

VIT-TBI ILD WORKSHOP 2024

Documentation Report - 2

Hardware

Name: Srija Koppar
Reg. No.: 23BCT0057

Introduction to Arduino: Implementation using hardware

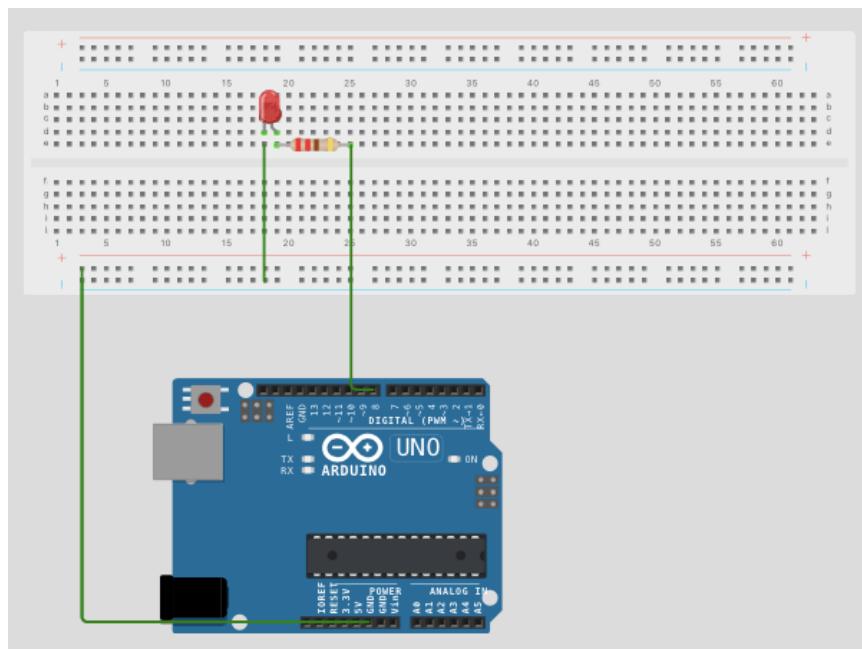
The following projects focus on basic Arduino applications and their implementation using hardware components. The main aim of the project is to understand how to make connections for various applications, how to upload the code from the Arduino IDE to the Arduino Uno board and view the output on the circuit built.

Project 1: LED Blinking [Hardware]

Aim: This project aims to make an LED bulb blink by connecting it to Arduino Uno.

Components Required: LED bulb, 220 Ohm resistor, Arduino Uno board, breadboard, 3 wires.

Circuit diagram:

