

## **IOT LAB CONSOLIDATED REPORT**

**USN:1BM18CS124**

**Name – Venkatesha Prasada CH**

**Program No. – 01**

**Program Title – LED Blinking**

---

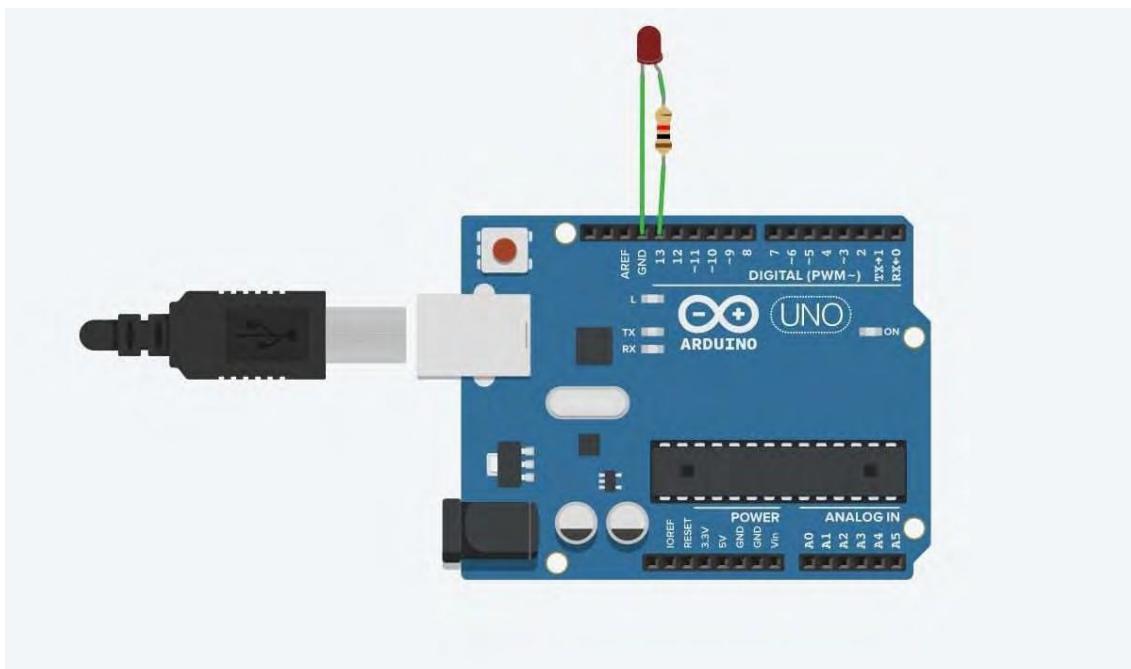
### **AIM**

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

### **HARDWARES REQUIRED**

- Arduino Board
- LEDs

### **CIRCUIT DIAGRAM**



## WRITE-UP

Venkatesha Prasadach

18M18CS124

### Code

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

void loop() {
    digitalWrite(13, HIGH);
    delay(1000);
    digitalWrite(13, LOW);
    delay(1000);
}
```

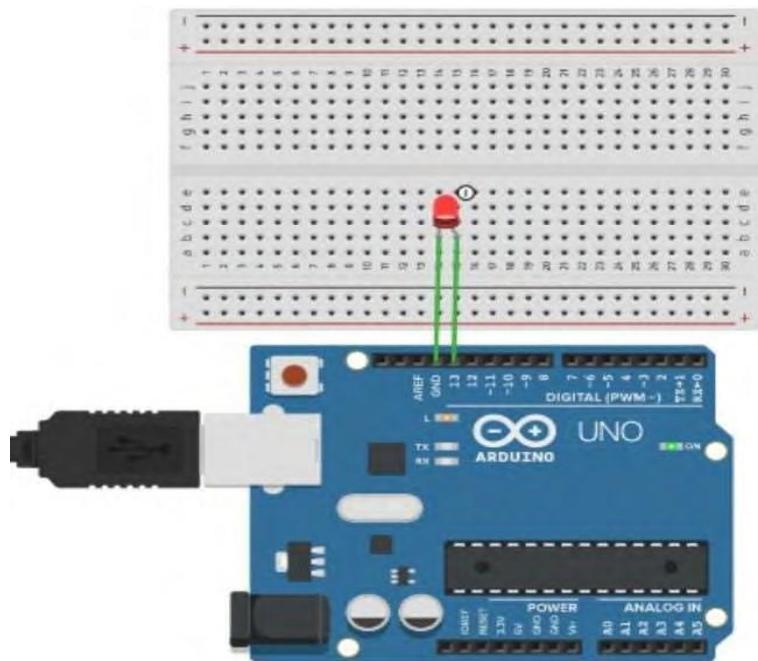
## CODE

```
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.



**Name – Venkatesha Prasada CH**

Program No. – 02

Program Title – Traffic Controller

---

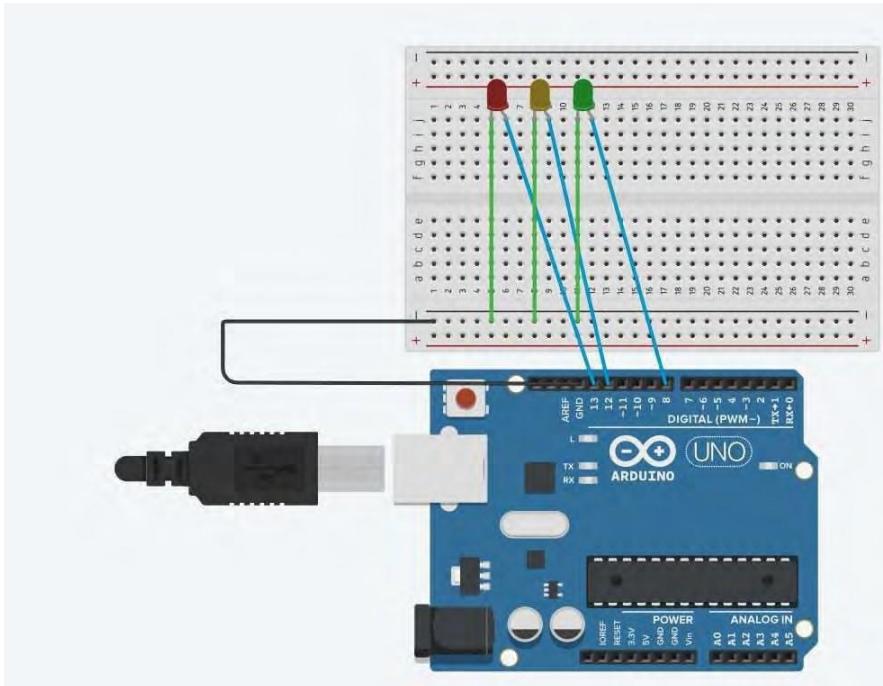
## AIM

Traffic Signal Simulator.

