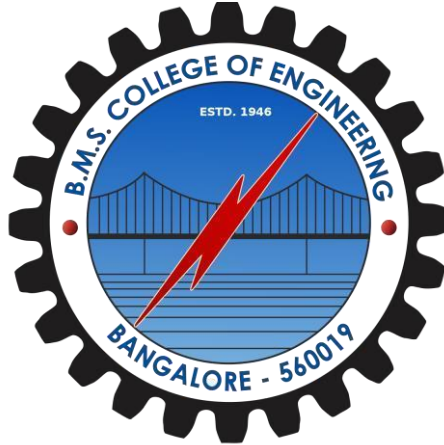


# IOT LAB MANUAL-2020



*Name: Varad Vithal KJ*

*USN: 1BM18CS122*

*CONSOLIDATED LAB PROGRAMS*

## Program no 00

### Program Title: LED-BLINK

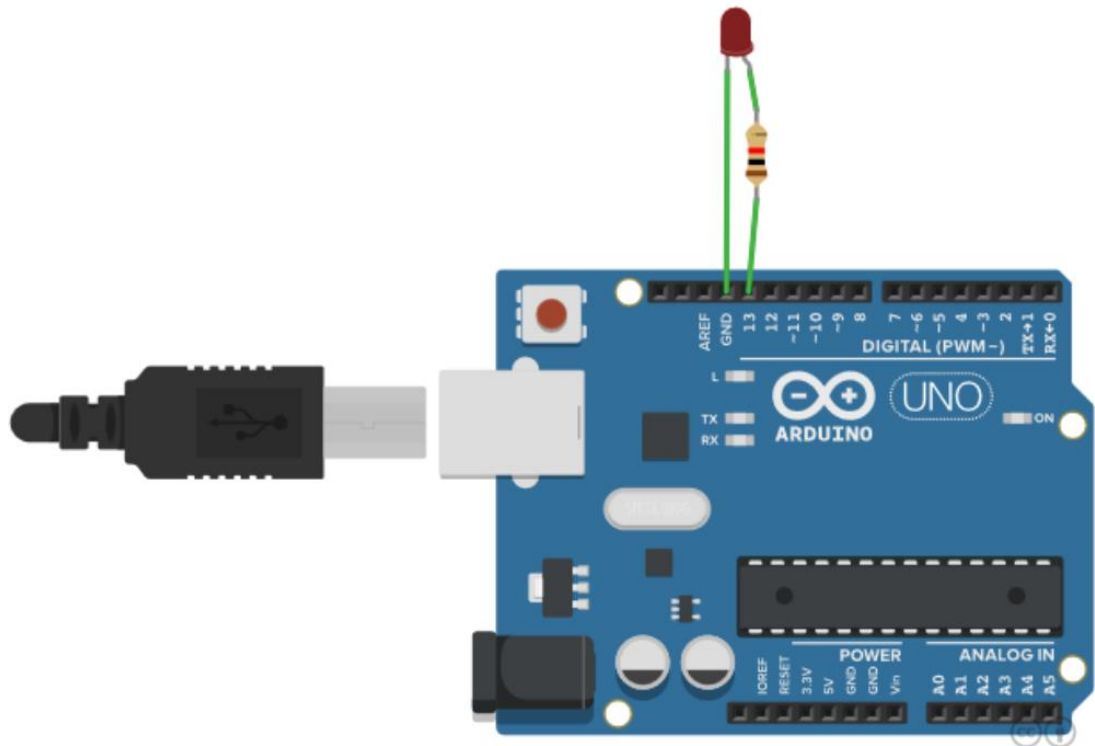
#### Aim

To demonstrate blinking of Led after delays

#### Hardware Required

- Arduino Board
- Led Light
- 1 ohm resistor

#### Circuit Diagram



#### Code:

```
int ledPin=13;
```

```
void setup() {  
  // put your setup code here, to run once:  
  pinMode(ledPin,OUTPUT);  
}
```

```
void loop() {  
  // put your main code here, to run repeatedly:  
  digitalWrite(ledPin,HIGH);  
  delay(1000);  
  digitalWrite(ledPin,LOW);  
  delay(1000);  
}
```

# Observation /Output:

The image displays three sequential screenshots of the Tinkercad web interface, illustrating the setup and execution of a Blink-Red circuit simulation.

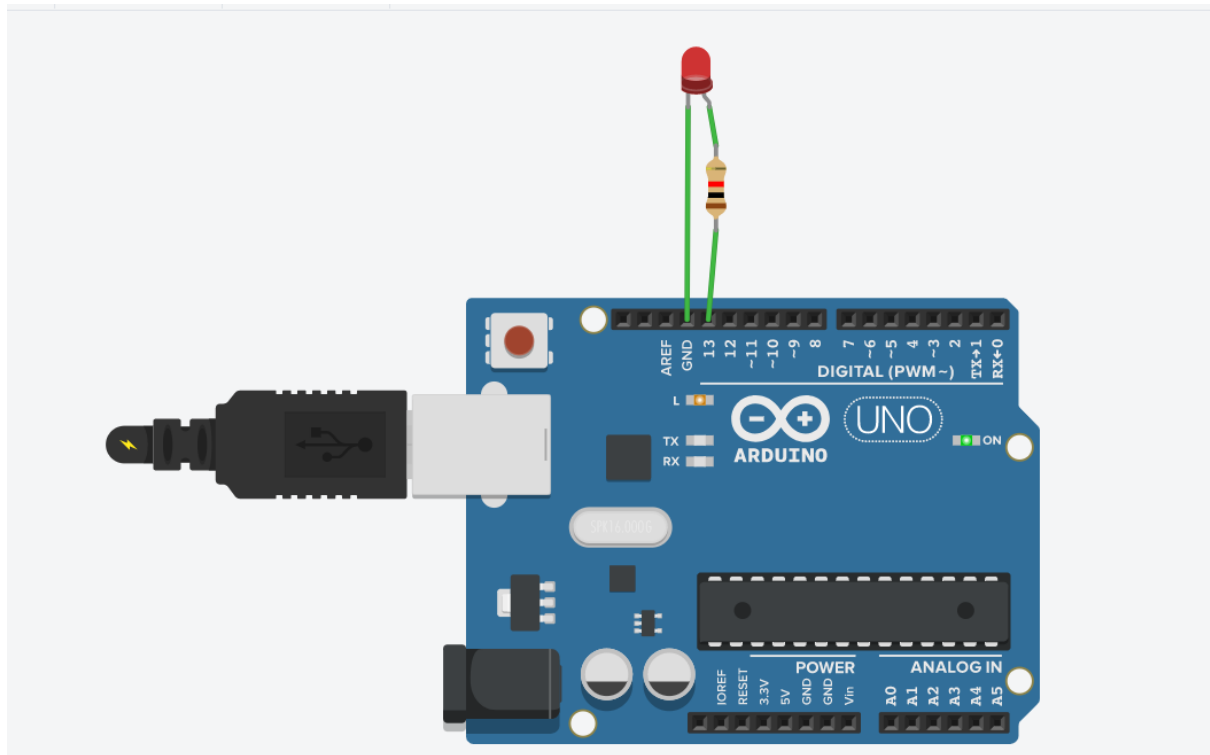
**Top Screenshot:** Shows the initial circuit setup. An Arduino Uno R3 is connected to a USB Type-A cable. A red LED is connected to digital pin 13, and a 220Ω resistor is connected between the LED's anode and pin 13, and its cathode to ground. The code editor on the right contains the following code:

```
1 int ledPin=13;
2
3 void setup() {
4   // put your setup code here, to run once:
5   pinMode(ledPin,OUTPUT);
6 }
7
8 void loop() {
9   // put your main code here, to run repeatedly:
10  digitalWrite(ledPin,HIGH);
11  delay(1000);
12  digitalWrite(ledPin,LOW);
13  delay(1000);
14 }
```

**Middle Screenshot:** Shows the simulation starting. The "Start Simulation" button is highlighted in green. The "Simulator time: 00:00:00" is displayed. The LED is currently off.

**Bottom Screenshot:** Shows the simulation in progress. The "Stop Simulation" button is highlighted in green. The "Simulator time: 00:00:05" is displayed. The LED is now on, indicating it has blinked.

