

1v1 Game Console

Autor: Proca Andreea-Dana - 332CC

Introducere

Consola portabilă cu jocuri 1v1 oferă o experiență de joc captivantă pentru doi jucători, încurajând competiția și distracția într-un format portabil. Aceasta permite jucătorilor să se bucure de jocuri împreună într-o manieră interactivă. Prin intermediul unui joystick, fiecare jucător poate controla mișcările personajului în joc, iar ecranul LCD afișează meniurile și grafica jocurilor într-un mod clar și captivant. De asemenea, prin intermediul unui difuzor integrat, consola redă muzică pentru o experiență audio completă. Am dorit să dezvolt o consolă portabilă cu jocuri 1v1 pentru a oferi o experiență de joc captivantă și interactivă pentru doi jucători. 😊

Descriere generală

Jucătorul poate controla mișcările folosind un joystick, iar meniul și jocurile vor fi afișate pe un ecran LCD, muzica va fi redată de un difuzor. Când unul dintre jucători câștigă un punct, ledul RGB se aprinde pentru câteva secunde în culoarea corespunzătoare celui jucător. La final, ledul RGB se aprinde intermitent schimbându-și culoarea, iar pe ecran vor fi afișate scorul și câștigătorul. Jocurile disponibile pe consola sunt **Snake 1v1** și **Whack-a-mole**.

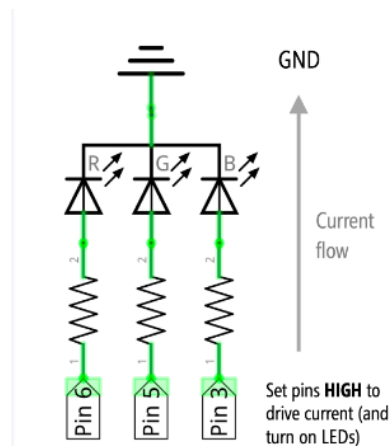
♦ **Snake 1v1** : fiecare jucător controlează un șarpe care își mărește lungimea odată cu trecerea timpului. Dacă unul dintre șerpi se intersectează cu celălalt, lovește un perete sau pe el însuși pierde 🤪. Scopul este de a rămâne cât mai mult timp în viață 😊

♦ **Whack-a-mole** : pe ecran vor fi afișate mai multe causte din care vor ieși aleator cartite. Jucătorii trebuie să le lovească selectând casuta, astfel câștigând un punct. Dacă ambii lovesc aceeași cartită, punctul va fi câștigat de primul care a lovit-o. Câștigă cel care are cele mai multe puncte. 😊

