

Smart Chessboard

Introducere

Proiectul constă în realizarea unei table de șah inteligente, capabilă să ofere următoarele mutări posibile pentru ambii jucători.

În plus, tabla integrează și un ceas de șah pentru meciuri de Speed Chess și oferă posibilitatea salvării mutărilor efectuate într-un joc pe un card SD.

Un astfel de proiect poate fi util pentru cineva care începe să joace șah și vrea să se antreneze într-un mod mai interactiv.

Descriere generală

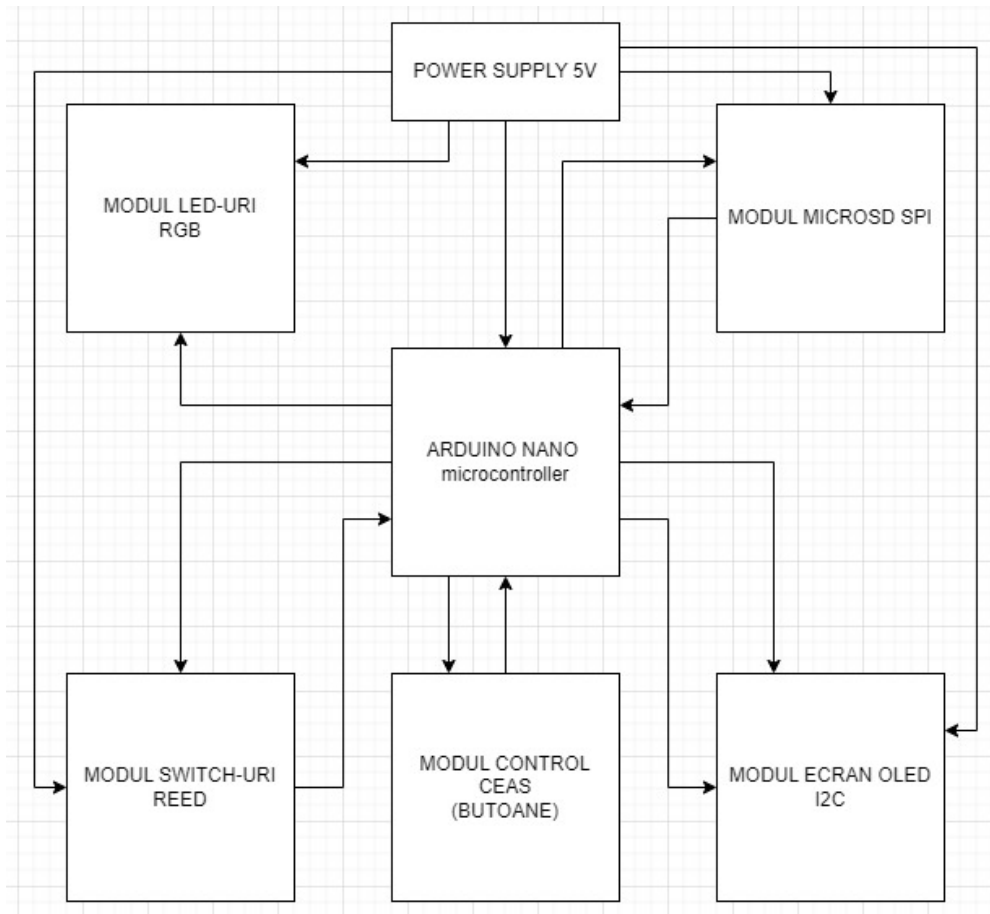
Tabla de șah este controlată cu ajutorul unui Arduino Nano cu microcontroller ATmega328P.

În fiecare din cele 64 de căsuțe de pe tablă există câte un switch magnetic (Reed) care comută la ridicarea piesei și câte un LED RGB care ajută la semnalizarea mutărilor posibile. Fiind multe switch-uri sunt folosite 8 registre Parallel-In-Serial-Out înseriate pentru a comunica cu microcontroller-ul printr-un singur pin de date. LED-urile sunt de tipul celor din benzile LED, care se pot înseria pentru a le aprinde folosind doar un pin de date.

Pentru afișajul ceasului de Speed Chess este folosit un modul de ecran OLED care comunică pe I2C, iar ceasul în sine este controlat folosind numărătoarele de pe ATmega și butoanele jucătorilor.

Un modul de card SD este folosit pentru a salva mutările din jocurile de șah, comunicând prin SPI.

În plus, o sursă de tensiune externă este adăugată pentru a oferi ceva mai reliable decât pinul de 5V de pe Arduino și conectorul USB-C.



