

# Bad Apple

**Autor: Cristian Cringasu, 336CB**

Mail: cristian.cringasu@stud.acs.upb.ro

## Introducere

### Ce face?

Proiectul va presupune realizarea unei matricii LED 16×16, pe care se vor putea reprezenta diverse filmulete provenite de pe un card SD conectat (optional) la arduino. Se va reda sunet folosindu-se 4 motoare de tip stepper, si se vor selecta filmuletele folosindu-se un ecran LCD + joystick (tot optional).

### Care este scopul proiectului?

Scopul acestui proiect presupune familiarizarea cu conversia video de la o rezolutie la alta, conversia audio la input pentru motoare de tip stepper (probabil in doi pasi: audio → midi, midi → stepper, optional folosindu-se AI pentru audio→midi), dar si familiarizarea cu interconectarea sistemelor hardware embeded la arduino, si implica si o parte tehnica (lipirea LEDurilor, si implementarea unui grid pentru selectie).

### Care a fost ideea de la care am pornit?

Am vazut prea multe videouri pe YT cu Bad Apple :)

### De ce cred ca este util pentru mine, cat si pentru altii?

Poate fi un proiect estetic atragator, si interesant din punct de vedere audio vizual :)

## Descriere generala a functionalitatii

Proiectul va contine un arduino de tip HUB: se vor prelua date de la cardul micro-sd, numele acestora se vor afisa pe ecranul LCD (cate un video pe entry), putandu-se scolla prin videouri cu ajutorul joystick-ului. Dupa ce se va selecta un video, acesta va fi eventual convertit sau doar afisat pe ecranul de tip LED, iar stepererele se vor comporta ca niste difuzoare pe 2 biti (am doar 4 steperere, eventual o sa incerc sa vad daca fac rost de mai multe). Daca conversia video → LED / audio → Stepper va fi prea complicata, voi folosi fisiere pre-calculata pentru audio / video.

Matricea LED va fi implementata ca un grid, vor exista pini pentru linii si coloane, un LED se va aprinde la conectarea inputul corespunzator pentru linie / coloana. Coloanele vor trebui legate la VCC in caz in care se va dori aprinderea unui LED de pe coloana `j`. Iar liniile se vor lega la GND, in cazul in care se va dori aprinderea unui LED de pe linia `i`.

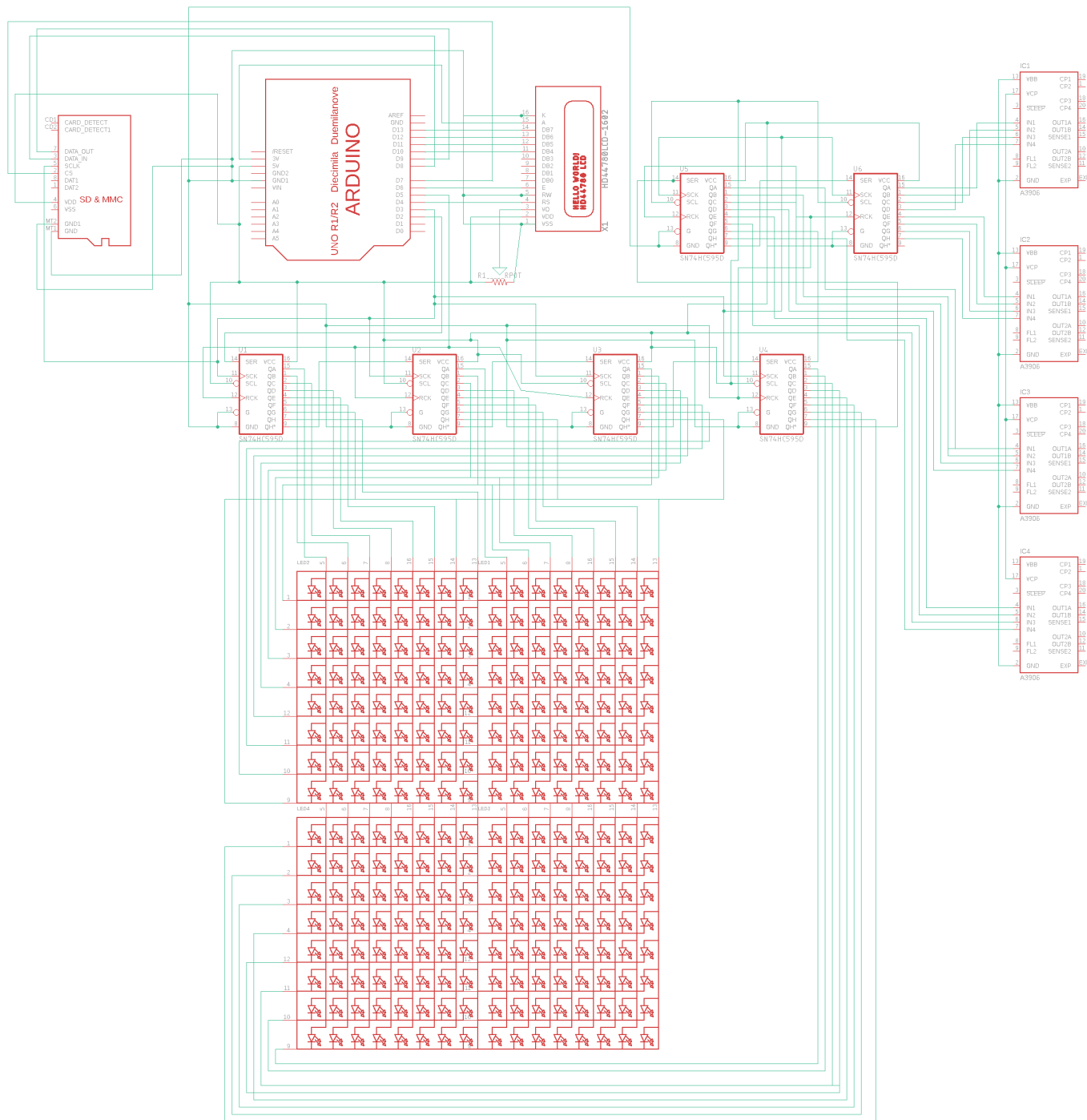
Pentru indexare se va folosi un decodificator serial - paralel, sau 2 decodificatoare (unul pentru linie, altul pentru coloana).

