

# HidroGarden

## Introducere

- Proiectul presupune realizarea unui sistem automatizat de udare a plantelor;
- Scopul sistemului este de a detecta atunci cand o plantă are nevoie de apă pentru a putea oferi oxigen și de a absorbi cantități cât mai mari de CO<sub>2</sub>;
- In functie de anotimp sau de mediul în care stă o plantă, are nevoie de o cantitate diferită de apă;
- Este util pentru mine, cat si pentru alti oameni care au plante in casa deoarece pot economisi timp datorita acestui sistem care va uda plantele in locul lor;

## Descriere generală

- Senzorul măsoară nivelul de umiditate din sol și transmite aceste date la microcontroller, care decide dacă plantele au nevoie de udare;
- Microcontroller-ul gestionează datele primite de la senzor si gestioneaza pornirea si oprirea pompei de apa;
- Led-uri funcționeaza asincron pe parcursul udării;
- Ecranul LCD va arata daca procesul de udare este activ/oprit și valorile de umiditate;
- Vom folosi un buzzer care va produce melodia de intro din Star Wars pe parcursul udării;



## Hardware Design

- lista piese:
  - ESP32- microcontroller(am ales acest microcontroller pentru că față de ATmega328P are suport nativ pentru Bluetooth sau Wi-fi);
  - senzor de umiditate a solului - detectează cand pământul este uscat si are nevoie de apă;
  - releu - controlat de ESP32 pentru a porni/opri alimentarea cu apa;
  - pompă de apă - produce apa;
  - Furtun/tuburi- pentru a transporta apa la plante;

- sursa de alimentare;
- cabluri;
- Display LCD1602 HD44780 cu ecran albastru iluminat;
- LED-uri;
- Rezistențe 220 Ohmi;
- difuzor- notifică prin niște sunete ca procesul de udare s-a terminat;
- Modul buzzer activ;
- Fire dupont tata-mama 20cm;
- Fire dupont tata-tata 20cm;
- Fire dupont mama-mama 20cm;
- Update in loc de releu - am folosit tranzistor NPN;



## Conectivitate

- ESP32 D21- display SDA
  - ESP32 D33- senzor umiditate AO
  - ESP32 D22- display SCL
  - ESP32 D27- buzzer module I/O
  - ESP32 D12- rezistenta LED verde
  - ESP32 D14- rezistenta LED rosu
  - ESP32 D32- releu IN(uodate am folosit tranzistor NPN)
- 
- VCC:- se conecteaza la toate dispozitivele care au nevoie;
  - GND:- se foloseste ca punct de referinta pentru toate tensiunile;
  - Am folosit un tranzistor NPN care are conectat la baza o rezistenta de 2000 de Ohm, conectata la randul ei la un pin digital.
  - Emitorul tranzistorului e conectat la GND, iar colectorul este conectat la plusul de la pompa de apa.
  - Rezistentele le am folosit pentru a proteja LED-urile si de a putea controla intensitatea lor;
  - Vom folosi baterii pentru a furniza energie pompei de ape, conectand - de la baterie la - de la pompaterminalul;
  - Toate componentele functioneaza

## Software Design

Descrierea codului aplicației (firmware):

- mediu de dezvoltare:Arduino IDE
- Am folosit librarii precum BlynkSimpleEsp32(specifica Blynk- platforma care ne ajuta sa controlam dintr-o aplicatie partea hardware, cu ajutorul codului pe care l-am implementat)
- Pentru buzzer parcurg vectorul in timp se pompa este pornita, si note va fi indicele care va parcurge notele din Star wars. Parcurgerea va fi oprita o data cu oprirea pompei, acesta reluandu-si activitate

dupa ce a fost pornit.

## Rezultate Obținute

Senzorul de umiditate din sol masoara nivelul umiditatii, iar in functie de nivelul pe care il afiseaza display-ul, cand va fi apasat butonul din aplicatie, se va decide daca planta are nevoie de apa. Cand se porneste pompa va fi afisat un mesaj pe display sub indicile care masoara umiditate, care ne arata daca pompa functioneaza(butonul corespunzator aplicatiei este ON). Pe parcursul udarii led-urile vor functiona asincron si buzzer-ul va functiona.

## Concluzii

Proiectul a fost interesat de implementat. Partea hardware mi s-a parut mai greu de implementat fata de software. Am avut probleme cu releu, deoarece cand apasam butonul de ON si apoi OFF din aplicatie, pompa de apa nu se mai oprea. Asa ca am folosit un tranzistor NPN alaturi de o rezistenta de 2000.

## Download

\*(In arhiva am 2 poze cu hardware-ul pe langa codul sursa) [watersystempreda.zip](#)

## Jurnal

- 12-19 mai- am implementat schema electrica alaturi cu releu si module buzzer activ
- 20-26 mai- am implementat partea soft
- 20-26 mai- am schimbat schema electrica din cauza ca am folosit tranzistor npn in loc de releu si buzzer activ in loc de module buzzer activ.

## Bibliografie/Resurse

<https://randomnerdtutorials.com/getting-started-with-esp32/>

<https://www.arduino.cc/en/software>

Listă cu documente, datasheet-uri, resurse Internet folosite, eventual grupate pe **Resurse Software** și **Resurse Hardware**.

[Export to PDF](#)

From:

<http://ocw.cs.pub.ro/courses/> - **CS Open CourseWare**

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2024/fstancu/ioan.preda>



Last update: **2024/05/26 21:45**