

SmartAccess with card

Introducere

Prezentarea pe scurt:

- **Ce face:** Acest dispozitiv citește și scrie cartele RFID și verifică accesul utilizatorilor, de exemplu, la o sală de sport.
- **Care este scopul lui:** Să controleze accesul pe bază de cartelă, într-un mod automatizat, sigur și configurabil.
- **Care a fost ideea de la care ați pornit:** Nevoia de a gestiona eficient accesul persoanelor într-un spațiu controlat, cu posibilitatea de a adăuga/scoate utilizatori și de a înregistra ora accesului.
- **De ce credeți că este util pentru alții și pentru voi:** Poate fi implementat în orice spațiu care necesită control de acces (săli de sport, birouri, laboratoare etc.) și oferă o experiență practică utilă pentru învățarea protocoalelor seriale (UART, I2C) și a programării microcontrolerelor.

Descriere generală

Funcționalități:

- * Citirea și validarea cartelelor RFID - Sistemul detectează apropierea unei cartele RFID și citește datele stocate pe aceasta cu ajutorul modulului RFID RC522. - Cartelele valide sunt recunoscute, iar accesul este permis; în caz contrar, accesul este refuzat.
- * Scrierea de date pe cartelă - Sistemul permite scrierea datelor de identificare pe o cartelă nouă sau actualizarea acestora (ex: ID utilizator, abonament activ etc.).
- * Afișarea orei curente - Un ceas software este implementat cu ajutorul timerelor interne ale microcontrollerului și afișat pe ecranul LCD folosind protocolul I2C.
- * Afișarea stării accesului și a orei - Ecranul LCD 16x2 afișează mesaje precum „Acces permis”, „Acces respins”, precum și ora curentă în timp real.
- * Interfață serială pentru configurare/loguri - Sistemul transmite prin UART informații de tip log, utile pentru depanare sau monitorizare pe un PC (ex: ID-ul ultimei cartele scanate, ora accesului).

Laboratoare folosite:

- **Laboratorul 0 (GPIO)** – Utilizat pentru controlul pinilor digitali necesari în citirea și scrierea către module externe (ex: control LED-uri/buzzer opțional).
- **Laboratorul 1 (USART - UART)** – Comunicația serială este folosită pentru interfața dintre Arduino și modulul RFID RC522, dar și pentru transmiterea de date către PC.
- **Laboratorul 2 (Întrepreri)** – Implementarea butoanelor.
- **Laboratorul 5 (I2C)** – Este utilizat pentru comunicarea eficientă cu afișajul LCD 16x2, printr-o conexiune I2C, reducând numărul de pini utilizați.

Module Hardware:

- Unitate de procesare – Placă de dezvoltare compatibilă cu Arduino UNO (ATmega328p și CH340)
- Cititor cartele – Modul RFID RC522 pentru citirea și scrierea de cartele RFID
- Afișaj – Display LCD 16x2 cu interfață I2C pentru afișarea orei și a mesajelor de status
- Interfață de comunicare – UART pentru conectarea la PC și debug/loguri
- Alimentare – Prin cablu USB sau sursă externă de 5V

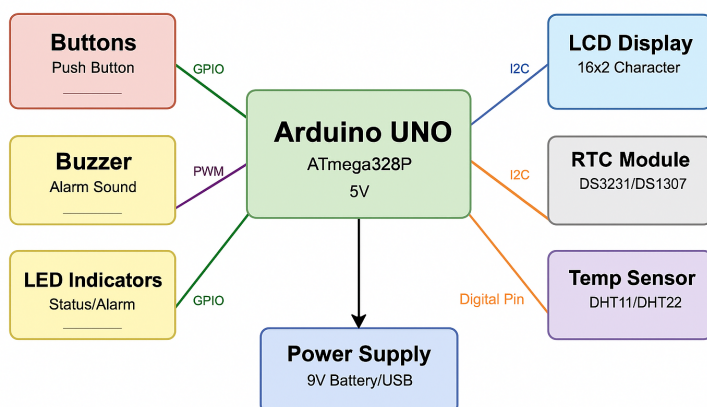
Module Software:

- Gestionare RFID – Citirea/scrierea datelor de pe cartelă folosind librăria MFRC522 și verificarea autenticității
- Control display – Actualizarea ecranului LCD cu ora și statusul accesului
- Ceas intern – Implementarea unui ceas software folosind Timere pentru a menține ora actuală
- Interfață serială – Trimiterea de mesaje de log către PC pentru monitorizarea funcționării
- Logica de acces – Verificarea cartelelor scanate în baza de date stocată local (ex: EEPROM)

Interacțiunea între module:

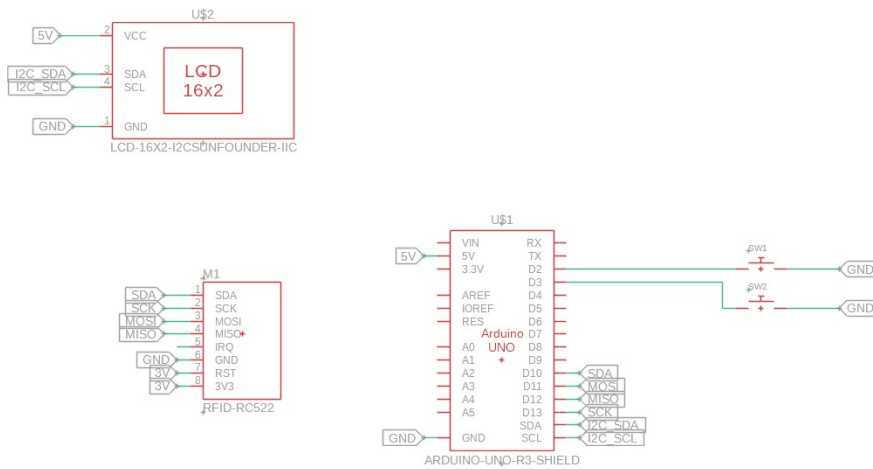
- Modulul RFID RC522 comunică cu Arduino prin UART sau SPI, pentru citirea cartelelor RFID.
- Display-ul LCD este controlat prin I2C pentru afișarea mesajelor și a orei.
- Ceasul software actualizează ora la fiecare secundă folosind Timerul intern al ATmega328p.
- Microcontrollerul gestionează logica de control acces, stochează datele relevante și comunică cu PC-ul prin interfața serială.
- La scanarea unei cartele, sistemul verifică dacă aceasta este validă și afișează rezultatul pe LCD, împreună cu ora exactă.

Schemă Bloc

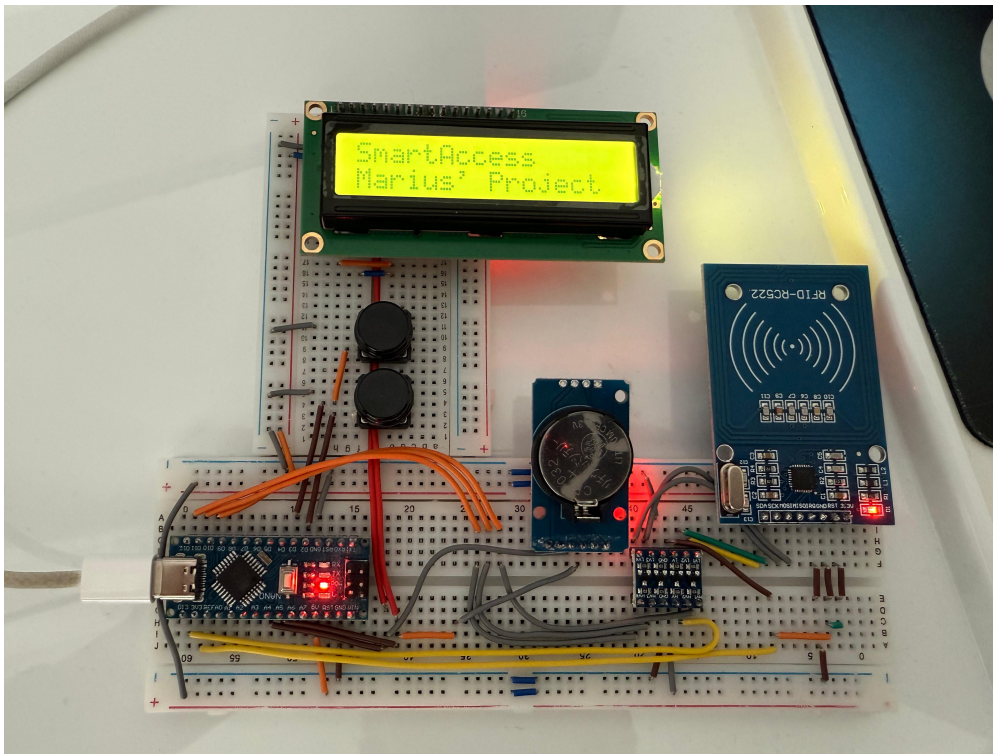


Hardware Design

Schema electrica realizata in fusion



Componentele hardware conectate



Proiect cu carcasa:



Listă de piese:

- Arduino Nano (cu ATmega328P și CH340)
- MFRC522 (RC522) - pentru citirea cardurilor RFID
- Display LCD 1602 cu interfață I2C
- 2x Butoane tactile simple
- Breadboard
- modul RTC

Software Design

Mediu de dezvoltare utilizat Aplicația este scrisă în Arduino C++. Mediul de dezvoltare folosit este Arduino IDE.

