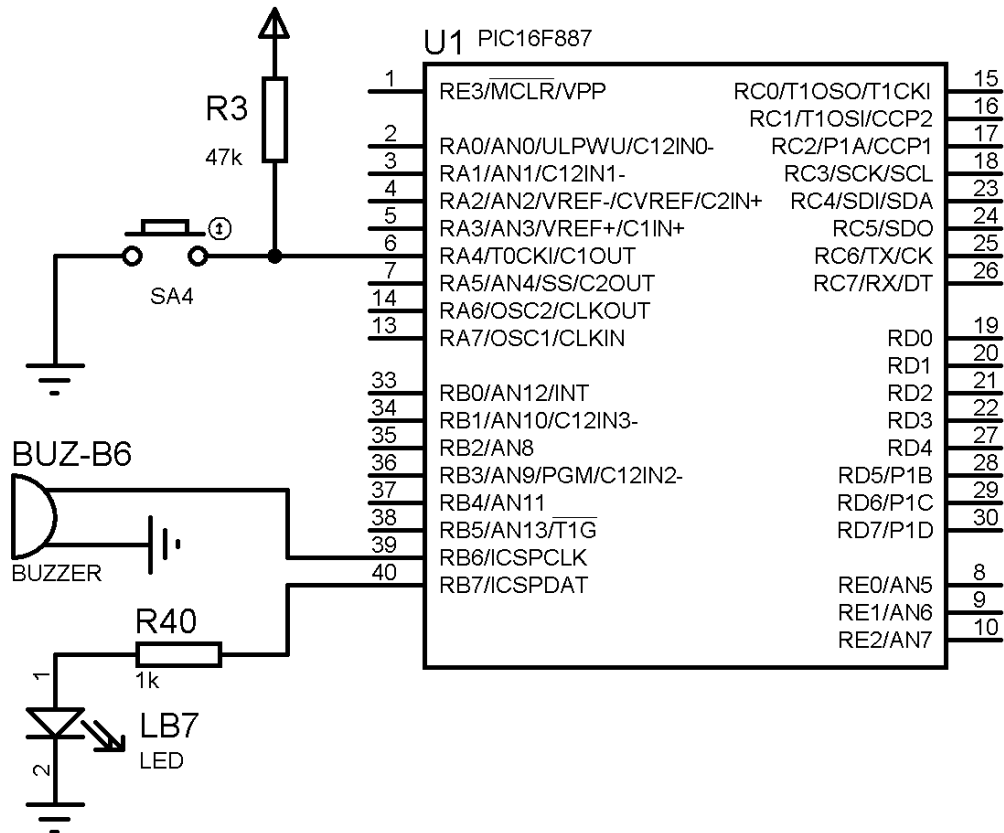


**1. Programul CCS C Compiler. Instalare. Creare proiect.**

Descarcati ultima versiune a programului de la adresa : <http://www.ccsinfo.com/compdemo.php>  
 Instalati programul si lansati-l in executie (vom utiliza meniurile si cuvintele cheie in limba engleza).

NOTA: Oricare versiune mai veche functioneaza asemanator.

Vom utiliza modulul PIC-AP20 cu schema simplificata desenata mai jos.



Aici am desenat doar intrerupatorul SA4 (conectat la A4), LED-ul LB7 (conectat la B7) si buzzer-ul BUZ-B6 (conectat la B6). Pe masura ce avansam vom desena si alte elemente.

Observati organizarea pe porturi a microcontroler-ului (A,B,C,D,E). Acestea au in general cite 8 terminale notate de la 0 la 7 (doar E are terminale de la 0 la 3).

