

Controler lumina

Introducere

Ce face proiectul?

Proiectul constă într-un sistem de control automat al iluminatului, care folosește senzori de mișcare și lumină pentru a aprinde sau stinge lămpile în mod inteligent. Sistemul detectează prezența unei persoane printr-un senzor PIR conectat la un pin de întrerupere și ajustează starea luminilor în funcție de nivelul de lumină ambientală măsurat cu ajutorul unui senzor conectat la un pin ADC. De asemenea, folosește timere pentru a opri lumina automat după o perioadă de inactivitate. Prin UART, datele pot fi transmise către o aplicație externă pentru monitorizare.

Care este scopul proiectului?

Scopul proiectului este de a crește eficiența energetică și confortul utilizatorilor prin automatizarea iluminatului în funcție de condițiile din mediu. Sistemul elimină necesitatea acționării manuale a întrerupătoarelor și previne risipa de energie atunci când luminile rămân aprinse inutil.

Care a fost ideea de la care am pornit?

Ideea a pornit de la observația că în multe spații - cum ar fi holurile, băile sau camerele de trecere - luminile rămân aprinse chiar și atunci când nu este nimeni prezent. Am dorit să dezvolt un sistem care să automatizeze procesul de aprindere/stingere a luminilor și să reacționeze atât la prezență, cât și la lumina din mediu, fără intervenție manuală.

De ce cred că este util?

Acest proiect este util pentru oricine dorește să reducă consumul de energie și să își automatizeze locuința sau spațiul de lucru. Este o soluție accesibilă și ușor de integrat, care oferă atât beneficii economice, cât și funcționalitate inteligentă.

Descriere generală

Acest proiect reprezintă un sistem de iluminat automat controlat de un microcontroler Arduino UNO, care folosește senzori de mișcare și lumină pentru a aprinde sau stinge luminile în mod inteligent. Scopul său este de a optimiza consumul de energie și de a îmbunătăți confortul utilizatorului, eliminând necesitatea acționării manuale a luminilor. }

Componente și interacțiuni

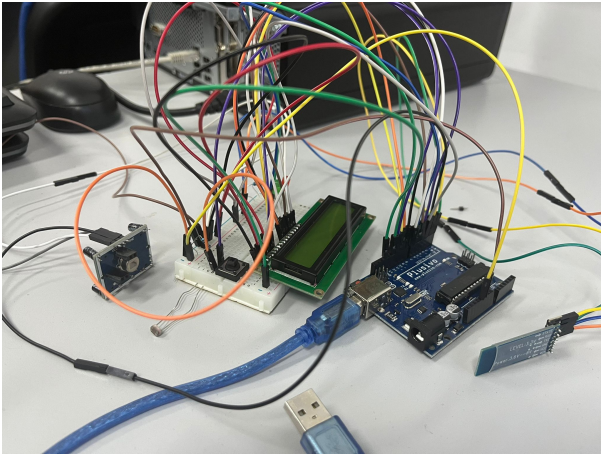
- **Arduino UNO** este unitatea centrală de control care gestionează toate componentele sistemului, atât pe partea de intrare (senzori), cât și pe partea de ieșire (relee/lumină, comunicație, cronometrare).
- **Senzor de mișcare (PIR)** este conectat la un pin GPIO configurat cu întreruperi. Când senzorul detectează mișcare, trimite un semnal către Arduino, care aprinde lampa.
- **Senzor de lumină (fotorezistor)** este conectat la un pin ADC. Arduino citește nivelul de lumină ambientală și decide dacă este nevoie să aprindă lampa (doar dacă e întuneric).
- **Releu** acționat de un pin GPIO controlează efectiv sursa de lumină (aprindere/stingere), în funcție de datele primite de la senzori.



Hardware Design

Lista piese:

1. Arduino ATmega328P
2. Senzor PIR
3. Buton
4. Led-uri
5. Bluetooth HC-05
6. LCD
7. Rezistente



Software Design

Link GitHub: https://github.com/TraistaruAndreea/Proiect_PM

Firmware-ul este scris in limbajul C++ folosind extensia Arduino C pentru platforma AVR. Aplicatia este impartita in mai multe componente logice:

- Initializare hardware (pini, PWM pentru contrast, UART pentru Bluetooth, LCD).
- Rutine de citire pentru buton cu debounce si senzor PIR.
- Controlul LED-urilor in functie de senzor si comenzi.
- Interfata cu utilizatorul prin LCD si Bluetooth UART.
- Logare a evenimentelor de miscare prin Bluetooth.