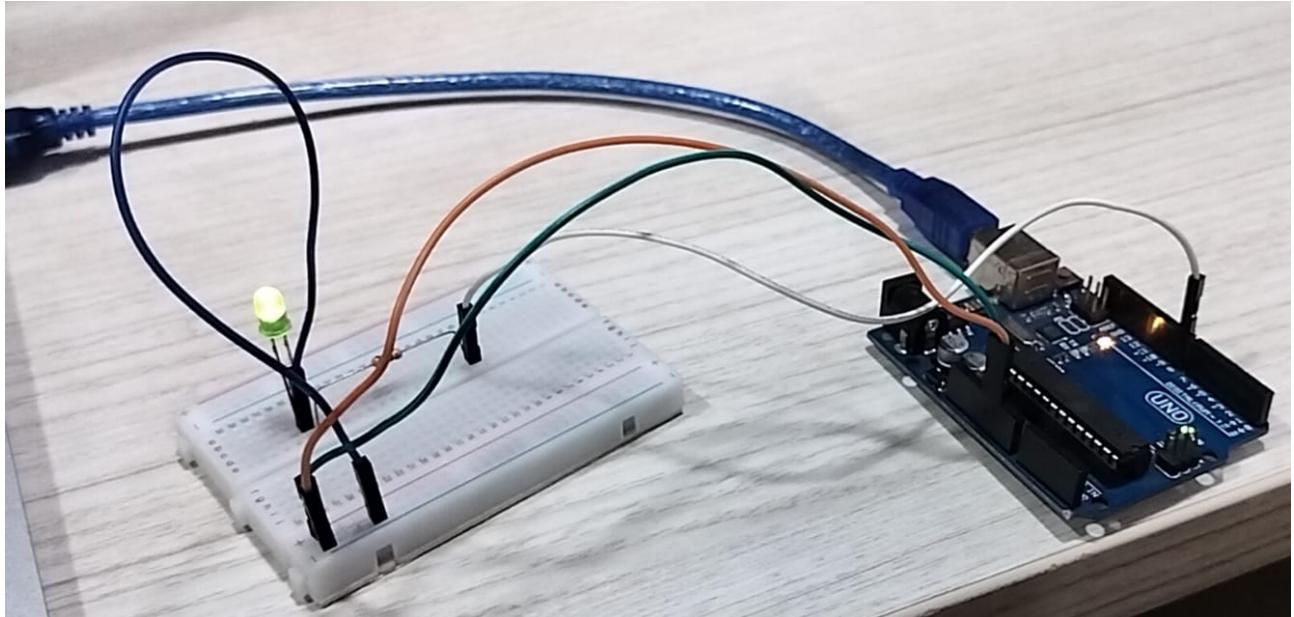


Code:

```
int redLed=8;  
void setup() {  
    // put your setup code here, to run once:  
    pinMode(redLed,OUTPUT);  
}  
  
void loop() {  
    // put your main code here, to run repeatedly:
```

```
digitalWrite(redLed,HIGH);
delay(9000);
digitalWrite(redLed,LOW);
delay(1000);
}
```

Output:



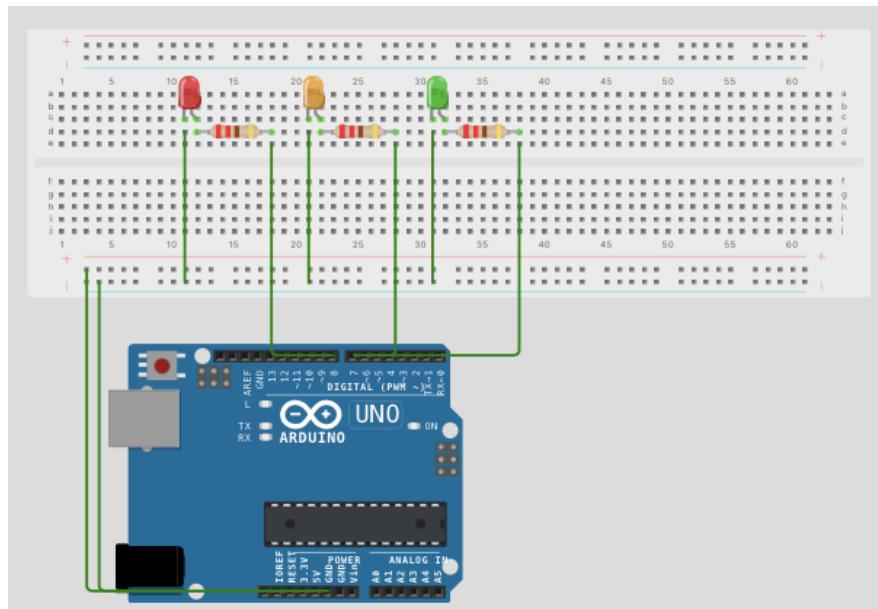
https://drive.google.com/file/d/1g8zxwXyDUIETclUe-wQAwH2gjzQm7VCz/view?usp=share_link

Project 2: Traffic Light [Hardware]

Aim: This project aims to create a traffic light setup by connecting three LEDs- red, orange and green in series to Arduino Uno. Here, we have used only 2 LEDs of two colours- green and white due to the availability constraints.

