

# Alexandru NECULAI (81997) - Maze Runner

Autorul poate fi contactat la adresa: **Login pentru adresa**

## Introducere

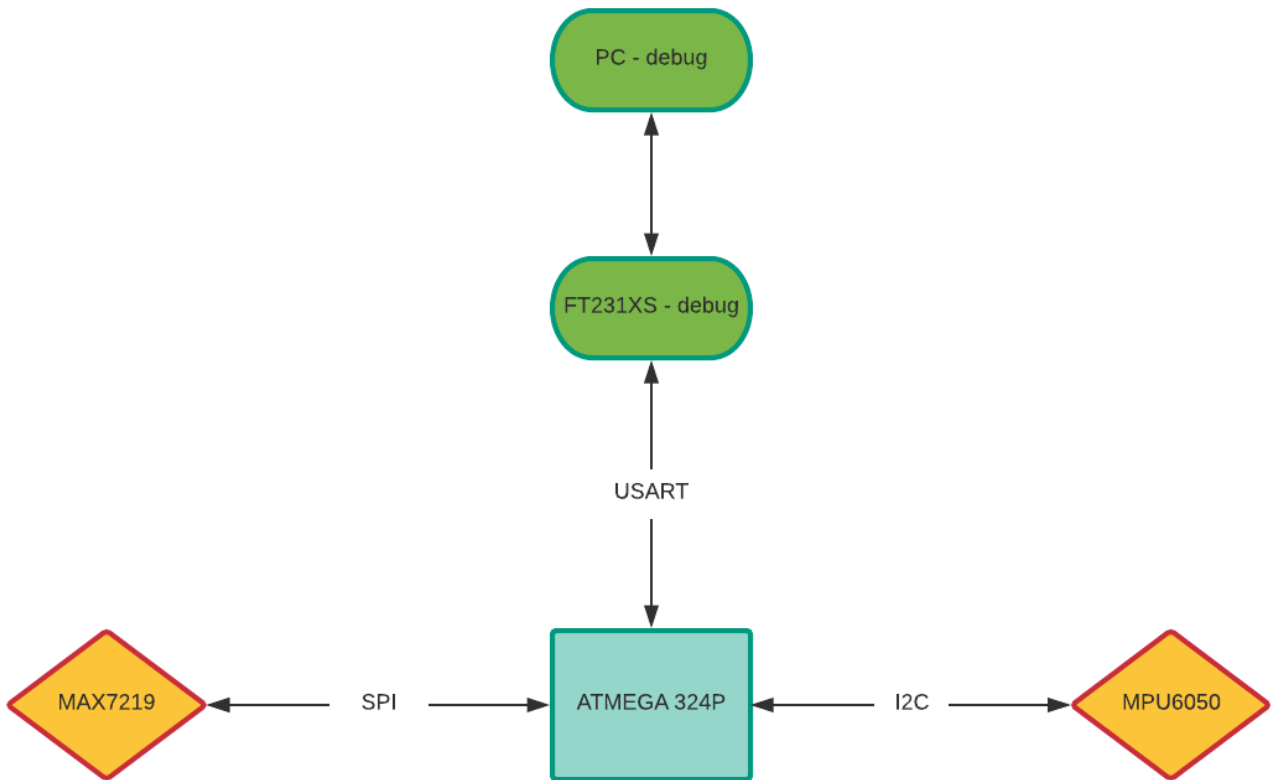
Proiectul isi propune implementarea unui joc ce consta in rezolvarea unor labirinturi, cu ajutorul unui modul de matrici de LED-uri, un accelerometru/giroscop si placuta de baza.

Scopul proiectului este acela de a aprofunda materia invatata pana acum la laborator intr-o maniera placuta, realizand un device ce foloseste majoritatea tehnologiilor prezentate in cadrul laboratoarelor.

Va rezulta un device ce va permite rezolvarea labirinturilor intr-un mod amuzant - bazat pe efecte vizuale si recompense, care va relaxa/amuza utilizatorul.

