

**Name – Parag Gattani**

Program No. – 01

Program Title – LED Blinking

---

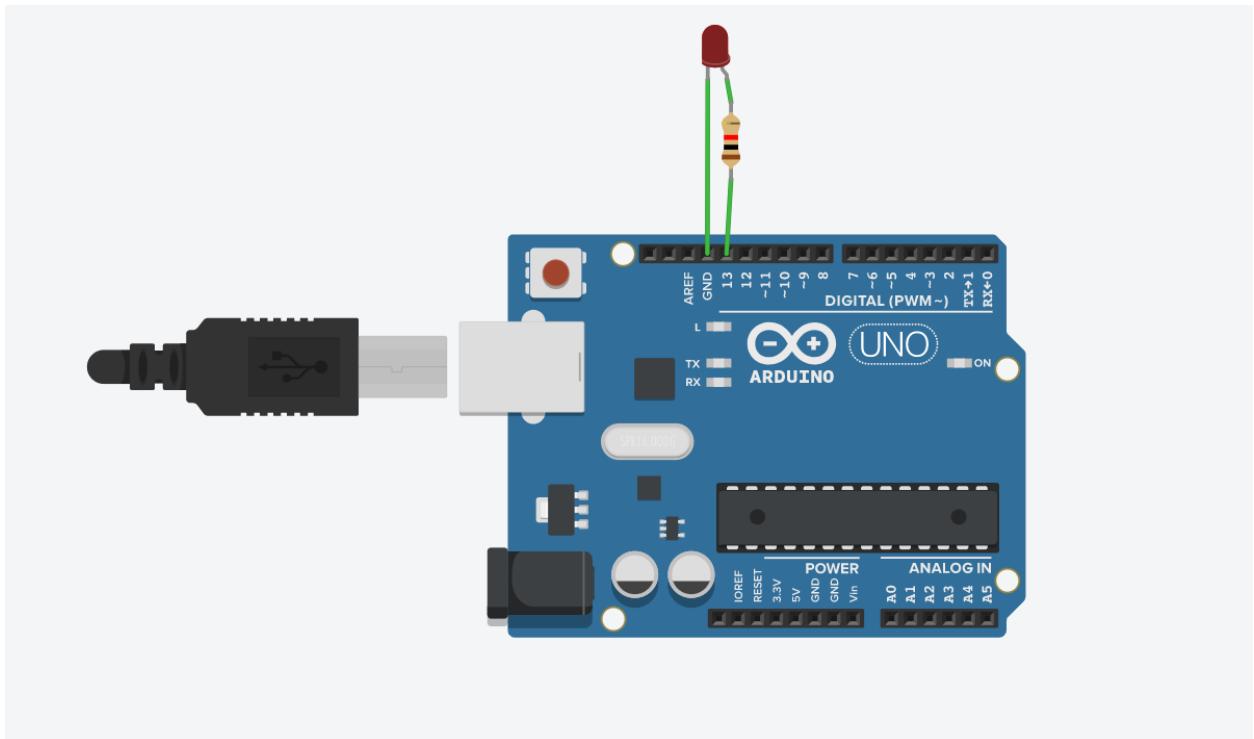
## **AIM**

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

## **HARDWARES REQUIRED**

- Arduino Board
- LEDs

## **CIRCUIT DIAGRAM**



## WRITE-UP

Name → Parag Gattani  
USN - 1BM18CS067

16/9/2020

Exp. No.-1.

LED Blanking

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

Hardware Required :-

- Arduino Board
- LEDs.

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 – Parag Gattani**

Program No. – 02

Program Title – Traffic Controller

---

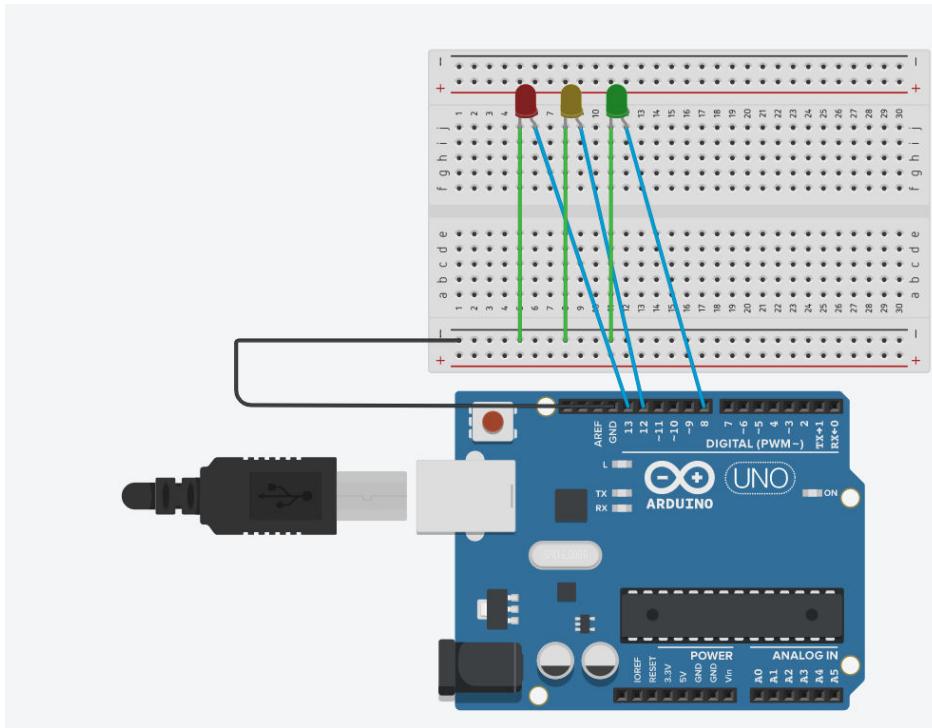
## **AIM**

Traffic Signal Simulator.

## **HARDWARES REQUIRED**

- Arduino Board
- LEDs
- Breadboard

## **CIRCUIT DIAGRAM**



## **WRITE-UP**

Name - Parag Gottani  
USN - 18M18CS067

16/09/2020

Exp. No. - 2

### TRAFFIC CONTROLLER

Aim:- Traffic signal Simulator.