Mediu de dezvoltare:

- Platforma utilizata: Arduino IDE
- Microcontroler: ATmega328P
- Toolchain: AVR-GCC + Arduino Core + AVR Libc
- Alte unelte: `avr/io.h`, `util/delay.h` pentru control low-level

Algoritmi si structuri implementate:

- Debounce software pentru buton, cu verificare temporala pentru a preveni declansari multiple false.
- Comunicare UART (Bluetooth): trimitere si receptie caractere pentru control de la distanta.
- Actualizare LCD inteligenta: verifica daca textul este diferit inainte de a rescrie liniile, pentru a reduce flicker-ul.
- Control LED-uri pe baza senzorului PIR: LED-ul se aprinde doar daca se detecteaza miscare si se respecta selectia (rosu sau verde).
- Bufferizare si formatare text pentru trimiterea datelor prin Bluetooth (`sprintf` in `printBuffer`).
- Protectie intreruperi (`cli()` / `sei()`) la actualizarea mesajelor pe LCD, pentru consistenta.

Rezultate Obținute

DEMO: https://drive.google.com/file/d/1_f0R-Ac0D2NqhR1YbWh6enbfPSf868oR/view?usp=sharing

Sistemul utilizează un senzor PIR pentru a detecta mișcarea. Două LED-uri (roșu și verde) indică starea sistemului:

- Roșu: aprins la comanda primită sau buton, dacă există mișcare.
- Verde: aprins la comanda primită sau buton, dacă există mișcare.
- Ambele LED-uri stinse dacă nu există mișcare.

Starea LED-urilor poate fi schimbată fie cu un buton fizic, fie prin Bluetooth. Afisajul LCD prezintă starea curentă (mișcare, LED aprins/stins). Cu ajutorul modulului Bluetooth, utilizatorul poate trimite comenzi pentru a schimba culoarea LED-ului aprins.



Concluzii

Download

[traistaru_andreea_cosmina_332cd_controler_lumina.zip](#)

Jurnal

- 5 mai: incepere documentare proiect
- 7 mai: comanda piese
- 14 mai: implementare hardware senzor pir + 1 led
- 15 mai: implementare hardware buton + 2 led-uri
- 15 mai: comanda bluetooth
- 16 mai: implementare hardware proiect final
- 17 mai: implementare software cu functii arduino
- 24 mai: implementare software utilizand concepte laborator
- 26 mai: am construit proiectul pe placuta de prototipare + modificare cod

Bibliografie/Resurse

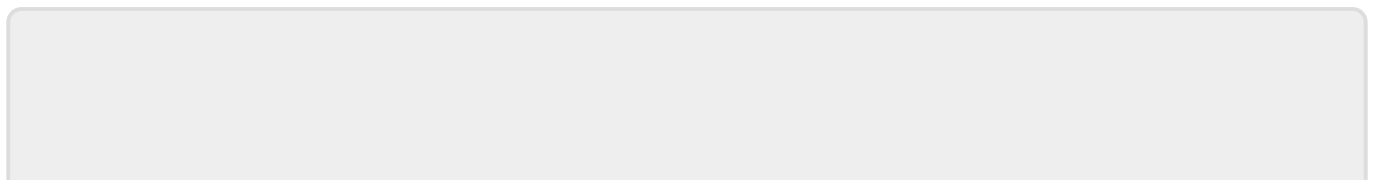
Resurse Hardware:

- [Placadedezvoltare Arduino UNO (ATmega328P)](<https://store.arduino.cc/products/arduino-uno-rev3>).
- [SenzorPIRHC-SR501](https://www.electronics-tutorials.ws/io/io_4.html) (datasheet: [pdf](<https://cdn.sparkfun.com/datasheets/Sensors/Proximity/HCSR501.pdf>))
- [ModulBluetoothHC-05](<https://components101.com/wireless/hc-05-bluetooth-module>) (datasheet: [pdf](<https://cdn.sparkfun.com/datasheets/Wireless/Bluetooth/rn-42-ds.pdf>))
- [DisplayLCD16x2HD44780](<https://www.electronicwings.com/nodemcu/lcd-16x2-interfacing-with-nodemcu>)(datasheet:[pdf](<https://www.sparkfun.com/datasheets/LCD/HD44780.pdf>))

Resurse Software:

- [ArduinoIDE](<https://www.arduino.cc/en/software>)
- [PlatformIO](<https://platformio.org/>)
- [LibreriaLiquidCrystal](<https://www.arduino.cc/en/Reference/LiquidCrystal>)
- [LibreriaSoftwareSerial](<https://www.arduino.cc/en/Reference/softwareSerial>)

[Export to PDF](#)



From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2025/ccristi/andreea.traistaru03>



Last update: **2025/05/27 20:42**