

# Monitorizare puls

Student: NEGRU Robert-Marius

Grupa: 342C5

## Introducere

Implementarea unui sistem de monitorizare al pulsului pentru mai multe categorii. Proiectul consta in verificarea valorilor pulsului unei persoane si stabilirea daca este sanatoasa sau nu. In functie de valorile obtinute, se vor folosi doua leduri, unul rosu pentru avertizare ca sunt probleme si unul verde daca persoana este in limitele normale. LED-urile vor blinkui in functie de valorile pulsului.

Se vor trata urmatoarele cazuri:

- Varsta
- Sex
- Efort

Algoritmul o sa trateze cazurile speciale in care o persoana depune efort inainte (e.g. face cateva exercitii inainte de masurarea pulsului) pentru a stabili nivelul pulsului la efort fizic dar si in functie de varsta sau sexul persoanei respective.

In functie de acesti parametrii led-urile vor blink-uri fie rosu fie verde.

## Descriere generală

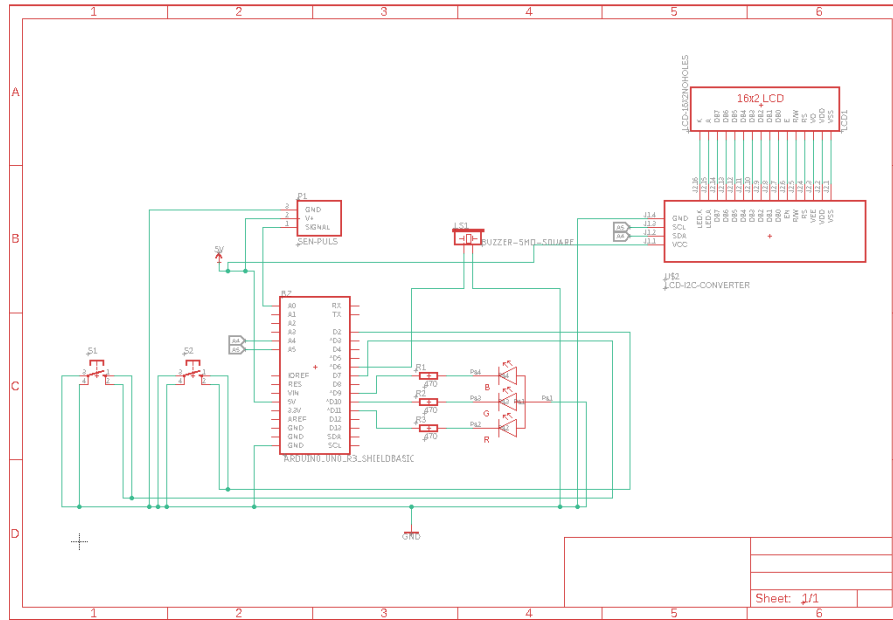
### Mod de funcționare

- Ne vom folosi de butonul de push pentru a incepe procesul si afisam pe ecranul LCD mai multe informatii.
- Utilizatorul se va folosi de senzorul de puls XD-58C pentru a incepe masurarea.
- Valoarea citita de catre senzorul de puls va fi redada pe ecranul LCD.
- In functie de valoarea citita, ne vom folosi de LED-rile rosu si verde pentru a stabili daca este sanatos sau nu.
- Se vor respecta parametrii (varsta, sex).
- LED-ul rosu se va aprinde daca nu sunt in limite normale, asadar daca pulsul este peste sau sub limita. Acesta va blink ui in functie de valorile respective, cu cat valoarea este mult peste sau sub limita, cu atat led-ul rosu va blink ui mai tare.
- LED-ul verde se va aprinde daca limitele sunt normale.
- In timp ce pulsul este masurat, un sunet va fi redat conform batailor inimii cu ajutorul unui buzzer.

## Schema bloc



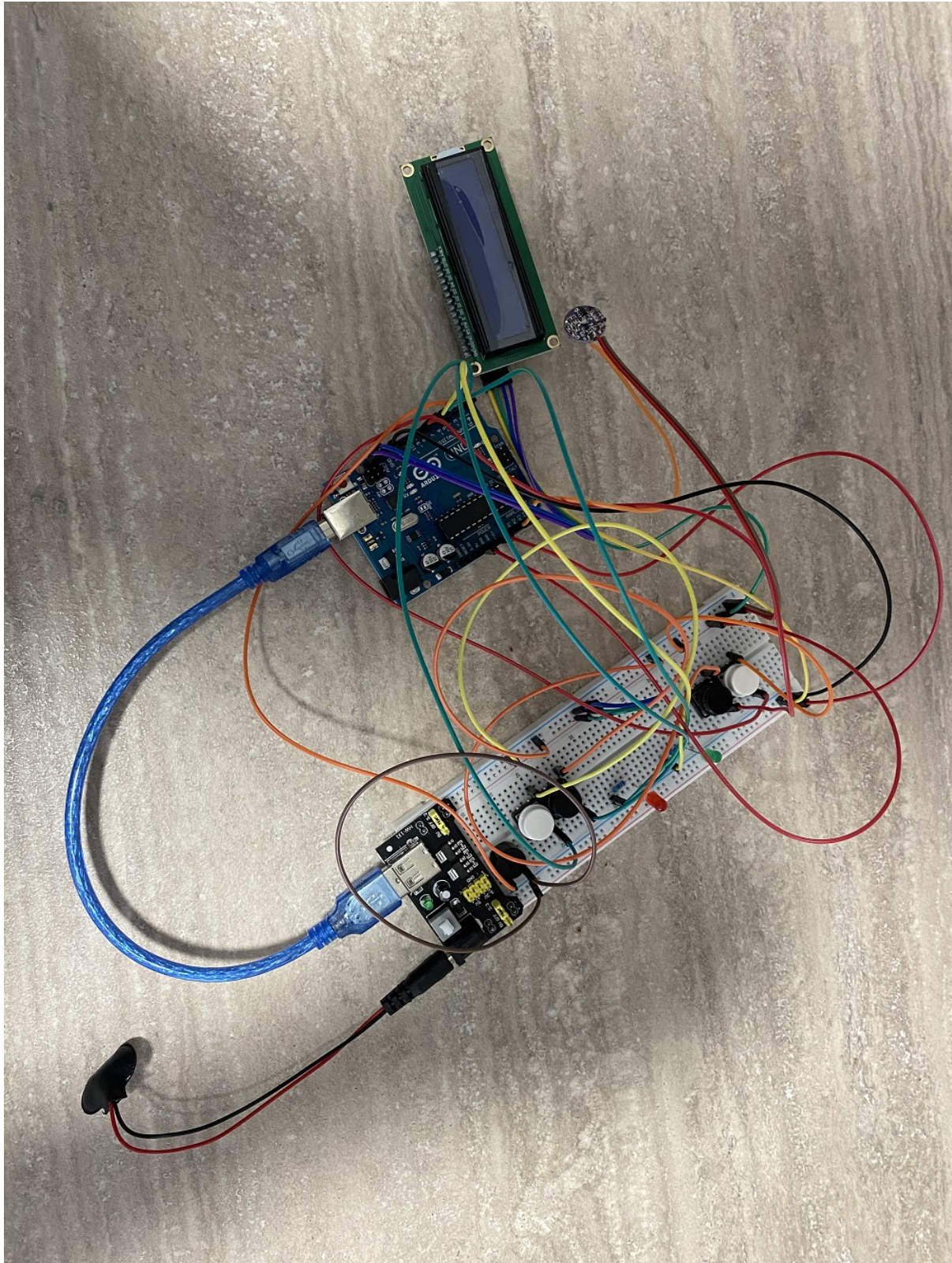
## Schema electrica



## Hardware Design

Listă de piese:

- Arduino Uno
- Breadboard
- Ecran LCD I2C
- Push Button
- Senzor Puls (XD-58C)
- LED rosu si verde
- Buzzer
- Rezistente
- Fire



## Software Design

Descrierea codului aplicației (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 plănuiți să le implementați
- (etapa 3) surse și funcții implementate

## 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

- 6 Mai: Creare pagină Wiki + documentație
- 10 Mai: Comanda piese
- 12 Mai: Piese livrate
- 18 Mai: Implementare hardware

## 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/prj2023/daniel/monitorizarepuls>



Last update: **2023/05/28 18:41**