UNIVERSITATEA ,,POLITEHNICA” din BUCUREȘTI

**Facultatea de Electronică, Telecomunicații și Tehnologia Informației**

**PROIECT**

Semnale si programare

Joc de memorie

-Model Arduino-

Pupazan Bogdan-Nicolae

Grupa 423C

# Scopul proiectului

Proiectul reprezinta un joc de memorie folosind placa integrata Arduino Uno, un display LCD cu interfata I2C, leduri, butoane si un buzzer atasate de un breadboard. Acesta functioneaza in felul urmator: unul dintre cele 4 leduri se aprinde iar buzzerul va reda un ton, dupa care jucatorul va trebui sa apese din nou ledul respectiv. Dupa aceea, se va aprinde acelasi led si un nou led la intamplare acompaniate de frecventele respective de la buzzer, dupa care jucatorul are de repetat secventa in care s-au aprins ledurile.

Pentru a retine mai usor secventa, buzzerul este programat sa produca o frecventa specifica fiecarui led.Jucatorul de asemenea poate sa vizualizeze runda la care se afla urmarind pe display numaratoarea intitulata “Scorul tau”, fiecare secventa corecta crescand-o pe aceasta cu 1.

Acest lucru continua pana la o secventa de 10 leduri, moment in care pe display se va afisa mesajul “Ai castigat!”. Dupa aceea, ledurile se vor aprinde rapid intr-o ordine aleatoarie si buzzerul va reda o melodie, semnificand victoria.

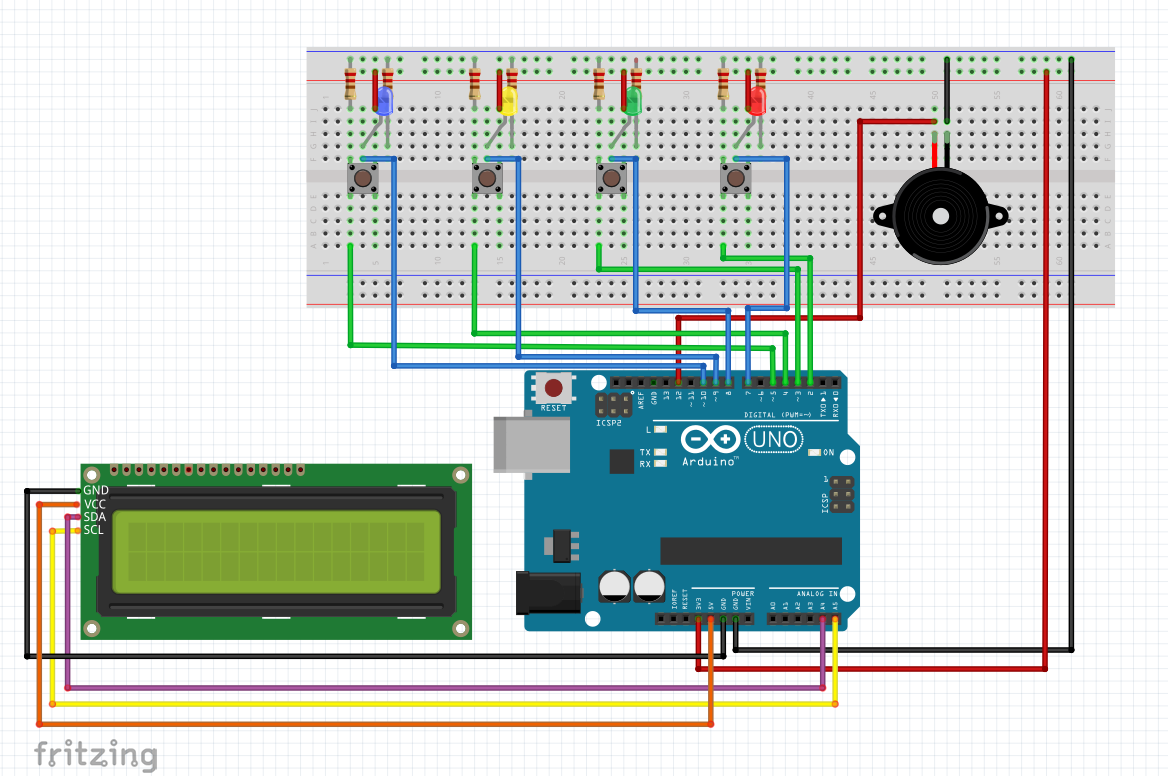
In continuare, secventa se va reseta si scorul se va intoarce la 1, dar de aceasta data ledurile vor sta mult mai putin timp aprinse, semnificand un grad de dificultate crescut.

In cazul in care jucatorul greseste secventa, acesta va fi intampinat de un ton jos de la buzzer si de doua iluminari ale tuturor ledurilor, semnificand infrangerea, dupa care jocul o ia de la capat, resetandu-si scorul pe display.