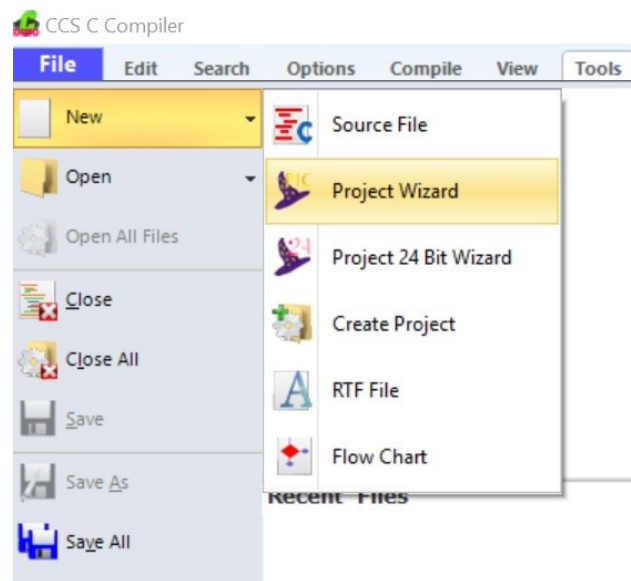
Ca orice sistem ce contine un procesor si acesta are nevoie de un oscilator. Avantajul acestui microcontroler este ca are un oscilator intern care poate fi programat sa functioneze la anumite frecvente. De obicei utilizam frecventa cea mai mare (8MHz) daca nu avem restrictii la consumul de curent.

Deoarece utilizam oscilatorul intern vom avea la dispozitie terminalele RA6 si RA7 pentru a le folosi ca intrari/iesiri. Daca dorim putem utiliza un cristal de cuarț conectat la aceste doua terminale. Conform schemei mai sint necesare doua condensatoare de cca 22pF.

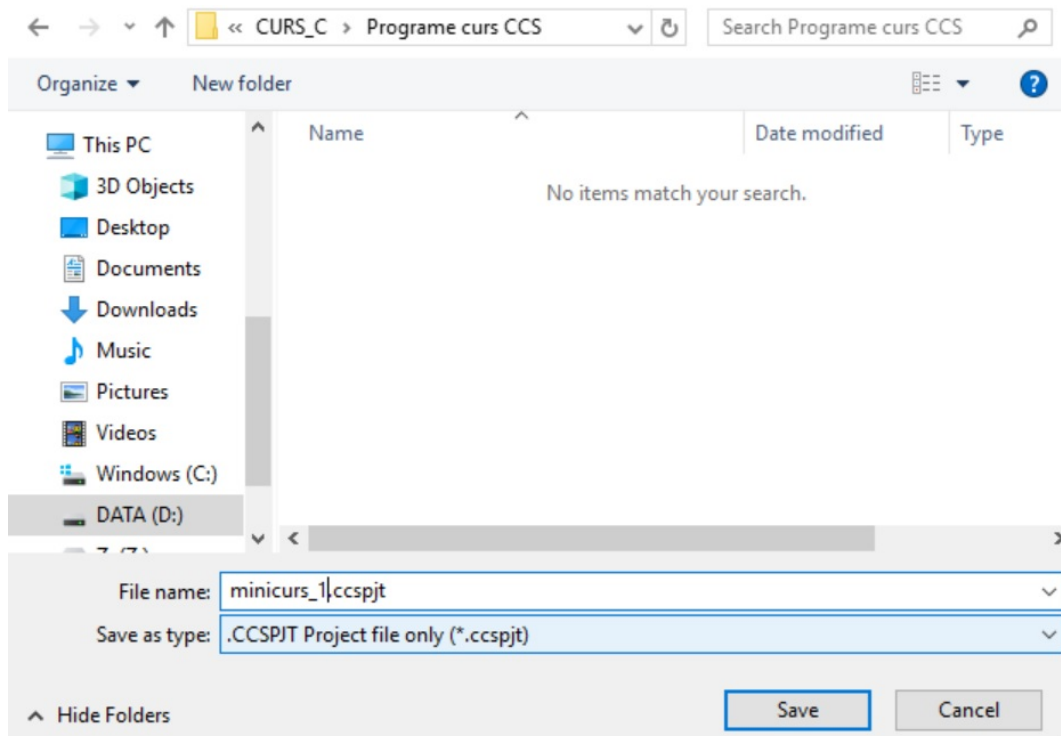
Pe baza acestei scheme vom incepe sa configuram parametri in CCS C Compiler (CCSC).

