

Pinball Game

- Autor: Ștefana-Ioana Dranca
- Grupa: 334CC

Introducere

Proiectul este o implementare a clasicului joc de pinball. Jocul conține mai multe ținte, ce odată lovite cresc scorul jucătorului (afișat pe ecranul LCD). Jucătorul lovește mingea cu ajutorul unor flippere, acționate prin butoane. Odată ce mingea pătrunde printre flippere de 3 ori sau jucătorul acumulează un scor maxim jocul se încheie!

Scopul proiectului este divertismentul celor care vor să joace acest arcade game iubit de generații întregi!

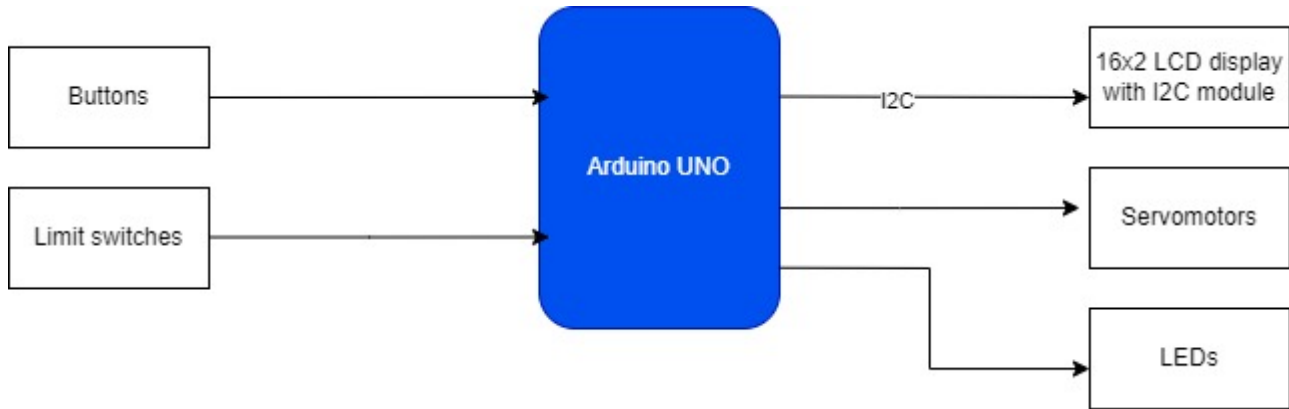
Descriere generală

Se acționează un buton de start, pentru a porni jocul. Pe tot parcursul lui, scorul curent este vizibil pe ecranul LCD. Mingea va fi propulsată printr-un mecanism mecanic în spațiul de joc. Spațiul de joc este construit din carton, sub forma unui plan înclinat. Jucătorul va acționa prin două butoane laterale flipper-ele. Flipper-ele sunt montate pe servomotoare. De fiecare dată când un buton este apăsat, flipper-ul corespunzător se rotește cu un număr de grade (același pe tot parcursul jocului), după care revine în poziția inițială.

Țintele pe care mingea trebuie să le atingă vor avea punctaje diferite, în funcție de cât de accesibile sunt, iar pentru a detecta când sunt lovite, acestea vor conține limit switch-uri.

Atunci când mingea trece printre flippere de trei ori, jocul se încheie, iar jucătorul pierde. Detectarea acestui incident se face tot cu ajutorul unui limit switch montat în planul înclinat, în zona de ieșire. Dacă jucătorul acumulează punctajul maxim înainte de a scăpa mingea de 3 ori printre flippere, acesta câștigă!

Schema bloc



Hardware Design

Listă componente:

- Arduino UNO
- Butoane (x 3)
- Servomotor (x 2)
- Modul LCD 1602 cu I2C
- Limit switch (x 4)
- LED
- Rezistență de 220Ω
- Rezistență de 10KΩ (x 4)
- Fire Dupont mamă-mamă, mamă-tată, tată-tată

Schema electrică:



În implementarea schemei de pe Tinkercad am folosit slide switch-uri, deoarece nu există limit switch-uri în Tinkercad. Slide switch-urile au, la fel ca limit switch-urile, 3 pini: COM, NO și NC.

Software Design

- Mediu de dezvoltare: Arduino IDE
- Biblioteci folosite: Servo.h pentru controlul servomotoarelor, LiquidCrystal_I2C.h și Wire.h pentru comunicarea cu ecranul LCD prin I2C

