

Puls - Oximetru

Mihalceanu Cristian-Ionut, grupa 331CC

Introducere

Proiectul urmareste implementarea unui aparat de masura a ritmului cardiac si al nivelului de oxigen din sange

Ideea de la baza o constituie folosirea acestui tip de sistem in timpul pandemiei, in timp ce utilitatea sa se regaseste in confirmarea valorilor oferite de alte puls-oximetre achizitionate de la farmacie la preturi scazute.

Personal, utilitatea proiectului se regaseste si in familiarizarea in lucrul cu Arduino



Descriere generală

Persoana mentine degetul apasat pe senzorul cardiac pentru un interval de timp mai mare decat timpul de stabilizare. Senzorul transmite informatia la Arduino, urmand ca valorile sa fie afisate pe ecranul LCD, buzzerul imitand bataile inimii. In cazul unor valori alarmante, ledul rgb lumineaza rosu, altfel inca verde sau galben.



Hardware Design

Lista piese folosite:

- Breadboard 400 puncte
- DISPLAY OLED IIC I2C 0.91" 128x32
- SENZOR PULS, OXIMETRU, OPTIC, GY MAX30100, I2C
- PLACA DEZVOLTARE UNO R3 ARDUINO

- Modul Buzzer
- LED RGB
- Rezistente, fire

Schema Electrica



Design Circuit



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 plănuți să le implementați
- (etapa 3) surse și funcții implementate

Pentru a realiza acest proiect, am folosit Arduino IDE.

Codul funcționează pentru senzorul de puls-oxigen MAX30102 + 128×32 OLED i2c + Buzzer and Arduino UNO

Ca librării am importat

SparkFun MAX3010x library pentru senzor

Adafruit SSD1306 si Adafruit GFX library pentru ecran si grafica

```
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET);
```

particleSensor.getIR(); - am folosit aceasta functie a senzorului de puls-oxigen pentru a detecta daca o persoana a atins senzorul cu degetul caz in care se declanseaza ritmic buzzerul:

```
tone(3,1000, 100);  
delay(100);  
noTone(3);  
digitalWrite(3, HIGH);
```

Butoanele sunt configurate pentru a modifica durata de functionare a LED ului pentru cazul in care pulsul citit are o valoare ridicata

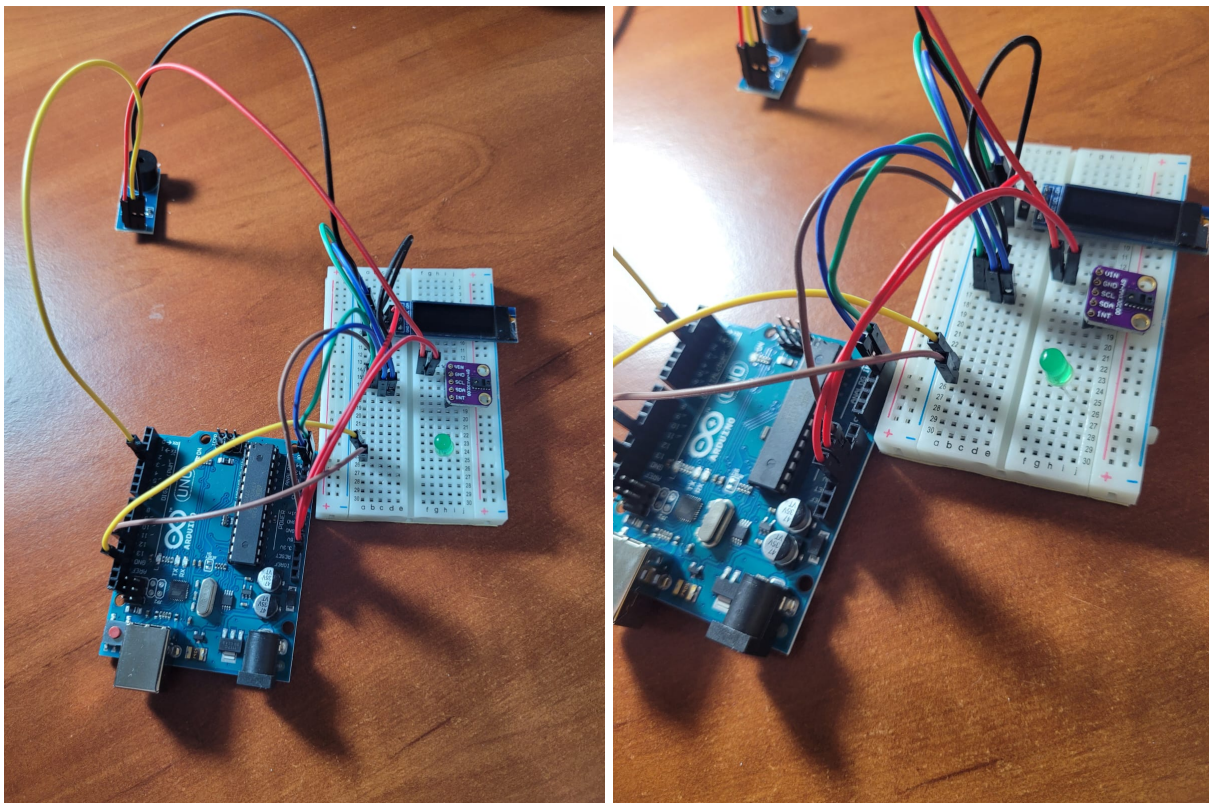
```
buttonState1 = digitalRead(buttonPin1); if(buttonState1 == 0){  
if(lastButton1Press + debounceInterval < millis()){
```

```
lastButton1Press = millis();  
interval -= STEP;  
} }
```

Rezultate Obținute

Care au fost rezultatele obținute în urma realizării proiectului vostru.

Rezultate obtinute dupe etapa Hardware




Concluzii

Videoclip cu rezultatul final:

<https://youtube.com/shorts/Bx30ruMsv18?feature=share>

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**.

Arhiva cu codul sursa se gaseste aici: [prpm1.zip](#)

Jurnal

27.04 - Alegere proiect si discutie in cadrul laboratorului
7.05 - Creare documentatie pe ocw si comanda piese
21.05 - Am adaugat schema electrica
28.05 - Am adaugat videoclip, codul sursa, bibliografie

Bibliografie/Resurse

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

<https://projecthub.arduino.cc/SurtrTech/eb4f74c6-bb73-4148-9aaf-9cf1363e5c6d>
<https://how2electronics.com/interfacing-max30100-pulse-oximeter-sensor-arduino/>
https://www.youtube.com/watch?v=cEtyMkubXj4&t=875s&ab_channel=SurtrTech
<https://forum.arduino.cc/t/2-buttons-1-led/535430>

[Export to PDF](#)

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

Permanent link:
<http://ocw.cs.pub.ro/courses/pm/prj2023/ndrogeanu/puls-oximetru> 

Last update: **2023/05/29 15:10**