Jocul Snake

Introducere

Scurta descriere

Proiectul consta din crearea unui sistem de jocuri pe un ecran LCD, cu folosirea unui sistem de texturi tile-based, sistem de sunet, si control cu un joystick cu buton incorporat si cu salvarea scorurilor in EEPROM. Pe baza acestui sistem voi implementa un joc Snake, cu 3 dificultati.

Scopul proiectului

Scopul este crearea unui framework simplu pentru jocuri tile-based pe Arduino si implementarea unui joc Snake ca exemplu.

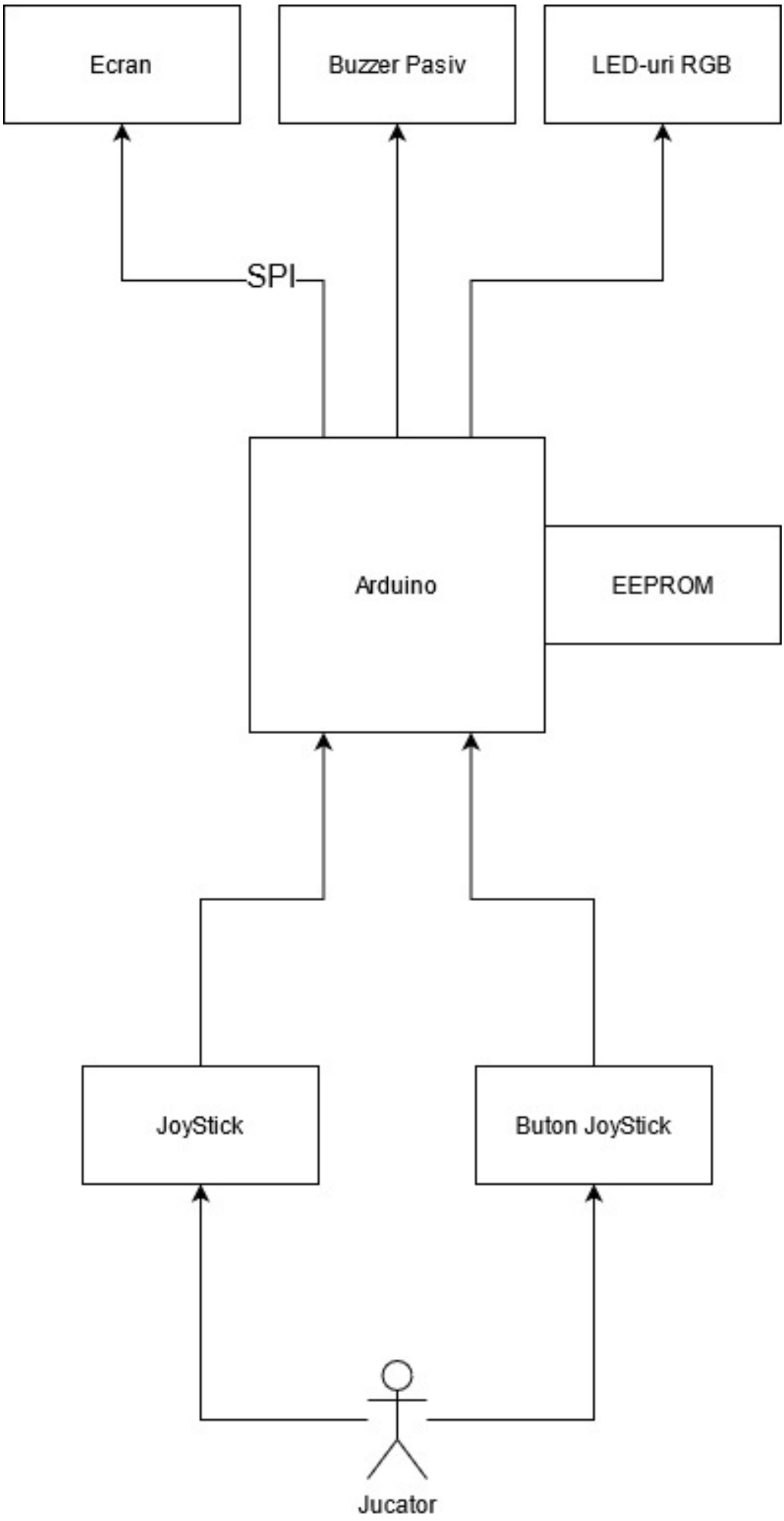
Inspiratia

Idea din spatele unui astfel de framework provine din jocurile clasice, ca exemplu Super Mario Bros. , Legend of Zelda sau Final Fantasy. In aceste jocuri, cea mai mare parte din partea grafica a jocului era compusa din tile textures, cu cateva elemente, de obicei inamici si caracterul jucatorului, care se puteau misca liber.

