

# Music Player

Nume: Panțucu Ștefan-Alexandru

Grupa: 333CB

## Introducere

### Prezentare

Proiectul consta intr-un music player minimalist, care va reda melodiile dintr-o selectie atent aleasa, prezenta pe un card SD. Titlul șlagărului in derulare poate fi vizualizat pe un ecran LCD, iar daca dorim sa ascultam hitul urmator sau sa il reauzim pe cel anterior, avem posibilitatea cu ajutorul butoanelor de next/prev. Pentru cand petrecerea ia o scurta pauza, exista si un buton de pause.

### Ideea

Nestiind ce as putea face ca proiect, am decis sa imi iau inspiratia din hobby-urile mele, iar muzica este unul dintre ele. Asadar m-am gandit ca un music player ar fi ideea perfecta pentru proiect.

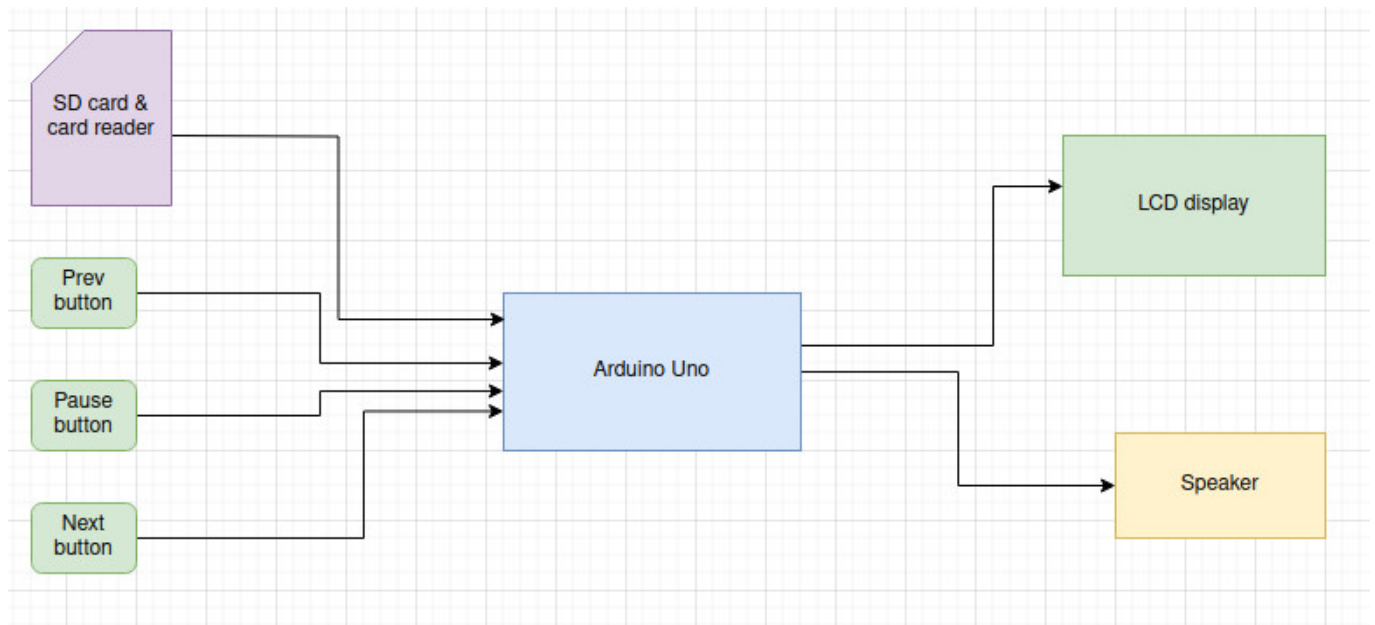
### Utilitate

Pot asculta melodiile preferate oriunde, de pe un dispozitiv creat de mine, chiar si fara internet (util daca nu am Spotify premium). In plus, faptul ca melodiile sunt citite de pe un card SD face foarte usoara tranzitia intre playlist-uri, putandu-se schimba cardul.