NOTA: evident pentru o intelegere superioara, este indicat sa studiat si fisa tehnica a PIC16F887.

Deschideti CCSC si selectati **File/New/Project Wizard** , ca in imaginea de mai jos:



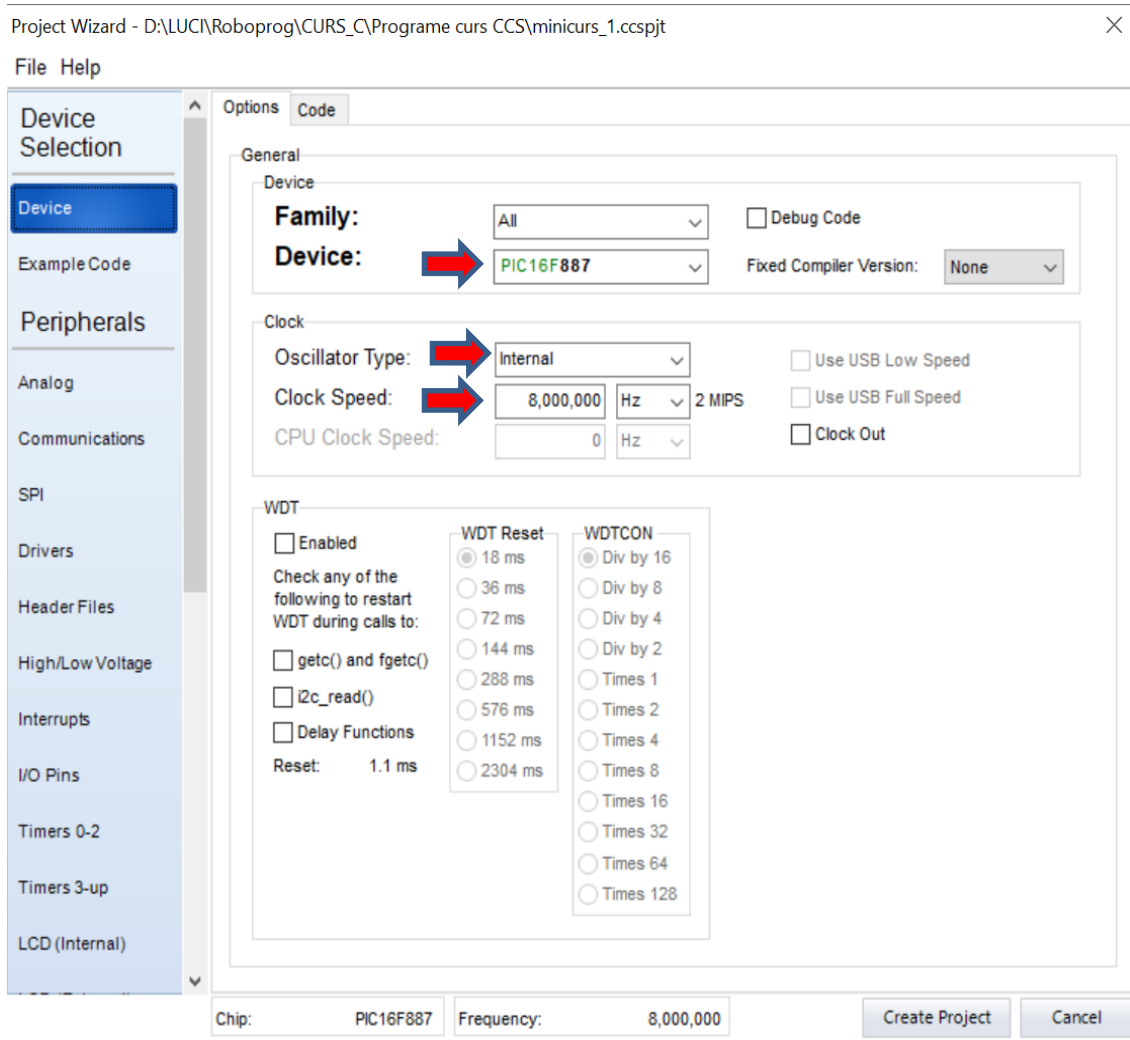
Se va deschide o feresatra unde trebuie sa salvati proiectul.