Hardware Required:-

- Arduino Board
- LEDs
- Bread Board

CODE :-

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

void red ()
{
    digitalWrite (3, 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);
}
```

Name - Parag Gattani  
USN - 18M18CS067

111

void loop()  
{

red();  
delay(3000);  
yellow();  
delay(1500);  
green();  
delay(3000);  
yellow();  
delay(1500);

}

## 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 – Parag Gattani**

Program No. – 03

Program Title – LED fading without potentiometer

---

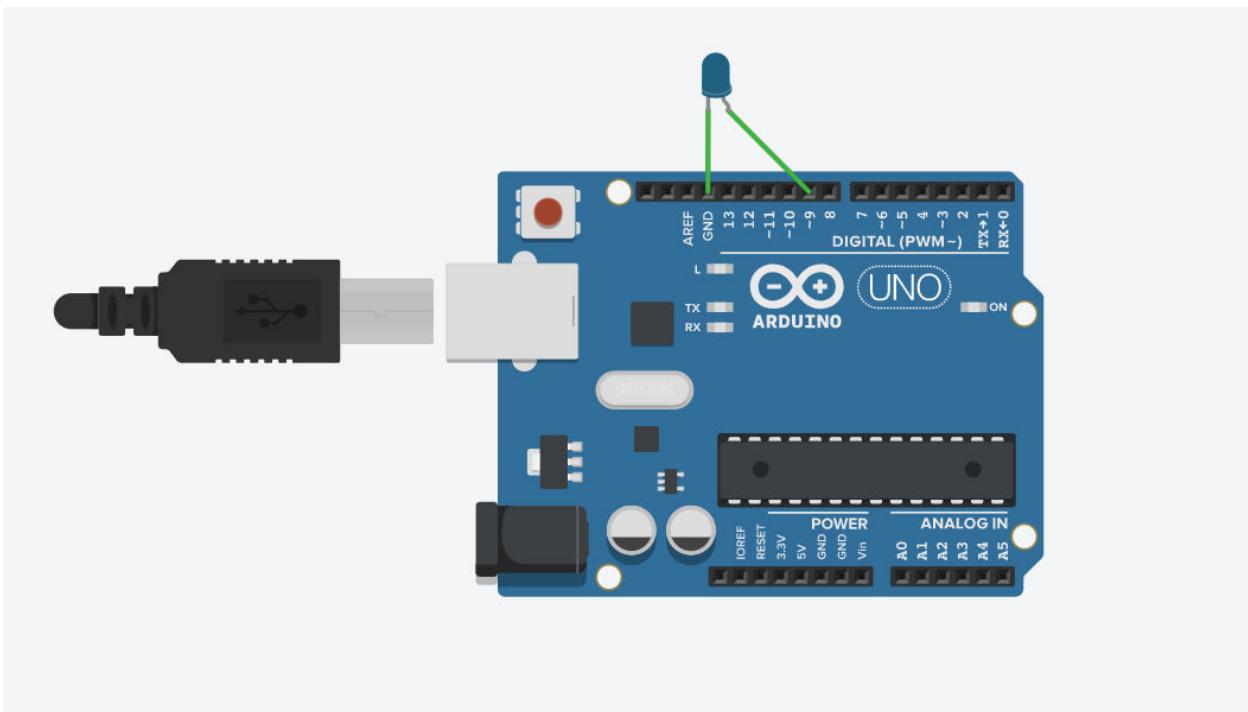
## **AIM**

Demonstrate to show LED fading.

## **HARDWARES REQUIRED**

- Arduino Board
- LED bulb

## **CIRCUIT DIAGRAM**



## WRITE-UP

Name - Panag Gattani  
USN - 1BM18CS067

23 / 09 / 2020

Exp. 4

### ★ LED FADING

Aim :-

Demonstrate to show LED fading.

Hardware - Required :-

Arduino Board

LED Bulb

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);
    }
}
```

## 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 – Parag Gattani**

Program No. – 04

Program Title – LED fading with potentiometer

---

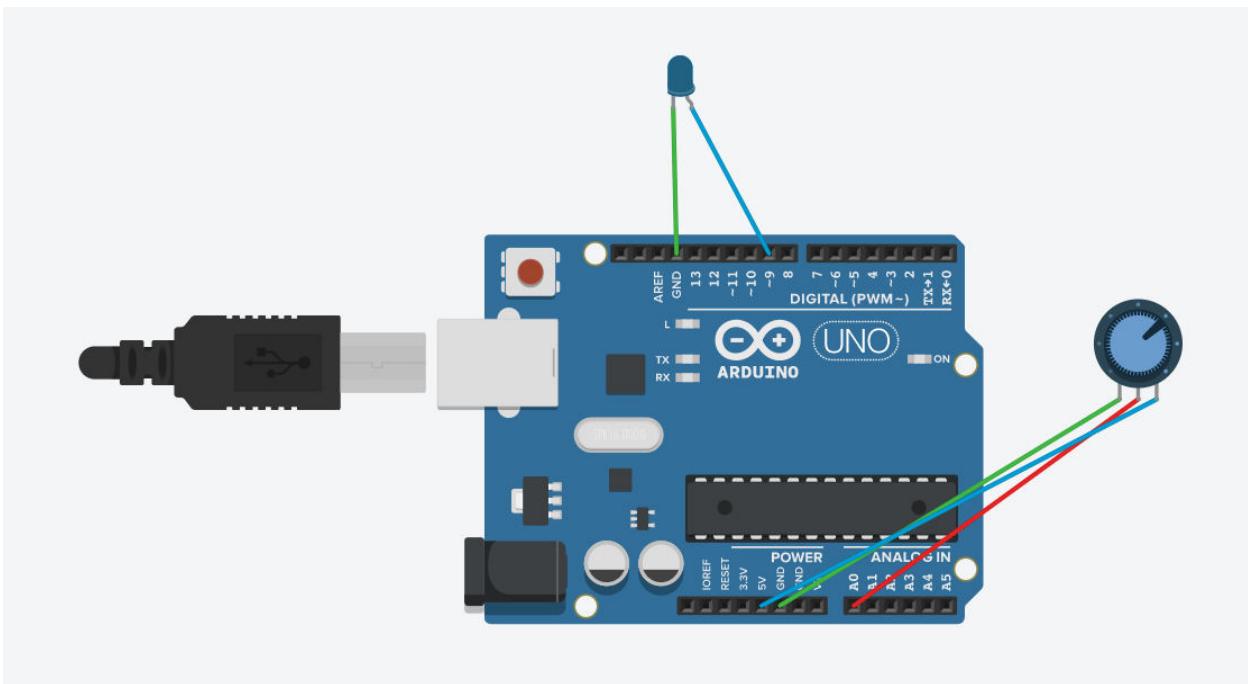
## AIM

Demonstrate to show LED fading(analog output).

## HARDWARES REQUIRED

- Arduino Board
- LED bulb
- Potentiometer

## CIRCUIT DIAGRAM



## **WRITE-UP**

Name - Parag Gattani  
USN - SBM18CS067

23/09/2020

Exp-3

## LED FADING WITH POTENTIOMETER

Aim :-

To Demonstrate ON/OFF of an LED Using Pushbutton



Demonstrate to show LED Fading (Analog Output)

Hardware Required :-

Arduino Board

Potentiometer

LED Bulb

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);  
}
```

## 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 – Parag Gattani**

Program No. – 05

Program Title – ON/OFF LED using Push Button

---

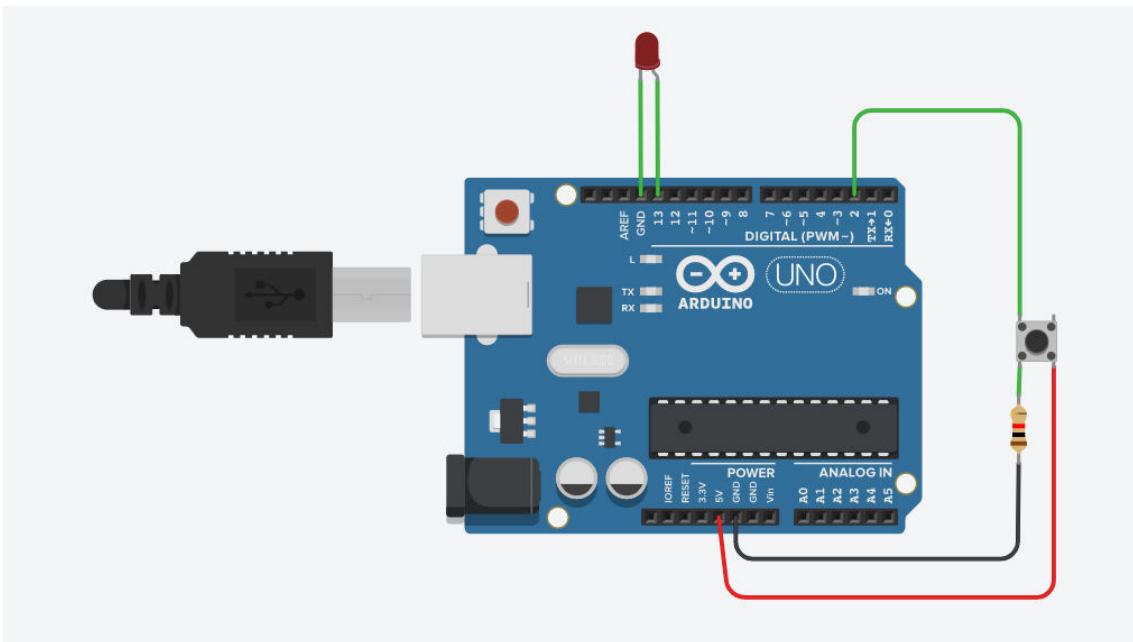
## **AIM**

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

Name - Parag Gattani  
USN - 1BM18CS067

23/09/2020

Exp. No. 5

ON/OFF LED using pushbutton

Aim:-

Demonstrate to show ON/OFF of a LED using pushbutton (Digital Output)

Hardware Required:-

Arduino Board

LED Bulb

Push Button

Resistor

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);  
    }  
}
```

