

Labirint

Nume: Popa Ioana-Theodora

Grupă: 336CB

Introducere

Proiectul meu este o interpretare a unui joc fizic, în care există un labirint, iar scopul este de a duce o minge dintr-un punct de plecare până în alt punct stabilit. Problema care apare este faptul că într-un astfel de joc fizic, labirintul nu poate fi schimbat des, așa că introducerea unei variante digitalizate ar putea să varieze mult mai des labirintul.

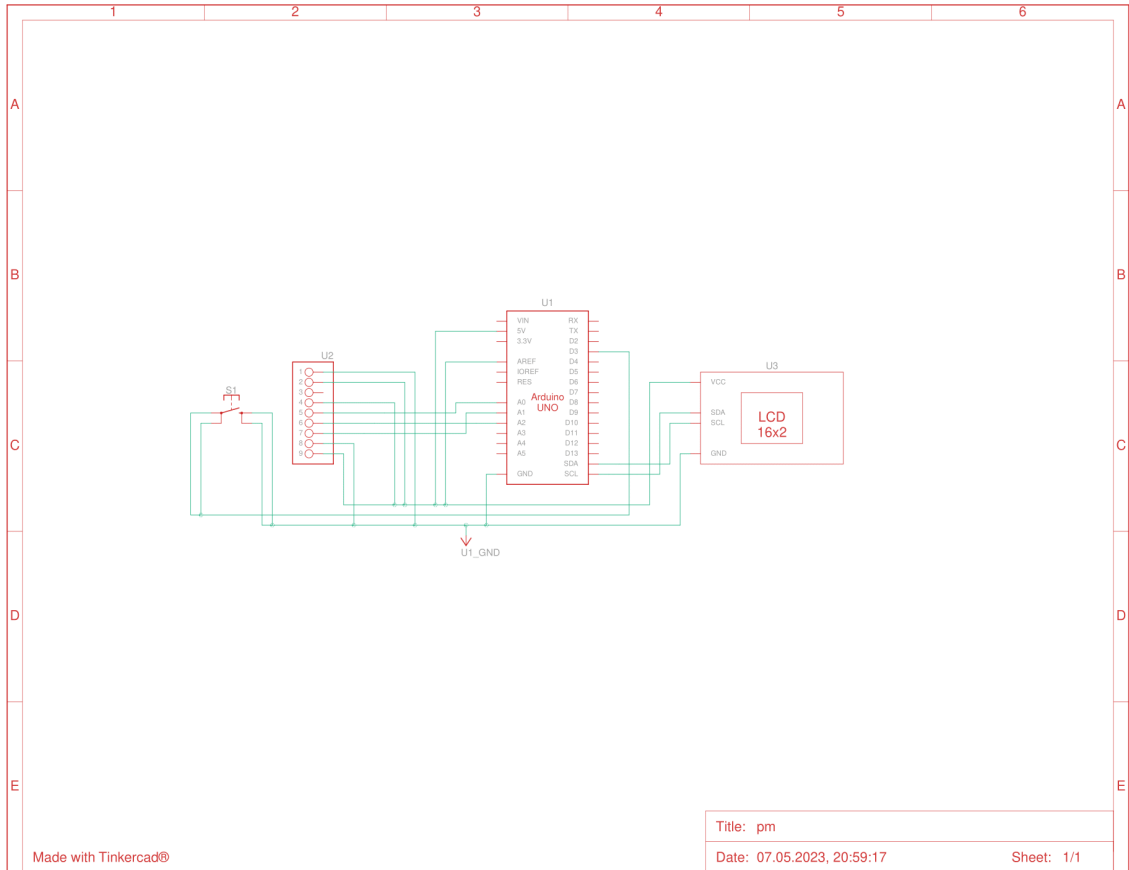
Descriere generală

Jocul va începe prin apăsarea unui buton. Mingea va fi afișată pe ecran pe o poziție inițială și utilizatorul poate să înceapă să o miște. Acest lucru se face prin mișcarea fizică a ecranului. Pentru a prelua informațiile asupra poziției și unghiului de inclinare din mediul extern, voi folosi un giroscop și accelerometru. Datele vor fi transmise și poziția mingii pe ecran se va modifica corespunzător. Astfel, jucătorul poate manipula poziția mingii și să o ducă în punctul final. De asemenea, în cazul coliziunii cu un perete al labirintului, mingea nu va fi lăsată să pătrundă în acea porțiune. Jocul se va desfășura într-un timp limită, iar la final, după expirarea acestuia sau după câștigul jucătorului, se va afișa un mesaj corespunzător.

