

Dungeon Explorer

Introducere

Proiectul consta intr-un joc de tip dungeon crawler genre Roguelike RPG, in care jucatorul controleaza un personaj ce exploreaza un labirint, infrunta inamici si colecteaza diferite resurse sau imbunatatiri, la fiecare explorare labirintul va fi diferit camerele vor fi generate procedural pe baza unor reguli astfel fiecare explorare va aduce o experienta noua pentru jucator . Scopul proiectului este de implementa un joc cu mai multe mecanici precum logica de miscare, gestionarea resurselor, AI basic pentru inamici, interactiunea jucator cu mediul inconjurator si toate acestea folosind niste resurse limitate. Ideea de la care am pornit a fost pasiunea mea pentru jocuri, astfel m-am gandit cum as putea face un proiect care sa fie interesant pentru mine, astfel am ajuns la aceasta idee. Un joc de tip dungeon crawler poate poate fi simplificat pentru a fi adaptat la constrangerile hardware inasa raman si o groaza de mecanici pentru a face jocul interant in continuare.

Descriere generală

Sistemul este compus din următoarele componente:

- Bloc de control pentru mișcare: microcontrolerul ATmega328P primește input de la joystick și keypad și transmite comenzile prin UART către placa principală Arduino Mega. De asemenea, acesta primește mesaje de la Arduino Mega pentru activarea buzzerului atunci când este necesar.
- Joystick-ul este utilizat pentru controlul mișcării caracterului din joc.
- Keypad-ul este folosit pentru atacarea inamicilor și pentru interacțiunea cu diferite elemente din joc.
- Bloc de control pentru display: Arduino Mega se ocupă de afișarea sprite-urilor pe ecran. Acesta primește comenzile prin UART de la cealaltă placă și mută jucătorul sau realizează interacțiuni cu elementele din joc în funcție de acestea.
- TFT Shield v2.2 este utilizat pentru conversia nivelului de tensiune de la 5V la 3.3V, deoarece pinii ecranului funcționează la 3.3V.
- Ecranul TFT LCD de 3.2 inch afișează harta, jucătorul și inamicii.



Hardware Design



Ecranul ILI9341 TFT LCD Display Module este conectat în modul 16-bit paralel la un Arduino Mega 2560. Pinii de comunicație ai ecranului funcționează la 3.3V, în timp ce plăcuța oferă semnale la 5V. Pentru a realiza conexiunea dintre ecran și microcontroller am utilizat un TFT Shield V2.2, care

realizează conversia nivelurilor de tensiune.

Shield-ul se conectează la plăcuță astfel:

- Ocupă toți pinii de power;
- Utilizează pinii digitali 0-7 și 22-53.

Această mapare a pinilor nu a fost realizată manual, deoarece shield-ul permite un singur mod de conectare la Arduino Mega.

Conexiunile dintre shield și ecran sunt următoarele:

- Pinii **DB0-DB15** sunt utilizați pentru comunicația în modul 16-bit paralel;
- Pinii **CS, RD, WR** și **RS** sunt utilizați pentru controlul ecranului și pentru funcționarea bibliotecii grafice;
- Pinii **VDD** și **GND** sunt utilizați pentru alimentare;
- Pinii **CLK, MOSI, MISO** și **SDCS** sunt utilizați pentru comunicarea cu slotul de card SD, necesar salvării datelor persistente.

Pentru conectarea joystick-ului am utilizat:

- **A0** pentru VRx;
- **A1** pentru VRy.

Acești pini au fost aleși deoarece semnalele VRx și VRy sunt citite analogic.

Pentru conectarea keypad-ului am utilizat:

- Pinii **A8-A11**, câte unul pentru fiecare buton (1-4).

Acești pini au fost aleși deoarece suportă întreruperi de tip PCINT, necesare pentru detectarea apăsării butoanelor.

Buzzer-ul a fost conectat la pinul digital 13, deoarece este necesar un pin cu suport PWM pentru controlul acestuia.

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

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/alexandru.jipa2803/acondorache> 

Last update: **2026/05/12 15:44**