## **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 – Parag Gattani**

Program No. – 06

Program Title – Fire Alarm using flame Sensor

---

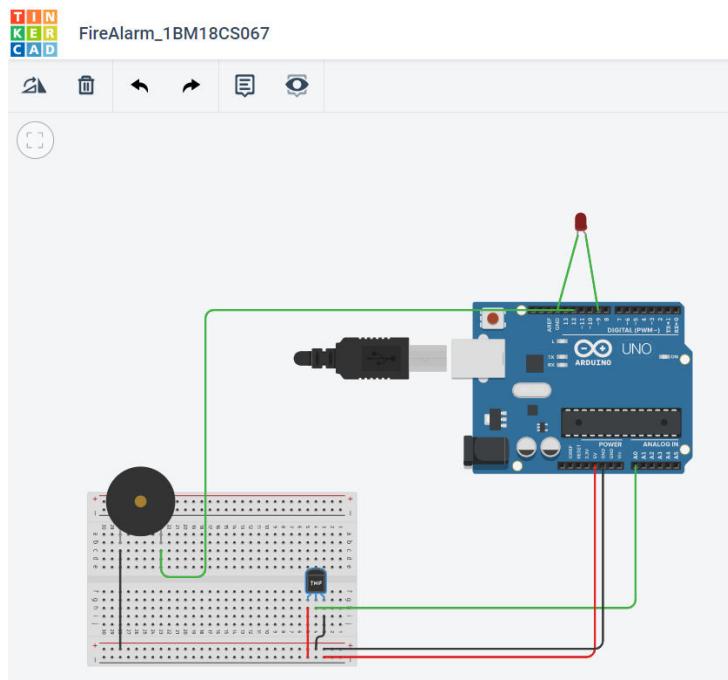
## **AIM**

Design an alert system using a flame sensor.

## **HARDWARES REQUIRED**

- Arduino Board
- Piezo
- Temperature Sensor
- Breadboard small

## **CIRCUIT DIAGRAM**



## WRITE-UP

Name - Parag Pattni  
USN - 1BM18CS067

07/10/2020

Topic:

Flame Sensor

Aim:

Design an alert system using flame sensor (temp. sensor).

Hardware Required:

Arduino Board

Breadboard Small

Jumper

Temp. Sensors.

Code:

```
const int temperaturePin = 0;  
int buzzer = 12;
```

```
void setup()
```

```
{
```

```
pinMode(buzzer, OUTPUT);
```

```
}
```

```
void loop()
```

```
{
```

```
float voltage, degreesC;
```

```
voltage = getVoltage(temperaturePin);
```

```
degreesC = (voltage - 0.5) * 100.0;
```

```
if (degreesC > 37)
```

```
digitalWrite(buzzer, LOW);
```

```
tone(12, 10000, 100);
```

```
}
```

```
}
```

float getVoltage (int pen)  
{  
 return (analogRead(pen) \* 0.004882814);  
}

## 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 – Parag Gattani**

Program No. – 07

Program Title – Distance Measurement using ultrasonic sensor

---

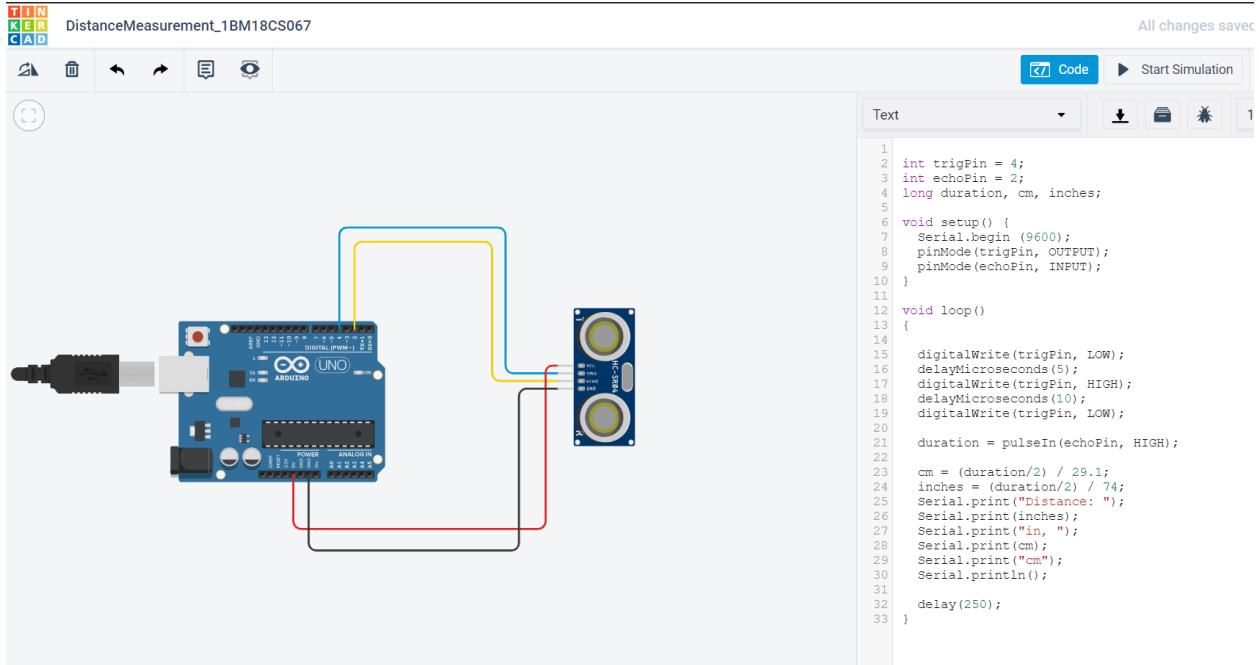
## AIM

Design a system to measure the distance between objects.

## HARDWARES REQUIRED

- Arduino Board
- Ultrasonic sensor HC-SR04

## CIRCUIT DIAGRAM



## WRITE-UP

Name - Parag Gattani  
VSN - 1B M18CS067

07/10/2020

exp.

### DISTANCE MEASUREMENT

Aim

Design a system to measure the distance between objects.