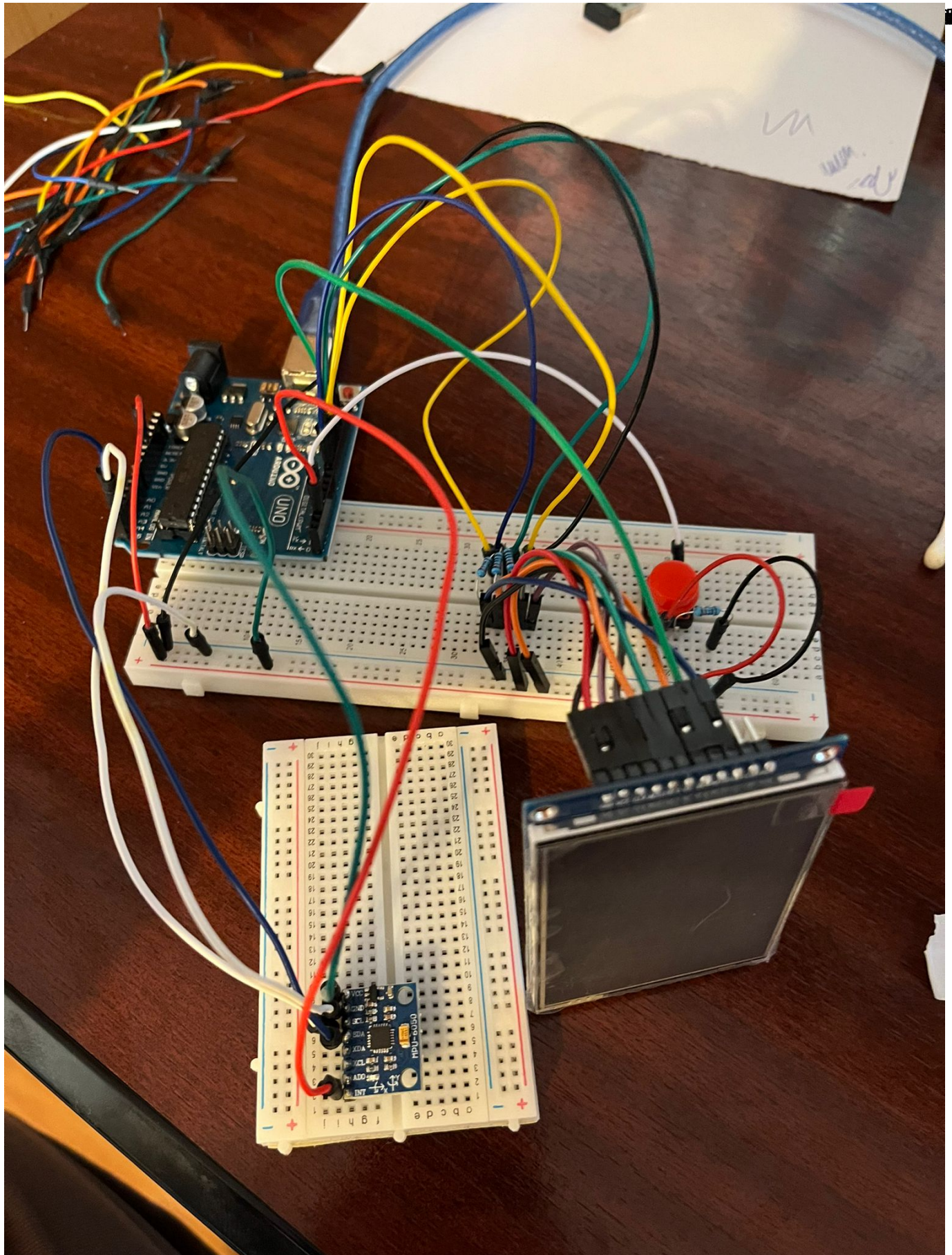
Hardware Design

Piese folosite pentru acest proiect sunt:

- Arduino UNO (ATMega328p)
- Ecran LCD 2.8" TFT 240×320
- Giroscop și Accelerometru MPU6050
- Breadboard
- Buton
- Fire de legătură



În cazul de față, am conectat LCD-ul, giroscopul și butonul la Arduino Uno. Deoarece modelul de LCD