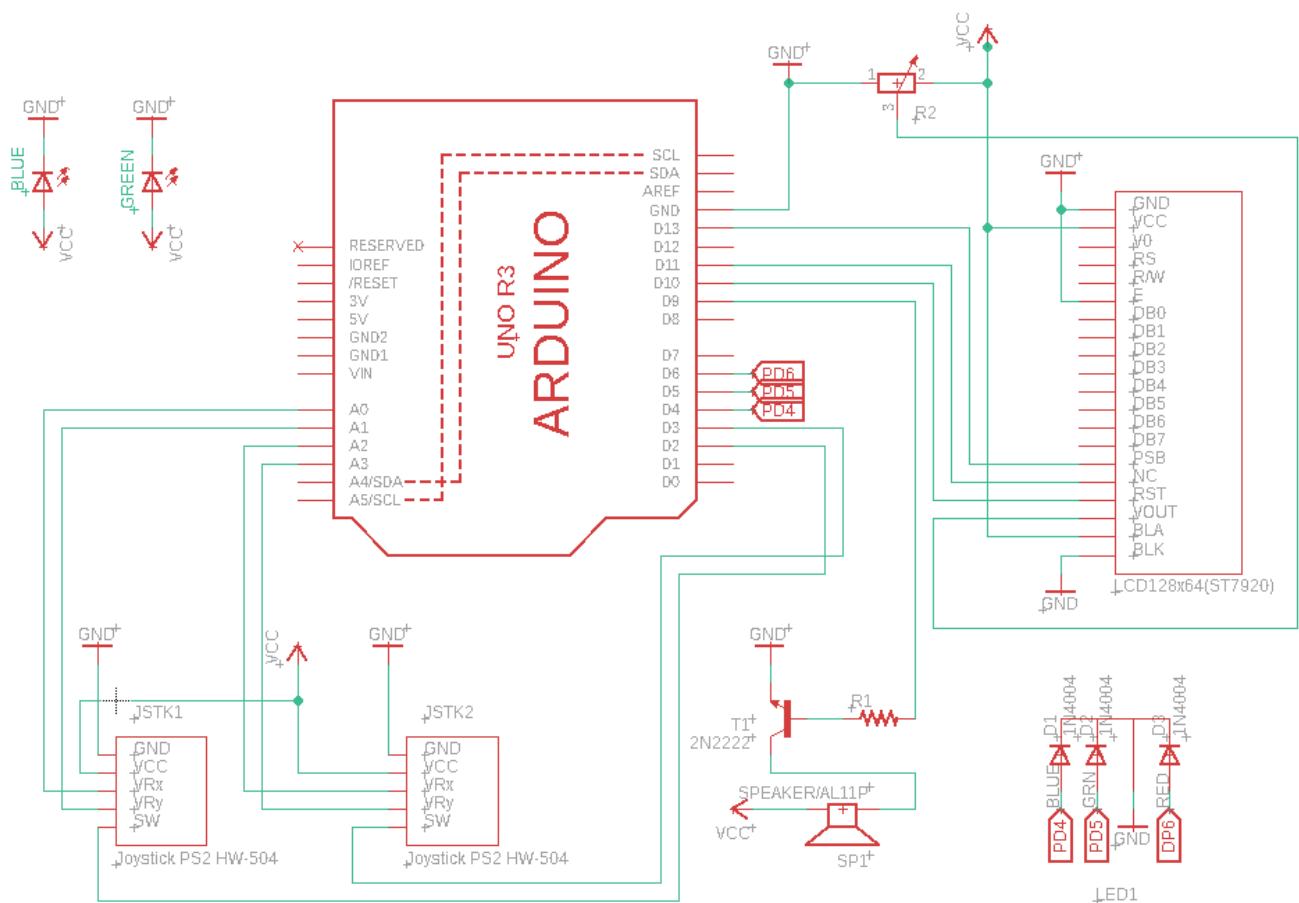
Schema bloc



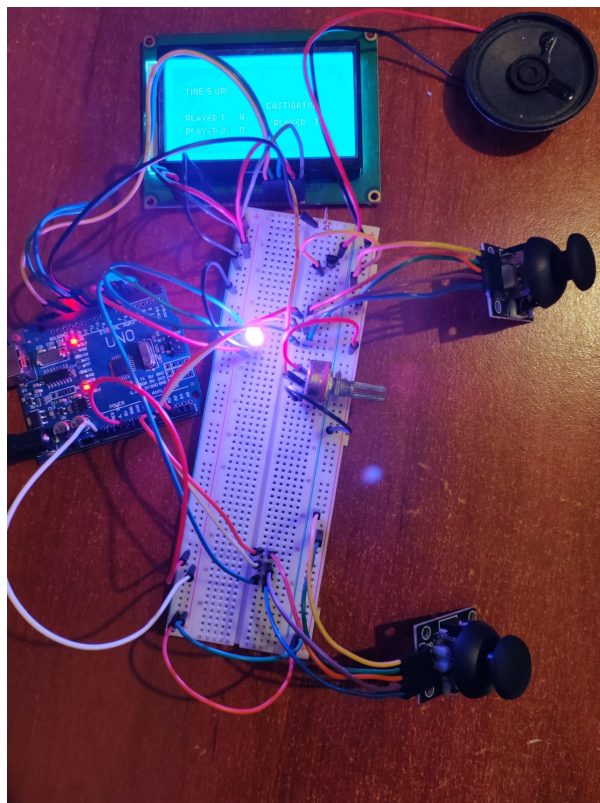
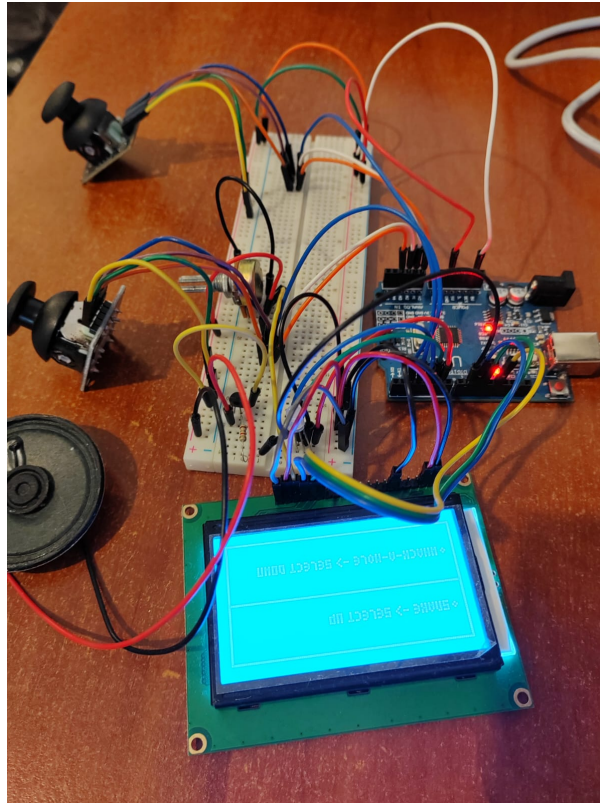
◆ Led RGB



