

Plant Watcher

Introducere

- Proiectul urmareste mediul in care se afla planta, mai precis, umiditatea solului, temperatura, presiunea aerului si altitudinea.
- Informatiile precizate mai sus sunt transmise prin SMS, cu un modul GSM. Sunt 3 cazuri in care se declanseaza trimiterea de informatii, detaliate mai jos, unde este descris Flow-ul.
- Scopul lui este sa ajute la intretinerea plantei si a mediului in care se afla aceasta, pentru a se dezvolta intr-un mod optim.
- Ideea a pornit de la plantele pe care uit sa le ud la timp si ele au de suferit pana imi amintesc. Cu proiectul asta, as primi mereu notificari si probabil ca nu as mai uita.
- Proiectul poate sa fie util atat pentru plantele tinute in apartament, dar si pentru fermierii care au suprafete mai mari, intr-un spatiu inchis (de ex. sere / solarii), care, prin stabilirea unor praguri de umiditate / temperatura pot controla cu mai mare precizie cantitatea de apa si temperatura pe care o ofera plantelor. De asemenea, si presiunea aerului este utila.

<https://pubmed.ncbi.nlm.nih.gov/11538810/>

Descriere generală

Schema bloc



Flow

- Arduino citeste valorile de la cei 3 senzori (de la BMP280 temperatura + presiune + altitudine, de la cel de sol, umiditatea solului, iar de la DHT11, umiditatea)
- Exista 3 situatii in care se poate declansa trimiterea unei notificari:
 1. Un trigger manual: un buton - util pentru cazul in care se vor informatiile in acel moment (are totusi un cooldown de 30 de secunde)
 2. Un trigger periodic: este prestabilit la 6 ore, se poate modifica - util pentru cazul in care vrem sa primim informatiile chiar daca nu suntem langa Arduino.
 3. Un trigger care actioneaza ca un watchdog: se uita tot timpul la valorile de temperatura si sol (pe acestea le-am considerat cele mai importante), iar cand face un numar de citiri succesive cu valori sub / peste pragurile prestabilite, declanseaza trimiterea unui mesaj de avertizare.
- Modulul SIM800L se ocupa de trimiterea notificarii.