## Descriere generală

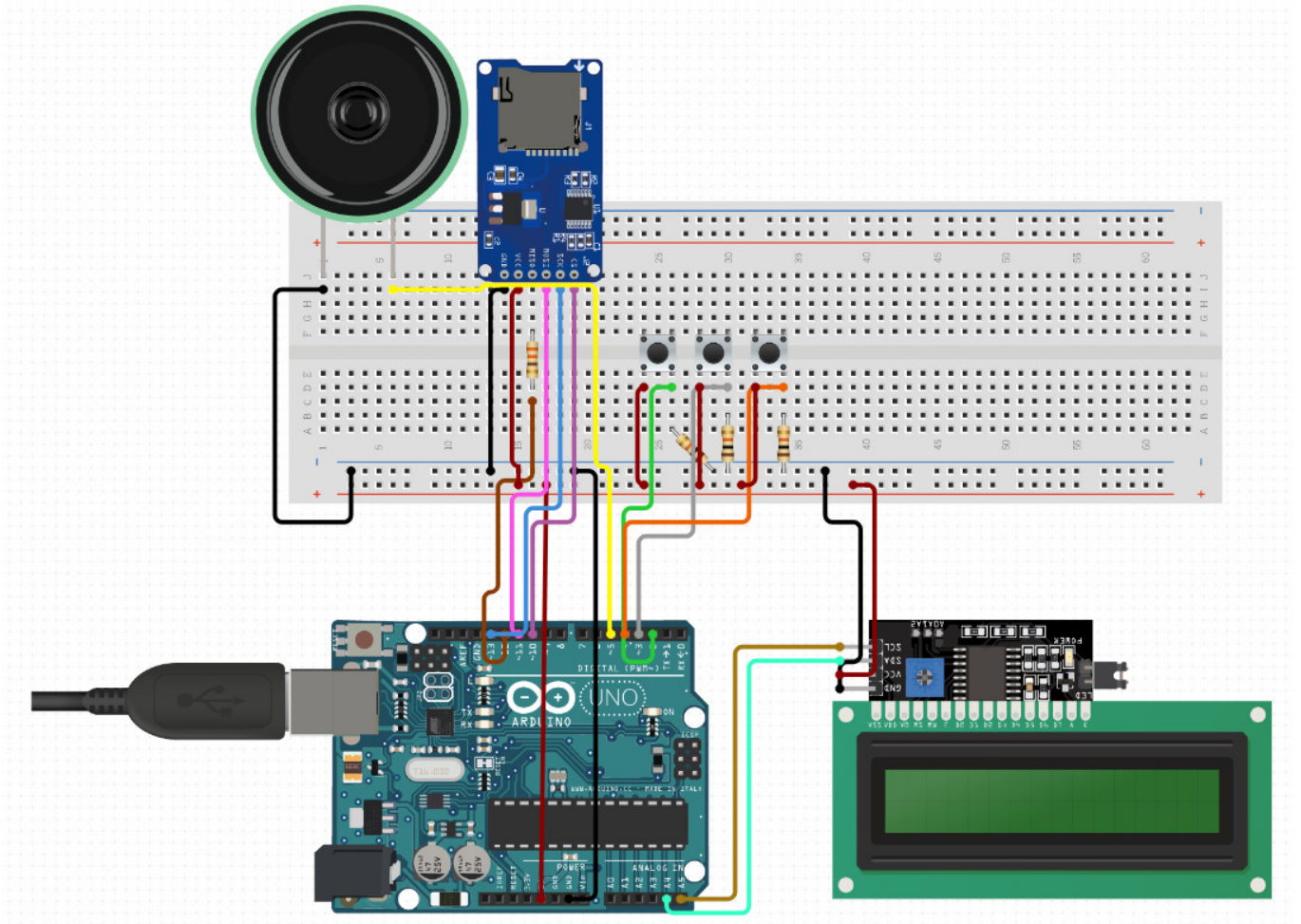
Playerul de muzica porneste atunci cand este conectat la o sursa de curent, incepand cu prima melodie aflata pe cardul SD. Din acest punct incepe distractia, putandu-ne plimba printre melodii si sa punem pauza oricand dorim. De mentionat faptul ca melodiile trebuie sa fie in format .wav.

## Schema Bloc

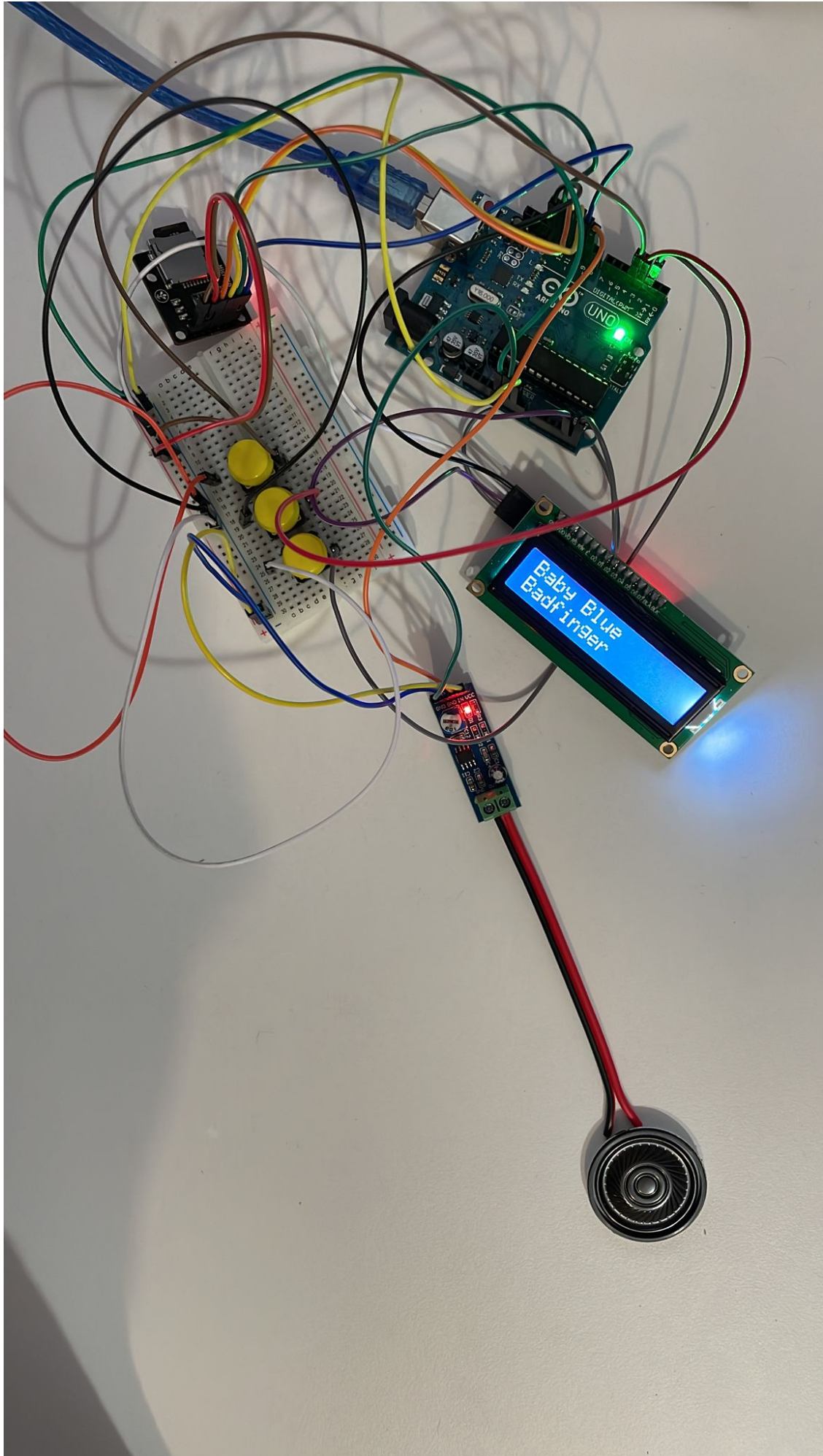


## Hardware Design

- Arduino Uno
- Butoane
- Modul difuzor cu amplificator
- Ecran LCD cu modul I2C
- SD card reader



**Rezultat final:**



# Software Design

## Mediul de dezvoltare

Pentru scrierea codului necesar implementarii proiectului am folosit Arduino IDE.


## Biblioteci folosite

- LiquidCrystal\_I2C → pentru comunicarea cu ecranul LCD prin intermediul modulului I2C
- TMRpcm → pentru speaker
- SD → pentru cardul SD
- SPI → pentru cardul SD

## Modul de functionare

In metoda de setup se initializeaza pinii folositi pentru butoane, se initializeaza citirea de pe cardul SD, selectand pin-ul de CS folosit, se citeste continutul din radacina cardului SD pentru a determina numarul total de melodii, se initializeaza speaker-ul pe pinul ales, cat si volumul si calitatea, iar apoi se initializeaza si ecranul LCD, urmand sa se afiseze un mesaj de bun venit. Dupa 3 secunde, prima melodie disponibila va incepe sa cante.

In loop se citeste inputul pe fiecare din cele 3 butoane si se actioneaza in conformitate cu actiunea ce a avut loc (se trece la melodia anterioara, se pune pauza sau se reia redarea sau se trece la melodia urmatoare). In cazul in care nicio melodie nu este in desfasurare sau melodia curenta s-a terminat, se va trece la melodia urmatoare.

Melodiile sunt stocate pe card in formatul <index>.wav,  $0 \leq \text{index} < \text{numarul de melodii}$ . Deoarece citirea de pe card functioneaza doar daca file systemul folosit este FAT32, care permite ca numele fisierelor sa aiba cel mult 8 caractere , mi-am dat seama ca daca vreau sa afisez numele artistului si al melodiei, trebuie sa gasesc o alta solutie. Astfel, fiecarei melodii ii corespunde un fisier <index>.in in care se gaseste pe primul rand numele melodiei, iar pe al doilea numele artistului. In momentul in care se apeleaza functia playSong(index), mai intai este apelata functia displaySong(index), care compune numele fisierului de intrare pentru a citi datele despre melodie de pe card, pentru a le putea afisa pe ecran. Apoi, revenind in playSong, se compune numele melodiei si se apeleaza metoda play() cu numele fisierului.

## Extra


Am scris un script de bash care ia toate fisierele cu extensia .mp3 si folosind ffmpeg, le converteste in format .wav pentru a putea fi redade de catre arduino prin speaker. Acest script creeaza si fisierele de

input cu datele despre melodie. Singura conditie inainte de rulara scriptului este ca melodiile sa fie in formatul "NumeMelodie;NumeArtist.mp3" pentru ca datele sa fie afisate corect pe ecranul lcd.

## Rezultate Obținute

<https://youtu.be/gFPt4ypNxlk>

## Concluzii

A fost o experienta interesanta sa lucrez la acest proiect si chiar daca poate nu este cel mai complicat, a fost rewarding sa il vad terminat. Am trecut prin diverse probleme de-a lungul implementarii, de la formatul potrivit atunci cand converteam melodiile din mp3 in wav, la probleme cu bouncing-ul la butoane sau faptul ca am crezut ca nu functioneaza modulul pentru cardul SD, dar de fapt pinii erau scrisi gresit pe placuta .

O provocare a fost afisarea datelor despre melodie pe ecran, data fiind limitarea provocata de FAT32, inasa cred ca am gasit un workaround destul de dragut (descriu mai sus), care te scuteste de introducerea manuala a datelor in prealabil.

## Download

stefan\_pantucu\_333cb\_proiect\_pm.zip

## Jurnal

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

## Bibliografie/Resurse

<https://reference.arduino.cc/reference/en/libraries/liquidcrystal-i2c/>

<https://reference.arduino.cc/reference/en/libraries/tmrpcm/>

<https://www.arduino.cc/reference/en/libraries/sd/>

[Export to PDF](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2023/drtranca/stefan.pantucu>



Last update: **2023/05/28 19:49**