Hardware Required

Arduino Board

Ultrasonic distance measure. HC - S04

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(echoPin, 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 (" in, ");  
Serial.print (cm);  
Serial.print (" cm");  
Serial.println();

delay (250);

}

## 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.

**Name – Parag Gattani**

Program No. – 08

Program Title – LDR

---

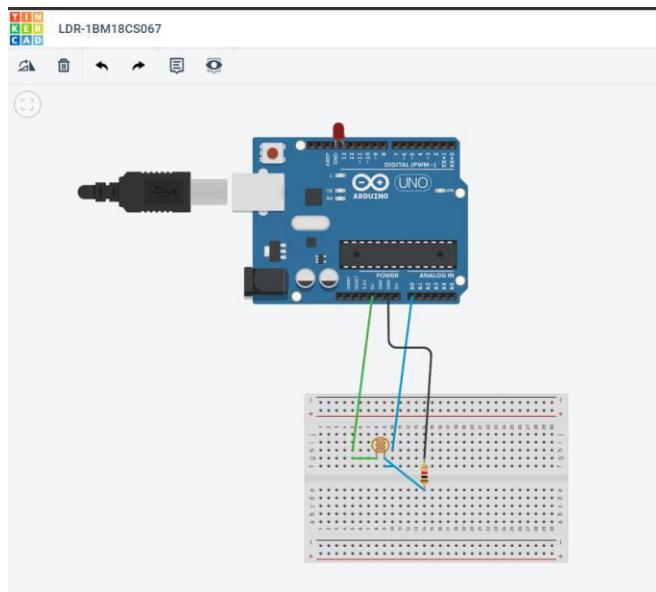
## **AIM**

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

Name - Parag Cattani  
UGN - IBMJBCS067

30/09/2020

Exp. 6

Aim :-

Demonstrate to show ON/OFF of a LED using LDR using LDR - Night light simulation.

Hardware Requirements.

Arduino Board

Photodiode

Resistor

LED

Breadboard small

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(" --- ");  
    }  
}
```

## 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 –Parag Gattani**

Program No. – 09

Program Title – PIR

---

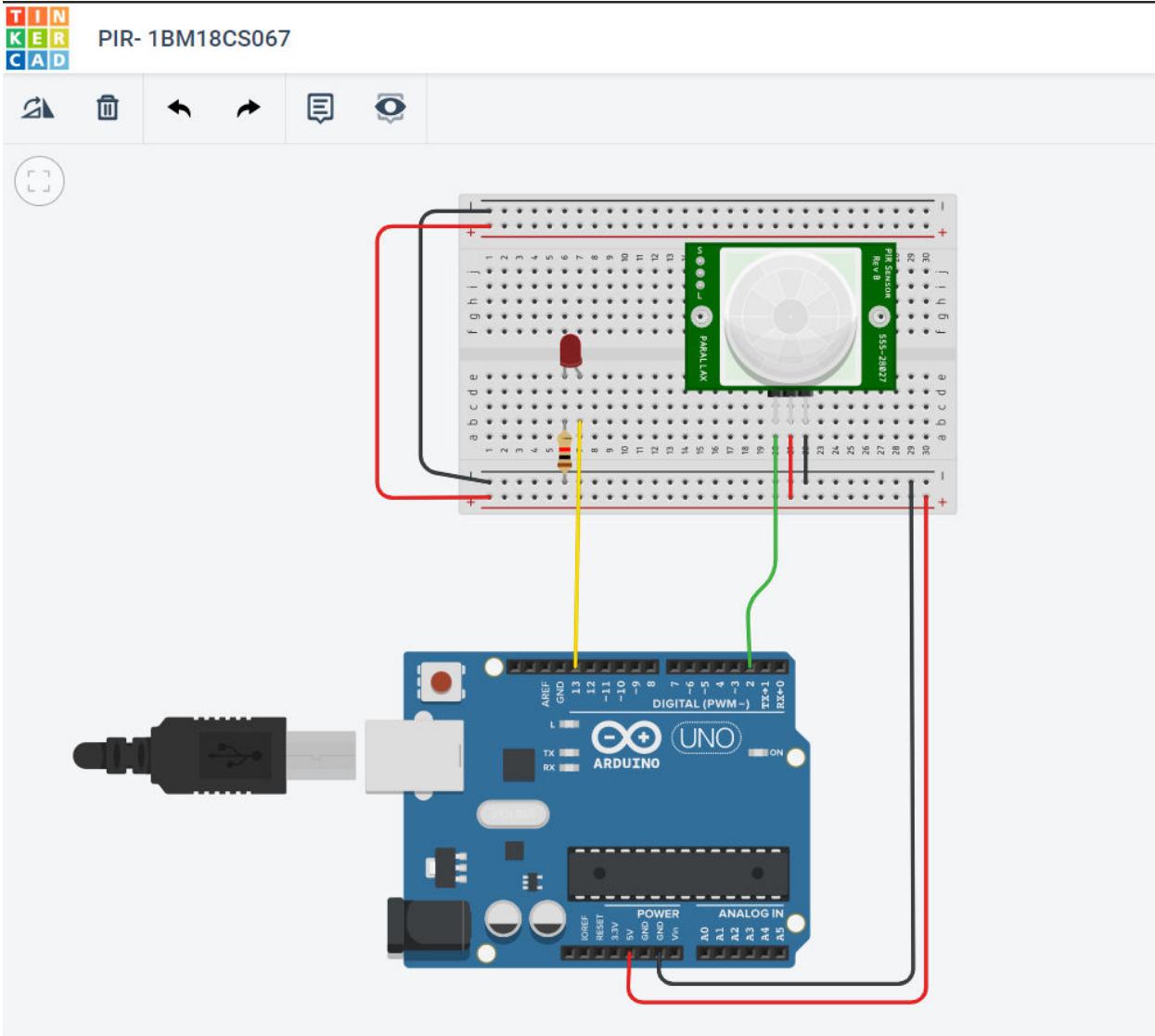
## **AIM**

Demonstrate to show working of PIR sensor.

## **HARDWARES REQUIRED**

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

## CIRCUIT DIAGRAM





## WRITE-UP

Name - Panag Gattani  
USN - 18M18C067

30/09/2020

Exp. 7.

Aim:-

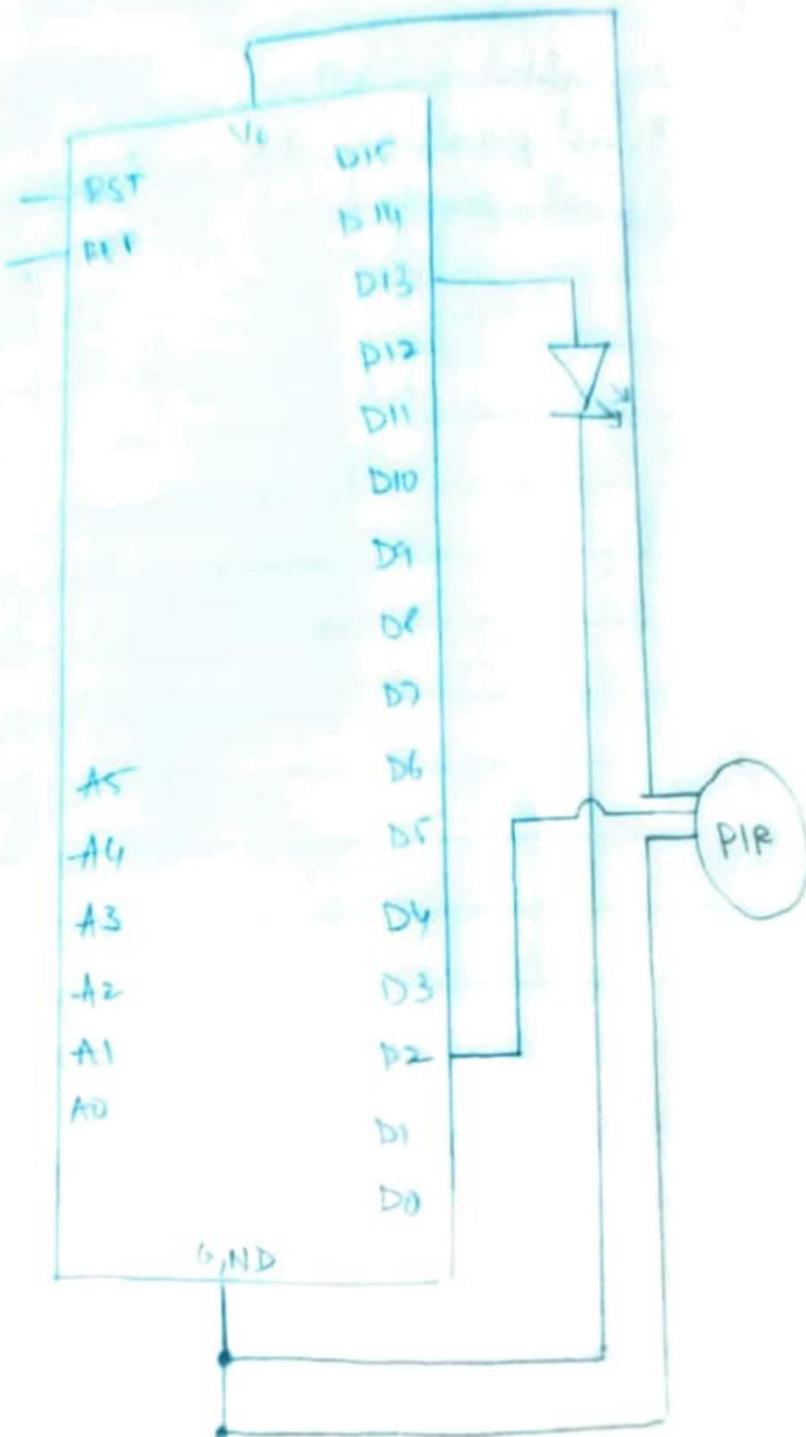
Demonstrate to show the working of PIR sensor.

Hardware Required:-

LED Pin, Arduino Board, Breadboard, Resistor,  
PIR Sensor.

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);  
    }  
    delay (10);  
}
```



