

# Tic-Tac-Toe

## Introducere

Proiectul consta in implementarea clasicului joc X si 0 pe un LCD cu touchscreen. Jocul se poate juca atat intre 2 jucatori(pvp), cat si intre un utilizator si un AI, avand 3 dificultati: Easy, Medium si Impossible.

## Descriere generală

Jucatorii se pot folosi atat de butoane, cat si de touchscreen-ul de pe LCD pentru partea de input. Display-ul LCD-ului(SPI) este componenta principala si esentiala pentru partea de output, afisandu-se in mod fluent ecranul corespunzator momentului curent din cadrul jocului. Este folosit un buzzer pentru partea de sunet a jocului, aceasta fiind utila in anumite cazuri(ex: cand o partida ajunge la final, la o mutare incorecta).



## Hardware Design

### Listă de piese:

- Arduino MEGA 2560
- LCD cu touchscreen(SPI) - ILI9341, XPT2046
- Translator de nivel bidirectional cu 8 canale (folosit pentru conectarea LCD-ului)
- 10 Butoane
- 1 Buzzer
- 1 Breadboard principal
- Cate un breadboard de mici dimensiuni pentru amplasarea butoanelor asociate fiecarui jucator
- Fire de legătură

### Schemă electrică:



## Software Design

Mediul de dezvoltare: Arduino IDE

Biblioteci externe utilizate:

- Ucglib.h : pentru LCD
- XPT2046.h : pentru touchscreen

Pentru implementarea algoritmului folosit in cazul modului de joc cu bot-ul, dificultatea Impossible, m-am folosit de urmatoarea strategie de joc:

- Daca bot-ul poate castiga intr-o mutare, mutarea castigatoare va fi facuta;
- Daca adversarul poate castiga intr-o mutare, bot-ul va bloca, alegand campul respectiv;
- Se va crea un scenariu in care bot-ul sa aibe 2 variante de a castiga, daca acest lucru este posibil;
- Daca jucatorul ar putea, la urmatoarea mutare, un scenariu care-i va oferi doua variante de a castiga, bot-ul va incerca sa il opreasca;
- Daca campul din mijloc este liber, bot-ul va alege acest camp pentru a muta;
- Daca un colt opus unui simbol al adversarului este liber, botul va alege acel colt;
- Se va alege orice colt liber, daca acesta exista;
- Se va alege orice camp liber ramas.

Astfel, folosind aceasta strategie, bot-ul nu va putea fi invins in niciun scenariu.

Pentru dificultatea Medium, bot-ul va incerca mai intai sa castige, daca acest lucru este posibil, iar altfel, daca adversarul poate castiga intr-o mutare, acesta va fi blocat. Daca niciunul dintre aceste 2 scenarii nu apare, se va face o mutare random.

Dificultatea Easy consta doar in mutari random.