The Led blinks after delays



Handwritten –

IBM18CS122

VARAD VITHAL K J

## TOT

### 1) LED - BLINK

```
int ledPin = 13;
```

```
void setup()
```

```
{
```

```
  pinMode(ledPin, OUTPUT);
```

```
}
```

```
void loop()
```

```
{
```

```
  digitalWrite(ledPin, HIGH);
```

```
  delay(1000);
```

```
  digitalWrite(ledPin, LOW);
```

```
  delay(1000);
```

```
}
```

Var



## Program no 02

### Program Title: PUSH-BUTTON

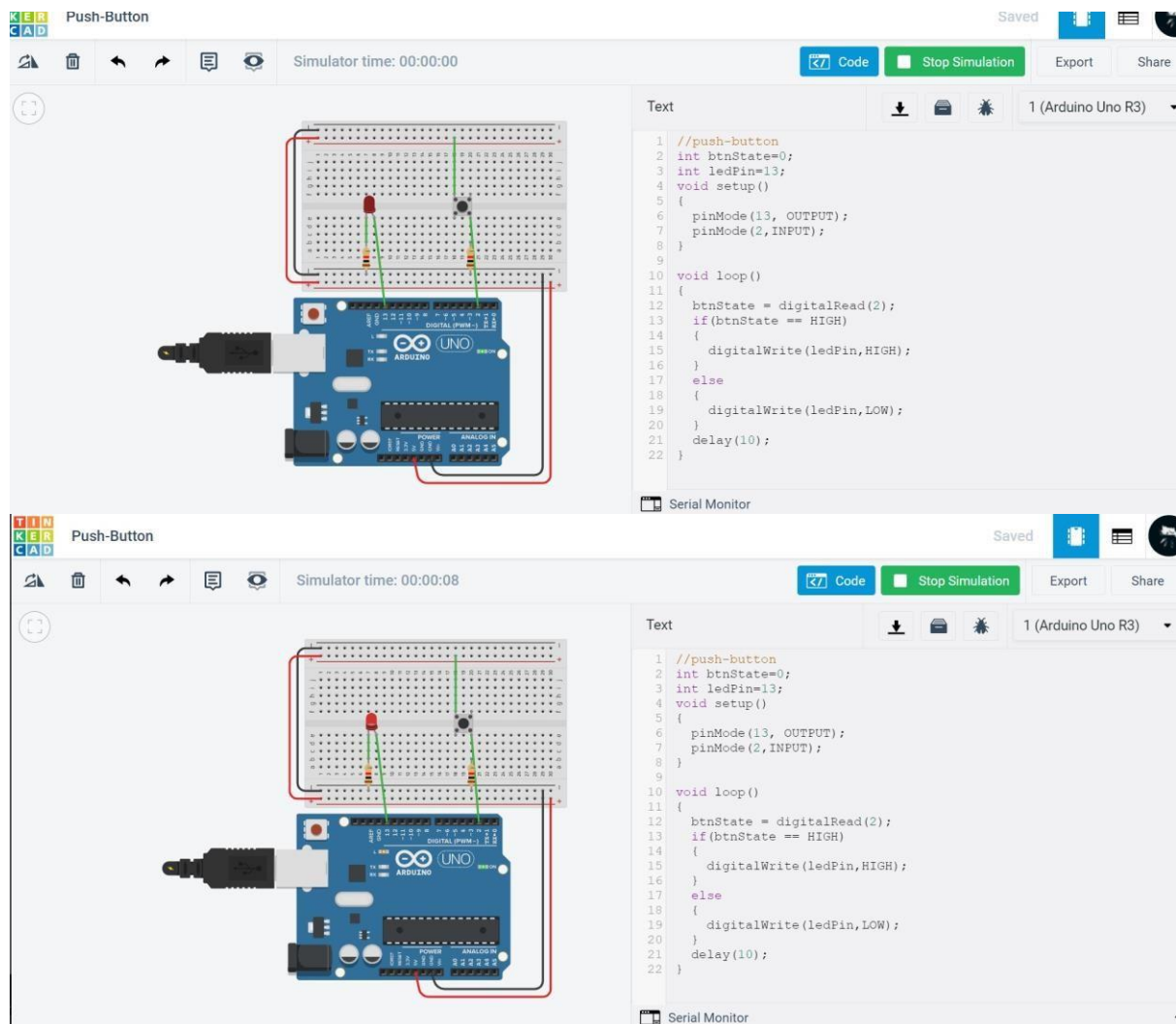
Name: Varad Vithal KJ, USN: 1BM18CS122

Aim : To demonstrate working of LED when button is pushed

#### Hardware Required

- Arduino Board
- Led Light
- 1 ohm resistor
- 1 Push Button

#### Circuit Diagram –





```
Code: //push-  
button int  
btnState=0; int  
ledPin=13; void  
setup()  
{  
    pinMode(13, OUTPUT);  
    pinMode(2,INPUT);  
}  
  
void loop()  
{  
    btnState = digitalRead(2);  
    if(btnState == HIGH)  
    {  
        digitalWrite(ledPin,HIGH);  
    }  
    else  
    {  
        digitalWrite(ledPin,LOW);  
    }  
    delay(10);
```

}

## Observation /Output:

Push-Button

Simulator time: 00:00:00

Code Stop Simulation Export Share

Text

```
1 //push-button
2 int btnState=0;
3 int ledPin=13;
4 void setup()
5 {
6   pinMode(13, OUTPUT);
7   pinMode(2, INPUT);
8 }
9
10 void loop()
11 {
12   btnState = digitalRead(2);
13   if(btnState == HIGH)
14   {
15     digitalWrite(ledPin,HIGH);
16   }
17   else
18   {
19     digitalWrite(ledPin,LOW);
20   }
21   delay(10);
22 }
```

Serial Monitor

1 (Arduino Uno R3)

Push-Button

Simulator time: 00:00:08

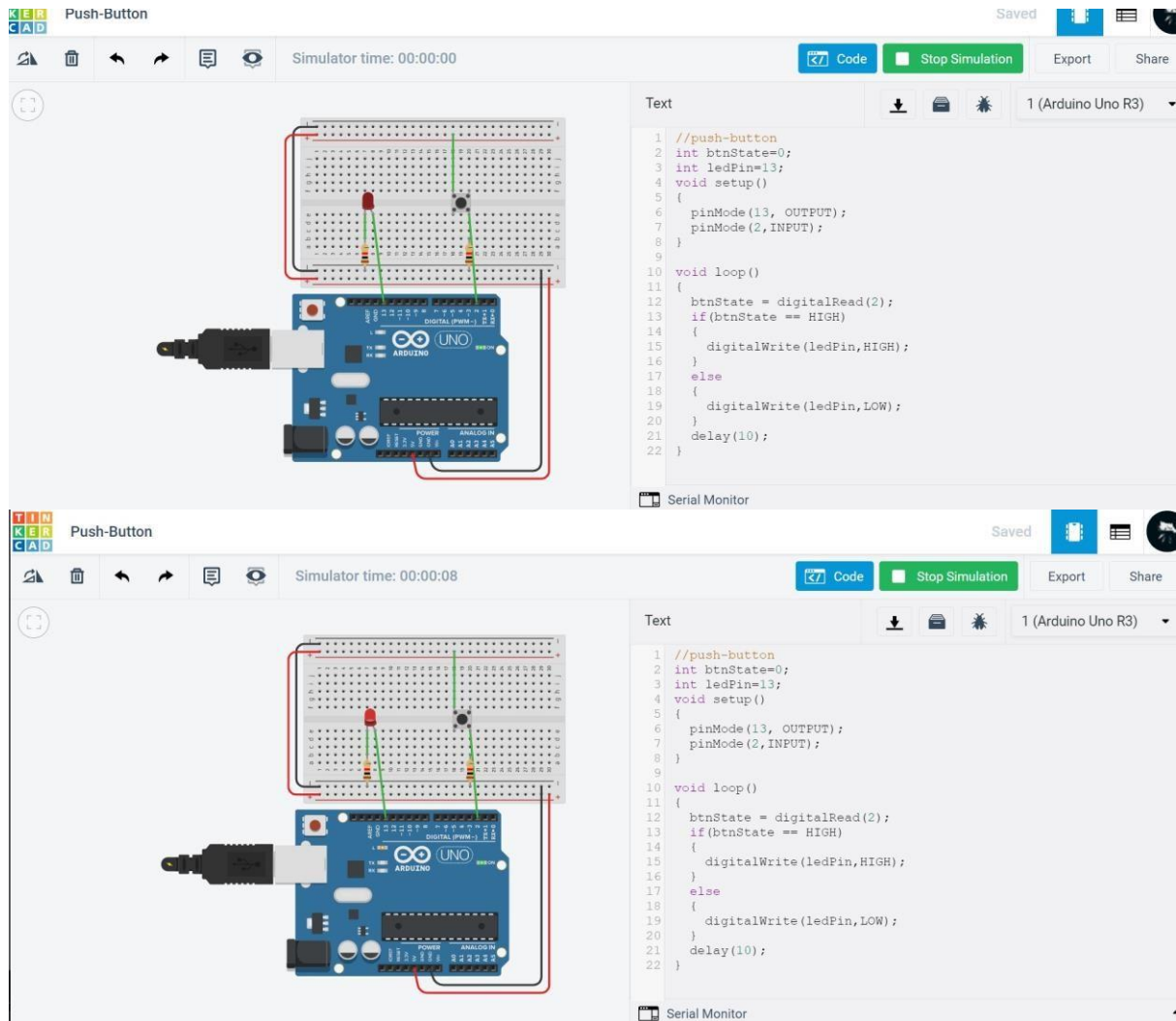
Code Stop Simulation Export Share

Text

```
1 //push-button
2 int btnState=0;
3 int ledPin=13;
4 void setup()
5 {
6   pinMode(13, OUTPUT);
7   pinMode(2, INPUT);
8 }
9
10 void loop()
11 {
12   btnState = digitalRead(2);
13   if(btnState == HIGH)
14   {
15     digitalWrite(ledPin,HIGH);
16   }
17   else
18   {
19     digitalWrite(ledPin,LOW);
20   }
21   delay(10);
22 }
```

