

## IOT LAB CONSOLIDATED REPORT

USN:1BM18CS143

Name – Rishav Agrawal

Program No. – 01

Program Title – LED Blinking

---

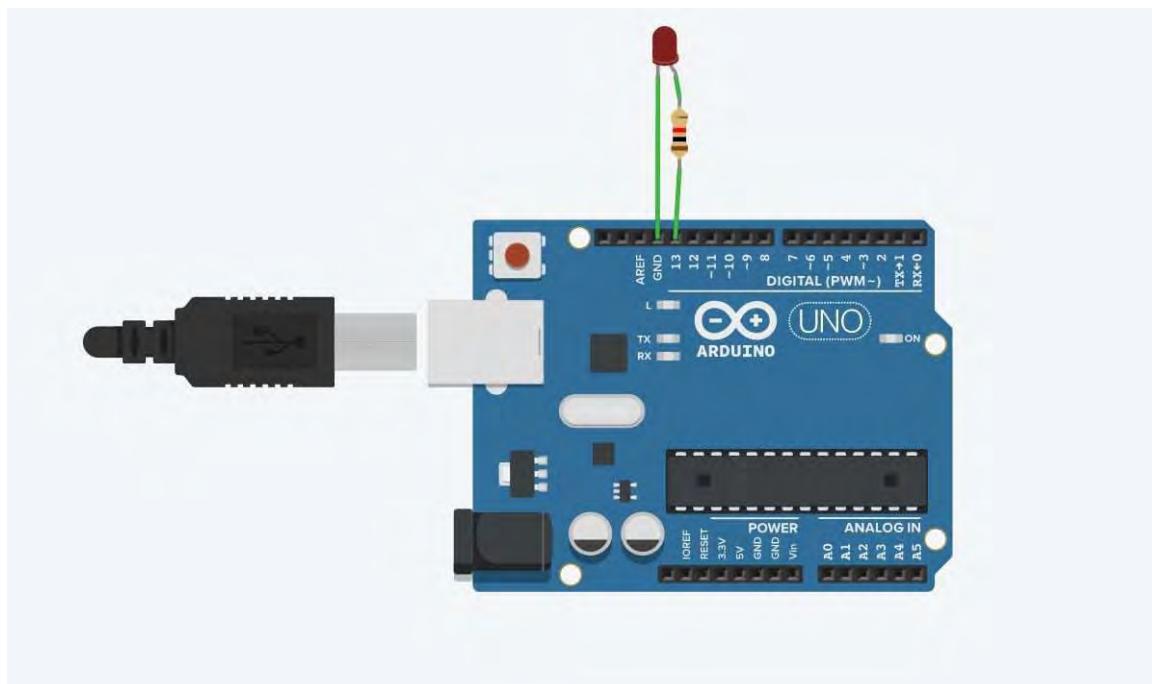
### AIM

Turn the LED on for a second, then off for a second, repeatedly.

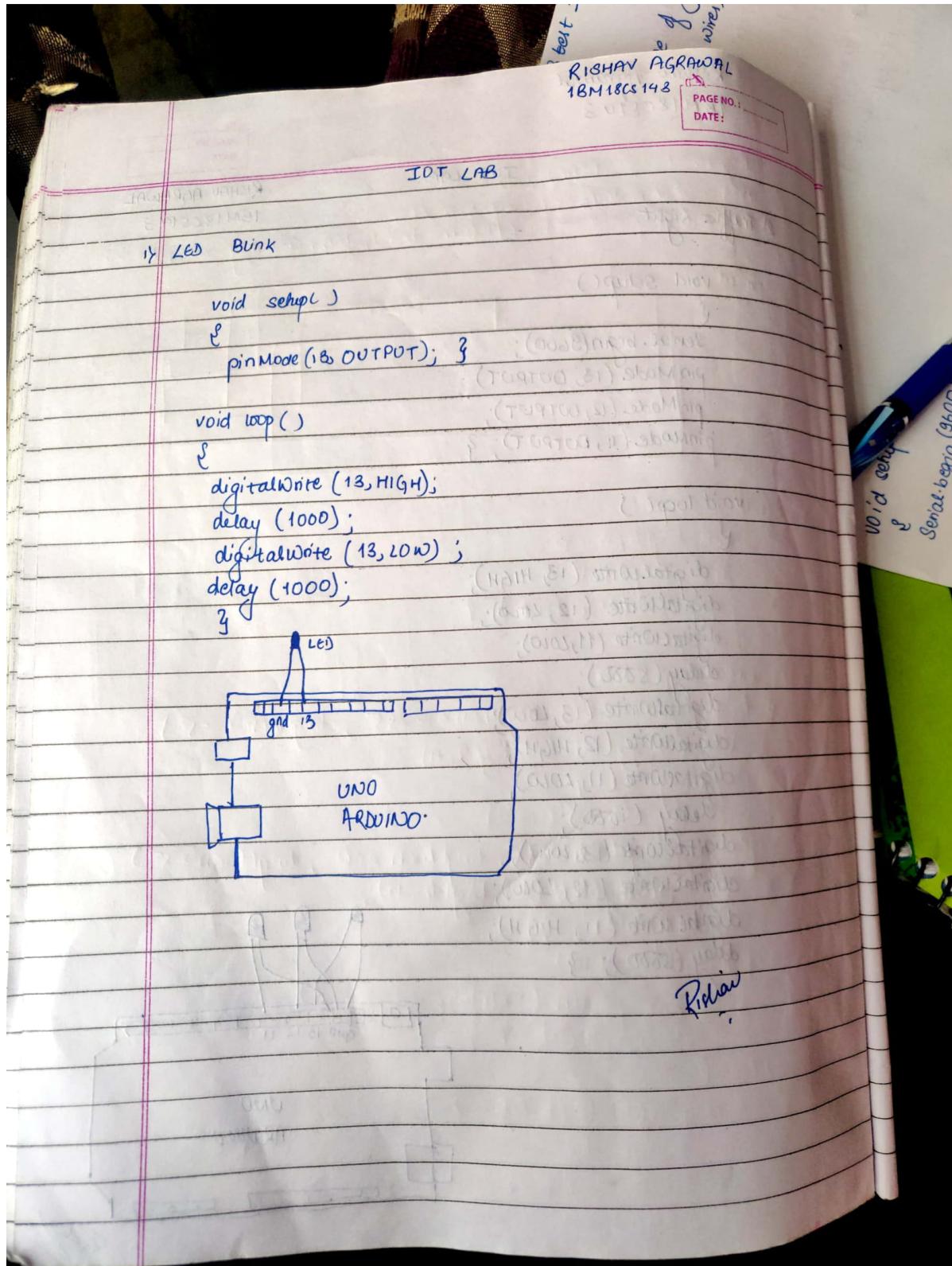
### HARDWARES REQUIRED

- Arduino Board
- LEDs

### CIRCUIT DIAGRAM



## WRITE-UP



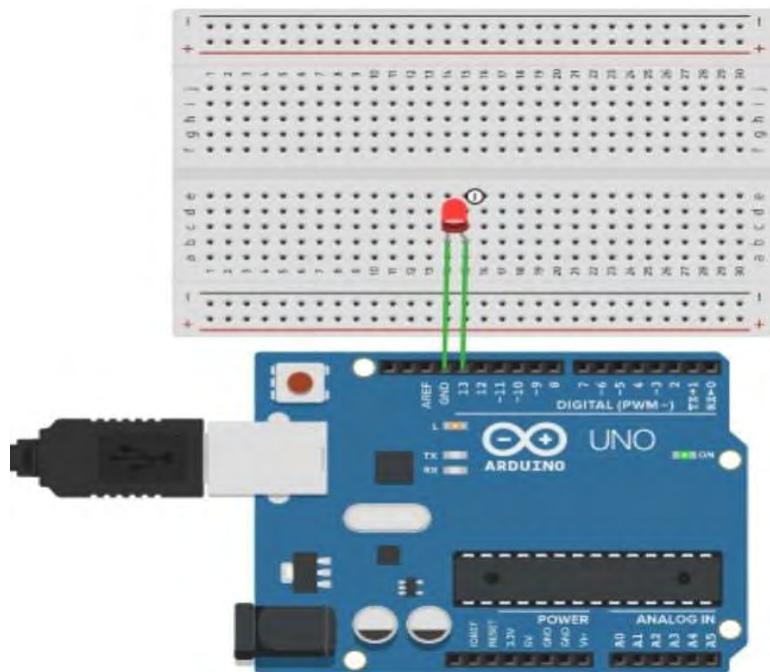
## CODE

```
void setup()
{
    pinMode(13, OUTPUT);
}

void loop()
{
    digitalWrite(13, HIGH);
    delay(1000); // Wait for 1000 millisecond(s)
    digitalWrite(13, LOW);
    delay(1000); // Wait for 1000 millisecond(s)
}
```

## OUTPUT

The LED was found to be blinking at an interval of 1000 ms.



Program No. – 02

Program Title – Traffic Controller

---

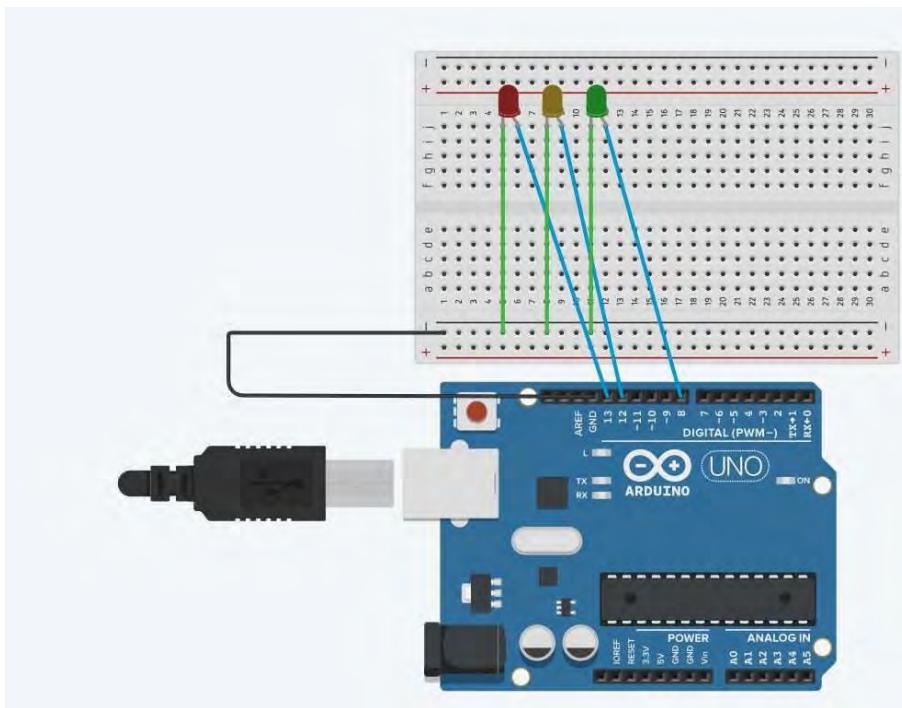
## AIM

Traffic Signal Simulator.

## HARDWARES REQUIRED

- Arduino Board
- LEDs
- Breadboard

## CIRCUIT DIAGRAM



## WRITE-UP

Rishav Agrawal  
18M18CS143

PAGE NO.: \_\_\_\_\_  
DATE: \_\_\_\_\_

IOT LAB

RISHAV AGRAWAL  
18M18CS143

2) Traffic light

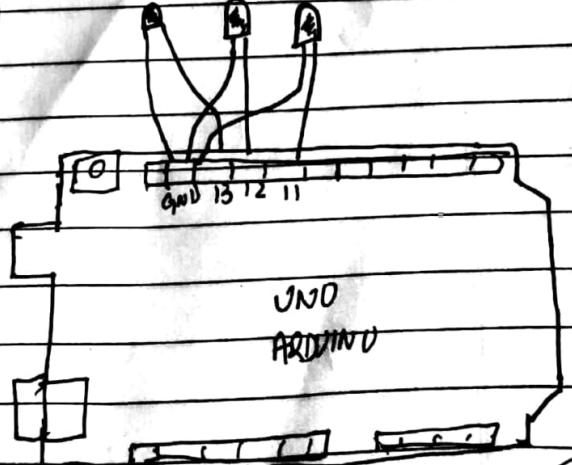
void setup()

{

```
Serial.begin(9600);  
pinMode(13, OUTPUT);  
pinMode(12, OUTPUT);  
pinMode(11, OUTPUT); }
```

void loop()

```
digitalWrite(13, HIGH);  
digitalWrite(12, LOW);  
digitalWrite(11, LOW);  
delay(5000);  
digitalWrite(13, LOW);  
digitalWrite(12, HIGH);  
digitalWrite(11, LOW);  
delay(1000);  
digitalWrite(13, LOW);  
digitalWrite(12, LOW);  
digitalWrite(11, HIGH);  
delay(5000); }
```



## CODE

```
void setup()
{
    pinMode(13, OUTPUT);
    pinMode(12,OUTPUT);
    pinMode(8,OUTPUT);

}

void red()
{
    digitalWrite(13, HIGH);
    digitalWrite(12,LOW);
```

```
    digitalWrite(8,LOW);  
}  
  
void yellow()
```

```
{  
    digitalWrite(13, LOW);  
    digitalWrite(12,HIGH);  
    digitalWrite(8,LOW);  
}
```

```
void green()  
{  
    digitalWrite(13, LOW);  
    digitalWrite(12,LOW);  
    digitalWrite(8,HIGH);  
}
```

```
void loop()  
{  
    red();  
    delay(3000);  
    yellow();  
    delay(1500);  
    green();  
    delay(3000);
```

```
yellow();  
delay(1500);  
}
```

## OUTPUT

All the three LEDs blink one after the other at an interval of 1000ms.

