Université BADJI Mokhtar Annaba

Faculté des sciences de l’ingénieur

Département d’électronique

Master 1 Système embarqué

Matière : DSP

**TP 2 : Prise en main de l'environnement CCS**

 **Programmation en C/C++**

1. Création d’un projet :

La création d’un nouveau projet se fait par la sélection de : Project 🡪 New

Saisir le nom du projet qui sera sine8\_LED

Sélectionner la cible TMS320C67XX

Cliquer sur Finish .



Ajouter un fichier source nommé :sine8\_LED.C au projet.

Ajouter le fichier source c6713dskinit.c au project.

c6713dskinit.c contient les définitions de fonctions d’un certain nombre de routines de bas niveau par lesquelles : comm.\_poll() et output\_left\_sample() .

Sélectionner Project --> Add Files to Project et chercher dans les fichiers d’extension ( \* .c, \* .ccc).

Faire un double clic sur c6713dskinit.c qui doit apparaitre dans la fenêtre "Project View".

Ajouter le fichier source vectors\_poll.asm qui contient la table de gestion des interruptions du 6713. Suivre la même procédure décrite précédemment mais en cherchant dans les fichiers de type ASM ( \* .a \* ). vectors\_poll.asm doit apparaitre dans la fenêtre "Project View".

Ajouter, maintenant, les fichiers de support bibliothèques en suivant toujours la même procédure mais en recherchant les fichiers ayant pour extension ( \* .o \* , \* .l \* ) :

rts6700.lib (run - time support pour l’ architecture des C67x),qui doit se trouver dans le répertoire : c:\CCStudio\_v3.1\c6000\cgtools\lib

dsk6713bsl.lib (board support pour le DSK C6713), qui doit se trouver dans le répertoire : c:\CCStudio\_v3.1\c6000\dsk6713\lib

csl6713.lib (chip support pour le processeur C6713)qui doit se trouver dans le répertoire c:\CCStudio\_v3.1\c6000\csl\lib

Ajouter le fichier c6713dsk.cmd au projet (fichier de commande linker) qui se trouve dans le répertoire: c:\CCStudio\_v3.1\myprojects\Support .

Pour cela, il faut suivre la même procédure de rajout de fichiers mais en sélectionnant les fichiers de type ( \* .cmd; \* . lcf) . c6713dsk.cmd doit apparaitre dans la fenêtre "Project View".

Pour rajouter les fichiers entêtes (headers), cliquer sur Project 🡪 Scan All File Dependencies.

Les fichiers c6713dskinit.h , dsk6713.h , et dsk6713\_aic23.h apparaitront dans la fenêtre "Project View".

La fenêtre "Project View" doit apparaitre telle que sur la figure suivante :



Fenêtre "Project View" après l’ajout des différents fichiers.

Le fichier dsk6713.gel est rajouté automatiquement au projet. Il initialise la carte DSK C6713 en faisant appel à la bibliothèque de support de la carte (bsl) afin d’utiliser la PLL pour fixer l’horloge du CPU à 225 MHz (par défaut, le C6713 fonctionne à 50 MHz).

GEL= General Extension Language (langage similaire à C).

Génération de code et options "Build"

Les outils utilisés par CCS permettant la génération du code exécutable (compilateur C, assembleur, et linker) ont chacun des options qu’il faudrait définir convenablement avant de lancer le "build" d’un projet. Une fois définies, ces options sont stockées dans un fichier

Configuration du compilateur (Définition des options):

Sélectionner Project -- > Build Options et cliquer sur l’onglet Compiler.

Choisir les options telles qu’indiquées par les figures suivantes :

 

*Basic* category: set *Target Version* to *C671x ( - mv6710) Advanced* category set *Memory Models* to *Far ( mem\_model:data=far)*



*Preprocessor* category set:

1. *Pre – Define Symbol* to *CHIP\_6713*
2. *Include Search Path* to *c:*\ *CCStudio\_v3.1* \ *C6000* \ *dsk6713* \ *include*

Configuration du Linker :

Cliquer sur l’onglet "*Linker" dans la fenêtre des options de build (voir figure ci-dessous):*

**

L’option –c est utilisée pour initialiser les variables au moment de l’exécution.

L’option –o sert à nommer le fichier exécutable

sine8\_LED.out .

Fichier fournissant des informations utiles sur l’emplacement mémoire des fonctions, etc

Identificateur du fichier de sortie (basé sur le nom du projet)

Bibliothèques incluses : rts6700.lib;dsk6713bsl.lib;csl6713.lib

Cliquer sur OK .

Construction (Building), téléchargement (Downloading), et exécution (Running) du Projet:

1. Construire ce projet.

Sélectionner Project --> Rebuild All (Ou cliquer sur l’icone avec 3 flèches vers le bas).

Ceci compile and assembles tous les fichiers C et assemble le fichier vectors\_poll.asm.

Les fichiers objets résultant de cette opération sont lies avec les fichiers bibliothèques pour créer le fichier exécutable sine8\_LED.out qui peut être chargé et exécuté par le DSP C6713.

Un fichier cc\_build\_Debug.log est alors créé et montre les fichiers qui sont compiles et assembles avec les options du compilateur sélectionnées. Il fournit aussi une liste des fonctions de support utilisées.

Un message apparait alors dans la fenêtre du CCS: “ Build Complete,0 Errors, 0 Warnings, 0 Remarks ” .

Il est possible qu’un avertissement (warning) soit généré sur la taille de la pile (Stack Size). Il peut être simplement ignoré ou il faut désélectionner l’option "Warn About Output Sections" dans la catégorie "Advanced" des options du "Linker Build" .

Le warning peut aussi être éliminé en choisissant une valeur appropriée pour la taille de la pile dans la catégorie "Advanced" des options du "Linker Build" (e.g., 0x1000 ).

Exécution :

Connexion à la carte the DSK: Sélectionner Debug -->Connect et vérifier qu’il est indiqué "connection" dans le coin inférieur gauche de la fenêtre du CCS.

1. Sélectionner File --> Load Program pour charger le fichier sine8\_LED.out qui doit se trouver dans le répertoire c:\CCStudio\_v3.1\MyProjects\sine8\_LED\Debug .

Sélectionner Debug --> Run .

Pour vérifier qu’un signal sinusoïdal de fréquence 1 kHz est présent au niveau de la "LINE OUT" et le "HEADPHONE" du DSK, lorsque DIP switch #0 est actionné vars le bas, utiliser un oscilloscope

connecté à "LINE OUT" et des écouteurs connectés à "HEADPHONE".

Remarque : pour arrêter le programme, sélectionner Debug 🡪 Halt.