

# 2048 Sounds

Autor: [Mihai-Daniel Radu 332CC](#)

## Introducere

Scopul proiectului este implementarea jocului 2048 folosind Arduino UNO. Input-ul utilizatorului va fi receptionat prin butoane, iar output-ul, starea curenta a jocului, va fi afisata pe un ecran LCD. Jocul va fi acompaniat de diverse efecte sonore si melodii in stil retro emise de un modul buzzer pasiv, in functie de starea in care se afla jocul.

## Descriere generala

Scopul jocului 2048 este ca, prin translatarea unor patrate numerotate cu multipli de 2 si situate intr-o matrice 4x4, sa se obtina un patrat care sa contina numarul 2048, fara ca toata matricea sa fie in totalitate ocupata si nicio miscare disponibila ramasa.

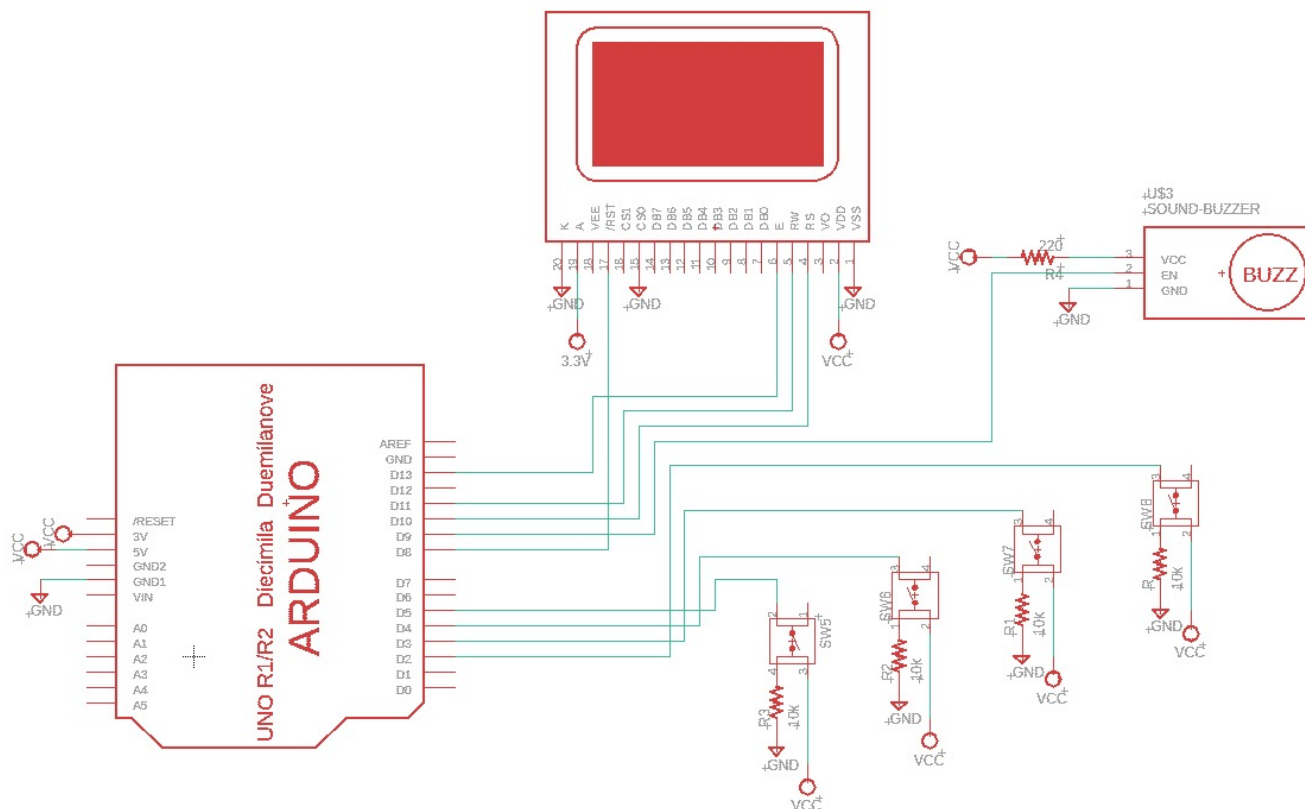
Input-ul jucatorului va fi captat folosind 4 butoane, care vor reprezenta directiile in care se vor misca patratele in matrice: stanga, sus, jos si dreapta.

