

Agregare senzori de gradina

Introducere

Eu am un istoric lung cu diverse plante pe care uit sa le ud, sa le pozitionez intr-un loc cu soare sau sa nu le las iarna in frig.

Scopul acestui proiect este dezvoltarea unui dispozitiv care sa urmareasca situatia unei plante si sa puna datele intr-un server web.

Odata agregate datele se pot face numeroase metode de notificare a situatiei plantelor (e.g. mail, mesaj pe mobil).

Descriere generală

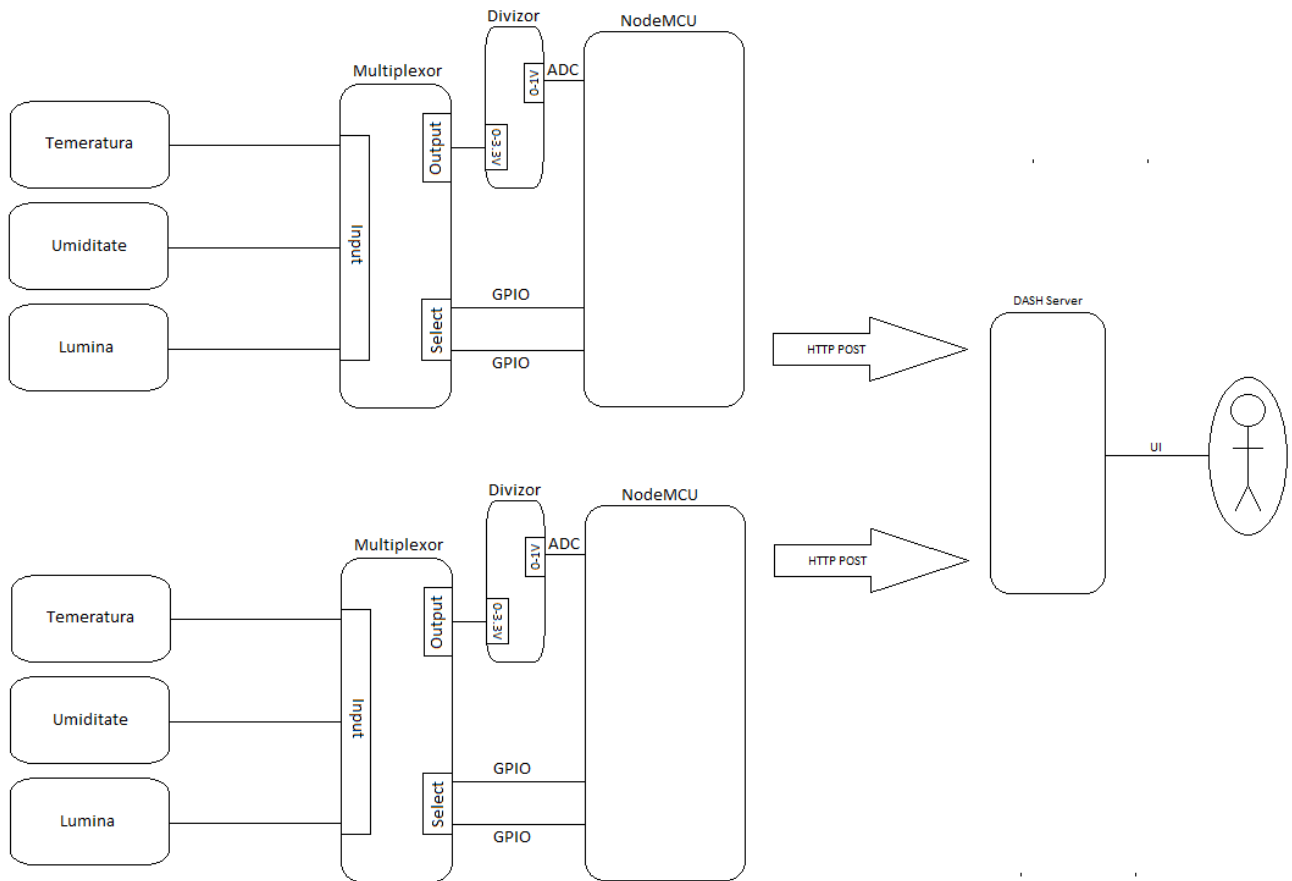
Am decis sa folosesc NodeMCU, dar acesta are un singur pin de conversie Analog-Digitala.

De aceea este nevoie de un multiplexor care sa selecteze semnalul analog primit de la senzori. Selectia multiplexorului este facuta tot de microcontroler prin GPIO.

De asemenea, pinul ADC de pe placuta NodeMCU accepta tensiune in valoarea **0-1V** de aceea este nevoie si de un divizor de tensiune care primeste ca intrare **0-3.3V**.

Rezultatul obtinut din divizorul de tensiune este interpretat si apoi trimis cu un request HTTP de tip POST catre un server in Google Cloud.

Datele pot fi vizualizate prin accesarea server-ului.



Hardware Design

O sa folosesc urmatoarele componente:

- placuta [NodeMCU](#)
- senzor temperatura [DS18B20](#)
- senzor fotorezistor [KY-018](#)
- senzor umiditate sol [YL-69](#)
- multiplexor [CD74HC4067](#)
- rezistente **100 Ohm** si **220 Ohm** pentru divizorul de tensiune
- un LED **rosu** care sa arate lipsa de conexiune la internet
- un LED **verde** care sa arate prezenta conexiunii la internet

Sursa de internet in este un hotspot facut pe telefonul mobil.

Software Design

Folosesc **Arduino Studio** ca IDE cu extensia pentru Node MCU.

Pentru partea de server am folosit **Visual Studio Code** si pyhon. Libraria de UI este **Dash** iar pentru HTTP server este **Flask**

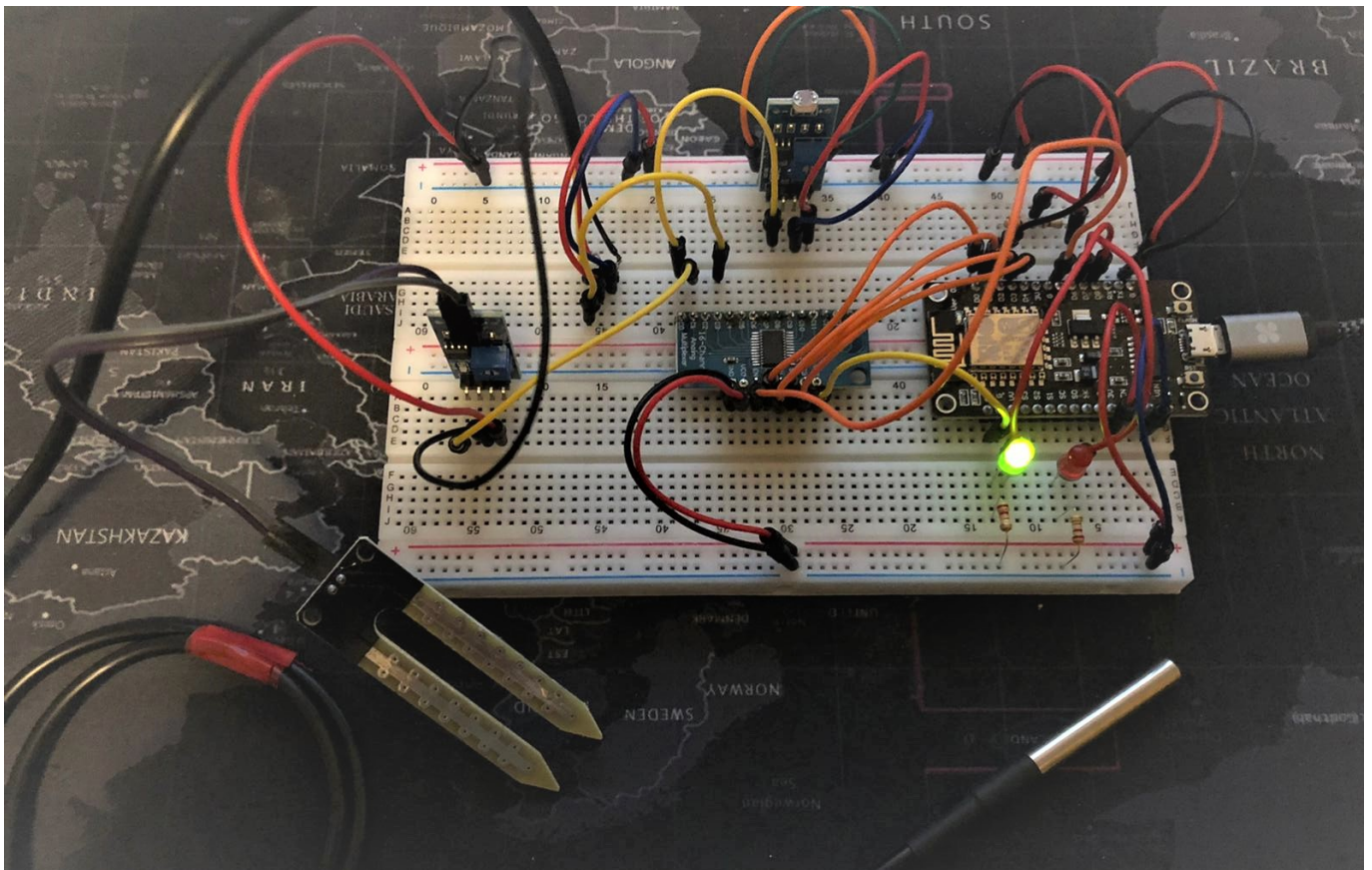
Pentru a citi senzori la un interval de timp mai mare (jumătate de secunda) am folosit **timer1** de pe placuta NodeMCU.

Comunicarea client-server este peste HTTP, folostind POST.

Rezultate Obținute

[AICI](#) se poate vedea dashboard-ul tinut in Google Cloud Platform.

Aici se poate observa o poza cu placuta in starea finala.



In dashboard se pot vedea la umiditate si la lumina modificari marcate cu galben. In cazul luminii

tensiunea creste cand acopar senzorul cu un deget; la umiditate tensiunea scade cand bag senzorul intr-un pahar cu apa.

Senzorul de temperatura are un semnal **gresit**, in urma investigatiilor nu am reusit sa imi dau seama de ce.



Concluzii

In urma acestui proiect am invatat:

- sa folosesc placuta **NodeMCU** pentru a ma conecta la o sursa de WiFi
- sa trimit request-uri de la placuta la un server
- sa multiplexez un port **ADC** pentru a utiliza mai multi senzori decat permite o placuta.
- sa utilizez un divizor de tensiune impreuna cu un grup de senzori diferiti.

In final doar 2 din cei 3 senzori functioneaza conform asteptarilor mele.

Download

Codul sursa atat pentru client cat si pentru server se poate gasi in link-urile de mai jos.

[Cod sursa Client](#) [Cod sursa Server](#)

Jurnal

Initial doream ca divizorul de tensiune sa fie pe semnalul de iesire din senzori, dar intr-un final am ajuns sa fac divizorul de tensiune pe VCC al senzorilor.

Resurse

Resurse Hardware:

- <https://github.com/nodemcu/nodemcu-devkit-v1.0>
- <https://pdf1.alldatasheet.com/datasheet-pdf/view/58557/DALLAS/DS18B20.html>
- <https://datasheetspdf.com/pdf/1402029/Joy-IT/KY-018/1>
- <https://usermanual.wiki/Pdf/instructions.261306374.pdf>
- <https://pdf1.alldatasheet.com/datasheet-pdf/view/27059/TI/CD74HC4067.html>

Resurse Software:

- <https://tttapa.github.io/ESP8266/Chap07%20-%20Wi-Fi%20Connections.html>
- <https://randomnerdtutorials.com/esp8266-nodemcu-http-get-post-arduino/>
- <https://plotly.com/python/time-series/>
- <https://stackoverflow.com/questions/45845872/running-a-dash-app-within-a-flask-app>

Export to PDF

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2022/sionescu/533>



Last update: **2022/05/27 20:14**