Schema electrica finala



Implementarea hardware



Software Design

Mediu de dezvoltare: Arduino IDE

Biblioteci și surse 3rd-party:

- "Wire.h" pentru comunicarea I2C
- "U8glib.h" pentru controlul ecranului LCD
- "Bounce2.h" pentru gestionarea debouncing-ului butoanelor
- "TimerOne.h" pentru gestionarea unui timer
- <https://github.com/robsoncoute/arduino-songs> pentru muzica de fundal

Implementare software:

- Pini utilizati:

```
U8GLIB_ST7920_128X64 u8g(13, 11, 10, U8G_PIN_NONE); // pini digitali ai displayului -> SPI Com: SCK = 13, MOSI = 11, CS = 10
Bounce button1 = Bounce();
Bounce button2 = Bounce();

const int speakerPin = 9; // pinul digital pentru speaker
const int ledR = 6; // pinul digital pentru culoarea rosu a ledului RGB
const int ledG = 5; // pinul digital pentru culoarea verde a ledului RGB
const int ledB = 4; // pinul digital pentru culoarea albastru a ledului RGB

const int player1XPin = A0; // pinul analogic pentru coordonata x a joystickului 1
const int player1YPin = A1; // pinul analogic pentru coordonata y a joystickului 1
const int player1ButtonPin = 2; // pinul digital pentru butonul joystickului 1
const int player2XPin = A2; // pinul analogic pentru coordonata x a joystickului 2
const int player2YPin = A3; // pinul analogic pentru coordonata y a joystickului 2
const int player2ButtonPin = 3; // pinul digital pentru butonul joystickului 2
```

- Structuri folosite:

In jocul Snake, serpii sunt reprezentati de structuri de tipul:

```
#define SNAKE_MAX_LENGTH 128 // lungimea maxima la care poate ajunge un sarpe
const byte SCREEN_WIDTH = 128; // Matrix width
const byte SCREEN_HEIGHT = 64; // Matrix height

struct Snake {
    byte x[SNAKE_MAX_LENGTH]; // coordonata x a sarpei
    byte y[SNAKE_MAX_LENGTH]; // coordonata y a sarpei
    byte length; // lungimea sarpei
    byte direction; // directia in care se deplaseaza: 0 = up, 1 = right, 2 = down, 3 = left
    byte noSnake; // numarul de ordine sarpei: 1 = jucator 1, 2 = jucator 2
    unsigned long lastUpdateTime; // ultima data cand sarpele si-a modificat lungimea
} player1Snake, player2Snake;
```

- Functii implementate:

1. Generale:

- void setup()
- void loop()
- void playMusic() → *porneste muzica folosind note muzicale si duratele lor declarate intr-un vector de tip int*
- void readCoords() → *citeste coordonatele jucatorilor transmise de joystickuri si le scaleaza la dimensiunea displayului*
- void setColor(int redValue, int greenValue, int blueValue) → *seteaza culoarea ledului RGB setand valori pentru cele 3 culori RGB*
- void setRGBOff() → *opreste ledul RGB*
- void displayMenu(int player1X, int player1Y, int player1Score, int player2Score) → *afiseaza fereastra de meniu de unde jucatorii pot alege jocul*

2. Whack-a-Mole:

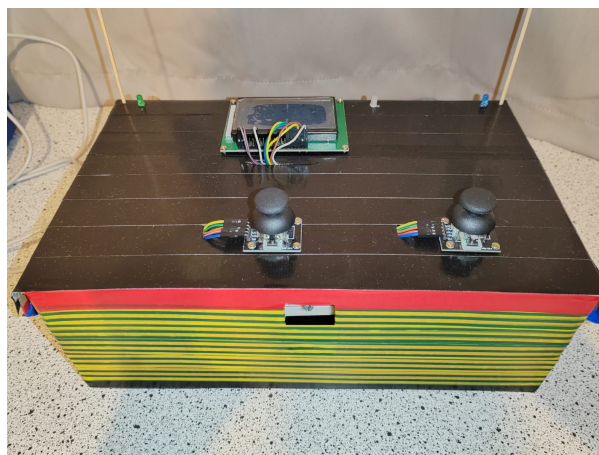
- void initializeMoleGrid()
- void spawnMole() → *adauga cartite noi in casute alese random*

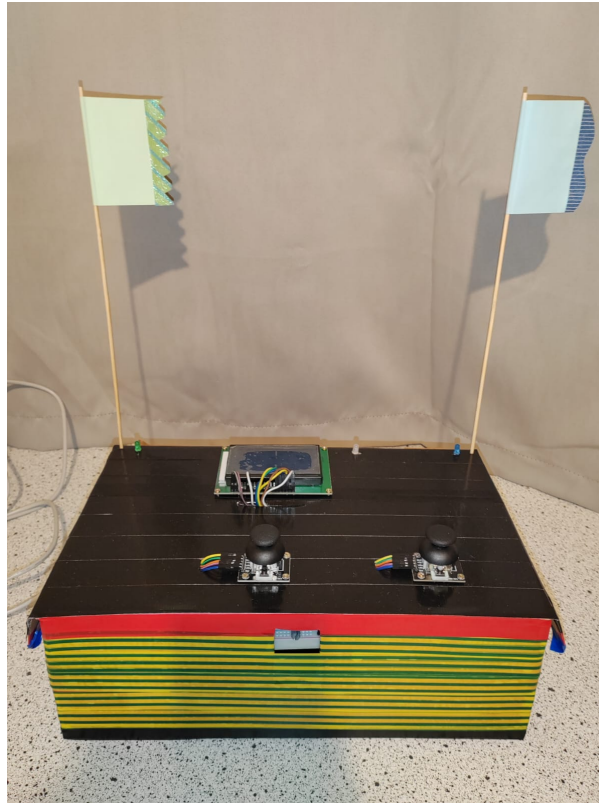
- void updateMoles() → *sterge cartitele au stat prea mult in joc sau care au fost prinse*
- void drawGrid() → *afiseaza tabla de joc de 2×4 formata din patrate*
- void drawMoles() → *afiseaza "cartitele" reprezentate de patrate verificand inainte daca casuta trebuie sa fie ocupata*
- void runWhackAMoleGame()
- void loadWhackAMole(int player1X, int player1Y, int player2X, int player2Y) → *afiseaza fereastra de incarcare a jocului*
- void playWhackAMole(int player1X, int player1Y, int player2X, int player2Y)
- void endGame() → *fereastra finala a jocului; aprinde ledul RGB intermitent in diferite culori, alege catigatorul in functie de punctaj, afiseaza pe ecran punctajele jucatorilor si cine a castigat*