### Schema bloc a proiectului:



## Hardware Design

### Schema Electrica



**Lista de componente:**

1. Placuta Arduino Uno
2. Ecran LCD
3. Rezistente (32 pentru fiecare / coloana din matricea LED)
4. Potentiometru pentru LCD
5. 256 LED-uri (16x16)
6. unul / doua decodificatoarea serial - paralel pentru indexare LED
7. 32 tranzistori NMOS (conexiune coloana / linie LED la VCC / GND)
8. Jumpers (probabil 100% vor fi lipite LED-urile, mai putin conexiunile la LCD / card SD reader)
9. Baterie (9V arduino alimentare + 12V pentru matricea LED)
10. Breadboard

11. PCB (matrice LED)
12. Pistol de lipit
13. Fire de staniu / cupru
14. Suport Plastic PLA 3D printed.

## Software Design

- Cod Arduino :
  - stepper\_music.ino → citește date de la arduino mega, redă audio corespunzător, dacă există
  - SerialStepper.h, SerialStepper.cpp, SerialStepperPlayer.h, SerialStepperPlayerLight.h, MidiFile.h, MidiFile.cpp : drivere și biblioteci implementate pentru steppere paralele adresate cu shift registre +

cod parsare fișiere midi (am renunțat totuși la midi pentru că nu am avut un al doilea card sd :( ) + playere care se folosesc de serialstepper pentru redare sunete în valori nominale (stepperele nu emit sunete audibile la toate frecvențele, depinde mult de cât de repede se poate efectua un pas asincron în cod, arduino e cam lent deci am făcut câteva conversii).

- midi\_player.ino → discontinued (lipsa card sd), redă fișiere midi folosind stepperele într-un model de workpool asincron
- badAppleSong.h → cod sursă parsat din fișiere midi folosind <https://sshdl-7.extramaster.net/tools/midiToArduino/processFile/>, modificat pentru a fi corespunzător cu funcțiile mele
- led\_video.ino → player video pentru fișiere custom de tip .VID, realizate cu ajutorul "led\_convertor.py" (vorbiți mai târziu ce face).
- led\_player.h → funcții ce își parsează fișiere .VID pe chunkuri și redau frame-uri pe matricea led, de asemenea avem și interfață cu shift registrele.
- NewLedI2c - librărie de pe git interfațează I2C pentru lcd
- Cod Python :
  - led\_convertor.py → convertește mp4 la VID, folosind un model scalat la 16×16 HIGH / LOW (led aprins stins calculat cu distanța euclidiană a pixelului la [0,0,0] - black / [255,255,255] - white)
  - midi.py - convertor audio - midi / convertor.py - convertor midi text / midicom parser de fișier
- **necesită un terminal cu python3 instalat**

