

Jumping Game

Autor: FLOREA Adrian - 336CB

Introducere

Scopul lucrării este de a folosi un ecran LCD, conectat cu ajutorul unui modul I2C, pentru a interacționa cu utilizatorul astfel încât acesta să poată juca un Mini-game, prin apăsarea unor butoane. Proiectul mă ajută să înțeleg cum pot lucra cu un LCD, în concordanță cu butoane și un buzzer.

Descriere generală

Proiectul constă în implementarea unui simplu joc, folosind un ecran LCD, butoane și un buzzer. Ecranul LCD este conectat la microcontroller printr-un modul I2C. Jucătorul poate alege dificultatea jocului, care se referă la viteza de deplasare a personajului. La apăsarea de butoane sau la diverse evenimente ale jocului, buzzerul este setat cu anumite melodii/sunete.



Hardware Design

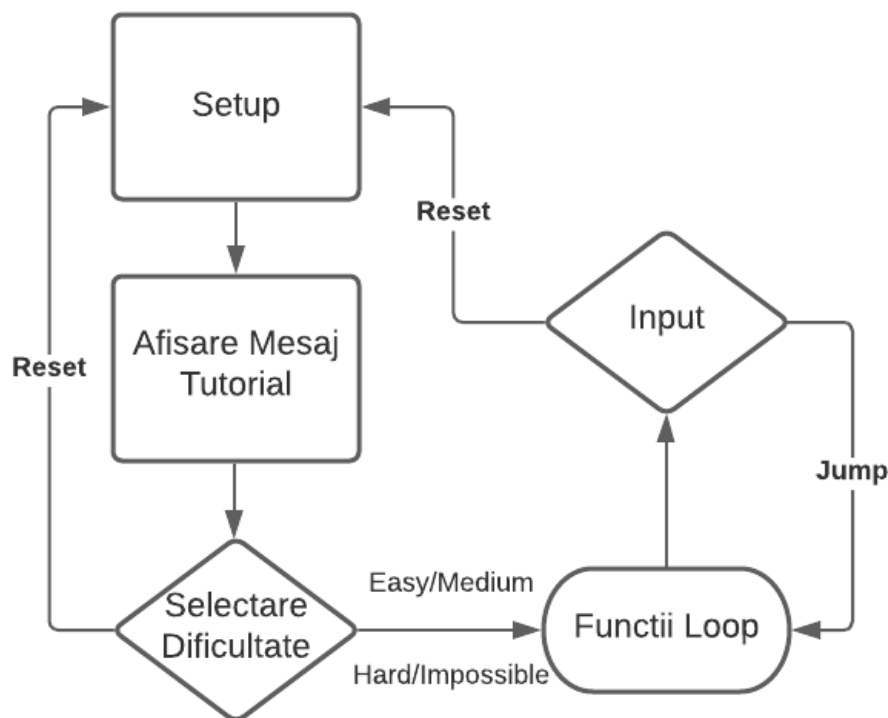
Lista Componente:

1. Arduino UNO
2. Standard LCD 16x2
3. Breadboard
4. Butoane
5. Modul I2C
6. Cabluri/Fire

Schema electrica:



Software Design



void setup()

1. seteaza pinii pentru buzzer, selectare dificultate, saritura (JUMP button);
2. creaza obiectele pentru scena folosind initializeGraphics()
3. loop in functia menu() pana cand utilizatorul alege o dificultate, dupa care programul trece in loop()
4. porneste lcd-ul declarat global

void loop()

1. in functie de dificultatea aleasa apeleaza functia principala draw_func(int delay_time)
2. delay_time se refera la viteza cu care se actualizeaza scena

void menu()

1. parseaza prin setarile de dificultate pana cand se foloseste butonul de jump pentru a incepe jocul

void beginning()

1. printeaza mesajul de inceput al jocului, inainte de selectarea dificultatii

void draw_func(int delay_time)

1. daca jucatorul nu a avut o coliziune, primeste input de la butonul de jump
2. daca s-a intamplat o coliziune, jocul se blocheaza si asteapta input pe jump
3. altfel scena se shifteaza la stanga constant folosind functia advanceTerrain()

bool drawHero(byte position, char* terrainUpper, char* terrainLower, unsigned int score)

