2025/05/25 15:19 1/11 GameBox

# **GameBox**

#### **Introducere**

GameBox este inspirat din jocurile copilăriei ([\*]imagine de referință), cele pe care le puteai lua oriunde după tine.

Aceasta este o consolă portabilă ce permite încărcarea a mai multor jocuri clasice din anii 2000, din care poți alege ce vrei să joci în orice moment.

Proiectul îmbină simplitatea și farmecul jocurilor retro cu portabilitatea și accesibilitatea unui dispozitiv modern.

Are ca scop entertainmentul și este util pentru a satisface nevoia sentimentului de nostalgie a celor pasionați de jocurile vechi, oferind totodată o alternativă relaxantă și fără complicații la gamingul actual.

Intr-o versiune avansata, GameBox ar putea fi extins pentru a suporta mai multe jocuri, în acest moment fiind în stagiul de demo va avea 1-3 jocuri.



# Descriere generală

Proiectul este structurat în jurul unei plăci de dezvoltare Arduino Mega 2560, care gestionează toate modulele hardware și logica software necesară pentru funcționarea unei console de jocuri. Sistemul este compus din următoarele module:

- Joystick pe 2 axe (XY) permite controlul direcțional în jocuri, prin mișcări pe axa X (stânga/dreapta) și Y (sus/jos). Este folosit pentru navigare și controlul personajului.
- Shield LCD pentru Arduino Mega de 3.5" afișează interfața grafică și jocurile. Funcția de touchscreen permite selecția jocului din meniu și interacțiunea în timpul jocului.
- 4 butoane push actionează ca butoane de acțiune pentru jocuri (ex: tragere, abilități speciale,

rotire, confirmare etc.), cu funcționalități personalizate în funcție de jocul activ.

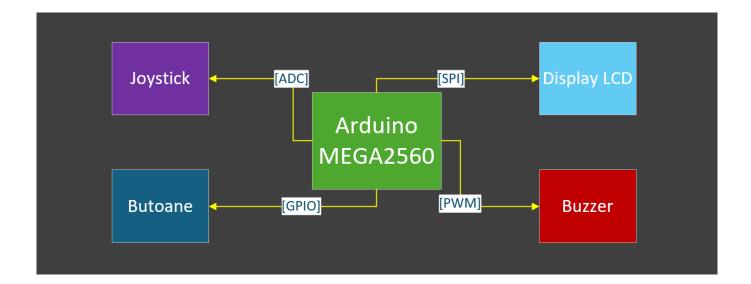
• Buzzer pasiv - generează efecte sonore și feedback audio în timpul jocului.

Toate modulele comunică și interacționează prin intermediul Arduino Mega 2560, care rulează un program ce:

- Interpretează datele de la joystick și butoane.
- Oferă interfață grafică pe display-ul TFT și permite selecția jocurilor din memorie.
- Gestionează logica de joc, afișajul și sunetul.

—-

# **Diagrama Bloc**



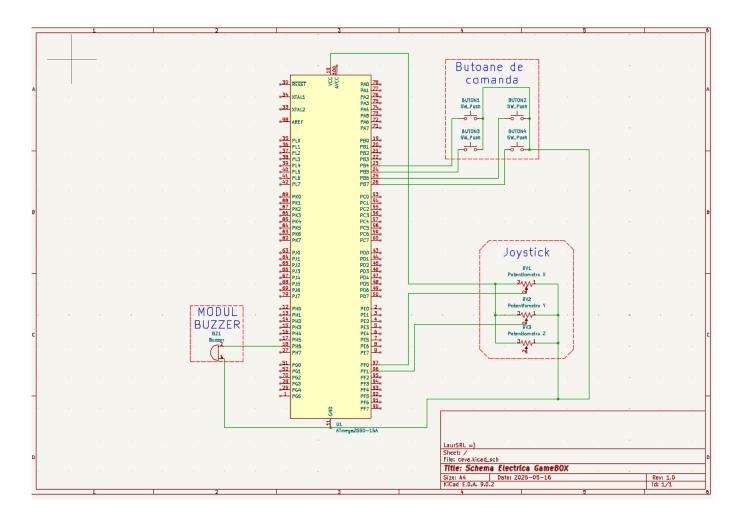
# **Hardware Design**

## **Componente Hardware:**

- placă de dezvoltare Arduino Mega 2560 cu microprocesor ATmega2560 [1]
- modul joystick pe 2 axe XY cu 5 pini [2]
- Shield LCD pentru Arduino Mega de 3.5" [3]
- 4 butoane push
- modul buzzer pasiv
- fire de legătură și eventual un breadboard