Components Required: 3 LED bulbs- red, orange and green colour, 3 220 Ohm resistors, Arduino Uno board, breadboard, 8 wires.

Circuit diagram:



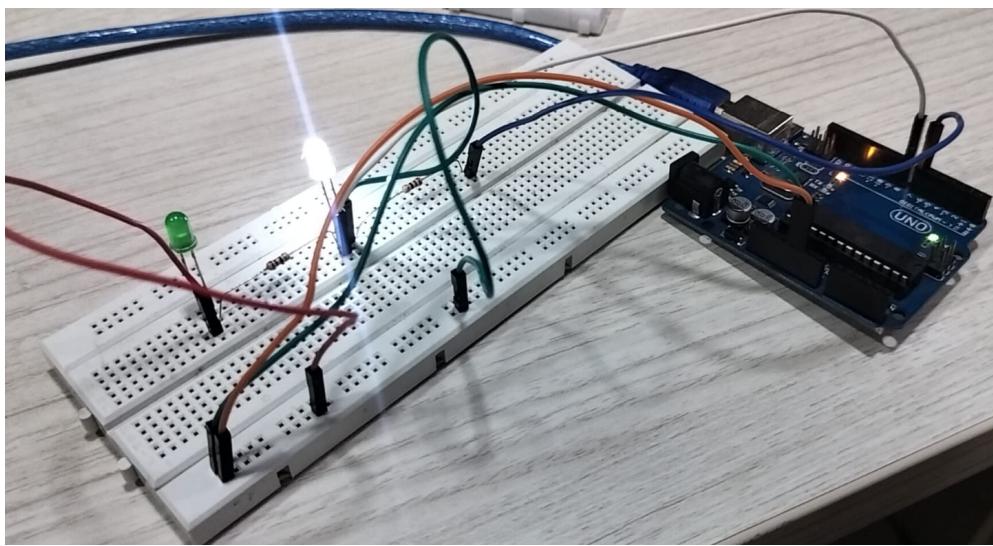
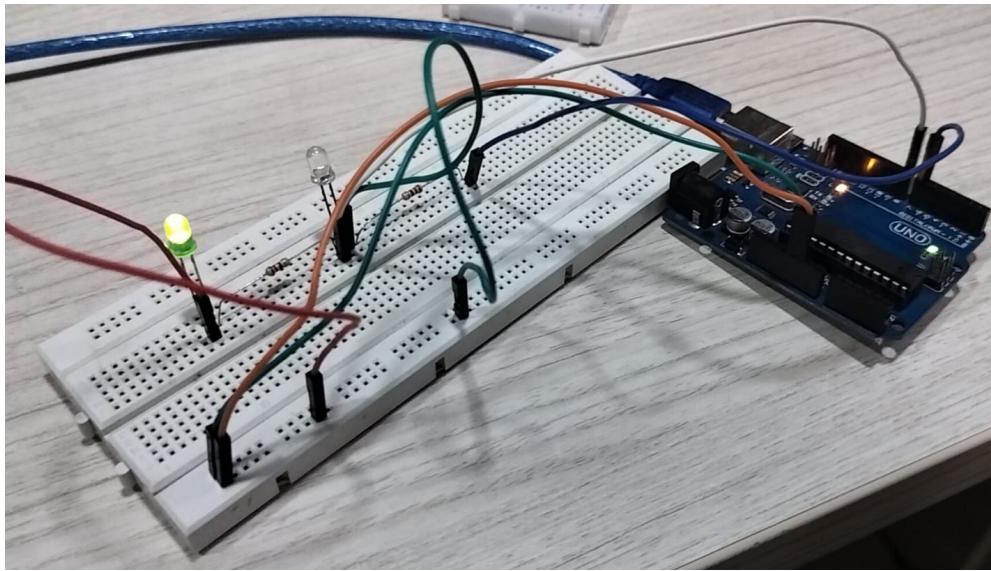
Code:

```
int redLed=8;
int orangeLed=7;
int greenLed=6;
void setup() {
    // put your setup code here, to run once:
    pinMode(redLed,OUTPUT);
    pinMode(orangeLed,OUTPUT);
    pinMode(greenLed,OUTPUT);
}

void loop() {
    // put your main code here, to run repeatedly:
    digitalWrite(redLed,HIGH);
    delay(5000);
    digitalWrite(redLed,LOW);
```

```
delay(1000);
digitalWrite(orangeLed,HIGH);
delay(5000);
digitalWrite(orangeLed,LOW);
delay(1000);
digitalWrite(greenLed,HIGH);
delay(5000);
digitalWrite(greenLed,LOW);
delay(1000);
}
```

