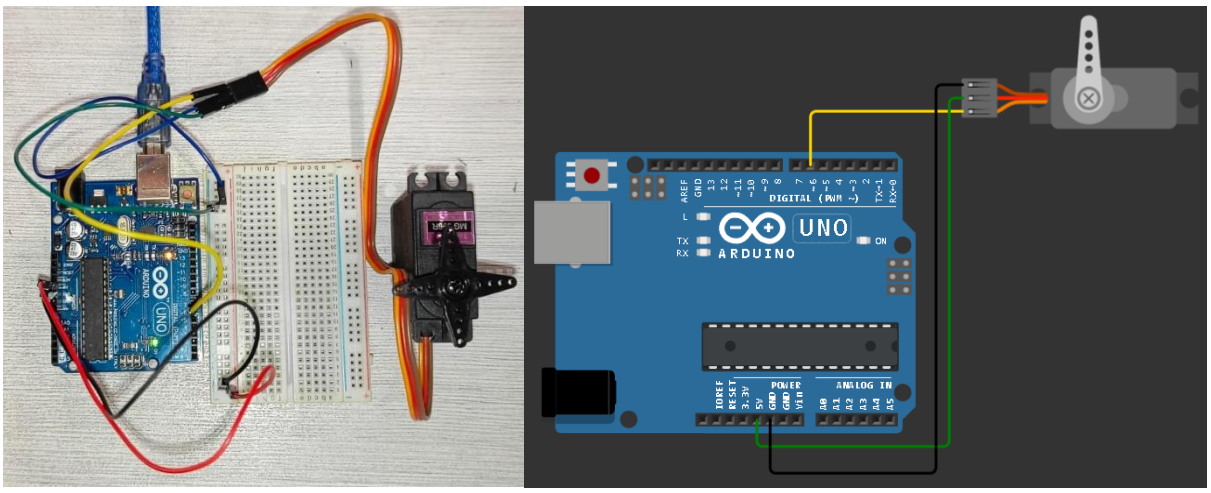
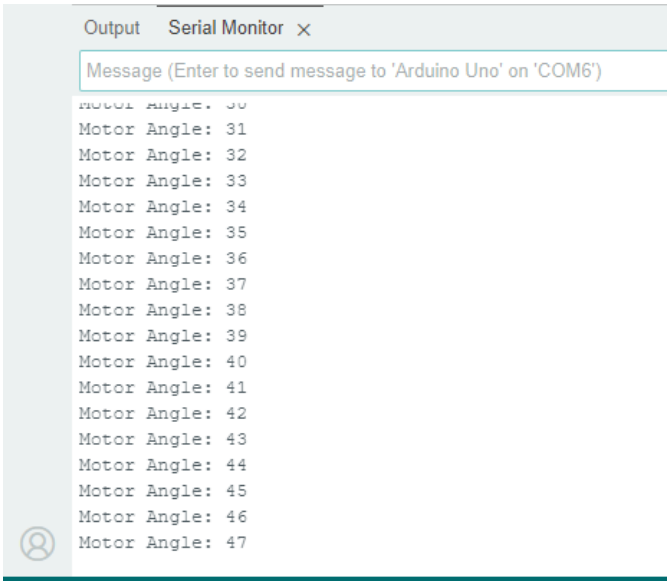
LAB TASKS

Q1: Develop the program which would generate a sweep motion (0°-180° and then 180°-0°) by using a servo motor. Print the values of angle and add it in your report.

