

Arduino Guitar Tuner

Autor: Fratila Sergiu-Marius

Grupa: 332CB

Introducere

Dispozitiv care ajuta la acordarea instrumentului (adaptat pentru chitara), inspirat din nevoile fiecarui muzician. Proiectul isi propune citirea datelor analogice sonore cu ajutorul unui microfon, apoi transformarea acestora in sample-uri si analiza fiecarui sample pentru a determina peak-urile si apoi frecventa. Aceasta este comparata cu una din notele de referinta (corzile chitarii), iar apoi pe ecran este afisata sugestia de a tensiona sau de a detensiona coarda chitarii.

Descriere generala

Dispozitivul functioneaza prin intermediul masuratorii frecventei sunetului captat de microfon. Daca frecventa corecta este atinsa, dispozitivul indica acest lucru. Altfel, sugereaza ca nota sa fie ridicata/coborata prin intermediul ecranului LCD.

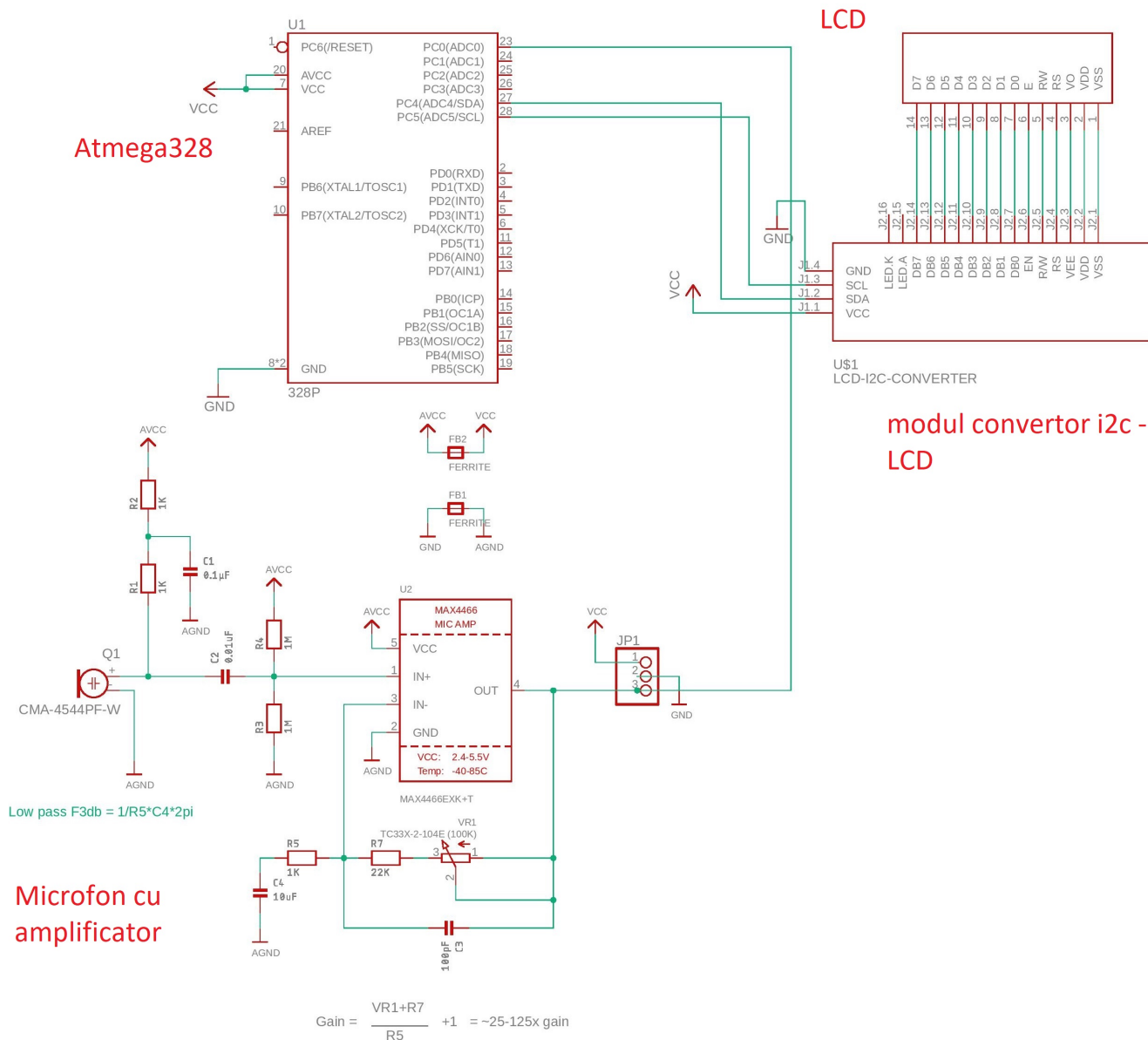
Datele citite cu ajutorul microfonului la fiecare 1ms din pinul analog respectiv reprezinta tensiunea in Volti. Analiza sunetului este facuta astfel: datele discrete sunt stocate intr-un array de x sample-uri, acestea fiind analizate sample cu sample pentru a determina peak-urile. Numarul de peak-uri gasite intr-un array este impartit la x (nr sample-uri) pentru a determina perioada in ms. Se calculeaza inversul acesteia pentru a afla frecventa.

Aceste frecvente obtinute sunt comparate cu unele de referinta (cele 6 note ale corzilor chitarilor), iar in functie de acuratetea notei, pe ecranul LCD va fi sugerata tensionarea/detensionarea corzii ce a fost ascultata.

Schema Bloc



Hardware Design



Lista de Piese

- Arduino UNO
- ecran LCD cu modul I2C [1]
- Microfon cu amplificator [2]
- fire de legatura
- breadboard

[1] <https://cleste.ro/ecran-lcd-1602-iic-i2c.html> [2] <https://www.robofun.ro/sunet/modul-amplificator-microfon-electret-5buc.html>

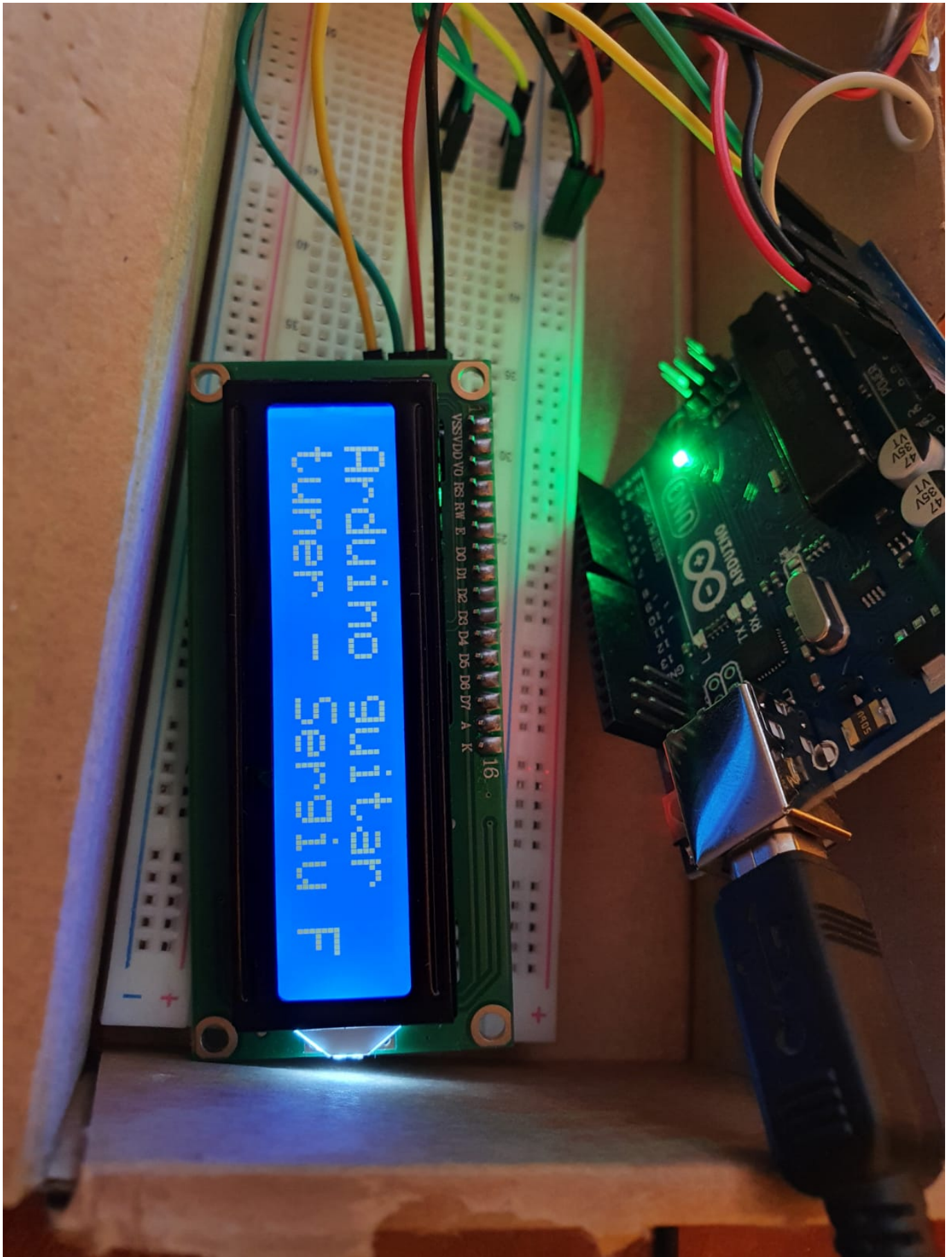
Software Design

- Pentru modulul LCD am folosit biblioteca <LiquidCrystal_I2C.h>
- Pentru stocarea datelor citite de la pinul A0, folosesc un array de 64 de sample-uri. Am incercat initial cu 128, insa avand in vedere ca am nevoie de un array si pentru partea imaginara a semnalului, atunci cand calculez FFT, varianta cu 128 de sample-uri ocupa 75-80% din memoria dinamica a placutei Arduino. Primeam warning-uri de stabilitate, asa ca am coborat la 64.
- Pentru a obtine frecventa sunetului din sample-urile citite de la pinul A0 cu ajutorul microfonului, am folosit cateva functii din biblioteca <arduinoFFT.h>. Fast Fourier Transform este o metoda

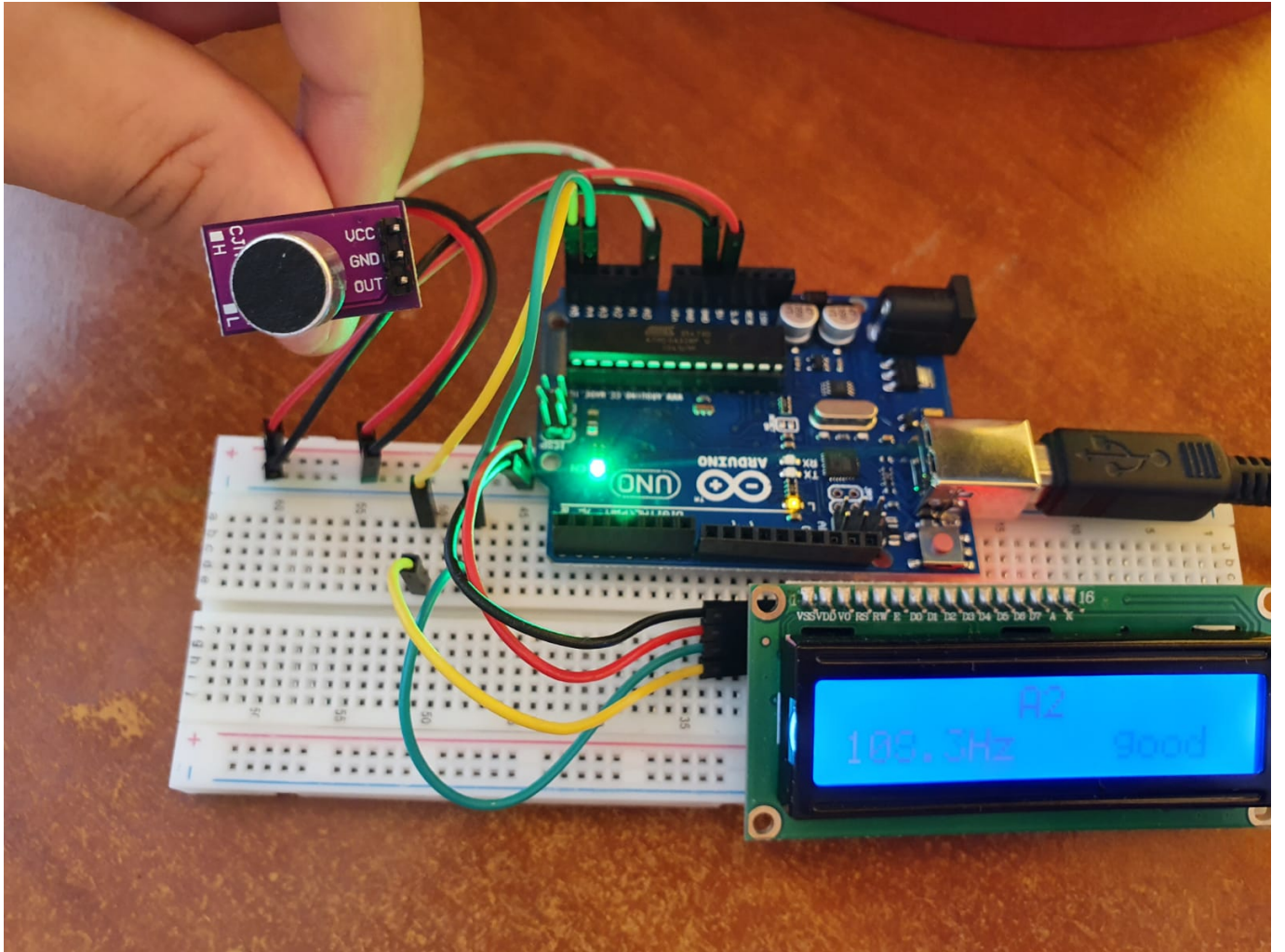
matematica de a analiza componentele de frecventa dintr-un semnal.

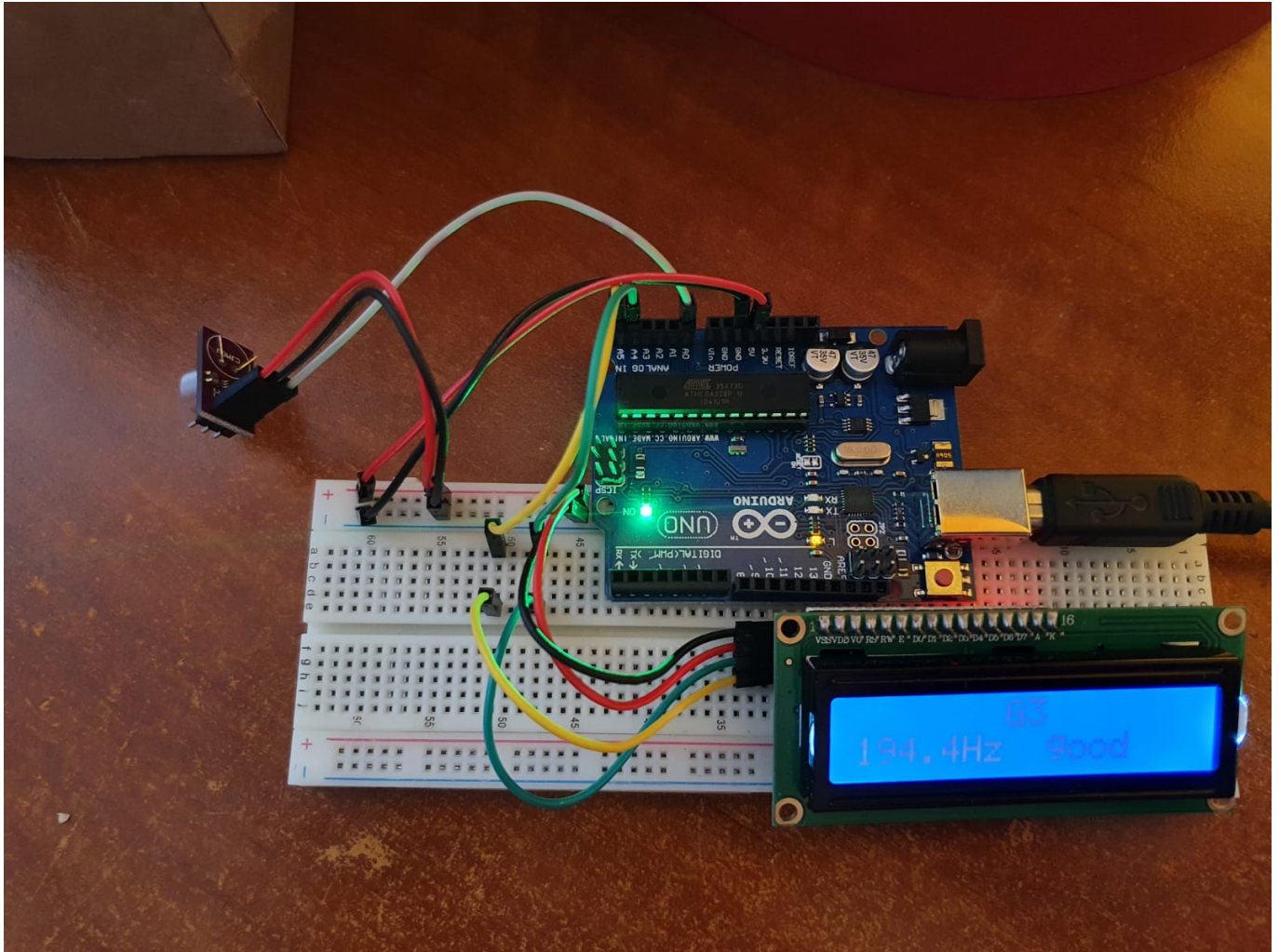
- In loop, frecventa intoarsa de functia FindFrequency(implementata cu FFT) este comparata cu cate un interval ce corespunde fiecărei corzi de chitara.
- Astfel, acordul meu este capabil sa recunoasca fiecare coarda din acordajul standard al unei chitare (E A D G B e).
- Dupa ce incadreaza frecventa gasita intr-unul dintre intervalele specifice corzilor, este verificat daca sunetul este mai inalt sau mai jos decat un alt interval mai restrans, care ar insemna ca acea coarda este acordata.
- In functie de cum se incadreaza in intervalele: mai mic, perfect, mai mare, ecranul LCD afiseaza "< <" pentru un sunet cu frecventa prea mica, "> >" pentru frecventa prea mare si "good" pentru cazul in care este acordata.
- Pe masura ce am scris codul ce incadra frecventa intr-un interval pentru fiecare coarda, am incercat sa calibrez indicatiile tunerului meu in functie de alte 2 acordoare.

Rezultate Obtinute



• Fig. 1: Mesaj de intampinare.





• Fig. 2, 3: Montaj complet.

[Demonstratie video](#)

Concluzii

- Implementarea acestui proiect nu a fost nici mai grea, nici mai usoara decat ma asteptam.
- A trebuit sa fac ceva research legat de analiza frecventei / alte proiecte arduino guitar tuner astfel incat sa aflu ca aveam nevoie de FFT pentru a obtine frecventa dintr-un sample.
- Desi Arduino nu este cea mai buna placuta pentru procesare audio, sunt multumit de rezultatul final si de acuratetea tunerului. In videoul demonstrativ se vede faptul ca este utilizabil ...cu toate ca mai da mici rateuri, asa fac si tunerele profesionale, in experienta mea:))

Jurnal

- **7 mai** - achizitionare componente + implementare afisare LCD
- **14 mai** - implementare beta (fara FFT) didn't work :(
- **21 mai** - research FFT + aflare frecventa in Hz
- **21-27 mai** - implementarea interpretarii frecventelor citite + afisare sugestii acordare + finalizare

Bibliografie/Resurse

[Arduino Frequency Detection Fourier Transform Arduino Guitar Tuner \(instructables\)](#)

Descarcare pagina/arhiva surse

Click [aici](#) pentru descarcarea PDF a paginii curente.

[Arhiva cu codul sursa](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2021/avaduva/guitartuner>



Last update: **2021/05/28 08:01**