

Daily

Autor

[Gabriela-Camelia Mircia](#)

Grupa: 332CA

Introducere

- Proiectul consta in implementarea unui ceas care are mai multe functionalitati precum: afisarea datei si a orei, afisarea zilei saptamanii, posibilitatea de a seta alarme, vizualizarea temperaturii si a umiditatii. Este usor de folosit prin butoane si o aplicatie simpla, dar si usor de vizualizat pe un ecran.
- Scopul proiectului este afisarea informatiilor utile pentru oricine dintr-o zi, de aici si numele.

Descriere generală

Se conecteaza la o placuta de Arduino UNO un modul RTC pentru afisarea datei si orei. Acestea pot fi modificate prin butoanele de pe breadboard. Utilizatorul are posibilitatea de a vizualiza ora fie in format AM, fie PM. Toate informatiile detaliate sunt afisate pe un LCD controlat cu ajutorul unui potentiometru.

La nivelul de temperatura si umiditate este folosit un senzor pentru preluarea datelor, acestea sunt interpretate folosind doua LED-uri: cel rosu se va aprinde daca temperatura depaseste o anumita limita (cea setata este de 24C), cel albastru va porni daca temperatura scade sub limita impusa.

Pentru a face mai interactiv proiectul am adaugat un senzor de sunet care determina pornirea si oprirea ceasului. Acesta recunoaste sunete de frecvente asemanatoare unei batai din palme. Spre exemplu, daca e folosit intr-o camera unde exista un dialog continuu, ceasul se va reporni permanent.

Spre final, pentru a complica putin setarea alarmei, a fost introdus un modul bluetooth care functioneaza impreuna cu o aplicatie simpla pentru mobile. Astfel utilizatorul poate seta alarma chiar din telefon. La acel moment buzzerul va suna si cele doua LED-uri vor lumina intermitent. Pentru adaptabilitate si pentru a nu depinde de un dispozitiv, a fost conectata o baterie de 9V.



Hardware Design

Componente

Piesa	Nr	Link	Datasheet/Utils
Arduino Uno	1	link	link
Modul RTC DS3231SN	2	link	link
Senzor sunet CZN-15E	3	link	link
Ecran LCD 1602A	4	link	link
Senzor temperatura AM2302	5	link	link
Modul bluetooth ZS-040	6	link	link
Potentiometru	7	link	link
Buzzer pasiv	8	link	link
Rezistente 1K Ω	9	link	link
LED-uri monocolare	10	link	link
Butoane	11	link	link
Breadboard	12	link	link
Fire tata-tata	13	link	link
Fire mama-tata	14	link	link
Baterie 9V	15	link	
Conector baterie 9V	16	link	

Schema electrica



Schema a fost realizata in Eagle folosind informatiile din laboratoarele de IC. Pentru a nu aglomera firele in schematic am folosit functia Mirror, am repositionat elementele si am folosit labeluri. Mentionez ca un pas greu aici a fost descarcarea a diferite biblioteci pentru inserarea componentelor potrivite. Voi include in arhiva de download si aceasta schema.

Observatii

Cel mai bine conectarea se observa in schema electrica. Pentru fiecare senzor am consultat datasheetul si diverse tutoriale, fiind un spatiu de lucru nou.

O dificultate hardware o consider asezarea componentelor. Pentru a incapsula intreg proiectul am folosit o cutie de carton alaturi de mici bucati de carton care mentin o stabilitate a pieselor. Am incercat sa folosesc fire de dimensiune potrivita fara a aglomera breadboardul. Am tinut cont de culori

in special la VCC si GND.

O alta problema intalnita a fost defectarea unui pin Arduino. Credeam ca este defect senzorul de temperatura, dar s-a dovedit a fi pinul.

Testarea componentelor s-a facut individual. Schema initiala nu includea si modulul bluetooth, astfel pentru testarea acestuia am folosit alta placuta de Arduino.

