
جامعة المستقبل
AL MUSTAQBAL UNIVERSITY

كلية العلوم

قسم الانظمة الطبية الذكية

Intelligent Medical Systems Department

# Lap: (6)

# Motors

**Subject: Embedded System**

**Class: Third**

**Lecturer: Prof.Dr. Mehdi Ebady Manaa**

**Programmer:- Fatima Hussein Jawad**

**Study Year: 2024-2025**

# Motors :

An electric motor is an electrical machine that converts electrical energy into mechanical energy. Most electric motors operate through the interaction between the motor's magnetic field and winding currents to generate force in the form of rotation.

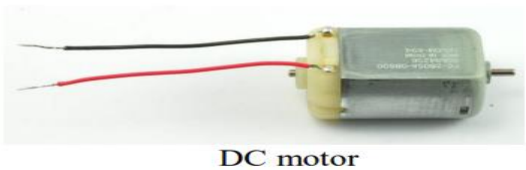
Electric motors can be powered by Direct Current (DC) sources, such as from batteries, motor vehicles or rectifiers, or by Alternating Current (AC) sources, such as a power grid, inverters or electrical generators.

The Arduino board can be used to control motors. Many electronics that have moving parts contain motors, e.g. car toys.

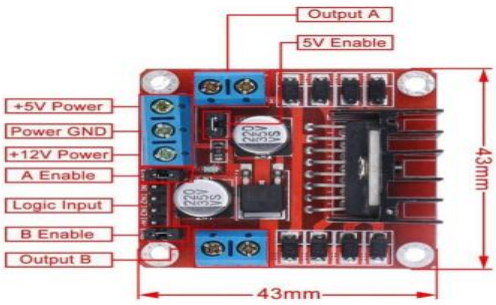
## Types of motors using with arduino

1- **DC motor:** is the most common type of motor. DC motors normally have just two leads, one positive and one negative. If you connect these two leads directly to a battery, the motor will rotate. If you switch the leads, the motor will rotate in the opposite direction.

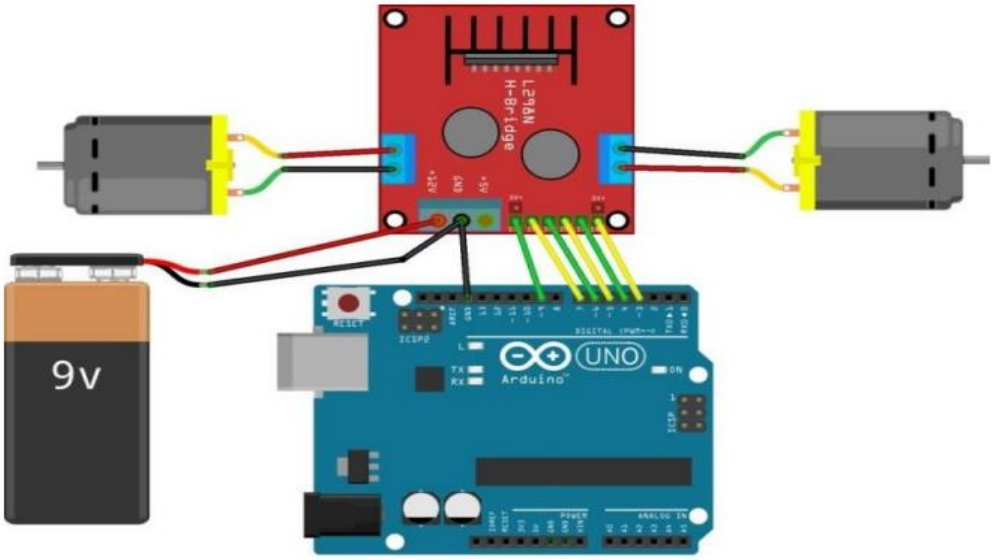
**Warning** – Do not drive the motor directly from Arduino board pins. This may damage the board. Use a driver Circuit (such as L298) or an IC.



