

Sistem Smart Home pentru climatizare

Student: Grosu Cristina-Geanina

Grupa: 331CC

Introducere

Descrierea Proiectului

Proiectul propune dezvoltarea unui sistem Smart-Home avansat, capabil să controleze automat și manual sistemele de încălzire și răcire (centrala termică și aerul condiționat) în funcție de anumiți parametri predefiniți. Sistemul poate funcționa în două moduri: automat și manual.

Mod de Funcționare

Mod Automat:

În modul automat, sistemul utilizează o cartelă RFID pentru a detecta prezența utilizatorilor. Când utilizatorul părăsește locuința și scanează cartela la ușă, sistemul dezactivează centrala termică sau aerul condiționat pentru a evita consumul inutil de energie în absența ocupanților. La întoarcerea utilizatorului, scanarea cartelei reactivează sistemul de încălzire sau răcire, în funcție de anotimp, și ajustează temperatura interioară la valoarea preferată de utilizator. Sistemul rămâne activ până când temperatura ambientală atinge nivelul dorit.

Mod Manual:

În modul manual, utilizatorii pot seta manual temperatura dorită pentru încăpere, independent de preferințele predefinite și de temperatura exterioară. Acest mod permite ajustări rapide și personalizate ale temperaturii interioare, fără a ține cont de setările automate.

Motivația Proiectului

Inspirația pentru acest proiect a apărut dintr-o nevoie personală de a reduce consumul inutil de energie și de a evita facturile mari la utilități. După multiple ocazii în care am uitat să opresc centrala termică sau aerul condiționat când am părăsit locuința, am realizat necesitatea unui sistem inteligent care să gestioneze automat aceste aparate. Astfel, acest sistem își propune să elimine grija utilizatorilor legată de oprirea echipamentelor de încălzire/răcire la plecarea din locuință.

Descriere generală

Acest proiect dezvoltă un sistem Smart-Home pentru controlul automat și manual al sistemelor de încălzire și răcire, utilizând trei plăcuțe Raspberry Pi Pico W. Una dintre acestea funcționează ca

unitate master, gestionând datele despre utilizatorii care intră și ies din locuință prin intermediul unui modul RFID RC522. Masterul controlează preferințele de temperatură ale utilizatorilor și decide activarea sau dezactivarea centralei termice și a aerului condiționat în funcție de anotimp și temperatura setată. De asemenea, masterul dispune de un buton pentru comutarea între modulele automat și manual.

Celelalte două plăcuțe Raspberry Pi Pico W controlează centrala termică și aerul condiționat, primind instrucțiuni wireless de la unitatea master. Plăcuța responsabilă de centrala termică are un potențiomtru care permite utilizatorului să seteze manual temperatura dorită în modul manual. Plăcuța responsabilă de aerul condiționat are un potențiomtru pentru setarea temperaturii curente în încăpere. Ambele dispozitive ajustează funcționarea până când temperatura dorită este atinsă și indică starea de funcționare (working/idle) prin intermediul a două LED-uri.

Astfel, proiectul oferă o soluție eficientă și convenabilă pentru gestionarea confortului termic în locuință, economisind energie și simplificând controlul sistemelor de încălzire și răcire.

Diagrama bloc



Hardware Design

Lista piese

- Raspberry Pi Pico W (Master)
- Raspberry Pi Pico W (Aer Conditionat)
- Raspberry Pi Pico W (Centrala)
- Modul RFID RC522
- Potentiometru pentru setat temperatura ambientala curenta
- Potentiometru pentru setat temperatura dorita in modul manual
- Buton pentru comutat intre modul manual si automat
- Led-uri pentru semnalizarea starii centralei/aerului conditionat (working/idle)

Schema electrica



Descrierea sistemului

Acest proiect dezvoltă un sistem Smart-Home destinat controlului automat și manual al sistemelor de încălzire și răcire, utilizând trei plăcuțe Raspberry Pi Pico W și diverse alte componente pentru asigurarea funcționalității dorite.

Componentele Sistemului

Raspberry Pi Pico W (Master):

- Funcție: Primește date despre utilizatori (intrare/ieșire) și gestionează preferințele de temperatură.
- Echipamente conectate:
 - Modul RFID RC522 pentru citirea cartelelor RFID.