Alegerea celor 2 LED-uri monoculare a venit din cauza faptului ca nu am avut suficiente pini, mai aveam ramas 2, dar pentru un RGB erau necesari 3. Am realizat ca o rezolvare a acestei probleme ar fi putut fi folosirea unui LCD cu I2C care nu necesita si un potentiometru, deci numarul pinilor ocupati ar fi scazut.

Software Design

- Limbaj: C
- Tehnologie: Arduino IDE Windows, diagrams.net

Software-ul se imparte in mai multe parti pentru fiecare functie. Majoritatea codului consta in notiuni de baza in C la care se adauga initializari si interpretarea datelor de pe placuta de Arduino. Codul contine comentarii care lamuresc deciziile luate.

Bibliotecile folosite au fost incluse in arhiva. Acestea depind de senzorii utilizati.

Exemple: DHT_sensor_library, DS3231, LiquidCrystal

before

- initializarea variabilelor folosite pe parcurs

setup()

- pornirea modulelor folosite: LCD, RTC, Serial, Bluetooth

buttoninfo(..)

- functie care mentine contorizarea apasarii butonului
- asigura ca se inregistreaza o singura apasare

loop()

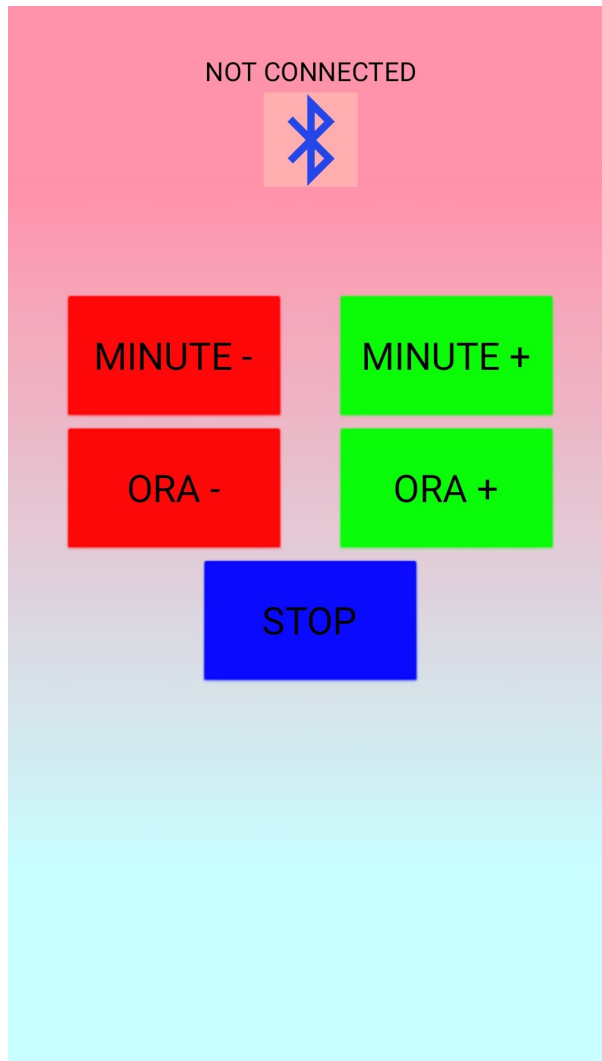
- citirea datelor transmise de senzori
- verificarea apasarii butoanelor
- logica de afisare principala se bazeaza pe senzorul de sunet, declansarea acestuia modifica afisajul de pe ecran: "on-off"-ul ceasului
- logica "paginilor": o apasare a primului buton modifica afisajul de pe LCD, astfel se creeaza un meniu interactiv
- pagini: ora, data, ziua saptamanii, temp/umiditate, alarma
- butonul 2: selectarea datei care se doreste a fi modificata
- butonul 3: permite modificarea valorilor orei/datei
- butonul 4: permite afisarea in format 12/24 ore
- afisarea se face in functie de pozitiile de pe LCD folosind setCursor(..)
- pagina intermediara: apare dupa apasarea butoanelor 2+1, adica dupa o modificare facuta de utilizator
- logica optiune temperatura: in functie de limita impusa, setez pe LOW/HIGH LED-ul de culoarea potrivita

