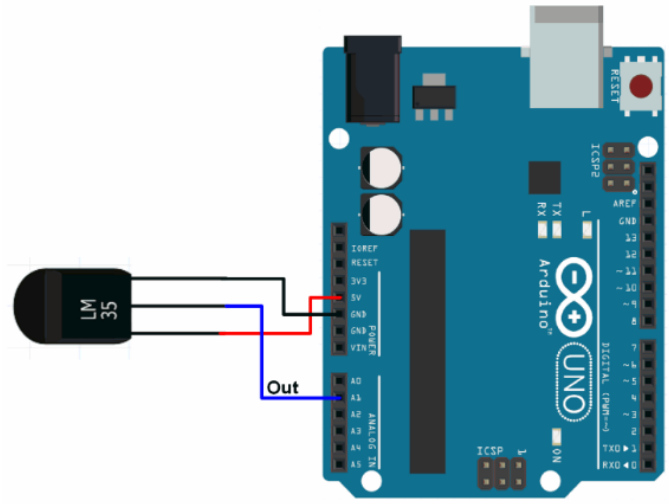
Task 1: LM35 and DHT11 Temperature Sensors

Name: Rahul Karthik S
Register Number: 21BEC1851
Date: 23-07-2024

Task 1.1: LM35 Temperature Sensors

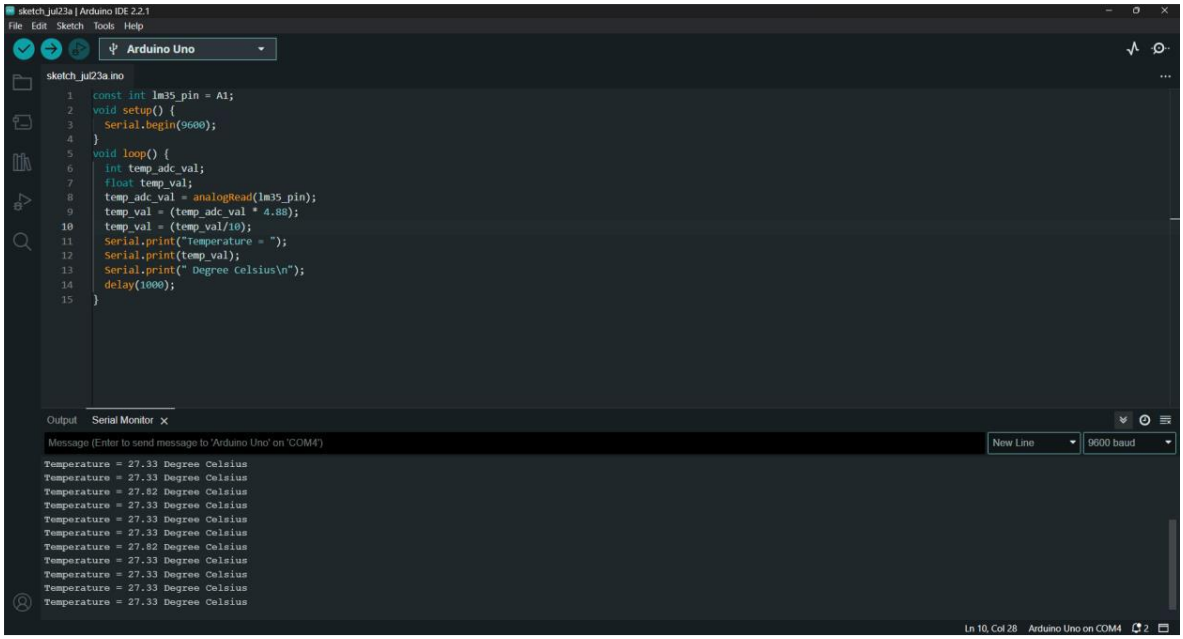
Circuit Diagram:



Program Code:

```
const int lm35_pin = A1;
void setup() {
Serial.begin(9600);
}
void loop() {
int temp_adc_val;
float temp_val;
temp_adc_val = analogRead(lm35_pin);
temp_val = (temp_adc_val * 4.88);
temp_val = (temp_val/10);
Serial.print("Temperature = ");
Serial.print(temp_val);
Serial.print(" Degree Celsius\n");
delay(1000);
}
```

Results:



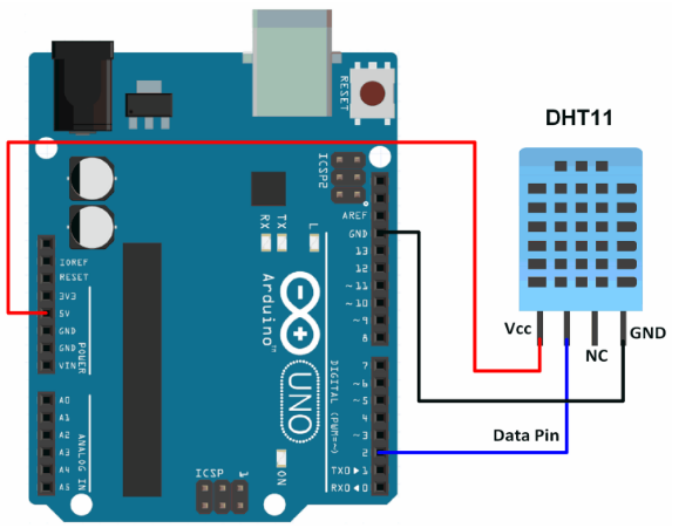
Verified Screenshot:

LM35:

```
const int lm35_pin = A1;
void setup() {
  Serial.begin(9600); }
void loop() {
  int temp_adc_val;
  float temp_val;
  temp_adc_val = analogRead(lm35_pin);
  temp_val = temp_adc_val * 4.88;
  temp_val = temp_val /10;
  Serial.print("Temperature = ");
  Serial.print(temp_val)
  serial.print(" Degree Celcius \n");
  delay(1000);
}
```

Task 1.2: DHT11 Temperature Sensors

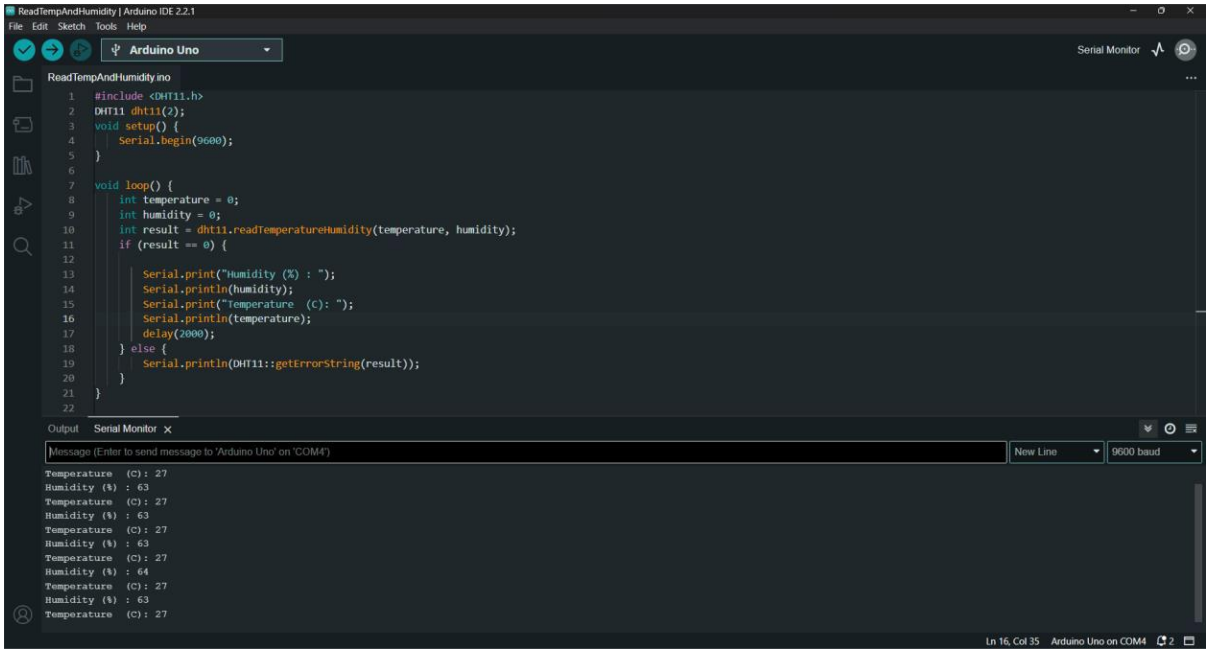
Circuit Diagram:



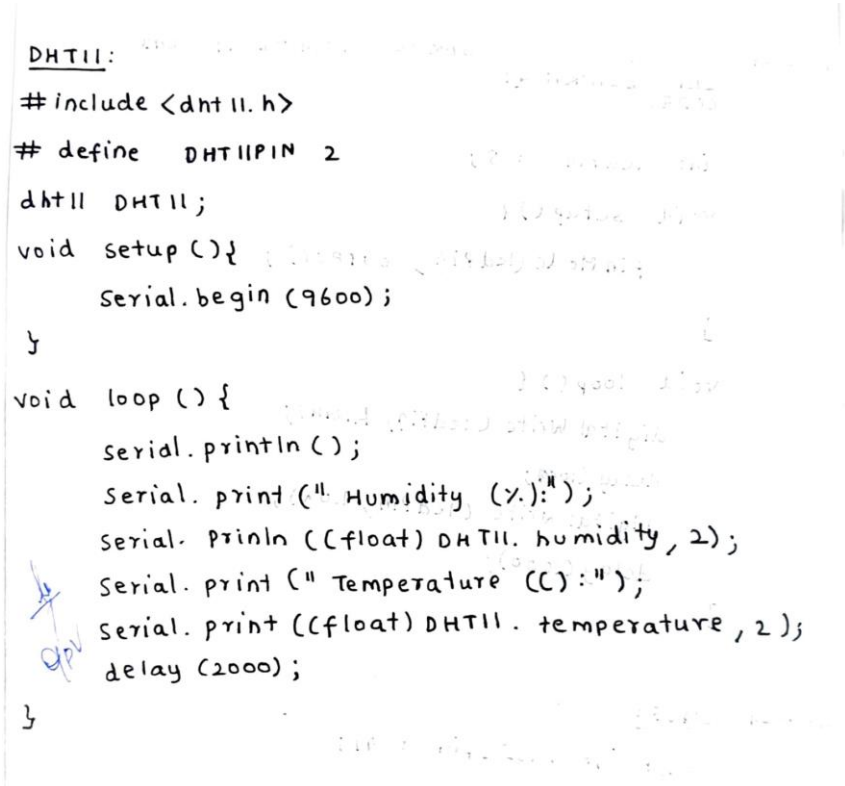
Program Code:

```
#include <dht11.h>
#define DHT11PIN 2
dht11 DHT11;
void setup()
{
  Serial.begin(9600);

}
void loop()
{
  Serial.println();
  int chk = DHT11.read(DHT11PIN);
  Serial.print("Humidity (%): ");
  Serial.println((float)DHT11.humidity, 2);
  Serial.print("Temperature  (C): ");
  Serial.println((float)DHT11.temperature, 2);
  delay(2000);
}
```

