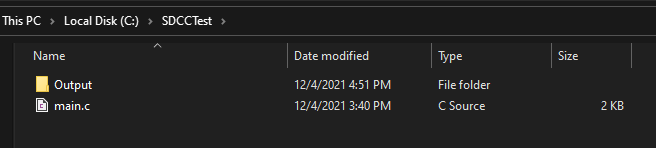
**Workflow pentru compilare cu SDCC**

Cuprins

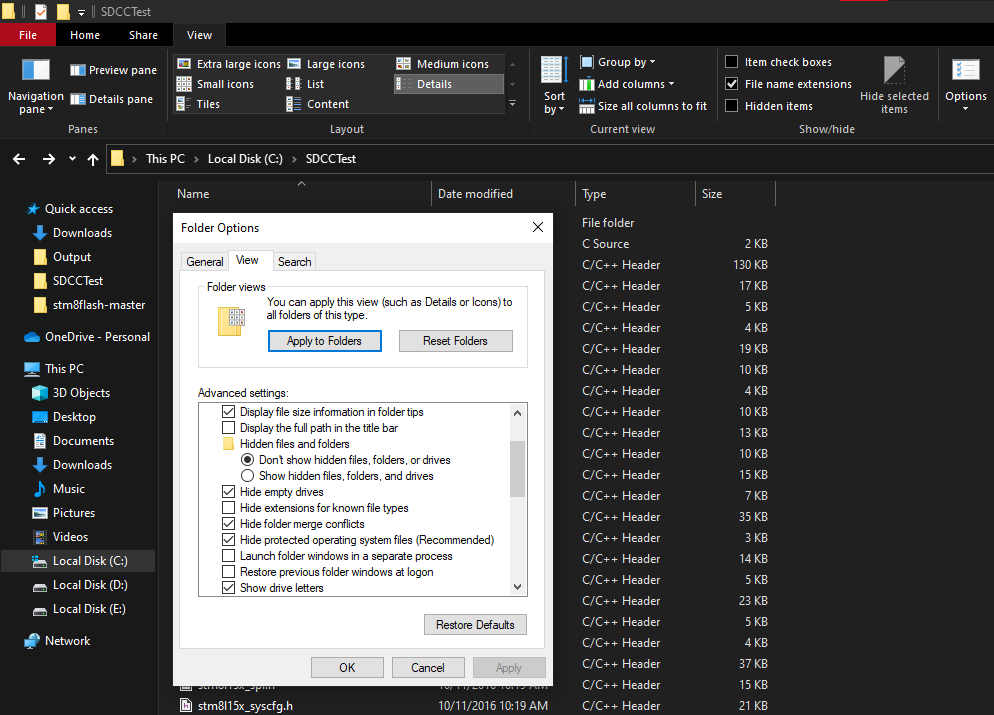
1. Pregătire
   1. Creare proiect de bază
   2. Instalare SDCC
   3. Verificare SDCC
2. Setup librării
   1. Copiere librării
   2. Prelucrare librării
3. Creare Makefile
   1. Introducție in batch
   2. Scrierea scriptului
4. Testare
   1. Compilare program de bază
   2. Compilare flashled
   3. Programare pe placă

1.1 - Creare proiect de bază

Pentru pregătirea proiectului de bază vom crea un nou folder numit “SDCCTest”. În acest proiect vom insera apoi un nou document text, a cărui denumire și extensie o vom modifica în “main.c” (fig. A). Dacă extensiile sunt ascunse, vor trebui activate manual din setările File Explorer (fig. B). În acest folder vom mai crea de asemenea un alt folder numit “Output”.