## **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 – Parag Gattani**

Program No. – 10

Program Title – Gas Sensor

---

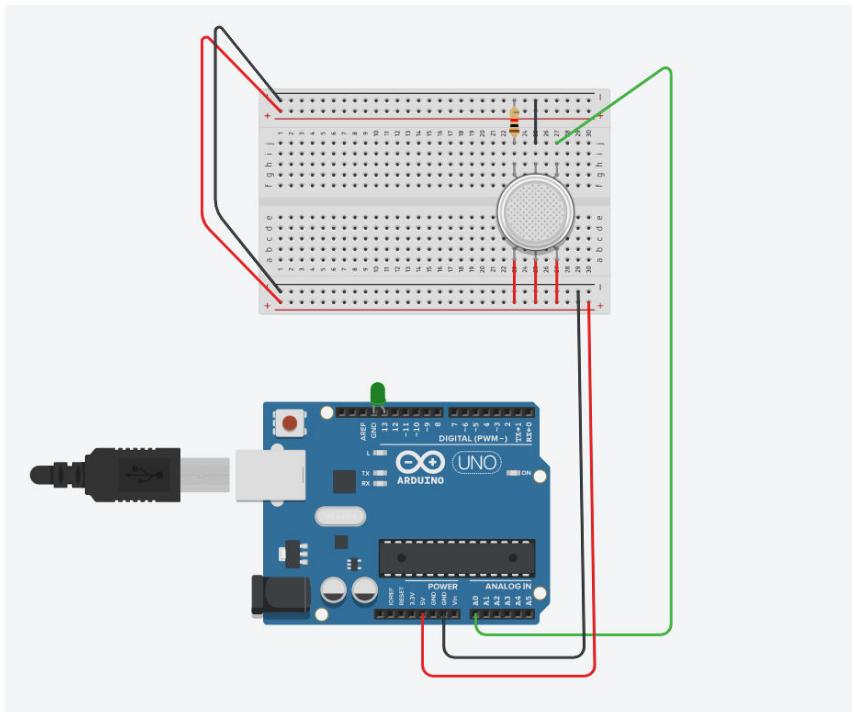
## **AIM**

To design a smart gas leakage indicator system.

## **HARDWARES REQUIRED**

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

## **CIRCUIT DIAGRAM**



## **WRITE-UP**

Name - Parag Gattani  
USN - 1BM18CS067

14 / 10 / 2020

Exp.

## Gas Sensor and LED

Aim

To design a smart gas leakage indicator system.

Hardware required

Gas Sensor, Arduino Board  
LED, resistor, Breadboard

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);  
    }  
}
```

```
digitalWrite(LED, low);  
Serial.print("Serial Value: ");  
Serial.println(serialValue);  
}  
delay(1000);
```

## 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 – Parag Gattani**

Program No. – 11

Program Title – TEMPERATURE SENSOR (LM35)

---

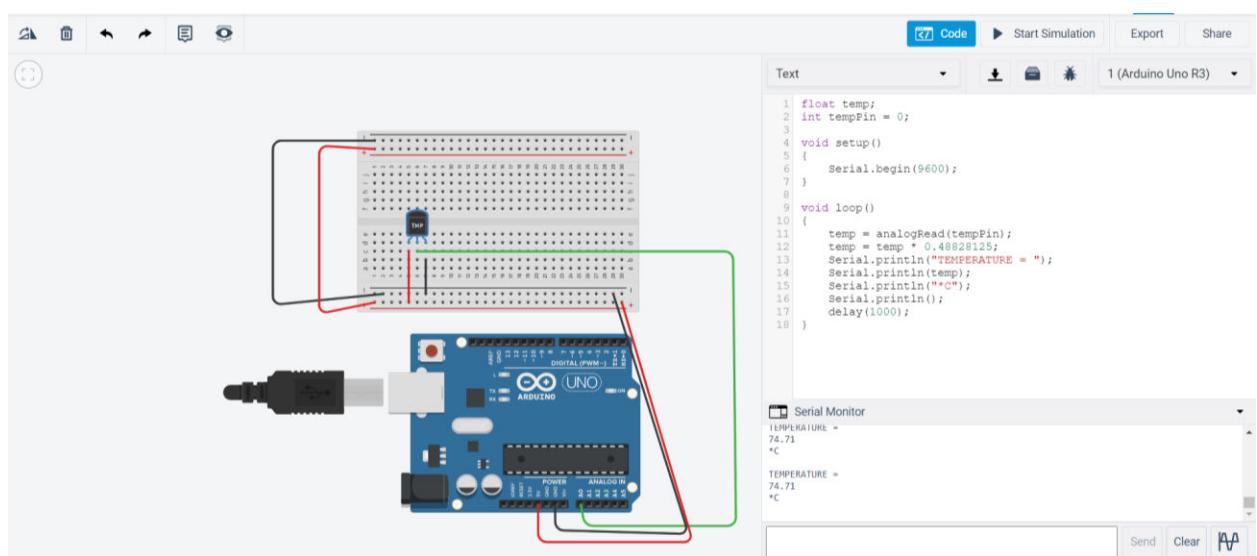
## AIM

Demonstrate and make your own temperature sensor by Arduino and LM35 sensor

## HARDWARES REQUIRED

- Arduino Board
- Breadboard
- LM35 Temperature Sensor

## CIRCUIT DIAGRAM



## **WRITE-UP**

Name - Parag Gattani  
USN - 1BM18CS067

07/10/2020

Topic:

Flame Sensor

Aim

Design an alert system using flame sensor (temp sensor).

Hardware Required.

Arduino Board

Breadboard Small

Jumper

Temp. Sensors.

Code

