

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.

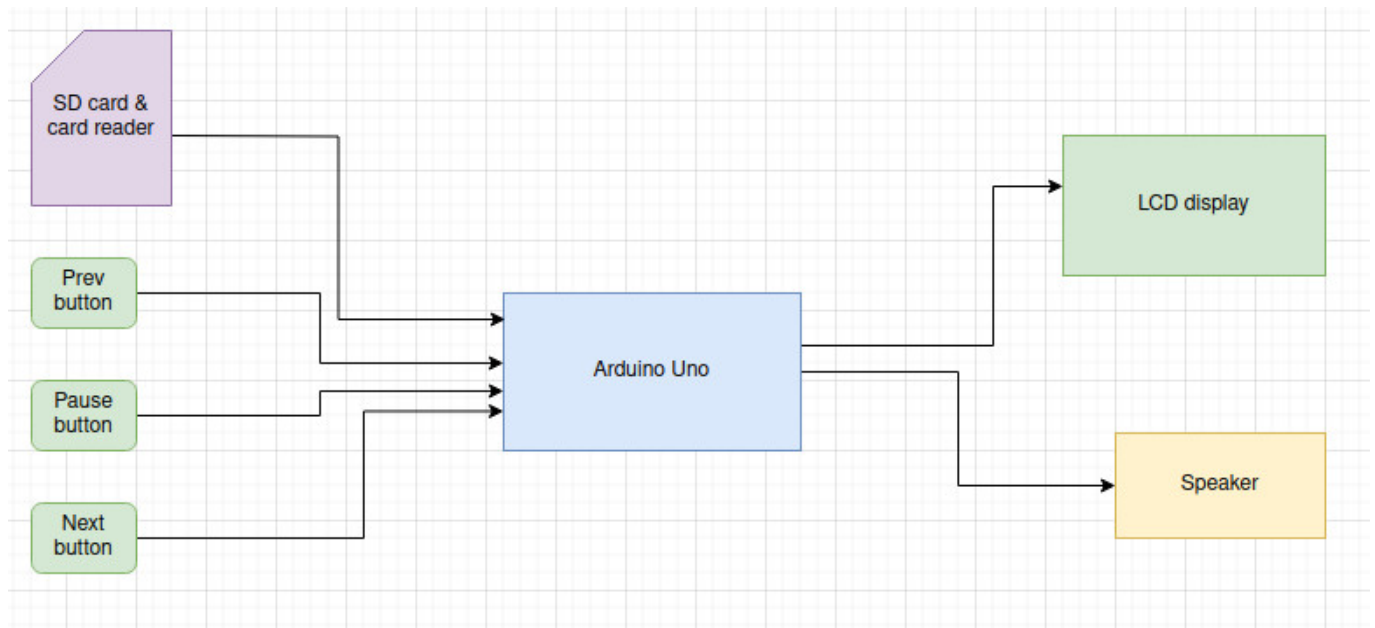
Utilitate

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

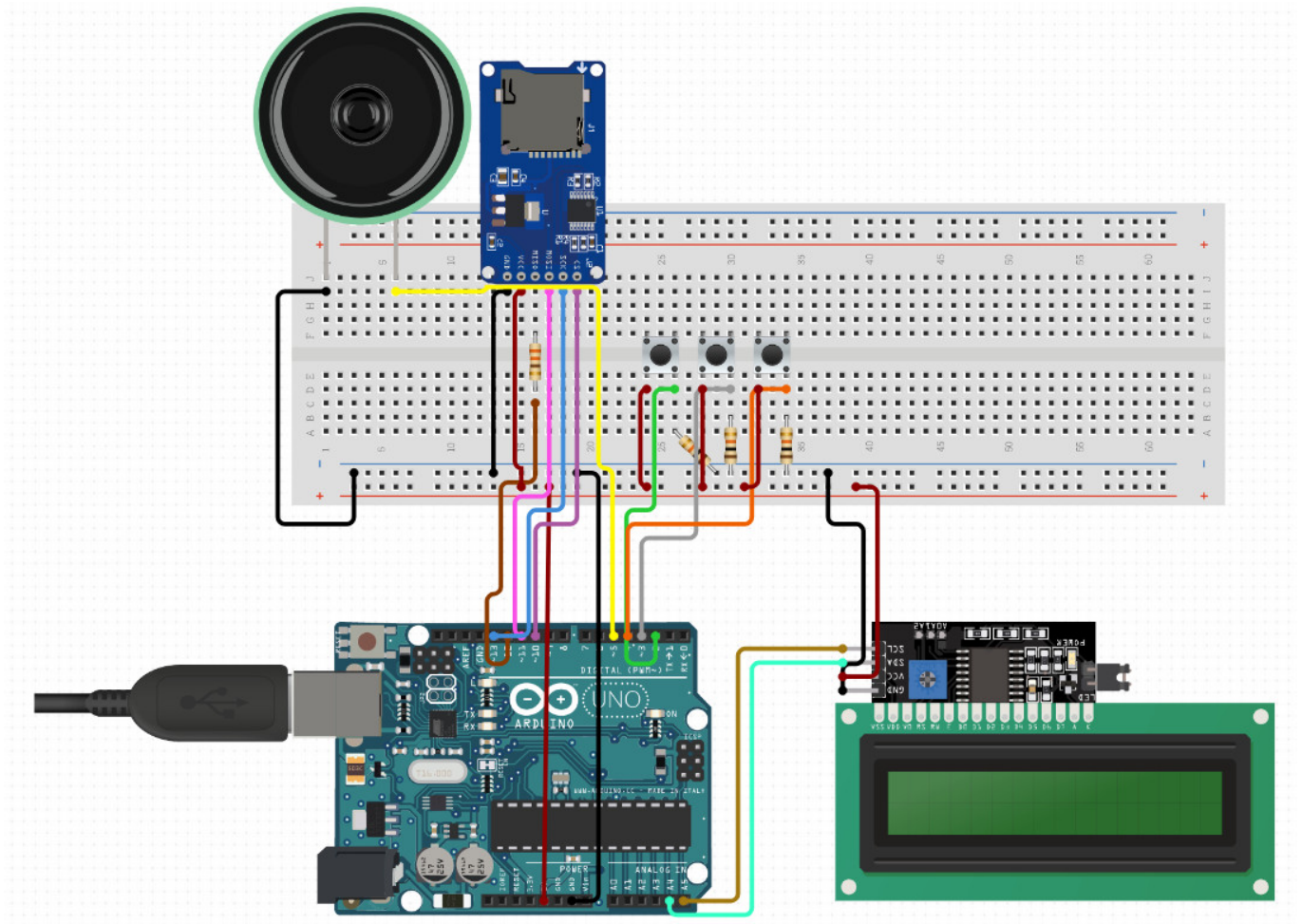
Playerul de muzica porneste atunci cand este conectat la o sursa de curent, incepand cu prima melodie aflata pe cardul SD. Din acest punct incepe distractia, putandu-ne plimba printre melodii si sa punem pauza oricand dorim. De mentionat faptul ca melodiile trebuie sa fie in format .wav.

Schema Bloc

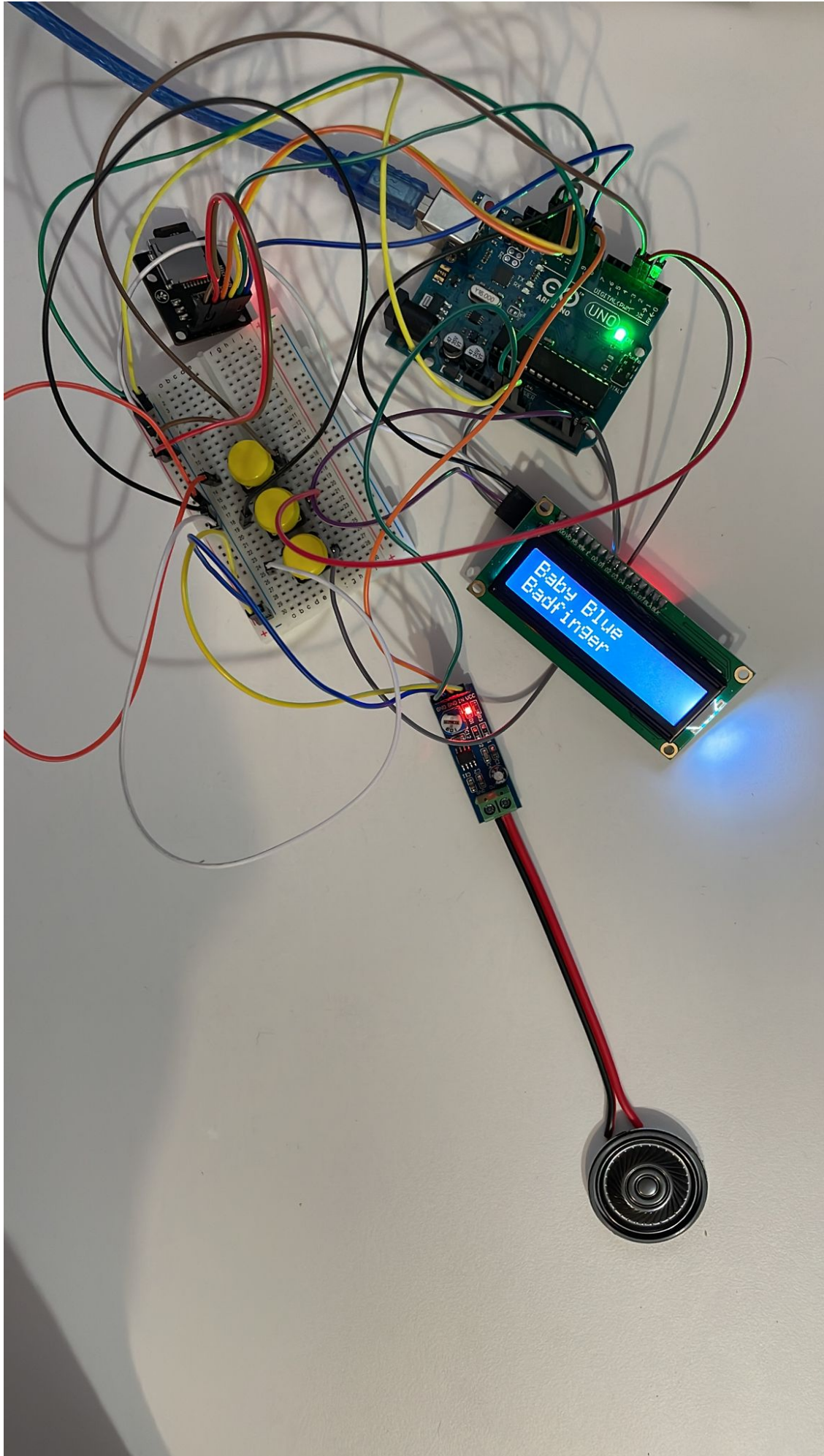


Hardware Design

- 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

În metoda de setup se initializează pinii folosiți pentru butoane, se inițializează citirea de pe cardul SD, selectând pin-ul de CS folosit, se citește conținutul din rădăcina cardului SD pentru a determina numărul total de melodii, se inițializează speaker-ul pe pinul ales, cât și volumul și calitatea, iar apoi se inițializează și ecranul LCD, urmând să se afișeze un mesaj de bun venit. După 3 secunde, prima melodie disponibilă va începe să cânte.

În loop se citește inputul pe fiecare din cele 3 butoane și se acționează în conformitate cu acțiunea ce a avut loc (se trece la melodia anterioară, se pune pauză sau se reia redarea sau se trece la melodia următoare). În cazul în care nicio melodie nu este în desfășurare sau melodia curentă s-a terminat, se va trece la melodia următoare.

Melodiile sunt stocate pe card în formatul <index>.wav, $0 \leq \text{index} < \text{numărul de melodii}$. Deoarece citirea de pe card funcționează doar dacă fișierul folosit este FAT32, care permite ca numele fișierelor să aibă cel mult 8 caractere , mi-am dat seama că dacă vreau să afișez numele artistului și al melodiei, trebuie să găsesc o altă soluție. Astfel, fiecărei melodii îi corespunde un fișier <index>.in în care se găsește pe primul rând numele melodiei, iar pe al doilea numele artistului. În momentul în care se apelează funcția playSong(index), mai întâi este apelată funcția displaySong(index), care compune numele fișierului de intrare pentru a citi datele despre melodie de pe card, pentru a le putea afișa pe ecran. Apoi, revenind în playSong, se compune numele melodiei și se apelează metoda play() cu numele fișierului.

Extra

Am scris un script de bash care ia toate fișierele cu extensia .mp3 și folosind ffmpeg, le convertește în format .wav pentru a putea fi redat de către arduino prin speaker. Acest script creează și fișierele 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/gFPt4ypNxIk>

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, insa cred ca am gasit un workaround destul de dragut (descrie 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**