

# Florentina PETCU (66939) - No More 5 Minutes

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

## Introducere

Proiectul consta in realizarea unui ceas cu alarma care "fuge" de utilizator atunci cand acesta incearca sa opreasca alarma.

Acesta va avea toate functionalitatile unui ceas cu alarma: se va putea seta ora curenta, se va putea seta ora alarmei si se va putea porni, amana sau opri alarma. Cand alarma suna si utilizatorul apropie degetul de butonul de amanare/oprire, ceasul va pleca inainte ca utilizatorul sa poata apasa butonul .

Scopul proiectului este sa faca utilizatorul sa se ridice din pat cand suna alarma si sa renunte la cele "inca 5 minute" .

Ideea a pornit de la o problema cu care se confrunta multa lume (printre care si eu): alarme ineficiente. Cu totii stim acele dimineti care ne sunt "stricate" de sunetul alarmei  si suntem tentati sa apasam pe butonul de amanare/oprire alarma si sa dormim in continuare. Proiectul meu are scopul de a combate acest obicei prost.

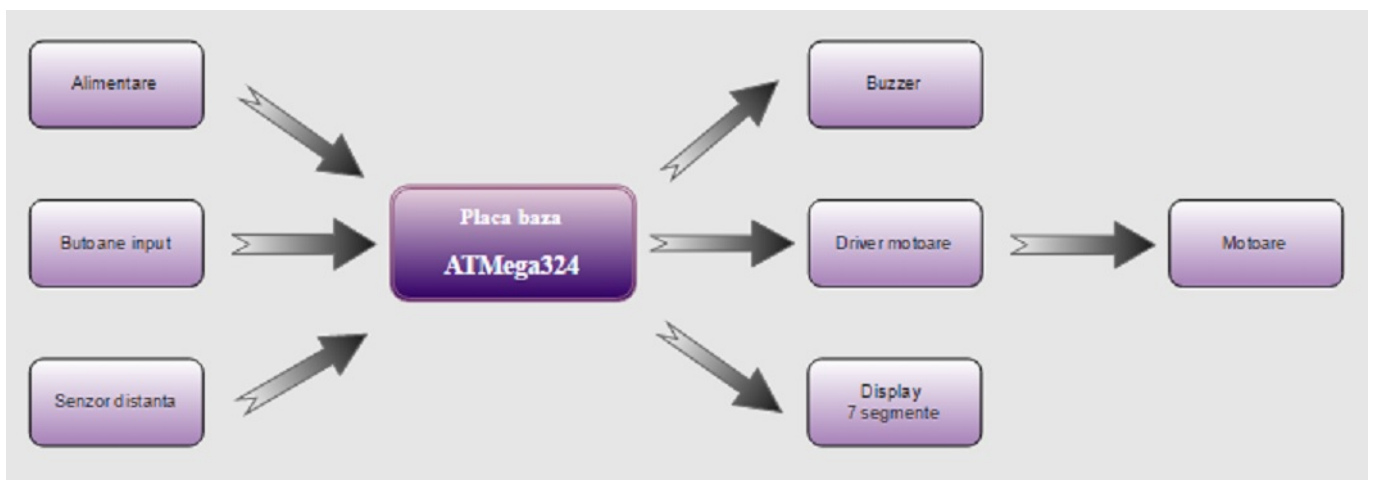
### **Aceste minute furate - oricat de delicioase par - nu merita.**

S-au realizat multe studii pe aceasta tema si toate au ajuns la concluzia ca nu este bine pentru organism sa amanam alarma si sa adormim din nou. Pentru mai multe detalii urmariti acest [video](#) sau accesati link-urile din bibliografie, sectiunea Articole interesante.





## Descriere generală



Utilizatorul va interactiona cu dispozitivul prin intermediul a 5 butoane care vor trimite datele la ATmega324: setare ora, setare minut, oprire alarma, ceas, alarma.

Microcontrolerul ATmega324 va prelua informatii de la senzorul de distanta care va fi asezat in apropierea butoanelor pentru a sesiza cand utilizatorul doreste sa apese butonul de amanare/oprire alarma.

Ora va fi afisata pe un display LED cu 7 segmente.

ATmega324 va reda sunetul alarmei prin intermediul unui difuzor. Acesta va incepe la ora setata si se va opri atunci cand utilizatorul apasa butonul oprire.

Motorul care deplaseaza ceasul va fi controlat de microcontroler cu ajutorul unui driver pentru motoare.

