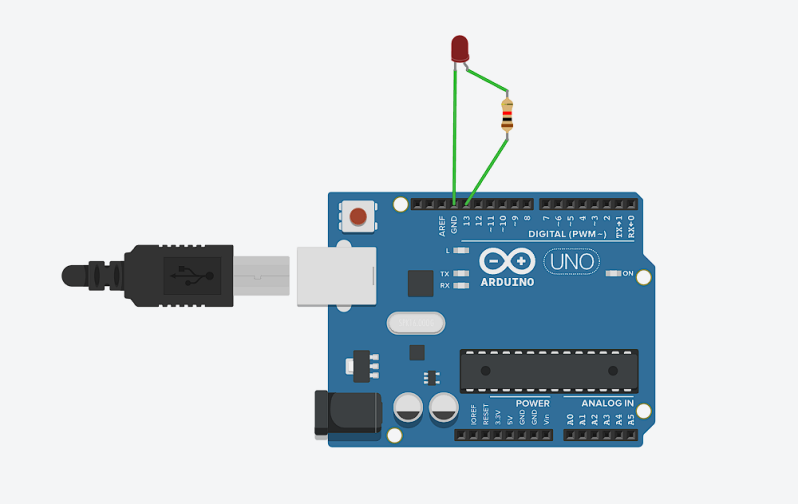
Experiment No. 1: Program to blink Arduino onboard LED.

Aim: Write a program to blink Arduino onboard LED, to interface external LED with Arduino and write a program to turn ON LED for 1 sec after every 2 seconds.

Components Used: Arduino Kit, LED, Resistor, Jumping wire

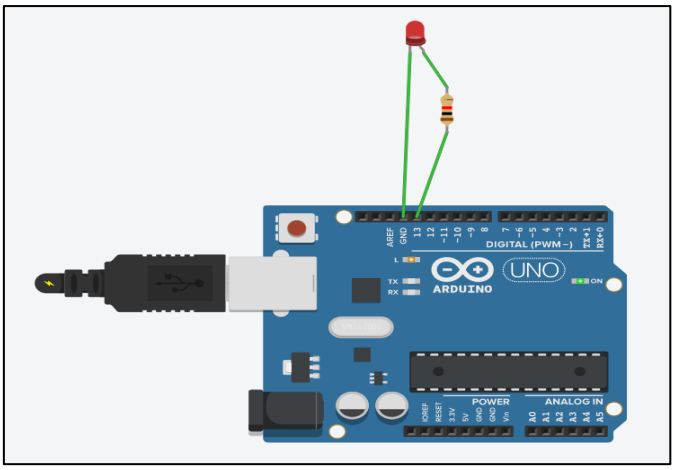
Circuit Diagram:



Code:

```
void setup ()
    {
        pinMode (13, OUTPUT);
    }
void loop ()
    {
        digitalWrite (13, HIGH);
        delay (1000); // Wait for 1000 millisecond(s)
        digitalWrite (13, LOW);
        delay (2000); // Wait for 2000 millisecond(s)
    }
```

Output:

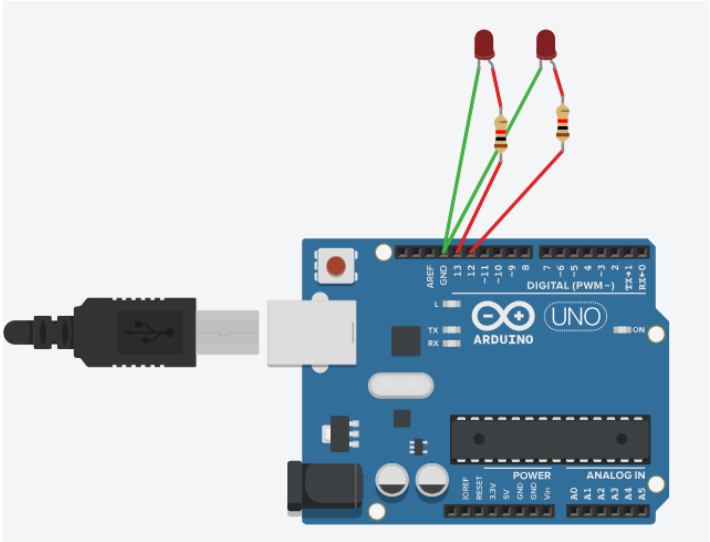


Experiment No. 2: Program to blink 2 Arduino onboard LEDs using 2 resistors.

Aim: Program to blink Arduino onboard two LED and two resistors to interface external LED with Arduino and write a program to turn ON LED for 1 sec after every 2 seconds.

Components Used: Arduino Kit, 2 LEDs, 2 Resistors, Jumping wire

Circuit Diagram:



Code:

Simultaneous blinking

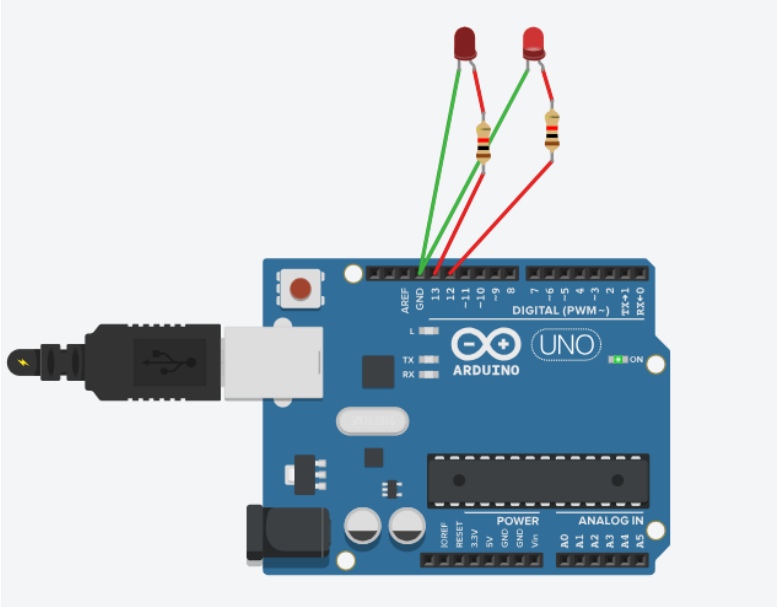
```
void setup ()
{
        pinMode (13, OUTPUT);
        pinMode (12, OUTPUT);
}
void loop ()
{
        digitalWrite (13, HIGH);
        digitalWrite (12, HIGH);
        delay (1000); // Wait for 1000 millisecond(s)
        digitalWrite (13, LOW);
        digitalWrite (12, LOW);
        delay (2000); // Wait for 2000 millisecond(s)
 }
```

Code: Alternative blinking

```
void setup ()
{
        pinMode (13, OUTPUT);
        pinMode (12, OUTPUT);
}
void loop ()
{
        digitalWrite (13, HIGH);
        delay (1000); // Wait for 1000 millisecond(s)
        digitalWrite (13, LOW);
        delay (2000); // Wait for 2000 millisecond(s)
        digitalWrite (12, HIGH);
        delay (1000);
```

```
        digitalWrite (12, LOW);
        delay (2000);
}
```

Output:

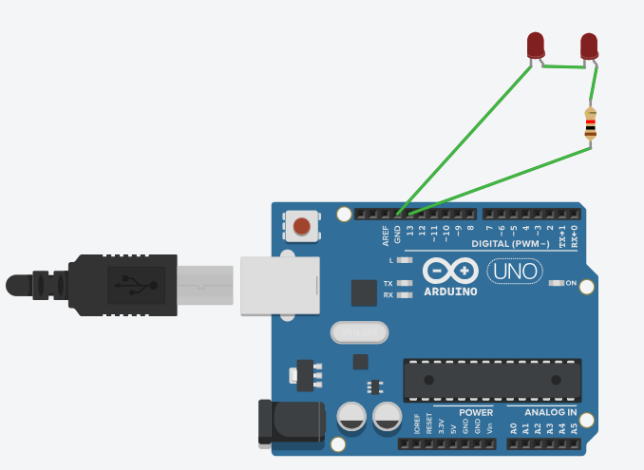


Experiment No. 3: Program to blink 2 Arduino onboard LEDs using single resistors.

