

U BRUNDA

Mail ID: [bu8486@srmist.edu.in](mailto:bu8486@srmist.edu.in)

## Project 1: RGB Pattern

### Abstract:

Creating color changing of RGB led using Arduino in Tinkercad platform.

### Introduction:

A color-changing RGB led is created by passing different inputs in the potentiometer through a well-structured code using an Arduino Uno.

### Methodology:

- Place a Arduino Uno, a breadboard, a LED RGB and three potentiometers.
- Place LED RGB on breadboard in pin number 4. Connect the LED RGB pin 4 to arduino input terminal of pin 11 through a normal wire.
- Similarly, to the pin numbers of 6 and 7 of LED RGB connect to the input terminals of arduino 10 and 9 respectively.
- To the 5 pin of LED RGB, place a resistor on the breadboard from d to f and connect from pin a to negative terminal of breadboard.
- To produce a RED color, a potentiometer is placed at pin 12. Similarly, to get GREEN and BLUE color potentiometers are placed at pin 18 and 24 respectively.
- For RED color potentiometer a wire is connected from negative terminal to f in the pin 12 on breadboard. Connect a wire from f to e and from a to **A0** in arduino. In pin 14 connect a wire from positive terminal to f.
- For BLUE color potentiometer a wire is connected from negative terminal to f of pin 18. Connect a wire from f to e and from a to **A1** in arduino. In pin 20, connect a wire from positive terminal to f.
- For GREEN color potentiometer a wire is connected from negative terminal to f of pin 24. Connect a wire from f to e and from a to **A2** in arduino. In pin 26, connect a wire from positive terminal to f.
- Connect a normal wire from **GND** of Arduino to negative terminal of breadboard and **5V** of Arduino to positive terminal of breadboard.