Output-ul jocului va fi afisat folosind un ecran LCD grafic, iar buzzer-ul va emite sunete la apasarea butoanelor si in fiecare ecran al jocului.

Schema bloc 2048Sounds:



## Hardware Design



Lista componentelor folosite pentru proiect:

Componente	Nr. bucati
Placă de dezvoltare UNO compatibilă Arduino - DIP	1
Cablu USB A-B 1m Arduino Mega, UNO, imprimanta	1
Breadboard 830 puncte MB-102	2
65 x fire jumper	1
Buton MIC-Push Button-trough hole Negru	4
Rezistor 1/4W - Valoare rezistenta : 10K	4
Rezistor 1/4W - Valoare rezistenta : 220	1
Display LCD 128*64 pixeli 5V iluminat Albastru	1
Modul Buzzer - Tip : Pasiv	1
Fire Dupont mama-tata 20cm	6
Fire Dupont tata-tata 20cm	10

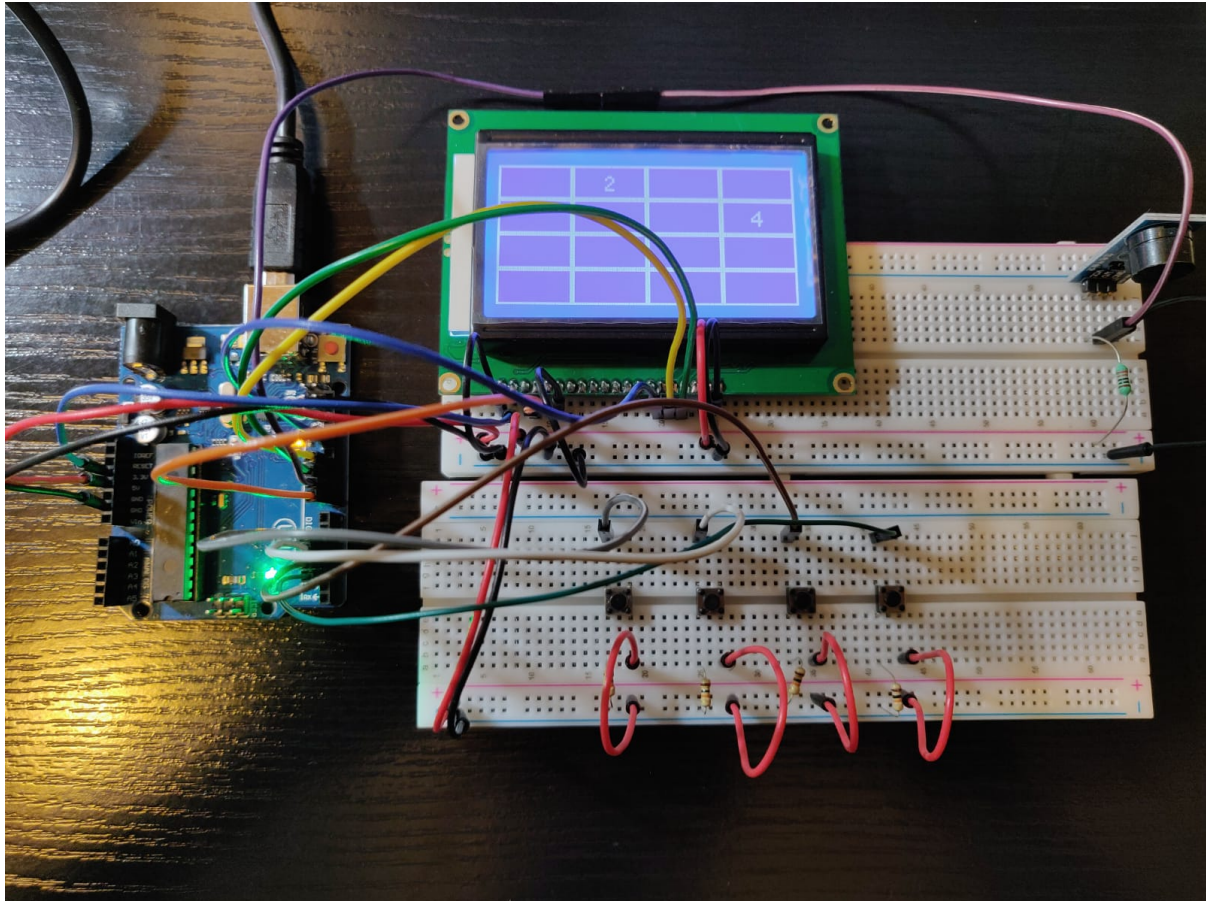
Primul lucru pe care l-am facut a fost sa sudez/lipesc o bareta de 20 pini la display pentru a-l putea conecta la breadboard si respectiv la Arduino. Nu a fost prima oara cand lucrez cu asta, dar a fost foarte rewarding cand am vazut cum se aprindea display-ul, mai ales ca nu prea am avut contact pana acum cu componente hardware la un nivel atat de low.

