

Elena Violeta AMZA (78517) - Bluetooth Controlled SingerMoodLamp

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

Introducere

SingerMoodLamp - dispozitiv de tip mood lamp care permite setarea cadrului ambiental dorit prin alegerea unui template de culori si/sau muzica - note muzicale calculate pornind de la valoarea RGB a culori setate.

Proiectul va fii comandat prin Bluetooth prin intermediul unei aplicatii Android MySingerLamp prin comenzi text si prin comenzi vocale.

Desi scopul principal al acestui dispozitiv este cel estetic, acesta are si un scop secundar, cel recreativ - mini joc de lumini si note muzicale deoarece faciliteaza schimbarea cadrului ambiental: lumina/muzica prin intermediul unei aplicatii care este intuitiv de folosit.

Descriere generală

Mod de functionare :

SingerMoodLamp va fii comandata prin Bluetooth prin intermediul unei aplicatii Android MySingerLamp cu 2 posibile optiuni:

1. **comenzi de tip text** date prin intermediul aplicatiei Android care sa permita:

```
selectarea unei anumite culori / a unei melodii / a unui mod
predefinit
on - se aprinde lampa in mod default / off - se stinge lampa
play/stop music
```

2 **comenzi vocale** prin intermediul aplicatiei Android pentru Voice Recongition care sa permita:

```
pornirea/oprirea unui mod predefinit
on/off
(not)music
red/green/blue
```

moduri predefinite:

1. **work**: joc de culori format din nuanțe de verde/galben ; melodie de studiu(notele date de culorile respective) ; intensitate medie .
2. **sleep**: joc de culori format din nuanțe de albatru ;melodie lenta care in timp se va stinge(notele date de culorile respective) ;intensitatea scade in timp .
3. **read**: joc de culori format din nuanțe de portocaliu/rosu(culori calde) ; melodie de studiu (notele date de culorile respective);intensitate slaba .
4. **wake-up**: joc de culori format din nuanțe de rosu/galben ; melodie ritmata care se intensifica in timp(notele date de culorile respective) ; intensitatea crește in timp ;
5. **default**: joc de culori format din toate nuanțele de culori existente ; melodie de studiu(notele date de culorile respective) ; intensitate medie .

OBSERVATIE: Schimbarea graduala a culorii in aceste moduri se va face pe ritmul muzicii.

SCHEMA BLOC:

Conectarea la lampa si setarea starii acesteia se va face prin intermediul unei aplicatii Android - conectare bluetooth(se foloseste modulului Bluetooth HC-05). Comenzile se dau deci pe interfata seriala. In urma interpretarii comenzii date se va aprinde modulul LED RGB din microcontroller conform setarii facute si dupa caz se va reda o nota muzicala.

Hardware Design

Lista piese :

1. placa de baza
2. placa test
3. modul led RGB de 3W
4. Modul Bluetooth HC-05
5. fire mama-mama
6. rezistente diferite valori (pentru divizorul de tensiune pentru modul de Bluetooth - 1k Ω ,2k Ω)
7. Modul cu Buzzer activ

OBSERVATIE Lista nu este in stadiul final.

Schema Electrica:



Probleme aparute si schimbare de planuri:

1. am eliminat partea cu cardul sd si am inlocuit cu note pentru fiecare culoare deoarece cardul sd imi seta si pinii aferenti pentru led-ul rgb (amandoua aveau nevoie de PWM si lucrau pe aceleasi registre) si mi-a fost aproape imposibil sa refac totul, am incercat sa le pun pe canale diferite de la timere dar si initializarile de baza difereau asa ca am decis sa refac aceasta parte cu functionalitatea pe care am zis-o mai devreme. Schema actuala este cea finala.

Software Design

- Sistem de operare: Windows
- Editare cod: Sublime
- Aplicatia android: Android Studio
- Biblioteci AVR și comunicarea cu placa: WinAVR Toolkit
- Încărcarea programului: HID Boot Flash (GUI version)
- Creare schemă electrică: Autodesk Eagle 9.0.0
- Creare schemă bloc: <https://www.draw.io/>

Librarii folosite:

pentru comunicatia pe seriala: biblioteka usart.h din cadrul laboratorului

Se genereaza o intrerupe la primirea unui caracter pe interfata seriala USART. Cand primesc caracterul final al unei comenzii (?),tratez comanda. Daca se primește o comanda se realizeaza lucrurile corespunzatoare acesteia:

Se primesc aceleasi comenzi in spate si pentru partea text/select si pentru voice control

- rgbRRRGGG BBB?

```
seteaza registrele folosite de led :0C0A , 0C0B si 0C2A cu valorile  
primite pentru a reproduce culoarea respectiva  
daca variabila de music_activ este activa(= 1) => se produce si nota  
asociata culorii respective
```

- music/not_music

```
seteaza variabila de music_activ = 1/0;
```

- on_lamp/off

```
seteaza variabila de on = 1/0;  
cand on == 0; lampa nu este aprinsa;
```

- wake_up/not_wake_up

```
seteaza variabila wake_up_mode = 1/0;
```

- sleep/not_sleep

```
seteaza variabila sleep_mode = 1/0;
```

- read/not_read

```
seteaza variabila read_mode = 1/0;
```

- work/not_work

```
seteaza variabila work_mode = 1/0;
```

- default/not_default

```
seteaza variabila default_mode = 1/0;
```

Implementarea primirii de comenzi se face folosind intreruperea de primire a USART0 de pe ATmega324 iar aprinderea LED-urilor se face folosind timerele t0 si t2 (pe 8 biti), pe canalele OC0A, OC0B si OC2A

Pentru a creea o nota aferenta unei culori, ii calculez frecventa in functie de valoarea RGB a culori si apoi pentru a reda pe buzzer alternez bitul PA0, corespunzator pin-ului I/O buzzer de un numar de ori.

Pentru fiecare mod specificat mai sus, parcurg cu un range de valori ([x,y] x >=0, y <= 255, valorile difera in functie de mod) pentru fiecare culoare a led-ului RGB si setez in mod corespunzator, de asemenea creez si nota aferenta.

Aplicatia android: Exista 7 switch-uri:

1. obiectul PickerColor
2. switch pentru on/off lamp
3. switch pentru music- culorile au si note muzicale asociate
4. switch pentru modul work
5. switch pentru modul read
6. switch pentru modul sleep
7. switch pentru modul default
8. switch pentru modul wake-up

Folosind Listener pentru fiecare switch(OnCheckedChangeListener()) si pentru obiectul pickerColor (OnColorChangeListener()), trimit mesaje prin Bluetooth

Pentru partea de Voice Recognition, am urmat tutorialul de la [android-voice-recognition-tutorial-388](#), eliminand obiectele care nu erau necesare pentru aplicatia mea(lista de sugestii, de exemplu). Cand activitatea respectiva intoarce un rezultat corect, se verifica comanda, am verificat si variatii ale cuvintelor(pentru read exista si Read, si tot felul de genul acesta), si pentru fiecare comanda daca este cazul se activeaza si switch-ul corespunzator. Am considerat sa fac acest lucru din doua motive : 1)vizual , 2)daca dupa vrei sa dezactivezi un mod sa poti sa faci asta lucru si direct din aplicatie fara sa spui o comanda daca nu vrei.

Rezultate Obținute

- demo :

[cEBeyD1_eEw](#)

[FaHw9K5SX7A](#)

[zg9ygEDaoBw](#)

6DkCEKUHvS

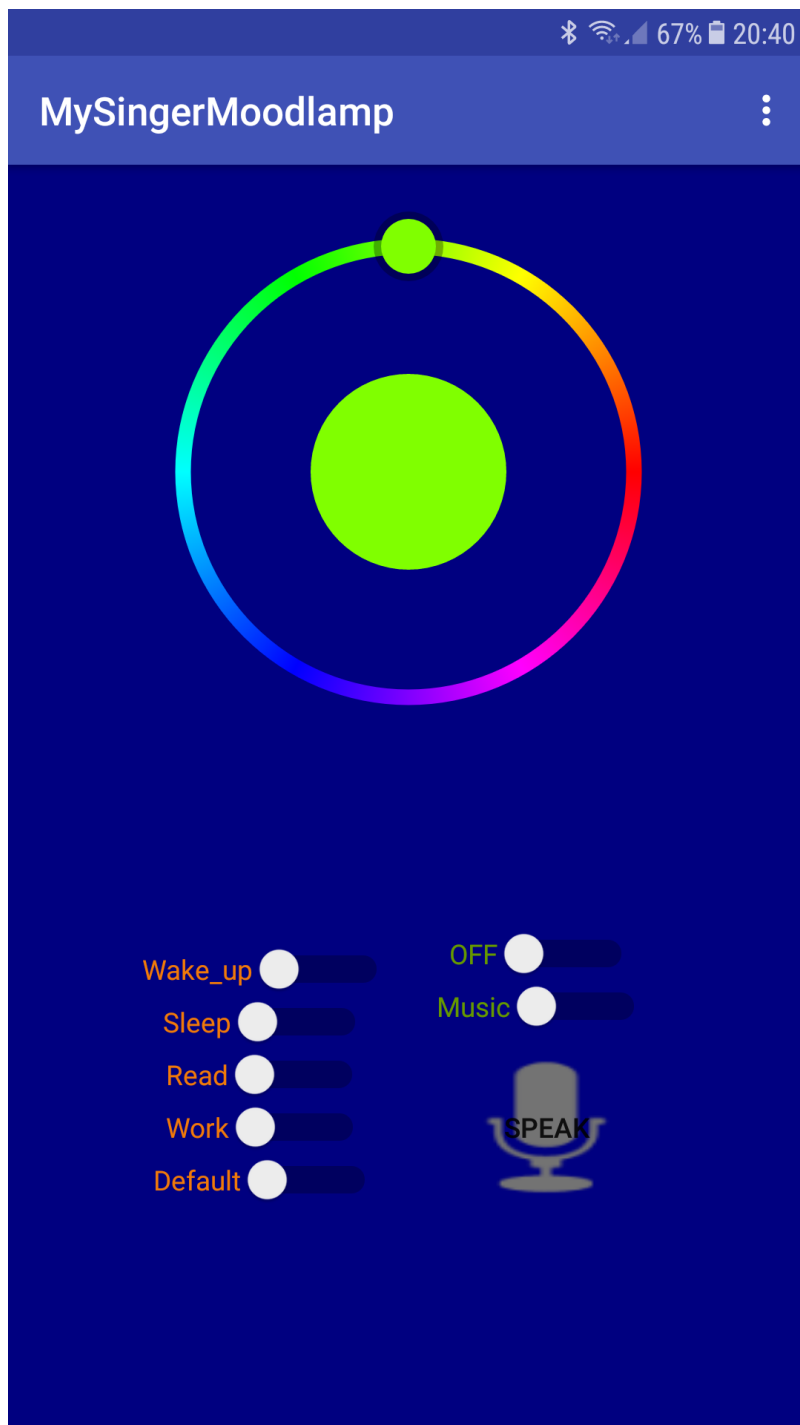
-cablaj:



-poze:



Aplicatia android:



Concluzii

Proiectul a fost unul destul de captivant, dar am intampinat multe probleme pe parcurs. Am stricat placuta de baza la inceput, am refacut-o (asa am devenit destul de buna la lipit). Apoi Modul de Bluetooth care nu avea prea multe valori comune cu ATmega324, iar acest lucru m-a incurcat putin la inceput(si inca mi se pare ca uneori comenzile se primesc cu un anumit delay). Apoi a trebui sa modific schema initiala deoarece nu am mai putut folosi card-ul sd cu toate functionalitatile deoarece el luca pe aceleasi registre ca modul de led si a trebui sa refac toata implementarea, de data asta cu note asociate culorilor. Totusi, sunt destul de incantata de ce am realizat pana aici, dorind ca pana la PM Fair sa mai modific cateva lucruri- atat in aplicatia de android cat si in cod, poate am timp sa adaug si functionalitatea de voice recognition pe care mai vroiam sa o implementez la inceput. . Am reusit sa realizez si partea de Voice Recognition(24 Mai), deci proiectul pare a fii in intregime functional, cu toate lucurile pe care le-am prezentat la inceput(bineinteles, cu mentiune de note pentru culori si nu melodiile de cardul sd).Ca si concluzie, desi m-am stresat destul de mult cu proiectul, s-a dovedit mai usor de gestionat si de implemetat decat am crezut. 

Download

Schema electrica: - formata din 2 pagini - prima este = pm2018.sch - schema oficiala a placutei; a doua este schema mea

[pm2018_mylamp_amza_elena_violeta.sch](#)

- aplicatia android:

[aplicatie_android_amza_elena_violeta.zip](#)

- proiect:

[amza_elena_violeta_proiect_pm_lamp.zip](#)

Jurnal

22.04.2018: Stabilire proiect

22.04.2018 → 04.05.2018: lipirea componentelor de placa de baza

04.05.2018: Finalizarea plăcii de bază

06.05.2018: Schema electrica

17.05.2018 → 20.05.2018: Familiarizare Android + setari pe led

21.05.2018 → 22.05.2018: Implementarea software-ului + schimbarea functionalitati muzica (card SD) cu note pentru fiecare culoare + schimbare schema electrica

23.05.2018: Actualizarea paginii de Wiki cu toate informațiile despre proiect până în momentul curent

24.05.2018:

1. Scriere modulara a codului
2. Partea de Voice Control a aplicatie
3. Actualizare si finalizare pagina de wiki

Bibliografie/Resurse

1. Wiki PM
2. Proiectele din anii trecuți - Mood Lamp + proiectele cu intrumente muzicale [Proiecte PM 2016](#)
3. Detalii de legare si folosire pentru modul de bluetooth
[153-modul-bluetooth-master-slave-hc-05-cu-adaptor.html](#)
4. Detalii de legare si folosire pentru modul led rgb: [737-modul-cu-led-rgb.html](#)
5. Detalii de legare si folosire pentru buzzer: [10-modul-cu-buzzer-activ.html](#)
6. [Laboratorul 1: USART, LCD](#)
7. [Laboratorul 3: Timere, Pulse Width Modulation \(PWM\)](#)
8. obiectul ColorPiker pentru android [HoloColorPicker](#)
9. [android-voice-recognition-tutorial-388](#) pentru partea de voice recognition pentru android

* Documentația în format [PDF](#)

From:

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

Permanent link:

http://ocw.cs.pub.ro/courses/pm/prj2018/adraghici/elena_violeta.amza

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

