

# Statie de monitorizare a temperaturii si umiditatii

## Autor

[Pavel Delia-Maria](#)

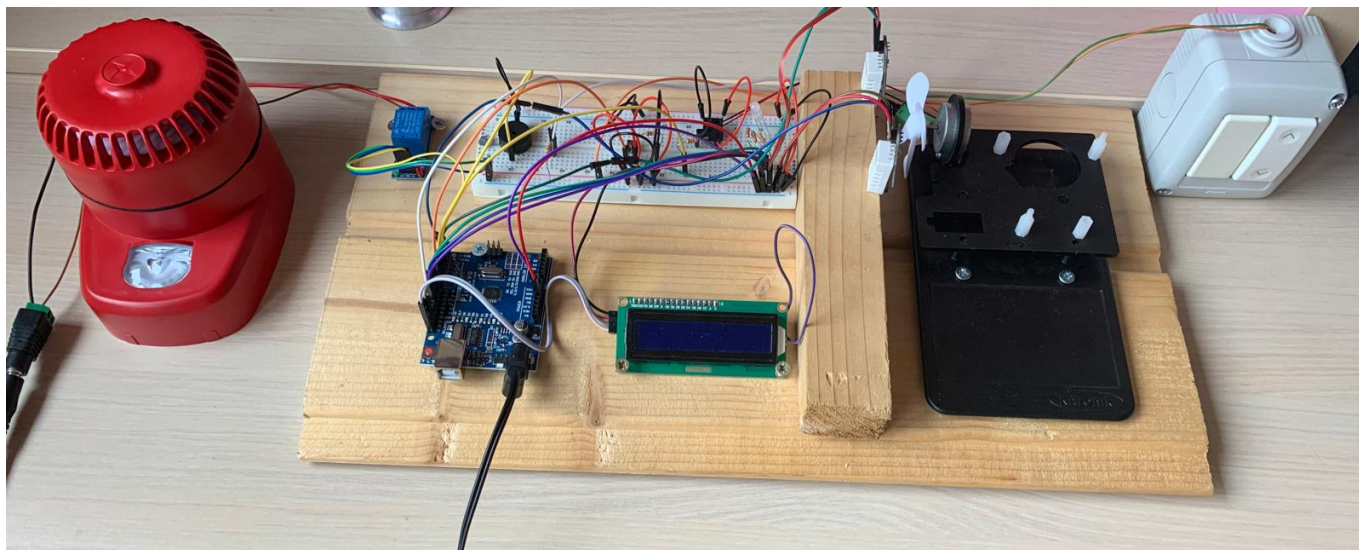
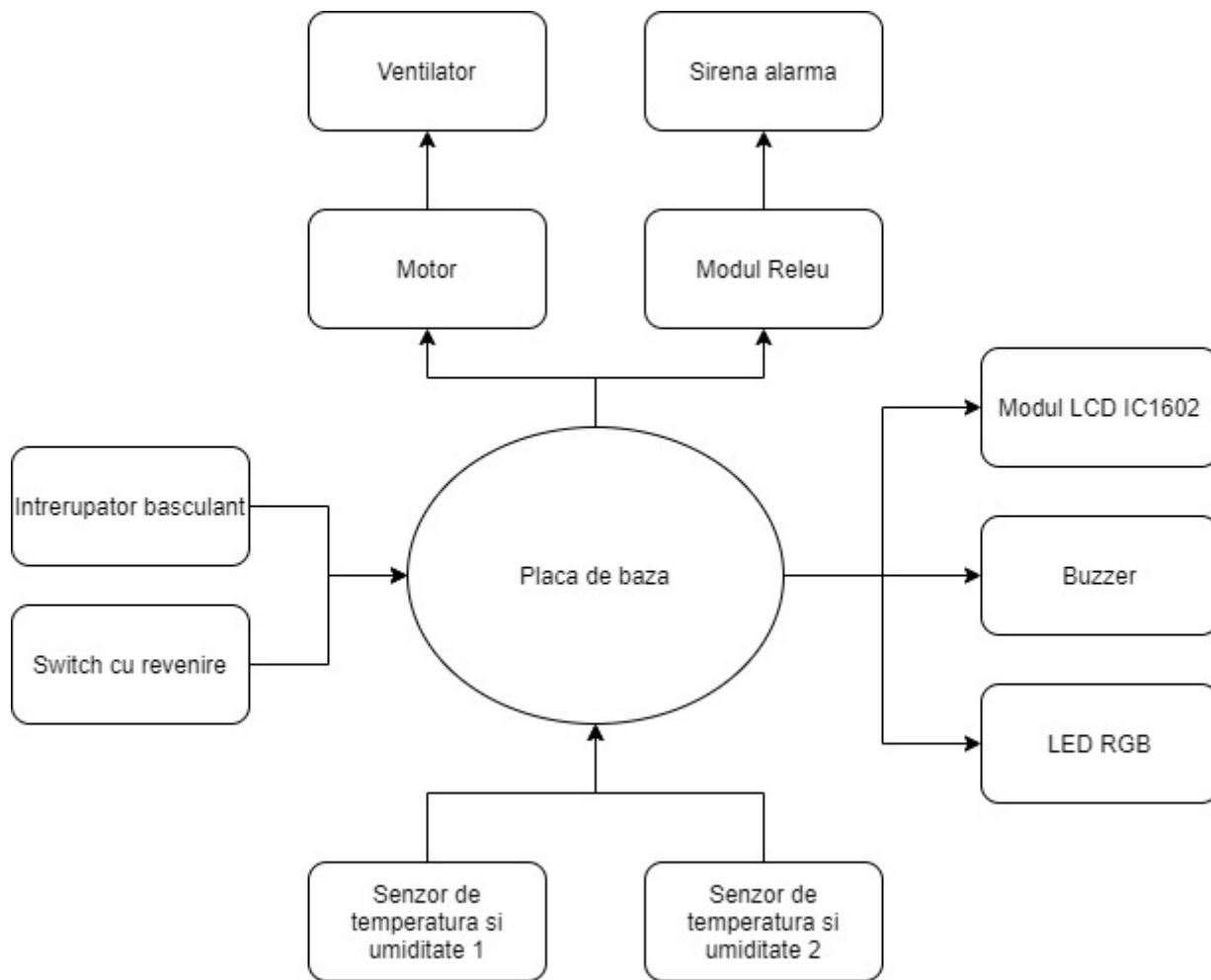
## Introducere