### CODE:

```
const int red = 11;
const int blue = 10;
const int green = 9;

void setup(){
  pinMode(red, OUTPUT); // Red
  pinMode(blue, OUTPUT); // Blue
  pinMode(green, OUTPUT); // Green
  pinMode(A0, INPUT); // Red
  pinMode(A1, INPUT); // Blue
  pinMode(A2, INPUT); // Green
  Serial.begin(9600);
}

void loop() {
  int r_value = analogRead(A0);
  int b_value = analogRead(A1);
  int g_value = analogRead(A2);

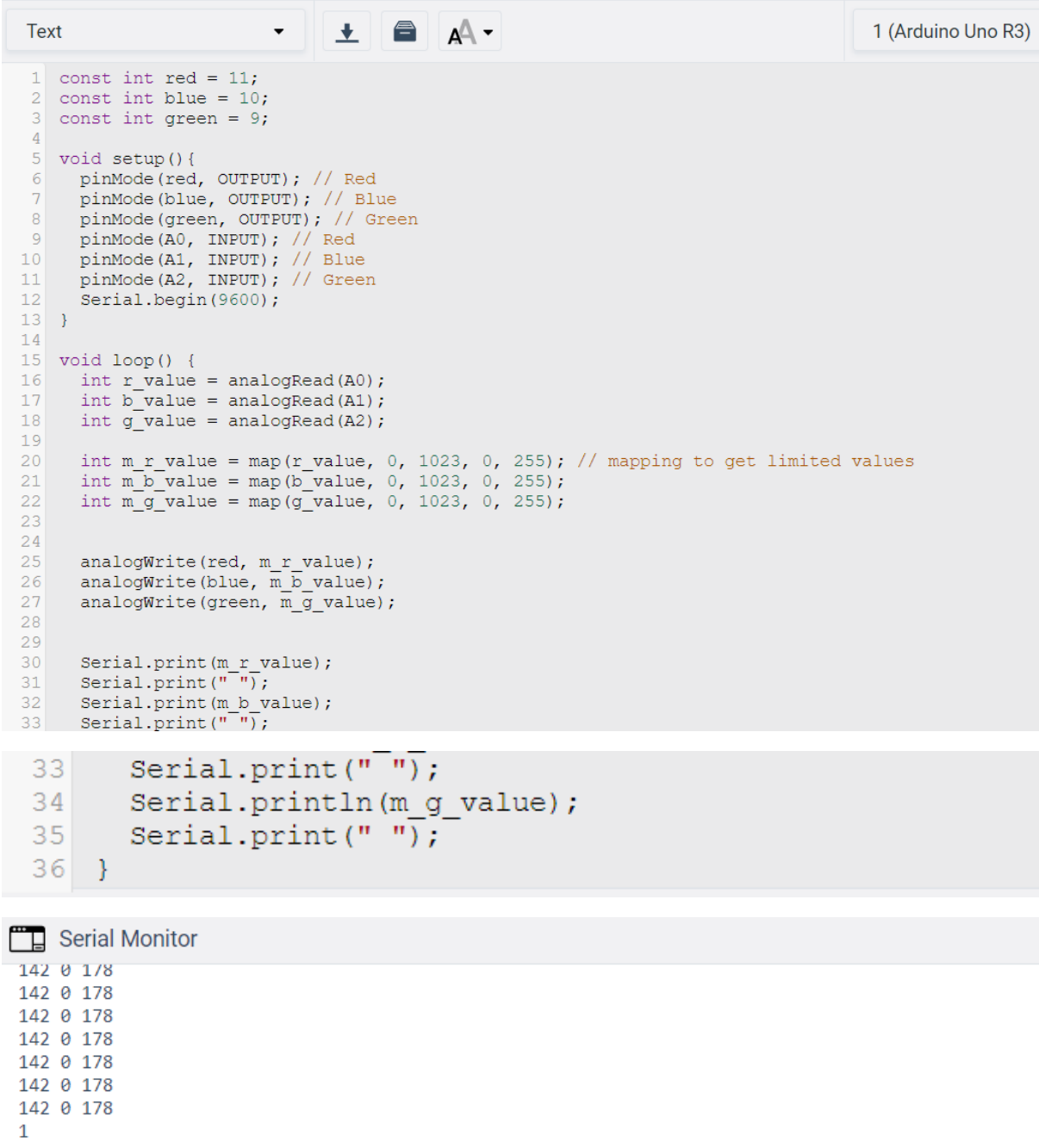
  int m_r_value = map(r_value, 0, 1023, 0, 255); // mapping to get limited values
  int m_b_value = map(b_value, 0, 1023, 0, 255);
  int m_g_value = map(g_value, 0, 1023, 0, 255);

  analogWrite(red, m_r_value);
  analogWrite(blue, m_b_value);
  analogWrite(green, m_g_value);

  Serial.print(m_r_value);
```

```
Serial.print(" ");  
  
Serial.print(m_b_value);  
  
Serial.print(" ");  
  
Serial.println(m_g_value);  
  
Serial.print(" ");  
  
}
```

## SCREENSHOTS:

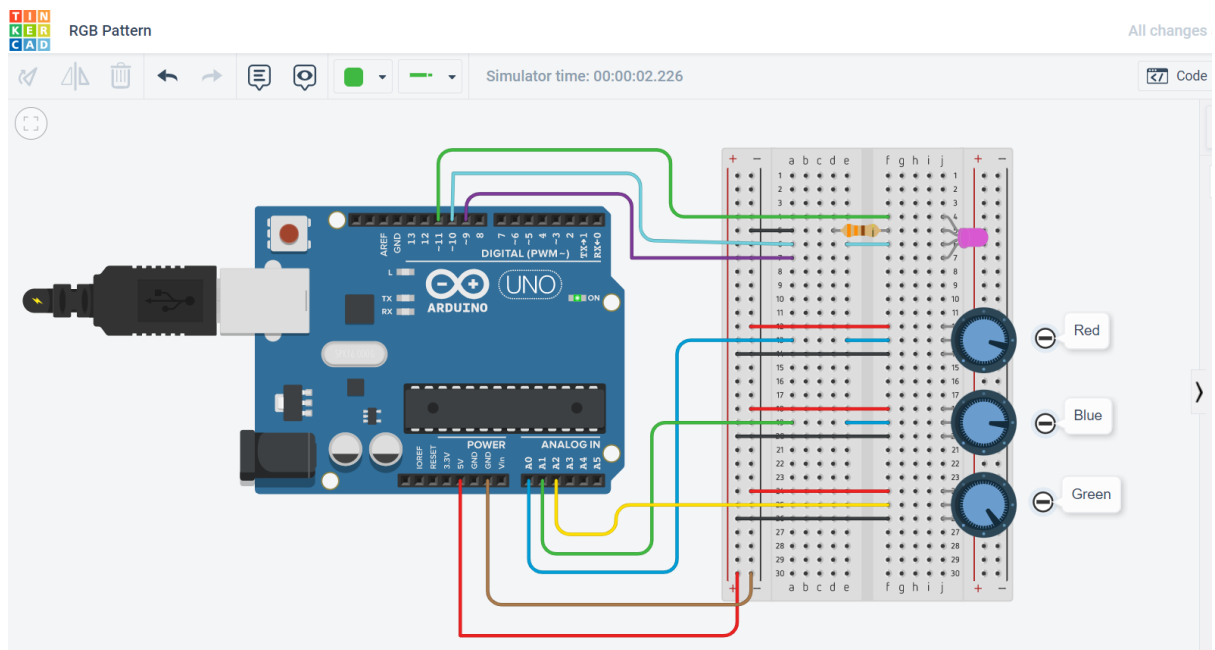
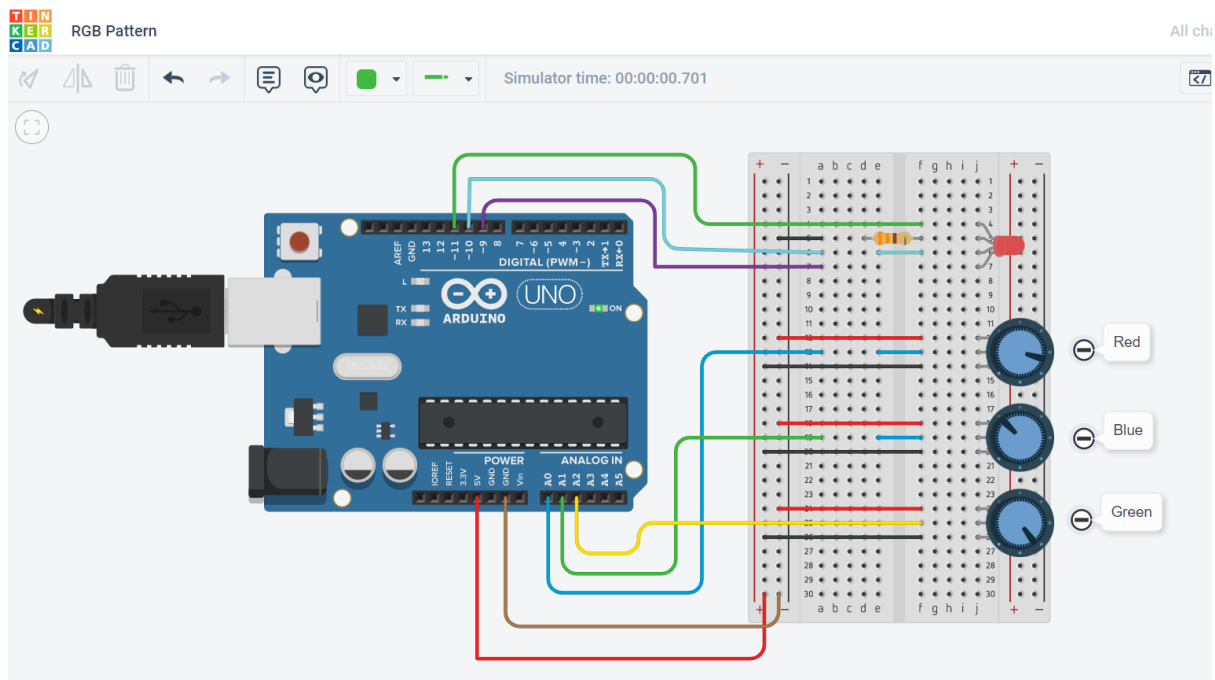


