

# VIN PROJEKT – GROWBOX

## 1. Uvod

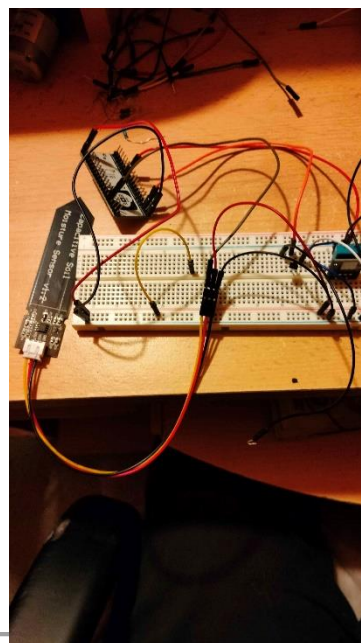
Projekt obravnava implementacijo sistema za avtomatizirano zalivanje rastlin in nadzor temperature s pomočjo ESP32 mikrokontrolerja. Uporabljamo DHT11 senzor za merjenje temperature, analogni senzor vlage, rele za prižig led luči ter črpalko za zalivanje rastlin. Sistem deluje na način, da črpalka zalije rastlino, ko je vlaga pod določenim pragom, in izklopi svetlobo, ko temperatura preseže kritično vrednost.

---

## 2. Cilji projekta

Cilji projekta so:

- Prebrati temperaturne podatke iz DHT11 senzorja in na podlagi temperature krmiliti rele (luč).
- Prebrati podatke iz analognega senzorja vlage in krmiliti črpalko za zalivanje, ko vrednost vlage pade pod določen prag.
- Implementirati funkcionalnost v programskem jeziku C, z uporabo razvojnega okolja ESP-IDF za ESP32.
- Omogočiti avtomatizirano delovanje sistema in njegovo prilagoditev z enostavno spremembo kode.
- Primerjati pridelek rastlin, če rastejo pod lučjo v škatli ter če zrastejo zunaj pod soncem.

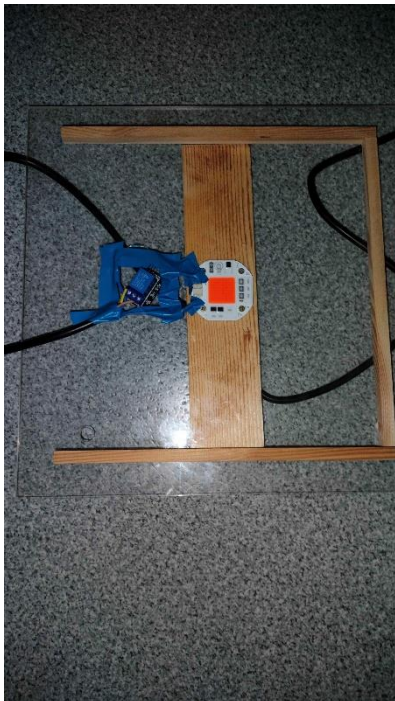


## 3. Opis uporabljenih komponent

V projektu smo uporabili naslednje komponente:

- **ESP32 WROOM-32:** Mikrokontroler z vgrajenim Wi-Fi in Bluetooth modulom, ki omogoča prebiranje analognih in digitalnih vhodov ter krmiljenje izhodnih naprav.

- **DHT11 senzor:** Senzor za merjenje temperature in relativne vlažnosti zraka. Uporabljamo ga za zaznavanje previsokih temperatur.
- **Analogni senzor vlage:** Senzor, ki meri vlažnost prsti. Njegov izhod je analogna vrednost, ki jo preberemo preko ADC vhoda ESP32.
- **Črpalka:** Naprava, ki je uporabljena za zalivanje rastlin, ko vlažnost prsti pade pod določen prag. Črpalka sicer ne deluje s polno močjo saj zahteva 6V priklopil se jo pa na 5V.
- **Rele:** Komponenta, ki omogoča preklapljanje napajanja za luč glede na temperaturo zraka.
- **BC547 tranzistor:** Tranzistor omogoča večji pretok toka do črpalke saj porabi več kot zmore pin na esp32.
- **LED luč:** Uporabljena led luč je priklopljena na 220V in uporablja 10k ohm upor saj drugače premočno sveti kar ni idealno za rastlino.
- **Powerbank:** Za napajanje vsega razen luči uporabljan 15000mA/h powerbank.



#### 4. Opis programske kode

Program je napisan v jeziku C in teče na ESP32 z uporabo ESP-IDF (Espressif IoT Development Framework).

1. **Inicializacija senzorjev in naprav:** Na začetku kode definiramo GPIO pin za DHT11 senzor, črpalko in rele. Nato inicializiramo analogni vhod za senzor vlage.

