

# Pian Electric

**Autor:** [Emilian Mănescu](#)

## Introducere

Proiectul presupune crearea unui pian electric capabil sa redea notele din gama Do Major.

Scopul acestuia este de a-l ajuta pe utilizator sa cante melodii simple si de a oferi un prim pas in invatarea unui instrument muzical.

Consider ca proiectul este util intrucat permite exersarea conceptelor de baza despre pian intr-un mod practic si la un pret redus.

## Descriere generala

Pianul va avea opt butoane corespunzatoare notelor din gama Do Major. Acestea vor fi redade prin intermediul unui buzzer.

Utilizatorul poate alege intre a canta la pian in mod liber, a reda melodiile din memoria pianului sau a le invata.

Pianul va avea ca interfata cu utilizatorul un ecran LCD pe care vor fi afisate melodiile disponibile. La alegerea unei melodii prin apasarea unui buton, notele ce trebuie cantate vor aparea pe ecran, una cate una. Un LED va semnala daca nota a fost cantata corect sau nu, dupa un cod de culori (rosu = gresit, verde = corect).

De asemenea, exista un buton pentru redarea melodiei selectate, pentru a oferi utilizatorului un exemplu de interpretare a melodiei.

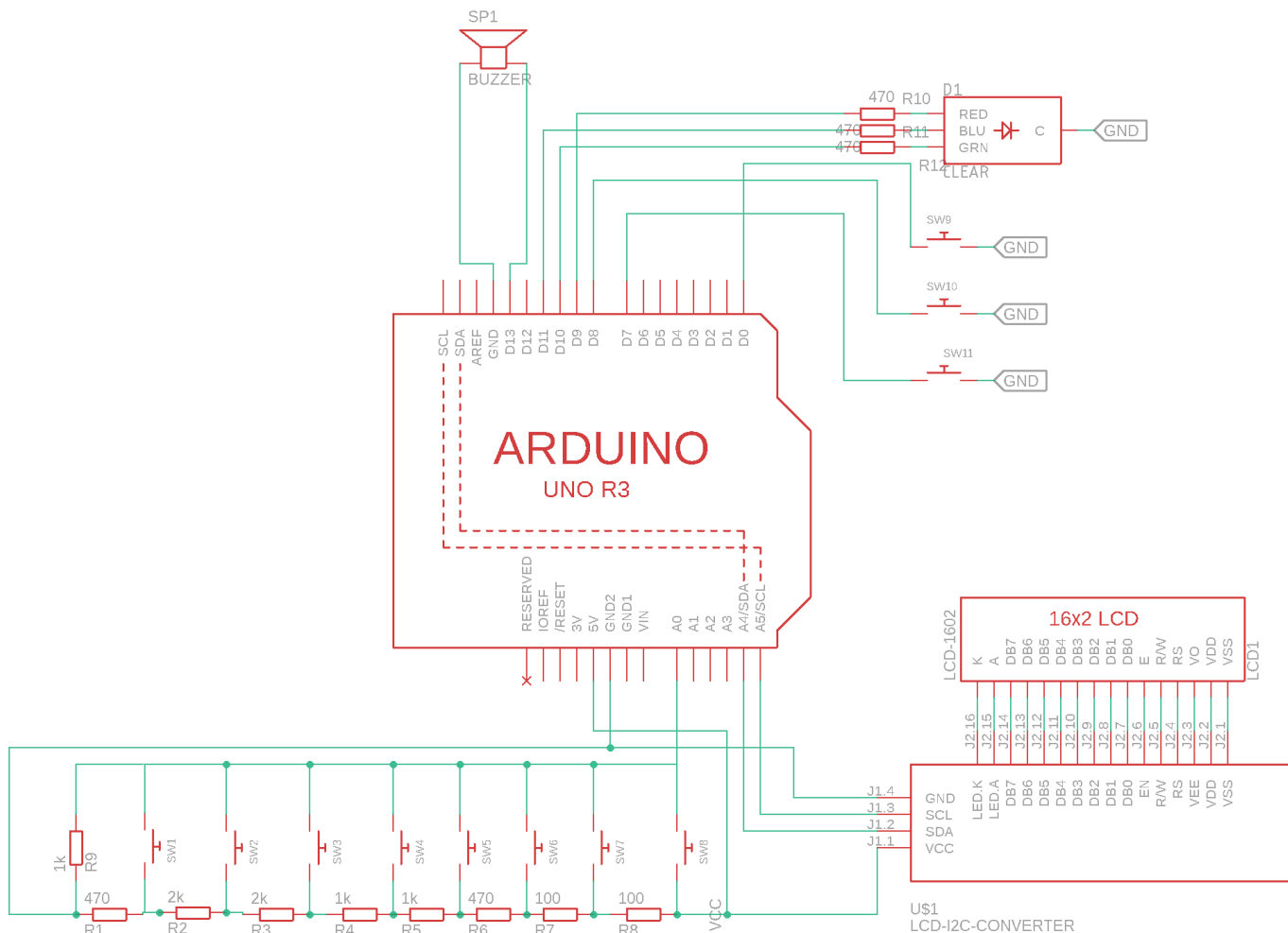
## Schema bloc:



## Hardware design

### Lista de piese:

- Arduino UNO
- Buzzer
- Modul LCD 1602
- Adaptor I2C
- Rezistente
- Butoane
- LED RGB
- Breadboard
- Fire



## Software design

Proiectul a fost realizat in Arduino IDE. Pentru comunicarea cu ecranul LCD, am folosit bibliotecile Wire si LiquidCrystal\_I2C.

## Descriere implementare

- functia **RGB\_color** coloreaza LED-ul RGB intr-o anumita culoare specificata (rosu sau verde)

- functia **setup** initializeaza pinii, ecranul LCD si LED-ul.
- functia **playNote** alege nota ce va fi redata de catre buzzer.
- functia **loop** cuprinde logica generala a programului, in care se poate schimba cantecul afisat pe ecran, se poate alege redarea unei melodii sau invatarea acesteia.
- functia **playCurrentSong** apeleaza functia de redare (**playSong**) pentru melodia afisata pe ecran.
- functia **getNoteString** returneaza un string reprezentand nota ce va fi afisata pe ecran in modul de invatare.
- functia **notCorrect** verifica daca nota apasata de utilizator si cea afisata pe ecran coincid.
- functia **learnCurrentSong** incepe procesul de invatare a melodiei afisate pe ecran.
- functia **learnSong** verifica daca toate notele unei melodii au fost cantate corect de utilizator.
- fisierul **pitches.h** cuprinde frecventele tuturor notelor.
- fisierul **songs.h** cuprinde informatii despre cele trei melodii disponibile (notele, durata notelor, numarul de note si tempo-ul de redare).

## Rezultate obtinute

[Demo](#)

## Concluzii

Mi-a placut sa lucrez la acest proiect, in special pentru ca am reusit sa imbin doua dintre domeniile care ma pasioneaza cel mai mult (tehnologia si muzica). Partea de hardware (desi nu foarte complicata) mi s-a parut mai greu de realizat, deoarece nu am mai lucrat niciodata la un proiect practic. Cel mai greu a fost sa realizez conectarea butoanelor pentru clape la un singur pin analogic (prin divizoare de tensiune). Totodata, imi doresc sa imbunatatesc acest proiect pe viitor si sa adaug functionalitati extra.

## Download

[Arhiva Proiect](#)

## Bibliografie/Resurse

- <https://lastminuteengineers.com/i2c-lcd-arduino-tutorial/>
- <https://www.the-diy-life.com/multiple-push-buttons-on-one-arduino-input/>
- <https://howtomechatronics.com/tutorials/arduino/how-to-use-a-rgb-led-with-arduino/>
- <https://create.arduino.cc/projecthub/rahulkhanna/arduino-tutorial-mini-piano-08f8b8>

From:

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

Permanent link:

[http://ocw.cs.pub.ro/courses/pm/prj2021/amocanu/pian\\_electric](http://ocw.cs.pub.ro/courses/pm/prj2021/amocanu/pian_electric)

Last update: **2021/06/02 19:24**

