

Name – Samir kumar

Program No. – 01

Program Title – LED Blinking

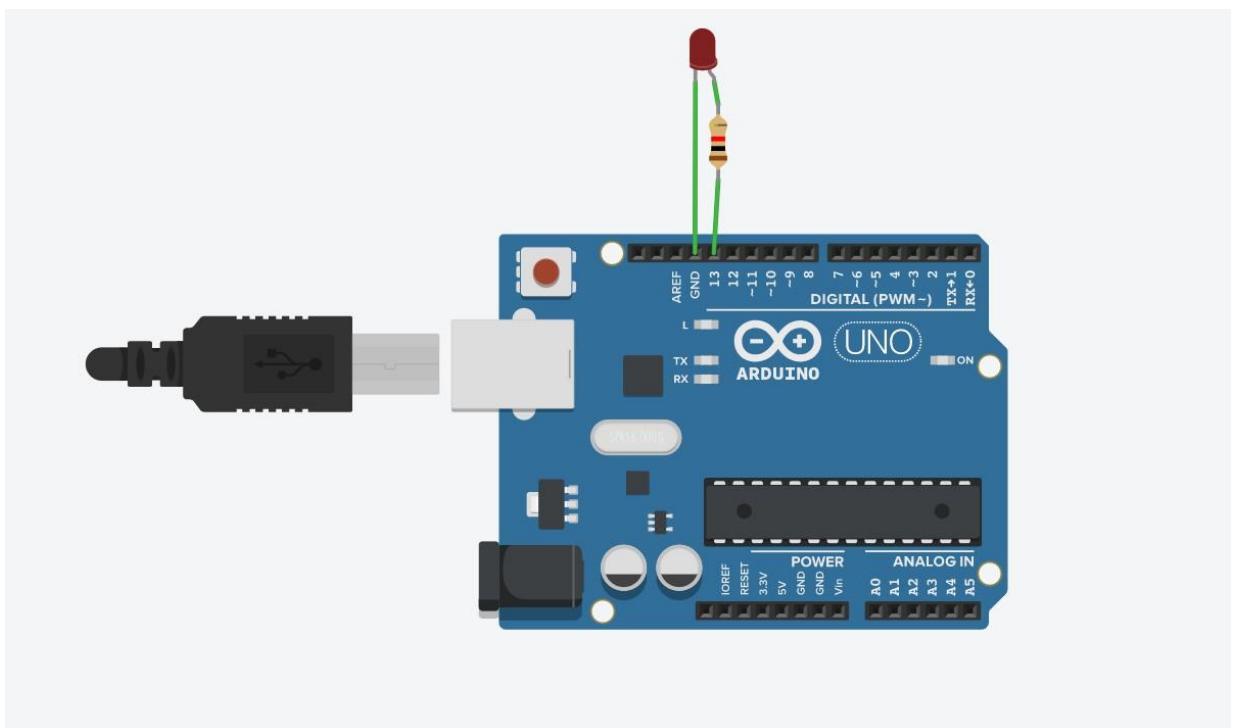
AIM

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

HARDWARES REQUIRED

- Arduino Board
- LEDs

CIRCUIT DIAGRAM



WRITE-UP

Name : SAMIR KUMAR

USN : 1BM18CS091

LED Blinking

Aim : Turn on the LED on for a record, then off for a record, repeatedly

Hardware Required :-

- Arduino Board
- LEDs

CODE :

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

Void loop()
{
    digital write (13, HIGH);
    delay (1000);
    digital write (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 – Samir kumar

Program No. – 02

Program Title – Traffic Controller

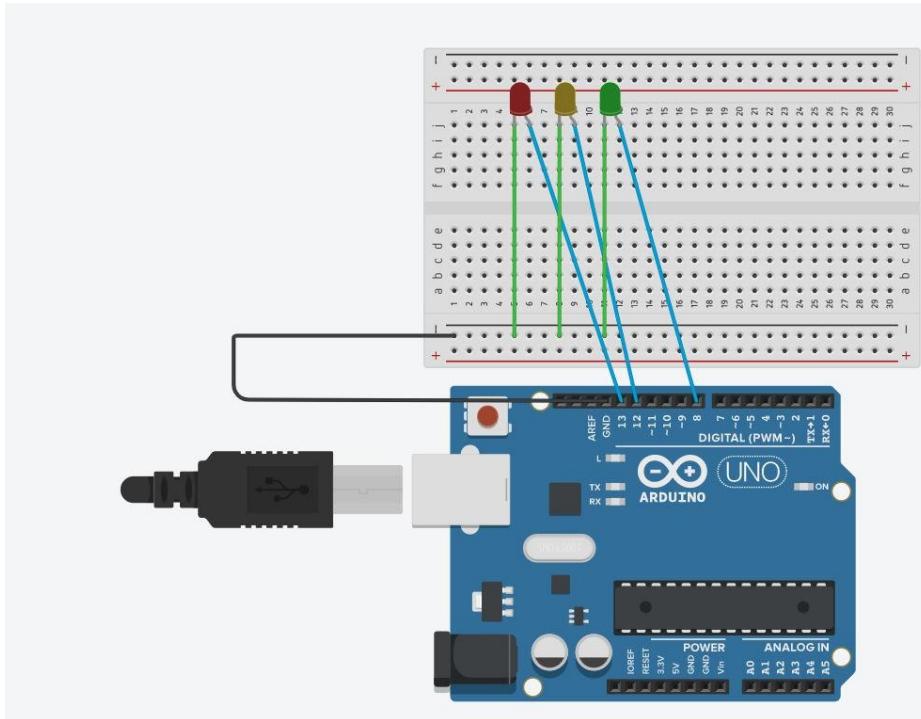
AIM

Traffic Signal Simulator.

