# Arduino With Servo Motor

**AIM:**To control the servo motor using Arduino

**Software Required:** Arduino IDE

**Components Required:**

1. System -1
2. Arduino Uno Board -1
3. Arduino dumping cable -1
4. Servo motor -1
5. Breadboard-1
6. Connecting Wires -Required

**Theory:-**A servomotor is a rotary actuator or linear actuator that allows for precise control of angular or linear position, velocity and acceleration. It consists of a suitable motor coupled to a sensor for position feedback.

**Types:-**

•DC Servo Motor

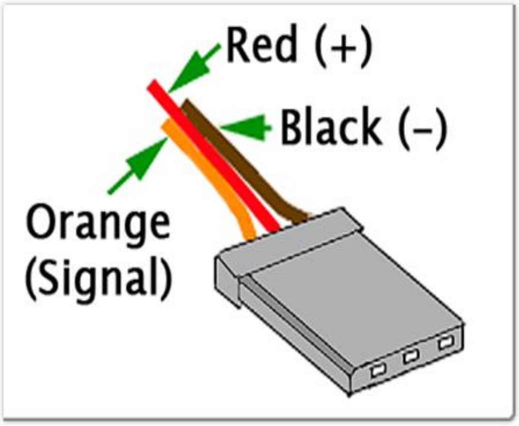
•AC Servo Motor

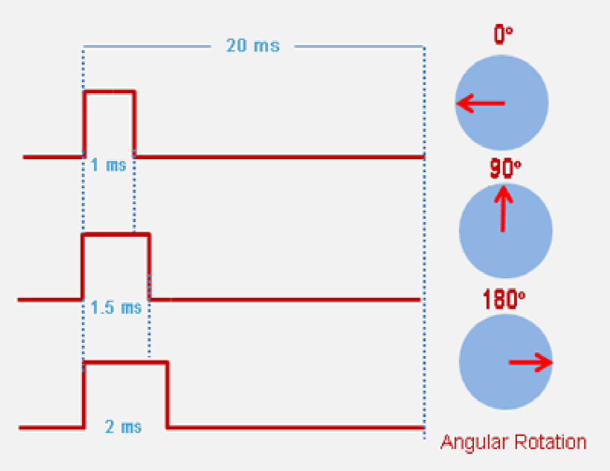
•Continuous Rotation Servo Motor

•Positional Rotation Servo Motor

•Linear Servo Motor

**Servo Motor Connections:-**





**Procedure:-**

1. Open Arduino IDE.
2. Write the code in the text editor.
3. Save the sketch with .ino extension.
4. Connect the hardware circuit and Connect your Arduino Board to the USB port of your computer.
5. Select the serial device of the Arduino board from the Tools | Serial Port menu.
6. Compile the file by clicking on the verify button.
7. If successful, the message "Done Compiling." will appear in the status bar.

8. If there are any errors it will show them in the Transcript window, rectify those errors and compile it again.

9. Push the reset button on the board then click the Upload button in the IDE. Wait a few seconds. If successful, the message "Done uploading." will appear in the status bar.

10. Click the serial monitor button in the toolbar and select the same baud rate used in the call to begin ().

**Code:-**

```
#include<Servo.h>
Servo myservo;
int pos = 0;
void setup()
{
  myservo.attach(3);
}
void loop()
{
myservo.write(0);
 delay(500);
myservo.write(180);
delay(500);
}
```

