

# ConnectFour

## Introducere

### Ce face proiectul?

Proiectul este o transpunere digitală a clasicului joc de strategie „Connect Four”. Utilizatorul interacționează cu jocul printr-un joystick analogic pentru a naviga și a lansa piese colorate pe o matrice de LED-uri RGB. Sistemul gestionează în timp real „căderea” pieselor, rândul jucătorilor și verifică automat condițiile de victorie, oferind în același timp feedback sonor prin intermediul unui buzzer.

### Care este scopul proiectului?

Scopul principal este crearea unui sistem de divertisment interactiv care să demonstreze integrarea între hardware și software. Din punct de vedere academic, proiectul urmărește aplicarea conceptelor fundamentale de microprocesoare (comunicația cu periferice, gestionarea evenimentelor în timp real și optimizarea algoritmilor de logică) într-un produs finit și funcțional.

### Care este ideea de la care am pornit?

Ideea a pornit de la dorința de a reinventa un joc „fizic” popular folosind tehnologia modernă a LED-urilor. Scopul a fost să transform o tablă de joc statică într-o experiență vizuală dinamică, unde culorile vii ale matricei să înlocuiască jetoanele de plastic, oferind o interfață mai atractivă.

### De ce este util?

- **Pentru alții:** Reprezintă o sursă de entertainment rapidă și un exemplu despre cum jocurile de societate pot fi digitalizate folosind componente electronice accesibile.
- **Pentru mine:** Realizarea acestui proiect îmi permite să fac trecerea de la simple exerciții de laborator la construirea unui produs complet. De asemenea, m-a ajutat să înțeleg cum să gestionez limitările de hardware (precum consumul de energie sau viteza de procesare) și cum să structurez un cod complex care să rămână ușor de utilizat.

## Descriere generală

Proiectul constă în realizarea jocului de strategie Connect Four pe o platformă Arduino Uno, utilizând o matrice LED RGB 8x8. Scopul este de a crea un sistem interactiv unde doi jucători se pot întrece folosind culori distincte pentru a-și marca piesele.

Sistemul este structurat pe trei niveluri de interacțiune:

- **Input:** Un joystick analogic este utilizat pentru navigarea pe axa orizontală (prin ADC) și pentru confirmarea mutării prin apăsarea butonului (gestionată prin întreruperi).

- **Procesare:** Microcontrolerul ATmega328P rulează logica de joc, gestionează animația de “cădere” a pieselor și execută algoritmul de verificare a victoriei pe patru axe (orizontal, vertical și două diagonale).
- **Output:** O matrice NeoPixel afișează starea tablei de joc, iar un buzzer pasiv oferă feedback sonor în momentele cheie (mutări sau victorie).

## Schema bloc



## Interacțiune și funcționare

Utilizatorul mișcă joystick-ul pentru a-și poziționa piesa în tabla de joc, vizibila pe primul rând al matricei. La apăsarea butonului, piesa „cade” pe coloana selectată până la prima poziție liberă. Sistemul verifică automat dacă s-a format o linie de patru piese și anunță câștigătorul prin animații vizuale și log-uri transmise prin UART către PC. Pentru siguranță, luminozitatea LED-urilor este limitată software pentru a permite alimentarea întregului ansamblu direct din portul USB.

## Hardware Design

Listă componente folosite:

Nr. crt.	Denumire Componentă	Cantitate	Rol în cadrul proiectului
1	<b>Placuta Arduino Uno</b>	1	Unitatea centrală de procesare care rulează codul în C și gestionează perifericele
2	<b>Matrice LED RGB 8x8 WS2812B</b>	1	Periferic de ieșire utilizat pentru afișarea tablei de joc, a pieselor jucătorilor și a animațiilor
3	<b>Modul Joystick Analogic</b>	1	Periferic de intrare pentru selectarea coloanelor (axa X) și lansarea pieselor (butonul SW)
4	<b>Buzzer Pasiv</b>	1	Periferic de ieșire utilizat pentru a oferi feedback sonor la mutări și în momentul victoriei
5	<b>Condensator 1000 <math>\mu</math>F</b>	1	Protejează matricea LED prin filtrarea vârfurilor de curent și stabilizarea alimentării
6	<b>Rezistență 470 <math>\Omega</math></b>	1	Protejează pinul de date al Arduino și primul LED al matricei împotriva reflexiilor de semnal
7	<b>LED-uri simple</b>	2	Periferice externe utilizate ca indicatoare vizuale pentru a semnaliza care jucător este activ
8	<b>Rezistențe 220 <math>\Omega</math></b>	2	Limitează curentul electric care trece prin LED-urile indicatoare pentru a preveni arderea lor
9	<b>Breadboard</b>	1	Suport hardware pentru realizarea conexiunilor electrice fără necesitatea lipirii componentelor
10	<b>Fire jumper</b>	?	Asigură interconectarea modulelor (matrice, joystick) și a componentelor discrete cu Arduino

Aici puneți tot ce ține de hardware design:

- listă de piese

- scheme electrice (se pot lua și de pe Internet și din datasheet-uri, e.g. <http://www.captain.at/electronic-atmega16-mmc-schematic.png>)
- diagrame de semnal
- rezultatele simulării

## Software Design

Descrierea codului aplicației (firmware):


- mediu de dezvoltare (if any) (e.g. AVR Studio, CodeVisionAVR)
- librării și surse 3rd-party (e.g. Procyon AVRlib)
- algoritmi și structuri pe care plănuți să le implementați
- (etapa 3) surse și funcții implementate

## Rezultate Obținute

Care au fost rezultatele obținute în urma realizării proiectului vostru.

## Concluzii

## Download

O arhivă (sau mai multe dacă este cazul) cu fișierele obținute în urma realizării proiectului: surse, scheme, etc. Un fișier README, un ChangeLog, un script de compilare și copiere automată pe uC crează întotdeauna o impresie bună .

Fișierele se încarcă pe wiki folosind facilitatea **Add Images or other files**. Namespace-ul în care se încarcă fișierele este de tipul **:pm:prj20??:c?** sau **:pm:prj20??:c?:nume\_student** (dacă este cazul).  
**Exemplu:** Dumitru Alin, 331CC → **:pm:prj2009:cc:dumitru\_alin**.

## Jurnal

Puteți avea și o secțiune de jurnal în care să poată urmări asistentul de proiect progresul proiectului.

## Bibliografie/Resurse

Listă cu documente, datasheet-uri, resurse Internet folosite, eventual grupate pe **Resurse Software** și **Resurse Hardware**.

[Export to PDF](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2026/atoader/teodora.sintea>



Last update: **2026/05/09 21:25**