## Descriere generală

- Microcontrollerul ATMEGA324P preia date de la modulul MPU6050 - care are accelerometru si giroscop - le prelucreaza si comanda modulul MAX7219 in mod corespunzator (in functie de nivelul la care jucatorul se afla - vor fi 3 nivele, cate unul pe fiecare matrice, iar a 4 a va fi folosita pentru a indica faptul ca miscarea este permisa sau nu) - efectul dorit este urmatorul: atunci cand accelerometrul este inclinat intr-o parte, jucatorul sa se miste o casuta din labirint in acea directie, asta doar daca casuta este libera - in caz contrar un efect de clipire si un X vor indica faptul ca miscarea nu este permisa;
- Cu ajutorul modulului FT231XS se face debug, informatiile preluate de la MPU6050 (dar nu numai) fiind transmise pe seriala (UART) calculatorului;
- Schema bloc:

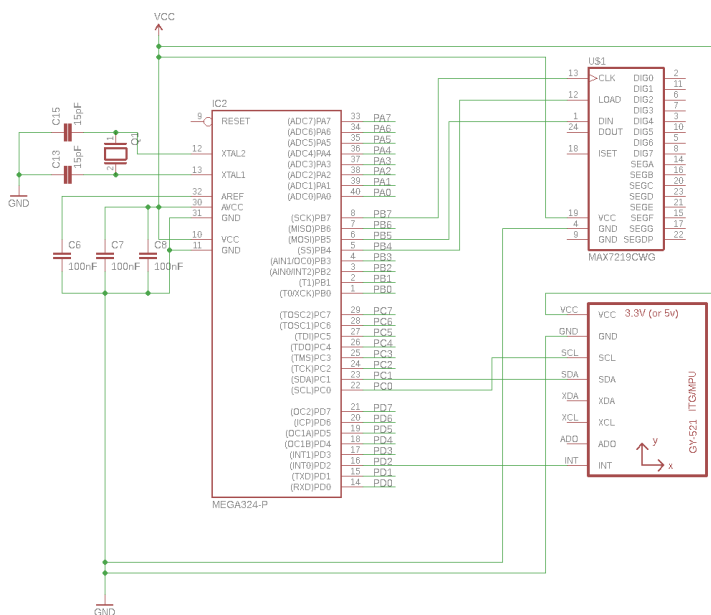


## Hardware Design

### Lista de piese:


- piesele pentru placuta de baza;
- modul accelerometru si giroscop cu 3 Axe MPU6050;
- modul cu 4 matrici de LED-uri MAX7219;
- modul/chip FT231XS;
- 3 condensatoare ceramice;
- plexiglas;
- suruburi;

### Schema electrica:




## Software Design

**Mediul de dezvoltare folosit:** Atmel Studio 7.0 (Windows) + HIDBootFlash (din resurse) pentru programarea efectiva;

**Biblioteci și surse 3rd-party:** cele prezente in Atmel Studio, setarea si utilizarea USART-ului din laborator si, foarte importante, doua biblioteci scrise de Davide Gironi (credits in the source code) atat pentru MPU6050, cat si pentru MAX7219 - daca cineva doreste sa utilizeze vreunul din aceste module sa apeleze cu incredere la aceste biblioteci deoarece sunt foarte utile 

### Surse și funcții implementate:

- Am avut in vedere o implementare cat mai modulara a functionalitatilor jocului - atat pentru claritatea si coerenta codului, cat si pentru usurinta depanarii in caz de erori; in acest sens exista functii specializate (denumite sugestiv si comentate) pentru fiecare actiune care trebuie luata pentru a implementa cu succes jocul;
- De asemenea - am folosit cat se poate de mult capacitatile placutei si ale modulelor prezente in cadrul proiectului - de ajutor au fost data-sheet-urile, bibliotecile anterior mentionate, laboratoarele si Google 
- Astfel, initial, device-ul afiseaza intr-o bucla mesajul "PUSH TO PLAY" - "spart" in 3 frame-uri pentru a putea fi scris pe modulul de matrice utilizat; acest lucru este realizat intr-o bucla cat timp nu a fost apasat butonul de pe placa PD6; detectarea apasarii butonului se face printr-un ISR, avand a priori initializat butonul (directie, rezistenta de pull-up etc), intreruperile pe PCINT3, dar si intreruperile globale - sei();
- Dupa ce a fost detectata apasarea butonului functia "start signal" da un mesaj ce contine "3 2 1 0 GO" - cu un efect de clipire - pentru a avertiza jucatorul de inceperea jocului;
- Se deseneaza primul labirint si jucatorul - acesta se misca prin inclinarea accelerometrului in

