

# Flappy Bird

Autor: [Chirnogeanu Maria-Andreea - 333CB](#)

## Introducere

Proiectul meu dorește să implementeze clasicul joc Flappy Bird folosind o plăcuță Arduino.

Inițial, jocul Flappy Bird a fost creat pentru platforme mobile. Scopul său este să faci o pasăre să treacă prin spațiul dintre două tuburi, prin activarea touchscreen-ului, care face pasărea să zboare în sus. Când jucătorul pierde, se afișează pe ecran punctajul obținut. Este un joc de mare dificultate, care cere răbdare și atenție.

Proiectul va folosi un LCD display care va afișa jocul propriu-zis. Pasărea este ghidată de un buton care, prin acțiunea sa, va face pasărea să zboare din ce în ce mai sus. În momentul în care se pierde jocul, jucătorul va primi o notificare pe telefon cu scorul obținut folosind un modul Bluetooth.

Ideea jocului nu este nouă, însă eu am dorit să îl implementez deoarece mi se pare un joc distractiv. Eu în general sunt pasionată de crearea de jocuri, astfel că proiectul va îmbina două aspecte care îmi plac: programarea de jocuri și crearea de circuite.

## Descriere generală

### Schema Bloc a componentelor



La momentul conectării la sursa de alimentare, va apărea pe LCD interfața inițială a jocului, reprezentată de un background și un buton pe care scrie "Start Game". Jucătorul va apăsa pe butonul fizic pentru a începe jocul, care va determina acțiunea pe ecran a butonului "Start Game".

Va dispărea butonul de pe ecran și va apărea pasărea. Urmează un mini tutorial care indică jucătorului că acțiunea de a ridica pasărea se face din butonul fizic. La apăsarea butonului, jucătorul va putea face pasărea să zboare mai sus, puțin câte puțin. După tutorial, încep să apară obstacolele, adică o secvență de perechi de tuburi care au un spațiu între ele. Scopul jocului este ca pasărea să treacă prin spațiul dintre tuburi, ca în jocul original. Scorul obținut se va afișa pe ecran cât timp jocul continuă.

Jucătorul pierde atunci când pasărea se va ciocni de unul dintre cele două tuburi. În acest moment, jucătorul va primi o notificare pe telefonul său mobil cu scorul obținut, iar pe ecran va apărea text "Game Over" și un buton pe care scrie "OK". Pentru a ieși din acel modul, jucătorul trebuie să apese pe butonul fizic, care va determina acțiunea pe ecran a butonului "OK".

Jocul revine la meniul principal, iar jucătorul poate începe să se joace din nou.

## Hardware Design

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

### Lista de piese folosite:

- 1 x Arduino Uno R3
- 1 x Cablu USB (pentru alimentare)
- 1 x Display LCD color de 2.4" cu driver ILI9341 si suport card SD
- 1 x Card SD (pentru stocarea de imagini, ex background, desen tuburi, desen pasare)
- 1 x Buton
- 1 x Modul Bluetooth HC-05
- 1 x Breadboard
- Fire
- Rezistoare