Aim: Program to blink Arduino onboard two LED and one resistor to interface external LED with Arduino and write a program to turn ON LED for 1 sec after every 2 seconds.

Components Used: Arduino Kit, 2 LEDs, 1 Resistor, Jumping wire

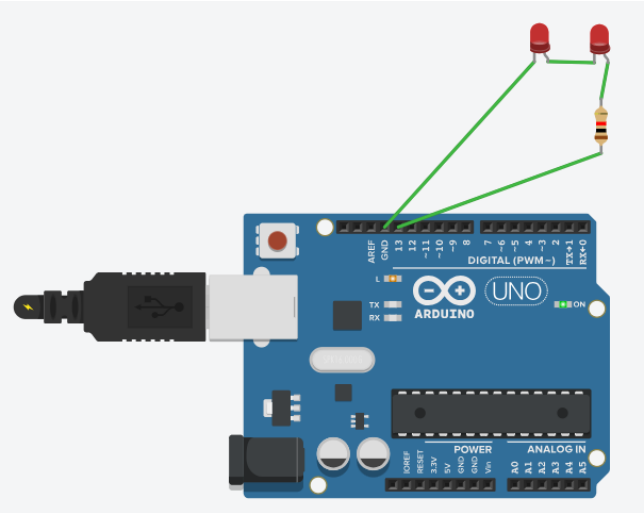
Circuit Diagram:



Code:

```
void setup ()
{
        pinMode (12, OUTPUT);
}
void loop ()
{
        digitalWrite (12, HIGH);
        delay (1000); // Wait for 1000 millisecond(s)
        digitalWrite (12, LOW);
        delay (2000); // Wait for 2000 millisecond(s)
}
```

Output:

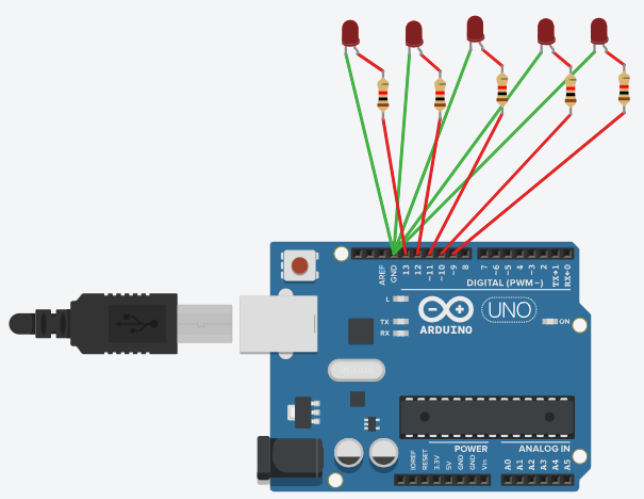


Experiment No. 4: To interface 5 LED‟s with Arduino and to blink 6 LEDs, one at a time, in a back & forth formation.

Aim: To interface 5 LED‟s with Arduino and write a program to blink 6 LEDs, one at a time, in a back-and-forth formation.

Components Used: Arduino Kit, 5 LEDs, 5 Resistors, Jumping wire

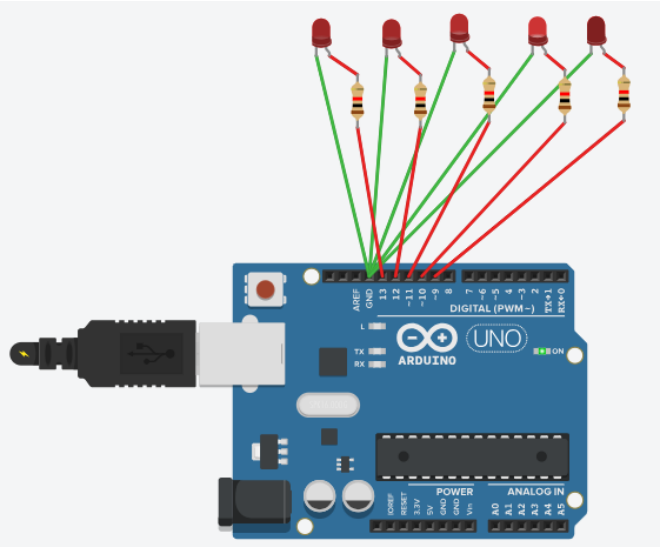
Circuit Diagram:



Code:

```
void setup ()
{
        pinMode (5, OUTPUT);
        pinMode (7, OUTPUT);
        pinMode (11, OUTPUT);
        pinMode (12, OUTPUT);
        pinMode (13, OUTPUT);
}
void loop ()
{
        digitalWrite (13, HIGH);
        delay (1000);
        digitalWrite (13, LOW);
        digitalWrite (12, HIGH);
        delay (1000);
        digitalWrite (12, LOW);
        digitalWrite (11, HIGH);
        delay (1000);
        digitalWrite (11, LOW);
        digitalWrite (7, HIGH);
        delay (1000);
        digitalWrite (7, LOW);
        digitalWrite (5, HIGH);
        delay (1000);
        digitalWrite (5, LOW);
}
```

Output:

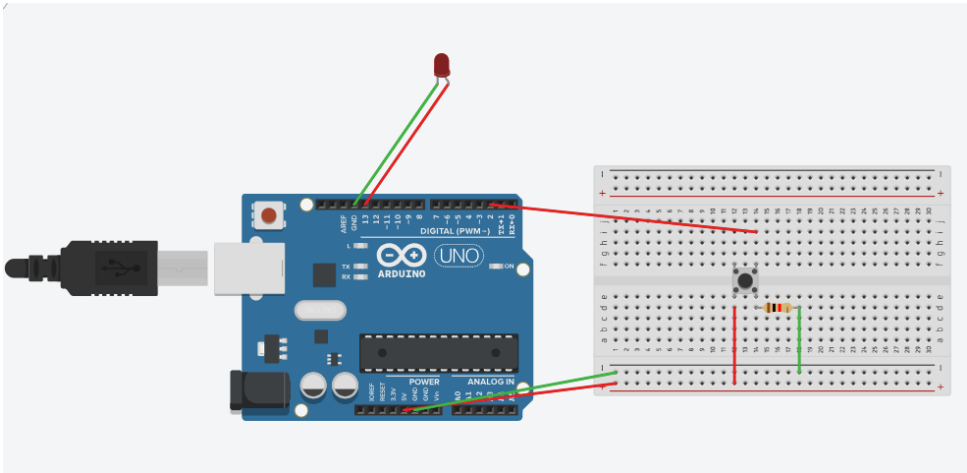


Experiment No. 5: To interface Push button with Arduino and to turn ON LED when push button is pressed.

Aim: To interface Push button with Arduino and write a program to turn ON LED when push button is pressed.

