Sistem de monitorizare a mediului inconjurator

Introducere

Sistemul avansat de monitorizare a mediului înconjurător este un dispozitiv care are scopul de a furniza informații esențiale pentru îngrijirea optimă a plantelor. Ideea de bază a proiectului a pornit de la nevoia de a oferi soluții tehnologice inovatoare care să faciliteze și să îmbunătățească procesul de creștere a plantelor în diverse medii. Sistemul utilizează senzori specializați pentru a colecta date despre umiditatea solului, temperatura ambientală și nivelul de lumină, oferind utilizatorului informații detaliate despre condițiile de mediu. Prin intermediul funcționalităților precum activarea LED-ului RGB în lipsa luminii sau pornirea unui buzzer în cazul scăderii umidității sub un anumit prag, proiectul nu doar monitorizează, ci și reacționează proactiv la schimbările din mediu. Sunt convinsă că acest sistem va fi util nu doar pentru mine, ci și pentru alții, oferindu-le posibilitatea de a îngriji plantele în mod mai eficient și sustenabil.

Descriere generală

Schemă bloc

Senzorii transmit informațiile colectate către Arduino (temperatura, umiditatea solului, intensitatea luminoasă), care se va folosi de ele pentru a le transmite catre LCD, buzzer și LED. Valorile colectate se vor afișa pe ecran și în diferite cazuri LED-ul se va aprinde sau buzzer-ul va porni.

Hardware Design

Circuit

Componente utiizate:

- **Arduino Uno ATmega328P**
- **Breadboard**
- **Senzor de temperatură și umiditate DHT11**
- **Senzor de intensitate luminoasă GY-302 BH1750**
- **Senzor de umiditate a solului**
- **Ecran LCD 1602 IIC/I2C**
- **LED**
- **Buzzer**

Pini

În acest proiect am folosit următorii pini:

- **Pin A0** → Conectat la output-ul analogic al senzorului de umiditate a solului. Acest pin citește valoarea analogică a umidității solului.
- **Pin A4** → Pinul de date pentru comunicația I2C, folosit pentru a comunica cu LCD-ul și senzorul de lumină BH1750.
- **Pin A5** → Pinul de ceas pentru comunicația I2C, folosit pentru a sincroniza datele cu LCD-ul și senzorul de lumină BH1750.
- **Pin 3** → Conectat la buzzer. Buzzer-ul se activează atunci când umiditatea solului depășește un anumit prag.
- **Pin 9** → Conectat la LED-ul galben. LED-ul își ajustează luminozitatea în funcție de nivelul de lumină ambientală măsurat de senzorul BH1750.
- **Pin 13** → Conectat la senzorul de temperatură și umiditate DHT11. Acest pin citește datele de temperatură și umiditate din mediul înconjurător.

Software Design

Pentru partea software, inițial am împărțit proiectul în mai multe părți: am luat fiecare componentă pe rând și am încercat s-o fac să funcționeze. Ulterior, după ce fiecare senzor a mers și am afișat datele pe Serial Monitor, am adăugat și codul pentru ecranul LCD și le-am afisat și pe ecran. Am adăugat și implementarea pentru LED și pentru buzzer și în final am ajuns la rezultatul dorit.

- Initializarea și scrierea pe display (prin I2C) se realizează utilizând biblioteca LiquidCrystal_I2C.h.

- Buzzer-ul și LED-ul sunt controlați folosind regiștrii.

- Citirea nivelul de lumină ambientală în unități de lux. Funcția 'map()' transformă valoarea de lux citită într-o valoare de luminozitate adecvată pentru LED. Intervalul de intrare (0-400 lux) este mapat la intervalul de ieșire (255-0) pentru controlul PWM al LED-ului. Aceasta înseamnă că un nivel

de lumină scăzut va corespunde unei luminozități mari a LED-ului și viceversa. Funcția 'constrain()' se asigură că valoarea mapată a luminozității este în intervalul acceptabil de 0 la 255.

- Citirea datelor de la DHT11 și senzorul de umiditate a solului.

- Controlul buzzer-ului pe baza umidității solului folosind registrii. Dacă umiditatea solului depășește pragul ('threshold'), buzzer-ul este activat. Dacă umiditatea solului este sub prag, buzzer-ul este dezactivat.

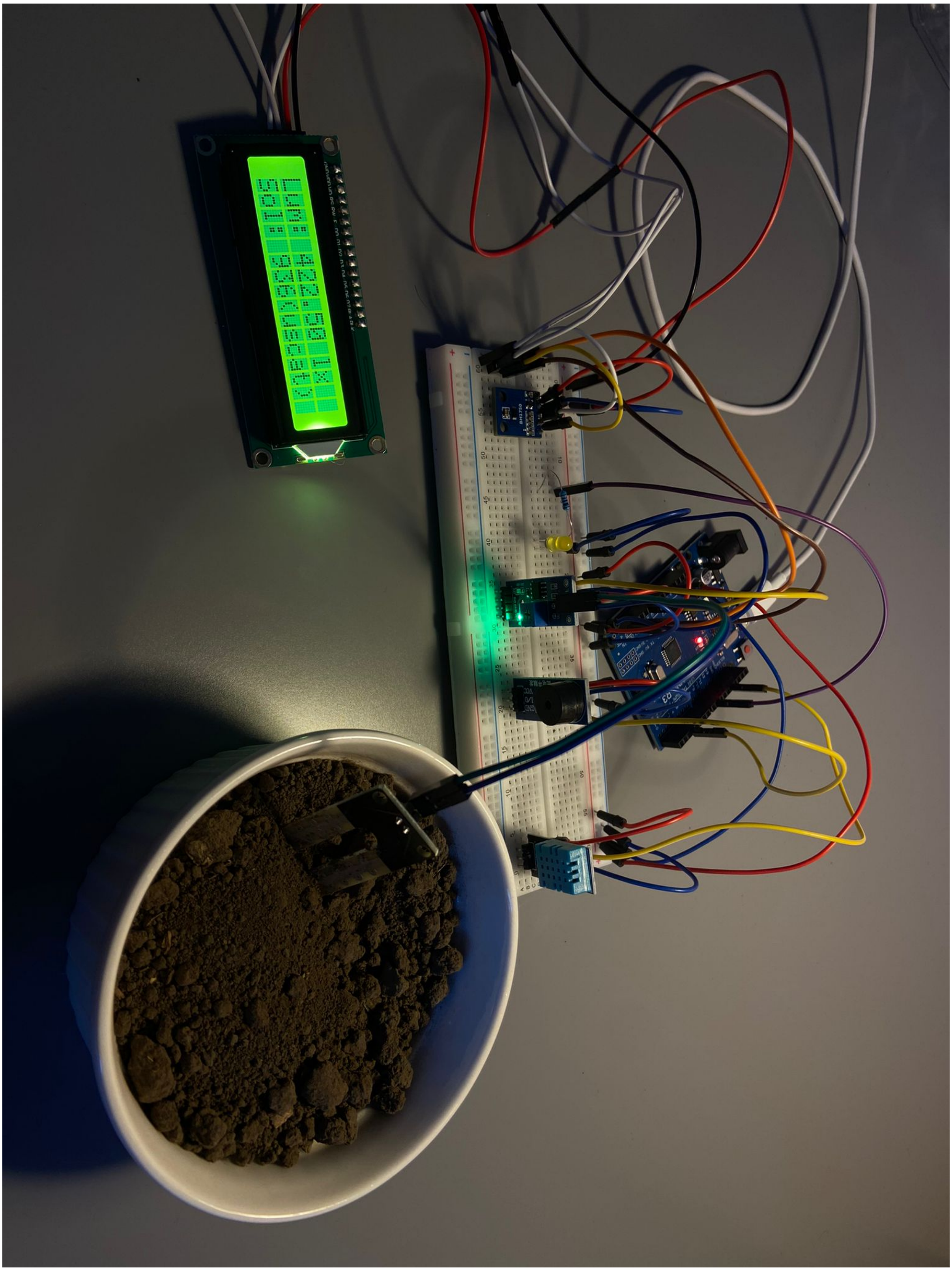
- Afișarea informațiilor pe LCD, variabila 'count' având rolul de a alterna afișarea informațiilor pe LCD între două seturi de date diferite la fiecare interval de timp definit.

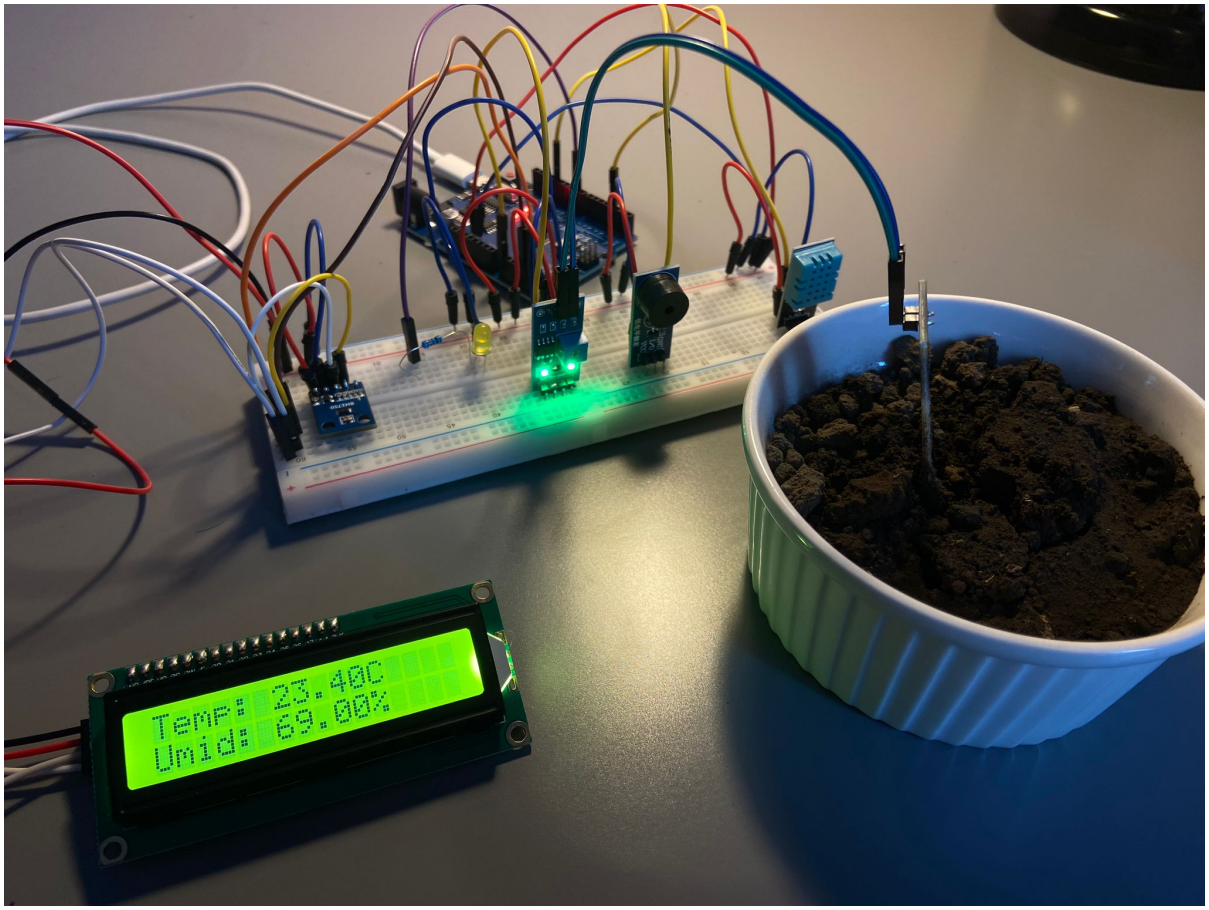
- Mediu de dezvoltare: Arduino IDE
- Librării folosite: DHT.h, LiquidCrystal_I2C.h, Wire.h, BH1750.h

