

Pergolă Automată

Introducere

Proiectul constă în realizarea unei pergole retractabile automatizate, controlată de un microcontroler Arduino Uno. Aceasta își poate extinde sau retrage acoperișul în funcție de condițiile meteorologice detectate de senzori. Astfel, starea pergolei este determinată de valorile înregistrate de senzori.

Scopul proiectului este de a crea un sistem autonom și eficient pentru protecția și confortul utilizatorilor în spațiile exterioare, cum ar fi terase, grădini sau foișoare. Prin automatizarea pergolei, se elimină nevoia de intervenție umană în funcționarea acesteia, crescând astfel gradul de confort și adaptabilitate la schimbările de vreme.

Ideea a pornit de la dorința de a automatiza un element frecvent întâlnit în zonele rezidențiale și comerciale, pentru a-l face mai inteligent și mai reactiv la mediul înconjurător. Am dorit să aduc un plus de funcționalitate unui element pasiv des întâlnit într-o grădină sau terasă.

Consider că proiectul este util deoarece poate fi aplicat în viața reală, oferind un plus de confort și protecție utilizatorilor. Reprezintă totodată și o idee de afacere ce duce spre zona de automatizări ce influențează modul de trai al utilizatorilor, putând fi integrat ulterior cu sisteme SmartHome.

Descriere generală

Proiectul este structurat în jurul unui microcontroler **Arduino Uno**, care gestionează toate modulele hardware și logica software a pergolei retractabile. Sistemul este compus din următoarele module:

- **Senzor de ploaie** – detectează prezența precipitațiilor. Dacă este detectată ploaie, transmite semnalul către Arduino pentru a închide pergola.
- **Senzor de temperatură și umiditate (DHT11)** – oferă date despre mediul înconjurător. În funcție de valorile citite, sistemul decide dacă pergola trebuie deschisă sau închisă (de exemplu, temperatură foarte ridicată și umiditate scăzută → deschidere pentru ventilație).
- **Ecran LCD 16x2 cu I2C** – afișează informații relevante pentru utilizator: starea curentă a pergolei (Deschisă / Închisă), temperatura și umiditatea.
- **Driver de motor L298N** – controlează direcția de rotație și alimentarea motoarelor, în funcție de comenzile primite de la Arduino.
- **Două motoare de 12V** – acționează mecanismul de extindere/retragere al pergolei.
- **Led verde, led roșu** - fiecare led se aprinde în funcție de starea pergolei: verde când pergola este deschisă, roșu când este închisă.
- **Alimentare** – sursa de curent care furnizează tensiunea necesară pentru Arduino și motoare.

Schema bloc

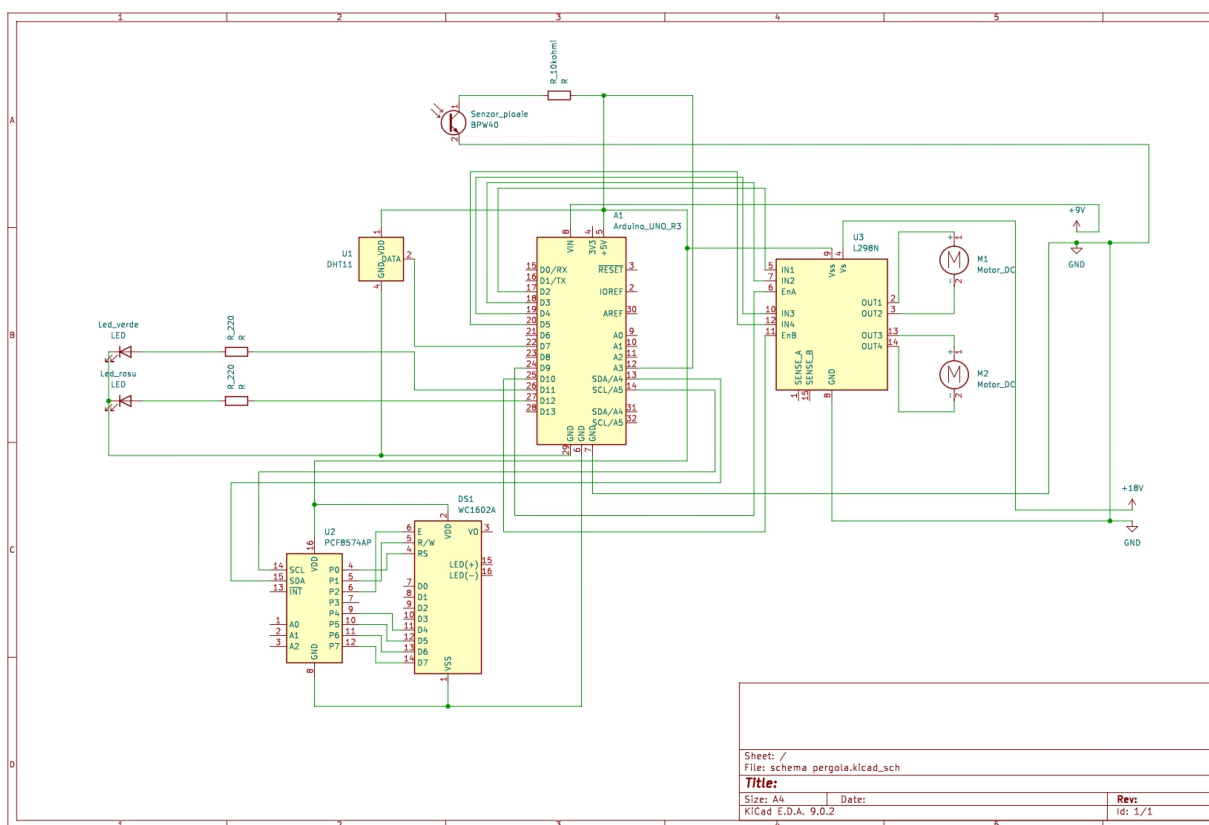
Toate aceste module comunică și interacționează prin intermediul Arduino-ului, care rulează un program ce interpretează datele de la senzori, ia decizii și controlează motoarele. În același timp, informațiile utile sunt afișate pe ecranul LCD.