## HARDWARES REQUIRED

- Arduino Board
- LEDs
- Breadboard

## CIRCUIT DIAGRAM



## WRITE-UP

Venkatesha Prasad  
1BM18CS124

code - Lab2

```
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(2000);
    digitalWrite(13, LOW);
    digitalWrite(12, HIGH);
    digitalWrite(11, LOW);
    delay(2000);
    digitalWrite(13, LOW);
    digitalWrite(12, LOW);
    digitalWrite(11, HIGH);
}
```

Amit

## 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.

**Name – Venkatesha Prasada CH**

Program No. – 03

Program Title – LED fading without potentiometer

---

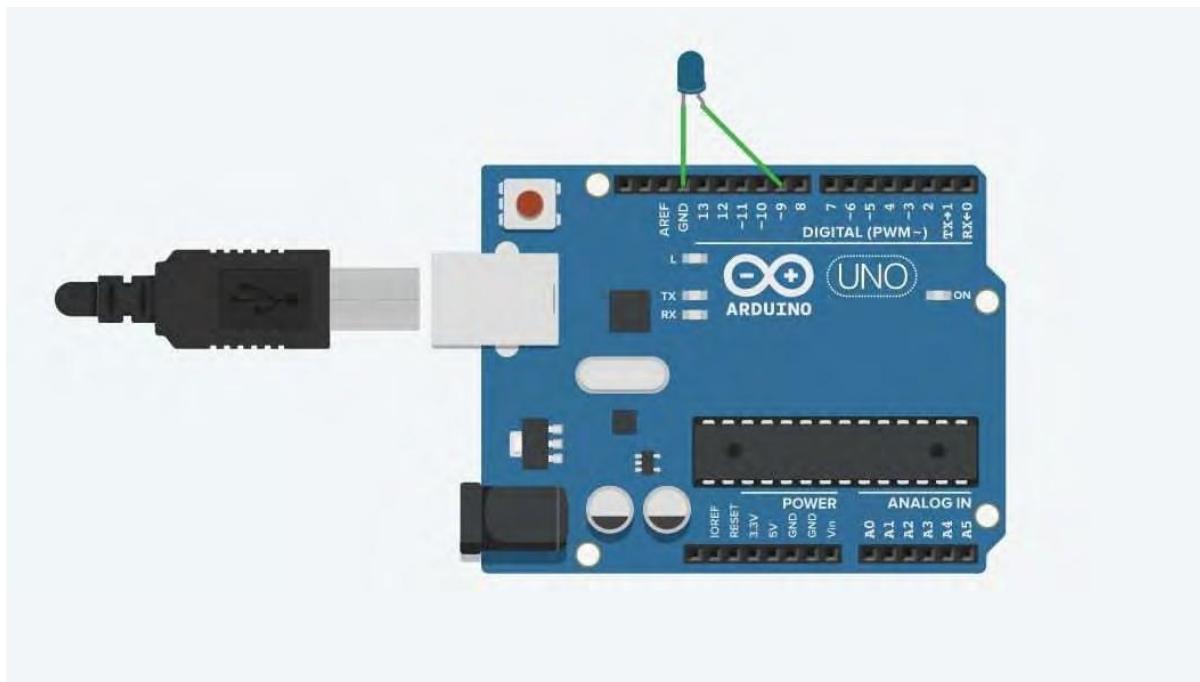
## **AIM**

Demonstrate to show LED fading.

## **HARDWARES REQUIRED**

- Arduino Board
- LED bulb

## **CIRCUIT DIAGRAM**



## **WRITE-UP**

Exp - 3

int ledPin = 9;  
int potPin = A0;

Venkatesha Prasad CT  
1BM18CS124

Aim : Demonstrate fading LED using potentiometer

Requirements : Arduino board, LED, connecting wire, resistor, potentiometer, breadboard, jumper wires.

Code :

```
void setup()
{
    Serial.begin(9600);
    pinMode(9, OUTPUT);
}

void loop()
{
    int analog = analogRead(A0);
    int brightness = map(analog, 0, 1023, 0, 255);
    analogWrite(a, brightness);
    Serial.print("Analog value : ");
    Serial.print(analog);
    Serial.print(" brightness value : ");
    Serial.print(brightness);
    Serial.print(" (0-255) pixels");
}
```

Submitted

## 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 – Venkatesha Prasada CH**

Program No. – 04

Program Title – LED fading with potentiometer

---

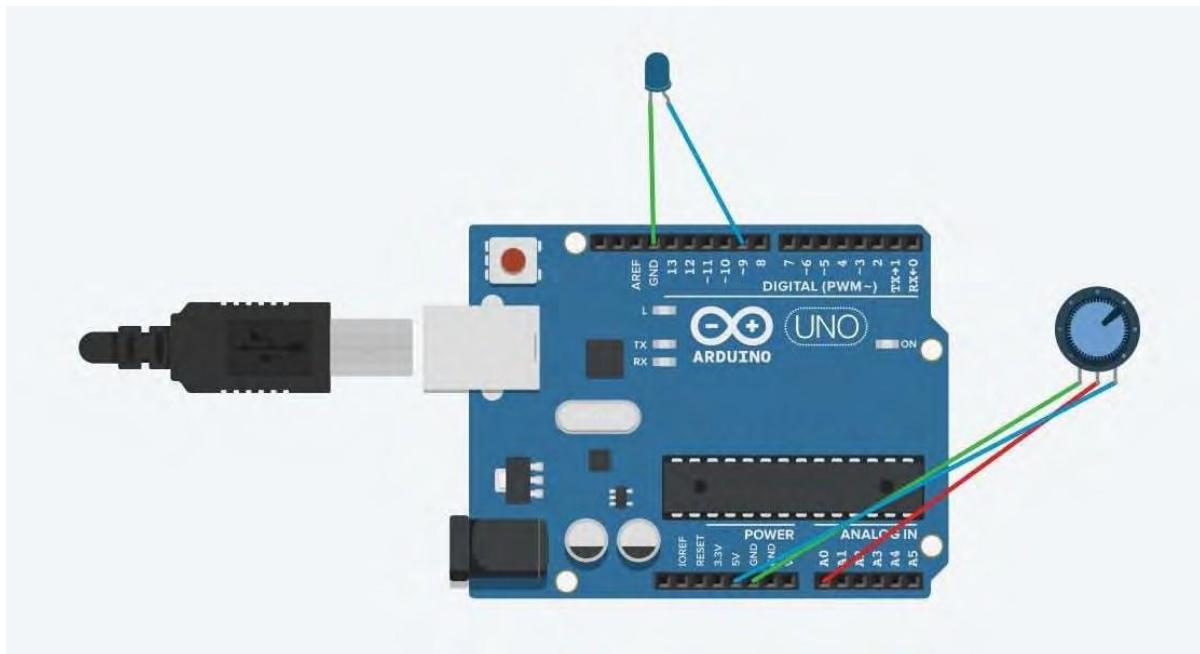
## **AIM**

Demonstrate to show LED fading(analog output).

## **HARDWARES REQUIRED**

- Arduino Board
- LED bulb
- Potentiometer

## **CIRCUIT DIAGRAM**



## **WRITE-UP**

exp-4

Venkatesha Dossada (14)  
13M18CS124

Aim: Fading without potentiometer.

Required : Arduino board, LED, wire, resistor.

Code:

```
void setup() {  
    Serial.begin(9600);  
    pinMode(9, OUTPUT);  
}  
  
void loop() {  
    for (brightness = 0; brightness <= 255; brightness++) {  
        analogWrite(9, brightness);  
        delay(30);  
    }  
    for (brightness = 255; (brightness) = 0; brightness -= 5) {  
        analogWrite(9, brightness);  
        delay(30);  
    }  
}
```