Dupa aceea, am conectat cele 4 butoane cu rezistente 10k de pull-down pentru a capta input-ul jucatorului.

Initial, am comandat 3 buzzere: 1 buzzer simplu activ, 1 buzzer simplu pasiv si 1 modul de buzzer pasiv. Dupa testarea acestora, am decis sa folosesc modulul de buzzer pasiv, avand cel mai bun sunet si fiind cel mai facil de folosit.

Modulul buzzer pasiv functiona anormal sau chiar prost, fiind conectat direct la 5V. Dupa ce am adaugat o rezistenta de 220 ohmi, modulul a functionat perfect.

Am ales sa folosesc 2 breadboard-uri MB-102 pentru integritatea fiecărei componente hardware si pentru a avea mai mult spatiu la dispozitie pentru a actiona cele 4 butoane. Pe primul breadboard am legat display-ul si modulul buzzer-ului pasiv, iar pe cel de-al doilea breadboard am pus in centru butoanele care vor actiona, in aceasta ordine, miscarile stanga, sus, jos si dreapta (de la stanga la dreapta pe breadboard).



## Software Design

Diagrama de flow:



Pentru a scrie cod cat mai curat si lizibil, am impartit in 3 codul: partea de Arduino si setup-ul necesar, mecanicile si logica de joc 2048 si fisierul cu efectele sonore.

Am inceput design-ul software prin a seta toti pinii folositi si prin a apela constructorul necesar pentru configuratia seriala prin SPI a display-ului 128x64 ST7920. Folosind biblioteca u8g2 si manualul de referinte si resurse din aceasta, am desenat fiecare ecran din cadrul jocului: ecranul de "Hello & Welcome!", ecranul de joc cu matricea si numerele din aceasta, ecranul de "Game Won" si ecranul de "Game Lost".

Implementarea ecranelor de vizualizare contine foarte putine valori hardcodate sau specifice doar

display-ului sau componentelor pe care le-am utilizat eu, deci am respectat o norma de "clean code".

Pentru a capta input-ul jucatorului, verific intotdeauna daca a existat vreo schimbare in starile butoanelor(apasat/neapasat), iar daca a existat, fac debouncing pentru a ma asigura ca este o miscare adevarata din partea jucatorului si nu o fluctuatie de curent sau orice altceva ce ar putea schimba starile butoanelor.