### Utilizare



Pe display este afisat ceasul la ora setata de utilizator, atata timp cat nu se apasa alte butoane. Pentru a accesa functiile dispozitivului, utilizatorul trebuie sa faca urmatoarele actiuni:

- Butonul 0 (in timp ce este afisata alarma) → incrementeaza ora alarmei. Alarma va mai fi afisata inca 5 secunde.
- Butonul 1 (in timp ce este afisata alarma) → incrementeaza minutele alarmei. Alarma va mai fi afisata inca 5 secunde.
- Butonul 2 → opreste alarma atunci cand aceasta e activata.
- Butonul 3 + butonul 0 (simultan) → incrementeaza ora ceasului.



corespunde tutoror celor 4 segmente A ale display-ului (identic pentru celelalte segmente). Punctul ce desparte ora si minutele l-am conectat la pinul comun al celui de-al doilea afisaj.

### Butoane

Cele 5 butoane sunt legate in paralel. Fiecare buton este legat in paralel cu un condensator ce are rolul de a elimina efectul de bounce. In serie cu acestea, este legata o rezistenta de pull-up.

### Difuzor

Difuzorul este legat la microcontroller, la masa avand un condensator.

### Senzor ultrasonic de distanta

Senzorul are legati direct la microcontroller pinii de trigger si echo.

### Motor

Motorul este legat la microcontroller print-un circuit alcatuit din octocuplor, tranzistor si releu. Acesta are alimentare separata de restul placutelor.

### Alimentare

Alimentarea este realizata de la acumulatori. Microcontrollerul este alimentat la 4.5V, iar motorul la 9V.

## Software Design

Mediul de dezvoltare software a fost Programmer's Notepad cu WinAVR. Am incarcat codul pe microcontroller cu HIDBootFlash.exe.

Pentru a intercepta apasarea butoanelor, am folosit intreruperi si am implementat rutina de tratare a intreruperilor ce efectueaza operatiile necesare.

Intreruperile, alaturi de timer-ul 1 al microcontroller-ului, le-am folosit pentru implementarea software-ului necesar functionarii senzorului de distanta. Acesta trimite un semnal pe pinul Trigger timp de 10us si asculta pe pinul Echo semnalul primit, masurand durata acestuia pe nivelul high. Intervalul de timp masurat reprezinta timpul necesar semnalului sa ajunga la obiectul pe care l-a detectat si sa se intoarca inapoi la senzor. Distanta (in centimetri) se calculeaza dupa formula: durata (in secunde) \* viteza sunetului \* 1/2.

Atunci cand senzorul detecteaza miscare la o distanta mai mica decat distanta setata, se trimite un semnal catre motor care deplaseaza masina.

## Rezultate Obținute

Rezultatul final poate fi vazut in acest [video](#) 

Componenta	Rezultat	Obervatii
<b>Hardware</b>		
Placa de baza	Complet ✓	
Conectare LED 7 Segment la microcontroller	Complet ✓	
Conectare difuzor la microcontroller	Complet ✓	

Conectare butoane la microcontroller	Complet ✓	
Conectare senzor la microcontroller	Complet ✓	
Conectare motor la microcontroller	Complet ✓	
<b>Software</b>		
Afisare ceas pe display	Complet ✓	
Afisare alarma pe display	Complet ✓	
Activare alarma la ora potrivita	Complet ✓	
Redare sunet alarma	Complet ✓	
Oprire alarma	Complet ✓	
Setare ora si minute ceas	Complet ✓	
Setare ora si minute alarma	Complet ✓	
Detectare prezenta folosind senzor	Complet ✓	
Deplasare masina	Complet ✓	


Proiectul este functional. Au fost implementate atat hardware, cat si software, toate functionalitatile dorite. Pe display se poate afisa ceasul sau alarma, in functie de ceea ce doreste utilizatorul. Se pot seta ora si minutele ceasului si alarmei. Alarma suna la ora la care a fost setata, iar utilizatorul o poate opri printr-o apasare de buton. Senzorul detecteaza apropierea mainii utilizatorului de butoane si declanseaza deplasarea masinii.



### Probleme

Cand am implementat software-ul, am descoperit ca sunt probleme cu buzzer-ul. Acesta avea nevoie de mai multa putere decat ii putea da microcontrollerul. L-am inlocuit cu un difuzor. Regulatorul de tensiune s-a ars. Nu mai e posibila alimentarea la 12V.

## Concluzii

A fost unul dintre cele mai interesante proiecte din cei 3 ani de facultate . Mi-a placut ca, in final, am realizat un obiect real si practic, nu doar un output care trece niste teste, asa cum se intampla la majoritatea temelor de la facultate. A fost interesant cum, de la gramada de piese, am ajuns sa contruiesc singura, pas cu pas, un obiect functional.

