

# Led Controller

## Introducere

Acest proiect, poreclit de mine "Visual-Audio Spectre", se axează pe controlarea unei benzi de LED-uri prin intermediul unui microcontroller Arduino, pentru a lumina în diferite moduri (selectate prin intermediul unui buton) bazate pe semnalele sonore (muzica) primite de la un laptop. Scopul acestui proiect este de a crea un setup de lucru și gaming captivant, care să ofere un mediu estetic plăcut pentru ochi și să îmbunătățească experiența utilizatorului.

Ideea de bază a proiectului a pornit din dorința de a combina pasiunea pentru tehnologie și design într-un singur produs inovator. Cred ca proiectul este util atât pentru mine, cât și pentru alții, deoarece îmbină funcționalitatea și estetica, oferind un element de atracție vizuală și dinamică în orice spațiu în care este implementat.

## Descriere generală

## Schema Bloc



## Hardware Design

### Lista piese

- 1 Microcontroller Arduino UNO R3
- 1 Banda LED WS2812B, 5M, 60 led/M, IP20, 5V
- 11 Jumper Wires
- 1 Condensator 1000 $\mu$ F
- 1 Rezistor 330Ohm
- 1 LM393 Sound Detection module
- 1 Buton
- O sursa de curent (220V  $\rightarrow$  5V,6A)

## Diagrama hardware



## Diagrama electrica



## Montaj



## Software Design

Descrierea codului aplicației (firmware):

- Mediu de dezvoltare Arduino IDE
- Biblioteci externe folosite: FastLED + OneButton
- Algoritmi și structuri: Algoritmi pre-implementati de \*random\*, unde sinusoidale ce pot fi compuse, algoritmi euristici și de decay

Codul a fost scris utilizând informații din trei laboratoare, mai exact

- Laboratorul 2: Întreruperi hardware → cod ce se rulează în momentul în care timer-ul ajunge la valoare dorita
- Laboratorul 3: Timere → cod pentru crearea unui timer ce ruleaza cu o frecventa de 0.5Hz
- Laboratorul 4: ADC → setarea unui pin ce primește informații în format analog de la un senzor de sunet

Logica de baza este destul de simpla. În funcția de setup, fac set-up la banda led folosind funcțiile de setup din biblioteca 'FastLED', configurez pinul A1 pentru a activa ADC-ul, setez timer-ul sa declanșeze o întrerupe odată la 2 secunde și leg butonul care este controlat de pinul 3 de funcția care schimba pattern-ul curent.

În functia principala de loop, citesc de fiecare data informația care îmi este data de senzor, o translatez în spațiul de valori care mă intereseaza [0-NUM\_LEDS-1], după care, cu ajutorul unui switch verific pe ce pattern de luminare sunt în momentul actual și apelez funcția corespunzătoare

pattern-ului.

Fiecare pattern este diferit în felul sau. Primele 5 pattern-uri sunt "standalone", nu se folosesc de valoare citită de pe senzor, și doar luminează pe baza unor funcții matematice ce conferă un aspect interesant și frumos (pe baza de testing). Ultimele 3 pattern-uri folosesc atât intensitatea sunetului citită de senzor pentru a lumina un număr de led-uri proporțional cu aceasta, cât și de întrerupere și timer pentru a schimba în timp culoarea.



Restul de cod poate fi vizualizat la următorul link de [Github](#)

