

# Alarma securitate

## Introducere

**Alarma securitate** este o alarma cu display care se declanseaza daca detecteaza miscare, putand fi dezactivata doar printr-o telecomanda.

## Descriere generală

La inceput alarma e pe off si poate fi activata prin telecomanda prin plimbarea printr-un meniu  afisat pe display-ul LCD 12C. In plus, tot din acest meniu se poate seta timer-ul la 5 sec sau 10 sec. Acest timer e folosit pt. a sti dupa cate secunde se porneste senzorul de miscare dupa alegerea activarii alarmei. Daca senzorul de miscare detecteaza ceva, o sa se auda un sunet ca de alarma care poate fi oprit doar din telecomanda iar alarma se va dezactiva, urmand ca apoi sa fie reactivata tot din telecomanda daca se doreste. In plus mai sunt 3 LED-uri, dintre care unul rosu care indica daca alarma e pe off, unul verde care indica daca e pe on si unul galben care indica cand alarma poate primi comenzi de la telecomanda intervalul fiind de minim 1 sec intre 2 comenzi succesive.

## Hardware Design

 *Lista piese:*

- Arduino Uno
- LCD 12C
- Senzor miscare
- Buzzer
- 3xLED
- Telecomanda
- Senzor
- Breadboard

## Software Design

Descrierea codului aplicatiei (firmware):

- mediu de dezvoltare (if any) (e.g. AVR Studio, CodeVisionAVR)
- librării și surse 3rd-party (e.g. Procyon AVRlib)
- algoritmi și structuri pe care plănuiați să le implementați
- (etapa 3) surse și funcții implementate

- Mediu de dezvoltare: Arduino IDE
- Librării și surse 3rd-party: IRremote.h, LiquidCrystal\_I2C.h

## Detalii de implementare

In setup() am configurat toti pinii pe de care aveam nevoie si am configurat intreruperile de tip timer. Pentru configurarile intreruperilor de tip timer m-am folosit de [laboratorul 3 de PM](#). In codul intreruperii, in functia **ISR()** am o var. **volatile long count** pe care o cresc mereu.

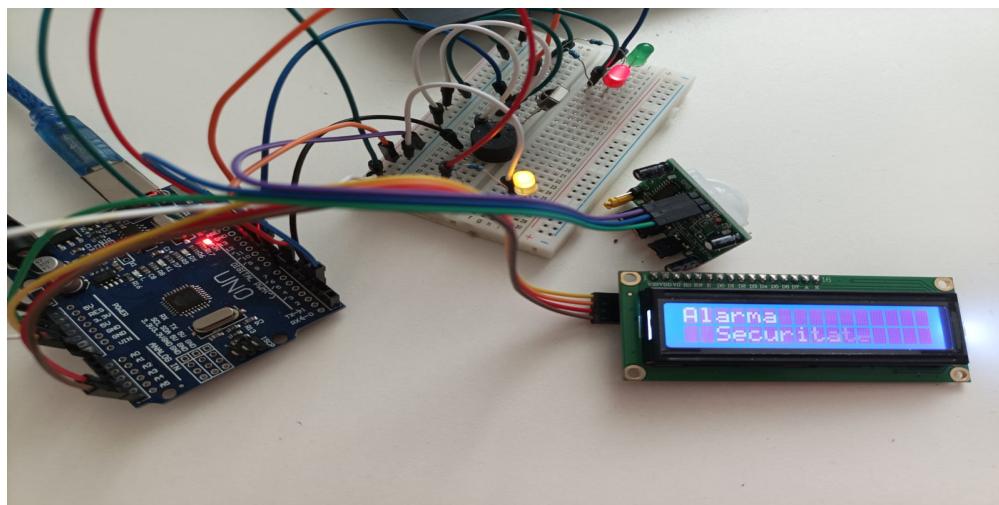
Pentru a sti ce tasta apas pe telecomanda m-am folosit de **IrReceiver.decodedIRData.command** care imi intoarcea un numar mai mare ca zero pentru fiecare tasta in parte, dupa care m-am definit cu define toate tastele cu numarul comenzi dat de senzorul IR cu define (#define TASTA\_OK 28). Cand apasam o data pe o tasta de cele mai multe ori era interpretata de mai multe ori lucru ce ducea la alegerea unor optiuni nedorite pe meniul afisat pe ecranul LCD, lucru ce a trebuit rezolvat prin folosirea unui timer, astfel trebuie sa astepti 1 sec intre 2 comenzi succesive.

Pentru afisarea meniului pe display m-am folosit de un switch case pe mai multe stari, iar in functie de ce tasta e apasata de pe telecomanda, se schimba var **int STATE = 0** cu o alta stare. Pentru a afisa text pe ecran m-am folosit de functiile **Lcd.clear()**, **Lcd.setCursor()** si **Lcd.print()**.

Cand se apasa tasta OK STATE ia val lui STATE\_4, iar var **int flag\_alarma = 0** devine 1, astfel se intra intr-un if statement unde se verifica cu **digitalRead(PIN\_TO\_SENSOR)** daca valoarea pe pin e LOW sau HIGH. Daca e HIGH insaemna ca a fost detectata miscare iar var **int alarma = 0** devine 1, care duce la apelarea functiei **sound()**. Pe pin-ul PIN\_TO\_SENSOR e conectat senzorul de miscare HC-SR501. Dupa ce se pornteste alarma prin selectarea optiunii de ON, nu se verifica imediat daca senzoul a detectat miscare si abia dupa **alarm\_timer** sec, care initial e setat pe 5 doar poate fi crescut la 10 prin selectarea unei optiuni pe display.

Functia **sound()** scrie valori de HIGH, urmate de valori de LOW prin **digitalWrite(buzzer)** la intervale de 1 sec pentru creearea sunetului de alarma, **const int buzzer = 3** e pin-ul pt buzzer.

## Rezultate Obținute



Care au fost rezultatele obținute în urma realizării proiectului vostru.

Video cu funcționalitatile: [Alarma securitate](#)

## Concluzii

Nu poti folosi functia tone() impreuna cu biblioteca IRremote deoarece, ambele lucreaza pe acelasi timer lucru care duce la incapabilitatea senzorului IR de a mai procesa comenzi. Eu avand nevoie doar de un sunet care sa sune a alarma am putut rezolva acest lucru prin scrierea pe pin-ul pe care e conectat buzze-rul semnale de HIGH sau LOW, evitand astfel utilizarea functiei tone(). O alta metoda era schimbarea timer-ului pe care functioneaza biblioteca IRremote.

## Download

Codul sursa → [Arhiva](#)

## Bibliografie/Resurse

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

## Resurse Software

- [Motion sensor HC-SR501](#)

- LCD 12C
- IR sensor
- Timer

[Export to PDF](#)

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

Permanent link:  
<http://ocw.cs.pub.ro/courses/pm/prj2023/danield/alarma-securitate>

Last update: **2023/05/28 18:01**