Components Used: Arduino Kit, 1 LED, 1 Resistor, 1 Push Button, Breadboard, Jumping wire

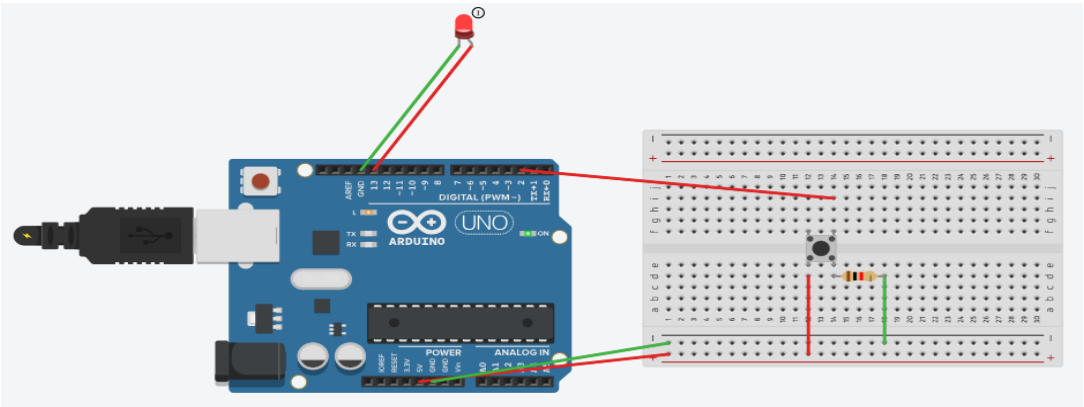
Circuit Diagram:



Code:

```
void setup ()
{
        pinMode (2, OUTPUT);
        pinMode (13, OUTPUT);
}
void loop ()
{
        if (digitalRead (2) ==1)
 {
        digitalWrite (13, HIGH);
 }
 else
 {
        digitalWrite (13, LOW);
 }
}
```

Output:

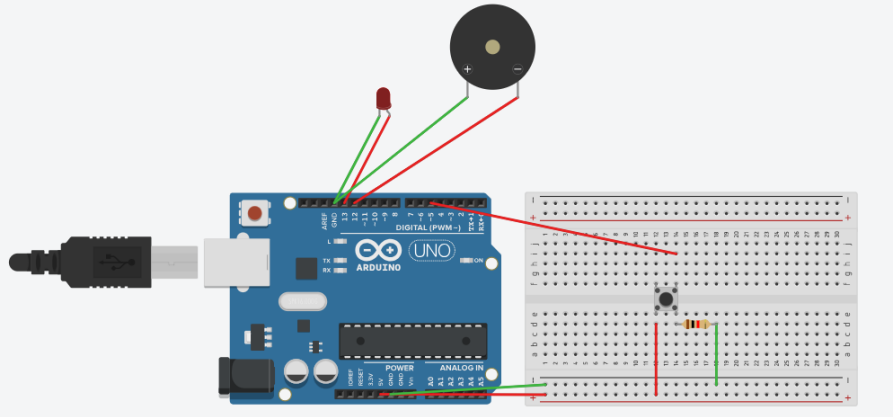


Experiment No. 6: To blink 1 led and buzz 1 buzzer after pushing a button.

Aim: To interface Push button, Speaker/buzzer with Arduino and write a program to turn ON LED and generate a note or tone when push button is pressed.

Components Used: Arduino Kit, 1 LED, 1 Resistor, 1 Push Button, 1 Buzzer, Breadboard, Jumping wire

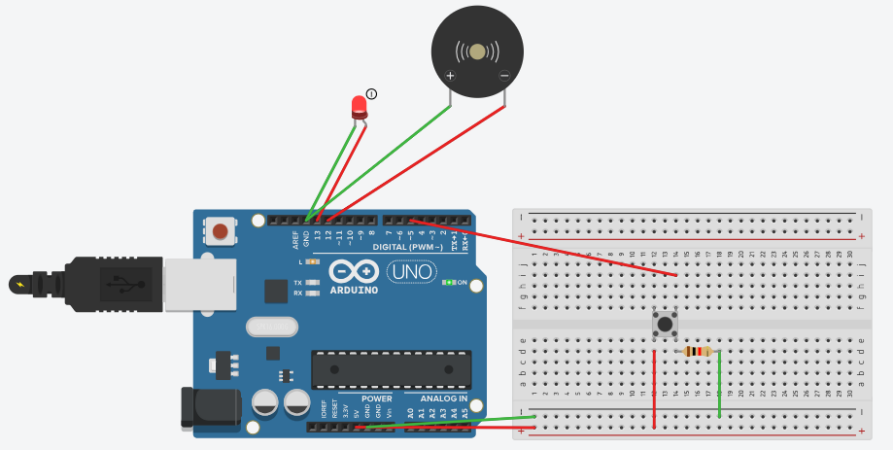
Circuit Diagram:



Code:

```
void setup ()
{
        pinMode (13, OUTPUT);
        pinMode (12, OUTPUT);
}
void loop ()
{
         if (digitalRead (5) ==1)
  {
        digitalWrite (13, HIGH);
        digitalWrite (12, HIGH);
  }
else {
        digitalWrite (13, LOW);
        digitalWrite (12, LOW);
  }
}
```

Output:

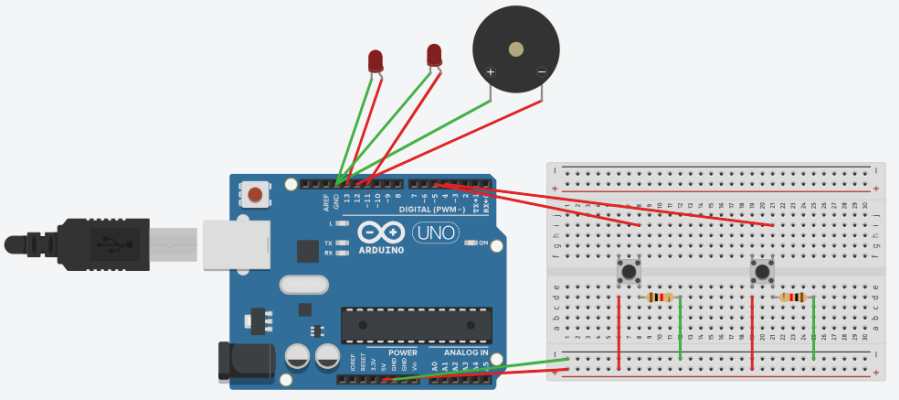


Experiment No. 7: To blink 1 led and buzz 1 buzzer after pushing a button and Another led and same buzzer after pushing 2nd button.

Aim: To interface 2 Push buttons, a Speaker with Arduino and write a program to turn ON LED and generate 2 different tones on two button keyboards.

Components Used: Arduino Kit, 2 LEDs, 2 Resistors, 2 Push Button, 1 Buzzer, Breadboard, Jumping wire

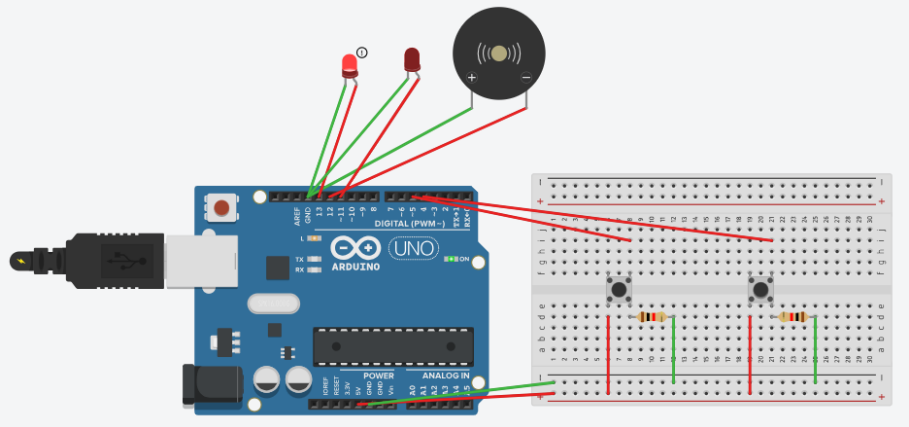
Circuit Diagram:



Code:

```
void setup ()
{
        pinMode (13, OUTPUT);
        pinMode (12, OUTPUT);
        pinMode (11, OUTPUT);
}
void loop ()
{
         if (digitalRead (5) ==1)
{
        digitalWrite (13, HIGH);
        digitalWrite (12, HIGH);
 }
  if (digitalRead (2) ==1)
{
        digitalWrite (11, HIGH);
        digitalWrite (12, HIGH);
 }
  else
{
        digitalWrite (13, LOW);
        digitalWrite (12, LOW);
        digitalWrite (11, LOW);
  }
}
```

Output:

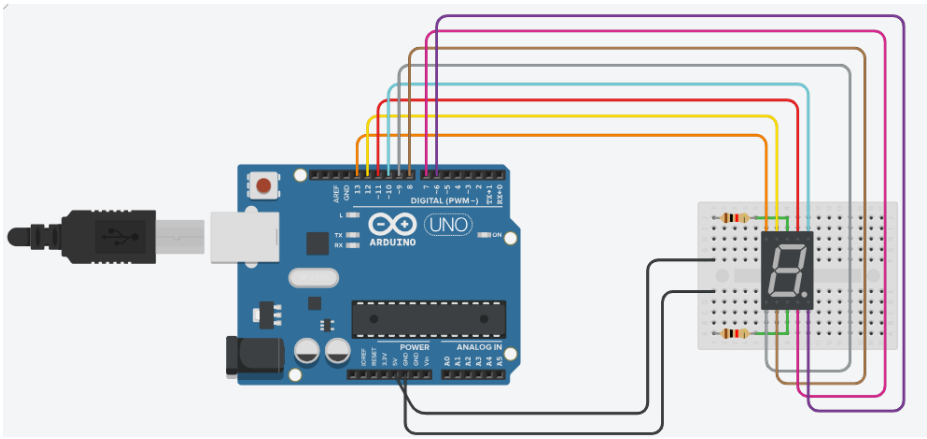


Experiment No. 8: To blink all segments of SSD to print numbers from 1-9

Aim: To interface Seven Segment Display (SSD) with Arduino and write a program to print numbers from 1 to 9 on SSD.

Components Used: Arduino Kit, 1 SSD, 2 Resistors, Breadboard, Jumping wire

Circuit Diagram:



Code:

```
unsigned const int A = 13;
unsigned const int B = 12;
unsigned const int C = 11;
unsigned const int D = 10;
unsigned const int E = 9;
unsigned const int F = 8;
unsigned const int G = 7;
unsigned const int H = 6;

void setup ()
{
        pinMode (A, OUTPUT);
        pinMode (B, OUTPUT);
        pinMode (C, OUTPUT);
        pinMode (D, OUTPUT);
        pinMode (E, OUTPUT);
        pinMode (F, OUTPUT);
        pinMode (G, OUTPUT);
        pinMode (H, OUTPUT);
}

void zero (void)
{
        digitalWrite (A, LOW);
        digitalWrite (B, HIGH);
        digitalWrite (C, HIGH);
        digitalWrite (D, HIGH);
        digitalWrite (E, HIGH);
        digitalWrite (F, HIGH);
        digitalWrite (G, HIGH);
        digitalWrite (H, LOW);
}
void one (void)
{
```

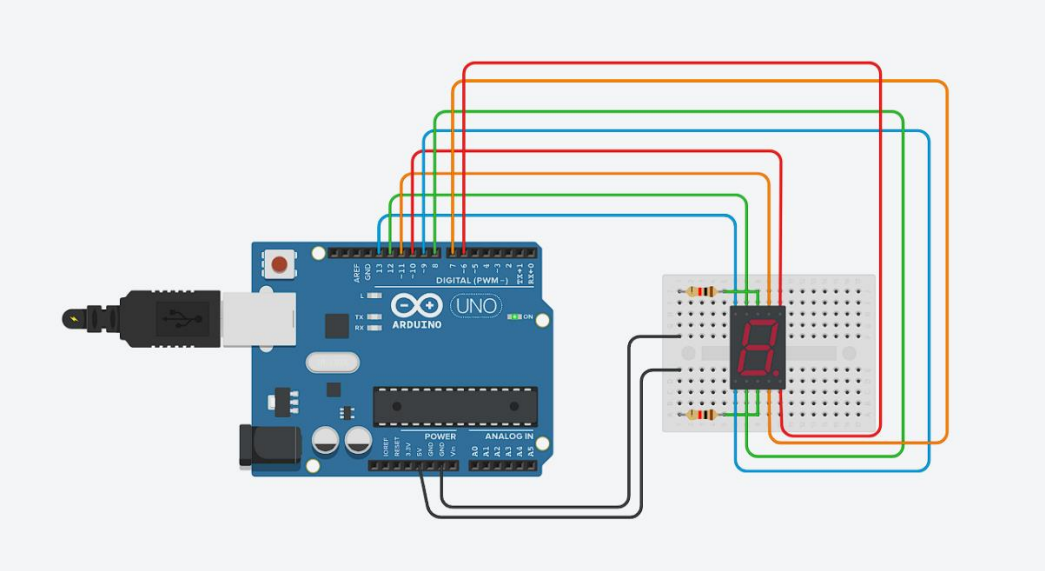
```
        digitalWrite (A, LOW);
        digitalWrite (B, LOW);
        digitalWrite (C, LOW);
        digitalWrite (D, HIGH);
        digitalWrite (E, LOW);
        digitalWrite (F, LOW);
        digitalWrite (G, HIGH);
        digitalWrite (H, LOW);
}
void two(void)
{
        digitalWrite (A, HIGH);
        digitalWrite (B, LOW);
        digitalWrite (C, HIGH);
        digitalWrite (D, HIGH);
        digitalWrite (E, HIGH);
        digitalWrite (F, HIGH);
        digitalWrite (G, LOW);
        digitalWrite (H, LOW);
}
void three(void)
{
        digitalWrite (A, HIGH);
        digitalWrite (B, LOW);
        digitalWrite (C, HIGH);
        digitalWrite (D, HIGH);
        digitalWrite (E, LOW);
        digitalWrite (F, HIGH);
        digitalWrite (G, HIGH);
        digitalWrite (H, LOW);
}
void four(void)
{
        digitalWrite (A, HIGH);
        digitalWrite (B, HIGH);
        digitalWrite (C, LOW);
        digitalWrite (D, HIGH);
        digitalWrite (E, LOW);
        digitalWrite (F, LOW);
        digitalWrite (G, HIGH);
        digitalWrite (H, LOW);
}
void five(void)
{
        digitalWrite (A, HIGH);
        digitalWrite (B, HIGH);
        digitalWrite (C, HIGH);
        digitalWrite (D, LOW);
        digitalWrite (E, LOW);
        digitalWrite (F, HIGH);
        digitalWrite (G, HIGH);
        digitalWrite (H, LOW);
}


void six(void)
```

```
{
        digitalWrite (A, HIGH);
        digitalWrite (B, HIGH);
        digitalWrite (C, HIGH);
        digitalWrite (D, LOW);
        digitalWrite (E, HIGH);
        digitalWrite (F, HIGH);
        digitalWrite (G, HIGH);
        digitalWrite (H, LOW);
}
void seven(void)
{
        digitalWrite (A, LOW);
        digitalWrite (B, LOW);
        digitalWrite (C, HIGH);
        digitalWrite (D, HIGH);
        digitalWrite (E, LOW);
        digitalWrite (F, LOW);
        digitalWrite (G, HIGH);
        digitalWrite (H, LOW);
}
void eight(void)
{
        digitalWrite (A, HIGH);
        digitalWrite (B, HIGH);
        digitalWrite (C, HIGH);
        digitalWrite (D, HIGH);
        digitalWrite (E, HIGH);
        digitalWrite (F, HIGH);
        digitalWrite (G, HIGH);
        digitalWrite (H, LOW);
}
void nine(void)
{
        digitalWrite (A, HIGH);
        digitalWrite (B, HIGH);
        digitalWrite (C, HIGH);
        digitalWrite (D, HIGH);
        digitalWrite (E, LOW);
        digitalWrite (F, HIGH);
        digitalWrite (G, HIGH);
        digitalWrite (H, LOW);
}
// Start
void loop(void)
{
        zero ();
        delay (1000);
        one ();
        delay (1000);
        two ();
        delay (1000);
        three ();
        delay (1000);
        four ();
        delay (1000);
```

```
        five ();
        delay (1000);
        six ();
        delay (1000);
        seven ();
        delay (1000);
        eight ();
        delay (1000);
        nine ();
        delay (1000);
}
```