Listă Piese

- plăcuță Arduino Nano cu USB-C (<https://www.sigmanortec.ro/placa-dezvoltare-nano-v3-ch340-atmega328p-au-type-c-16m-5v>)
- baterii
- 64 de switch-uri Reed (<https://www.sigmanortec.ro/Comutator-magnetic-Reed-N-O-p161249015>)
- 64 de LED-uri WS2812B-MINI (<https://www.tme.eu/ro/details/ws2812b-mini/diode-led-smd-colorate/worldsemi/>)
- 8 shift registers SN74HC165N (<https://www.tme.eu/ro/details/sn74hc165n/registri-de-deplasare/texas-instruments/>)
- modul card SD pe SPI (<https://www.optimusdigital.ro/en/memories/1516-microsd-card-slot-module.html>)
- ecran OLED pe I2C
- rezistori

- condensatoare
- butoane

Hardware Desgin

Pentru partea de tablă de șah am ales să-mi construiesc un PCB care să încorporeze LED-urile, switch-urile și regiștrii PISO. De pe acest PCB mă conectez la un Arduino NANO, aflat pe un breadboard alături de 2 butoane și un display OLED (care compun ceasul de șah), dar și de modulul microSD pe care se salveaza mutările.

Schema electrică:

-regiștrii, înseriați câte 4, dar comandați de același semnal de LATCH și CLOCK:



-switch-urile Reed, conectate cu rezistențe de pull-down:



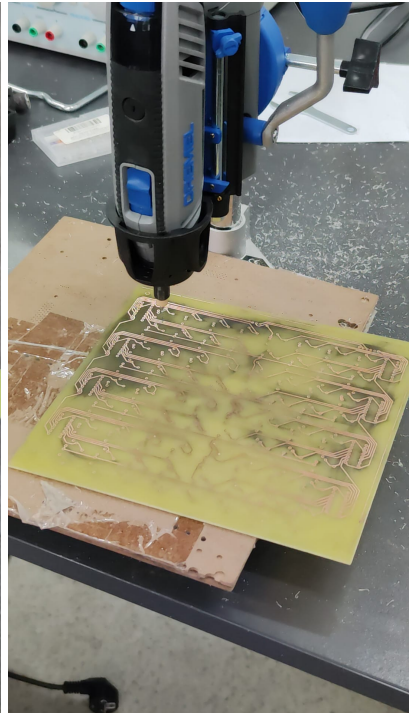
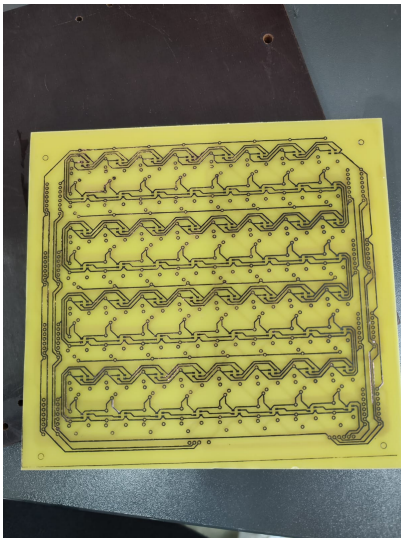
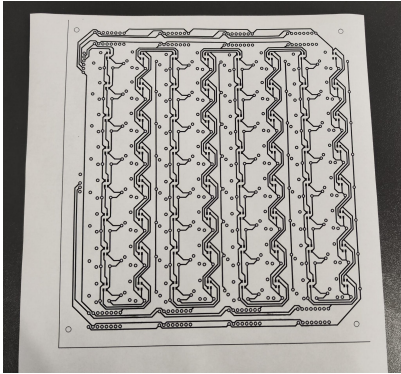
-LED-urile, care au câte un condensator de decoupling de 100nF:

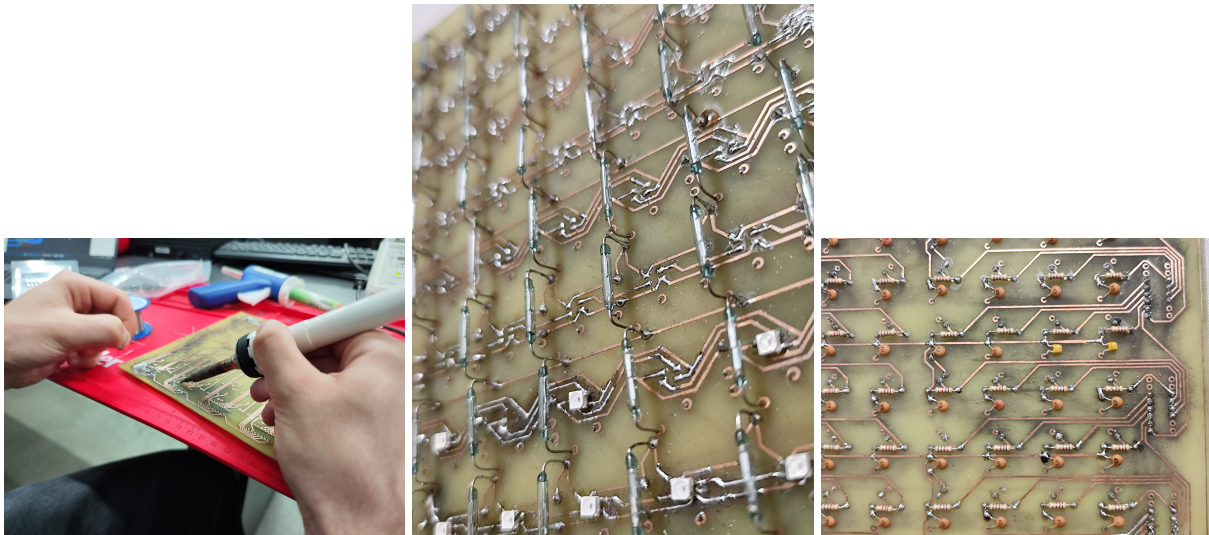


Design-ul PCB-ului l-am făcut cu EasyEDA. Este un design în 2 layere care sunt conectate între ele de unele borne ale componentelor:



Am scos la imprimantă top layer-ul și bottom layer-ul PCB-ului și l-am transferat pe o placă de textolit laminată cu cupru. Am corodat placa cu persulfat de sodiu și am început să lipesc piesele.





Pe breadboard componentele sunt conectate astfel:

- cele 2 butoane sunt conectate la pinii D2 și D3 ai Arduino-ului pentru a folosi întreruperile externe INT0 și INT1, folosind rezistențele interne de pull-up
- display-ul OLED este conectat la 5V, la GND, la A4 - SDA și la A5 - SCL
- modulul de card microSD este conectat la 5V, la GND, la D10 - CS, la D11 - MOSI, la D12 - MISO și la D13 - SCK
- ceilalți pini: D4 - intrare de la primii 4 regiști, D7 - intrare de la ceilalți 4 regiști, D5 - ieșire de ceas pentru regiști, D6 - ieșire de latch pentru regiști - de pe breadboard se conectează la 5V și la GND și PCB-ul, pinii de CLK_INH ai regiștrilor se conectează tot la GND



Software Design

Descrierea codului aplicației (firmware):

- mediu de dezvoltare: PlatformIO
- librării: Arduino.h, SSD1306Ascii.h, SSD1306AsciiWire.h, Wire.h, SD.h, SPI.h

Din punct de vedere software am implementat proiectul ca pe un automat de stări. Am realizat aceasta cu ajutorul unui vector de funcții, câte una pentru fiecare stare.



Am pornit de la framework-ul Arduino cu (setup() și loop()). În loop() se rulează funcția care desemnează starea curentă care se actualizează cu starea returnată de aceasta. Tot în loop se afișează și pe display-ul OLED timpul rămas pentru jucătorul comun.



După acest exemplu de pe Wokwi, <https://wokwi.com/projects/306024460940476993>, am făcut o funcție care citește din ieșirile regiștrilor înseriați prezența pieselor pe tabla de șah în 2 valori unsigned int pe 32 de biți. Când LATCH este 0 regiștri copiază valorile pe care le primesc la intrările paralele. Apoi LATCH trece pe 1, iar prin toggle pe CLK se citește câte un bit de pe ieșirea serială.



Pentru GPIO am folosit configurare cu regiștri ca la laborator. Pentru apăsarea butoanelor de pe breadboard am folosit întreruperile externe INTO și INT1 care comută jucătorul curent.

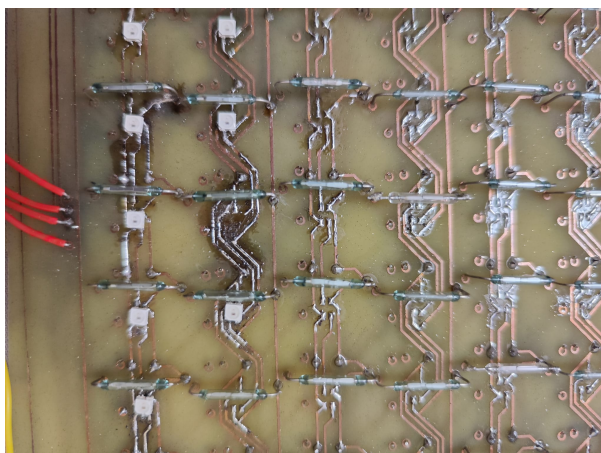
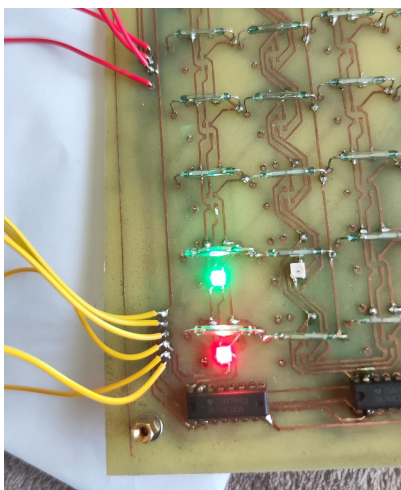


Pentru ceas am configurat TIMER1 de pe microcontroller în modul CTC astfel încât să genereze câte o întrerupere la fiecare 1 secundă.

