

Cannon

Introducere

Autor: George-Bogdan Angheloiu

Grupa: 334CB

Proiectul constă într-un tun rotit de două servomotoare, care folosește video preluat de la o cameră pentru a ținti. Acest video este trimis către un display montat. Lansarea proiectilului este realizată tot cu ajutorul unui servomotor.

Scopul proiectului este de a fi folosit într-un joc, se pot realiza multe jocuri pe baza unei turete. Nu există o utilitate propriu-zisă în afară de divertisment.

Descriere generală

✘ Controller-ul este un Arduino UNO. Este responsabil de a controla mișcarea și tragerea tunului acționând servomotoarele la apăsarea butoanelor, de a prelua video de la cameră și de a-l trimite către ecran.

Servo Orizontal/Vertical rotesc tunul stânga-dreapta sau sus-jos.

Buton rotație verticală/orizontală controlează rotația la apăsare.

Servo Tragere pune în funcțiune mecanismul de lansare a proiectilului.

Buton Tragere acționează mecanismul de tragere, se pune în funcțiune și nu mai servește nicio funcție până nu se completează tragerea și revine la poziția inițială.

Camera OV7670 este folosită pentru țintire, trimite video către controller pentru a fi afișat.

Display TFT LCD de 2.4" afișează imaginile preluate de la controller, din cauza vitezei reduse a microcontroller-ului și a faptului că se trimite video prin SPI, nu se poate folosi cu un refresh rate obișnuit pentru video, dar este conceptual utilizabil pentru țintire.

Hardware Design

Aici puneți tot ce ține de hardware design:

- listă de piese
- scheme electrice (se pot lua și de pe Internet și din datasheet-uri, e.g. <http://www.captain.at/electronic-atmega16-mmc-schematic.png>)
- diagrame de semnal
- rezultatele simulării

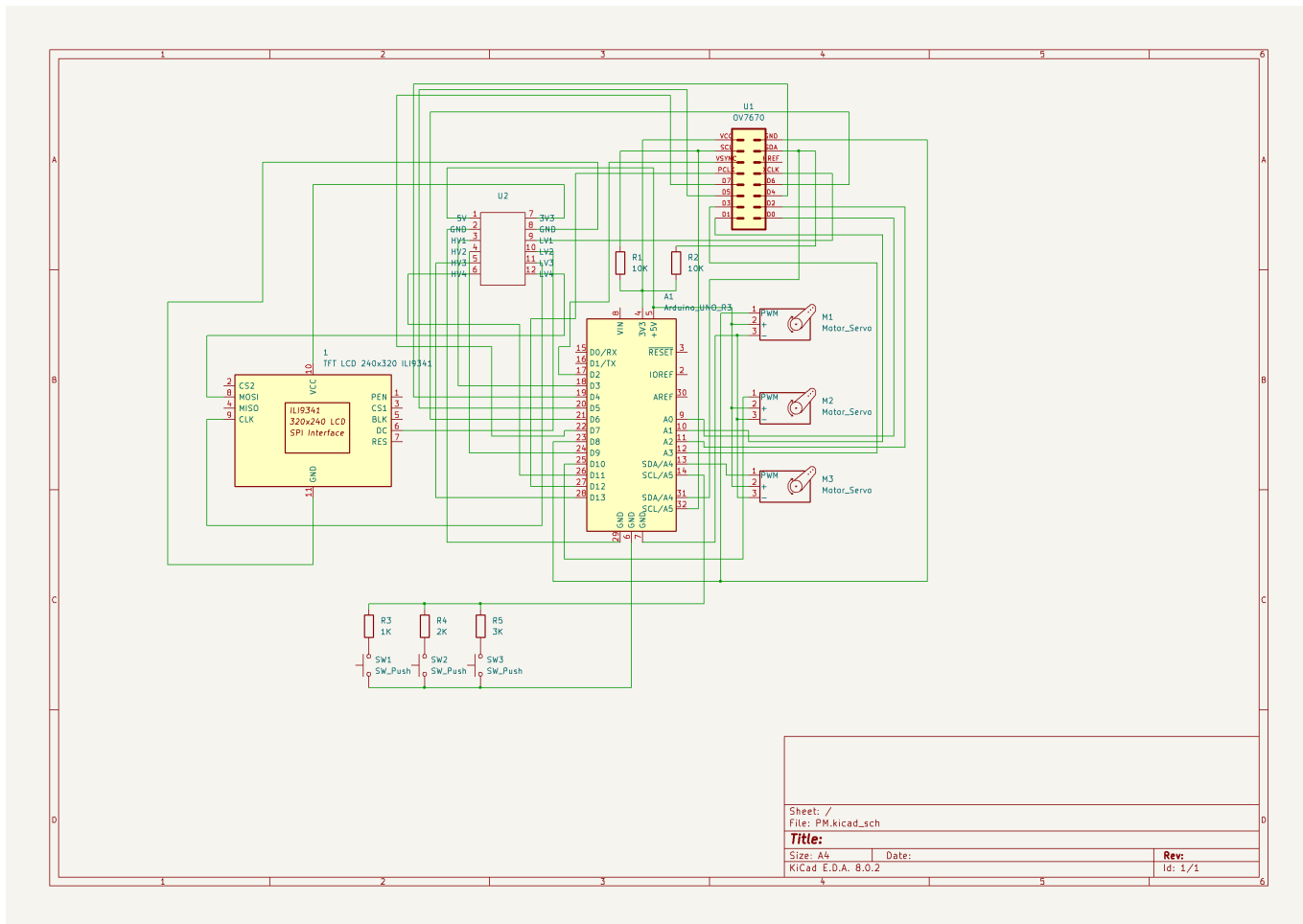
Listă componente pentru proiectul original:

- 1x Arduino UNO R3
- 3x Servomotor SG90
- 3x push buttons
- 1x level shifter
- 1x Modul Camera OV7670
- 1x Modul LCD de 2.8" cu SPI și Controller ILI9341
- 1x rezistență 1K Ω
- 1x rezistență 2K Ω
- 1x rezistență 3K Ω
- 2x rezistențe 10K Ω
- fire tată-tată și mamă-mamă

Listă componente suplimentare:

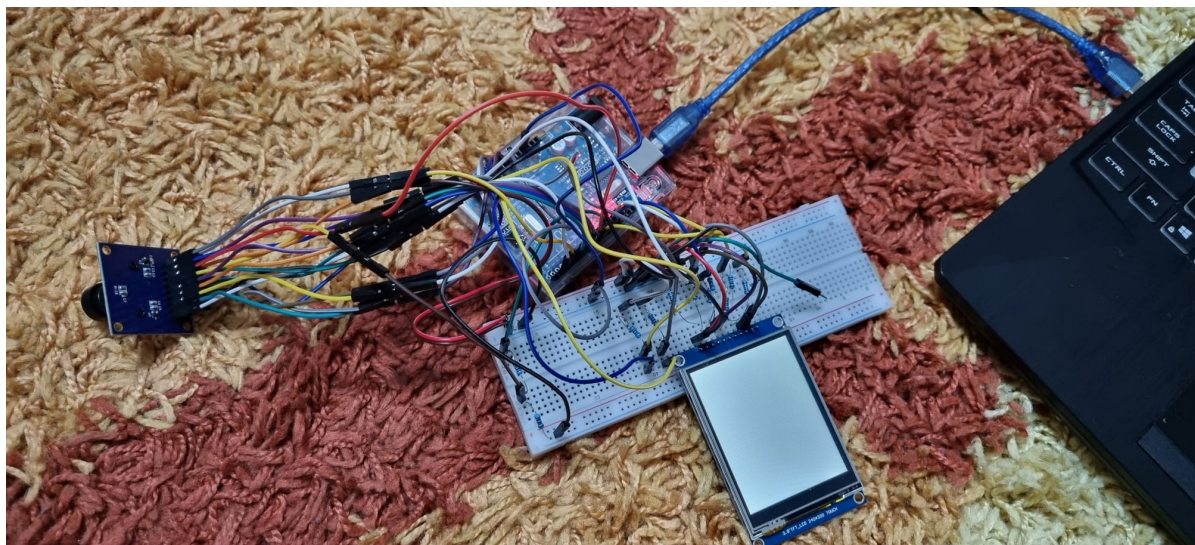
- 6x rezistență 1K Ω
- 6x rezistență 2K Ω

Schema electrică:



Modificări aduse la schema electrică:

- Am folosit rezistențe suplimentare în locul level shifter-ului care nu mai mergea cum trebuia
- Anumiți pini au fost schimbați și anumite legături adăugate din cauza problemelor cu display-ul și camera



Software Design

Mediu de dezvoltare: Visual Studio Code + PlatformIO

Librării folosite:

- Arduino
- Adafruit ILI9341
- [Modul cameră OV7670 cu Arduino](#)

Ideal, aş fi folosit proiectul LiveOV7670 la care aş fi adus modificări pentru a acţiona servomotoarele. Din păcate, acest proiect nici măcar nu a compilat, asemenea majorităţii librărilor pe care am încercat să le folosesc.

Adafruit ILI9341 a fost folosit pentru interacţiunea cu display-ul. Am folosit librăria Arduino pentru funcţionalităţile plăcuţei folosite (Arduino Uno).

Pentru iniţializarea camerei, am folosit următoarele comenzi:

```
□arduinoUnoInut();  
camInit();  
setResolution();  
setColor();  
writeReg(0x11, 13);
```

Pentru a captura imagini şi a le trimite către display am apelat în loop:

```
captureImgTft(tft, 320, 240);
```

Rezultate Obţinute

Am obţinut un dispozitiv care realizează conexiunea dintre cameră şi display, după multe teste şi schimbări pe care le-am realizat. După trei schimbări de display, ultimul dintre ele a funcţionat în aproximativ 50% din încercări. Camera a suferit aceleaşi schimbări aleatoare: funcţiona la început, iar apoi selectiv: se trimiteau pixeli, însă foarte încet şi valorile erau eronate de multe ori.


