

Stație meteo cu predicție

- Nume autor: **Șendre Mihai-Alin**
- Grupa: 332CC
- e-mail de contact: **mihai.alin.sendre@gmail.com**

Introducere

Proiectul presupune implementarea unei **stații meteo de predicție a vremii** prin compararea presiunii atmosferice obținute prin măsurători efectuate în timestamp-uri anterioare. Stația meteo nu doar că mă va ajuta să aleg vestimentația potrivită în funcție de vreme, dar și să înțeleg algoritmi de predicție a vremii în funcție de presiunea atmosferică înregistrată de-a lungul unei zile - am pornit de la ideea că ar fi un bun exercițiu atât hardware, cât și software. 😊

Descriere generală

Predicția vremii poate fi făcută comparând presiunea atmosferică înregistrată de-a lungul timestamp-urilor anterioare folosind mai mulți senzori specializați (vezi secțiunea *Hardware design /schemă bloc*).

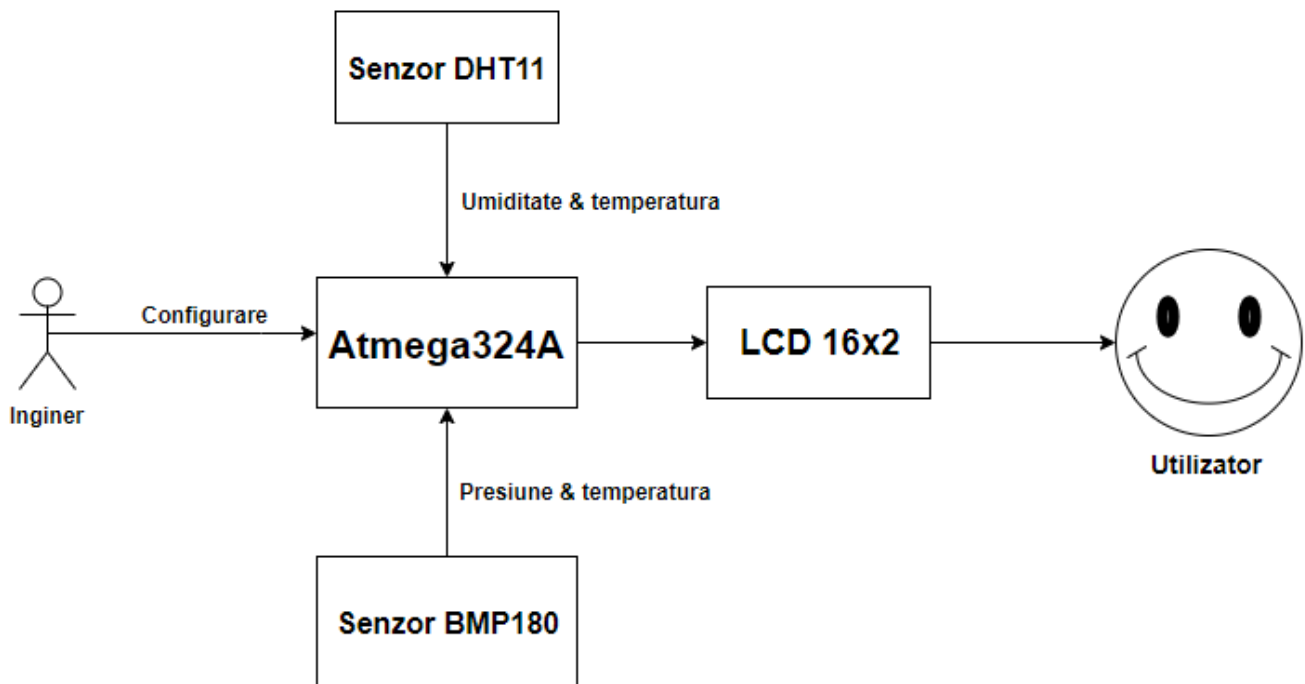
În locul timer-ului de pe *Atmega324A*, voi utiliza un modul **Real Time Clock (RTC)** pentru a obține ora și minutul cu o precizie superioară. Acest modul **RTC** poate fi configurat să genereze o întrerupere o dată la câteva ore, la intrarea într-un nou timestamp.

