Proiect Proiectare cu microprocesoare: Battleship

Andercou Alexandru grupa 30231



1. Introducere: Problem statement:

Proiectul propus este jocul de tip board game Avioane sau Battleship

Regurile jocului

Jocul se joaca intre 2 jucatori. Fiecare din cei 2 jucatori au o harta pe care isi pozitioneaza navele de batalie si scopul final este sa doboare toate navele aflate pe harta adversarului. Fiecare nava este formata dintr-un numar de parti diferite. La o singura lovitura un jucator poate dobora doar o bucata dintr-o nava, si in functie de nava va primi un punctaj anume.

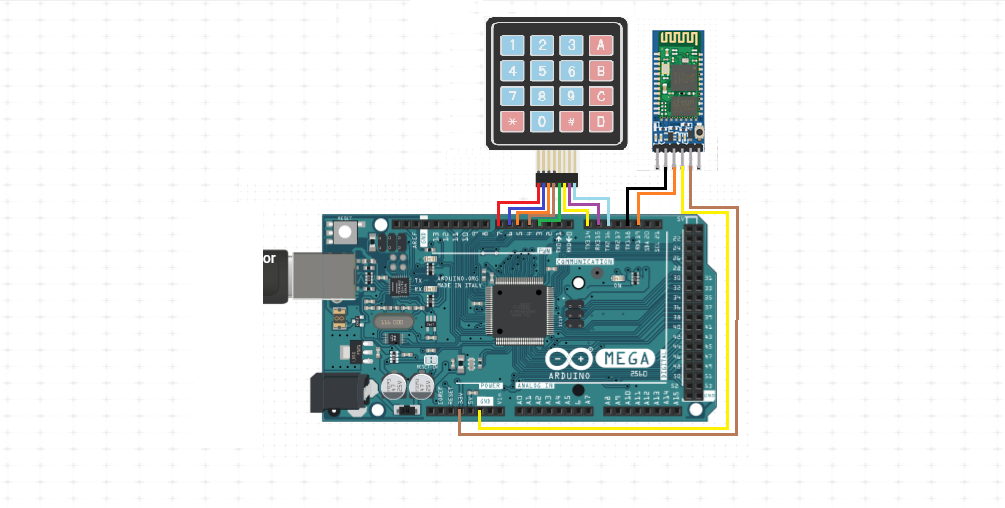
Pentru un jucator exista 5 nave: Destroyer care este formata dintr-o singura bucata si costa un punct, Submarine format din 3 parti fiecare parte costand 2 puncte, un Cruiser din 3 parti fiecare costand 3 puncte, o nava de batalie Battleship din 4 parti costand 4 puncte si un Carrier din 5 parti costand fiecare parte 5 puncte. Jocul se termina cand un jucator a lovit toate navele altui jucator,sau un jucator a ratat toate navele , cel care are cel mai mare scor castiga, cand se doboara o nava acest lucru se anunta jucatorului advers.

Solutia

Solutia aleasa implementeaza jocul folosind o placuta arduino, un modul bluetooth,o tastatura Keypad, si monitorul serial al calculatorului.

Jocul este implementat sub forma de joc tip text si grid. Gridul poate varia in lungime intre 5 si 8 dar valoarea default este 8. Fiecare dintre cei doi jucarori detine doua harti(matriceHidden1, matricePlayer1, matriceHidden2, matricePlayer2),una pe care are amplasate navele proprii(matriceHidden1, matriceHidden2) si una pe care amplaseaza pozitiile unde crede ca se afla navele adversarului, in caz de reusita (hit) se va amplasa un caracter H pe pozitia respectiva, in caz de esec se va scrie un caracter X.La fiecare nava doborata jucatorul este anuntat.

1. Schema Proiectului



Pentru implementare am decis ca primul jucator sa joace pe monitorul serial al unui calculator si sa foloseasca o tastatura externa folosind clasa Serial si Keybord iar cel de-al doilea jucator folosind un terminal de bluetooth de pe un telefon care suporta contact bluetooth , pentru care se foloseste un modul de bluetooth si clasa Serial1.

Modulul bluetooth se numeste HC-05. Pentru comunicare cu programul principal se foloseste protocolul UART si clasa Serial1, pinii modului sunt conectati la pinii placutei de vin:+3.3 V, GND si TX1: 18 si RX1: 19. Functia de scriere folosita este Serial1.print() si functia de citire Serial1.available(),Serial1.read().,Serial1.flush().

Pentru comunicarea cu monitorul serial se foloseste de asemenea protocolul UART si clasa Serial, metodele folosite sunt:Serial.print(),Serial.read() si Serial.flush().

