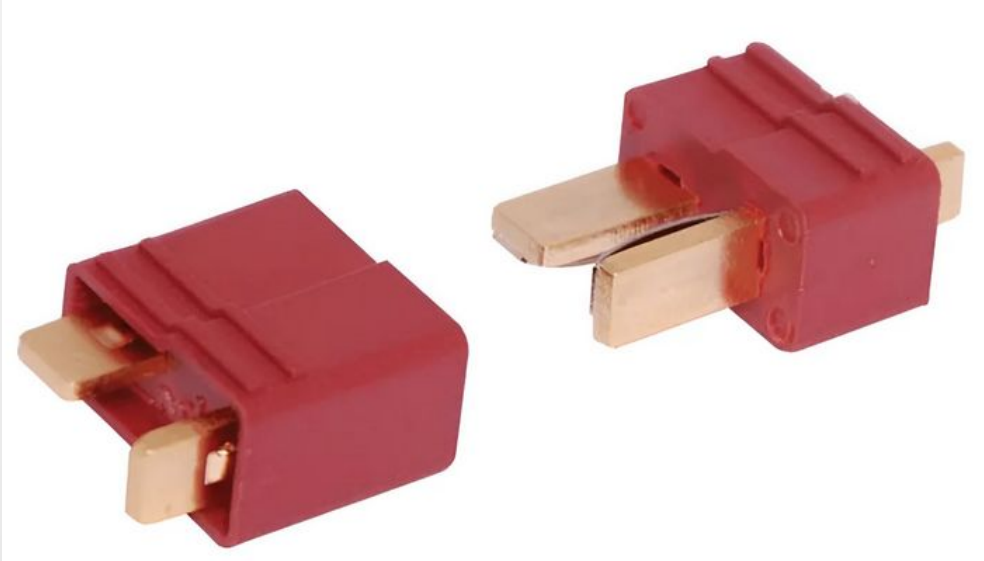
Types of Switches:

SPST (Single Pole Single Throw) → Basic ON/OFF switch

DPDT (Double Pole Double Throw) → Can reverse motor direction

T-Plug (Deans Connector)

Used in high-current applications (e.g., drones, RC cars, Li-Po batteries).
Supports up to 50A current.
Provides a secure connection, minimizing power loss.
Example: Connecting a 3S Li-Po battery to a motor driver



What is a PCB?

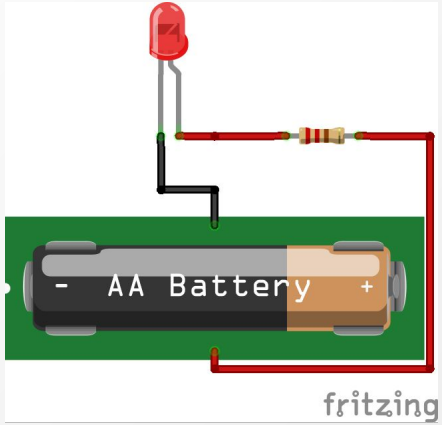
PCB (Printed Circuit Board) is a solid board that holds electronic components.

Fritzing

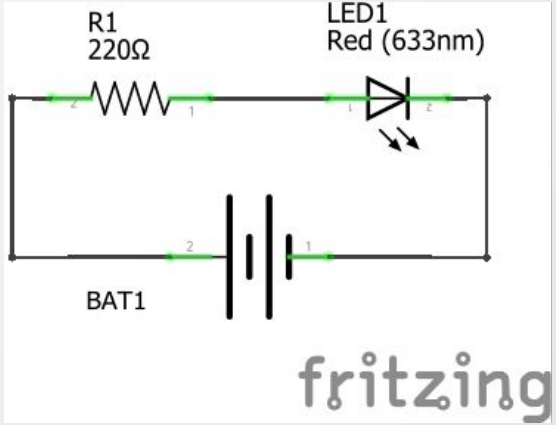
Fritzing is an open-source hardware initiative that makes electronics accessible as a creative material for anyone.

Link: https://fritzing.org/

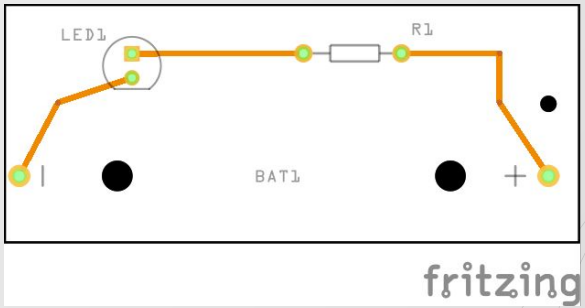
Wiring Using Fritzing



Breadboard Layout



Schematic Layout



PCB Layout

CODES

Arduino IDE

Integrated Development Environment for writing, compiling, and uploading code to Arduino.

First-Time Setup:

Connect Arduino board via USB.