directia dorita de mers; daca directia este buna (adica nu este un perete in noua pozitie/daca nu se iese de pe harta) player-ul isi modifica pozitia si un "tick" este desenat in cea de a 4 a matrice, iar in caz contrar este desenat un X; (aici sunt utilizate datele de la accelerometru pentru a detecta intentia de miscare)

- La sfarsitul unui nivel - cand player-ul a ajuns la iesirea din labirint, se trece in noul nivel (un alt labirint pe urmatoarea matrice de LED-uri), iar in cel vechi este desenat un smiley face; acest lucru se intampla la fiecare nivel; la sfarsitul celui de al treilea nivel este afisat mesajul "GGEZ" (end\_game\_animation) folosind un efect de clipire pentru a anunta jucatorul de finalizarea jocului;
- Demn de mentionat: am structuri interne pentru un nivel (Matrix) - fiecare structura continand locul de plecare al playerului in cadrul nivelului, dar si locul de final al nivelului alaturi de harta respectiva; am abstractizat la nivel software labirintul de LED-uri cu o matrice de valori binare (world\_x) pentru a putea detecta miscarea corecta/gresita (check\_collision, check\_border, move etc) si pentru a putea actualiza cu succes noua pozitie a jucatorului (aceste elemente se gasesc in header-ul utils.h); efectele de clipire sunt realizate cu un array de delay-uri (delay\_array) si o functie speciala de delay (Delay\_ms) care primeste o variabila drept parametru, si nu o constanta precum functia clasica "\_delay\_ms()"; formele pentru semne, litere, cifre le-am implementat eu;

## Rezultate Obținute

Rezultatul este un device ce implementeaza un joc de labirinturi, avand miscarea jucatorului controlata din accelerometru - complet functional;







## Concluzii

Proiectul a fost o activitate interesanta, care m-a invatat cum sa utilizez un microcontroller alaturi de diverse module (datasheet, biblioteci etc), dar si cum sa fiu mai rabdator si cat de complex poate fi un device chiar si de mici dimensiuni.

## Download

[alexandru\\_neculai331ca.zip](#)

## Jurnal

2018-04-07 - alegerea proiectului  
2018-04-12 - achizitionarea componentelor pentru placa de baza  
2018-04-17 - receptia placii de baza  
2018-04-22 - gandirea proiectului, componentele necesare, schema bloc  
2018-05-03 - lipirea componentelor pe placa de baza  
2018-05-06 - schema electrica a proiectului (EAGLE)  
2018-05-08 - bootloader si primul program pus pe placa  
2018-05-19 - scriere cod + teste  
2018-05-23 - finalizare documentatie + aranjare cod

## Bibliografie/Resurse

Biblioteca MAX7219 -

<http://davidegironi.blogspot.ro/2013/08/avr-atmega-multiple-8x8-led-matrix.html>

Biblioteca MPU6050 -

<http://davidegironi.blogspot.ro/2013/02/avr-atmega-mpu6050-gyroscope-and.html#.WwVstUjRCUk>

EAGLE - <https://www.autodesk.com/products/eagle/free-download>

AtmelStudio - <https://www.microchip.com/mplab/avr-support/atmel-studio-7>

- Documentația în format [PDF](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2018/amocanu/alexandru.neculai>



Last update: **2021/04/14 15:07**

