

TeleAlarm

NUME: Paius Marian-Cosmin
GRUPA: 332CC

Introducere

Proiect: Sistem de alarmă cu notificare pe telefon ☐

Ce face: Proiectul constă în crearea unui sistem de alarmă simplu care trimite notificări pe telefonul deținătorului atunci când detectează mișcare. Folosim un microcontroler ESP8266 și senzorul de mișcare PIR în combinație cu platforma IFTTT pentru a trimite notificări.

Scopul acestui proiect este de a crea un sistem de securitate accesibil și eficient care să poată fi folosit în locuințe, birouri sau alte spații care necesită monitorizare.

Ideea de la care am pornit: Am pornit de la ideea de a dezvolta un sistem de securitate ușor de implementat și personalizat care să ofere o soluție accesibilă și eficientă pentru monitorizarea spațiilor.

Utilitate: Acest sistem de alarmă poate fi util pentru toți cei care doresc să-și protejeze și să-și monitorizeze spațiile într-un mod simplu și eficient, fără a fi nevoie de investiții mari. Pe lângă protecția pe care o oferă, acest proiect este o oportunitate bună pentru a învăța despre microcontrolere, senzori și comunicarea cu IFTTT.

Descriere generală

☐ Proiectul vizează realizarea unui sistem de alarmă simplu, care trimite notificări pe telefon atunci când detectează mișcare. Acesta utilizează un microcontroler ESP8266 și un senzor de mișcare PIR pentru a monitoriza mișcarea într-o zonă și trimite notificări prin intermediul platformei IFTTT.

Schema bloc:

1. Microcontroler ESP8266: Acesta controlează toate componentele hardware și comunică cu platforma IFTTT pentru a trimite notificări.
2. Senzor de mișcare PIR: Detectează mișcarea și trimite semnalul la microcontrolerul ESP8266.
3. Platforma IFTTT: Primește semnalul de la ESP8266 și trimite notificări către telefonul deținătorului.

Interacțiunea dintre module:

Senzorul de mișcare PIR trimite semnalul de mișcare detectată către microcontrolerul ESP8266. Acesta, la rândul său, transmite informația către platforma IFTTT, care generează și trimite notificări

pe telefonul personal.

Exemplu de schemă bloc:



Hardware Design

În această secțiune, voi prezenta detaliile legate de designul hardware al sistemului de alarmă:

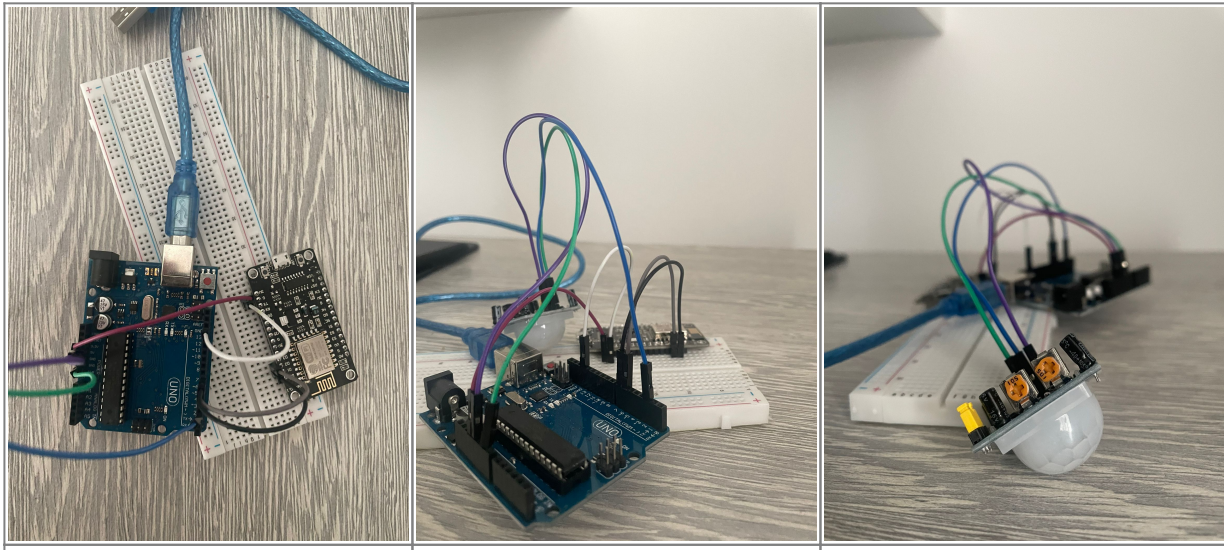
Lista de piese:

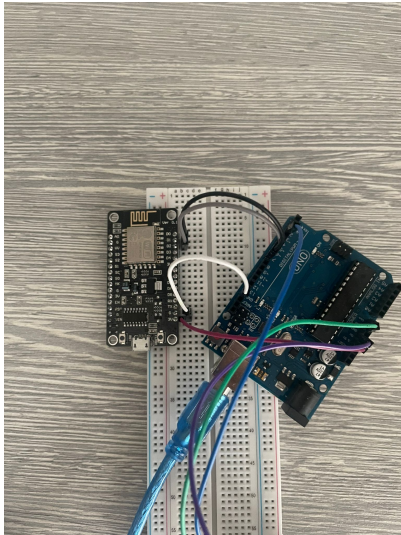
- Arduino UNO
- Microcontroler ESP8266
- Senzor de mișcare PIR
- Breadboard
- Cabluri jumper

Scheme electrice: Pentru conectarea componentelor, puteți consulta următoarea schemă electrică, care ilustrează conexiunile necesare între placuta Arduino, microcontrolerul ESP8266 și senzorul de mișcare PIR:



Imagini:





Rezultatele simulării: După conectarea și programarea componentelor, puteți verifica funcționarea corectă a sistemului prin simularea unei mișcări în fața senzorului PIR. Dacă sistemul funcționează corect, veți primi o notificare pe telefonul dvs. prin intermediul aplicației IFTTT.

