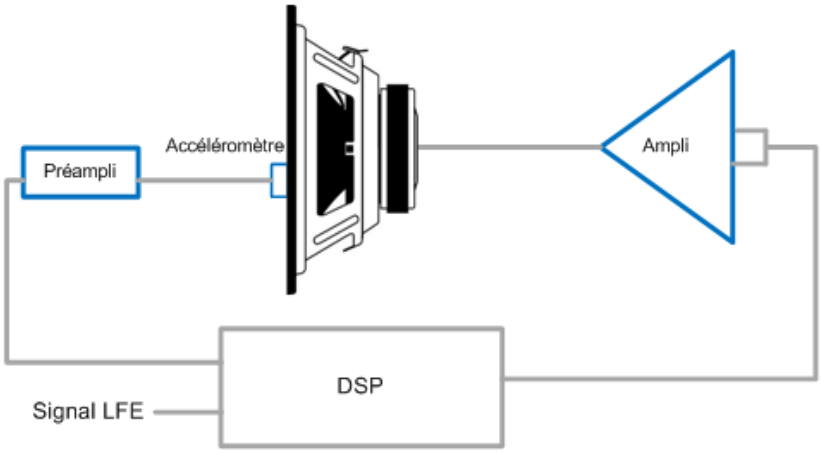
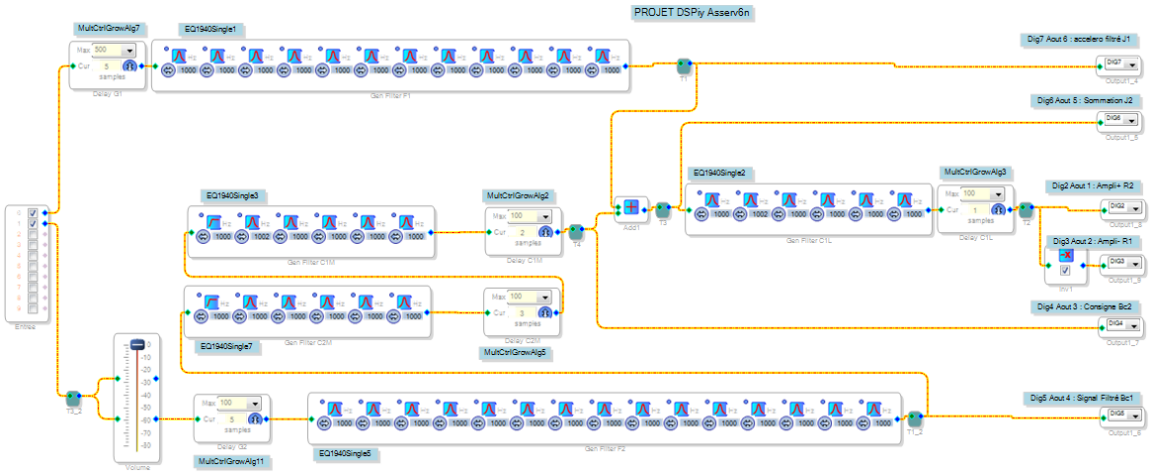
CREATION OU MODIFICATION D'UNE APPLICATION DSTUDIO

VERSION 1.0 DATE : 01-2016





Objet : création ou modification d'une application sigma studio hébergée par un DSPIY V1 ou V2 à base de dsp ADAU1701. Prise en compte de la nouvelle application par DStudio.

Version DSPIY : V1, V2 (l'Adau 1452 n'est pas traité ici)

Version DStudio : 4.22 beta 3

Version Sigma Studio : 3.11 (sur Windows 7 – 32 bits)

Documentations connexes : Description et utilisation du fichier .dctl

Table des matières

1 Introduction

Ce petit guide montre comment créer une application « custom » DStudio à partir d'une des applications préparées par Thierry.

Note : certains écrans DStudio présentés dans ce document sont modifiés par le fichier .dctl L'utilisation et le paramétrage de ce fichier n'est pas abordé ici. Il est décrit dans ce document : http://www.dspiy.be/forum/

Le projet qui va servir d'exemple (asservissement d'un subwoofer) peut se dessiner de la façon suivante :

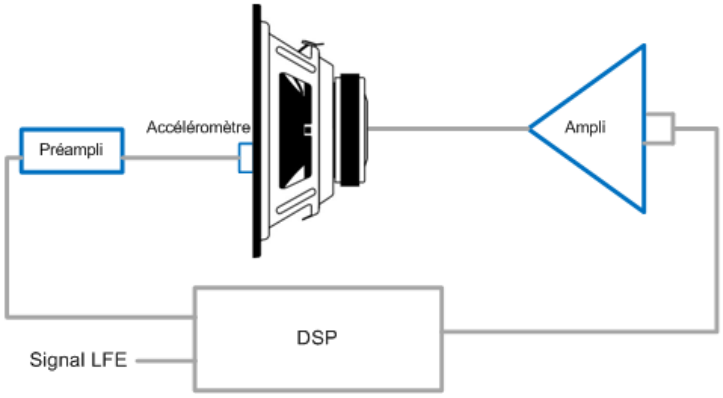


Figure 1 : Schéma de principe

Avec pour « cahier des charges » sommaire :

- On entre un signal BF sur une entrée analogique sur lequel on doit pouvoir au moins :
- Filtrer en passe haut et passe bas
- Appliquer un gain
- Retarder
- Appliquer des égalisations paramétriques

- On entre en analogique le signal d'un accéléromètre 8mv/g, pouvant aller jusqu'à 100g (800mv crête) préamplifié par 2, donc 1.6v crête, sur lequel on doit pouvoir au moins :
- Retarder
- Appliquer des égalisations paramétriques
- Appliquer un gain
- Filtrer

On compare les deux signaux, et on sort en symétrique le signal erreur, issu de cette comparaison.

On devra pouvoir sur ce signal direct (c'est adire dans la chaine directe de l'asservissement) :

- Filtrer
- Retarder
- Appliquer un gain

2 *Précautions à prendre avant de modifier*

On va affiner l'utilisation du DSPIY et de l'application DStudio 2x1+1dans ce cadre, mais il y a un certain nombre de précautions à prendre. En effet, l'application DStudio a été écrite pour prendre en compte les différents paramètres de l'application Sigma Studio originelle. Chaque paramètre, ou bloc de paramètres, Sigma Studio a sa « case » de programmation dans DStudio. Il ne faut donc pas modifier ces correspondances si on veut que ça fonctionne, et que la nouvelle application DStudio programme correctement le DSPIY .