1. verificarea coliziei dintre personaj si scena

2. afiseaza personajul si animatiile sale

```
void advanceTerrain(char* terrain, byte newTerrain)
```

1. afiseaza piese in scena si genereaza parti noi

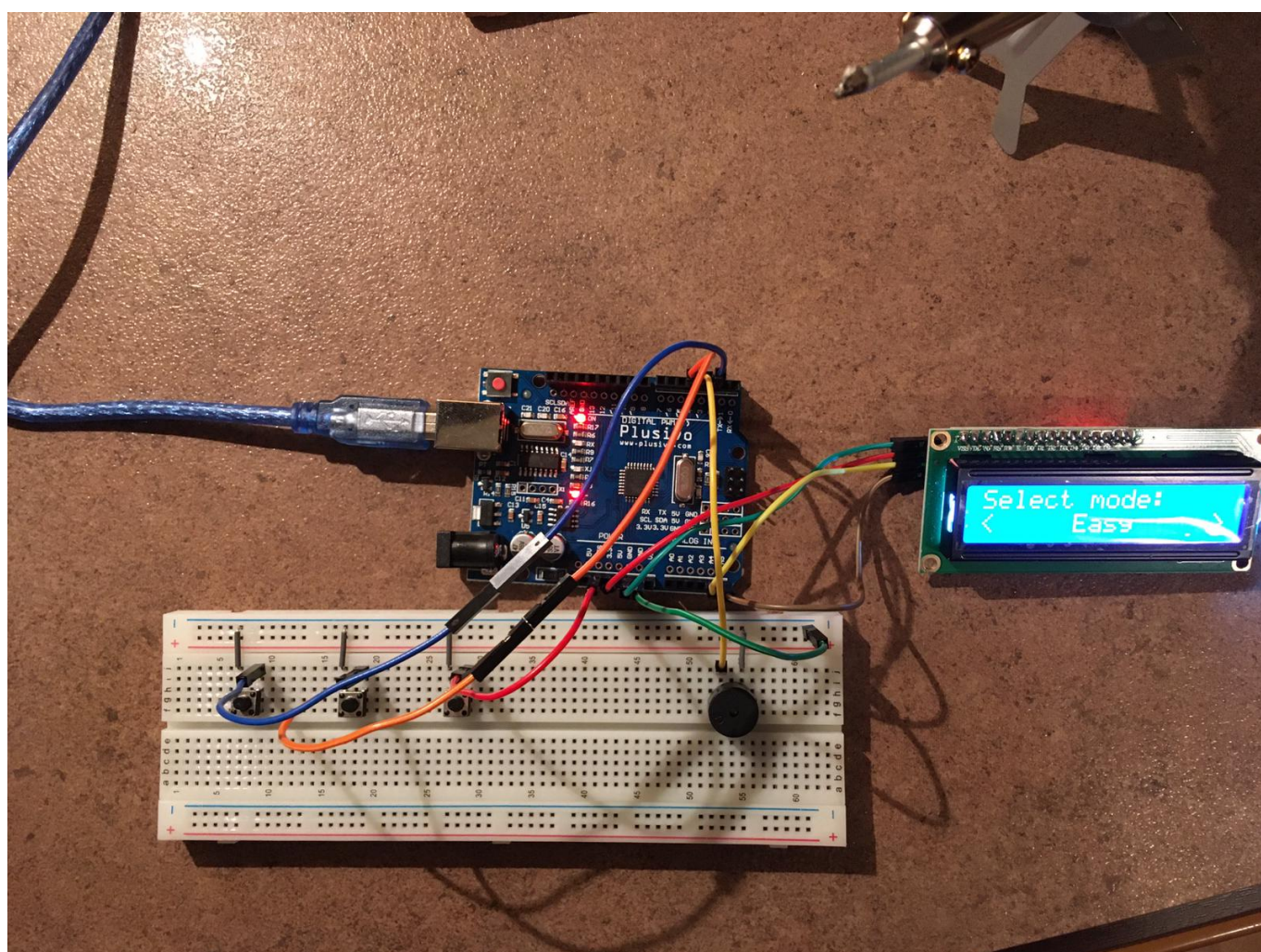
PIN 4: select

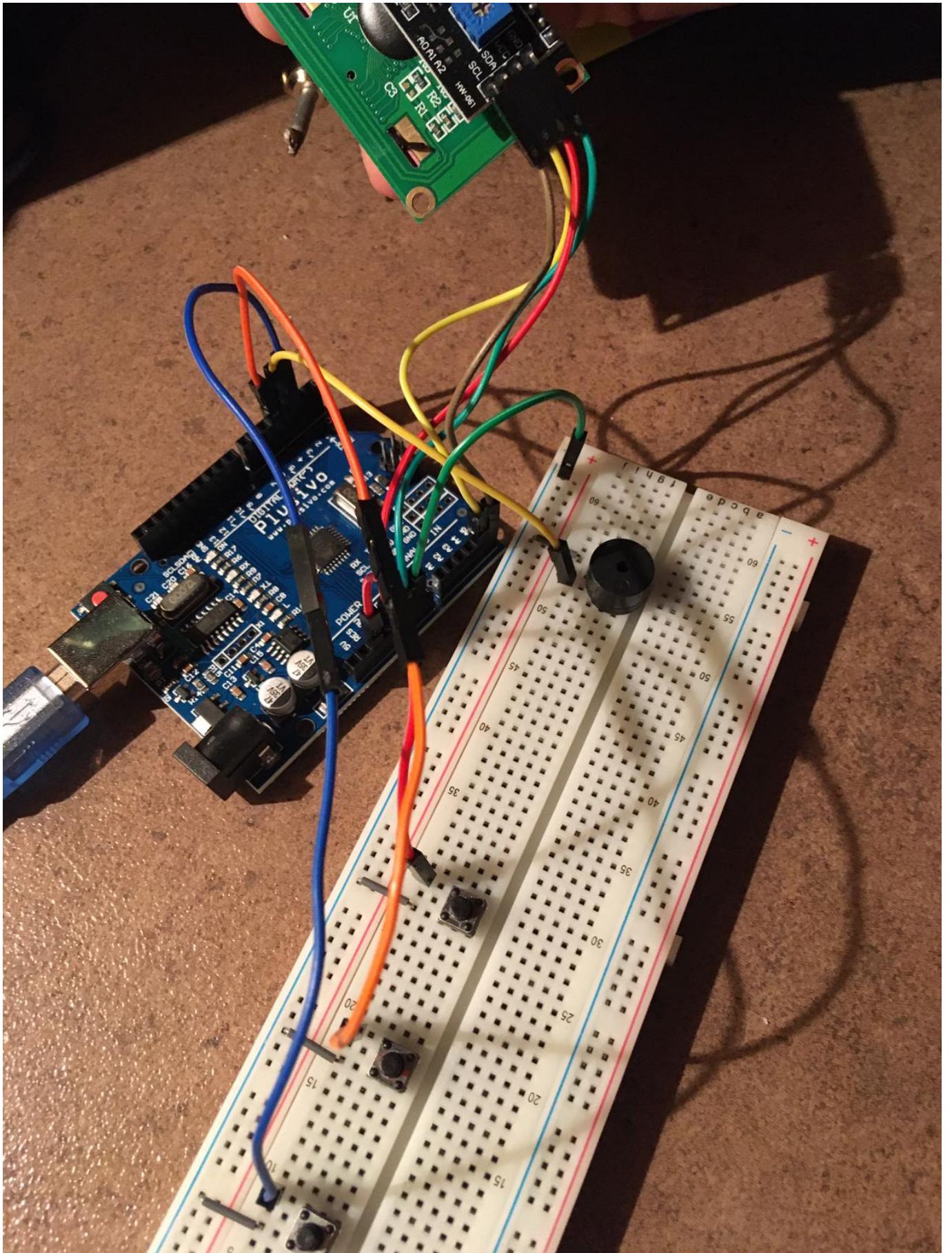
PIN 3: buzzer

