

Sistem de monitorizare a calității aerului

Introducere

Sistemul își propune să urmărească o serie de parametri care influențează calitatea aerului din mediul ambiant și tot odată confortul unei persoane aflate într-o încăpere. Este vorba despre temperatura, umiditatea și presiunea aerului, dar și măsurând calitatea aerului pe baza acestor parametri.

Prezentarea proiect:

- dispozitivul măsoară presiunea, temperatura și nivelul umidității din aer
- acest sistem are rolul de a verifica calitatea aerului din atmosferă
- având în vedere nivelul ridicat de poluare care există, un astfel de dispozitiv este important în vederea sănătății oamenilor
- un astfel de dispozitiv este necesar pentru a asigura respirarea unui aer cât mai calitativ

Descriere generală

O schemă bloc cu toate modulele proiectului vostru, atât software cât și hardware însoțită de o descriere a acestora precum și a modului în care interacționează.

Exemplu de schemă bloc: <http://www.robs-projects.com/mp3proj/newplayer.html>

Hardware Design

Componente:

- placă Arduino Uno
- breadboard
- buzzer activ
- cablu UART
- ecran LCD 1602 cu Interfață I2C
- cabluri conductoare
- senzor pentru umiditate și temperatură DHT22
- senzor pentru presiune BMP280



Software Design

În cadrul acestui proiect, am folosit Arduino IDE ca mediu de dezvoltare, iar ca și biblioteci pentru Arduino am utilizat:

- **Wire** pentru comunicația cu ecranul LCD;
- **Adafruit_BMP280** pentru senzorul BMP280;
- **DHT** pentru senzorul DHT22;
- **LiquidCrystal_I2C** pentru afișarea datelor pe ecran.

Diagrama de stări pentru sistem:



Această diagramă ilustrează funcționarea sistemului de monitorizare a calității aerului utilizând o mașină de stări. Fiecare stare și tranzițiile dintre ele sunt reprezentate grafic:

1. INIT (Inițializare):

- Inițializarea componentelor hardware (senzorii DHT22 și BMP280, ecranul LCD și buzzer-ul);
- Tranziția către starea READ_SENSORS.

2. READ_SENSORS (Citirea senzorilor):

- Citirea datelor de la senzorii DHT22 (temperatură și umiditate) și BMP280 (presiune);
- Tranziția către starea DISPLAY_DATA.

3. DISPLAY_DATA (Afișarea datelor):

- Afișarea datelor citite pe ecranul LCD;
- Tranziția către starea SEND_UART.

4. SEND_UART (Trimiterea datelor prin UART):

- Trimiterea datelor citite de la senzori prin UART către un dispozitiv extern;
- Tranziția către starea CHECK_AIR_QUALITY.

5. CHECK_AIR_QUALITY (Verificarea calității aerului):

- Calcularea unui indice simplu al calității aerului pe baza datelor citite;
- Activarea buzzer-ului dacă indicele de calitate a aerului depășește un prag prestabilit;
- Tranziția înapoi către starea READ_SENSORS după un interval de timp, reluând ciclul.

Această structură permite o gestionare clară și organizată a funcționalităților sistemului.



Rezultate Obținute

În urma realizării proiectului, am obținut informații cu privire la calitatea aerului din atmosferă (umiditate, temperatură și presiune).

Concluzii

Download

O arhivă (sau mai multe dacă este cazul) cu fișierele obținute în urma realizării proiectului: surse, scheme, etc. Un fișier README, un ChangeLog, un script de compilare și copiere automată pe uC crează întotdeauna o impresie bună 😊.

Fișierele se încarcă pe wiki folosind facilitatea **Add Images or other files**. Namespace-ul în care se încarcă fișierele este de tipul **:pm:prj20??:c?** sau **:pm:prj20??:c?:nume_student** (dacă este cazul).
Exemplu: Dumitru Alin, 331CC → **:pm:prj2009:cc:dumitru_alin**.

Jurnal

Puteți avea și o secțiune de jurnal în care să poată urmări asistentul de proiect progresul proiectului.

Bibliografie/Resurse

- <https://howtomechatronics.com/tutorials/arduino/dht11-dht22-sensors-temperature-and-humidity-tutorial-using-arduino/>
- <https://www.instructables.com/How-to-Use-the-Adafruit-BMP280-Sensor-Arduino-Tuto/>

[Export to PDF](#)

From:

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

Permanent link:

http://ocw.cs.pub.ro/courses/pm/prj2024/avaduva/nelu_rares.carausu



Last update: **2024/05/23 22:55**