Abhishek

## CODE

```
int LED_PIN = 9;

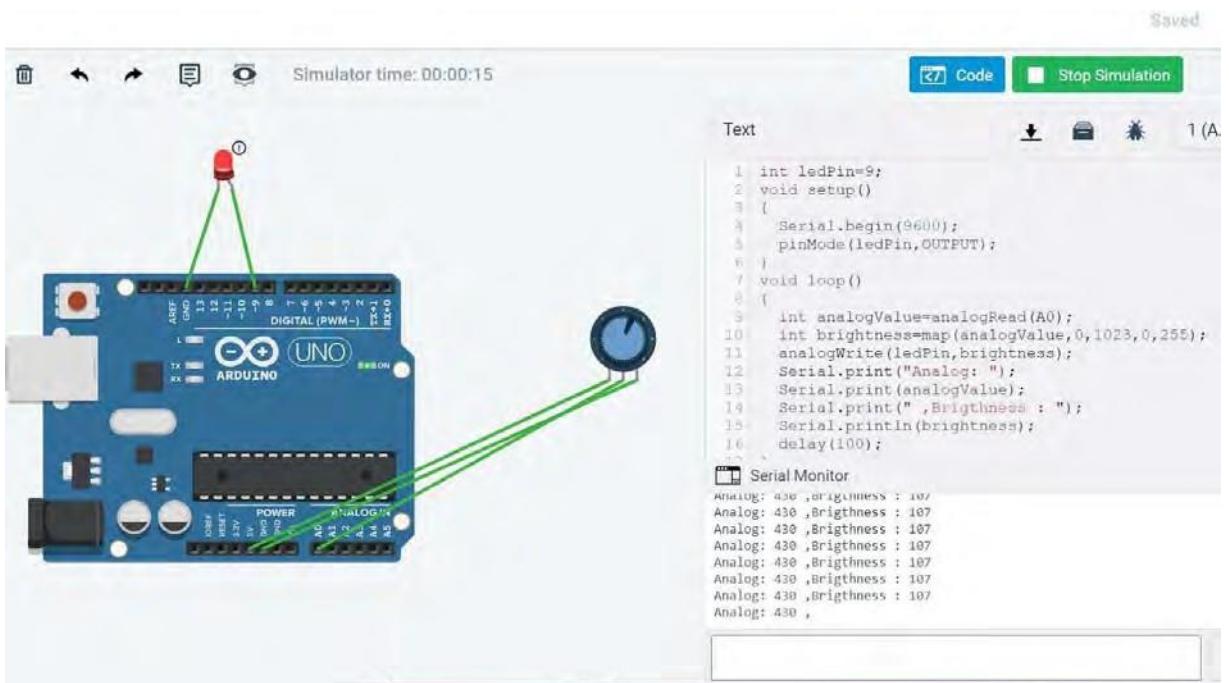
void setup()
{
    Serial.begin(9600);
    pinMode(LED_PIN, OUTPUT);
}

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 – Venkatesha Prasada CH**

Program No. – 05

Program Title – ON/OFF LED using Push Button

---

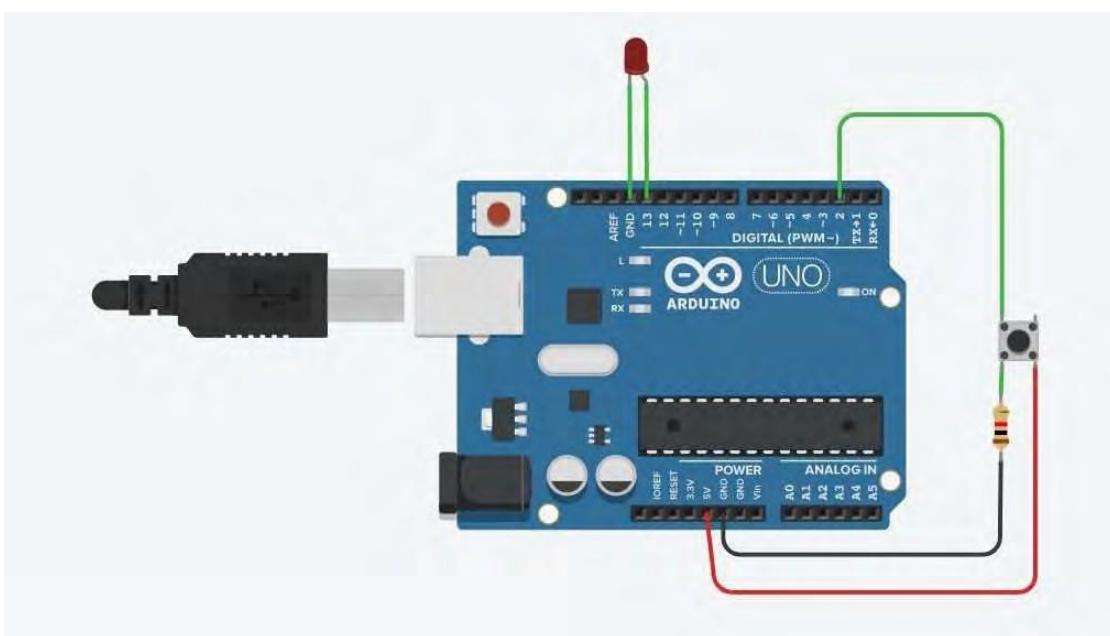
## **AIM**

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**

exp-6

Venkatesha Preysadaiah  
18M18CS124

Aim: On/off LED using LDR

Requirement: Arduino board, LED, breadboard, Resistor, Wires, photomistor.

```
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 dark, LED is on");  
    }  
    else  
    {  
        digitalWrite(ledPin,LOW);  
        Serial.println("----");  
    }  
}
```

Alka

## **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 –** Venkatesha

Prasada CH

**Program No. – 06**

**Program Title – LDR**

---

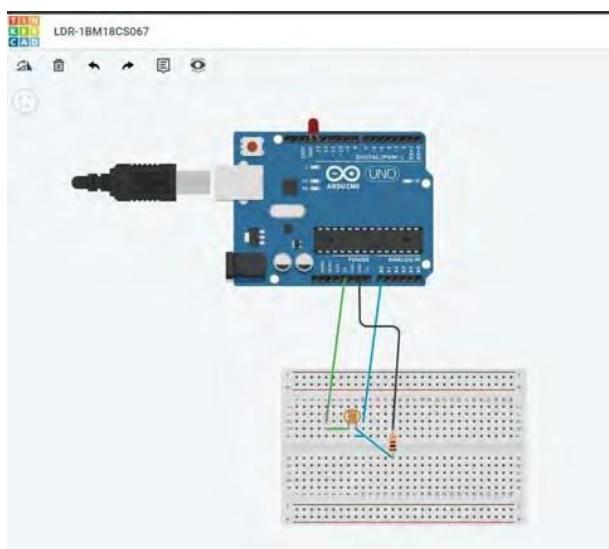
## **AIM**

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**

Venkatesha Preysada/H  
18M18CS124

exp-1

Aim: On/Off LED using LDR

Requirement: Arduino board, LED, breadboard, resistor, wires, 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("----");  
    }  
}
```

Oldest

## 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

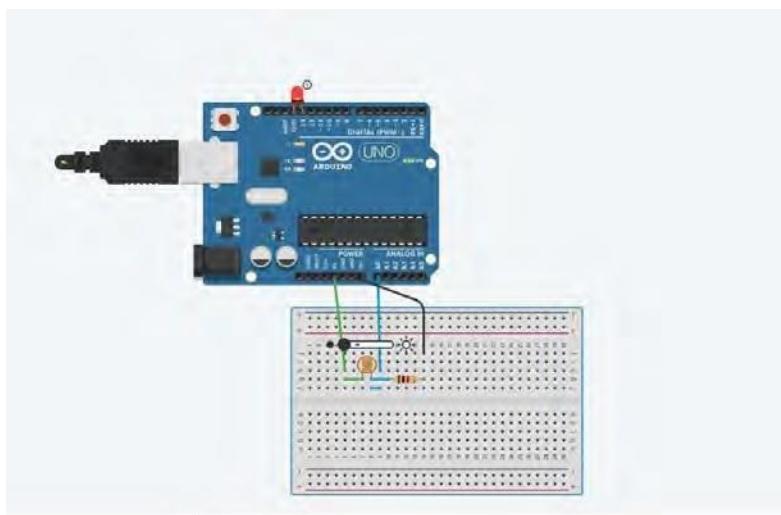
{
    digitalWrite(ledPin, LOW);
    Serial.println(" -----");
}

