

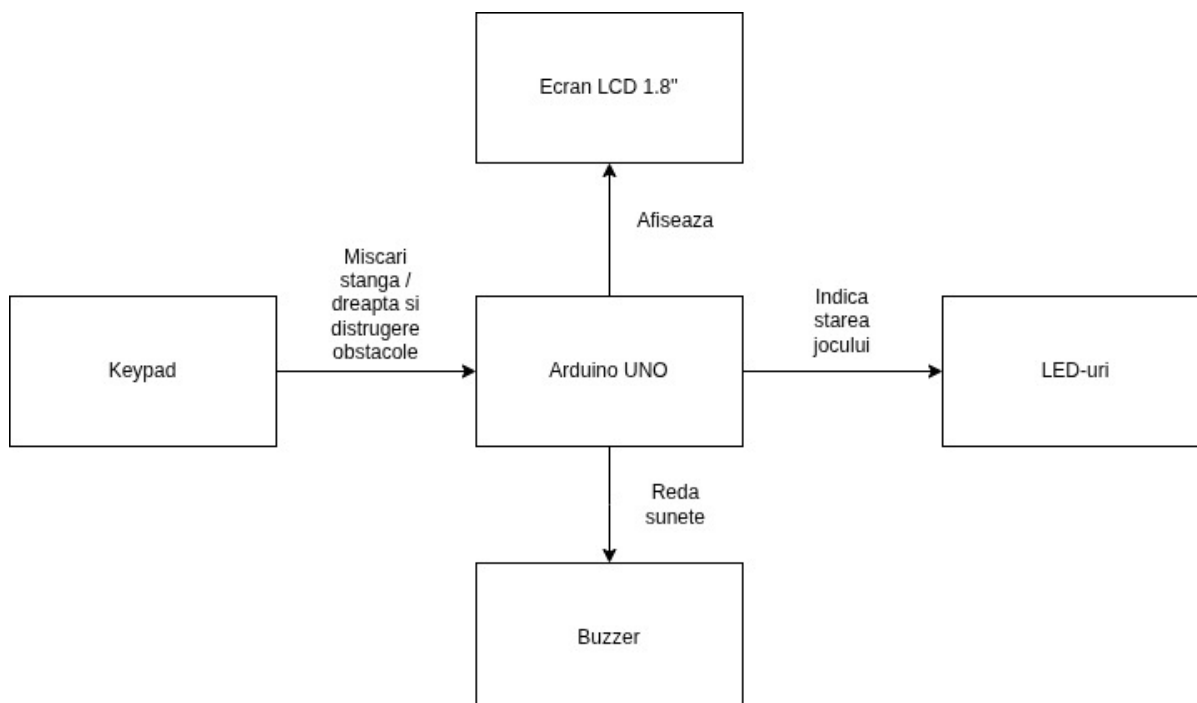
Avoidino - Destroy 'n Run

Introducere

Prezentare pe scurt:

- Un joc simplu inspirat puțin din jocul Subway Surfers, afișat pe un ecran LCD și controlat cu un keypad.
- Jocul constă în a evita obstacole prin mutări la stânga sau la dreapta, cu posibilitatea de a distruge obstacolele prin tragerea de gloanțe pentru a te feri și a câștiga mai multe puncte, toate prin intermediul keypad-ului.
- În timpul jocului, un LED de culoare verde va rămâne aprins, iar la pierderea jocului prin coliziunea cu un obstacol, se va stinge LED-ul verde și se va aprinde cel roșu.
- Se vor auzi zgomote la distrugerea obstacolelor și la pierderea jocului prin intermediul unui buzzer.
- Scopul jocului este pur recreativ.

Descriere generală



Hardware Design

Lista de componente utilizate:

- [Arduino Uno](#)
- [Buzzer](#)
- Breadboard
- Fire
- [Ecran LCD 1.8" SPI](#)
- [Keypad](#)
- Rezistențe
- 2 LED-uri RGB

Schema electrica:



Detalii despre pini:

- Alocarea pinilor s-a făcut cu scopul de a asigura compatibilitate cu perifericele folosite și de a permite un control eficient la nivel de registru atunci când este necesar.
- Ecranul TFT este conectat prin magistrala SPI, utilizând pinii standard pentru Arduino UNO: pinul digital 10 pentru CS (chip select), 9 pentru DC (data/command) și 12 pentru reset (RST).
- Tastatura matricială (keypad-ul) este conectată la pinii analogici A0-A4, împărțiți în patru pini de rând (A1-A4) și un pin de coloană (A0), ceea ce permite multiplexarea tastelor fără a consuma pini digitali suplimentari.
- Buzzer-ul este conectat pe pinul digital 3 (PD3), care corespunde ieșirii de tip PWM OC2B a Timerului 2. Alegerea acestui pin permite generarea de semnal audio controlat în mod direct prin setarea registrelor TCCR2A/B și OCR2A/B, în locul folosirii funcției `high-level tone()`.
- LED-urile roșu și verde sunt conectate pe pinii PD5 și PD6, controlați direct prin portul PORTD.

Software Design

Pentru realizarea componentei software a proiectului, am folosit mediul de dezvoltare Arduino IDE, împreună cu două biblioteci externe: `TFT.h` și `SPI.h`. Acestea sunt utilizate pentru afișarea graficii jocului pe un ecran TFT conectat prin interfață SPI, oferind funcționalități de desenare rapidă a textului și formelor.

