# EXPERIMENT NO 06 INTERFACING ANALOG OUTPUT SERVO MOTOR WITH ARDUINO

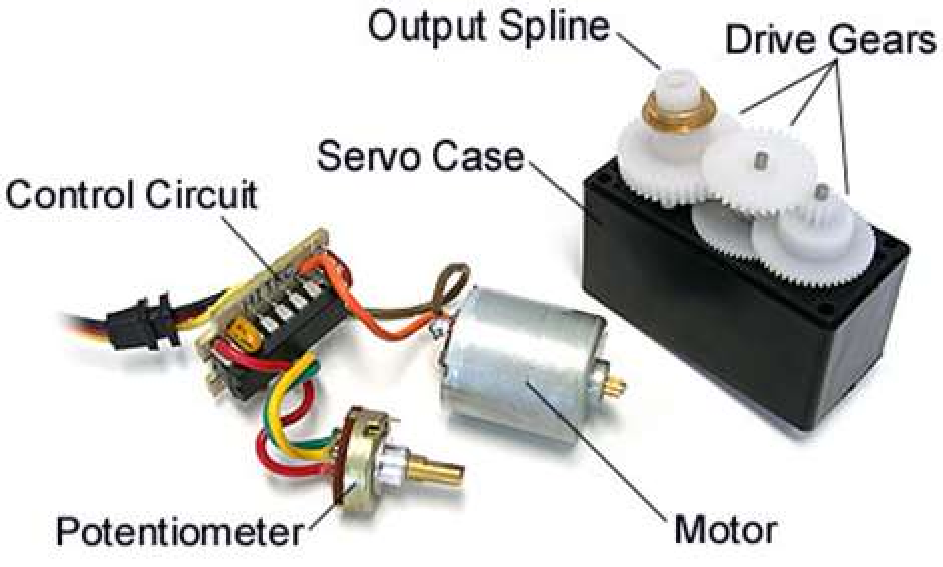
## AIM

To interface an Analog output (servo motor) and modify the angular displacement of the servo using PWM signal . COMPONENTS REQUIRED:

1. Servo motor of choice (9v is preferred )
2. 1 KΩ resistor
3. Arduino Uno
4. USB Interfacing cable
5. Connecting wires
6. Servo rated power supply (dc source )

## THEORY

Servo motors and are constructed out of basic DC motors, by adding: • gear reduction • a position sensor for the motor shaft • an electronic circuit that controls the motor's operation Typically, a potentiometer (variable resistor) measures the position of the output shaft at all times so the controller can accurately place and maintain its setting. Servo motors are used for angular positioning, such as in radio control airplanes. They typically have a movement range of 180 deg but can go up to 210 deg.The output shaft of a servo does not rotate freely, but rather is made to seek a particular angular position under electronic control.



### Figure-01 SERVO MOTOR SPLIT VIEW

Control An external controller (such as the Arduino) tells the servo where to go with a signal know as pulse proportional modulation (PPM) or pulse code modulation (which is often confused with pulse width modulation, PWM). PWM uses 1 to 2ms out of a 20ms time period to encode its information.