Până la urmă, am reuşit să leg display-ul de cameră pentru câteva teste. Acum, această conexiune nu mai funcţionează dintr-un motiv necunoscut: nu am modificat nimic, nici pe hardware, nici pe software timp de mai puţin de 24 de ore, iar circuitul care funcţiona parţial înainte nu mai merge acum.


Concluzii

A fost o experienţă neplăcută, chiar chinuitoare. Am întâmpinat probleme şi la cel mai mic pas. De fiecare dată când funcţiona ceva, se strica altceva. Am avut probleme de hardware, fiindcă componentele pe care le-am comandat au avut diverse erori, am avut probleme de software în lucrul cu bibliotecile de care aveam nevoie şi de multe ori am avut probleme chiar şi de upload a codului pe plăcuţă.

Mă așteptam să fie o activitate interesantă, care să mă ajute să înțeleg mai bine materia și mi-am dorit de la început să aleg un proiect interesant. În schimb, am rămas dezamăgit, deoarece am stat mai mult să fac debugging, să schimb firele, componentele sau să caut diverse metode pentru a rezolva probleme din cod.

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**. **</note> ===== Jurnal ===== <note tip> Puteți avea și o secțiune de jurnal în care să poată urmări asistentul de proiect progresul proiectului. </note> ===== Bibliografie/Resurse ===== <note> Listă cu documente, datasheet-uri, resurse Internet folosite, eventual grupate pe **Resurse Software** și **Resurse Hardware**. </note> <html>Export to PDF</html>===== Nume proiect ===== Introducere ===== <note tip> Prezentarea pe scurt a proiectului vostru: ** ce face * care este scopul lui * care a fost ideea de la care ați pornit * de ce credeți că este util pentru alții și pentru voi </note> ===== Descriere generală ===== <note tip> O schemă bloc cu toate modulele proiectului vostru, atât software cât și hardware însoțită de o descriere a acestora precum și a modului în care interacționează. Exemplu de schemă bloc: <http://www.robs-projects.com/mp3proj/newplayer.html> </note> ===== Hardware Design ===== <note tip> Aici puneți tot ce ține de hardware design: ** listă de piese * scheme electrice (se pot lua și de pe Internet și din datasheet-uri, e.g. <http://www.captain.at/electronic-atmega16-mmc-schematic.png>) * diagrame de semnal * rezultatele simulării </note> ===== Software Design ===== <note tip> 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 plănuți să le implementați * (etapa 3) surse și funcții implementate </note> ===== Rezultate Obținute ===== <note tip> Care au fost rezultatele obținute în urma realizării proiectului vostru. </note> ===== Concluzii ===== ===== Download ===== <note warning> 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**.*****

cannon.zip

Jurnal

3 mai: idee și descriere proiect

12 mai: selectare componente necesare și achiziționare hardware

16 mai: soldare și testare componente, realizare schemă electrică, implementare hardware parțială

21-25 mai: am întâmpinat probleme cu componentele, pe care am încercat să le rezolv din hardware și software

26-27 mai: testare + update proiect actual

Probleme întâmpinate:

- o parte din firele din pachetul nou-nouț s-au stricat
- am avut 3 display-uri diferite pe care le-am încercat pe rând, fiindcă niciunul dintre ele nu funcționa în mod constant (uneori da, uneori nu); doar le conectam la alimentare și nu se aprindeau întotdeauna
- nu am găsit suficiente resurse pentru a conecta camera de monitor; am avut o singură referință pe care am urmat-o și care nu a funcționat
- probleme de upload a codului pe plăcuța Arduino Uno atunci când sunt conectate componente; pentru testare deconectam 3v3 sau 5v de la Arduino Uno

Bibliografie/Resurse

Resurse Software:

- <https://circuitdigest.com/microcontroller-projects/how-to-use-ov7670-camera-module-with-arduino>
- <https://registry.platformio.org/libraries/adafruit/Adafruit%20OV7670>
- https://registry.platformio.org/libraries/arduino-libraries/Arduino_OV767X
- https://registry.platformio.org/libraries/arducam/arducam_dvp
- <https://www.hackster.io/techmirtz/visual-capturing-with-ov7670-on-arduino-069ebb>
- <https://www.youtube.com/watch?v=R94WZH8XAvM&t=4s>

Resurse Hardware:

- Tutoriale de soldare și legare a componentelor hardware
- Datasheets ale componentelor hardware
- <https://www.youtube.com/watch?v=R94WZH8XAvM&t=4s>

[Export to PDF](#)

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

Permanent link:
<http://ocw.cs.pub.ro/courses/pm/prj2024/azamfir/george.angheloiu>

Last update: **2024/05/27 14:34**