Serial Monitor

1 (Arduino Uno R3)





## 3.) Push button

// push - button

int btn State = 0;

int led Pin = 13;

void Setup ()

{

Pin Mode (13, OUTPUT);

pinMode (2, INPUT);

}

void loop ()

{

btn State = digitalRead (2);

if (btn State == HIGH)

{

digitalWrite (led Pin, HIGH);

}

else

{

digitalWrite (led Pin, LOW);

}

delay (10);

}

✓

# IOT LAB - 5th Sem

Name : Varad Vithal KJ , USN : 11BM18CS122

To demonstrate working of LED fading in and out

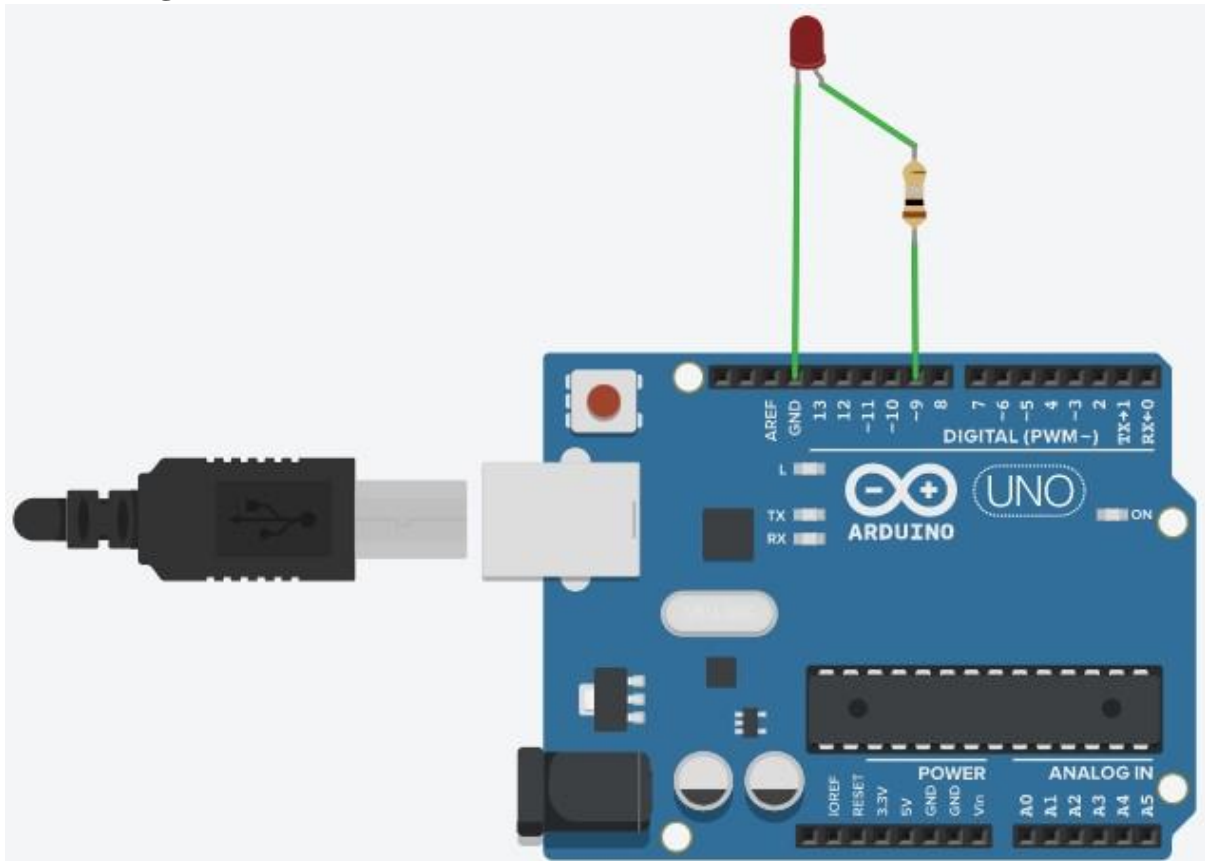
Aim :

To fade a bulb on and off using an Arduino Uno board.

## Hardware Required :

- Arduino Uno Board
- LED bulb
- Resistor

## Circuit Diagram –



Written Code :

```
void setup()  
{  
  pinMode(9, OUTPUT);  
}  
void loop()  
{  
  for (int i=0; i<300; i+=10)  
  {  
    delay(100);  
    analogWrite(9, i);  
  }  
  for (int i=300; i>0; i-=10)  
  {  
    delay(100);  
    analogWrite(9, i);  
  }  
}
```

## **Observation /Output:**

The LED bulb was faded on and off with a delay of 100ms.

## **Program no 03**

### **Program Title: POTENTIOMETER-LED**

#### **Aim**

**To demonstrate brightness of LED according to Potentiometer resistance**

#### **Hardware Required**

- Arduino Board
- Led Light
- 1 ohm resistor
- 1 Potentiometer

#### **Circuit Diagram -**



Potentiometer

Saved

Code Start Simulation Export Share

Text 1 (Arduino Uno R3)

```
1 //potentiometer
2 void setup()
3 { Serial.begin(9600);
4   pinMode(9, OUTPUT);
5 }
6
7 void loop()
8 {
9   int analog=analogRead(A0);
10  int brightness=map(analog,0,1023,0,255);
11  analogWrite(9,brightness);
12  Serial.print("\n analog value");
13  Serial.print(analog);
14  Serial.print("\n brightness value");
15  Serial.print(brightness);
16 }
```

Serial Monitor

Simulator time: 00:00:03

Potentiometer

Saved

Code Stop Simulation Export Share

Text 1 (Arduino Uno R3)

```
1 //potentiometer
2 void setup()
3 { Serial.begin(9600);
4   pinMode(9, OUTPUT);
5 }
6
7 void loop()
8 {
9   int analog=analogRead(A0);
10  int brightness=map(analog,0,1023,0,255);
11  analogWrite(9,brightness);
12  Serial.print("\n analog value");
13  Serial.print(analog);
14  Serial.print("\n brightness value");
15  Serial.print(brightness);
16 }
```

Serial Monitor

Simulator time: 00:00:03

Potentiometer

Saved

Code Stop Simulation Export Share

Text 1 (Arduino Uno R3)

```
1 //potentiometer
2 void setup()
3 { Serial.begin(9600);
4   pinMode(9, OUTPUT);
5 }
6
7 void loop()
8 {
9   int analog=analogRead(A0);
10  int brightness=map(analog,0,1023,0,255);
11  analogWrite(9,brightness);
12  Serial.print("\n analog value");
13  Serial.print(analog);
14  Serial.print("\n brightness value");
15  Serial.print(brightness);
16 }
```

Serial Monitor

Simulator time: 00:00:07

## 4) Potentiometer - LED

```
// potentiometer
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 value");
  Serial.print(analog);
  Serial.print("\n brightness value");
  Serial.print(brightness);
}
```

**Observation /Output:**

The brightness of the LED bulb increased/decreased upon turning the potentiometer knob..



