

Arduino Graphix

Nume: Popa Andrei

Grupă: 333CA

Introducere

Arduino Graphix este un demo de pipeline grafic software implementat pe microcontrolerul ATmega328P. Se afișează pe un ecran LCD un mediu 3D, iar utilizatorul se poate folosi de un keypad pentru a modifica perspectiva de vizualizare (pentru a mișca camera).

Sistemul este format din 2 plăcuțe cu microcontroler ATmega328P:

- plăcuța 1 joacă rolul unui GPU: va fi responsabilă să deseneze scena pe un ecran LCD într-un timp cât mai scurt.
- plăcuța 2 joacă rolul unui CPU: va ține cont despre obiectele din scenă și despre cameră și va transmite informațiile relevante către "GPU" și va procesa inputul de la keypad (utilizator).

Motivul pentru care am ales acest proiect este interesul meu față de modul de funcționare a procesoarelor video și modul prin care acestea interacționează cu restul sistemului de calcul pentru a prezenta/desena informații pe un display.

Scopul meu principal este de a învăța prin experiență cum se implementează un pipeline grafic (chiar dacă este un model software foarte limitat și simplificat) și totodată, ca scop secundar, să scriu de mână un "driver" simplu pentru ecranul LCD.

Ca utilitate pentru mine (și alții), vreau ca proiectul să servească ca un exemplu de compromisuri ce trebuie făcute pentru a obține un pipeline grafic 3D funcțional, dar și utilizabil (cu framerate decent) pe un sistem hardware foarte limitat.

Schemă bloc



Hardware Design

Bill of Materials

Componentă	Cantitate	Link	Preț unitar (RON)
Placă dezvoltare Arduino Uno AtMega 328p	2	Link	30.93
Display LCD SPI	1	Link	40.62
Breadboard	1	Link	6.62
Tastatură membrană 3×4	1	Link	4.30
Fire Dupont Tată-Tată 10cm	TBD	Link	0.15
Rezistențe 1k	TBD	Link	0.19
Preț total			113.4 RON

Schema electrică



Pinout

Display LCD:

Pin Display	Pin Arduino	Pin ATmega328p
BL	5V	
RST	5V	
DC	D10	PB2
CS	GND	
CLK	D13	PB5 / SCK
DIN	D11	PB3 / MOSI
GND	GND	
VCC	5V	

Deoarece există doar un singur periferic conectat prin SPI, am conectat pinul de CS (chip select) la GND, astfel acesta este selectat continuu. Pinul de RST (reset) este conectat la 5V (acesta funcționează în logică negată), deoarece nu am nevoie să resetez software display-ul.

Keypad 4×4:

Pin Keypad	Pin Arduino	Pin ATmega328p
1 (Column)	D6	PD6 / OC0A
2 (Column)	D5	PD5 / OC0B
3 (Column)	D9	PB1 / OC1A
4 (Column)	D10	PB2 / OC1B
5 (Row)	D2	PD2 / PCINT18
6 (Row)	D3	PD3 / PCINT19
7 (Row)	D4	PD4 / PCINT20
8 (ROW)	D7	PD7 / PCINT23

Ținând doar câte o coloană a keypad-ului pe high la un moment dat, în funcție de care rând este pe low, obținem ce buton a fost apăsat. Am ales pinii pentru coloană pentru a putea schimba eventual coloana activă prin timere. Am ales pinii pentru rând pentru a primi întreruperi în momentul apăsării unui buton.

ATmega328P:

Pin Arduino (1)	Pin ATmega(1)	Pin Arduino(2)	Pin ATmega (2)
D0	PD0 / RXD	D1	PD1 / TXD
D1	PD1 / TXD	D0	PD0 / RXD

Pinii aleși sunt pinii pentru USART și sunt legați RX ↔ TX.

Software Design

Rezultate Obținute

Concluzii

Download

<https://github.com/andarkrc/Basic-Arduino-Game>

Jurnal

- 7.04.2026 - Alegere tema proiect si confirmare.
- 7.05.2026 - Creare pagină ocw pentru proiect.

Bibliografie/Resurse

Resurse Hardware

Resurse Software

- Laboratoarele
- [Datasheet display](#)
- [Datasheet ATmega328P](#)

[Export to PDF](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2026/victor.stoica0203/andrei.popa0810>



Last update: **2026/05/12 07:05**