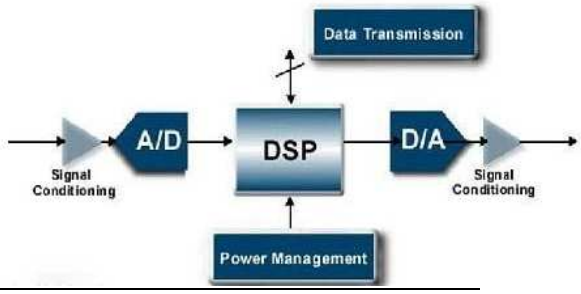

Travaux Pratiques II
(DSP)

L'objectif de ce TP est de se familiariser avec l'environnement de développement intégré « Code Composer Studio » tout en écrivant, compilant et exécutant des programmes C et assembleur.

Partie 1: Calcule de la somme n+ (n-1)+.....+1

Dérouler le code suivant pour n = 10.

```
//sum.c Finds n+(n-1)+...+1. Calls ASM function sumfunc

#include <stdio.h>
main()
    {
     short n=10;                    // set value
     short result;                  // Result from asm function
     result = sumfunc(n);           //Call ASM function sumfunc
     printf("sum = %d", result);    // Print result from asm function
}
```

; Sumfunc.asm Assembly function to find n + (n-1) + ... + 1

;sumfunc.asm Assembly function to find n + (n---1) + ... + 1

```
                .def        _sumfunc ;      function called from C
_sumfunc:       MV      .L1 A4,A1;          setup n as loop counter
                SUB     .S1 A1,1,A1 ;       decrement n
LOOP:           ADD     .L1 A4,A1,A4 ;      accumulate in A4
                SUB     .S1 A1,1,A1 ;       decrement loop counter
        [A1]    B       S2  LOOP ;          branch to LOOP if A1#0
                NOP         5 ;             five NOPs for delay slots
                B       .S2 B3 ;            return to calling routine
                NOP         5 ;             five NOPs for delay slots
                .end
```

Cet exemple illustre un programme C appelant une fonction assembleur C6000. Le programme source C **sum.c** appelle la fonction codée en assembleur sumfunc.asm. La valeur de n est définie dans le programme C principal. Il passe par le registre A4 (par convention). Par exemple, l'adresse de plusieurs valeurs peut être transmise à la fonction sumfunc via A4, B4, A6, B6, etc. La somme résultante de la fonction **Sumfunc** (asm) est renvoyée pour générer le résultat dans le programme C, qui imprime ensuite cette somme résultante.

Le nom de la fonction assembleur est précédé d'un trait de soulignement (par convention). La valeur n dans le registre A4 de la fonction asm est déplacée vers le registre A1 pour définir A1 comme compteur de boucle puisque seuls A1, A2, B0, B1 et B2 peuvent être utilisés comme registres conditionnels. A1 est alors décrémenté. Une section de code en boucle commence par l'étiquette ou l'adresse.

Partie 2: Programmation en langage C

a) Ecrire un programme C qui calcule la somme des N premiers termes de la série harmonique :

$$1 + 1/2 + 1/3 + \ldots + 1/N$$

b) Ecrire un programme C qui permet de calculer le produit scalaire de deux vecteurs d'entiers U et V de même dimension N=3.

$$U = \begin{bmatrix} 1 \\ 0.5 \\ 0 \end{bmatrix} \quad et \quad V = \begin{bmatrix} 0 \\ 0.5 \\ 1 \end{bmatrix}$$

c) Ecrire un programme C qui réalise le produit de deux matrices carrées (A et X), N=3.

$$A = \begin{bmatrix} 1 & 1 & 0.5 \\ 0 & -1 & 2 \\ 0.5 & 0 & 0.5 \end{bmatrix} \quad et \quad X = \begin{bmatrix} 1 & 0 & 1 \\ 2 & 0 & -1 \\ 0.5 & 0 & 0.5 \end{bmatrix}$$

d) Ecrire un programme C qui calcule la somme suivante :

$$y = \sum_{i=1}^{N} \frac{x^i}{i!}, \quad \text{avec} \quad N = 5, x = 0.5$$

1. Calcule de n ! = n (n-1).....1 (C et Assembleur).
2. Ecrire un programme C qui calcule la somme suivante

$$y = \sum_{i=1}^{N} \frac{x^{i!}}{i!}, \text{ avec N=5, x=0.5}$$

   En utilisant deux fonctions pour le calcul de la puissance xi (puis (i,x)) et le factoriel i !, fact(i).

3. Résoudre le problème 2 en utilisant le C et l'assembleur C6000.