HARDWARES REQUIRED

- Arduino Board
- LEDs
- Breadboard

CIRCUIT DIAGRAM



WRITE-UP

Name : SAMIR KUMAR

USN: 18M18(509)

TRAFFIC CONTROLLER

Aim: Traffic signal simulator

Hardware required:

- Arduino Board
- LED
- Bread Board

CODE :-

```
void setup () {  
    pin mode (13, output);  
    pin mode (12, output);  
    pin mode (8, output);  
}
```

```
void read () {
```

```
    digital write (13, HIGH);  
    digital write (12, LOW);  
    digital write (8, LOW);  
}
```


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 – Samir kumar

Program No. – 03

Program Title – LED fading without potentiometer

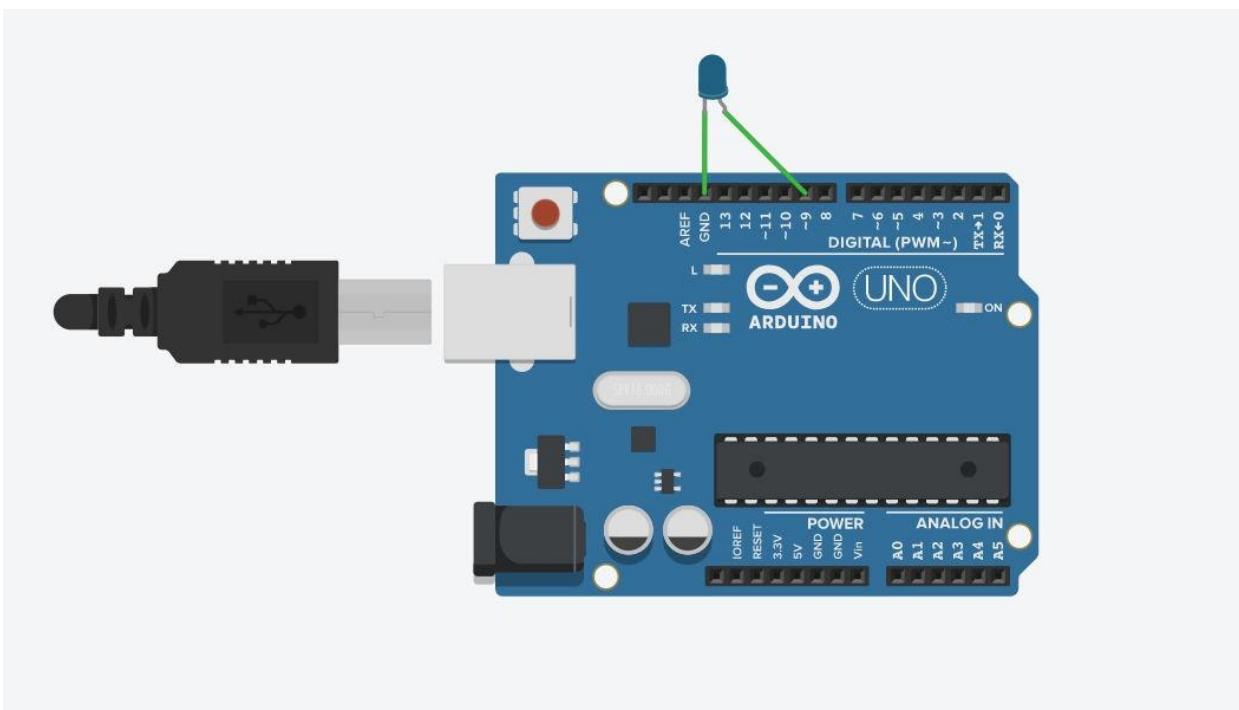
AIM

Demonstrate to show LED fading.

HARDWARES REQUIRED

- Arduino Board
- LED bulb

CIRCUIT DIAGRAM



WRITE-UP

Name: Samir Kumar
USN: 1 BM18CS091

CLASSEmate

Date _____
Page _____

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);
```

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 – Samir kumar

Program No. – 04

Program Title – LED fading with potentiometer

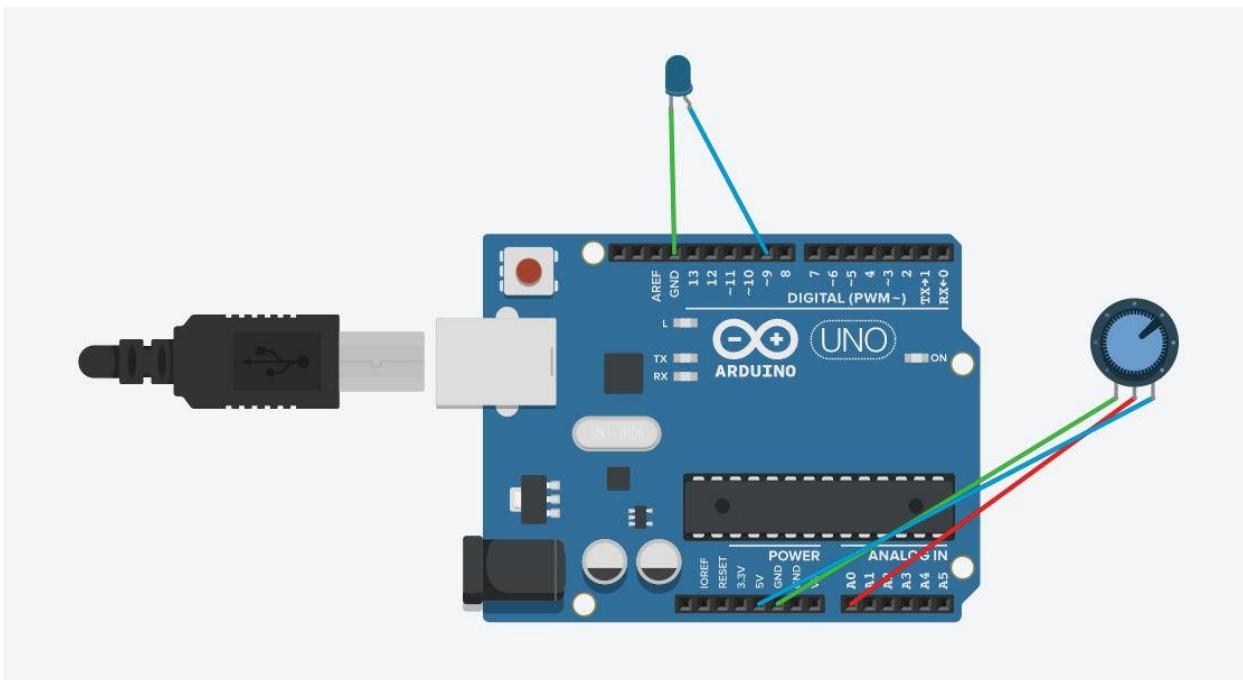
AIM

Demonstrate to show LED fading(analog output).

HARDWARES REQUIRED

- Arduino Board
- LED bulb
- Potentiometer

CIRCUIT DIAGRAM



WRITE-UP

Name: SANJAY KUMAR
VSN: 18M18C8091

classmate

Date _____
Page _____

EXPS

LED FADEING WITH POTENTIOMETER

Aim :-

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);
```