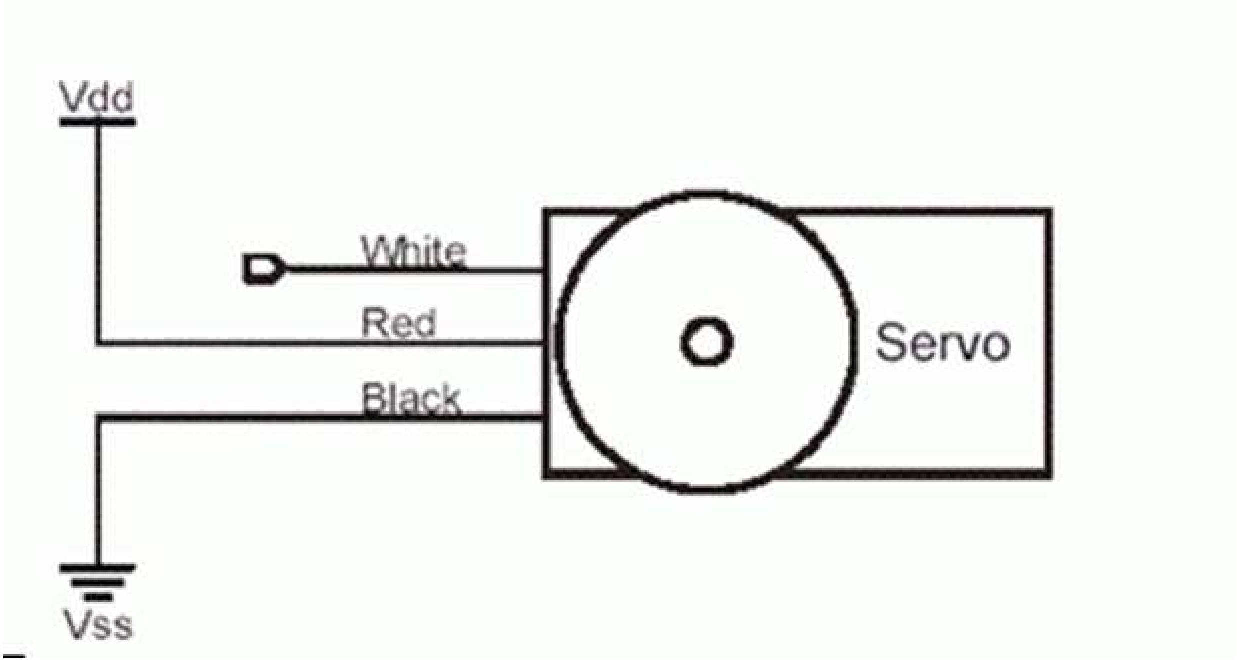


Figure-02 SERVO MOTOR PINS

