UNIVERZITET U TUZLI
FAKULTET ELEKTROTEHNIKE

SMART LIGHTING SYSTEM

SATELITSKE TELEKOMUNIKACIJE

STUDENT

AMELA KULIĆ

TUZLA, JUN, 2023. GODINE

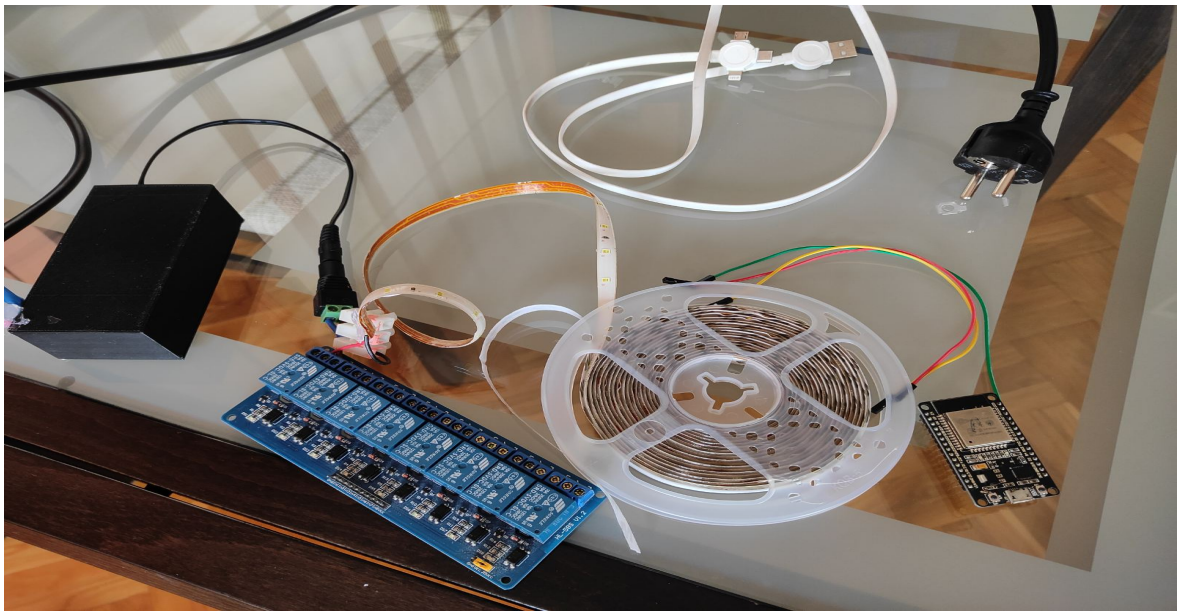
Pametna rasvjeta u kući se odnosi na upotrebu tehnologije i automatizacije kako bi se osvjetljenje u kući kontrolisalo na inteligentan način. Umjesto tradicionalnih prekidača i svjetiljki, koriste se pametne svjetiljke, senzori pokreta, daljinske upravljači ili aplikacije za pametne telefone kako bi se prilagodilo osvjetljenje prema potrebama korisnika. Evo nekoliko prednosti i mogućnosti koje pametna rasvjeta u kući može pružiti:

- Prilagodljivost
- Automatizacija
- Daljinsko upravljanje
- Integracija sa drugim pametnim uređajima

Daljinsko upravljanje

Kroz ovaj izvještaj upoznat ćete se sa tehnikama izrade pametne rasvjete u smislu uključivanja i isključivanja rasvjete i u slučajevima kada niste prisutni u prostoriji. Za ovaj projekat korištena je sljedeća oprema:

- ESP32 mikrokontroler
- Relay modul
- Jumperi
- LED traka
- USB kablo
- 220 V to 12 V converter



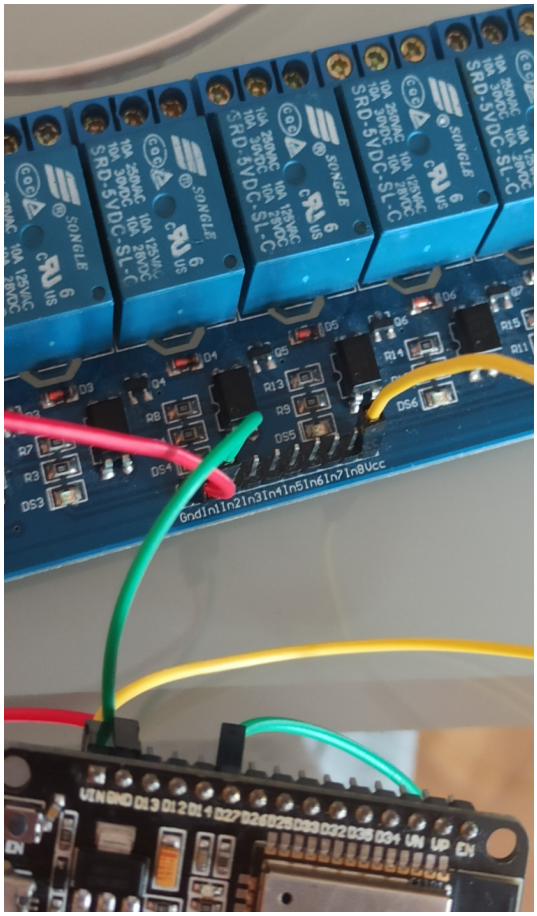
ESP32 je popularni mikrokontroler koji se koristi za razvoj IoT (Internet of Things) projekata. IoT (Internet of Things) projekti su projekti koji uključuju povezivanje fizičkih uređaja s internetom kako bi se omogućila razmjena podataka, upravljanje i nadzor tih uređaja putem mreže. U ovom slučaju pokazat ćemo kako postaviti ESP32 kao pristupnu tačku koristeći Arduino IDE. U nastavku se nalazi objašnjenje koda u Arduino IDE alatu pomoću kojeg se vrši instalacija ESP32 mikrokontrolera kao pristupne tačke.

- Znači, pristupna tačka (AP) omogućava ESP32 mikrokontroleru da se ponaša kao samostalna WiFi mreža. To znači da uređaji mogu povezati s ESP32 direktno, bez potrebe za postojećom WiFi mrežom.
- Korištenje ESPAsyncWebServer biblioteke omogućava brzo i jednostavno postavljanje web server na ESP32 mikrokontroleru. Ova biblioteka podržava asinhroni rad, što omogućava obradu više zahtjeva istovremeno.
- Uz pomoć HTML, CSS i JavaScript koda, kreirali smo vlastitu web stranicu koja se prikazuje kada korisnik poveže s ESP32 pristupnom tačkom.

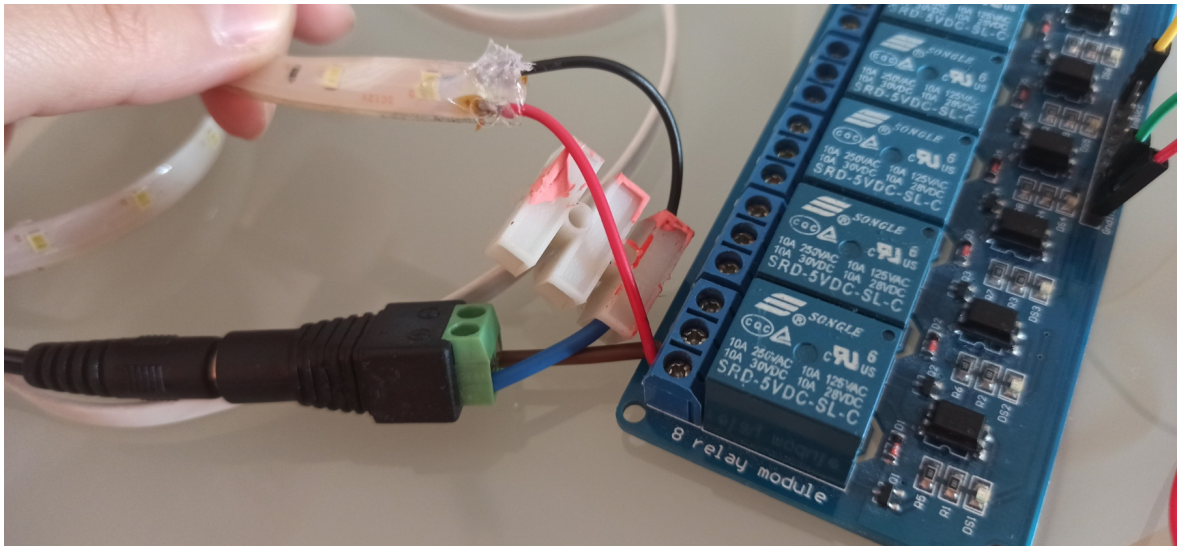
U zaključku, postavljanje ESP32 mikrokontrolera kao pristupne tačke i web servera pruža nam mogućnost da kreiramo samostalne WiFi mreže i kontrolišemo uređaje putem web interfejsa.
Nakon instaliranja ESP32 ploče u Arduino IDE, izvršeno je spajanje mikrokontrolera sa relay modulom. Korištena su 3 jumpera i spajanje je izvršeno na sljedeći način:

- GND mikrokontrolera spaja se za GND relaya
- Vin pin mikrokontrolera spaja se na Vcc pin relaya
- GPIO26/27 pin mikrokontrolera spaja se na input1 relaya

Prethodno opisani postupak prikazan je na sljedećoj slici:



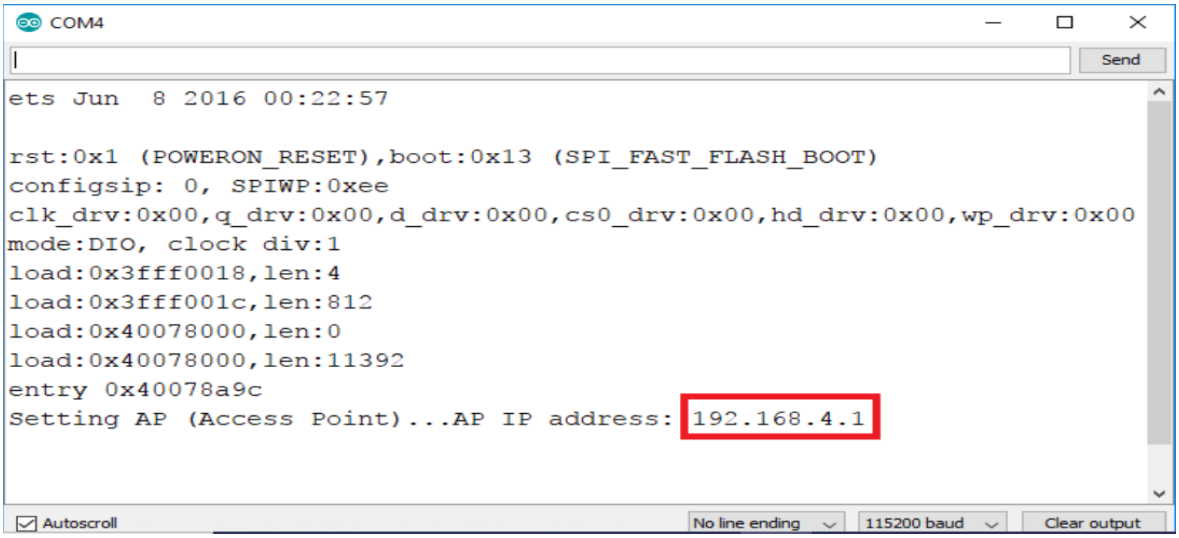
Nakon toga se mikrokontroler spaja na izvor napajanja preko USB kabla.
Zalemili smo vodiče na izlazne pinove LED trake i povezali ih na relay modul. Kada povežemo LED traku sa relay modulom, relay modul će djelovati kao prekidač za napajanje LED trake. Relay modul ima ulazni pin koji je povezan sa mikrokontrolerom ESP32, a izlazni pin koji je povezan sa napajanjem LED trake. Također, LED traka se priključuje na izvor napajanja preko 220V to 12V konvertera.



Kao što je navedeno na slikama koje objašnjavaju kod, postavlja se naziv mreže u varijablu "ssid". SSID (Service Set Identifier) predstavlja ime Wi-Fi mreže koja se koristi za identifikaciju i pristup mreži. U kontekstu priloženog koda, SSID je varijabla koja sadrži naziv pristupne tačke koju ESP32 mikrokontroler stvara.
Zatim moramo dobiti IP adresu pristupne tačke pomoću softAPIP() metoda.
WiFi.softAP(ssid, password); je funkcija u ESP32 Arduino biblioteci koja se koristi za postavljanje ESP32 mikrokontrolera kao pristupne tačke (Access Point - AP) u WiFi mreži. Ova funkcija omogućava ESP32 da kreira vlastitu WiFi mrežu na koju se drugi uređaji mogu povezati. Argumenti ssid i password predstavljaju naziv i lozinku mreže koju će ESP32 kreirati.

```
IPAddress IP = WiFi.softAPIP();
Serial.print("AP IP address: ");
Serial.println(IP);
```

Nakon izvršenja ovog dijela koda, na serijskom monitoru će se ispisati IP adresa pristupne tačke.



Nakon povezivanja na WiFi mrežu koju smo napravili, u web browseru kucamo IP adresu pristupne tačke koju stvara ESP32 mikrokontroler. U ovom slučaju to je 192.168.4.1. Prikazuje nam se izgled aplikacije definisane pomoću HTML, CSS i JavaScript-a. Ovisno o tome koji pin mikrokontrolera je spojen na relay, vrši se paljenje i gašenje rasvjete. U našem slučaju, u kodu smo tu mogućnost definisali za pinove GPIO26 i GPIO27. Izgled aplikacije prikazana je na sljedećoj slici, a sama funkcionalnost će biti predstavljena na odbrani završnog projekta.