TermoSmart Scheduler

Introducere

Proiectul vizeaza dezvoltarea unui termostat inteligent care monitorizeaza continuu temperatura ambientala si, in functie de necesar, porneste un ventilator compact (care imita aerul conditionat) pentru racire sau activeaza rezistoare de incalzire (care imita caloriferul), mentinand astfel permanent un confort termic optim in incapere si optimizand consumul de energie.

Ce face: Masoara temperatura ambientala si controleaza automat centrala si aerul conditionat.

Care este scopul lui: Ofera economii de energie si confort optim prin pornirea/opirea HVAC conform programarilor.

Care a fost ideea de la care ati pornit: Am plecat de la ideea de a simplifica controlul temperaturii, eliminând nevoia de a jongla între termostat și telecomanda aerului condiționat. Un singur dispozitiv integrat permite menținerea usoara a temperaturii ideale și optimizeaza consumul de energie.

De ce credeti ca este util pentru altii si pentru voi: Simplifica administrarea climatizarii in locuinte si reduce costurile cu incalzirea si racirea.

Descriere generala

Functionalitati: Incalzire si racire automate – Porneste incalzirea sau racirea pana la atingerea setpoint-ului definit sau pana la oprire manuala.

Mentinere temperatura constanta – Odata atins pragul setat se comuta in modul „mentinere” pentru a pastra temperatura dorita pana la primirea altei comenzi.

Alarma de blocaj – Semnaleaza daca sistemul functioneaza prea mult timp exclusiv in modul incalzire sau racire. Se semnaleaza printr-un semafor in functie de nivelul de blocaj si un buzzer.

Interfata utilizator intuitiva – Meniu cu butoane dedicate („incalzire”, „racire”, „mentinere”) si afisaj digital al deciziei curente.

Laboratoare folosite

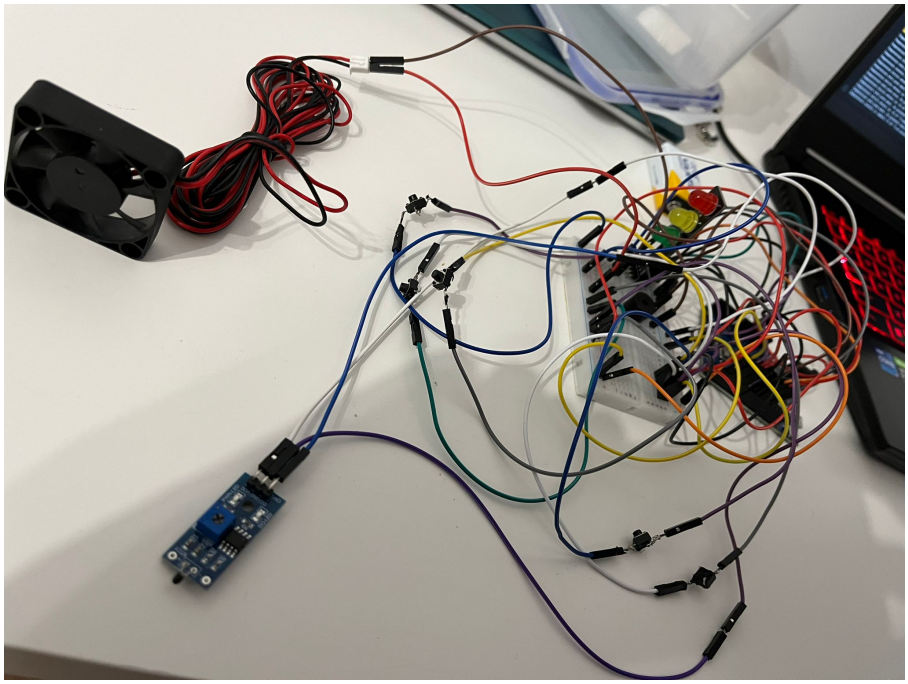
Laboratorul 0 (GPIO) – Citirea semnalelor de la butoanele de selectare a modului (incalzire, racire, mentinere) si controlul semaforului cu LED-uri (verde, galben, rosu) prin software, pe baza pragurilor de temperatura.

Laboratorul 1 (UART) – Interfata seriala cu PC-ul pentru configurarea pragurilor de temperatura si afisarea in timp real a starii sistemului.

Laboratorul 3 (Timere & PWM) – Generarea intreruperilor periodice (ex. la fiecare 1 s) necesare esantionarii senzorului de temperatura si modularea vitezei ventilatorului prin semnal PWM.

Laboratorul 4 (ADC) – Citirea semnalului analogic de la senzorul NTC 10 kΩ si conversia lui in valori digitale de temperatura.

