Video Link:
https://drive.google.com/file/d/1VOJpux7i1j6ztUbygJpxWNiS1TWfbwth/view?usp=sharing

Part1:

1. Wiring

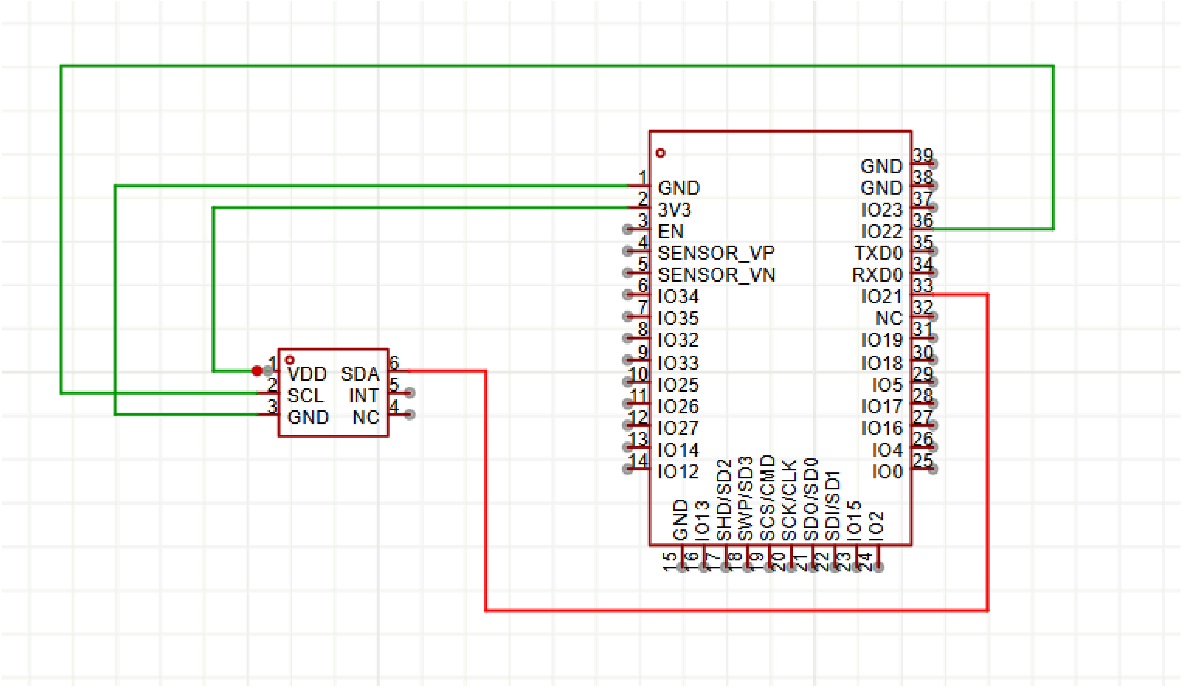
   #RGB sensor

TCS34725	ESP32
3v3	3v3
SCL	P22
SDA	P21
GND	GND

   #On board LED

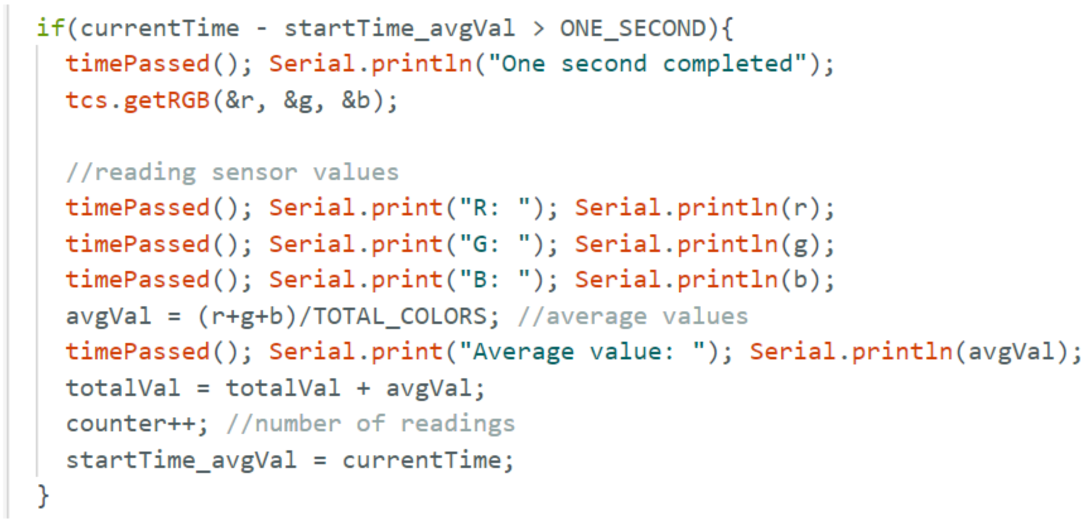
   ESP32 - PIN2

2. Schematic



3. Code

I have used Adafruit_TCS34725.h library to get RGB values from RGB sensor.



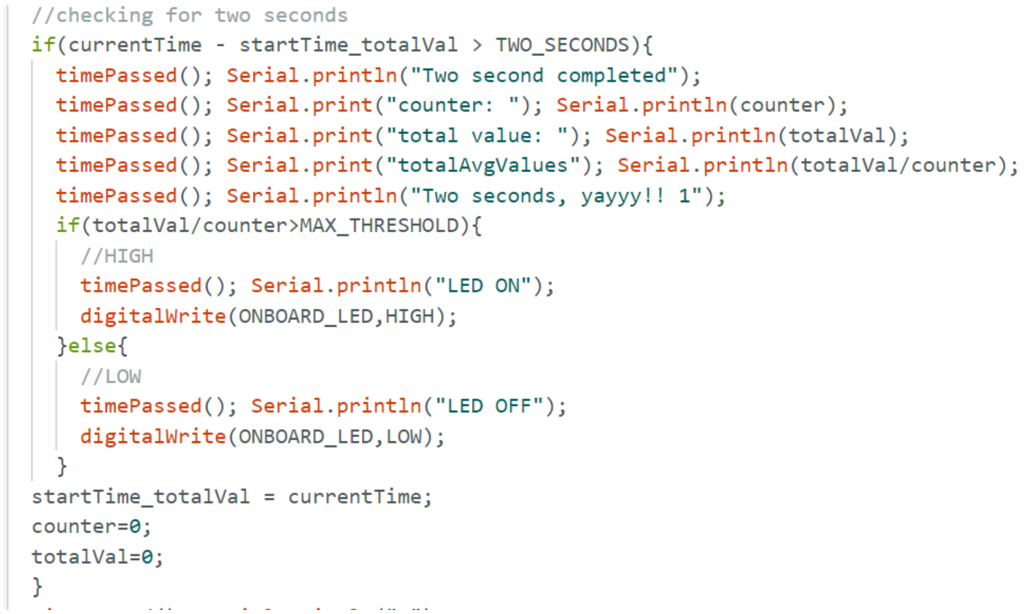
```
if(currentTime - startTime_avgVal > ONE_SECOND){
  timePassed(); Serial.println("One second completed");
  tcs.getRGB(&r, &g, &b);

  //reading sensor values
  timePassed(); Serial.print("R: "); Serial.println(r);
  timePassed(); Serial.print("G: "); Serial.println(g);
  timePassed(); Serial.print("B: "); Serial.println(b);
  avgVal = (r+g+b)/TOTAL_COLORS; //average values
  timePassed(); Serial.print("Average value: "); Serial.println(avgVal);
  totalVal = totalVal + avgVal;
  counter++; //number of readings
  startTime_avgVal = currentTime;
}
```

In the void loop, we read the sensor value every 1 second using the check mentioned in the screenshot above.

Each time (every second), the program reads the RGB values and increments both totalValue and counter.

The totalValue variable is used to calculate the total RGB sensor values over a 2-second period, while the counter variable keeps track of how many times the values have been read.



```
//checking for two seconds
if(currentTime - startTime_totalVal > TWO_SECONDS){
  timePassed(); Serial.println("Two second completed");
  timePassed(); Serial.print("counter: "); Serial.println(counter);
  timePassed(); Serial.print("total value: "); Serial.println(totalVal);
  timePassed(); Serial.print("totalAvgValues"); Serial.println(totalVal/counter);
  timePassed(); Serial.println("Two seconds, yayyy!! 1");
  if(totalVal/counter>MAX_THRESHOLD){
    //HIGH
    timePassed(); Serial.println("LED ON");
    digitalWrite(ONBOARD_LED,HIGH);
  }else{
    //LOW
    timePassed(); Serial.println("LED OFF");
    digitalWrite(ONBOARD_LED,LOW);
  }
startTime_totalVal = currentTime;
counter=0;
totalVal=0;
}
```

The code runs every two seconds and checks the totalValue and counter variables to calculate the average sensor value. It then compares this average with the maximum threshold. If the average value is greater than the threshold, the onboard LED is turned on; otherwise, the onboard LED is turned off.

Part2:

1. Wiring

#RGB sensor

TCS34725 - ESP32

3v3 - 3v3

SCL - P22

SDA - P21

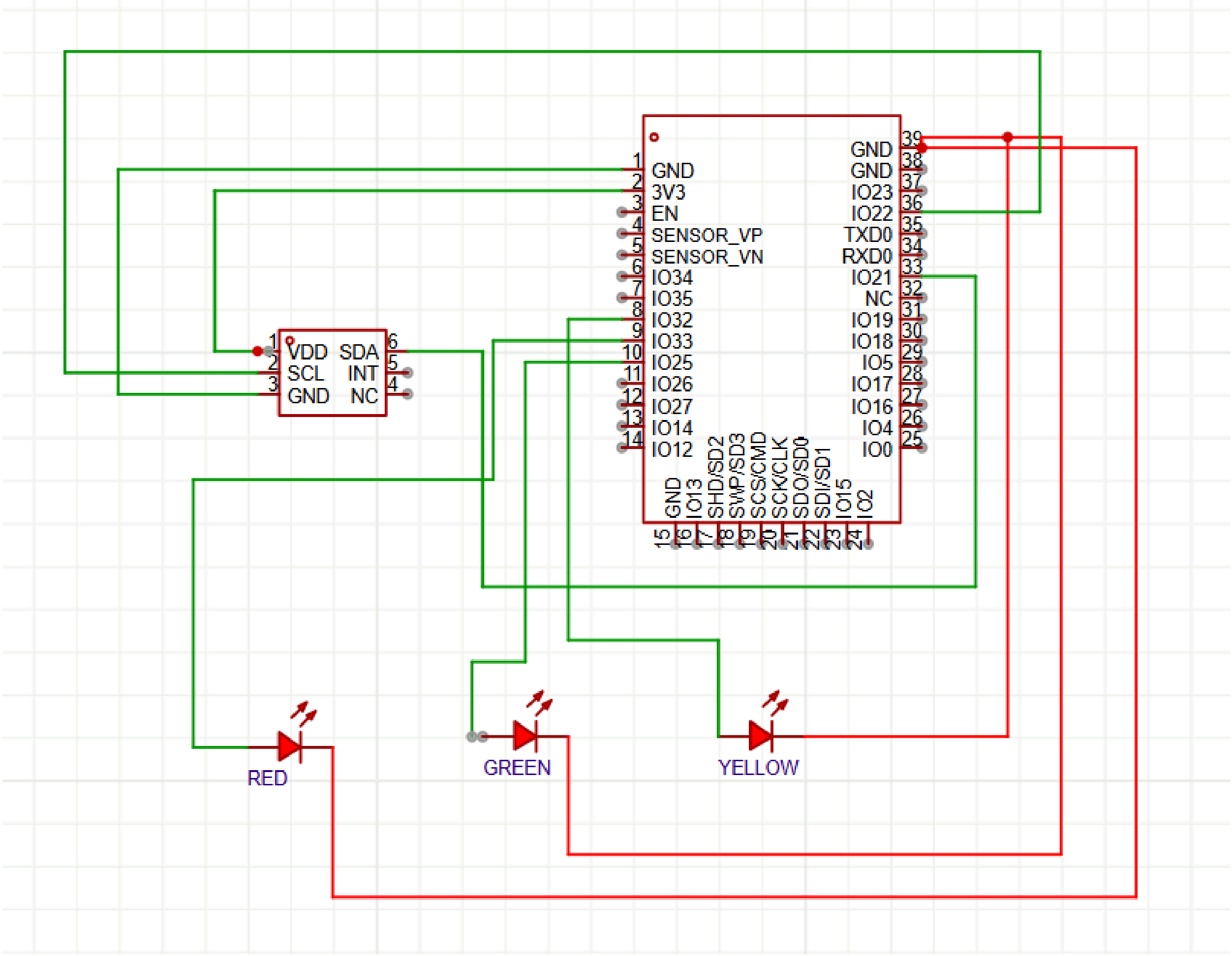
GND - GND

GREEN LED - 25

RED LED - 33

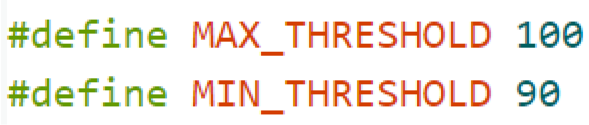
YELLOW LED - 32

2. Schematic

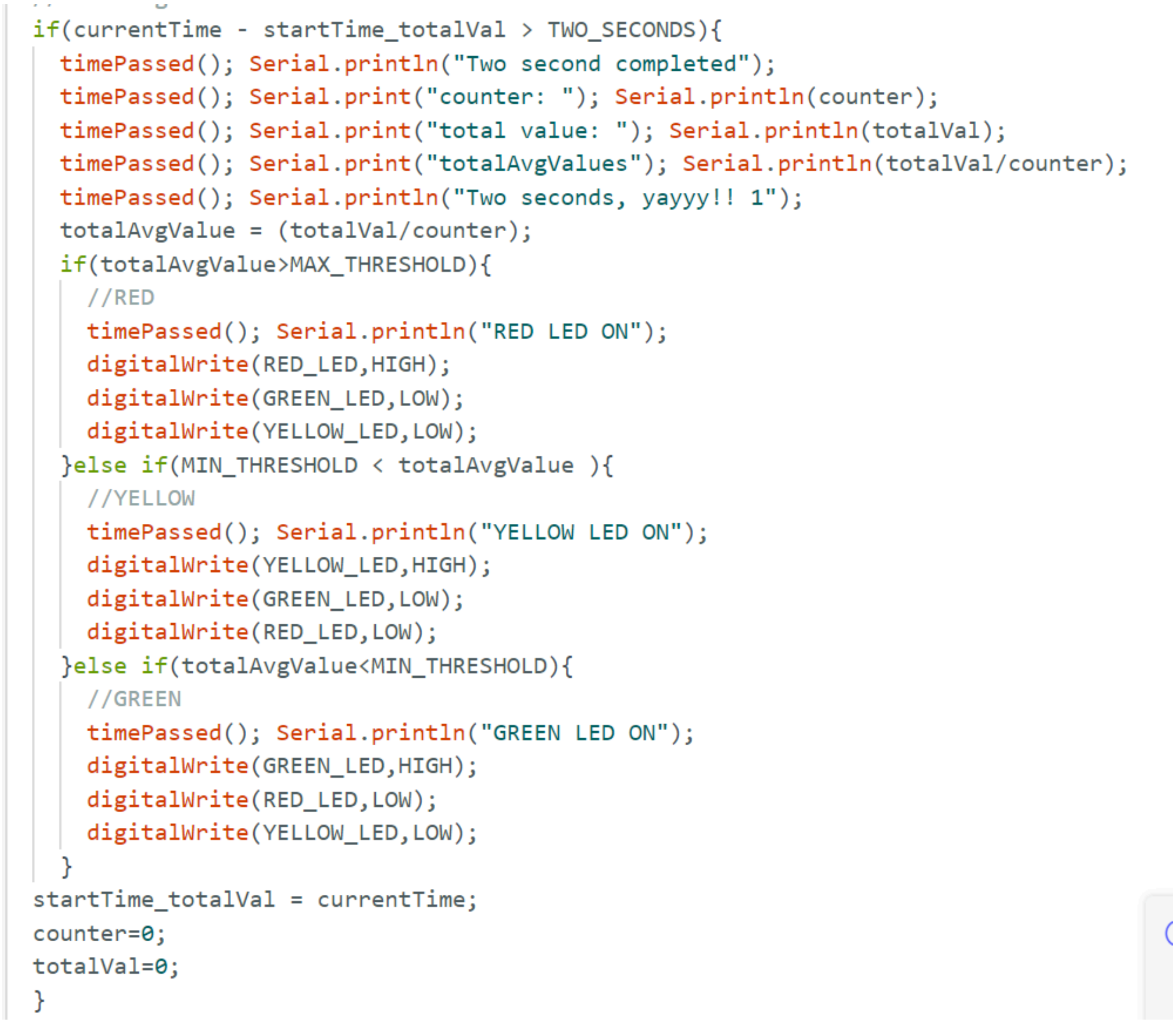


3. Code

Defined



```
#define MAX_THRESHOLD 100
#define MIN_THRESHOLD 90
```



```
if(currentTime - startTime_totalVal > TWO_SECONDS){
  timePassed(); Serial.println("Two second completed");
  timePassed(); Serial.print("counter: "); Serial.println(counter);
  timePassed(); Serial.print("total value: "); Serial.println(totalVal);
  timePassed(); Serial.print("totalAvgValues"); Serial.println(totalVal/counter);
  timePassed(); Serial.println("Two seconds, yayyy!! 1");
  totalAvgValue = (totalVal/counter);
  if(totalAvgValue>MAX_THRESHOLD){
    //RED
    timePassed(); Serial.println("RED LED ON");
    digitalWrite(RED_LED,HIGH);
    digitalWrite(GREEN_LED,LOW);
    digitalWrite(YELLOW_LED,LOW);
  }else if(MIN_THRESHOLD < totalAvgValue ){
    //YELLOW
    timePassed(); Serial.println("YELLOW LED ON");
    digitalWrite(YELLOW_LED,HIGH);
    digitalWrite(GREEN_LED,LOW);
    digitalWrite(RED_LED,LOW);
  }else if(totalAvgValue<MIN_THRESHOLD){
    //GREEN
    timePassed(); Serial.println("GREEN LED ON");
    digitalWrite(GREEN_LED,HIGH);
    digitalWrite(RED_LED,LOW);
    digitalWrite(YELLOW_LED,LOW);
  }
startTime_totalVal = currentTime;
counter=0;
totalVal=0;
}
```

Here we are checking values and turning LED ON
If, total average value is greater than max threshold (100) - RED ON
If, total average value is between min threshold(90) and max threshold(100) - YELLOW
If, total average value is less than min threshold than - GREEN

Part3:

4. Wiring

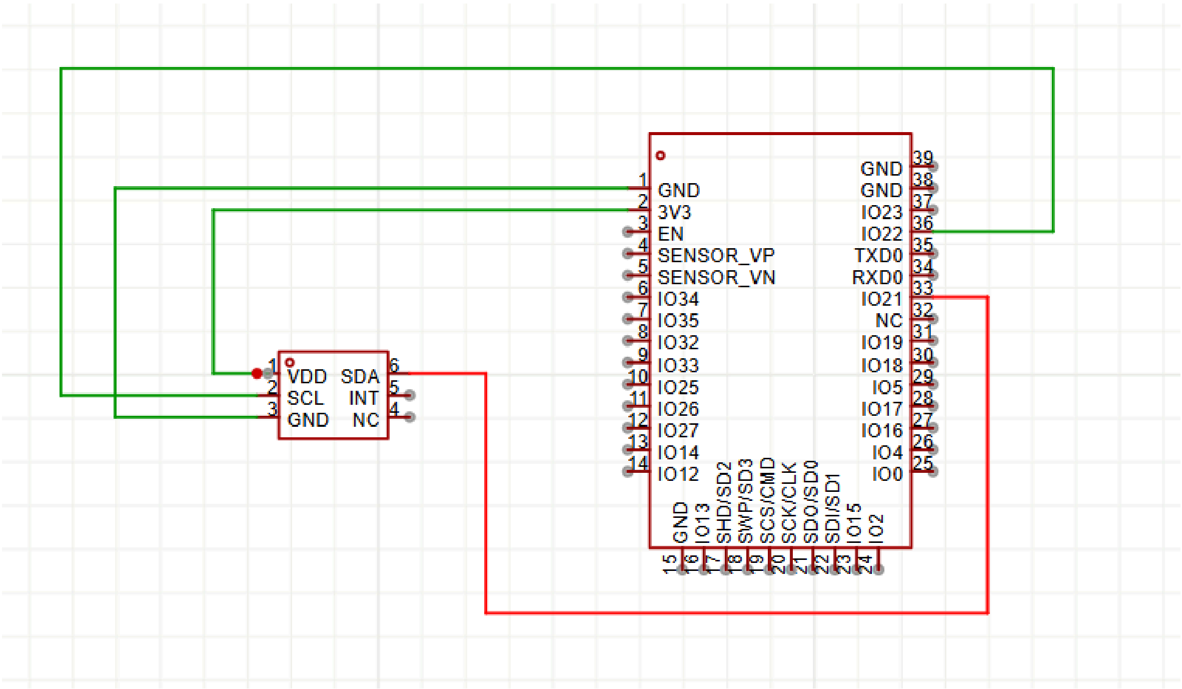
#RGB sensor

TCS34725	ESP32
3v3	3v3
SCL	P22
SDA	P21
GND	GND

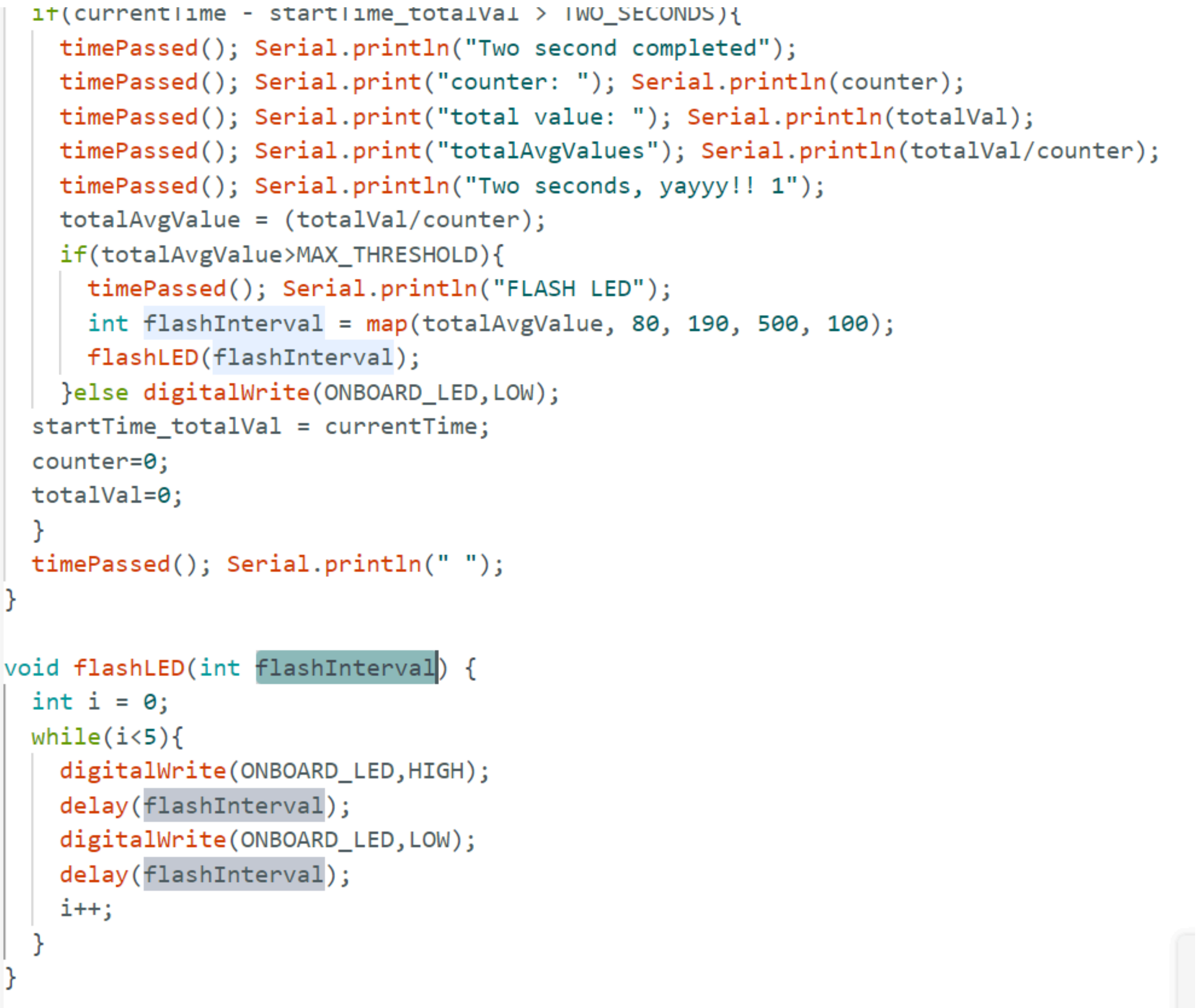
#On board LED

ESP32 - PIN2

5. Schematic



3. Code



```
  if(currentTime - startTime_totalVal > TWO_SECONDS){
    timePassed(); Serial.println("Two second completed");
    timePassed(); Serial.print("counter: "); Serial.println(counter);
    timePassed(); Serial.print("total value: "); Serial.println(totalVal);
    timePassed(); Serial.print("totalAvgValues"); Serial.println(totalVal/counter);
    timePassed(); Serial.println("Two seconds, yayyy!! 1");
    totalAvgValue = (totalVal/counter);
    if(totalAvgValue>MAX_THRESHOLD){
      timePassed(); Serial.println("FLASH LED");
      int flashInterval = map(totalAvgValue, 80, 190, 500, 100);
      flashLED(flashInterval);
    }else digitalWrite(ONBOARD_LED,LOW);
  startTime_totalVal = currentTime;
  counter=0;
  totalVal=0;
  }
  timePassed(); Serial.println(" ");
}

void flashLED(int flashInterval) {
  int i = 0;
  while(i<5){
    digitalWrite(ONBOARD_LED,HIGH);
    delay(flashInterval);
    digitalWrite(ONBOARD_LED,LOW);
    delay(flashInterval);
    i++;
  }
}
```

The code is similar to Part 1, with the only change being that the flashInterval decreases as the average value increases — this is done using the map() function.

The flashLED() function flashes the onboard LED based on the calculated total average value: higher average values make the LED flash faster, while lower averages make it flash slower.