Correction: m-am hotărât să folosesc totuși timer-ul de pe *Atmega324A*, întrucât la *PM Fair* e doar un *proof of concept*, cu întreruperi o dată la 30 secunde.

Persistența datelor măsurate în timestamp-urile anterioare se poate asigura stocând datele obținute în memoria non-volatilă (**EEPROM**) a microcontroller-ului *Atmega324A*.

Senzorul de presiune funcționează în parametri optimi doar atunci când este alimentat la tensiunea de **3.3V** și nu tolerează tensiunea de **5V** de pe pinii logici - astfel, voi folosi un **stabilizator de tensiune de 3.3V** pentru a alimenta senzorul, iar semnalele folosite în comunicarea cu *Atmega324A* vor trebui trecute printr-un **translator de nivel** voi alimenta senzorul BMP180 de la pin-ul de 3.3V de pe plăcută.

Rezultatul prelucrărilor se va afișa pe un **LCD 16x2** - vor fi vizibile valorile actuale și anterioare ale parametrilor de interes și predicția vremii pentru ziua curentă.



Hardware design

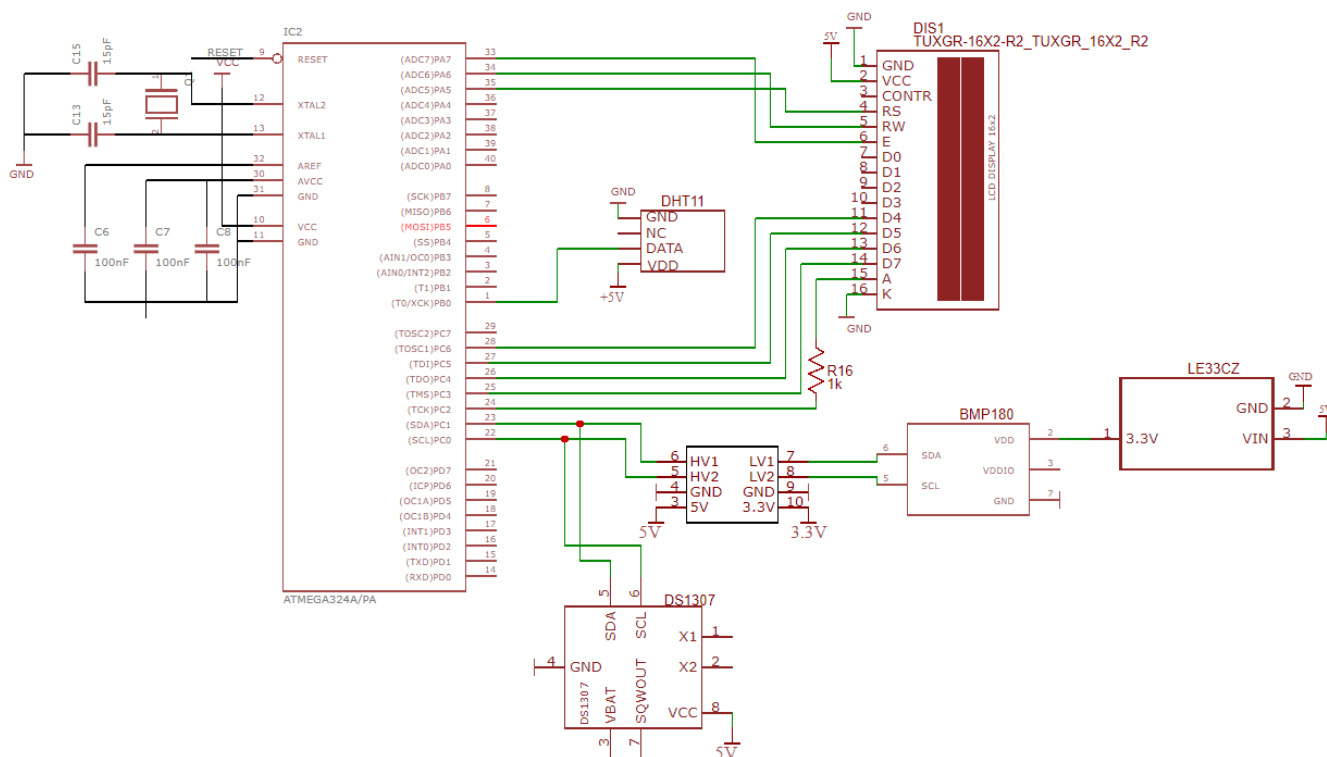
Lista de piese

- Kit PM de bază
- Senzor BMP180 de temperatură și presiune
- Senzor DHT11 de temperatură și umiditate
- LCD 16×2 cu backlight albastru, alimentare 5V
- Traductor de nivel
- Stabilizator de tensiune de 3.3V
- Modul Real Time Clock DS1307 cu baterie proprie