Hardware Design



BOM (Bill Of Materials)

Arduino Uno R3 : Creierul proiectului; alimentare 9V extern - Link

Semafor LED (Roșu/Galben/Verde) : Semnalizare stare/eroare; D9–D11 - Link

Buzzer Activ : Semnalizare stare/eroare; - Link

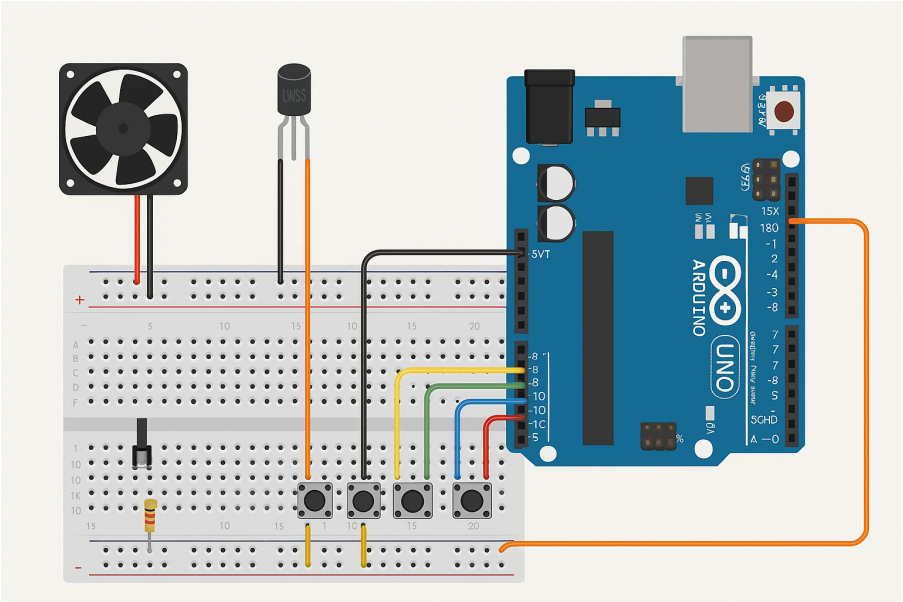
Senzor temperatura : Senzorul de detectare al temperaturii; - Link

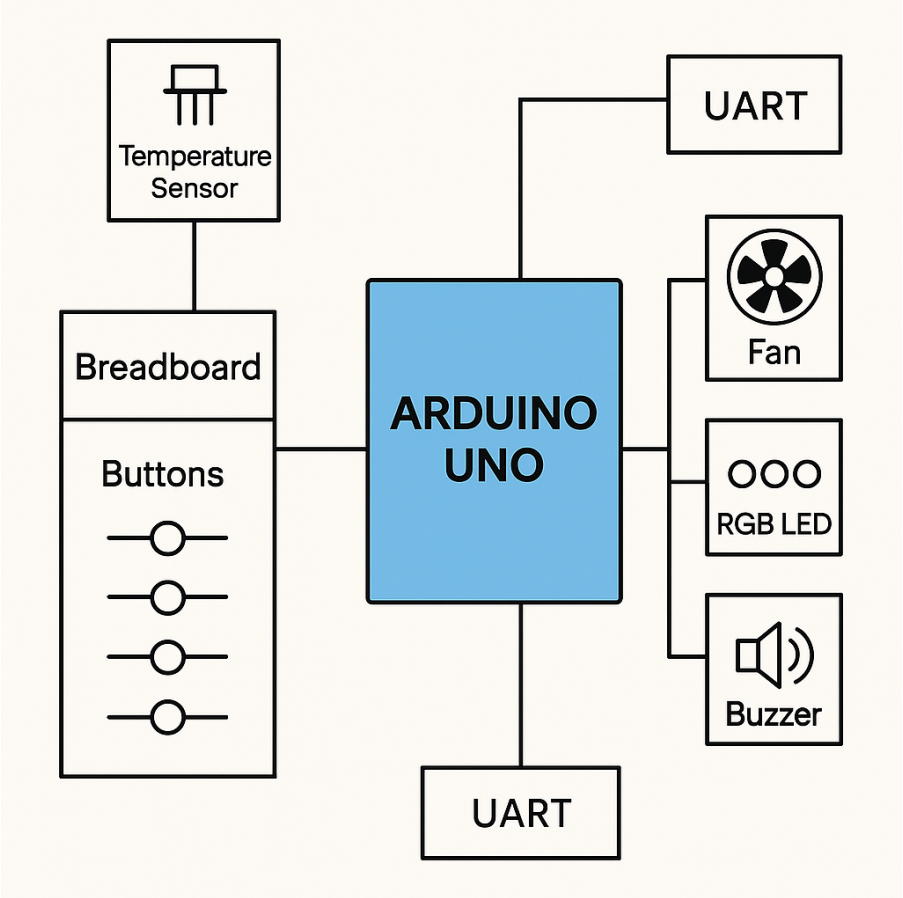
Ventilator : Pentru a raci senzorul de temperatura, imita aerul conditionat; - Link

