UNIVERSITI TEKNOLOGI PETRONAS

# LAB ASSIGNMENT

SEP 2025 SEMESTER


**'AFFAN NAJIY BIN RUSDI**

22010453

BACHELOR OF COMPUTER SCIENCE

INTERNET OF THINGS


TFB2093

# Contents

# Code

## Task 1

```
//Task 1A
//Potentiometer on A0 → % on Serial Monitor

const int POT_PIN = A0;

void setup() {
  Serial.begin(9600);
}

void loop() {
  int raw = analogRead(POT_PIN); //0..1023
  float pct = (raw / 1023.0) * 100.0; //map to 0..100%
  Serial.print("raw=");
  Serial.print(raw);
  Serial.print("\tpercent=");
  Serial.println(pct);
  delay(50);
}
```

```
//Task 1B
//Potentiometer on A0 → % on Serial Plotter

const int POT_PIN = A0;

void setup() {
  Serial.begin(9600);
}

void loop() {
  int raw = analogRead(POT_PIN);
  Serial.println(sensorValue);
  delay(50);
}
```

## Task 2

```
//Task 2
//LDR Voltage Divider with Thresholds

const int LDR_PIN = A0;

void setup() {
  Serial.begin(9600);
}

void loop() {
  int raw = analogRead(LDR_PIN); //0..1023
  Serial.print("Light Level=");
  Serial.print(raw);

  if (raw > 700) Serial.println("Bright");
  else if (raw > 300) Serial.println("Normal");
  else Serial.println("Dark");

  delay(100); //Updates
}
```

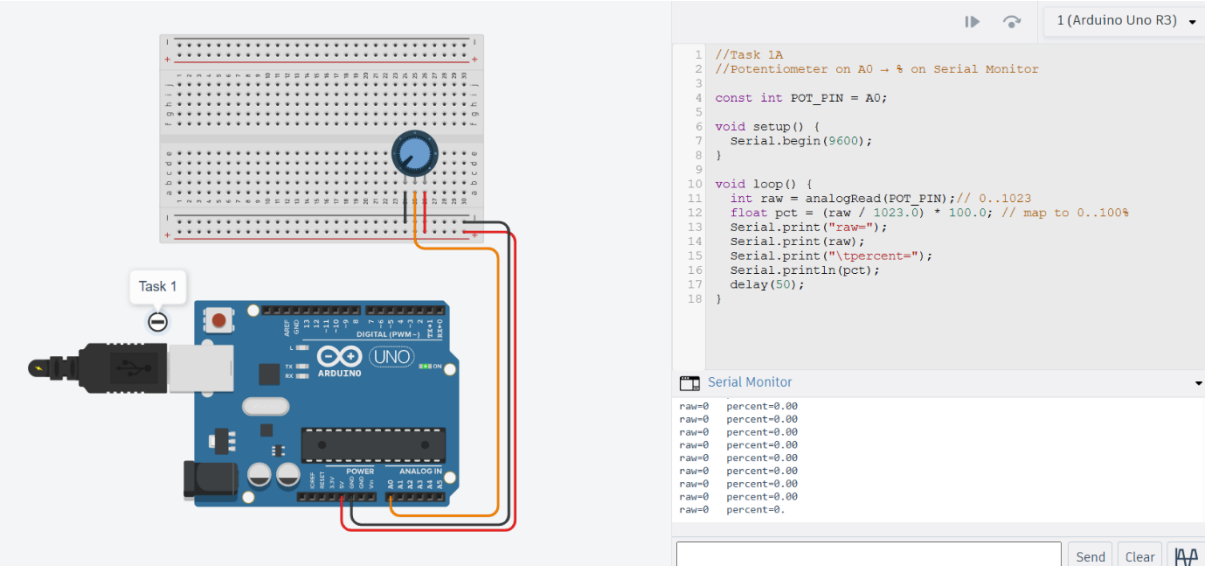
## Task 3

```
//Task 3
//Servo from Potentiometer with Rate-Limit/Deadband

#include <Servo.h>

const int POT_PIN = A0;
const int SERVO_PIN = 9;
Servo myServo;

int lastAngle = 0;
unsigned long lastMove = 0;
const int DEAD_BAND = 10; //Ignore changes < 10
const unsigned long MOVE_DELAY = 20; //Rate limit (ms)

void setup() {
  Serial.begin(9600);
  myServo.attach(SERVO_PIN);
}

void loop() {
  int raw = analogRead(POT_PIN); //0..1023
  int targetAngle = map(raw, 0, 1023, 0, 180); //Map to 0-180°

  // Deadband: Only move if change is significant
  if (abs(targetAngle - lastAngle) > DEAD_BAND) {
    if (millis() - lastMove >= MOVE_DELAY) {
      lastMove = millis();
      myServo.write(targetAngle);
      lastAngle = targetAngle;
      Serial.print("Angle=");
      Serial.println(targetAngle);
    }
  }

  delay(10); //Small delay
}
```