modul Bluetooth

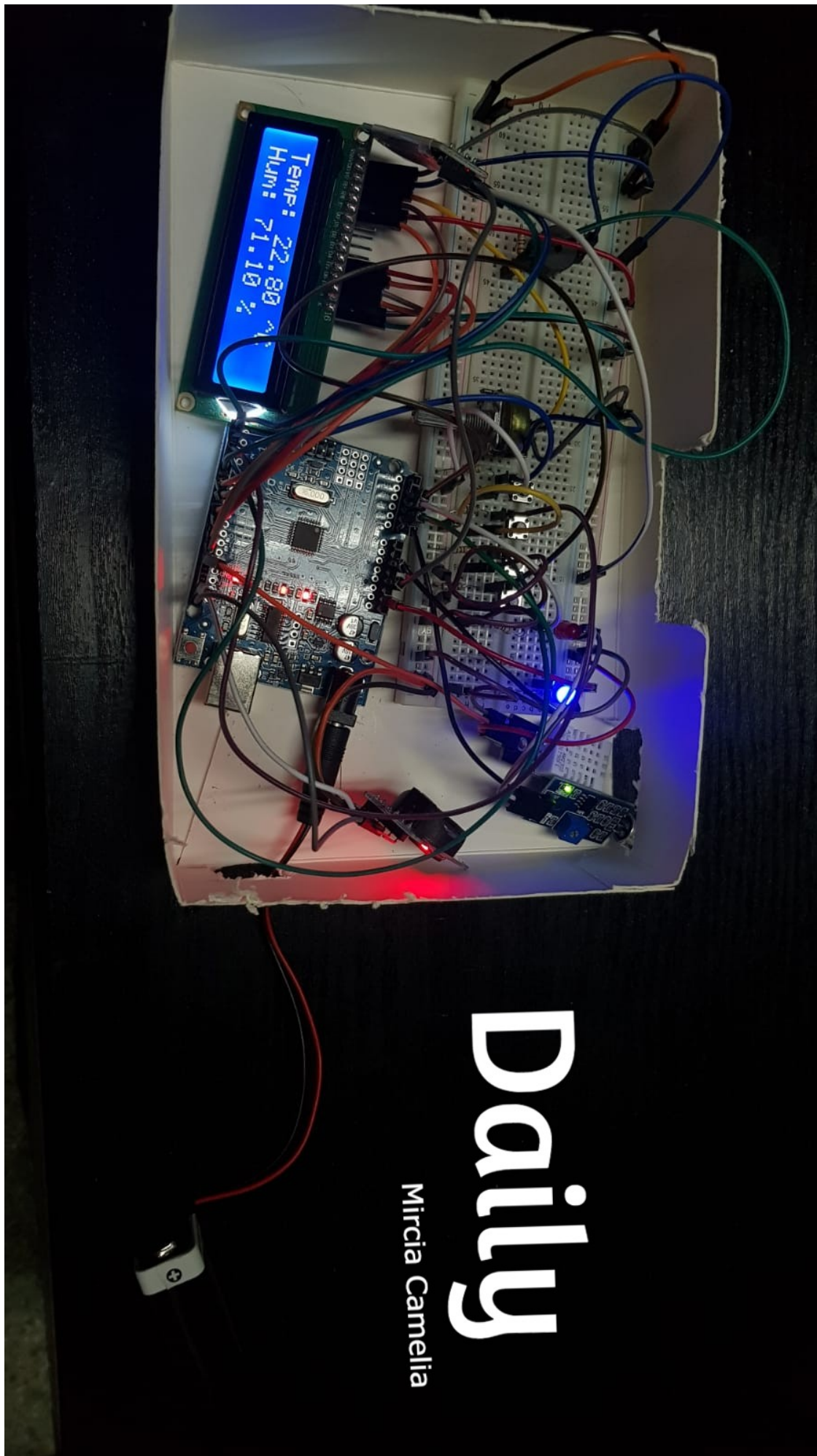
- pentru a interactiona cu acest modul am creat o aplicatie pe Android cu ajutorul [Mit App Inventor](#)
- programul se bazeaza pe blocuri si imagini, de unde mai apoi este generata o aplicatie
- contine 6 butoane: unul pentru conectarea la Bluetooth, unul pentru a opri conectarea, 2 pentru modificarea orei, 2 pentru modificarea minutului
- aplicatia trimite caractere ('a','b','c','d') care sunt interpretate printr-o serie de blocuri conditionate in C, un caracter reprezinta o modificare a unui minut/unei ore