Result:



Verified Screenshot:



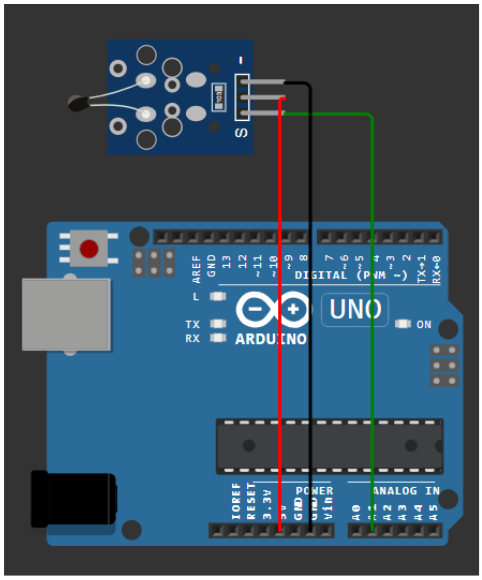
DHT11:

```
#include <dht11.h>
#define DHT11PIN 2
dht11 DHT11;
void setup(){
    Serial.begin(9600);
}
void loop(){
    Serial.println();
    Serial.print("Humidity (%):");
    Serial.println((float)DHT11.humidity, 2);
    Serial.print("Temperature (C):");
    Serial.print((float)DHT11.temperature, 2);
    delay(2000);
}
```

Task 2: Thermistor Sensor – Interfacing Characteristic/Calibration Analysis using Arduino Uno

Name: Rahul Karthik S
Register Number: 21BEC1851
Date: 30-07-2024

Circuit Diagram:



Program Code:

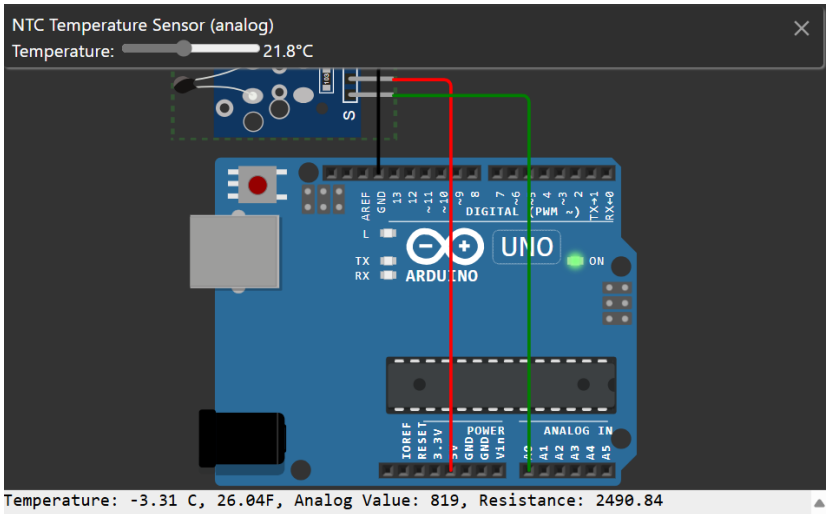
```
const float BETA = 3950;
void setup() {
  Serial.begin(9600);
}
void loop() {
  int analogValue = analogRead(A0);
  float c = 1 / (log(1 / (1023. / analogValue - 1)) / BETA + 1.0 /
298.15) - 273.15;
  float f = ((c*(9.0/5.0))+32);
  float r = 10000*((1023.0/analogValue)-1.0);
  Serial.print("Temperature: ");
  Serial.print(c);
  Serial.print(" C, ");
  Serial.print(f);
  Serial.print("F, ");
  Serial.print("Analog Value: ");
  Serial.print(analogValue);
  Serial.print(", Resistance: ");
  Serial.println(r);
  delay(1000);
}
```

Results:

1. *Arduino IDE:*

```
Temperature: 27.89 ℃ , 82.20 F ; Analog Value: 479 Resistance: 11356.99
Temperature: 27.89 ℃ , 82.20 F ; Analog Value: 479 Resistance: 11356.99
Temperature: 30.35 ℃ , 86.64 F ; Analog Value: 452 Resistance: 12632.74
Temperature: 27.89 ℃ , 82.20 F ; Analog Value: 479 Resistance: 11356.99
Temperature: 27.89 ℃ , 82.20 F ; Analog Value: 479 Resistance: 11356.99
Temperature: 29.16 ℃ , 84.49 F ; Analog Value: 465 Resistance: 12000.00
Temperature: 35.02 ℃ , 95.04 F ; Analog Value: 403 Resistance: 15384.61
Temperature: 28.80 ℃ , 83.83 F ; Analog Value: 469 Resistance: 11812.37
Temperature: 30.91 ℃ , 87.64 F ; Analog Value: 446 Resistance: 12937.22
Temperature: -273.15 ℃ , -459.67 F ; Analog Value: 0 Resistance: inf
Temperature: 32.03 ℃ , 89.66 F ; Analog Value: 434 Resistance: 13571.43
Temperature: 33.56 ℃ , 92.41 F ; Analog Value: 418 Resistance: 14473.68
Temperature: 33.08 ℃ , 91.54 F ; Analog Value: 423 Resistance: 14184.40
Temperature: 32.98 ℃ , 91.37 F ; Analog Value: 424 Resistance: 14127.36
Temperature: 32.98 ℃ , 91.37 F ; Analog Value: 424 Resistance: 14127.36
Temperature: 32.03 ℃ , 89.66 F ; Analog Value: 434 Resistance: 13571.43
Temperature: 31.85 ℃ , 89.32 F ; Analog Value: 436 Resistance: 13463.30
Temperature: 31.75 ℃ , 89.15 F ; Analog Value: 437 Resistance: 13409.61
Temperature: 31.28 ℃ , 88.31 F ; Analog Value: 442 Resistance: 13144.80
Temperature: 30.91 ℃ , 87.64 F ; Analog Value: 446 Resistance: 12937.22
Temperature: 30.54 ℃ , 86.97 F ; Analog Value: 450 Resistance: 12733.33
Temperature: 30.17 ℃ , 86.30 F ; Analog Value: 454 Resistance: 12533.04
Temperature: 29.89 ℃ , 85.81 F ; Analog Value: 457 Resistance: 12385.12
Temperature: 29.62 ℃ , 85.31 F ; Analog Value: 460 Resistance: 12239.13
Temperature: 29.43 ℃ , 84.98 F ; Analog Value: 462 Resistance: 12142.86
Temperature: 29.25 ℃ , 84.65 F ; Analog Value: 464 Resistance: 12047.41
Temperature: 29.07 ℃ , 84.32 F ; Analog Value: 466 Resistance: 11952.79
Temperature: 28.89 ℃ , 84.00 F ; Analog Value: 468 Resistance: 11858.97
Temperature: 28.80 ℃ , 83.83 F ; Analog Value: 469 Resistance: 11812.37
```

2. *Wokwi:*




```
Temperature: -3.31 C, 26.04F, Analog Value: 819, Resistance: 2490.84
Temperature: -0.37 C, 31.33F, Analog Value: 792, Resistance: 2916.67
Temperature: 4.74 C, 40.52F, Analog Value: 741, Resistance: 3805.67
Temperature: 11.51 C, 52.72F, Analog Value: 667, Resistance: 5337.33
Temperature: 11.51 C, 52.72F, Analog Value: 667, Resistance: 5337.33
Temperature: 18.11 C, 64.60F, Analog Value: 591, Resistance: 7309.64
Temperature: 21.82 C, 71.27F, Analog Value: 548, Resistance: 8667.88
```

Verified Screenshot:

1851

Task-2:

21BEC1851

```
const float BETA = 3950;
void setup(){
    Serial.begin(9600);
}
void loop(){
    int analogValue = analogRead(A0);
    float c = 1/(log(1/(1023./analogValue -1))/
              BETA + 1.0/298.15) - 273.15;
    float f = ((c*(9.0/5.0)) + 32);
    float r = 10000 * ((1023.0/analogValue)-1
    Serial.print("Temperature : ");
    Serial.print(*c);
    Serial.print(" c, ");
    Serial.print(f);
    Serial.print(" F, ");
    serial.print("Analog Value : ");
    Serial.print(analogValue);
    Serial.print(", Resistance ");
    Serial.println(r);
    delay(1000);
}
```