2. **Branje temperature in vlage:**

- Temperaturo preberemo s funkcijo `dht_read_temperature()`, ki pridobi vrednosti s pomočjo DHT11 senzorja.
- Vrednost senzorja vlage preberemo preko ADC (Analogno-Digitalni Pretvornik) vhoda z uporabo funkcije `adc1_get_raw()`, ki vrne vrednost od 0 do 4095.

3. **Krmiljenje naprav:**

- Ko senzor vlage zazna vrednost, nižjo od 3300, se črpalka vklopi za 15 sekund, nato se izklopi.

- Ko temperatura preseže 45°C, se rele vklopi, kar simulira izklop luči, da se prepreči pregrevanje. Rele ostane vklopljen, dokler temperatura ne pade pod 45°C.
4. **Zakasnitve:** Sistem vključuje zakasnitve med odčitki senzorjev in krmiljenjem naprav, da preprečimo prehitro izvajanje operacij ter porabimo manj el. Energije saj se vse razen luči napaja iz baterije. Te zakasnitve so implementirane z uporabo FreeRTOS funkcij.
- 

## 5. Struktura kode

Programska koda je modularno zasnovana in je razdeljena na več funkcij za boljšo preglednost in prilagodljivost.

- **Funkcija `init_gpio()`:** Inicializira GPIO pine za črpalko in rele.
  - **Funkcija `init_adc()`:** Inicializira ADC vhod za prebiranje analognega senzorja vlage.
  - **Funkcija `read_moisture_sensor()`:** Prebere in vrne vrednost senzorja vlage.
  - **Funkcija `app_main()`:** Glavna funkcija, kjer je implementirana logika za nadzor temperature in vlage ter krmiljenje naprav.
- 

## 6. Rezultati in testiranje

Sistem je bil uspešno testiran s fizičnimi napravami. Temperatura in vlažnost sta bili pravilno odčitani, črpalka in rele pa sta se odzvala na pogoje, določene v kodi. Da sem pridobil pogoje, ki sem jih uporabil v kodi sem testiral kakšne izpise sem dobil od senzorja vlažnosti v različnih pogojih in s temi odčitku sem okvirno postavil mejo za zalivanje.

- Pri nizkih vrednostih vlage se je črpalka vklopila in zalila rastlino.
- Pri visokih temperaturah se je rele vklopil, kar izklopi luč za zaščito pred pregrevanjem.

## 7. Možnosti izboljšav

Možnost za izboljšavo projekta je več kot dovolj, torej bom izpostavil par ključnih izboljšav

- **Ogrodje za vezje:** V prihodnje bom nadgradil projekt in izgled samega projekta tako da bom naredil ogrodje za samo vezje, ker trenutno mi povsod visijo žice in esp ker je vse samo prilepljeno na plexisteklo z izolirnim trakom.
- **Integracija z Wi-Fi:** ESP32 podpira Wi-Fi, kar bi omogočilo oddaljeno spremljanje in nadzor sistema preko spleta.
- **Dodajanje nadzora nad svetlobo:** Poleg temperature bi lahko sistem spremljal tudi koliko je svetlo okoli same škatle ter glede na to prilagodil svetlost (če je škatla na soncu izklopi luč)
- **Sistem pumpanja vode:** Pri tem sistemu nastane problem saj ima pumpa ki jo imam kratke cevi in na način kakor sem naredil je možnost da voda začne izpuščati iz plastenke in tudi izgled ni pretirano lep. To bi spremenil tako da bi platenko z vodo dal izven same škatle in podaljšal cev (platenko bi lahko tudi obrnil pokončno posledično, da ne more spuščati voda). Poleg tega bi pa za daljšo cevjo lahko tudi bolj lepo zalival rastlino.



---

## 8. Zaključek

Projekt je dokaj dosegel zastavljene cilje in demonstriral, kako lahko s pomočjo ESP32 mikrokontrolerja avtomatiziramo nadzor temperature in vlage v okolju rastlin. Sistem omogoča osnovno avtomatizirano zalivanje in nadzor nad pregrevanjem, kar bi bilo mogoče še razširiti z dodatnimi funkcijami, kot so oddaljen dostop in izboljšani senzorji ter lepši izgled in ogrodje. Sicer sem projekt uspešno testiral, ampak na žalost mi je zmanjkalo časa, da bi dlje testiral in primerjal rastline iz škatle ter rastline zunaj.

---

## 9. Literatura

- Espressif Systems. ESP-IDF Programming Guide. Dostopno na: <https://docs.espressif.com/projects/esp-idf/en/latest/esp32/get-started/index.html>

- G. Schorcht. DHT Sensor library for ESP-IDF. Dostopno na:  
<https://github.com/gschorcht/dht>