

Seif Deștept

Nume: Avram Isabela-Maria

Grupa: 331CA

Introducere

Prezentarea pe scurt a proiectului ales:

- Proiectul pe care doresc să-l implementez constă într-un seif deștept. Acesta se poate deschide cu un cifru care a fost setat inițial (introdus printr-o tastatură).
- Scopul proiectului este ca să pot avea un loc mai sigur pentru depozitarea lucrurilor importante și de valoare.
- Ideea de la care am pornit a fost faptul că aveam un cub din lemn în casă care nu avea nicio întrebuințare substanțială, era practic doar un suport, așa că m-am gândit că aș putea să-l folosesc cumva la proiectul de la PM (deși am sfârșit prin a folosi polistiren). Dat fiind faptul că este un cub, am zis că ar trebui să realizez un proiect prin care mă pot folosi de faptul că pot depozita ceva în interiorul cubului, așa mi-a venit ideea de seif.
- Pentru mine este util deoarece, așa cum am menționat mai sus, mi-ar prinde bine să am un loc mai sigur pentru depozitarea lucrurilor importante și de valoare, iar pentru alții ar fi de folos tot din același motiv.

Descriere generală

Proiectul este un seif care poate fi deschis printr-un cifru (stabilit inițial în interiorul codului, deci este ceva prestabilit). Dacă cifrul corespunde parolei stabilite inițial, atunci este acționat servo motorul care deschide (printr-un zăvor) ușa seifului. În cazul în care cifrul introdus nu corespunde celui corect, atunci ușa va rămâne închisă. Totodată, la începutul programului, ușa se va închide automat. Se vor afișa mesaje corespunzătoare fiecărei etape în care se află seiful. Seiful va avea 3 melodii încorporate, acestea fiind redade la începutul fiecărei acțiuni importante din cadrul programului (pornire seif, deschidere ușă, cod greșit).



Hardware Design

1. Listă de piese folosite:

- Arduino
- Tastatură
- Servomotor
- Ecran LCD I2C
- Fire
- Buzzer Pasiv
- Rezistență 100Ω
- Breadboard

2. Schema electrică:



Software Design

Mediul de dezvoltare folosit este Arduino IDE.

Librării și surse 3rd-party:

- Wire.h
- Keypad.h
- LiquidCrystal_I2C.h
- Servo.h
- pitches.h

Descrierea funcționalității codului:

- Prima etapă a fost să declar toți parametrii de care am avut nevoie.
- Funcționalitatea dată seifului este practic de a se deschide când este introdusă parola corectă, altfel acesta este menținut închis. Parolele (cea setată = corectă și cea introdusă de la tastatură) sunt verificate caracter cu caracter. Programul nu acceptă mai mult de 4 caractere pentru parolă, fapt pentru care vor fi luate în considerare doar primele 4 caractere introduse de la tastatură.
- Sistemul de închidere este format dintr-un servo motor, un zăvor metalic și o sârmă, astfel că modul în care funcționează acesta depinde în totalitate de corectitudinea parolei introduse, dacă este corectă, servo motorul își va schimba poziția în "unlockPos", iar dacă este incorectă, servo motorul va rămâne în poziția "lockPos". Din poziția "unlockPos", utilizatorul poate alege să închidă seiful apăsând pe butonul cu caracterul "#", astfel servo motorul schimbându-și poziția în "lockPos". Procesul se poate relua astfel la nesfârșit.

Funcțiile implementate în cod sunt următoarele:

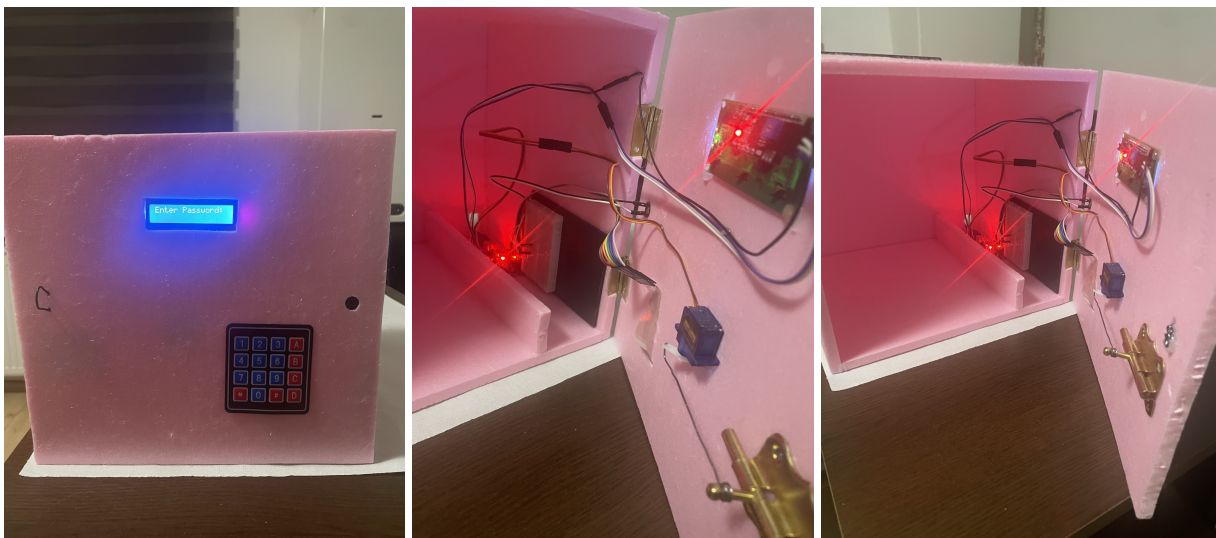
- **void setup():**
 - se configurează pinul buzzer-ului;
 - se închide ușa (servo motorul va trece în poziția "lockPos");
 - se inițializează ecranul LCD și se afișează un mesaj de start;
 - se redă una dintre melodiile definite;
- **void loop():**
 - aici se verifică starea ușii, dacă este deschisă (caz în care se verifică dacă se vrea închiderea ușii cu ajutorul tastei "#", dacă este apăsată, atunci se va trece servo motorul în poziția "lockPos")

sau nu (se va ține ușa deschisă cu ajutorul funcției **void Open()**);


- **void clearData():**
 - această funcție șterge parola introdusă anterior de utilizator pentru a putea avea buffer-ul gol la următoarea introducere noi parole;
- **void ServoOpen():**
 - această funcție aduce servo motorul în poziția "unlockPos", ceea ce înseamnă că ușa se va deschide;
- **void ServoClose():**
 - această funcție aduce servo motorul în poziția "lockPos", ceea ce înseamnă că ușa se va închide;
- **void Open():**
 - în această funcție se așteaptă introducerea unei parole de acces la seif;
 - când o tastă este apăsată, aceasta va fi reținută într-un string care, după introducerea celor 4 caractere necesare, va fi comparat cu parola setată implicit în program (care este și cea corectă);
 - dacă cele două stringuri sunt egale ca valoare, atunci se va apela funcția amintită anterior **void ServoOpen()**, altfel doar se va modifica valoarea variabilei care reține dacă ușa este încuiată sau nu (noua valoare va fi 1, însemnând că ușa este închisă);
 - la final se va apela funcția **void clearData()** amintită mai sus;
- **void lionSleepsTonight(), void nokia(), void pinkPanther():**
 - aceste funcții au fost preluate și folosite pentru a reda melodii în cadrul programului, sursele vor fi adăugate la final în Bibliografie.

Rezultate Obținute

Link către un mic demo al proiectului: <https://youtu.be/Zww7OANcRqY>



Concluzii

A fost destul de interesant să lucrez la acest proiect, deși cred că una dintre cele mai mari provocări a fost construirea efectivă a seifului. Mi-am dorit să existe și o notă de personalitate, așa că am adăugat și melodiile pentru puțină atmosferă drăguță .

Download

[proiectpm_avramisabelamaria.zip](#)

Jurnal

- 29.05.2023: Definitivare documentație OCW
- 28.05.2023: Finalizare parte software
- 21.05.2023: Implementare hardware + schemă bloc + schemă electrică
- 01.05.2023: Adăugare informații pe pagină
- 21.04.2023: Stabilire temă proiect

Bibliografie/Resurse

1. Resurse software:

- <https://github.com/hibit-dev/buzzer/tree/master/src>
- <https://forum.arduino.cc/>
- <https://ocw.cs.pub.ro/courses/pm/lab/lab1-2022>
- <https://ocw.cs.pub.ro/courses/pm/lab/lab3-2023>
- <https://ocw.cs.pub.ro/courses/pm/lab/lab6-2022>

2. Resurse hardware:

- <https://www.sigmanortec.ro/>

[Export to PDF](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2023/apredescu/seifdestept>



Last update: **2023/05/29 22:30**