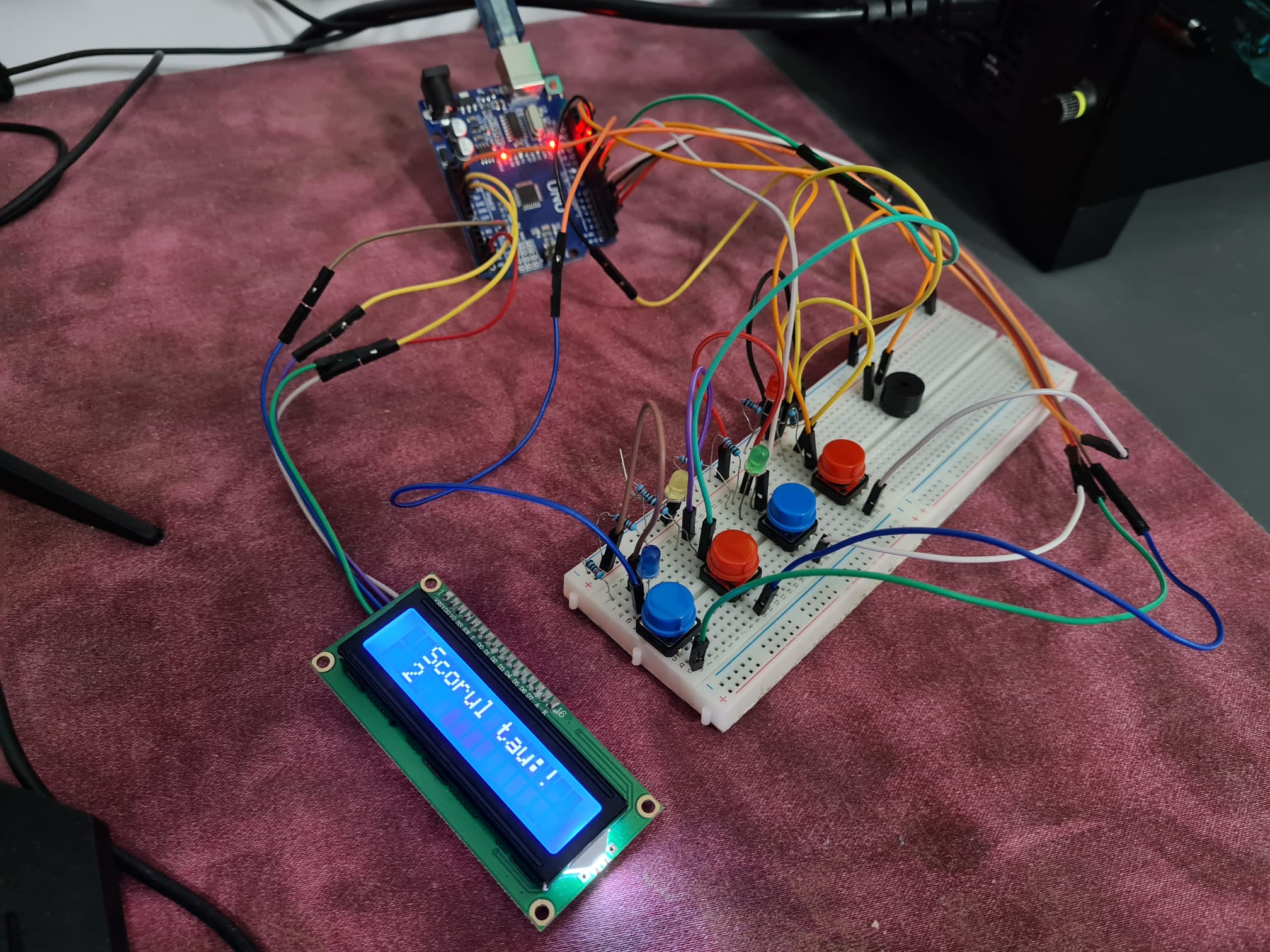
Schema proiectului este formată din:

* Arduino UNO
* 4 Leduri (albastru, galben, verde, rosu)
* 8 Rezistente de 220Ω
* Buzzer
* 4 Butoane
* Display LED + interfata I2C
* Cabluri jumper
* Breadboard

# Montaje



**Schema montajului**

**- realizata folosind programul de proiectare “Fritzing”** -

**Montajul fizic**

# Demonstratii video

[**Demonstratie joc**](https://youtu.be/wYB1bGsFFKM)[**Demonstratie pierdere**](https://youtu.be/bRbLGVJb0QY)[**Demonstratie castig**](https://youtu.be/SYlGh3Xak2A)

# Codul Arduino

/\*

joc arduino memorie

pt proiect semnale si programare

\*/

#include <Wire.h> // Library for I2C communication

#include <LiquidCrystal\_I2C.h> // Library for LCD

LiquidCrystal\_I2C lcd **=** LiquidCrystal\_I2C**(**0x27**,** 16**,** 2**);** // setez adresa la display 0x27, si tipul de display 16\*2

// constante

const int button1 **=** 2**;** // primul buton Blue LED

const int button2 **=** 3**;** // al doilea buton Yellow LED

const int button3 **=** 4**;** // al treilea buton Green LED

const int button4 **=** 5**;** // al patrulea buton Red LED

const int led1 **=** 7**;** // Blue LED

const int led2 **=** 8**;** // Yellow LED

const int led3 **=** 9**;** // Green LED

const int led4 **=** 10**;** // Red LED

const int buzzer **=** 12**;** // buzzerul

const int tones**[]** **=** **{**1915**,** 1700**,** 1519**,** 1432**,** 2700**};** // tonuri led, ultimul ton e pt cand pierzi

// variabile

int buttonState**[]** **=** **{**0**,**0**,**0**,**0**};** // stadiu actual buton

int lastButtonState**[]** **=** **{**0**,**0**,**0**,**0**};** // stadiu anterior buton

int buttonPushCounter**[]** **=** **{**0**,**0**,**0**,**0**};** // nr apasari

// functie de ton

void playTone**(**int tone**,** int duration**)** **{**

**for** **(**long i **=** 0**;** i **<** duration **\*** 1000L**;** i **+=** tone **\*** 2**)** **{**

digitalWrite**(**buzzer**,** HIGH**);**

delayMicroseconds**(**tone**);**

digitalWrite**(**buzzer**,** LOW**);**

delayMicroseconds**(**tone**);**

**}**

**}**

void setup**()** **{**

// lcd

lcd**.**init**();**

lcd**.**backlight**();**

// initialize inputs :

randomSeed**(**analogRead**(**0**));**

pinMode**(**button1**,** INPUT**);**

pinMode**(**button2**,** INPUT**);**

pinMode**(**button3**,** INPUT**);**

pinMode**(**button4**,** INPUT**);**

// initializez outputurile:

pinMode**(**led1**,** OUTPUT**);**

pinMode**(**led2**,** OUTPUT**);**

pinMode**(**led3**,** OUTPUT**);**

pinMode**(**led4**,** OUTPUT**);**

pinMode**(**buzzer**,** OUTPUT**);**

//Serial.begin(9600);

**}**

int game\_on **=** 0**;**

int wait **=** 0**;**

int currentlevel **=** 1**;** // scorul actual / nr de apasari pt urmatorul nivel

long rand\_num **=** 0**;** // variabila pt random

int rando **=** 0**;** // variabila pt loopgame

int butwait **=** 500**;** //timp de asteptat pt urm input (workaround de-bounce)

int ledtime **=** 500**;** //cat sta ledul aprins

int n\_levels **=** 10**;** //scor necesar pt castig

int pinandtone **=** 0**;** //pt secventa

int right **=** 0**;** //practic un ok; 1 pt urmatorea secventa

int speedfactor **=** 5**;** //viteza leduri si sunet; creste pe masura ce castigi

int leddelay **=** 200**;** //timp initializare led; scade pe masura ce castigi

void loop**()** **{**

int n\_array**[**n\_levels**];**

int u\_array**[**n\_levels**];**

int i**;**

//curat ambele array uri si incep un joc nou

**if** **(**game\_on **==** 0**){**

**for(**i**=**0**;** i**<**n\_levels**;** i**=**i**+**1**){**

n\_array**[**i**]=**0**;**

u\_array**[**i**]=**0**;**

rand\_num **=** random**(**1**,**200**);**

**if** **(**rand\_num **<=** 50**)**

rando**=**0**;**

**else** **if** **(**rand\_num**>**50 **&&** rand\_num**<=**100**)**

rando**=**1**;**

**else** **if** **(**rand\_num**>**100 **&&** rand\_num**<=**150**)**

rando**=**2**;**

**else** **if** **(**rand\_num**<=**200**)**

rando**=**3**;**

//salveaza nr random in array

n\_array**[**i**]=**rando**;**

**}**

game\_on **=** 1**;**

lcd**.**setCursor**(**2**,** 0**);** // setez cursorul primul rand a treia coloana

lcd**.**print**(**"Scorul tau:"**);** // printez numaratoare

lcd**.**setCursor**(**2**,** 1**);** //setez cursorul urm rand

lcd**.**print**(**currentlevel**-**1**);** // printez scorul

**}**

//aprind secventa curenta

**if** **(**wait **==** 0**){**

delay **(**200**);**

i **=** 0**;**

**for** **(**i **=** 0**;** i **<** currentlevel**;** i**=** i **+** 1**){**

leddelay **=** ledtime**/(**1**+(**speedfactor**/**n\_levels**)\*(**currentlevel **-** 1**));**

