

Security system

Student: **Pană Luca - 334CC**

Introducere

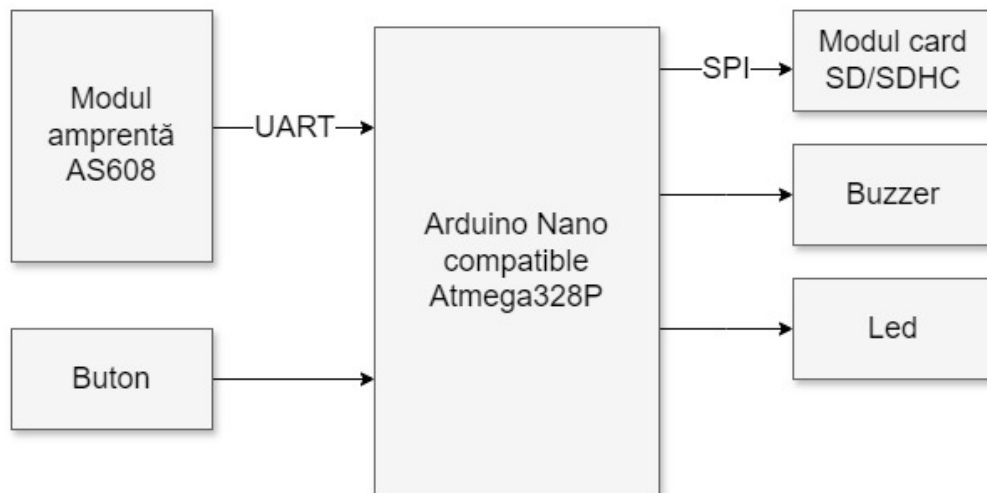
Proiectul isi propune crearea unui mini-sistem de securitate, care ar permite accesul fizic într-un anumit spațiu numai pe bază de autentificare. Înglobează în continuare și capacitatea de autentificare, dar ulterior i-am adăugat și capacitatea de a stoca persistent imaginea amprentei pe SD.

- Scopul: Adaugă un nivel de protecție în plus unui anumit spațiu unde ținem obiecte de valoare, acte, mementouri./ Stochează informații biometrice despre o persoană.
- Ideea de baza: Devine incomod sa cari multe chei cu tine, iar un astfel de sistem elimină nevoie de a le ține la tine.
- Utilitatea: Poate fi util pentru persoanele care nu trăiesc singure și vor să aibă un sertar securizat exclusiv al lor, sau pentru a înregistra intrările într-o casă/firmă și de cine sunt făcute.

Descriere generală

Un modul de citit amprenta care comunica cu arduino si care stocheaza pe un sd amprentele cunoscute cu mod de operare REGISTER (in care face store) si CHECK (in care face search pe SD dupa o amprenta matching) si ofera intr-un fel feedback. In functie de daca o sa folosesc sau nu biblioteci high-level feedbackul asta o sa fie dat ori de un led/buzzer ori un de un LCD.

Schema bloc



Hardware Design

Piese:

- Arduino Nano
- Senzor optic amprenta AS608 [see](#)
- Modul card SD/SDHC + card SDHC 16GB hama
- Breadboard, fire, LED
- Rezistor 10K

Montaj:

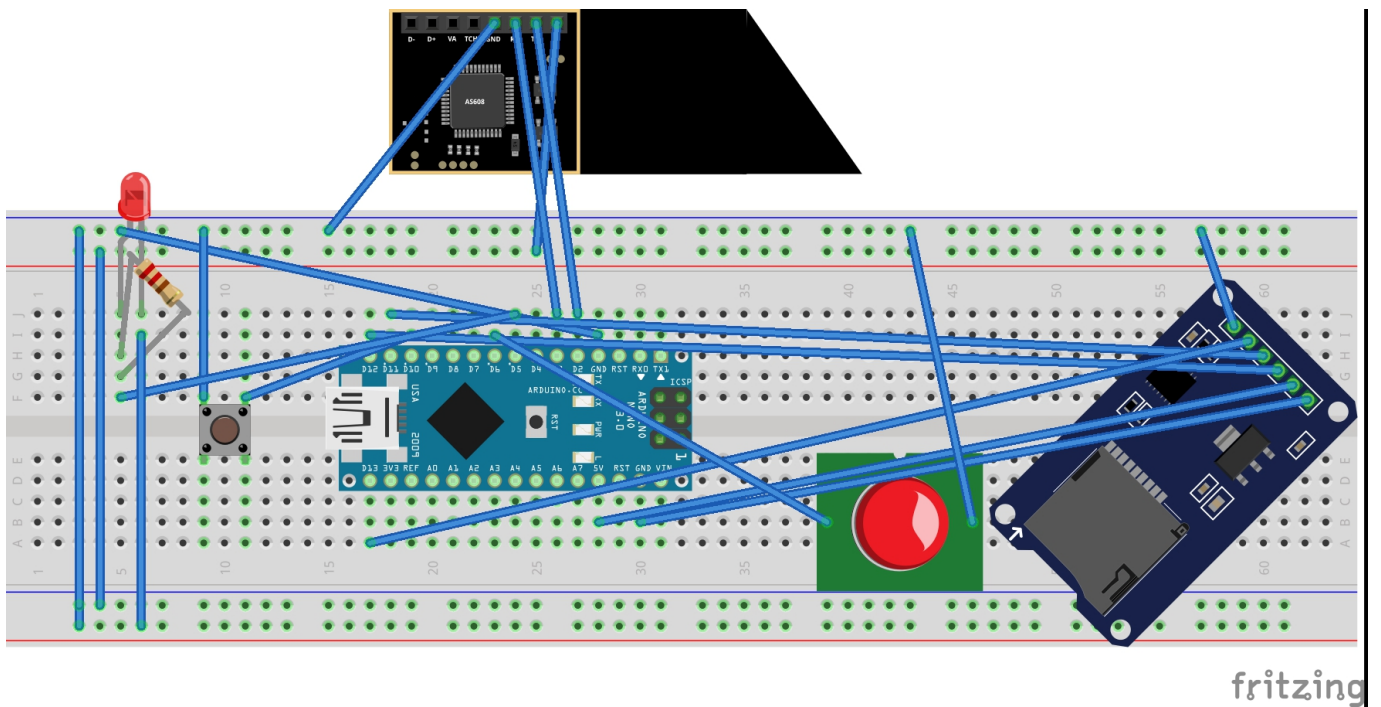
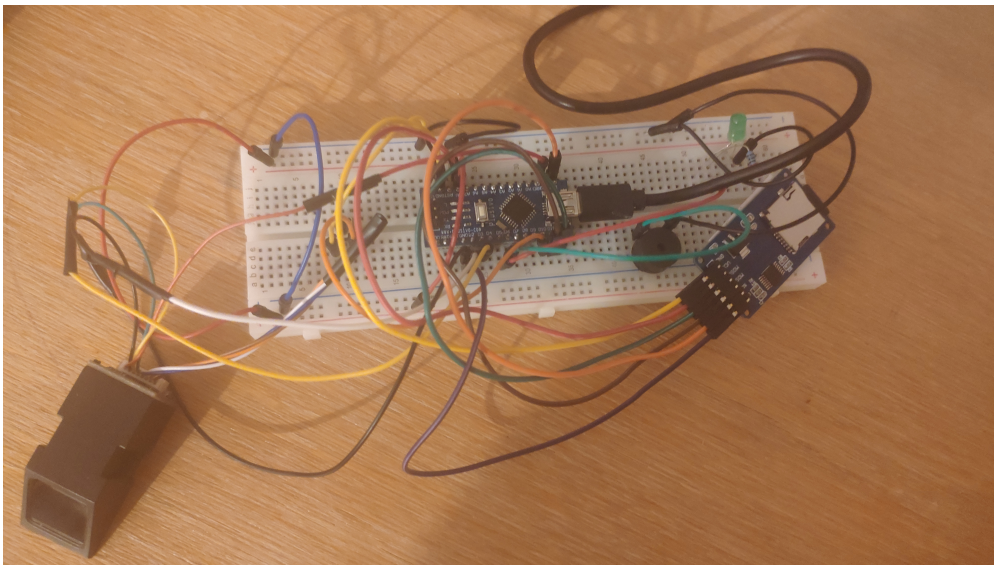