```
const int temperaturePin = 0;
```

```
int buzzer = 12;
```

```
void setup()
```

```
{
```

```
pinMode(buzzer, OUTPUT);
```

```
}
```

```
void loop()
```

```
{
```

```
float voltage, degreesC;
```

```
voltage = getVoltage(temperaturePin);
```

```
degreesC = (voltage - 0.5) * 100.0;
```

```
if (degreesC > 37)
```

```
digitalWrite(buzzer, LOW);
```

```
tone(12, 10000, 100);
```

```
{
```

```
}
```

— / —

```
float getVoltage (smt pen)
{
    return (analogRead (pen) * 0.004882814);
```

## CODE

```
float temp;  
int tempPin = 0;  
  
void setup()  
{  
    Serial.begin(9600);  
}  
  
void loop()  
{  
    temp = analogRead(tempPin);  
    temp = temp * 0.48828125;  
    Serial.println("TEMPERATURE = ");  
    Serial.println(temp);  
    Serial.println("*C");  
    Serial.println();  
    delay(1000);  
}
```

## **OUTPUT**

Temperature is being measured.

**Name – Parag Gattani**

Program No. – 12

Program Title – Vibration motor and LDR

---

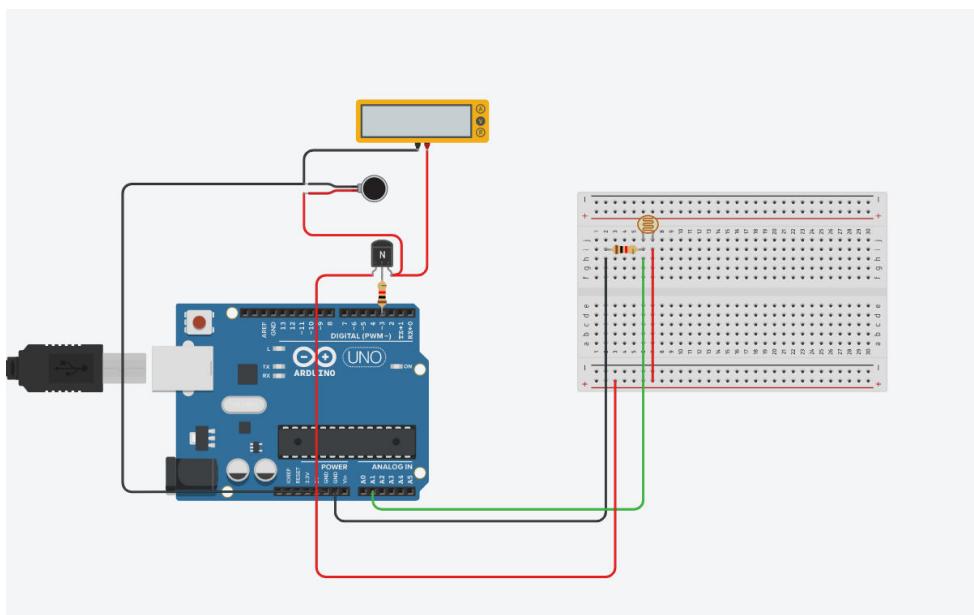
## **AIM**

To design an automated day indicator system.

## **HARDWARES REQUIRED**

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

## **CIRCUIT DIAGRAM**



## WRITE-UP

Name - Panag Gattani  
USN- 18M18CS067

14/10/2020

Exp.

Vibration Motor and LDR

Aim

Design an automated day indicator system.

Hardware Required

NPN Transistor, Resistor, Vibration Motor,  
Multimeter, Photoresistor, Arduino Board.

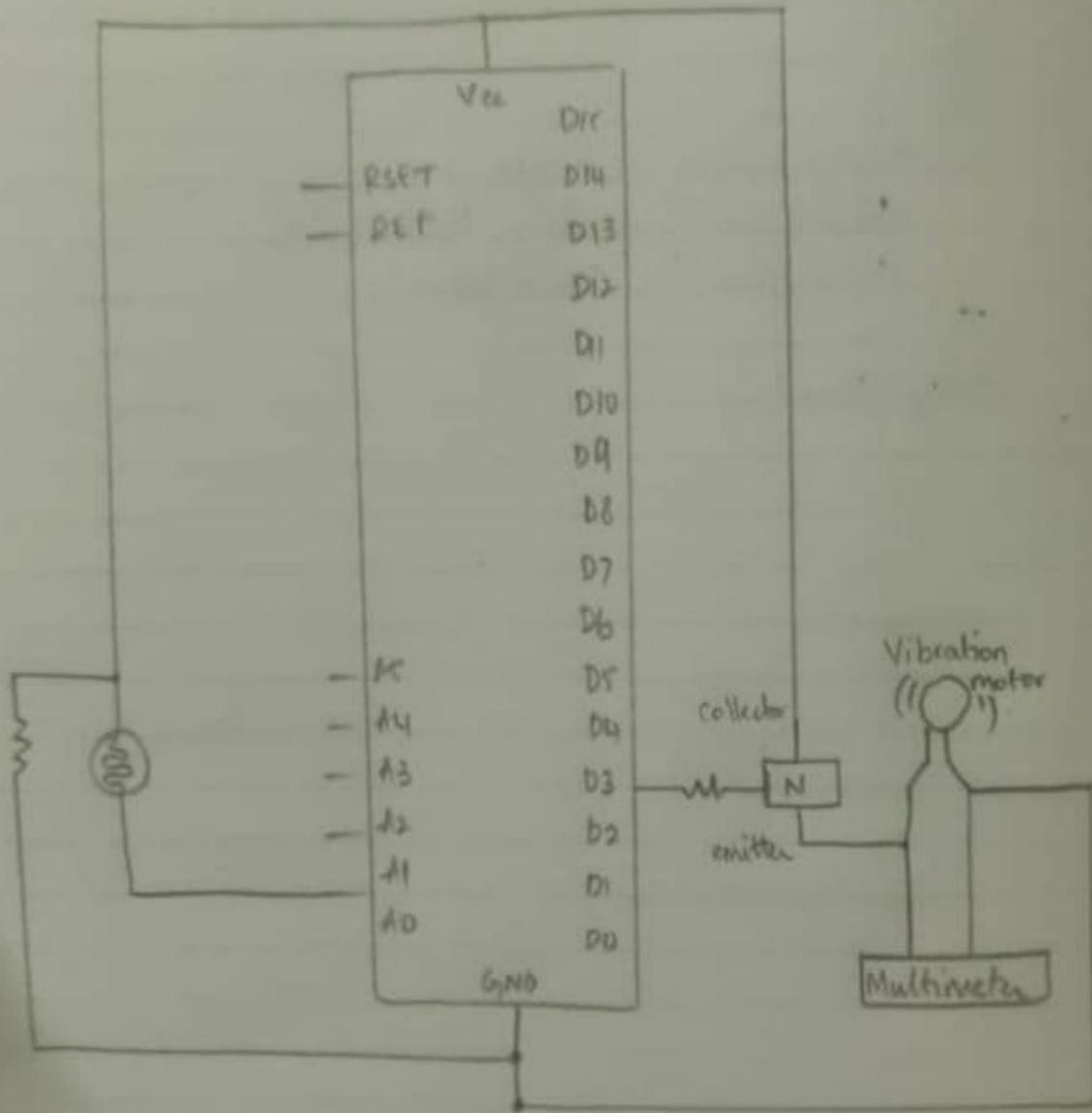
Code

```
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);  
    }
```



## 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
```

```

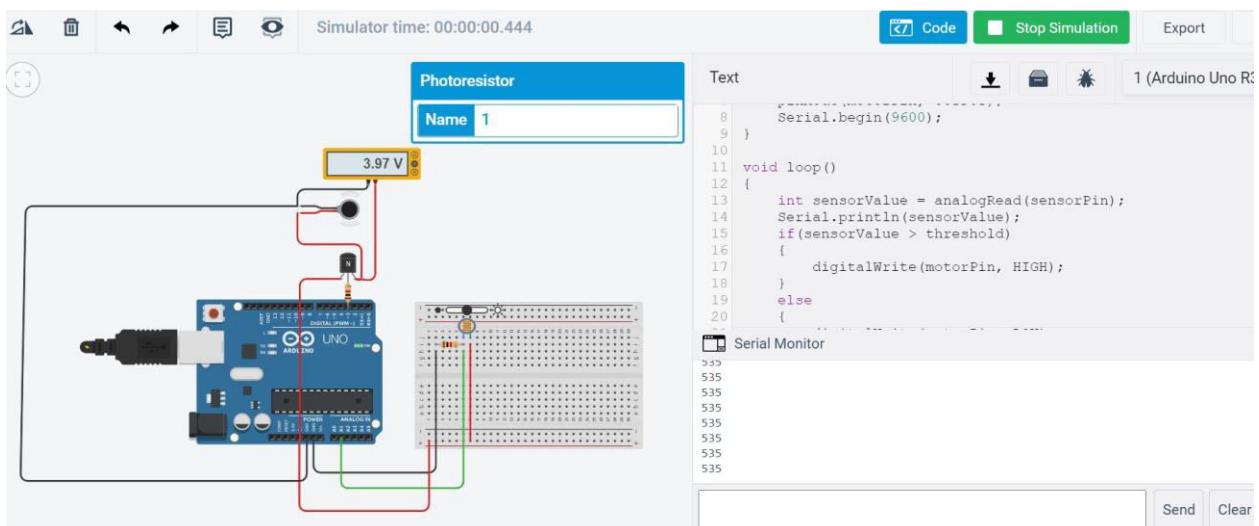
    {
        digitalWrite(motorPin, LOW);
    }

}

```

## OUTPUT

Designed an automated day indicator system.



**Name – Parag Gattani**

Program No. – 13

Program Title – Tilt Sensor

---

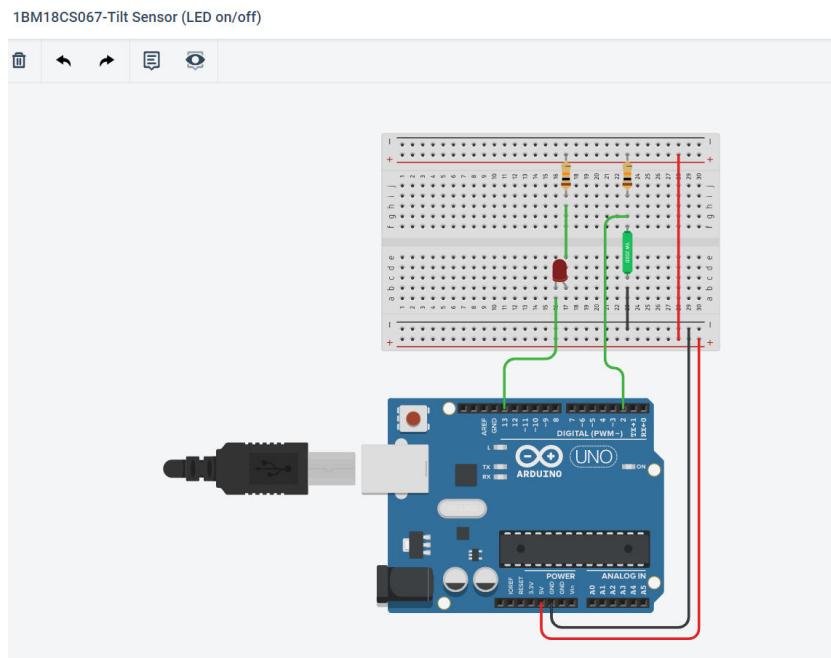
## **AIM**

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

## **HARDWARES REQUIRED**

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

## **CIRCUIT DIAGRAM**



## WRITE-UP

Name - Parag Gattane  
USN - 1BM18CS067

28/10/2020

Exp-12

Tilt Sensor

Aim :-

Design a Smart Package handling system.

Hardware Required:-

Arduino Board, Breadboard Small, LEDs,  
Tilt Sensor, Resistor.

Code:-

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

## **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 – Parag Gattani**

Program No. – 14

Program Title – IR based SERVO Motor controller

---

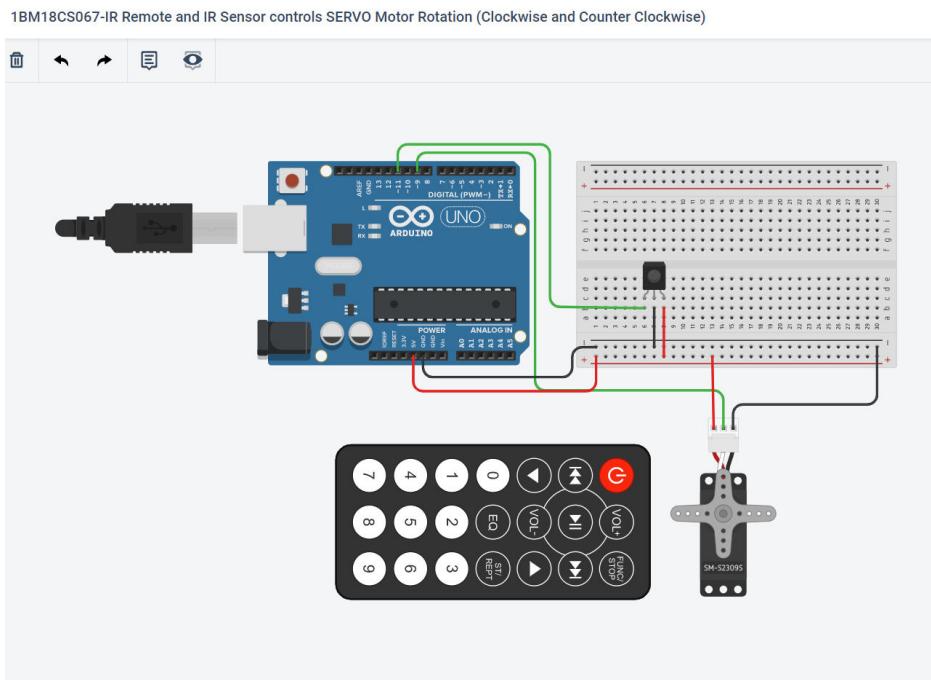
## **AIM**

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

Name - Parag Gattani  
USN - 1BM18CS067

28/10/2020

exp. 13

IR based SERVO Motor Controller.

Aim.

Design IR based SERVO Motor Controller

Hardware Required:-

Arduino Board, Breadboard, Small, IR Sensor,  
IR Remote, MicroServo

Code:-

```
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 0xFDDOFF:
```

```
       myservo.attach(9);  
       Serial.println("start");  
       break;
```

— / —

Case 0xFD609F:

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

Case 0xFD60DF:

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

default:

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

```
xServo.release();
```

```
}
```