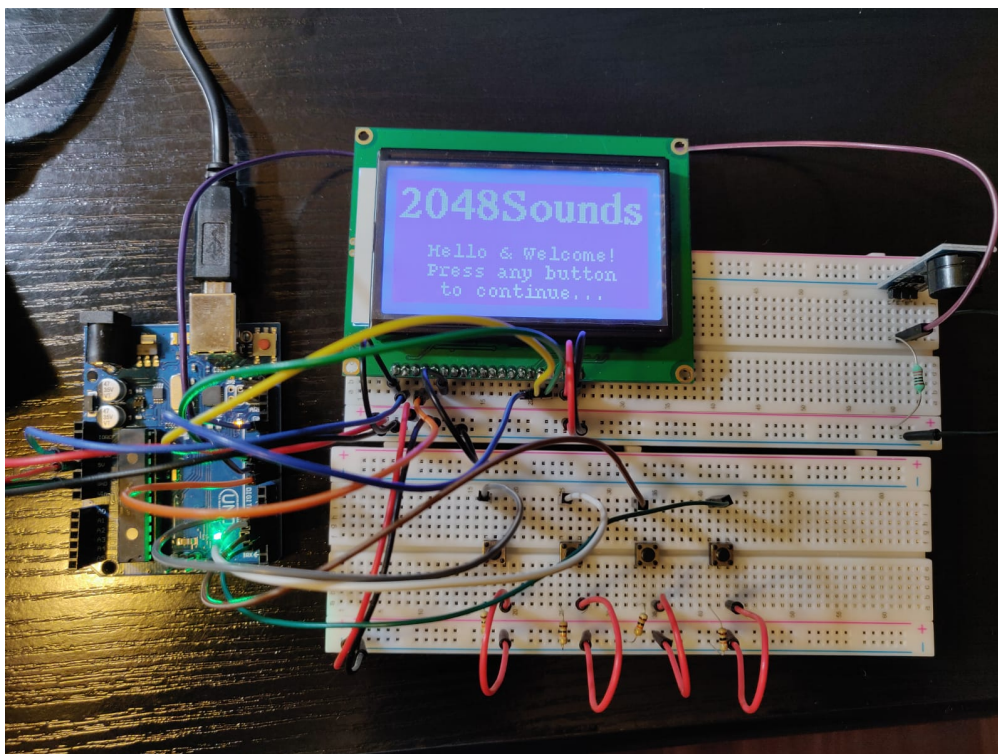
In ceea ce priveste jocul de 2048, in matricea de joc pastrez valorile puteri ale lui 2, deci valorile posibile in cadrul jocului sunt cuprinse intre 0 si 11, iar transformarea este efectuata la vizualizarea ecranelor, nefiind importanta pentru logica jocului decat exponentul bazei 2. Prin intermediul mai multor functii si flag-uri, verific daca jocul este inceput, castigat(exista valoarea 2048 in matrice) sau pierdut(nu mai exista miscari posibile valide in matrice), iar in functie de acestea decid ce ecran afisez sau ce efecte sonore sunt activate. La apasarea unui buton, verific daca exista miscari valide in directia impusa de buton, iar daca exista o efectuez si adaug un nou numar in matrice.

Efectele sonore joaca un rol important in jocul meu si de aceea am construit mai multe melodii si sunete care se potrivesc foarte bine cu jocul. Theme song-ul care canta pe fundal in ecranul de "Hello & Welcome!" este partea de inceput din "The Godfather", modificata de mine pentru a complimenta atmosfera in jocul de 2048. La apasarea butoanelor se aude un sunet inspirat din jocurile retro arcade, iar la ecranele de "Game Won" si "Game Lost" sunt redade sunete din aceeasi sfera retro potrivite pentru castigul sau pierderea jocului.