Butoane : Cate un buton pentru fiecare mod (incalzire/racire/mentinere) si 2 butoane pentru a creste sau a scadea in mentinere; - Link

Breadbord : Montaj prototip; alimentare rails și grupuri de câte 5 orizontal; - Link

Adaptor DC 9 V : Alimentare externă pentru Arduino (în jack); - Link





Software Design

Descrierea codului :

* **Mediu de dezvoltare:** Arduino IDE

* **Librarii si surse 3rd-party:**

- Functii native Arduino (`analogRead()`, `digitalWrite()`, `pinMode()`, `tone()`, `noTone()`, `Serial`, `millis()`, `delay()`)

* **Acces GPIO prin registri (Laboratorul 3 – Intreruperi / GPIO avansat):**

- Configurare intrari cu pull-up pentru butoane pe PD2...PD6 folosind `DDRD` si `PORTD`
- Citire stare butoane direct din `PIND` cu masca `_BV(PDx)`

* **Algoritmi si structuri planificate:**

- Citire analogica de la senzorul de temperatura si conversie ADC→°C (Laboratorul 4 – ADC)
- Gestionare moduri de functionare (Incalzire, Racire) si pornire semafor (verde/galben/rosu) (Laboratorul 0 – GPIO de baza)
- Reglare manuala temperatura dorita in modul Mentenanta (Laboratorul 0 – GPIO de baza)
- Semaforizare stari sistem:

- verde 0–10s (LED_GREEN + buzzer 500 Hz)
- galben 10–20s (LED_YELLOW + buzzer 1000 Hz)
- rosu ≥20s (LED_RED + buzzer 1500 Hz)
- Control ventilator si incalzire (`digitalWrite(FAN_PIN)`) dupa temperatura curenta vs. dorita
- Feedback UART pentru stare curenta, mesaje de actiune si erori (Laboratorul 1 – UART)

* **Surse si functii implementate:**

- `setup()`

- Configurare butoane PD2–PD6 ca intrari cu pull-up prin `DDRD`/`PORTD` (Lab 3)
- `pinMode()` pentru LED-uri, buzzer si ventilator
- `Serial.begin(9600)` (Lab 1)
- `loop()`

- `readTemperature()` – citire ADC + conversie in °C (Lab 4)
- Detectie butoane `BUTTON1`/`BUTTON2`/`BUTTON3` prin citire din `PIND` (Lab 3)
- `semaforActive`, `startTime = millis()` – pornire semafor (Lab 0)
- Reglare mentenanta: modificare `temperaturaDorita`, mesaj UART, control ventilator (Lab 0)
- `updateSemaphore()` (in-line) – gestionare LED-uri si buzzer cu `digitalWrite()` si `tone()` (Lab 0)
- `afiseazaReglaj()` – afisare pe UART a temperaturii dorite vs. curente si control ventilator (Lab 1)

> **Notă:** Registrii sunt folosiți exclusiv pentru detectia rapidă a stării butoanelor și pentru a ilustra tehnici de întreruperi/GPIO avansat (Laboratorul 3). Restul funcționalităților sunt implementate cu API-ul Arduino pentru claritate și portabilitate.

Rezultate Obtinuite

In urma realizarii proiectului, software-ul implementat functioneaza conform specificatiilor si indeplineste obiectivele stabilite. Cu toate acestea, am intampinat dificultati minore in gestionarea ventilatorului, cauzate de limitarile hardware, ceea ce a determinat ca ventilatorul sa nu functioneze intotdeauna optim.

Concluzii

* Proiectul s-a dovedit provocator din punct de vedere hardware si mediu din punct de vedere software. Cu toate acestea, am reusit sa finalizam ambele componente cu succes. Am dedicat aproximativ 45 de ore acestui proiect si nu am intampinat probleme majore, cum ar fi defectarea componentelor hardware. In concluzie, am obtinut un proiect functional la standardele dorite.

Download

Proiectul este incarcat pe GitHub.

Jurnal

Jurnalul orientativ al proiectului.

* 05/05/2025 – Am ales tema

* 07/05/2025 – Am terminat prima parte din documentatie

* 13/05/2025 – Am achizitionat piese

* 15/05/2025 – Am ars un semafor; am comandat altul

* 20/05/2025 – Aproape am terminat partea Hard

* 21/05/2025 – Am terminat partea Soft

* 22/05/2025 – Am terminat partea Hard

* 23/05/2025 – Am updatat partea Soft

* 28/05/2025 – Am terminat documentatia

Bibliografie/Resurse

Resurse Software

- Lab0

- Lab1

- Lab2

- Lab3

- Lab4

- ChatGPT

Resurse Hardware

- Transistor

- Arduino

- ChatGPT

Export to PDF