Hardware Design

Componente hardware utilizate

Componentă	Funcție	Detalii tehnice
Arduino UNO	Unitate de control principală	14 pini digitali, 6 pini analogici, 5V
Senzor de ploaie	Detectează prezența apei pe suprafață	leșire analogică, conectat la A3
Senzor DHT11	Măsoară temperatura și umiditatea	Temperatură (0-50°C), umiditate (20-90%), pin digital 7
Modul L298N	Control motor bidirecțional	2 canale H-Bridge, alimentare 18V
Motoare DC x2	Extind și retrag pergola	Alimentate la 18V prin driver L298N
LCD 16×2	Afișează starea pergolei și valorile meteo	I2C, adresă 0x27, conectat la A4/A5
LED verde	Indică pergolă retrasă	Conectat la pin digital 11, cu rezistență 220Ω
LED roșu	Indică pergolă extinsă	Conectat la pin digital 12, cu rezistență 220Ω
Surse baterii 9V	Alimentare separată pentru Arduino și motoare	1x 9V pentru Arduino (jack), 2x 9V în serie pentru motoare

Schema electrică



Explicații

Componentă	Pini folosiți	Descriere comportament
Arduino UNO	A3, D7, A4, A5, D2-D5, D9-D12	Primește date de la senzori, controlează motoarele, LCD-ul și LED-urile
Driver motoare L293D	IN1-IN4 (D2-D5), ENA/ENB (D9/D10), VS, VSS, GND	Controlează sensul și viteza motoarelor, primește semnale de la Arduino
Motoare DC x2	OUT1-OUT4 (driver)	Extind și retrag fizic pergola, comandate de L293D
Senzor de ploaie	A3 (semnal analogic)	Trimite un semnal variabil în funcție de cantitatea de apă detectată
Senzor DHT11	D7	Trimite temperatura și umiditatea către Arduino
LCD 16x2 cu I2C	A4 (SDA), A5 (SCL), 5V, GND	Afișează alternativ starea pergolei și valorile senzorilor
Modul I2C (PCF8574)	SDA, SCL, P0-P7 → LCD	Controlează LCD-ul cu doar 2 fire, mapat intern la RS, E, D4-D7
LED roșu	D12 (prin rezistor)	Se aprinde când pergola este extinsă
LED verde	D11 (prin rezistor)	Se aprinde când pergola este retrasă
Baterii 9V (x3)	VIN (Arduino), VS (driver), GND	1 baterie → Arduino, 2 în serie → alimentare motoare; GND comun