Output:

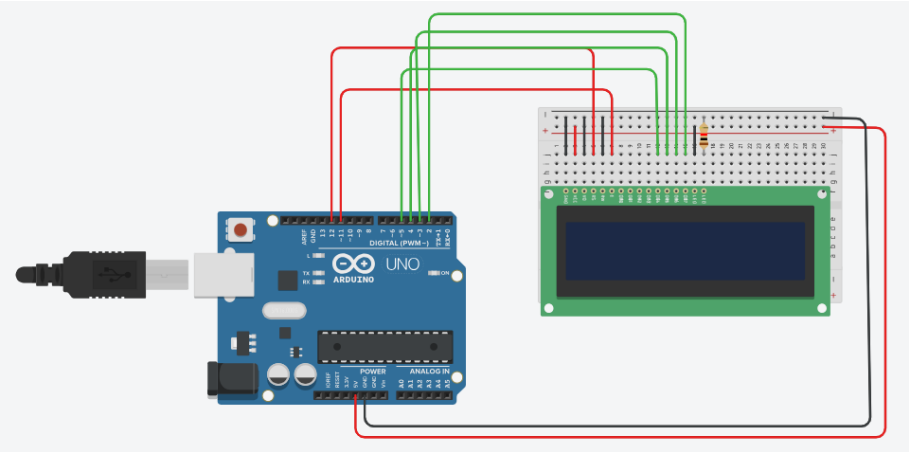


Experiment No. 9: Program to interface LCD with Arduino and to display messages

Aim: To interface LCD with Arduino and write a program to display message on LCD.

Components Used: Arduino Kit, 1 LCD, 1 Resistor, Breadboard, Jumping wire

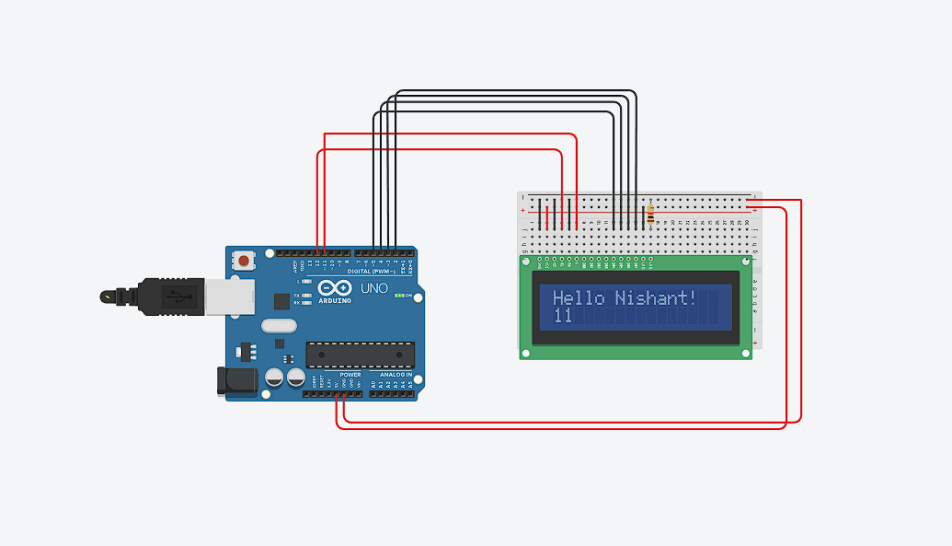
Circuit Diagram:



Code:

```
#include <LiquidCrystal.h>
const int rs = 12, en = 11, d4 = 5, d5 = 4, d6 = 3, d7 = 2;
LiquidCrystal lcd (rs, en, d4, d5, d6, d7);
void setup ()
{
        lcd.begin (16, 2);
        lcd.print ("Hello Nishant!");
}
void loop ()
{
        lcd.setCursor (0, 1);
        lcd.print (millis () / 1000);
}
```

Output:

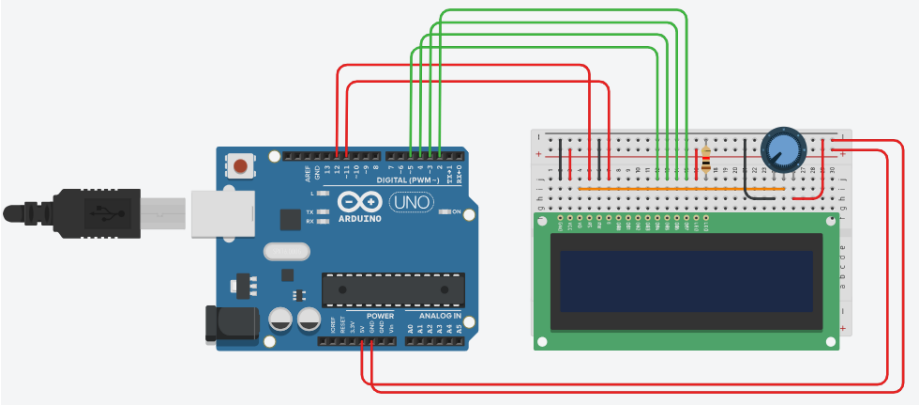


Experiment No.10: Program to interface LCD and potentiometer with Arduino and to display messages

Aim: To interface LCD, potentiometer with Arduino and write a program to display message on LCD.

Components Used: Arduino Kit, 1 LCD, 1 Resistor, 1 Potentiometer, Breadboard, Jumping wire

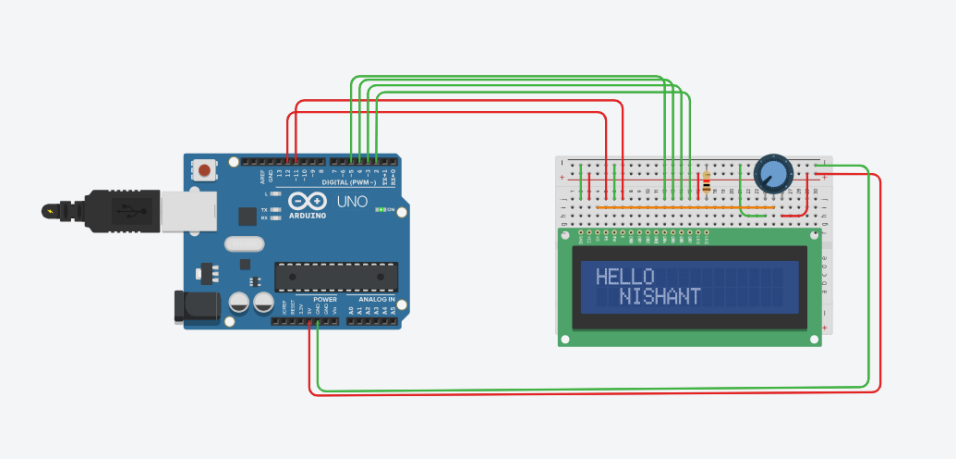
Circuit Diagram:



Code:

```
#include<LiquidCrystal.h>
LiquidCrystal lcd (12, 11, 5, 4, 3, 2);
void setup ()
{
        lcd.begin (16, 2);
}
void loop ()
{
lcd.setCursor (0,0);
lcd.print ("Hello");
lcd.setCursor (2,1);
lcd.print ("Nishant");
}
```