Detalii de implementare

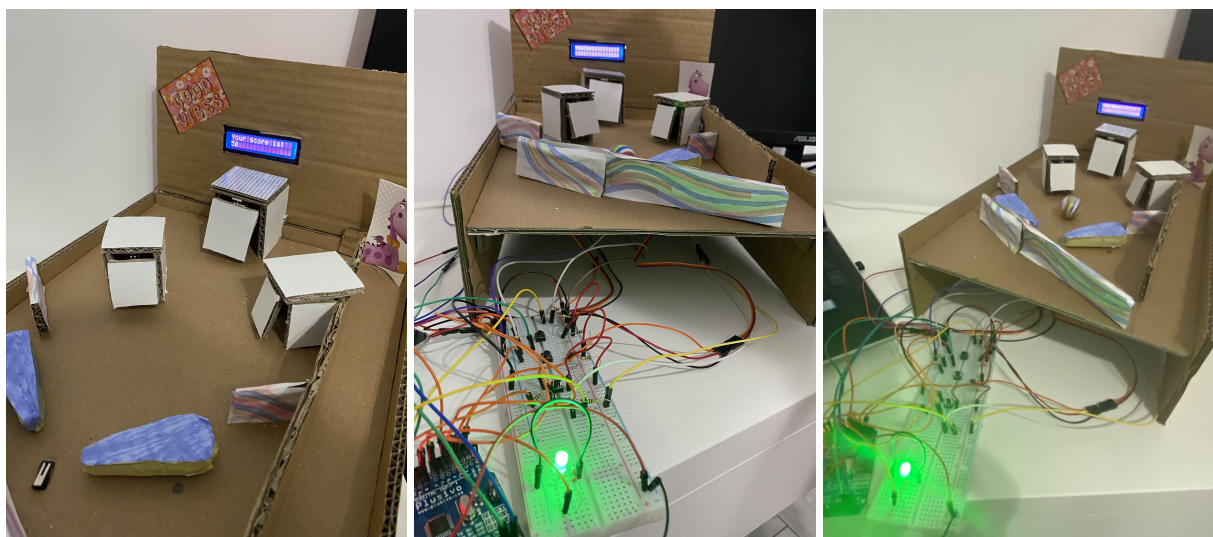
În **setup()** am configurat pinii de input și output și lcd-ul, am atașat pinii corespunzători variabilelor de tip Servo prin care se controlează cele două servomotoare. În **loop()** se apelează mai întâi funcția **checkStart()** prin care jocul poate fi pornit sau repornit prin apăsarea butonului de start. Dacă jocul este pornit se pot acționa cele două servomotoare prin apăsarea butoanelor (**buttonControl()**), se verifică toate target-urile (limit switch-uri) prin funcțiile corespunzătoare, dar și dacă jucătorul a atins un scor maxim (**checkWin()**) sau dacă mingea a ieșit din joc (**checkSwOut()**). Dacă target-urile sunt atinse scorul este incrementat. Dacă este atins switch-ul montat la ieșirea din spațiul de joc, înainte de atingerea scorului maxim, jucătorul pierde. Dacă jucătorul atinge scorul maxim a câștigat!

Pe parcursul jocului ecranul LCD afișează status-ul jocului: "Game started!", scorul curent, "You won!" sau "You lost!". Pentru afișarea mesajelor pe display a fost necesar să găsim mai întâi adresa I2C a acestuia, cu ajutorul unui loop for care a iterat printre adresele posibile și al funcțiilor **Wire.beginTransmission(address)** și **Wire.endTransmission()**. După găsirea adresei (0x27), am eliminat această bucată de cod.

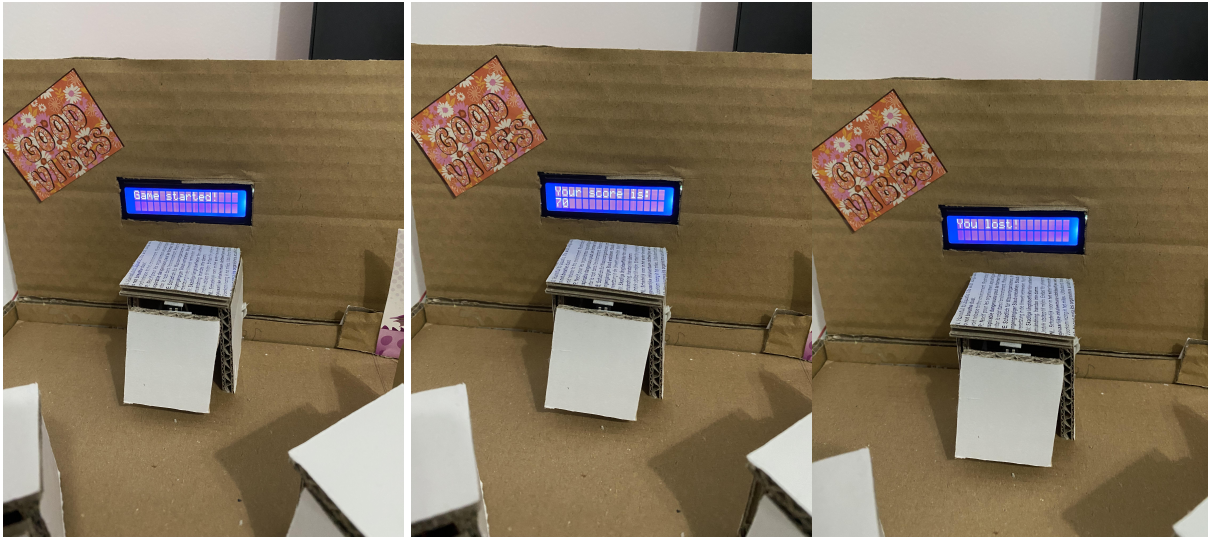
În funcțiile care detectează dacă switch-urile au fost acționate, am folosit tehnica de **debouncing** pentru a evita rezultate neașteptate, cum ar fi incrementarea scorului de mai multe ori atunci când un switch este apăsat o singură dată. Pentru a obține timpii când au fost acționate switch-urile am utilizat funcția **millis()**.