2025/05/25 15:19 3/11 GameBox

# Schema Electrica (fara display):



## Descriere funcționalitate hardware:

- 1. Microcontroller Arduino Mega 2560 (ATmega2560)
- MCU cu 256 KB memorie flash, 8 KB RAM, 16 MHz cristal.
- Multi pini digitali și analogici, interfețe multiple: UART, SPI, I2C, Timere hardware, PWM, ADC etc.
- Este potrivit pentru proiecte complexe cu multe periferice.
- 2. Modul joystick pe 2 axe XY
- Interfață: Analog input ADC pentru poziție continuă, permite citirea valorii tensiunii variabile proporțional cu poziția joystick-ului.
- Functionalitate: Determinarea poziției joystick-ului pe cele două axe pentru control în interfață/joc.
- Consum estimat: câțiva mA (~5 mA).
- 3. Shield LCD 3.5" pentru Arduino Mega
- Interfață: General, shield-ul se conectează direct la header-ele Mega (digital + analog).
- Folosește pinii digitali și SPI pentru comunicare cu ecranul tactil și afișajul.
- Permite afișarea grafică de înaltă rezoluție.

• Consum: Poate ajunge până la 100-150 mA când este iluminat și afișează imagini.

#### 4. 4 butoane push

- Configurate ca input cu rezistențe pull-up interne activate.
- Folosite pentru input simplu on/off, pentru controale în joc (ex. selecție, acțiuni).
- Consum neglijabil.

#### 5. Modul buzzer pasiv

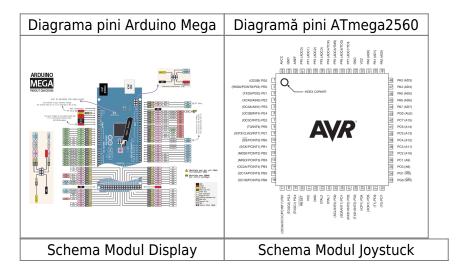
- Controlat prin generarea unui semnal pătrat la frecvențe audibile.
- Implementare prin Timer1 și ISR pentru toggle manual pe pin (nu folosim hardware PWM clasic, dar generăm semnal PWM prin ISR).
- Consum redus (~20-30 mA).
- Permite redarea tonurilor/efectelor sonore în joc.

## Managementul consumului de energie

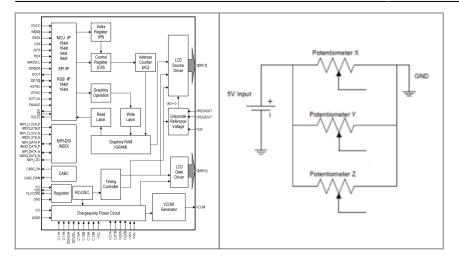
Tensiune alimentare: 5V stabilizată de la sursa Arduino Mega (USB sau adaptor).

Consum total estimat: Mega 2560: ~50 mA în funcțiune normală LCD 3.5": ~100-150 mA Joystick + butoane + buzzer: ~20-30 mA Total maxim estimat: ~200-250 mA.

# Scheme și Diagrame:



2025/05/25 15:19 5/11 GameBox



### **Bill Of Materials:**

Componenta	Pret	Link	Datasheet
3.5inch Arduino Display-Mega2560	69,99 lei	https://www.optimusdigital.ro/ro/shield-uri-pentru-arduino/872-shield-lcd-pentru-arduino-mega-de-35html	http://www.lcdwiki.com/res/MAR3513/Datasheet-ILI9486.pdf
Modul Joystick	4,88 Lei	https://www.emag.ro/modul-joystick-conexiune-digitala-3-3v-5v-compatibil-arduino-ai0262-s362/pd/D6449GBBM/	https://naylampmechatronics.com/img/cms/Datasheets/000036%20-%20datasheet%20KY-023-Joy-IT.pdf
Placă de dezvoltare Arduino Mega 2560	87,99 lei	https://www.optimusdigital.ro/ro/compatibile-cu-arduino-mega/1687-clona-arduino-mega-2560-3-atmega/2560-atmega/260-atmega	https://makerhero.com/img/files/download/MEGA-Datasheet.pdf?srsitid=AfmBOorDsPIMo5rFe6UU6zdO5jQiH-NPPSS3SBkoj8nBwnV9nRWTNV_B
Modul Buzzer Pasiv	1,69 lei	https://www.optimusdigital.ro/ro/componente-electronice/12598-modul-buzzer-pasiv.html	https://www.handsontec.com/dataspecs/module/passive%20buzzer.pdf
Butoane	0,6 lei/bucata	https://www.robofun.ro/kit-25-comutatoare-tactile-apasare.html	N-am gasit
Fire rigide	12,49 lei	https://www.optimusdigital.ro/ro/fire-fire-nemufate/899-set-de-fire-pentru-breadboard-rigide.html	
Breadboard	6,90 lei	https://www.emag.ro/placa-test-breadboard-400-bb400/pd/D5WBP7MBM/	
Carton	0 lei	Au venit piesele in el 🕹)	Nu cred ca are