Output:

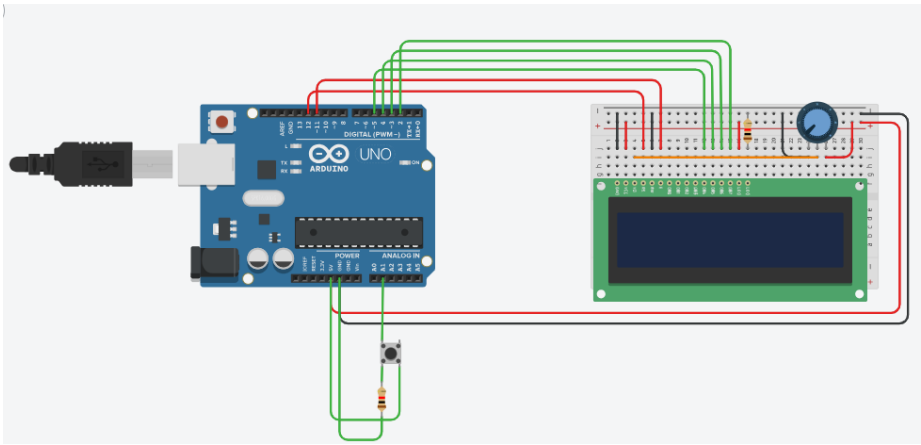


Experiment No.11: Program to interface LCD, push button, potentiometer with Arduino and to display a message

Aim: To interface LCD, push button, potentiometer with Arduino and write a program to display message on LCD when push button is pressed.

Components Used: Arduino Kit, 1 LCD, 2 Resistors, 1 Potentiometer, 1 Push Button, Breadboard, Jumping wire

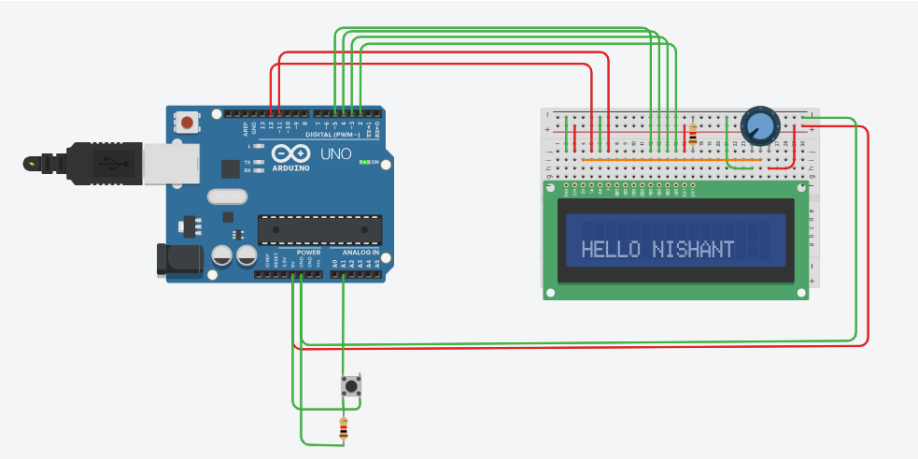
Circuit design:



Code:

```
#include<LiquidCrystal.h>
LiquidCrystal lcd (12, 11, 5, 4, 3, 2);
void setup ()
{
lcd.begin (16, 2);
pinMode (A1, INPUT);
}
void loop ()
{
lcd.setCursor (0,1);
  if (digitalRead (A1) ==HIGH)
  {
lcd.print ("Hello Nishant");
  }
}
```

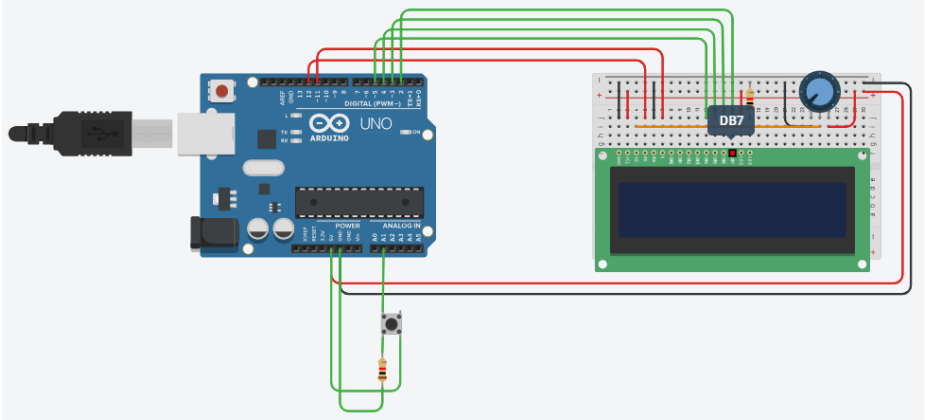
Output:



Experiment No.12: Program to interface LCD, push button, potentiometer with Arduino and to display no. of times push button is pressed

Name of the Program: To interface LCD, push button, potentiometer with Arduino and write a program to display the no. of times (count) the push button is pressed on LCD.
Components Used: Arduino Kit, 1 LCD, 2 Resistors, 1 Potentiometer, 1 Push Button, Breadboard, Jumping wire

Circuit design:

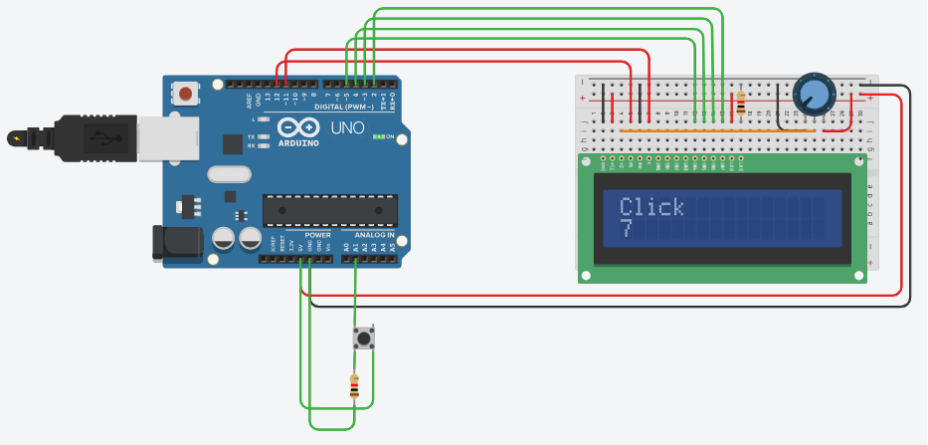


Code: -

```
#include <LiquidCrystal.h>
int val;
int count=0;
int press;
int Y;
const int rs = 12, en = 11, d4 = 5, d5 = 4, d6 = 3, d7 = 2;
LiquidCrystal lcd (rs, en, d4, d5, d6, d7);
void setup ()
{
lcd.begin (16, 2);
pinMode (A1, INPUT);
}
void loop ()
{
val=digitalRead(A1);
  if (val== HIGH)
{
    press=count++;
    Y=1*press+1; //y= mx +b
```

```
delay (250);
  }
lcd.setCursor (0, 0);
lcd.print ("Click");
lcd.setCursor (0, 1);
lcd.print(Y);
}
```

Output:

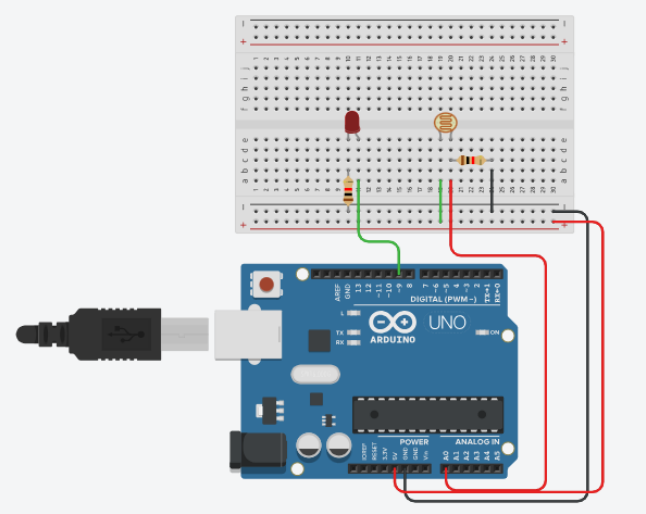


Experiment No.13: Program to interface LED and Photoresistor (LDR) with Arduino to increase and decrease the brightness of the LED

Aim: To interface LED, Photo resistor (LDR) with Arduino and write a program to increase and decrease the brightness of the LED based on the amount of light

Components Used : Arduino kit, 1 LED, 2 Resistors, 1 Photoresistor, Breadboard, Jumping wire

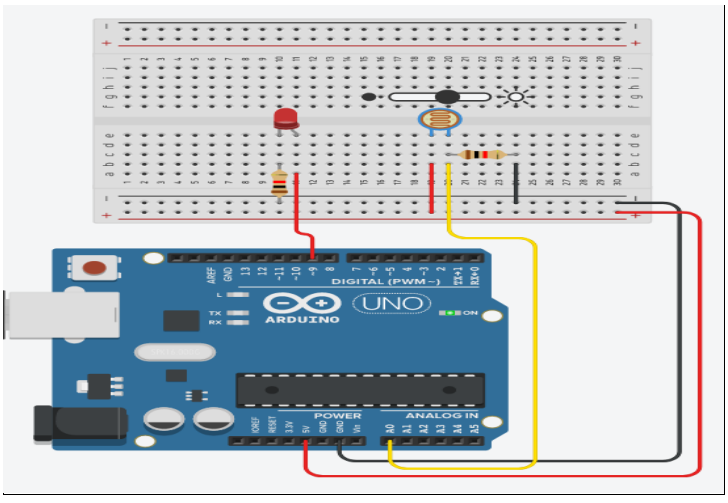
Circuit design:



Code:

```
int photosensor = 0;
void setup ()
{
        pinMode (A0, INPUT);
        Serial.begin (9600);
        pinMode (9, OUTPUT);
}
void loop ()
{
        photosensor = analogRead (A0);
        Serial.println (photosensor);
        analogWrite (9, map (photosensor, 0, 1023, 0, 255));
        delay (1000);
}
```

Output:

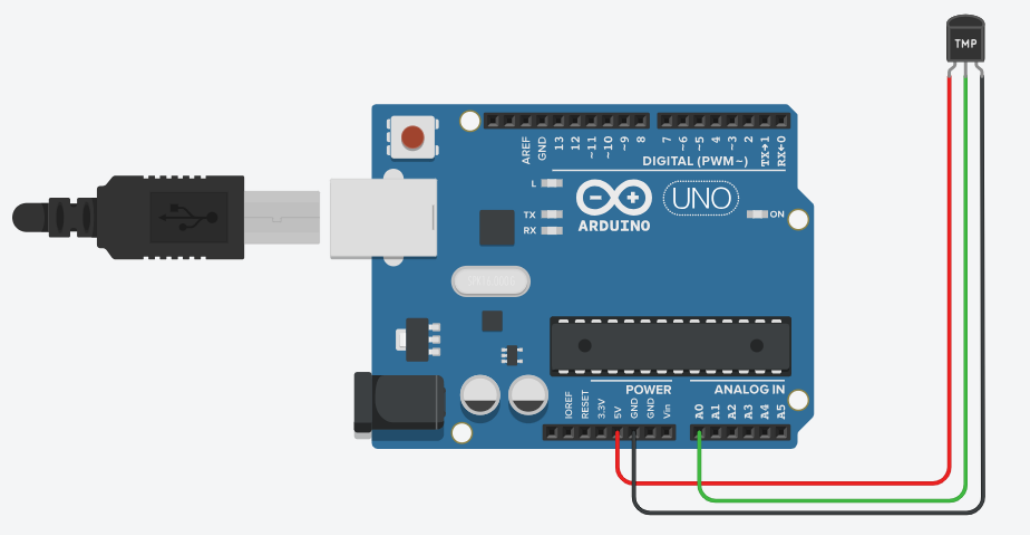


Experiment No.14: Program to interface DHT11 sensor with Arduino to display temperature and humidity data

Aim: To interface *DHT11* sensor with Arduino and write a program to display temperature and humidity data on serial monitor.

Components Used: Arduino kit, 1 DHT11 sensor, Jumping wire

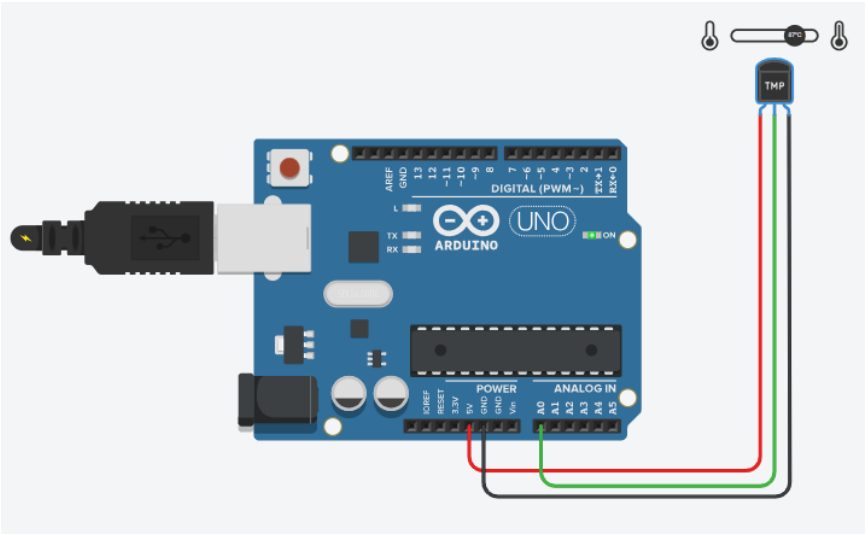
Circuit design:



Code:

```
void setup ()
{
        Serial.begin(9600);
}
void loop ()
{
  int sensorValue= analogRead(A0);
//serial.println (sensorValue);
  float volt=(sensorValue/1023.0) *4.9;
//serial.println(volt);
Delay (2000);
  float tempC= (volt -0.5) *100;//celcius
Serial.println(tempC);
Delay (2000);
}
```

Output:

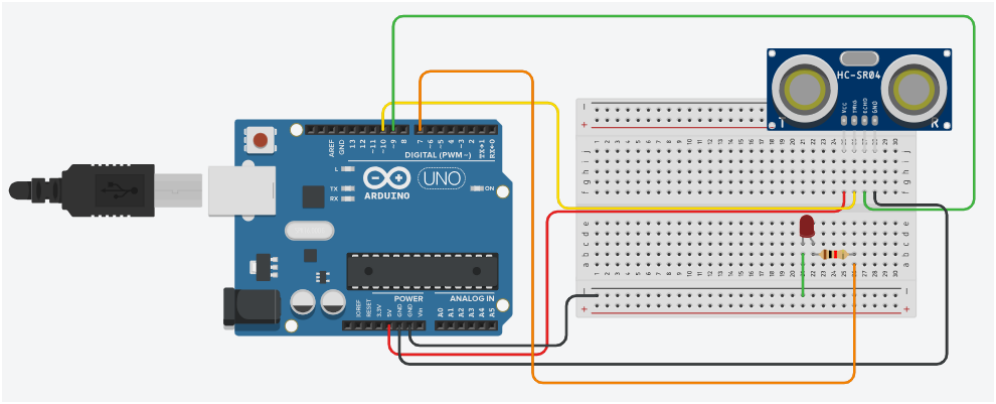


Experiment No.15: Program to interface PIR/ Ultrasonic sensor with Arduino depending on motion detection/sound detection.

