

Fit Tracker □

Autor: [Noaghea Marian-Florentin](#)

Introducere

Descriere

Proiectul reprezinta un tracker pentru numarul de pasi parcursi intr-o zi, si masurarea in timp real a pulsului persoanei. Pentru a vizualiza informatiile poti urmari displayul dispozitivului sau telefonul mobil, acesta avand o conexiune bluetooth, trackerul iti mai poate aduce aminte printr-un semnal sonor sa faci miscare, informatiile referitoare la puls si la numarul de pasi poate fi vazut atat pe telefon cat si pe display.

Utilitate

Consider ca acest dispozitiv este unul bun pentru orice persoana care isi doreste sa duca o viata mai activa si mai sanatoasa.

Schemă bloc



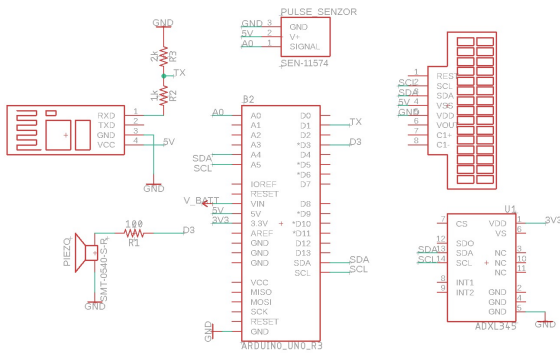
Hardware Design

Componente

Nume	Număr Piese
Arduino Uno	1
Modul Bluetooth HC-05	1
Piezo Buzzer	1
Accelerometru ADXL345	1
Senzor puls	1

Ecran LCD I2C	1
Breadboard	1
Rezistențe	3
Baterie 9V	1
Conector baterie 9V	1

Schema electrică



Software Design

Mediul de dezvoltare

- **Arduino IDE** - scriere cod si incarcare pe arduino
- **Autodesk Eagle** - realizarea schemei electrice
- **draw.io** - realizarea schemei bloc

Biblioteci folosite

- **Wire.h** - comunicarea cu I2C
- **Adafruit_ADXL345_U.h & Adafruit_Sensor.h** - accelerometru ADXL345
- **LiquidCrystal_I2C.h** - comunicare LCD I2C
- **PulseSensorPlayground.h** - senzor puls

Descriere implementare

Pentru calcularea numarului de pasi mai intai iterez un for de cateva 10 de ori, for in care accelerometrul se calibreaza (calculeaza average-ul acceleratiei). In functie de acest average calculez acceleratia la un moment t folosind o formula $\sqrt{\text{Suma}(\text{diff}(\text{acc} - \text{avgacc}))}$ pe toate cele 3 axe. Voi masura acceleratia la 2 momente, t (momentul curent) si t + delay, daca diferenta acceleratiei va fi mai mare decat unul dintre thresholdurile setate de mine numarul pasilor se vor actualiza. Am folosit 2 thresholduri, unul pentru cand s-ar face 2 pasi si unul pentru cand s-ar face 1 pas. Pentru

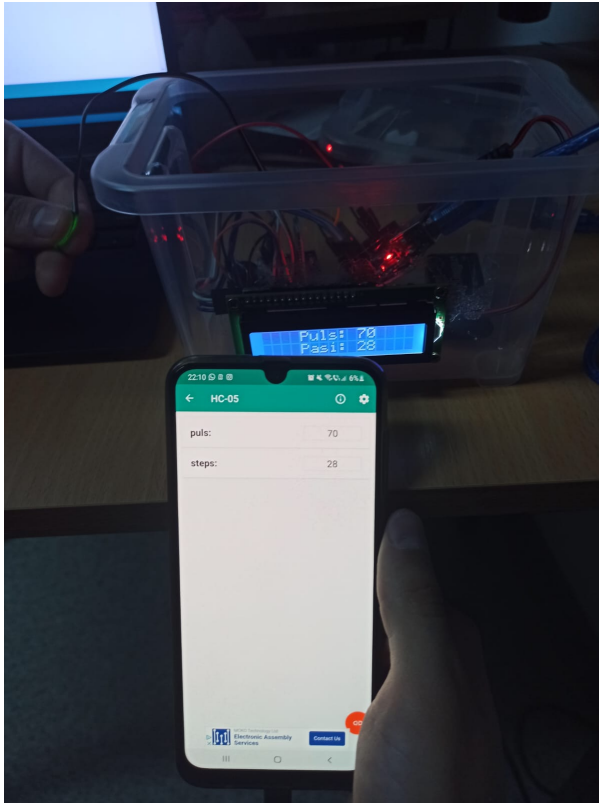
calcularea pulsului folosesc in mare parte de PulseSensorPlayground cu mici ajustari. La fiecare moment de timp printez atat pe LCD (folosind obiectul LiquidCrystal_I2C lcd1) cat si in aplicatia de pe telefon. Pentru fiecare asteptare mai mare de 15 secunde atat buzzerul cat si ecranul LCD vor atentiona ca este momentul sa faci miscare.

Rezultate Obținute

Circuit



Final look



Demo

[Youtube Link](#)

Bibliografie/Resurse

Configurare I2C:

https://www.youtube.com/watch?v=xVCOX_PE_XE

Senzor puls:

<https://pulsesensor.com/pages/getting-advanced>

Pedometru:

<https://www.engineersgarage.com/arduino-based-walking-steps-distance-calculator-adxl345/>

How to Do Timers Correctly:

https://create.arduino.cc/projecthub/Arduino_Scuola/limitations-of-delay-how-to-do-timers-correctly-833e9e

[Export to PDF](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2022/cstan/tracker>



Last update: **2022/05/27 21:26**