}

```

## OUTPUT

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



**Name –** Venkatesha

Prasada CH

Program No. – 07

Program Title – PIR

---

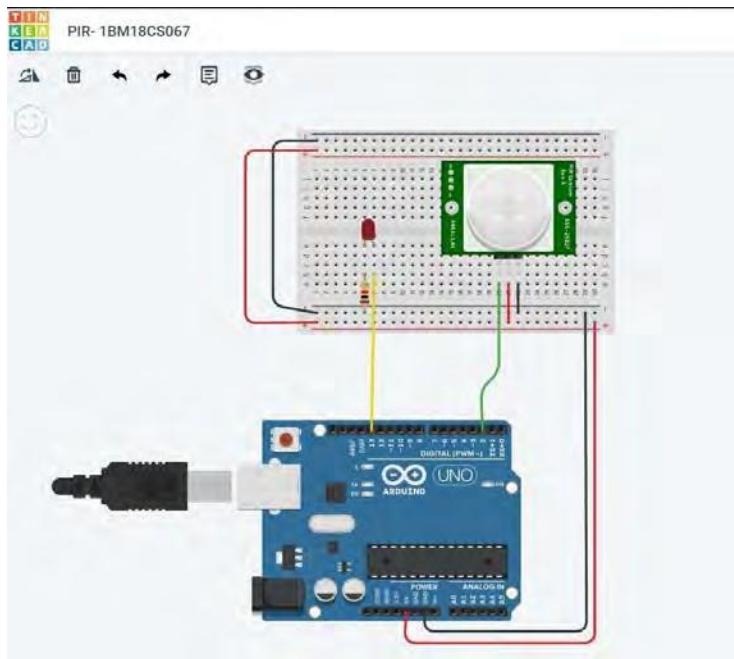
## AIM

Demonstrate to show working of PIR sensor.

## HARDWARES REQUIRED

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

## CIRCUIT DIAGRAM



## **WRITE-UP**

Venkatesha Prasada C14

1BME8CS124

exp - 7

Aim: Demonstrate working of PIR sensor

Requirement : Arduino board, led, resistor, wire, PIR sensor.

Code:

```
void setup() {
    pinMode(13, OUTPUT);
    pinMode(6, INPUT);
    Serial.begin(9600);
}

void loop() {
    val = digitalRead(6);
    if (val == HIGH) {
        digitalWrite(13, HIGH);
        delay(1000);
        print(" motion detected");
    }
    if (val == LOW) {
        digitalWrite(13, LOW);
        print(" ---");
    }
}
```

Arleat



## CODE

```
int sensorState = 0;

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

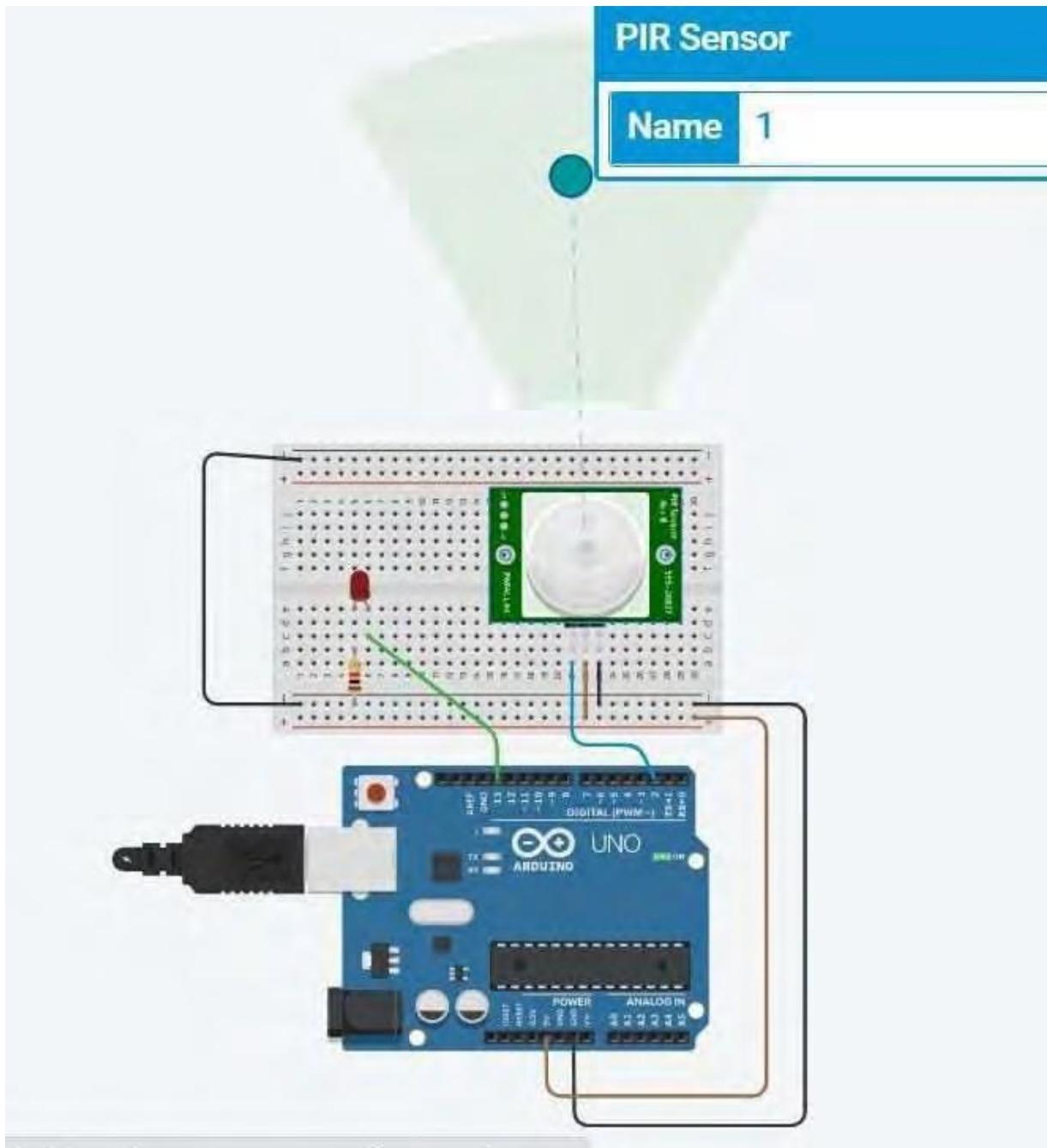
void loop()
{
    sensorState = digitalRead(2);

    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 – Venkatesha Prasada CH**

Program No. – 08

Program Title – Distance Measurement using ultrasonic sensor

---

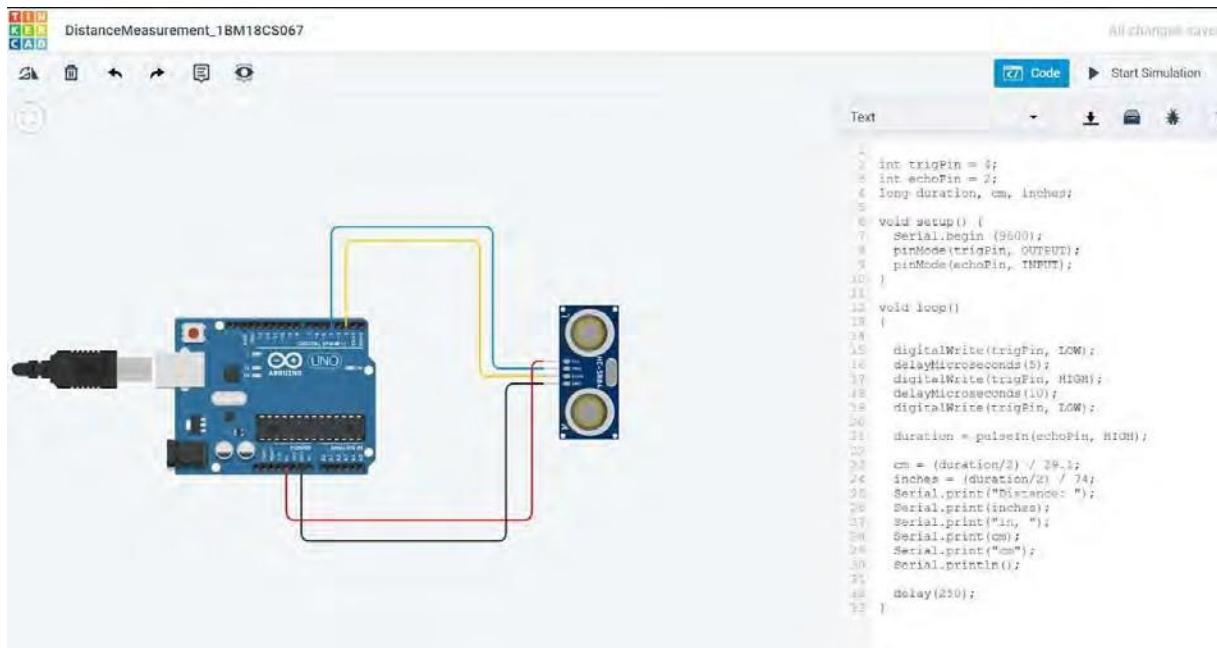
## AIM

Design a system to measure the distance between objects.