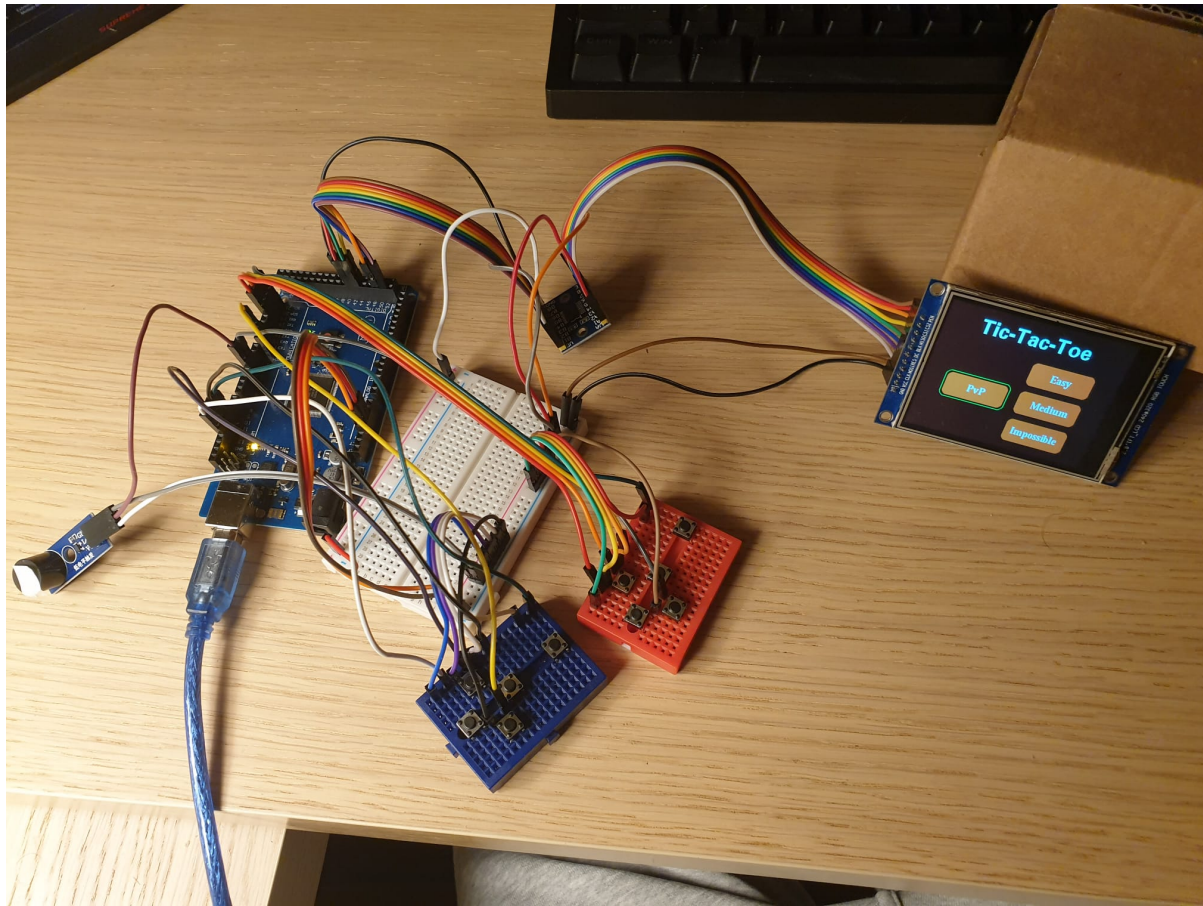
Sunt implementate mai multe ecrane, oferindu-le jucatorilor mai multa libertate de alegere:

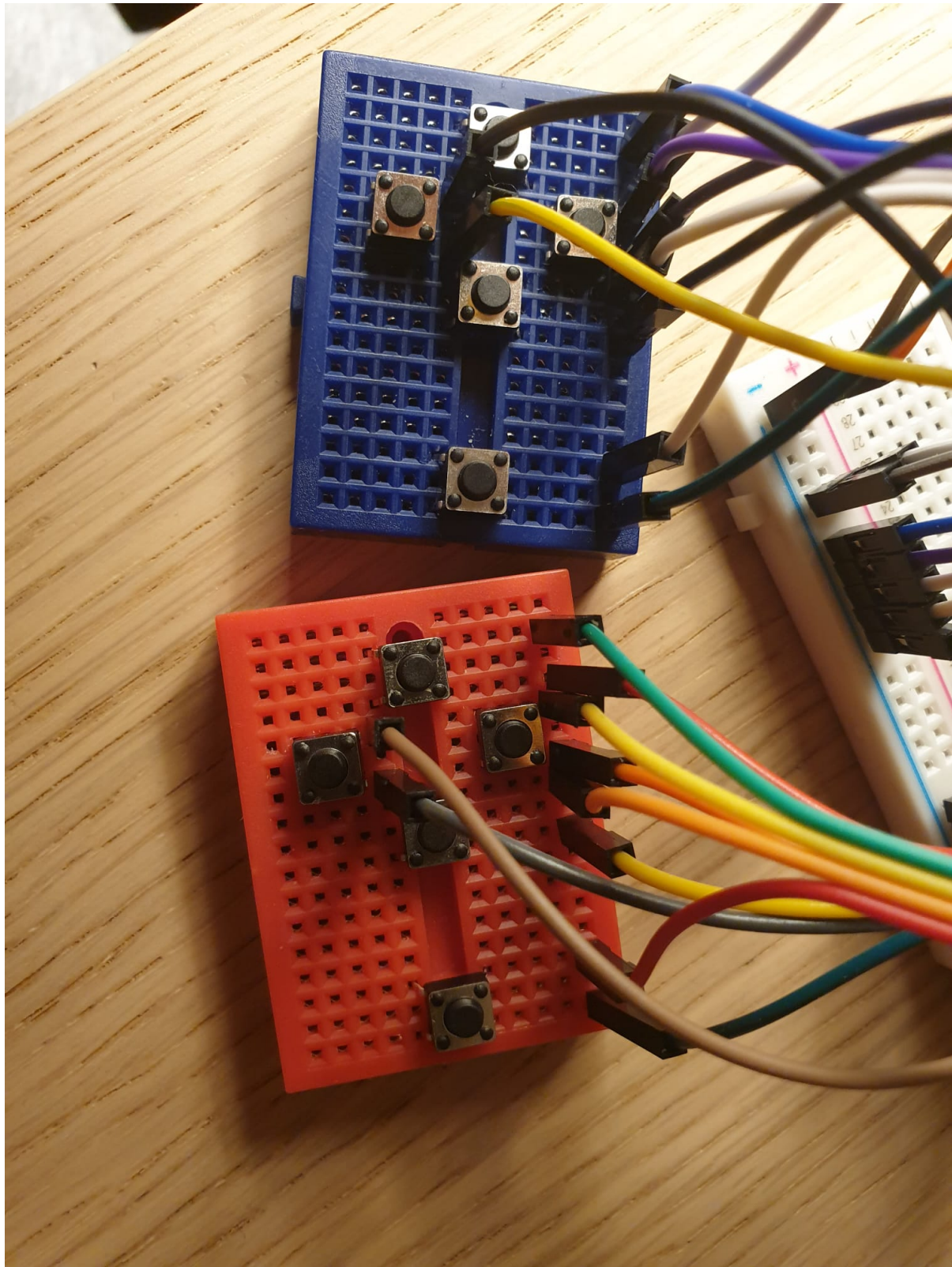
- initial, jucatorii vor alege modul de joc
- apoi, in cazul in care se alege un mod de joc contra 'AI'-ului, un singur jucator va putea juca, astfel ca acesta isi va putea alege culoarea(Rosu/Albastru)
- La finalul unui joc, jucatorii pot alege daca sa continue sa joace in acest mod de joc, pastrandu-se scorul, sau sa se intoarca in meniul de start.

In cazul jucarii mai multor runde, simbolurile asociate bot-ului, respectiv jucatorilor, vor alterna, oferind sanse egale, insa culorile se vor mentine.

