

Cristian-Andrei SANDU (66894) - Minesweeper

Autorul poate fi contactat la adresa: cristian.sandu2006@stud.acs.upb.ro

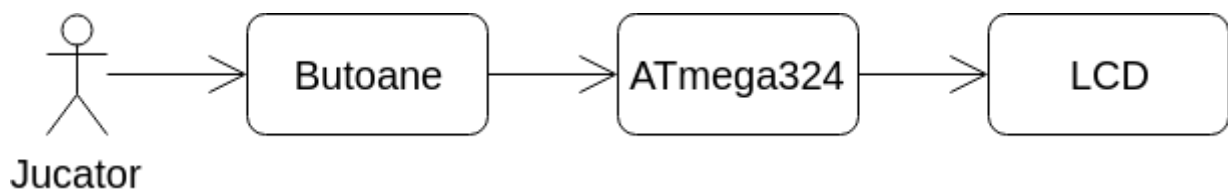
Introducere

Proiectul consta in implementarea jocului Minesweeper pe LCD. Jucatorul trebuie sa expandeze toate casutele tablei de joc, in afara de bombele ascunse, cu ajutorul hint-urilor numerice care spun cate bombe sunt adiacente unei casute.

Am ales acest proiect pentru ca Minesweeper este un joc care imi place foarte mult si poate fi jucat cu usurinta de oricine.

Descriere generală

Schema bloc:



Jucatorul interactioneaza cu jocul cu ajutorul a 4 butoane directionale, care muta cursorul in cele 4 directii, un buton pentru a marca bombele si un buton pentru a realiza selectia/expandea patratelul de sub cursor.

Jocul este afisat pe LCD(84×48 pixeli) sub forma unui grid de dimensiuni fixe, cursorul reprezentand un highlight in jurul patratelului selectat.