## HARDWARES REQUIRED

- Arduino Board
- Ultrasonic sensor HC-SR04

## CIRCUIT DIAGRAM



## **WRITE-UP**

exp-8

Venkatesha Prasadulu  
1BM18CS124

Aim: Distance measurement using ultrasonic sensor

Requirements: Arduino board, Ultrasonic sensor

Code:

```
const int pingPin = 7;  
  
void setup()  
{  
    Serial.begin(9600);  
}  
  
void loop()  
{  
    long duration, inch, cm;  
    pinMode(pingPin, OUTPUT);  
    digitalWrite(pingPin, LOW);  
    delayMicroseconds(2);  
    digitalWrite(pingPin, HIGH);  
    delayMicroseconds(10);  
    digitalWrite(pingPin, LOW);  
    pinMode(7, INPUT);  
    duration = pulseIn(7, HIGH);  
    inch = duration / 147;  
    cm = inch * 2.54;  
    Serial.print("Distance(in cm) : ");  
    Serial.println(cm);  
    Serial.print("Distance(in inch) : ");  
    Serial.println(inch);  
}  
delay(500);
```

✓ Shukla



## 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 | 1 x ! ! J Circ u it design Amazing Fyra[n] x + X

iJD!ll Circ u it design Amazing Fyra[n] x

131[: object distance f:HJ(j)

Simulator time: 00:00:07

**fjh++@FH**

Text .i.

1 (Arduino Uno R3)

```

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

**Name – Venkatesha Prasada CH**

Program No. – 09

Program Title – Fire Alarm using flame Sensor

---

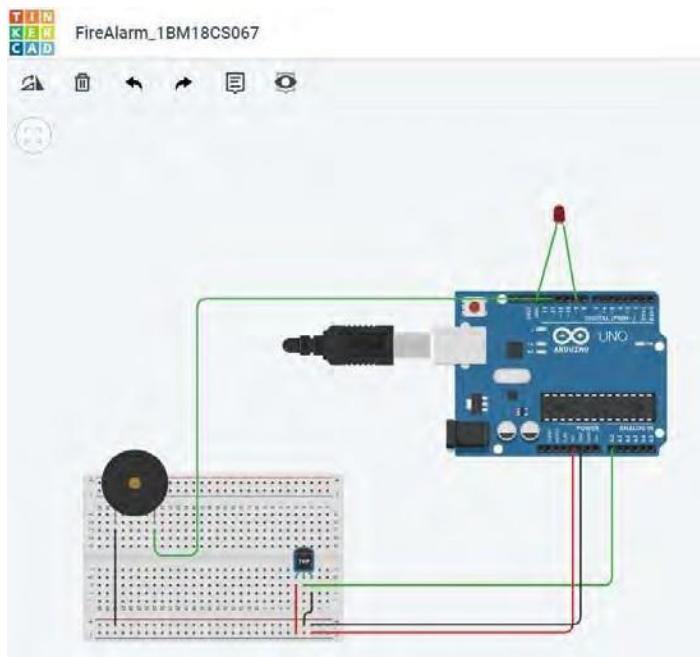
## **AIM**

Design an alert system using a flame sensor.

## **HARDWARES REQUIRED**

- Arduino Board
- Piezo
- Temperature Sensor
- Breadboard small

## **CIRCUIT DIAGRAM**



## **WRITE-UP**

exp - 9

Verlesterha Prasada Ch

IBM 18CS124

Aim: Fire alarm system

Requirements: Arduino board, Temperature sensor, LED.

Code:

```
void setup() {
    pinMode(13, OUTPUT);
    pinMode(12, OUTPUT);
    serial.begin(9600);
}

void loop() {
    int rawvolt, temp;
    float millivolt;
    rawvolt = analogRead(A0);
    millivolt = (rawvolt/1024.0)*5000;
    temp = millivolt/10;
    delay(100);
    if (temp > 37) {
        serial.println("Fire Alert");
        digitalWrite(12, HIGH);
        digitalWrite(13, HIGH);
    }
    else {
        serial.println("Safe");
        digitalWrite(12, LOW);
        digitalWrite(13, LOW);
    }
}
```

Chileant



## 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)
```

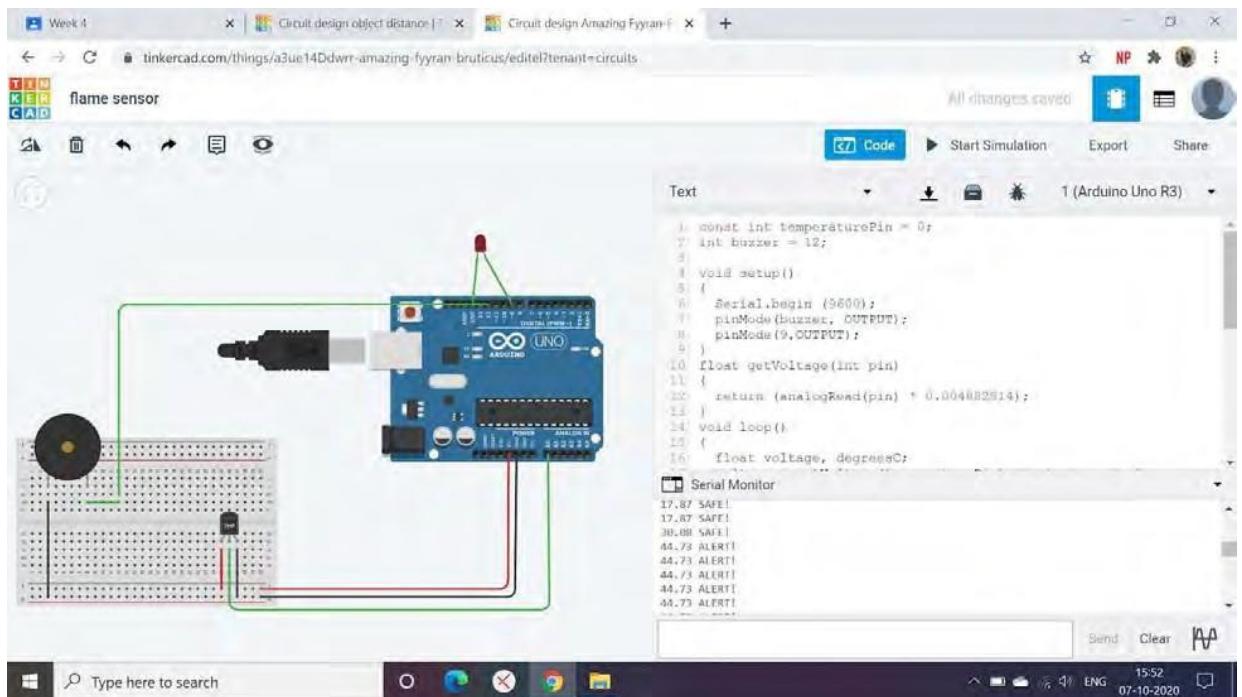
```
{  
    Serial.print(degreesC);  
    Serial.println(" SAFE!");  
}  
  
}
```

```
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 – Venkatesha Prasada CH**

## Program No. – 10

## Program Title – Gas Sensor

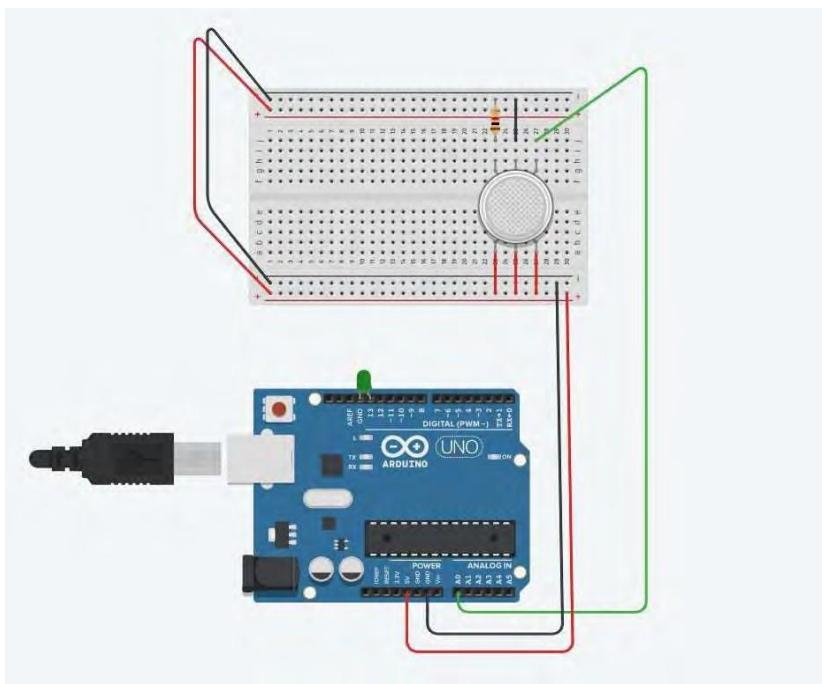
AIM

To design a smart gas leakage indicator system.

## **HARDWARES REQUIRED**

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

## CIRCUIT DIAGRAM



## **WRITE-UP**

exp -10

Venkateswaran Prasad

1BM18CS124

Aim : To design smart gas leakage indication system

Code :