Execuția codului este structurată în modul următor:

- Se declară variabilele globale necesare pentru poziția jucătorului, a glonțului, a inamicilor și a stării jocului (viață, scor, viteză inamic, coliziune).
- În funcția `setup()` se inițializează ecranul TFT, porturile de control pentru LED-uri, precum și sistemul de sunet. Tot aici se generează o valoare aleatorie pentru poziția inamicilor, folosind `analogRead()` pentru a inițializa `randomSeed()`.
- Jocul începe cu un ecran de tip intro, care explică comenzile de control: butoanele A și B mută jucătorul stânga/dreapta, C trage un glonț, iar D începe jocul.
- În cadrul funcției `loop()`, jocul rulează permanent: se citește tastatura, se mișcă inamicul și glonțul, se detectează coliziunile, se redă sunetul și se actualizează scorul și afișajul. Jocul se oprește în

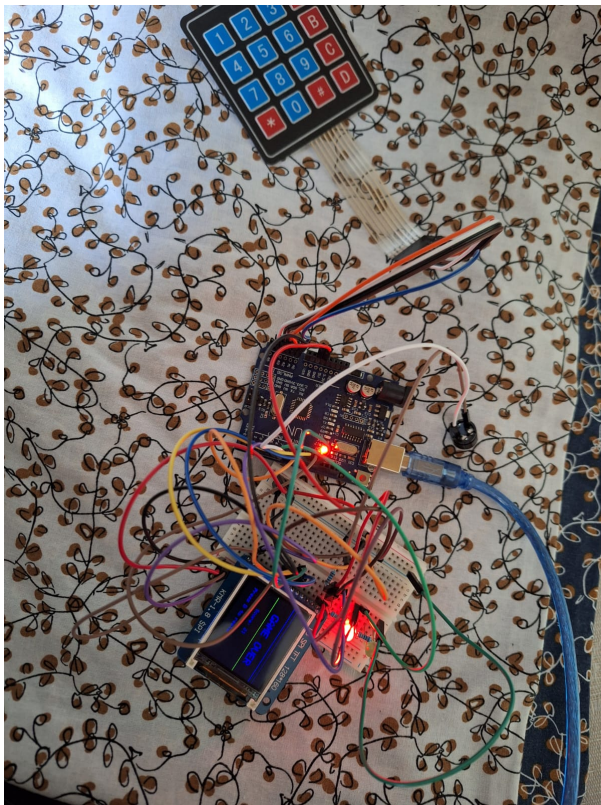
cazul în care se pierd toate viețile, afișând mesajul GAME OVER.

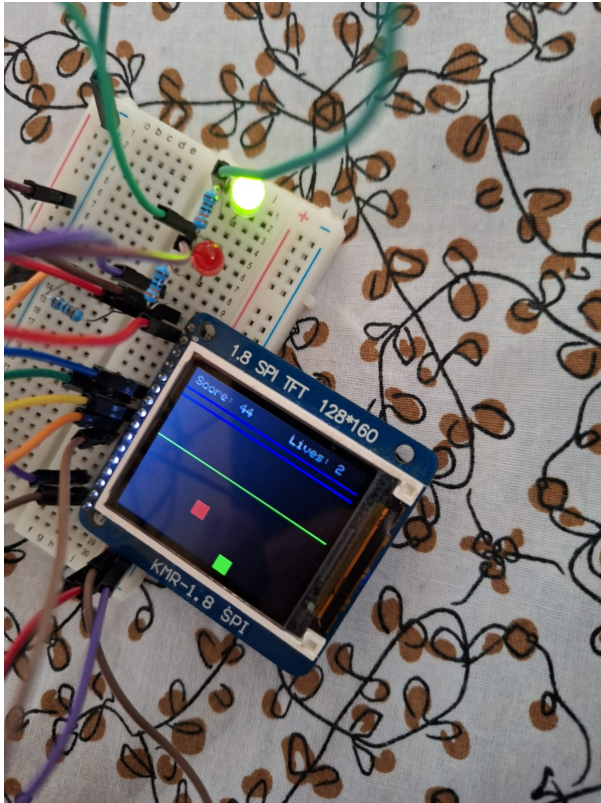
- Jocul funcționează pe un principiu simplu: jucătorul este poziționat în partea de jos a ecranului și poate trage proiectile pentru a distruge inamicii care cad de sus. La fiecare atingere de jos a unui inamic, scorul crește, iar la intervale fixe, viteza acestora se accelerează. La coliziune directă, jucătorul pierde o viață.

În scopul optimizării performanței și pentru o mai bună înțelegere a hardware-ului, au fost făcute mai multe modificări:

- S-a renunțat la utilizarea bibliotecii Keypad.h, implementând în schimb o citire manuală a tastaturii matriciale 4x1. Această abordare permite control complet asupra citirii și debouncing-ului.
- În locul funcției tone(), care este blocantă și abstractă, a fost implementat un sistem propriu de generare a sunetului, folosind Timer2 în mod Fast PWM direct pe pinul OC2B (pinul 3 / PD3). Sunetul este redat atât în mod non-blocking (în timpul jocului), cât și în mod blocking (pentru secvențele de la Game Over). Astfel, s-a reușit reducerea latenței în reacțiile vizuale și sonore.

Rezultate Obținute





Cod sursă - Download

[Link download cod sursă](#)

Demo

[Link Demo](#)

Export to PDF

[Export to PDF](#)

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

Permanent link:
<http://ocw.cs.pub.ro/courses/pm/prj2025/avaduva/andrei.cozma0903>

Last update: **2025/05/25 14:10**