Les blocs de base disponibles en fonction des applications DStudio déjà crées sont accessibles :

- Par les deux onglets filtre (voie1 et voie 2), de DStudio, qui permettent chacun, et de façon indépendante sur chacune des voies, d'ajouter une tempo globale (ms), de filtrer avec jusqu'à 15 biquads, de gérer le gain de la voie (AdB), et d'inverser la phase (+/-) :

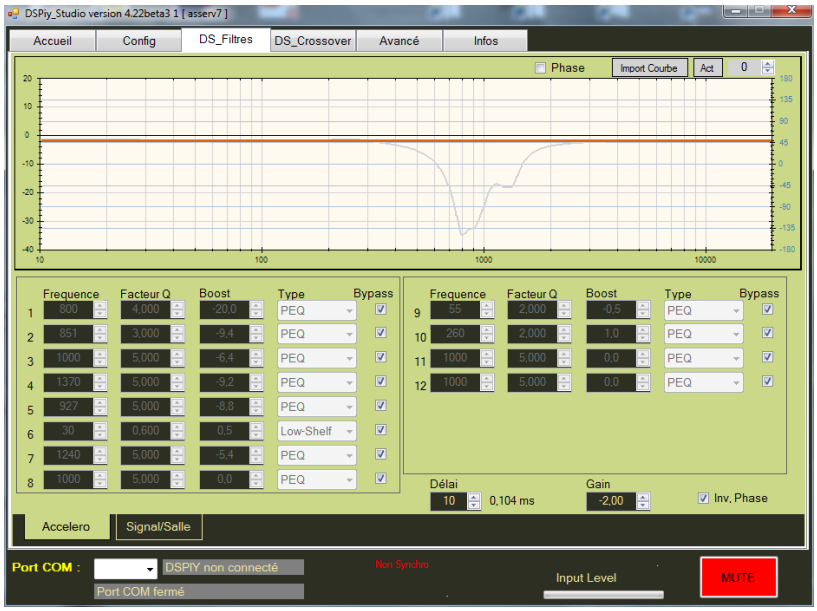
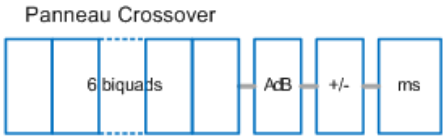


Figure 2 : Ecran Filtre

- Par les trois panneaux (Low, Mid et High) des deux onglets crossover (voie 1 et voie2). Chaque panneau permet de filtrer, d'inverser la phase (+/-), de gérer le gain (AdB), et d'ajouter un délai (ms).



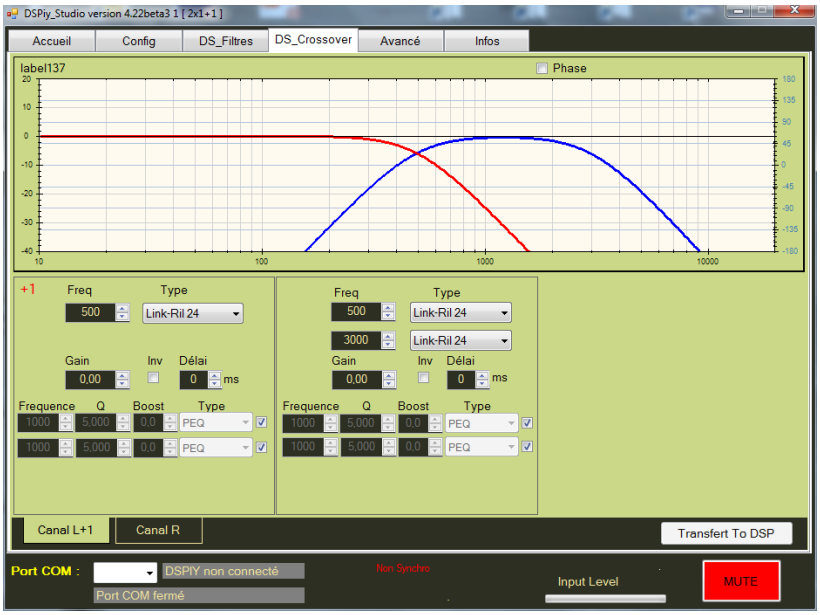
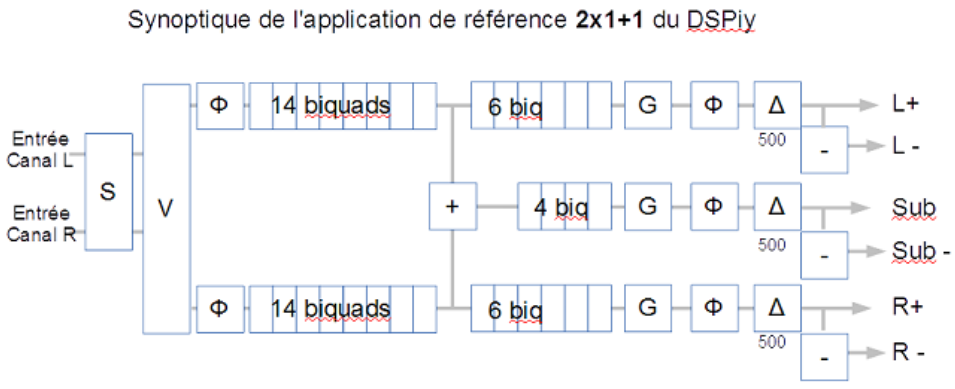


Figure 3 : Ecran Crossover (ici 2 panneaux : Low et Mid)

A partir de ces briques de base on va construire notre process DSP :

On choisit l'application DStudio « 2x1+1 » qui semble avoir les blocs nécessaires à notre exercice :



3 *Modification de l'application Sigma Studio*

On pourrait par exemple modifier selon ce principe :

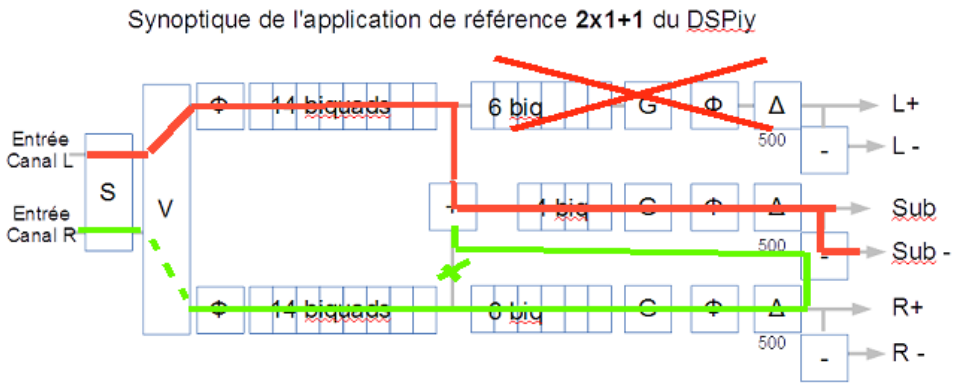


Figure 4 : Modifications à appliquer

Pour cela, il faudra :

Diminuer le nombre de biquads sur la voie L (rouge) de 14 à 12.

