

# ReMINDer

## Introducere

“ReMINDer” este un dispozitiv ce vine in ajutorul utilizatorilor uituci ce vor sa duca la bun sfarsit toate task-urile, respectand deadline-urile acestora. Cu ajutorul lui “ReMINDer”, utilizatorul poate configura task-urile pe care le are, setand ziua si ora corespunzatoare acestora, si un titlu reprezentativ asociat. De aici, “ReMINDer” va avea responsabilitatea de a semnala utilizatorul cu privinta la task-urile de indeplinit, in zilele si orele care au fost configurate.

Ideea de la care am pornit este aplicatia software “Reminders” de pe iOS, folosind-o in mod frecvent pentru task-urile de la facultate. Insa, uneori, fiind distras de multimea de notificari ale celorlalte aplicatii de pe telefon, nu observ notificari privitoare la task-urile pe care le am de indeplinit. Asa ca m-am gandit ca ar fi mai util in a avea un dispozitiv separat pentru aceasta necesitate de a manageria timpul si task-urile avute.

## Descriere generală

Schema bloc:



Utilizatorul va interactiona cu dispozitivul prin cele 4 butoane care sunt definite astfel: buton de back, un buton de next/confirm, 2 butoane de iterare (+/-) prin data, ora si alfabetul necesar setarii titlului

Cand data si ora sistemului vor face matching cu una din datele si orele unui task configurat de utilizator, sistemul va activa buzzerul, aprinderea ledului si va afisa pe LCD titlul task-ului de indeplinit, pana in momentul in care utilizatorul apasa butonul “confirm”.

In afara momentelor in care sistemul se afla in starea de reminding (buzzerul e activat, ledul e aprins si display-ul afiseaza titlul task-ului), acesta afiseaza data, ora, temperatura si umiditatea (ultimele doua sunt preluate prin intermediul senzorului).

## Hardware Design

Aici puneți tot ce ține de hardware design:

- listă de piese
- scheme electrice (se pot lua și de pe Internet și din datasheet-uri, e.g.)

<http://www.captain.at/electronic-atmega16-mmc-schematic.png>)

- diagrame de semnal
- rezultatele simulării

## Software Design

Descrierea codului aplicației (firmware):


- mediu de dezvoltare (if any) (e.g. AVR Studio, CodeVisionAVR)
- librării și surse 3rd-party (e.g. Procyon AVRlib)
- algoritmi și structuri pe care plănuți să le implementați
- (etapa 3) surse și funcții implementate

## Rezultate Obținute

Care au fost rezultatele obținute în urma realizării proiectului vostru.

## Concluzii

## Download

O arhivă (sau mai multe dacă este cazul) cu fișierele obținute în urma realizării proiectului: surse, scheme, etc. Un fișier README, un ChangeLog, un script de compilare și copiere automată pe uC crează întotdeauna o impresie bună .

Fișierele se încarcă pe wiki folosind facilitatea **Add Images or other files**. Namespace-ul în care se încarcă fișierele este de tipul **:pm:prj20??:c?** sau **:pm:prj20??:c?:nume\_student** (dacă este cazul).  
**Exemplu:** Dumitru Alin, 331CC → **:pm:prj2009:cc:dumitru\_alin**.

## Jurnal

Puteți avea și o secțiune de jurnal în care să poată urmări asistentul de proiect progresul proiectului.

## Bibliografie/Resurse

Listă cu documente, datasheet-uri, resurse Internet folosite, eventual grupate pe **Resurse Software** și **Resurse Hardware**.

[Export to PDF](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2022/reminder>



Last update: **2022/04/22 21:19**