

# Mini-Tutorial pentru pian

## Introducere

**Proiectul** constă în implementarea unui **dispozitiv** ce facilitează *învatarea și recunoașterea* notelor muzicale din gama **Do Major**.

Se vor folosi **7 butoane**, fiecare asociat cu o notă muzicală. Vor exista **5 nivele** de dificultate din care utilizatorul poate alege. După redarea unei secvențe de note muzicale prin **buzzer**, utilizatorul are la dispoziție un interval prestabilit de timp pentru a reda secvența corect. Dispozitivul oferă o modalitate *interactivă și distractivă* de a învăța notele muzicale, utilizatorul fiind îndrumat de **led-urile luminoase** și **ecranul LCD** în acest proces.

## Descriere generală

### Schema Bloc



### Mod de funcționare

Inițial, utilizatorul va avea 10 secunde pentru a alege cu ce nivel de dificultate dorește să continue.

**Buzzer-ul** va reda o secvență de note muzicale, iar ulterior, utilizatorul va avea la dispoziție un interval de timp măsurat cu ajutorul unui **timer** pentru a reproduce "melodia" auzită. În cazul în care timpul expiră sau un buton greșit este apăsat, pe **ecranul LCD** se va afișa un mesaj sugestiv, **led-urile** vor deveni roșii, iar nivelul selectat se va relua.

Utilizatorii vor fi ghidați de către o **bandă led** ce va avea câte un led asociat fiecărui buton, iar acesta se va aprinde în cazul în care nota muzicală ce trebuie redată corespunde butonului respectiv. Pentru a îmbunătăți procesul de *recunoaștere și învățare* al notelor muzicale, la fiecare buton apăsat, numele notei muzicale careia îi corespunde va fi afișat pe ecranul LCD.

Odată ce secvența va fi reprodusă corect, **led-urile** devin verzi și se oferă posibilitatea trecerii la un alt nivel, în cazul în care se dorește continuarea folosirii dispozitivului.

# Hardware Design

## Lista piese:

- Arduino UNO
- Ecran LCD 12C
- Buzzer
- 7 butoane redare note muzicale
- 1 buton alegere nivel
- Banda LED WS2812
- Breadboard
- Fire

## Schema Electrica



# Software Design

- Mediu de dezvoltare : Arduino IDE
- Librarii importate : LiquidCrystal\_I2C.h pentru comunicarea cu ecranul LCD, Adafruit\_NeoPixel.h pentru adresarea bandei led, Arduino Tone Library pentru redarea notelor muzicale

## Detalii de implementare

In **setup()** am configurat pinii de input (butoanele corespunzatoare notelor muzicale si butonul corespunzator nivelului) si pinul de output (buzzer-ul), LCD-ul si banda led.

Am implementat functii separate pentru a *modulariza* codul, intrucat rationamentul folosit este identic in cele 5 nivele. Functiile **playTone()** si **playNote()** sunt responsabile pentru redarea unui ton pe buzzer, respectiv a unei note muzicale. Cu ajutorul functiei **playNivel()** se va reda secventa corespunzatoare nivelului selectat, aceasta primind ca input o serie de caractere care definesc notele muzicale si durata lor.

Functia **sing()** este responsabila cu functionalitatea "pianului", intrucat reda frecventa notei muzicale asociata butonului apasat cu ajutorul functiei **tone()** si afiseaza pe ecranul LCD denumirea acesteia.

Functiile **succesLed()**, **failedLed()** si **clearLed()** confera un feedback vizual utilizatorului referitor la redarea secventei auzite.

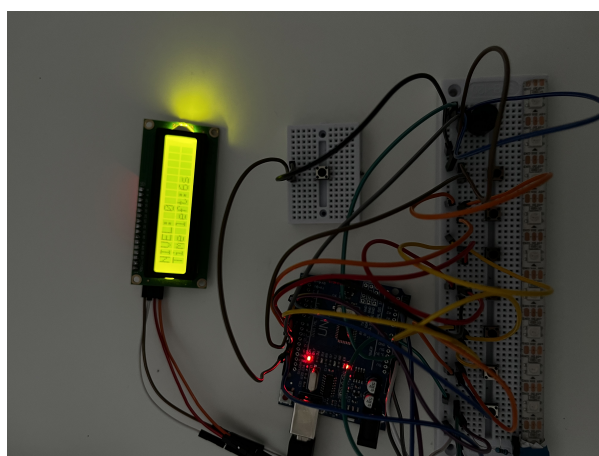
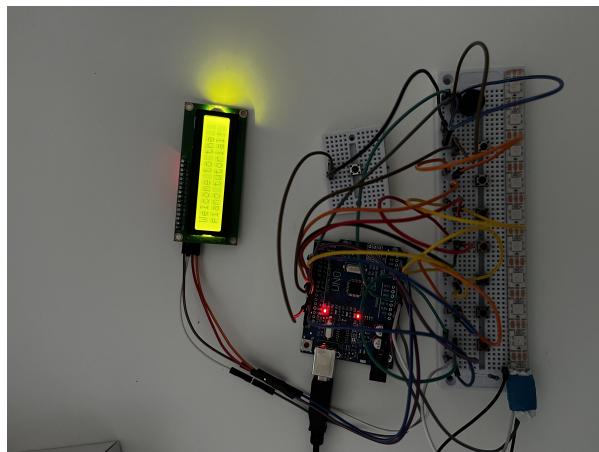
Funcția **anotherButton()** primește drept argument indicele butonului ce trebuie apasat și verifică dacă un alt buton este apasat în schimb.