21BEC1851

Task 3: LDR, IR and Ultrasonic Sensors

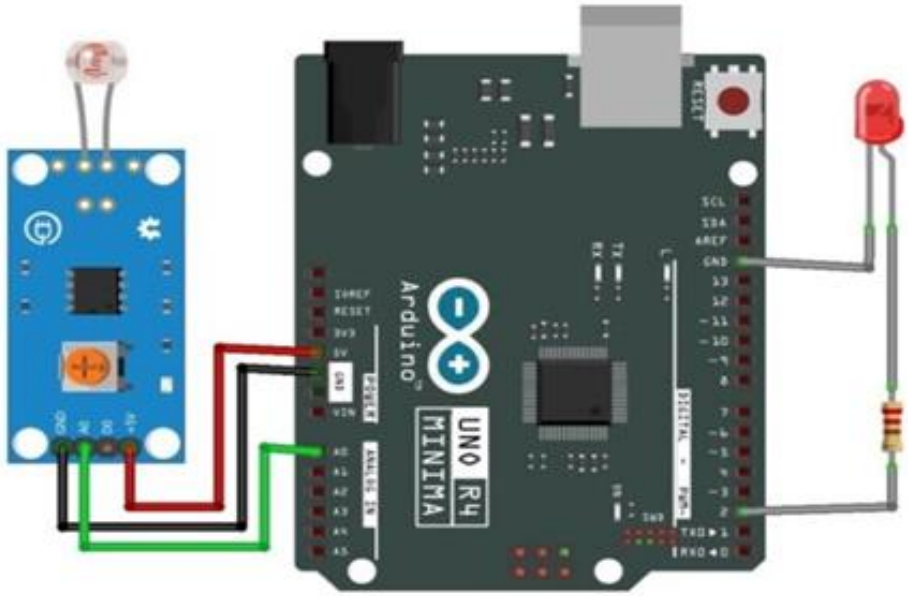
Name: Rahul Karthik S

Register Number: 21BEC1851

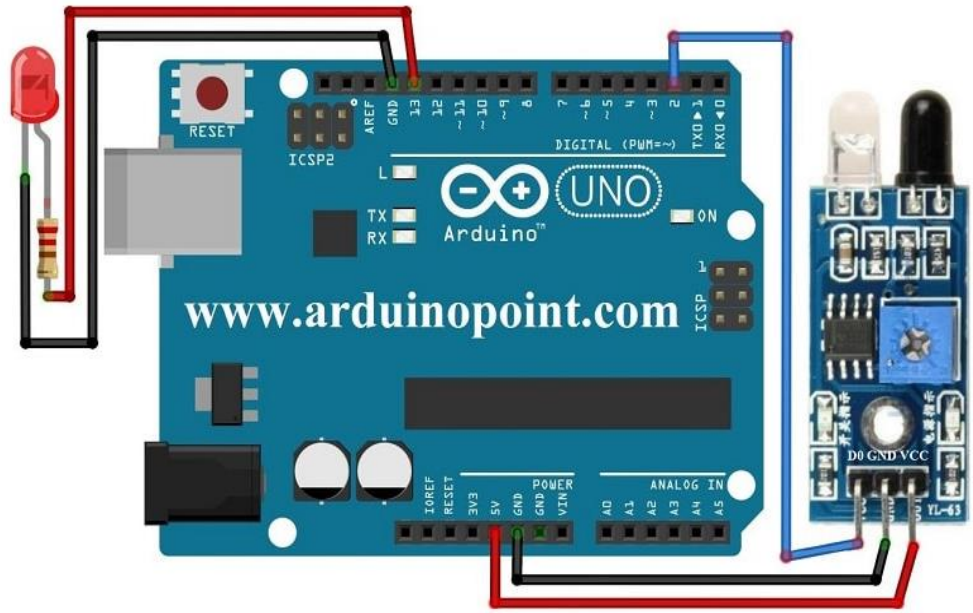
Date: 06-08-2024 and 13-08-2024

Circuit Diagram:

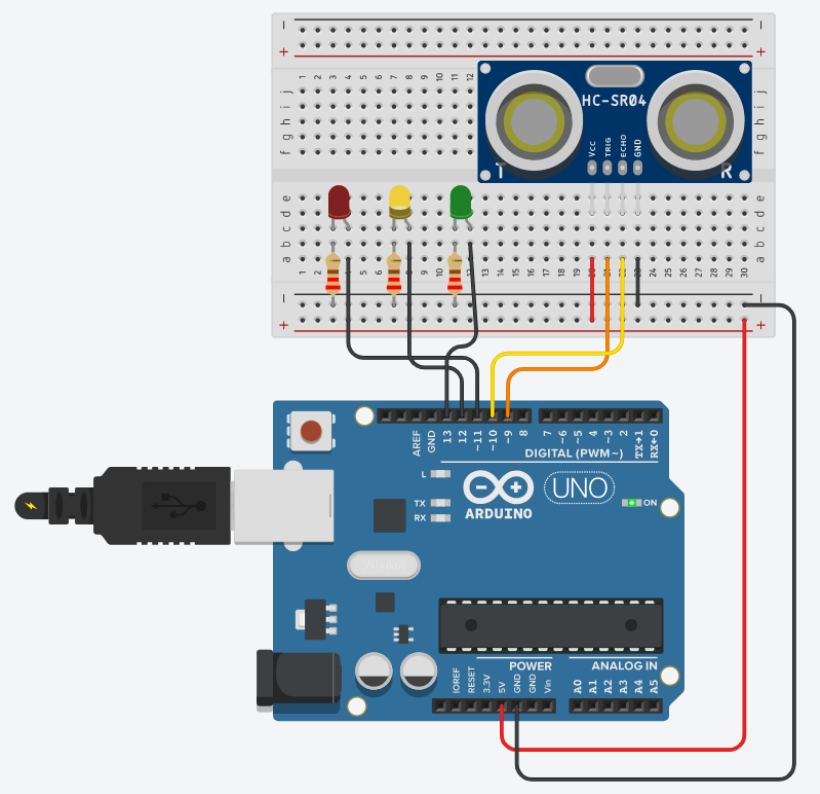
LDR Sensor:



IR Sensor:



Ultrasonic Sensor:



Program Code:

LDR Sensor:

```
const int ledPin = 2;
const int ldrPin = A0;
 void setup()
{
  pinMode(ledPin, OUTPUT); // Set the LED pin as output
  Serial.begin(9600);
}
 void loop()
{
  int ldrValue = analogRead(ldrPin);
  Serial.println(ldrValue);
   // Assuming a lower value means more light
  if (ldrValue > 600)
  {
    digitalWrite(ledPin, HIGH);
  }
  else
  {
    digitalWrite(ledPin, LOW);
  }
   delay(500);
}
```

IR Sensor:

```
int SensorPin = 2;
int OutputPin = 13;
```

```
void setup() {
  pinMode(OutputPin, OUTPUT);
  pinMode(SensorPin, INPUT);
  Serial.begin(9600);
}
void loop() {
  int SensorValue = digitalRead(SensorPin);

  Serial.print("SensorPin Value: ");
  Serial.println(SensorValue);
  delay(100);
   if (SensorValue==LOW){ // LOW MEANS Object Detected
    digitalWrite(OutputPin, HIGH);
  }
  else
  {
    digitalWrite(OutputPin, LOW);
  }
}
```

Ultrasonic Sensor:

```
int trig = 9;
int echo = 10;
int led1 = 11;
int led2 = 12;
int led3 = 13;
```

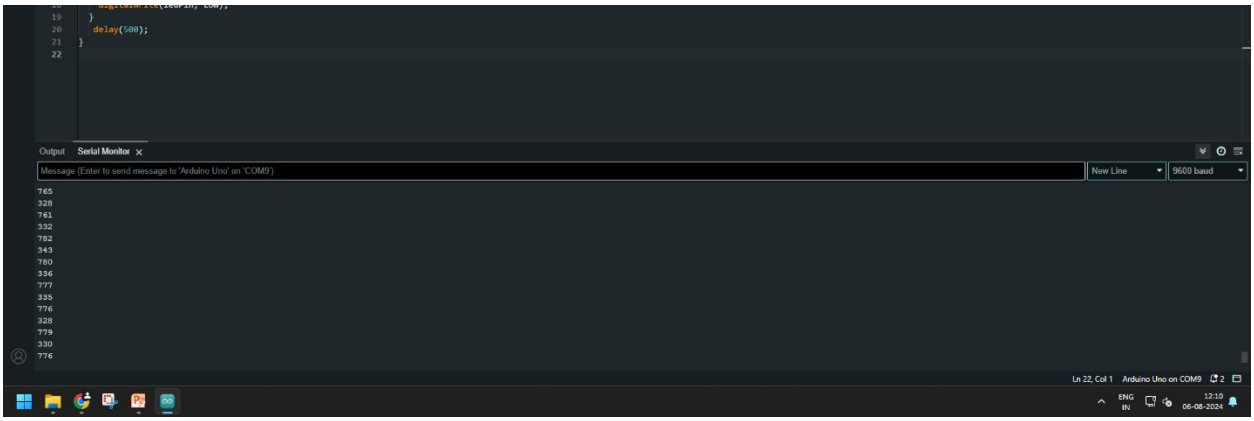
```
long duration = 0;
int cm = 0;
int in = 0;

void setup()
{
  pinMode(trig, OUTPUT);
  pinMode(echo, INPUT);
  pinMode(led1, OUTPUT);
  pinMode(led2, OUTPUT);
  pinMode(led3, OUTPUT);
  Serial.begin(9600);
  Serial.println("Serial Started...");
}

void loop()
{
  digitalWrite(trig, LOW);
  digitalWrite(trig, HIGH);
  digitalWrite(trig, LOW);

  int duration = pulseIn (echo, HIGH);
  cm = duration*0.034/2;
  in = duration*0.0133/2;
```

```
Serial.println(in);


if (in >= 108) {
  digitalWrite(led3, LOW);
  digitalWrite(led2, LOW);
  digitalWrite(led1, LOW);
  delay(1000);
  digitalWrite(led3, HIGH);
  digitalWrite(led2, LOW);
  digitalWrite(led1, LOW);
  delay(1000);
}


 else if (in < 108 && in > 36){
  digitalWrite(led3, LOW);
  digitalWrite(led2, LOW);
  digitalWrite(led1, LOW);
  delay(600);
  digitalWrite(led3, LOW);
  digitalWrite(led2, HIGH);
  digitalWrite(led1, LOW);
  delay(600);
}
```

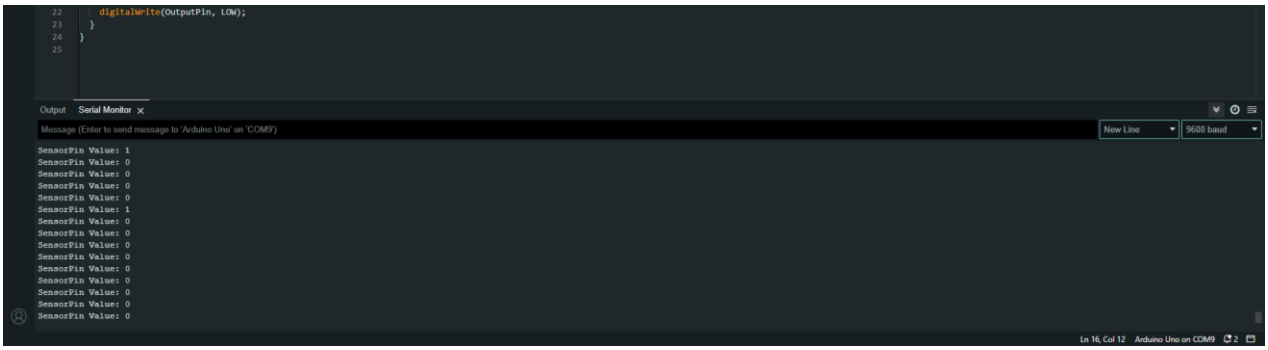
```
  else if (in <= 36 ){
   digitalWrite(led3, LOW);
   digitalWrite(led2, LOW);
   digitalWrite(led1, LOW);
   delay(300);
   digitalWrite(led3, LOW);
   digitalWrite(led2, LOW);
   digitalWrite(led1, HIGH);
   delay(300);
  }
}
```

Output:

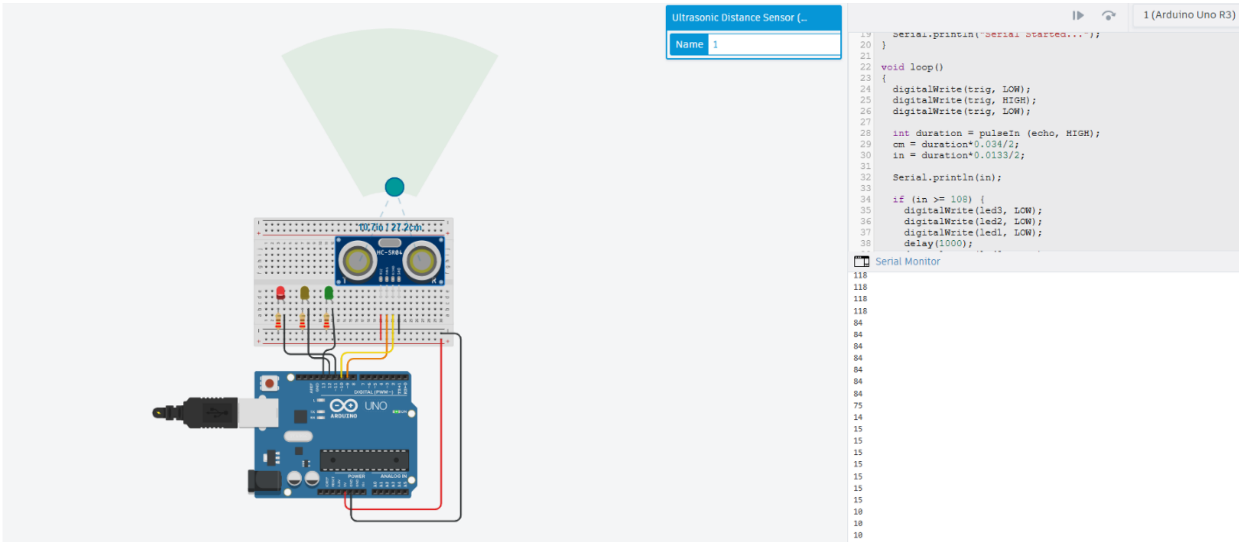
LDR Sensor Output:

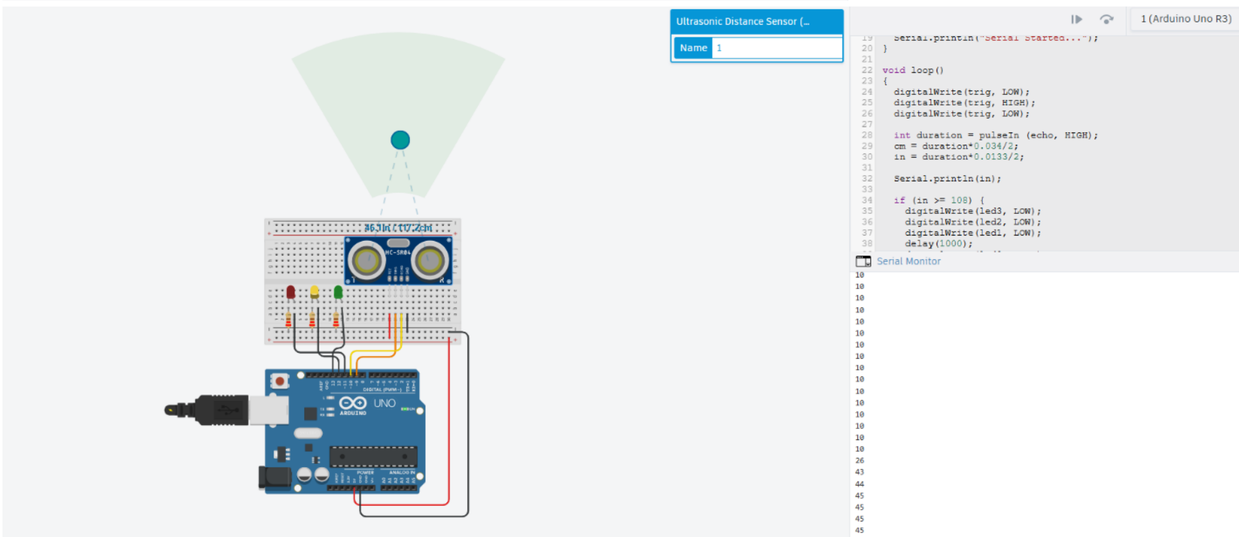


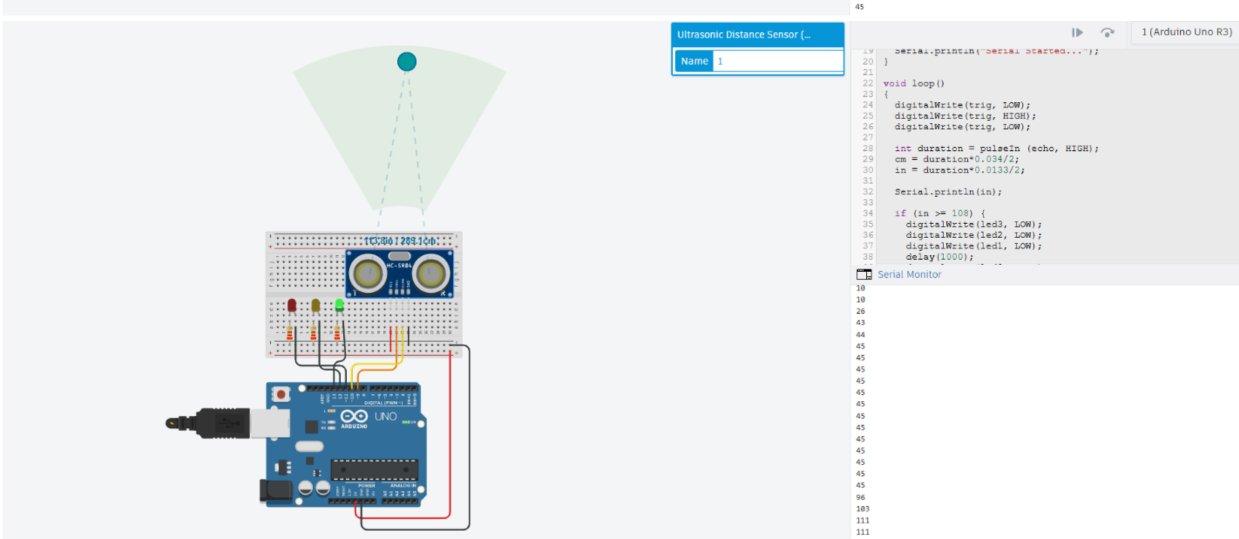
IR Sensor Output:



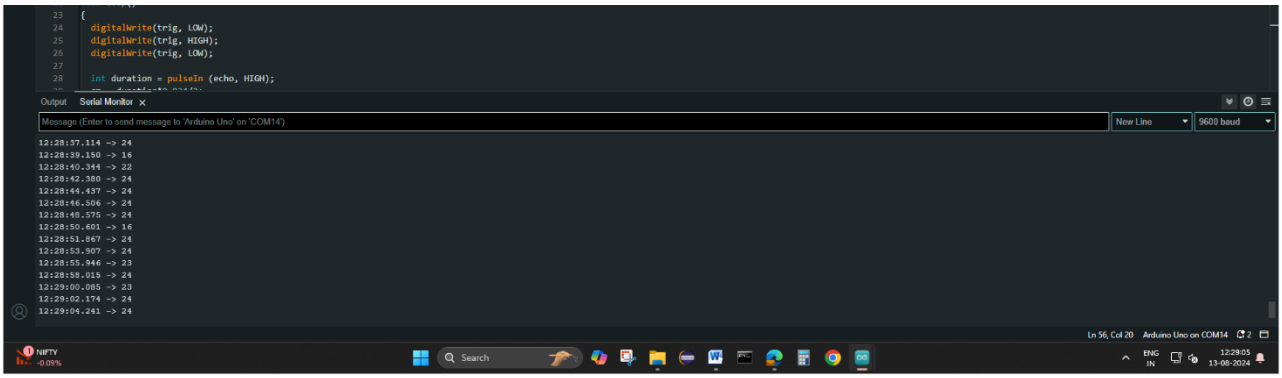
Ultrasonic Sensor Tinker CAD Output:





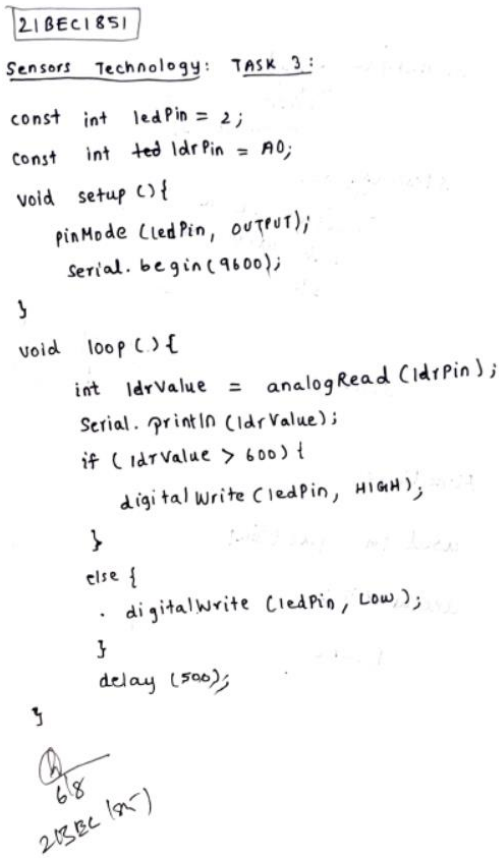


Ultrasonic Sensor Arduino IDE:



Verified Screenshots:

LDR Sensor:



IR Sensor:

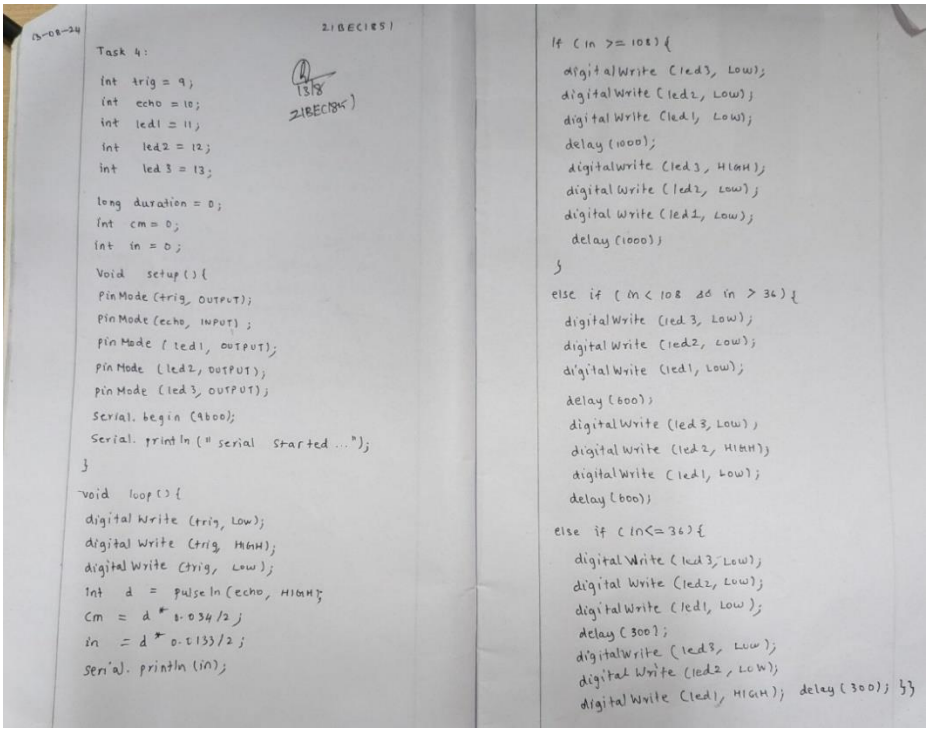
21BEC1851

Task 3.1:

```
int SP = 2;
int OP = 13;

Void setup() {
    Pin Mode (OP, OUTPUT);
    pin Mode (SP, INPUT);
    Serial.begin (9600);

}
Void loop() {
    int sensorVal = digital Read (~~Sens~~ SP);
    Serial.print (" sensor Value is:");
    Serial.println (sensorVal);
    delay (100);
    if (sensorVal == LOW) {
        digitalwrite (OP, HIGH);
    }
    else {
        digital write (OP, LOW);
    }
}
```

21BEC1851

Ultrasonic Sensor:



```
Task 4:
int trig = 9;
int echo = 10;
int led1 = 11;
int led2 = 12;
int led 3 = 13;

long duration = 0;
int cm = 0;
int in = 0;
Void setup() {
Pin Mode (trig, OUTPUT);
Pin Mode (echo, INPUT);
Pin Mode (led1, OUTPUT);
Pin Mode (led2, OUTPUT);
Pin Mode (led 3, OUTPUT);
Serial.begin (9600);
Serial.println (" serial Started ...");
}
Void loop() {
digital Write (trig, Low);
digital Write (trig, HIGH);
digital Write (trig, Low);
int d = pulse In (echo, HIGH);
Cm = d * 0.034 / 2;
in = d * 0.0133 / 2;
Serial.println (in);

if (in >= 108) {
digitalWrite (led3, Low);
digitalWrite (led2, Low);
digital Write (led1, Low);
delay (1000);
digitalwrite (led 3, HIGH);
digital Write (led2, Low);
digital Write (led1, Low);
delay (1000);
}
else if (in < 108 && in > 36) {
digitalWrite (led 3, Low);
digital Write (led2, Low);
digitalWrite (led1, Low);
delay (600);
digitalWrite (led 3, Low);
digitalWrite (led2, HIGH);
digitalWrite (led1, Low);
delay (600);
else if (in<= 36) {
digitalWrite (led 3, Low);
digital Write (led2, Low);
digitalWrite (led1, Low);
delay (300);
digitalWrite (led 3, Low);
digital Write (led2, Low);
digital Write (led1, HIGH); delay (300); }}
```

Task 4: Barometric Sensor

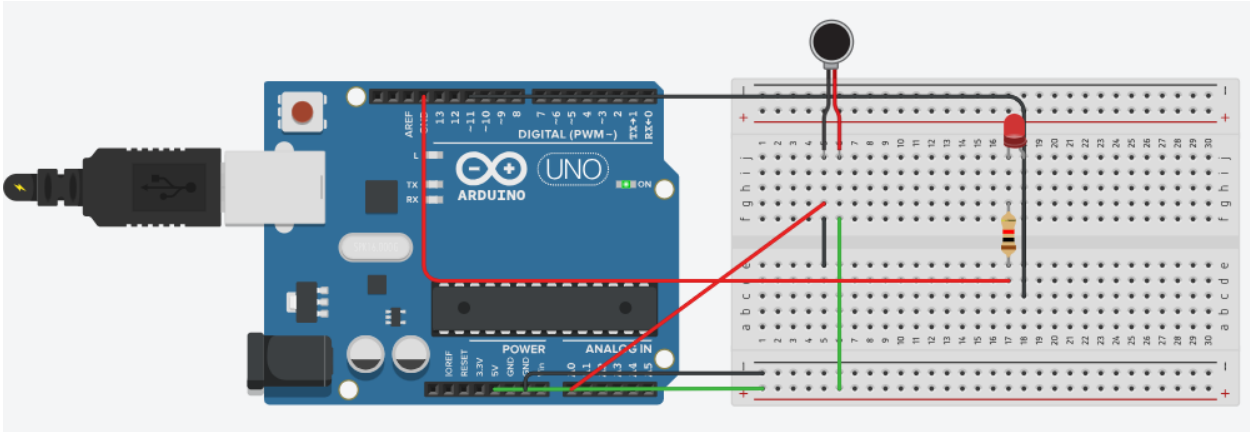
Name: Rahul Karthik S

Register Number: 21BEC1851

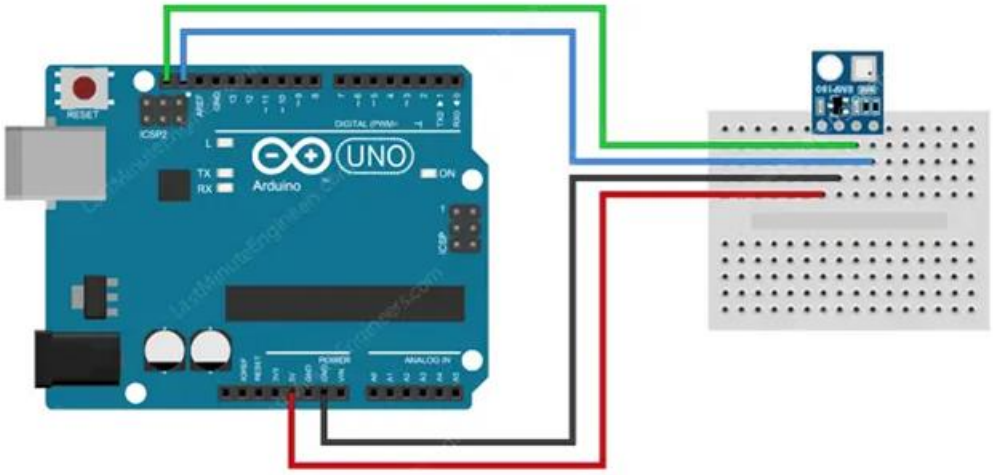
Date: 03-09-2024

Circuit Diagram:

1. *TinkerCAD*



2. *Arduino IDE*



Program Code:

1. *TinkerCAD:*

```
int fsrAnalogPin = 0; // FSR is connected to analog 0
```

```
int LEDpin = 11;      // connect Red LED to pin 11 (PWM pin)
int fsrReading;      // the analog reading from the FSR resistor divider
int LEDbrightness;

void setup(void) {
  Serial.begin(9600);   // We'll send debugging information via the Serial monitor
  pinMode(LEDpin, OUTPUT);
}

void loop(void) {
  fsrReading = analogRead(fsrAnalogPin);
  Serial.print("Analog reading = ");
  Serial.println(fsrReading);

  // we'll need to change the range from the analog reading (0-1023) down to the range
  // used by analogWrite (0-255) with map!
  LEDbrightness = map(fsrReading, 0, 1023, 0, 255);
  // LED gets brighter the harder you press
  analogWrite(LEDpin, LEDbrightness);

  delay(100);
}
```

2. *Arduino IDE:*

```
#include <Wire.h>
#include <Adafruit_BMP085.h>
```

```
#define seaLevelPressure_hPa 1013.25
Adafruit_BMP085
void setup() {
  Serial.begin(9600);
  if (!bmp.begin()) {
  Serial.println("Could not find a valid BMP085 sensor, check wiring!");
  while (1) {}
  }
}

void loop() {
    Serial.print("Temperature = ");
    Serial.print(bmp.readTemperature());
    Serial.println(" *C");

    Serial.print("Pressure = ");
    Serial.print(bmp.readPressure());
    Serial.println(" Pa");

    Serial.print("Altitude = ");
    Serial.print(bmp.readAltitude());
    Serial.println(" meters");

    Serial.print("Pressure at sealevel (calculated) = ");
    Serial.print(bmp.readSealevelPressure());
    Serial.println(" Pa");
```

```
    Serial.print("Real altitude = ");
    Serial.print(bmp.readAltitude(seaLevelPressure_hPa * 100));
    Serial.println(" meters");


    Serial.println();
    delay(5000);
}
```

Output:

1. *TinkerCAD:*

```
Analog reading = 1023
Analog reading = 1023
Analog reading = 1023
Analog reading = 1023
Analog reading =
```

2. *Arduino IDE:*

```
Temperature = 28.20 *C
Pressure = 100422 Pa
Altitude = 74.78 meters
Pressure at sealevel (calculated) = 100427 Pa
Real altitude = 74.53 meters

Temperature = 28.10 *C
Pressure = 100421 Pa
Altitude = 75.12 meters
Pressure at sealevel (calculated) = 100430 Pa
Real altitude = 74.62 meters
```

Verified Screenshots:

21BEC1851

03-09-24 Interface BMP180 Barometer Pressure & Temperature Sensors with Arduino

Code:

```
#include <Wire.h>
#include <Adafruit_BMP085.h>
#define seaLevelPressure_hPa 1013.25

Adafruit_BMP085 bmp;

void setup() {
  Serial.begin(9600);
  if (!bmp.begin()){
    Serial.println("Could not find valid
                    sensor");
    while(1){}
  }
}

void loop() {
  Serial.print("Temperature = ");
  Serial.print(bmp.readTemperature());
  Serial.println(" *C");

  Serial.print("Pressure = ");
  Serial.print(bmp.readPressure());
  Serial.println(" Pa");

  Serial.print("Altitude = ");
  ~~Serial.print(bmp.readSealevelPressure())~~
  Serial.print(bmp.readAltitude());
  Serial.println(" meters");

  Serial.print("Pressure at Sea Level = ");
  Serial.print(bmp.readSeaLevelPressure());
  Serial.println(" Pa");

  Serial.println("Real Altitude = ");
  Serial.print(bmp.realAltitude
                (seaLevelPressure_hPa
                 *100));
  Serial.println(" meters");
  delay(5000);
}
```

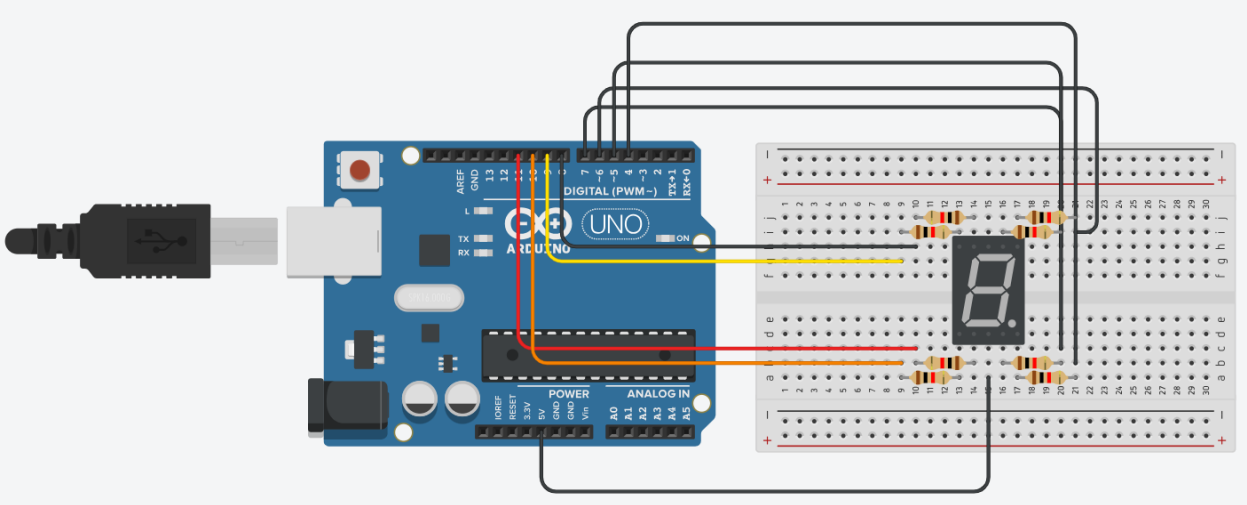
Task 5.1: 7-Segment Display

Name: Rahul Karthik S

Register Number: 21BEC1851

Date: 24-09-2024

Circuit Diagram:



Program Code:

```
int a=7;

int b=6;

int c=5;

int d=11;

int e=10;

int

  f=8;

int g=9;

int dp=4;

//display number 1

void display1(void)
```

```
{

   digitalWrite(b,LOW);
   digitalWrite(c,LOW);
}
//display number2
void display2(void)
{
   digitalWrite(a,LOW);
   digitalWrite(b,LOW);

   digitalWrite(g,LOW);
  digitalWrite(e,LOW);
   digitalWrite(d,LOW);
}

// display number3
void display3(void)
{
   digitalWrite(a,LOW);

   digitalWrite(b,LOW);

  digitalWrite(c,LOW);
   digitalWrite(d,LOW);
```

```
   digitalWrite(g,LOW);
}
// display number4
void display4(void)
{

   digitalWrite(f,LOW);
   digitalWrite(b,LOW);
   digitalWrite(g,LOW);

  digitalWrite(c,LOW);

}
// display number5
void display5(void)

{
   digitalWrite(a,LOW);
   digitalWrite(f,LOW);
   digitalWrite(g,LOW);

  digitalWrite(c,LOW);
   digitalWrite(d,LOW);
}
// display number6
void
```

```
  display6(void)
{
    digitalWrite(a,LOW);
    digitalWrite(f,LOW);

    digitalWrite(g,LOW);
   digitalWrite(c,LOW);
    digitalWrite(d,LOW);

    digitalWrite(e,LOW);
}
// display number7
void display7(void)

{
   digitalWrite(a,LOW);
    digitalWrite(b,LOW);
    digitalWrite(c,LOW);
}

// display number8
void display8(void)
{
    digitalWrite(a,LOW);

    digitalWrite(b,LOW);
```

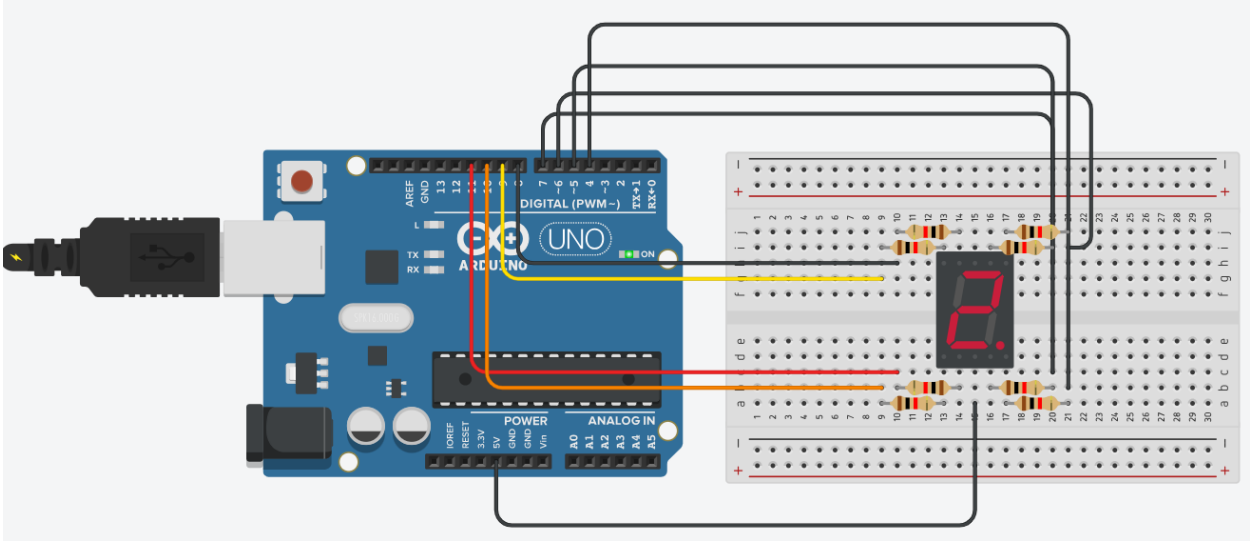
```
   digitalWrite(g,LOW);
  digitalWrite(c,LOW);

   digitalWrite(d,LOW);
   digitalWrite(e,LOW);
  digitalWrite(f,LOW);

}
void clearDisplay(void)
{
   digitalWrite(a,HIGH);
   digitalWrite(b,HIGH);

   digitalWrite(g,HIGH);
  digitalWrite(c,HIGH);
   digitalWrite(d,HIGH);

   digitalWrite(e,HIGH);
  digitalWrite(f,HIGH);
}
void display9(void)

{
   digitalWrite(a,LOW);
   digitalWrite(b,LOW);
   digitalWrite(g,LOW);
```

```
  digitalWrite(c,LOW);
    digitalWrite(d,LOW);
  digitalWrite(f,LOW);

}
void display0(void)
{
    digitalWrite(a,LOW);
    digitalWrite(b,LOW);

    digitalWrite(c,LOW);
    digitalWrite(d,LOW);
    digitalWrite(e,LOW);

  digitalWrite(f,LOW);
}
void setup()
{
    int i;
    for(i=4;i<=11;i++)

        pinMode(i,OUTPUT);
}
void loop()
{
```

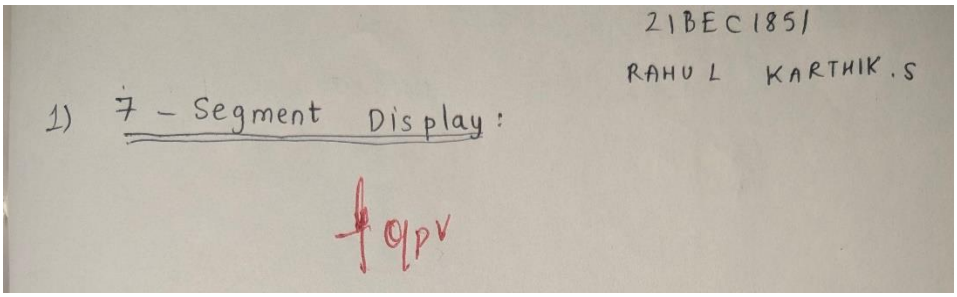
```
  while(1)

  {   clearDisplay();
display0();
      delay(1000);
      clearDisplay();

      display1();
      delay(1000);
      clearDisplay();
      display2();

      delay(1000);
      clearDisplay();
      display3();

      delay(1000);
      clearDisplay();
      display4();
      delay(1000);

      clearDisplay();
      display5();
      delay(1000);
      clearDisplay();
```

```
    display6();
    delay(1000);
    clearDisplay();

    display7();
    delay(1000);
    clearDisplay();
    display8();

    delay(1000);
    clearDisplay();
    display9();

    delay(1000);
  }
}
```

Output:

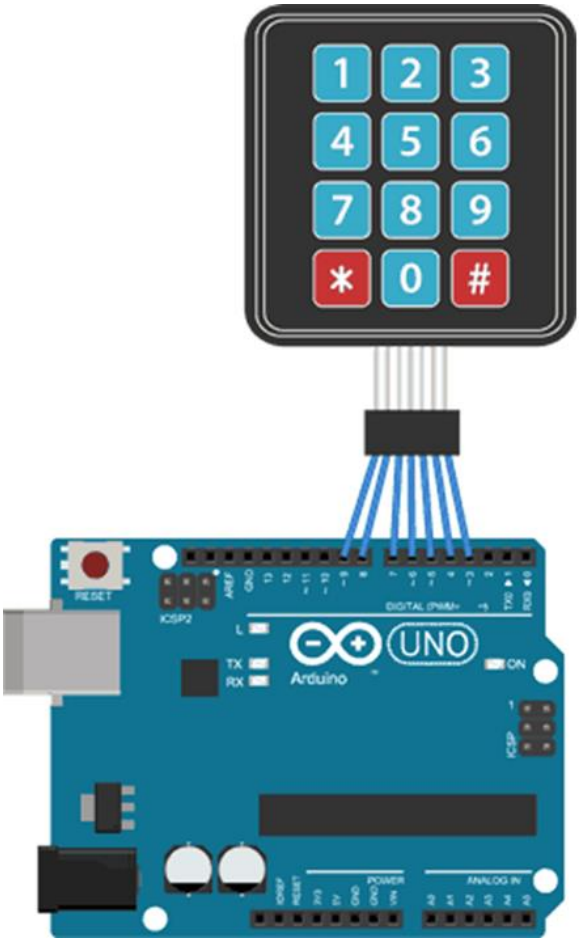


Verified Screenshots:



Task 5.2: Keypad 4 X 3

Circuit Diagram:



Program Code:

```
#include <Keypad.h>
```

```
const int ROW_NUM = 4; //four rows
const int COLUMN_NUM = 3; //three columns

char keys[ROW_NUM][COLUMN_NUM] = {
 {'1','2','3'},
 {'4','5','6'},
 {'7','8','9'},
 {'*','0','#'}
};

byte pin_rows[ROW_NUM] = {9, 8, 7, 6}; //connect to the row pinouts of the keypad
byte pin_column[COLUMN_NUM] = {5, 4, 3}; //connect to the column pinouts of the keypad

Keypad keypad = Keypad( makeKeymap(keys), pin_rows, pin_column, ROW_NUM,
COLUMN_NUM );

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

void loop(){
 char key = keypad.getKey();

 if (key){
  Serial.println(key);
 }
```

}

Output:

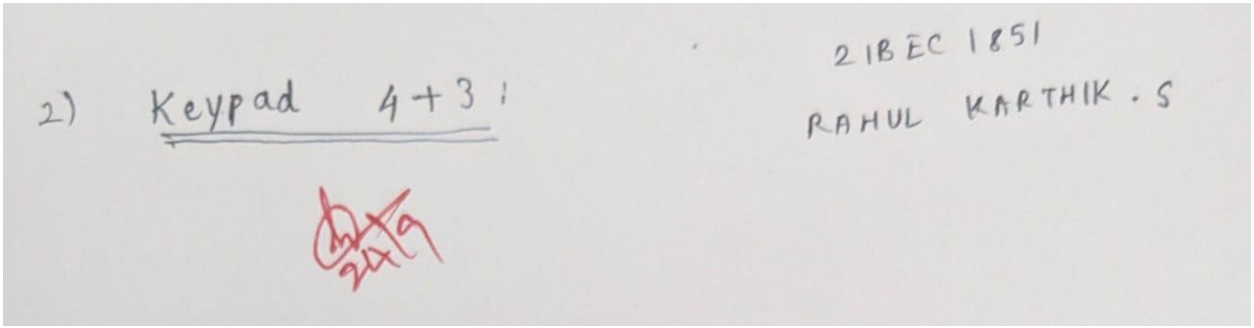
```
17
18  void setup(){
19    Serial.begin(9600);
```

Output Serial Monitor ×

Message (Enter to send message to 'Arduino Uno' on 'COM19')

```
12:35:14.287 -> 1
12:35:15.511 -> 8
12:35:16.171 -> 4
12:35:23.146 -> 6
12:35:25.816 -> 1
12:35:26.745 -> 5
12:35:27.399 -> 3
12:35:28.897 -> 1
12:35:29.437 -> 8
12:35:29.841 -> 5
12:35:30.307 -> 3
12:35:31.835 -> 1
12:35:32.402 -> 8
12:35:32.872 -> 5
12:35:33.327 -> 1
12:35:33.864 -> 1
12:35:34.398 -> 8
12:35:34.868 -> 4
12:35:36.705 -> 1
12:35:37.352 -> 0
12:35:38.669 -> 9
```

Verified Screenshots:



Task 6.1: Gas Sensor with Indication

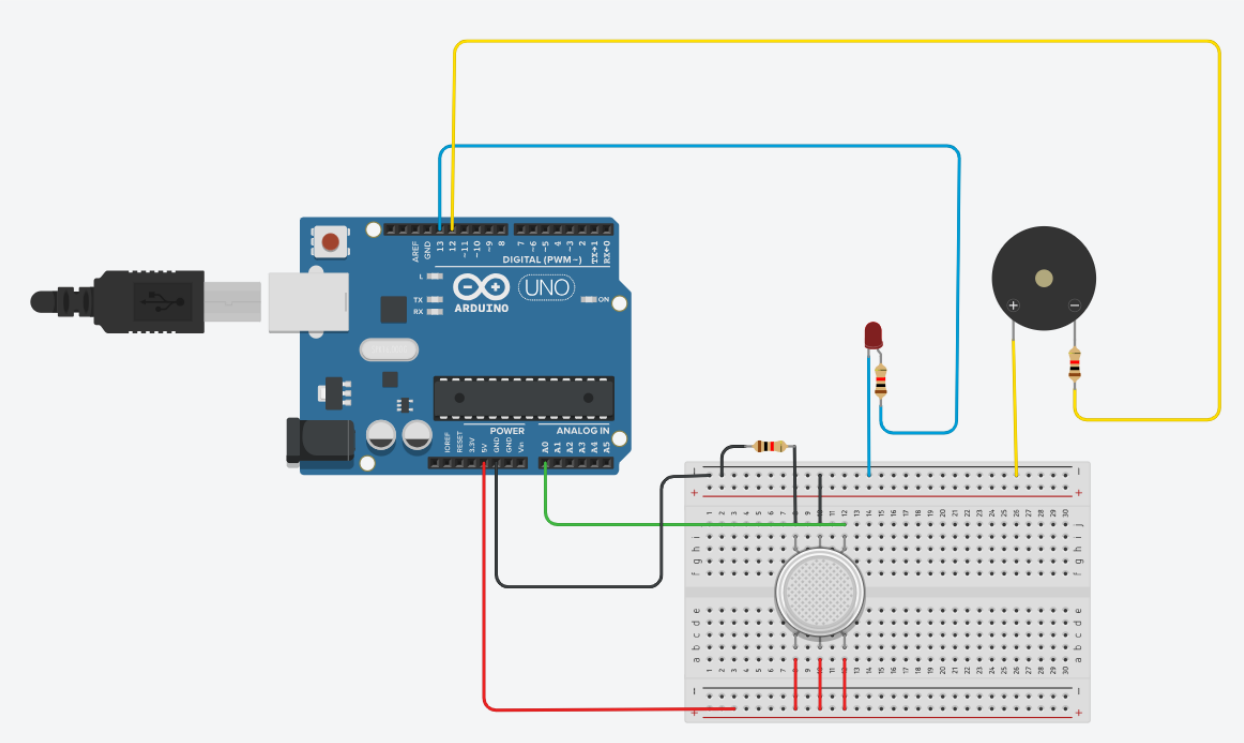
Name: Rahul Karthik S

Register Number: 21BEC1851

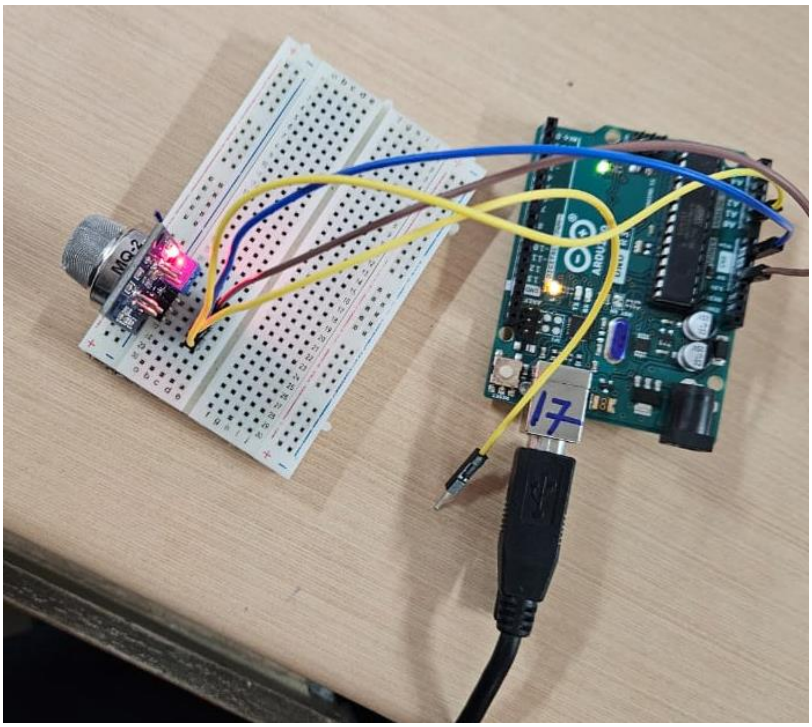
Date: 22-09-2024

Circuit Diagram:

1. *TinkerCAD:*



2. *Arduino IDE:*



Program Code:

```
int redLed = 12;
int greenLed = 11;
int buzzer = 10;
int smokeA0 = A5;
// Your threshold value
int sensorThres = 400;

void setup() {
  pinMode(redLed, OUTPUT);
  pinMode(greenLed, OUTPUT);
  pinMode(buzzer, OUTPUT);
  pinMode(smokeA0, INPUT);
  Serial.begin(9600);
}
```

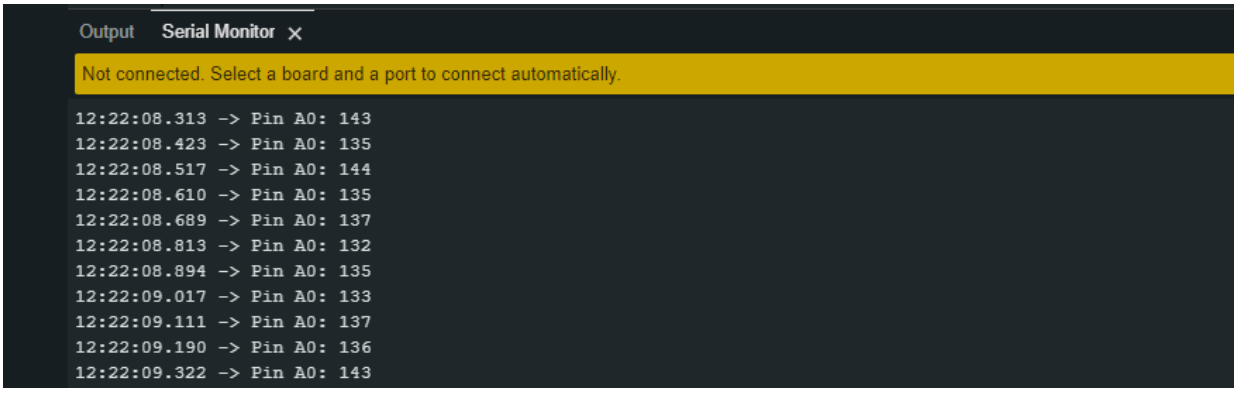
```
void loop() {
  int analogSensor = analogRead(smokeA0);


  Serial.print("Pin A0: ");
  Serial.println(analogSensor);
  // Checks if it has reached the threshold value
  if (analogSensor > sensorThres)
  {
    digitalWrite(redLed, HIGH);
    digitalWrite(greenLed, LOW);
    tone(buzzer, 1000, 200);
  }
  else
  {
    digitalWrite(redLed, LOW);
    digitalWrite(greenLed, HIGH);
    noTone(buzzer);
  }
  delay(100);
}
```

Output:

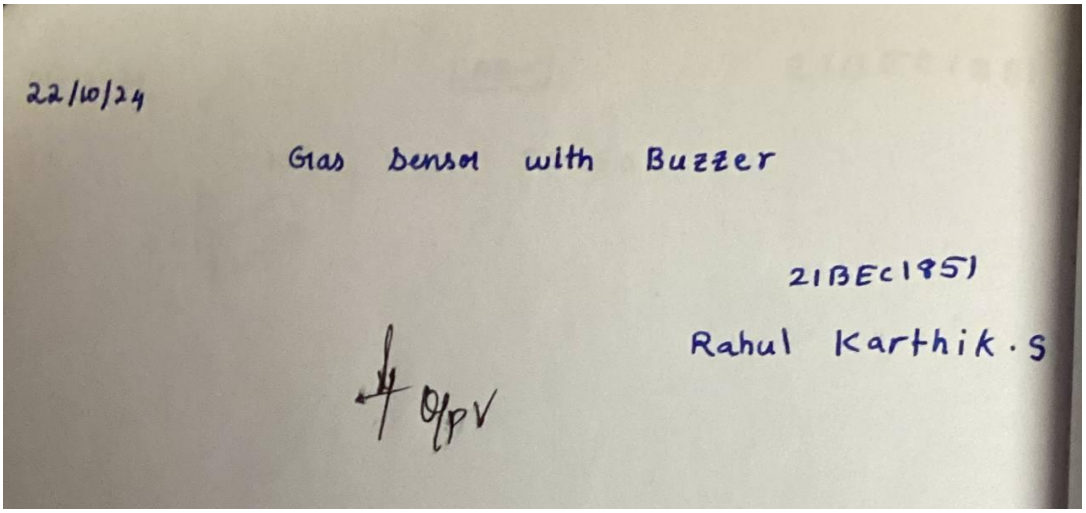
1. *TinkerCAD:*

```
sensor value is :
119.00
No gas Detected
sensor value is :
119.00
No gas Detected
284.00
Gas detected, Buzzer is ON
sensor value is :
128.00
No gas Detected
sensor value is :
134.00
No gas Detected
sensor value is :
134.00
No gas Detected
sensor value is :
134.00
No gas Detected
sensor value is :
134.00
No gas Detected
sensor value is :
134.00
No gas Detected
sensor value is :
134.00
No gas Detected
sensor value is :
134.00
No gas Detected
sensor value is :
134.00
No gas Detected
```

2. *Arduino IDE:*



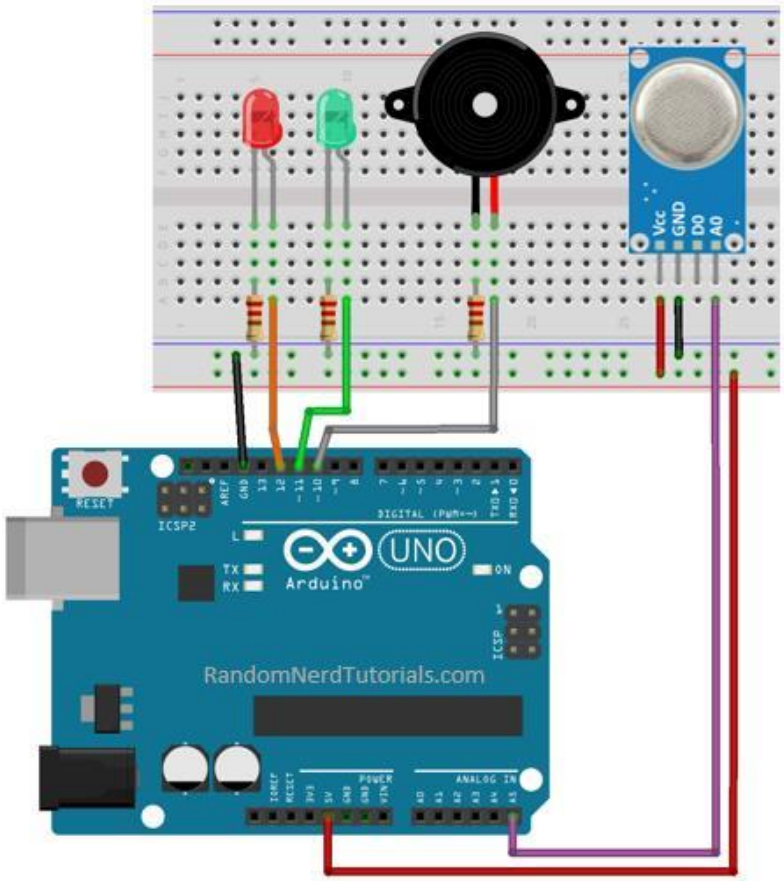
Verified Screenshots:



Task 6.2: Gas Sensor Calibration

Date: 01-10-2024

Circuit Diagram:



Program Code:

```
#define    MQ_PIN                    (0)     //define which analog input channel you are going to
use

#define    RL_VALUE                  (5)     //define the load resistance on the board, in kilo ohms

#define    RO_CLEAN_AIR_FACTOR       (9.83)  //RO_CLEAR_AIR_FACTOR=(Sensor resistance
in clean air)/RO,

                                     //which is derived from the chart in datasheet



#define    CALIBARAION_SAMPLE_TIMES  (50)    //define how many samples you are going
to take in the calibration phase

#define    CALIBRATION_SAMPLE_INTERVAL  (500)   //define the time interal(in milisecond)
between each samples in the

                                     //cablibration phase

#define    READ_SAMPLE_INTERVAL      (50)    //define how many samples you are going to
take in normal operation

#define    READ_SAMPLE_TIMES         (5)     //define the time interal(in milisecond) between
each samples in

                                     //normal operation



#define    GAS_LPG                   (0)

#define    GAS_CO                    (1)

#define    GAS_SMOKE                 (2)



float      LPGCurve[3]  =  {2.3,0.21,-0.47};   //two points are taken from the curve.
```

```
                                    //with these two points, a line is formed which is "approximately
equivalent"
                                    //to the original curve.
                                    //data format:{ x, y, slope}; point1: (lg200, 0.21), point2: (lg10000,
-0.59)
float         COCurve[3]  =  {2.3,0.72,-0.34};    //two points are taken from the curve.
                                    //with these two points, a line is formed which is "approximately
equivalent"
                                    //to the original curve.
                                    //data format:{ x, y, slope}; point1: (lg200, 0.72), point2: (lg10000,
0.15)
float         SmokeCurve[3] ={2.3,0.53,-0.44};    //two points are taken from the curve.
                                    //with these two points, a line is formed which is "approximately
equivalent"
                                    //to the original curve.
                                    //data format:{ x, y, slope}; point1: (lg200, 0.53), point2: (lg10000,
-0.22)
float         Ro         =  10;              //Ro is initialized to 10 kilo ohms


void setup()
{
  Serial.begin(9600);                       //UART setup, baudrate = 9600bps
  Serial.print("Calibrating...\n");
  Ro = MQCalibration(MQ_PIN);                   //Calibrating the sensor. Please make sure the
sensor is in clean air
                                    //when you perform the calibration
  Serial.print("Calibration is done...\n");
  Serial.print("Ro=");
```

```
  Serial.print(Ro);
  Serial.print("kohm");
  Serial.print("\n");
}

void loop()
{
   Serial.print("LPG:");
   Serial.print(MQGetGasPercentage(MQRead(MQ_PIN)/Ro,GAS_LPG) );
   Serial.print( "ppm" );
   Serial.print("    ");
   Serial.print("CO:");
   Serial.print(MQGetGasPercentage(MQRead(MQ_PIN)/Ro,GAS_CO) );
   Serial.print( "ppm" );
   Serial.print("    ");
   Serial.print("SMOKE:");
   Serial.print(MQGetGasPercentage(MQRead(MQ_PIN)/Ro,GAS_SMOKE) );
   Serial.print( "ppm" );
   Serial.print("\n");
   delay(200);
}



float MQResistanceCalculation(int raw_adc)
{
  return ( ((float)RL_VALUE*(1023-raw_adc)/raw_adc));
```

```
}



float MQCalibration(int mq_pin)
{
  int i;
  float val=0;

  for (i=0;i<CALIBARAION_SAMPLE_TIMES;i++) {           //take multiple samples
    val += MQResistanceCalculation(analogRead(mq_pin));
    delay(CALIBRATION_SAMPLE_INTERVAL);
  }
  val = val/CALIBARAION_SAMPLE_TIMES;                  //calculate the average value

  val = val/RO_CLEAN_AIR_FACTOR;                       //divided by RO_CLEAN_AIR_FACTOR yields
the Ro
                                                       //according to the chart in the datasheet

  return val;
}

float MQRead(int mq_pin)
{
  int i;
  float rs=0;
```

```
  for (i=0;i<READ_SAMPLE_TIMES;i++) {
    rs += MQResistanceCalculation(analogRead(mq_pin));
    delay(READ_SAMPLE_INTERVAL);
  }

  rs = rs/READ_SAMPLE_TIMES;

  return rs;
}

int MQGetGasPercentage(float rs_ro_ratio, int gas_id)
{
  if ( gas_id == GAS_LPG ) {
     return MQGetPercentage(rs_ro_ratio,LPGCurve);
  } else if ( gas_id == GAS_CO ) {
     return MQGetPercentage(rs_ro_ratio,COCurve);
  } else if ( gas_id == GAS_SMOKE ) {
     return MQGetPercentage(rs_ro_ratio,SmokeCurve);
  }

  return 0;
}

int  MQGetPercentage(float rs_ro_ratio, float *pcurve)
{
  return (pow(10,( ((log(rs_ro_ratio)-pcurve[1])/pcurve[2]) + pcurve[0])));
```

}

Output:

```
12:36:40.575 -> LPG:0ppm    CO:0ppm    SMOKE:0ppm
12:36:41.528 -> LPG:0ppm    CO:0ppm    SMOKE:0ppm
12:36:42.507 -> LPG:0ppm    CO:0ppm    SMOKE:0ppm
12:36:43.469 -> LPG:0ppm    CO:0ppm    SMOKE:11786ppm
12:36:44.416 -> LPG:3309ppm    CO:24432ppm    SMOKE:3802ppm
12:36:45.365 -> LPG:586ppm    CO:-2579ppm    SMOKE:7617ppm
12:36:46.324 -> LPG:1110ppm    CO:-20799ppm    SMOKE:868ppm
12:36:47.291 -> LPG:71ppm    CO:27621ppm    SMOKE:23486ppm
12:36:48.226 -> LPG:116ppm    CO:6101ppm    SMOKE:418ppm
12:36:49.205 -> LPG:100ppm    CO:1172ppm    SMOKE:53ppm
12:36:50.161 -> LPG:37ppm    CO:1685ppm    SMOKE:-24936ppm
12:36:51.105 -> LPG:3511ppm    CO:15622ppm    SMOKE:1102ppm
12:36:52.051 -> LPG:32ppm    CO:62ppm    SMOKE:321ppm
12:36:53.030 -> LPG:21071ppm    CO:-6340ppm    SMOKE:1893ppm
12:36:53.984 -> LPG:141ppm    CO:
```

Verified Screenshots: