

Simon Says

Autor:

- [Mara-Ioana Nicolae](#)

Introducere

Proiectul implementează o versiune single-player a popularului joc „Simon Says”, rolul lui Simon este de a transmite o serie de semnale pe care utilizatorul trebuie să le refacă testându-și atenția și precizia. Cu ajutorul unui joystick și a unui ecran LCD, jucătorul va trebuie să refacă secvența dictată de către Simon.

Meniul principal va fi afișat pe ecranul LCD, iar preferințele utilizatorului vor fi selectate tot cu ajutorul joystick-ului. Astfel se poate selecta durata unui semnal și lungimea secvenței ce trebuie reprodusă. Jocul devine mai complicat cu cât durata dintre două semnale consecutive este mai mică, iar lungimea secvenței este mai mare.

Descriere generală

Jocul începe odată ce utilizatorul va apăsa joystick-ul, în acest moment va fi întâmpinat de prompt-ul pentru preferințe; nivelul de dificultate pentru fiecare rundă putând fi configurat. Se va dicta o secvență de semnale, când pe ecran va apărea mesajul de finalizare al secvenței, jucătorul va fi invitat să refacă secvența manevrând joystick-ul în direcțiile corespunzătoare. În funcție de input-ul jucătorului, se vor emite semnale sonore pentru a anunța dacă secvența a fost reprodusă corect sau nu.

Schema bloc



Hardware Design

Componente folosite

- Arduino UNO
- Breadboard
- Joystick
- LED-uri
- Butoane
- Ecran LCD 1602 cu interfață I2C
- Buzzer pasiv

Interacțiunea cu jucătorul

Instrucțiunile jocului vor fi redade prin **ecranul LCD**, jucătorul fiind nevoit să acorde atenție mesajelor transmise de către acesta. În meniul de start, se poate configura dificultatea runde curente. Totodată, secvențele alese de Simon vor apărea pe ecran pentru o perioadă scurtă, timp în care utilizatorul va trebui să rețină secvența dictată.

Prin manevrarea **joystick**-ului utilizatorul va putea controla direcția dorită; în funcție de coordonatele citite pe axa verticală și cea orizontală se va determina poziția aleasă și se va verifica dacă se potrivește cu cea dictată de Simon. De asemenea, tot prin intermediul joystick-ului se va seta dificultatea runde; prin mișcarea la stânga și la dreapta se vor seta lungimea secvenței ce se va reproduce și durata dintre două simboluri consecutive. Butonul integrat în joystick are rolul de a confirma alegerea jucătorului pentru o dificultatea configurată.

Pentru selectarea unei culori se vor utiliza cele patru **butoane**, la apăsarea unuia dintre acestea se va aprinde și **LED**-ul colorat corespunzător pentru a înștiința utilizatorul în legătură cu opțiunea aleasă.

Buzzer-ul va oferi sunete de alertă pentru a atenționa jucătorul în trei scenarii diferite:

- începerea unei noi runde
- decodificarea greșită a unuia din simboluri și pierderea runde
- câștigarea runde

Schema electrică



Software Design

Pentru fiecare început de rundă, se va emite prin buzzer secvența sonoră corespunzătoare prin funcția *startSound* și se va verifica valoarea variabilei **gameOn** ce indică dacă jocul a început sau nu. Runda va începe odată cu apăsarea butonului atașat joystick-ului; pe ecranul LCD vor apărea instrucțiunile necesare pentru începerea jocului în funcția **printStartMessage** cu ajutorul bibliotecii **LiquidCrystal_I2C**.

Următorul pas din fluxul aplicației este reprezentat de meniul de configurare al runde, prezent în funcția **chooseDifficulty**. În funcție de preferințele sale jucătorul va putea ajusta lungimea secvenței dictate și perioada dintre două simboluri personalizând nivelul de dificultate. După ce nivelul a fost configurat, Simon va începe să dicteze o secvență de simboluri și codificarile acestora.