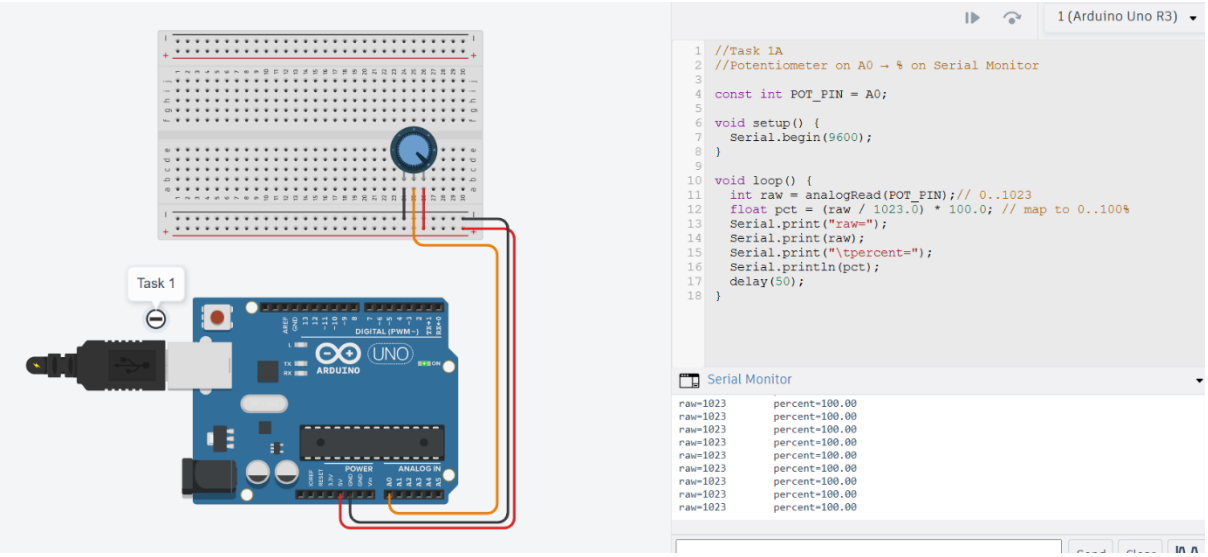
# Output

Tinkercad Link: [Affan Task 1]

