

Maria-Luiza ȘERBAN (67340) - Temperature Mood Lamp

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

Introducere

Proiectul consta in realizarea unui model de *mood lamp* avansat, ce va avea urmatoarele functionalitati:

- In timpul functionarii, temperatura e afisata pe ecran;
- In timpul functionarii, cu o bataie din palme, poate fi selectat un mod culoare (se lumineaza becul cu o culoare din cele disponibile) sau modul temperatura (se lumineaza becul cu o culoare in functie de temperatura).

Am vrut sa creez ceva ce are un aspect placut, dar este si util in acelasi timp. De cativa ani imi doresc sa imi achizitionez o lampa de tipul mood lamp, dar nu am gasit un model care sa imi placa foarte mult, asa ca am decis sa construiesc eu unul. De asemenea, mi-a placut proiectul cu statia meteo construit la AD in anul al II-lea de facultate, si m-am gandit ca ar fi util sa integrez masurarea temperaturii si a umiditatii in cadrul proiectului meu.

Descriere generală

Schema bloc:



Mod de functionare:

La identificarea unui sunet scurt (ex: o bataie din palme), receptat de catre microfon, se va aprinde becul in primul mode de functionare. Cat timp becul este aprins:

- temperatura va fi afisata pe ecran;
- becul va sta aprins in modul de functionare selectat;
- la identificarea unui sunet, se va schimba modul de functionare, fiind notat pe ecran.

Hardware Design

Lista de piese:

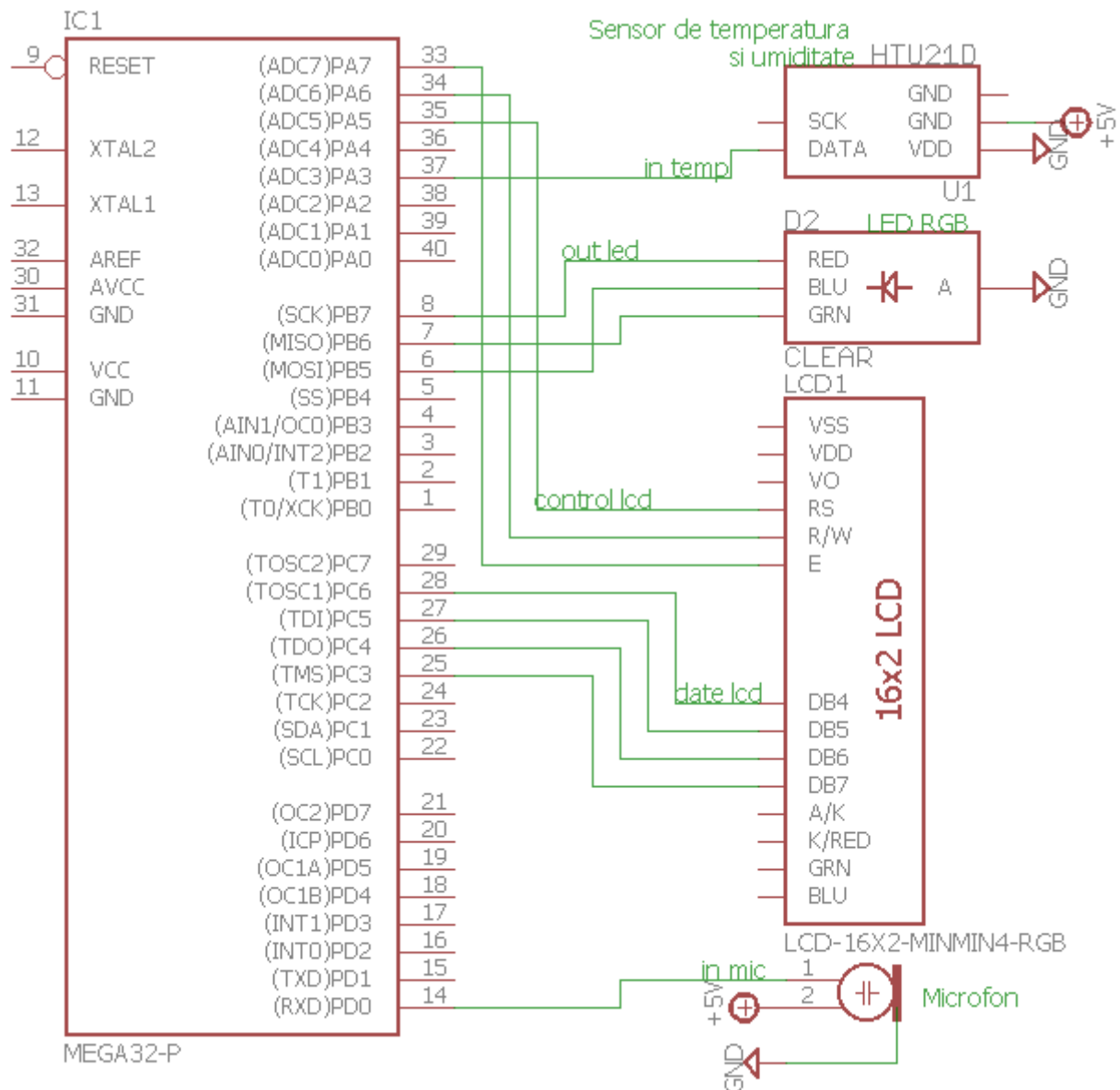
- placa de baza;

- componente pentru placa de baza;
- rezistente;
- fire mama-mama si mama-tata;
- header pentru lcd;
- modul LCD 1602 cu backlight albastru de 5 V;
- modul senzor sunet;
- senzor de temperatura brick;
- led RGB.

Componente extra folosite:



Schema electrica:



Conexiuni:

- PORTA → senzor temperatura: PA3 → OUT si control LCD: PA5, PA6, PA7;
- PORTB → modul led RGB: PB5 → B, PB6 → G, PB7 → R;
- PORTC → date LCD: PC3, PC4, PC5, PC6;
- PORTD → modul sunet: PD0 → OUT.

Software Design

Mediul de dezvoltare:

- Programmers Notepad [WinAvr]
- BootloaderHID

Am pornit dezvoltarea de la **laboratorul 1** de PM, folosind functiile implementate pentru LCD.

Am organizat dezvoltarea codului in **4 etape**:

- simplu: se afiseaza un text pe LCD, alcatuit dintr-o linie a temperaturii (ce va fi identificata la etapa 3) si o linie cu modul de functionare;
- complex: am adaugat functionalitatea de schimbare a culorii becului la identificarea unui sunet;
- complet: am adaugat functionalitatea de schimbare a culorii becului in albastru la temperaturi sub 20 de grade Celsius si in rosu la temperaturi mai mari, cand modul temperatura este selectat;
- int: am adaugat intrerupere pentru citirea temperaturii la intervale regulate de timp, am adaugat un mesaj de start si am reorganizat codul.

Cele 4 etape de dezvoltare pot fi analizate la sectiunea **Download**.

Algoritmul:



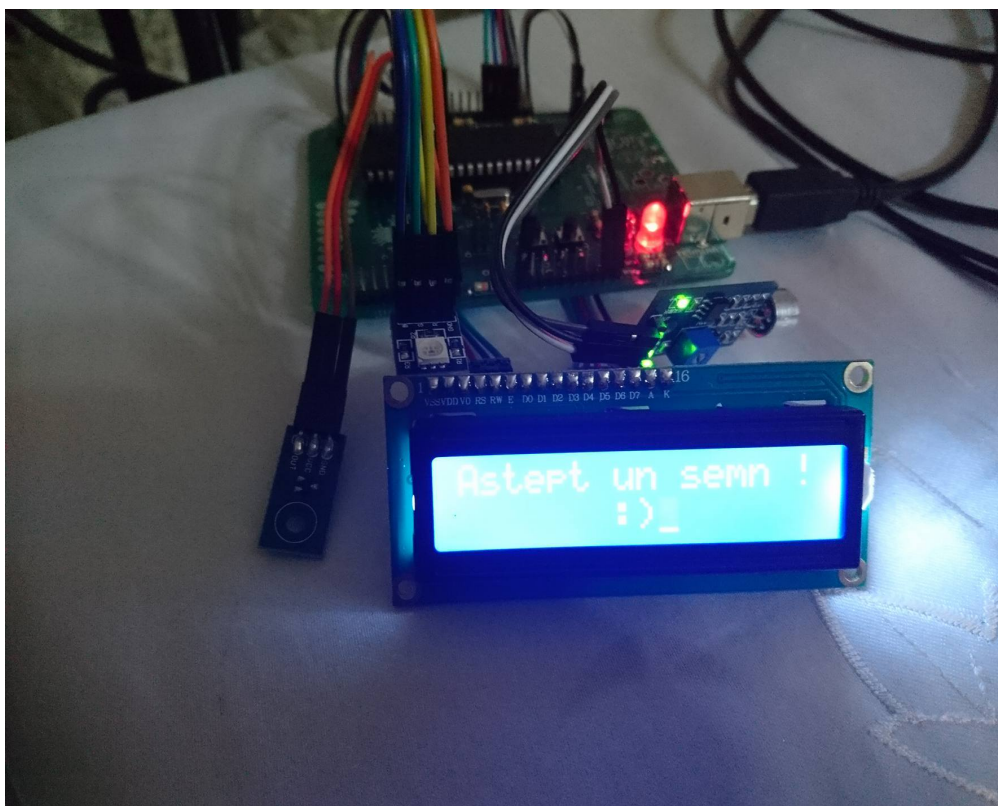
Temperatura este citita la cateva secunde prin intermediul intreruperii scrise.

Surse:

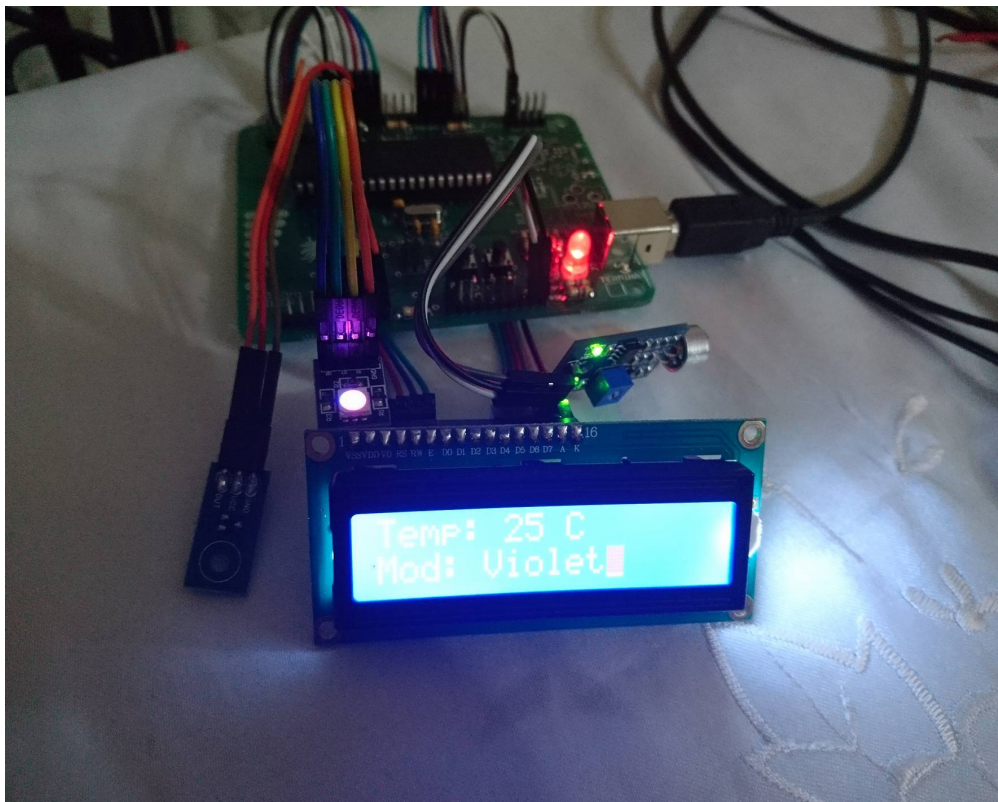
- lab1.c > implementarea algoritmului si a functiilor de interactiune cu senzorii;
- lcd.c > implementarea functiilor specifice lcd-ului.

Rezultate Obținute

Mesaj de start:



Functionare in mod Violet:



Alte moduri de functionare disponibile sunt: Rosu, Portocaliu, Verde, Albastru, Temperatura.

Clip **video** de prezentare: <https://youtu.be/TghUhcJM6BM>

Concluzii

Am reusit sa implementez ideea initiala si sa utilizez corect diferite datasheet-uri. Pentru scrierea codului, am urmat pasii mentionati la Software Design pentru a putea testa, pe rand, buna functionare a fiecarei componente extra. Pentru mine, cea mai mare provocare a fost reprezentata de alegerea pinilor si conectarea componentelor la acestia, avand in vedere utilizarea corecta a lor in cod.

Download

[Arhiva cu surse, impartite in cele 4 etape de dezvoltare, si scheme:](#)

maria_luiza.serban_pm.zip

Jurnal

- 4 mai: am lipit placa de baza;
- 11 mai: am primit componentele extra;
- 17 mai: am scris varianta simpla a codului;
- 18 mai: am lipit headere si rezistente pentru conexiunea lcd-ului;
- 19 mai: am scris varianta complexa a codului;
- 21 mai: am scris varianta complexa a codului;

- 23 mai: am scris varianta cu intreruperi a proiectului;
- 24 mai: am realizat ultimele teste si verificari, completand documentatia.

Bibliografie/Resurse

Resurse **software**:

- folosire lcd: <http://cs.curs.pub.ro/wiki/pm/lab/lab1>;
- folosire convertor analogic-digital si senzor temperatura: <http://cs.curs.pub.ro/wiki/pm/lab/lab6>,
https://www.robofun.ro/senzor-temperatura-brick?search=temperatura&sub_category=true;
- folosire intreruperi: <http://cs.curs.pub.ro/wiki/pm/lab/lab2>,
http://cs.curs.pub.ro/wiki/pm/_media/prezentare_proiect_termometru.pdf.

Resurse **hardware**:

- ATmega324 [atmega324a.pdf](#);
- modul senzor sunet [sound.pdf](#);
- senzor temperatura [temp.pdf](#);
- lcd [lcd.pdf](#).

Documentația în format [PDF](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2017/amocanu/101>



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