Rezultate Obținute

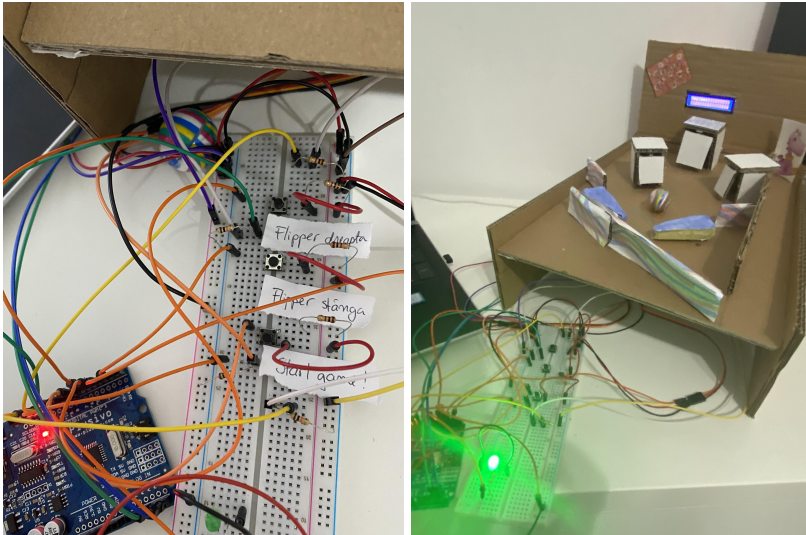
Masa de joc



Mesaje display LCD



Butoane și led



[Link Demo](#)

După mai multe încercări de poziționare a elementelor și multe ore de tăiat și lipit carton, am reușit să construiesc masa de joc și să integrez toate piesele. Atât motoarele, cât și display-ul LCD și limit switch-urile funcționează corect. Mingea este detectată de cele mai multe ori de către limit switch-urile din target-uri, atunci când le lovește cu viteză. Mingea nu este detectată însă de limit switch-ul montat la ieșirea din tabla de joc din cauza greutateii reduse a bilei și lovește rar țintele de 25 respectiv 50 de puncte. Switch-ul de la ieșire funcționează corect atunci când este acționat de o forță mai mare.

Concluzii

Am lucrat cu plăcere la proiect, deoarece am învățat lucruri noi și am pus în practică cunoștințe obținute de la laborator, iar asamblarea și testarea proiectului a fost amuzantă. Am văzut importanța

rezistențelor de pull-up și pull-down, dar și a verificării circuitelor cu ajutorul multimetrului. Am realizat cât de necesară este o analiză a componentelor electrice, dar și a celorlalte elemente ce se folosesc într-un proiect real (din punct de vedere al greutateii, al design-ului, al materialelor din care sunt făcute). Am avut câteva dificultăți la integrarea tuturor elementelor și la cablarea lor, deoarece dimensiunea jocului este destul de mare, iar componentele sunt numeroase, însă le-am depășit. Am încercat mai multe mingi, iar cea aleasă are rata cea mai mare de acționare a limit switch-urilor.

Proiectul a însemnat de asemenea o ocazie bună de re folosire a cartoanelor mai vechi din casă.

Download

- Cod sursă, README și schemă electrică: [pinball_game.zip](#)

Jurnal

11.05.2022: Realizare schemă pe Tinkercad și schemă electrică

12.05.2022: Realizare schema bloc

20.05.2022: Testarea primei versiuni de cod pe Tinkercad

21.05.2022 - 22.05.2022: Asamblare circuit și testare a componentelor

23.05.2022 - 27.05.2022: Construcția mesei de joc

27.05.2022: Finalizare pagină wiki

Bibliografie/Resurse

Resurse Software:

- <https://create.arduino.cc/projecthub/abdularbi17/how-to-scan-i2c-address-in-arduino-eaadda>
- <https://www.arduino.cc/en/Tutorial/BuiltInExamples/Debounce>
- <https://www.arduino.cc/reference/en/libraries/servo/>
- <https://arduinogetstarted.com/tutorials/arduino-lcd-i2c>
- <https://ocw.cs.pub.ro/courses/pm/lab/lab3-2022>

Resurse Hardware:

- <https://create.arduino.cc/projecthub/akshayjoseph666/interface-i2c-16x2-lcd-with-arduino-uno-just-4-wires-273b24>
- <https://arduinogetstarted.com/tutorials/arduino-limit-switch>

[Export to PDF](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2022/bogdanc/pinball-game>



Last update: **2022/05/27 21:48**