Rezultate Obţinute

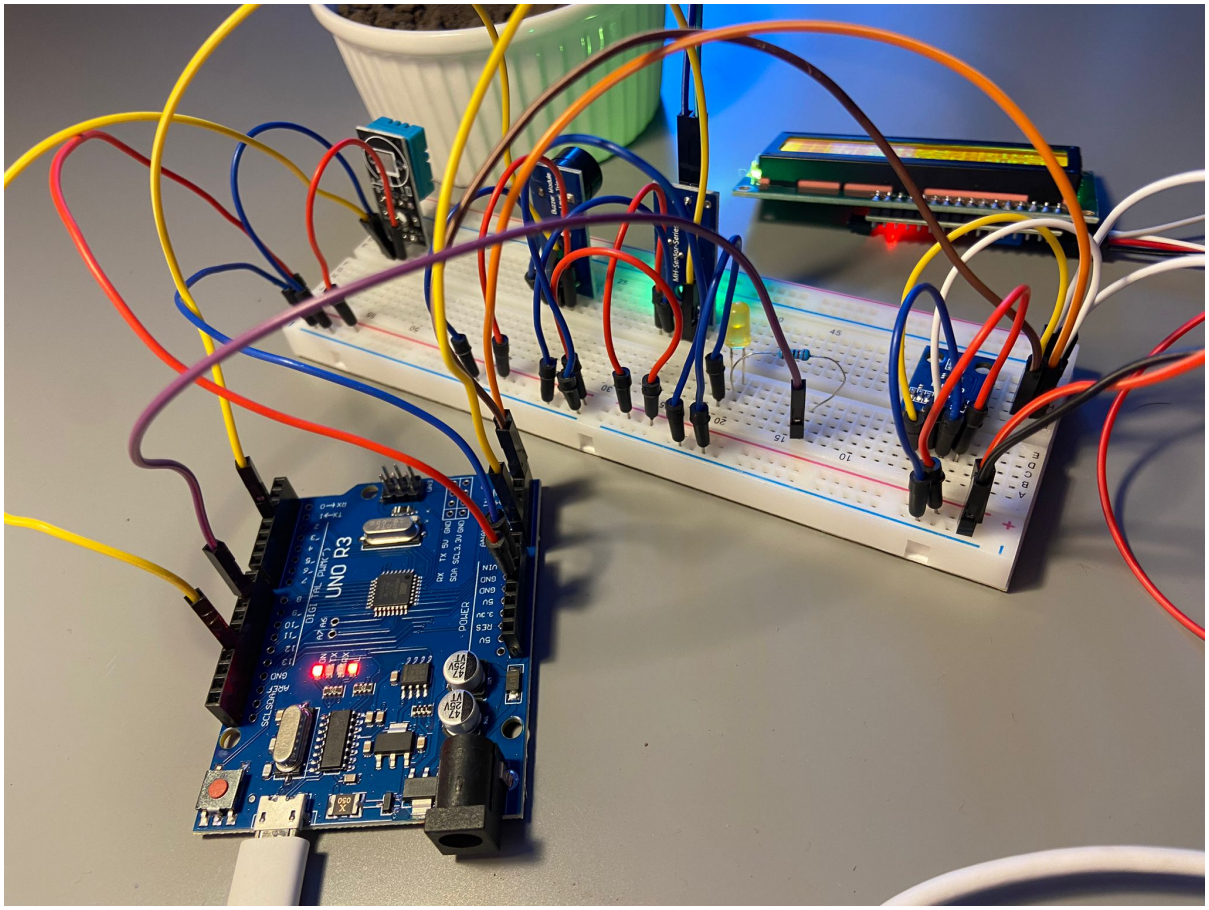
Rezultatele obţinute în urma realizării proiectului: https://www.youtube.com/watch?v=1X3rE3mfLKI

General


Lum: 422.50 lx
Soil: 936(Wet)



Cablaj



Concluzii

Mi-a plăcut să lucrez la acest proiect și sunt mulțumită de rezultatul final. Am invățat să lipesc componente și am întâmpinat situații dificile în care componentele nu funcționau, dar am reușit să rezolv problemele. Cu siguranță voi folosi proiectul în viitor, ba chiar voi încerca să implementez mai multe funcționalități.

Download

Arhivă proiect: delia_maria.rosu_pm.zip

Jurnal

27 aprilie - Alegere temă proiect

3 mai - Achiziționare piese

7 mai - Crearea paginii de documentație

17 mai - Milestone Hardware

21 mai - Întâmpinare problemă LCD

23 mai - Finalizare proiect

24 mai - Milestone Software

25 mai - Finalizarea paginii de documentație

Bibliografie/Resurse

https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf

https://docs.arduino.cc/resources/pinouts/A000066-full-pinout.pdf

https://www.youtube.com/watch?v=x45Fwd0VQjU

https://www.youtube.com/watch?v=CvqHkXeXN3M&t=481s

Export to PDF