U BRUNDA

Mail ID: 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.

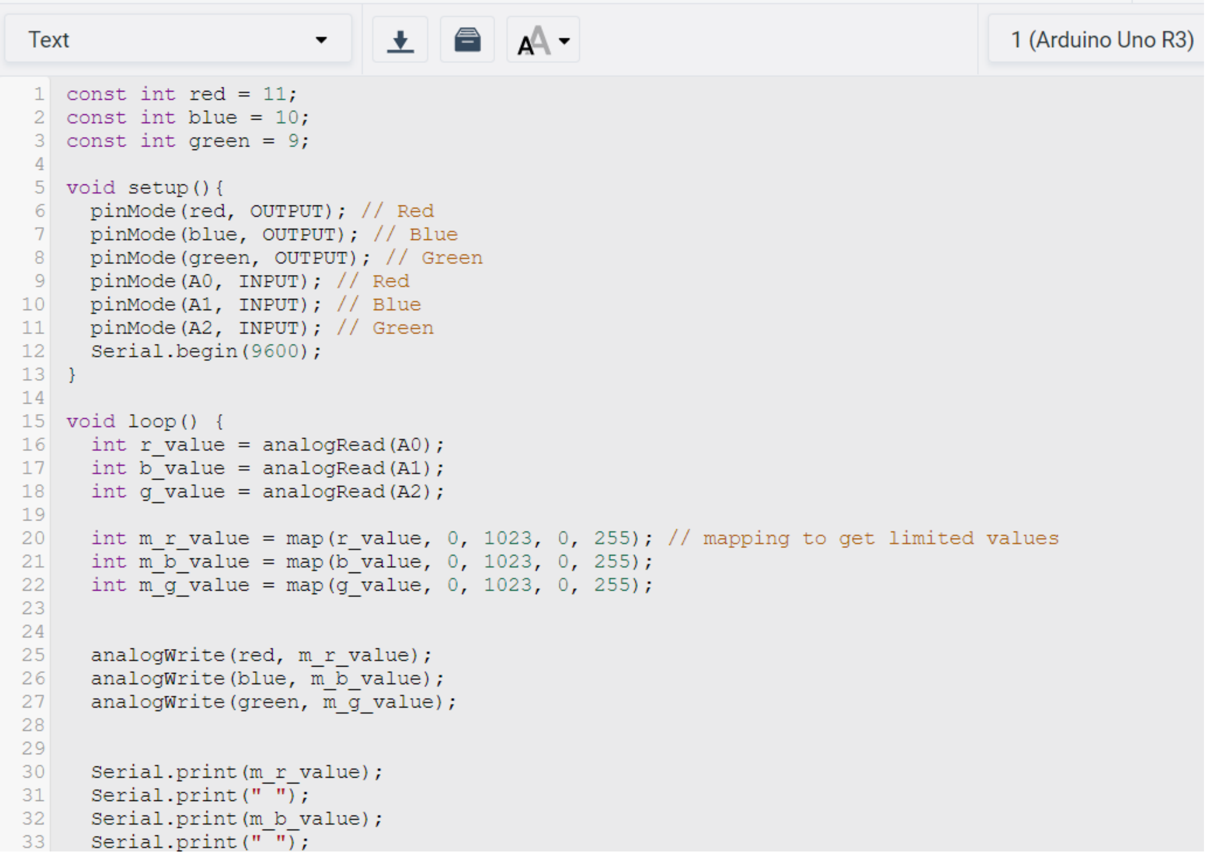
Methodology:

- 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.

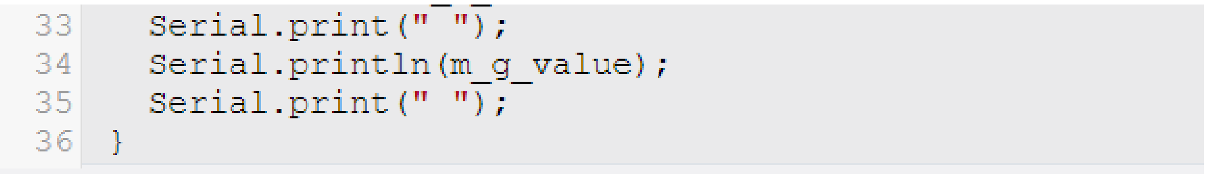
CODE:

```
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(" ");

}
```