Pentru comunicarea prin tastatura externa se foloseste biblioteca Keypad.h, un obiecty de tip keypad kpd si metoda getKey() pentru a recepta inputul. Pentru construirea obiectului se foloseste metoda: makeKeymap(). Pentru montajul hardware , se folosesc pimii digitali: 7,6,5,4,3,14,15,16. Pinii 7,6,5,4 sunt folositi pentru receptionarea liniei din tastatura iar pinii 3,14,15,16 sunt pentru receptionarea coloanei. Metoda :getKey() returneaza un caracter corespuzator caracterului de pe tastatura.

Codul

Codul a fost structurat in 4 fisiere. 3 fisiere header si unul cu tipul .ino. Acestea sunt :InputDevices.h,OutputDevices.h, Game\_logics.h si BattleshipProject.ino

aceste fisiere modularizeaza functiile programului , separand logica jocului , de interfata de intrare si interfata de iesire care ulterior pot fi modificate in functie de hardware-ul folosit sau de o logica imbunatatita, si ofera flexibilitate in vederea unor modifiari ulterioare.

Logica jocului este impartita in mai multe functionalitati: Game\_run este sursa de pornire a jocului Functiile win,lose sunt folosite pe post de conditii de terminare a jocului si PlacedBatleships1 si PlaceBatleships2 se folosesc pe post de conditii de terminare a etapei initiale in care jucatorii isi plaseaza navele pe hartiile proprii. Metoda Sinked verifica daca a fost doborata o nava ca sa anunte acest eveniment. Functia TryToHit se foloseste in etapa a doua a jocului si trateaza functionalitatea de a incerca de-a dobora o nava a adversarului, aceasta poate rezulta intr-un hit sau miss is schima matricea ascunsa a adversarului si matticea vizibila a jucatorului care a initializat atacul. In aceasta functie creste si scade scorul in functie de hit or miss.

Vectorii battleships\_player1 si battleships\_player2 ,sunt vectori de lungime 5 umpluti cu valoarea 1 se folosesc in functiile: PlacedBatleships1 PlacedBatleships2 si are rolul de a retine ce nave au fost amplasate pe harta ascunsa pentru fiecare din jucator.

Functia Game\_run ruleaza amplasarea navelor cat timp functiile PlacedBatleships1 si PlacedBatleships2 nu returneaza niciuna true.In acest timp se trece printr-o bucla for prin toate tipurile de nave(5) si se efectueaza amplasarea unei nave.

Procesul de asezare este urmatorul: se afiseaza un mesaj initial pe ecranul jucatorului advers de asteptare dupa jucatorul prezent prin functia print\_wait\_message,se printeaza harttiile jucatorului curent cu functia printOnMonitor sau printOnBluetooth, se printeaza nava care urmeaza sa fie afisata,se intreaba pentru locatia si pozitia la care sa fie amplsata cu functia ask\_for\_location,se incearca plasarea navei cu functia Place\_battle\_ship\_on\_grid daca operatia esueaza va returna –1 si se va afisa un mesaj de eroare cu functia print\_eroare1 si se printeaza din nou matriciile modificate.

Partea a doua, jocul efectiv are loc cat nici-un jucator nu a doborat navele adversarului lucru indicat de functia win sau pana cand mai exista spatii libere care nu au fost lovite din gresela de un jucator , caracteristica indicata de functia lose.

Procesul atacului este urmatorul:

Se afiseaza un mesaj de asteptare pe ecranul adversarului,se printeaza hartiile, seintreaba pentru input, locatiia unde sa aiba loc atacul cu functia ask\_for\_location2,se incearca un atac cu functia TryToHit, se verifica pentru corabii doborate la moment=dut dat si se anunta adversarul ca i s-a scufundat o corabie,se reafiseaza pe ecranul jucatorului curent hartiile schimbate.

La sfarsitul jocului se afiseaza pe ambele ecrane scorul final si castigatorul cu functia print\_final\_score,

Jucatorul Player1 de pe smonitorul serial are prioritate jucatorului player2 de pe telefonul mobil, atat la amplasare cat si la atac.