Descriere generală

O sa incerc sa creez un framework cat mai general de creare a unui joc bazat pe tile-uri, de tipul jocurilor clasice de NES si SNES. Pentru crearea jocurilor, dezvoltatorul va completa un numar de functii, unele chemate in setup(), altele in loop(), de tipul initializare texturi sau next frame. Folosind acest framework, o sa scriu un joc de Snake, cu 3 nivele de dificultate: jocul simplu, in care snake poate trece pe partea cealalta a ecranului, o versiunea in care marginea ecranului e un obstacol, si o versiunea la care, pe langa margine, avem si obstacole pe restul hartii, generate aleator.

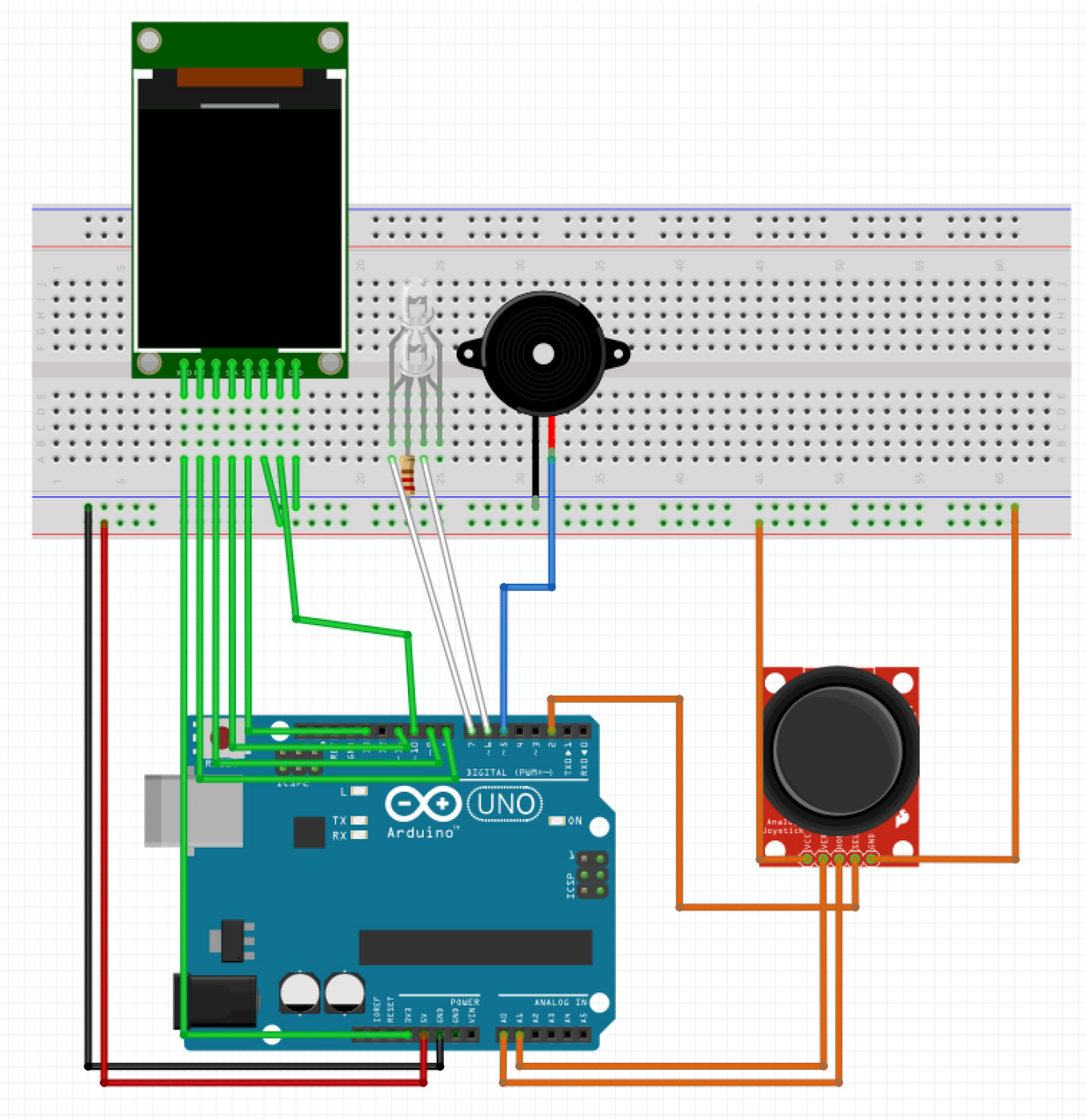
Schema bloc


Ecran
Buzzer Pasiv
LED-uri RGB
SPI
Arduino
EEPROM
JoyStick
Buton JoyStick
Jucator