Output:



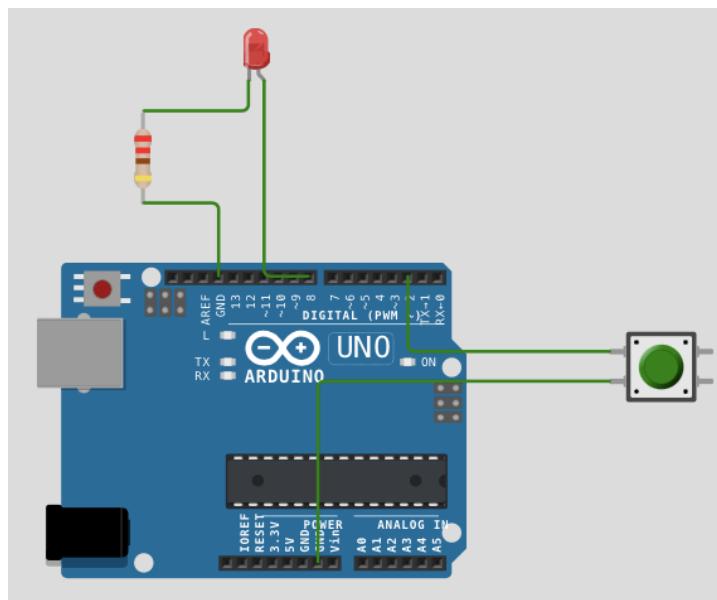
https://drive.google.com/file/d/199lZ-F9xieyuahHpN7AHJEj5B-Dxfus_/view?usp=share_link

Project 3: Button Controlled LED [Hardware]

Aim: This project aims to control the blinking of an LED bulb using a pushbutton by connecting them to Arduino Uno.

Components Required: LED bulb, 220 Ohm resistors, pushbutton, Arduino Uno board, breadboard, 5 wires.

Circuit diagram:



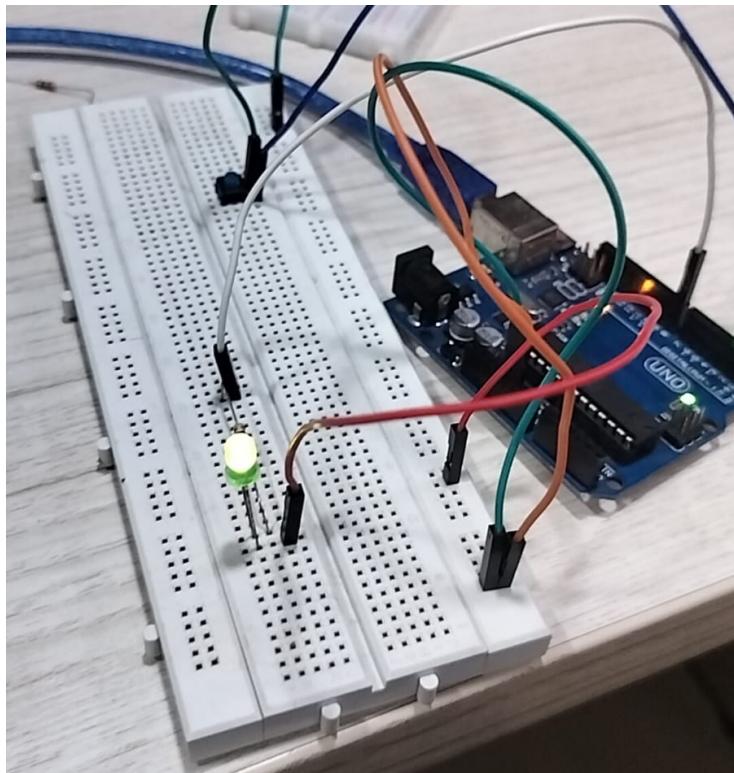
Code:

```
int ledPin=8;
int buttonPin=2;
bool buttonState = false;
void setup() {
    // put your setup code here, to run once:
    pinMode(ledPin,OUTPUT);
    pinMode(buttonPin, INPUT);
}

void loop() {
    // put your main code here, to run repeatedly:
    buttonState=digitalRead(buttonPin);
    if(buttonState==HIGH)
    {
        digitalWrite(ledPin,HIGH);
    }
}
```

```
else
{
  digitalWrite(ledPin,LOW);
}
}
```

Output:



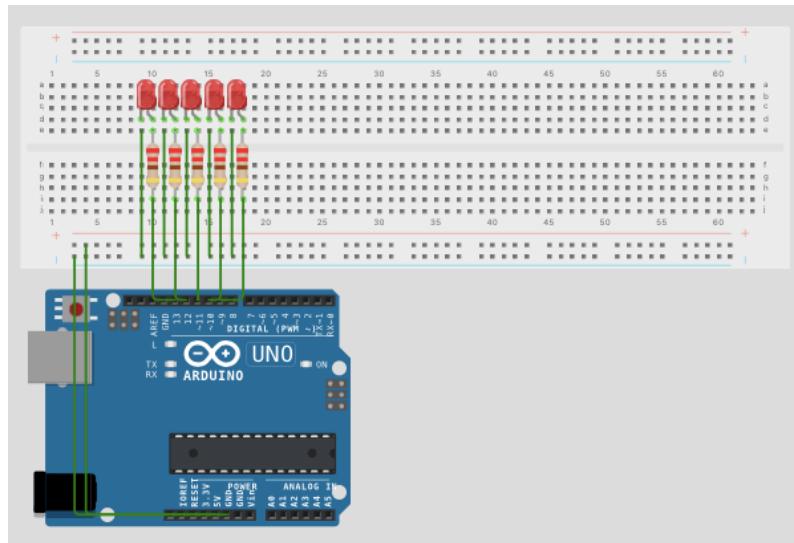
<https://drive.google.com/file/d/12Oq3fysypyCXyXQWEBLJHynltvcIUn5L/view?usp=sharing>

Project 4: Consecutive LED blinking [Hardware]

Aim: This project aims to make LED connected in a row blink consecutively by connecting them to Arduino Uno.

Components Required: 5 LED bulbs, 5 220 Ohm resistors, Arduino Uno board, breadboard, 12 wires.

Circuit diagram:



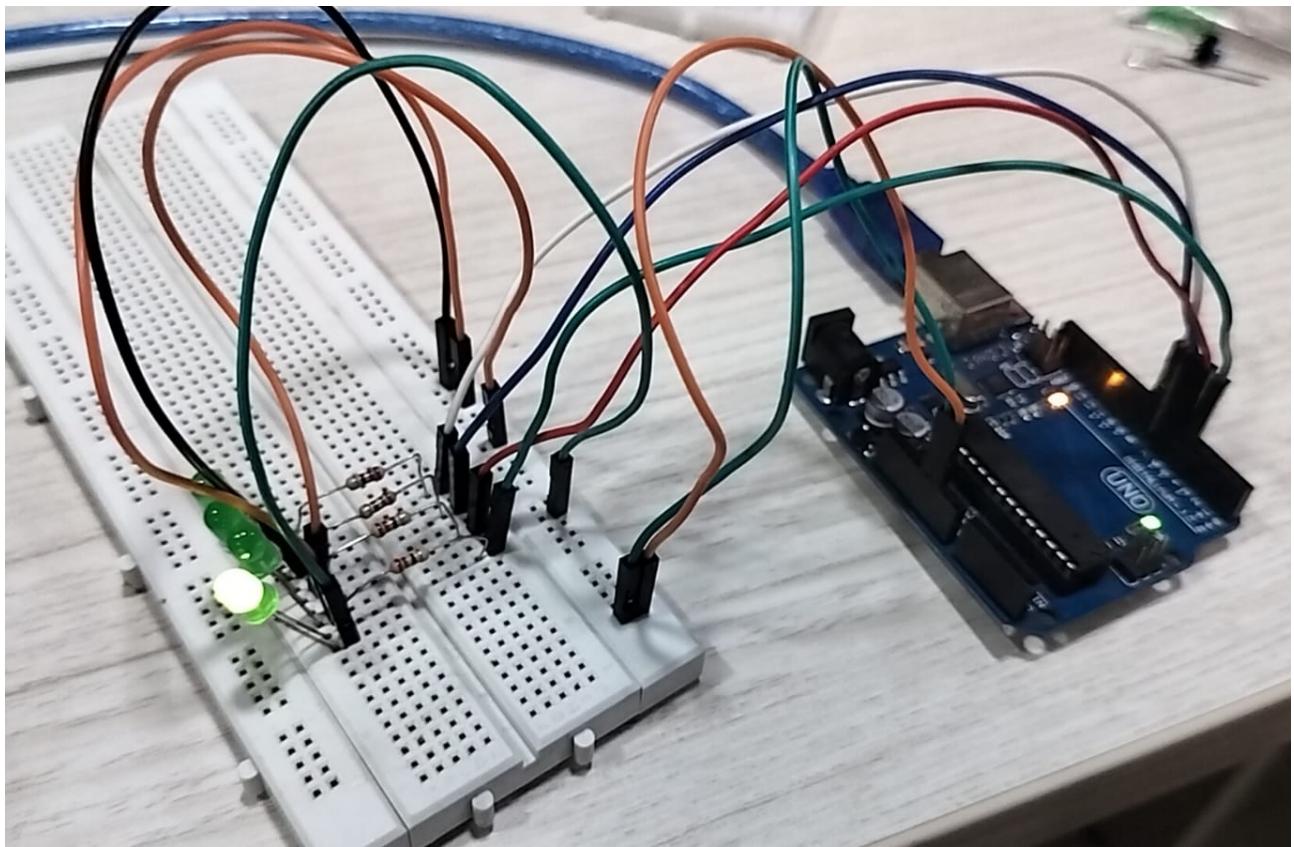
Code:

```
const int beta = 3950;

void setup() {
    Serial.begin(9600);
}

void loop() {
    int value = analogRead(A0);
    float celsius = 1 / (log(1 / (1023. / value - 1)) / beta + 1.0 / 298.15) - 273.15;
    Serial.print("Temp: ");
    Serial.print(celsius);
    Serial.print(" C\n");
    delay(1000);
}
```

Output:



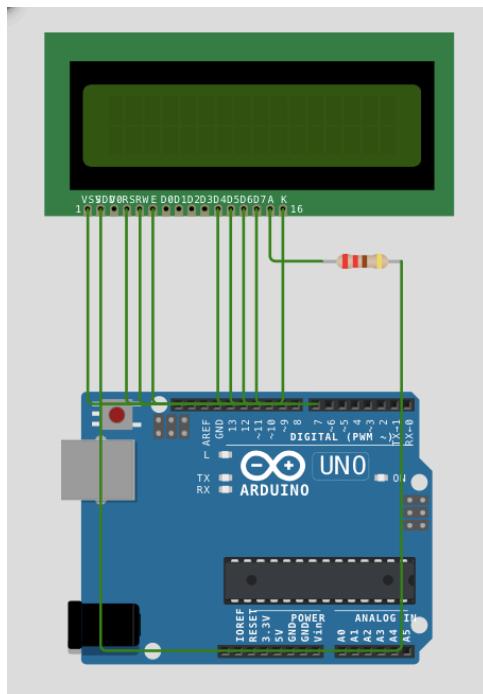
https://drive.google.com/file/d/1WcLkdLiokZAcY_-qyynn8XUcFUhpw22/view?usp=share_link

Project 5: Displaying “Hello World!” On LCD Screen [Hardware]

Aim: This project aims to make a display blinking Hello World message on an LCD Screen by connecting it to Arduino Uno.

Components Required: LCD (16x2) screen, Arduino Uno board, 4 wires.

Circuit diagram:

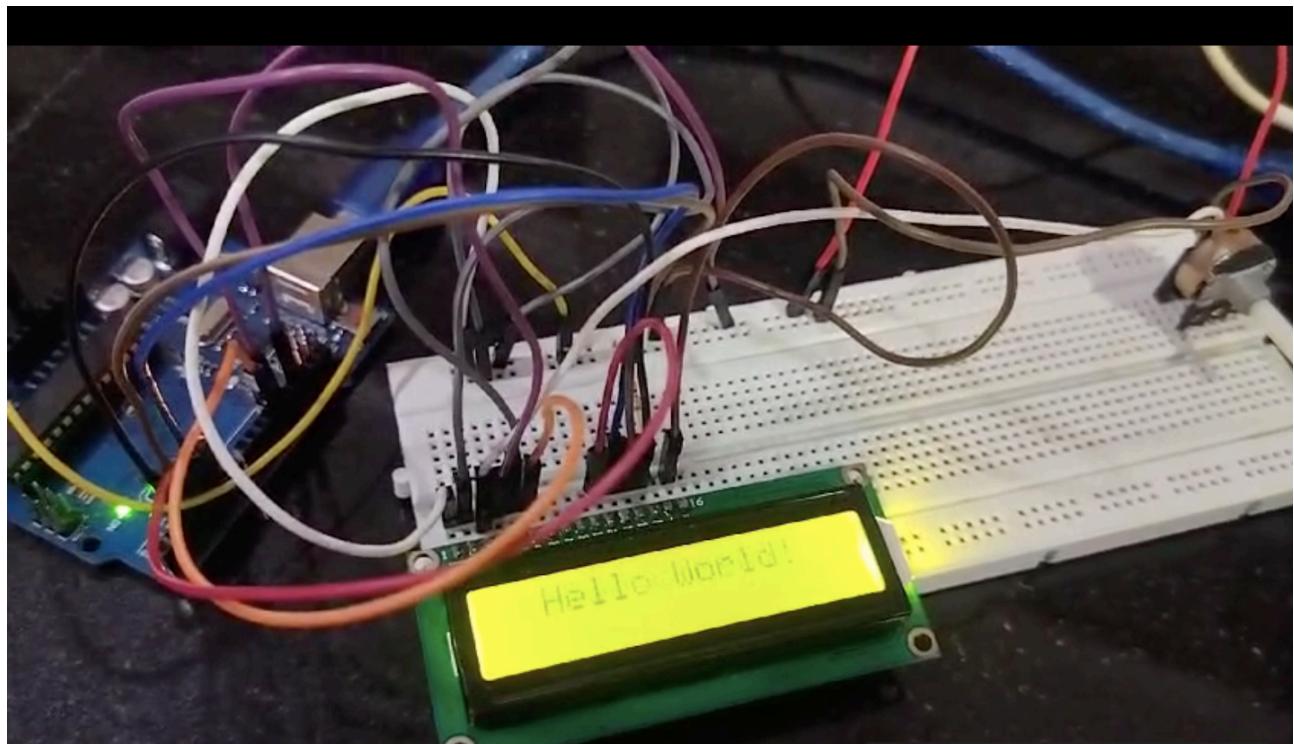


Code:

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(12,11,10,9,8,7);
void setup() {
    // put your setup code here, to run once:
    lcd.begin(16,2);
    lcd.print("Hello World!");
}

void loop() {
    // put your main code here, to run repeatedly:
    lcd.scrollDisplayLeft();
    delay(200);
    // lcd.scrollDisplayRight();
    // delay(200);
}
```

Output:



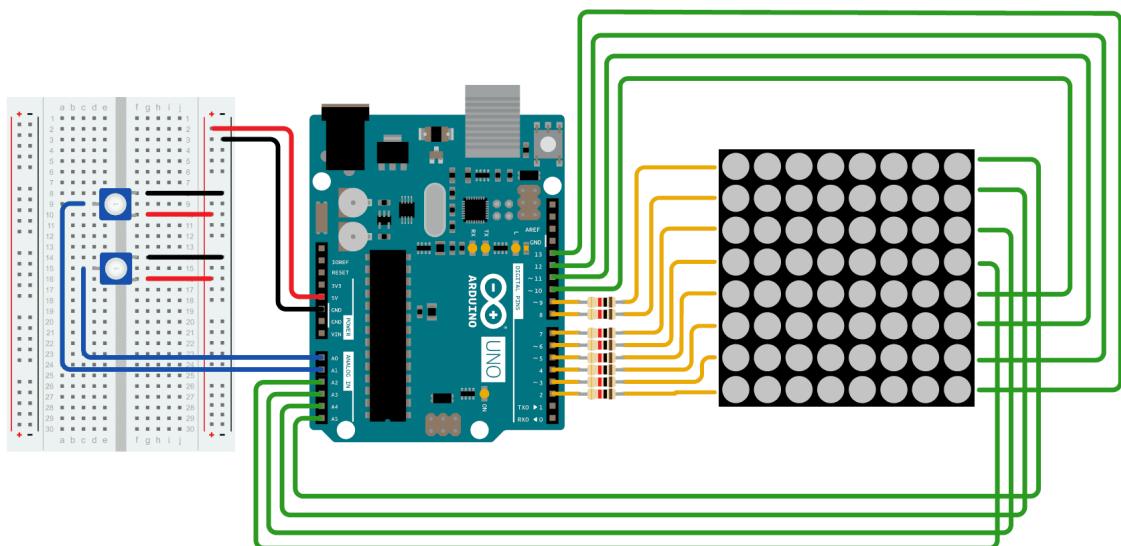
https://drive.google.com/file/d/19hznjFongie9YGwqz4eLrzv_ZrqOMswR/view?usp=share_link
https://drive.google.com/file/d/1awnxpJaf4FM3Si2JER32WAI8pFqE-X4g/view?usp=share_link

Project 5: Displaying name On LED matrix display [Hardware]

Aim: This project aims to make a display blinking name on an LED matrix display by connecting it to Arduino Uno.

Components Required: LED matrix display (32x8) screen, Arduino Uno board, 4 wires.

Circuit diagram:



Code:

```
#include <MD_Parola.h>
#include <MD_MAX72xx.h>
#include <SPI.h>

// Define hardware type and pin connections
#define HARDWARE_TYPE MD_MAX72XX::FC16_HW // Type of LED matrix hardware
#define MAX_DEVICES 4 // Number of 8x8 matrices in your setup

#define CS_PIN 10 // Chip Select pin

// Create a MD_Parola object
MD_Parola p = MD_Parola(HARDWARE_TYPE, CS_PIN, MAX_DEVICES);

void setup() {
    // Initialize the LED matrix
    p.begin();
```

```
p.setIntensity(5); // Set brightness (0-15)
p.displayText("Srija Koppar", PA_CENTER, 100, 0, PA_SCROLL_LEFT);
}

void loop() {
// Update the display
if (p.displayAnimate()) {
  p.displayReset();
}
}
```

Output:

