**TRAFFIC LIGHT CONTROL USING BLYNK SOFTWARE AND NODEMCU-ESP8266 MODULE**

1. **BLYNK:**

Blynk.Console is a web application that allows users to manage devices, users, and organizations on the Blynk IoT platform.

It also allows users to remotely monitor and control devices.

**Features:**

**Device configuration**: Configure how connected devices work on the platform, including application settings

**Remote control**: Remotely control and monitor devices in a web browser

**User and organization management**: Manage users, organizations, and locations

**Over-the-air updates**: Perform over-the-air (OTA) firmware updates

**Dashboard creation**: Create a dashboard with multiple tabs for interacting with devices via Widgets

**Data analytics**: Perform data analytics

**Alerts and automations**: Set up alerts and automations

**USERS:**  Businesses, Developers, and IoT developers.

**Working**

* Blynk is a low-code IoT platform that allows users to build and customize apps without writing code
* Blynk's apps allow users to manage devices, users, and data remotely.

**CONFIGURATION:**

BLYNK\_TEMPLATE\_ID "TMPL3-B6Qx-Qf"

BLYNK\_TEMPLATE\_NAME "TLIGHTS"

BLYNK\_AUTH\_TOKEN "aFXtugni8NIPE3OwUJG1btrgb5wTQq6\_"

1. **ESP32**

The ESP32 is a low-cost, low-power microcontroller with built-in Wi-Fi and Bluetooth. It's a system-on-a-chip (SoC), which means it's an integrated circuit that contains an entire electronic system.

What it's used for Mobile devices, Wearable electronics, Internet of Things (IoT) applications, and Industrial environments.

**Features:**

**Wi-Fi**: Supports 802.11b/g/n standard at 2.4 GHz band, up to 150 Mbps

**Bluetooth**: Supports Bluetooth v4.2 BR/EDR and Bluetooth LE specifications

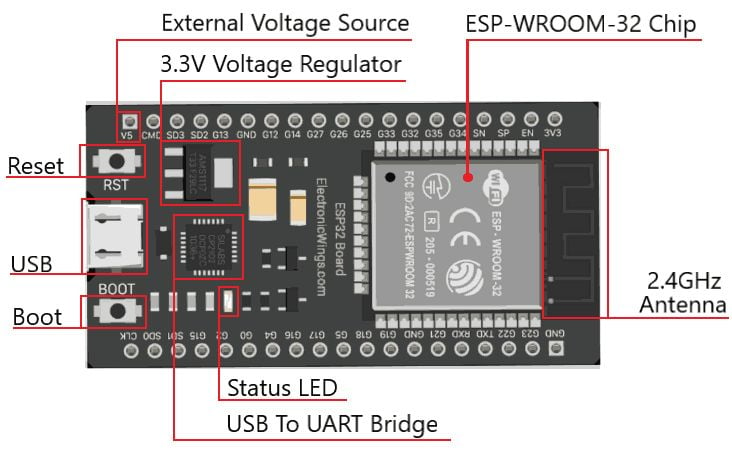
**Power consumption**: Ultra-low power consumption

**Operating temperature**: Can function reliably in industrial environments, with an operating temperature ranging from –40°C to +125°C

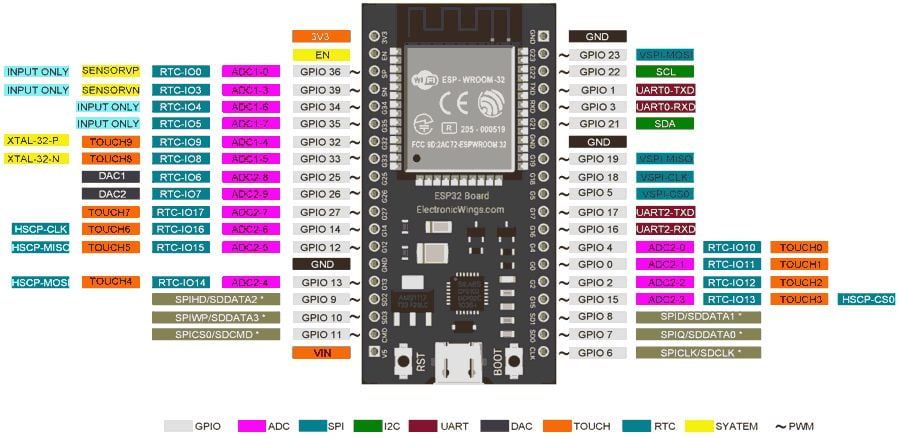
**Digital pins**: 34 digital pins that can be used to connect sensors, buttons, buzzers, and more

Espressif Systems, a Chinese company based in Shanghai

Manufactured by TSMC using their 40 nm process



**PIN DIAGRAM:**



1. **WOKWI**

* Wokwi is an online Electronics simulator.
* Used it to simulate Arduino, ESP32, STM32, and many other popular boards, parts and sensors.

**Features**[**​**](https://docs.wokwi.com/?utm_source=wokwi#unique-features)**:**

* [**WiFi simulation**](https://docs.wokwi.com/guides/esp32-wifi) - Connect your simulated project to the internet. You can use MQTT, HTTP, NTP, and many other network protocols.
* [**Virtual Logic Analyzer**](https://docs.wokwi.com/guides/logic-analyzer) - Capture digital signals in your simulation (e.g. UART, I2C, SPI) and analyze them on your computer.
* [**Advanced debugging with GDB**](https://docs.wokwi.com/gdb-debugging) - Powerful Arduino and Raspberry Pi Pico debugger for advanced users.
* [**SD card simulation**](https://docs.wokwi.com/parts/wokwi-microsd-card) - Store and retrieve files and directories from your code. [Paying users](https://wokwi.com/pricing?ref=docs_sdcard) can also upload binary files (such as images)
* [**Chips API**](https://docs.wokwi.com/chips-api/getting-started) - Create your own custom chips and parts, and share them with the community.
* [**Visual Studio Code integration**](https://docs.wokwi.com/vscode/getting-started) - Simulate your embedded projects directly from VS Code.

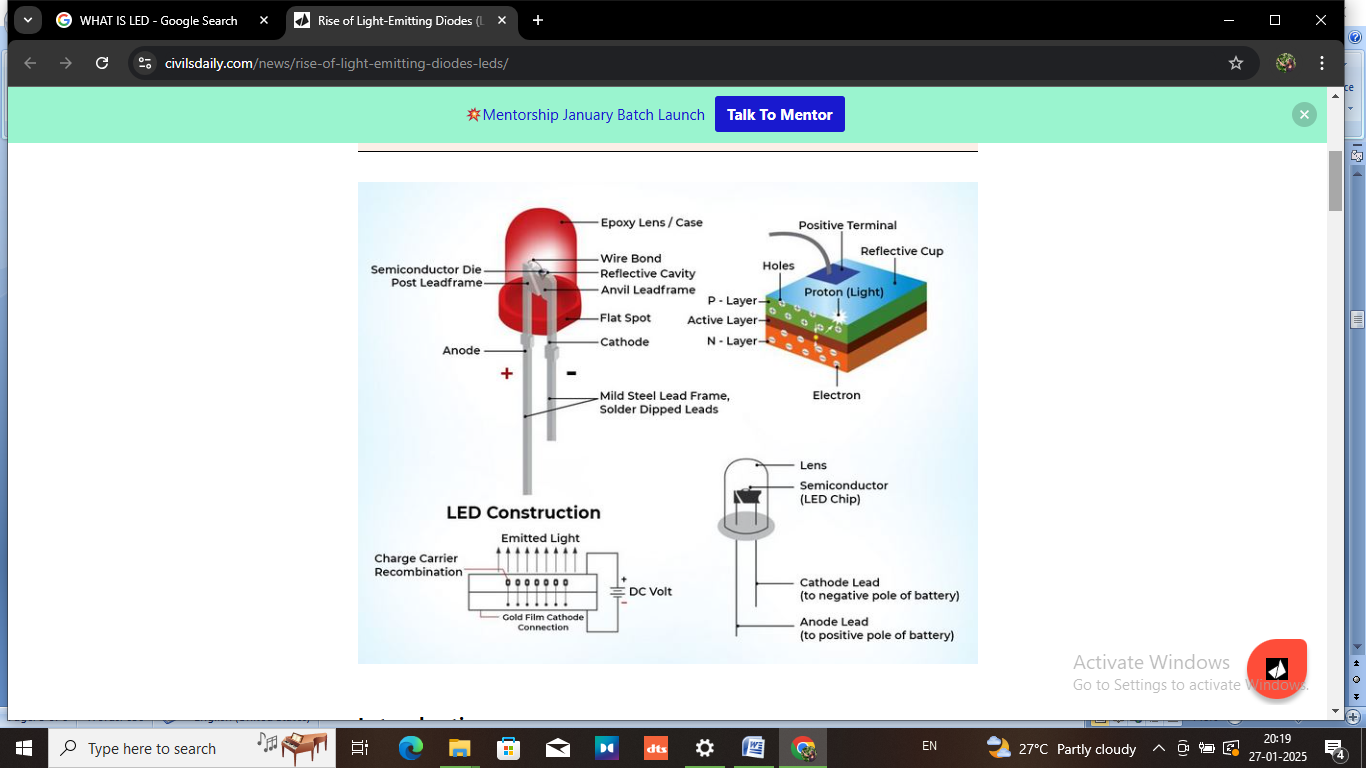
1. **LED**

LED stands for light-emitting diode. It's a semiconductor device that emits light when an electric current passes through it. LEDs are used in many devices, including mobile phones, traffic lights, and street lights.

**WORKING:**

* When an electric current passes through the LED, electrons and holes recombine at the P-N junction.
* This recombination releases energy in the form of photons.
* The color of the light depends on the energy required for electrons to cross the semiconductor's band gap.

Benefits of LEDs

****

1. **SKETCH CODE:**

#define BLYNK\_TEMPLATE\_ID "TMPL3-B6Qx-Qf"

#define BLYNK\_TEMPLATE\_NAME "TLIGHTS"

#define BLYNK\_AUTH\_TOKEN "aFXtugni8NIPE3OwUJG1btrgb5wTQq6\_"

#include <WiFi.h>

#include <BlynkSimpleEsp32.h>

const char\* ssid = "Wokwi-GUEST";

const char\* password = "";

#define RELAY1\_PIN  25

#define RELAY2\_PIN  26

#define RELAY3\_PIN  27

#define SWITCH1\_VPIN V0

#define SWITCH2\_VPIN V1

#define SWITCH3\_VPIN V2

BLYNK\_WRITE(SWITCH1\_VPIN) {

  int buttonState = param.asInt();

  digitalWrite(RELAY1\_PIN, buttonState ? HIGH : LOW);

}

BLYNK\_WRITE(SWITCH2\_VPIN) {

  int buttonState = param.asInt();  digitalWrite(RELAY2\_PIN, buttonState ? HIGH : LOW);

}

BLYNK\_WRITE(SWITCH3\_VPIN) {

  int buttonState = param.asInt();

  digitalWrite(RELAY3\_PIN, buttonState ? HIGH : LOW);

}

void setup() {

**Serial**.begin(115200);

  pinMode(RELAY1\_PIN, OUTPUT);

  pinMode(RELAY2\_PIN, OUTPUT);

  pinMode(RELAY3\_PIN, OUTPUT);

  digitalWrite(RELAY1\_PIN, LOW);

  digitalWrite(RELAY2\_PIN, LOW);

  digitalWrite(RELAY3\_PIN, LOW);

**Serial**.print("Connecting to WiFi");

  WiFi.begin(ssid, password);

  while (WiFi.status() != WL\_CONNECTED) {

    delay(500);

**Serial**.print(".");

  }

**Serial**.println("\nConnected to WiFi");

**Serial**.println("Connecting to Blynk...");

  Blynk.begin(BLYNK\_AUTH\_TOKEN, ssid, password);

**Serial**.println("Connected to Blynk");

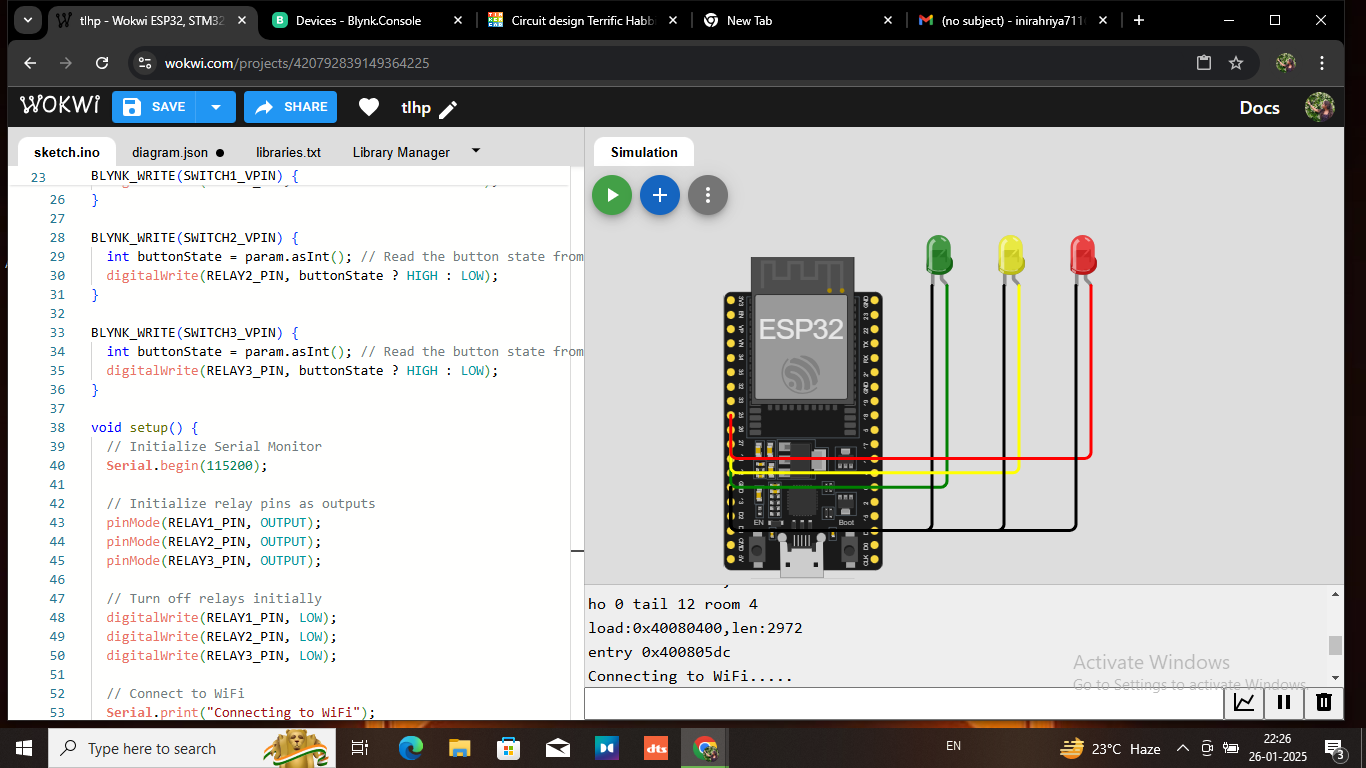
}

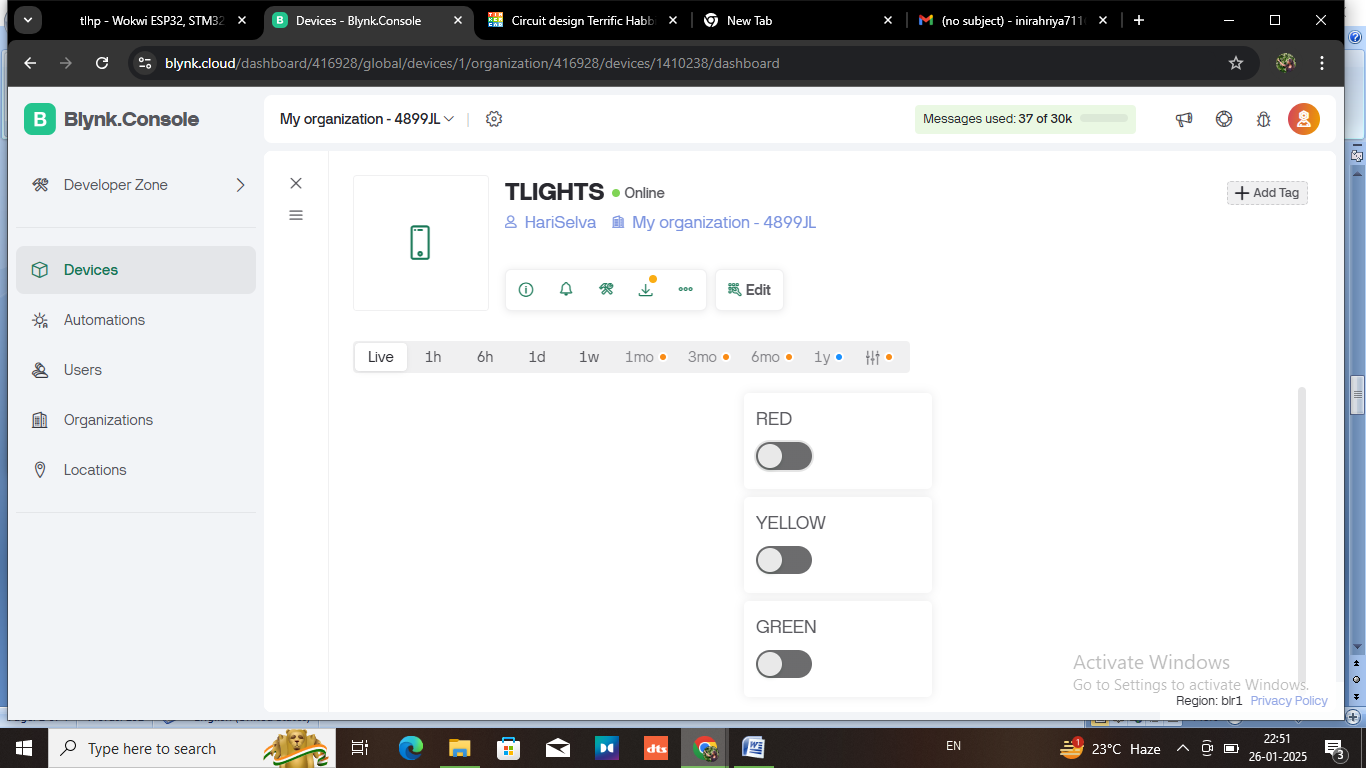
void loop() {

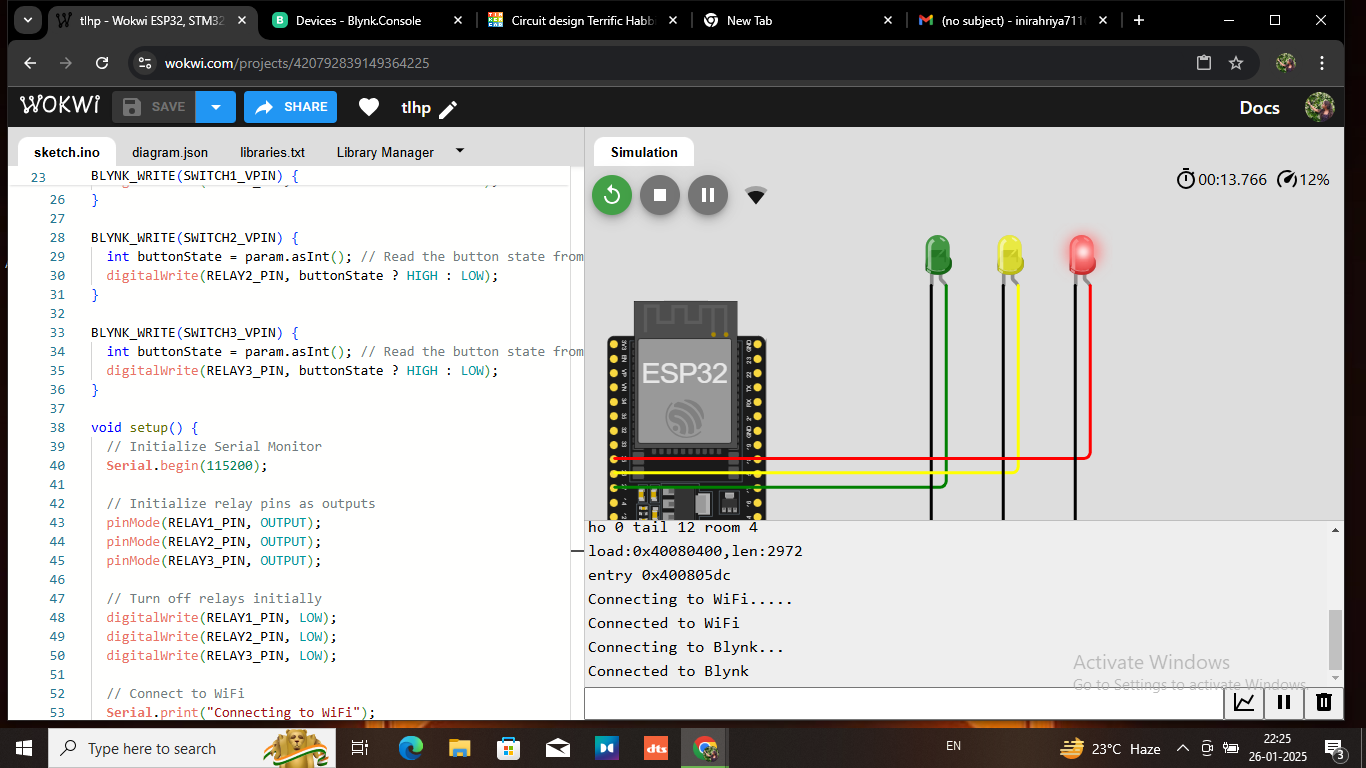
  Blynk.run();