## Diagrama Activității

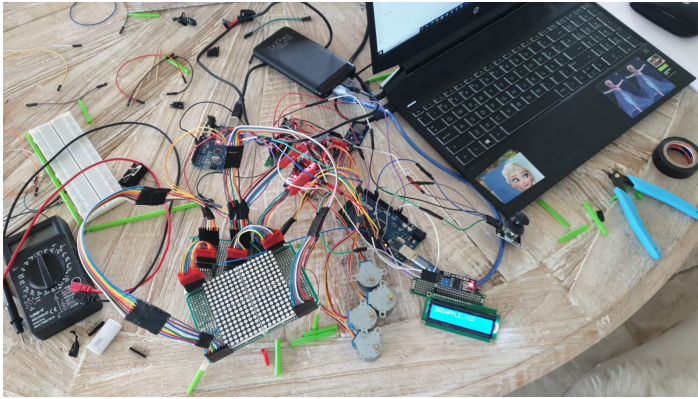


## Rezultate Obținute

### STAGE 1

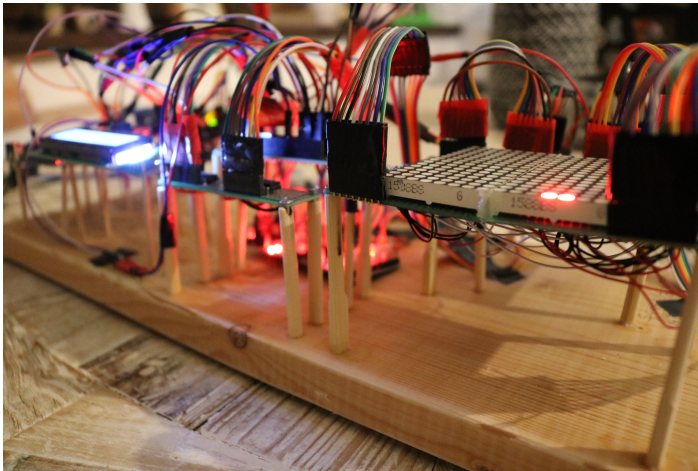
Componentele sunt lipite așezate și gata (nu am alte poze mai vechi, dar oricum, s-a muncit la matricea led pentru lipit fire, la shift registre pentru cablaj, și la steppere relativ am făcut mufe tip rainbow, să stea cât de cât fixe, mai avem și o placută lipite tip breadbord 2 coloane cu GND și VCC, hub pentru celelalte componente, și mai am o placută cu i2c\_lcd).





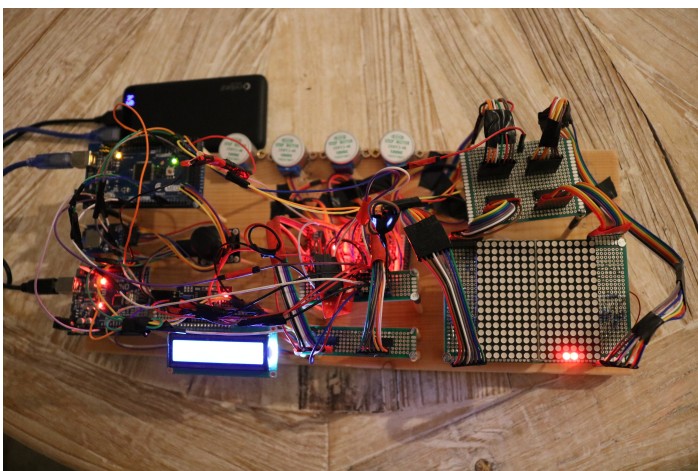
## STAGE 2

Realizare suport de lemn, cu betisoare de sustinere si suruburi.



## STAGE 3

Cable management si vedere de ansamblu.



## VIDEO

[Demo](#)

## Concluzii

## Reusite si Probleme

- Implementarea a fost cat de cat ok, cu probleme la midi file player si in general cu audio al stepperelor.. aparent cu cat sunt mai multe calcule de facut in bucla loop, cu atat audio va fi mai prost, si de asemenea am replicat audio de la stepper ul '0', tradeoff pentru calitate vs cantitate de note cantate simultan (mono acum)
- La lcd au fost probleme mai mult hardware, dar software a fost ok.

## Posibilitati de extindere

- Se foloseste o baterie externa in prezent
- Se poate folosi si un suport mai adecvat, entual cu suruburi si placa de sustinere.. dar alta data facem si asta :))

## Ce am invatat din proiect?

- Sa ma uit pe datasheet in disperare, sa lipesc ca un pro ;) si sa fac debugging pana la 5 dimineata :D <3

## Bibliografie/Resurse

[Midi Arduino Online Convertor](#)

[Arduino Stepper Default Library](#)

[Conversie Frecvente Note - Intervale](#)

[Midi SdCard Player - Timed](#)

[midi-note-to-freq.md](#)

[74HC595 Datasheet](#)

[8x8 dot matrix 1588BS, Pin out](#)

[Arduino Mega 2560 Pin Mapping](#)

[Repository Github](#)

Export în PDF

<https://ocw.cs.pub.ro/courses/pm/prj2021/abirlica/badapple>

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2021/abirlica/badapple>



Last update: **2021/06/04 15:01**