DC motor




L298




## Code

```
int EnA = 10;
int EnB = 5;
int In1 = 9;
int In2 = 8;
int In3 = 7;
int In4 = 6;

void setup() {
  // Set all control and enable pins as outputs
  pinMode(EnA, OUTPUT);
  pinMode(EnB, OUTPUT);
  pinMode(In1, OUTPUT);
  pinMode(In2, OUTPUT);
  pinMode(In3, OUTPUT);
  pinMode(In4, OUTPUT);
}

void loop() {
  // Clockwise motion
  digitalWrite(In1, HIGH);
  digitalWrite(In2, LOW);
  analogWrite(EnA, 200);  // Control speed for motor A
```

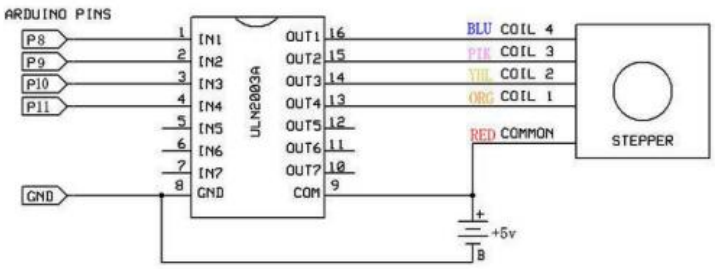
```
  digitalWrite(In3, HIGH);
  digitalWrite(In4, LOW);
  analogWrite(EnB, 200);  // Control speed for motor B
  delay(2000);            // Run motors clockwise for 2 seconds

  // Counter-clockwise motion
  digitalWrite(In1, LOW);
  digitalWrite(In2, HIGH);
  analogWrite(EnA, 200);  // Control speed for motor A
  digitalWrite(In3, LOW);
  digitalWrite(In4, HIGH);
  analogWrite(EnB, 200);  // Control speed for motor B
  delay(2000);            // Run motors counter-clockwise for 2 seconds

  // Stop the motors
  digitalWrite(In1, LOW);
  digitalWrite(In2, LOW);
  digitalWrite(In3, LOW);
  digitalWrite(In4, LOW);
  delay(1000);  // Optional delay to hold motors in stopped state
}
```

## 2- Stepper motors:

Stepper motors can be found in electronics where high precision is important such as scanners and printers. A stepper motor, in contrast to the DC motor, can be very precise with both position and speed.

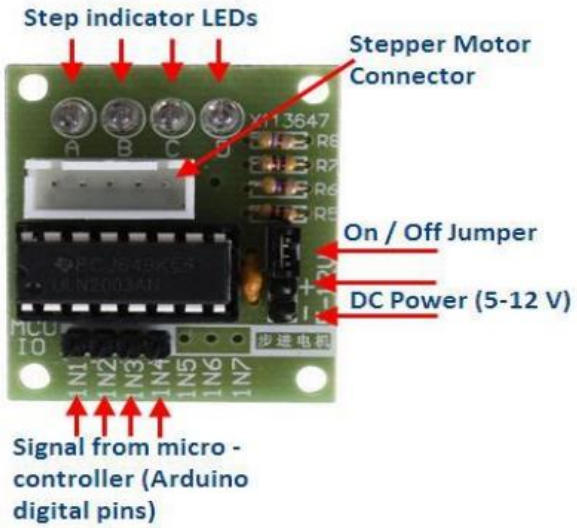
A stepper motors full rotation is divided into equally big steps and you can control the motor to stop at each of these steps. The steps are measured in degrees, usually 1.8, 3.6 or 7.2. The smaller the steps mean the more precise. This makes it very useful when repeated positioning is needed.