Select Board & Port:
Tools > Board & Port

Open Example Sketch:
File > Examples > Basics > Blink

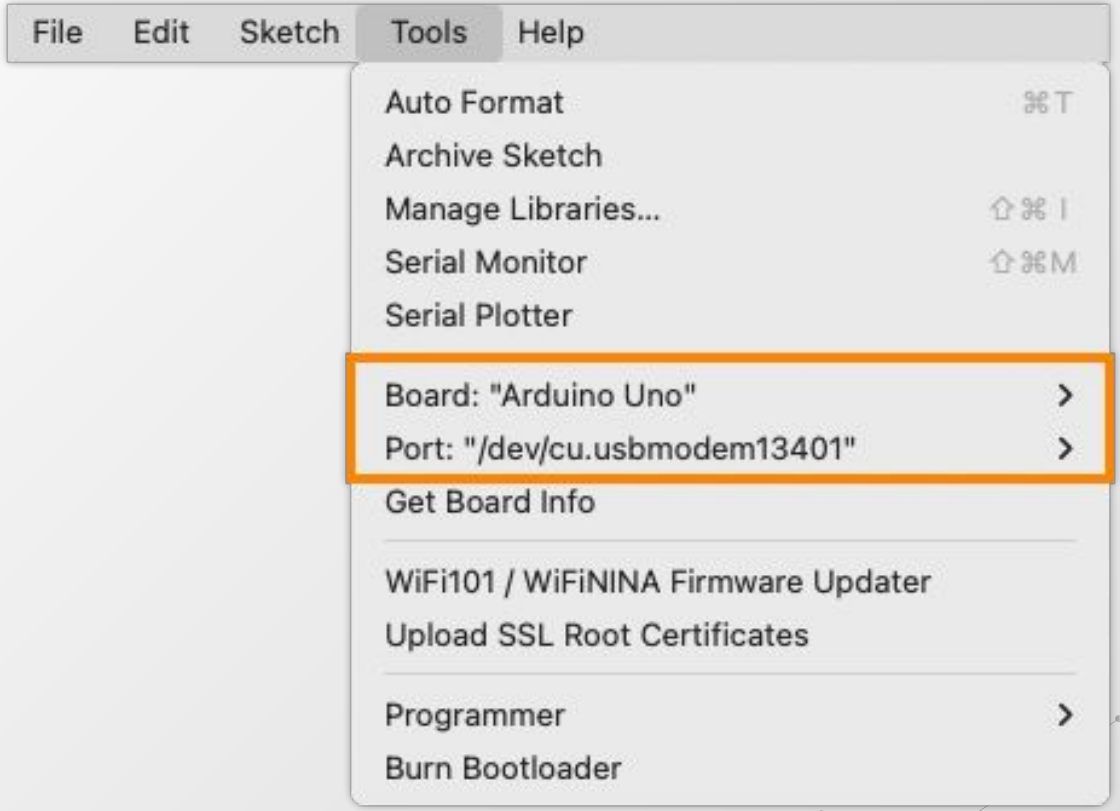
Download from
www.arduino.cc

Supports C/C++ language with additional Arduino functions.

Arduino IDE

Documentation: https://docs.arduino.cc/software/ide-v1/tutorials/Environment/

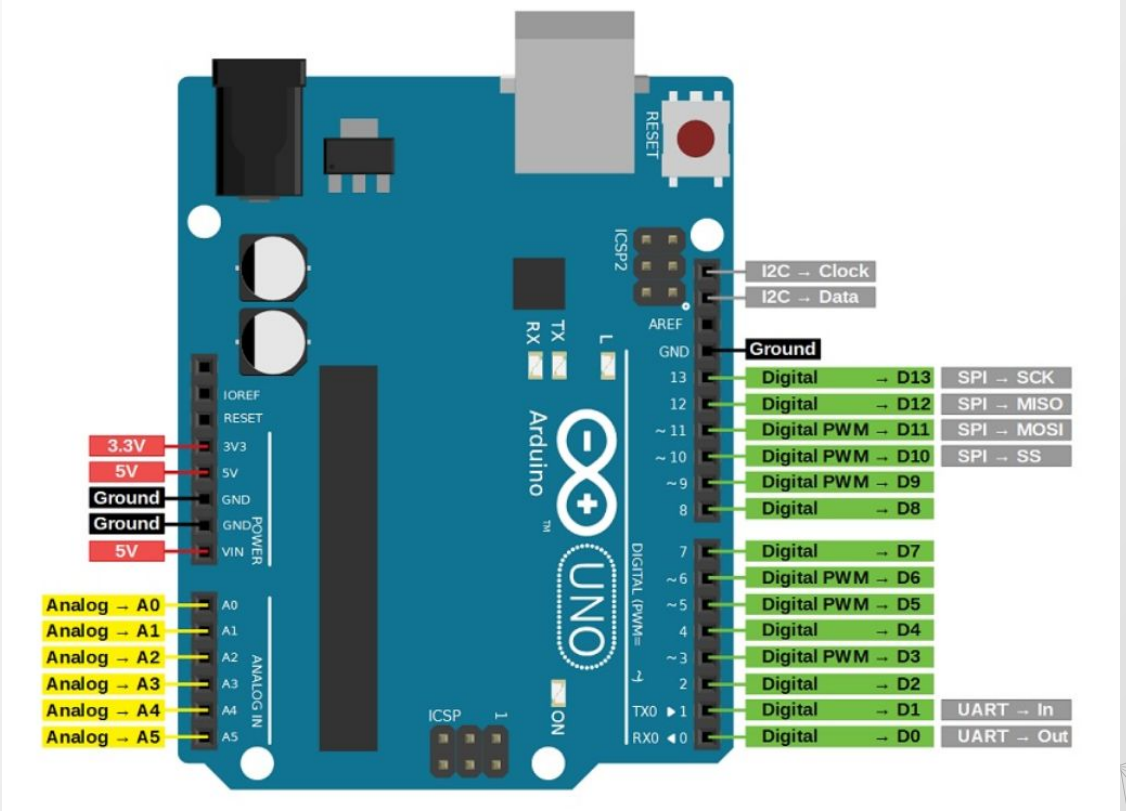
Select Board & Port



Arduino Sketch (Code) Format:

```
void setup() {
  // put your setup code here, to run once:
  // Used to initialize pins, start communication, etc
}
void loop() {
  // put your main code here, to run repeatedly:
  // Runs continuously after setup().
}
```

Arduino Uno Pin Diagram



pinMode(PinNumber, Mode)

Configures a pin as INPUT, OUTPUT, or INPUT_PULLUP.

```
void setup() {
  pinMode(7, INPUT);      // Set pin 7 as input
  pinMode(13, OUTPUT);    // Set pin 13 as output
  pinMode(A1, OUTPUT);    // Set pin A1 as output
  pinMode(2, INPUT_PULLUP); // Enable internal pull-up resistor
}
```

digitalWrite(pin, value)

Writes HIGH (1 = 5v) or LOW (0 = 0v) to a digital pin.

```
void loop() {
digitalWrite(13, HIGH); // Turn LED ON
digitalWrite(13, LOW);  // Turn LED OFF
}
```

digitalRead(pin)