**Result:-**

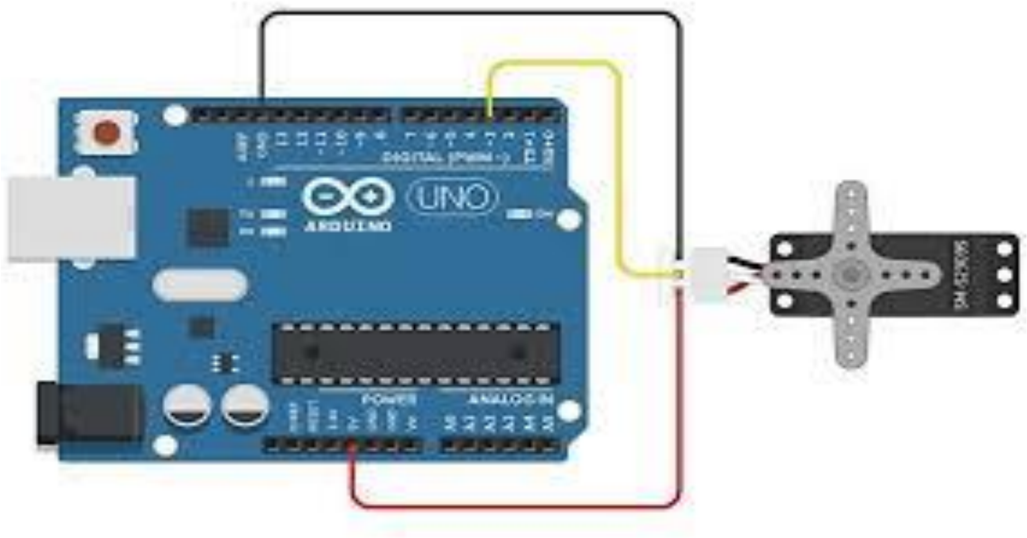


**Fig:- Servo Motor With Arduino**

# Servo motor controlling with potentiometer

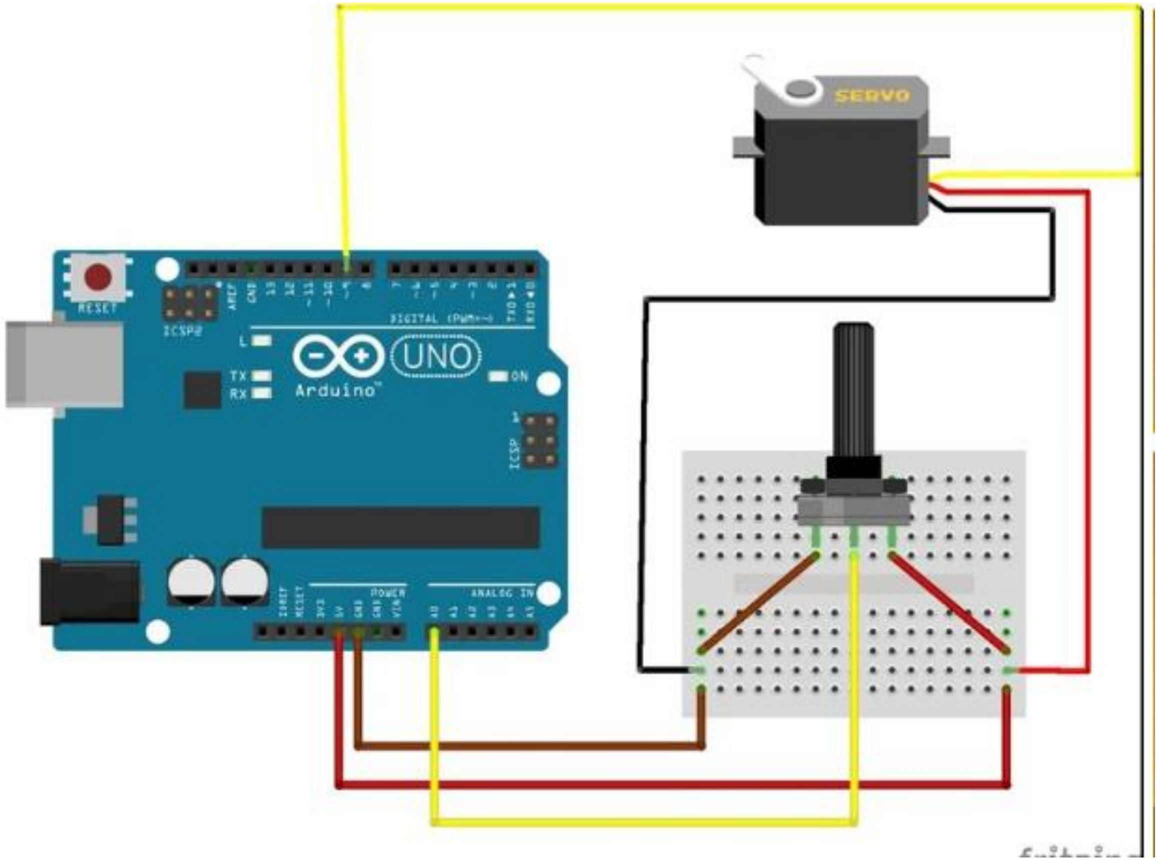
**AIM:**To control the servo motor with potentiometer using Arduino

**Software Required:**Arduino IDE

**Components Required:**

1. System -1
2. Arduino Uno Board -1
3. Arduino dumping cable -1
4.Servo motor -1
5. Resistor 330 ohms-1
6. Breadboard-1
7. Connecting Wires -Required
8.potentiometer

**Theory:-**Servo motors have three wires: power, ground, and signal. The power wire is typically red, and should be connected to the 5V pin on the Arduino or Genuino board. The ground wire is typically black or brown and should be connected to a ground pin on the board. The signal pin is typically yellow or orange and should be connected to pin 9 on the board.The potentiometer should be wired so that its two outer pins are connected to power (+5V) and ground, and its middle pin is connected to analog input 0 on the board.I recommend it! You do not connect the servo motor to the Arduino. I suggest you use external power to the servo. SG90 Mini Servo motors can be use



## Procedure:-

1. Open Arduino IDE.
2. Write the code in the text editor.
3. Save the sketch with .ino extension.
4. Connect the hardware circuit and Connect your Arduino Board to the USB port of your computer.
5. Select the serial device of the Arduino board from the Tools | Serial Port menu.
6. Compile the file by clicking on the verify button.
7. If successful, the message "Done Compiling." will appear in the status bar.
8. If there are any errors it will show them in the Transcript window, rectify those errors and compile it again.
9. Push the reset button on the board then click the Upload button in the IDE. Wait a few seconds. If successful, the message "Done uploading." will appear in the status bar.
10. Click the serial monitor button in the toolbar and select the same baud rate used in the call to begin ().

## Code:-

```
#include <Servo.h>  // add servo library
Servo myservo;  // create servo object to control a servo
int potpin = A0;  // analog pin used to connect the potentiometer
int val;    // variable to read the value from the analog pin
void setup() {
myservo.attach(9);  // attaches the servo on pin 9 to the servo object
}
void loop() {
val = analogRead(potpin);            // reads the value of the potentiometer (value between 0
and 1023)
val = map(val, 0, 1023, 0, 180);     // scale it to use it with the servo
                                  (value between 0 and 180)
 myservo.write(val);                  // sets the servo position according
                                  to the scaled value
delay(15);                           // waits for the servo to get there
}
```