Dupa salvare se va deschide o noua fereastră unde trebuie sa faceti citeva *setari*.

In partea din stanga se observa parametri care trebuie *setati*. Puteti naviga pentru a vedea despre ce este vorba . Nu vom folosi decat citiva parametri pentru inceput.

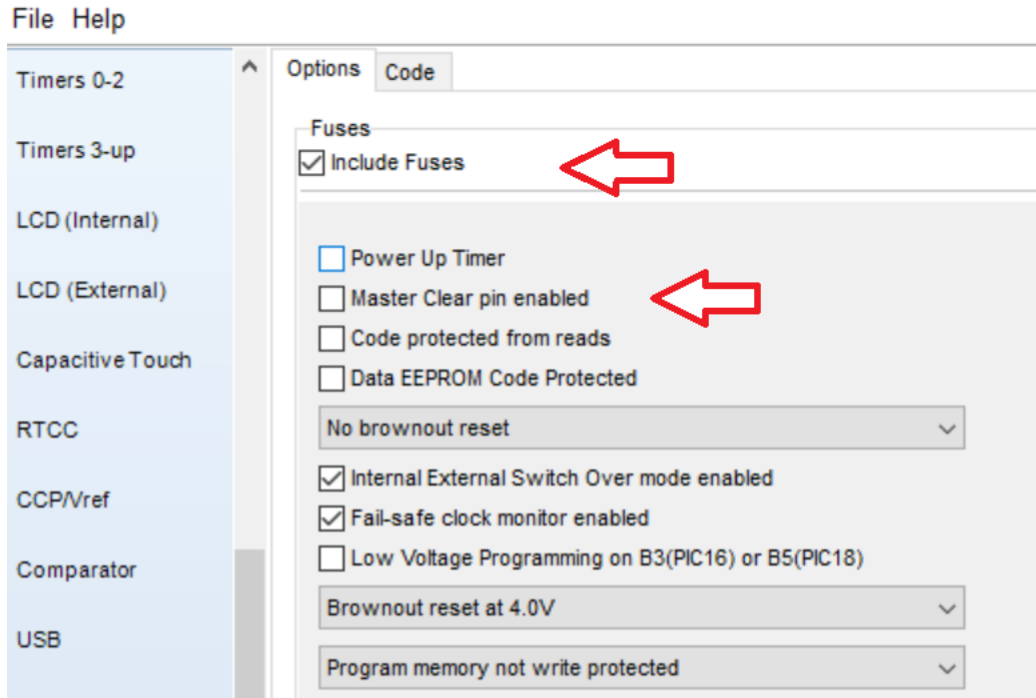
In partea dreapta vom alege tipul microcontroler-ului, oscilatorul intern si frecventa la care va functiona oscilatorul. Trebuie sa retineti ca frecventa reala de tact este de 4 ori mai mica decat cea specificata acolo.



Derulati lista din stanga pina la ultimul parametru.

Deși acum nu ne este necesar este o idee bună să invalidăm terminalul de reset. În acest fel dispunem de încă o intrare (acest terminal poate fi doar intrare nu și ieșire). Bifați opțiunea Include Fuses.