Librării și surse third-party utilizate:

- **Wire.h** - Comunicație I2C. Folosită pentru a comunica cu LCD.
- **LiquidCrystal_I2C.h** - Pentru afișaj LCD 16x2 cu interfață I2C și permite afișarea textului fără control direct al pinilor LCD.
- **SPI.h** - Comunicație SPI. Folosită pentru a comunica cu modulul RFID și cu modulul microSD.
- **MFRC522.h** - Folosită pentru a comunica cu modulul RFID.
- **RTCLib.h** - Folosita pentru modulul RTC.

Algoritmi și structuri implementate

- ===== 1. Citire card RFID =====
 - Se utilizează modulul RFID MFRC522.
 - Se citește UID-ul cardului și se verifică dacă are drept de acces.
 - Se verifică în blocul 4 dacă există marcajul „OK”.
 - Dacă accesul este permis, se citește și afișează numele utilizatorului din blocurile 5 și 6.
- ===== 2. Scriere card nou (SCRIERE_NOU) =====
 - Utilizatorul poate introduce un nume de maxim 7 caractere, folosind 2 butoane:
 - Un buton pentru a schimba litera curentă (A-Z).
 - Un buton pentru a trece la următoarea poziție.
 - După confirmare, se scrie:
 - „OK” în blocul 4.
 - Numele în blocurile 5 și 6.
- ===== 3. Copiere card =====
 - Se copiază blocul 4 (starea de acces) de pe un card sursă pe un card destinație.
 - Variante extinse pot copia și blocurile 5 și 6 (numele utilizatorului).
- ===== 4. Interfață meniu pe LCD =====
 - Afișaj LCD 16x2 I2C pentru interacțiune cu utilizatorul.
 - Meniu controlat cu 2 butoane:
 - Navigare prin opțiuni.
 - Confirmare selecție.
 - Opțiuni disponibile:
 - Citire card
 - Scriere (submeniu cu Copiere și Scriere nouă)
 - View Logs (ultimele 5 UID-uri citite)
- ===== 5. Securitate ieșire (în modul Citire) =====
 - Pentru a ieși din modul Citire, utilizatorul trebuie să prezinte un card administrator cu UID-ul **62787441**.
 - Dacă UID-ul nu corespunde, rămâne în modul de citire.
- ===== 6. Salvare acces în EEPROM (View Logs) =====
 - Se salvează în EEPROM ultimele 5 UID-uri ale cardurilor scanate.
 - La alegerea „View Logs” din meniu, se afișează pe ecran lista acestor carduri.
- ===== 7. Gestionare butoane =====
 - Se utilizează întreruperi hardware pentru ambele butoane (Select și Confirm).
 - Se aplică un sistem de anti-rebound cu **millis()** și delay de 200ms pentru debounce.

STRUCTURA GENERALĂ A CODULUI

Firmware-ul este structurat astfel:

- Inițializare componente hardware și variabile globale
- Loop principal (loop()): gestionează modurile de funcționare
- Funcții pentru Citire/Scriere/etc.

Rezultate Obținute

În urma implementării proiectului, am obținut un sistem funcțional care îndeplinește următoarele cerințe:

- **Citirea corectă a UID-ului unui card RFID** folosind un modul MFRC522;
- **Verificarea accesului** prin citirea unui marcaj stocat în card;
- **Afișarea unui meniu interactiv** pe un ecran LCD 16×2 cu interfață I2C, controlat cu doar 2 butoane;
- **Funcționalitate de scriere nouă** a unui card, cu alegerea unui nume personalizat (maxim 7 caractere), care este salvat în blocurile 5 și 6;
- **Funcționalitate de copiere a cardului** (blocurile 4-6) între două carduri;
- **Afișarea numelui de utilizator** dacă acesta este valid și are acces;
- **Stocarea automată a ultimelor 5 UID-uri** în memoria EEPROM și afișarea acestora în meniul „View Logs”;
- **Protecție la ieșirea din modul de citire**, posibilă doar după autentificarea cu un card administrator;
- **Implementarea corectă a debounce-ului** pentru butoane folosind întreruperi și verificare temporală cu `millis()`.

Link videoclip youtube:

<https://youtu.be/e5lpyr1-J40>

Concluzii

Download

Contul de git cu restul resurselor proiectului:

<https://github.com/MariusS16/SmartAccess>

Jurnal

Bibliografie/Resurse

<https://docs.arduino.cc/libraries/liquidcrystal-i2c/>

<https://github.com/adafruit/RTCLib>

<https://docs.arduino.cc/tutorials/nano-esp32/cheat-sheet/>

[Export to PDF](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2025/rnedelcu/marius.slincu>



Last update: **2025/05/28 09:08**