

Sistem All-In-One de securizare incapere

Introducere

- Ciorgan Andrei-Florian
- 332CD
- Indrumator: Razvan Virtan

Proiectul are la baza creerea unui sistem de inchidere ce poate fi atasat unei usi si un sistem de detectare de incendiu care sa deschida usa in cazul unui astfel de eveniment. Prin acest dispozitiv, doresc sa creez un model simplu si usor de utilizat care poate fi incorporat usor de catre un utilizator pentru a spori securitatea unei incaperi.

Cand a trebuit sa aleg un proiect am decis sa ma focusez pe ceva ce ar putea fi folosit si dupa facultate si ceva care sa ma forteze sa lucrez cu cat mai multe tipuri de senzori ca sa imi permita sa inteleg mai bine creerea si proiectarea unor astfel de dispozitive. Utilitatea proiectului final este relativ evidenta prin faptul ca poate oferi o metoda usoara de securizare al unei incapari, lasandu-i la dispozitie utilizatorului o gama variata de metode de deschidere.

Pe langa toate astea, sistemul are la baza si o masura in caz de urgenta: daca se detecteaza o crestere prea mare de temperatura sau o cantitate prea mare de fum, sistemul va porni o alarma si va deschide implicit usa pentru a putea elibera premisa cat mai rapid.

Descriere generală

Un utilizator poate interactiona efectiv cu trei componente: senzor de amprente, de RFID sau prin cod numeric pentru a deschide usa. Restul componentelor (senzorii si buzzerul) sunt folositi in detectarea posibilelor incendii.



Hardware Design

In realizarea proiectului se vor folosi:

- Arduino UNO

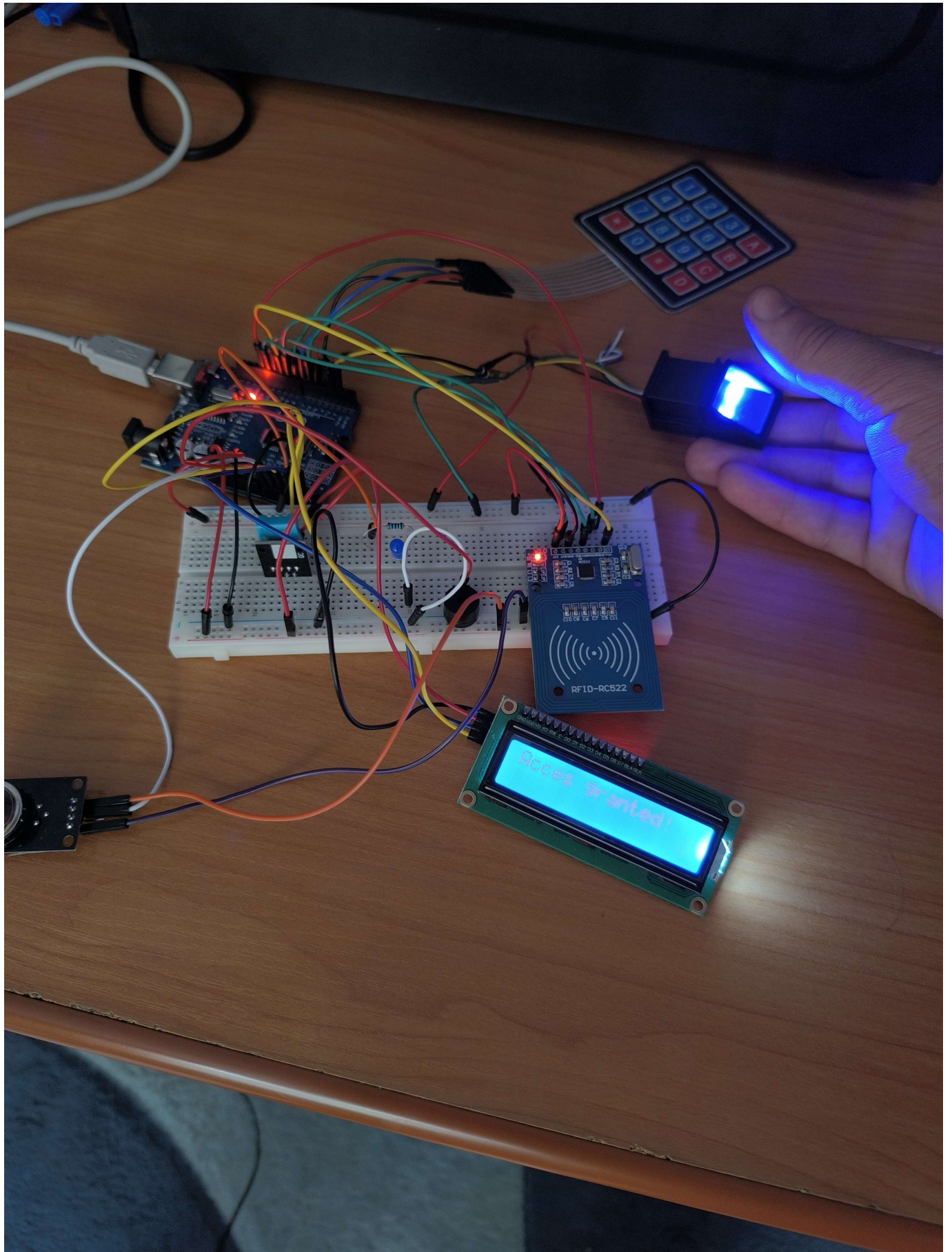
- Modul senzor optic de amprenta
- Modul RFID RC522
- Tastatura matriceala 4x4
- Senzor temperatura DHT11
- Senzor de fum MQ2
- Buzzer pasiv
- Yala electromagnetica ORNO OR-EZ-4033