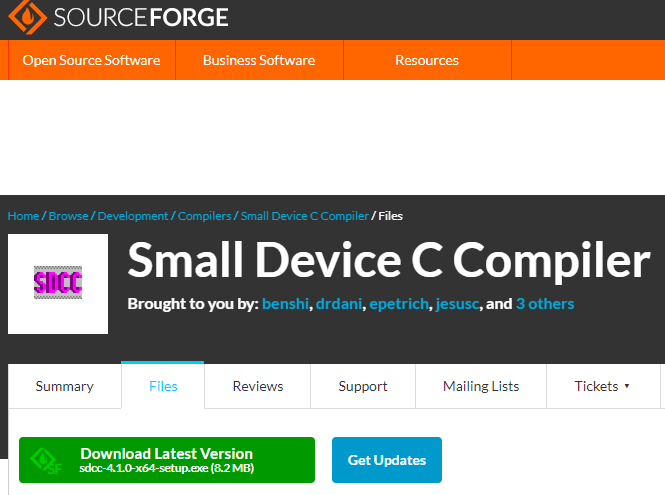
*Figura A, Proiectul de bază*



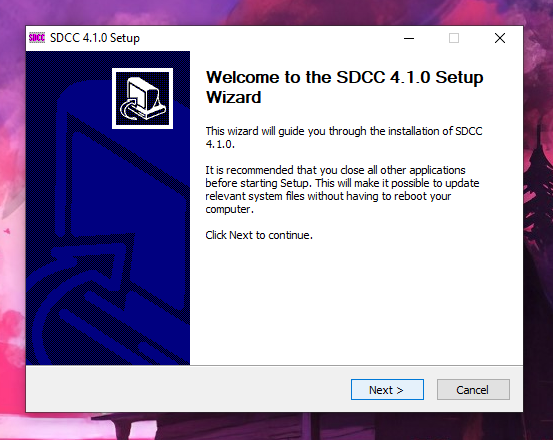
*Figura B, Activarea extensiilor în File Explorer*

1.2 - Instalare SDCC

SDCC (*Small Device C Compiler*) poate fi găsit la următorul link de download: <https://sourceforge.net/projects/sdcc/files/> (Fig. C). La setup nu sunt nevoie de configurări adiționale la niciun pas (Fig. D).