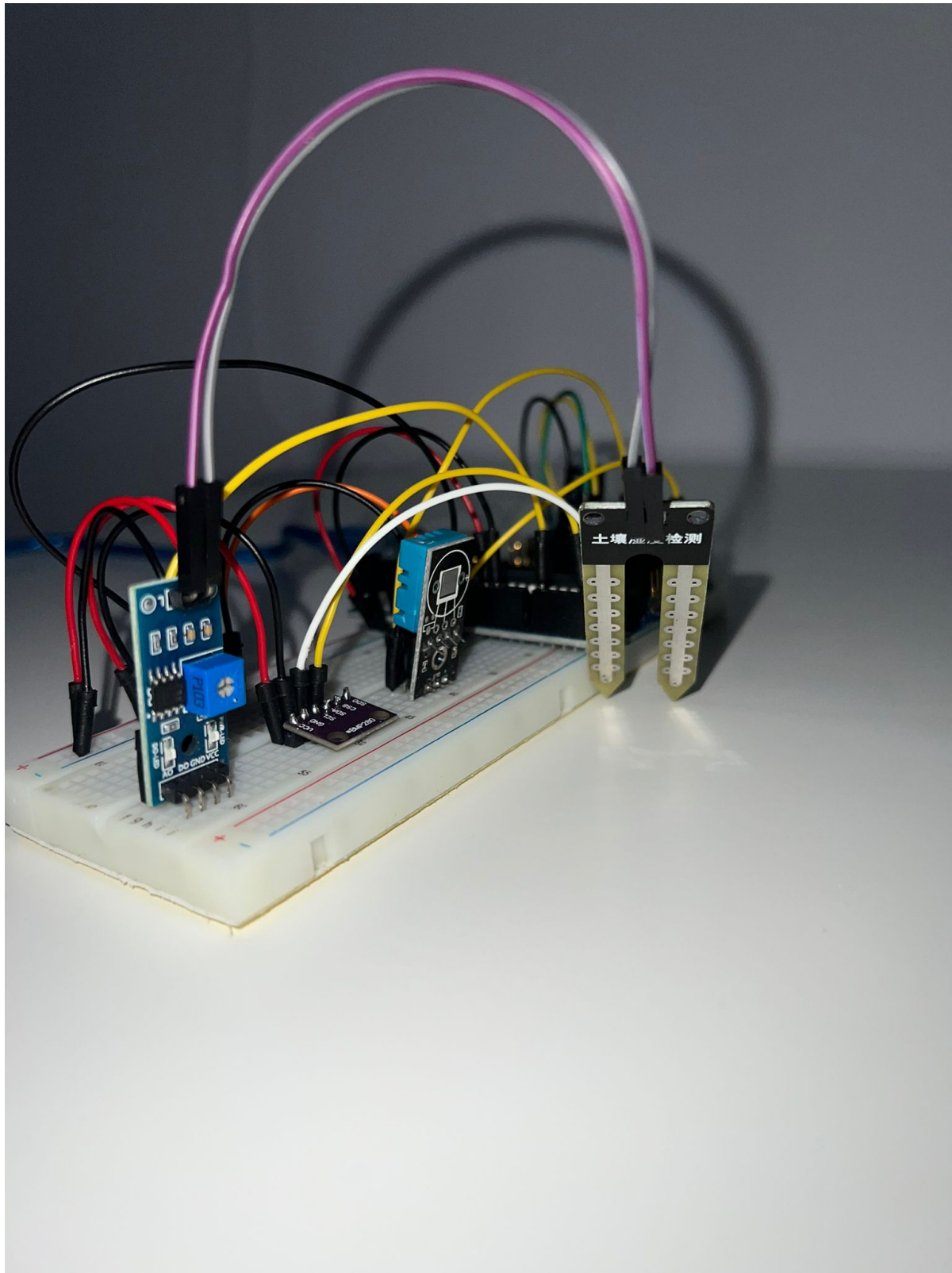
Toate pragurile, cat si perioadele se pot modifica.

