

# Statie Meteo

## Introducere

Proiectul propus este un sistem de monitorizare a mediului ambiental care utilizează mai multe senzori pentru a măsura și afișa temperatura, umiditatea, presiunea atmosferică și nivelul de radiație UV. Sistemul utilizează un microcontroler Arduino pentru a prelua datele de la senzori și un afișaj LCD pentru a le afișa în timp real.

Prin intermediul acestui proiect, utilizatorul poate obține informații esențiale despre condițiile de mediu din jurul său, ceea ce poate fi util în diverse aplicații, cum ar fi monitorizarea vremii, controlul climatului în interiorul unei încăperi sau monitorizarea nivelului de radiație UV pentru protecția sănătății.

Proiectul combină mai multe componente hardware și software pentru a crea un sistem funcțional și versatil de monitorizare a mediului.

## Descriere generală



Această schemă bloc arată modul în care aceste module interacționează între ele pentru a colecta date de la senzori și a le afișa pe un afișaj LCD.

### 1. Hardware:

- Arduino Microcontrolerul Arduino este componenta centrală a proiectului și este responsabil de preluarea datelor de la senzori și controlul afișajului LCD
- Senzorul KY-015: Acest senzor măsoară temperatura și furnizează datele către Arduino
- Senzorul DHT11: Acest senzor măsoară temperatura și umiditatea și furnizează datele către Arduino
- LCD (modul afișaj)
- Senzorul BMP180: Acest senzor măsoară temperatura și presiunea atmosferică și furnizează datele către Arduino
- Senzorul UV: Acest senzor măsoară nivelul de radiație UV și furnizează datele către Arduino
- Afișaj LCD: Afișajul LCD este utilizat pentru a afișa datele colectate de la senzori
- Baterie externa

### 2. Software:

- Bibliotecile: Proiectul utilizează diverse biblioteci precum Wire.h, LiquidCrystal\_I2C.h, dht.h,

Adafruit\_BMP085.h și UVSensor.h pentru a interacționa cu senzorii și afișajul LCD.

- Configurare: Software-ul include funcții de configurare a senzorilor și a afișajului LCD pentru a asigura funcționarea corectă a acestora.
- Prelevarea datelor: Arduino preia datele de la senzori prin intermediul interfeței hardware corespunzătoare și le procesează.
- Afișarea datelor: Arduino afișează datele colectate pe afișajul LCD într-un format ușor de citit.

Interacțiunea dintre modulele hardware și software este următoarea: Arduino inițializează și configurează senzorii și afișajul LCD. Apoi, Arduino preia datele de la senzori în mod regulat și le afișează pe afișajul LCD. Astfel, utilizatorul poate vedea în timp real informațiile despre temperatura, umiditatea, presiunea atmosferică și nivelul de radiație UV în mediul ambiental.

Această schemă bloc și descrierea asociată oferă o viziune generală asupra modului în care modulele hardware și software ale proiectului interacționează între ele pentru a crea un sistem de monitorizare a mediului ambiental funcțional și eficient.

## Hardware Design

Listă de piese:

- Senzor KY-015 (temperatură și umiditate)
- Senzor BMP180 (presiune atmosferică)
- Senzor UV 43 (senzor UV)
- LCD (modul afișaj)
- Arduino Uno (placă de dezvoltare)
- Breadboard (placă de conexiuni)
- Fire de conexiune
- Baterie externa

Schema electrica a circuitului:



Rezultate ale simulării: Simulare1: Linia 1:

- Temperatura: 25°C
- Umiditate: 47%
- Intensitate UV: 0UV

Linia 2:

- Presiune: 1005.21hPa



Simulare1: Linia 1:

- Temperatura: 26°C
- Umiditate: 48%
- Intensitate UV: 0UV

Linia 2:

- Presiune: 1008.74hPa



# Software Design

Mediul de dezvoltare:

Pentru acest proiect am folosit Arduino IDE pentru a programa pe Arduino UNO intrucat este o aplicatie ușor de utilizat și gratuită, care oferă o serie de funcționalități utile pentru dezvoltarea proiectelor pe platforma Arduino.