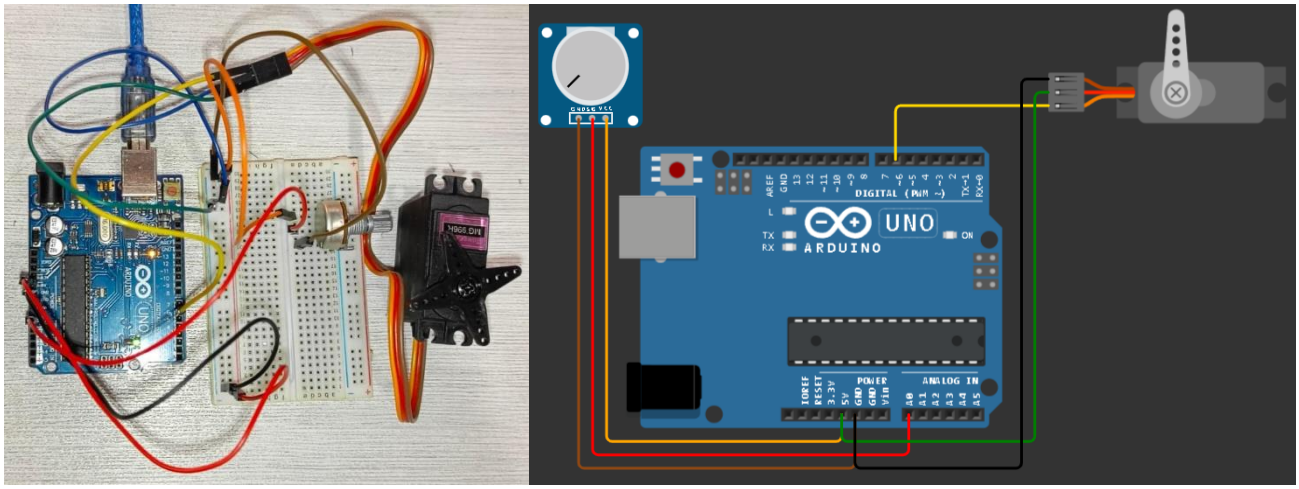
```
#include <Servo.h> // Defining the servo motor library

#define servoPin 6 // Defining a global variable to store the input pin position of the motor

Servo myservo; //Calling a servo motor function from the library

int Ser; // Initializing a variable

void setup () {
  Serial.begin(9600); // Calling serial monitor for data display
  myservo.attach(6); // Defining the library function with the input position of the servo
motor
}

void loop() {
  for (Ser = 0; Ser <= 180; Ser++){ // Defining a for loop for multiple forward repetition
    myservo.write(Ser); // Sending an output to the motor for it to rotate
    Serial.print("Motor Angle: "); // Printing data on the serial monitor
    Serial.println(Ser); // Printing the value stored in Ser variable

    delay(100); // Adding a time delay
  }

  for (Ser = 180; Ser >= 0; Ser--){ // Defining a for loop for multiple backward repetition
    myservo.write(Ser); // Sending an output to the motor for it to rotate
    Serial.print("Motor Angle: "); // Printing data on the serial monitor
    Serial.println(Ser); // Printing the value stored in Ser variable

    delay(100); // Adding a time delay
  }
}
```





Q2: Develop a program that would allow user to control a servo motor with the help of rotary potentiometer. Print both results on serial monitor and them in your report when the potentiometer is at its middle value.

```
#include <Servo.h> // Defining the servo motor library

#define servoPin 6 // Defining a global variable to store the input pin position of the motor
#define PMpin A0 // Defining a global variable to store the input pin position of the
potentiometer

Servo myservo; //Calling a servo motor function from the library

int val; // Initializing a variable
int mval; // Initializing a variable

void setup () {
  Serial.begin(9600); // Calling serial monitor for data display
  myservo.attach(6); // Defining the library function with the input position of the servo
motor
}

void loop() {
  val = analogRead(PMpin); // Reading the data from potentiometer and storing into val variable
  mval = map(val,0,1023,0,180); // Mapping the analog data from potentiometer to angle range (0
to 180)

  myservo.write(mval); // Sending an output to the motor for it to rotate

  Serial.print("Potentiometer Value: "); // Printing data on the serial monitor
  Serial.print(val); // Printing the value stored in val variable
  Serial.print (", Servo Motor Value: "); // Printing data on the serial monitor
  Serial.println(mval); // Printing the value stored in mval variable

  delay(100); // Adding a time delay
}
```




```
Output  Serial Monitor ×
Message (Enter to send message to 'Arduino Uno' on 'COM6')
Potentiomneter Value: 419, Servo Motor Value: 73
Potentiomneter Value: 424, Servo Motor Value: 74
Potentiomneter Value: 408, Servo Motor Value: 71
Potentiomneter Value: 427, Servo Motor Value: 75
Potentiomneter Value: 450, Servo Motor Value: 79
Potentiomneter Value: 470, Servo Motor Value: 82
Potentiomneter Value: 500, Servo Motor Value: 87
Potentiomneter Value: 531, Servo Motor Value: 93
Potentiomneter Value: 567, Servo Motor Value: 99
Potentiomneter Value: 605, Servo Motor Value: 106
Potentiomneter Value: 650, Servo Motor Value: 114
Potentiomneter Value: 690, Servo Motor Value: 121
Potentiomneter Value: 720, Servo Motor Value: 126
Potentiomneter Value: 755, Servo Motor Value: 132
Potentiomneter Value: 765, Servo Motor Value: 134
Potentiomneter Value: 793, Servo Motor Value: 139
Potentiomneter Value: 806, Servo Motor Value: 141
Potentiomneter Value: 846, Servo Motor Value: 148
Potentiomneter Value: 854, Servo Motor Value: 150
Potentiomneter Value: 927, Servo Motor Value: 163
```

POST LAB TASKS

Q1: Give any 2 examples of practical applications of a 180° servo motor.