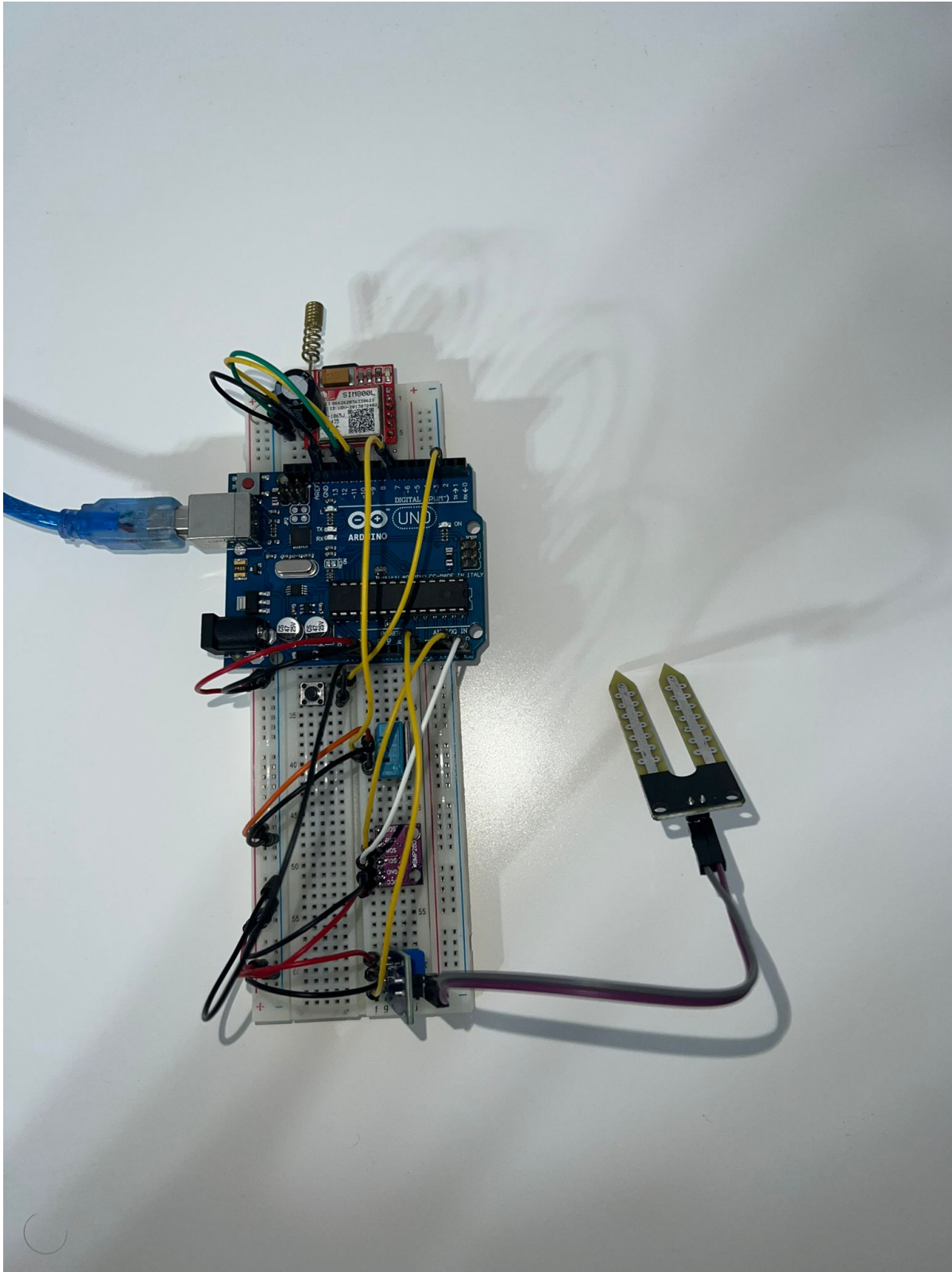
Hardware Design

Diagrama Hardware



Implementare Hardware





Lista componente

- Arduino UNO
- Modul Sensor de Temperatura și Umiditate DHT11
- FC-28 Soil moisture sensor
- Condensator 1500uF

- Dioda 1N4007
- BMP280: Senzor de temperatura si presiune
- SIM800L GSM Module
- Push-down button
- Breadboard

Software Design

Biblioteci

- [Software Serial](#)
- [Simple DHT](#)


Notiuni din laboratoare

- I2C - pentru BMP280
- ADC - pentru senzorul de sol
- Intreruperi hardware - pentru buton
- Timere - pentru trimiterea mesajului de informare, periodic

Toate au fost implementate utilizand registre!

Descrierea codului

Am modularizat implementarea, lucrul cu senzorii se face in fisiere separate, exportand apoi functii in headere.

- BMP280: Se foloseste si de functiile de I2C implementate separat, functii implementate si in laboratorul 6. Se cauta adresa, apoi ca in datasheet, se trimite un struct cu 12 campuri  si se configureaza sampling-ul si filtrarea. (Am ales x1 pentru temperatura, x16 pentru presiune si x16 la filtru). Se prelucreaza presiunea si temperatura ca in datasheet. Mai departe, cu ajutorul presiunii determin si altitudinea.
- FC-28 Soil Moisture: Un simplu ADC, fac setup-ul ca in laborator si citesc rezultatul. Acesta este ulterior transformat in procent.
- DHT11: Pentru umiditatea aerului, folosesc direct biblioteca DHT11. Functia read calculeaza si temperatura, dar o ignor, optand pentru cea calculata cu BMP280.
- In fisierul main, plant_watcher.ino, apelez functiile de setup de la senzori, apoi cu SoftwareSerial, fac setup-ul modulului GSM (acesta afiseaza la seriala daca a reusit sa se conecteze si cat de puternic este semnalul, pe o scala de la 0-30, 30 insemnand cea mai buna conexiune). De asemenea, fac setup-ul si pentru intreruperi, configurand intreruperea externa pe pinul 2, unde este butonul si Timer1.
- In Loop, citesc valorile de la senzori si cu functia add_info, construiesc in buffer mesajul. Pe langa

asta, gestionez si partea de avertizare, verificand valorile pentru temperatura, umiditate si tin cont de cooldown-ul pentru notificari.