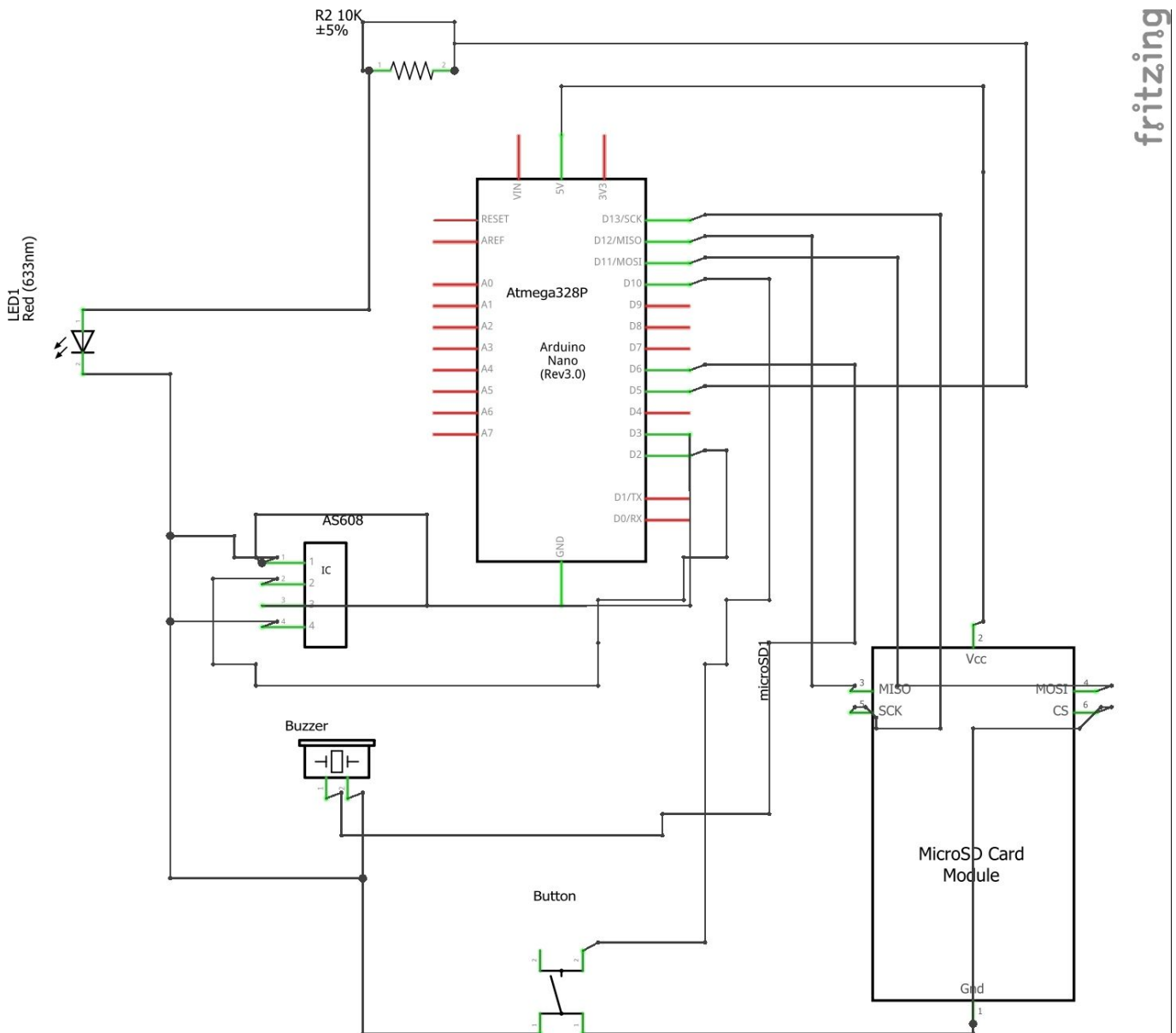


Diagrama electrica



Software Design

Procesările care conțin algoritmul efectiv de matching/ fingerprint analysis/ feature extraction sunt în firmware și neaccesibile programatorului. Reverse engineering la firmware ar fi consumat mult timp dacă ar fi fost posibil și ar fi avut puține beneficii. Ca urmare, am folosit funcțiile de matching și store din bibliotecile corespunzătoare.

- mediu de dezvoltare: Arduino IDE
- surse 3rd party și biblioteci: SD.h, SPI.h, SoftwareSerial.h, Adafruit_Fingerprint.h, FPM.h

Bibliotecile au fost alese în general din considerente de economisire maximă a timpului și lizibilitate a codului pentru că mai există și alte câteva deadline-uri prezente în paralel. Varianta de FPM.h inclusă în arhivă are câteva linii modificate de mine pentru a-mi facilita comunicarea cu SD-ul, versus comunicarea cu Serial pentru care fusese inițial proiectată.

Formarea headerului imaginii .bmp și apoi umplerea conținutului efectiv al imaginii este inspirată din/ portarea scriptului getImage.py <https://github.com/brianrho/FPM/blob/master/extras/getImage.py> pe

arhitectura AVR, tăind PC-ul intermediar din transfer.

Există 2 moduri de operare separate prin apăsarea butonului din montaj. Modul în care pornește este cel de STORE al unei amprente, după store putându-se da switch la modul de match + upload imagine amprentă pe cardul SD. Imaginea este un bitmap de dimensiune 256×286. Feedbackul la interacțiunea cu senzorul este oferit utilizatorului la monitorul serial și prin buzzer. LED-ul se aprinde de 3 ori la terminarea transferului către SD.

Verificarea ulterioară a imaginii de pe SD necesită un device cu display și adaptor sd-usb sau reader sd.

Laboratoare care mi-au fost de folos: Lab0_GPIO, Lab1_USART, LAB5_SPI

Calibrarea elementelor de senzorică a constat în reconfigurarea baud rateului senzorului de amprente de la 57.6k (out-of-the-box default) la 9.6k, care e o schimbare ceva mai permanentă și nu se poate realiza dinamic la runtime.

Nu a fost cazul de optimizări mari. S-ar putea considera mini-optimizări că am fost nevoit să mut niște string literals din RAM în EEPROM și comunicarea arduino - SD făcută buffered printr-un char[64].

Rezultate Obținute / DEMO

Un montaj funcțional care permite stocarea persistentă într-un format cunoscut a unor imagini cu amprente, și care are o formă de autentificare biometrică. La o scară mai neprofesionistă și neconformă cu standarde legale, este similar cu ce ar putea fi găsit într-o secție de făcut pașapoarte/ într-un aeroport.

- [Match fingerprint 1:N + upload bitmap image to SD](#)
- [Enroll fingerprint, reject invalid/messy images, accept\(2 scans to form a template\)](#)

Concluzii

Este foarte important să analizezi dinainte să cumperi senzori / periferice dacă au biblioteci software bine documentate și cu funcționalitățile dorite sau măcar sunt ușor extensibile.

Comunicarea pe UART cuplată cu un microcontroller cu o frecvență care nu permite 0% error rate la niciun baud rate duce la pierderi de date.

Atmega328P este insuficient pentru 99% din procesările de imagini pe care ni le-am putea dori să le facem.

Download

[security_system-airport_fingerprint_scanner.zip](#)

Jurnal

Senzorii de amprentă de la Adafruit sunt destul de slab documentați, mai ales pe partea de interacțiune programatică, existând propriu-zis un singur manual de utilizator machine-translated din chineză.

Frecvența de 16MHz de pe Atmega328P duce la un 0.2% error rate pentru baud rate 9600, cât am ales să folosesc în proiect, error rate vizibil la transferurile de 34Kb pe care am avut nevoie să le fac pentru a scrie fișierul .bmp pe cardul sd. Baud rate-urile mai mari duceau la semnificativ mai multe erori și nu merită menționate.

Însăși extragerea imaginii cu amprenta generată de senzor nu era posibilă doar cu ajutorul bibliotecii oficiale Adafruit, ci numai cu o altă bibliotecă 3rd party înlocuitoare și relativ necunoscută (FPM.h).

Bibliografie/Resurse

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

- <https://github.com/adafruit/Adafruit-Fingerprint-Sensor-Library/issues/114>
- <https://forum.arduino.cc/t/reading-writing-struct-to-sd-card/187722/11>
- <https://www.arduino.cc/reference/en/libraries/sd/>
- <https://arduino.stackexchange.com/questions/23163/arduino-and-2-adafruit-fingerprint-modules-working-together>
- <https://www.engineersgarage.com/arduino-adafruit-r30x-r307-fingerprint-scanner/>

[Export to PDF](#)

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

Permanent link:
<http://ocw.cs.pub.ro/courses/pm/prj2024/alucaci/luca.pana>



Last update: **2024/05/22 13:53**