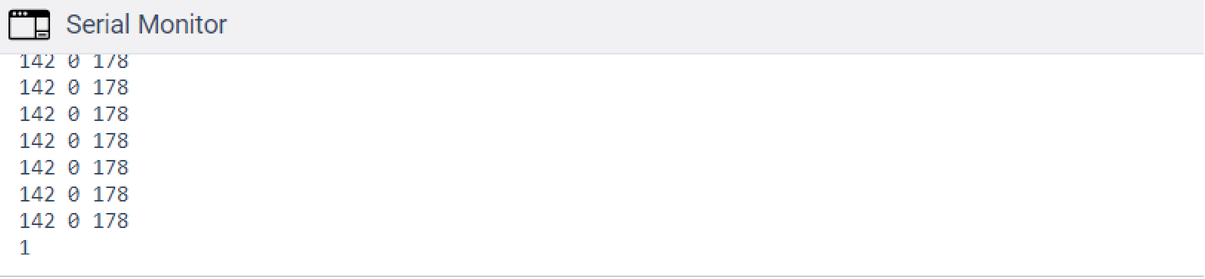
SCREENSHOTS:



```
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(" ");
```

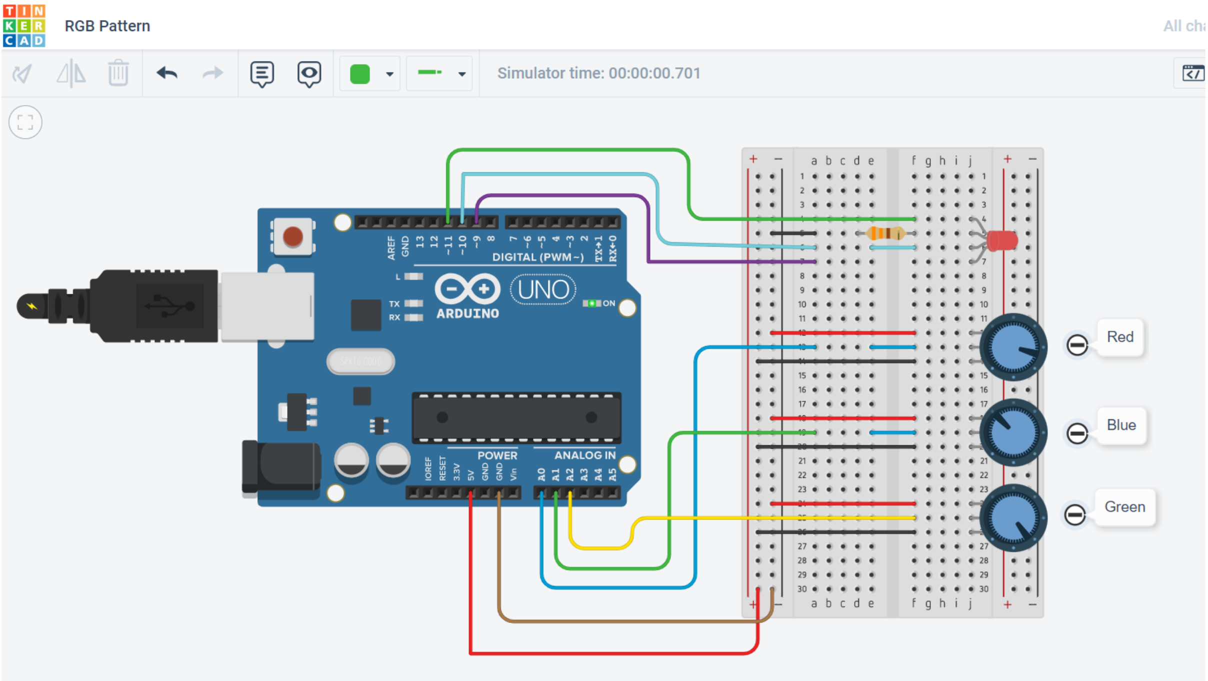


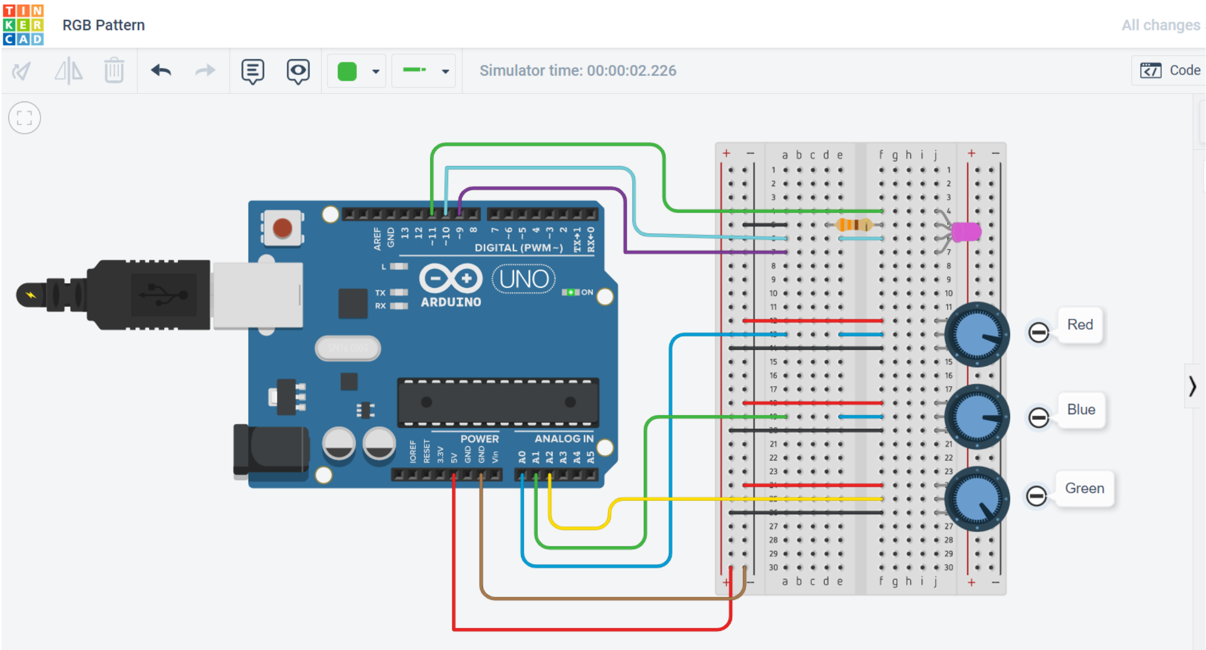
```
  Serial.print(" ");
  Serial.println(m_g_value);