Augmenter le nombre de biquads sur la voie R (verte) de 14 à 15.

Augmenter le nombre de biquads sur le crossover après l'additionneur de 4 à 6.

Supprimer la liaison entre la sortie des 14 biquads et l'additionneur.

Reboucler la sortie du crossover R+/R- vers l'additionneur.

Supprimer le crossover de la voie L+/L-, ou le déplacer sur la voie R (verte).

N'appliquer le volume que sur la voie R (verte)

Rajouter un délai global au début de chaque de voie.

On aura alors :

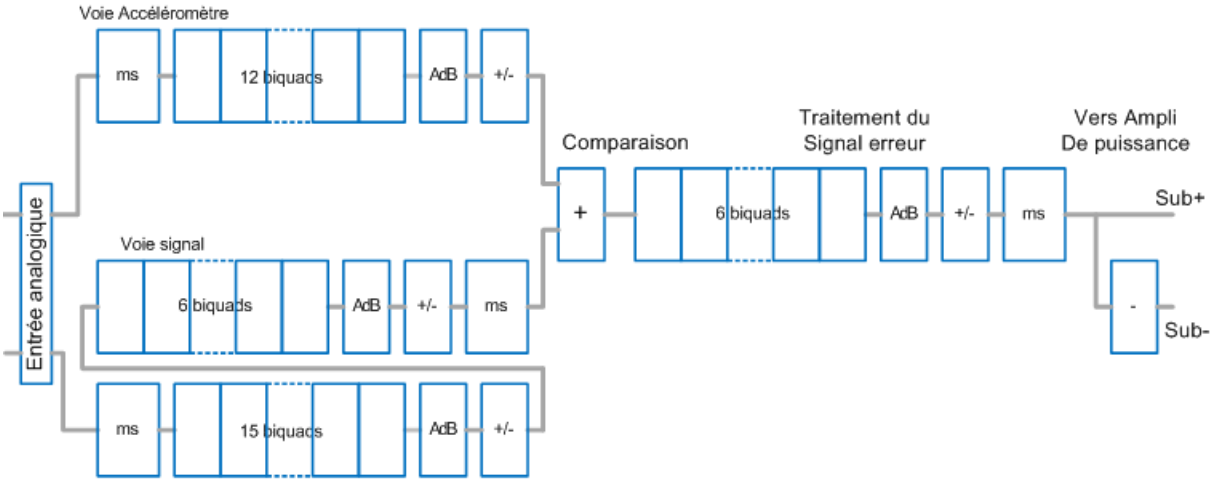


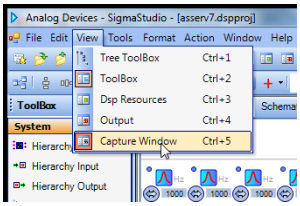
Figure 5 : Synoptique de l'application Asservissement avec DSP

Pour être sûr de garder la bonne numérotation des blocs dans l'application Sigma Studio, on relève leur nom avant modification des interconnexions. Ces noms sont du type :

- Egaliseurs (groupes de biquads) : Eq1940Single1, 2, 3, 4....
- Délais : MultCtlDelGrowAlg1, 2, 3...

Comment faire :

Une fois l'application à modifier (2x1+1.dspproj ici) chargée et affichée dans Sigma-Studio, on sélectionne la « capture window (Barre de Menu, View, Capture Window),



On change une valeur dans le bloc dont on veut connaitre le nom (on change la valeur du délai par exemple, ou la fréquence de coupure d'un filtre). L'action est tracée dans la « capture window », et on peut y lire le nom du bloc concerné.

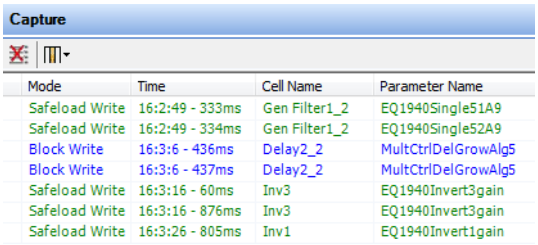


Figure 6 : Capture Window

Pour être prudent, on peut inscrire ces noms sur l'application de départ en prenant soin de bien les conserver au cours des modifications, **ces noms ne doivent en aucun cas changer** :

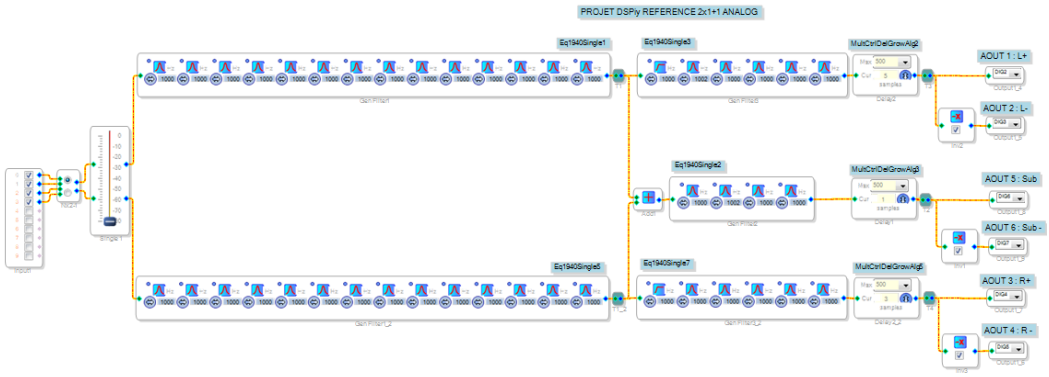


Figure 7 : Blocs nommés

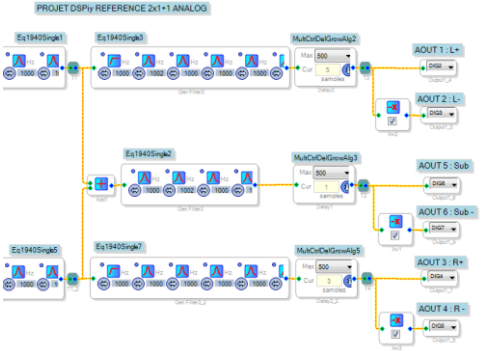


Figure 8 : Détail des blocs nommés

Il est ainsi facile de savoir « qui fait quoi » dans l'application DStudio. Cela va nous être utile pour modifier notre application.