Lista de piese poate suferi mici modificări pe parcursul dezvoltării proiectului, la identificarea unor noi necesități.

Ulterior, am eliminat următoarele piese care îngreunau implementarea **Hardware** într-un mod inutil: *Translator de nivel* și *Stabilizatorul de tensiune 3.3V* le-am scos după ce am aflat că pot alimenta cu 3.3V direct de la pin-ul de pe plăcută, iar *DS1307 RTC* l-am eliminat întrucât nu aveam nevoie de precizie foarte mare pentru un *proof of concept* la *PM Fair* - ar dura mult prea mult să iau suficiente date dacă aș folosi *DS1307* pentru întreruperi o dată la o ora, așa că m-am rezumat la *Timer1* de pe plăcută.

Schema electrică



Schema electrică poate suferi mici modificări pe durata dezvoltării proiectului.

Software design

Proiectul presupune implementarea unor **algoritmi de predicție a vremii** bine-cunoscuți și simplificați care vor prelucra datele colectate de la senzori în timestamp-urile anterioare. Ca mediu de dezvoltare, am folosit **Atom**, iar codul a fost dezvoltat pe **Ubuntu 18.04**.

Biblioteci folosite

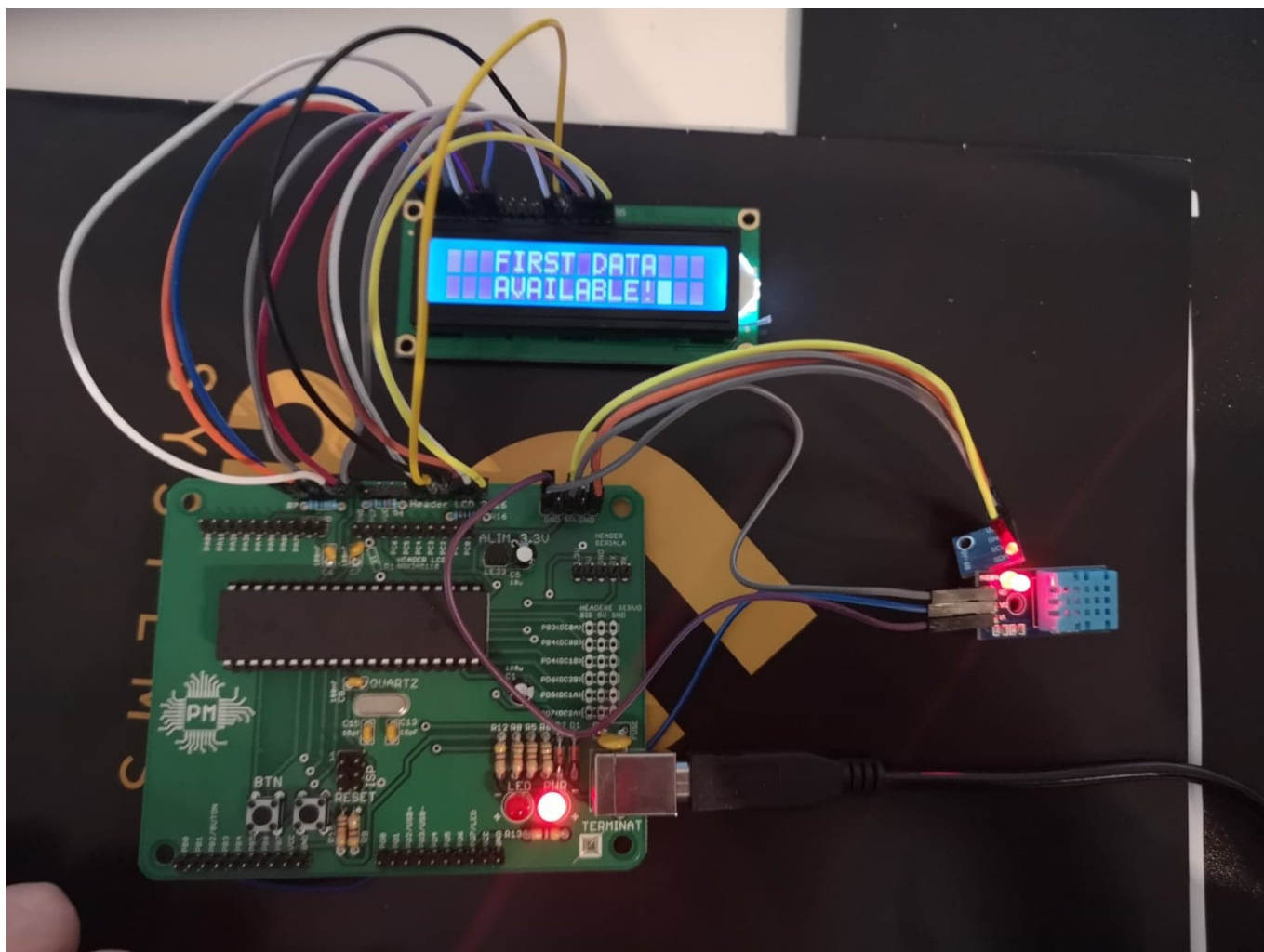
- **atmega_bmp180_lib** - folosită în comunicația cu senzorul *BMP180* de temperatura & presiune.
- **dht** - folosită în comunicația cu senzorul *DHT11* de temperatura & umiditate.
- **eeeprom** - folosită pentru a scrie/citi date în/din memoria *EEPROM* a microcontroller-ului.
- **lcd** - folosită pentru interacțiunea cu LCD 16x2.
- **twi_lib** - bibliotecă de **i2c** folosită în comunicația cu senzorul *BMP180*.