Program No. – 03

Program Title – LED fading without potentiometer

---

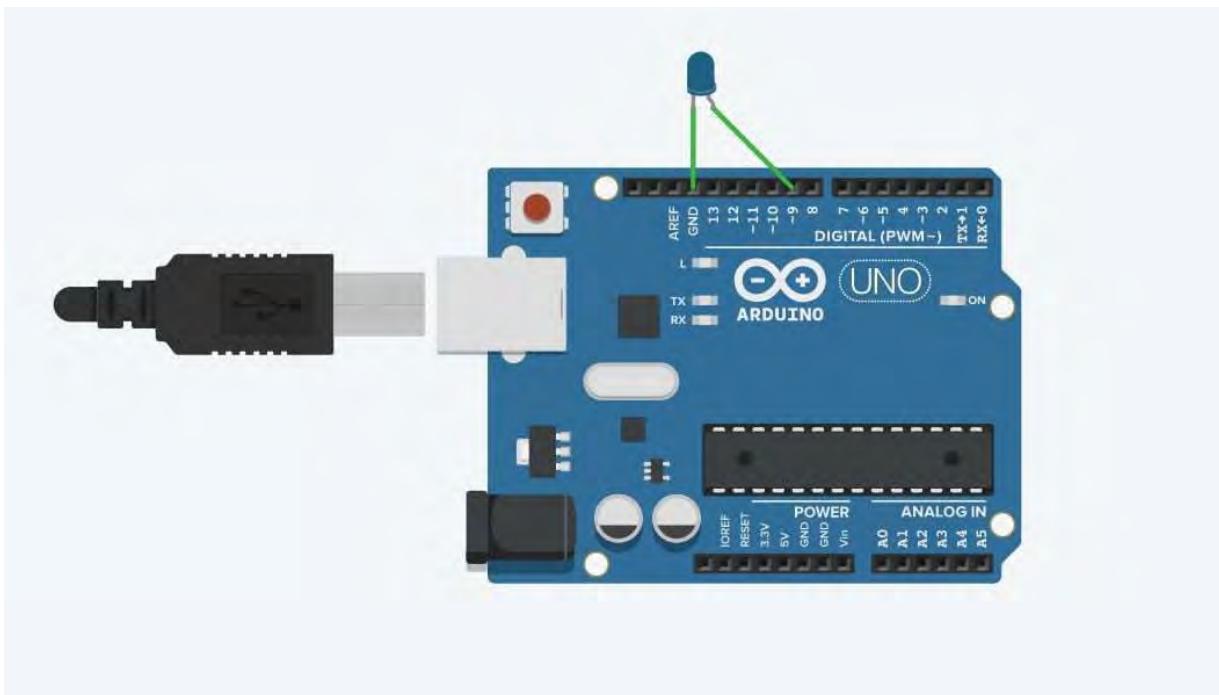
## AIM

Demonstrate to show LED fading.

## HARDWARES REQUIRED

- Arduino Board
- LED bulb

## CIRCUIT DIAGRAM



## WRITE-UP

Rishav Agrawal  
18N18CS143

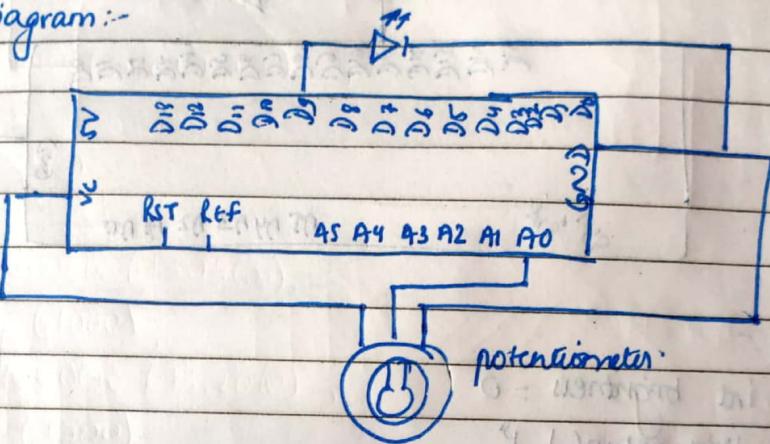
PAGE NO.:  
DATE:

Program : 3 - fading LED using Potentiometer

Aim :- Demonstrate a fading LED (analog Output)

Hardware Requirements :- Arduino board, LED, connecting wire, resistor,

CIRCUIT Diagram:-



CODE :-

```
void setup ()  
{ serial.begin (9600)  
pinMode (9, OUTPUT); }  
  
void loop ()  
{ int analog = analogRead (A0);  
int brightness = map (analog, 0, 1023, 0, 255);  
analogWrite (9, brightness);  
Serial.print ("\n Analog Values :");  
Serial.print (analog);  
Serial.print ("\n Brightness value");  
Serial.print (brightness); }
```

Rishav

## CODE

```
void setup()
{
    pinMode(2, OUTPUT);
}

void loop()
{
    for(int fade =0;fade <=255; fade+=5)
    {
        analogWrite(9,fade);
        delay(30);
    }
    for(int fade = 255; fade>=0;fade-=5)
    {
        analogWrite(9, fade);
        delay(30);
    }
}
```

}

## **OUTPUT**

Fading of LED.

**Name – Rishav Agrawal**

Program No. – 04

Program Title – LED fading with potentiometer

---

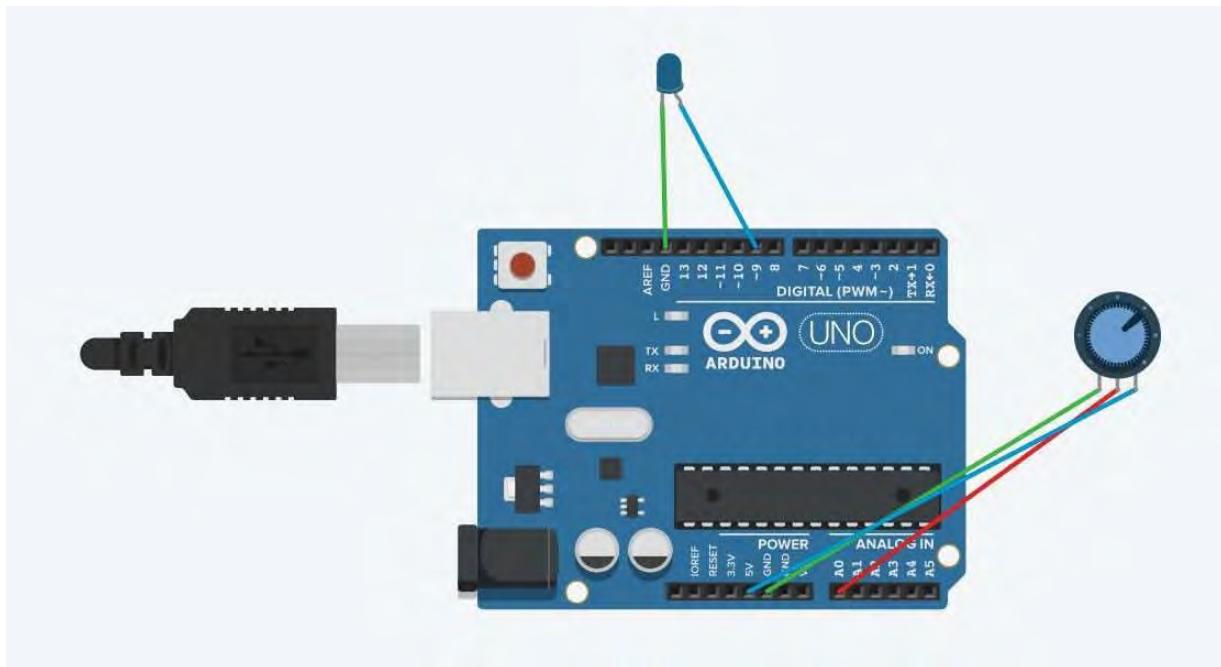
## **AIM**

Demonstrate to show LED fading(analog output).

## **HARDWARES REQUIRED**

- Arduino Board
- LED bulb
- Potentiometer

## **CIRCUIT DIAGRAM**



## WRITE-UP

Rishav Agrawal  
1B M18CS143

PAGE NO.: \_\_\_\_\_  
DATE: \_\_\_\_\_

Program: 4 :- fading Led

Aim: Demonstrate fading of a Led

Hardware Required:- (Raspberry board, Led, connecting wire, resistor)

Circuit :-



Code :-

```
int brightness = 0  
void setup() {  
    pinMode(11, output); }
```

```
void loop()  
{ for (brightness = 0; brightness <= 255; brightness += 5)  
    { analogWrite(11, brightness);  
    delay(30); } }
```

```
for (brightness = 255; brightness >= 0; brightness -= 5)  
{ analogWrite(11, brightness);  
delay(30); }
```

Rishav

## CODE

```
int LED_PIN = 9;

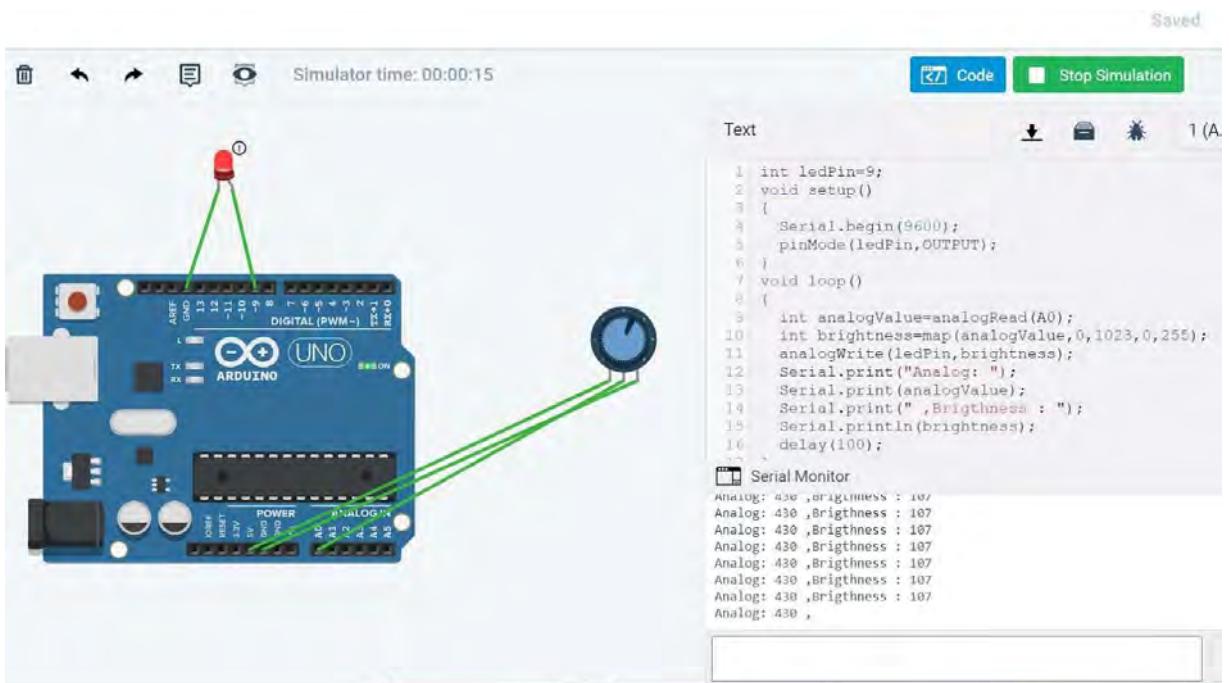
void setup()
{
    Serial.begin(9600);
    pinMode(LED_PIN, OUTPUT);
}

void loop()
{
    int analogValue = analogRead(A0);
    int brightness = map(analogValue, 0, 1023, 0, 255);
    analogWrite(LED_PIN, brightness);
    Serial.print("Analog: ");
    Serial.print(analogValue);
    Serial.print(", Brightness : ");
    Serial.println(brightness);
```

```
delay(100);  
}
```

## OUTPUT

## Fading of LED with potentiometer.



**Name – Rishav Agrawal**

Program No. – 05

Program Title – ON/OFF LED using Push Button

---

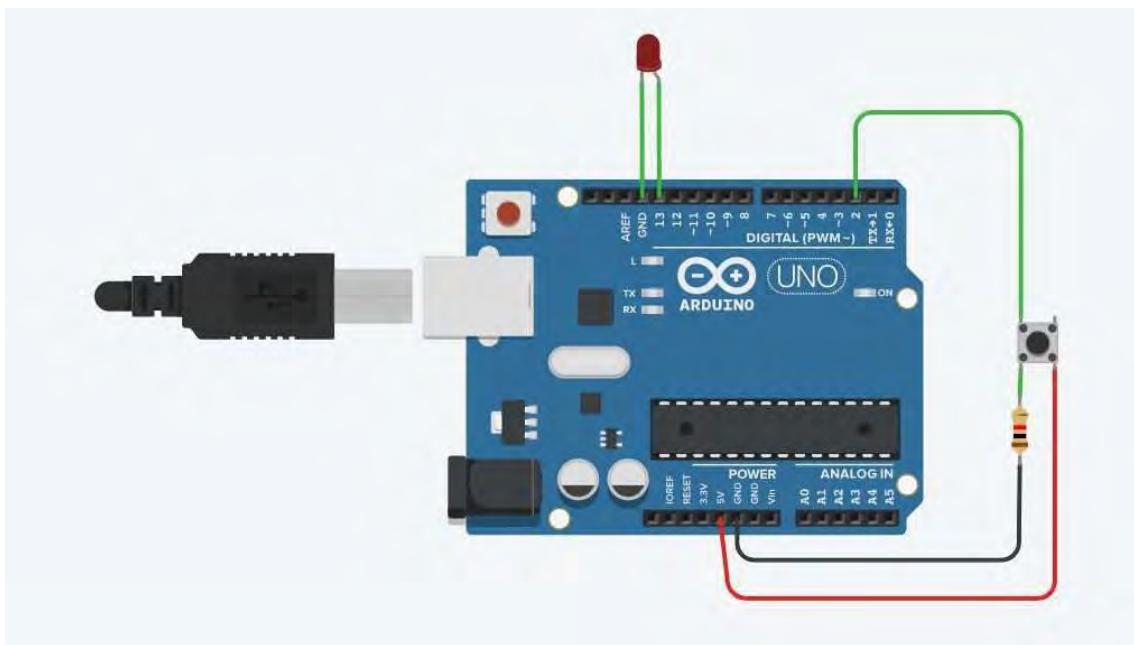
## **AIM**

Demonstrate to show ON/OFF of a LED using push button (Digital Output).

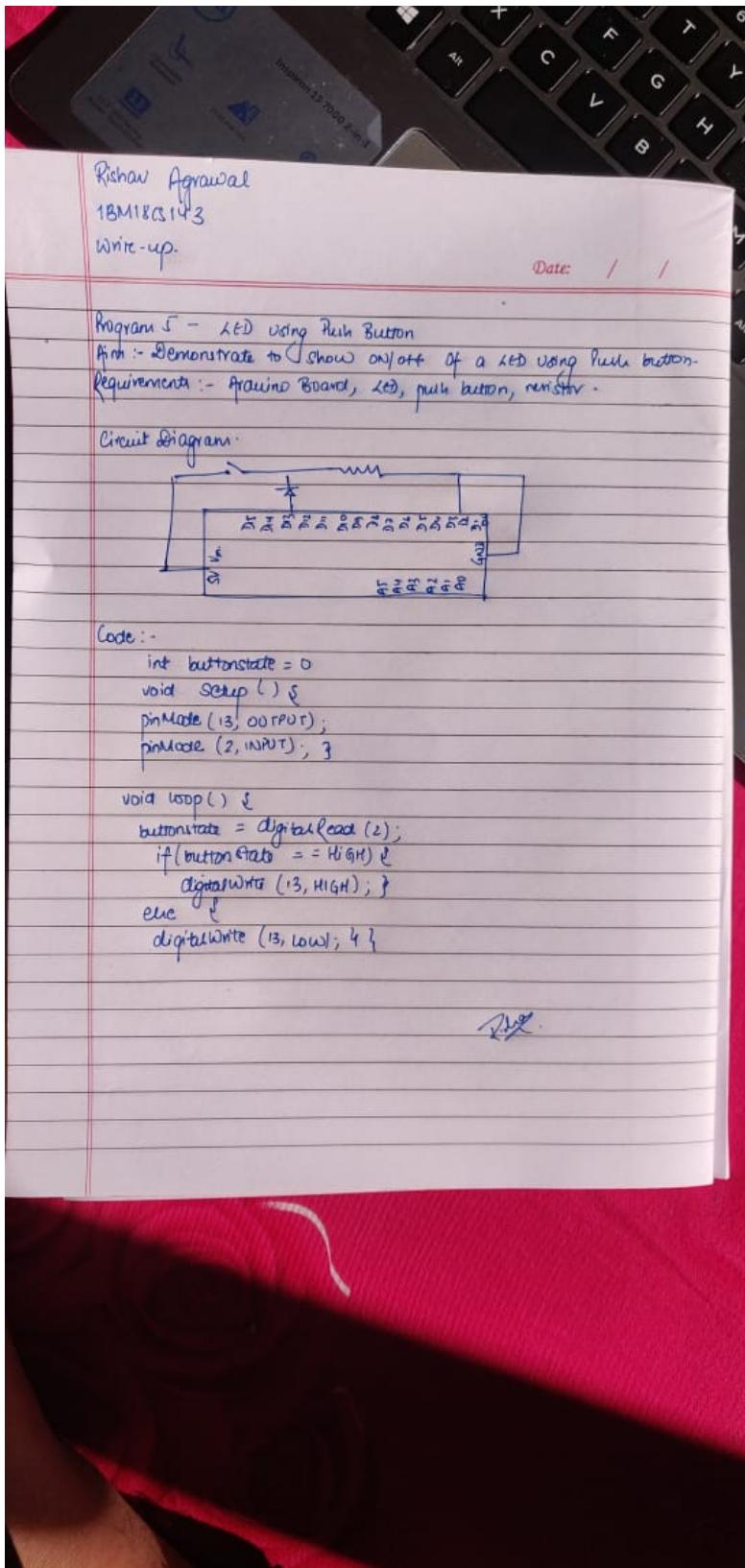
## **HARDWARES REQUIRED**

- Arduino Board
- LED bulb
- Push Button
- Resistor

## **CIRCUIT DIAGRAM**



## WRITE-UP



## **CODE**

```
int buttonstate=0;  
  
void setup()  
{  
    pinMode(13, OUTPUT);  
    pinMode(2, OUTPUT);  
  
}  
  
void loop()  
{  
    buttonstate=digitalRead(2);  
    if(buttonstate == HIGH)  
    {digitalWrite(13,HIGH);}  
    else  
    {digitalWrite(13,LOW);}  
}
```

## **OUTPUT**

ON/OFF of a LED using push button (Digital Output).

**Name – Rishav Agrawal**

Program No. – 06

Program Title – LDR

---

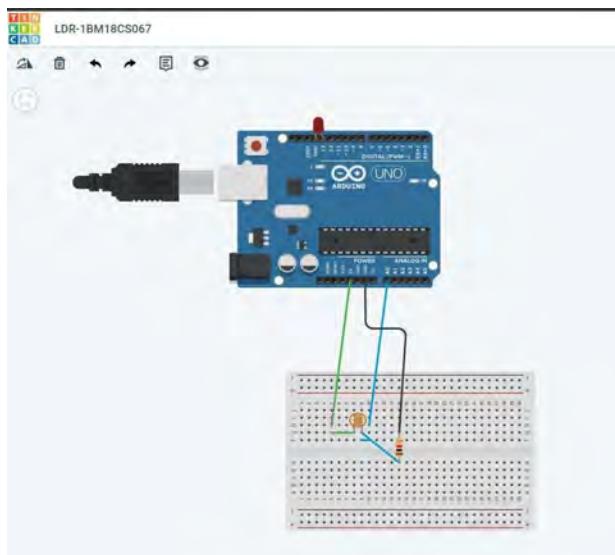
## **AIM**

Demonstrate to show on/off of a LED using LDR night light simulation.

## **HARDWARES REQUIRED**

- Arduino Board
- Photoresistor
- Resistor
- LED
- Breadboard Small

## **CIRCUIT DIAGRAM**



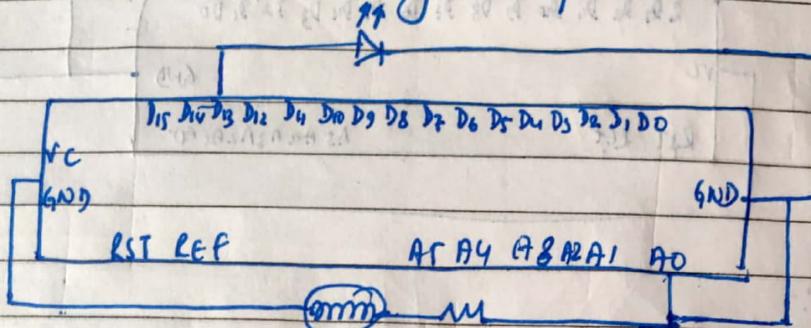
## WRITE-UP

Rishav Agrawal  
1BM18CS143

PAGE NO.:  
DATE:

Program :- 6 ON/OFF LED USING LDR.  
Aim :- Demonstrate to show ON/OFF of a LED using LDR - Nightlight Simulator

Hardware Requirements :- Arduino board LED, breadboard, resistor, connecting wire, photodiode.



Code :-

```
const int ledPin = 8;  
const int ldrPin = A0;  
void setup()  
{  
  Serial.begin(9600);  
  pinMode(ledPin, OUTPUT);  
  pinMode(ldrPin, INPUT);  
}
```

```
void loop()  
{  
  int ldrStatus = analogRead(ldrPin);  
  Serial.println(ldrStatus);  
  if (ldrStatus <= 10)  
  {  
    digitalWrite(ledPin, HIGH);  
    Serial.println("LDR is dark, LED is ON");  
  }  
  else
```

```
{  
  digitalWrite(ledPin, LOW);  
  Serial.println("LDR is bright");  
}
```

## CODE

```
const int ledPin = 13;

const int ldrPin = A0;

void setup()
{
    Serial.begin(9600);
    pinMode(ledPin, OUTPUT);
    pinMode(ldrPin, INPUT);
}

void loop()
{
    int ldrStatus =analogRead(ldrPin);
    Serial.println(ldrStatus);
    if(ldrStatus <=10)
    {
```

```

        digitalWrite(ledPin, HIGH);
        Serial.println("LDR is DARK, LED is ON");

    }

else

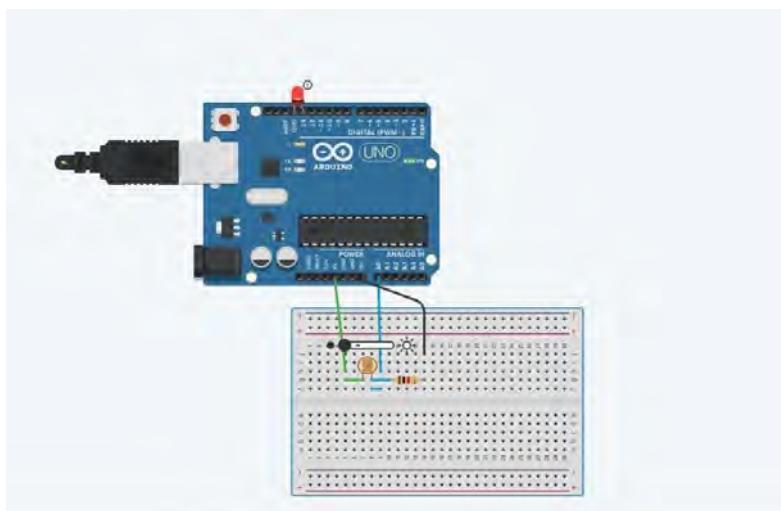
{
    digitalWrite(ledPin, LOW);
    Serial.println(" -----");
}

}

```

## OUTPUT

Design a system to show on/off of a LED using LDR night light simulation.



**Name – Rishav Agrawal**

Program No. – 07

Program Title – PIR

---

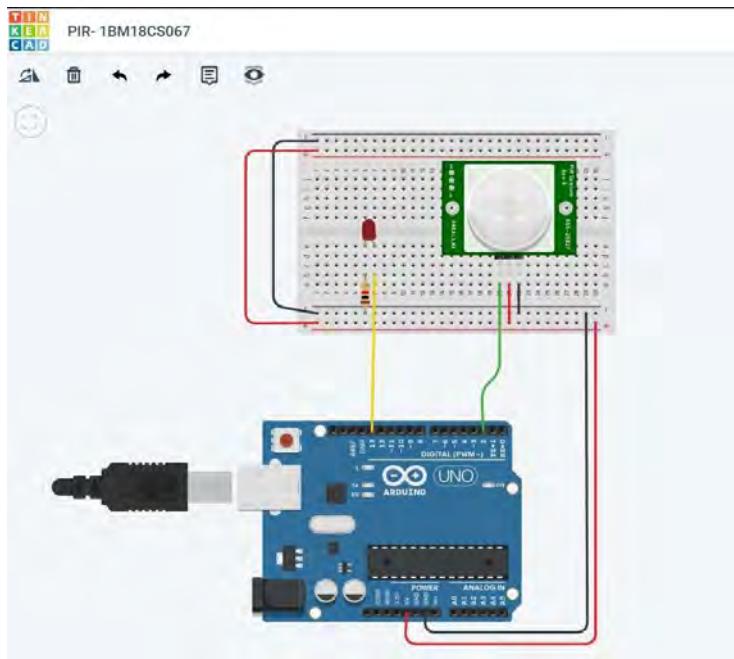
## **AIM**

Demonstrate to show working of PIR sensor.

## **HARDWARES REQUIRED**

- Arduino Board
- PIR sensor
- Resistor
- LED pin
- Breadboard Small

## **CIRCUIT DIAGRAM**



## WRITE-UP

Rishav Agrawal  
18M18CS143

PAGE NO.: \_\_\_\_\_  
DATE: \_\_\_\_\_

Program 7 :- PIR sensor.

Circuit Diagram :-

Code :-

```
int led = 13;
int sensor = 1;
int state = LOW;
int val = 0;

void setup()
{
    pinMode(led, OUTPUT);
    pinMode(sensor, INPUT);
    Serial.begin(9600);
}

void loop()
{
    val = digitalRead(sensor);
    if (val == HIGH)
    {
        digitalWrite(led, HIGH);
        delay(10);
        if (state == LOW)
        {
            Serial.println("Motion detected!");
            state = HIGH;
        }
    }
    else
    {
        digitalWrite(led, LOW);
        delay(10);
        if (state == HIGH)
        {
            Serial.println("Motion stopped");
            state = LOW;
        }
    }
}
```

Rishav Agrawal

## CODE

```
int sensorState = 0;

void setup()
{
    pinMode(2, INPUT);
    pinMode(13, OUTPUT);
    Serial.begin(9600);
}

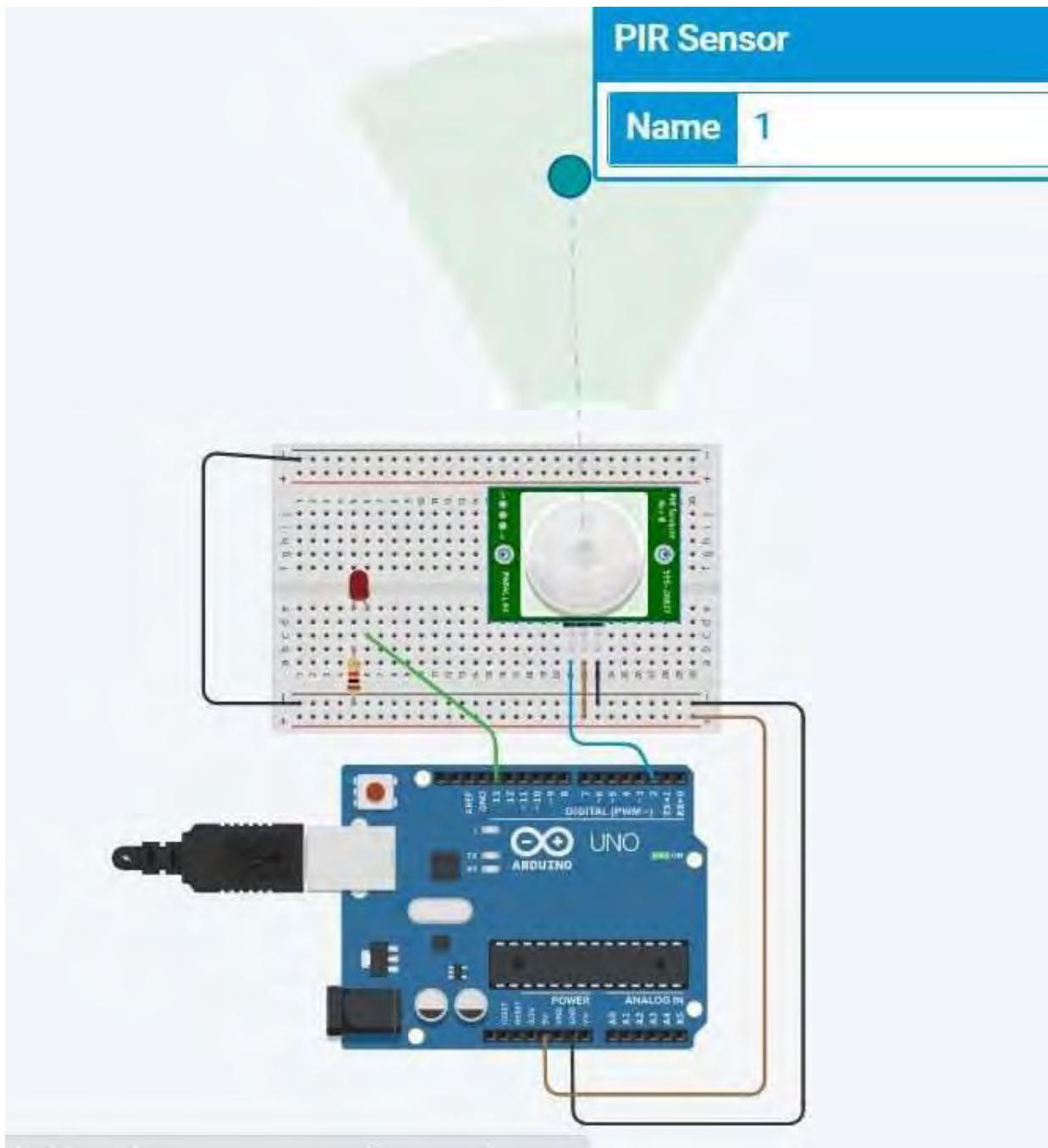
void loop()
{
    sensorState = digitalRead(2);

    if (sensorState == HIGH) {
        digitalWrite(13, HIGH);
        Serial.println("Sensor activated!");
    } else {
        digitalWrite(13, LOW);
        Serial.println("Sensor deactivated!");
    }
}
```

```
delay(5); }
```

## OUTPUT

Designed a system to show working of PIR Sensor.



Name – Rishav Agrawal

Program No. – 08

Program Title – Distance Measurement using ultrasonic sensor

---

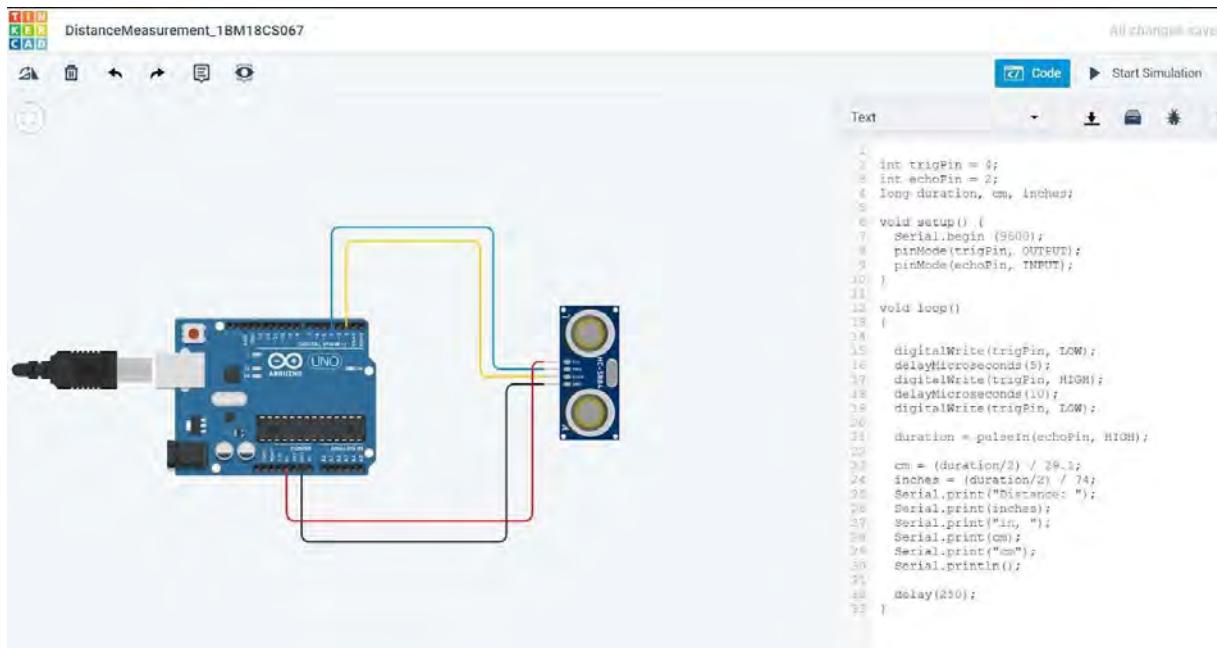
## AIM

Design a system to measure the distance between objects.

## HARDWARES REQUIRED

- Arduino Board
- Ultrasonic sensor HC-SR04

## CIRCUIT DIAGRAM



## WRITE-UP

Rishav Agrawal  
1BM18CS143

Date: / /

Program 9 :- Ultrasonic Range Finder.

Aim :- Design a system to measure distance between object.

Hardware Requirements :- Arduino Uno, HC-SR04.

Circuit :-

Code :-

```
int trigPin = 13;
int echoPin = 10;

void setup() {
    Serial.begin(9600);
    pinMode(trigPin, OUTPUT);
    pinMode(4, OUTPUT);
    pinMode(2, OUTPUT);
    pinMode(echoPin, INPUT);
}

void loop() {
    float duration, distance;
    digitalWrite(trigPin, LOW);
    delayMicroseconds(2);
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigPin, LOW);
    duration = pulseIn(echoPin, HIGH);
    distance = (duration / 2) * 0.0344;
}
```

Date:

```
if (distance >= 200 || distance <= 2) {  
    Serial.print ("Distance =");  
    Serial.println ("Out of Range");  
    digitalWrite (2, HIGH);  
    digitalWrite (4, LOW); }  
  
else {  
    Serial.print ("Distance =");  
    Serial.print (distance);  
    Serial.println ("");  
    delay (500);  
    digitalWrite (4, HIGH);  
    digitalWrite (2, LOW); }  
delay (100); }
```

## CODE

```
int trigPin = 4;  
int echoPin = 2;  
long duration, cm, inches;  
  
void setup() {  
    Serial.begin (9600);  
  
    pinMode(trigPin, OUTPUT);  
  
    pinMode(echoPin, INPUT);  
}  
  
void loop()  
{
```

```
digitalWrite(trigPin, LOW);
delayMicroseconds(5);
digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);

duration = pulseIn(echoPin, HIGH);
cm = (duration/2) / 29.1;
inches = (duration/2) / 74;
Serial.print("Distance: ");
Serial.print(inches);
Serial.print("inch, ");
Serial.print(cm);
Serial.print("cm");
Serial.println();
delay(250);

}
```

## OUTPUT

Design a system to measure the distance between objects using ultrasonic device.

B Week4

C Circ u it design obJec t dista nce

iJD!!! Circ u it design Amazing Fyra n!!

131) object distance

131(j)

Simulator time: 00:00:07

**fjh++@FHII**

Text

```

int trigPin = 4;
int echoPin = 2;
long duration, cm inches;

void setup() {
  Serial.begin(9600);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
}

void loop()
  
```

P Type here to search

Send Clear

15:53  
07-10-2020

\* 3

```

1.4 digitalWrite(trigPin, LOW);
1.5 delayMicroseconds(5);
1.6 digitalWrite(trigPin, HIGH);

Lii Serial Monitor
ULtrasonic<:cJLJL>, L:J:BL
Distance 52inches, 133cm
Distance 51inches, 133cm
Distance 52inches, 133cm
Distance 52inches, 133cm
Distance 51inches, 133cm
Distance 52inches, 133cm
Distance 52inches, 133cm

```

**Name - Rishav Agrawal**

Program No. – 09

Program Title – Fire Alarm using flame Sensor

---

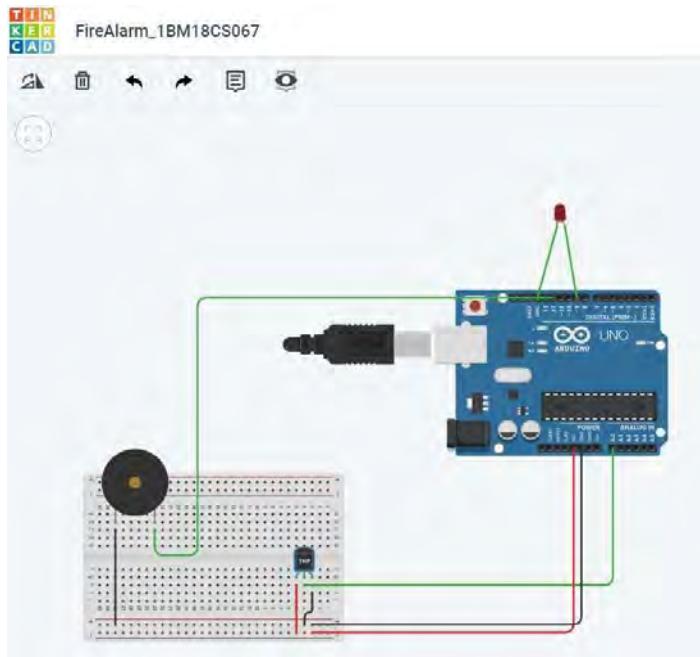
## **AIM**

Design an alert system using a flame sensor.

## **HARDWARES REQUIRED**

- Arduino Board
- Piezo
- Temperature Sensor
- Breadboard small

## **CIRCUIT DIAGRAM**



## WRITE-UP

Rishav Agrawal  
18M1PCL143

Date: / /

Program 8 :- Fire Alert System.

Aim :- Design an alert system using flame sensor.

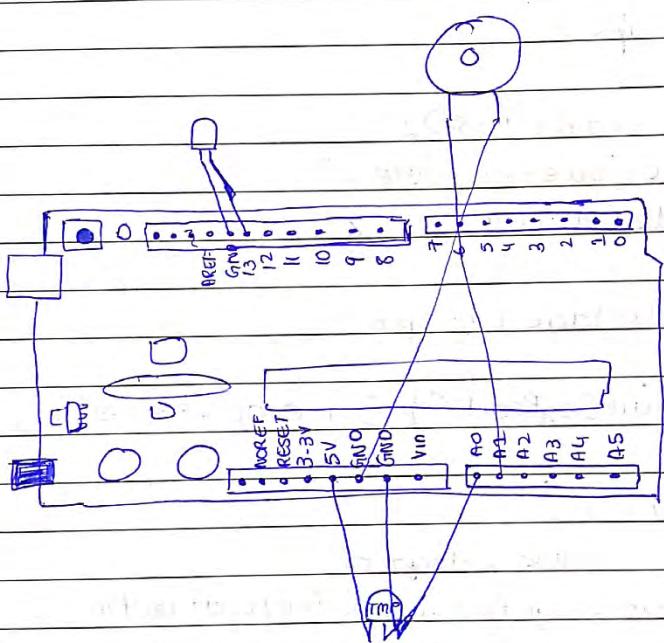
Requirement :- Arduino Uno, temperature sensor, LED

Code:-

```
int temperaturePin = 0;  
int buzzer = 12;  
void setup() {  
    Serial.begin (9600);  
    pinMode (buzzer, OUTPUT);  
    pinMode (9, OUTPUT);  
  
    float getVoltage (int pin) {  
        return (analogRead (pin) + 0.004882.814);  
  
    void loop () {  
        float voltage, degreeC;  
        voltage = getVoltage (temperaturePin);  
        degreeC = (voltage - 0.5) * 100;  
        digitalWrite (9, LOW);  
        if (degreeC > 37) {  
            Serial.print (degreeC);  
            Serial.print (" Alert !");  
            digitalWrite (buzzer, HIGH);  
            //digitalWrite (9, HIGH);  
            tone (12, 1000, 100);  
            delay (200);  
        }  
        else {  
            Serial.print ("degree");  
            Serial.print ("Save!");  
            delay (200);  
        }  
    }  
}
```

WZ.

Diagram :



## CODE

```
const int temperaturePin = 0;  
int buzzer = 12;  
  
void setup()  
{  
    Serial.begin (9600);  
    pinMode(buzzer, OUTPUT);  
    pinMode(9, OUTPUT);  
}  
  
void loop()  
{  
    float voltage, degreesC;  
    voltage = getVoltage(temperaturePin);  
    degreesC = (voltage-0.5)*100.0;  
  
    if(degreesC < 37)
```

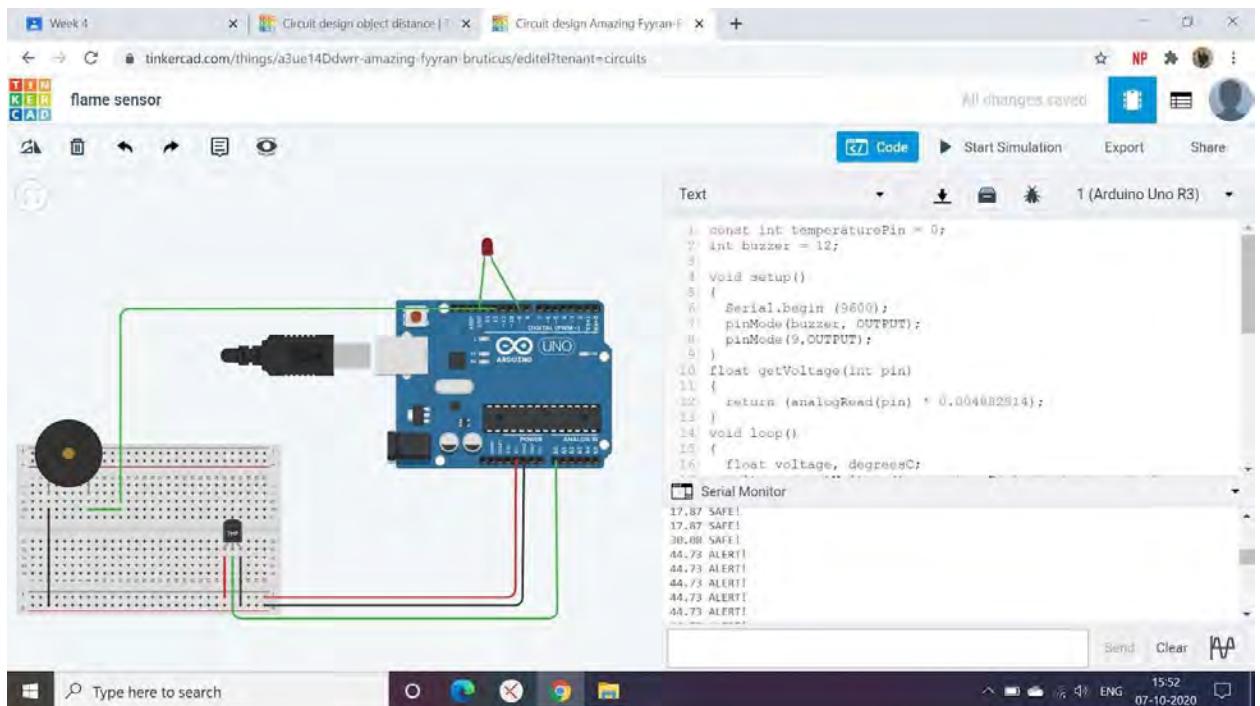
```
{  
    Serial.print(degreesC);  
    Serial.println(" SAFE!");  
}  
  
}
```

```
if(degreesC > 37)  
{  
    Serial.print(degreesC);  
    Serial.println("FIRE !!!");  
    digitalWrite(9, HIGH);  
    digitalWrite(buzzer, LOW);  
    tone(12, 10000,100);  
    delay(100);  
}  
  
}
```

```
float getVoltage(int pin)  
{  
    return (analogRead(pin) * 0.004882814);  
}
```

## OUTPUT

Designed an alert system using flame sensor.



**Name – Rishav Agrawal**

Program No. – 10

Program Title – Gas Sensor

---

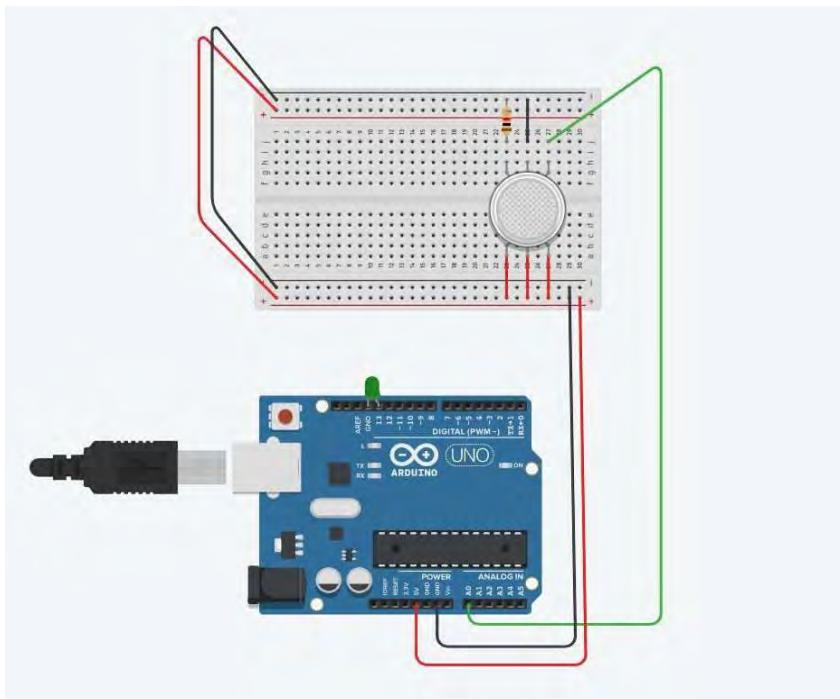
## **AIM**

To design a smart gas leakage indicator system.

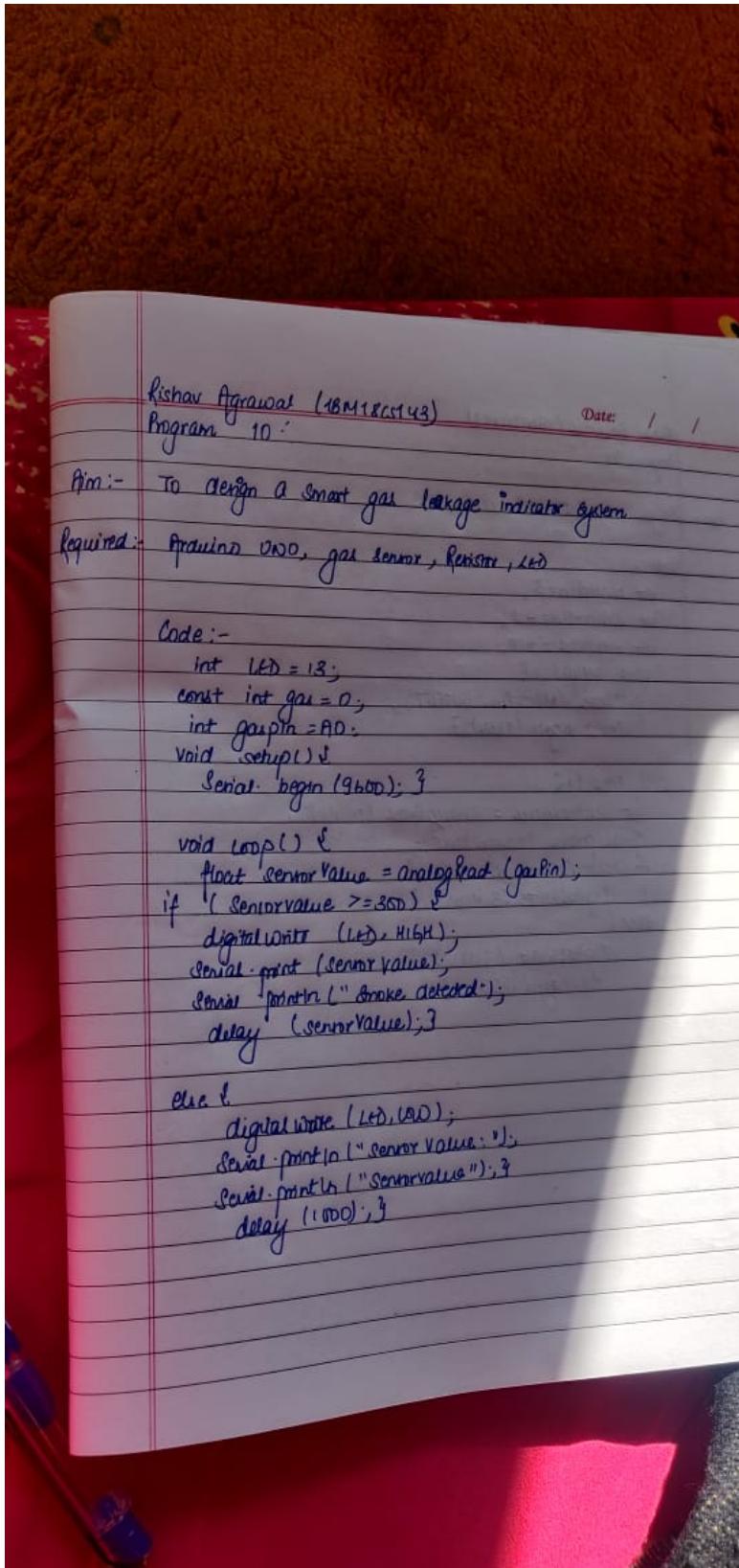
## **HARDWARES REQUIRED**

- Arduino Board
- Gas sensor
- Resistor
- LED
- Breadboard Small

## **CIRCUIT DIAGRAM**



## WRITE-UP



## CODE

```
int LED = 13;  
  
const int gas = 0;  
  
int Gaspin = A0;  
  
  
void setup()  
{  
    Serial.begin(9600);  
}  
  
void loop()  
{  
    float sensorValue = analogRead(Gaspin);  
  
    if(sensorValue >= 300)  
    {  
        digitalWrite(LED, HIGH);  
  
        Serial.print(sensorValue);  
  
        Serial.println(" *** SMOKE DETECTED ***");  
  
        delay(sensorValue);  
    }  
}
```

```

}

else

{

    digitalWrite(LED, LOW);

    Serial.println("Serial Value : ");

    Serial.println(sensorValue);

}

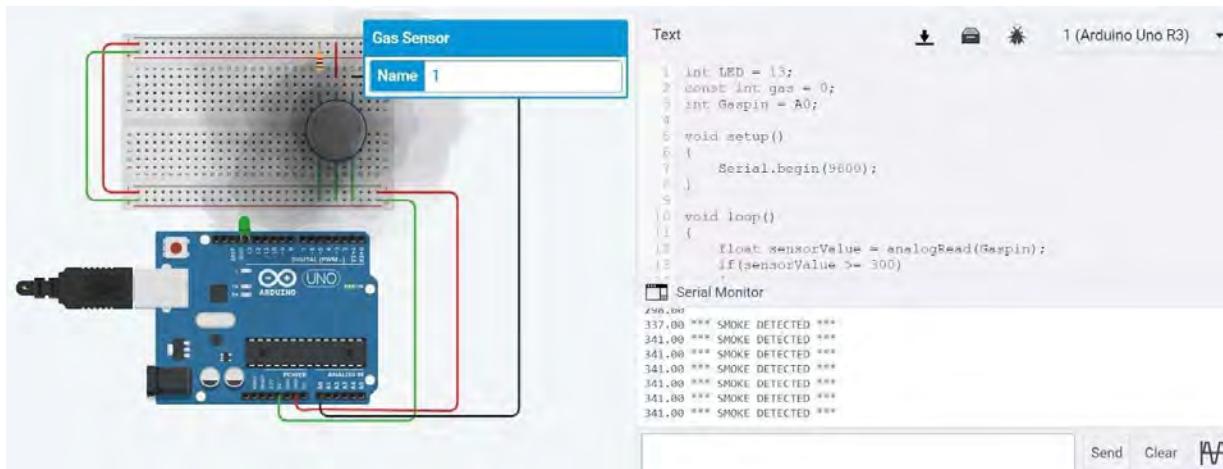
delay(1000);

}

```

## OUTPUT

Designed a smart gas leakage indicator system.



**Name – Rishav Agrawal**

Program No. – 11

Program Title – Vibration motor and LDR

---

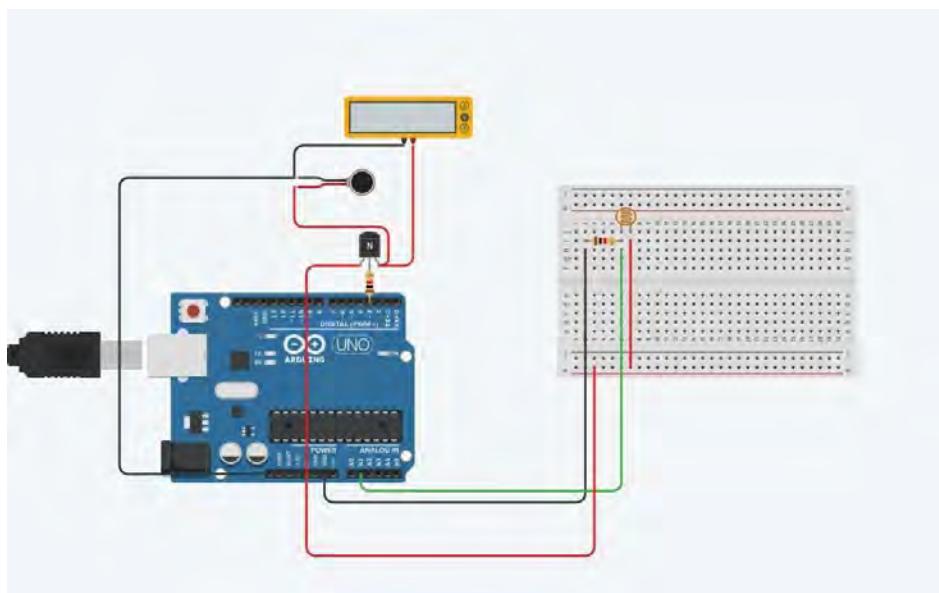
## **AIM**

To design an automated day indicator system.

## **HARDWARES REQUIRED**

- Arduino Board
- NPN Transistor
- Resistor
- Vibration motor
- Multimeter
- Photoresistor

## **CIRCUIT DIAGRAM**



## WRITE-UP

Date: / /

Rishav Agrawal (1BN18CL143)

Program 11:

Aim:- Design an automated motor IDR

Code:-

```
int motorPin = 3;  
int sensorPin = A1;  
int threshold = 400;  
void setup () {  
    pinMode (MotorPin, OUTPUT);  
    Serial.begin (9600);}
```

```
void loop () {  
    int sensorValue = analogRead (sensorPin);  
    Serial.println (sensorValue);  
    if (sensorValue > threshold) {  
        digitalWrite (3, HIGH);}  
    else {  
        digitalWrite (3, LOW);}  
    delay (1000);}
```

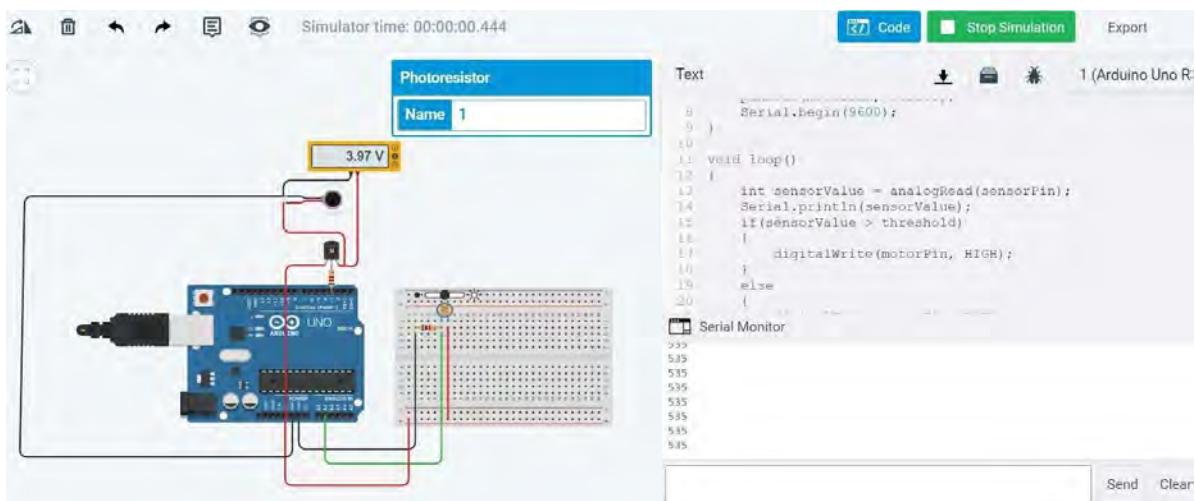
## CODE

```
int motorPin = 3;  
int sensorPin = A1;  
int threshold = 400;  
  
void setup()  
{  
    pinMode(motorPin, OUTPUT);  
    Serial.begin(9600);  
}  
  
void loop()  
{  
    int sensorValue = analogRead(sensorPin);  
    Serial.println(sensorValue);  
    if(sensorValue > threshold)  
    {  
        digitalWrite(motorPin, HIGH);  
    }  
}
```

```
    }  
  
    else  
  
    {  
  
        digitalWrite(motorPin, LOW);  
  
    }  
  
}
```

## OUTPUT

Designed an automated day indicator system.



**Name – Rishav Agrawal**

Program No. – 12

Program Title – Tilt Sensor

---

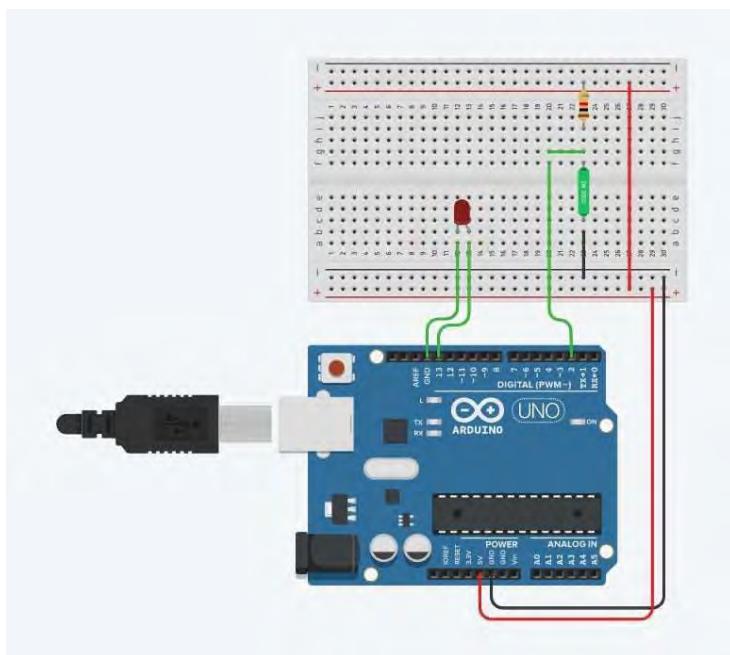
## **AIM**

Design a Smart Package handling system (Tilt sensor and LED)