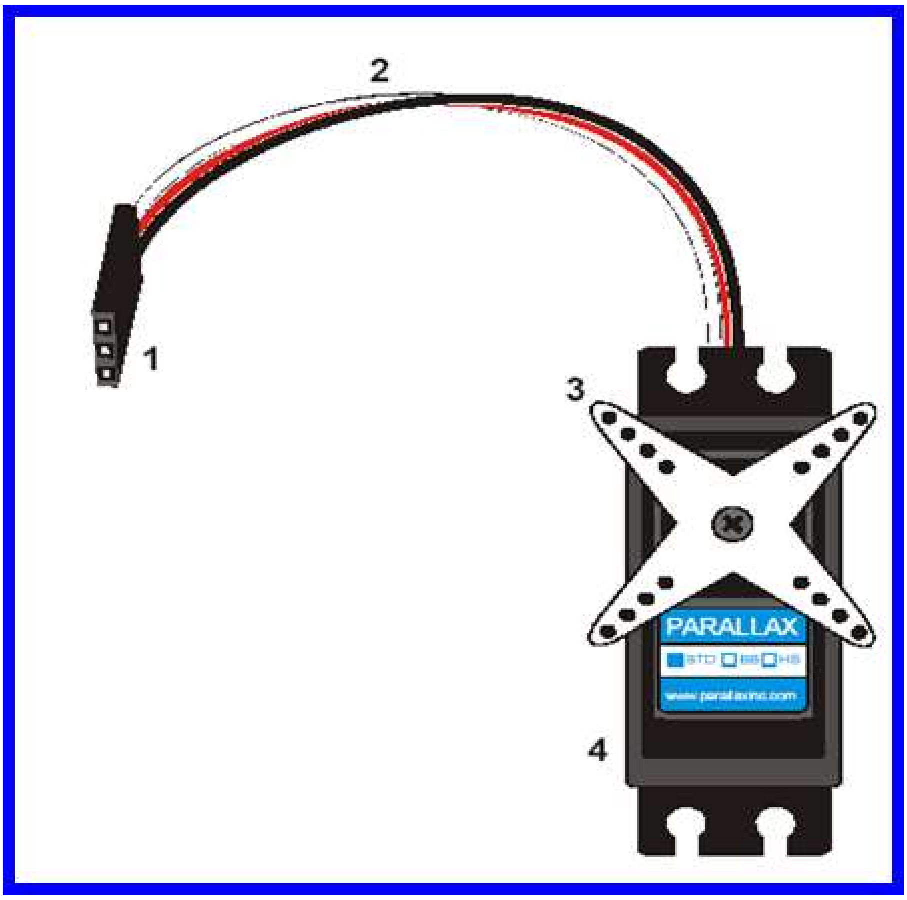
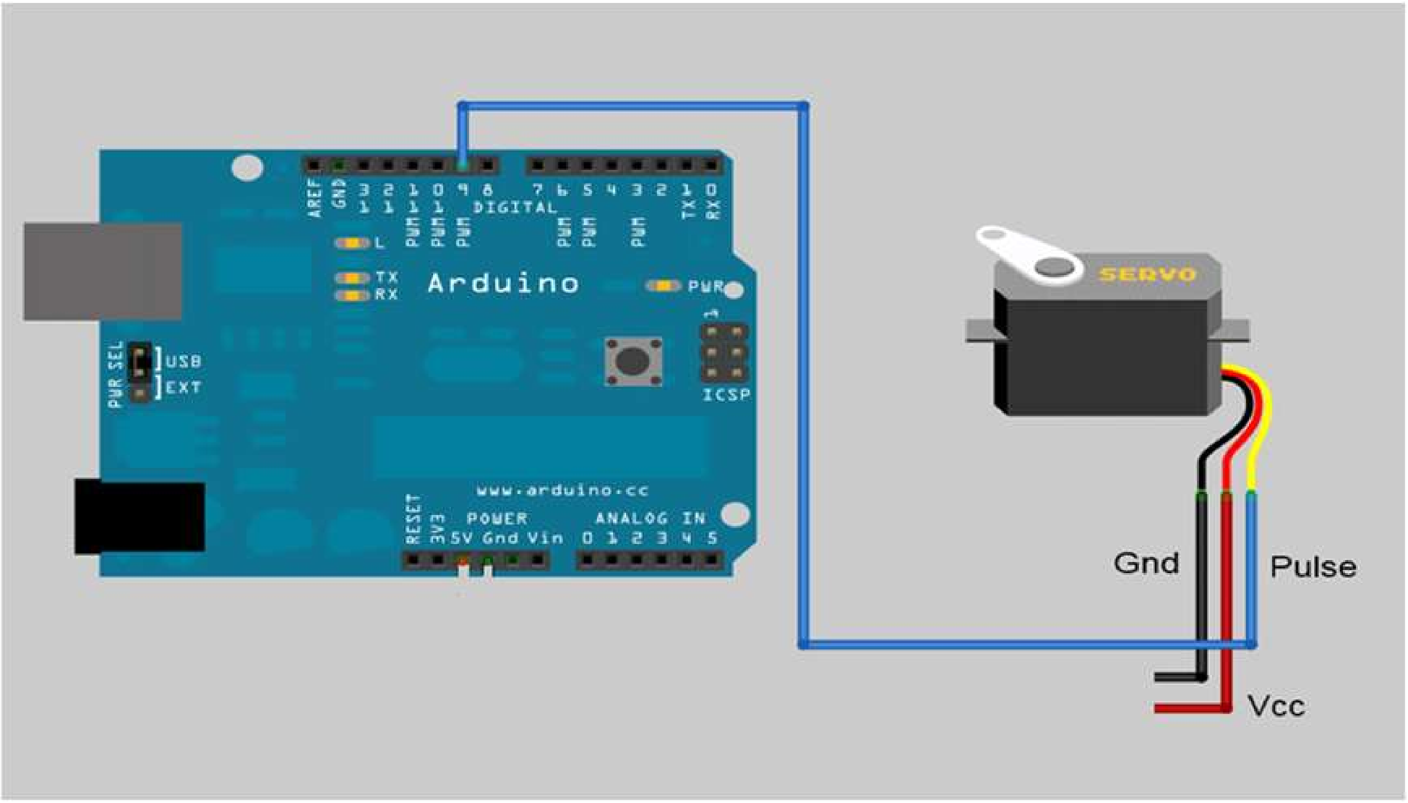