Pentru o **prezentare succintă** a modului în care funcționează proiectul, vezi secțiunea *Rezultate obținute*. Pentru o **prezentare** mai **amănunțită**, vezi **README** și alte coduri sursă din secțiunea *Download*.

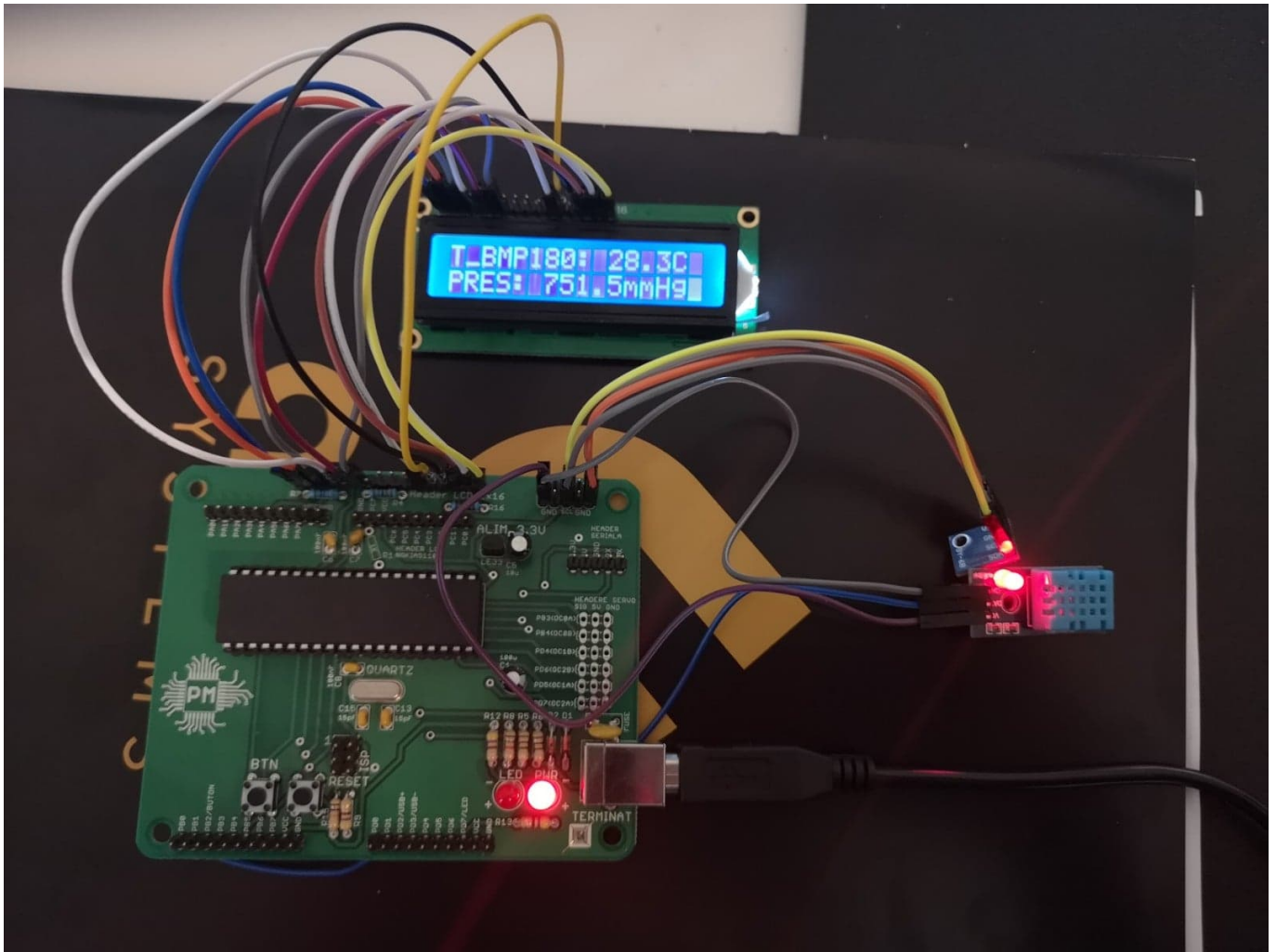
Rezultate obținute

Ca produs final, am obtinut un program pe placuta care afiseaza in continuu **ultimele 4 masuratori facute** si, o data la **30 secunde**, primeste o intrerupere si extrage date noi de la cei 2 senzori (**BMP180** si **DHT11**). De asemenea, inainte de a trece prin toate datele anterioare, face si o predictie comparand valoarea curenta a presiunii cu valoarea anterioara, conform algoritmului listat in sectiunea de *Bibliografie*. Informatiile sunt scrise/citite in/din memoria **EEPROM** a microcontroller-ului, fiind organizate logic sub forma unei liste circulare.