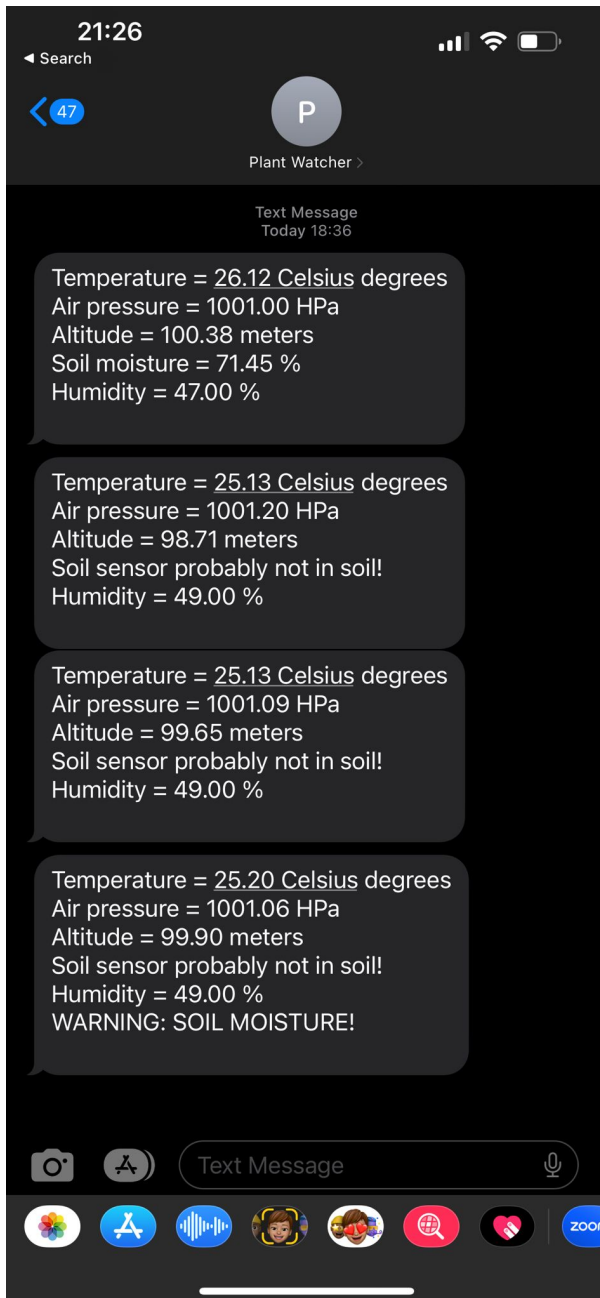
- O variabila foarte importanta este sendTrigger, de ea depinzand trimiterea mesajului. Aceasta se poate face true, fie din intreruperea de la buton, fie dupa un anumit timp pentru trimiterea mesajului de informare periodic, fie prin declansarea unei avertizari.

Github

[Link catre repo](#)

Rezultate Obținute

[Link Youtube](#)



Primul este mesajul de la testul facut in video. Urmatoarele sunt teste pentru informarea periodica + informarea de avertizare. (Testele sunt facute cu senzorul de sol scos din pamant, de aceea apare acea informatie in dreptul valorii de sol)

Bibliografie/Resurse

- Laboratoarele 0-5
- [Last minute engineers - SIM800L](#)
- [Acest comentariu excelent](#)
- [Biblioteca de BMP280 de la Adafruit](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2023/alexau/plantwatcher>



Last update: **2023/05/28 19:29**