# IOT LAB - 5th Sem

Name : Varad Vithal KJ , USN : 11BM18CS122

Program Title : Fire Detection

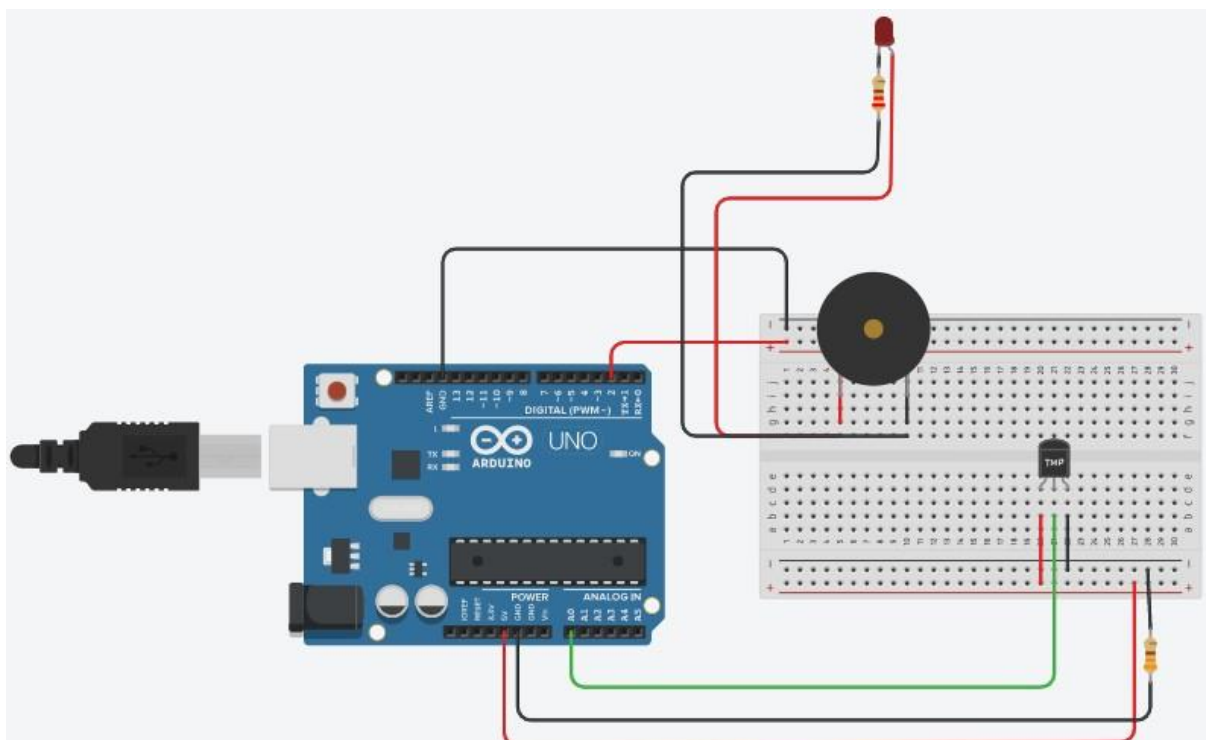
Aim :

To turn on a LED and buzzer upon detecting a fire (aka high temp) using an Arduino Uno board.

Hardware Required :

- Arduino Uno Board
- LED
- Buzzer
- Temperature Sensor
- 330 Ohm Resistor
- 220 Ohm Resistor

Circuit Diagram :



Written Code :

### 8) Temperature Sensor

```
int tmpSensor = A0;  
int outputpin = 0;  
float tmp;  
void Setup()  
{  
  Serial.begin(9600);  
}  
void loop()  
{  
  int rawvoltage = analogRead(outputpin);  
  float millivolts = (rawvoltage / 1024.0) * 500;  
  float celsius = millivolts / 10;  
  Serial.print(celsius);  
  Serial.print("Degrees in Celsius,");  
  Serial.print((celsius * 9) / 5 + 32);  
  Serial.println("Degrees Fahrenheit");  
  delay(1000);  
}
```

**Observation /Output :**

The LED and Buzzer are turned on when a fire is detected.

# IOT LAB - 5th Sem

Name : Varad Vithal KJ , USN : 11BM18CS122

Program Title : Light Sensor

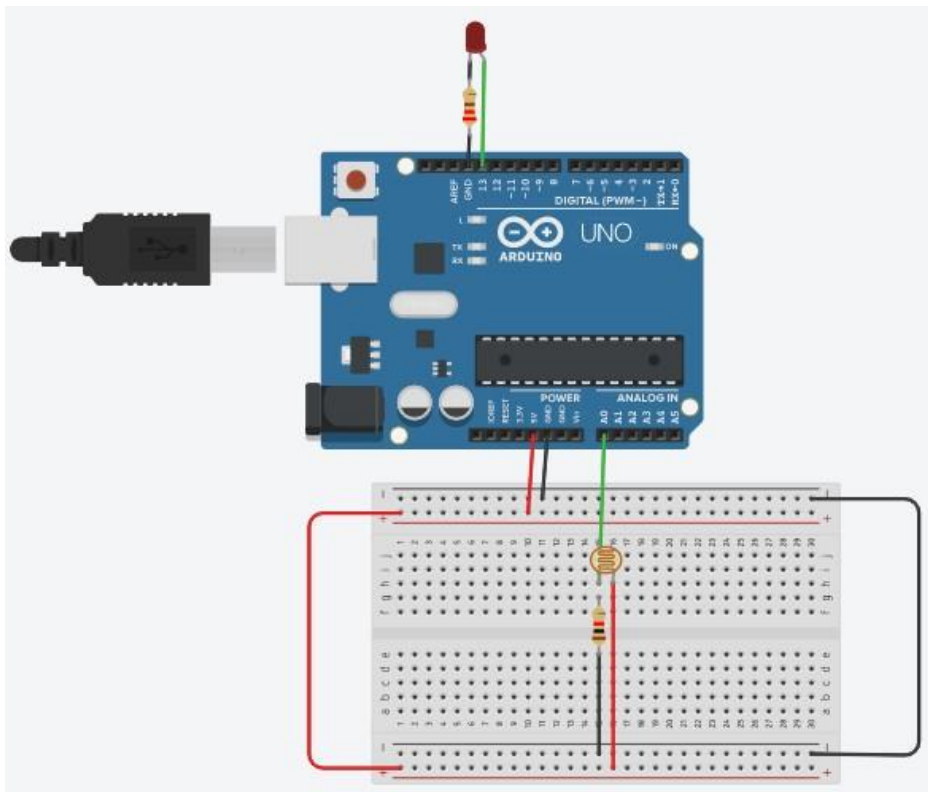
Aim :

To turn on and off an LED beyond a certain threshold using an Arduino Uno board.

## Hardware Required

- Arduino Uno Board
- LDR
- LED
- 2 Resistors

## Circuit Diagram -



**Written Code :**



```
{  
  digitalWrite(13, LOW);
```

```
  int led = 13;  
  int sensor = 6;  
  delay(1000);
```

```
}  
  
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 (Val == LOW) {
```

IBM 18CS122

K.J. Varad - vithal

6) LDR

```
intldr=A0;  
intldrvalue=0;  
int light_sensitivity = 500;
```

```
Void Setup()
```

```
{
```

```
Serial.begin(9600);
```

```
pinMode(13,OUTPUT);
```

```
}
```

```
Void loop()
```

```
{
```

```
ldrvalue = analogRead(ldr);
```

```
Serial.println(ldrvalue);
```

```
if(ldrvalue < light_sensitivity)
```

```
{
```

```
digitalWrite(13,HIGH);
```

```
}
```

```
else
```

## **Observation /Output:**

The LED turned off as the light increased beyond the threshold.