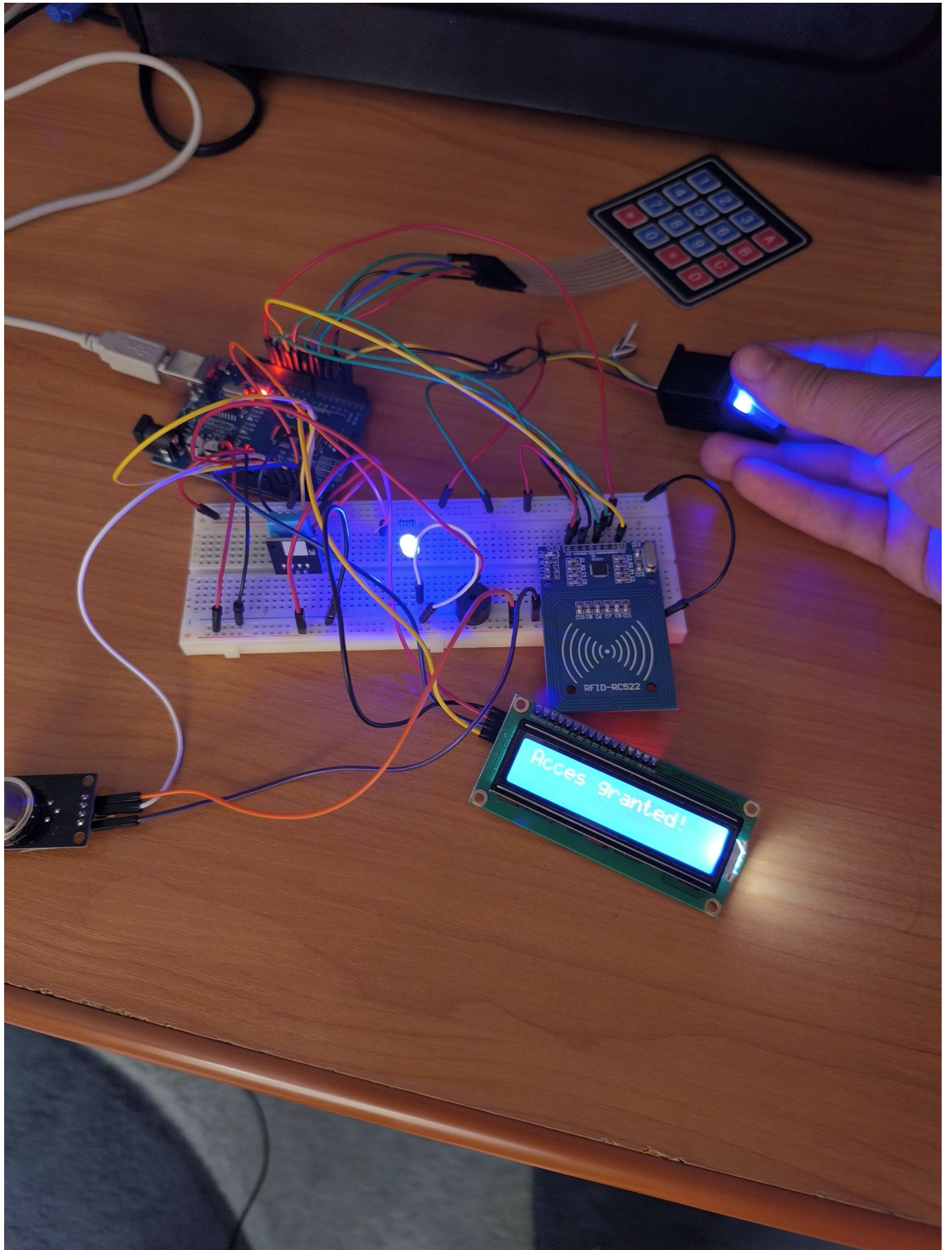


Am inceput proiectarea circuitului prin impartirea pinilor pe fiecare componenta utilizata. Impartirea acestora este urmatoarea:

- Pinii A4 si A5 i-am pastrat pentru comunicatia I2C cu LCD-ul
- Pinii 2,3 sunt pentru comunicatia seriala cu cititorul de amprente
- Pinii 4,5,6,7,8 ii folosesc in comunicatia cu modulul de Keyboard
- Pinii 9,10,11,12,13 sunt alocati comunicarii pe SPI cu senzorul de RFID
- Pinul A0 este asignat citirii valorii calitatii aerului (senzorul de fum)
- Pinul A1 este folosit pentru controlul buzzerului pasiv
- Pinul A2 este rezervat citirii temperaturii
- Pinul A3 este folosit pentru controlul outputului circuitului (in versiunea finala, planuiesc sa inlocuiesc led-ul de pe iesirea acesta cu un releu care sa coordoneze iala electronica)

Se poate observa si cablajul fizic in pozele de mai jos care evidentiaza si functionalitatea senzorului de amprenta.





Software Design

Project link: [GitHub](#)

Am inceput prin crearea unei structuri de baza care sa imi separe functionalitatile senzorilor de foc (temperatura si fum) de modulele ce interactioneaza direct cu utilizatorul (cele de citire). Acest lucru l-am facut deoarece doresc sa cietsc de pe cei doi senzori de incendiu odata la cateva secunde, nu in continuu.

Ca librarii, am folosit:

- LiquidCrystal_I2C.h - utilizat pentru comunicatia cu LCD-ul
- Adafruit_Fingerprint.h, SoftwareSerial.h - folosit pentru comunicarea cu senzorul de amprente