Figure-03 SERVO MOTOR OVERVIEW

CIRCUIT DIAGRAM



## FIGURE 04 CIRCUIT DIAGRAM

## PROCEDURE:

1. Connect the circuit as per the circuit diagram
2. Connect the board to your computer via the USB cable.
3. If needed, install the drivers.
4. Launch the Arduino IDE.
5. Select your board (If you the board is arduino uno, select accordingly).
6. Select your serial port, accordingly ( E.g. COM5 )
7. Open the file of the program and verify the error , clear if any errors that are existing
8. Upload the program and check for the physical working.
9. Ensure safety before powering up the device.

## PROGRAM :

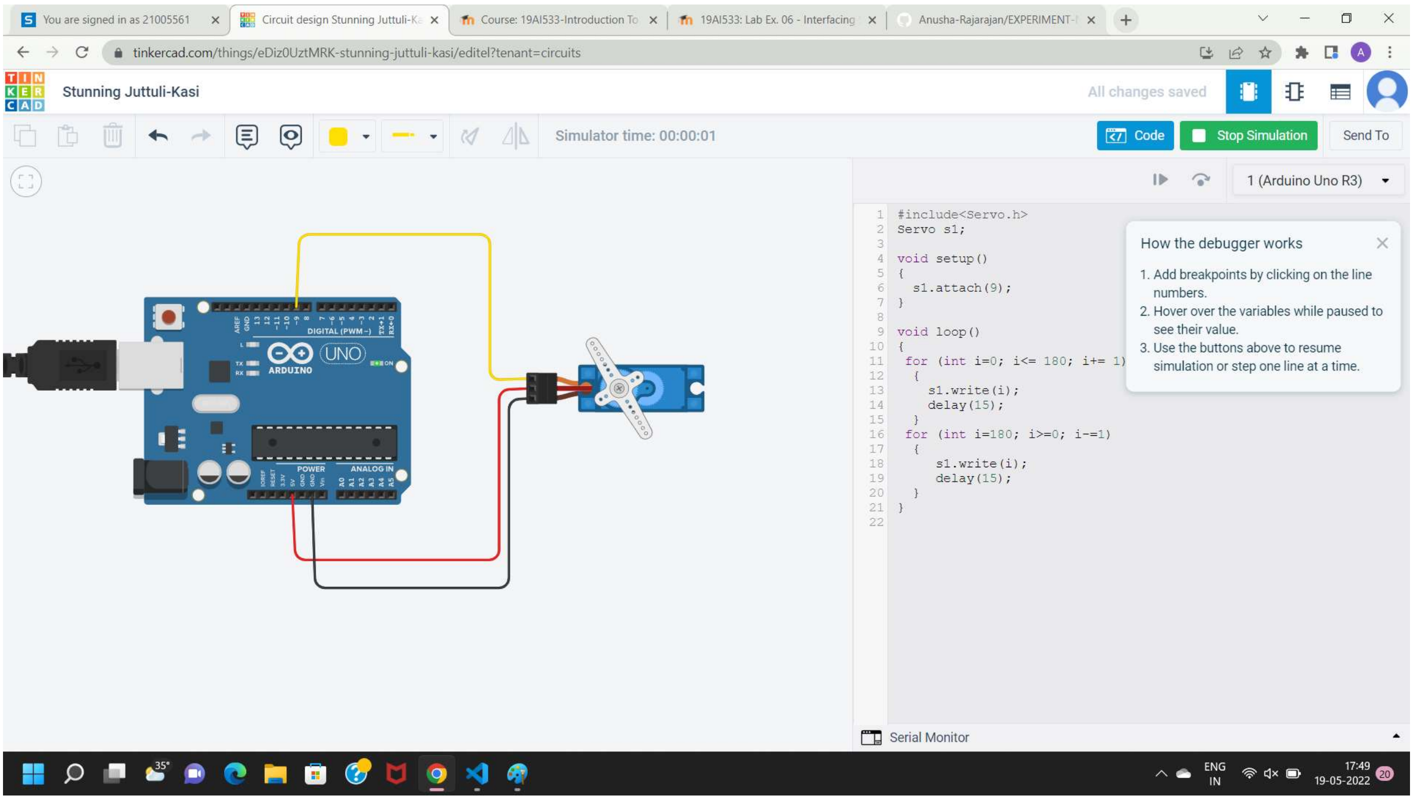
```
#include<Servo.h>
Servo s1;

void setup()
{
  s1.attach(9);
}

void loop()
{
 for (int i=0; i<= 180; i+= 1)
```

```
  {
    s1.write(i);
    delay(15);
  }
 for (int i=180; i>=0; i-=1)
  {
         s1.write(i);
         delay(15);
  }
}
```

## OUTPUT:


You are signed in as 21005561
Circuit design Stunning Juttuli-Ka
Course: 19AI533-Introduction To
19AI533: Lab Ex. 06 - Interfacing
Anusha-Rajarajan/EXPERIMENT-
tinkercad.com/things/eDiz0UztMRK-stunning-juttuli-kasi/editel?tenant=circuits
Stunning Juttuli-Kasi
All changes saved
Simulator time: 00:00:01
Code
Stop Simulation
Send To
1 (Arduino Uno R3)
UNO
ARDUINO
DIGITAL (PWM~)
POWER
ANALOG IN
Serial Monitor
ENG
IN
17:49
19-05-2022

```
#include<Servo.h>
Servo s1;

void setup()
{
  s1.attach(9);
}

void loop()
{
 for (int i=0; i<= 180; i+= 1)
  {
    s1.write(i);
    delay(15);
  }
 for (int i=180; i>=0; i-=1)
  {
     s1.write(i);
    delay(15);
  }
}
```

How the debugger works

1. Add breakpoints by clicking on the line numbers.
2. Hover over the variables while paused to see their value.
3. Use the buttons above to resume simulation or step one line at a time.

## RESULTS:

Arduino uno interfacing with servo motor is learned and angular position is controlled using PWM signal.