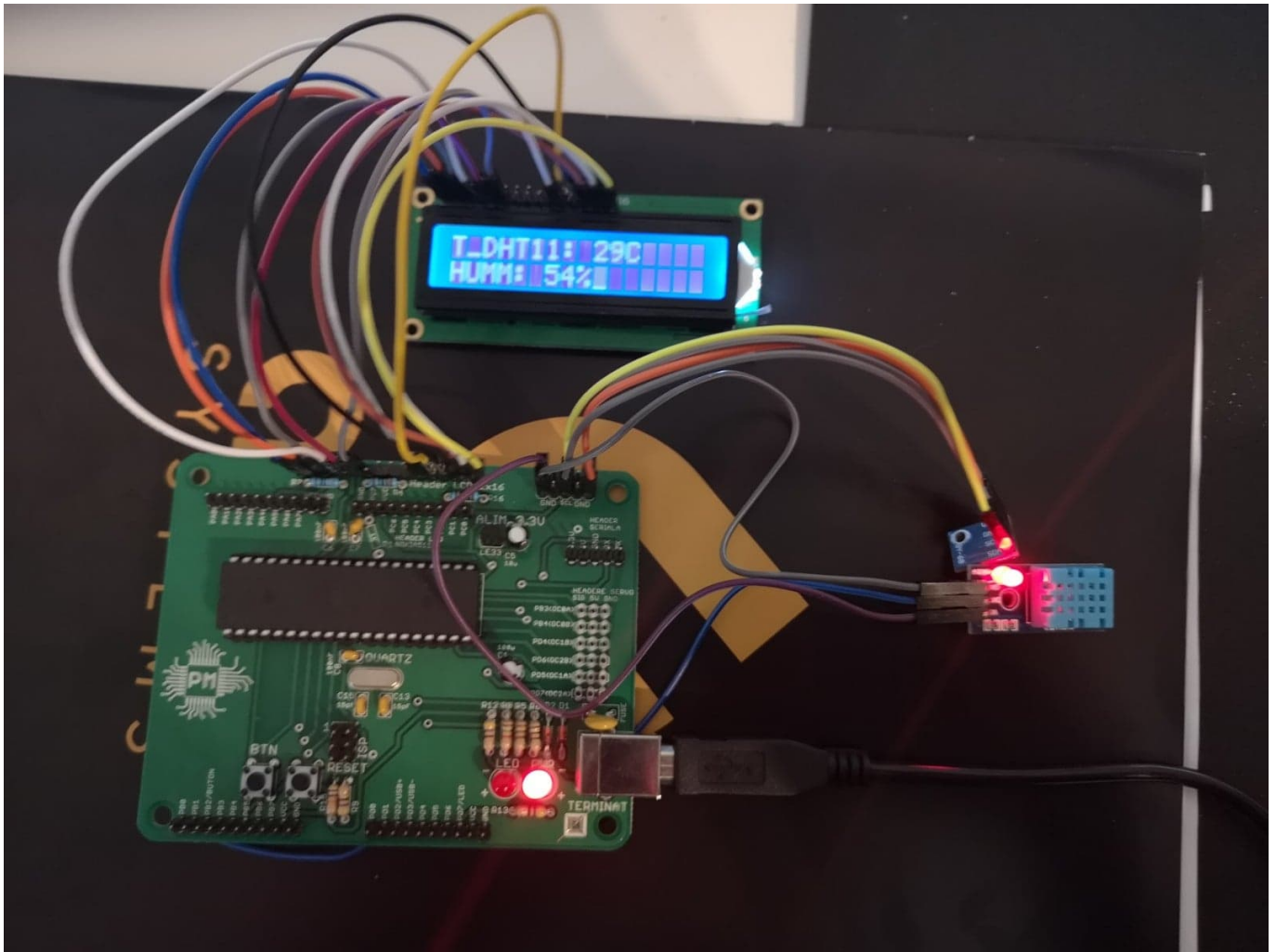
Primul mesaj afișat



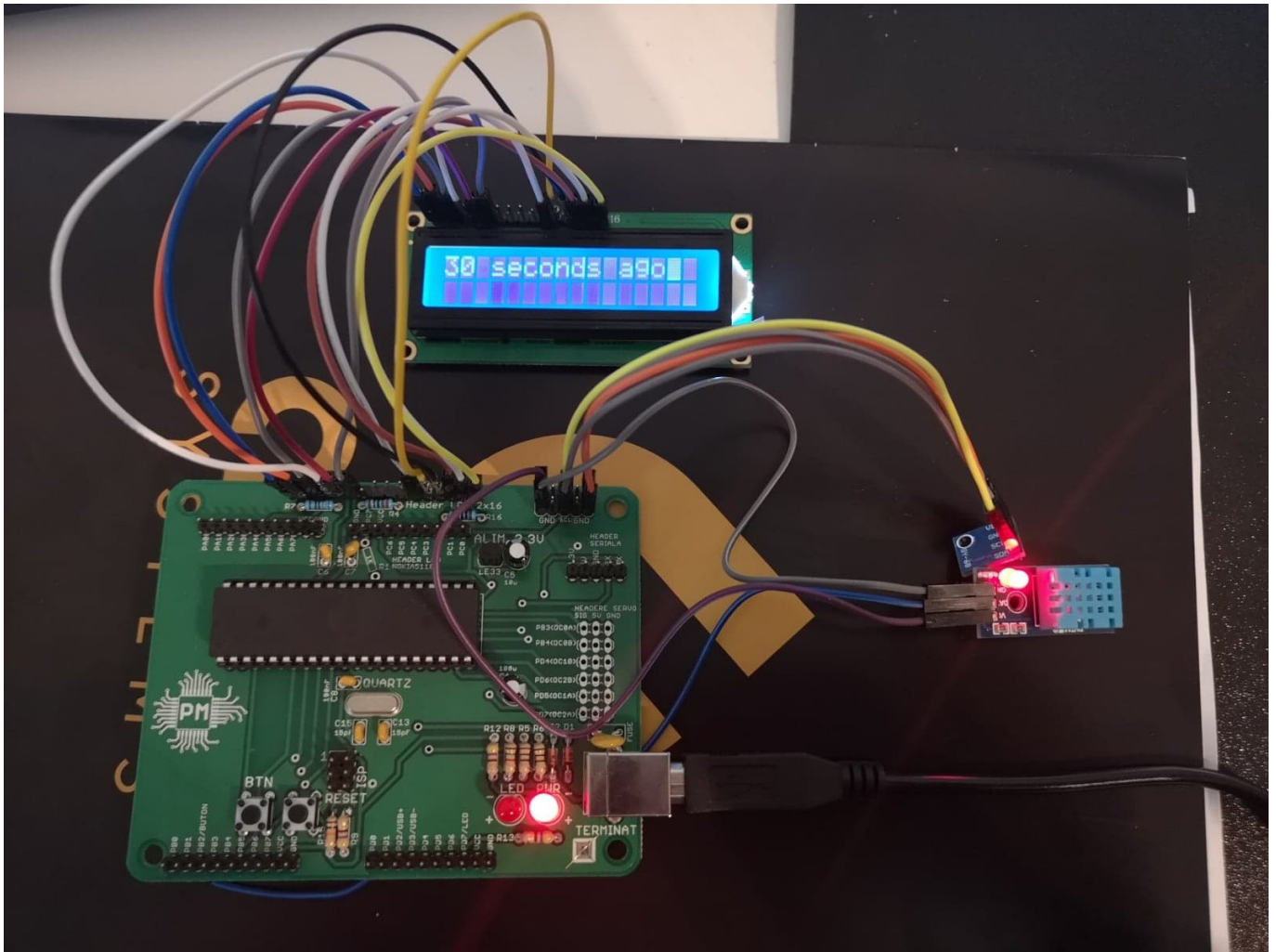
Afișare parametri senzor BMP180



Afișare parametri senzor DHT11



Afişare timp trecut de la datele afişate



Afișare predicție



Concluzii

Dezvoltarea proiectului a fost interesantă, dar ușor frustrantă din când în când, mai ales până să-mi dau seama cum funcționează comunicația cu senzorii. Per total a fost fain, iar costurile aproximative sunt undeva în jurul a **120 RON**.

Download

Conținut arhivă

Bibliotecile **atmega_bmp180_lib**, **dht**, **lcd**, **twi_lib**, fișier **README**, **Makefile** pentru compilare + încărcare pe plăcută, **bootloadHID** pentru