Software Design

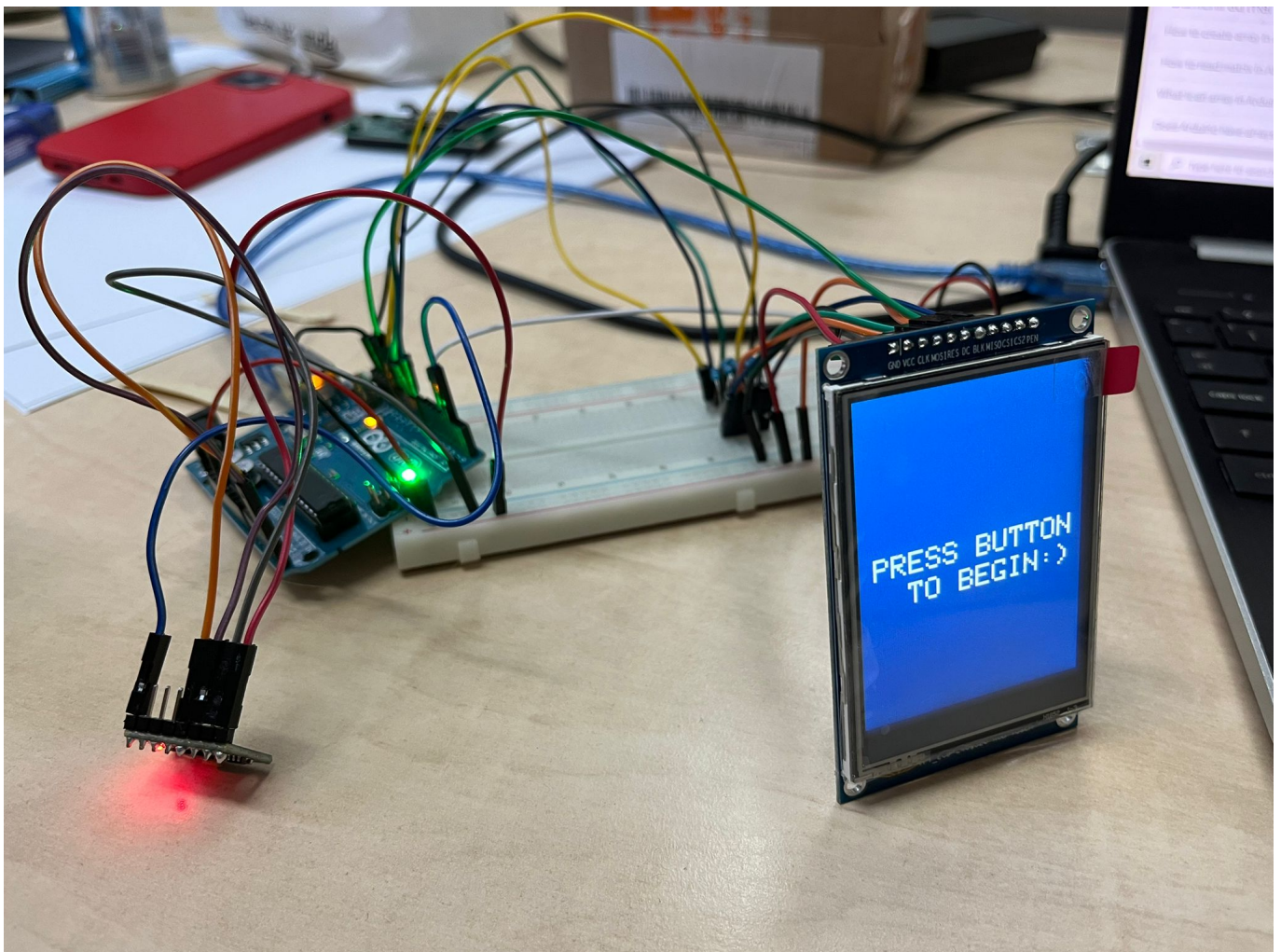
Jocul se imparte in patru stari de joc:

- initializare: se randează background-ul de start cu textul "PRESS BUTTON TO BEGIN!"
- draw labyrinth: la apăsarea butonului de pe plăcuță, va avea loc o întrerupere prin intermediul căreia se intră din starea de inițializare la starea de creare a labirintului și poziționare a mingii
- start game: jocul efectiv începe, iar giroscop-ul transmite constant date despre unghiul în care se află. Pe baza acestuia, se randează poziția mingii, adăugând la poziția mingii un echivalent pentru unghiul în care se află. În cazul coliziunilor, jocul este pierdut și se trece în starea finală. Dacă ajunge la punctul final, jocul este câștigat.
- end game: va arăta rezultatul jocului pe baza unei variabile.

Pentru implementare, am folosit două biblioteci:

- MPU6050_tockn.h - pentru a obține datele de la giroscop și a le folosi ulterior
- LCDWIKI_GUI.h - pentru a lucra cu LCD-ul

O problemă pe care am întâmpinat-o a fost generarea efectului de mișcare a bilei fără spike-uri, deoarece exista constant un lag. Am rezolvat această problemă făcând mingea mai mică și punând un delay între un frame și altul, dând o senzație aproximativ reală.





Rezultate Obținute

În final, am obținut un joculeț interesant, o interpretare a celui fizic, care permite adăugarea ușoară a altor moduri de joc, mișcarea mingii oferind senzația de realitate, iar coliziunile de asemenea.

Concluzii

Prin intermediul acestui joc, am reușit să învăț mai multe despre citirea unei documentații pentru anumite piese, despre interacțiunea pieselor cu plăcuța de Arduino și despre cum trebuie abordate anumite probleme în această instanță.

Download

proiect_pm.zip

Jurnal

- 27 aprilie - alegerea temei proiectului
- 4 mai - relizarea documentației
- 5 mai - alegerea pieselor necesare pentru proiect
- 10 mai - comandarea pieselor
- 17 mai - începerea implementării hardware
- 18 mai - terminarea implementării hardware
- 18 mai - începerea implementării software
- 25 mai - terminarea implementării software
- 26 mai - update wiki

Bibliografie/Resurse

- [Conectare LCD la Arduino](#)
- [Interfața MPU6050](#)
- [Bibliotecă pentru Interfața LCD Arduino](#)
- [Bibliotecă pentru Interfața Modul MPU6050 și Arduino](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2023/iotelea/labirint>



Last update: **2023/05/29 18:09**