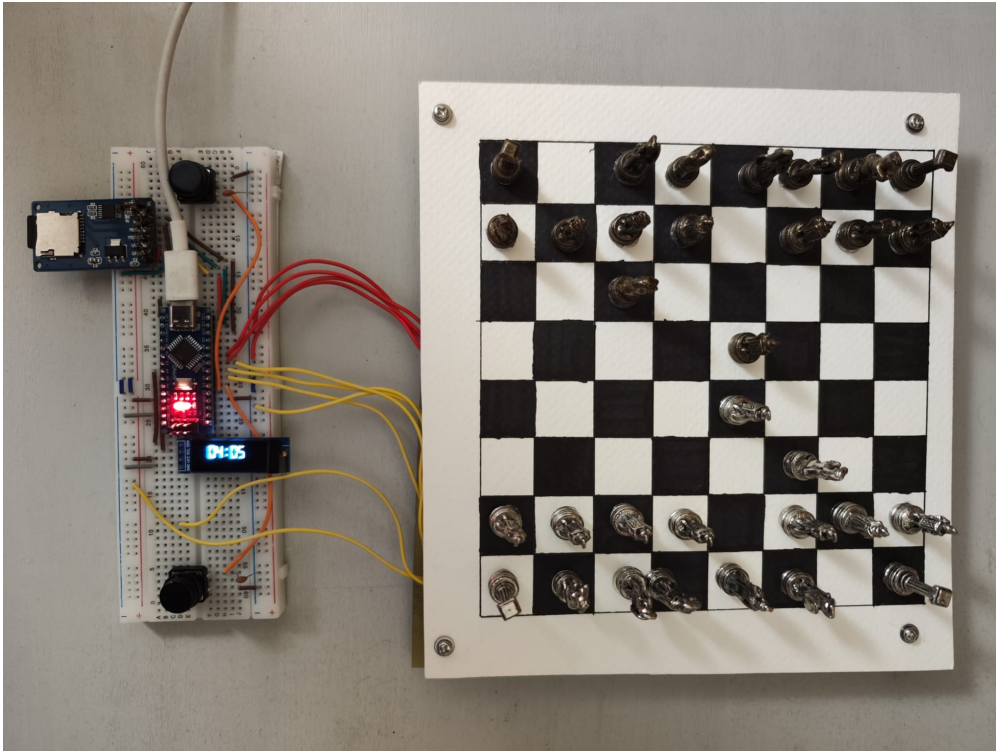


Rezultate Obținute

Nu am reușit să lipesc LED-urile, fiind prea mici am stricat câteva topindu-le când am încercat să le lipesc pe PCB. Am dezlipit și niște circuit de pe PCB în încercarea de a le lipi și m-am lăsat păgubaș. Îmi pare rău de ele, erau mișto la aspect și foarte ușor de programat :(.



În ceea ce privește tabla de șah, regiștrii funcționează ok, însă switch-urile cam dau rateuri, se sparg foarte ușor, având un înveliș de sticlă, și cele 2 piese care ar trebui să facă contact când apropii magnetul de switch devin prea îndepărtate/fac mereu contact. Nu am reușit să joc un meci de șah :(.



Ceasul funcționează corect totuși :).

Concluzii

Ar fi fost mai ușor dacă mă apucam mai din timp de PCB poate-l comandam de la vreo farbică.

Trebuia să găsesc altă soluție pentru detectarea pieselor, ceva mai puțin sensibil.

Poate era mai bine doar să simulez căsuțele de pe placă cu niște switch-uri simple. Cam așa ceva:

<https://wokwi.com/projects/398615863123489793>

Download

[pmdarius.rar](#)

Bibliografie/Resurse

https://www.ti.com/lit/ds/symlink/sn74hc165.pdf?ts=1716707719745&ref_url=https%253A%252F%252Fwww.google.pl%252F

<https://docs.wokwi.com/parts/wokwi-74hc165>

<https://www.studiopieters.nl/arduino-nano-pinout/>

[Export to PDF](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2024/vstoica/darius.stefan>



Last update: **2024/05/27 00:37**