Home Monitor Environment - RESULI Rrapo

Introducere

Proiectul Home Monitor Environment este un sistem conceput pentru a monitoriza temperatura, umiditatea și calitatea aerului într-o încăpere. Folosește senzori pentru a colecta date și un afișaj LCD pentru a prezenta aceste informații utilizatorului. Sistemul poate fi controlat de la distanță folosind o telecomandă IR, oferind o modalitate simplă și eficientă de a menține un mediu sănătos.

Scopurile Proiectului: - Monitorizarea continuă a parametrilor de mediu - Afișarea în timp real a datelor pe un ecran LCD - Control de la distanță folosind o telecomandă IR

Ideea: - Inspirat de nevoia de a monitoriza condițiile de mediu pentru a asigura un climat optim pentru sănătatea umană și protecția dispozitivelor electronice.

Utilitate: - Util pentru cei care doresc să îmbunătățească calitatea aerului în locuințe sau birouri. - Oferă o bază solidă pentru proiecte IoT și sisteme de automatizare a locuinței.

Descriere Generală



Proiectul este un sistem de monitorizare a mediului interior care:

- măsoară temperatura, umiditatea și calitatea aerului
- procesează valorile pe un Arduino Nano
- afișează datele pe un ecran LCD I2C 16×2
- permite controlul prin intermediul unei telecomenzi IR
- (opțional) stochează datele în EEPROM sau le trimite prin Serial Monitor

1. Module Hardware

- Data Acquisition
 - Senzor DHT22 măsoară temperatura și umiditatea relativă
 - Senzor MQ135 detectează niveluri de CO2 și compuși volatili
- Processing (Arduino Nano)
 - Unitate principală care colectează și prelucrează datele de la senzori
- User Interface
 - LCD Display I2C afișează valorile curente și mesaje
 - Telecomandă IR controlează modurile de afișare sau declanșează acțiuni
- Storage (opțional)
 - EEPROM intern pentru salvarea ultimelor valori sau praguri

2. Module Software (pe Arduino Nano)

- Sensor Drivers
 - Biblioteci: `DHT`, `IRremote`, `LiquidCrystal_I2C`
- Data Processing Algorithm
 - Citirea senzorilor la intervale regulate
 - Compararea valorilor cu praguri și calcularea mediilor
- Display Driver
 - Scriere pe ecranul LCD cu informații actualizate
- IR Remote Handler
 - Interpretarea comenzilor de la telecomandă pentru schimbarea modului de afișare
- EEPROM Manager (opțional)
 - Salvarea și citirea datelor persistente între porniri
- Serial Communication Module
 - $_{\circ}\,$ Trimiterea datelor brute către PC pentru debugging sau logare

Hardware Design

Conexiuni electrice

Alimentare generală

- Arduino Nano 5 V \rightarrow şina + a breadboard-ului
- Arduino Nano GND \rightarrow şina a breadboard-ului

MQ135

- VCC → 5 V
- GND \rightarrow GND
- AO \rightarrow Arduino AO

DHT22 (temp + umiditate)

- VCC → 5 V
- GND \rightarrow GND
- DATA \rightarrow Arduino D2

TSOP1838

- VCC → 5 V
- GND \rightarrow GND
- DAT → Arduino D11

LED-uri simple (roşu / verde / albastru)

- Arduino D4/ D5/ D6- rezistor 220 Ω anod LED roşu/ verde/ albastru
- catod LED roşu/ verde/ albastru \rightarrow GND

LCD 16×2 cu interfață I²C (adresă 0x27)

- VCC \rightarrow 5 V
- GND \rightarrow GND
- SDA → Arduino A4
- SCL → Arduino A5

Schema Electrica

×

Software Design

Mediu de dezvoltare: Arduino IDE

Librării folosite:

- `Wire.h` pentru comunicarea I2C cu LCD-ul
- `LiquidCrystal_I2C.h` controlul afişajului LCD 16×2 prin I2C
- `DHT.h` pentru senzorul DHT22 (temperatură și umiditate)
- `MQ135.h` pentru senzorul de calitate a aerului (gaz)
- `IRremote.hpp` pentru interpretarea comenzilor de la telecomanda IR

Moduri de funcționare (comutabile cu telecomanda IR):

• mode_0 (Temperatură):

Se citește temperatura ambientală cu DHT22.

```
Pe LCD apare mesajul:
`Temp: xx.x°C`
Este trimis și în `Serial Monitor`. LED-ul albastru (pin D5) se aprinde.
```

• mode_1 (Umiditate):

Se citește umiditatea relativă (%) cu DHT22.

```
Pe LCD apare mesajul:
`Humidity: xx.x%`
Este trimisă și în `Serial Monitor`. LED-ul verde (pin D6) se aprinde.
```

mode_2 (Calitate aer):

Se citește valoarea de la senzorul MQ135 și se estimează nivelul de ppm.

Pe LCD apare mesajul:

```
`Air Quality: xxx ppm`
Este trimis și în `Serial Monitor`. LED-ul roșu (pin D4) se aprinde.
```

Control prin telecomandă IR:

- Comenzile de pe telecomandă sunt mapate la:
 - Buton `1`: schimbă în modul Temperatură
 - Buton `2`: schimbă în modul Umiditate
 - Buton `3`: schimbă în modul Calitate Aer
- Codurile sunt recepționate prin pinul digital D11, folosind `IRremote.hpp`.

Funcționalități suplimentare:

- Afișaj LCD I2C cu iluminare, actualizat la fiecare secundă
- LED-uri de status pe pini dedicați:
 - D5 albastru (Temperatură)
 - D6 verde (Umiditate)
 - D4 roșu (Calitate aer)
- Serial Monitor activ pentru debugging sau vizualizare valori în timp real

Sistemul este simplu, intuitiv și oferă o bază solidă pentru extindere ulterioară cu module precum EEPROM, RTC sau SD-card dacă este necesar.

Rezultate Obținute

GitHub Repository: https://github.com/resulirrapo/Home_Monitor_Environment/tree/master



×

7/7

Concluzii

Download

O arhivă (sau mai multe arhive dacă este cazul) cu fișierele obținute din proiect: surse, scheme etc. Un fișier README, un ChangeLog și un script pentru compilare și încărcare automată pe microcontroller creează întotdeauna o impresie bună ⁽²⁾.

Fișierele ar trebui încărcate pe wiki folosind facilitatea **Add Images or other files**. Namespace-ul pentru încărcarea fișierelor ar trebui să fie de forma **:pm:prj20??:c?** sau **:pm:prj20??:c?:nume_student** (dacă este cazul). **Exemplu:** Ion Popescu, 331CC → **:pm:prj2009:cc:ion_popescu**.

Jurnal

Puteți avea și o secțiune de jurnal în care asistentul de proiect poate urmări progresul proiectului vostru.

Bibliografie/Resurse

Listă cu documente, fișe tehnice și resurse de pe internet utilizate, eventual grupate în **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/prj2025/fstancu/rrapo.resuli

Last update: 2025/05/28 08:47