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 – Samir kumar

Program No. – 05

Program Title – ON/OFF LED using Push Button

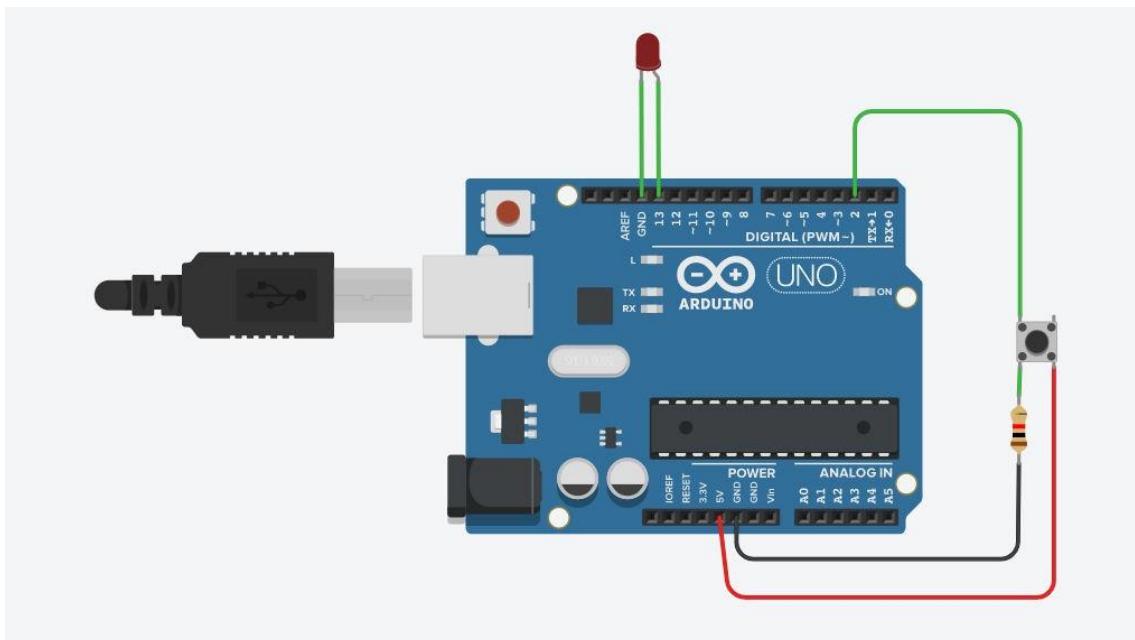
AIM

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 : ~~00000000~~ SANIR KUMAR
USN : 10M18(509)

classmate

Date _____

Page _____

BP-S

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()
```

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 – Samir kumar

Program No. – 06

Program Title – Fire Alarm using flame Sensor

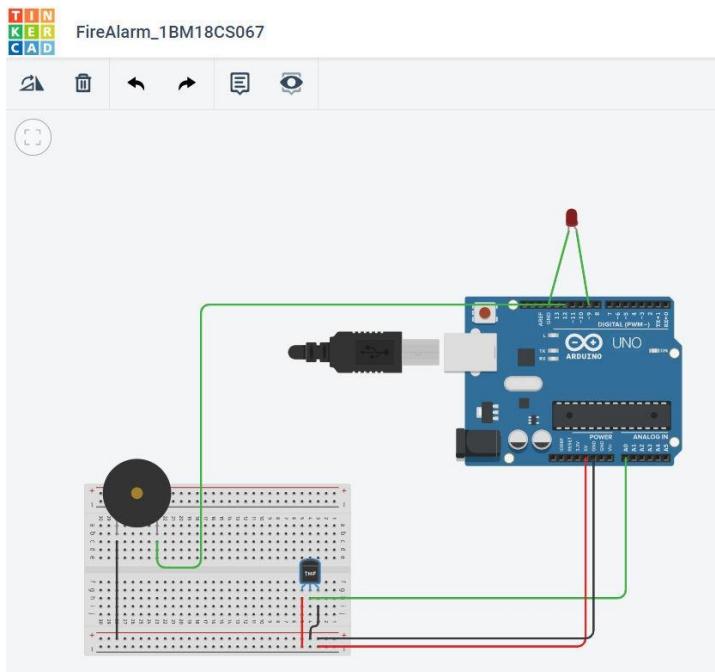
AIM

Design an alert system using a flame sensor.

HARDWARES REQUIRED

- Arduino Board
 - Piezo
 - Temperature Sensor
 - Breadboard small

CIRCUIT DIAGRAM



WRITE-UP

Experiment-6

Aim: Design a alert system using flame sensors (use temp sensor for experiment in tinkerCAD)

Hardware requirement:

Arduino Board

Bread Board

TMPSensor sensor

Piezo

Code:

```
int temppin = A0;
```

```
float temp = 0;
```

```
int Buzz = 12;
```

```
void setup()
```

```
{
```

```
    serial.begin(9600);
```

```
    pinMode(Buzz, OUTPUT);
```

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