### Pini folositi:

1. Modul joystick pe 2 axe (XY) – 4 pini utilizați (fără SW) Pini folosiți:

VCC - conectat la +5V de pe Arduino

GND - conectat la GND

VRx - conectat la A0 (citire axa X, analog)

VRy - conectat la A1 (citire axa Y, analog)

- Axa X şi Y oferă tensiuni analogice între 0 şi 5V → necesită pini analogici (A0, A1).
- 2. Shield LCD 3.5" pentru Arduino Mega Pini folosiți: Aceste shield-uri folosesc în mod tipic pinii digitali dedicați magistralei paralele (8-bit) și alți pini pentru control. La Mega, aceștia sunt predefiniți:

LCD Data pins (D0-D7) → Mega Digital 22-29

Control pins:

RS → D38

WR → D39

 $CS \rightarrow D40$ 

RST → D41

15:19

 $RD \rightarrow D43$ 

Backlight → D44

• Este un shield special pentru Mega: se potrivește fizic și logic pe Mega.

3. 4 butoane push Pini folosiți : Button 1 → D10

Button 2 → D11

Button 3 → D12

Button 4 → D13

Button 5 → D8

(Toți ca input cu pull-up activ)

- Pinii digitali 10-13 sunt liberi și accesibili.
- Butoanele funcționează cu citire simplă digitală, deci nu au nevoie de ADC.
- Se configurează cu INPUT\_PULLUP pentru a evita rezistențe externe.
- 4. Modul buzzer pasiv Pin folosit:

Signal → PH6 (Digital Pin 9)

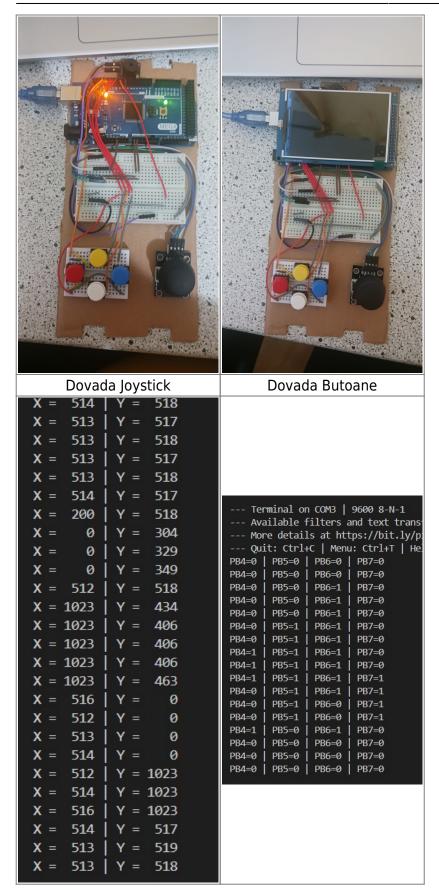
GND → la GND

• Este implementat toggling manual prin Timer1 și ISR, ceea ce permite generarea de sunet pe orice pin digital, inclusiv PH6.

#### **POZE**

-1		
1	Compunente(fara LCD)	Compunente(cu LCD)
-	L Compunenteriala ECD1	Compunentettu ECD1

2025/05/25 15:19 7/11 GameBox



Joystick: am folosit un cod de test care citeste si afiseaza datele primite de la joystick. Acesta trimite valor intre 0 si 1024 pozitia neutra fiind 516, 0 fiind pozitie stanga pe axa X si jos pe axa Y respectiv 1024 fiind dreapta pe axa X si sus pe axa Y

Butoane: de asemenea am folosit un cod care citeste si afiseaza starile celor 4 butoane. 0 - butonul este neapasat iar 1 - butonul este apasa

UPDATE: am mai adaugat un buton pentru functia de select.