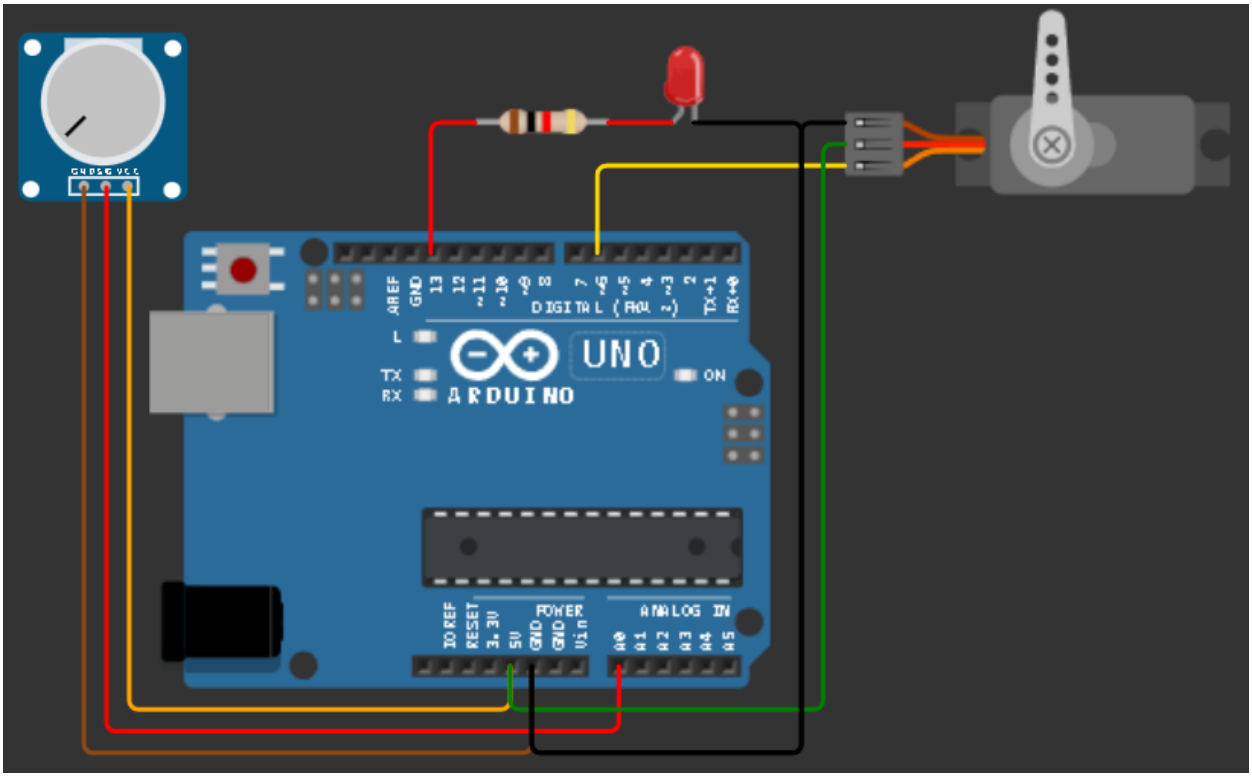
1. In a Speedometer: A 180° servo motor could be utilized in a speedometer system to control the needle's movement indicating the vehicle's speed. The servo motor, when triggered by the speed sensor, can rotate the needle on the speedometer dial within a range of 180°.

2. In a Locking Mechanism: A 180° servo motor can play a role in a locking mechanism, such as in electronic door locks or safes. By integrating the servo motor with the locking system, it can control the movement of the locking bolt or mechanism. For instance, upon receiving the correct input (like a password or signal), the servo motor is activated to rotate within its 180° range, either engaging or disengaging the lock, thus securing or releasing the door or safe.

Q2: By expanding the application of question # 2, write a code which includes blinking of an LED if the motor has reached its maximum rotation (in other words when the potentiometer output is 1023). Explain your code by comment on each line and draw the circuit diagram.

```
#include <Servo.h> // Defining the servo motor library

#define servoPin 6 // Defining a global variable to store the input pin position of the motor
```

```
#define PMpin A0 // Defining a global variable to store the input pin position of the potentiometer
#define LEDpin 13 // Defining a global variable to store the input pin position of the LED

Servo myservo; //Calling a servo motor function from the library

int val; // Initializing a variable
int mval; // Initializing a variable

void setup () {
  Serial.begin(9600); // Calling serial monitor for data display
  myservo.attach(6); // Defining the library function with the input position of the servo motor
  pinMode(LEDpin, OUTPUT); // Defining LED with LED pin and output
}

void loop() {
  val = analogRead(PMpin); // Reading the data from potentiometer and storing into val variable
  mval = map(val,0,1023,0,180); // Mapping the analog data from potentiometer to angle range (0 to 180)

  myservo.write(mval); // Sending an output to the motor for it to rotate

  Serial.print("Potentiometer Value: "); // Printing data on the serial monitor
  Serial.print(val); // Printing the value stored in val variable
  Serial.print (", Servo Motor Value: "); // Printing data on the serial monitor
  Serial.println(mval); // Printing the value stored in mval variable

  if (val == 1023){ // Using an if statement to check and run the code when the condition is true
    digitalWrite(LEDpin, 1); // Sending output to LED with on state
    delay(400); // Adding a time delay between on and off state of the LED so the blinking is visible
    digitalWrite(LEDpin, 0); // Sending output to LED with off state
    delay(300); // Adding a time delay between on and off state of the LED so the blinking is visible. 300 ms is defined to that it adds with the 100 ms below and equals 400 ms
  }

  delay(100); // Adding a time delay
}
```


UNO
ARDUINO
DIGITAL (PWM~)
POWER
ANALOG IN