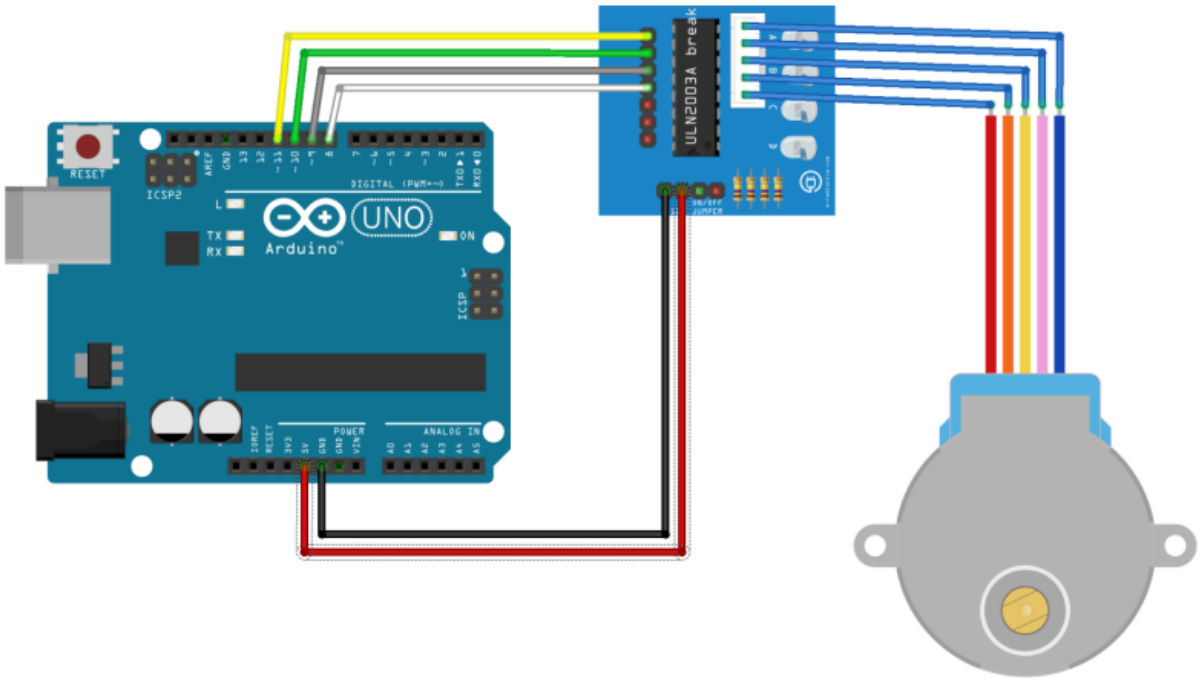
However, stepper motors will never be very fast compared to a DC motor. It generally has 4 or more wires and usually needs more than 5 volts to work. This means it cannot be powered by an Arduino, but we can use an external power supply.


Stepper motor (28BYJ-48)

To control work of the stepper motor needed connect it to a driver. There are many Types of Drivers, L293, ULN2003, A3967SLB, and more .The 28-BYJ48 even comes with Breakout using ULN2003 as a Motor driver chip.






## Code

```
#define IN1 8
#define IN2 9
#define IN3 10
#define IN4 11

int Steps = 0;
boolean Direction = true;
unsigned long last_time;
unsigned long currentMillis;
int steps_left = 4095;
long time;

void setup() {
  Serial.begin(115200);
  pinMode(IN1, OUTPUT);
  pinMode(IN2, OUTPUT);
  pinMode(IN3, OUTPUT);
  pinMode(IN4, OUTPUT);
}
```

```
void loop() {
  time = 0;  // Reset time counter for each cycle
  while (steps_left > 0) {
    currentMillis = micros();
    if (currentMillis - last_time >= 1000) {
      stepper(1);
      time += currentMillis - last_time;
      last_time = currentMillis;
      steps_left--;
    }
  }
  Serial.println(time);
  Serial.println("Wait...!");
  delay(2000);
  Direction = !Direction;
  steps_left = 4095;
}

void stepper(int xw) {
  for (int x = 0; x < xw; x++) {
    switch (Steps) {
      case 0:
        digitalWrite(IN1, LOW);
        digitalWrite(IN2, LOW);
        digitalWrite(IN3, LOW);
        digitalWrite(IN4, HIGH);
        break;
      case 1:
        digitalWrite(IN1, LOW);
        digitalWrite(IN2, LOW);
        digitalWrite(IN3, HIGH);
        digitalWrite(IN4, HIGH);
        break;
      case 2:
        digitalWrite(IN1, LOW);
        digitalWrite(IN2, LOW);
        digitalWrite(IN3, HIGH);
        digitalWrite(IN4, LOW);
        break;
      case 3:
```