Par exemple :

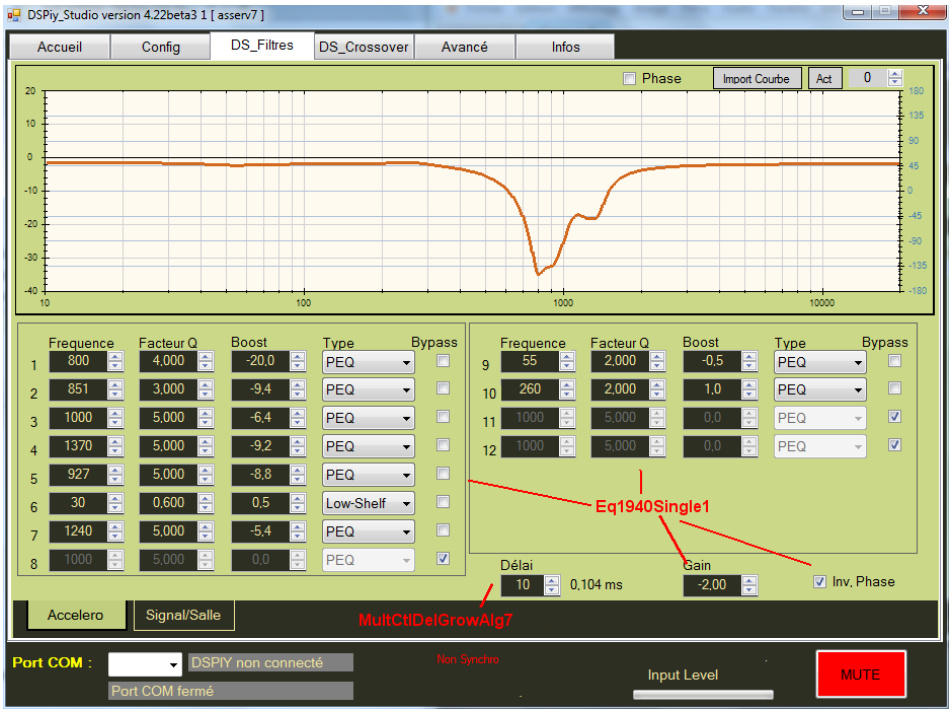


Figure 9 : nom des blocs dans 2x1+1 / Filtres

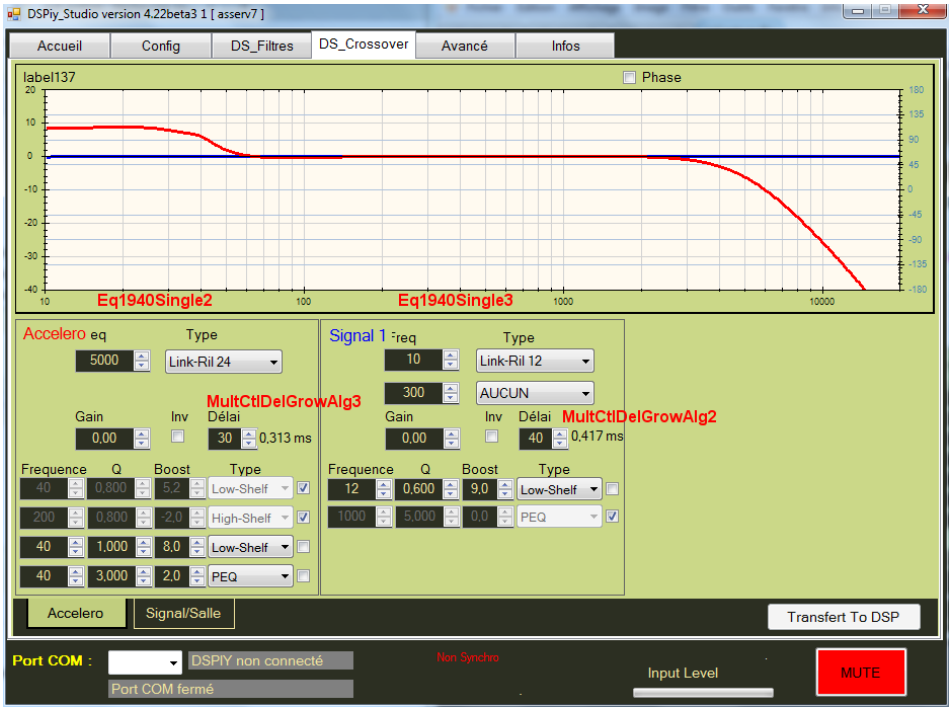


Figure 10 : Nom des blocs dans 2X1+1/crossover

Pour que l'appli DStudio fonctionne avec la nouvelle appli Sigma Studio, il est impératif de garder les noms originaux aux blocs, afin que ces derniers restent affectés aux bonnes « cases » de DStudio.

4 *Création de la nouvelle application dans sigma studio*

On peut supprimer des blocs, changer toutes les interconnexions, rajouter des biquads dans les blocs, modifier des délais et rajouter des opérateurs.

Si par mégarde on a effacé un bloc et qu'on veut le recréer, il faut à tout prix que le nouveau bloc porte le bon nom, qui sera affecté à la bonne case DStudio. Mais c'est Sigma Studio qui incrémente le N° des bloc, donc on peut être amené à recréer plusieurs blocs avant d'avoir le bon N°, et à effacer les inutiles ensuite.

Par exemple, le délai global, qui n'existe pas dans l'application 2x1+1 est un bloc MultCtlDelGrowAlg11 sur la voie 2 (Right). Il faudra donc créer les blocs MultCtlDelGrowAlg 7, 8, 9, 10, et enfin 11 par copier coller (Le bloc 6 étant le dernier dans cette application). On effacera ensuite les blocs MultCtlDelGrowAlg8, 9, 10.

A la fin, on obtient :

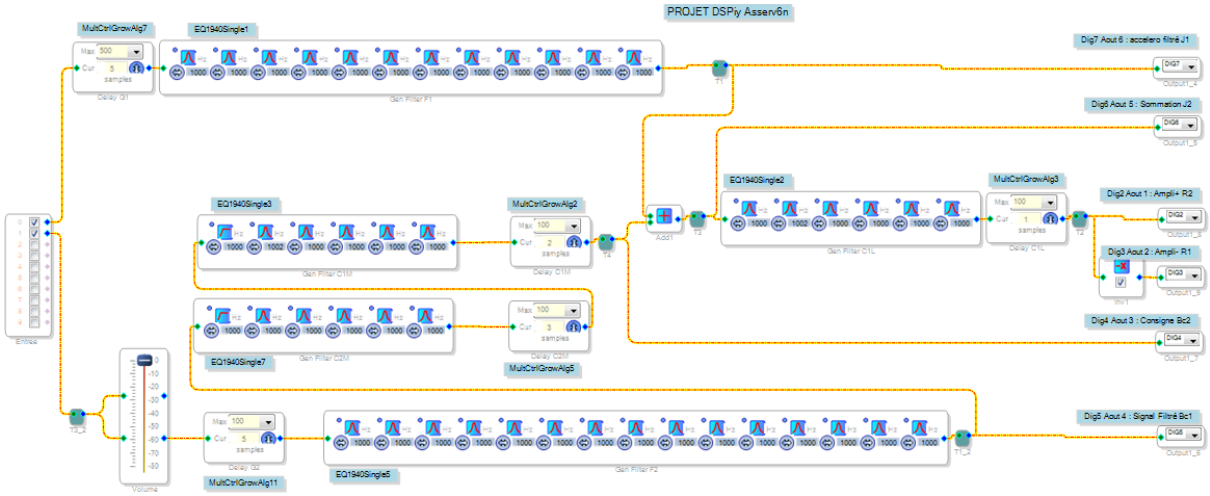
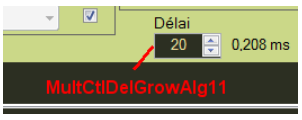