Reads HIGH (1 = 5v) or LOW (0 = 0v) from a digital input pin.

```
void loop() {
int sensorValue = digitalRead(7);  // Read
sensor data
}
```

analogWrite(pin, value)

Writes a **PWM (Pulse Width Modulation)** or **Analog** signal **(0-255)** to a **PWM supported pin (~)** or **Analog Pin (A0 - A5)** . Here 0 = 0v and 255 = 5v and 1 - 254 means between 0v - 5v

```
void loop() {
analogWrite(9, 128); // Set 50% brightness on
LED (PWM)
analogWrite(A1, 128);  // Set 50% brightness on
LED
}
```

analogRead(pin)

Reads analog values (0-1023) from an analog pin (A0-A5) .

```
void loop() {
int sensorValue = analogRead(A0);  // Read
sensor data
}
```

delay(ms)

Pauses execution for **ms** milliseconds.

```
void loop() {
delay(1000);  // Wait for 1000 ms or 1 second
}
```

Serial.begin(baudrate)

The Arduino Serial Monitor is a built-in tool in the Arduino IDE that allows you to send and receive data between your Arduino board and a computer, facilitating debugging, monitoring sensor readings, and interacting with your projects in real-time

Starts Serial Communication with a defined baudrate.

```
void loop() {
Serial.begin(9600);   // Start Serial Monitor at
9600 baud
}
```

Serial.print(value) / Serial.println(value)

Prints data to the Serial Monitor.

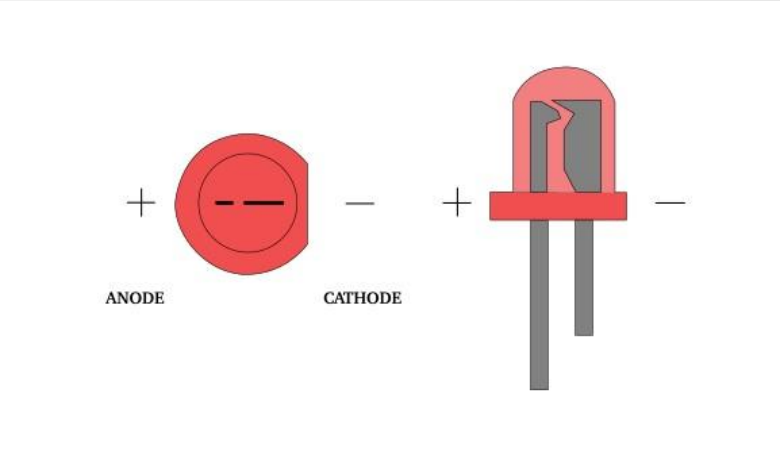
```
void loop() {
Serial.print("Sensorvalue: ");
Serial.println(sensorValue);
}
```

Example 1: Blink LED (Basic Output)

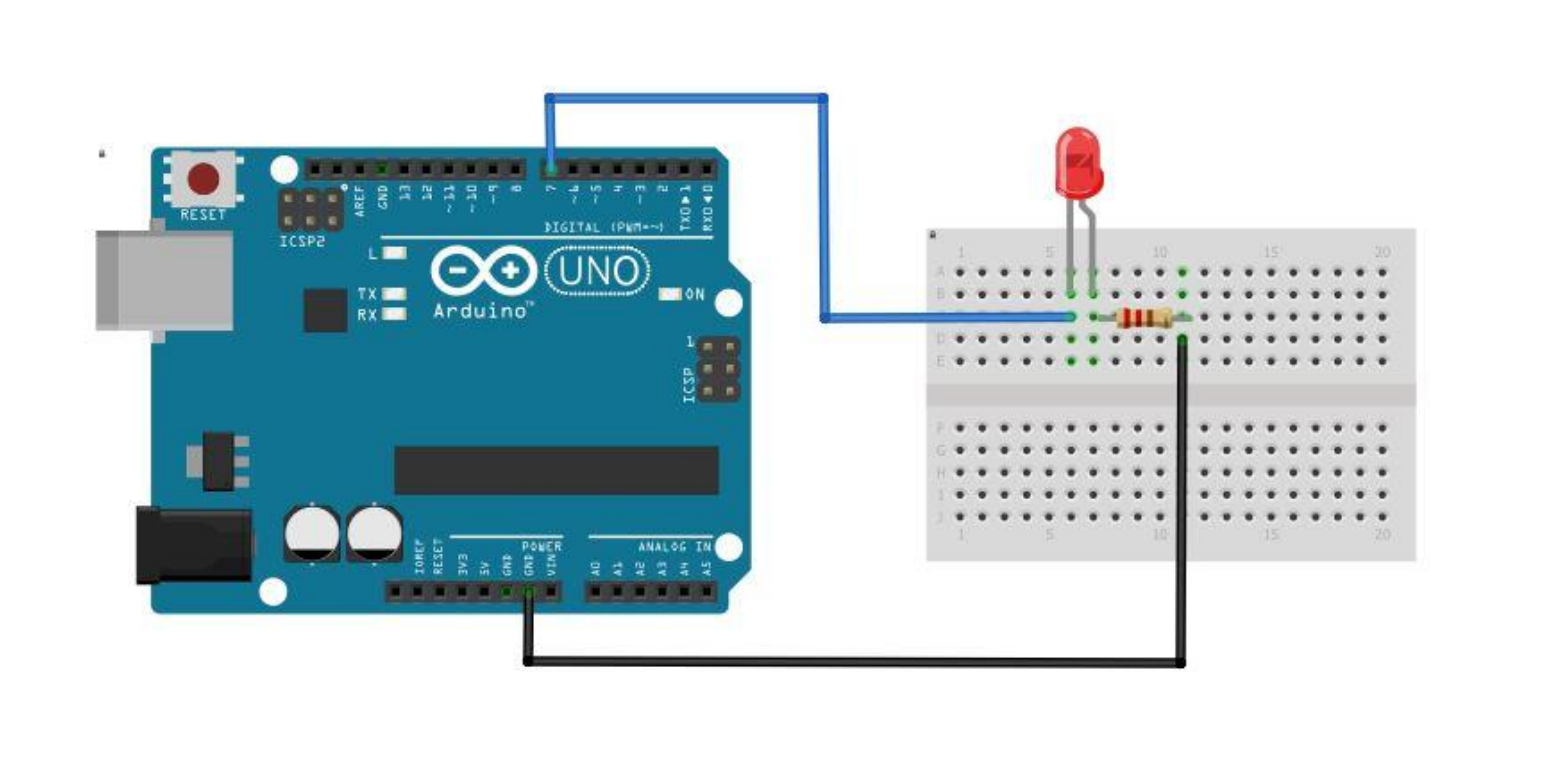
```
int ledPin = 7;// The LED is connected to pin 7
void setup() {
   pinMode(ledPin, OUTPUT);// Set pin 7 as an OUTPUT pin
}
void loop() {
  digitalWrite(ledPin, HIGH);// Turn the LED ON (HIGH = 5V)
  delay(1000);// Wait for 1000 milliseconds (1 second)
  digitalWrite(ledPin, LOW);// Turn the LED OFF (LOW = 0V)
  delay(1000);// Wait for another 1 second
}
```