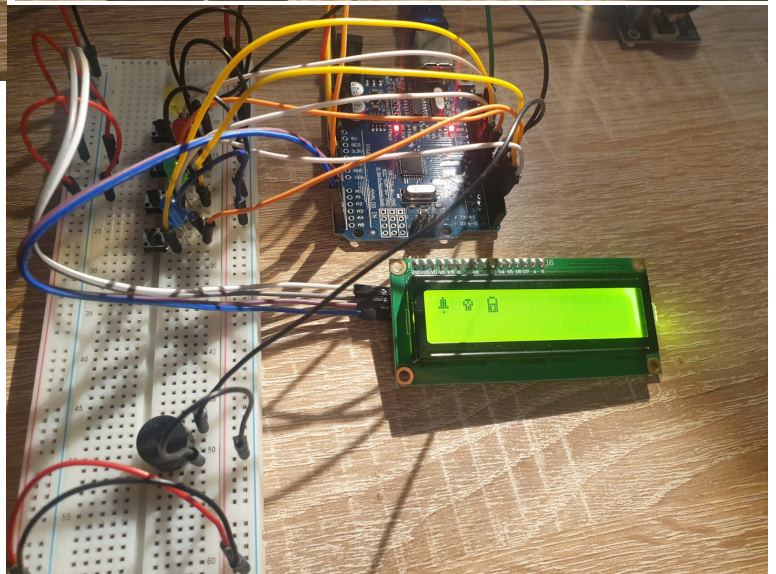
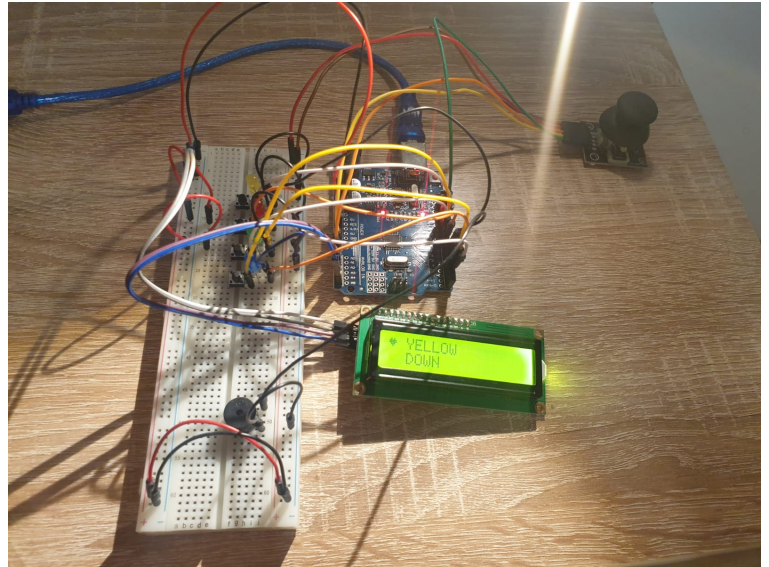
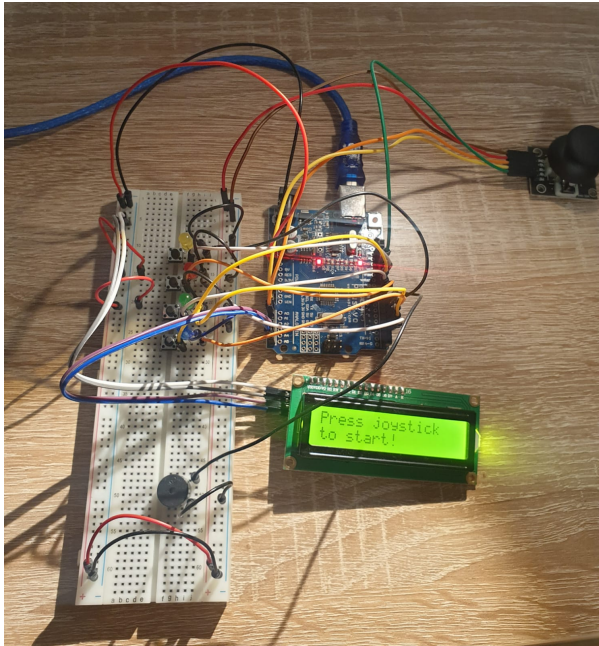
Întreaga logică a jocului se regăsește în interiorul funcției **playGame**, folosind mesaje intuitive, utilizatorul va naviga prin seria de instrucțiuni pentru a încerca să câștige runda. Se vor alege aleator simbolurile (**chooseEmoji**), culorile (**chooseColor**) și pozițiile (**choosePosition**), apoi va fi redată din nou secvența completă formată doar din simboluri.