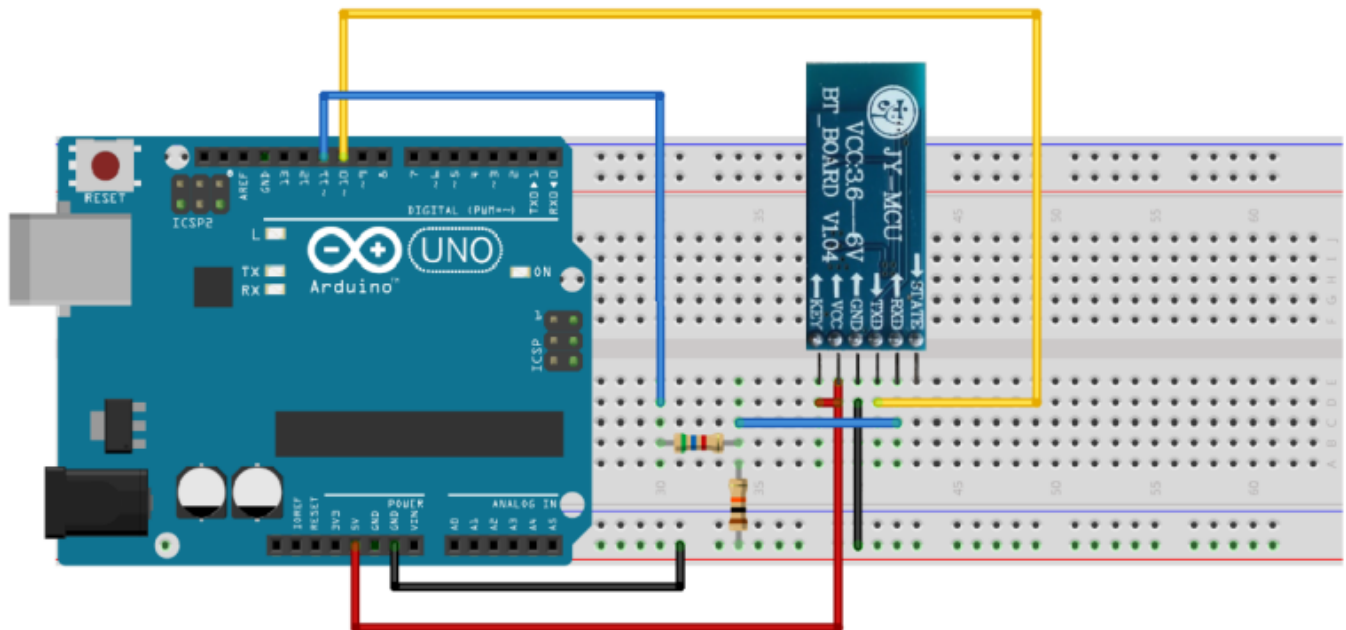
Am adăugat mai jos datasheet-uri pentru componente în secțiunea *Bibliografie/Resurse*

### Exemplu de scheme electrice care conțin componente din cadrul proiectului:

Schema pentru plăcuța Arduino conectată la display LCD și la un buton



Schema pentru plăcuța Arduino conectată la un modul Bluetooth HC-05



Con las dos resistencias creamos un divisor de tensión en la salida del pin 11 del Arduino para limitar el voltaje a 3,3V que es el voltaje que soporta el pin RXD del módulo Bluetooth.

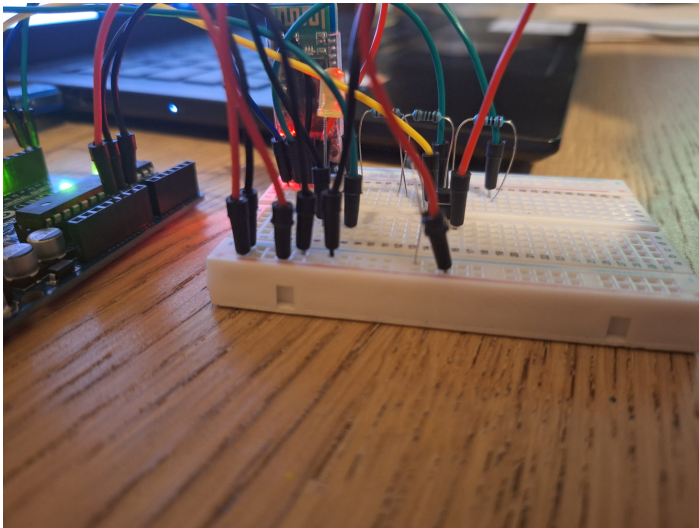
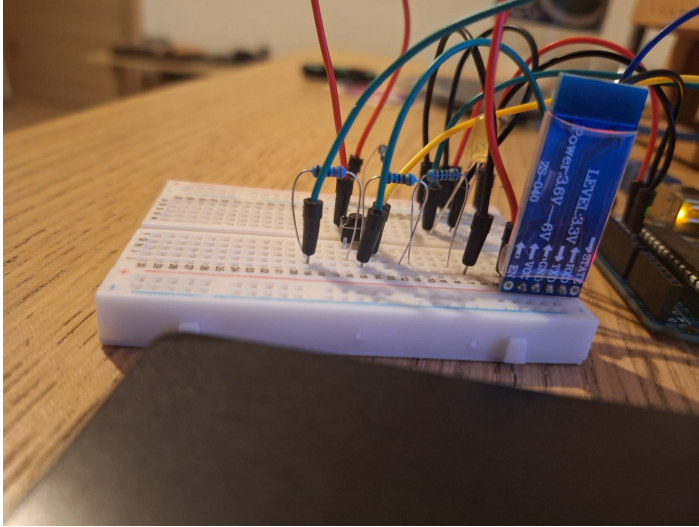
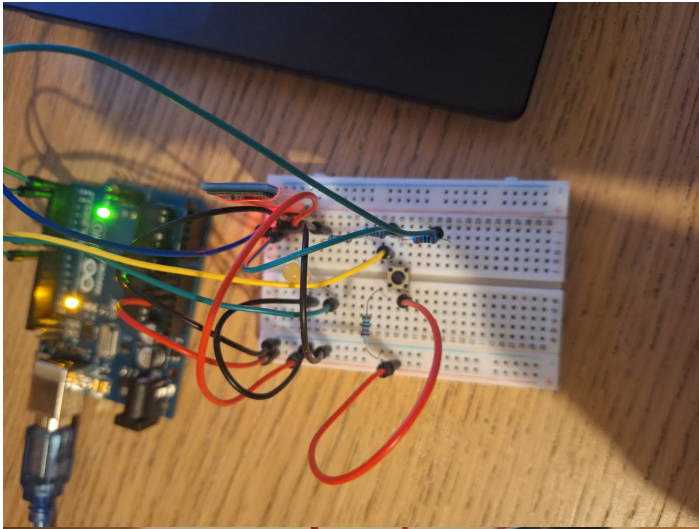
## Schema electrică a proiectului meu

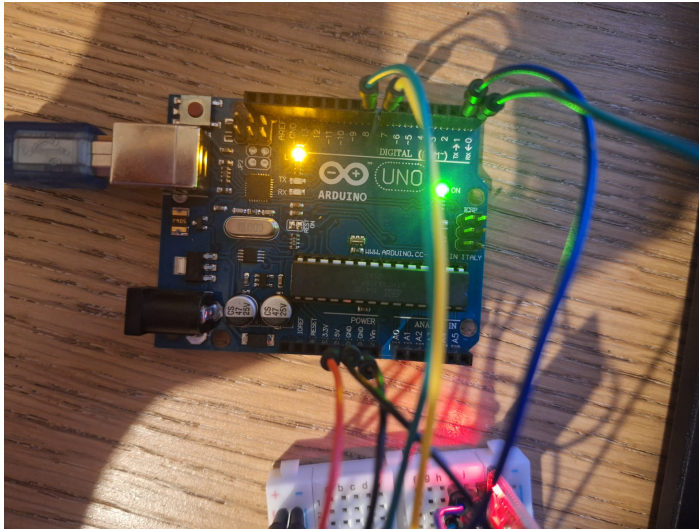


## Descriere pini folosiți

- GND → ground, aici se pune negativul
- 5V → ieșire pentru piesele montate la Arduino (aici conectez pin-ul LED și VCC de la LCD, butonul și pinul VCC de la modulul Bluetooth)
- Pin 1 → TX (Transmițător) - transmite date seriale către RX
- Pin 0 → RX (Receptor) - primește date seriale de la TX
- Pin 7 → Pin I/O responsabil pentru detectarea apăsării de buton
- Pin 9 → DC - semnalul de register pentru LCD
- Pin 10 → CS - semnalul de selecție chip pentru LCD
- Pin 11 → MOSI - scrie date pe LCD
- Pin 12 → MISO - citește date de pe LCD
- Pin 13 → CLK - semnalul de clock

## Circuitul meu în acest moment





## 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 plănuieți să le implementați
- (etapa 3) surse și funcții implementate

### Concepte folosite din cadrul laboratorului:

- GPIO - pentru buton și ecranul LCD
- Întreruperi - pentru buton
- UART - pentru modulul Bluetooth

Pentru programarea propriu-zisă a proiectului, voi folosi Arduino IDE.