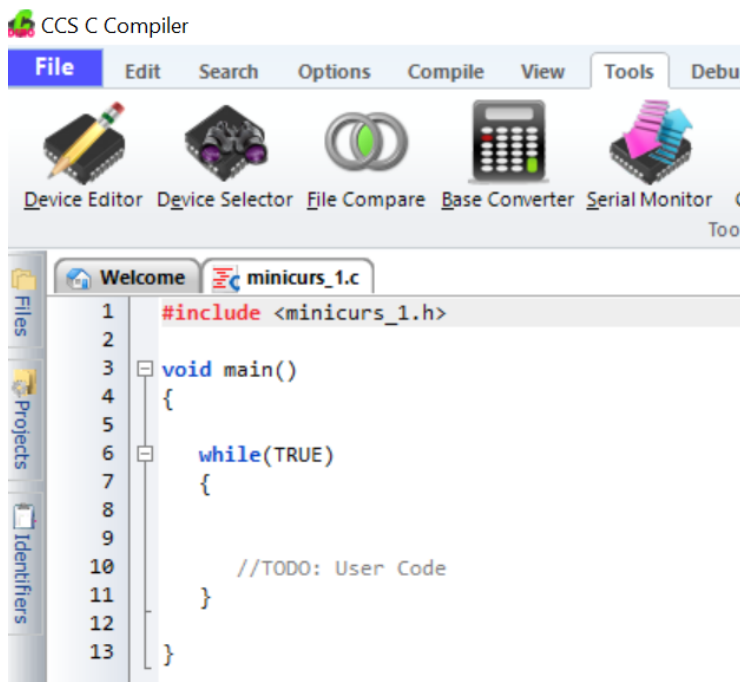
Observați că puteți alege să blocați citirea neautorizată a programului și a memoriei EEPROM.



Pentru inceput setarile sint suficiente.

Este interesant ca la fiecare parametru puteti vedea ce cod introduce in program. Selectati **Code** in partea dreapta sus pentru a vedea codul scris de compilator.

Dupa aceste setari apasati butonul **Create Project** (dreapta jos).



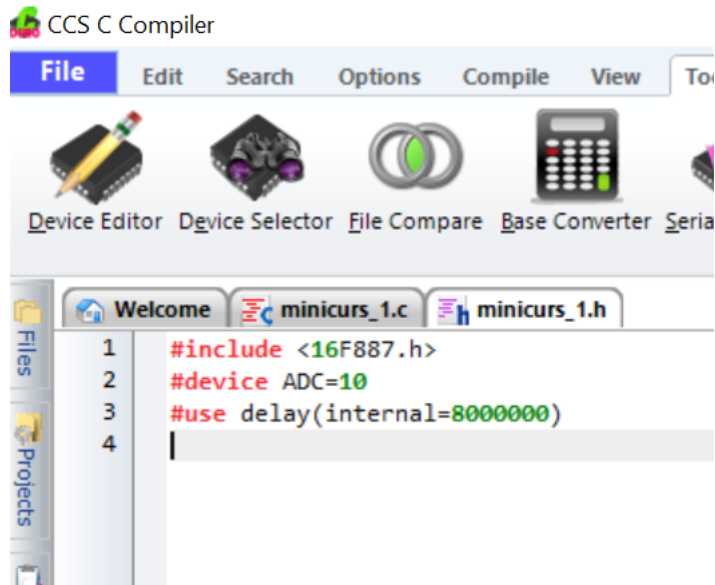
In acest moment se va afisa o pagina cu citeva "instructiuni".

Cel mai important este ca ni se indica unde putem scrie programul nostru:

**//TODO: User Code**

Apasati tasta F9 a PC-ului pentru a compila programul (sau selectati meniul **Compile** si apasati **Compile**).

In stanga, pe verticala este scris : "Files". Daca apasati acel buton veti putea selecta fisierul de tip .h care contine setarile initiale realizate in *wizard* (in general nu este foarte interesant). Se pot face editari aici, de exemplu se poate schimba frecventa la 4MHz.



#include<16F887.h> este o fila specifica acestui microcontroler

#device ADC=10 convertorul ADC functioneaza pe 10 biti

#use delay(internal=8000000)  
oscilator intern cu frecv=8MHz

Din acest moment sintem pregatiti sa scriem primul program.