## Rezultate Obținute

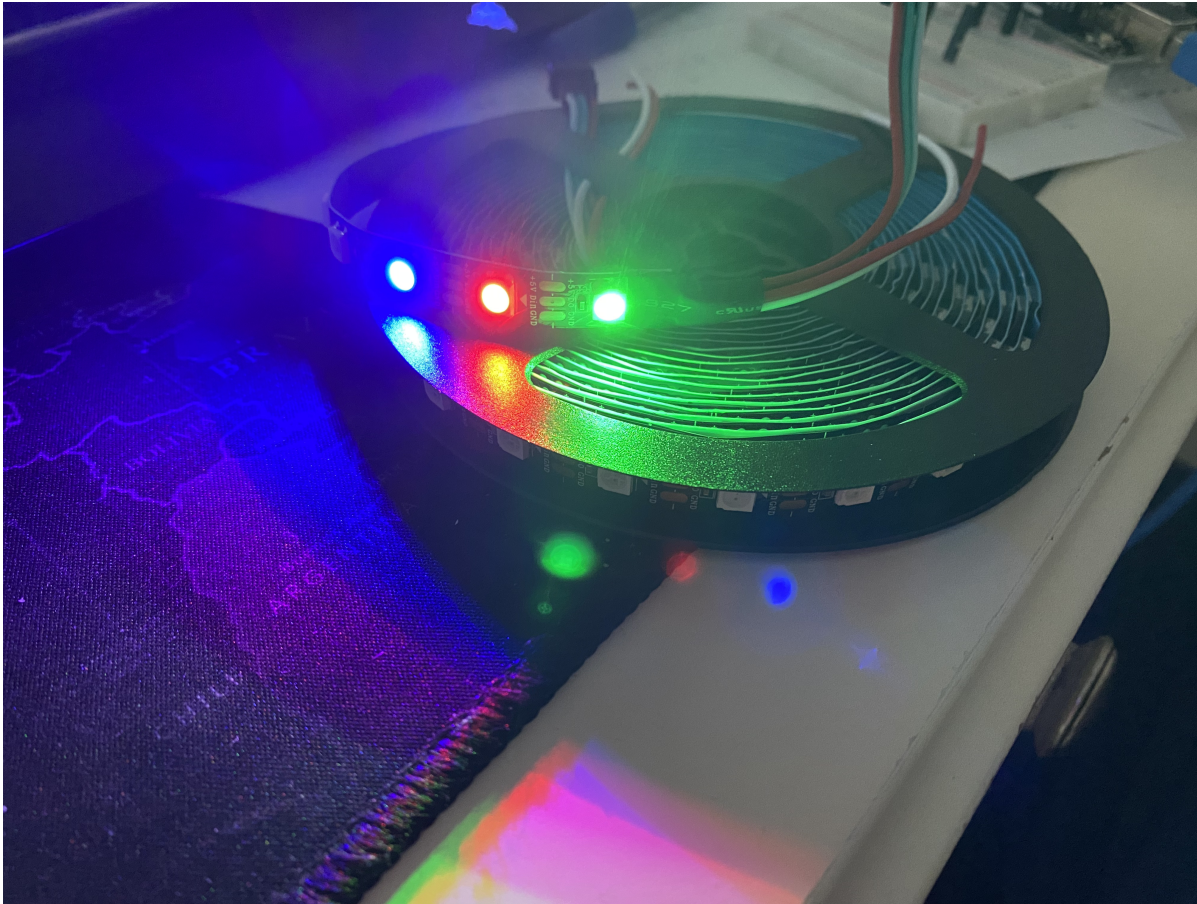
Link video YouTube: [Rezultat](#)

## Concluzii

Consider ca rezultatul final este unul foarte satisfăcător, atingând mai multe milestone-uri decât mi-am propus inițial (6 pattern-uri în plus) și sunt chiar recunoscător ca am avut acest assignment de la facultate deoarece m-a împins să ies din aria mea de confort într-o zonă în care voiam să intru dar nu puteam să încep. Am învățat din greșeli, cum ar fi faptul că trebuie să refac două comenzi deoarece nu am citit bine data sheet-urile, am învățat că un usb normal poate să dea în jur de 500mA-900mA, iar LED-urile pe care le-am achiziționat, în funcție de intensitatea la care lucrează poate să consume și 10A (am cumpărat și atașat o sursă externă de 5V). Per total, sunt foarte mulțumit de această experiență.

## Jurnal

- 20.04.2023 Alegere proiect
- 30.04.2023 Comanda piese
- 05.05.2023 Comandat piesa lipsa
- 06.05.2023 Test initial



- 07.05.2023 Test input intensitate muzica
- 15.05.2023 Adăugare 3 pattern-uri
- 20.05.2023 Adăugare încă 5 pattern-uri

## Bibliografie/Resurse

- <https://www.youtube.com/watch?v=4Ut4UK7612M>
- <https://github.com/FastLED/FastLED>
- <https://github.com/mathertel/OneButton>
- <https://github.com/NeverPlayLegit/Rainbow-Fader-FastLED/blob/master/rainbow.ino>

[Export to PDF](#)

From:

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

Permanent link:

[http://ocw.cs.pub.ro/courses/pm/prj2023/alexau/led\\_controller](http://ocw.cs.pub.ro/courses/pm/prj2023/alexau/led_controller)



Last update: **2023/05/21 18:34**