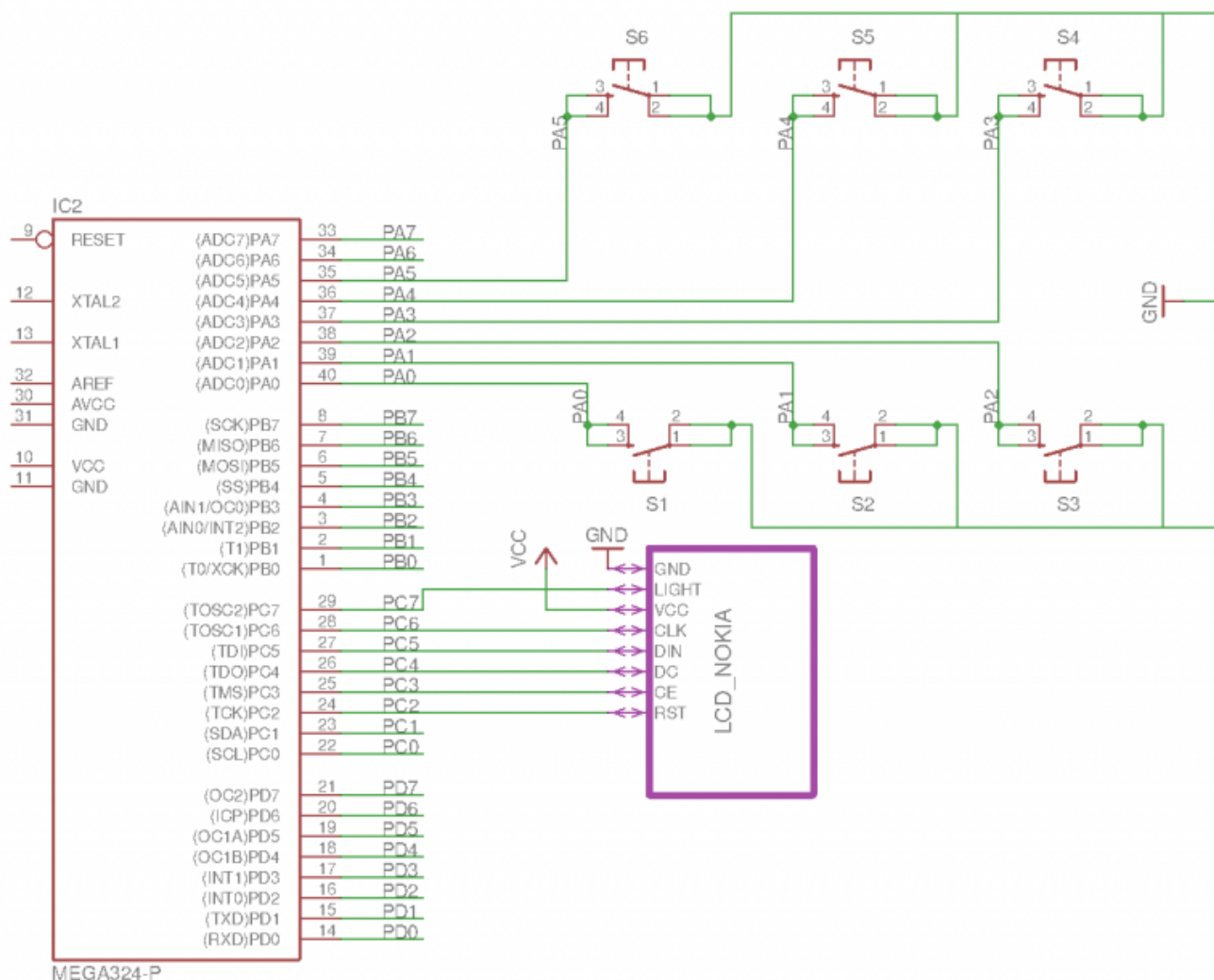
Hardware Design

Lista de piese:

- ATmega324
- componente pentru realizarea placii de baza
- 6x push buttons

- modul LCD cu Controller PCD8544
- regulator tensiune LE33 Ip2950cz
- 1x condensator 10uF
- 1x rezistenta de 1K, pentru backlight
- placuta de prototipare
- conectori

Schema electrica:



Am ales sa folosesc PORTC pentru a conecta display-ul, pentru ca avea pinii dispusi la fel ca cei ai lcd-ului (GND, PC7 cu rezistenta pe traseu pentru backlight, VCC etc.) si PORTA pentru butoane, pentru ca era pe aceeasi parte a placutei.

Software Design

Semnificatia fisierelor

- Biblioteca display, preluata de aici si adaptata pentru cerintele mele:

- nokia5110.c|h - functii de lucru direct cu display-ul: initializari si desenare primitive(caracter, sir de caractere, linie, dreptunghi), ultimele doua fiind scrise de mine.
- nokia5110_chars.h - array cu caractere ASCII printabile(fontul standard 3x5), stocate in PROGMEM; luate de [aici](#)
- Desenarea elementelor jocului:
 - graphics.c|h - diverse functii pentru desenarea board-ului, a cursorului, a status-ului din partea stanga a ecranului, mesaj de sfarsit de joc etc.
- Logica pentru joc:
 - game.c|h - design-ul este descris mai jos
- Altele:
 - utils.c|h - cuprinde initializare de timer pentru ceasul jocului, initializare seed pentru random()

Functiile propriu-zise sunt descrise pe larg in surse.

Design-ul jocului

Jocul contine 6x12 casute in care pot fi bombe, starea fiecărei casute fiind retinuta in structura:

```
enum field_state { HIDDEN, CLICKED, FLAGGED };
/* Minefield struct */
typedef struct {
    uint8_t isBomb;
    uint8_t bombCount; /* no. of bombs around this field */
    enum field_state state;
} minefield_t;
```

Informatiile despre jocul general fiind retinute intr-o structura de tipul:

```
/* Game states */
enum game_state { READY, ONGOING, WON, LOST };

/* Game info struct, for easier argument passing */
typedef struct {
    cursor_t cursor;
    minefield_t board[GAME_HEIGHT][GAME_WIDTH];
    enum game_state state;
    uint8_t hidden_count;
    uint8_t flagged_count;
    volatile uint16_t seconds_passed;
} game_info_t;
```

Logica este tratata la fiecare apasare de buton, in functie de game_state-ul in care se afla jocul si de butonul apasat:

- tastele directionale muta cursorul cu o pozitie in directia corespunzatoare(in spate programul curata cursor-ul vechi, face mutarea si redeseneaza cursorul la noua pozitie)
- la apasarea SELECT se expandeaza casuta curenta si vecinii care nu au bombe in jurul lor(fiind apelata recursiv pentru acestia) si se verifica sfarsitul jocului(LOSS daca "s-a dat click" pe o bomba sau WIN daca numarul de casute neexpandate este egal cu numarul bombelor)

- la apasarea FLAG:
 - se face toggle pentru casuta curenta intre starile FLAGGED/HIDDEN, putand fi un numar maxim de flag-uri egal cu numarul de bombe
 - la sfarsitul jocului, se reseteaza jocul

In partea stanga a ecranului este o zona “rezervata” pentru informatii ce tin de status:

- timpul, in secunde, ce a trecut de la inceperea jocului(porneste la apasarea oricarui buton)
- numarul de bombe ramase / numarul total de bombe
- mesajul de sfarsit de joc, cu indicatie pentru resetarea jocului

Chestii ce tin de hardware, dar nu se potrivesc la categoria Hardware Design:

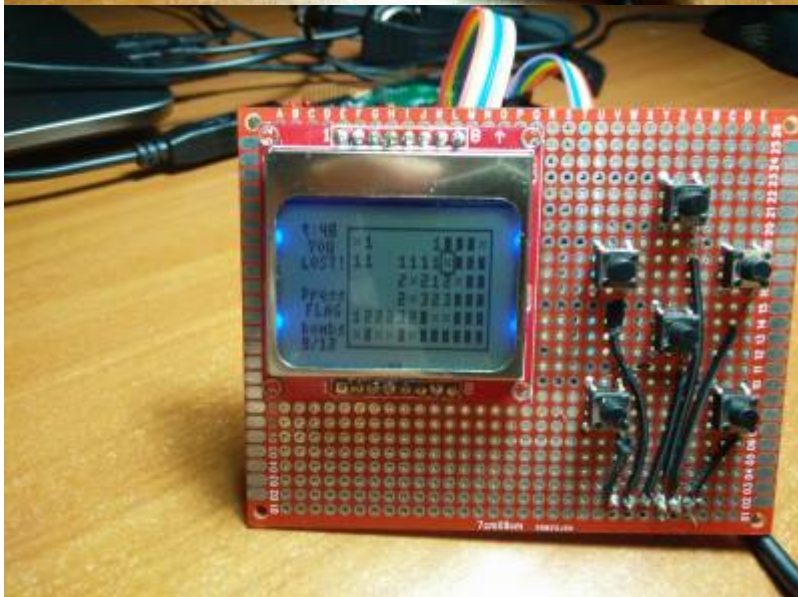
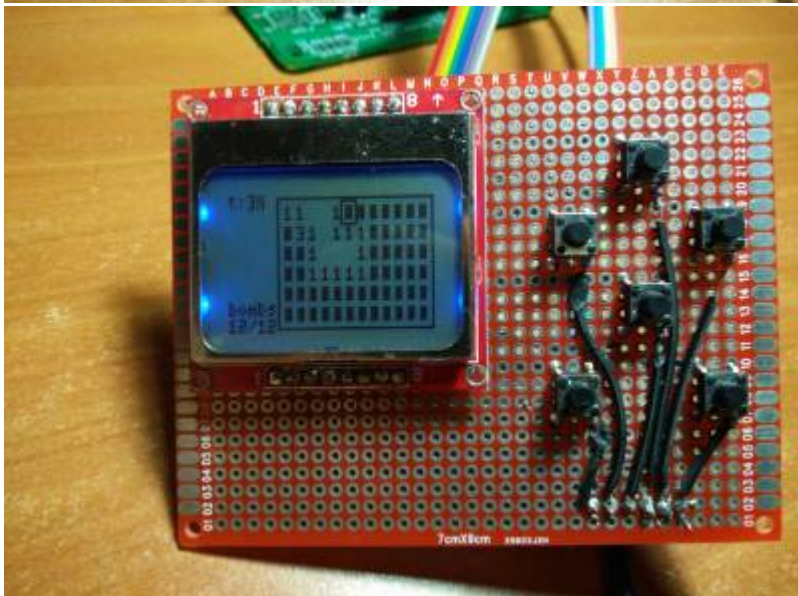
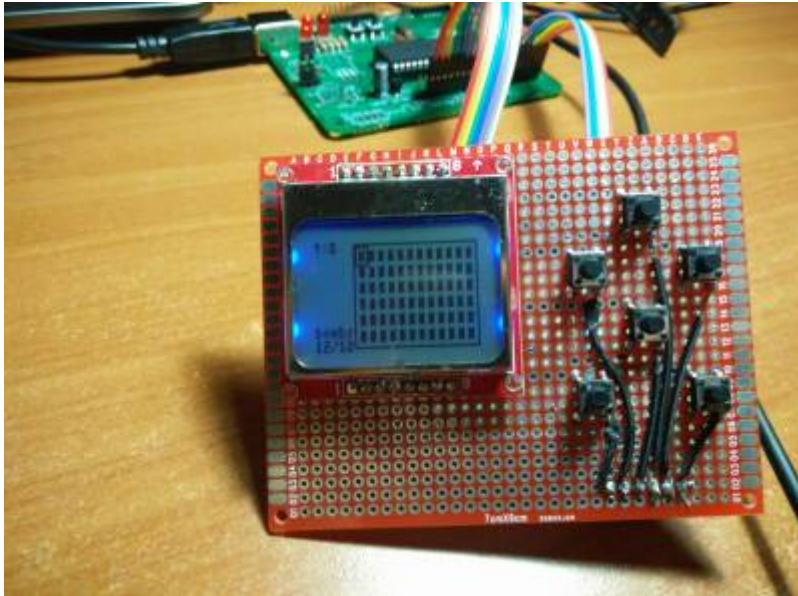
- am folosit ADC-ul pe un pin lasat in aer pentru obtine niste entropie pentru a stabili seed-ul pentru rand(), necesar la generarea aleatoare a bombelor
- am folosit Timer1 pentru a genera intreruperi la fiecare secunda, pentru a mentine timpul trecut de la inceperea jocului
- semnificatia butoanelor este urmatoarea:
 - SUS - muta cursorul o casuta in sus
 - JOS - muta cursorul o casuta in jos
 - STANGA - muta cursorul o casuta in stanga
 - DREAPTA - muta cursorul o casuta in dreapta
 - STANGA-JOS - butonul SELECT(asemanator click-stanga in cazul jocului pe desktop)
 - DREAPTA-JOS - butonul FLAG(asemanator click-dreapta in cazul jocului pe desktop)