- Buton pentru comutarea între modul automat și manual (gestionat prin întreruperi).
- Comunicare wireless cu celelalte două plăcuțe Raspberry Pi Pico W (Centrala și Aerul Condiționat).

Raspberry Pi Pico W (Centrala):

- Funcție: Controlează centrala termică.
- Indicatori: 2 LED-uri (Starea de funcționare: working/idle).
- Potențiomtru pentru setarea temperaturii dorite in modul manual.
- Receptează date wireless de la Master și funcționează până când temperatura atinge valoarea preferată.

Raspberry Pi Pico W (Aer Condiționat):

- Funcție: Controlează aerul condiționat.
- Indicatori: 2 LED-uri (Starea de funcționare: working/idle).
- Potențiomtru pentru setarea temperaturii curente în încăpere.
- Receptează date wireless de la Master și funcționează până când temperatura atinge valoarea preferată.

Modul RFID RC522:

- Funcție: Citirea cartelelor RFID pentru detectarea utilizatorilor care intră și ies din locuință.
- Conectat direct la Raspberry Pi Pico W (Master) prin SPI.

Potențiomtru la Raspberry Pi Pico W (centrala):

- Funcție: Setarea temperaturii dorite în modul manual.

Potențiomtru la Raspberry Pi Pico W (aer conditionat):

- Funcție: Setarea temperaturii resimtite in incapere.

Funcționarea Sistemului

Mod Automat

Când un utilizator părăsește locuința și scanează cartela RFID, sistemul dezactivează centrala termică sau aerul condiționat pentru a economisi energie. La întoarcere, scanarea cartelei reactivează sistemul de control al temperaturii în funcție de anotimp și preferințele utilizatorului, ajustând temperatura ambientală până la atingerea valorii dorite.

Mod Manual

În modul manual, utilizatorul poate seta manual temperatura dorită în încăpere folosind un potențiomtru, ignorând setările automate.

Software Design

Mediu de Dezvoltare

Pentru dezvoltarea codului aplicației a fost utilizat Thonny IDE, un mediu de dezvoltare integrat popular pentru programarea în Python, dedicat în special pentru microcontrolere.

Biblioteci folosite

În cadrul acestui proiect, au fost utilizate următoarele biblioteci externe:

- [umqtt](#): O librărie simplificată pentru protocolul MQTT, utilizată pentru comunicarea cu broker-ul MQTT Mosquitto.
- [MFRC522](#): O librărie pentru interfațarea cu modulul RFID RC522, utilizată pentru citirea cardurilor RFID.
- [picozero](#): O librărie ușor de folosit pentru controlul componentelor GPIO.

Algoritmi și Structuri Implementate

1. [master_board.py](#)

Codul aplicației este structurat pentru a îndeplini următoarele funcționalități principale:

Conectarea la rețeaua Wi-Fi: Realizată prin intermediul funcției `connect_to_wifi()`, care utilizează configurația specificată într-un fișier JSON.

Comunicarea MQTT: Configurarea și gestionarea clientului MQTT pentru a primi și trimite mesaje către și de la celelalte plăcuțe Raspberry Pi Pico W (centrala și aerul condiționat).

Determinarea anotimpului curent: Implementată prin funcția `get_current_season()`, care utilizează data curentă pentru a stabili dacă este iarnă sau vară.

Gestionarea utilizatorilor: Funcții pentru încărcarea, salvarea, adăugarea și ștergerea utilizatorilor și preferințelor lor de temperatură.

Citirea cardurilor RFID: Funcția `scan_rfid()` care utilizează biblioteca MFRC522 pentru a citi și identifica utilizatorii.

Gestionarea temperaturii: Funcția `check_temperature()` pentru a verifica și ajusta temperatura ambientală în funcție de preferințele utilizatorilor sau setările manuale.

Gestionarea modurilor de operare: Funcția `button_handler()` care permite comutarea între modurile automat și manual prin intermediul unui buton.

2. [air_conditioner_board.py](#)

Codul aplicației este structurat pentru a îndeplini următoarele funcționalități principale:

Conectarea la rețeaua Wi-Fi: Realizată prin intermediul funcției `connect_to_wifi()`, care utilizează configurația specificată într-un fișier JSON.