Librarii:

- Wire: Este o bibliotecă standard în Arduino IDE și este utilizată pentru comunicarea I2C. Este necesară pentru a comunica cu senzorul BMP180 și permite transferul datelor digitale între Arduino și senzor.
- LiquidCrystal\_I2C: Această bibliotecă permite controlul afișajului LCD cu interfață I2C. Este utilizată pentru a afișa valorile temperaturii, umidității, presiunii atmosferice și nivelului de radiație UV pe afișajul LCD.
- dht: Această bibliotecă este folosită pentru a citi datele de la senzorul DHT11. Ea oferă funcții pentru a citi temperatura și umiditatea ambientală și este utilizată în cod pentru a obține valorile temperaturii și umidității.
- Adafruit\_BMP085: Această bibliotecă este dezvoltată de Adafruit și este utilizată pentru a interacționa cu senzorul BMP180. Ea oferă funcții pentru a citi presiunea atmosferică și temperatura de la senzorul BMP180.

Listarea algoritmului:

```
#include <Wire.h> #include <LiquidCrystal_I2C.h> #include <dht.h> #include <Adafruit_BMP085.h>
```

```
#define dht_apin A0 #define UV_SENSOR_PIN A1
```

```
LiquidCrystal_I2C lcd(0x3F, 16, 2); Adafruit_BMP085 bmp; dht DHT;
```

```
void setup() {
```

```
  lcd.init();  
  lcd.clear();  
  lcd.backlight();  
  bmp.begin();
```

```
}
```

```
void loop() {
```

```
  DHT.read11(dht_apin);  
  
  float temperature = DHT.temperature;  
  float humidity = DHT.humidity;
```

```
float pressure = bmp.readPressure() / 100.0;
int uvValue = analogRead(UV_SENSOR_PIN);

lcd.clear();

lcd.setCursor(0, 0);
lcd.print("T:");
lcd.print((int)temperature);
lcd.print("C");

lcd.setCursor(6, 0);
lcd.print("H:");
lcd.print((int)humidity);
lcd.print("%");

lcd.setCursor(0, 1);
lcd.print("P:");
lcd.print(pressure);
lcd.print("hPa");

lcd.setCursor(12, 0);
lcd.print("UV:");
lcd.print(uvValue);

delay(5000);

}
```

Detaliere a algoritmului:

Initializarea:

- Se includ bibliotecile necesare pentru comunicația prin I2C, citirea senzorilor DHT11 și BMP180
- Se definește pinul analogic pentru senzorul UV
- Se creează obiectele pentru afișajul LCD și senzorii DHT11 și BMP180.
- În funcția setup(), se inițializează afișajul LCD, senzorul BMP180 și se curăță ecranul

Citirea datelor:

- În bucla principală loop(), se citesc valorile de temperatură și umiditate utilizând funcția DHT.read11().
- Se citește presiunea atmosferică utilizând funcția bmp.readPressure(), iar apoi rezultatul este împărțit la 100.0 pentru a obține valoarea în hPa.
- Se citește nivelul de radiație UV utilizând funcția analogRead() pentru pinul senzorului UV.
- Valorile citite pentru temperatură, umiditate, presiune și nivelul de radiație UV sunt stocate în variabile corespunzătoare.

Afișarea datelor pe LCD:

- Se curăță ecranul LCD utilizând funcția lcd.clear().
- Se setează cursorul LCD pentru afișarea valorilor citite.
- Utilizând funcția lcd.print(), se afișează valorile citite pentru temperatură, umiditate, presiune și

nivelul de radiație UV pe LCD.

Pauză și repetare:

- Se folosește funcția delay() pentru a realiza o pauză de 5 secunde între actualizările afișajului.
- Algoritmul se repetă în mod continuu, actualizând valorile și afișându-le pe LCD.