```
float sensorValue;  
void setup() {  
    Serial.begin(9600);  
}  
void loop() {  
    sensorValue = analogRead(A0);  
    if (sensorValue >= 200) {  
        digitalWrite(13, HIGH);  
        Serial.print(sensorValue);  
        Serial.println(" Danger...");  
        delay(1000);  
    }  
    else {  
        digitalWrite(11, LOW);  
        digitalWrite(12, LOW);  
        Serial.println(" Sensor Value : ");  
        Serial.println(sensorValue);  
    }  
    delay(1000);  
}
```

Oliveat



## 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);

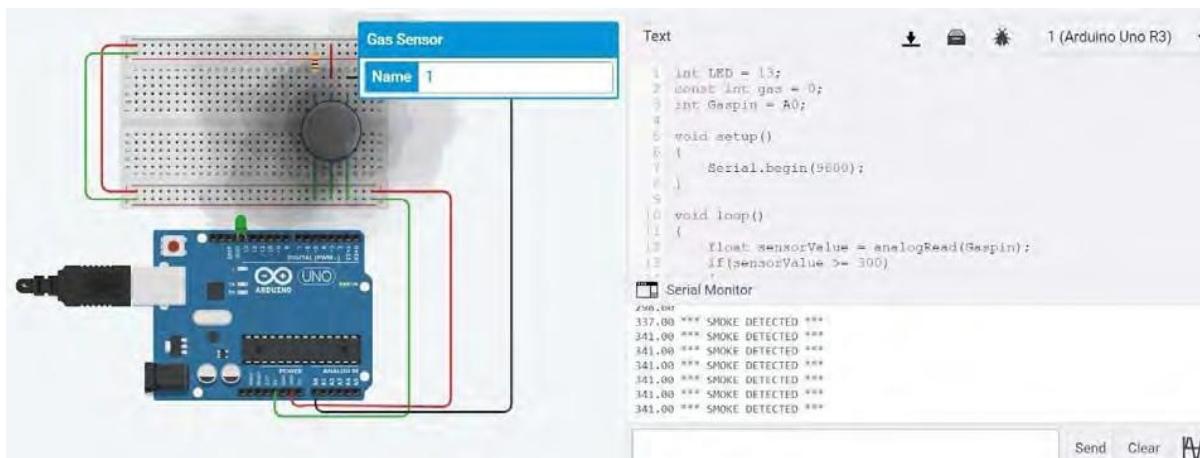
}

delay(1000);

}
```

## OUTPUT

Designed a smart gas leakage indicator system.



**Name – Venkatesha Prasada CH**

Program No. – 11

Program Title – Vibration motor and LDR

---

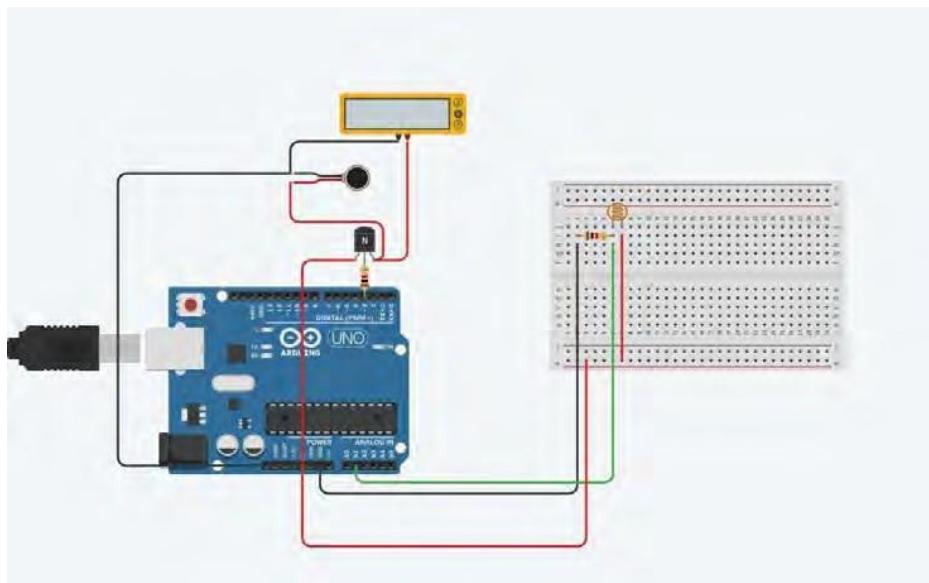
## **AIM**

To design an automated day indicator system.

## **HARDWARES REQUIRED**

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

## **CIRCUIT DIAGRAM**



## **WRITE-UP**

exp - 11

venkatesha messada  
IBM18CS124

Aim: Design automated day indicator.

code:

```
int sensorValue;  
void setup()  
{  
    pinMode(3, OUTPUT);  
    Serial.begin(9600);  
}  
  
void loop()  
{  
    sensorValue = analogRead(A0);  
    Serial.println(sensorValue);  
    if (sensorValue > 400) {  
        digitalWrite(3, HIGH);  
    }  
    else {  
        digitalWrite(3, LOW);  
    }  
}
```