# IOT LAB - 5th Sem

Name : Varad Vithal KJ , USN : 11BM18CS122

Program Title : Passive Infrared Sensor

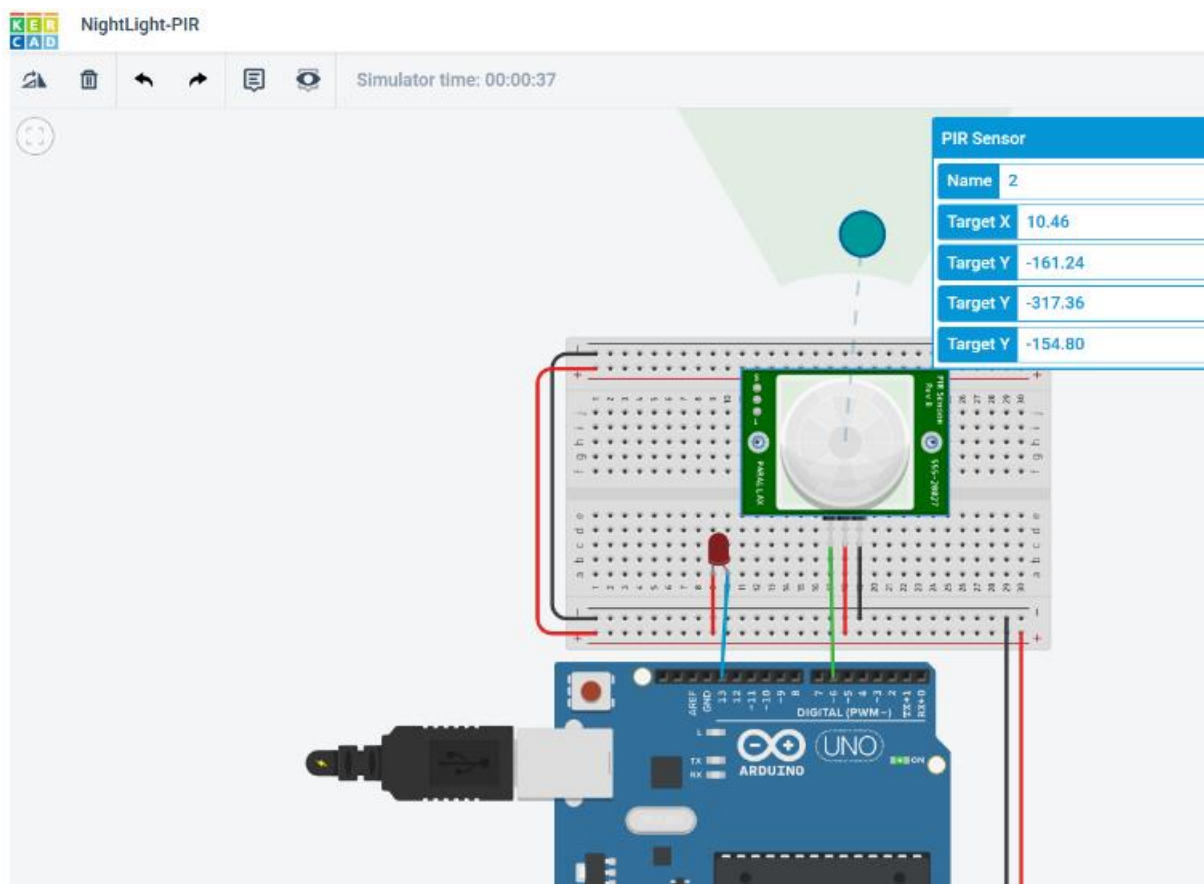
Aim :

To turn on a LED when motion is detected using an Arduino Uno board.

## Hardware Required

- Arduino Uno Board
- LED
- PIR
- 330 Ohm Resistor

## Circuit Diagram -





**Written Code :**

1BM48CS122

K. J. Varad-vithal

NightLight-PIR

int led = 13;

int sensor = 6;

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) {

IBM18CSI22

K.J. Varad-Vitha

```
Serial.println("Motion detected!");  
State = HIGH;
```

```
}
```

```
}
```

```
else {
```

```
    digitalWrite(LED, LOW);  
    delay(10);
```

```
if (State == HIGH) {
```

```
    Serial.println("Motion stopped!");  
    State = LOW;
```

```
}
```

```
}
```

```
}
```



## **Observation /Output:**

The LED is turned on when motion is detected.

# IOT LAB - 5th Sem

Name : Varad Vithal KJ , USN : 1BM18CS122

Program No : 09

Program Title : Fire Detection

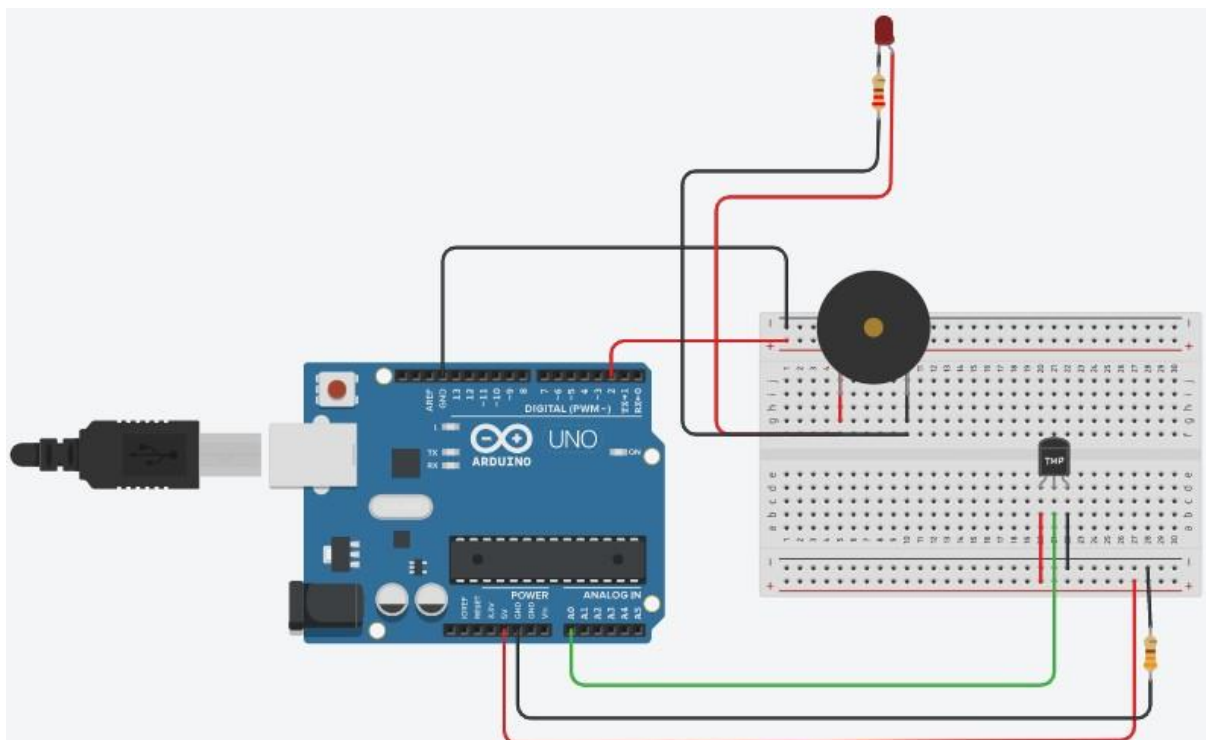
Aim :

To turn on a LED and buzzer upon detecting a fire (aka high temp) using an Arduino Uno board.

Hardware Required :

- Arduino Uno Board
- LED
- Buzzer
- Temperature Sensor
- 330 Ohm Resistor
- 220 Ohm Resistor

Circuit Diagram :



```

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

void loop()
{
  int temp = analogRead(A0);
  float x = map(x, 31, 368, -40, 125);

  if (x > 70)
  {
    Serial.println("Buzzz");
    digitalWrite(2, HIGH);
    delay(2000);
    digitalWrite(2, LOW);
  }
  else
  {
    Serial.println("Idle");
  }
}

```

### Observation /Output :

The LED and Buzzer are turned on when a fire is detected.

# IOT LAB - 5th Sem

Name : Varad Vithal KJ , USN : 11BM18CS122

Program Title : Ultrasound Distance Sensor

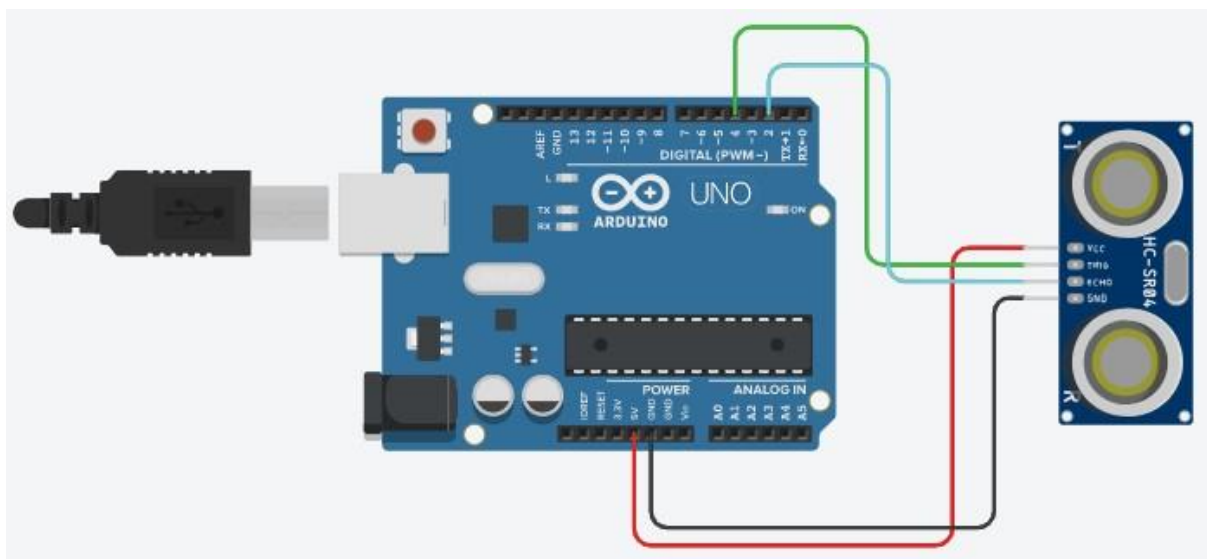
Aim :

To measure distance using an ultrasound sensor and an Arduino Uno board.