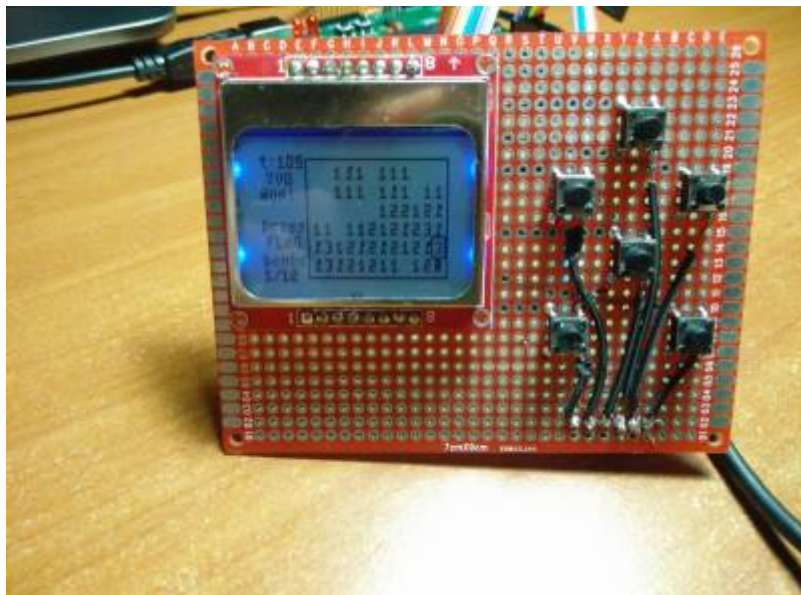
Rezultate Obținute

Rezultatele sunt conform asteptarilor initiale.

Am reusit cu succes sa implementez un joc de minesweeper controlat prin 6 butoane simple si afisat pe un ecran LCD de Nokia, conectat la uController-ul ATMEGA324.

Sunt foarte multumit de cum a iesit partea de software, iar in ce priveste hardware-ul, pe viitor m-as gandii mai mult la cum as atasa display-ul pe placuta de test(aici, pinii “ies prea putin in afara”, si conexiunile cu cablurile mama sunt cam subrede).





Concluzii

Nu a fost un proiect indeosebi de dificil, dar mi-a placut sa lucrez la el. Planuiesc ca pe viitor sa mai fac si alte "proiecte", utilizand microcontroller-ul si placuta de baza, vazand potentialul acestora si fiind incantat de rezultatele obtinute aici.

Download

[Arhiva cu proiectul](#)

Jurnal

- 1-7 mai 2017: realizare si verificare placuta de baza
- 8-14 mai 2017: finalizare parte hardware - lipit componente, testat display si butoane folosind functiile din biblioteca pe care planuiesc sa o utilizez; [pic1](#), [pic2](#)
- 15-21 mai 2017: finalizare parte software
 - schimbat biblioteca pentru display folosita, pentru ca am gasit alta mai simpla si care se mapa mai bine pe ce aveam eu nevoie
 - proiectat aplicatie, asa cum am descris la sectiunea de software design
 - scris cod
 - testat, debugging
- 22 mai 2017: finalizare proiect
 - prezentare proiect la laborator
 - rezolvat un mic bug ce aparea atunci cand toate flag-urile erau utilizate prematur
 - completat documentatie pe wiki

Bibliografie/Resurse

Resurse software

- [Biblioteca pentru LCD](#), luata de [aici](#) si extinsa.

Resurse hardware

- [Datasheet controller display](#)
- [Datasheet ATMEGA324](#)
- Documentația în format [PDF](#)

From:
<http://cs.curs.pub.ro/wiki/pm/> - **PM Wiki**

Permanent link:
<http://cs.curs.pub.ro/wiki/pm/prj2017/aaldescu/minesweeper>

Last update: **2017/05/24 00:51**