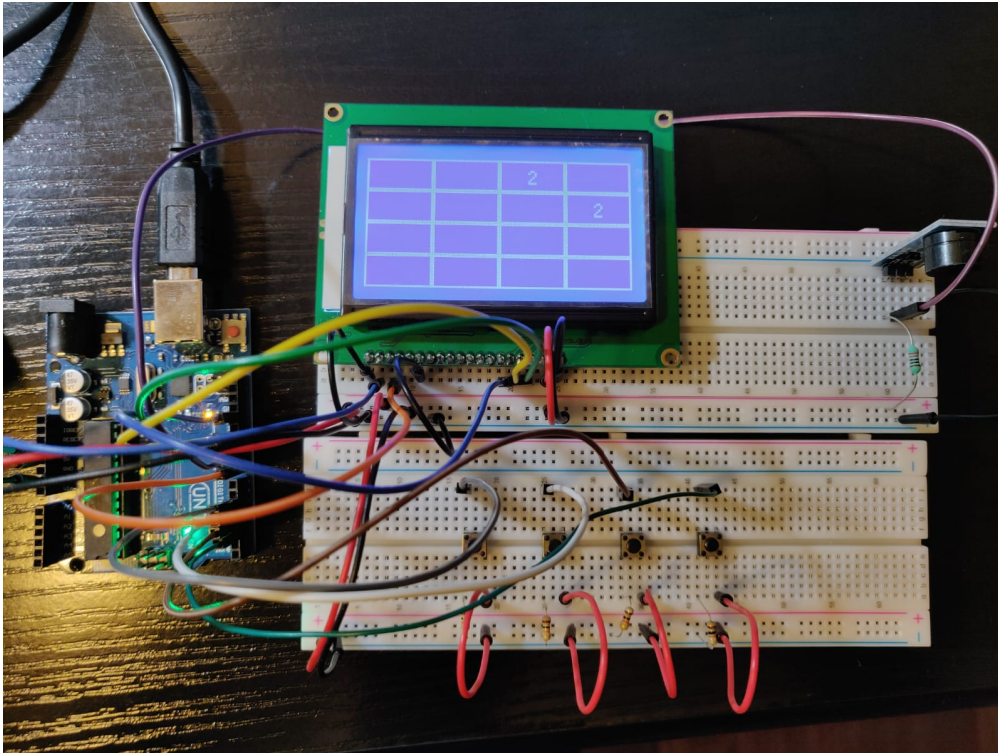
La final, este afisat scorul obtinut in urma jocului, iar la apasarea oricarui buton dupa afisarea ecranelor de castigare/pierdere, jucatorul este trimis inapoi la ecranul de inceput "Hello & Welcome!".

## Rezultate obtinute

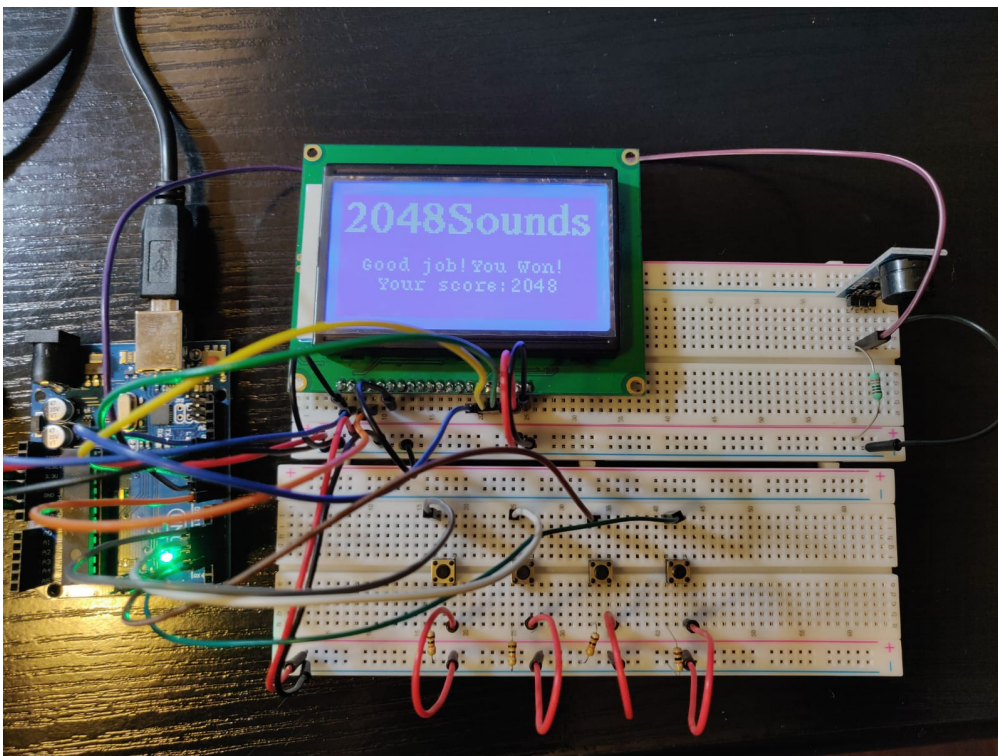
Ecran "Hello & Welcome!":