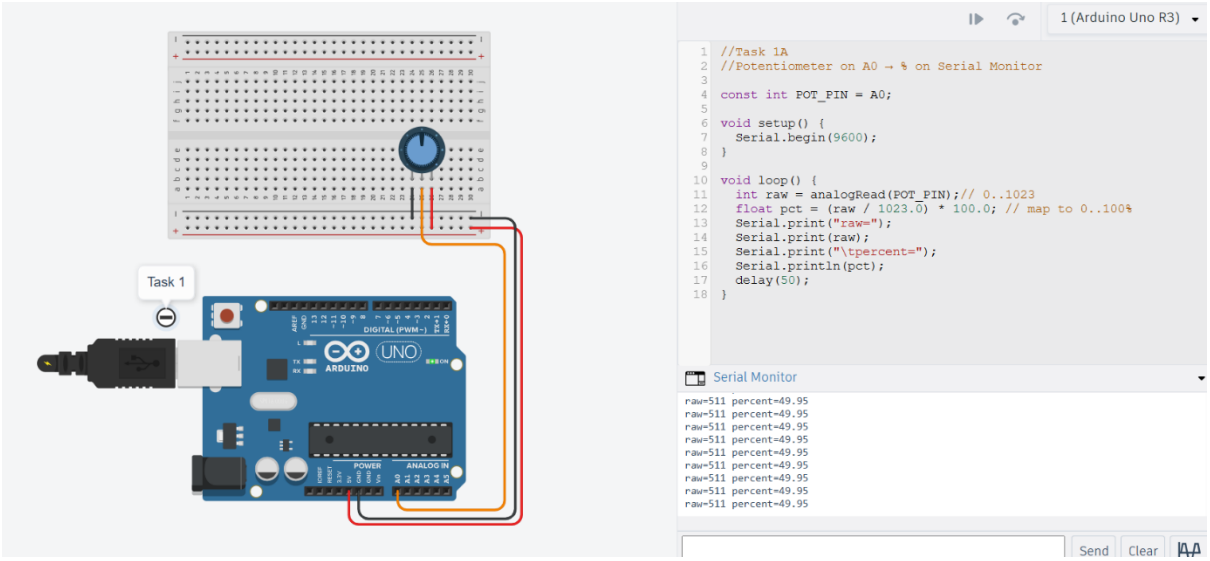
## Task 1



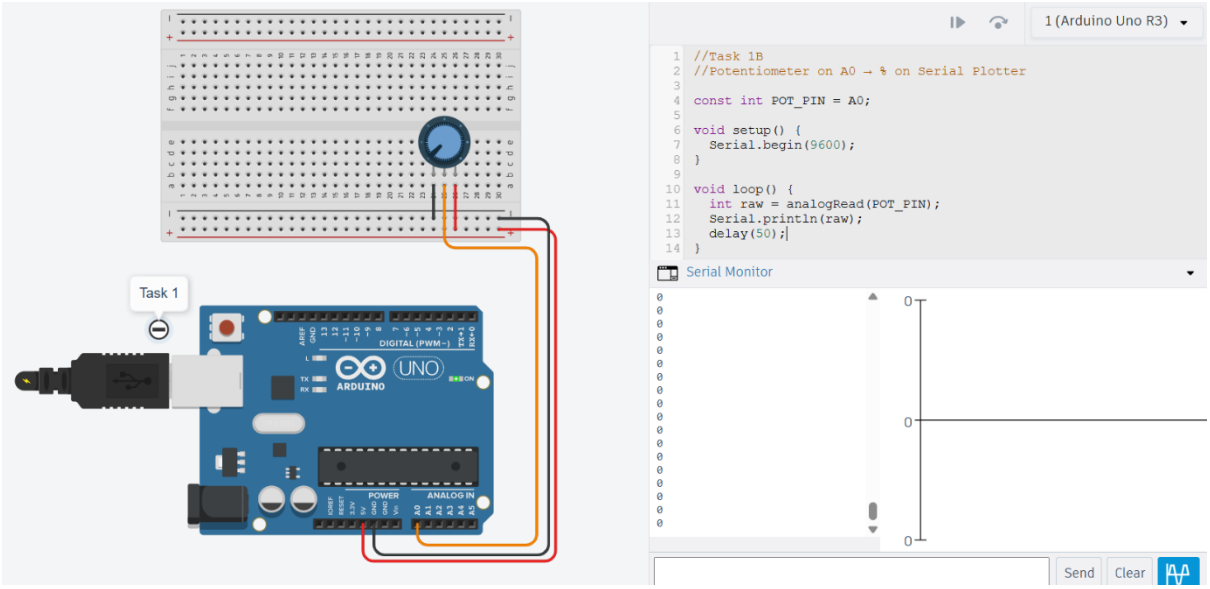

*Min on Potentiometer for Serial Monitor*



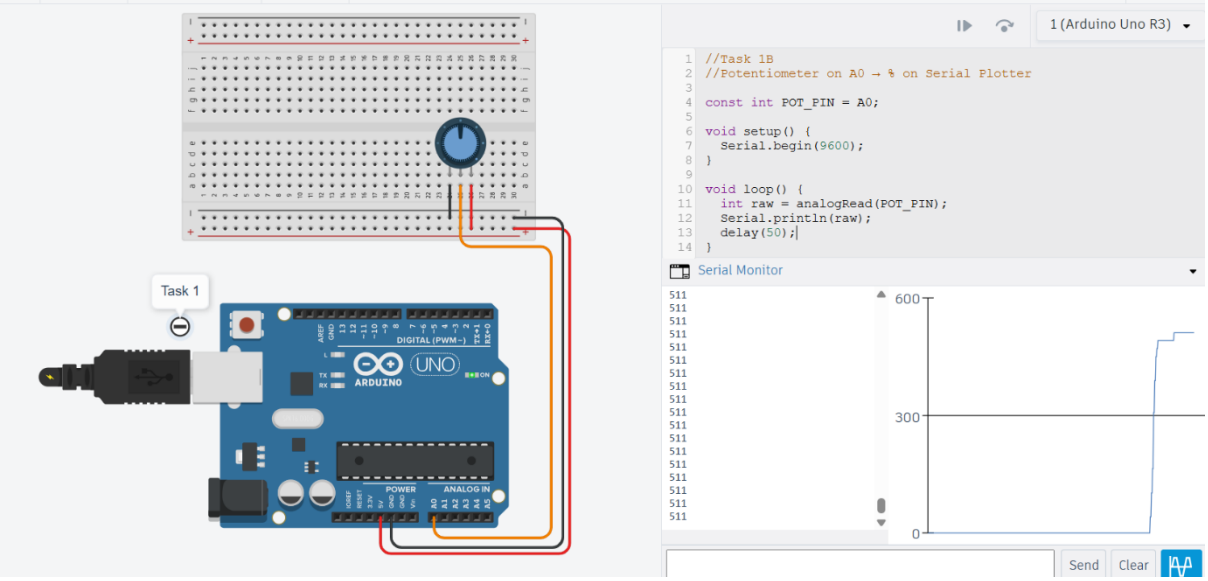