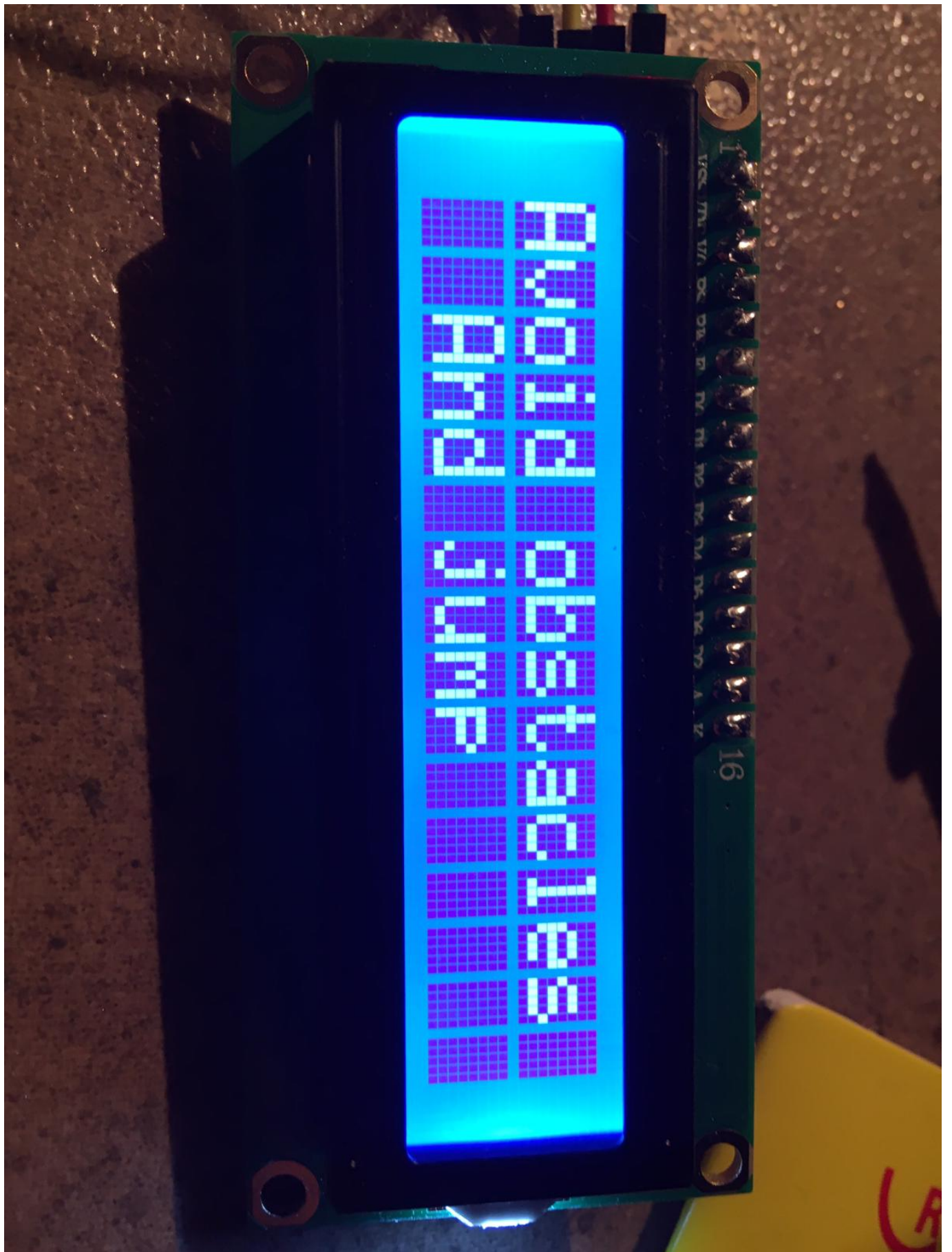
PIN 2: jump

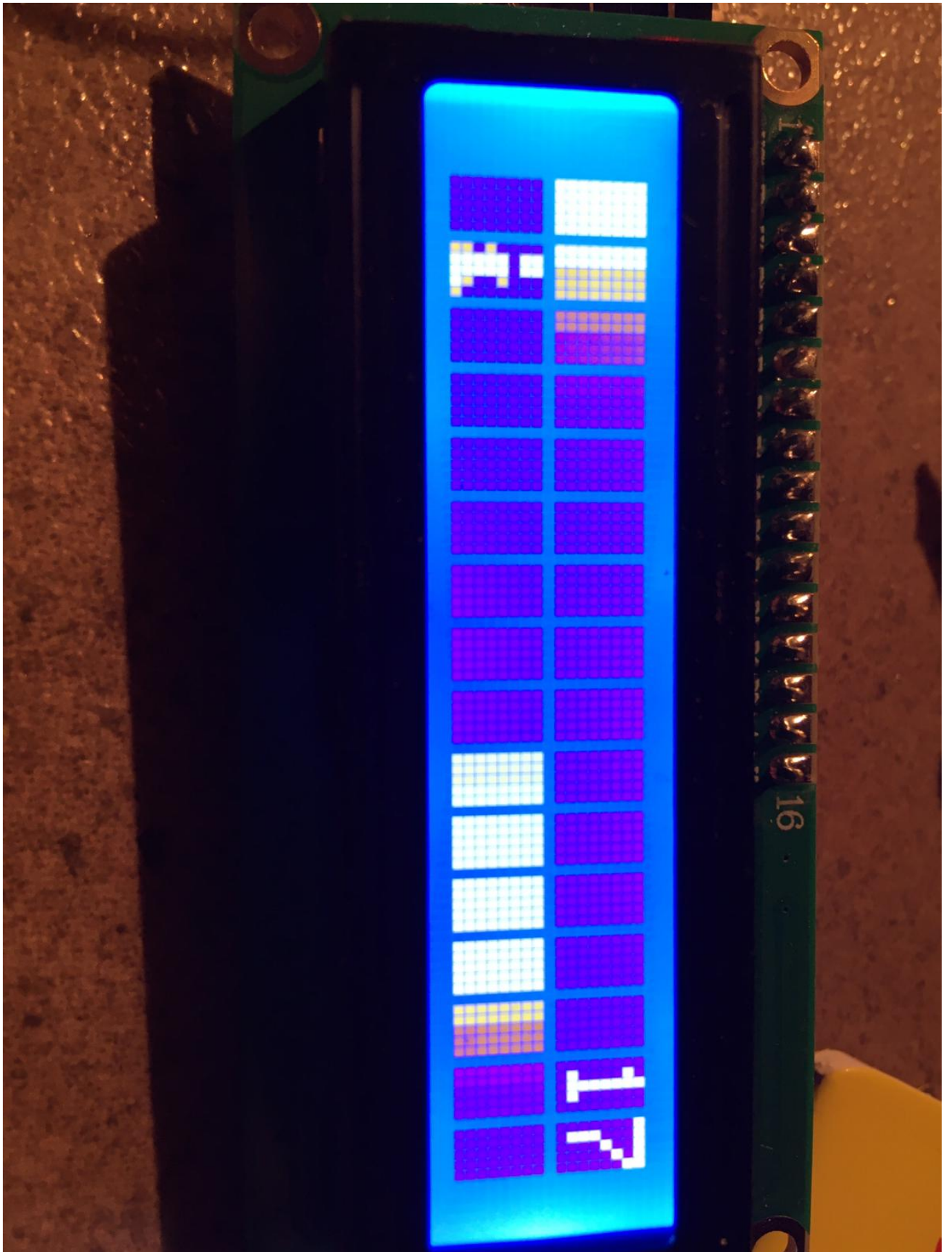
Biblioteca folosita pentru comunicarea I2C este atasata in sectiunea [Bibliografie/Resurse](#).

Rezultate Obținute









<< DEMO >>

Concluzii

Proiectul a fost amuzant si relativ simplu de realizat, si cu datorita lui am invatat (cat de cat) cum se foloseste un ciocan de lipit, dar mai ales cum functioneaza un ecran LCD cu modul I2C si cum se poate lucra cu acesta in diverse proiecte.

Jurnal

2021/04/27 [Jumping Game](#) Pagina creata, pregatire lista componente, schema bloc

2021/05/11 [Jumping Game](#) Schema electrica

2021/05/28 [Jumping Game](#) Demo

2021/05/30 [Jumping Game](#) Finalizare pagina

Bibliografie/Resurse

[Pagina sub forma PDF](#)

[Driver Plusivo](#)

[Tutorial LCD 1602 I2C](#)

[Cod Sursa](#)

[Biblioteca LCD I2C](#)

From:

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

Permanent link:

http://ocw.cs.pub.ro/courses/pm/prj2021/abirlica/jumping_game



Last update: **2021/05/30 16:17**