Diagramă de semnal



Explicații diagramă de semnal - Extindere automată pergolă

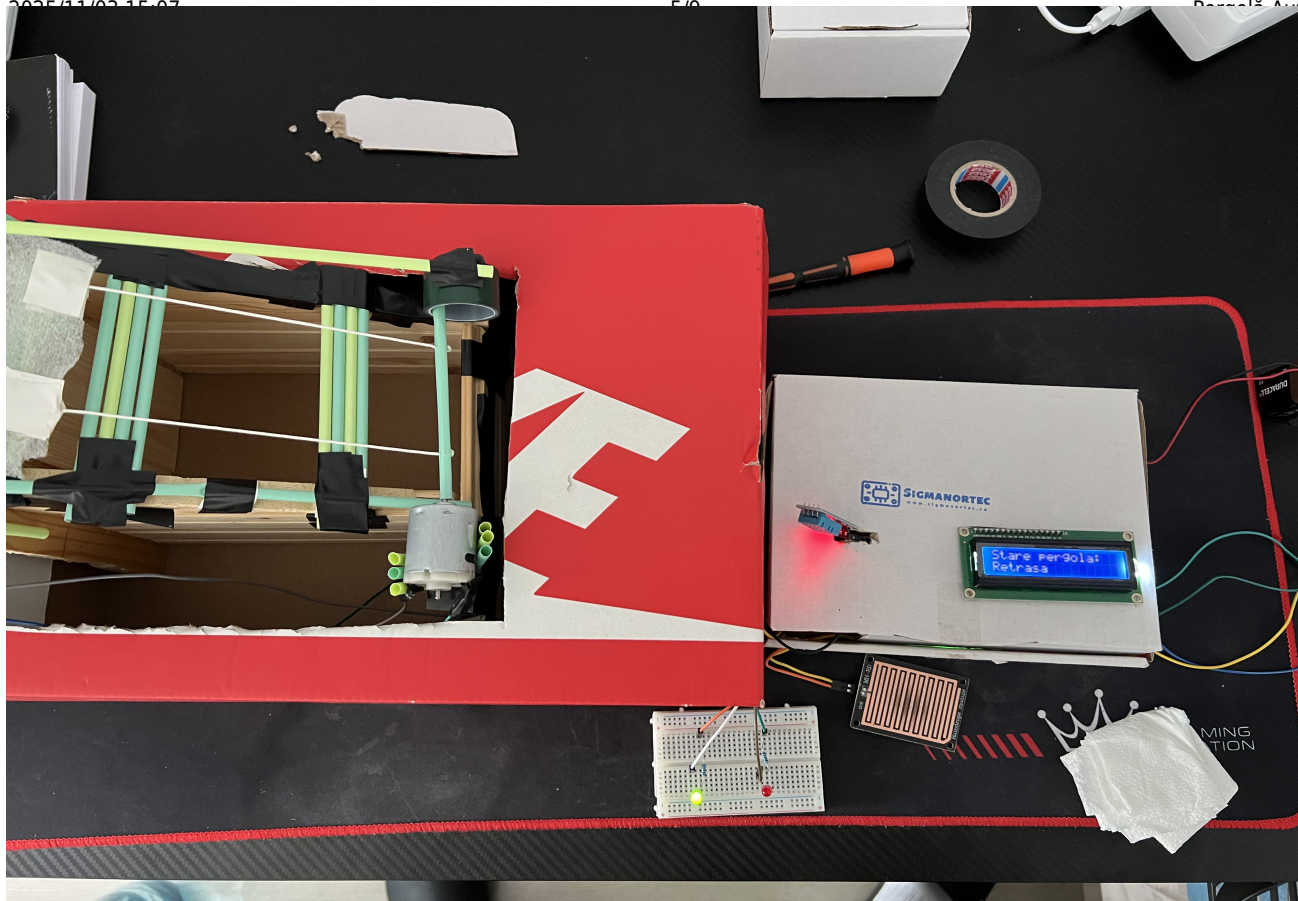
Diagrama de semnal evidențiază comportamentul sistemului în momentul în care sunt întrunite

- 'RainSensor_A3' - semnal analogic de la senzorul de ploaie. Când începe să plouă, nivelul tensiunii scade sub pragul de detecție, indicând prezența apei.
- 'DHT11_Trigger' - semnal digital generat în urma citirii temperaturii și umidității. Acesta devine activ (HIGH) dacă umiditatea depășește 60% sau temperatura scade sub 15°C.
- 'Motor_OUT (PWM)' - semnal de control trimis către driverul de motoare. Devine activ pentru o durată scurtă, corespunzătoare timpului de extindere a pergolei.
- 'LED_Rosu (D12)' - semnal care indică starea "pergola extinsă". Se activează după ce extinderea s-a finalizat.
- 'LED_Verde (D11)' - semnal care indică starea "pergola retrasă". Este activ înainte de extindere și se dezactivează odată cu începerea procesului.