încărcarea pe plăcută și codul sursă **prj.c** în care am dezvoltat logică programului.

[sendre_mihai_alin_332cc_pm2019_prj.zip](#)

Jurnal

Milestone 1

- 20.04.2019 - completare *date despre autor, Introducere și Descriere generală*. Secțiunile de *Design* sunt în faza incipientă, la fel și *Bibliografia*.

Milestone 3

- 07.05.2019 - adăugat *schema electrică* în secțiunea *Hardware Design*.
- 09.05.2019 - cumpărat piesele necesare pentru proiect.
- 17.05.2019 - realizat implementarea *Hardware* a proiectului.
- 22.05.2019 - am simplificat proiectul eliminând componente *Hardware* care nu erau neapărat necesare.
- 24.05.2019 - ultimul update al paginii.. pentru posteritate :).

Bibliografie/resurse

- [mihai.alin.sendre_pm2019.pdf](#)
- <https://www.thoughtco.com/how-to-read-a-barometer-3444043>
- <https://electrosome.com/interfacing-lcd-atmega32-microcontroller-atmel-studio/>
- https://www.microchip.com/webdoc/AVRLibcReferenceManual/group_avr_eeprom.html
- <http://www.avislab.com/blog/bmp180/>
- <http://davidegironi.blogspot.com/2012/12/reading-temperature-and-humidity-on-avr.html#.XOfu99fg5k>
- **Datasheet ATmega324A**

From:
<http://cs.curs.pub.ro/wiki/pm/> - **PM Wiki**

Permanent link:
<http://cs.curs.pub.ro/wiki/pm/prj2019/astratulat/mihai.alin.sendre>

Last update: **2019/05/24 15:02**