Aim: To interface PIR/ Ultrasonic sensor with Arduino and write a program to turn on and off LED depending on motion detection/sound detection.

Components Used: Arduino kit, 1 Ultrasonic sensor, 1 LED, 1 Resistor, Breadboard, Jumping Wires.

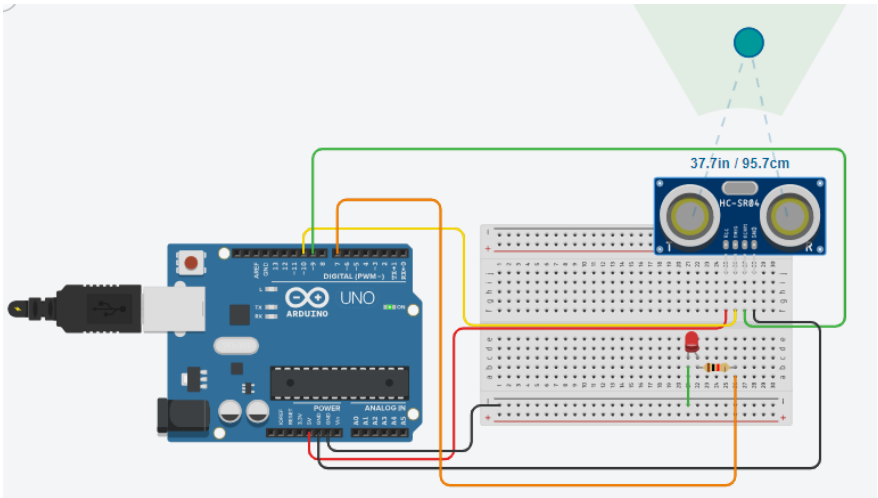
Circuit design:



Code:

```
void setup ()
{
  Serial.begin(9600);
  pinMode (10, OUTPUT);
  pinMode (7, OUTPUT);
  pinMode (9, INPUT);
  digitalWrite (10, HIGH);
}
void loop ()
{
  While (digitalRead (9) == HIGH)
          {
                    digitalWrite (7, LOW);
                    delay (3000);
                    digitalWrite (7, HIGH);
                   delay (2000);
          }
}
```

Output:

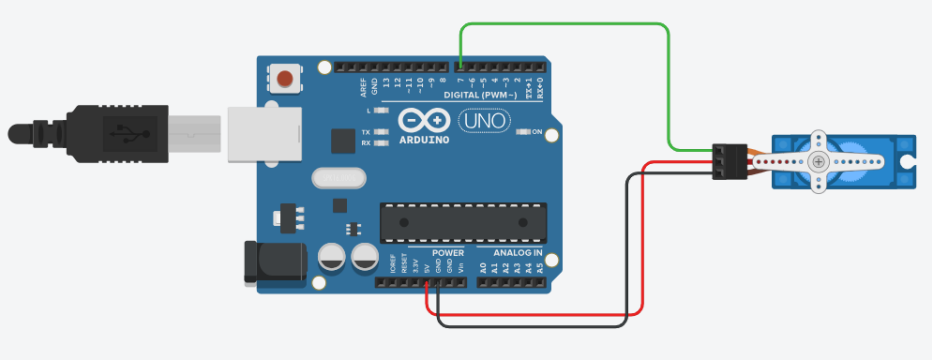


Experiment No.16: Program to interface servo motor/DC motor with Arduino to sweep a servo back and forth

Aim: To interface servo motor/DC motor with Arduino and write a program to sweep a servo back and forth through its full range of motion/ to control a DC motor.

Components Used: Arduino kit, 1 Micro Servo Motor, Jumping wire

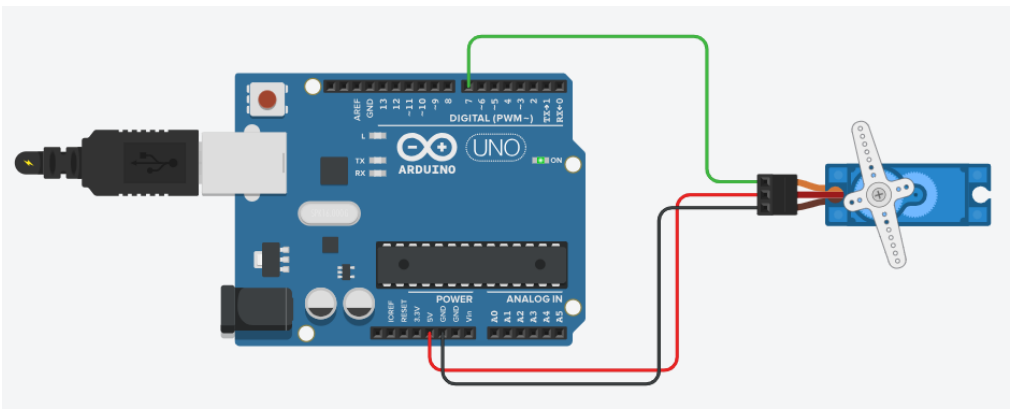
Circuit design:



Code:

```
#include<Servo.h>
Servo myservo;
void setup ()
{
        myservo.attach (7);
}
void loop ()
{
for (int ang=0; ang<180; ang++)
{
        myservo.write (ang);
        delay (50);
  }
for (int ang=180; ang>0; ang--)
{
        myservo.write (ang);
        delay (50);
  }
}
```

Output:

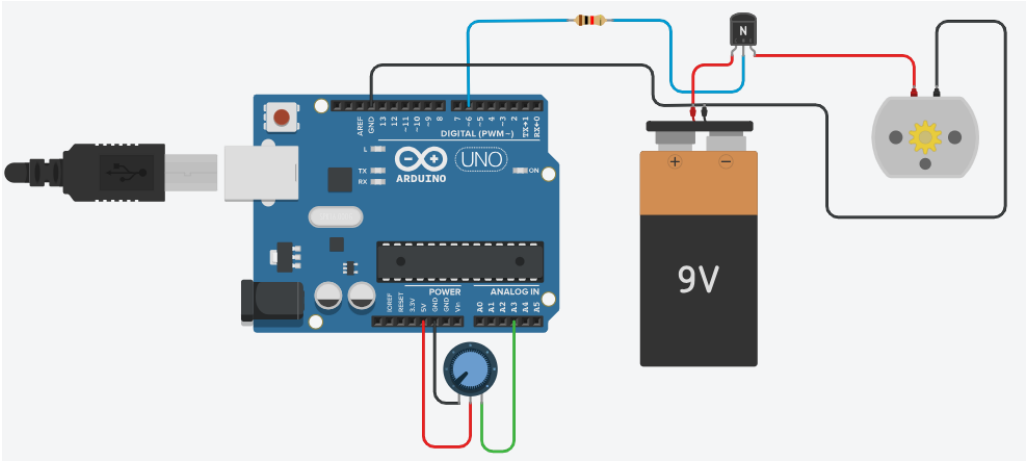


Experiment No.17: Program to interface DC motor with Arduino to control speed of a DC motor

Aim: To interface DC motor with Arduino and write a program to control Speed of a DC motor.

Components Used: Arduino kit,1 NPN Transistor, 1 Potentiometer, 1 9-Volt Battery, 1 Resistor, 1 DC Motor, Jumping wire.

Circuit design:



Code:

```
const int poten = A3;
int var;
void setup ()
{
        Serial.begin (9600);
        pinMode (6, OUTPUT);
}
void loop ()
{
        var = analogRead(poten);
        analogWrite (6, var);
        Serial.println(var);
}
```

Output: