

Lampa 3 in 1

- Popescu Maria-Teodora,
- 332CC

Introducere

Ce face?

Proiectul constă în implementarea unei **lămpi de birou** care să **afișeze** nu numai **data și ora**, dar și **temperatura și umiditatea** dintr-o încăpere.

Care este scopul lui?

La o scurtă privire, s-ar putea părea că dezvoltarea unei lămpi de birou este o idee simplă. În ciuda acestui fapt, este un amalgam de funcționalități care nu sunt folosite doar pentru a **ilumina o încăpere**, ci și un **reper pentru a ajusta confortul personal**. Acest „obiect” merge pe vechea idee că o lampă simplă de birou poate fi pusă la un alt nivel de funcționare, nu numai pe funcția primară de bază cu care este asociată.

Care a fost ideea de la care am pornit?

Ideea ce a stat la baza acestui „obiect” a fost **nevoia de a aduce o îmbunătățire în sfera utilității cotidiene** a unei lămpi simple de birou, dat fiind faptul că odată cu avansul în tehnologie, lucrurile uzuale, simple, au evoluat dincolo de simpla lor utilitate de bază.

Așadar, esența acestui sistem modest constă în faptul că nu poate face doar iluminare, ci și oferă informații esențiale despre mediul în care este amplasat. Temperatura și umiditatea camerei sunt factori determinanți pentru confortul și eficiența numeroaselor activități zilnice.

De ce crezi că este util și pentru alții, dar și pentru tine?

După părerea mea, acest dispozitiv nu este doar un lucru obișnuit, ci însoțitorul meu permanent. Prin urmare, **urmăresc cea mai înaltă performanță** posibilă și bunăstare zi de zi. În ceea ce privește relevanța sa pentru diferiți oameni, de obicei, lămpile de birou sunt dispuse unde se pune accent pe lucru manual și cognitiv, nu numai în cazul elevilor și studenților, ci și al tuturor persoanelor care au nevoie de lumină. Acest sistem, însă, oferă o gamă variată de beneficii tangibile, cum ar fi data și ora și temperatura și umiditatea, **promovând confortul spațiului de lucru și acasă**.

Descriere generală

Proiectul este compus din două părți: **hardware și software**.

Partea de **software** este reprezentată de întreg procesul de comunicare ar perifericelor cu microcontroller-ul, dar și partea de achiziții de date de la senzori și afișarea măsurilor pe ecran.

Partea de **hardware**, în schimb, constă în piesele interconectate între ele.

Astfel, o schemă de principiu ar fi următoarea:



Ecranul are 2 stări de afișare, cea de dată și oră, și cea de temperatură și umiditate.

Prima stare:

Data și ora sunt afișate pe rânduri separate, pe primul rând fiind data, în format DD/MM/YYYY, iar pe al doilea - ora, în format HH:MM, 24h.

A doua stare:

Temperatura și umiditatea sunt detectate prin intermediul senzorului și al termistorului. Semnalul generat de termistor va fi direcționat către convertorul analog-digital al plăcuței(ADC). Display-ul LCD de 16×2 caractere cu I2C încorporat, va afișa în timp real rezultatul pe două rânduri: pe primul - temperatura în grade Celsius de la senzor și de la termistor (pentru comparație), iar pe al doilea - umiditatea în procente.

Butoanele:

- unul care schimbă afișajul ecranului, adică din modul ceas+data îl face în modul temperatură și umiditate → BUTON DATA-ORA/TEMPERATURA-UMIDITATE,
- unul care aprinde și stinge lumina/led-ul/led-bar-ul → BUTON LUMINA,
- unul care schimbă modul de setare al datei și al orei → BUTON MOD,
- unul care incrementează data/ora → BUTON +,
- unul care decrementează data/ora → BUTON -.

Hardware Design

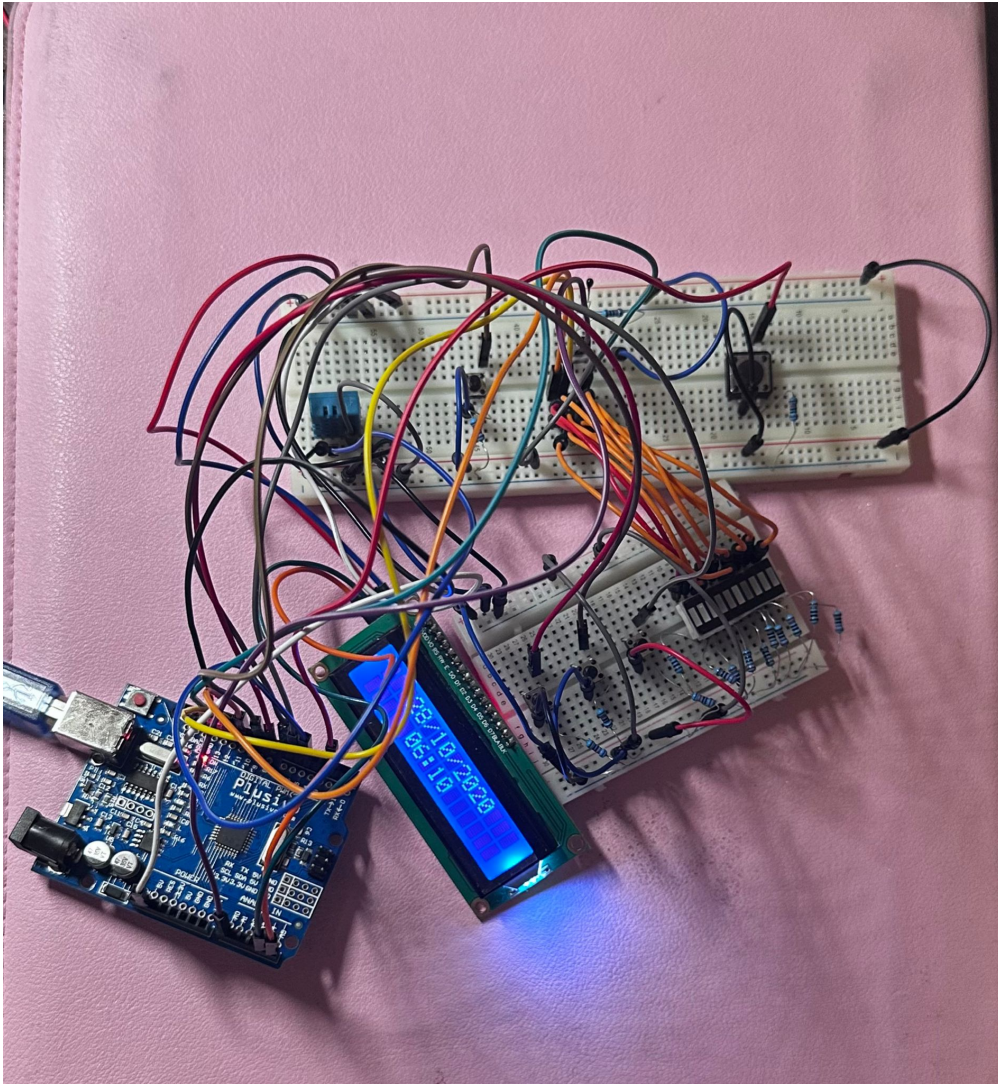
Listă piese:

- plăcuță arduino uno(ATmega328p),
- senzor temperatură și umiditate DHT-11,
- termistor ntc,
- ecran LCD 16×2 I2C,
- 5 mini butoane push,
- fire mama-tata, tata-tata, cu dimensiuni între 20-40cm, multicolore + jumper,
- rezistențe 220R, 1k și 10k,
- baterie/sursă USB PC,
- 10 segment led bar sparkfun.

Scheme electrice:



Proiectul fizic



Software Design

Biblioteci folosite:

- Arduino.h - am lucrat în PlatformIO VSCode în C, dar cu ajutorul bibliotecii Arduino
- Wire.h pentru - I2C
- "LiquidCrystal_I2C.h" - tot pentru ecran, I2C
- EEPROM.h - pentru a stoca data și ora
- Adafruit_Sensor.h - pentru senzorul DHT-11

Pentru **început**, am conectat **led bar-ul** cu rezistențe de 220 la pinii 13, 12, 11 și cele cinci butoane la plăcuță, folosind rezistențe de pull-up de 1K. Inițial, am programat numai **butonul de pe pinul 10** (inițial pe acest pin, căci în final l-am pus pe pinul 2 pentru a face cu întreruperi), cel care aprinde și stinge led bar-ul. Așa am testat aprinderea și stingerea unui led, folosirea unui buton.

Apoi am conectat **ecranul** și am testat să văd dacă funcționează să afișez ceva, motiv pentru care a mai trebuit să modific intensitatea curentului din potențiometrul său propriu. Desigur că am testat și biblioteca de I2C a ecranului.

Cum deja funcționa un buton, led bar-ul și ecranul și mai aveam de testat 4 butoane, m-am gândit să implementez logica din spatele acestora.

Butonul de pe pinul 9 schimbă modul de afișare al ecranului, care inițial afișa sugestiv DD/MM/YYYY

cu HH:MM sau T: cu H: pentru măsurătorile următoare.

Butonul de pe pinul 8 - modul de modificare al datei și al orei - de aceea se va face MODULO 5, fiind 5 posibilități:

- modificare zi - %5 == 0
- modificare lună - %5 == 1
- modificare an - %5 == 2
- modificare oră - %5 == 3
- modificare minut - %5 == 4

Butoanele de pe pinii 7 și 6 se folosesc pentru a incrementa, respectiv a decrementa, valorile conform mode-ului butonului de pe pinul 8. Aici am avut grijă ca incrementarea și decrementarea să se realizeze conform ordinii firești a naturii/calendarului, astfel:

- dacă este an bisect, luna a doua are 29 de zile
- la decrementarea zilei, dacă se ajunge la 0 și este lună cu 31 de zile, automat se trece la luna precedentă, cu ziua 30 (similar pentru lunile cu 30 de zile)
- incrementarea zilei se face similar, dacă este lună cu 30 de zile și ziua este 30, când se incrementează trece la luna următoare, ziua 1, similar și la lunile cu 31 de zile
- incrementarea lunii, indiferent de zi, dacă luna este a 12-a și se incrementează, se mărește cu o unitate anul (similar dacă se decrementează, din luna 12 dacă se decrementează se ajunge la luna 1, anul precedent)
- incrementarea minutelor - din 59 minute se adaugă încă o oră și minutele se resetează (similar la decrementare, din 00 se scade o oră și minutele devin 59)
- incrementarea orei - indiferent de minute, dacă ora este 23 și se incrementează, se trece la ziua următoare, iar acolo se ține cont în continuare de logica incrementării zilelor (similar și la decrementare, din 00, se trece la ziua precedentă)

Pentru a înțelege mai bine logica, a se vedea videoclipul de pe youtube din link-ul de mai jos, din secțiunea **Rezultate obținute**

În plus, cum este ceas, **la fiecare 60 secunde, se incrementează minutele** - timpul trece/se măsoară trecerea timpului

Desigur că, **orice modificare** în trecerea timpului și în setarea datei și a orei, acestea (data și ora) **sunt păstrate/actualizate în memoria EEPROM** și luate/încărcate de acolo la setup().

Pe urmă, am adăugat **senzorul DHT-11** și am afișat inițial măsurile în Serial, apoi și pe ecran, conform stării butonului de pe pinul 9.

Pentru că aveam nevoie de 3 laboratoare, am încercat să găsesc încă o metodă de a calcula măsura temperaturii, cu ajutorul unui **termistor ntc** la care am folosit ADC și câteva calcule matematice.

Astfel, afișez în paralel temperatura din senzor cu temperatura din termistor (mai întâi, în partea stângă este temperatura din DHT, urmată lângă/în dreapta de cea din termistor).

La definitivarea codului, am trecut la **implementarea întreruperilor**, și am făcut aceasta pentru butonul care aprinde și stinge led bar-ul. Pentru a reuși, a trebuit să schimb pinul său (inițial 10) pe pinul 2 și am folosit ISR și funcție handler de întrerupere.

Rezultate Obținute

- Pentru întreruperi, funcționează numai pinii INT, nu și pinii PCINT.
- Se pare că la întreruperi nu este necesar ca niciun buton apăsat să nu facă nimic, întreruperile să fie independente/să nu interfereze una cu cealaltă (+ dacă nu sunt și alți pini de întrerupere folosiți).
- Valoarea de la termistor se schimbă repede și brusc, dată fiind sensibilitatea sporită a acestui dispozitiv.


Folosesc 3 tehnologii/protocoale/laboratoare:

- termistorul ntc cu ADC
 - ecranul LCD cu I2C
 - butonul care stinge și aprinde lumina cu întreruperi
-
- Nu este user-friendly
 - Se văd firele
 - Nu are carcasă
 - Nu este aesthetic
 - Nu are lucru cu registre / nu este full C
 - lumina de la led-uri nu este puternica sa fie folosita chiar ca o lampa
 - ESTE FUNCȚIONAL

Demo youtube

<https://youtu.be/1dynAAyy78c?si=CjgWNPrbNHvFxq0D>

Concluzii

Înainte de acest proiect, am avut 0 contact cu arduino și programare pe C embedded, iar în urma realizării acestuia, am învățat multe prin hand-on experience și am realizat că nu este atât de complicat pe cât credeam + ușor de realizat, repede de făcut/implementat. Utilitatea practică a proiectului a făcut lucrurile și mai interesante și plăcute, iar conceptul de a programa de la zero piese conectate între ele, m-a făcut să mă simt ca un copil mic ce învață prin joacă .

Download

Numai cod, imaginile sunt la secțiunea de Hardware, la data-sheet-uri sunt link-uri mai jos, mai sus este demo pe YouTube + jos este jurnal cu fiecare pas realizat

[Tot proiectul in cod](#)

Jurnal

- **26 aprilie:** comanda piese + completare tabel Excel
- **4 mai:** creare pagină OCW + finalizare documentație Milestone 1, crearea primei versiuni a schemei electrice
- **9 mai:** primire piese și prima conectare de fire (butoane + led-bar).
- **16 mai:** conectarea tuturor pieselor și poză circuit inițial pentru Milestone 2
- **17 mai:** programarea butoanelor de pe pinii 10 și 9
- **18 mai:** programarea celorlalte butoane + adăugare termistor
- **20 mai:** comandă noi piese (fire, breadboard, butoane suplimentare, de orice eventualitate)
- **22 mai:** primire comandă, adăugare întreruperi în cod + definitivare cod + definitivare documentație cu secțiunile Software, Rezultate, Concluzii + realizare demo youtube - Finalizare Milestone 3

Bibliografie/Resurse

- ATmega328P (pe arduino): http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf
- Plăcuța de dezvoltare arduino: <https://store.arduino.cc/products/arduino-uno-rev3>
- DHT-11: <https://pdf1.alldatasheet.com/datasheet-pdf/download/1440068/ETC/DHT11.html>
- Ecran LCD1602: <https://pdf1.alldatasheet.com/datasheet-pdf/download/89725/ETC/SC1602A.html>

[Export to PDF](#)

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

Permanent link:
<http://ocw.cs.pub.ro/courses/pm/prj2024/alucaci/maria.popescu2812>



Last update: **2024/05/22 23:35**