Comunicarea MQTT: Configurarea și gestionarea clientului MQTT pentru a primi și trimite mesaje către și de la plăcuța `master_board`.

Citirea și filtrarea valorii de la potențiometrul: Implementată prin funcțiile `read_potentiometer()` și `map_value()` pentru a citi și scala valoarea de la potențiometrul în intervalul de temperaturi 15°C - 40°C.

Controlul LED-urilor: Funcții pentru aprinderea și stingerea LED-urilor în funcție de comenzile primite pentru a indica starea de funcționare (cooling/idle).

Transmiterea temperaturii curente: Funcția principală care trimite periodic temperatura curentă citită de la potențiomtru către plăcuța Master prin MQTT.

3. [heating_system_board.py](#)

Codul aplicației este structurat pentru a îndeplini următoarele funcționalități principale:

Conectarea la rețeaua Wi-Fi: Realizată prin intermediul funcției `connect_to_wifi()`, care utilizează configurația specificată într-un fișier JSON.

Comunicarea MQTT: Configurarea și gestionarea clientului MQTT pentru a primi și trimite mesaje către și de la plăcuța Raspberry Pi Pico W Master.

Citirea valorii de la potențiomtru: Implementată prin funcția `read_potentiometer()` pentru a citi și scala valoarea de la potențiomtru în intervalul de temperaturi 15°C - 40°C.

Controlul LED-urilor: Funcții pentru aprinderea și stingerea LED-urilor în funcție de comenzile primite pentru a indica starea de funcționare (heating/idle).

Transmiterea temperaturii manuale: Funcția `publish_manual_temperature()` trimite temperatura setată manual de la potențiomtru către plăcuța Master prin MQTT.

Rezultate Obținute

Proiectul a demonstrat implementarea unui sistem Smart-Home integrat, constând din trei plăcuțe Raspberry Pi Pico W conectate wireless, care comunică eficient între ele utilizând protocolul MQTT prin broker-ul Mosquitto. Am reușit să realizez o soluție funcțională pentru controlul automat și manual al sistemelor de încălzire și răcire, asigurând ajustarea temperaturii în locuință în funcție de prezența utilizatorilor și preferințele acestora. Sistemul s-a dovedit responsiv, reușind să gestioneze comenzile și să transmită datele necesare în timp real.

Concluzii

Proiectul a realizat cu succes integrarea unui sistem Smart-Home pentru controlul automat și manual al sistemelor de încălzire și răcire, demonstrând fiabilitatea comunicării wireless între cele trei plăcuțe Raspberry Pi Pico W. Utilizând protocolul MQTT prin broker-ul Mosquitto, sistemul a dovedit capacitatea de a gestiona eficient controlul temperaturii în funcție de prezența și preferințele utilizatorilor.

Download

[grosu_cristina_final_smart_home_system.zip](#)

Jurnal

7.03.2024: Comandat piese.

29.04.2024: Lipit pini headers pentru componente.

4.05.2024: Scris documentatie wiki pentru partea de hardware.

17.05.2024: Conectat toate componentele corespunzator si implementat logica in cod pentru citire card, scanare si adaugare useri intr-un JSON file, controlat potentiometru pentru temperatura, comutat prin buton cu intreruperi intre mod automat si mod manual.

18.05.2024: Adus documentatia la un nivel mai satisfacator.

24.05.2024: Implementat protocolul mosquito pentru comunicatia wireless.

26.05.2024: Finalizat documentatia si codul cu toate functionalitatile necesare.

Bibliografie/Resurse

Resurse

[1] <https://diyprojectslab.com/rfid-door-lock-with-raspberry-pi-pico>

[2] <https://randomnerdtutorials.com/raspberry-pi-pico-analog-inputs-micropython/>

[3] <https://www.raspberrypi.com/documentation/microcontrollers/images/picow-pinout.svg> ← most important

[4] <https://mosquitto.org/>

[5] https://www.youtube.com/watch?v=ybCMXqsQyDw&ab_channel=CoreElectronics

[6] https://www.youtube.com/watch?v=jw9zTjKqoUA&ab_channel=Shilleh

[7] <https://randomnerdtutorials.com/raspberry-pi-pico-w-mqtt-micropython/>

[Export to PDF](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2024/vstoica/cristina.grosu>



Last update: **2024/05/27 11:35**