LED (Light Emitting Diode)

LEDs have two leads. One lead is shorter, that is the cathode(-ve) and another one is longer, that is the anode(+ve). You can also identify the cathode by spotting the flat side of the LED and another one is the anode. LEDs have polarities. So the long leg should connect to a digital pin on the Arduino board. And shorter legs should go to the GND.



Example 1: Blink LED (Basic Output)



COMMUNICATION DEVICES

Bluetooth Module (HC-05)

Enables wireless serial communication (UART) between devices.

Used for smart home automation, robotic control, and IoT projects.

FlySky 6CH RC Radio System

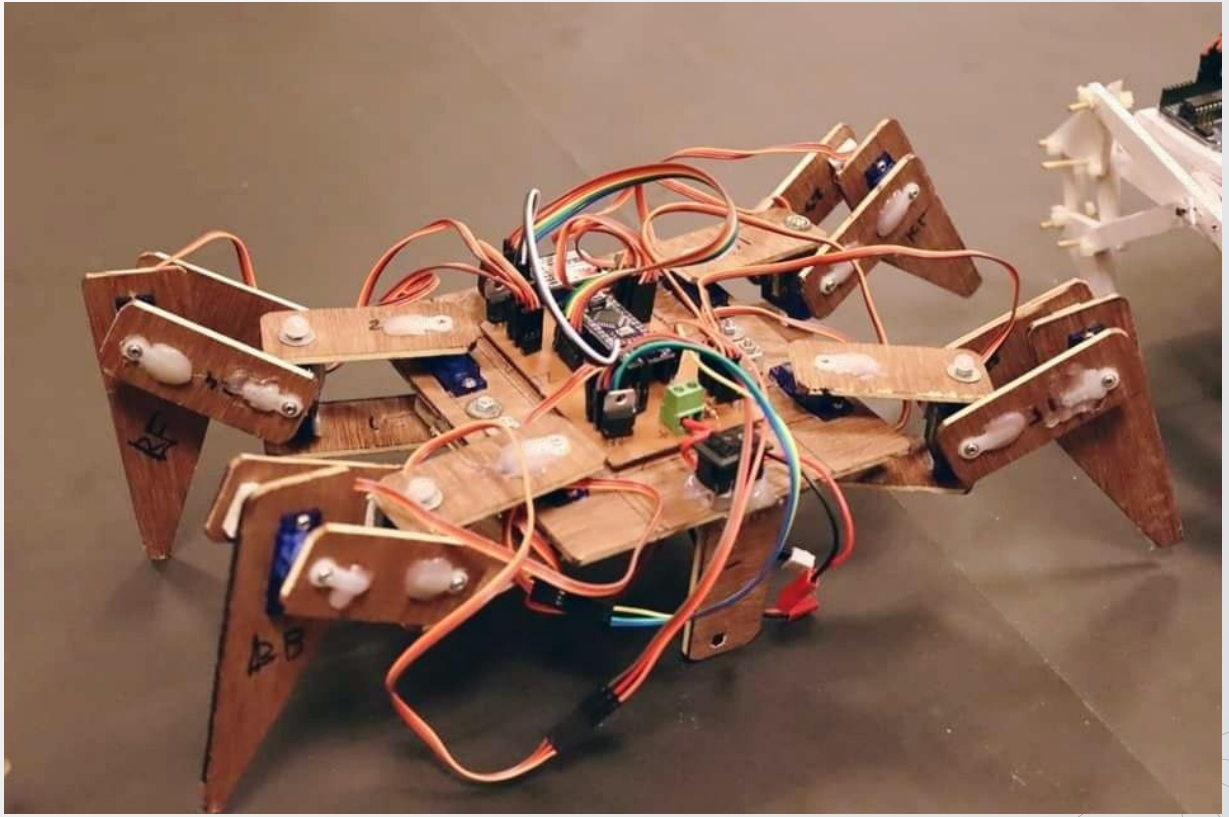
FlySky is a radio control system used to wirelessly control RC cars, boats, drones, and robots.

Low Latency: Instant response for drones/robots.
Long Range: 500m–2km communication.
Multiple Channels: Control servos, ESCs, and motors.
Failsafe Mode: Prevents crashes if the signal is lost.