## 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.

**Name – Parag Gattani**

Program No. – 15

Program Title – Irrigation

---

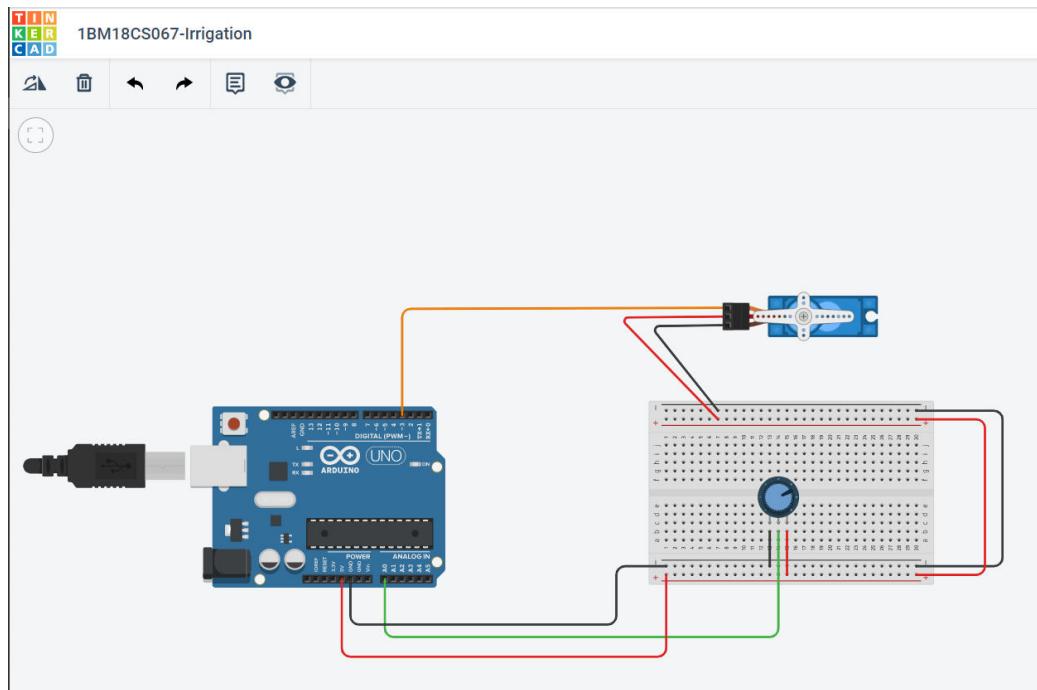
## **AIM**

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

## **HARDWARES REQUIRED**

- Arduino Board, Breadboard Small, Potentiometer
- Micro Servo

## **CIRCUIT DIAGRAM**



## WRITE-UP

Name - Panag Gattani  
USN - IBM18CS067

04 / 11 / 2020

Exp. 14

Invigation

Aim

Design a display system to print the RED, BLUE, and GREEN Colors (RGB LED and LCD)

Hardware Required

Arduino Board, Breadboard, Potentiometer, Microservo

Code

```
#include<Servo.h>
Servo myservo;
int pos=0;
int sensorPin = A0;
int sensorValue = 0

void setup()
{
    myservo.attach(3);
    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)

    mystwo.write(pos);  
    delay(15);

} delay(1000);

}

## 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 the
position

}

for (pos = 180; pos >= 0; pos -= 1) { // goes from 180 degrees to 0
degrees

    myservo.write(pos);           // tell servo to go to position in variable
'pos'

    delay(15);                  // waits 15ms for the servo to reach the
position

}

}

delay (1000);

}

```

## OUTPUT

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

**Name – Parag Gattani**

Program No. – 16

Program Title – RGB LED

---

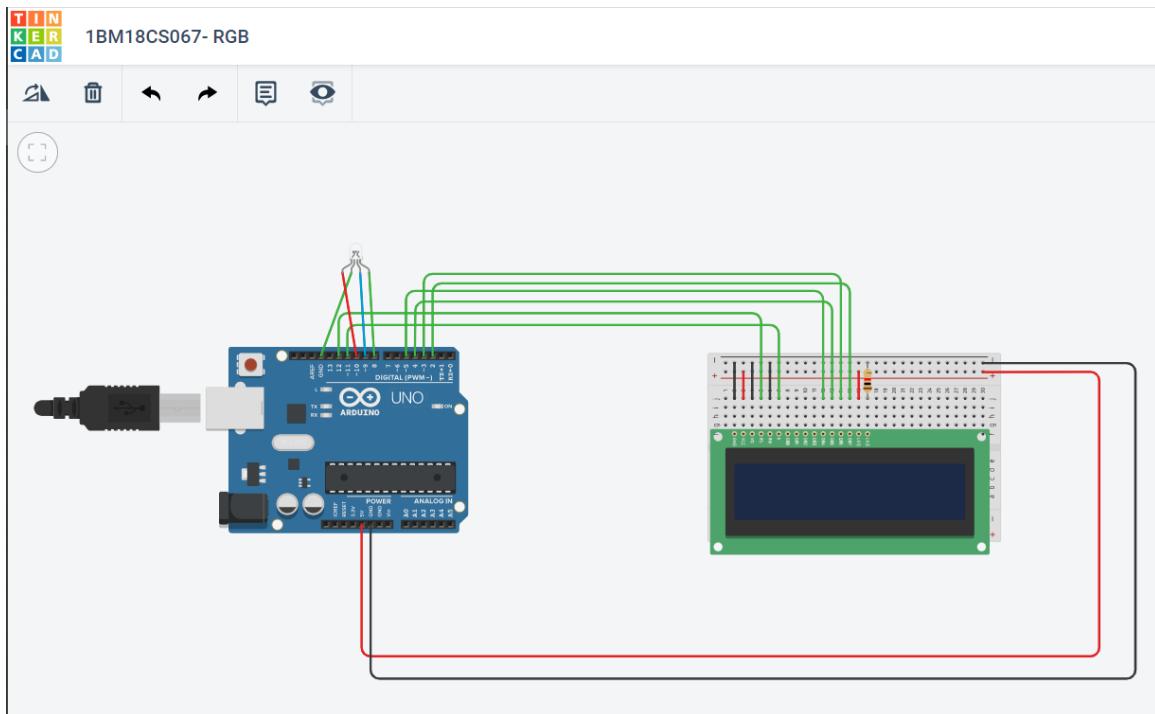
## **AIM**

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

## **HARDWARES REQUIRED**

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

## **CIRCUIT DIAGRAM**



## WRITE-UP

Name - Paavag Gattani  
USN-1 BM18S062

4/11/2020

Exp-15

RGB LED

Aim

Design a smart irrigation system.

Hardware Required.

Arduino Board, Breadboard Small, LED RGB,  
LCD 16x2, Resistor.

Code

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

```
int red_light_pin = 10;
int green_light_pin = 8;
int blue_light_pin = 9;
```

```
void setup()
```

```
{  
    pinMode(red_light_pin, OUTPUT);  
    pinMode(green_light_pin, OUTPUT);  
    pinMode(blue_light_pin, OUTPUT);  
}
```

```
void loop()
```

```
{  
    lcd.setCursor(0, 0);  
    RGB_color(255, 0, 0);  
    lcd.print("RED");  
    delay(1000);  
    lcd.clear();  
}
```

RGB-color(0, 0, 255);  
lcd.print("Blue");  
delay(1000);  
lcd.clear();

RGB-color(255, 255, 255);  
lcd.print("WHITE");  
delay(1000);  
lcd.clear();

}

void RGB-color(int red, int green, int blue)  
{

analogwhite(red + light-pin, red);  
analogwhite(green-light-pin, green);  
analogwhite(blue-light-pin, blue);

}

## CODE

```
#include <LiquidCrystal.h>

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

//Parameters: (rs, enable, d4, d5, d6, d7)

int red_light_pin= 10;
int green_light_pin = 8;
int blue_light_pin = 9;

void setup() {

    pinMode(red_light_pin, OUTPUT);
    pinMode(green_light_pin, OUTPUT);
    pinMode(blue_light_pin, OUTPUT);

}

void loop() {

    lcd.setCursor(0,0);

    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(255, 255, 255); // White

    lcd.print("WHITE");

    delay(1000);

    lcd.clear();

}

void  RGB_color(int  red_light_value,  int  green_light_value,  int
blue_light_value)

{
```

```
analogWrite(red_light_pin, red_light_value);  
analogWrite(green_light_pin, green_light_value);  
analogWrite(blue_light_pin, blue_light_value);  
}
```

## OUTPUT

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