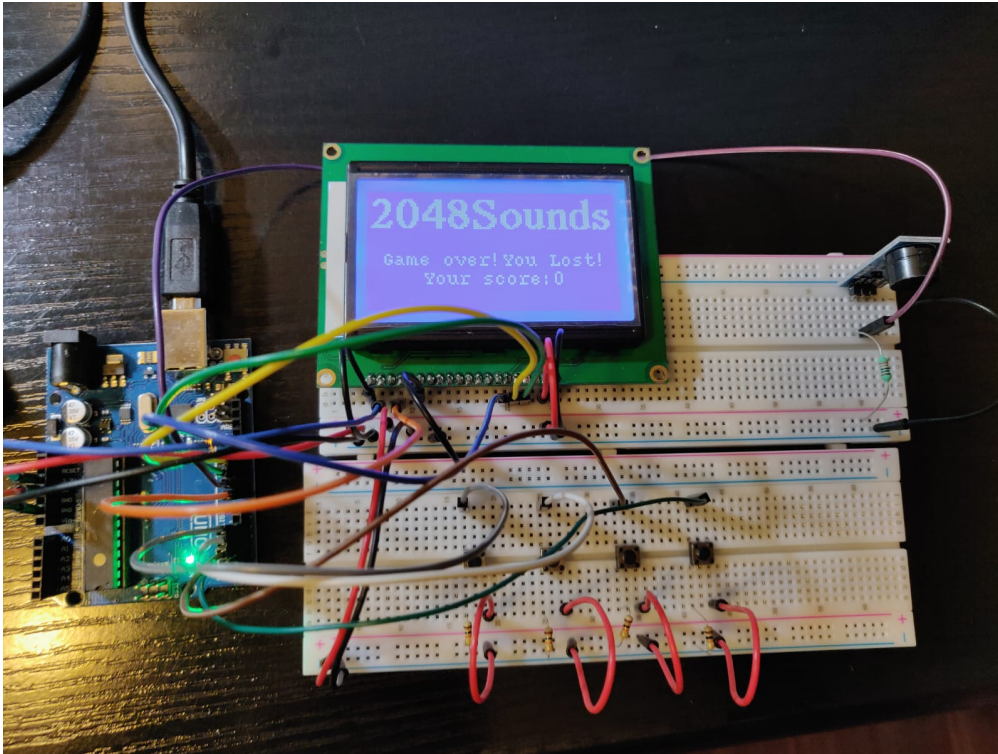
## Ecran matricea de joc:



## Ecran "Game Won!":



## Ecran "Game Lost!":



[Demo 2048Sounds Youtube](#)

## Concluzii

Consider ca a fost unul dintre cele mai interesante proiecte din facultate, nu doar datorita oportunitatii de a cunoaste si a intelege hardware-ul, dar si datorita faptului ca am reusit sa creez ceva de la 0 si sa mearga impecabil!

## Download

[Download Cod 2048Sounds](#)

## Jurnal

1. 25.04.2021 : Creare pagina + prima editare
2. 26.04 - 08.05.2021 : Research in vederea design-ului hardware
3. 08.05.2021 : Plasare comanda Ardushop
4. 10.05.2021 : Obtinere componente
5. 10.05 - 12.05.2021 : Research in vederea design-ului software
6. 14.05.2021 : Sudare display
7. 15.05.2021 : Finalizare hardware
8. 16.05 - 18.05.2021 : Finalizare in proportie de 85% software

9. 20.05 - 31.05.2021 : Refactoring cod, hardware updates, bug fixes
10. 01.06.2021 : Finalizare 100%

## Bibliografie si Resurse

Componente:

- [Display grafic 128x64 ST7920](#)
- [Breadboard MB-102](#)
- [Placa dezvoltare Arduino UNO DIP](#)
- [Modul buzzer pasiv](#)

Resurse:

- [Biblioteca u8g2 folosita pt ST7920](#)
- [Tutorial de folosire display](#)
- [Exemplu util debounce](#)
- [Tutorial util soldering](#)
- [Partitura theme song The Godfather](#)
- [Fisier note muzicale buzzer](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2021/avaduva/2048sounds>

Last update: **2021/06/02 00:10**

