

# MPC Turntable

## Introducere

Autor: Nicu-Catalin Iovita 333CC

Acest proiect are în vedere realizarea unei plăci asemănătoare cu un MPC, un set de DJ, prin care se pot asculta diferite piese muzicale preinterpretate, sau se pot încerca multiple note muzicale. Acest board conține și diferite ilustrații vizuale care diferă de la nota la nota. Rolul proiectului este de familiarizare cu placa tip Arduino, cât și a dispozitivelor de tip I2C.

## Schema bloc







## Descriere generală

Matricea de leduri, va printa inițial mesajul de bun venit, "HELLO, iar ecranul LCD, de asemenea un mesaj de bun venit. În acest moment, utilizatorul poate apăsa unul din butoanele aflate pe matricea de 4x4, pentru a emite un sunet, din diferită gama

O data ce va apăsa unul dintre butoane, matricea de leduri se va colora, având forme și modele diferite. De asemenea, exista opțiunea ca utilizatorul sa apese butonul 'A', respectiv 'B', pentru reglarea sunetului, având la dispoziție cele 3 difuzoare.

Totodată, la apăsarea butonului 'D', se poate intra în modul de Loop; nota apăsata anterior va cicla, creând un efect de acompaniament. De asemenea, utilizatorul poate regla nivelul de BPM (Beats Per Minute), apăsând tastele de '\*' și '#', creând un ritm mai alert sau mai puțin alert.

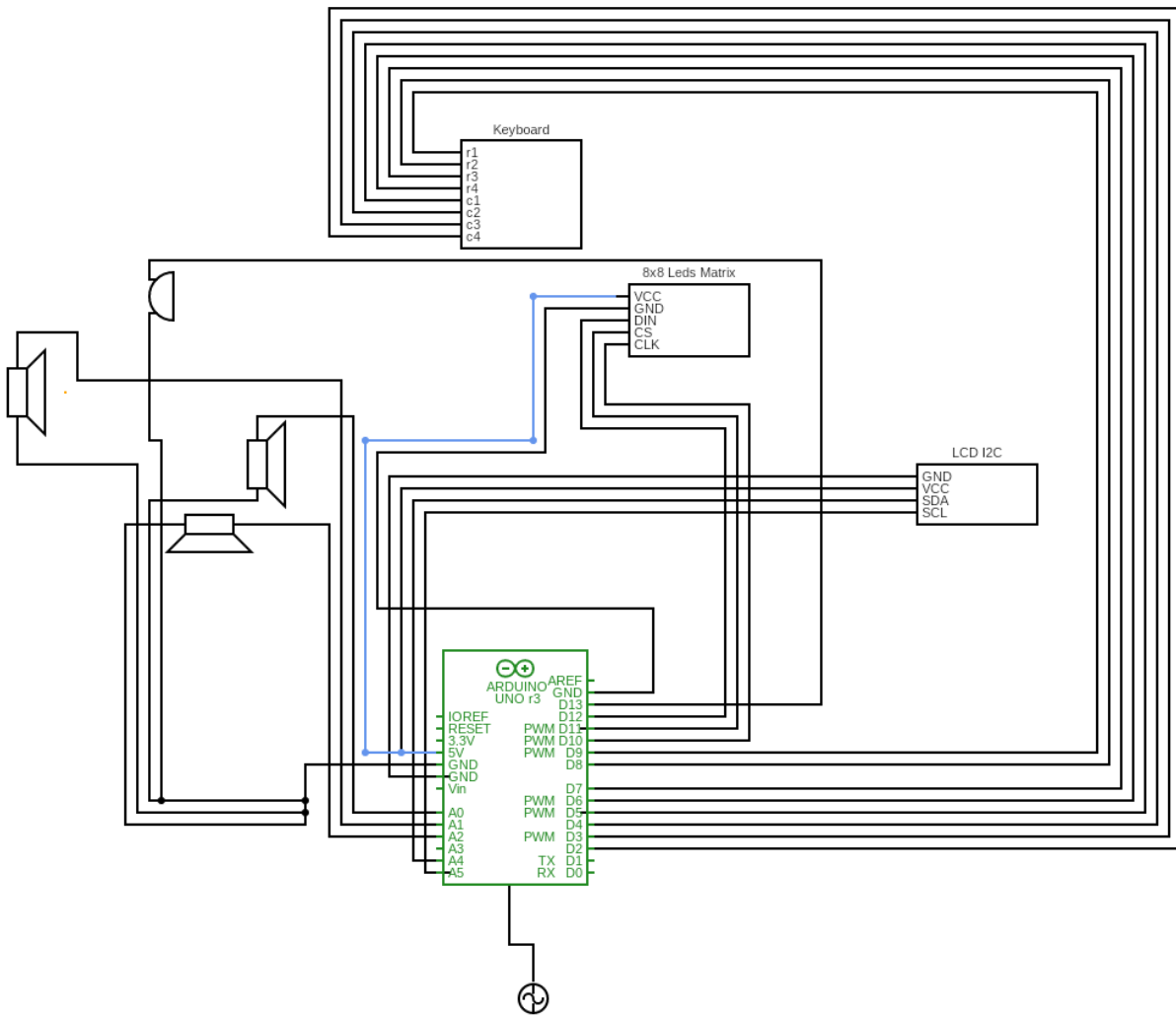
La apăsarea butonului '0', utilizatorul poate asculta o melodie întreaga, admirând notele line.

## Hardware Design

- Arduino UNO
- 8x8 Leds Matrix
- 4x4 Matrix Keyboard
- LCD 1602 IIC/I2C

- 3x Speakers
- Breadboard
- Fire legatura
- 1x Buzzer

## Schema electrica



## LCD I2C

Era necesar sa fie conectat la pinii A4, respectiv A5 din configuratie, așadar i-am legat de breadboard și apoi am tras alte fire către placa Arduino

## 4x4 Keypad

Am conectat fiecare linie, respectiv coloana către breadboard, apoi spre Arduino UNO, folosind pinii de la 2, la 9

## 3x Speakers

Difuzoarele le-am legat la GND pe fiecare (în breadboard), iar fiecare fir la câte o intrare de tip A (A0, A1, A2)

## 8x8 Matrix

Matricea de leduri am legat-o la pinii rămași D9, D10, D11 pentru CLK, DIN

## 1x Buzzer

Buzzer-ul l-am legat la pin-ul 13 rămas pe placa Arduino.

## Software Design

Ca si biblioteci, am folosit:

- <Keypad.h>
- <LedControl.h>
- <LiquidCrystal\_I2C.h>

## Ecran LCD

Ca si implementare, pentru ecranul LCD I2C, mi-am creat funcții care vor printa pe ecran diferite mesaje. Aceste funcții (printVolume, printMelodyTitle, etc), vor apela `lcd.clear()`, pentru curățarea ecranului, apoi vor poziționa cursorul pe ecran și vor afișa mesajul dorit.

## Matrice leduri

Pentru afisarea pe matricea de leduri, m-am folosit de o funcție ce parcurge fiecare led din interiorul matricei (printByte), apelând-o cu diferiți bytes. Inițial, am creat o funcție ce se va apela în modul de StandBy (mod în care utilizatorul nu dorește sa acționeze). În acest mod, matricea va cicla cu niște modele specifice, pana când utilizatorul va apăsa o tasta anume. De fiecare data când utilizatorul apăsa o tasta, modelul de pe matrice se va recolora. Pentru recolorarea matricilor, am folosit un software găsit online, numit Form, ce te ajuta sa creezi modele pe aceasta matrice 8x8, generand o secvența de octeti în binar sau hexazecimal.

<https://www.youtube.com/watch?v=lwH9vzC4BKI>

## Keypad

Pentru tastatura cu membrana, nu a fost complicata partea de software: în funcția de loop voi prelua de fiecare data caracterul apăsat de utilizator, și voi verifica valoarea acestuia. Cum am specificat, pentru tastele de la 1 la 9, se vor emite sunete generate de către difuzoare. La apăsarea tastei D, voi emite sunetul memorat anterior pana când tasta D va fi din nou apăsată.

## Difuzoare

Pentru difuzoare, m-am folosit de funcțiile tone() și noTone(). Ce nu am reușit prin implementarea mea a fost sa elimin partea de delay din apelul acestor funcții. Eu doream sa pot emite un sunet cu buzzerul (independent) și sa pot emite sunete diferite prin cele trei difuzoare; nu a fost posibil, deoarece sunetele se emiteau pe rând. Am încercat sa elimin o parte din delay, spre exemplu din afișarea pe matrice, creând funcția simulateDelay(), ca sa pot emite sunete în același timp și sa colorez matricea în timp real.

Pentru melodia cântată automat (Game of Thrones theme), am găsit repository-ul lăsat în documentație pe GitHub, unde notele erau generate și melodia era deja existenta (împreună cu delay-ul notelor). Inițial am pus mai multe melodii, însă am renunțat și am lăsat doar una în final, deoarece am vrut sa implementez și partea de Loop, cât și incrementul și decrementul BPM.

Partea de increment și decrement BPM se referă la simularea unui metronom; un instrument ce rulează pe fundal, de obicei niște note mai joase, pentru ca tu sa te poți sincroniza cu acesta, când dorești sa creezi diferite note (funcția freeStyle() din cod)

## Buzzer

Cum spuneam, buzzerul va lua rolul de 'metronom', creând ciclicitatea la dorința userului de a reda sunete joase sau înalte, pe care iei poate exersa talentul.

## Rezultate Obținute

<https://github.com/cataiovita/MPC-turntable>

<https://youtu.be/2Of8jpestDtQ>

## Concluzii

A fost un proiect deosebit, mi-a plăcut foarte mult sa lucrez în lumea Arduino. Cel mai satisfăcător lucru a fost reușită de a percepe rezultatele în mod real; fiecare licărire de bec a matricei îmi provoca puțin extaz. M-a bucurat foarte mult sa găsesc, totodata, o comunitate foarte primitoare, unde iti poți găsi inspirația pentru proiecte, de asemenea și rezolvarea unor probleme la nivel hardware, cât și software.

Cele mai folosite componente, s-au dovedit a fi, cele cu suport I2C, neavând foarte multi pini disponibili la un moment-dat pe placa Arduino. De asemenea, m-a ajutat foarte mult și breadboard-ul, chiar aş spune ca o consider o componenta necesară pentru dezvoltare.

Partea software, nu a fost foarte diferită de ce am experimentat pana acum (programarea în C fiind foarte comuna), însă cea mai atractiva latura a fost partea de hardware, facandu-ma sa cred ca, desi am conceput un proiect simplu, m-a ajutat sa mă dezvolt.

## Download

[mpc\\_turntable.ino.zip](#)

## Jurnal

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

## Bibliografie/Resurse

Listă cu documente, datasheet-uri, resurse Internet folosite, eventual grupate pe **Resurse Software** și **Resurse Hardware**.

<https://www.circuitbasics.com/how-to-set-up-a-keypad-on-an-arduino/>

<https://github.com/robsoncouto/arduino-songs>

[Export to PDF](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2022/cstan/800>



Last update: **2022/05/30 00:30**