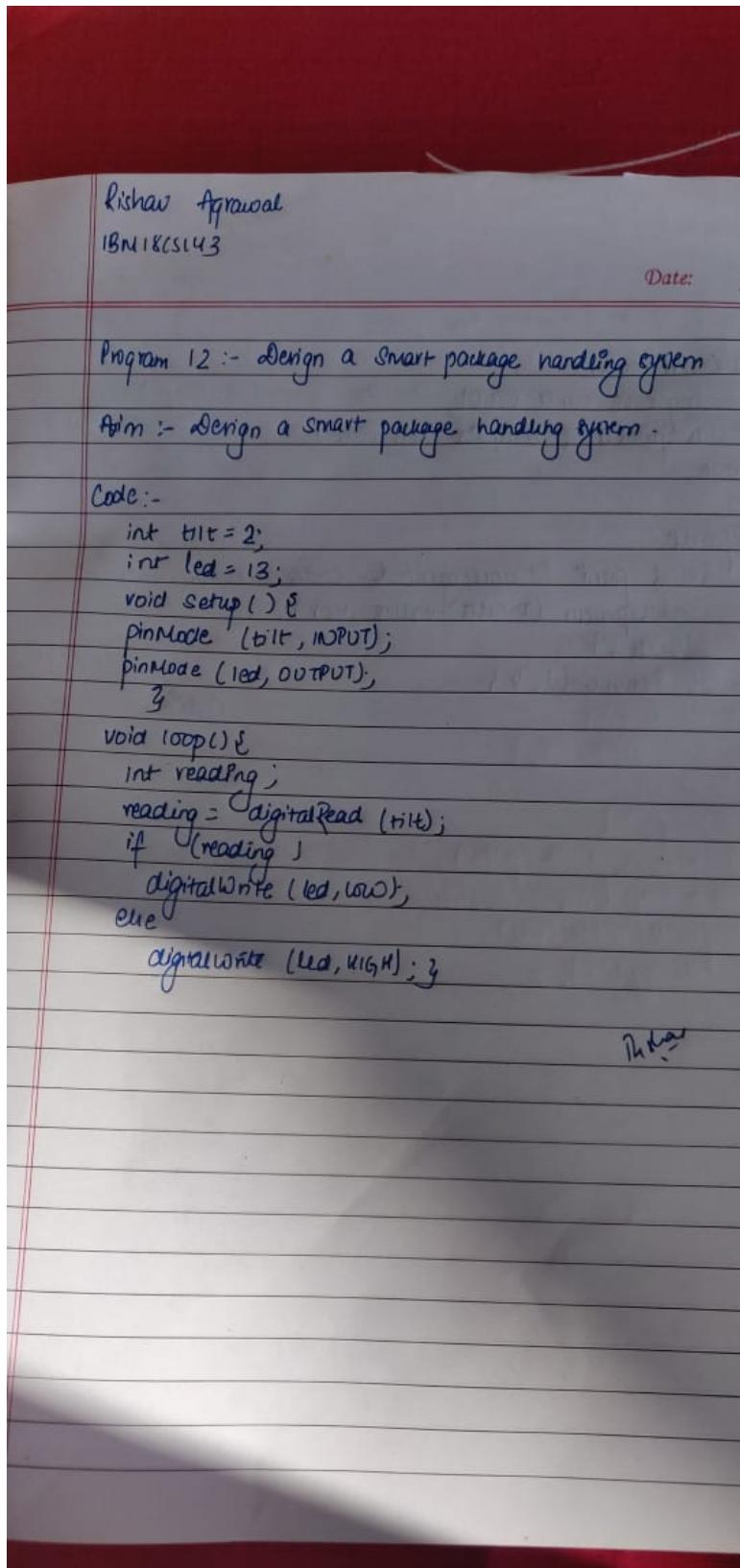
## **HARDWARES REQUIRED**

- Arduino Board
- Breadboard Small
- LEDs
- Tilt Sensor
- Resistor

## **CIRCUIT DIAGRAM**



## WRITE-UP

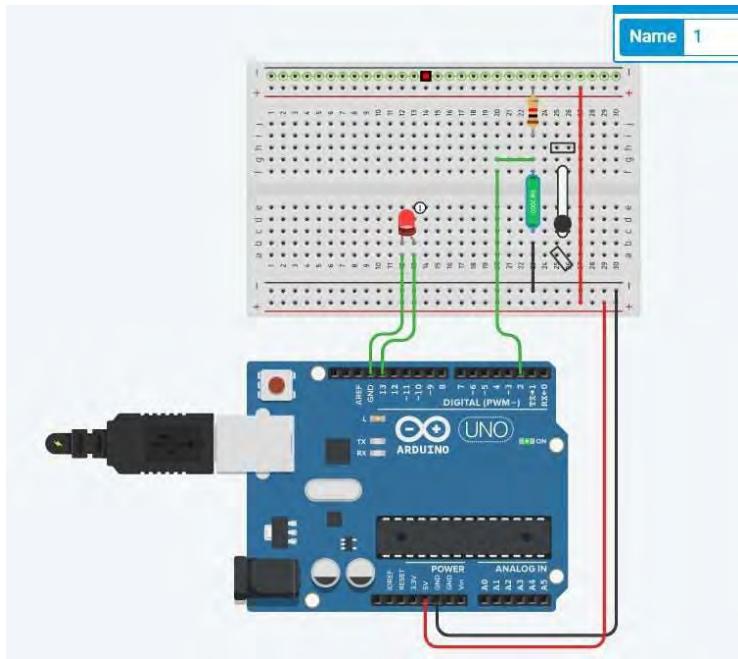


## CODE

```
int tilt = 2;  
  
int led = 13;  
  
void setup()  
{  
    pinMode(tilt, INPUT);  
    pinMode(led, OUTPUT);  
}  
  
void loop()  
{  
    int reading;  
    reading = digitalRead(tilt);  
    if(reading)  
        digitalWrite(led, LOW);  
    else  
        digitalWrite(led, HIGH);  
}
```

## OUTPUT

Designed a Smart Package handling system using Tilt Sensor and LED.



**Name – Rishav Agrawal**

Program No. – 13

Program Title – IR based SERVO Motor controller

---

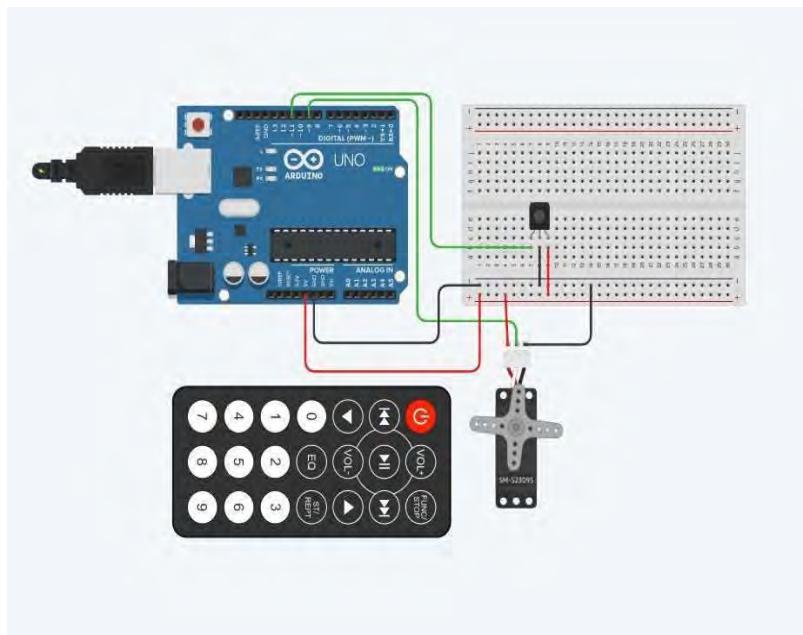
## **AIM**

Design IR based SERVO Motor controller. (Clockwise and Counterclockwise rotation of shaft).

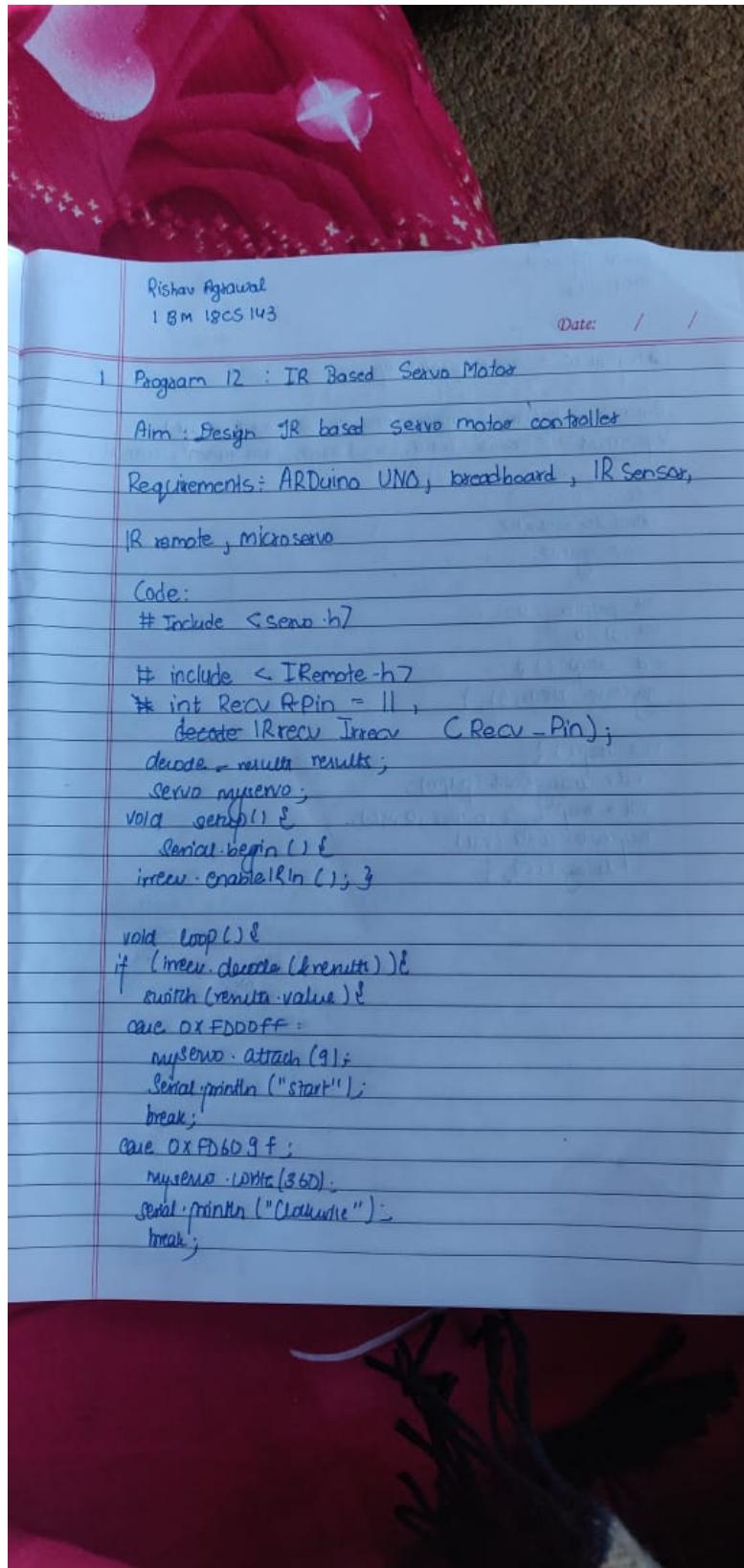
## **HARDWARES REQUIRED**

- Arduino Board,
- Breadboard Small,
- IR Sensor,
- IR Remote,
- Micro Servo

## **CIRCUIT DIAGRAM**



## WRITE-UP



Date: / /

Case 0xFD20DF:

```
myserio.write(360);  
Serial.println("counter clockwise");  
break;
```

default:

```
Serial.print("unrecognized code received: ");  
Serial.println(result.value, HEX);  
break; }  
imrcv.rename(); }
```

W9

## CODE

```
#include <Servo.h>

#include <IRremote.h>

int RECV_PIN = 11;

IRrecv irrecv(RECV_PIN);

decode_results results;
```

```
Servo myservo;

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

void loop(){
    if (irrecv.decode(&results))
    {
        switch (results.value)
        {
            case 0xFD00FF:
                myservo.attach(9);
                Serial.println("Start");
                break;
            case 0xFD609F:
                myservo.write(360);
                Serial.println("Clockwise");
                break;
            case 0xFD20DF:
                myservo.write(-360);
                Serial.println("Counter Clockwise");
                break;
            default:

```

```
    Serial.print("Unrecognized code received: 0x");
    Serial.println(results.value, HEX);
    break;
}

irrecv.resume();

}
}
```

## OUTPUT

Designed a Smart Package handling system using Tilt Sensor and LED.



The screenshot shows the Arduino Serial Monitor window. The title bar says "Serial Monitor". The main area displays the following text:

```
Starting..
Clockwise..
Clockwise..
Counter Clockwise..
Counter Clockwise..
Unrecognized code received: 0xFD48B7
```

**Name – Rishav Agrawal**

Program No. – 14

Program Title – **RGB Led and LCD**

---

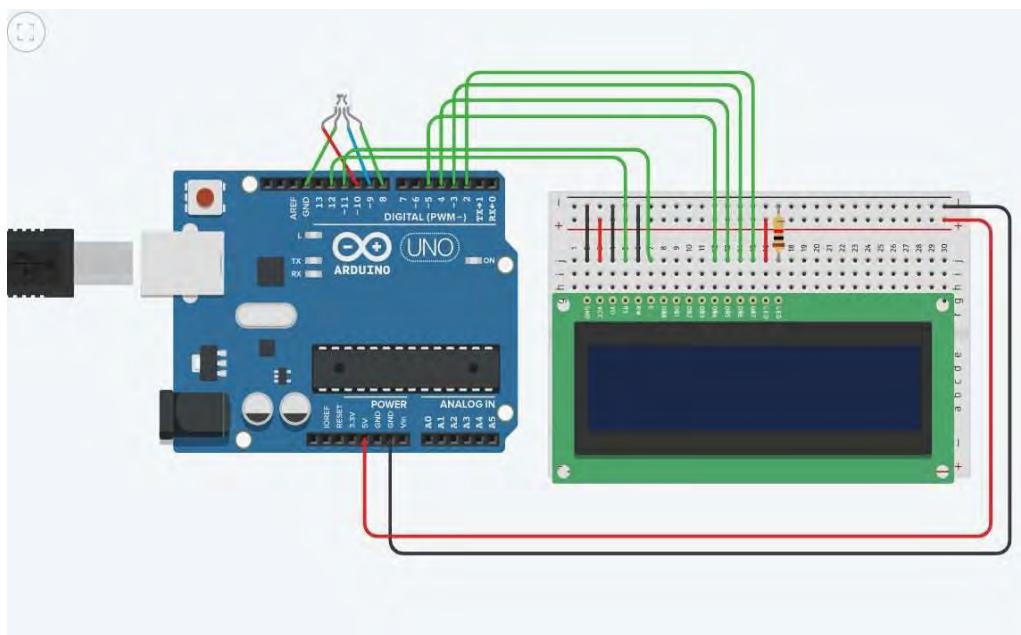
## **AIM**

Design a display system to print the RED, BLUE and Green colors (RGB Led and LCD).

## **HARDWARES REQUIRED**

- Arduino Board
- Breadboard Small
- LCD 16x2
- RGB LED
- Resistor

## **CIRCUIT DIAGRAM**



## WRITE-UP

Rishav Agrawal  
1BM18CST43

PAGE NO.: \_\_\_\_\_  
DATE: \_\_\_\_\_

RGB with LCD

Aim:- Display stem to print colour served by RGB led.

Hardware :- Arduino board, breadboard, LCD x 16x2, RGB LED, Resistor.