3. Snake:

- void initializeSnakes()
- void drawSnakes()
- void updateSnake(struct Snake& snake, byte x, byte y)
- bool checkCollision(const Snake& snake1, const Snake& snake2) → *verifica daca cei doi serpi s-au intersectat*
- bool checkWallCollision(Snake& snake) → *verifica daca unul dintre serpi s-a lovit de unul dintre pereti*
- bool checkSelfCollision(const Snake& snake) → *verifica daca unul dintre serpi s-a intersectat cu el insusi*
- void runSnakeGame()
- void loadSnake(int player1X, int player1Y, int player2X, int player2Y) → *afiseaza fereastra de incarcare a jocului*
- void playSnake(int player1X, int player1Y, int player2X, int player2Y)
- void endSnakeGame() → *fereastra finala a jocului; aprinde ledul RGB intermitent in diferite culori, afiseaza pe ecran cine a castigat*

Design & Rezultate Obținute





Concluzii

- Am invatat sa folosesc pistolul de lipit cu fludor.
- Costurile pentru dezvoltarea proiectului au fost accesibile.
- A fost interesant sa creez un proiect pornind de la 0, sa invat lucruri noi, sa documentez fiecare pas si apoi sa vad rezultatul final.

Download

Arhiva contine codul sursa, bibliotecile folosite, schema electrica finala, schema display-ului, schema joystick-ului, README si prezentarea jocurilor.

[1v1gameconsolearchive.zip](#)

Jurnal

27.04 - Alegere temei de proiect

29.04 - Achiziționarea pieselor

05.05 - Crearea paginii Wiki

11.05 - Lipire pini display LCD și difuzor

13.05 - Am facut schema electrica a proiectului folosind Eagle

15.05 - Am facut si testat partea hardware a proiectului

18.05 - Am inceput sa lucrez la partea software

19.05 - Am reusit sa adaug muzica de fundal folosind un timer

21.05 - Am terminat de implementat jocul Whack-a-Mole

22.05 - Am facut LED-ul RGB sa lumineze intermitent la finalul jocului

25.05 - Am terminat de implementat jocul Snake 1v1

27.05 - Am inceput sa lucrez la design-ul proiectului

28.05 - Am adaugat un led verde si unul albastru care semnalizeaza culoarea fiecarui jucator

29.05 -

- Am terminat designul proiectului 🤖
- Am stabilizat joystickurile si displayul cu suruburi
- Am reparat cateva mici bug-uri 🛠️ **Fix Me!**
- Am finalizat proiectul! 😊

Bibliografie/Resurse

Resurse Software

Laburile de PM 😊

<https://github.com/olikraus/u8glib/wiki/userreference#nextpage>

<https://github.com/PaulStoffregen/TimerOne>

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

Resurse Hardware

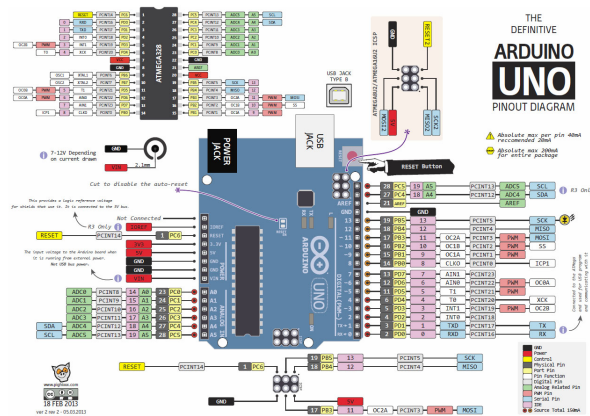
https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf

https://components101.com/sites/default/files/component_datasheet/Joystick%20Module.pdf

<https://drive.google.com/file/d/1JZtmwgvvPqeIQlOTkSLYoDFpSMMQ7dUe/view>

<https://ardushop.ro/ro/home/129-display-lcd-12864-pixeli-5v-iluminat.html>

* Arduino UNO pinout



Export to PDF

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

Permanent link:
<http://ocw.cs.pub.ro/courses/pm/prj2023/dene/1v1gameconsole>

Last update: **2023/05/30 11:35**