pinandtone **=** n\_array**[**i**];**

digitalWrite**(**pinandtone**+**7**,** HIGH**);**

playTone**(**tones**[**pinandtone**],** leddelay**);**

digitalWrite**(**pinandtone**+**7**,** LOW**);**

delay**(**100**/**speedfactor**);**

**}**

wait **=** 1**;**

**}**

i **=** 0**;**

int buttonchange **=** 0**;**

int j **=** 0**;** // poz curenta in secventa

**while** **(**j **<** currentlevel**){**

**while** **(**buttonchange **==** 0**){**

**for** **(**i **=** 0**;** i **<** 4**;** i **=** i **+** 1**){**

buttonState**[**i**]** **=** digitalRead**(**i**+**2**);**

buttonchange **=** buttonchange **+** buttonState**[**i**];**

**}**

**}**

**for** **(**i **=** 0**;** i **<** 4**;** i **=** i **+** 1**){**

**if** **(**buttonState**[**i**]** **==** HIGH**)** **{**

digitalWrite**(**i**+**7**,** HIGH**);**

playTone**(**tones**[**i**],** ledtime**);**

digitalWrite**(**i**+**7**,** LOW**);**

wait **=** 0**;**

u\_array**[**j**]=**i**;**

buttonState**[**i**]** **=** LOW**;**

buttonchange **=** 0**;**

**}**

**}**

**if** **(**u\_array**[**j**]** **==** n\_array**[**j**]){**

j**++;**

right **=** 1**;**

**}**

**else{**

right **=** 0**;**

i **=** 4**;**

j **=** currentlevel**;**

wait **=** 0**;**

**}**

**}**

//joc pierdut

**if** **(**right **==** 0**){**

lcd**.**clear**();**

lcd**.**setCursor**(**2**,** 0**);**

lcd**.**print**(**"Ai pierdut!"**);**

delay**(**300**);**

i **=** 0**;**

game\_on **=** 0**;**

currentlevel **=** 1**;**

**for** **(**i **=** 0**;** i **<** 4**;** i **=** i **+** 1**){**

digitalWrite**(**i**+**7**,** HIGH**);**

**}**

playTone**(**tones**[**4**],** ledtime**);**

**for** **(**i **=** 0**;** i **<** 4**;** i **=** i **+** 1**){**

digitalWrite**(**i**+**7**,** LOW**);**

**}**

delay **(**200**);**

**for** **(**i **=** 0**;** i **<** 4**;** i **=** i **+** 1**){**

digitalWrite**(**i**+**7**,** HIGH**);**

**}**

playTone**(**tones**[**4**],** ledtime**);**

**for** **(**i **=** 0**;** i **<** 4**;** i **=** i **+** 1**){**

digitalWrite**(**i**+**7**,** LOW**);**

**}**

delay**(**500**);**

game\_on **=** 0**;**

**}**

// secventa corecta, urm nivel

**if** **(**right **==** 1**){**

currentlevel**++;**

wait **=** 0**;**

lcd**.**setCursor**(**2**,** 1**);**

lcd**.**print**(**currentlevel**-**1**);**

**}**

//castig joc

**if** **(**currentlevel **==** n\_levels**){**

lcd**.**clear**();**

lcd**.**setCursor**(**2**,** 0**);**

lcd**.**print**(**"Ai castigat!"**);**

delay**(**500**);**

// The following is the victory sound:

int notes**[]** **=** **{**2**,** 2**,** 2**,** 2**,** 0**,** 1**,** 2**,** 1**,** 2**};**

int note **=** 0**;**

int tempo**[]** **=** **{**200**,** 200**,** 200**,** 400**,** 400**,** 400**,** 200**,** 200**,** 600**};**

int breaks**[]** **=** **{**100**,** 100**,** 100**,** 200**,** 200**,** 200**,** 300**,** 100**,** 200**};**

**for** **(**i **=** 0**;** i **<** 9**;** i **=** i **+** 1**){**

note **=** notes**[**i**];**

digitalWrite**(**note**+**7**,** HIGH**);**

playTone**(**tones**[**note**],** tempo**[**i**]);**

digitalWrite**(**note**+**7**,** LOW**);**

delay**(**breaks**[**i**]);**

delay**(**1000**);**

**}**

lcd**.**clear**();**

//setez jocul la 0 ca sa inceapa unul nou

game\_on **=** 0**;**

currentlevel **=** 1**;**

n\_levels **=** n\_levels **+** 2**;**

speedfactor **=** speedfactor **+** 1**;**

**}**

**}**

# Bibliografie

* + <https://github.com/fdebrabander/Arduino-LiquidCrystal-I2C-library>
  + <https://www.arduino.cc/reference/en/>
  + <https://randomnerdtutorials.com/projects-arduino/>