

Pian Electric Intelligent

Student: **Popa Athanasia**

Grupa: 333CC

Introducere

Proiectul ales de mine este un pian electric inteligent. Acesta pune la dispozitie utilizatorului functionalitati multiple si are patru moduri de functionare:

- **canta**: utilizatorul poate canta la pian, precum la oricare alt pian obisnuit prin apasarea clapetelor
- **inregistrare**: utilizatorul poate inregistra melodia cantata intr-o sesiune
- **redare**: utilizatorul poate reda melodiile inregistrate in sesiunile anterioare
- **joc**: utilizatorul se poate juca Piano Tiles (jucatorul trebuie sa apese clapetele corespunzatoare notelor muzicale afisate pe ecran).

Pianul electric inteligent creat de mine cumuleaza facilitati multiple, facand mai usoara munca oricarui pasionat de muzica.

Descriere generală



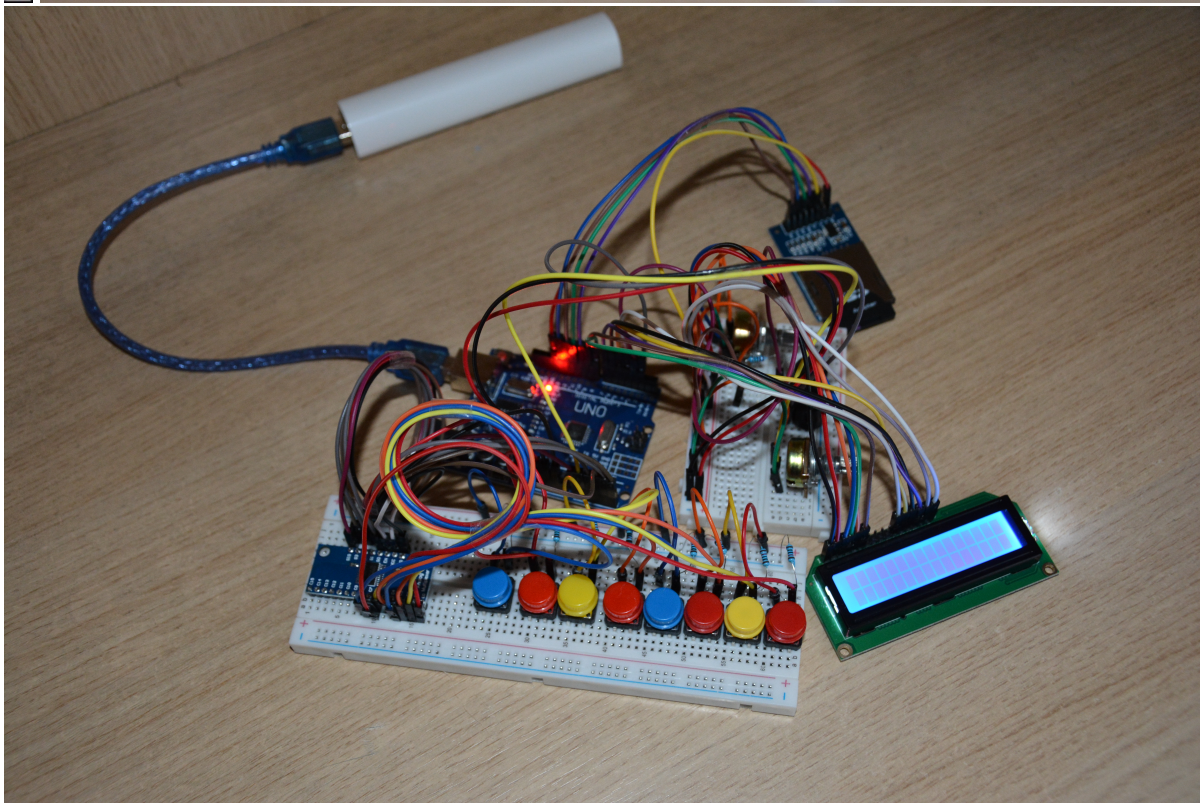
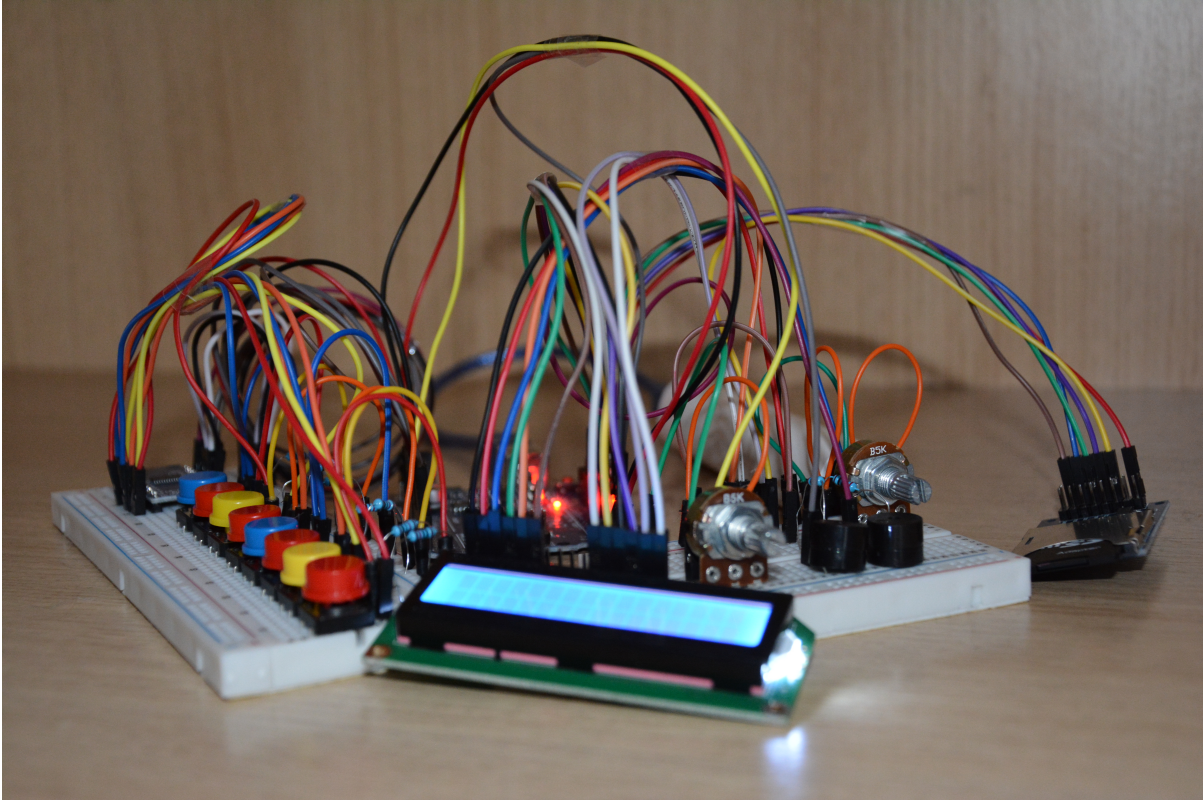
Hardware Design