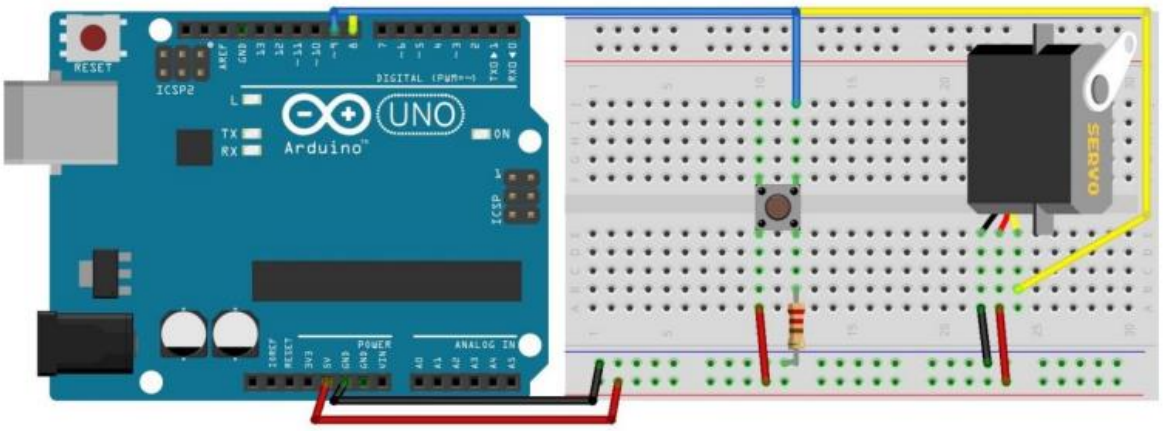
```
      digitalWrite(IN1, LOW);
      digitalWrite(IN2, HIGH);
      digitalWrite(IN3, HIGH);
      digitalWrite(IN4, LOW);
      break;
    case 4:
      digitalWrite(IN1, LOW);
      digitalWrite(IN2, HIGH);
      digitalWrite(IN3, LOW);
      digitalWrite(IN4, LOW);
      break;
    case 5:
      digitalWrite(IN1, HIGH);
      digitalWrite(IN2, HIGH);
      digitalWrite(IN3, LOW);
      digitalWrite(IN4, LOW);
      break;
    case 6:
      digitalWrite(IN1, HIGH);
      digitalWrite(IN2, LOW);
      digitalWrite(IN3, LOW);
      digitalWrite(IN4, LOW);
      break;
    case 7:
      digitalWrite(IN1, HIGH);
      digitalWrite(IN2, LOW);
      digitalWrite(IN3, LOW);
      digitalWrite(IN4, HIGH);
      break;
    default:
      digitalWrite(IN1, LOW);
      digitalWrite(IN2, LOW);
      digitalWrite(IN3, LOW);
      digitalWrite(IN4, LOW);
      break;
  }
  SetDirection();
 }
}

void SetDirection() {
```

```
  if (Direction) { Steps++; }
  else { Steps--; }
  if (Steps > 7) { Steps = 0; }
  if (Steps < 0) { Steps = 7; }
}
```

## 3- Servo motors:

Servo motors are often used in robotics and toys. These types of motors are simple to connect and to control from an Arduino. They have three wires: one for power, one for ground, and one for the control signal.

**There are two types**: **a standard rotation servo** and a **continuous rotation servo**. The standard rotation servo can rotate 180 degrees with precision like the stepper motors. The continuous rotation servo is similar to **a DC motor** and can spin in both directions, but not as fast. You can control both the speed and the direction without the use of transistors or driver Circuit.




## Code

```
#include <Servo.h>

const int servoPin = 8;
const int buttonPin = 9;
int buttonState = 0;
int lastButtonState = 0; // To track the last state of the button
int directionState = 0;
Servo myservo;
int pos = 0;

void setup() {
  myservo.attach(servoPin);
  pinMode(buttonPin, INPUT_PULLUP); // Enable internal pull-up resistor
}

void loop(){
  buttonState = digitalRead(buttonPin);

  // Check if the button is pressed (transition from LOW to HIGH)
  if (buttonState == HIGH && lastButtonState == LOW) {
    // Toggle direction state
    directionState = !directionState;

    if (directionState == 1) {
```

```
      // Move servo from 0 to 180 degrees
      for (pos = 0; pos <= 180; pos += 1) {
        myservo.write(pos);
        delay(15);
      }
    } else {
      // Move servo from 180 to 0 degrees
      for (pos = 180; pos >= 0; pos -= 1) {
        myservo.write(pos);
        delay(15);
      }
    }
  }

  lastButtonState = buttonState;      // Update the last button state
}
```

## Home work

1- Compare between wired and wireless connection?

2- What are benefits of using IOT?