#### Nota de final:

N-am reusit sa realizez o diagrama si o simulare in Wokwi deoarece nu am gasit componentele si nici librarii care sa le contina acolo.

## **Software Design**

# Descrierea codului aplicației

Codul este incarcat si executat pe un microcontroler ATmega2560 de pe placa Arduino MEGA2560. Aplicația rulează jocul clasic tetris și citește constant valorile butoanelor, respectiv joystick-ului pentru controlul pieselor din joc.

### Mediu de dezvoltare utilizat

Dezvoltarea a fost realizată folosind:

- Arduino IDE pentru scrierea, compilarea și încărcarea codului pe placă
- PlatformIO pentru realizarea de teste ale pieselor individuale
- KiCad 7 pentru proiectarea schemei electrice
- Visio pentru diagramei bloc

# Librării utilizate

Au fost utilizate următoarele librării:

- LCDWIKI GUI.h pentru afișarea pe LCD 3.5" mega shield.
- LCDWIKI KBV..h pentru comunicatia cu SD Cardul. (Card prea mare. Maximul acceptat este de 2GB)

Ambele librăriile sunt speciale pentru Display-urile ce folosesc driver ILI9486 și nu se regasesc în librăriile din Arduino IDE așa ca le-am încărcat extern .

2025/05/25 15:19 9/11 GameBox

# Justificarea utilizării funcționalităților din laborator

Conceptele învățate în laborator care au fost aplicate:

- GPIO pentru semnalele de la butoanele simple.
- PWM pentru utilizarea buzzer-ului pasiv pe diferite frecvențe.
- ADC pentru citirea valorii analogice de la Joystick.
- SPI pentru comunicația cu modulul LCD

#### Structura

Structura proiectului este organizată în jurul unei bucle infinite loop() ce conține mai multe stări (meniu, joc, gameover):

- 1. Initial se porneste un meniu.
- 2. Selectand play începe jocul propriu-zis ce este controlat prin butoane si/sau joystick. Aceasta stare are o melodie transmisă prin buzzer.
- 3. Pierderea jocului trimite aplicația în starea de gameover une de asemenea exista o alta melodie și așteaptă apăsarea butonului SELECT pentru a fi retrimis în starea de meniu.
- 4. Aplicație are și un ScoreBoard care se updatează după fiecare joc dacă acesta are mai mult de 0 puncte.

### **Demo video**

Link către demo YouTube Incepere Joc

Link către demo YouTube Joc Pierdut

## **Rezultate Obtinute**

În urma realizării aplicației s-au atins următoarele obiective:

- Utilizarea cu succes a urmatoarelor piese: Butoane, Buzzer, Joystick, Ecran LCD
- Realizarea unui joc complet functional pe consola fizică inclusiv cu un meniu, scoreboard, ecran de gameover și piese aferente transmise prin buzzer pentru joc si pierderea acestuia.

Obiective neatinse și concluzia acestora:

 Utilizarea unui SD Card de pe care să extrag resursele jocurilor și informațiile suplimentare (precum scorul chiar și după resetarea sistemului): Cititorul SD de pe modulul Display LCD este limitat la carduri de maxim 2GB, ceea ce a dus la timp pierdut depanării problemelor cu SD Cardul fara reusită, iar achizitionarea unui card SD dura prea mult pentru a mai implementa acest aspect.

• O varietate de jocuri: În lipsa unei memorii mai mari precum SD-ul nu puteam să dezvolt mai multe jocuri, în plus timpul pentru a programa si alte jocuri a fost irosit pe problema cu SD-ul. Concluzie: Citeste mai bine documentatia pieselor!!!

### **Download**

Fisierele din Github

pm-proiect-main.zip

### **Link Github**

https://github.com/Laurentiu2014/PM-Proiect/tree/main

# Bibliografie/Resurse

### **Resurse Hardware:**

• [1] - ATmega2560:

https://ww1.microchip.com/downloads/aemDocuments/documents/OTH/ProductDocuments/DataSheets/ATmega640-1280-1281-2560-2561-Datasheet-DS40002211A.pdf

• [2] - Modul Joystick:

https://www.laskakit.cz/user/related\_files/joystick\_module.pdf

• [3] - Modul Display:

http://lcdwiki.com/3.5inch\_Arduino\_Display-Mega2560

## **Resurse Software:**

• [1]

**Export to PDF** 

2025/05/25 15:19 11/11 GameBox

From:

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

Permanent link:

http://ocw.cs.pub.ro/courses/pm/prj2025/avaduva/laurentiu.panaite03

Last update: 2025/05/25 15:19