Chukat

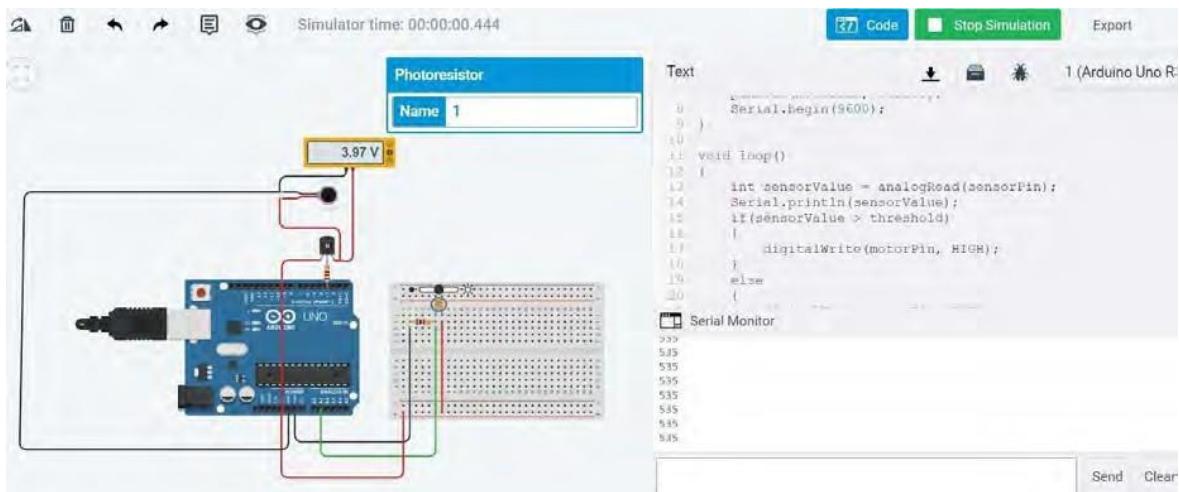
## CODE

```
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 –** Venkatesha Prasada

CH

Program No. – 12

Program Title – Tilt Sensor

---

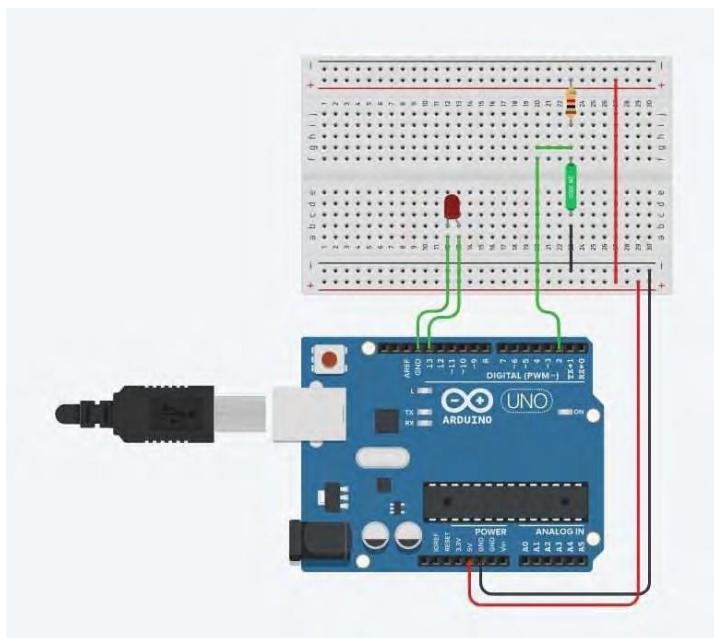
## AIM

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

## HARDWARES REQUIRED

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

## CIRCUIT DIAGRAM



## **WRITE-UP**

exp-12

Verkadecha Prasada Reddy

1BM18CS124

Aim: Littscreen sensor demonstration

Code:

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

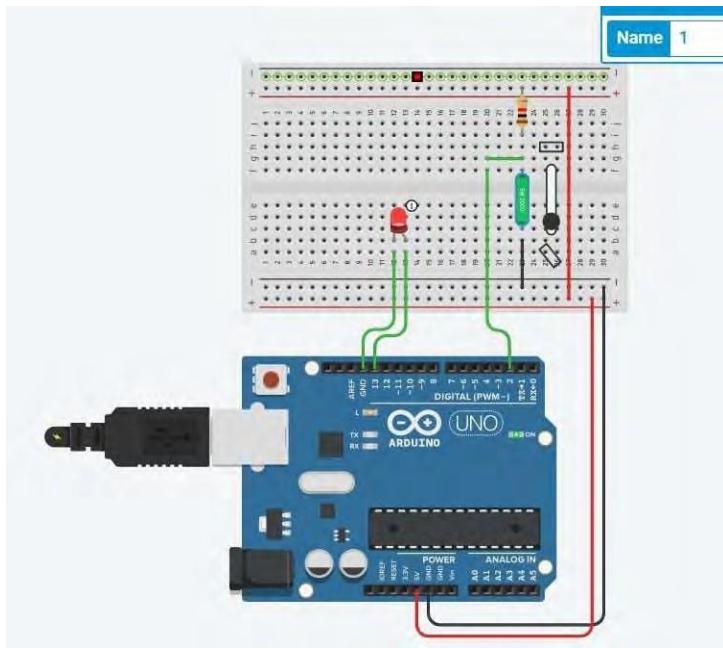
Chileet

## 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 – Venkatesha Prasada CH**

## Program No. – 13

## Program Title – IR based SERVO Motor controller

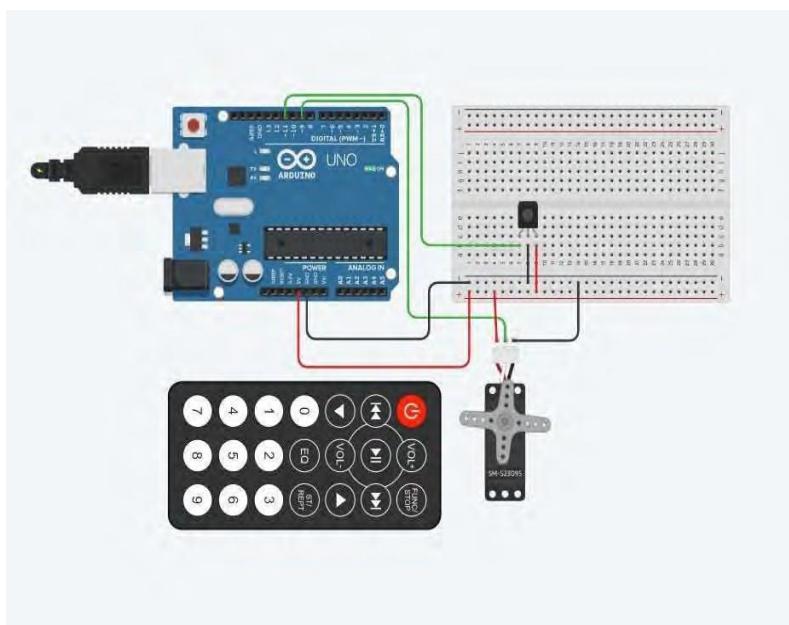
AIM

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**

exp-13

Verkanteshwar Prasad Ch

18M18CS124

Aim: Design IR based servo motor.