Hardware Required :

- Arduino Uno Board
- HC-SR04 Ultrasound Sensor

Circuit Diagram :



Written Code :

## 10) ULTRASONIC SENSOR.

```
int sensor = 7;
```

```
int red = 13;
```

```
const int blue = 10;
```

```
int green = 9;
```

```
void setup()
```

```
{
```

```
  pinMode(blue, OUTPUT);
```

```
  pinMode(green, OUTPUT);
```

```
  pinMode(red, OUTPUT);
```

```
  pinMode(7, OUTPUT);
```

```
  Serial.begin(9600);
```

```
}
```

```
long duration, inches, cm;
```

```
void loop()
```

```
{
```

```
  pinMode(sensor, OUTPUT);
```

```
  digitalWrite(sensor, LOW);
```

```
  delay(2);
```

```
  digitalWrite(sensor, HIGH);
```

```
  delay(5);
```

```
  digitalWrite(sensor, LOW);
```

P111 will (Sensor, HIGH);

duration = pulseIn(Sensor, HIGH); // reading  
duration for a HIGH pulse

// time -> distance

inches = microsecondsToInches(duration);

cm = microsecondsToCentimeters(duration);

Serial.print(inches);

Serial.print("in, ");

Serial.print(cm);

Serial.print("cm ");

Serial.println();

if (inches < 10)

{

digitalWrite(red, HIGH);

digitalWrite(green, LOW);

digitalWrite(blue, LOW);

}

else if (inches > 10 & inches < 50)

{

digitalWrite(red, LOW);

digitalWrite(green, HIGH);

digitalWrite(blue, HIGH);

}



else

{

digital Write (red, LOW);

digital write (green, HIGH);

digital Write (blue, LOW);

}

}

long microseconds ToInches (long microsecond

{

return microseconds / 74 / 2;

}

long microseconds ToCentimeters (long microse

{

return microseconds / 29 / 2;

}

}

**Observation /Output :**

The distance was measured using the ultrasound sensor.



# IOT LAB - 5th Sem

Name : Varad Vithal KJ , USN : 11BM18CS122

Program Title : Gas Sensor

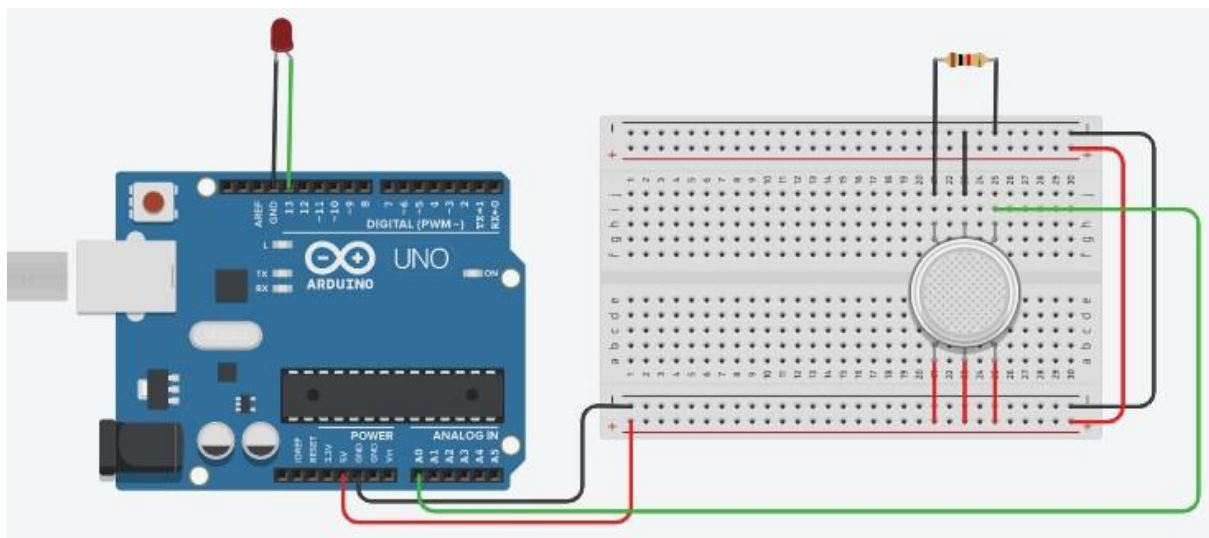
Aim :

To notify if there is a gas leakage near the sensor using an Arduino Uno board.

Hardware Required :

- Arduino Uno Board
- LED
- 330 Ohms Resistor
- Gas Sensor

Circuit Diagram :



Written Code :

## 12) Gas - Detection

```
int const gasPin = A0;
```

```
int red = 12;
```

```
int orange = 9;
```

```
int green = 7;
```

```
void setup()
```

```
{
```

```
  pinMode (12, OUTPUT);
```

```
  pinMode (orange, OUTPUT);
```

```
  pinMode (green, OUTPUT);
```

```
  Serial.begin(9600);
```

```
}
```

```
void loop()
```

```
  int value = analogRead(gasPin);
```

```
  Serial.print("Gas value=");
```

```
  Serial.print(value);
```

```
  Serial.println("");
```

```
  digitalWrite(green, HIGH);
```

```
digitalWrite(orange, value >= 90 ? HIGH : LOW);  
digitalWrite(red, value >= 150 ? HIGH : LOW);  
delay(100);  
}
```

**Observation /Output :**

The LED is turned on when gas is detected.

# IOT LAB - 5th Sem

Name : Varad Vithal KJ , USN : 1BM18CS122

Program Title : Vibration + LDR

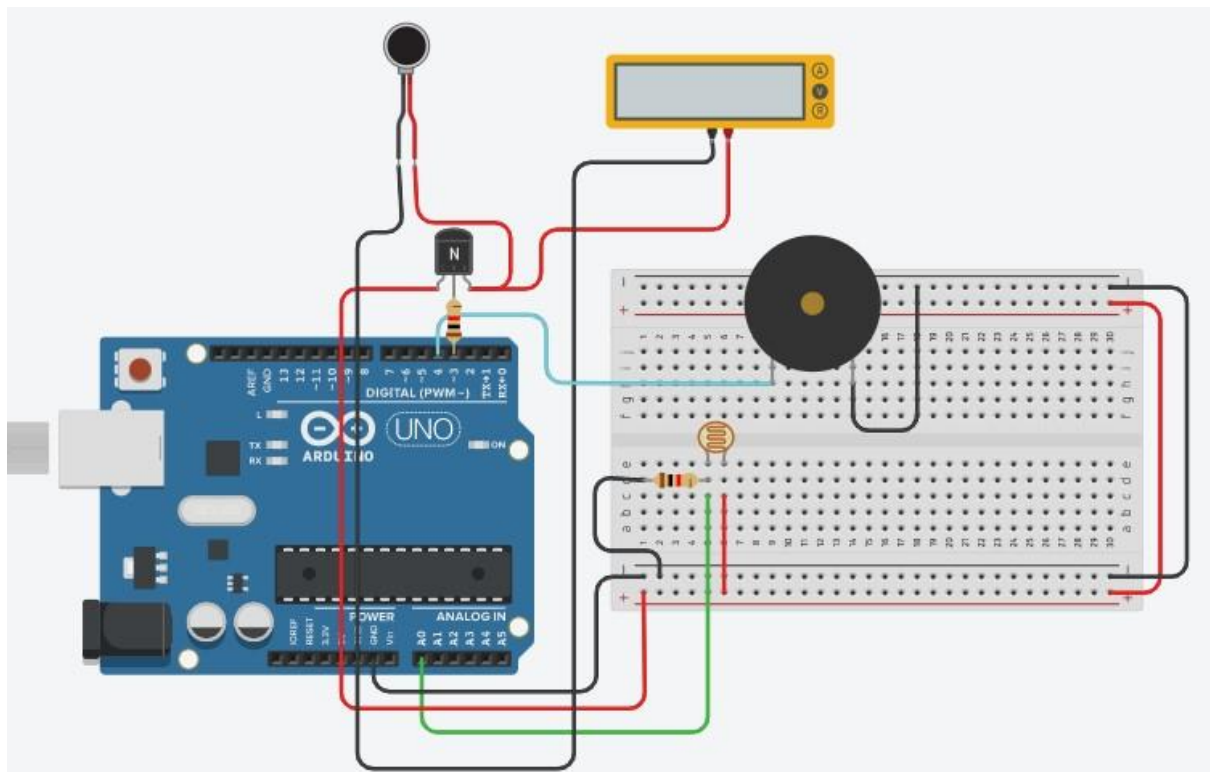
Aim :

To turn on the vibrator when the LDR detects light using an Arduino Uno board.