Pentru interfața grafică, voi folosi biblioteca Adafruit GFX Graphics Library și Adafruit ILI9341 (corespunzătoare driver-ului LCD-ului meu).

Pentru aplicația mobilă prin care transmit scorul, am folosit MIT App Inventor, care îți permite să faci aplicații mobile simple.

### Diagrama de flow a aplicației



### Funcții din cadrul codului

1. Setup - se ocupa cu setarea inițială a elementelor jocului, cum ar fi highscore-ul, butonul, precum și

pagina principală a jocului, ce conține un buton de start și titlul jocului Flappy Bird

2. Loop - se ocupă de derularea jocului propriu-zis, ce cuprinde trecerea de la meniul principal la joc, animațiile păsării și ale turnurilor, precum și calcularea scorului și detecției de ciocnire

3. drawLoop - este cea unde se creează animația păsării și a turnurilor

4. checkCollision - se detectează ciocniri cu solul și cu turnuri și oprește jocul dacă pasărea se ciocnește

## Rezultate Obținute

Care au fost rezultatele obținute în urma realizării proiectului vostru.

Rezultatele nu sunt tocmai așteptate... Din păcate jocul merge destul de greu, însă este funcțional, cu o grafică bună și care nu prezintă erori.

## Concluzii

Proiectul acesta a fost pe cât de distractiv, pe atât de stresant. Cel mai tare m-a stresat circuitul, consideram că greșesc la orice colț, mai ales la LCD, unde am avut probleme cu firul de LED și asta a cauzat probleme pentru etapa software.

Mi-a plăcut să scriu codul pentru joc și mi-a plăcut și să învăț lucruri noi.

## 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**.

Link github: <https://github.com/MariaAC2/Proiect-PM-Flappy-Bird>

# Jurnal

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

## Grafic Gantt pentru a seta țelurile proiectului



### Țeluri personale de atins în cadrul proiectului:

- Să funcționeze corespunzător componentele circuitului
- Să fac un joc Flappy Bird cât mai distractiv și aproape de original
- Să termin proiectul (cel mai important)
- Să fac modulul Bluetooth să transmită scorul obținut de către jucător
- Să am o grafică frumoasă
- Să am cât mai puține bug-uri

### Update-uri proiect

19.05 - am adăugat documentația pentru hardware design, incluzând o schemă electrică ce are toate componentele folosite de mine, precum și descrierea fiecărui pin folosit

19.05 - am făcut conexiunea la Bluetooth pe care o testez prin intermediul unei aplicații mobile create de mine

20.05 - am adăugat butonul și un led care testează că acesta funcționează

25.05 - am creat un repository cu proiectul în care am inclus funcționalitatea jocului

## Bibliografie/Resurse

### Informații generale

- [Flappy Bird - Informatii generale](#)
- [Flappy Bird - Gameplay](#)

### Hardware design

Link-uri pentru schemele electrice:

- [Schema Arduino Uno R3 + LCD + Buton](#)
- [Schema Arduino Uno R3 + Modul Bluetooth HC-05](#)

Software folosit în crearea schemelor electrice:

- [Tinkercad](#)
- [Fritzing](#)



Resurse pentru datasheet-urile componentelor:

- [Datasheet Arduino Uno R3](#)
- [Datasheet LCD Display cu driver ILI9341](#)
- [Datasheet Bluetooth Module HC-05](#)

### Software design

- [Adafruit GFX Graphics Library](#)

[Export to PDF](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2024/ccontasel/maria.chirnogeanu>



Last update: **2024/05/26 16:36**