Figure 11 : Application Sigma Studio modifiée

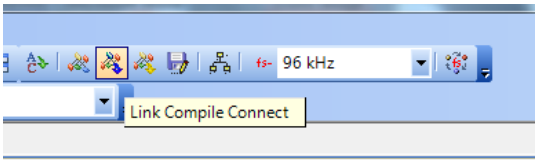
Remarques :

- Le potentiomètre est resté double, une sortie peut rester en l'air sans problème pour la compilation, par contre, les deux entrées doivent être raccordées, sous peine d'erreur à la compilation.
- L'entrée des signaux est devenue simple, sans sélecteur.
- Le crossover de la voie du haut a été placé en série avec celui de la voie du bas. Il aurait pu être supprimé.
- Les délais globaux ajoutés en tête des filtres Voie 1 et 2 sont en MultCtlDelGrowAlg 7 et 11. Ils ont été créés à partir de la version DStudio V4.22 beta 3.
  Ils apparaissent sur l'écran filtre si on les valide dans le fichier .dctl.



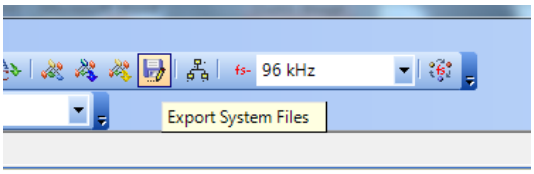
Compilations des liens :
Une fois l'application modifiée, faire :

- Save as asserv.dsppproj, dans un répertoire toto créé pour l'occasion.
- « Action / link compil connect » (ou bouton barre de menu).



Normalement, pas d'erreur, sinon, corriger!
(c'est souvent un dépassement de ressources ou un lien oublié entre deux opérateurs).

- "action / export system file" (ou bouton sur la barre de menu)



Le programme demande un nom pour la config (asserv par exemple, sans extension)
Sigma Studio va exporter un certain nombre de fichiers dont asserv.params dans le répertoire toto.

On vérifie dans le fichier « compiler_output.txt » que les ressources ne sont pas dépassées :

Number of instructions used (out of a possible 512 (en 96k)) = 409
Data RAM used (out of a possible 2048) = 898 (la place pour les tempos est prise ici)
Parameter RAM used (out of a possible 1024) = 180

Le fichier est dans le répertoire "…..\toto\IC1_asserv\net_list_out2\"

5 *Intégration de l'application dans Dstudio*

Lancer DStudio, peu importe l'application qui vient avec.

onglet "avancé", remplir :

- "selection param SigmaStudio" avec le « asserv.params » de l'export de Sigma Studio.

Le fichier « asserv.params » ne doit pas être déplacé de son répertoire d'origine (toto).

- "sélection paramètre µc" avec un uc-param.ucp qui va bien (ici uc-param_rc5.ucp modifié par mes soins parce que télécommande Philips).

- choisir le type d'appli DStudio (celle qui a servi de modèle et dont on veut utiliser les écrans, ici 2x1+1)

puis cliquer "créer appli", donner un nom (asserv par ex), qui va créer les fichiers asserv.apd et asserv.deq

Une fois créée, la nouvelle application est active, mais il est prudent de fermer DStudio et de le rouvrir, afin que tout soit bien initialisé pour cette nouvelle application.

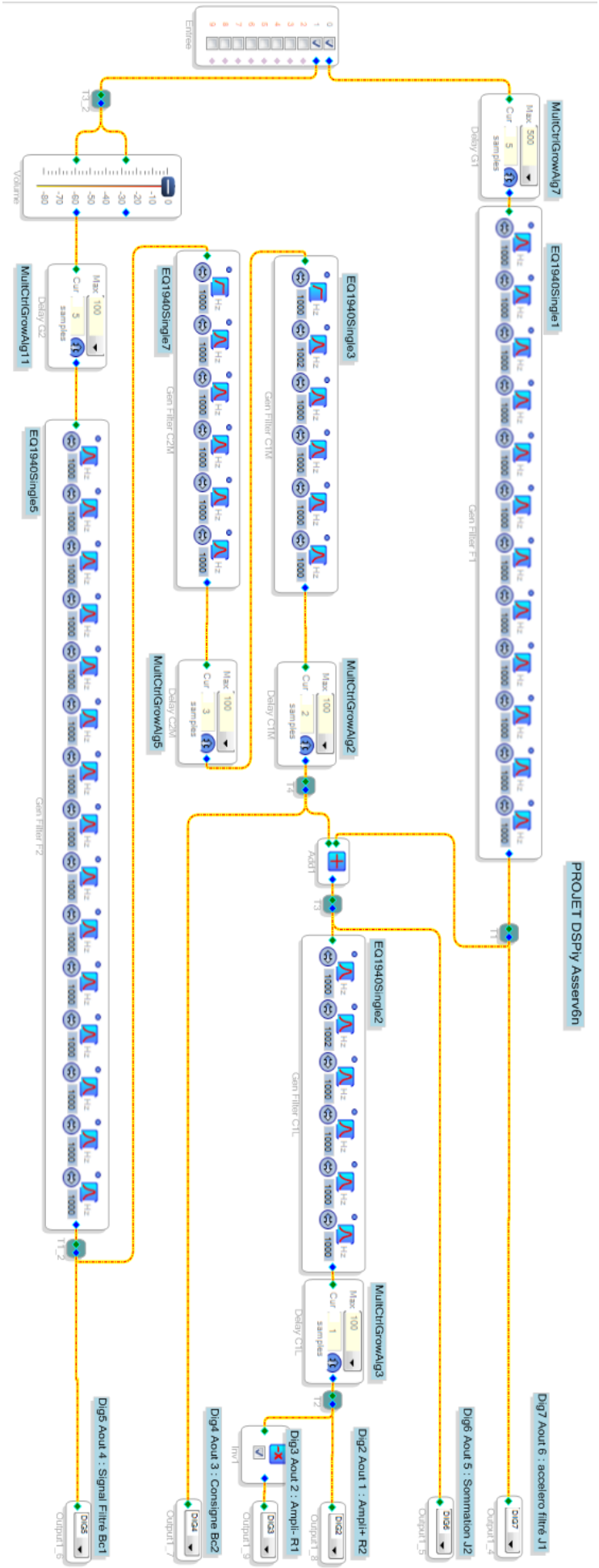
6 Annexes

6.1 Détail des paramètres dsp de l'application dans le fichier APD

parametre DSP

Gain1940AlgNS1 1 1	Potentiomètre
Gain1940AlgNS2 2 1	Potentiomètre
EQ1940Single10B1 3 -0,779838519633667	EQ1940Single1 (12 biquads)
:	
:	
EQ1940Single12A12 62 0	
EQ1940Single20B1 201 0,021620718376497	EQ1940Single2 (6 biquads)
:	
:	
EQ1940Single22A6 230 0,999222540061361	
EQ1940Single30B1 170 0,999672857810604	EQ1940Single3 (6 biquads)
:	
:	
EQ1940Single32A6 199 0	
MultCtrlDelGrowAlg3 231 31	Delai 3
EQ1940Single50B1 64 -0,779838519633667	EQ1940Single5 (15 biquads)
:	
:	
EQ1940Single52A15 138 0	
EQ1940Single70B1 139 0,999672857810604	EQ1940Single7 (6 biquads)
:	
:	
EQ1940Single72A6 168 0	
MultCtrlDelGrowAlg5 169 51	Delai 5
EQ1940Invert1gain 232 -1	Inverseur de la sortie symétrique
MultCtrlDelGrowAlg7 0 11	Delai global 7
MultCtrlDelGrowAlg2 200 41	Delai 2
MultCtrlDelGrowAlg11 63 21	Delai global 11

6.2 *Détail de l'application modifiée*



6.3 Affectation des blocs aux écrans DStudio

L'application DStudio est composée de 8 panneaux :

2 écrans Filtres de 1 panneau chacun : voie 1 et voie 2

2 écrans Crossover : Voie 1 et Voie 2, de 3 panneaux chacun : L M et H

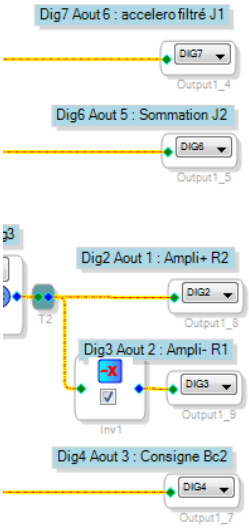
Les blocs « EQ1940Single x » correspondent à des ensembles de biquads, et à des panneaux de l'application DStudio.

Les blocs « MultCtrlDelAlg x » correspondent à des délais que l'on trouve dans les panneaux DStudios.

On a la répartition suivante :

Onglet	Voie	Panneau	Bloc de Biquads	Bloc Délai
Filtre	Left		EQ1940Single1	MultCtrlDelAlg7
Crossover	Left	Low	EQ1940Single2	MultCtrlDelAlg3
Crossover	Left	Medium	EQ1940Single3	MultCtrlDelAlg2
Crossover	Left	High	EQ1940Single4	MultCtrlDelAlg1
Filtre	Right		EQ1940Single5	MultCtrlDelAlg11
Crossover	Right	Low	EQ1940Single6	MultCtrlDelAlg6
Crossover	Right	Medium	EQ1940Single7	MultCtrlDelAlg5
Crossover	Right	High	EQ1940Single8	MultCtrlDelAlg4

6.4 Affectation des sorties

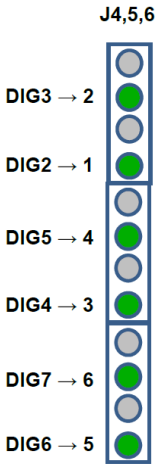


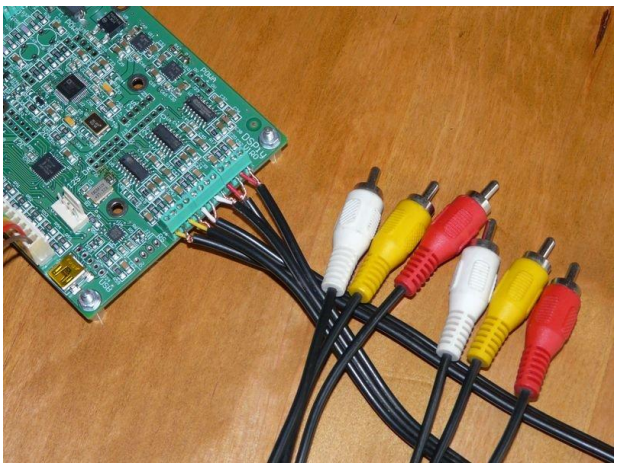
Les sorties du DSP sont nommées Dig 2 à Dig 7.

Celles-ci peuvent être affectées à n'importe quel bloc ou ensemble de bloc.

Ici, je les ai arrangées de façon à ce que ce soit simple au niveau câblage, sachant qu'on a les correspondances suivantes avec les points de sortie sur la carte via les connecteurs J4, J5 et J6:

Sortie DSP	Sortie Carte	Mon câblage
Dig2	Aout1	Ampli + / Rouge 2
Dig3	Aout2	Ampli - / Rouge 1
Dig4	Aout3	Consigne / Bc 2
Dig5	Aout4	Signal après filtre / Bc1
Dig6	Aout5	Sommation / Jaune 2
Dig7	Aout6	Accéléro après filtre / Jaune 1





6.5 Fichier des paramètres µC

Ce fichier est nécessaire pour créer une nouvelle application, et doit être renseigné dans l'onglet « avancé ». Il sert à configurer l'onglet config, et donc le comportement du DSPIY par rapport à son environnement :

Télécommande, triggers, rôle des BP, etc….

Pour avoir une config conformes aux autres applications que l'on a on peut soit :

utiliser le fichier fourni (uc_param_008.ucp) et recharger une config à partir d'une autre application après la création de la nouvelle application (import config).

avoir son propre fichier de config, que l'on peut obtenir simplement en créant soi-même son fichier uc_param_xxx.ucp.

Création d'un fichier .ucp personnalisé :

charger le fichier .apd d'une application dont la config vous convient dans le bloc note de Windows.

Sélectionner les lignes à partir de

parametre uc
D01_VOLUME 1 -30
D01_VOLUME_CTL 2 2
D01_VOLUME_MAX 3 0……..
……
D01_ADR_VOL3 67 0
D01_ADR_VOL4 68 0
D01_PRESET_NAME 100 Analog-1-2
D01_APP1452 69 1

En résumé, on sélectionne toutes les lignes préfixées **D01_** sauf le dernier **D01_APP1452 (et paramètre uc)**

puis copier coller dans un nouveau fichier texte que l'on nommera : mes_param-a-moi.ucp

(Seule l'extension .ucp est importante)

On pourra alors se servir de ce nouveau fichier pour les applications que l'on va créer, qui seront tout de suite paramétrées « comme on aime ».