Software Design

Mediu de dezvoltare: Vom folosi Arduino IDE pentru a dezvolta și încărca firmware-ul pe microcontrolerul ESP8266.

Librării și surse 3rd-party:

ESP8266WiFi - Biblioteca pentru gestionarea conexiunii Wi-Fi pe ESP8266.

ArduinoHttpClient - Biblioteca care permite realizarea de cereri HTTP într-un mod simplu și eficient, folosită pentru a comunica cu serviciul IFTTT Webhooks.

Algoritmi și structuri pe care plănuți să le implementați:

- Conectarea la rețeaua Wi-Fi și obținerea unei adrese IP valide.
- Monitorizarea stării senzorului PIR și detectarea mișcării.
- Trimiterea unei notificări prin intermediul serviciului IFTTT Webhooks atunci când este detectată mișcare.

Surse și funcții implementate: Funcția `setup()` - Inițializează conexiunea Wi-Fi(NodeMCU), setează pentru placuta Arduino pinul de intrare pentru senzorul PIR și configurează comunicarea serială pentru depanare.

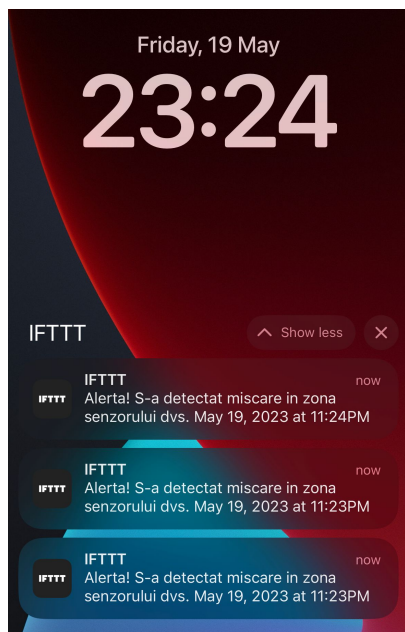
Funcția `loop()` - Verifică în mod continuu starea senzorului PIR și trimite o notificare prin IFTTT atunci când detectează mișcare.

Funcția `loop()(NodeMCU)` - Realizează o cerere HTTP către IFTTT Webhooks pentru a trimite o notificare pe telefonul utilizatorului.

Rezultate Obținute

Daca senzorul detecteaza miscare, timp de 5 secunde nu mai poate detecta alta miscare, pentru a nu

trimite o cantitate mare de notificari.



Download

O arhivă (sau mai multe dacă este cazul) cu fișierele obținute în urma realizării proiectului: surse, scheme, etc. Un fișier README, un ChangeLog, un script de compilare și copiere automată pe uC crează întotdeauna o impresie bună .

Fișierele se încarcă pe wiki folosind facilitatea **Add Images or other files**. Namespace-ul în care se încarcă fișierele este de tipul **:pm:prj20??:c?** sau **:pm:prj20??:c?:nume_student** (dacă este cazul).
Exemplu: Dumitru Alin, 331CC → **:pm:prj2009:cc:dumitru_alin**.

— [Marian-Cosmin PAIUS \(117101\)](#) 2023/05/07 15:50

Jurnal

- **28.04.2023** - Alegere tema proiect □
- **30.04.2023** - Comanda de piese a fost plasata □
- **07.05.2023** - Documentatie + Schema □
- **10.05.2023** - Am obtinut toate piesele necesare □
- **19.05.2023** - Finalizare proiect 

Bibliografie/Resurse

□ Voi prezenta mai jos o listă cu documente, datasheet-uri și resurse pe care le-am folosit pentru proiect, grupate pe **Resurse Software** și **Resurse Hardware**.

Resurse Software:

[Arduino IDE](#) - Mediul de dezvoltare utilizat pentru a programa microcontrolerul ESP8266.

[ESP8266WiFi](#) - Biblioteca pentru gestionarea conexiunii Wi-Fi pe ESP8266.

[ArduinoHttpClient](#) - Biblioteca pentru realizarea de cereri HTTP într-un mod simplu și eficient.

Resurse Hardware:

[ESP8266 Datasheet](#) - Datasheet-ul pentru microcontrolerul ESP8266 folosit în acest proiect.

[HC-SR501 Datasheet](#) - Datasheet-ul pentru senzorul de mișcare PIR HC-SR501.

[NodeMCU Documentation](#) - Documentație care include microcontrolerul ESP8266.

[IFTTT](#) - Platforma utilizată pentru a trimite notificări pe telefonul utilizatorului.

[Export to PDF](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2023/alucaci/telealarm>



Last update: **2023/05/19 21:30**