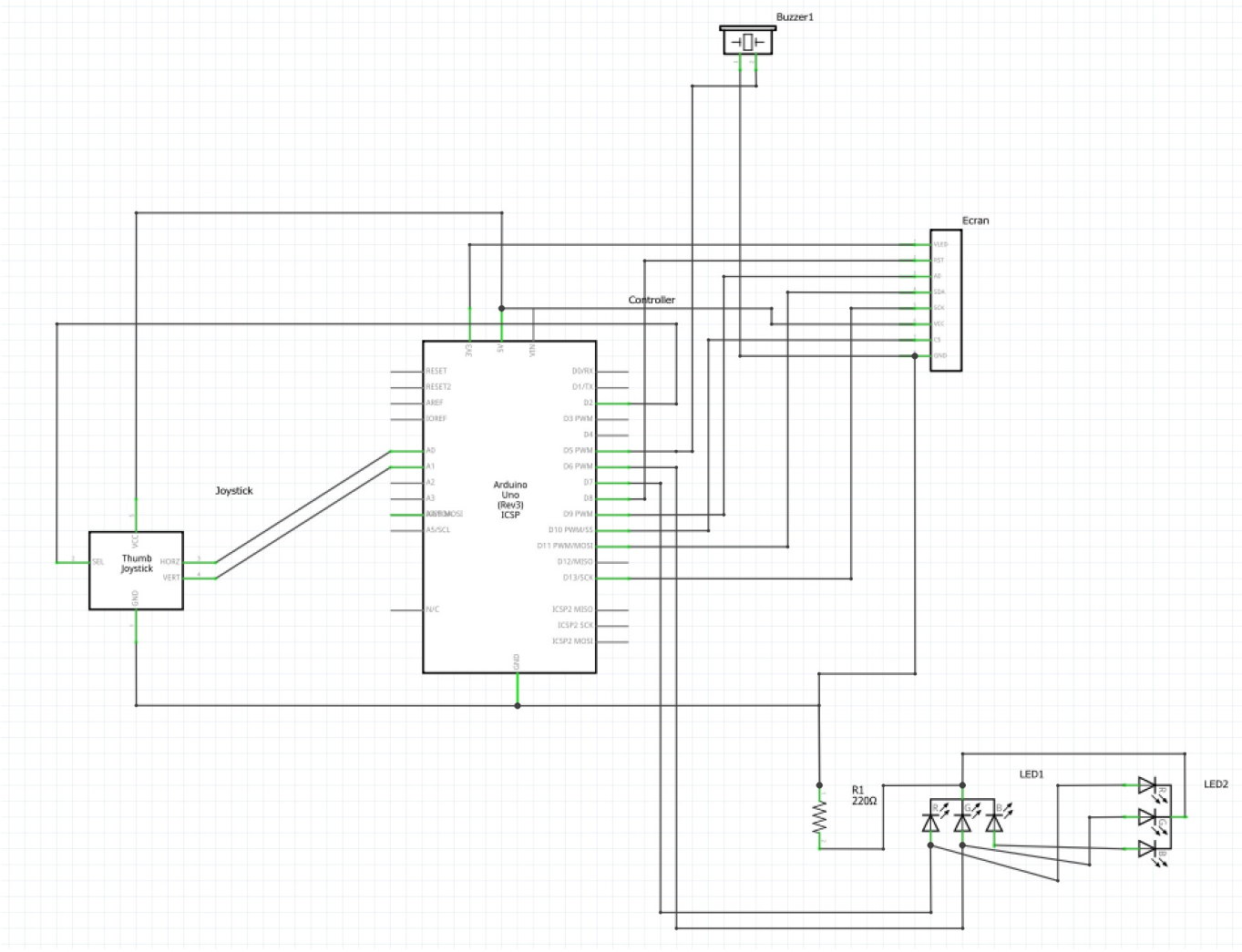
Hardware Design

- Arduino Uno
- Breadboard
- ecran LCD SPI
- joystick
- 1 buzzer
- fire
- led-uri RBG

Schema Hardware



Schema Electrica



Software Design

Programul urmareste structura standard a programelor Arduino, avand functiile setup si loop. In functia setup se initializeaza unele valori, dar cea mai mare parte a operatiilor se fac in loop, folosind o masina de stari. De asemenea, in loop ruleaza si partea de muzica.

