

Stație meteo pentru interior

Autor: — [Laura-Elena GHEORGHE \(101329\)](#)

Introducere

Proiectul presupune crearea unui sistem care măsoară temperatura, umiditatea și presiunea în cameră, afișând aceste informații pe un ecran. De asemenea, acesta trimite datele și în cloud, cu posibilitatea de a le urmări în timp real de pe o platformă IoT (Ubidots).

Toți acești parametri (temperatura, umiditatea, presiunea) pot afecta starea de sănătate a persoanelor care petrec mult timp într-o cameră. Astfel, ar fi utilă monitorizarea acestora, pentru a se putea interveni în cazul detectării unor valori anormale.

Descriere generală



- Senzorul BME280 - este cel care se va ocupa de detectarea temperaturii, umidității și a presiunii, trimițând aceste valori către Arduino.
- Arduino Uno - procesează datele primite de la senzor. Le afișează pe ecranul TFT în culori sugestive și le trimite în cloud către platforma IoT.
- Platforma IoT - primește informațiile de la Arduino (prin modulul WiFi) și oferă posibilitatea de a fi accesate în timp real. Protocolul folosit este MQTT.

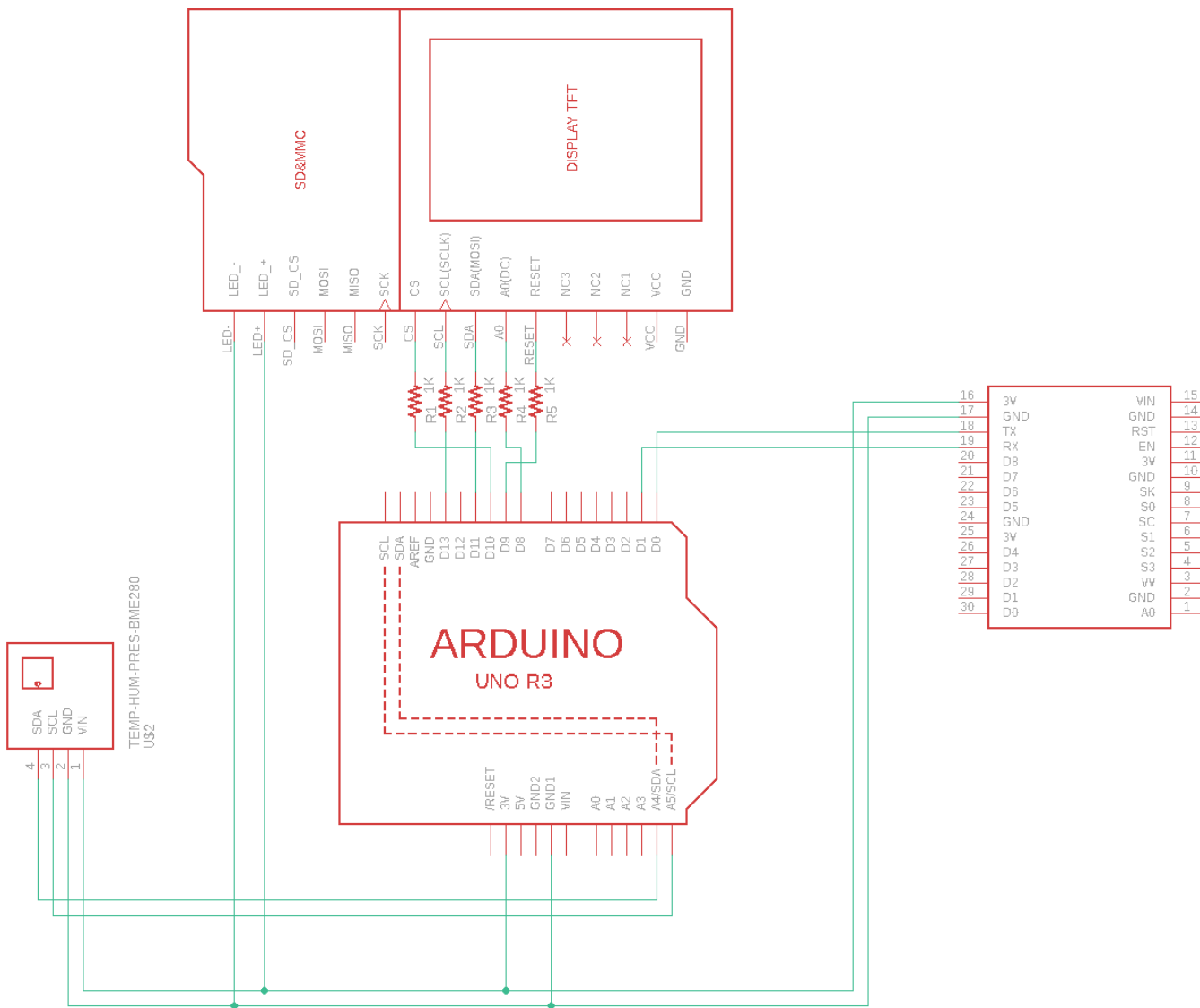
Hardware Design

Lista de piese

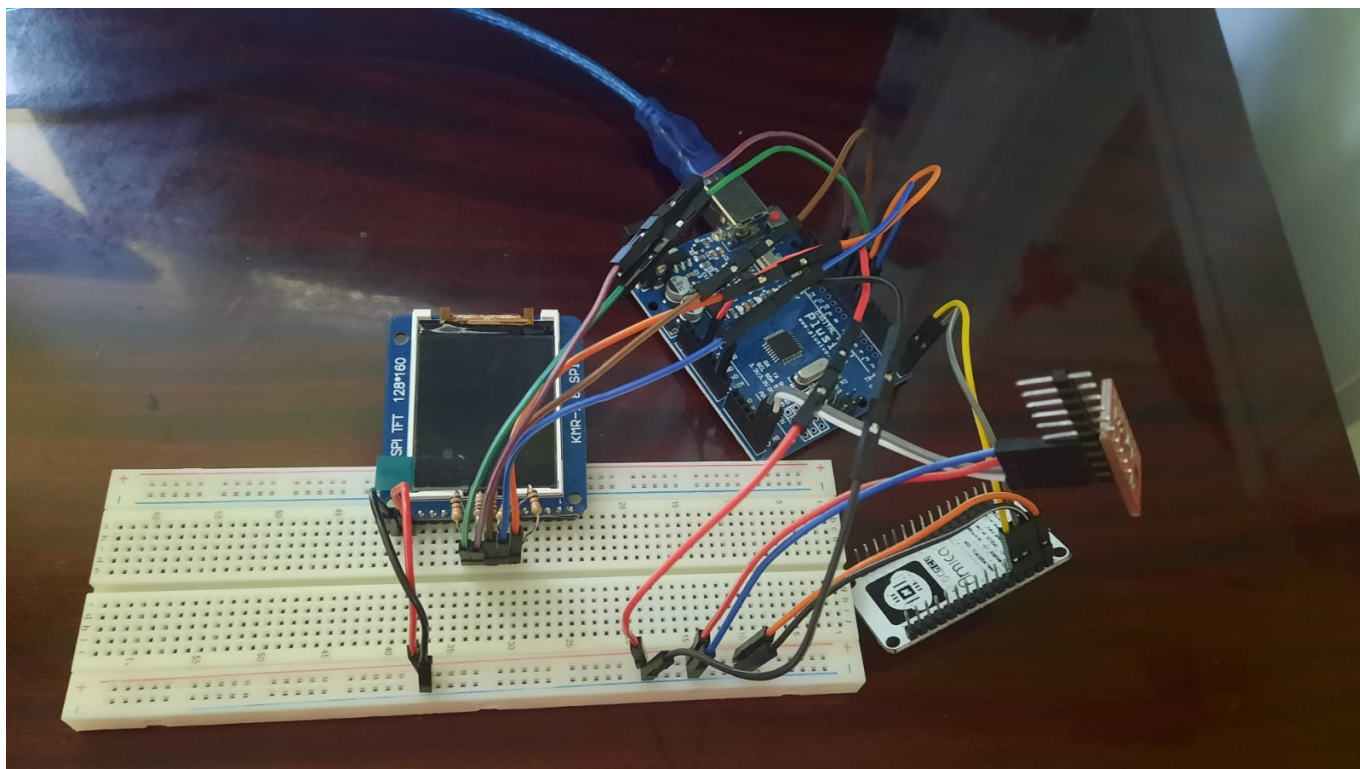
- Arduino Uno
- Ecran LCD TFT 1.8" ST7735
- Senzor BME280
- Modul WiFi NodeMCU ESP8266
- 5 rezistențe 1K
- Breadboard
- Fire mamă-tată, tată-tată

Pe ecranul LCD sunt afișate datele: temperatura (în °C), umiditatea (umiditatea relativă în %) și presiunea (în hPa), date preluate de către senzor. Placa Arduino trimite valorile către modulul WiFi, iar modulul le trimite mai departe către platforma Ubidots. Rezistențele sunt folosite pentru ecranul LCD.

Schema electrică



Hardware



Software Design

Componenta software cuprinde două fișiere: unul pentru preluarea datelor de pe senzor și afișarea pe ecran (`statie_meteo.ino`) și unul pentru comunicarea cu Ubidots (`wifi_module.ino`).

`statie_meteo.ino`

Am folosit următoarele biblioteci:

- `ArduinoJson` - pentru parsarea valorilor în format JSON către modulul WiFi
- `Adafruit GFX`, `Adafruit ST7735` - pentru ecranul TFT
- `Adafruit BME280` - pentru senzor

Pini `RST`, `CS`, `DC` ai ecranului sunt conectați pe placa Arduino la pini 9, 10, respectiv 8. Am determinat adresa I2C ca fiind `0x77` folosind un cod separat pentru determinarea acesteia.

În setup, se inițializează ecranul: fundalul este negru, se afișează „TEMPERATURE”, „HUMIDITY”, „PRESSURE” și alt text care nu se schimbă și nu este suprascris pe parcursul rulării. Tot aici se inițializează și senzorul BME280, verificându-se dacă este conectat. Dacă există vreo problemă de conexiune, se afișează un mesaj corespunzător pe ecran, iar rularea nu continuă.

În loop, se preiau temperatura, umiditatea și presiunea de la senzor și se afișează pe ecran, cu niște culori caracteristice. Apoi se creează stringul JSON ce le conține, string ce va fi trimis către modulul WiFi.

`wifi_module.ino`

Am folosit următoarele biblioteci:

- `ArduinoJson` - pentru extragerea valorilor primite de la Arduino în format JSON
- `UbidotsESPMQTT` - pentru comunicarea cu Ubidots prin protocolul MQTT.

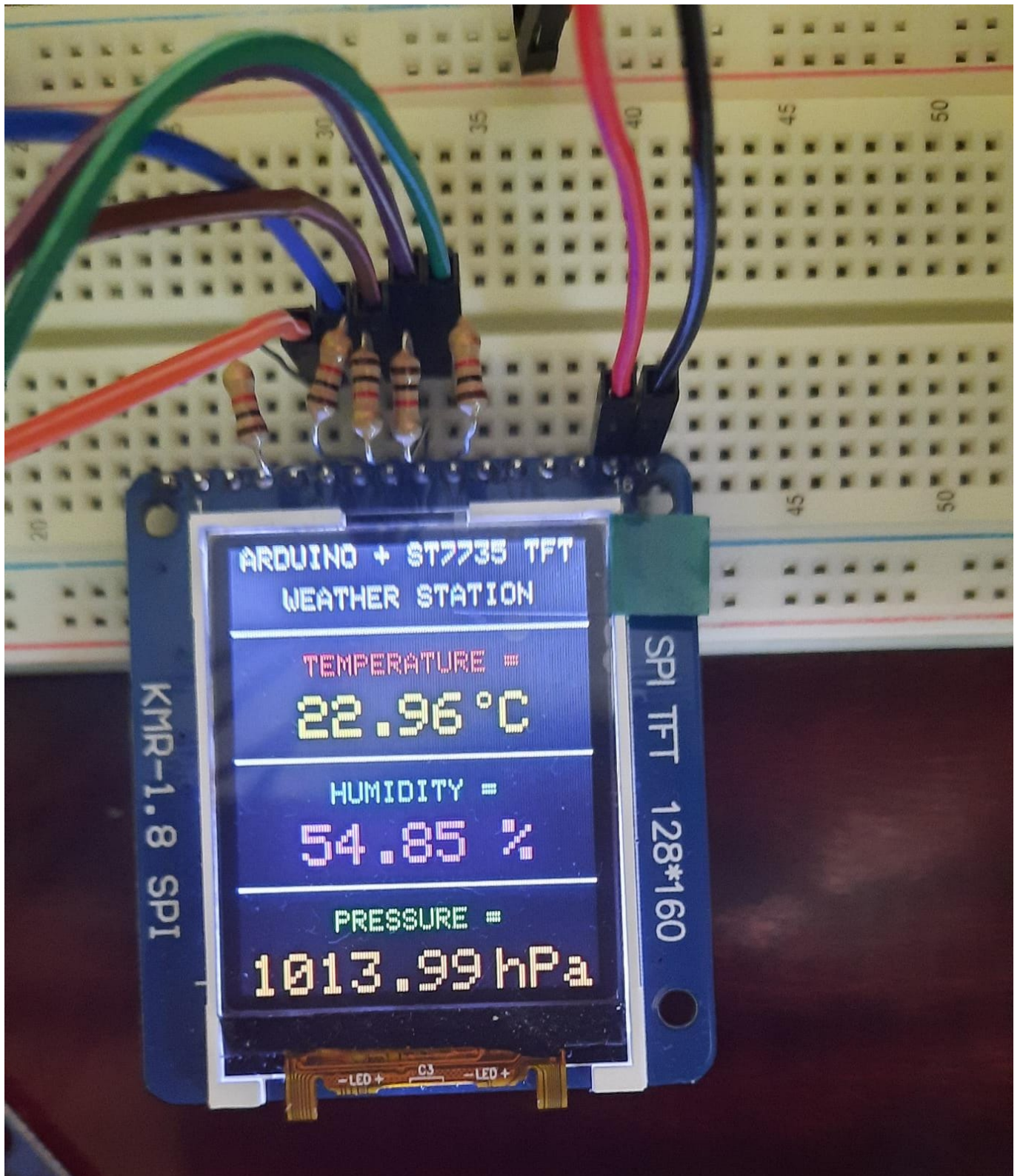
- PubSubClient - tot pentru comunicarea cu Ubidots

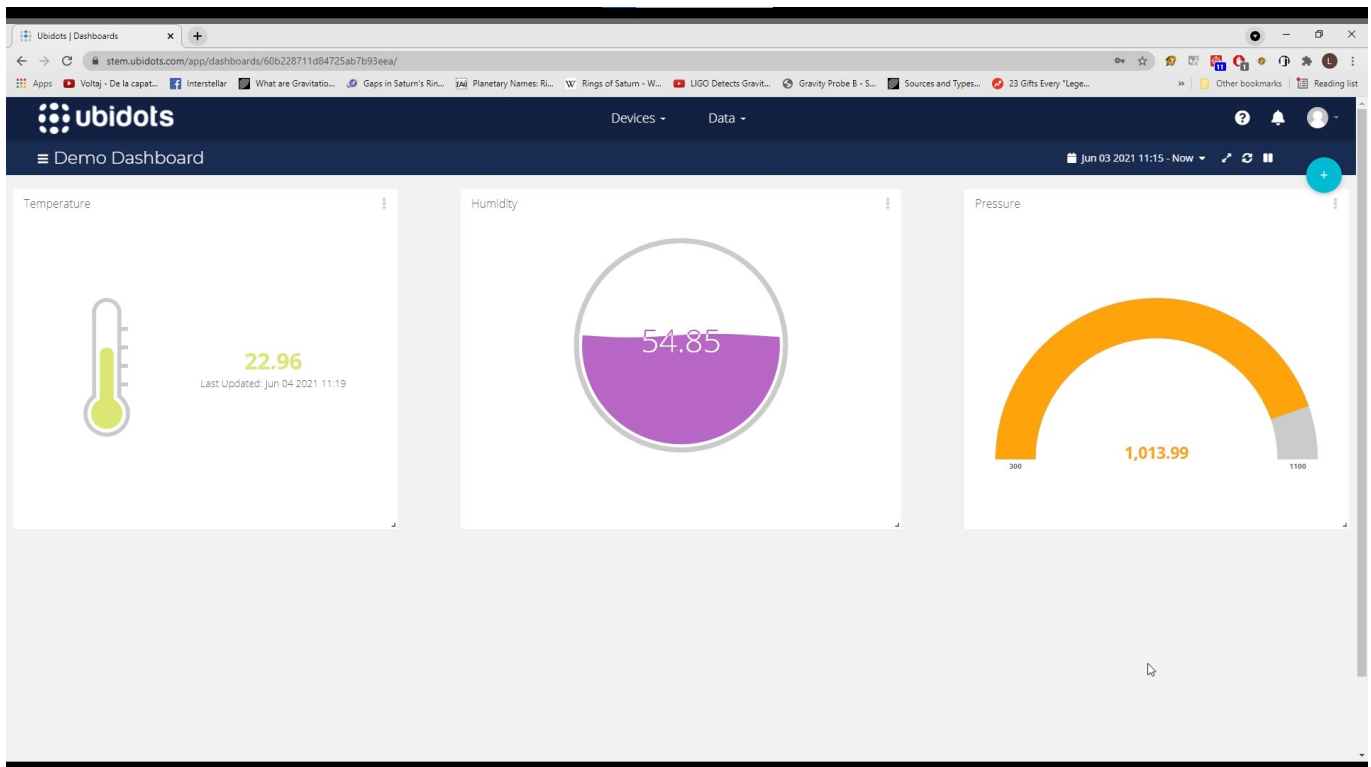
În setup, se inițializează conexiunea cu platforma Ubidots.

În loop, se verifică de fiecare dată dacă s-a întrerupt conexiunea, caz în care se încearcă reluarea acesteia. Apoi se așteaptă datele de la Arduino (sub formă de string JSON). Când este primit un mesaj, acesta este deserializat și se extrag valorile pentru temperatură, umiditate și presiune. Apoi, aceste valori sunt trimise către Ubidots. Pe această platformă, am creat un dashboard cu widgeturi pentru fiecare parametru.

Rezultate Obținute

Am reușit să implementez tot ce mi-am propus. Uneori am observat că există o desincronizare între datele trimise și datele vizibile în timp real pe Ubidots, dar acest lucru nu se întâmplă prea des și am pus-o pe seama comunicării cu serverul. Datele furnizate de senzor par în regulă și variază corespunzător cu mediul în care se află tot dispozitivul.





Este disponibil și un [demo](#) al proiectului.

Concluzii

Am învățat foarte multe pe parcursul realizării acestui proiect. Inițial neștiind nici măcar cum să leg firele, am simțit că evoluez și eu împreună cu el și am avut un sentiment de satisfacție uriaș de fiecare dată când reușeam să mai fac câte o funcționalitate sau să mai rezolv vreo problemă apărută. Totodată, am înțeles mult mai bine despre comunicarea serială, despre cum se îmbină componentele hardware și despre cum să fac debugging eficient.

Download


Arhiva conține fișierele `statie_meteo.ino`, `wifi_module.ino`, `README`, schema electrică și două fișiere folosite pentru debugging, `i2c_scan.ino` și `bme280test.ino`.

[lauragheorghestatiemeteodownload.zip](#)

Jurnal

- 2 mai 2021 - am creat pagina wiki (scurtă descriere, schema bloc, lista de componente)
- 12 mai 2021 - sosesc ecranul TFT și senzorul BME280. Placa Arduino, breadboardul, firele și rezistențele erau deja cumpărate.
- 13 mai 2021 - sosește comanda cu modulul WiFi.
- 19 mai 2021 - reușesc să conectez ecranul și senzorul la Arduino. A fost dificil deoarece ecranul meu are 16 pini iar singurul tutorial în care se explica ce înseamnă labelurile pentru fiecare pin (care oricum nu m-a ajutat prea mult) se numea „How to work with a 1.8” SPI TFT with strange incorrect labelling”. Ulterior am găsit o schemă de conectare pe un site care acum nu mai este disponibil și a trebuit să-l accesez cu Wayback Machine.
- 27 mai 2021 - încep schema electrică, dar nu adaug modulul WiFi până nu sunt sigură că merge.
- 28 mai 2021 - documentare despre cum se conectează modulul WiFi la Arduino și cum comunică

acesta cu Ubidots.

- 29 mai 2021 - reușesc să simulez niște șiruri JSON pe serială către modulul WiFi, iar acesta face parsarea și trimite corect datele către Ubidots prin MQTT. De asemenea, reușesc să fac ca Arduino să parseze corect parametri, deși observ niște output-uri ciudate uneori. Încă nu am legat Arduino de modulul WiFi.
- 30 mai 2021 - Arduino nu mai primește deloc datele de la senzor. După ce am mai modificat puțin așezarea componentelor pe breadboard, nici ecranul nu mai mergea. Panică. Am reușit să rezolv înlocuind rezistențele și am fost nevoită să conectez direct senzorul la Arduino, nu prin intermediul breadboardului. Acum așezarea nu mai e așa frumoasă, dar funcționează.
- 31 mai 2021 - reușesc să fac comunicarea Arduino - modul WiFi, dar uneori observ că parsarea la Arduino nu se face corect, astfel încât pe serială apar niște caractere ciudate în plus. Problema apărea pentru că am alocat prea multă memorie pentru JsonDocument .

Bibliografie/Resurse

- [Preluarea datelor de la senzor și afișarea pe ecran](#)
- [Comunicarea Arduino - NodeMCU și parsarea JSON](#)
- [Comunicarea NodeMCU - Ubidots \(prin MQTT\)](#)
- [Conectarea ecranului TFT cu 16 pini la Arduino](#)

Documentul în format PDF: [Statie meteo pentru interior](#)

From:

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

Permanent link:

http://ocw.cs.pub.ro/courses/pm/prj2021/apredescu/statie_meteo_pentru_interior 

Last update: **2021/06/04 12:22**