alarma

- o logica simpla: cand timpul dictat de aplicatie corespunde cu timpul ceasului, se pornesc LED-urile si buzzerul
- nu poate functiona in acelasi timp cu optiunea pentru temperatura deoarece nu ar trebui confundata lumina LED-urilor



Rezultate Obținute



Demo:

<https://youtu.be/ILNGMWWlvSg>

Cuprins video:

- filmarea intregii functionalitatii cu toate interactiunile cu utilizatorul
- demonstratie utilizare senzor de sunet prin batai din palme si diferite miscari
- optiunea AM/PM
- observarea schimbarii diferentei de temperatura


Ce a iesit?

In final, proiectul s-a dovedit a fi usor de folosit, toate functionalitatile dorite au putut fi observate. S-au folosit componente cu functii diversificate pentru un scop comun: sa ofere utilizatorului informatii si optiuni utile zilnic printr-o interactiune cat mai accesibila.

Concluzii

Consider ca am recuperat o mare parte din lipsa interactiunii cu partea de hardware din acesti 3 ani de facultate. Din teama de a nu reusi sa conectez o placuta Arduino, am ajuns sa fac un proiect initiat de o idee proprie care mi-a placut mult.

M-a pasionat si partea de organizare, diversitatea programelor folosite si primul contact cu wiki-ul.

“Daily” m-a ajutat sa inteleg mai bine partea de hardware si sa ma familiarizez cu cateva componente care se folosesc in automatele pe care le-am utilizat fiecare dintre noi. A meritat toata munca. 

Download

Contine bibliotecile folosite, imaginile inserate, codul, schema electrica.

[arhiva_cami.zip](#)

Jurnal

- 24 aprilie - Alegere Tema Proiect
- 24 aprilie-10 mai - Documentare, idei, comanda piese
- 24 mai - Testare LCD si senzori
- 25-27 mai - Hardware + Software..
- 28 mai - Upgrade la baterie 9V
- 29 mai - Upgrade la modul bluetooth, Finalizare schema electrica
- 1 iunie - Filmare si finalizare demo
- 3 iunie - Editare pagina wiki

Bibliografie/Resurse

PDF

- <https://create.arduino.cc/projecthub/MisterBotBreak/how-to-use-a-real-time-clock-module-ds3231-bc90fe>
- <https://core-electronics.com.au/tutorials/use-lcd-arduino-uno.html>
- <https://randomnerdtutorials.com/guide-for-microphone-sound-sensor-with-arduino/>
- <https://www.instructables.com/Interfacing-Buzzer-to-Arduino/>
- <https://forum.arduino.cc/t/rgb-led-on-analog-pins-a0-a1-a2/414785>
- <https://www.arduino.cc/en/Reference/Libraries>
- <https://appinventor.mit.edu/>

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2021/agrigore/daily>



Last update: **2021/06/03 05:39**