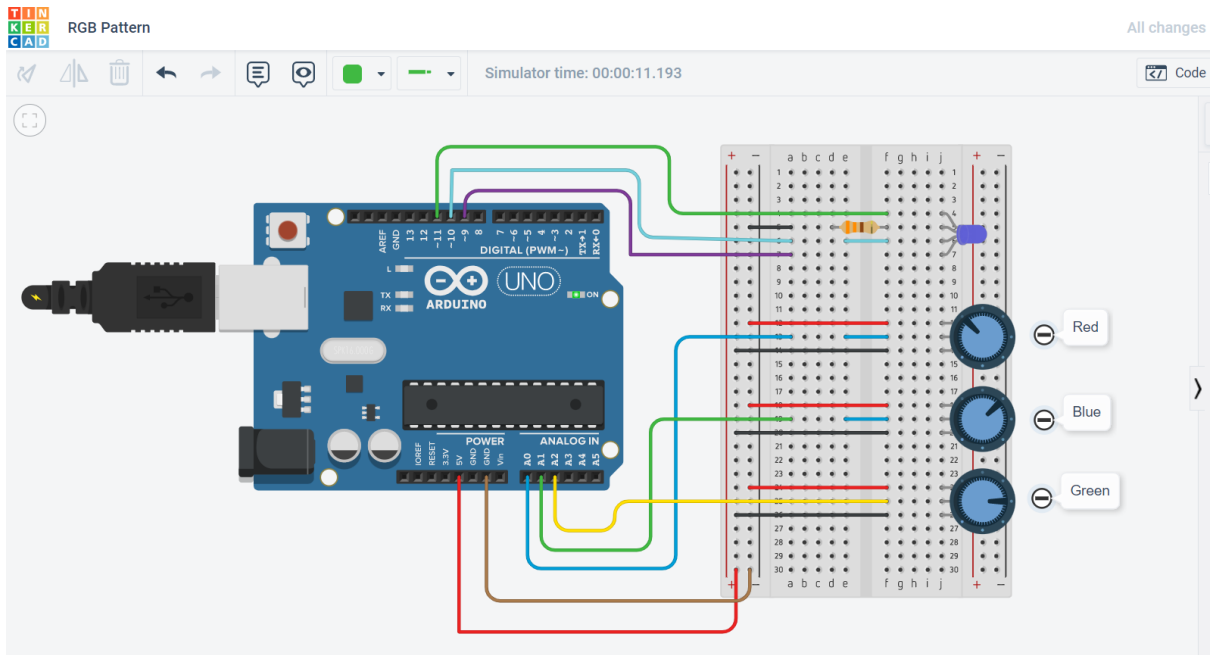
The screenshot displays the Arduino IDE interface. The top toolbar includes a dropdown menu set to 'Text', icons for saving, opening, and a font size selector, and a board selector set to '1 (Arduino Uno R3)'. The main code editor shows the following code:

```
1  const int red = 11;  
2  const int blue = 10;  
3  const int green = 9;  
4  
5  void setup() {  
6    pinMode(red, OUTPUT); // Red  
7    pinMode(blue, OUTPUT); // Blue  
8    pinMode(green, OUTPUT); // Green  
9    pinMode(A0, INPUT); // Red  
10   pinMode(A1, INPUT); // Blue  
11   pinMode(A2, INPUT); // Green  
12   Serial.begin(9600);  
13 }  
14  
15 void loop() {  
16   int r_value = analogRead(A0);  
17   int b_value = analogRead(A1);  
18   int g_value = analogRead(A2);  
19  
20   int m_r_value = map(r_value, 0, 1023, 0, 255); // mapping to get limited values  
21   int m_b_value = map(b_value, 0, 1023, 0, 255);  
22   int m_g_value = map(g_value, 0, 1023, 0, 255);  
23  
24  
25   analogWrite(red, m_r_value);  
26   analogWrite(blue, m_b_value);  
27   analogWrite(green, m_g_value);  
28  
29  
30   Serial.print(m_r_value);  
31   Serial.print(" ");  
32   Serial.print(m_b_value);  
33   Serial.print(" ");  
34   Serial.println(m_g_value);  
35   Serial.print(" ");  
36 }
```

Below the code editor, the 'Serial Monitor' window is open, showing the output of the code. It displays a series of values: 142, 0, 178, repeated six times, followed by a line number 1.

## LED COLOR CHANGING FOR DIFFERENT INPUTS:





### Conclusion:

RGB LED has implemented successfully with well-structured code and with proper connections of components.