*Mid on Potentiometer for Serial Monitor*



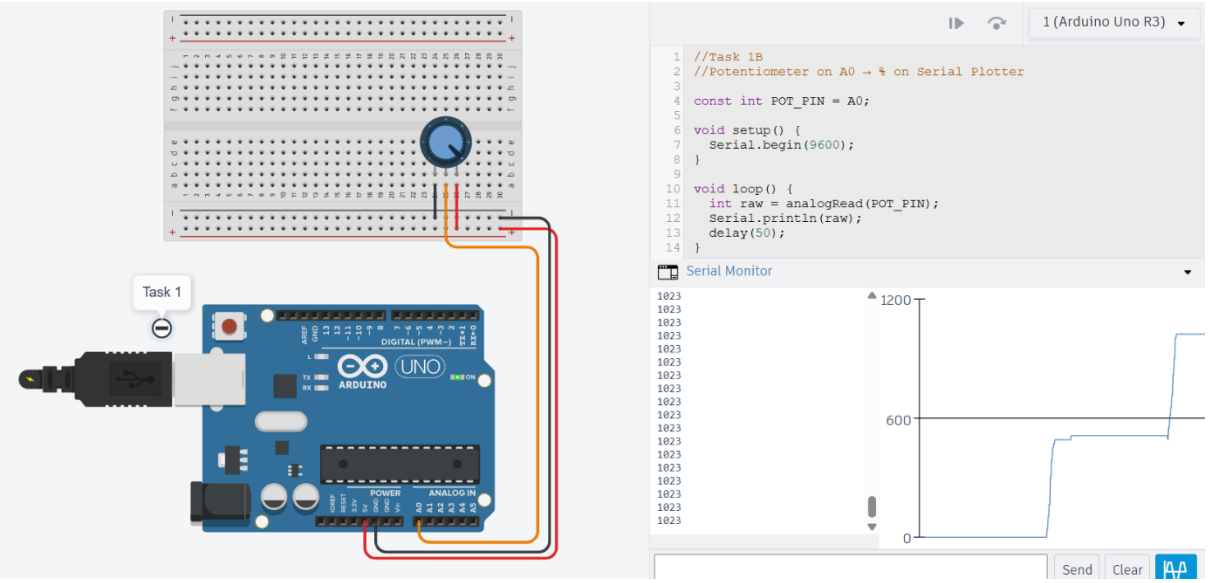

*Max Potentiometer for Serial Monitor*




*Min Potentiometer for Serial Plotter*



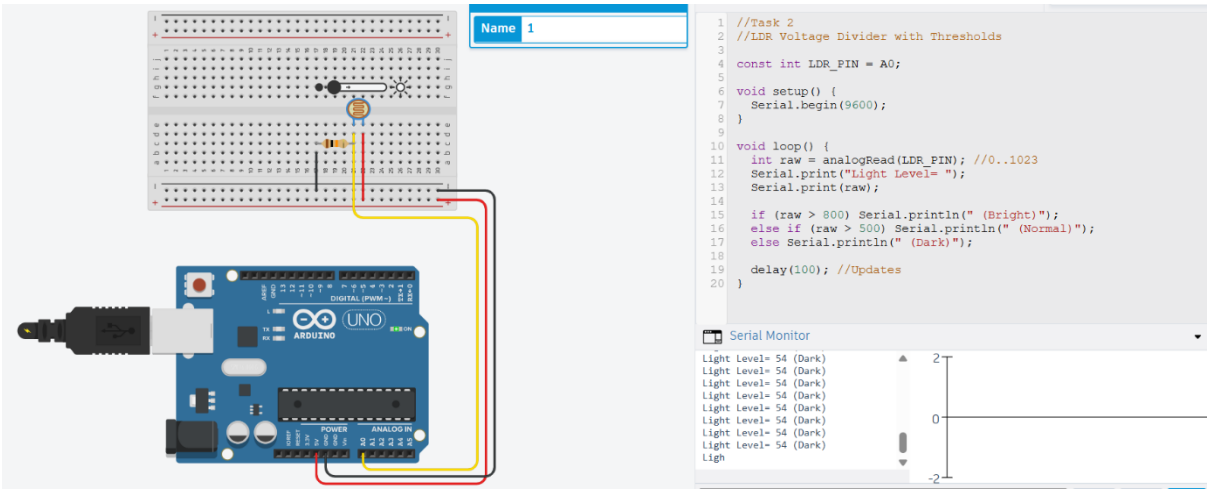

*Mid Potentiometer for Serial Plotter*



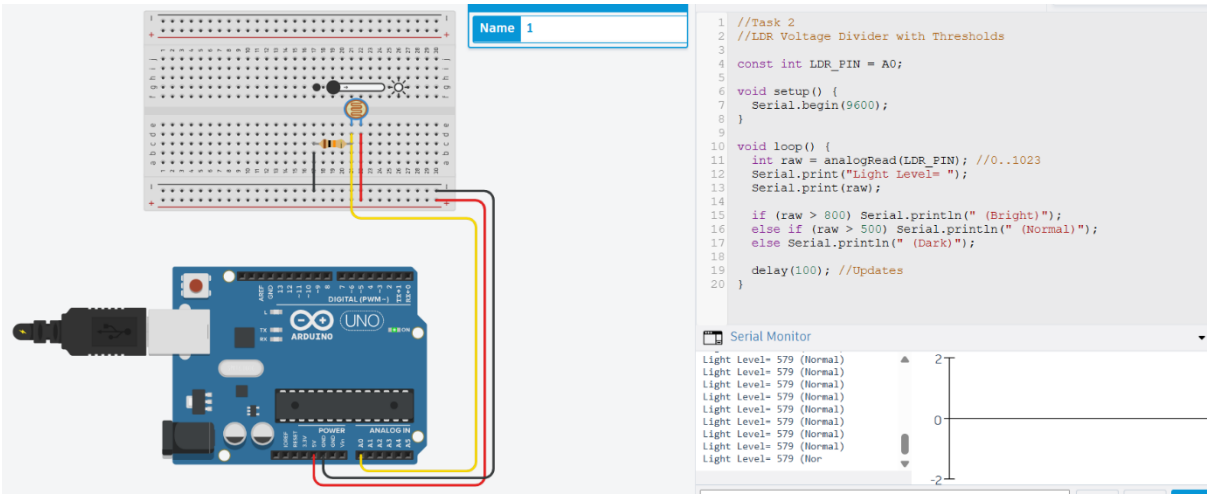

*Max Potentiometer for Serial Plotter*

## Task 2

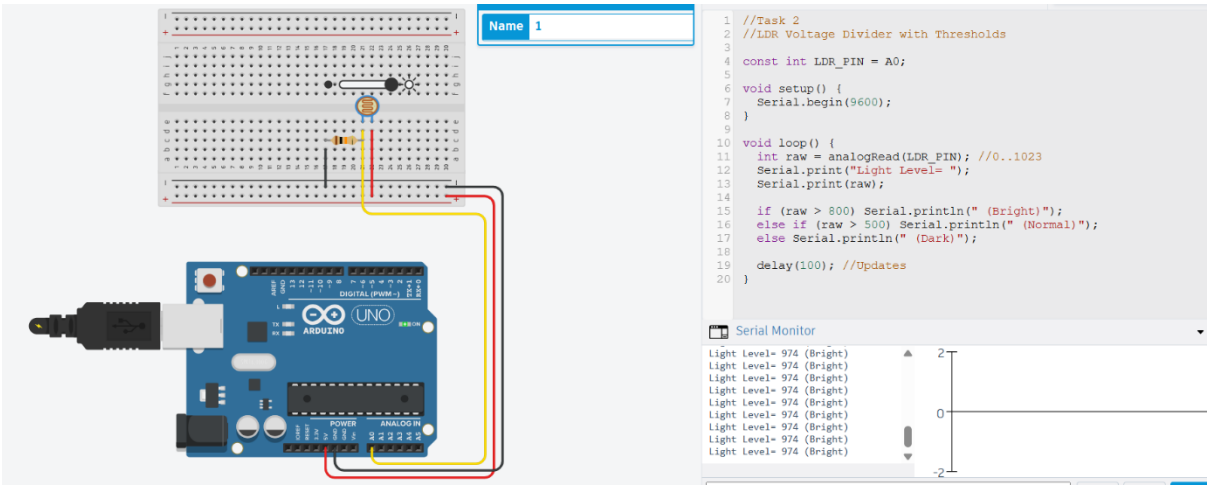
Tinkercad Link: [Affan Task 2]