Hardware Required :

- Arduino Uno Board
- NPN Transistor
- Multimeter
- Vibrator Motor
- 1K Ohm Resistor X 2
- LDR
- Buzzer

Circuit Diagram :



Written Code :

### 13) Vibration Motor - LDR

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

#### Observation /Output :

The vibrator and buzzer turns on once the LDR detects light.

# IOT LAB - 5th Sem

Name :Varad Vithal KJ, USN : 1BM18CS122

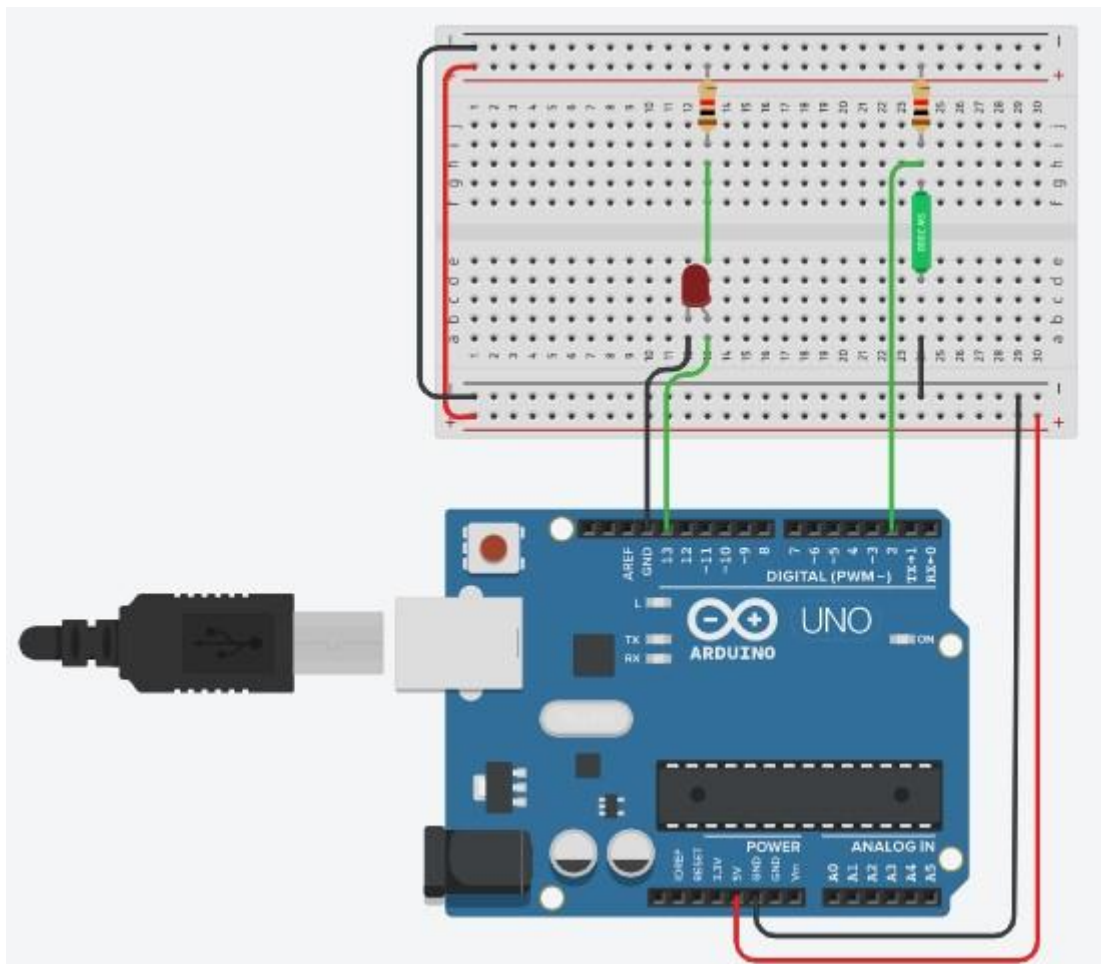
Program Title : Tilt Sensor

Aim :

To design a smart package handling system (Tilt & LED) using an Arduino Uno board.

Hardware Required :

- Arduino Uno Board
  - LED
  - Tilt Sensor
  - 2 Resistors – 1K Ohm
- Circuit Diagram :**



Written Code :



```
int tilt = 2;  
int led = 13;
```

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

```
void loop()  
{  
  int read; // would specify = 0  
  read = digitalRead (tilt);  
  Serial.println (read);  
  
  if (read == 1)  
    digitalWrite (led, LOW);  
  else  
    digitalWrite (led, HIGH);  
}
```

### Observation /Output :

The LED was lit once tilt was detected.

# IOT LAB - 5th Sem

Name : Varad Vithal KJ, USN : 1BM18CS122

Program No : 14

Program Title : IR based SERVO Motor Controller

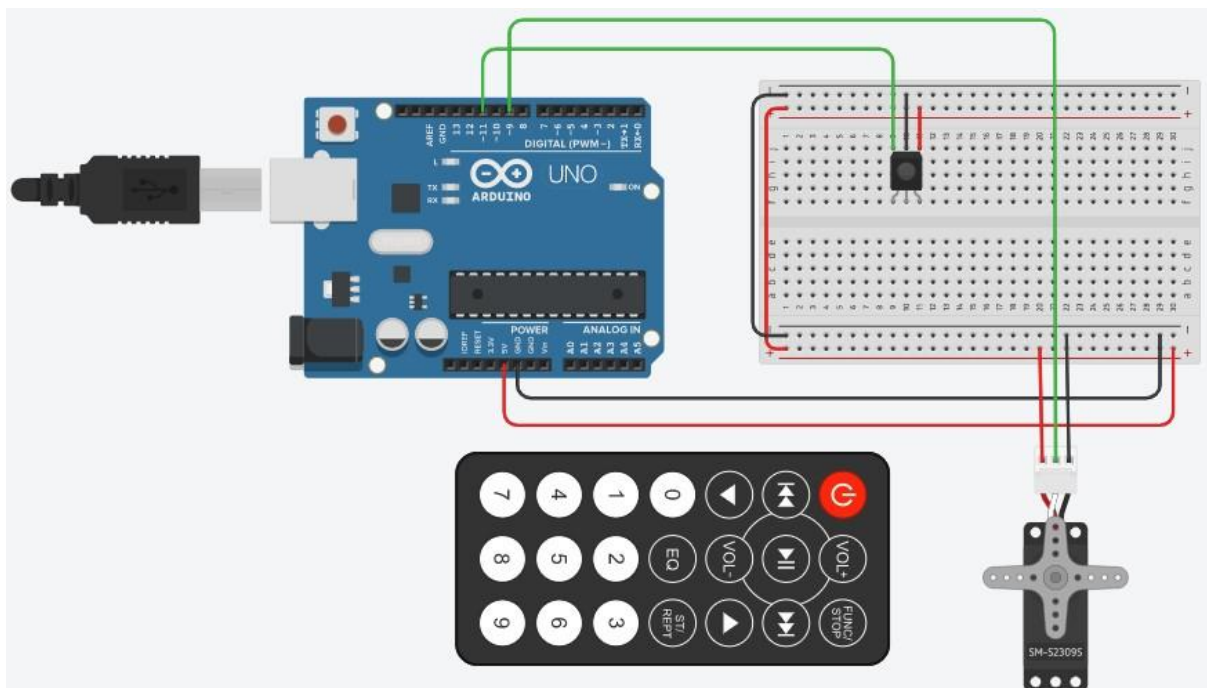
**Aim :**

To design IR based SERVO motor controller (Clockwise and Counter) using an Arduino Uno board.

**Hardware Required :**

- Arduino Uno Board
- IR Sensor
- IR Remote
- Micro Servo

**Circuit Diagram :**



**Written Code :**



## IOT LAB - 14

```
#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 0xFDA05F:
        myservo.write(-360);
        Serial.println("Counter Clockwise");
        break;

      case 0xFDA05F:
        myservo.attach(7);
        Serial.println("Stop");
        break;
    }
    irrecv.resume();
  }
}
```

### Observation /Output :

The Servo motor turns clockwise and counter clockwise upon detection of IR signal from the remote.