BODY

5mm PVC Robot Chassis

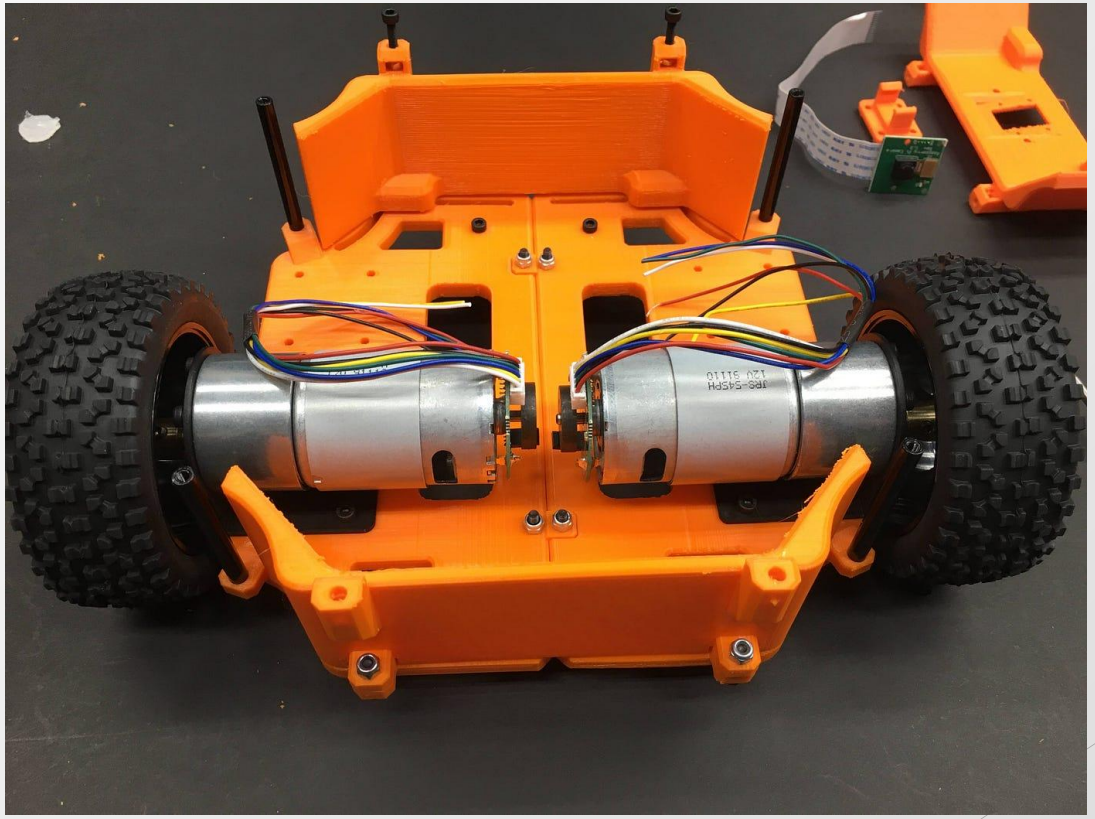
3mm Plywood Robot Chassis



Acrylic Robot Chassis

Metal Robot Chassis

3D Printed Robot Chassis



3D Design Tools

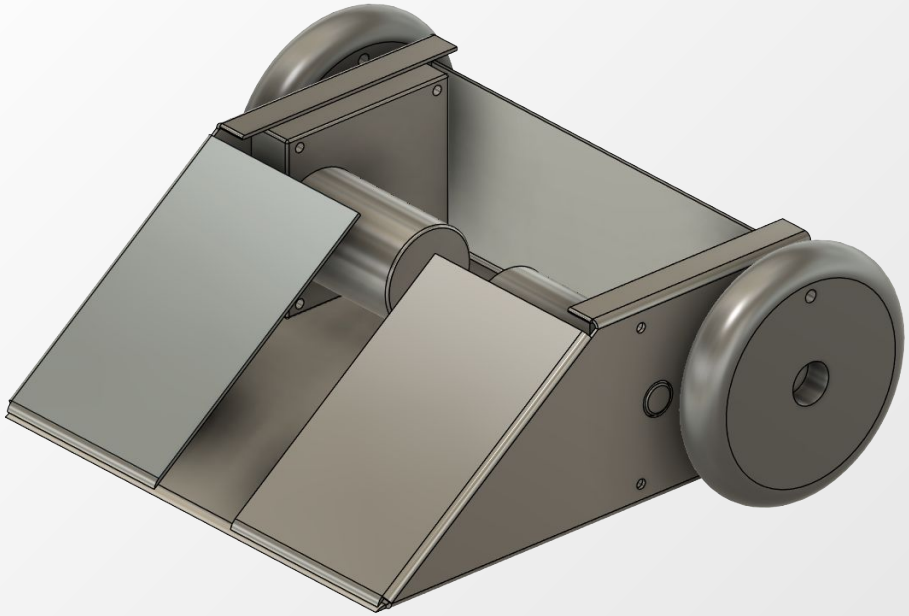
Solidworks

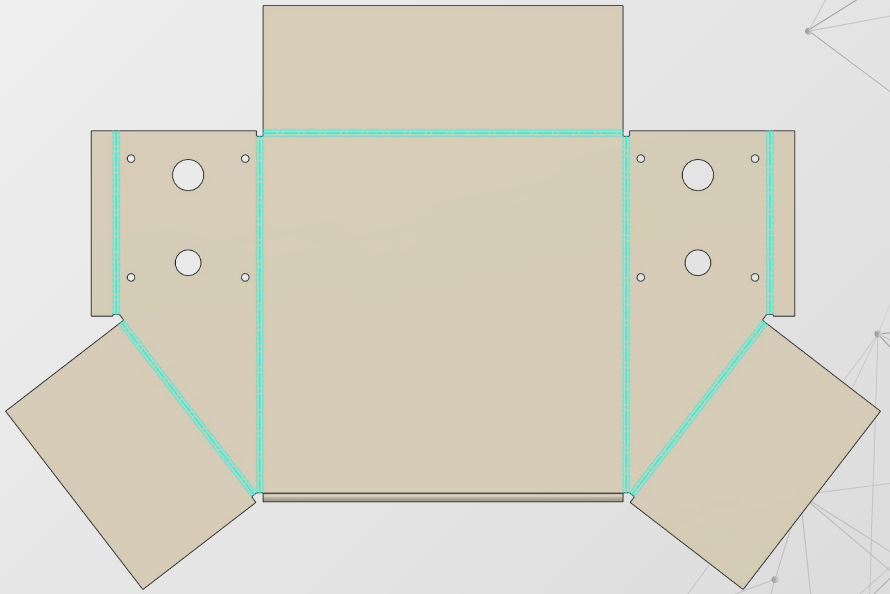
Fusion 360

Onshape

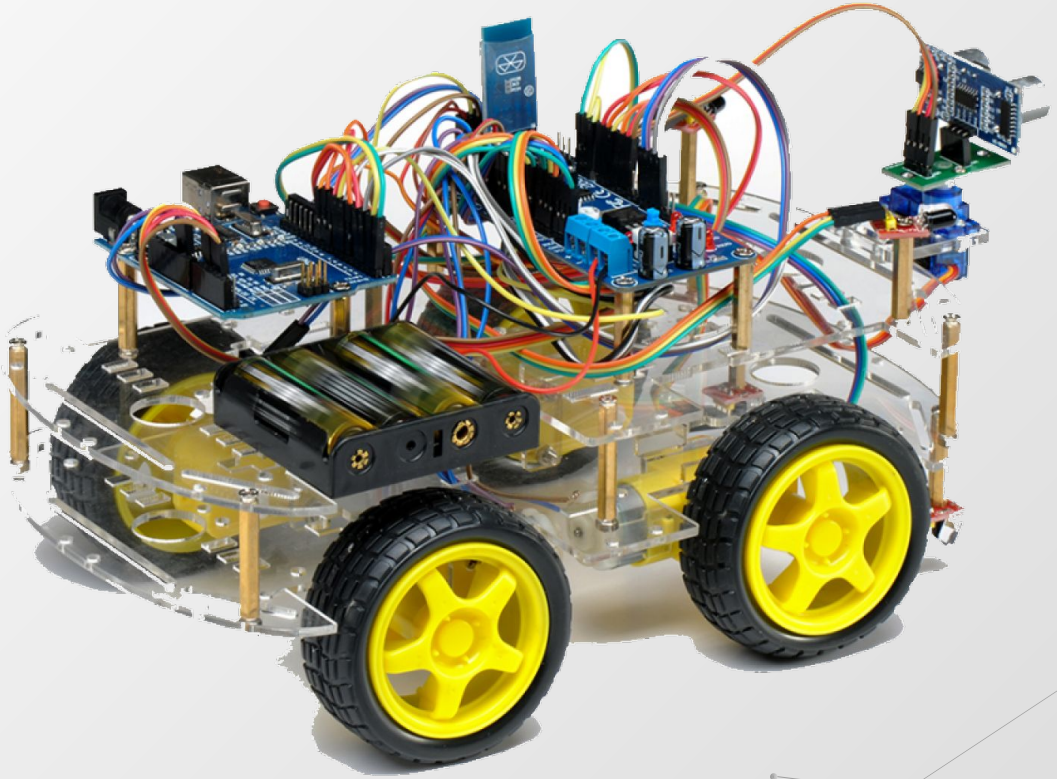
Sketchup

3D Design Using Fusion 360





Let's Build A RC Car



How To Accomplish A Projects

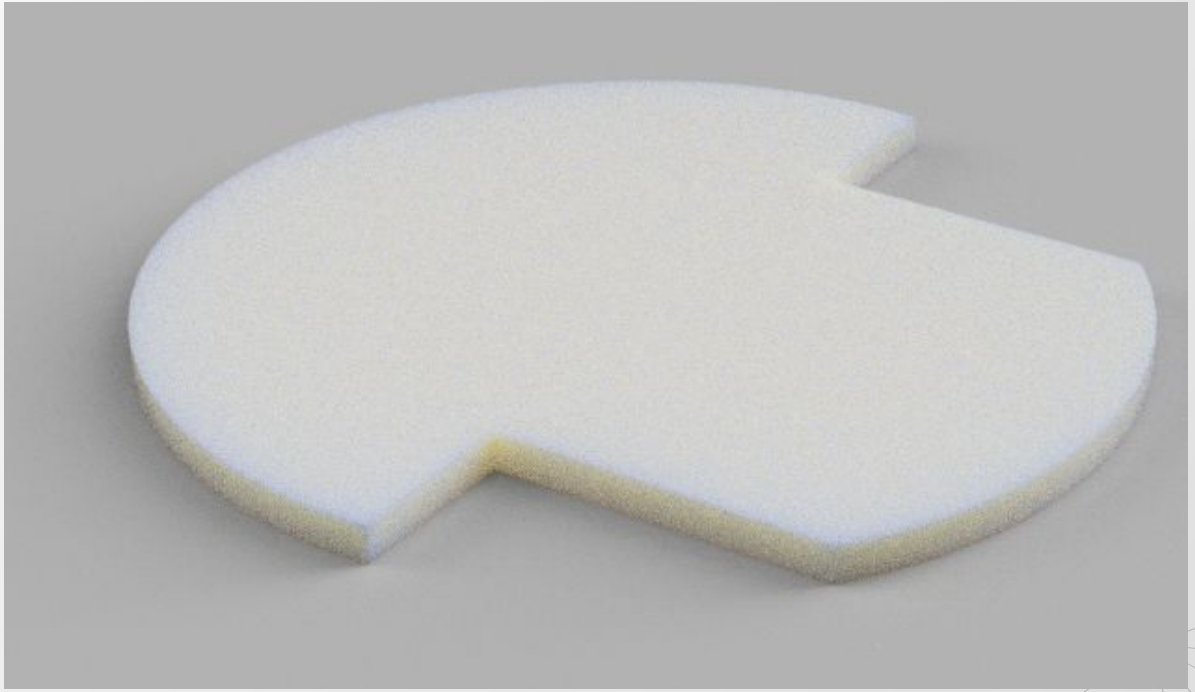
1. **Brainstorming** : Understand the problem and generate solution ideas.
2. **Research** : Explore how your idea will work and gather necessary information.
3. **Experimentation** : Test different approaches to find the most effective solution.
4. **Prototyping** : Build an initial model and refine it through simulations or small-scale testing.
5. **Final Build** : Complete the project with a polished design and optimized functionality.