Lista de piese:

- Arduino Uno
- Potentiometru
- Card SD + Modul scriere/citire card SD
- Ecran LCD
- Buzzer
- Butoane
- Multiplexor 16 canale

- Rezistente, fire si breadboard

Schema electrica



Software Design

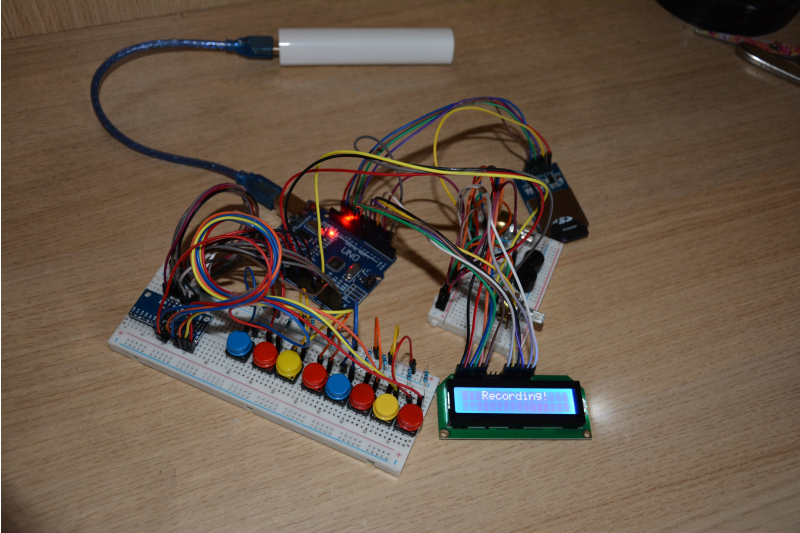
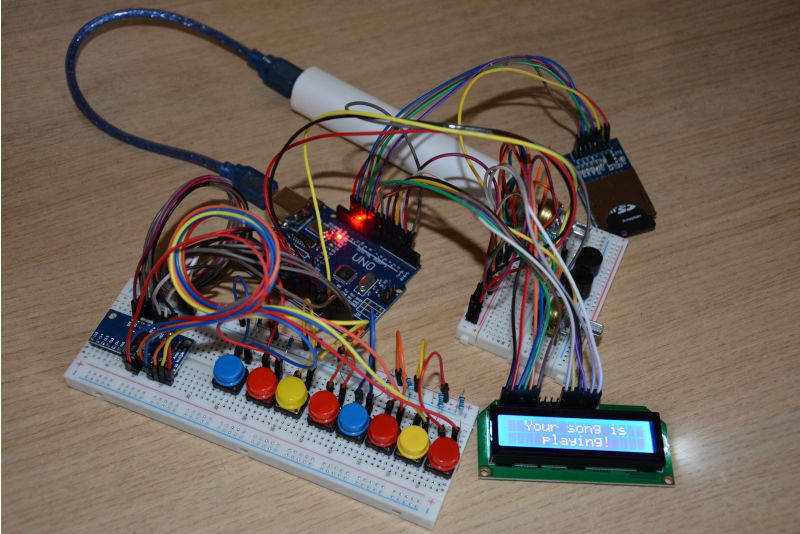
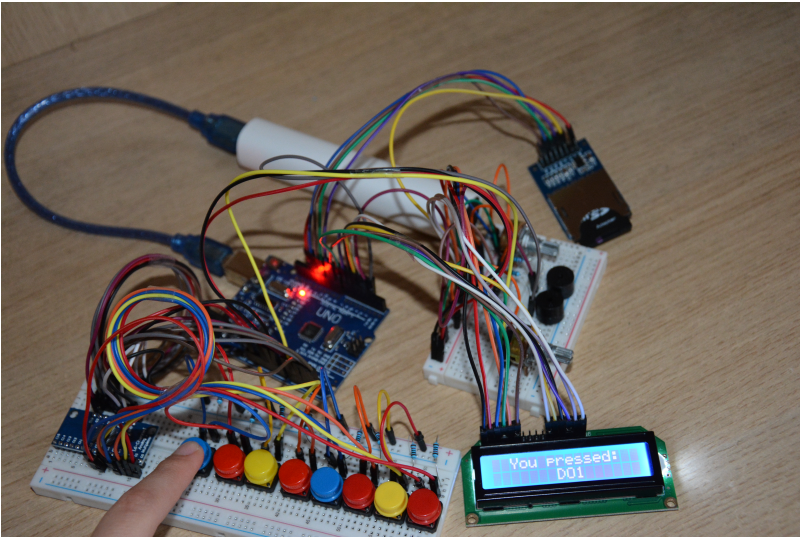
Mediul de dezvoltare folosit a fost Arduiono IDE, iar bibilotecile utilizate sunt:

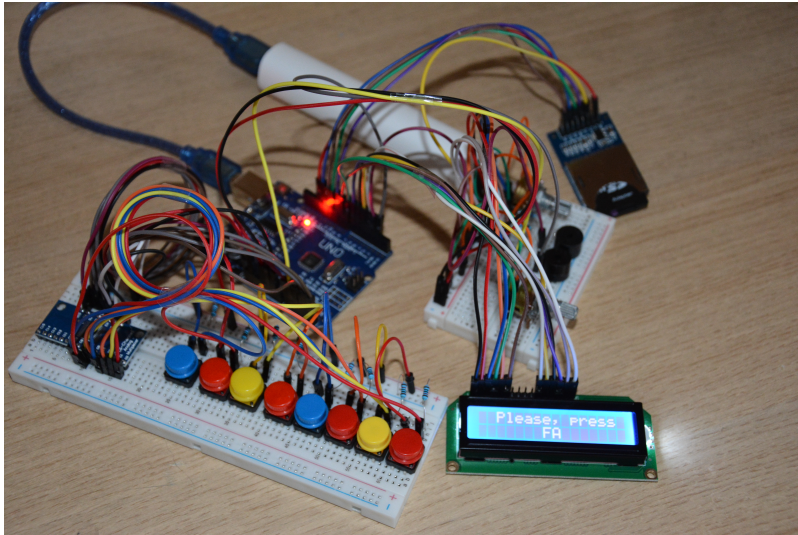
- <LiquidCrystal.h> pentru utilizarea modului LCD I2C
- <Tone.h> pentru redarea sunetelor
- <SD.h> si <SPI.h> pentru folosirea modului de scriere/citire a cardului SD

Implementare:

- global se initializeaza si declarat un array cu valorile frecventelor specifice notelor muzicale, un array cu denumirea acestora si pini utilizati;
 - modul in care se afla pianul (canta, inregistrare, redare, joc) este reprezentat de variabila state care corespunde una dintre aceste stari
 - setup():se declar modul fiecarui pin, se initializeaza lcd-ul si se seteaza seed-ul pentru generatorul de numere random folosit la partea de joc
 - changeState():se citeste valoarea receptate de la potentiometru si se mapeaza aceasta valoare cu modul in care se afla pianul, prin incadrarea valorii citite in intr-un interval specific modului respectiv
 - pressedButton():determina ce clapa/buton a fost apelat si intoarce indexul acestuia sau -1 in cazul in care niciun buton nu a fost apasat prin selectarea secventiala a canalelor multiplexorului si determinarea starii de HIGH/LOW a acestora
 - loop(): se verifica in ce mod este pianul(cu ajutorul functiei state) si se executa codul specific starii respective
1. starea 0(canta): se determine butonul/clapa apasata (cu ajutorul functiei pressedButton()), se reda nota corespunzatoare clapei respective si se afiseaza pe ecranul lcd denumirea notei muzicale
 2. starea 1(inregistrare): se determine butonul/clapa apasata (cu ajutorul functiei pressedButton()), se reda nota corespunzatoare clapei respective, se deschide fisierul corespunzator inregistrarii de pe cardul SD si se scrie indexul ntei tocmai apasta in acesta; in cazul in care nu a fost introdus cardul sd se va afisa un mesaj specific
 3. state 2(redare): se citesc din fisierul corespunzator inregistrarii, de pe cardul sd, notele muzicale ce compun melodia si se redau; in cazul lipsdi cardului sd sau a inregistrarilor se va afisa un mesaj specific pe ecranul lcd;
 4. state 3(joc): se va afusa pe ecranul lcd nota muzicala random ce se doreste a fi apasata, se asteapta inputul utilizatorului, daca aceasata este corect se genereaza o alta nota muzicala, altfel se va afisa pe ecran un mesaj specific

Rezultate Obținute





Concluzii

Realizarea acestui proiect a contribuit la aprofundarea subiectiva a cunostintelor dobandite atat la laborator, cat si la curs. Acesta a reprezentat un prilej spre a a-mi consolida skill-urile de utilizare si interconectare a componentelor hardware, aceasta nefiind o activitatea tocmai familiara mie. De asemenea, gasirea solutiilor cu privire la implementare au reprezentat o provocare: repartizarea eficienta a pinilor de pe placuta Arduino, utilizarea unui multiplexor pentru a reduce numarul de pini corespunzatori butoanelor-caple, utilizarea modulului scriere/citire pentru cardul SD si a ecranului LCD.

Download

Arhiva resurse software: [proiect_pm_popa_athanasia.zip](#)

Jurnal

- 05.05.2023 - Milestone proiect - documentație: alegerea temei proiectului și a funcționalităților de bază
- 19.05.2023 - Milestone - hardware: conectarea elementelor hardware și realizarea schemei electrice
- 26.05.2023 - Milestone - software: implementarea codului și finalizarea proiectului

Bibliografie/Resurse

Resurse Software:

- <https://arduinogetstarted.com/tutorials/arduino-write-variable-to-sd-card>
- <https://reference.arduino.cc/reference/cs/language/functions/advanced-io/tone/>
- laboratoare ocw

Resurse Hardware:

- <https://www.youtube.com/watch?v=R00FpEipWY4&list=WL&index=20>
- <https://forum.arduino.cc/t/using-the-cd74hc4067-multiplexer-with-push-buttons/636536>
- https://www.electrokit.com/uploads/productfile/41015/41015739_-_SD_Card_Module.pdf
- laboratoare ocw

[Export to PDF](#)

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

Permanent link:
<http://ocw.cs.pub.ro/courses/pm/prj2023/dene/pianinteligent>



Last update: **2023/05/29 20:58**