Comportamentul general reflectă o reacție automată a sistemului la condiții meteo nefavorabile, cu o secvență logică bine determinată: detecție → comandă motoare → actualizare vizuală a stării prin LED-uri. Sistemul rămâne în starea extinsă atâta timp cât persistă condițiile de ploaie sau umiditate ridicată.

Stadiul actual al proiectului





Software Design

Descrierea codului aplicației

Firmware-ul controlează în totalitate comportamentul automat al pergolei, în funcție de valorile returnate de senzori (ploaie, temperatură, umiditate). Codul este structurat modular și este executat pe un microcontroler 'ATmega328P' de pe placa 'Arduino UNO'.

Aplicația face citiri constante de la senzori, procesează valorile înregistrate și declanșează acțiuni în funcție de condițiile definite (extindere/retragere pergolă), oferind și feedback vizual printr-un LCD 16x2 și două LED-uri de stare.

Mediu de dezvoltare utilizat

Dezvoltarea a fost realizată folosind:

- 'Arduino IDE' - pentru scrierea, compilarea și încărcarea codului pe placă
- 'Tinkercad Circuits' - pentru simularea inițială a montajului
- 'KiCad 7' - pentru proiectarea schemei electrice
- 'WaveDrom' - pentru generarea diagramei de semnal

Librării și surse 3rd-party utilizate

Au fost utilizate următoarele librării open-source:

- 'LCD_I2C.h' - pentru afișarea pe LCD 16x2 prin I2C
- 'Wire.h' - pentru comunicația I2C
- 'DHT.h' - pentru senzorul DHT11 (temperatură și umiditate)

Toate librăriile sunt instalabile direct din Library Manager în Arduino IDE.

Motivația alegerii bibliotecilor

Bibliotecile au fost alese pentru simplitatea integrării cu platforma Arduino și suportul extins oferit în comunitate. Acestea oferă interfețe standardizate și permit un timp redus de dezvoltare, permițând focusarea pe logica aplicației și nu pe detalii de nivel inferior.

Element de noutate al proiectului

Spre deosebire de o simplă acționare pe bază de ploaie, proiectul combină mai mulți parametri meteo (temperatură, umiditate, ploaie) pentru a lua decizii inteligente, asemănător unui sistem de predicție meteo simplificat. Integrarea cu feedback vizual și animație pe LCD, împreună cu semnalizare LED, oferă un sistem intuitiv și robust.

Justificarea utilizării funcționalităților din laborator

Următoarele concepte învățate în laborator au fost aplicate:

- 'GPIO' - pentru controlul LED-urilor
- 'PWM' - pentru acționarea precisă a motoarelor DC
- 'ADC' - pentru citirea valorii analogice de la senzorul de ploaie
- 'I2C' - pentru comunicația eficientă cu modulul LCD
- 'Temporizare' - pentru afișare alternativă și evitarea 'delay()'

Explicația scheletului și interacțiunii

Structura proiectului este organizată în jurul unei bucle infinite

```
loop()
```

care:

1. Citește senzorii
2. Evaluează condițiile de extindere/retragere
3. Actualizează LCD-ul și LED-urile
4. Acționează motoarele prin funcții dedicate

Toate componentele sunt sincronizate logic, iar testarea a fost realizată incremental, pornind de la simulări în Tinkercad și verificări în consolă (Serial Monitor) pentru validarea fiecărei funcționalități.

Calibrarea elementelor de senzorică

Senzorul de ploaie a fost testat empiric, prin observarea valorilor analogice returnate în stare uscată și umedă. Pragul de 700 a fost ales în urma testelor. Pentru DHT11, valorile de temperatură și umiditate au fost verificate comparativ cu un dispozitiv extern, iar pragurile de decizie au fost ajustate în cod pentru a reflecta condițiile reale.

Optimizări realizate

- S-a eliminat utilizarea excesivă a `delay()` , folosind în schimb `millis()` pentru afişare ciclică.
- Comenzile către motoare sunt executate doar dacă starea sistemului s-a schimbat, evitând comenzi redundante.
- A fost adăugată o funcție `animateLCD()` pentru a face tranziția de stare mai lizibilă pentru utilizator.

Demo video

[Link către demo YouTube](#)

Videoclipul de mai sus prezintă funcționarea completă a sistemului de pergolă automatizată, evidențiind declanșarea extinderii sau retragerii în funcție de condițiile meteo detectate de senzori, afișajul dinamic pe LCD și semnalizarea vizuală cu LED-uri.