Required Components

1. Chassis for the car frame - 1
2. Arduino Uno - 1
3. DC Motors - 2
4. Mounts - 2
5. Wheels - 2
6. Ball Castor - 1
7. L298N Motor Driver - 1
8. HC-05 Bluetooth Module - 1
9. Jumper Wires - Some
10. Li-po Battery - 1
11. Smartphone with a Bluetooth controller app - 1

Design And Build

Use any tool to design body and later on build it using any suitable way possible.

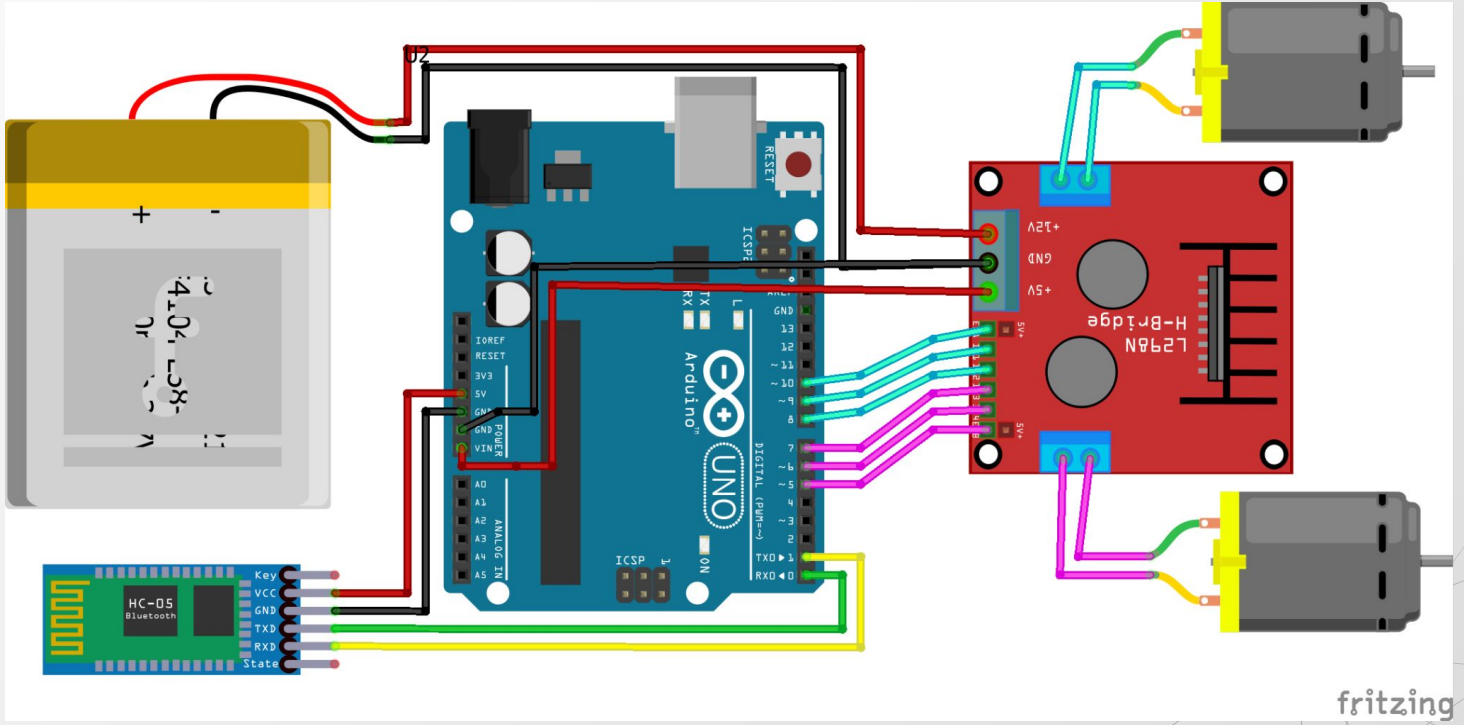


Assembling The Bot

Attach All The Components With Chesis

Design The Circuit

Connect all the pin according to the following diagram using jumper wire



Code

```
#define ENA 5
#define IN1 6
#define IN2 7
#define ENB 10
#define IN3 8
#define IN4 9
char command;
```

```
void setup() {
  pinMode(ENA, OUTPUT);
  pinMode(ENB, OUTPUT);
  pinMode(IN1, OUTPUT);
  pinMode(IN2, OUTPUT);
  pinMode(IN3, OUTPUT);
  pinMode(IN4, OUTPUT);
  Serial.begin(9600);
}
```

```
void loop() {
  if (Serial.available() > 0) {
    command = Serial.read();
    Stop();
    switch (command) {
      case 'F':
        forward();
        break;
      case 'B':
        back();
        break;
      case 'L':
        left();
        break;
      case 'R':
        right();
        break;
      default:
        Serial.println("Invalid");
    }
  }
}
```

```
void forward() {
  analogWrite(ENA, 200);
  analogWrite(ENB, 200);
  digitalWrite(IN1, HIGH);
  digitalWrite(IN2, LOW);
  digitalWrite(IN3, HIGH);
  digitalWrite(IN4, LOW);
}
```

```
void back() {
  analogWrite(ENA, 200);
  analogWrite(ENB, 200);
  digitalWrite(IN1, LOW);
  digitalWrite(IN2, HIGH);
  digitalWrite(IN3, LOW);
  digitalWrite(IN4, HIGH);
}
```

```
void left() {
  analogWrite(ENA, 150);
  analogWrite(ENB, 200);
  digitalWrite(IN1, LOW);
  digitalWrite(IN2, HIGH);
  digitalWrite(IN3, HIGH);
  digitalWrite(IN4, LOW);
}
```

```
void right() {
  analogWrite(ENA, 200);
  analogWrite(ENB, 150);
  digitalWrite(IN1, HIGH);
  digitalWrite(IN2, LOW);
  digitalWrite(IN3, LOW);
  digitalWrite(IN4, HIGH);
}
```