  Serial.print(" ");
}
```

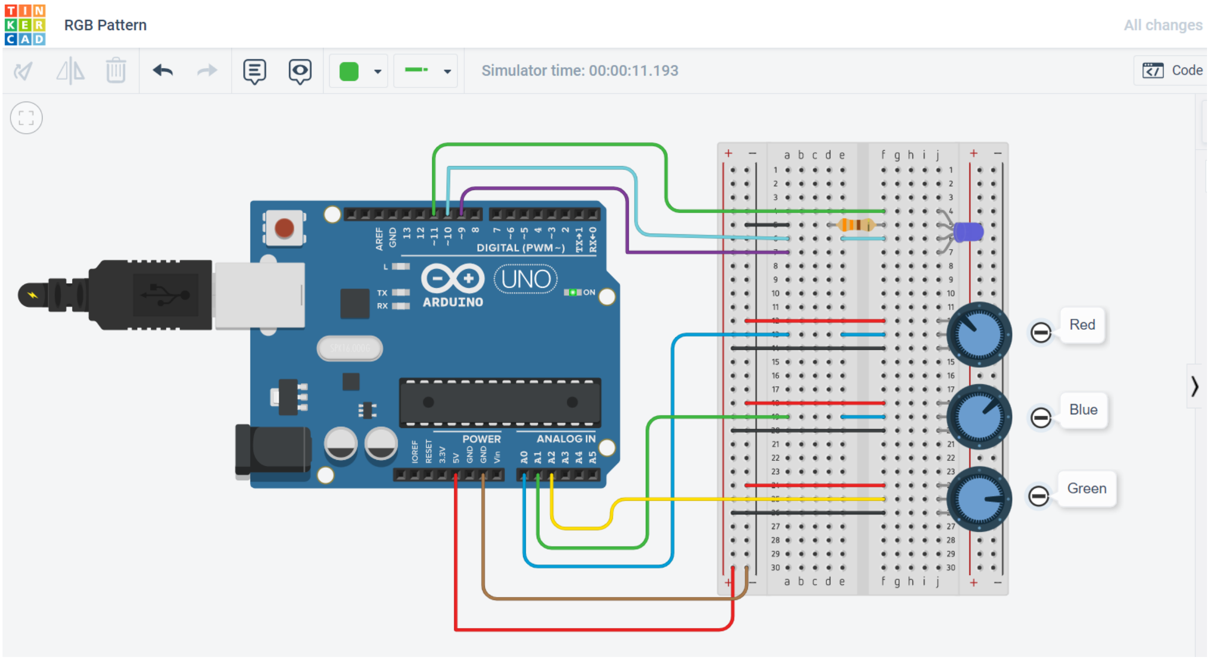


```
142 0 178
142 0 178
142 0 178
142 0 178
142 0 178
142 0 178
142 0 178
1
```

LED COLOR CHANGING FOR DIFFERENT INPUTS:







Conclusion:

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