Rezultate Obținute

Proiectul a atins cu succes toate obiectivele propuse inițial, fiind realizat un sistem complet funcțional de pergolă automatizată, capabil să reacționeze inteligent la condițiile meteorologice detectate în timp real. În urma implementării, au fost obținute următoarele rezultate concrete:

- Pergola se extinde automat în cazul detectării ploii, a umidității ridicate sau a temperaturilor scăzute, pe baza datelor de la senzori;
- Retragera se realizează automat în absența acestor condiții, menținând astfel un spațiu deschis optim în condiții favorabile;
- Afișajul LCD 16×2 comută în mod ciclic între starea pergolei și valorile actuale de temperatură și umiditate, oferind feedback clar utilizatorului;
- LED-urile roșu și verde semnalizează vizual în timp real dacă pergola este extinsă sau retrasă;
- Sistemul a fost validat prin testare fizică și simulare, comportându-se stabil și previzibil;
- Codul firmware este modular, eficient și permite extinderi ulterioare (ex: integrare cu aplicații mobile sau senzori suplimentari).

Proiectul demonstrează integrarea eficientă a mai multor funcționalități studiate în laborator, într-un sistem embedded cu aplicabilitate practică reală.

Concluzii

Proiectul realizat demonstrează posibilitatea construirii unui sistem automatizat de control al unei pergole, folosind concepte fundamentale din domeniul microcontrolerelor și al sensoristicii. Prin integrarea mai multor funcționalități studiate în cadrul laboratorului, am reușit să dezvolt o soluție capabilă să reacționeze în timp real la condițiile de mediu și să ofere un feedback clar utilizatorului.

Implementarea s-a realizat incremental, cu validare individuală a fiecărui modul, ceea ce a condus la o structură de cod stabilă și extensibilă. Utilizarea afișajului LCD și a LED-urilor de stare a contribuit la o interfață de utilizare intuitivă, în timp ce gestionarea semnalelor PWM, ADC și I2C a oferit un exemplu concret de aplicare practică a noțiunilor teoretice.

Proiectul poate fi îmbunătățit pe viitor prin integrarea unei conexiuni wireless (Bluetooth sau WiFi), adăugarea unui sistem de logare a datelor sau conectarea la o aplicație mobilă. În forma actuală, sistemul este complet funcțional și demonstrează eficiența soluțiilor embedded în automatizarea diverselor spații.

Download

Arhiva de mai jos conține toate fișierele relevante pentru proiectul „Pergolă Automatizată”, inclusiv codul sursă, schemele electrice, documentația auxiliară și un videoclip demonstrativ.

Fișierele incluse:

- Codul sursă complet: `main\_pergola.ino`
- Scheme electrice realizate în KiCad (`kicad\_sch`) și versiune exportată `.jpg`
- Diagramă logică de semnal (`png`)
- Fotografii cu montajul fizic al circuitului
- Fișier `README.txt` cu instrucțiuni detaliate
- `ChangeLog.txt` cu istoricul modificărilor
- Videoclip de prezentare a funcționării sistemului

[Descarcă arhiva completă a proiectului \(.zip\)](#)

Bibliografie/Resurse

Resurse Hardware

- Datasheet ATmega328P – [https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf Datasheet oficial Microchip]
- Specificații senzor DHT11 – [<https://sigmanortec.ro/senzor-temperatura-si-umiditate-dht22> Senzor DHT11 - Sigmanortec]
- Modul senzor ploaie – [<https://www.optimusdigital.ro/ro/senzori-senzori-de-umiditate/5775-modul-senzor-de-ploaie.html> Modul senzor ploaie - Optimus Digital]
- Driver motoare L298N – [<https://www.handsontec.com/dataspecs/L298N%20Motor%20Driver.pdf> Datasheet oficial L298N]
- LCD 16x2 cu I2C – [<https://www.optimusdigital.ro/en/lcds/62-1602-lcd-with-i2c-interface-and-yellow-green-backlight.htm> | LCD cu I2C - Optimus Digital]

Resurse Software

- Arduino IDE – [<https://www.arduino.cc/en/software>]
- Biblioteca DHT – [<https://github.com/adafruit/DHT-sensor-library>]
- Biblioteca LiquidCrystal_I2C – [https://github.com/johnrickman/LiquidCrystal_I2C]
- WaveDrom editor – [<https://wavedrom.com/editor.html>]
- Tinkercad Circuits – [<https://www.tinkercad.com/circuits>]
- KiCad EDA – [<https://www.kicad.org/>]

Alte resurse

- Forum Arduino – [<https://forum.arduino.cc/>]

From:

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

Permanent link:

http://ocw.cs.pub.ro/courses/pm/prj2025/avaduva/ioan_calin.ilie



Last update: **2025/05/21 15:05**