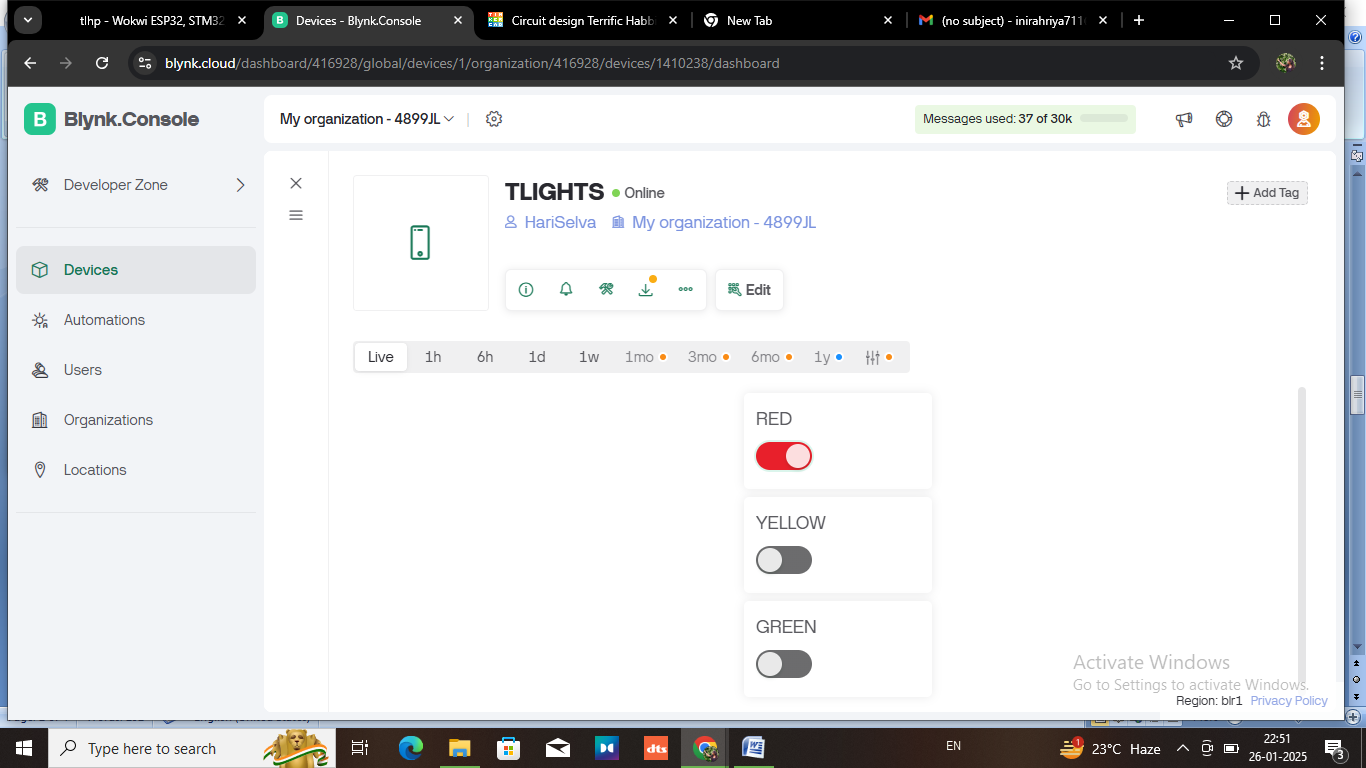
}

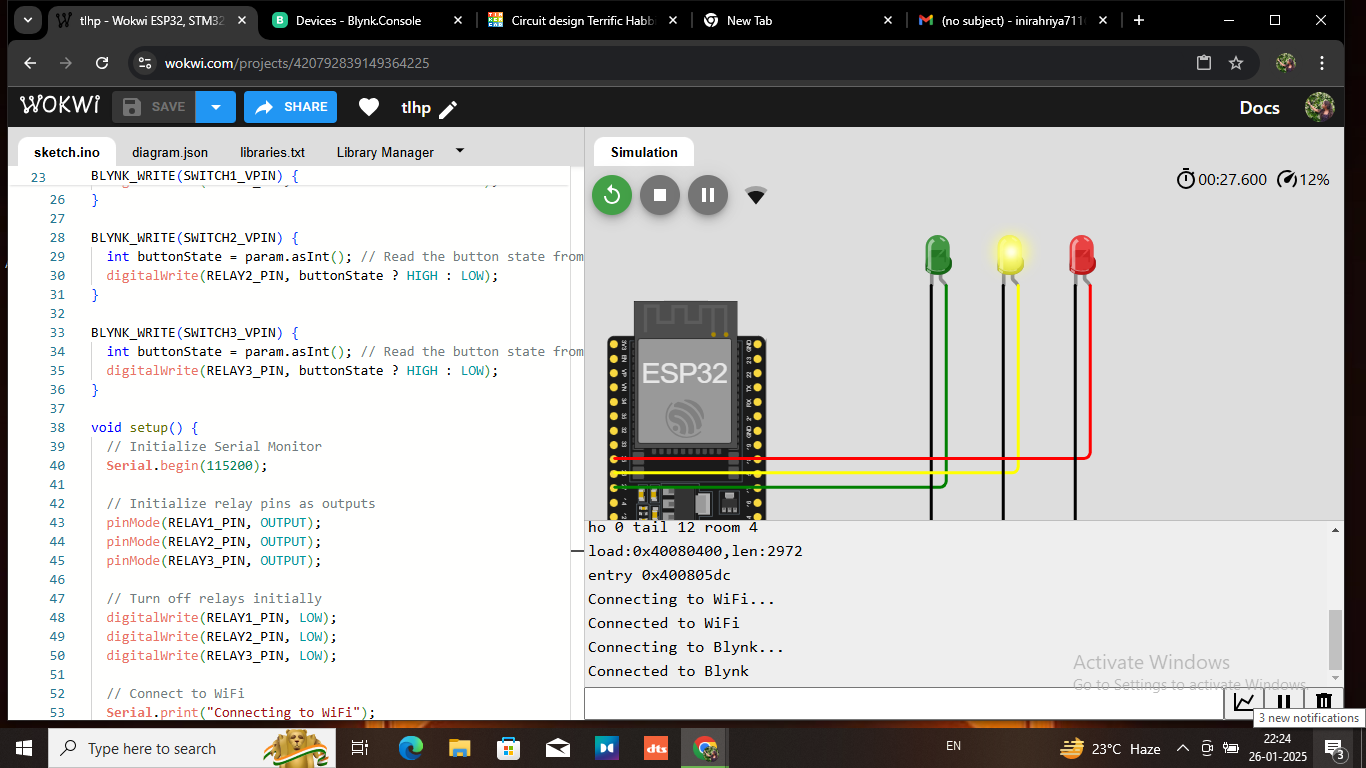
**OUTPUT:**

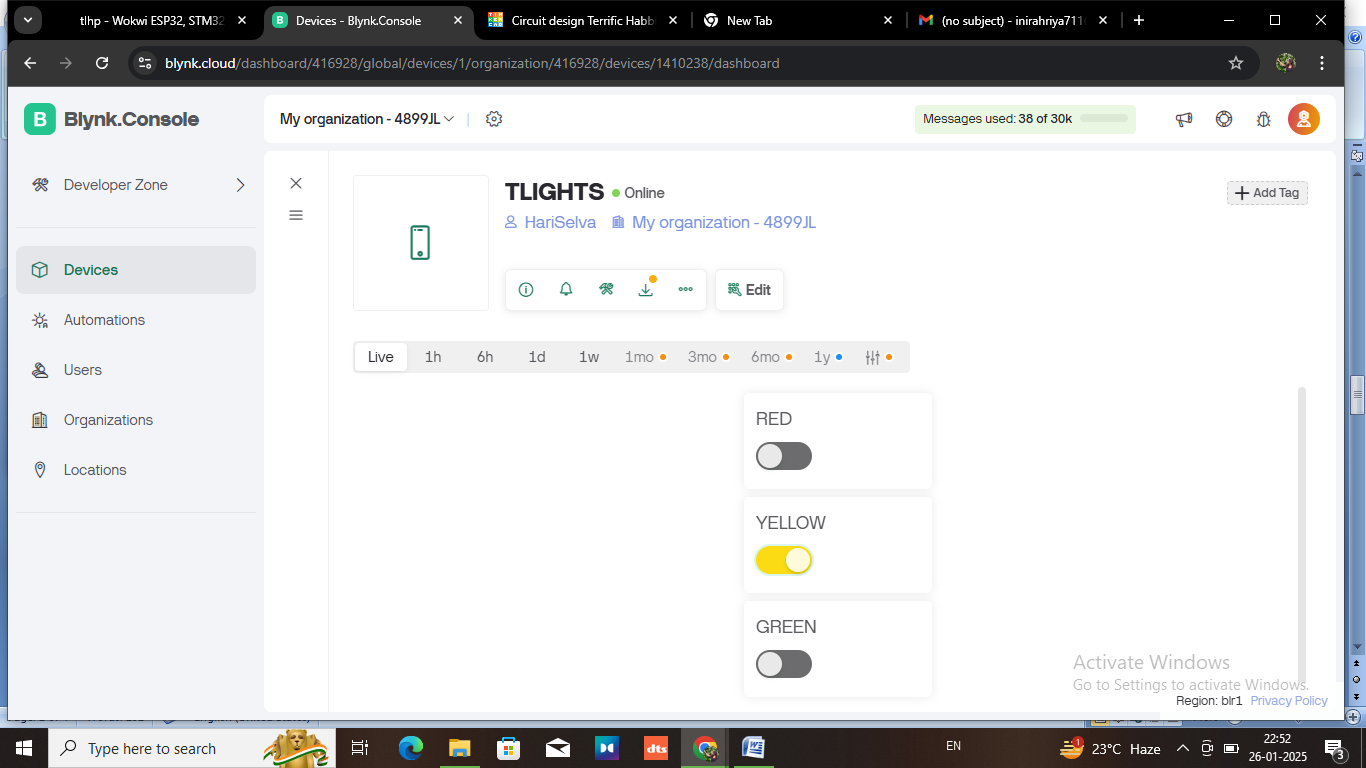
****

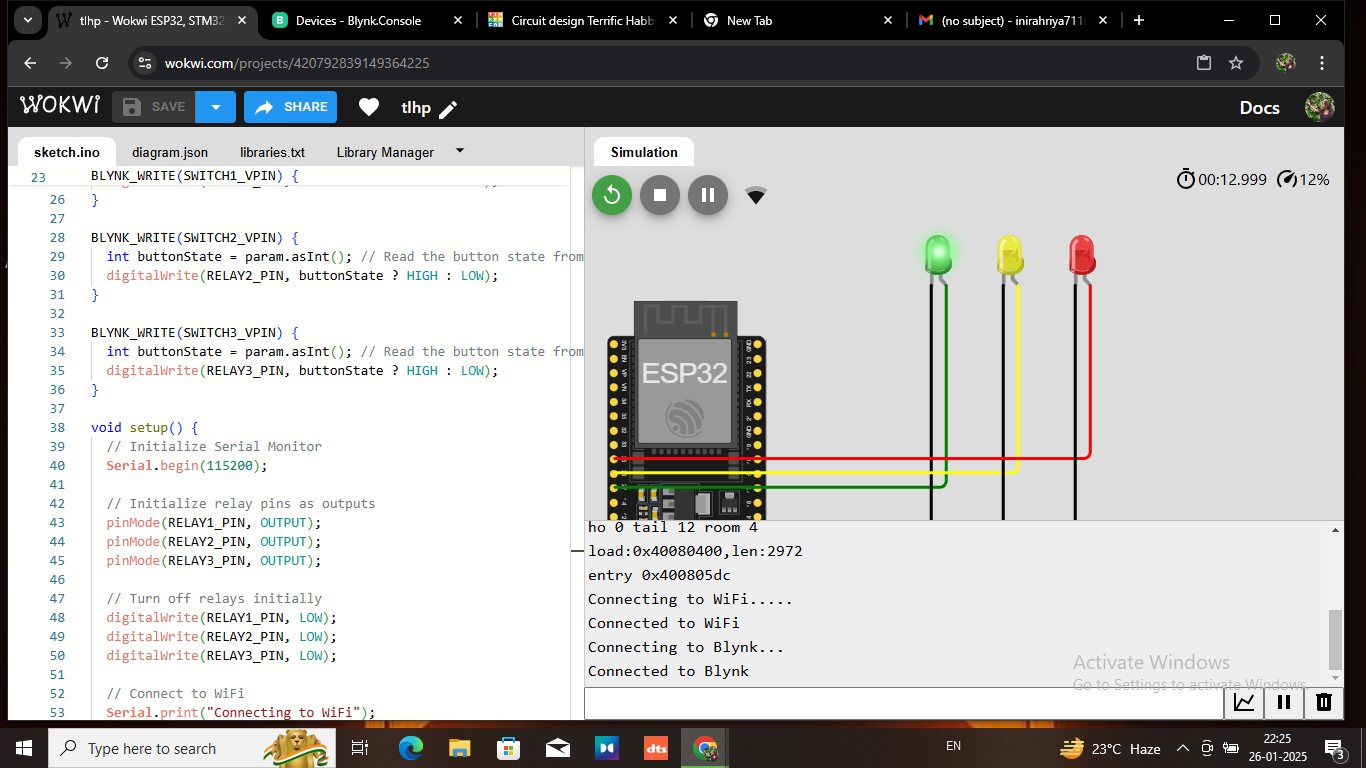


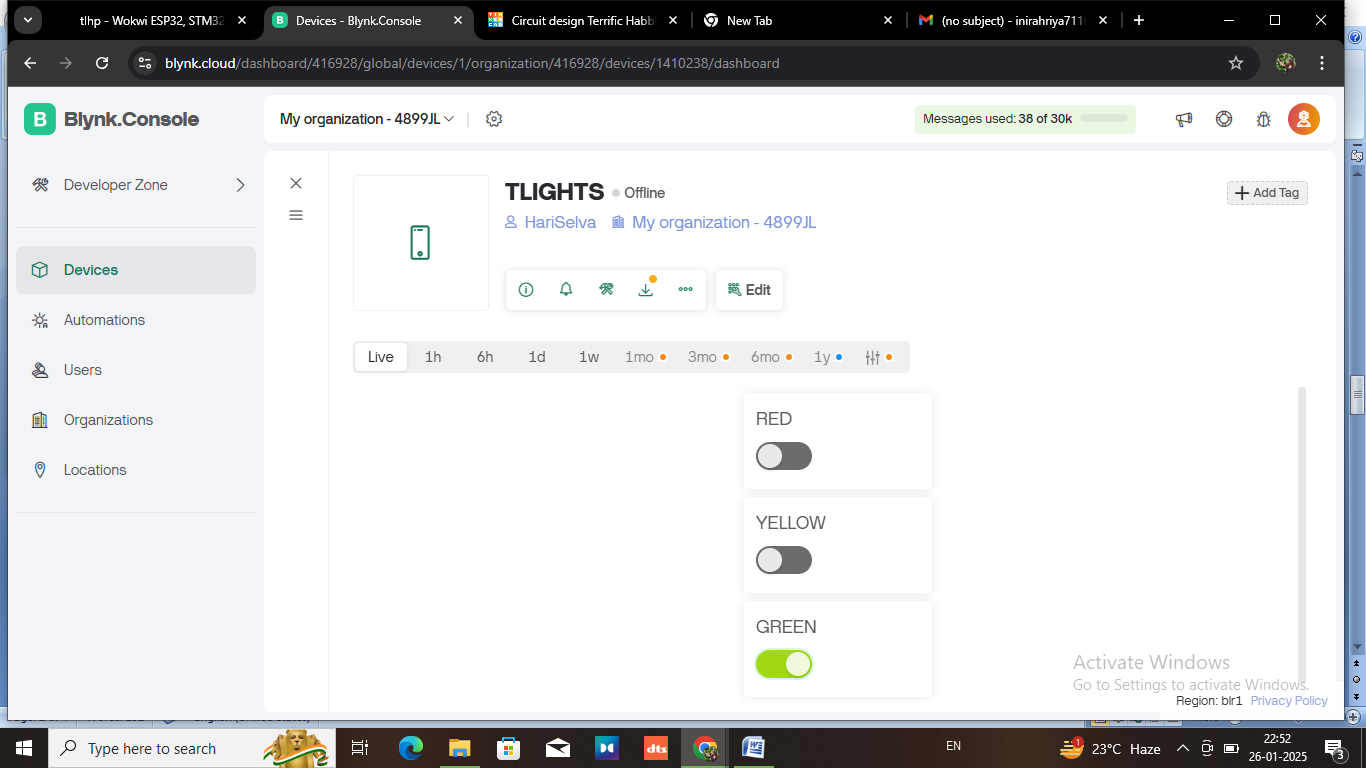
****



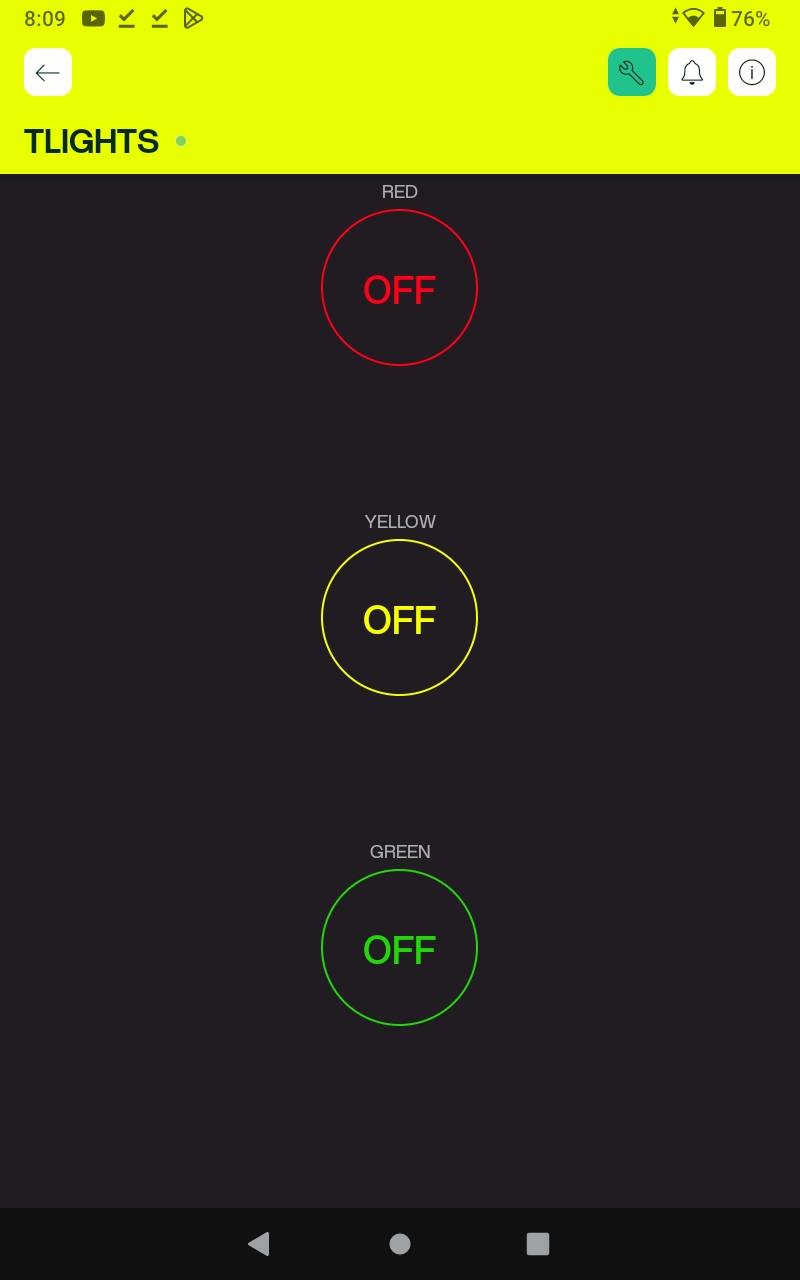
****



****



**MOBILE APP**

****