## Rezultate Obținute



1. Afișarea temperaturii: Pe afișajul LCD, veți putea vedea temperatura ambientală în grade Celsius. Valorile ar trebui să fie precise și să se actualizeze la intervale de 5 secunde.
2. Afișarea umidității: Pe afișajul LCD, veți putea vedea nivelul de umiditate ambientală exprimat în procente (%). Valorile ar trebui să fie precise și să se actualizeze împreună cu temperatura.
3. Afișarea presiunii atmosferice: Pe afișajul LCD, veți putea vedea presiunea atmosferică exprimată în hectopascale (hPa). Valorile ar trebui să fie precise și să se actualizeze împreună cu temperatura și umiditatea.
4. Afișarea nivelului de radiație UV: Pe afișajul LCD, veți putea vedea nivelul de radiație UV citit de senzorul UV. Valorile ar trebui să reflecte nivelul de radiație UV la care senzorul este expus. Este important să rețineți că pentru o măsurare corectă a radiației UV, senzorul trebuie să fie expus la sursa de radiație UV.

## Concluzii

În cadrul proiectului, am reușit să dezvoltăm un sistem de monitorizare a temperaturii, umidității, presiunii atmosferice și radiației UV. Am integrat cu succes mai multe componente hardware și am implementat un cod software adecvat pentru a obține măsurători precise și pentru a afișa aceste informații pe un afișaj LCD.

Sistemul dezvoltat oferă o monitorizare în timp real a parametrilor climatici și a nivelului de radiație UV. Utilizatorul poate vizualiza aceste informații pe afișajul LCD, ceea ce îi permite să obțină o perspectivă asupra condițiilor de mediu. Aceasta poate fi utilă în diverse domenii, cum ar fi agricultura, monitorizarea mediului sau aplicații de siguranță.

Proiectul are o valoare practică semnificativă în monitorizarea și înțelegerea condițiilor de mediu. Permite utilizatorilor să obțină date relevante despre temperatură, umiditate, presiune atmosferică și nivelul de radiație UV, ajutându-i să ia decizii informate în diverse domenii de activitate.

Posibile îmbunătățiri:

- Calibrarea și ajustarea senzorilor pentru a obține măsurători mai precise.
- Implementarea unei funcționalități de înregistrare a datelor pentru a putea analiza și urmări evoluția parametrilor de mediu pe o perioadă mai lungă de timp.
- Extinderea sistemului prin adăugarea de funcționalități suplimentare, cum ar fi conectivitatea la rețea sau integrarea cu un sistem de control automat.

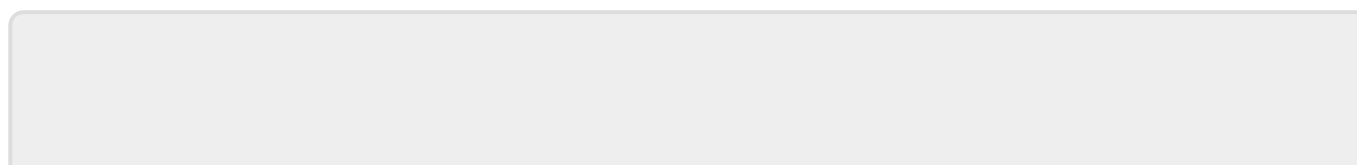
## Download

[arhivaa.rar](#)

## Bibliografie/Resurse

Listă cu documente, datasheet-uri, resurse Internet folosite, eventual grupate pe **Resurse Software** și **Resurse Hardware**.

[Export to PDF](#)



From:

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

Permanent link:

[http://ocw.cs.pub.ro/courses/pm/prj2023/razvans/statie\\_meteo](http://ocw.cs.pub.ro/courses/pm/prj2023/razvans/statie_meteo)



Last update: **2023/05/30 12:08**