6.6 Fichier uc_param_011.ucp

Début .ucp
D01_VOLUME 1 -55
D01_VOLUME_CTL 2 2
D01_VOLUME_MAX 3 0
D01_VOLUME_POWER_ON 4 0
D01_BOUTONP1 5 0
D01_BOUTONP2 6 0
D01_BOUTONP3 7 3
D01_BOUTONP4 8 2
D01_BOUTONP5 9 1
D01_BOUTONP6 10 0
D01_BOUTONP7 11 0
D01_BOUTONP8 12 0
D01_ISO_IN 15 13
D01_RELAI 16 0
D01_ISO_OUT 17 0
D01_LED1 18 0
D01_LED2 19 0
D01_LED3 20 0
D01_LED4 21 0
D01_LED5 22 0
D01_BUS 24 0
D01_MUTE 25 1
D01_IR_CODE 27 1
D01_IR_ADR 28 1
D01_IR_SUB_ADR 29 0
D01_IR_PRESET_PLUS 30 16
D01_IR_PRESET_MOIN 31 17
D01_IR_PRESET_1 32 0
D01_IR_PRESET_2 33 1
D01_IR_PRESET_3 34 2
D01_IR_PRESET_4 35 3
D01_IR_PRESET_5 36 4
D01_IR_PRESET_6 37 5
D01_IR_PRESET_7 38 6
D01_IR_PRESET_8 39 7
D01_IR_PRESET_9 40 8
D01_IR_MUTE 41 20
D01_IR_VOLUME_PLUS 42 18
D01_IR_VOLUME_MOIN 43 19

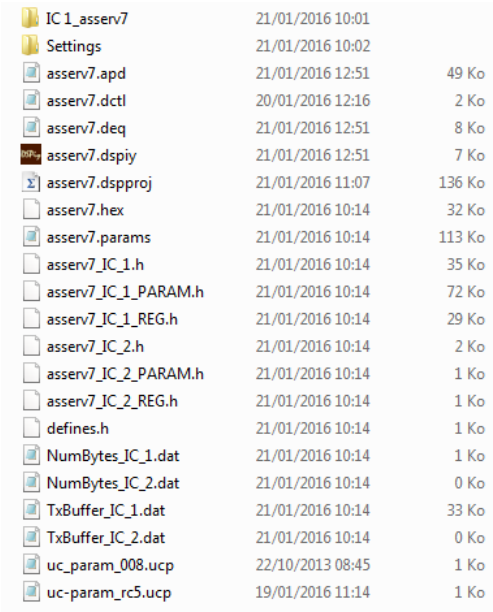
Suite .ucp
D01_IR_OFF 44 21
D01_IR_ON 45 21
D01_IR_BALANCE_R 46 56
D01_IR_BALANCE_L 47 81
D01_NWAITCODE 48 0
D01_STEPVOLUME 49 0
D01_IR_NLACTION 50 50
D01_LED2STB 51 0
D01_STBMODE 52 0
D01_AUTOSTB 53 0
D01_ADR_MUX_LR 54 255
D01_ADR_MUX 55 255
D01_PRESET_LINK 56 0
D01_LEVEL_DETECT_1 57 78
D01_LEVEL_DETECT_2 58 81
D01_ECRAN_CONTRASTE 59 50
D01_DSP 60 1
D01_PRESET_POWER_ON 61 0
D01_INPUT_SOURCE 62 0
D01_ECRAN 63 2
D01_EXTENSION 64 0
D01_ADR_VOL1 65 255
D01_ADR_VOL2 66 255
D01_ADR_VOL3 67 0
D01_ADR_VOL4 68 0
D01_ADR52_VOL1H 70 0
D01_ADR52_VOL1L 71 0
D01_ADR52_VOL2H 72 0
D01_ADR52_VOL2L 73 0
D01_ADR52_SAFEH 74 0
D01_ADR52_SAFEL 75 0
D01_LED4STB 76 0
D01_NO_SOURCE_01 77 0
D01_NO_SOURCE_02 78 0
D01_IR_SOURCE_MOINS 79 0
D01_IR_SOURCE_PLUS 80 0
D01_IR_CONTRAST 81 0
D01_PRESET_NAME 100 0
Fin .ucp

6.7 Contenu du répertoire de travail "Toto"

Après avoir créé le répertoire « toto » et fait les actions suivantes :

- Save as asserv7.dsppproj dans le répertoire « toto »
- Action / link compil connect
- Action / export system file
- Puis avoir créé l'application dans DStudio à l'aide de l'onglet avancé avec asserv7.params et un .ucp (on pourra le copier dans le répertoire « toto » pour avoir tout sous la main)

On obtient la liste des fichier suivante :



IC 1_asserv7	21/01/2016 10:01	
Settings	21/01/2016 10:02	
asserv7.apd	21/01/2016 12:51	49 Ko
asserv7.dctl	20/01/2016 12:16	2 Ko
asserv7.deq	21/01/2016 12:51	8 Ko
asserv7.dspiy	21/01/2016 12:51	7 Ko
asserv7.dsppproj	21/01/2016 11:07	136 Ko
asserv7.hex	21/01/2016 10:14	32 Ko
asserv7.params	21/01/2016 10:14	113 Ko
asserv7_IC_1.h	21/01/2016 10:14	35 Ko
asserv7_IC_1_PARAM.h	21/01/2016 10:14	72 Ko
asserv7_IC_1_REG.h	21/01/2016 10:14	29 Ko
asserv7_IC_2.h	21/01/2016 10:14	2 Ko
asserv7_IC_2_PARAM.h	21/01/2016 10:14	1 Ko
asserv7_IC_2_REG.h	21/01/2016 10:14	1 Ko
defines.h	21/01/2016 10:14	1 Ko
NumBytes_IC_1.dat	21/01/2016 10:14	1 Ko
NumBytes_IC_2.dat	21/01/2016 10:14	0 Ko
TxBuffer_IC_1.dat	21/01/2016 10:14	33 Ko
TxBuffer_IC_2.dat	21/01/2016 10:14	0 Ko
uc_param_008.ucp	22/10/2013 08:45	1 Ko
uc-param_rc5.ucp	19/01/2016 11:14	1 Ko

Le fichier « compiler_outpout.txt contenu dans IC1_asserv7\net_list_out2 :

Analog Devices Graphical Compiler Tool for Sigma DSP build date = 2008
Done Reading Nodelist ...
Done reading file,
Done Reading Parameter list ...
################## Summary ########################
Number of instructions used (out of a possible 1024) = 503 *Attention, c'est écrit 1024, mais la limite est 512 en 96kHz*
Data RAM used (out of a possible 2048) = 1194
Parameter RAM used (out of a possible 1024) = 233
Files written:
program_data.dat - load file for downloading code using ADI loader
hex_program_data.dat - load file for downloading code using microcontroller
spi_params.dat - file of parameter values for each instance, used by gen_spi program to make download file
spi_map.dat - Parameter RAM locations for each schematic instance
trap.dat - lists the values to enter in the trap registers to output a signal to the data-capture output pin.