```
}
```

```
void loop()
```

```
{
```

```
    temp = analogRead(temppin);
```

```
    temp = temp * 0.48828125;
```

```
    serial.println(temp);
```

```
    delay(1000);
```

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 – Samir kumar

Program No. – 07

Program Title – Distance Measurement using ultrasonic sensor

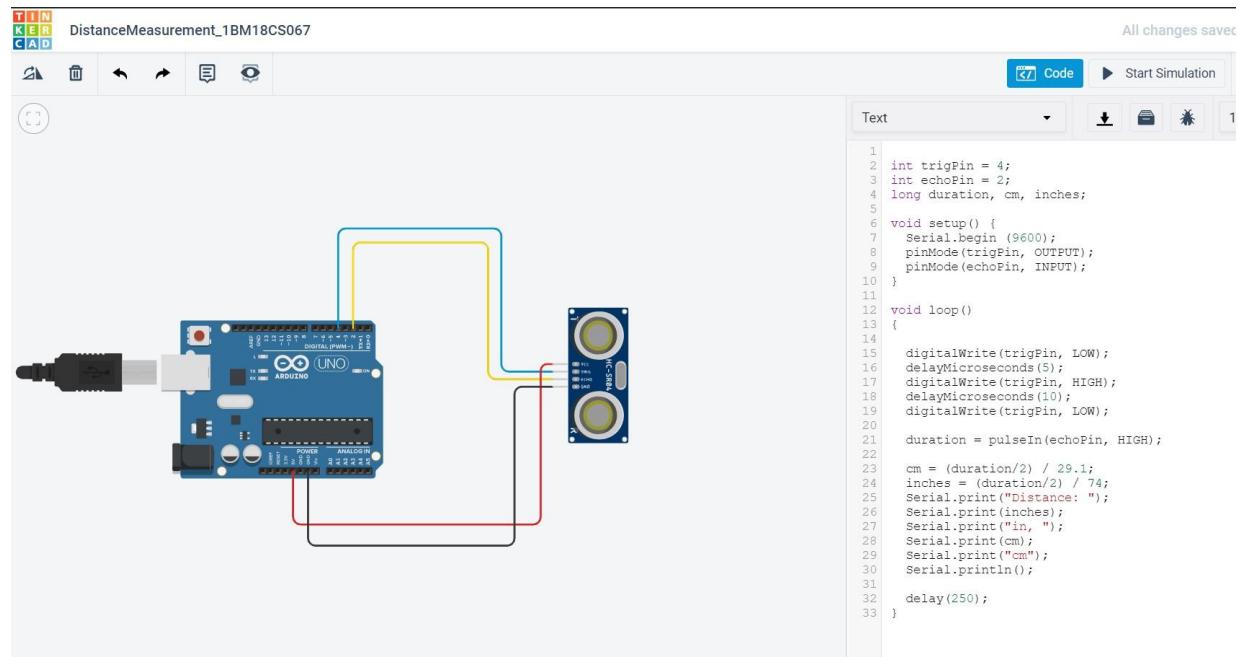
AIM

Design a system to measure the distance between objects.

HARDWARES REQUIRED

- Arduino Board
- Ultrasonic sensor HC-SR04

CIRCUIT DIAGRAM



WRITE-UP

Exp-7

Aim: Design a system to measure the distance between objects.

Hardware Required:

Arduino Board

Ultrasonic distance measure.

Code:

@te

void loop()

```
{  
    long duration, inch, cm;  
    pinMode(9, OUTPUT);  
    digitalWrite(9, HIGH);  
    delayMicroseconds(2);  
    digitalWrite(9, LOW);  
    delayMicroseconds(10);  
    digitalWrite(9, HIGH);  
    delayMicroseconds(10);  
    digitalWrite(9, LOW);  
}
```

duration = pulseIn(7, HIGH);

delayMicroseconds(10);

digitalWrite(9, HIGH);

pinMode(7, INPUT);

duration = pulseIn(7, HIGH);

delayMicroseconds(10);

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 – Samir kumar

Program No. – 08

Program Title – LDR

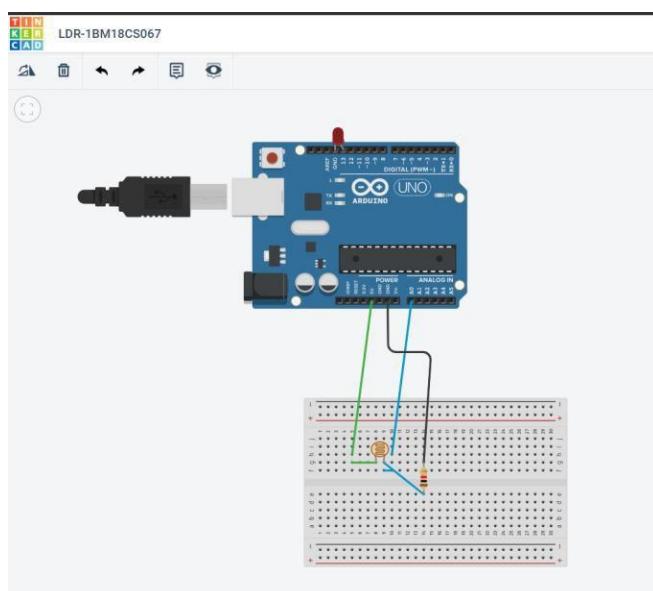
AIM

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

TINKER CAD Ldr

Simulator time: 00:00:00.506

All changes saved

Code Stop Simulation Export Share 1 (Arduino Uno R3)

Text

```
9  pinMode(ldrPin, INPUT);
10 }
11 void loop()
12 {
13   int ldrStatus = analogRead(ldrPin);
14   Serial.println(ldrStatus);
15   if(ldrStatus <=10)
16   {
17     digitalWrite(ledPin, HIGH);
18     Serial.println("LDR is DARK, LED is ON");
19   }
20   else
21   {
22     digitalWrite(ledPin, LOW);
23     Serial.println("-----");
24   }
25 }
```

Serial Monitor

```
6
LDR is DARK, LED is ON
```

Send Clear 15:55 30/09/2020

The screenshot shows a Tinkercad simulation environment. On the left, there's a breadboard with a blue Arduino Uno. A photoresistor (LDR) is connected to digital pin A0, and a red LED is connected to digital pin 13. A 10k pull-down resistor is also present. The breadboard is connected to a USB cable. On the right, the code for the Arduino Uno is displayed:

```
9  pinMode(ldrPin, INPUT);
10 }
11 void loop()
12 {
13   int ldrStatus = analogRead(ldrPin);
14   Serial.println(ldrStatus);
15   if(ldrStatus <=10)
16   {
17     digitalWrite(ledPin, HIGH);
18     Serial.println("LDR is DARK, LED is ON");
19   }
20   else
21   {
22     digitalWrite(ledPin, LOW);
23     Serial.println("-----");
24   }
25 }
```

The serial monitor on the right shows the output of the code, which prints the value of the LDR reading every loop iteration. The values are consistently low (around 6), indicating that the LED is turned on because the LDR is dark.

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 –Samir kumar

Program No. – 09

Program Title – PIR

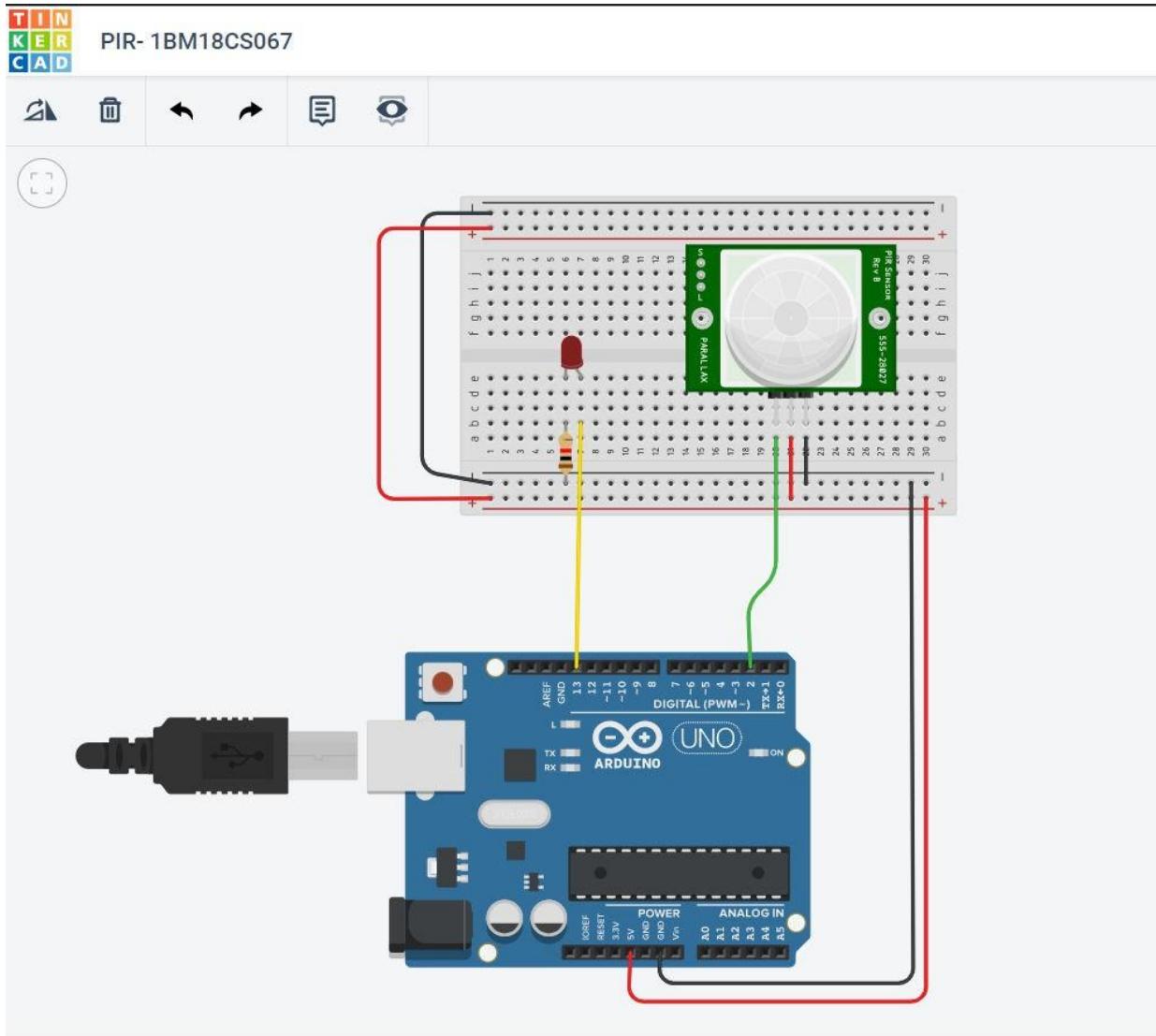
AIM

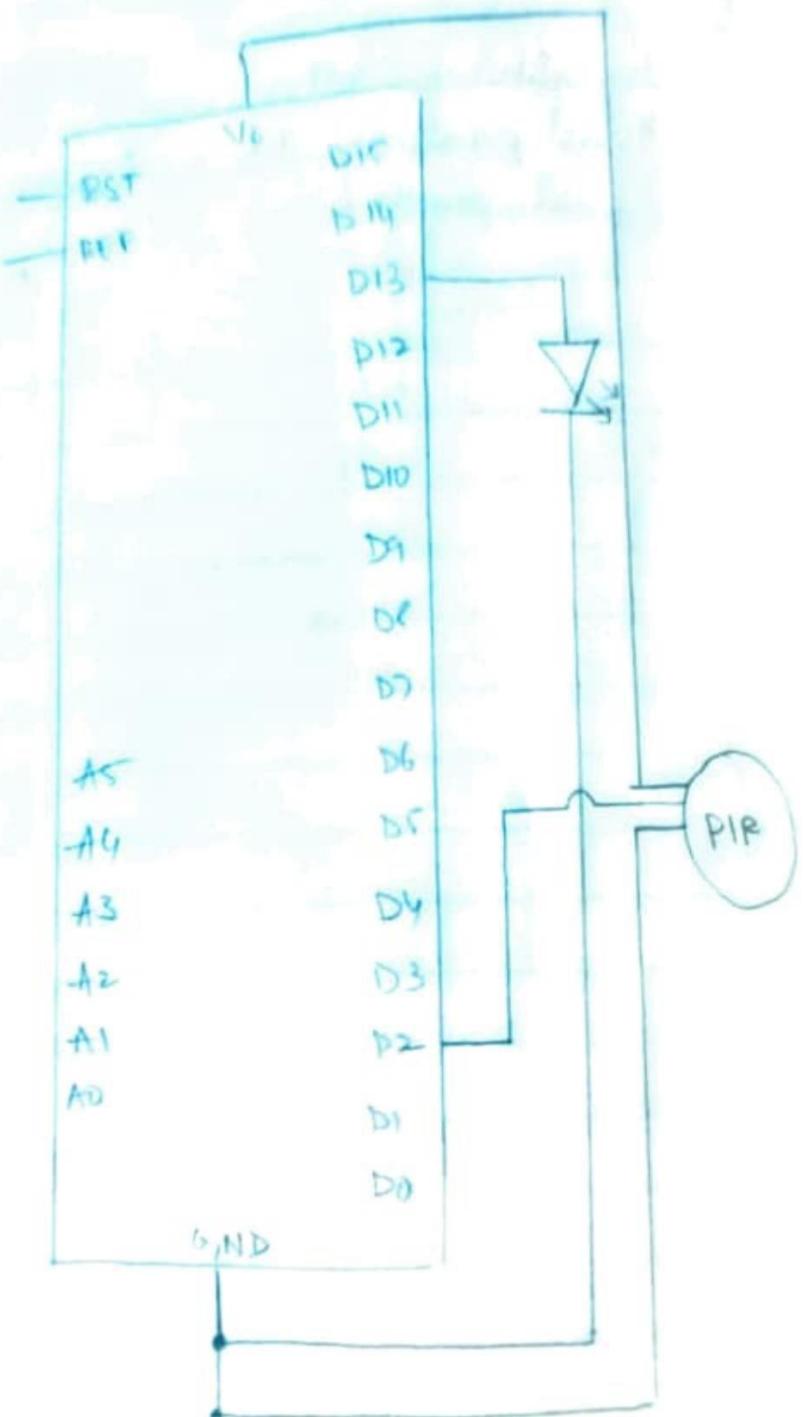
Demonstrate to show working of PIR sensor.

HARDWARES REQUIRED

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

CIRCUIT DIAGRAM





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 – Samir kumar

