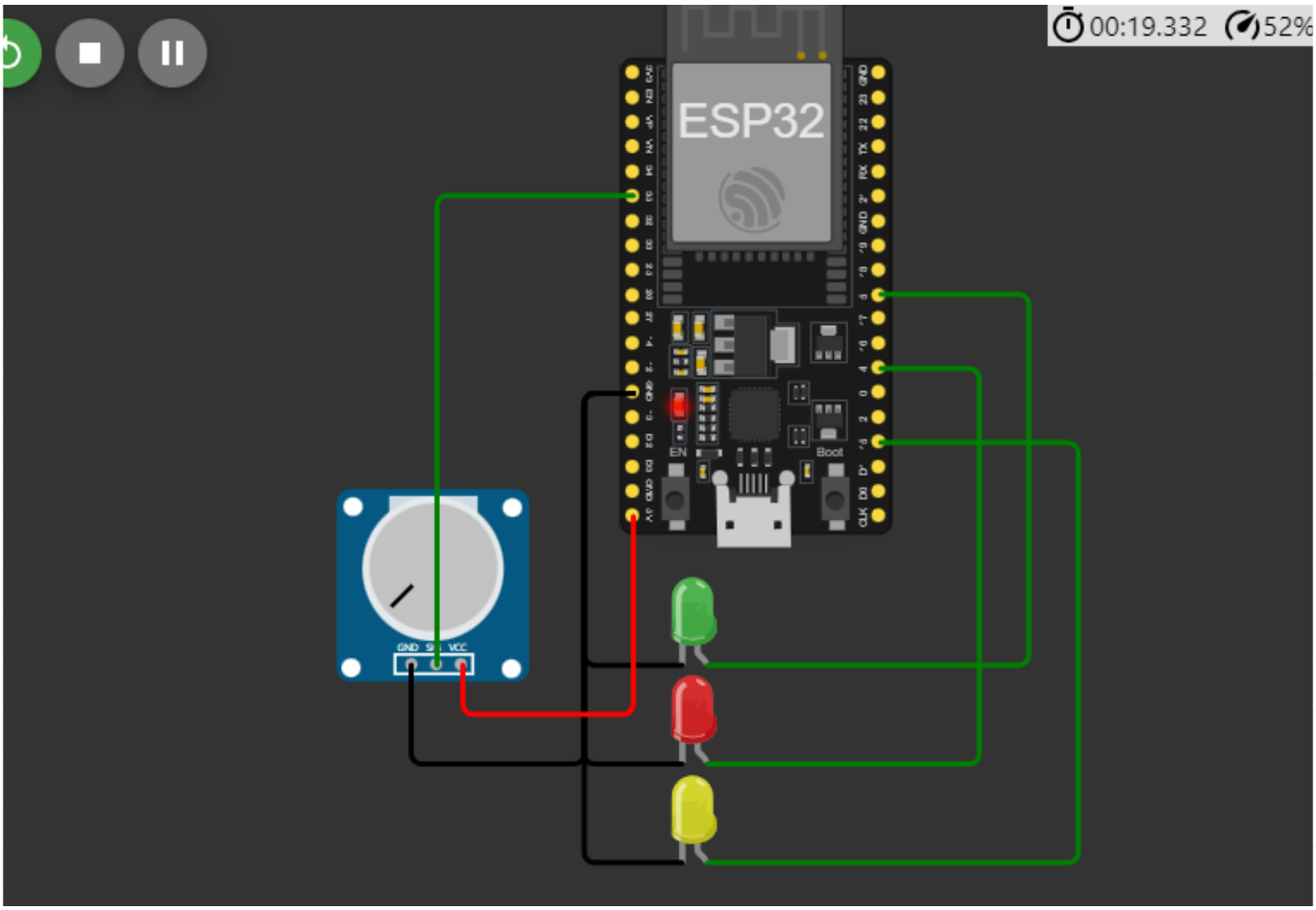
Microprocessor

HardWare project

Control LEDs Brightness

الرقم الجامعي	الاسم
21010881	عمر محمد أحمد شويخ محفوظ
21010947	فرح السيد السيد مصطفى
21011022	مارك زكريا سليمان سعيد
21010339	ايه احمد سليمان

Circuit diagram:



Code:

```
#include <BluetoothSerial.h>

// LED pins
const int ledPin1 = 5;
const int ledPin2 = 4;
const int ledPin3 = 15;

BluetoothSerial SerialBT;

void setup() {
  // Initialize serial communication
  Serial.begin(9600);

  // Set the LED pins as outputs
  pinMode(ledPin1, OUTPUT);
  pinMode(ledPin2, OUTPUT);
  pinMode(ledPin3, OUTPUT);

  // Initialize Bluetooth serial communication
  SerialBT.begin("LED_Control");

  // Wait until Bluetooth is connected
  while (!SerialBT.connected()) {
    delay(100);
  }
}

void loop() {
  // Check if data is available from Bluetooth
  if (SerialBT.available()) {
    // Read the incoming brightness value
    int brightness = SerialBT.parseInt();

    // Control the LEDs based on brightness value
    if (brightness <= 250) {
      ledcWrite(ledPin1, brightness);
      analogWrite(ledPin2, 0);
      analogWrite(ledPin3, 0);
    } else if (brightness <= 500) {
      analogWrite(ledPin1, 250);
      analogWrite(ledPin2, brightness - 250);
      analogWrite(ledPin3, 0);
    } else {
      analogWrite(ledPin1, 250);
      analogWrite(ledPin2, 250);
      analogWrite(ledPin3, brightness - 500);
    }

    // Print the brightness value to the serial monitor
    Serial.println(brightness);
  }

  // Delay for stability
  delay(10);
}
```

we include the BluetoothSerial library and define the LED pins (ledPin1, ledPin2, and ledPin3) that correspond to the three LEDs we want to control. We also initialize the serial communication with a baud rate of 9600 and configure the LED pins as outputs.

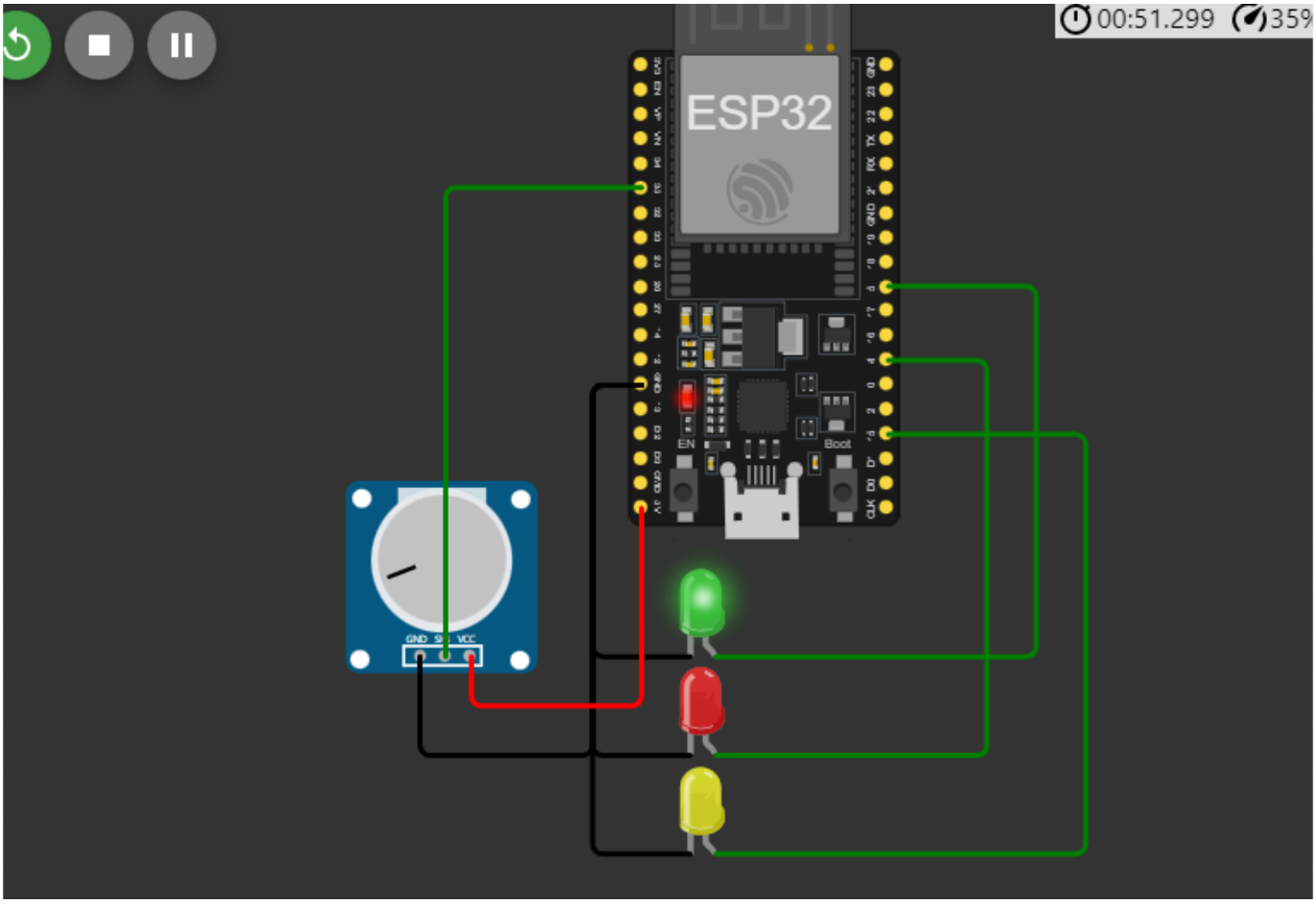
Inside the loop() function, we continuously check if there is data available from Bluetooth using SerialBT.available(). If there is data available, we proceed to read the incoming brightness value using SerialBT.parseInt().

After controlling the LEDs, we print the brightness value to the serial monitor using Serial.println(brightness).

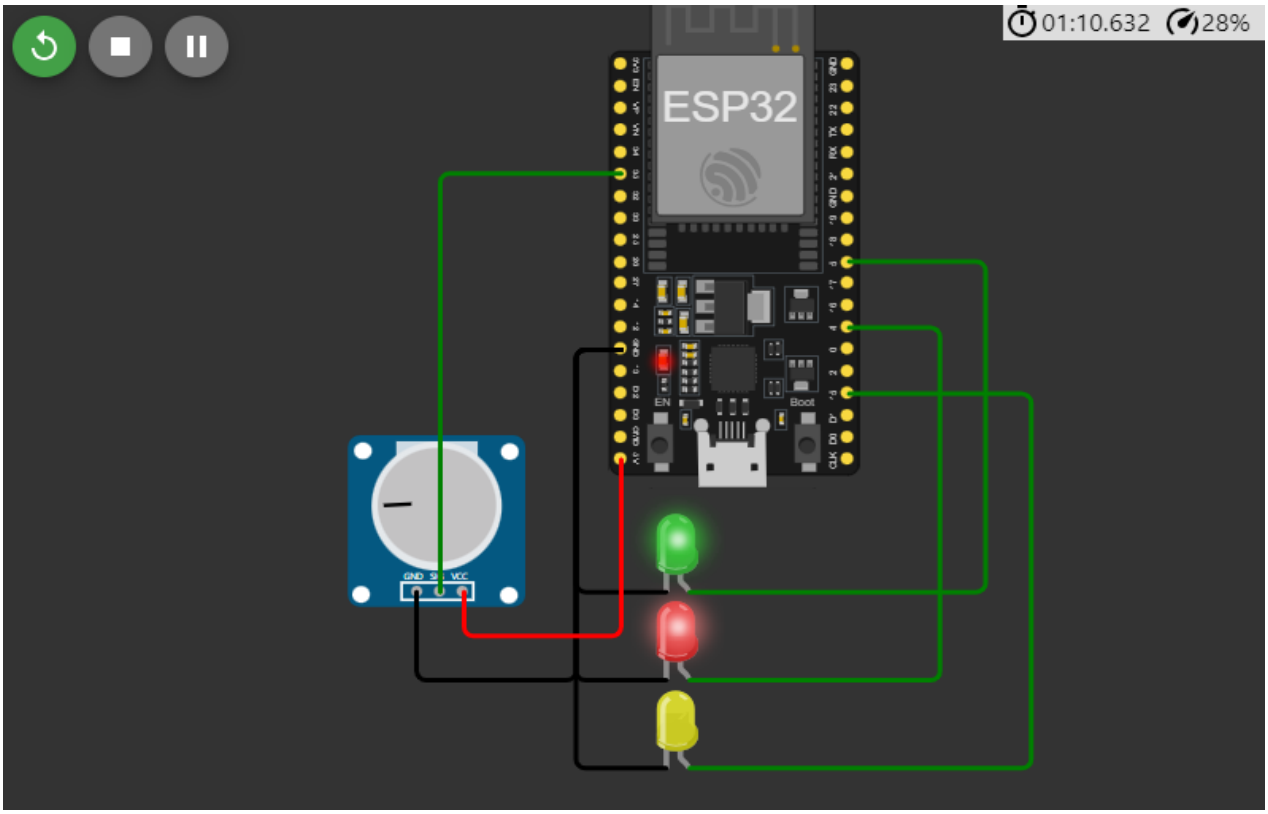
Finally, we add a small delay of 10 milliseconds using delay(10) to ensure stability and avoid overwhelming the system with rapid updates.

Based on the received brightness value, we control the LEDs gradually:

1) If brightness is less than or equal to 250, only the first LED (ledPin1) will be lit with the corresponding brightness value. The second and third LEDs are turned off.



2) If brightness is between 251 and 500, the first LED (ledPin1) is set to its maximum brightness (250), and the second LED (ledPin2) will light up with the remaining brightness value (brightness - 250).



3) If brightness is higher than 500, the first LED (ledPin1) is at maximum brightness (250) and the second LED (ledPin2) is set to a constant brightness (250), and the third LED (ledPin3) represents the remaining brightness (brightness - 500).