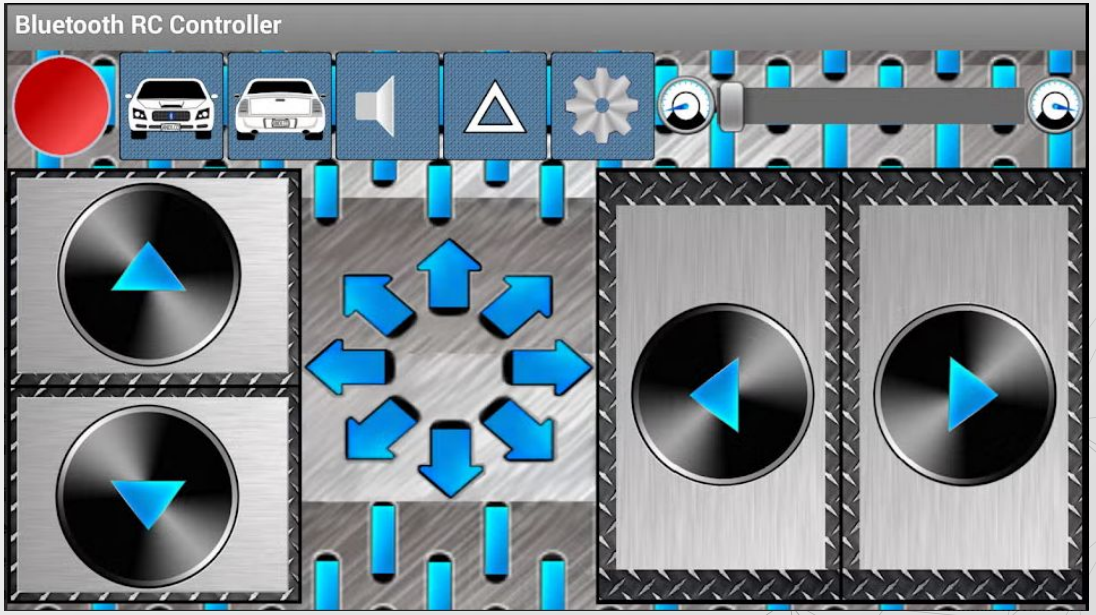
```
void Stop() {
  analogWrite(ENA, 0);
  analogWrite(ENB, 0);
  digitalWrite(IN1, LOW);
  digitalWrite(IN2, LOW);
  digitalWrite(IN3, LOW);
  digitalWrite(IN4, LOW);
}
```

Smartphone Application

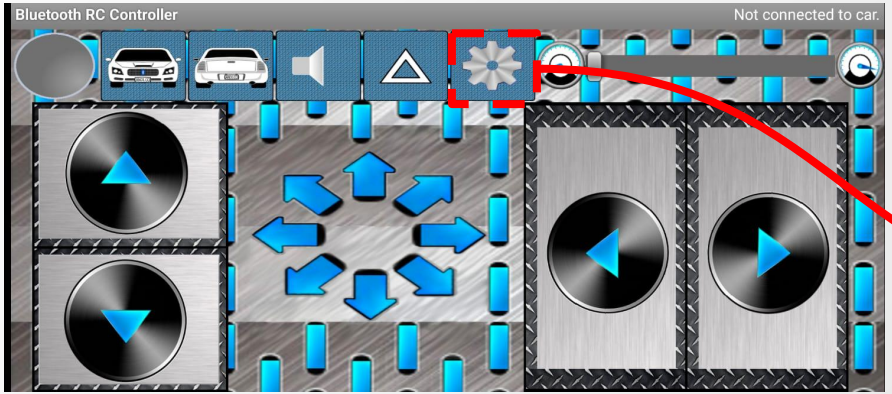
Install the app and connect it with the bluetooth module.

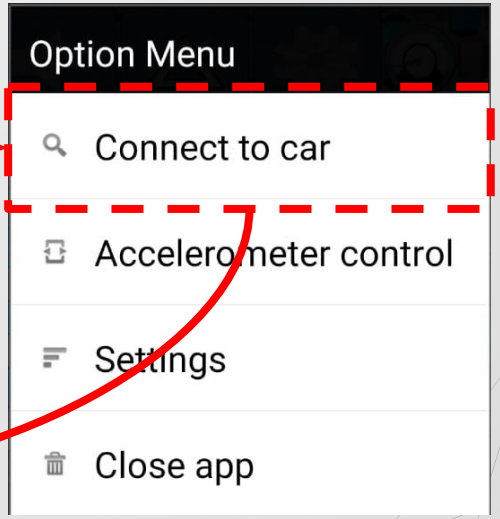
Bluetooth RC Controller

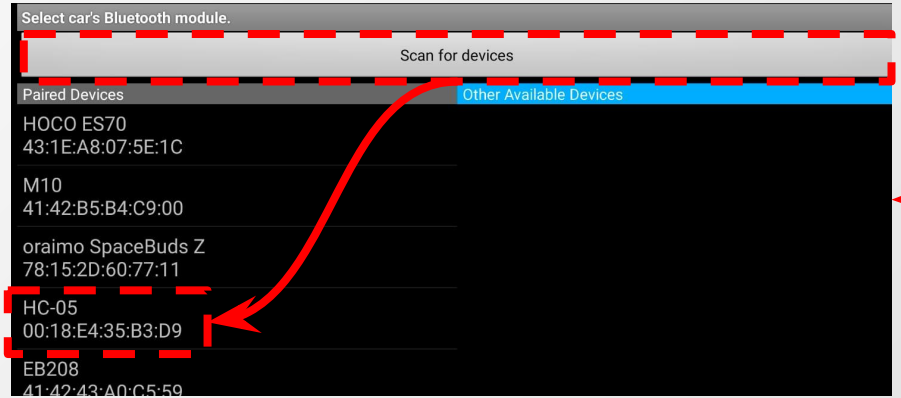
Link:
https://bluetooth-rc-car.en.softonic.com/android



Connect APP With Bluetooth Module







DEBUG & DRIVE

Resource

https://github.com/SaifAhammod/Build-A-RC-Car

QUESTION?