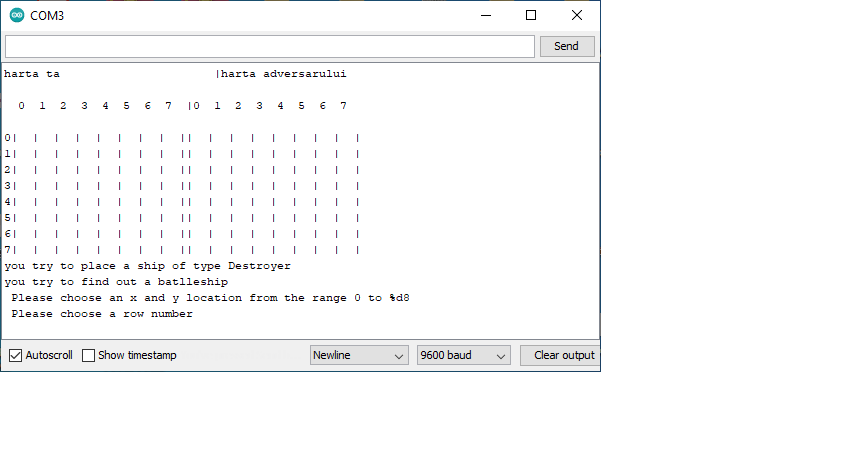
In headerul Input.h sunt declarate functiile ask\_for\_location si ask\_for\_location\_2,este declarata tastatura Keypad kpd si structura location careretine informatiile pentru amplasarea unei nave separat pentru fiecare jucator. Matricea keys si vectorii rowPins si columnPins se folosesc pentru a defini tastele si montajul tastaturii, pentru creiarea tastaturii se foloseste si functia makeKeyMap in interirul constructorului obiectului de tip KeyPad.

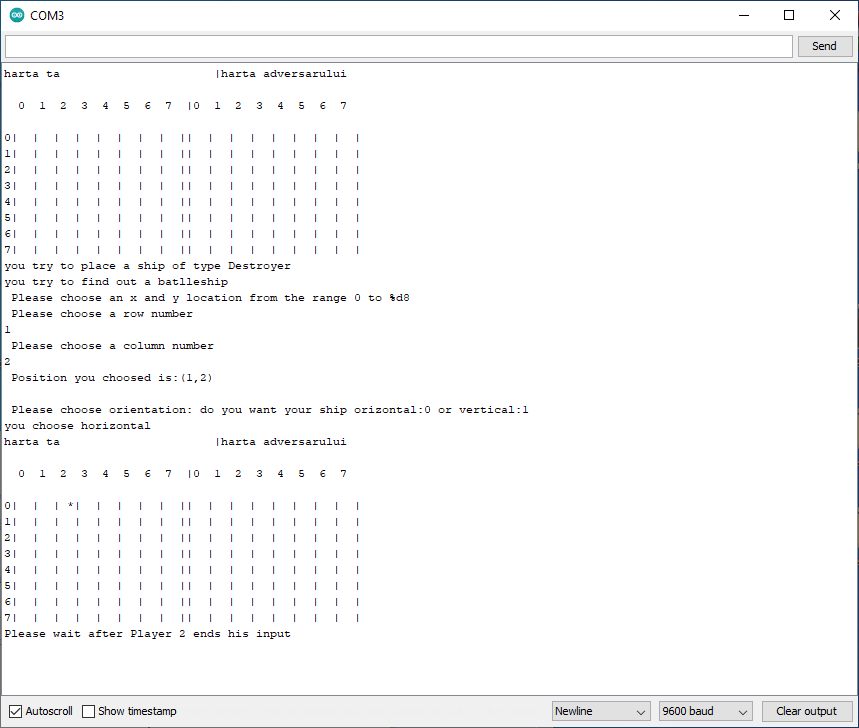
In output.h unt declarate toate functiile de printare inclusiv cele folosite in interiorul headerului input.h,astfel apar functiile : printOnMonitor si printOnBluetooth pentru printarea hartiilor si meja de ajutor care printeaja mesaje text: printHit,printMiss,placeShip, print\_ask\_start, print\_ask\_choose, print\_ask\_column, print\_ask\_orientation1, print\_ask\_orientation2, print\_auxiliary\_x\_location, print\_auxiliary\_y\_location, print\_final\_score, print\_eroare1, print\_Sinked, print\_wait\_message,

1. Imagini ale proiectului.

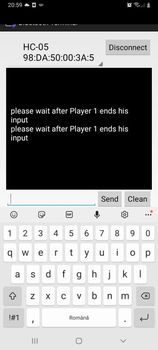
Proiectul are doua parti care interactioneaza intre ele. Prima este monitorul serial din ide-ul de arduino si al doilea este telefonul mobil al celui de-al doilea jucator

Monitorul Serial (parte initiala).





Telefon(Partea initiala)

:

4.Biografie

Jocul Battleship(avioane):

https://en.wikipedia.org/wiki/Battleship\_(game)

Keypad: <https://components101.com/misc/4x4-keypad-module-pinout-configuration-features-datasheet>

HC-05 Bluetooth Module: <https://ardushop.ro/ro/home/132-modul-bluetooth-hc-06-cu-3-pini.html?gclid=CjwKCAiArOqOBhBmEiwAsgeLmVNx76>

<https://users.utcluj.ro/~rdanescu/pmp-lab8-Bluetooth-RO.pdf>

Monitorul Serial si UART:

<https://biblioteca.utcluj.ro/files/carti-online-cu-coperta/336-3.pdf>

Paginile: pag 46-51