Program No. – 10

Program Title – Gas Sensor

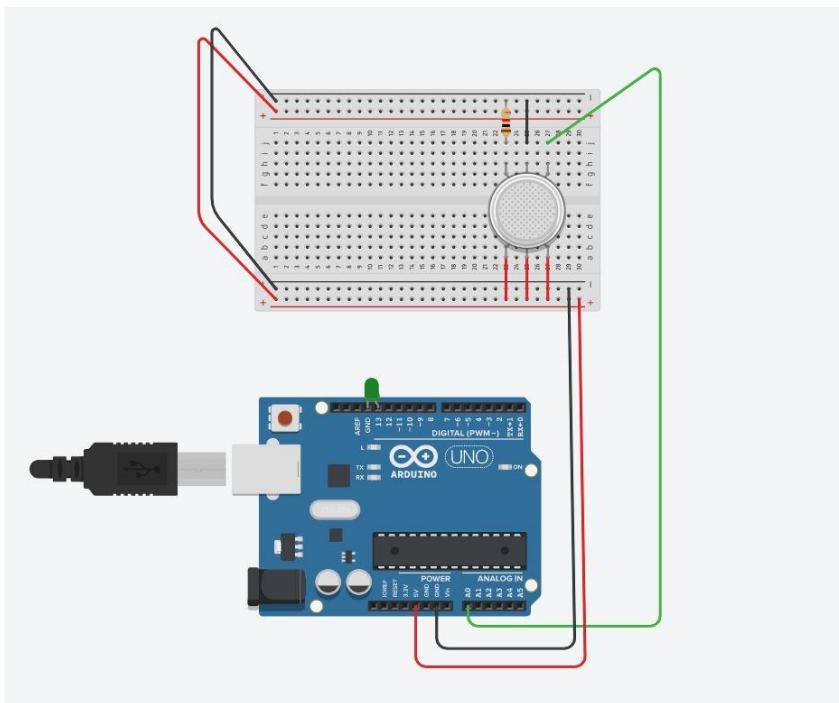
AIM

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 – Samir kumar

Program No. – 11

Program Title – Vibration motor and LDR

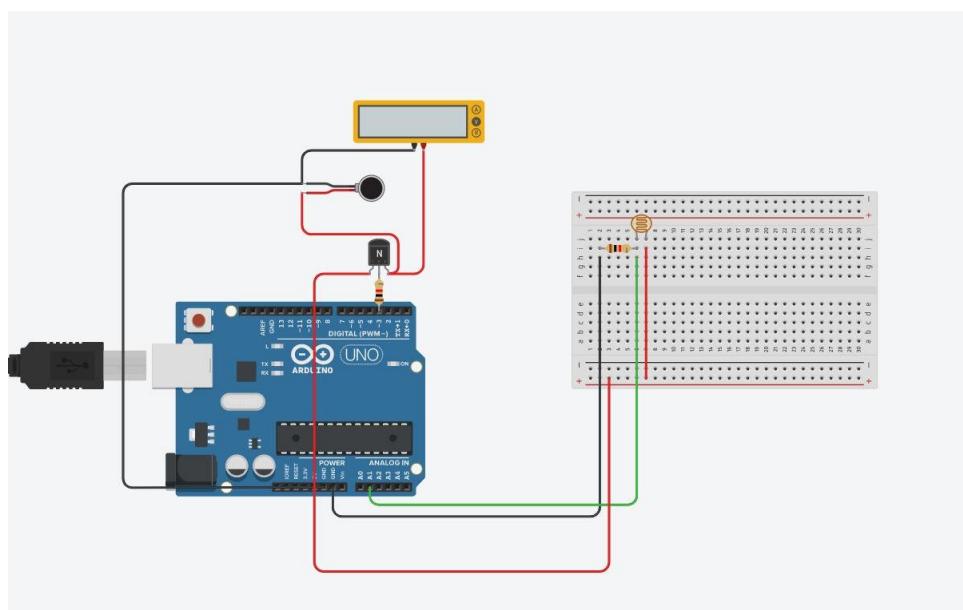
AIM

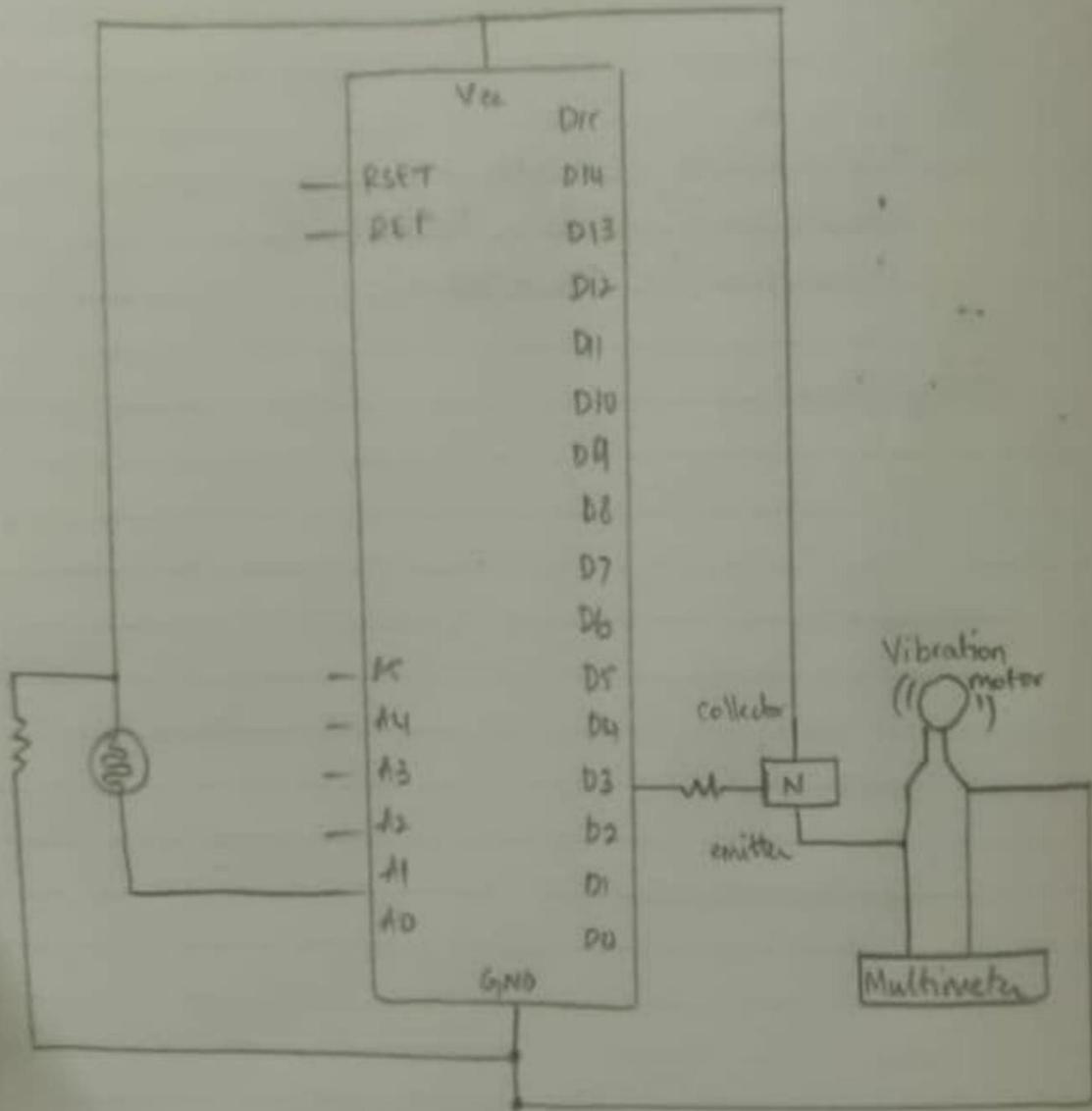
To design an automated day indicator system.

HARDWARES REQUIRED

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

CIRCUIT DIAGRAM





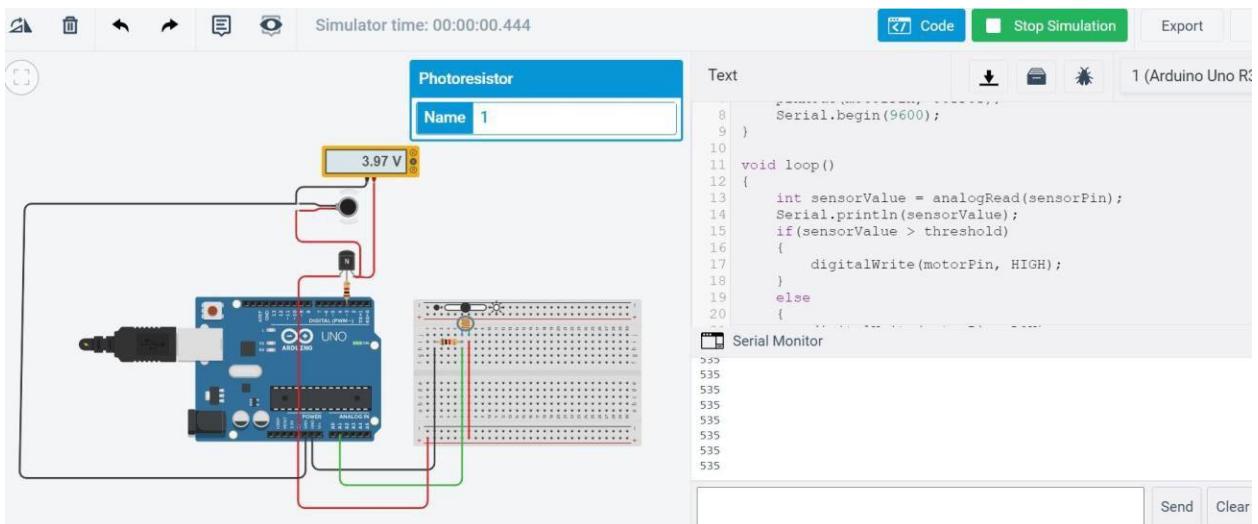
CODE