*Dark on Light Dependent Resistor*



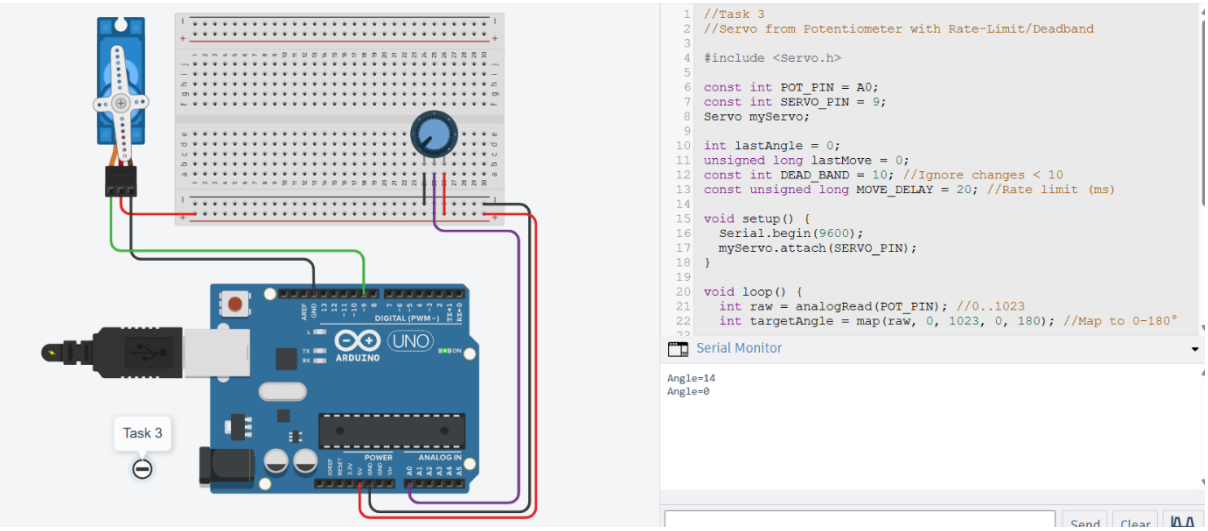

*Normal on Light Dependent Resistor*



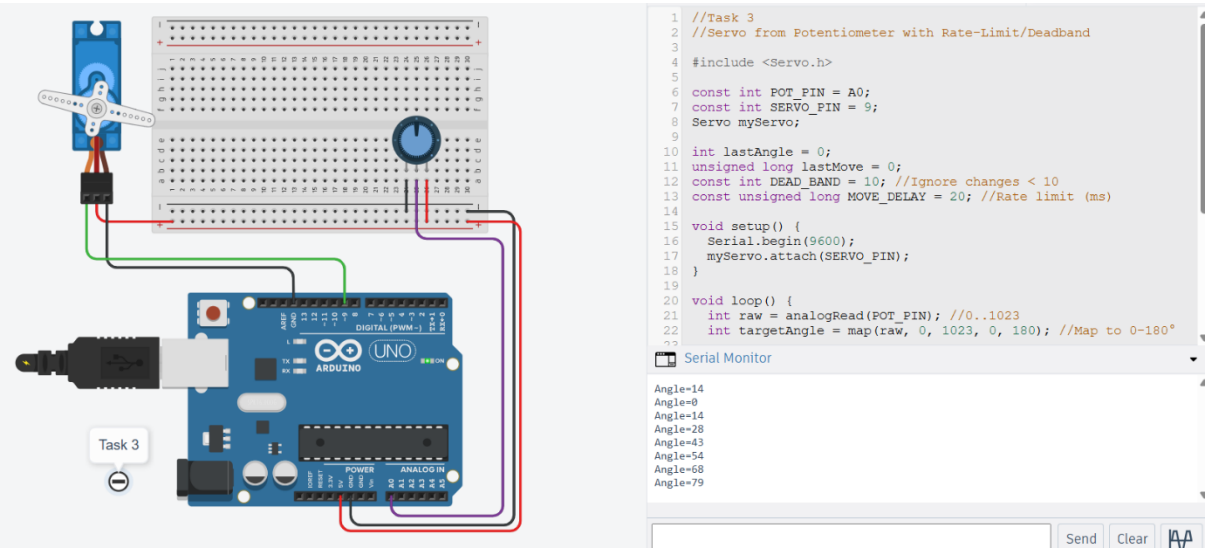

*Bright on Light Dependent Resistor*

## Task 3

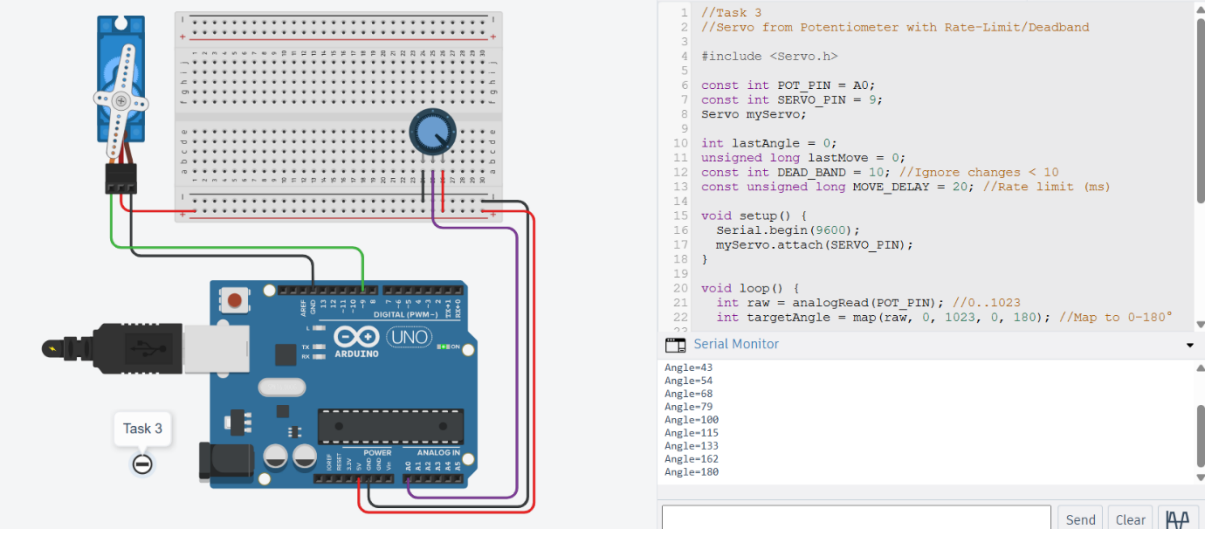
Tinkercad Link: [Affan Task 3]




*0-degree angle for Servo*




*79-degree angle for Servo*




*180-degree angle for Servo*

# Reflections

## Task 1

Mapping is a process whereby scaling the number a range which can range from the minimum to the maximum value. So, the task scaled the raw sensor reading of the potentiometer which is using an analogue value which was 0 to 1023 then it put into percentage from 0 to 100%.

## Task 2

One limitation of Light Dependent Resistor (LDR) is a slow response time which is a disadvantage if an application needed a fast detection in the changes of light which can lead to inaccurate readings. A digital light sensor would be a better choice as it is faster and more accurate instrument.

## Task 3

There exists a correlation between smoothness and responsiveness whereby a larger delay will reduce the jitter, but this slows the servo's reaction to its own movement. An application may need a higher responsiveness, but they must be prepared to sacrifice the smoothness of the servo.