

Pulse Meter

Patrascu Maria-Larisa

Grupa: 334CB

Introducere

Proiectul reprezinta un aparat care masoara pulsul. Atunci cand nu este niciun buton apasat se va afisa pe ecran mesajul "Press button to take your pulse!" pentru a anunta utilizatorul ca trebuie sa tina butonul alb apasat in timpul masurarii pulsului. Se ruleaza programul de python pentru a putea introduce datele utilizatorului in baza de date, apoi in timpul apasarii butonului alb se masoara pulsul si va functiona un buzzer, iar in functie de valoarea pulsului, un led se va aprinde pe o anumita culoare (verde, galben, rosu). Daca senzorul nu depisteaza nimic, ledul va fi alb. Dupa preluarea datelor se apasa butonul albastru pentru a trimite valorile masurate de catre senzorul de puls catre baza de date, pentru a putea fi inregistrat utilizatorul.

Descriere generală

In implementarea proiectului voi folosi, pe langa componentele de baza, un senzor de puls, un Display OLED, un LED RGB, un buzzer si doua butoane. Atunci cand este apasat butonul alb si se pune degetul pe senzor, se asteapta afisarea valorii pulsului pe ecran. In functie de aceasta culoarea LED-ului o sa se schimbe, iar buzzer-ul o sa functioneze pe intreaga perioada a masurarii. Se apasa butonul albastru pentru a trimite valorile masurate de catre senzorul de puls catre baza de date. Daca butonul alb nu este apasat, atunci se va afisa pe ecran un mesaj pentru indrumarea utilizatorului.

Hardware Design

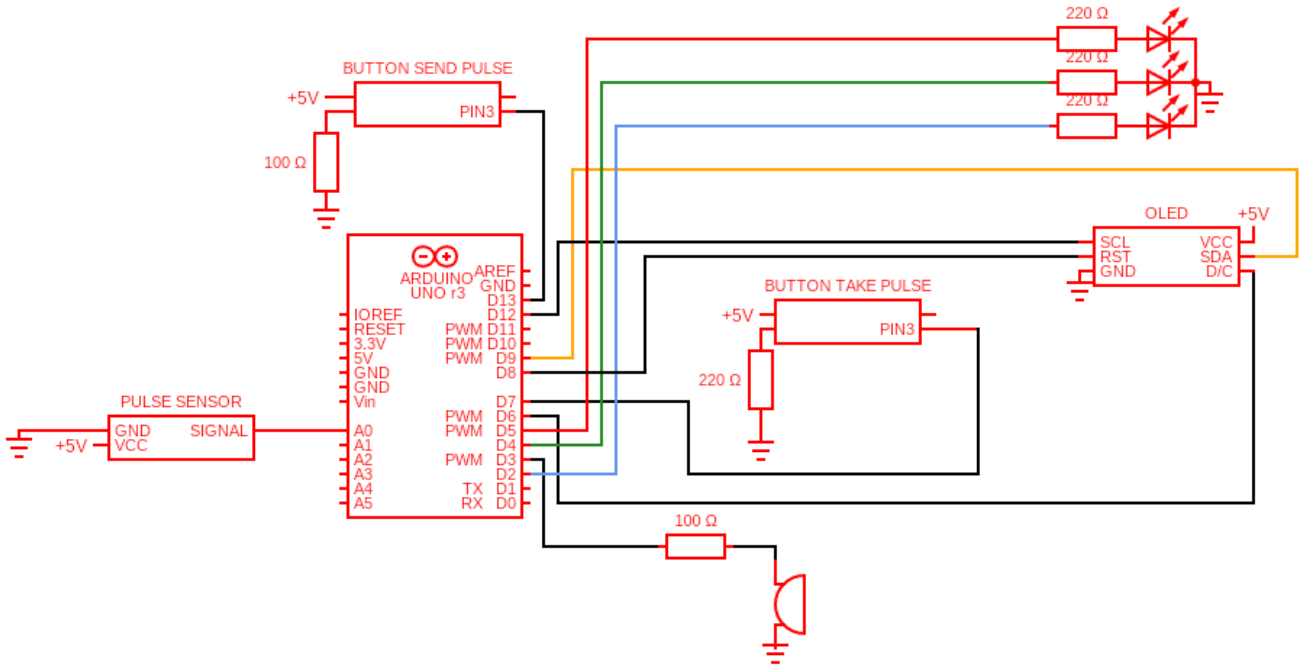
Componente:

1. Placuta Arduino
2. Breadboard
3. Senzor de puls
4. LED RGB
5. Display OLED
6. Buzzer
7. Butoane
8. Fire
9. Rezistente

Schema bloc:



Schema electrica:



Software Design

Mediu de dezvoltare:

1. Dezvoltare cod si incarcarea acestuia pe arduino: **Arduino IDE, PyCharm**
2. Realizare schema bloc: **draw.io**
3. Realizare schema electrica: **circuit-diagram.org**

Biblioteci:

1. Arduino:
 - o Adafruit_GFX, Adafruit_SSD1306
 - o Wire
 - o SPI
 - o PulseSensorPlayground
2. Python:
 - o tkinter
 - o sqlite3

- serial

Descrierea codului:

1. Faza I:

- In acest stadiu, pe ecran se afiseaza mesajul care-l anunta pe utilizator ca trebuie mentinut butonul alb apasat pentru a-si putea masura pulsul, ledul fiind stins deoarece nu se primeste nimic de la senzor

2. Faza II:

- Se ruleaza programul python pentru a putea introduce datele utilizatorului (Nume, Prenume, Varsta) in baza de date
- In momentul in care utilizatorul tine butonul alb apasat, trebuie sa puna degetul pe senzor pentru a putea sa se afiseze pulsul pe ecran
- Dupa preluarea valorilor de la senzorul de puls, se apasa butonul albastru pentru a trimite valorile catre programul din python, unde se va calcula pulsul mediu si se va introduce in baza de date in coloana BPM
- Am adaugat si un buzzer care va scoate un sunet de fiecare data cand culoarea LED-ului se schimba
- LED-ul isi schimba culoarea in functie de valorile pe care le masoara senzorul, astfel:
 - GALBEN: Cand pulsul este sub 60 bpm
 - VERDE: Cand pulsul este intre 60 si 100 bpm
 - ROSU: Cand pulsul trece de 100 bpm
 - ALB: Nu se masoara pulsul

Funcții:

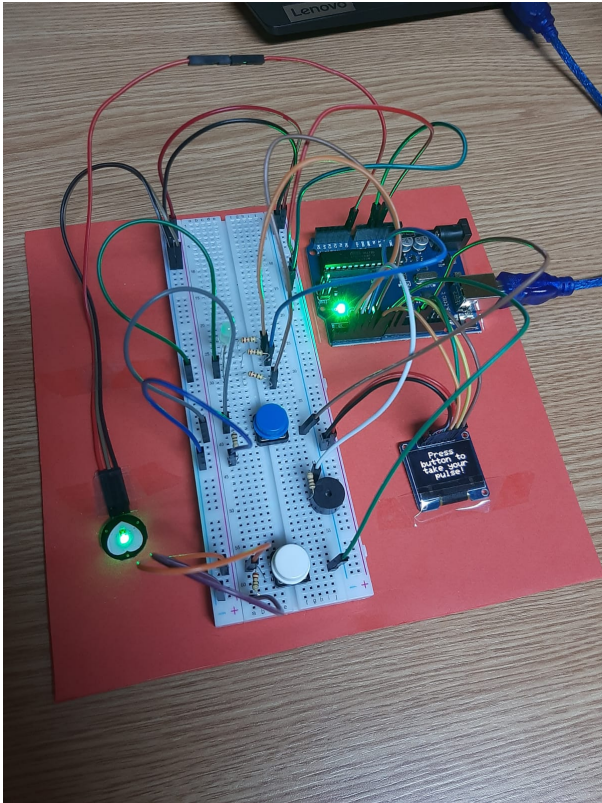
1. Arduino:

- setup() → initializarea display-ului OLED, a senzorului de puls, a butonului si a LED-ului
- RGB_color(red, green, blue) → seteaza valorile pinilor LED-ului pentru a arata culoarea dorita
- loop() → in functie de starea butonului alb se citesc sau nu valorile primite de la senzor, iar pe parcursul apasarii butonului alb se afiseaza valorile pe ecran, se aprinde LED-ul si buzzer-ul este pus in functiune. Cand se apasa butonul albastru se trimit valorile masurate de senzor catre programul python

1. Python:

- Clasa Data, in care am functiile:
 - initPage → adaug detaliile pentru pagina in care sunt introduse datele utilizatorului (Nume, Prenume, Varsta)
 - SaveDetails → creez baza de date, preiau pulsul de la arduino, calculez media lui, apoi inserez in baza de date toate informatiile despre utilizator (Nume, Prenume, Varsta, BPM)
 - main → creez pagina unde se introduc datele utilizatorului

Rezultate Obținute

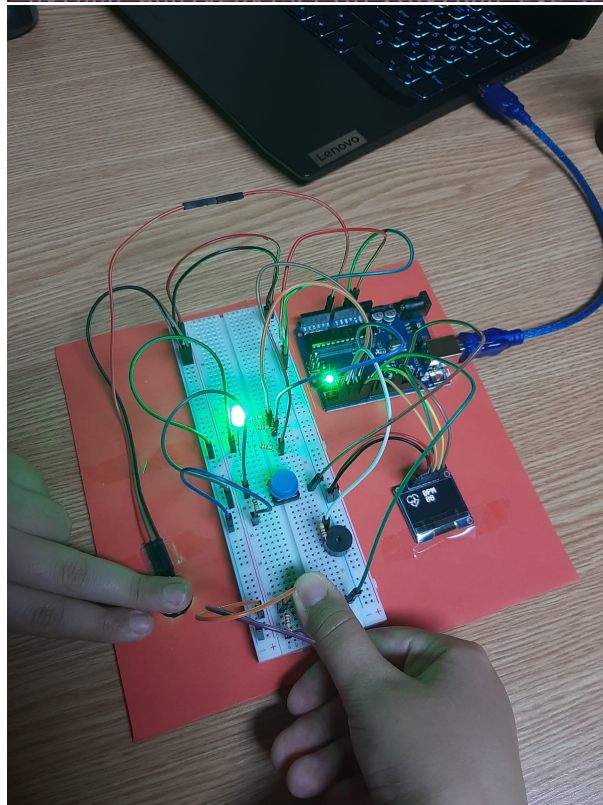
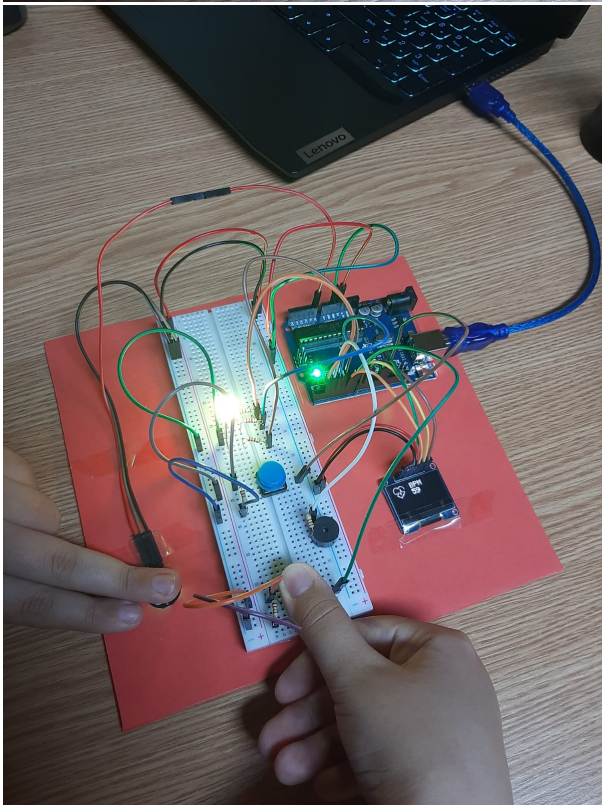


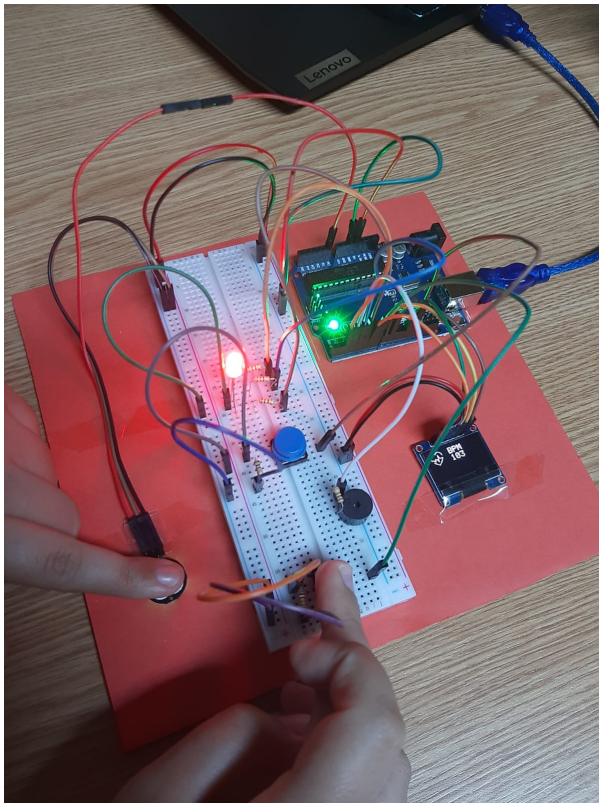
Insert datas

First Name:

Last Name:

Age:





	fname	lname	age	bpm
1	Dumitru	Biriboaca	21	85
2	Ana	Pop	23	79
3	Alina	Maniceanu	21	103
4	Larisa	Patrascu	21	90

Concluzii

In urma realizarii acestui proiect, am invatat ca trebuie sa te documentezi foarte bine in legatura cu utilizarea componentelor, deoarece unele nu pot fi compatibile impreuna. A fost dificil sa conectez display-ul OLED, pentru ca nu mai lucrasem cu un astfel de display.

A fost interesant sa lucrez la acest proiect, pentru ca vedeam cum usor, usor prinde forma tot ce am realizat.

Download

[pulse_meter.zip](#)

<https://github.com/Larisa2307/ProiectPM.git>

Jurnal

- 19.04.2022 - alegere tema proiect
- 02.05.2022 - comandare pise

- 04.05.2022 - sosire piese
- 08.05.2022 - etapa 1 wiki
- 14.05.2022 - conectare senzor puls
- 15.05.2022 - conectare display OLED
- 25.05.2022 - finalizare proiect
- 26.05.2022 - etapa 2 wiki

Bibliografie/Resurse

Pulse Meter

<https://ocw.cs.pub.ro/courses/pm/prj2022/sgherman/pulsemeter>

<https://create.arduino.cc/projecthub/Ingeimaks/diy-heart-rate-sensor-a96e89>

<https://www.engineersgarage.com/arduino-ssd1306-oled-display/>

<https://www.arduino.cc/en/Tutorial/BuiltInExamples/Button>

<https://create.arduino.cc/projecthub/muhammad-aqib/arduino-rgb-led-tutorial-fc003e>

<https://create.arduino.cc/projecthub/SURYATEJA/use-a-buzzer-module-piezo-speaker-using-arduino-un-o-89df45>

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2022/sgherman/pulsemeter>



Last update: **2022/06/02 13:29**