# **IOT LAB - 5th Sem**

**Name : Varad Vithal KJ , USN : 1BM18CS122**

**Program No : 15**

**Program Title : RGB LED interfacing with LCD panel**

**Aim :**

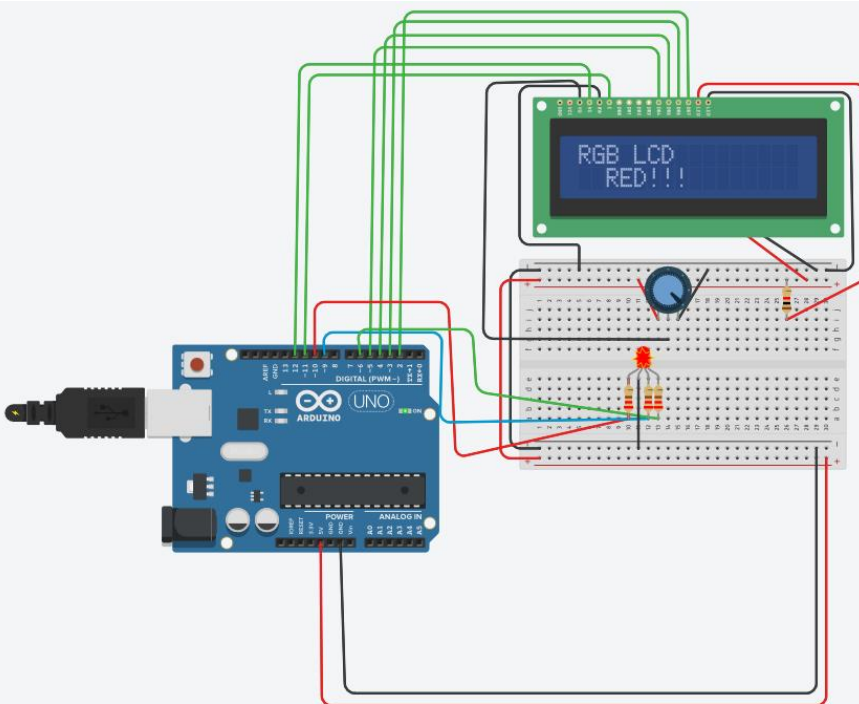
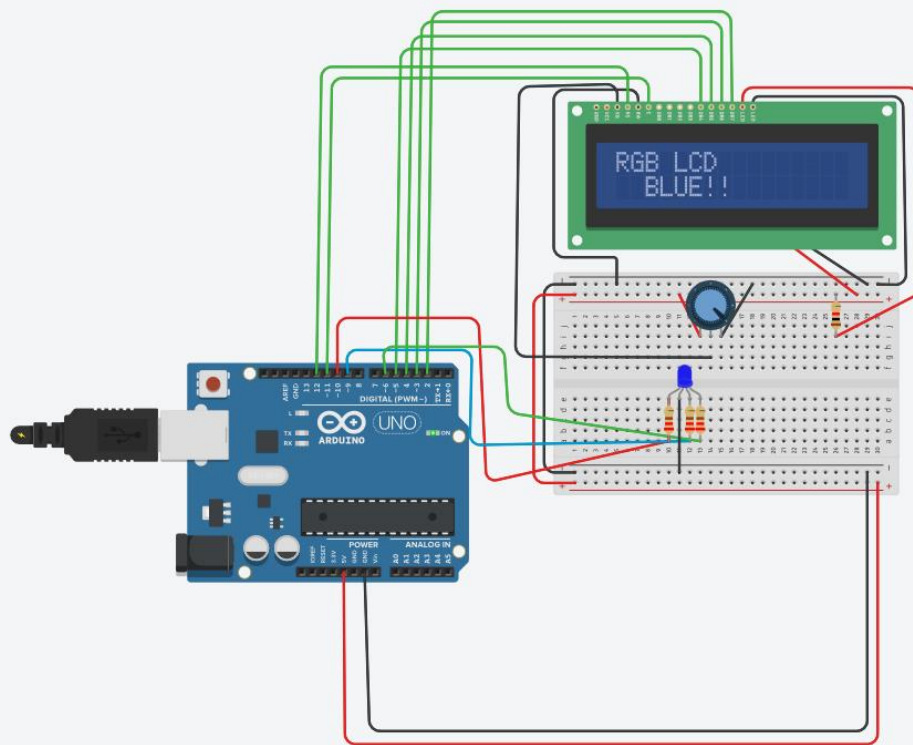
To switch colours using a RGB led and display the current colour in the LCD display using an Arduino Uno board.

**Hardware Required :**

- Arduino Uno Board
  - RGB LED
  - LCD Panel
  - 4 X 240 Ohm Resistor
- Circuit Diagram :**



Simulator time: 00:00:24





```

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

void setup ()
{
  pinMode (8, OUTPUT);
  pinMode (9, OUTPUT);
  pinMode (10, OUTPUT);

  lcd.begin (16, 2);
  lcd.print ("RGB-check");
}

void loop ()
{
  lcd.setCursor (0, 1);
  lcd.print ("Red ");
  digitalWrite (10, HIGH);
  digitalWrite (9, LOW);
  digitalWrite (8, LOW);
  delay (500);
  lcd.setCursor (0, 1);
  lcd.print ("Blue ");
  digitalWrite (10, LOW);
  digitalWrite (9, HIGH);
  digitalWrite (8, LOW);
  delay (500);
  lcd.setCursor (0, 1);
  lcd.print ("Green ");
  digitalWrite (10, LOW);
  digitalWrite (9, LOW);
  digitalWrite (8, HIGH);
  delay (500);
}

```

### Observation /Output :

The LED turns from Red – Blue – Green and the name of the current colour is displayed on the LCD.

# IOT LAB - 5th Sem

Name : Varad Vithal KJ , USN : 1BM18CS122

Program No : 16

Program Title : Smart Irrigation

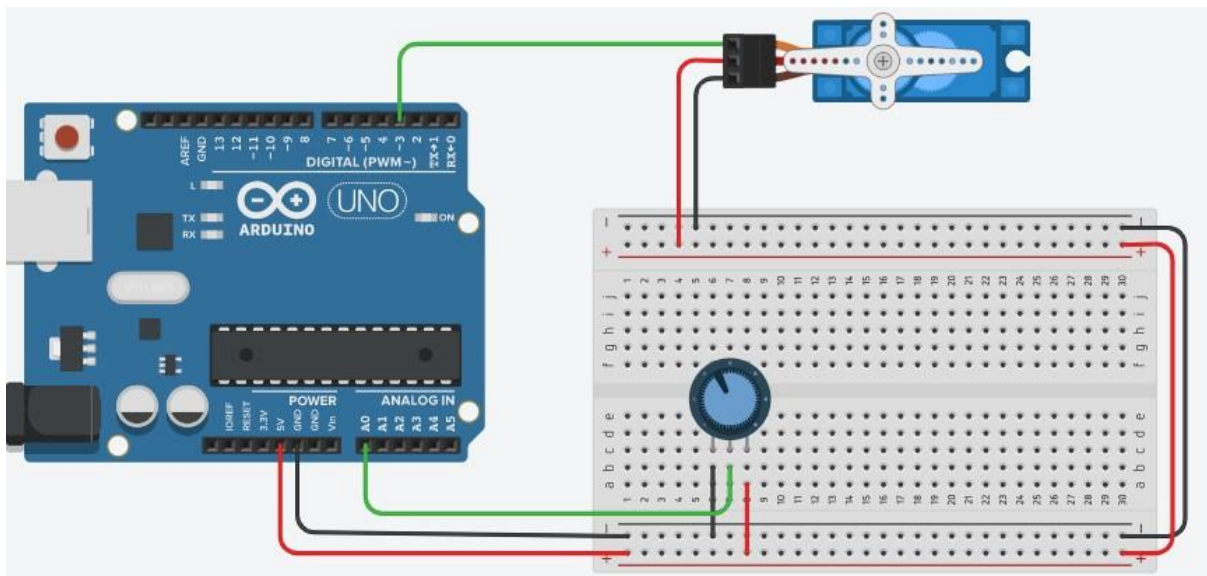
Aim :

To design a smart irrigation system (Potentio & Servo) using an Arduino Uno board.

Hardware Required :

- Arduino Uno Board
- Potentiometer
- Micro Servo

Circuit Diagram :



Written Code :



## TOT LAB - 15

```
#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)
    {
      myservo.write(pos);
      delay(15);
    }
  }
  delay(1000);
}
```

### Observation /Output :

The Servo moves when the potentiometers resistance is above a certain level.