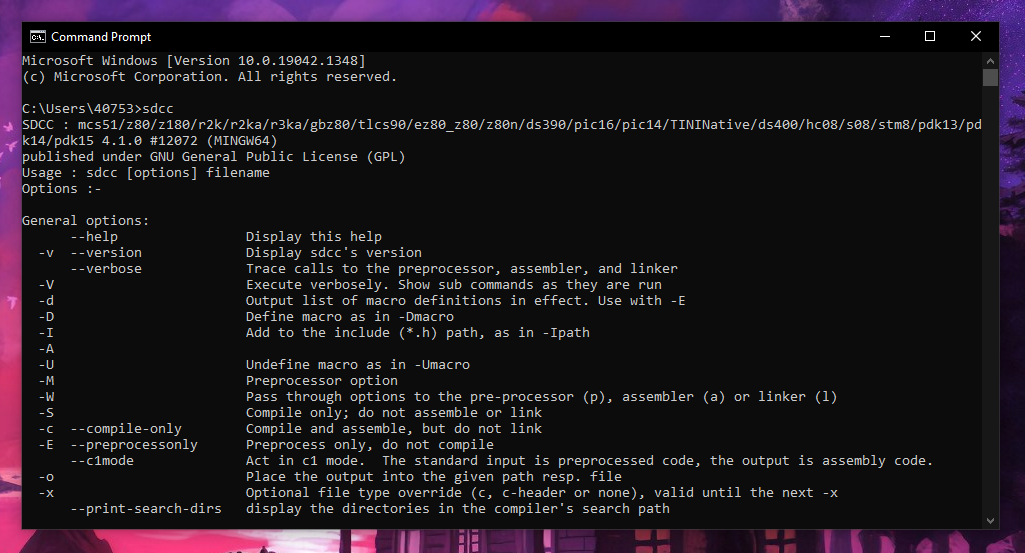
*Figura C, Se utilizează butonul verde*



*Figura D, Setup SDCC*

1.3 - Verificare SDCC

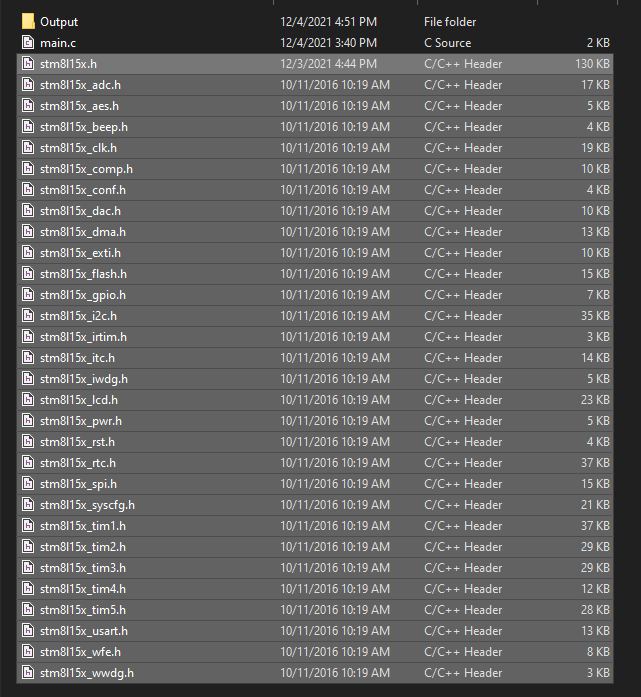
Pentru verificarea instalării corecte a SDCC-ului putem chema SDCC direct din Command Prompt. Utilizând meniul de start, căutați “cmd” și deshideți Command Prompt. Scrieți “sdcc”, fără niciun alt argument, și apăsați tasta enter. Ar trebui să se afișeze informațiile de help, vizibil în figura E.



*Figura E, Informațiile de help de la SDCC*

2.1 - Copierea librăriilor

Librăriile necesare pentru compilarea proiectelor cu SDCC, împreună cu toate fișierele necesare pentru proiect, pot fi găsite pe GitHub la următorul link: <https://github.com/Hexagon-77/STM8-Build>. Acestea provin direct din SPIRIT DK de la STM8. Se vor copia toate fișierele de tip header (\*.h) în folderul nostru principal (Fig. F).



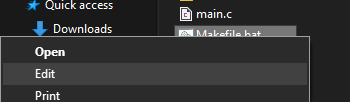
*Figura F, Librăriile necesare*

2.2 - Prelucrare librării

Pentru a face librăriile SPIRIT compatibile cu SDCC, va fi necesar să facem câteva modificări la fișierul “*stm8l115x.h*”, respectiv vom emula un compiler COSMIC adăugând linia “*#define \_\_CSMC\_\_*” (linia 65). Apoi pentru a optimiza librăria pentru placa STM8L152C6 vom decomenta definiția de la linia 34: “*#define STM8L15X\_MD*”. Datorită unui conflict cu librăria “*intrinsics.h*” necesară pentru compiler-ul IAR va trebui să comentăm și linia 2895.

3.1 - Introducție în batch

Pentru a face procesul de compilare mai ușor ne vom crea un Makefile în batch. Fișierul batch este un script care rulează o serie de comenzi în Windows sub Command Prompt (CMD). Acestea au extensia “*.bat*” și se pot edita folosind Notepad prin meniul de context (Fig. G).



*Figura G, Editarea fișierelor batch*

Comenzile de bază necesare în batch sunt următoarele:

“echo *text*“ - Afișează o secvență text

“pause” - Așteaptă un input de la tastatură

“cd *directory*” - Schimbă folderul de lucru

“dir” - Afișează conținutul folderului de lucru

“REM” - Marchează un comentariu

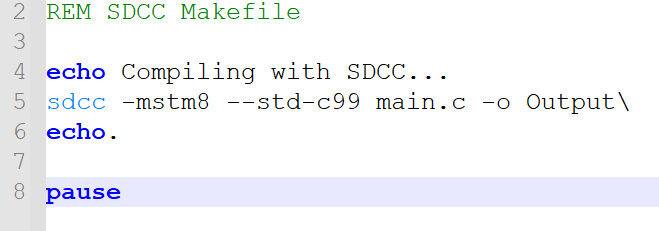
3.2 - Scrierea scriptului

Makefile-ul nostru va afișa un scurt text de debugging, după care va apela SDCC cu parametrii următori:

“mstm8” - Specifică placa de dezvoltare (platforma STM8)

“std-c99” - Utilizăm standardul de C numit C99

“-o *directory*” - Realizăm compilare fișierelor în folderul specificat



*Figura H, Makefile-ul SDCC*

4.1 - Compilarea programului de bază

Utilizăm următorul program de bază pentru a testa procesul inițial de compilare:

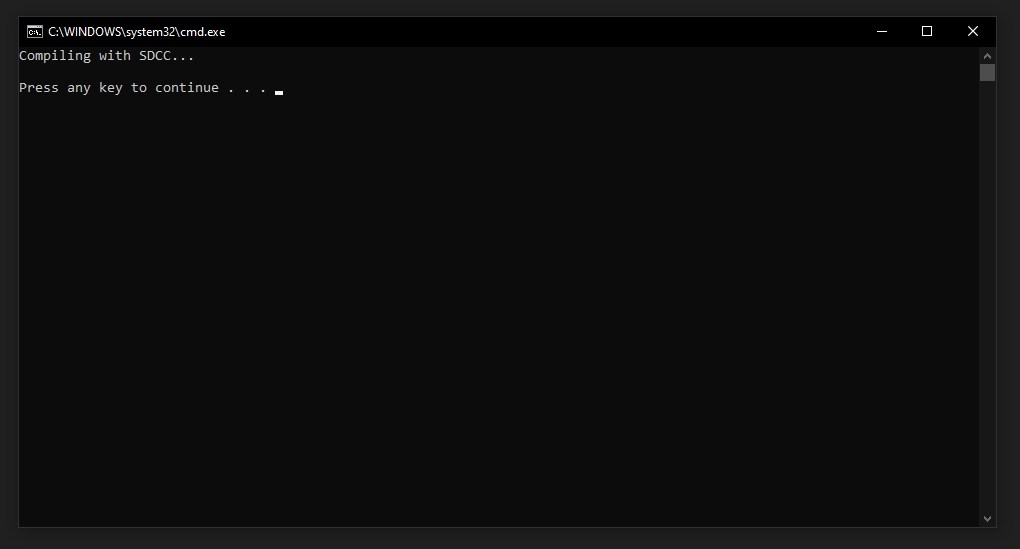
int main() {

int a = 1, b = 3;

int c = a+b;

}

Rulând makefile-ul ar trebui sa obținem următorul output:



*Figura E, rularea makefile-ului pentru programul de bază*

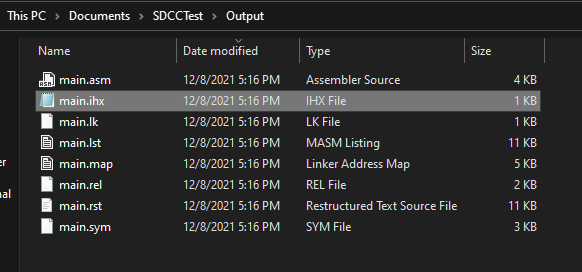
4.2 - Compilarea flashled

Următorul program pe care îl vom compila este flashled, disponibil de asemenea pe GitHub la următorul link: <https://github.com/Hexagon-77/STM8-Build/blob/main/main.c>

Programul va aprinde LED-urile de pe placă intermitent în ciclu infinit.

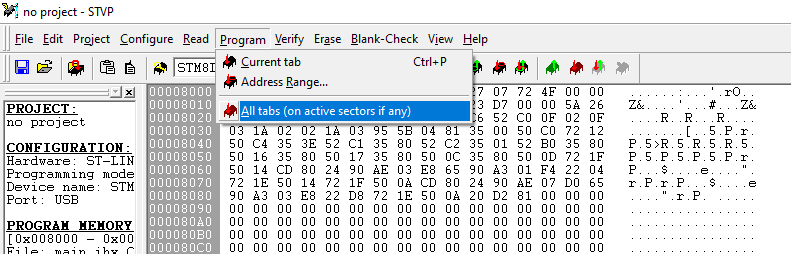
4.3 - Programarea pe placă

Folosim STVP pentru a programa pe placă fișierul final IHX compilat.



*Figura I, Fișierul IHX compilat*

Fișierul compilat IHX se află în folderul de Output al proiectului. Deschidem cu aplicația ST Visual Programmer (STVP) fișierul IHX, fie prin drag & drop, fie prin meniul *File* » *Open*, după care selectăm din submeniul de *Program* » *All tabs* pentru a programa pe placă (Fig. J). După ce bara de progres se finalizează trebuie să închidem programul STVP pentru a putea testa programul final pe placă.



*Figura J, Programarea pe placă*