Starile posibile

Level select

In aceasta stare se alege dintre cele 3 nivele posibile

Level init

Aici se initializeaza nivelul si se trece direct la Level running dupa o rulare a functiei.

In nivelul 1 nu avem nici un obstacol. In nivelul 2 terenul de joc este inconjurat de obstacole. In nivelul 3, pe langa zidul exterior, avem 4 obstacole in interior.

In oricare dintre cele 3 nivele se genereaza pozitia a 2 mere. Pentru a fi siguri ca nu generam un mar unde nu se poate, numaram cate tile-uri sunt libere pe parcursul jocului si apoi parcurgem vectorul de tile-uri pana la un tile liber random si generam acolo marul.

Din EEPROM, citim high-score-ul pentru nivelul selectat

Level running

Initial, calculam pozitia noua a capului sarpelui, daca acolo este un mar, il mancam si crestem in lungime, daca acolo este un obstacol, jocul se termina. Daca este un spatiu gol, sarpele se muta mai in fata cu o pozitie.

Score-ul este calculat astfel: Avem un scor maxim, initial de 100, ce creste cu 100 pentru fiecare mar mancat. La fiecare miscare a sarpelui, scorul creste cu 10, pana la scorul maxim.

Game Over

Daca avem un nou high score, il scriem doar odata in EEPROM. Afisam pe ecran GAME OVER, scorul nostru si high-score-ul.

Sistemul de tile rendering

Avem un vector ce retine ce tile corespunde unei anumite pozitii. Pentru a salva memorie, in loc sa avem o matrice de 16×16, avem un vector de 256 de pozitii. Salvam memorie deoarece putem retine un singur byte pentru pozitiile sarpelui, in loc de 2 coordonate. Cand setam un tile nou, il randam doar daca acesta este diferit de cel deja pe ecran. Astfel limitam timpul petrecut trimitand date ecranului.

Sistemul de muzica

Avem un vector ce retine notele si unul ce retine duratele. Folosim tone pentru durata corespunzatoare notei, iar dupa ce aceasta se termina, trecem la nota urmatoare.

Rezultate Obţinute

Link-ul la video-ul de prezentare Jocul Snake

Rezultatul este un ansamblu pe care se poate juca o versiune simpla a jocului Snake, cu muzica de fundal, si cu salvarea high-score-urilor in memoria EEPROM a Arduino-ului.

Concluzii

Sunt multumit de proiectul finalizat. Am avut unele dificultati cauzate de memoria mica a Arduino Uno. De exemplu, nu am mai putut folosi texturi luate de pe cardul SD sau sa salvez un istoric al scorurilor pe card. Biblioteca de carduri SD necesita multa memorie si am avut instabilitati la rulare.

In continuare, as putea sa fac 3 modificari majore:

- Sa adaug o sursa externa de 5V, astfel incat sistemul sa nu mai fie legat de calculator
- Sa lipesc componentele pe o placa, astfel aducand sistemul la o marime mai mica si facand-ul portabil. In momentul de fata, numerele de fire este destul de mare.
- Sa creez niste tile-uri mai creative, timpul de rendering al unui frame este destul de mic si as putea folosi niste functii mai complexe

Download

snake_arduino_uno_st7735_melinte_paul_eduard.zip

Jurnal

1. 02/05/2021 Finalizare alegere si descriere generala a proiectului
2. 19/05/2021 Am programat basic gameplay loop-ul si cele 3 nivele de dificultate
3. 23/05/2021 Am updatat schema bloc si piesele necesare, am adaugat schema hardware si schema electrica. Am adaugat la bibliografie
4. 31/05/2021 Am terminat implementarea
5. 1/06/2021 Am finalizat documentatia

Bibliografie/Resurse

Documentația în format PDF

- https://ocw.cs.pub.ro/courses/pm/prj2021/dbrigalda/snakegame
- https://github.com/adafruit/Adafruit-GFX-Library
- https://github.com/adafruit/Adafruit-ST7735-Library
- https://www.electronics-lab.com/project/using-st7735-1-8-color-tft-display-arduino/
- https://github.com/robsoncouto/arduino-songs/blob/master/tetris/tetris.ino