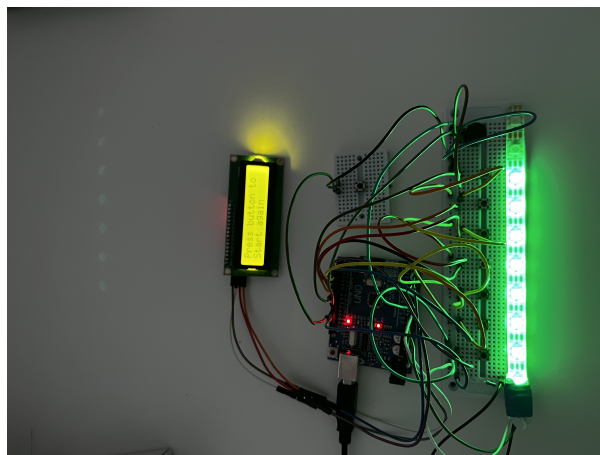
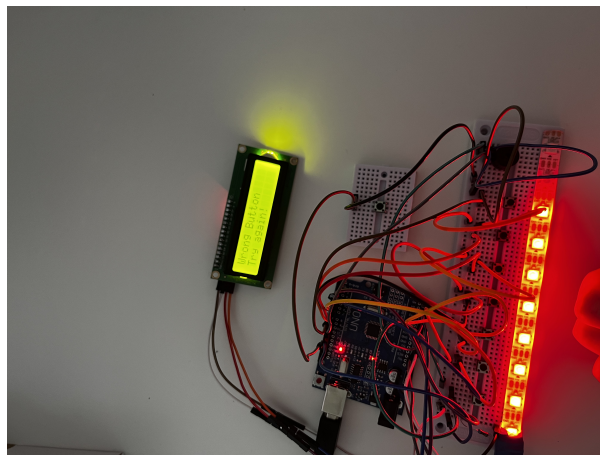
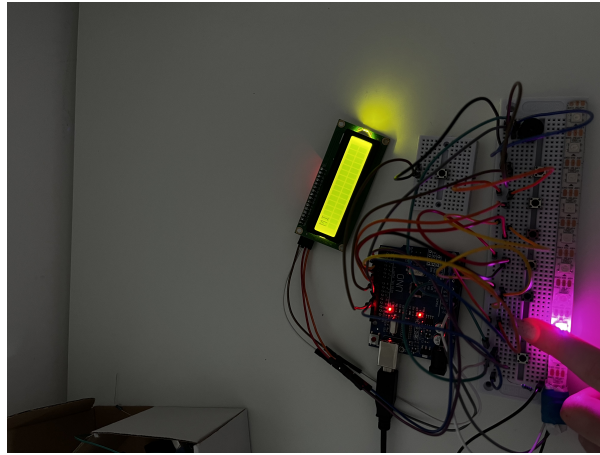
Intrucât folosim un timer pentru a măsura timpul de reproducere a melodiei, funcția **setupTimer()** se va ocupa de configurarea **Timer1**, iar **ISR(TIMER1\_COMPA\_vect)** va fi rutina de întrerupere este declanșată atunci când **timer-ul** atinge valoarea specificată în **registru OCR1A**.

În funcția **loop()** este implementat flow-ul programului:

1. Mesaj de bun-venit afișat pe **ecranul LCD**
2. Alegerea nivelului într-un timp de 10 secunde ce va fi măsurat folosind funcția **millis()**. La fiecare apăsare a butonului, nivelul va fi incrementat și utilizatorul va putea vedea cât timp mai are la dispoziție pentru alegere.
3. Redarea melodiei corespunzătoare nivelului
4. Îndrumarea utilizatorului pentru a reproduce secvența auzită și feedback-ul în funcție de corectitudinea rezultatului.

## Rezultate Obținute





## Concluzii

Ma bucur de alegerea facuta in legatura cu tema proiectului, deoarece mi-a facut placere sa il realizez, atat pe partea de hardware, cat si pe partea de software.

Pentru realizarea hardware-ului am urmarit diverse tutoriale pe youtube, care mi-au fost de mare ajutor, iar pentru scrierea codului am incercat sa implementez initial pe module (functionalitatea pianului, functionalitatea ecranului LCD, redarea melodiilor, adresarea bandei led), iar apoi le-am imbinat pentru a avea functionalitatea dorita. Consider ca am creat un dispozitiv interactiv, distractiv

si usor de utilizat datorita mesajelor sugestive afisate pe ecranul LCD si a feedback-urilor luminoase oferite de banda led.

## Download

[mini-tutorial\\_pentru\\_pian.zip](#)

## Jurnal

Puteți avea și o secțiune de jurnal în care să poată urmări asistentul de proiect progresul proiectului.

## Bibliografie/Resurse

### Resurse Software

- <https://gist.github.com/baojie/4522173>
- <https://www.youtube.com/watch?v=CvZ-SGJ8fGo>
- <https://ocw.cs.pub.ro/courses/pm/lab/lab3-2023>
- <https://forum.arduino.cc/t/using-millis-for-timing-a-beginners-guide/483573>
- <https://arduinogetstarted.com/tutorials/arduino-neopixel-led-strip>

### Resurse Hardware

- <https://www.youtube.com/watch?v=CvZ-SGJ8fGoo>
- <https://www.youtube.com/watch?v=EAeuxjtkumM>

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

Permanent link:  
<http://ocw.cs.pub.ro/courses/pm/prj2023/danield/mini-tutorial-pentru-pian>



Last update: **2023/05/29 12:09**