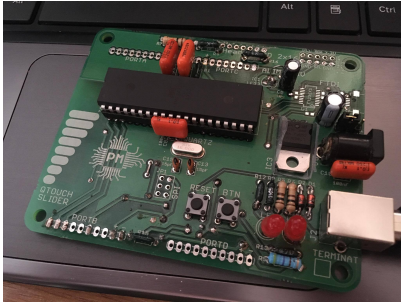
## Download

[Hardware](#)

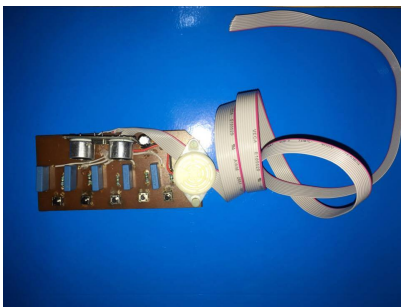
## Software

# Jurnal

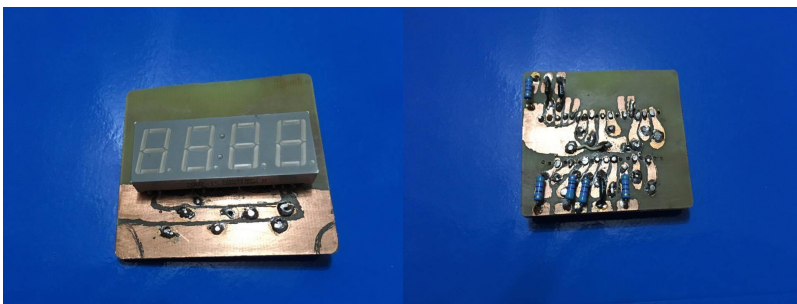
- 18.04.2017 - Finalizarea placii de baza.



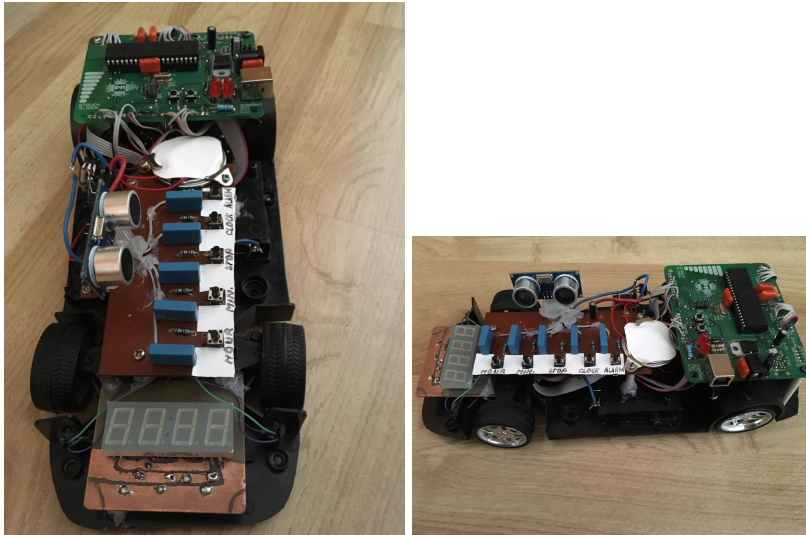
- 22.04.2017 - Alegerea temei, introducere, lista piese, schema bloc.
- 06.05.2017 - Realizarea schemei electrice a proiectului.
- 08.05.2017 - Realizare placuta butoane, buzzer si senzor distanta.



- 11.05.2017 - Realizare placuta display.



- 14.05.2017 - Conectare placuta butoane si placuta display la placa de baza.
- 15.05.2017 - Implementare software pentru afisarea orei, a alarmei si pentru alarma.
- 15.05.2017 - Inlocuire buzzer cu difuzor.
- 16.05.2017 - Implementare software pentru butoane.
- 19.05.2017 - Implementare software pentru senzorul de distanta.
- 20.05.2017 - Realizare circuit pentru conectarea motorului la microcontroller.
- 20.05.2017 - Realizare alimentare de la acumulatori.
- 21.05.2017 - Implementare software deplasare masina.



## Bibliografie/Resurse

[No More 5 Minutes - final result](#)

### Resurse software

Laboratoare PM ([Laborator 0](#), [Laborator 2](#)).

### Resurse hardware

[Datasheet ATmega324](#)

[Datasheet HC-SR04](#)

[Motor](#)

### Articole interesante

[Why hitting snooze is bad for health](#)

[The science behind why hitting the snooze button is bad for you](#)

[Don't hit snooze](#)

- Documentația în format [PDF](#)

From:

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

Permanent link:

[http://ocw.cs.pub.ro/courses/pm/prj2017/ideaconu/no\\_more\\_5\\_min](http://ocw.cs.pub.ro/courses/pm/prj2017/ideaconu/no_more_5_min)



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