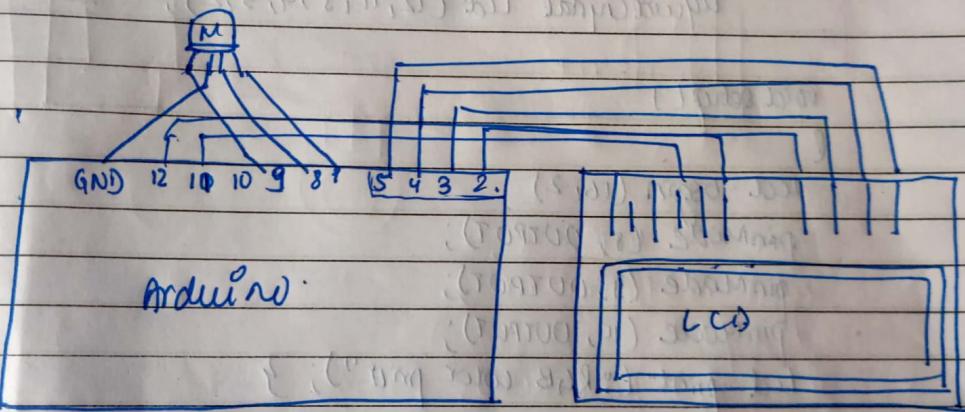
Code:-

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(12,11,5,4,3,2);

void setup()
{
    lcd.begin(16,2);
    pinMode(8, OUTPUT);
    pinMode(9, OUTPUT);
    pinMode(10, OUTPUT);
    lcd.print("RGB color print"); }

void loop()
{
    delay(1000);
    digitalWrite(10,HIGH);
    digitalWrite(9,LOW);
    digitalWrite(8,LOW);
    lcd.print("RED");
    delay(1000);
    lcd.clear();
    digitalWrite(9,HIGH);
    digitalWrite(10,LOW);
    digitalWrite(8,LOW);
    lcd.print("BLUE");
    delay(1000); }
```

```
lcd.clear();  
digitalWrite (8, HIGH);  
digitalWrite (10, LOW);  
digitalWrite (9, LOW);  
lcd.print ("GREEN");  
delay (1000);  
lcd.clear();
```



## CODE

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(12,11,5,4,3,2);
```

```
int red=10;
```

```
int green=8;
```

```
int blue=9;
```

```
void setup()
```

```
{
```

```
pinMode(10, OUTPUT);
```

```
pinMode(9, OUTPUT);
```

```
pinMode(8, OUTPUT);
```

```
}
```

```
void loop()
```

```
{
```

```
lcd.setCursor(0,0);
```

```
lcd.print("RGB Color Print!");
```

```
delay(1000);
```

```
lcd.clear();
```

```
RGB_color(255,0,0);//red
```

```
lcd.print("RED");
```

```
delay(1000);
```

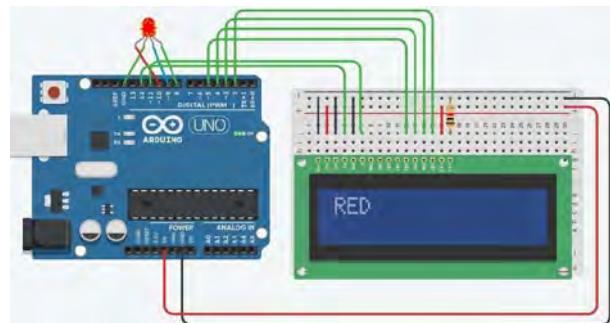
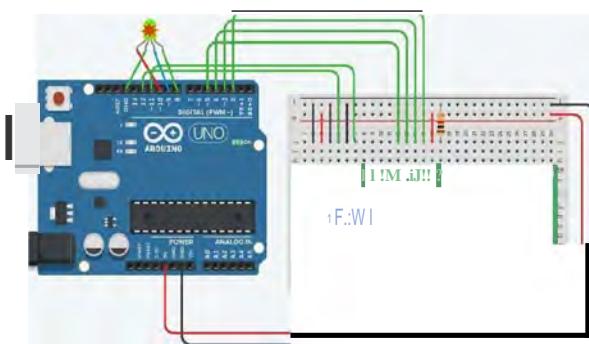
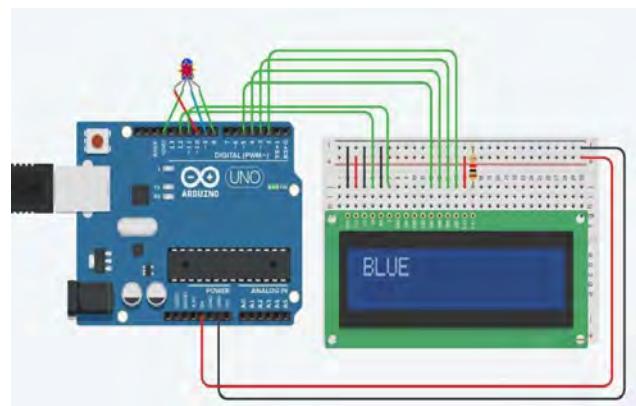
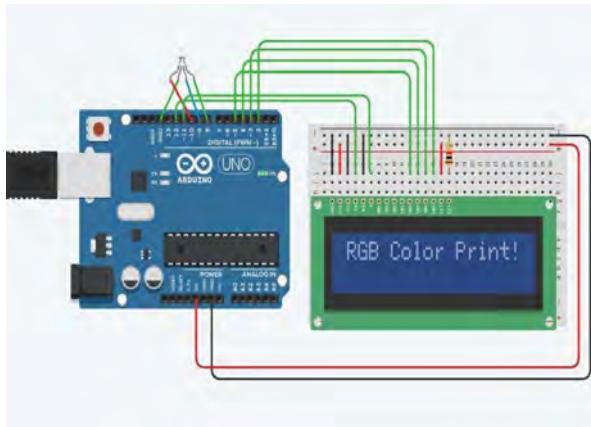
```
lcd.clear();  
  
RGB_color(0,255,0);//Green  
lcd.print("GREEN");  
delay(1000);  
lcd.clear();  
  
RGB_color(0,0,255);//Blue  
lcd.print("BLUE");  
delay(1000);  
lcd.clear();
```

```
RGB_color(0,0,0);//White
lcd.print("WHITE");
delay(1000);
lcd.clear();
}

void RGB_color(int red_value, int green_value, int blue_value)
{
    analogWrite(red,red_value);
    analogWrite(green,green_value);
    analogWrite(blue,blue_value);
}
```

## OUTPUT

Designed a display system to print the RED,BLUE and Green colors (RGB Led and LCD).



**Name – Rishav Agrawal**

**Program No. – 15**

**Program Title – Smart irrigation system**

---

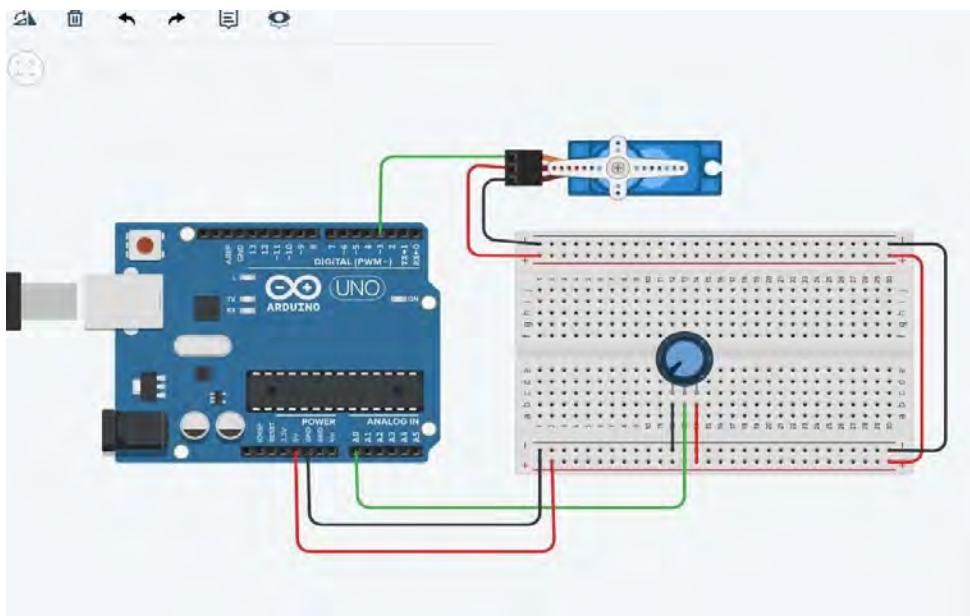
## **AIM**

Design a smart irrigation system (Potentiometer, Servo motor shaft).

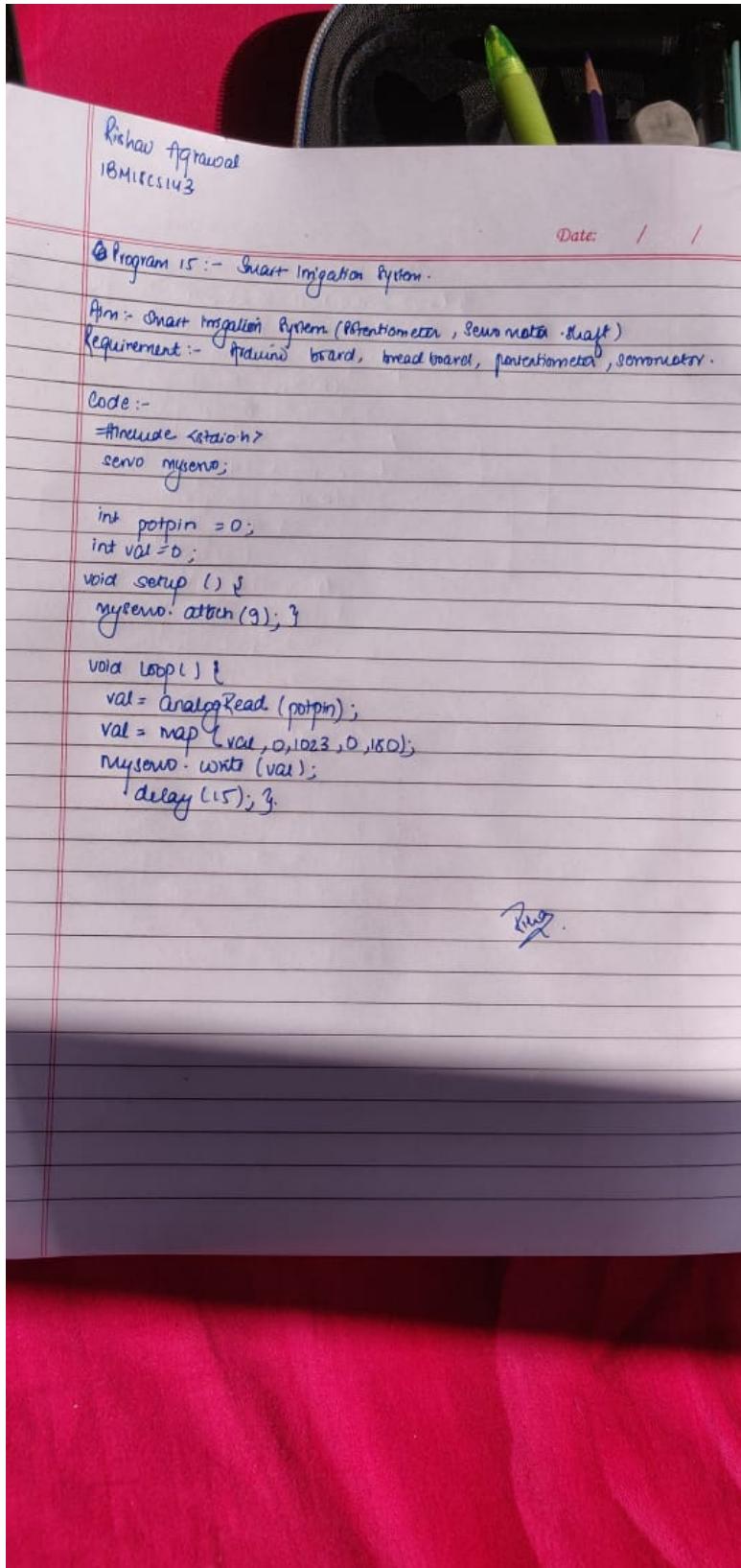
## **HARDWARES REQUIRED**

- Arduino Board
- Breadboard Small
- Potentiometer
- Servo motor shaft

## **CIRCUIT DIAGRAM**



## WRITE-UP



## CODE

```
#include <Servo.h>

Servo myservo; // create servo object to control a servo
// twelve servo objects can be created on most boards

int pos = 0;    // variable to store the servo position
int sensorPin = A0; // select the input pin for the potentiometer
int sensorValue = 0; // variable to store the value coming from the sensor

void setup() {
  myservo.attach(3); // attaches the servo on pin 9 to the servo object
  Serial.begin(9600);
```

```
}

void loop() {
    // read the value from the sensor:
    sensorValue = analogRead(sensorPin);
    Serial.println (sensorValue);
    if(sensorValue>500)
    {
        for (pos = 0; pos <= 180; pos += 1) { // goes from 0 degrees to 180
            degrees
            // in steps of 1 degree
            myservo.write(pos);           // tell servo to go to position in variable'pos'
            delay(15);                  // waits 15ms for the servo to reach theposition
        }
        for (pos = 180; pos >= 0; pos -= 1) { // goes from 180 degrees to 0degrees
            myservo.write(pos);           // tell servo to go to position in variable 'pos'
            delay(15);                  // waits 15ms for the servo to reach theposition
        }
    }
    delay (1000);
}
```

## OUTPUT

Designed a smart irrigation system (Potentiometer, Servo motor shaft).