```
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 – Samir kumar

Program No. – 12

Program Title – Tilt Sensor

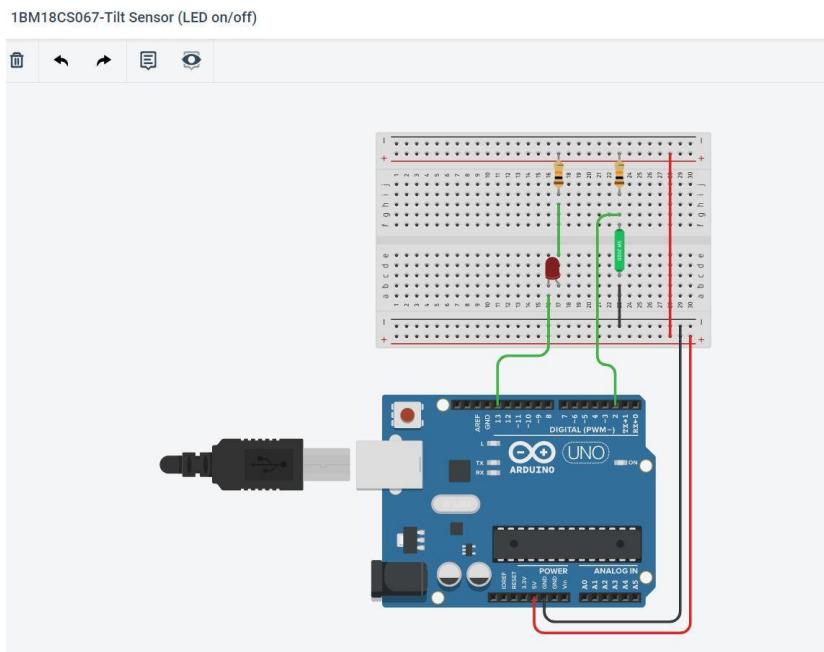
AIM

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

HARDWARES REQUIRED

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

CIRCUIT DIAGRAM



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 – Samir kumar

Program No. – 13

Program Title – IR based SERVO Motor controller

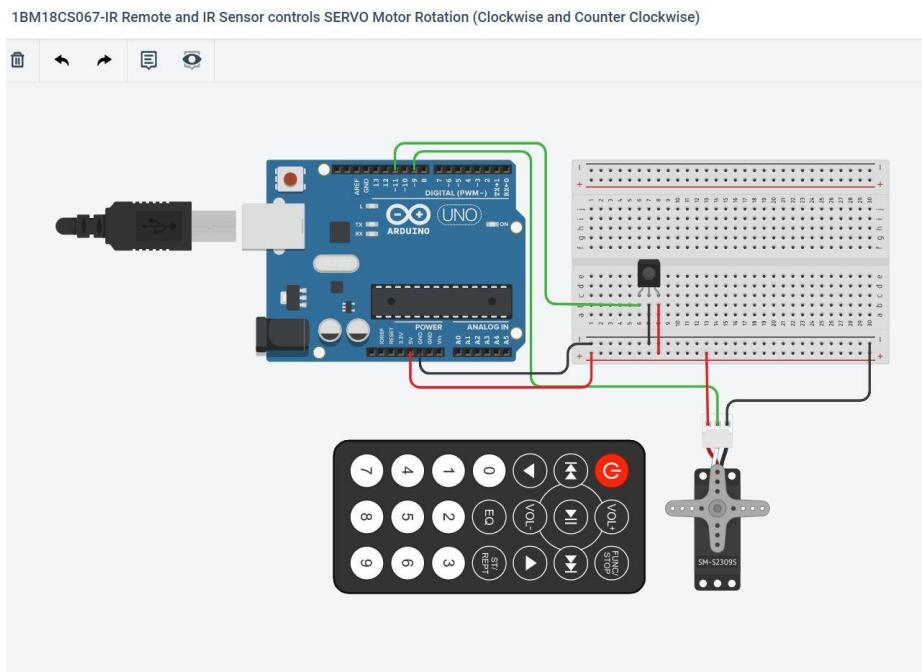
AIM

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



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 – Samir kumar

Program No. – 14

Program Title – Irrigation

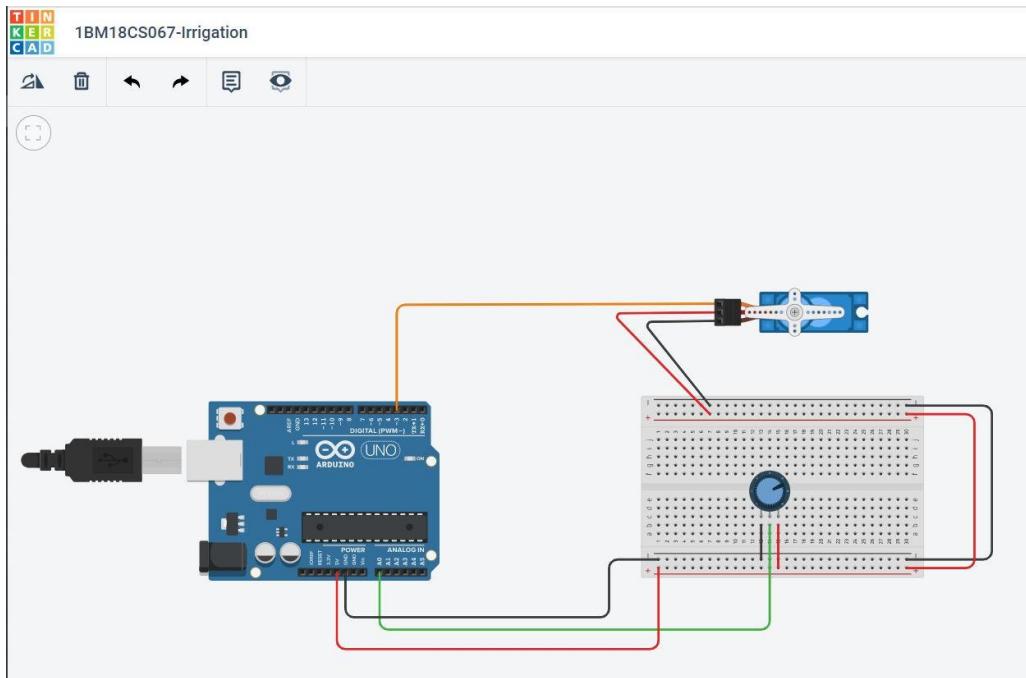
AIM

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




```
    delay(15);  
}  
for (pos = 180; pos >= 0; pos -= 1)  
{  
    myservo.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 – Samir kumar

Program No. – 15

Program Title – RGB LED

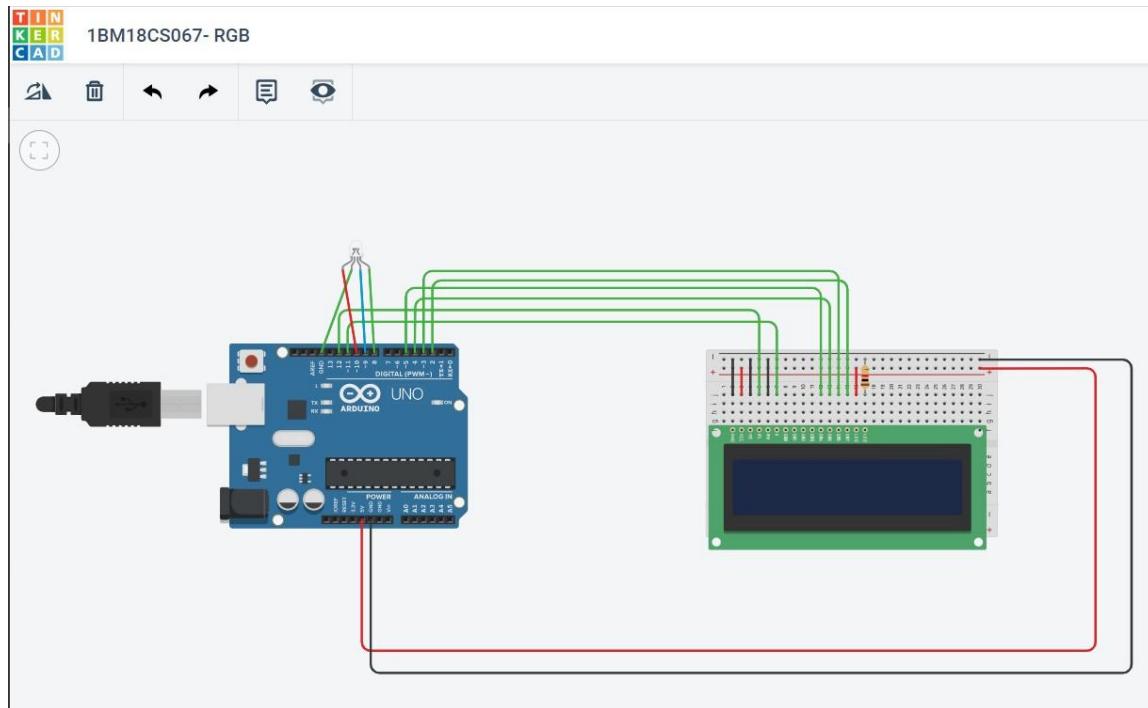
AIM

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

HARDWARES REQUIRED

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

CIRCUIT DIAGRAM



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).