Jucătorul își va pune la încercare memoria și acuratețea încercând să refacă secvența dictată de Simon. Prin citirea valorilor de pe axele verticale și orizontale ale joystick-ului se va analiza direcția aleasă de jucător, iar culoarea se va identifica în funcție de butonul ales. În funcție de rezultatul final se va afișa un mesaj pe ecranul LCD, împreună cu o secvență sonoră care anunță câștigarea runde prin **winSound**, sau pierderea acesteia prin **failSound** folosind header-ul **pitches.h**. Jocul va

reîncepe automat o nouă rundă.

Rezultate Obținute

Montajul complet al componentelor



Demo

[Demo proiect](#)

Cod sursă

simonsays.ino

Jurnal

- **18 aprilie - 25 aprilie:** aprobarea proiectului, setarea obiectivelor și achiziționarea componentelor necesare
- **25 aprilie - 2 mai:** realizarea unei scurte descrieri și a schemei bloc
- **2 mai - 9 mai:** acomodarea cu Arduino IDE, testarea componentelor
- **9 mai - 20 mai:** adăugarea funcționalităților:
 1. determinarea direcției alese de jucător prin interpretarea poziției joystick-ului și conectarea butoanelor la LED-uri
 2. afișarea instrucțiunilor prin ecranul LCD
 3. personalizarea a trei mesaje sonore ale buzzer-ului ce indică: începerea unui nou joc, câștigarea unei runde sau identificarea greșită a secvenței
 4. construirea secvenței de simboluri ce conține elemente unice
- **20 mai - 23 mai:** completarea schemei electrice, introducerea meniului principal prin care jucătorul poate personaliza gradul de dificultate al jocului
- **23 mai - 28 mai:** testarea jocului și depanarea erorilor, realizarea documentației

Concluzii

Sunt mândră de rezultatul obținut în urma implementării acestui proiect, deși ideea nu are o mare aplicabilitate sau impact în viață de zi cu zi. Mă bucur că am reușit să îmbin elemente hardware și software, fiind o experiență nouă pentru mine cu primul meu proiect Arduino.

Proiectul este funcțional și am reușit să îndeplinesc toate obiectivele pe care mi le-am setat la început de drum. Consider că, am reușit să integrez multe aspecte predate la laboratoare dar să aduc în implementarea mea și concepte noi.

Bibliografie/Resurse

[Link proiect](#)

Resurse Hardware

[Tutorial joystick](#)

[Tutorial LCD cu interfață I2C](#)

[Tutorial buzzer](#)

Resurse Software

[Personalizare emoticoane LCD](#)

[Identificare adresă LCD](#)

[Biblioteca LiquidCrystal](#)

[Header pitches.h](#)

[tone\(\)](#)

[noTone\(\)](#)

[createChar\(\)](#)

Laboratoare

[Laboratorul 0](#)

[Laboratorul 1](#)

[Laboratorul 2](#)

[Laboratorul 4](#)

[Laboratorul 6](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2021/abasoc/simonsays>

Last update: **2021/06/02 21:49**