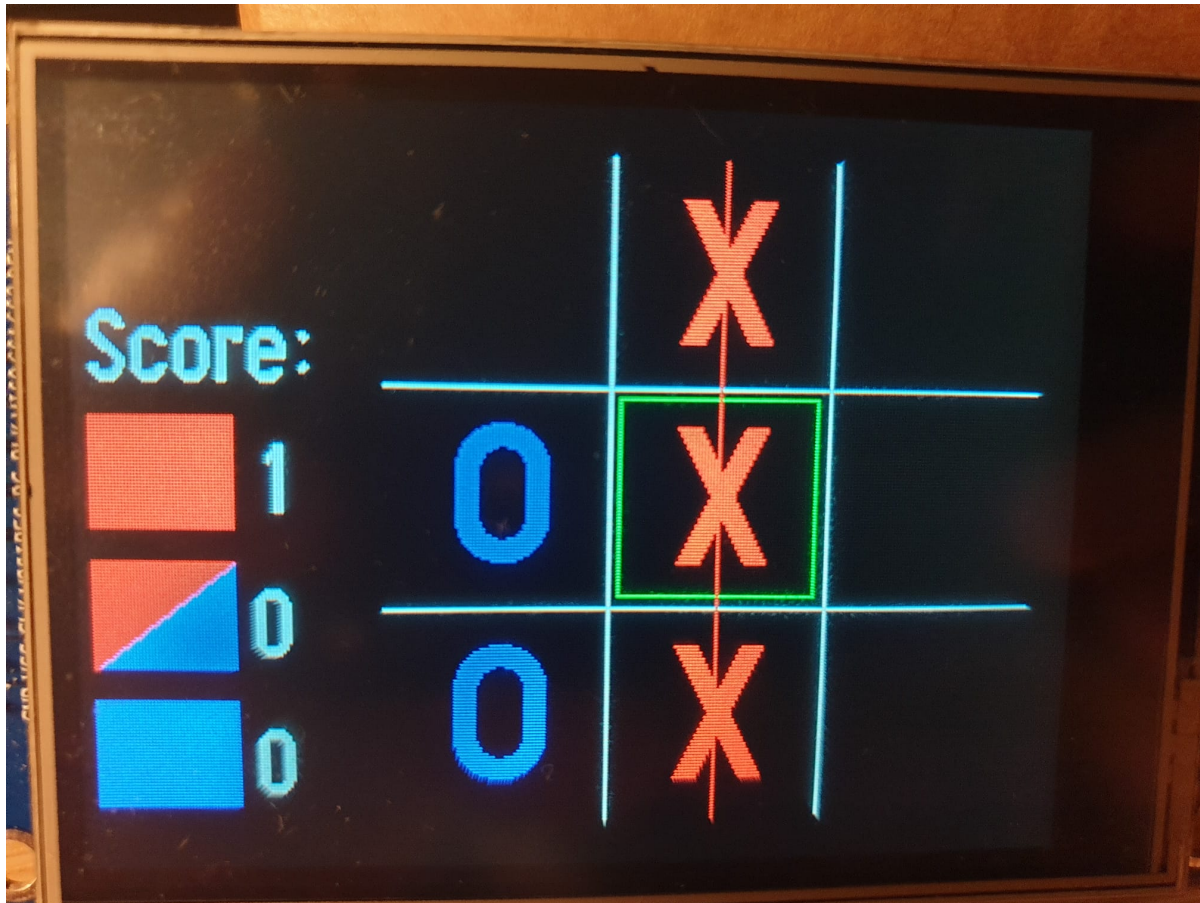
De mentionat este si faptul ca am folosit intreruperi PCINT pentru 5 butoane si intreruperi INT pentru celelalte 5, folosind doar anumite flaguri pentru a implementa si varianta utilizarii butoanelor in loc de utilizarea touchscreen-ului.

## Rezultate Obținute









## Concluzii

Realizarea proiectului a fost utila pentru familiarizarea cu Arduino IDE si realizarea acestui tip de circuite, folosirea eficienta a unor biblioteci externe si intelegerea mai profunda a subiectelor abordate la laborator precum utilizarea intreruperilor, comunicarea folosind SPI si lucrul cu registre.

De asemenea, confruntandu-ma cu probleme aparute pe parcurs precum faptul ca dimensiunea alocata pentru program space pentru un Arduino Uno R3 este insuficienta pentru proiectul ales(fiind necesara inlocuirea acestuia cu un Arduino MEGA) sau cele cauzate de rolul specific al anumitor pini de pe placuta m-au ajutat sa invat cum sa abordez cat mai corect astfel de proiecte.

## Download

[constantinescu\\_ciprian\\_tic-tac-toe.zip](#)

## Bibliografie/Resurse

- [1] [https://www.optimusdigital.ro/ro/optoelectronice-lcd-uri/3544-modul-lcd-spi-de-28-cu-touchscreen-controller-ili9341-i-xpt2046-240x320-px.html?search\\_query=lcd&results=198](https://www.optimusdigital.ro/ro/optoelectronice-lcd-uri/3544-modul-lcd-spi-de-28-cu-touchscreen-controller-ili9341-i-xpt2046-240x320-px.html?search_query=lcd&results=198)
- [2] <https://github.com/olikraus/ucglib>
- [3] <https://github.com/spapadim/XPT2046>
- [4] <https://www.cs.ou.edu/~fagg/classes/general/atmel/ArduinoMega.pdf>
- [5] <https://en.wikipedia.org/wiki/Tic-tac-toe>

[Export to PDF](#)

From:

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

Permanent link:

[http://ocw.cs.pub.ro/courses/pm/prj2022/agmocanu/tic\\_tac\\_toe](http://ocw.cs.pub.ro/courses/pm/prj2022/agmocanu/tic_tac_toe)



Last update: **2022/05/28 01:27**