- I2CKeyPad.h, Wire.h - se folosesc pentru a citi date de la keypad, tot pe baza de I2C
- TimerOne.h - utilizat pentru crearea intreruperilor pentru citirea senzorilor de foc
- SPI.h, MFRC522.h - necesar pentru comunicarea cu modulul de RFID
- dht.h - folosit pentru citirea temperaturii
- pitches.h - util pentru crearea de note pentru buzzer

Ca implementare propriu-zisa, avem cateva functii care trebuie mentionate:


- checkFingerprint, checkRFID, checkPassword - acestea se bazeaza pe citirea datelor de la user si verifica daca se indeplinesc conditiile pentru deschiderea usii
- fireCheck - se apeleaza odata la 8 secunde si citeste temperatura si valoarea calitatii aerului. Daca conditiile sunt specifice unui incendiu, va lasa usa deschisa si va porni o alarma
- openDoor - metoda ce apeleaza deschiderea usii si genereaza un jingle

Rezultate Obținute

Care au fost rezultatele obținute în urma realizării proiectului vostru.

Concluzii

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**.

Jurnal

Puteți avea și o secțiune de jurnal în care să poată urmări asistentul de proiect progresul proiectului.

Bibliografie/Resurse

Listă cu documente, datasheet-uri, resurse Internet folosite, eventual grupate pe **Resurse Software** și **Resurse Hardware**.

[Export to PDF](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2024/rvirtan/all-in-one-lock>



Last update: **2024/05/22 17:15**