Proiectul implementeaza un sistem de monitorizare a temperaturii si umiditatii intr-o camera tehnica, afisand parametrii pe un ecran LCD si comandand, la depasirea unui prag de temperatura, un ventilator.

Utilitatea proiectului consta in gasirea unei aplicatii practice, permitand mentinerea unei temperaturi relativ constante intr-o camera de acest tip, in care supraincalzirea echipamentelor ar cauza multe probleme.

## Descriere generală

Senzorii de temperatura si umiditate vor citi incontinuu si vor trimite datele catre placa de baza. Datele vor fi afisate pe ecranul LCD si vor fi prelucrate. In functie de valorile date de senzor, se decide daca se va porni sau nu ventilatorul cu ajutorul modulului releu. De asemenea, modulul de alimentare ne va permite sa folosim statia de monitorizare fara ca aceasta sa fie conectata la calculator.



## Hardware Design

Lista de piese:



## Descrierea codului aplicației:

- mediu de dezvoltare: Arduino IDE
- librării și surse 3rd-party
  - pentru LCD: <LiquidCrystal\_I2C.h>; <LCD.h>; <Wire.h>
  - pentru senzor: <DHT.h>
- algoritmi și structuri de date:
  - logică simplă (if, while, for, etc.)
  - structuri de bază C: variabile, constante etc.
- surse și funcții implementate:
  - funcția setup(): initializeaza lcd-ul, seteaza ventilatorul ca output, initializam senzorii, setam releul ca output, setam led-ul rgb ca output, si cele doua butoane (cel de pornire a statiei de monitorizare si cel de afisare a parametrilor RealFeel) ca input
  - funcția loop(): citim starea de pe butonul de start. Daca acesta nu este apasat, opresc releul, ventilatorul, led-ul si afisez pe ecran detalii despre proiect. Daca este apasat, inseamna ca pot incepe sa monitorizez temperatura si umiditatea. Vedem daca butonul RealFeel este apasat sau nu. Daca este apasat, afisam pe lcd detalii despre RealFeel. Daca nu, afisam temperatura si umiditatea pentru cei doi senzori. Verificam daca temperaturile ambilor senzori au depasit pragul maxim. Daca da, sirena de alarma va suna, se va aprinde led-ul rosu si va porni ventilatorul. Daca doar temperatura unui singur senzor a depasit acest prag, va suna buzzerul, se va porni ventilatorul si se va porni led-ul pe culoarea magenta. Daca temperatura niciunui senzor nu a depasit acest prag, se va opri ventilatorul, releul, buzzerul si ledul va afisa culoarea verde.
  - functia get\_realfeel(float temp, float hum): calculeaza RealFeel pe baza unei formule gasite pe Wikipedia, folosind temperatura si umiditatea primite ca parametri
  - functia setColorRGB(unsigned int red, unsigned int green, unsigned int blue): seteaza pe pinii led-ul rgb valorile primite ca parametru

## Rezultate obtinute

- [Cod sursă](#)
- [Video Demo](#)

## Concluzii

Proiectul la PM a fost mai interesant de cat m-as fi asteptat initial. Am reusit sa ma familiarizez pe cat posibil cu anumite concepte ale lucrului cu Arduino, acest proiect starnindu-mi interesul de a mai lucra si pe viitor cu aceste concepte. Partea cea mai dificila la proiectul meu consider ca a fost reglarea temperaturii pentru a verifica functionalitatea atat a senzorilor, cat si a intregului proiect care se bazeaza pe schimbarea temperaturii.

## Bibliografie/Resurse

- [https://en.wikipedia.org/wiki/Heat\\_index](https://en.wikipedia.org/wiki/Heat_index)
- <https://github.com/fdebrabander/Arduino-LiquidCrystal-I2C-library>
- <https://github.com/adafruit/DHT-sensor-library>

[Export to PDF](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2021/amocanu/statie-monitorizare-temperatura>



Last update: **2021/05/30 15:04**