```
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 '0x1D00FF':
                      myservo.attach(9);
                      Serial.println("start");
                      break;
                  case 0xF0L09F:
                      myservo.write(360);
                      Serial.println("clockwise");
                      Serial.println(results.value, HEX);
                      break;
                  case 0xFD20DF:
                      myservo.write(-360);
                      Serial.println("Anti-clockwise");
                      Serial.println(results.value, HEX);
                      break;
              }
          }
      }
```

```
default : Serial.print("Unrecognised code : ");
Serial.println(results.value, HEX);
```

```
}
```

```
ifrecv.available();
```

```
}
```

```
}
```

shukat

## 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 – Venkatesha Prasada CH**

Program No. – 14

Program Title – **RGB Led and LCD**

---

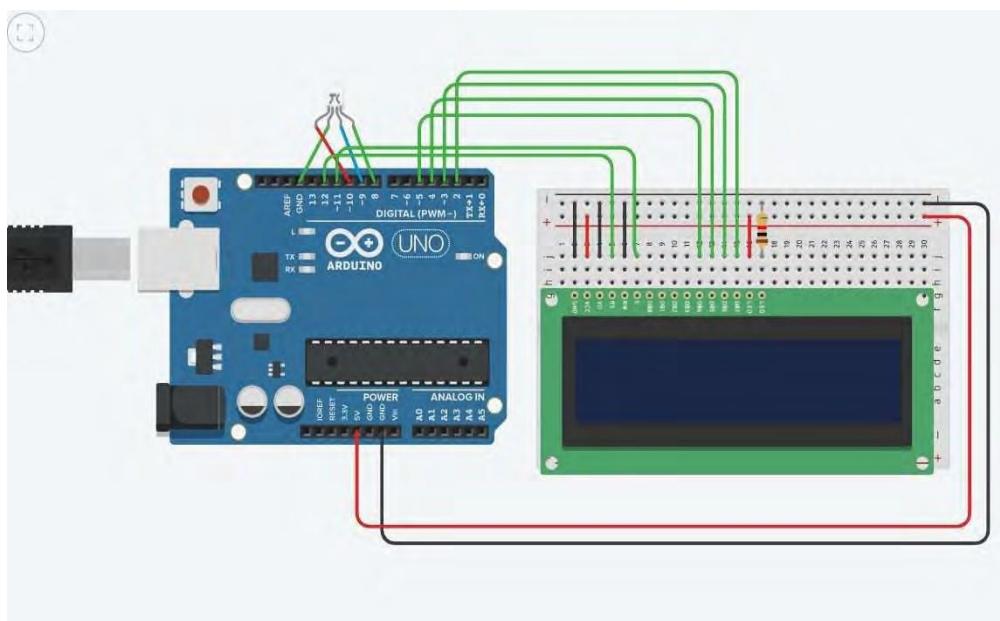
## **AIM**

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**

exp - 14

PSI O81/M/91

Verleeketha Prasade CSE  
1BM18CS124

Aim : Design RGB LCD & configura LCD

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

Dhileet



## 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);

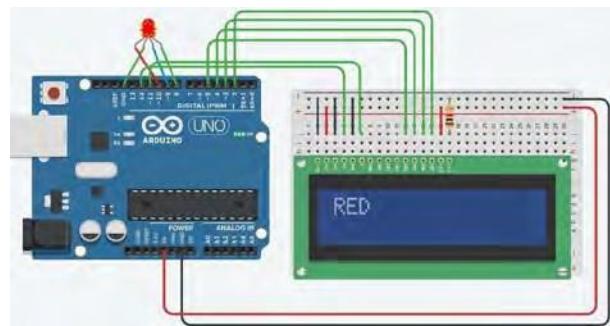
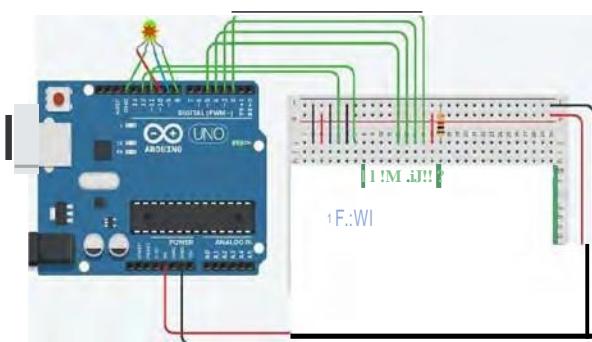
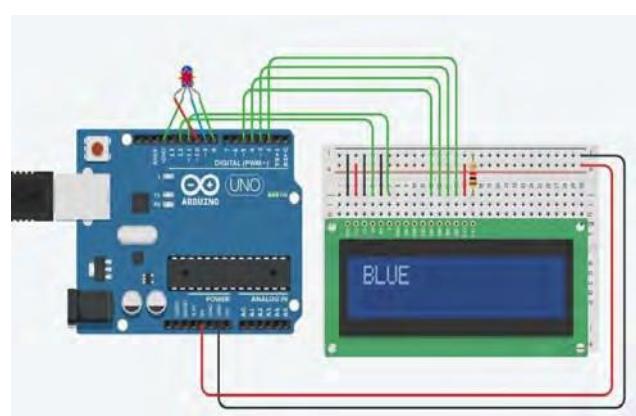
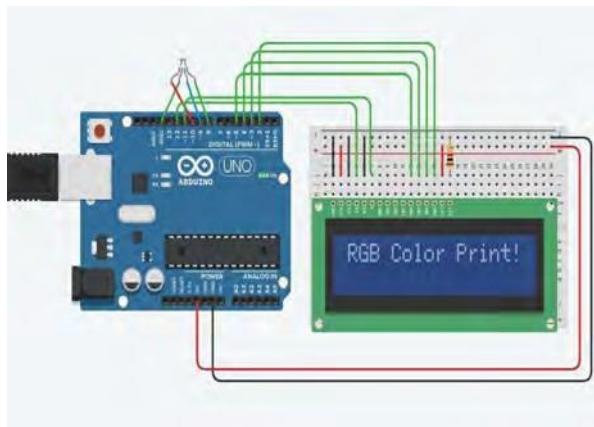
lcd.clear();
```

```
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 – Venkatesha Prasada CH**

## Program No. – 15

# Program Title – Smart irrigation system

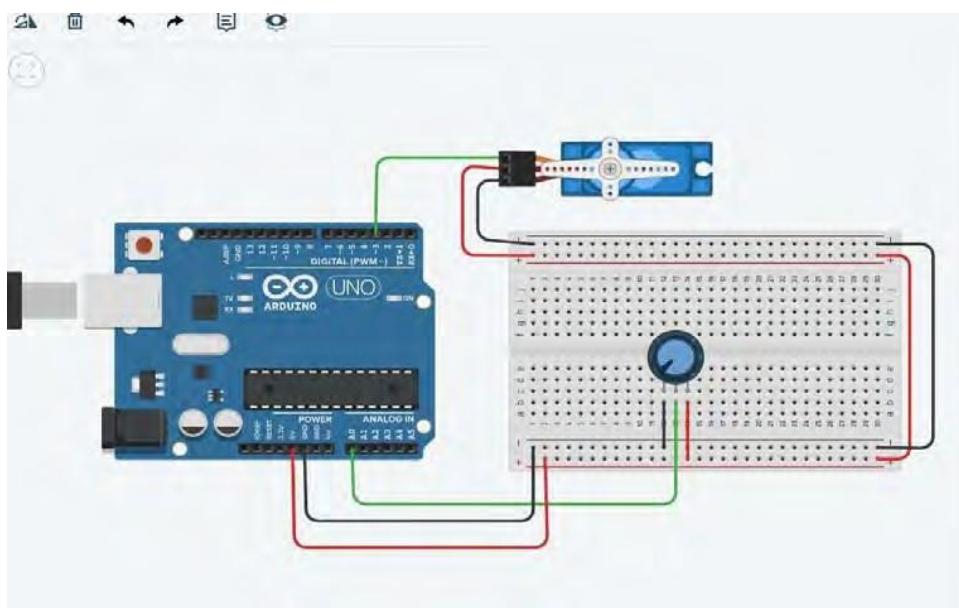
AIM

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

## **HARDWARES REQUIRED**

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

## CIRCUIT DIAGRAM



## **WRITE-UP**

Exp - 15

Venkatesha Prasada (H)  
18M18CS124

Aim: Design Smart irrigation system.

Code: #include <servo.h>

```
servo myservo;  
int pos=0;  
int val=0;  
int sensorPin = A2;  
void setup(){  
    myservo.begin();  
    Serial.begin(9600);  
}  
void loop(){  
    sensorValue = analogRead(sensorPin);  
    Serial.println(sensorValue);  
    if(sensorValue > 500){  
        for(pos=0 ; pos <= 180 ; pos+=1){  
            myservo.write(pos);  
            delay(15);  
        }  
        for( pos=180 ; pos >= 0 ; pos-=1){  
            myservo.write(pos);  
            delay(15);  
        }  
        delay(1000);  
    }  
}
```

Chikat

## 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).