6.8 Changement de la fréquence de l'application

L'intérêt de passer de 96kHz à 48kHz, est de bénéficier du double d'instructions, et de délais deux fois plus longs.

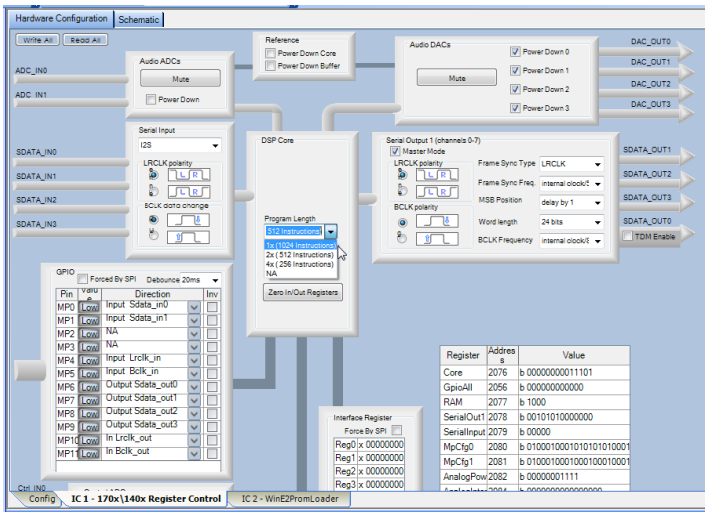
L'inconvénient est de perdre en résolution évidemment, mais pour une application telle que l'asservissement d'un subwoofer, où les fréquences ne dépassent pas 1kHz, c'est largement suffisant.

Dans Sigma Studio :

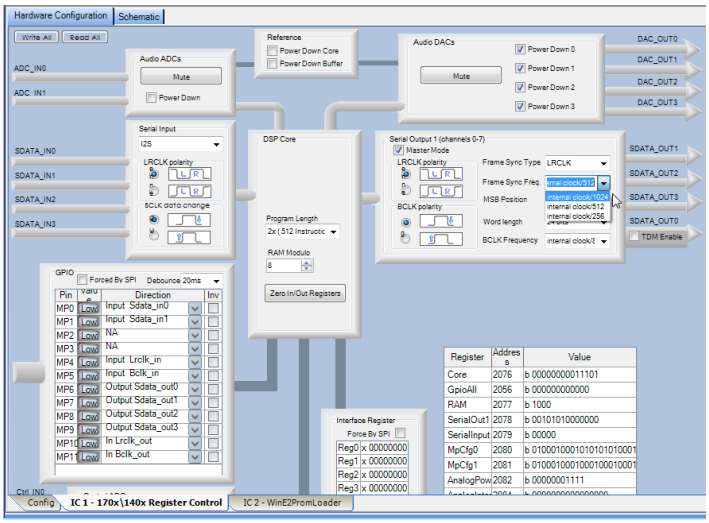
On quitte l'écran «« Schematic » (le dessin de l'appli), pour aller dans l'écran « Hardware configuration » par l'onglet en haut du schéma, et « IC1-170x\140x Register Control » par les onglets du bas de l'écran :

Une fois dans cet écran, on modifie :

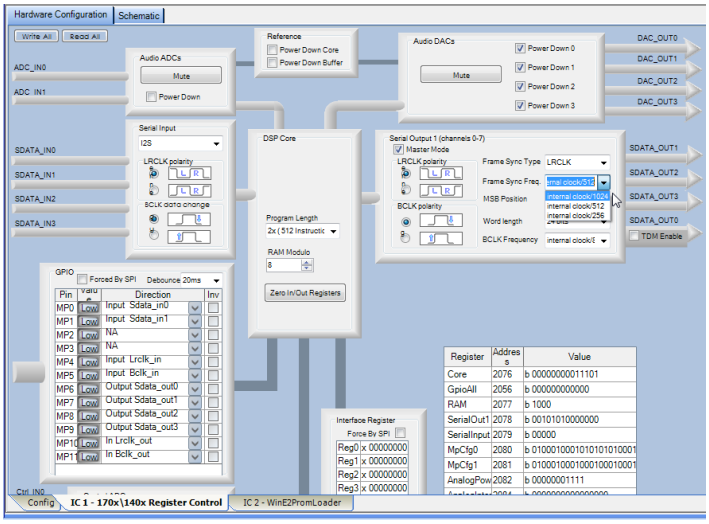
Dans le bloc « DSP Core »
« program length » en le passant de
512 à 1024



Dans « Serial output »
« Frame Sync Freq » en la passant de
Internal clock/512 à
Internal clock/1024

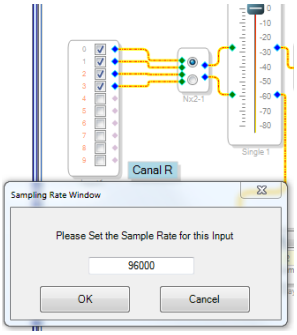


« BCLK Frequency » en la passant de
Internal clock/8 à
Internal clock/16

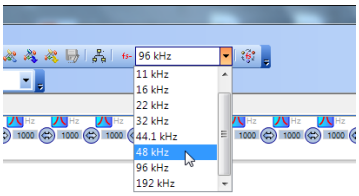


On revient sur l'écran « Schematic » (le dessin de l'appli),

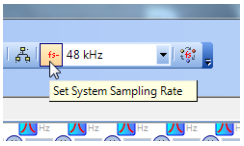
On règle la fréquence d'échantillonnage des entrées
(clic droit sur le bloc des entrées, et selection de
« set sampling rate »



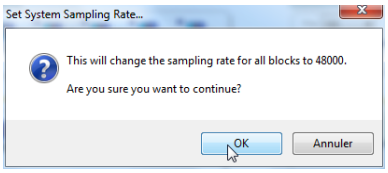
On passe la fréquence à 48kHz, en haut de l'écran,



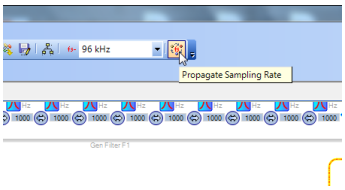
On valide la valeur en cliquant sur fs-…



…et sur OK dans la fenêtre qui apparait



Enfin, on clique sur « Propagate Sampling Rate »



Puis on reprend la séquence normale de création de l'application Sigma Studio avec « Save as » puis « Link compil », en haut de la page 10

Lors de la création de l'application DStudio (onglet avancé de DStudio, voir paragraphe 5, page 11), il faudra cocher la case « 48kHz ».