

# DSP Contrôle TP

Transcription : Hyperion

10 Mars 2022

## SUJET

**OBJET :** L'objet de ce contrôle TP est d'implémenter et de mettre en oeuvre un ensemble de filtres numériques de manière à restituer les trois composantes d'un signal composite. Comme pour les TP précédents, nous utiliserons les deux périphériques (ePWM et ADC).

**RAPPORT :** Le rapport est à rendre en version papier exclusivement et en fin de séance. Vous trouverez en annexe 2 une check liste des éléments à fournir dans le rapport. Ce dernier sera très probablement noté par un autre enseignant que votre encadrant, il doit donc être suffisamment explicite pour que vous obteniez la note optimale et donc comporter des éléments de vérifications (code, graphes,...). Écrire que le programme fonctionne n'apporte rien sans élément tangible.

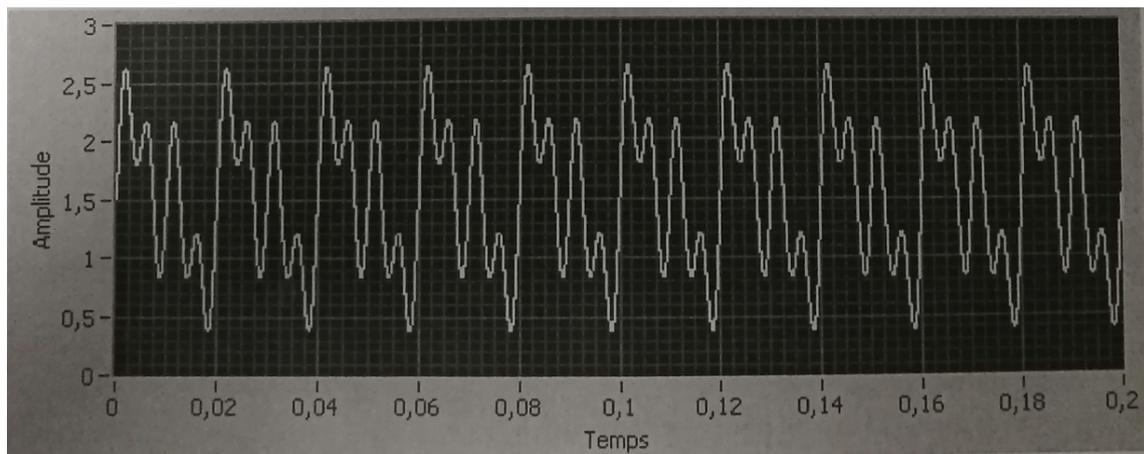
**NOTATION :** La notation comprendra, le rapport de contrôle et le fonctionnement des différents points du travail à réaliser.

**SUPPORT :** Code Composer Studio, Guide de référence du module ePWM, Guide de référence du module ADC, documentation de la carte eZDSP.

**MATÉRIEL :** Carte d'évaluation eZDSP 2808 de Spectrum Digital.

## INTRODUCTION

Pour ce TP nous allons utiliser un exécutable réalisé à partir de Labview et de la carte d'acquisition reliée au PC par port USB. L'application DSP permet de générer sur la sortie AO0 du module USB6221 un signal périodique somme de trois signaux sinusoïdaux de fréquence et d'amplitude différentes. Votre objectif sera de reproduire chacune des composantes du signal original sur les sorties GPIO0-EPWM1A, GPIO2-EPWM2A et GPIO3-EPWM3A.



Le signal provenant de la sortie analogique 1 de la carte PC sera connectée à l'entrée *ADCINA0* du DSP.

## TRAVAIL A EFFECTUER

Vous disposez d'un rappel des opérations à effectuer au paragraphe suivant. Les caractéristiques des filtres sont données en annexes. Faires contrôler par l'enseignant les différentes étapes de votre travail.

1. Configurer le module ePWM pour générer un signal à la fréquence de **200 KHz** (choisir le mode **incrément** et générer par défaut un rapport cyclique de 50%). Paramétrer l'entrée analogique 1 de manière à effectuer la conversion analogique du signal d'entrée **provenant du module NI**. Visualisez le signal obtenu sur un graphique à partir d'un tableau de 100 éléments (adaptez les fréquences pour un meilleur rendu). Joignez cet élément à votre rapport.
2. Calculer la relation de récurrence et programmer le filtre passe-bas dont l'équation vous est fournie en annexe. Vous utiliserez de préférence, mais pas obligatoirement, une représentation en virgule fixe pour les coefficients du filtre. Filtrez le signal d'entrée, stockez le résultat dans un tableau de 100 éléments puis visualisez le signal filtré avec Code Composer. Ensuite, générez le signal filtré sur la sortie EPWM1A.
3. Procédez comme pour le point 2 avec le filtre passe-haut et recréez le signal filtré sur la sortie EPWM1A en remplacement du précédent signal. Calculez les caractéristiques du filtre analogique à utiliser pour obtenir une fréquence de coupure voisine de 5.5 KHz.
4. Procédez comme pour les points 1 et 2 pour le filtrage passe-bande et recréez le signal filtré sur la sortie EPWM1A.
5. Modifiez la configuration des modules ePWM de manière à sortir les 3 signaux filtrés sur les sorties respectives EPWM1A, EPWM2A et EPWM3A.

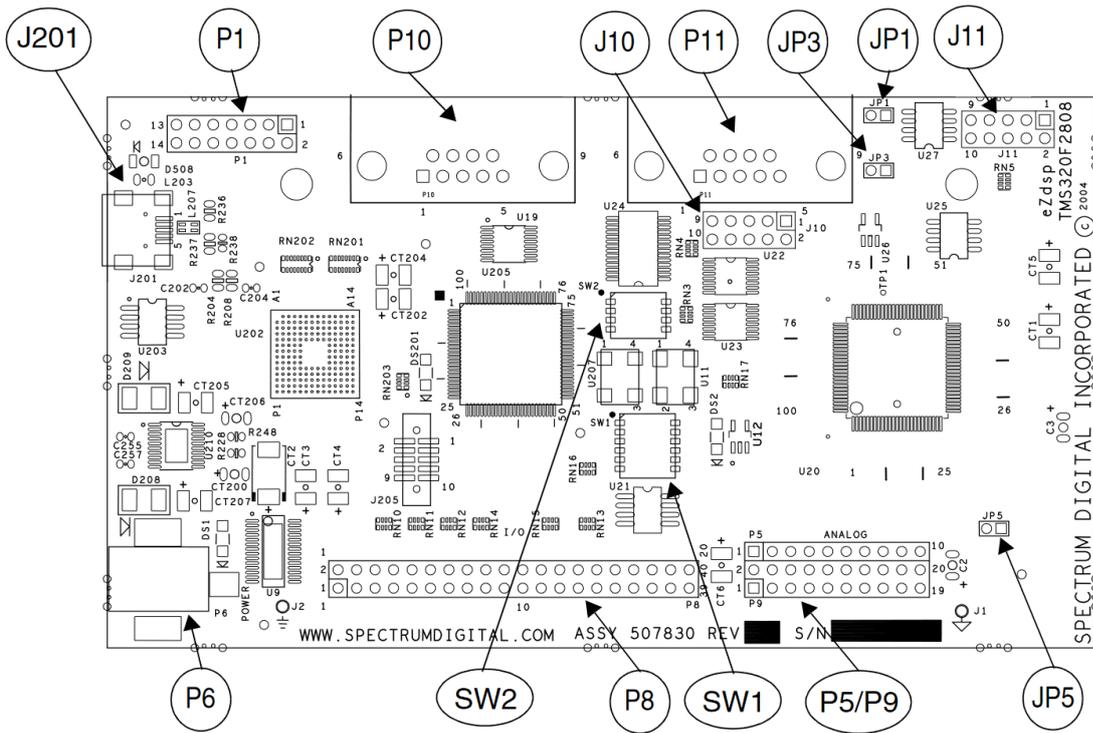


Figure 2-3, eZdsp™ F2808 Connector and Switch Positions

Figure 1: Positionnement des connecteurs et cavaliers de la carte eZDSP

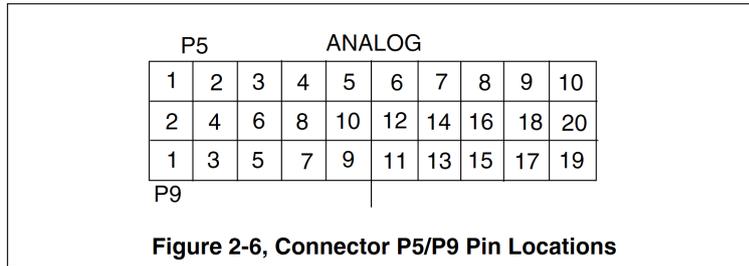


Figure 2-6, Connector P5/P9 Pin Locations

Figure 2: Définition des connecteurs P9 et P5

Table 4: P5/P9, Analog Interface Connector

P5 Pin #	Signal	P9 Pin #	Signal	P9 Pin #	Signal
1	ADCINB0	1	GND	2	ADCINA0
2	ADCINB1	3	GND	4	ADCINA1
3	ADCINB2	5	GND	6	ADCINA2
4	ADCINB3	7	GND	8	ADCINA3
5	ADCINB4	9	GND	10	ADCINA4
6	ADCINB5	11	GND	12	ADCINA5
7	ADCINB6	13	GND	14	ADCINA6
8	ADCINB7	15	GND	16	ADCINA7
9	ADCREFM	17	GND	18	VREFLO *
10	ADCREFP	19	GND	20	No connect

Figure 3: Définition de l'interface analogique

## RAPPEL DES OPÉRATIONS

Programmez le fichier "DSP280x\_Adc.c" de manière à récupérer sur la voie 0 du module ADC le signal analogique composite.

Dans la fonction "ADCINT\_ISR(void)" du fichier "DSP280x\_DefaultIsr.c" vous devez stocker dans un tableau de 100 éléments les valeurs converties lors de chaque séquence de conversion.

Lancez le débogue comme pour les TP précédents et affichez le graphique correspondant au signal mesuré et joignez-le à votre rapport (utilisez de basses fréquences pour l'affichage graphique).

L'objectif est maintenant d'implémenter un ensemble de trois filtres passe-bas, passe-bande et passe-haut dont les fréquences de coupures sont respectivement : 100 Hz, 500 Hz à 1200 Hz et 4500 Hz pour chaque de ces filtres de manière à extraire les composantes de base du signal composite.

Placez un filtre RC de fréquence de coupure de l'ordre de **200 Hz** entre la masse et la broche 9 (signal PWM) pour visualiser le signal de 100 Hz (cf. Figure 1, 5, 6)

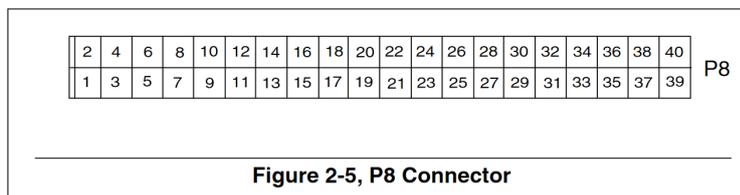


Figure 4: Définition du connecteur P8

Visualisez les signaux filtrés sur l'oscilloscope (joindre une image du résultat).

Conclure.

**Table 3: P4/P8, I/O Connectors**

<b>P8 Pin #</b>	<b>P8 Signal</b>	<b>P8 Pin #</b>	<b>P8 Signal</b>
1	+3.3V/+5V/NC *	2	+3.3V/+5V/NC *
3	MUX_GPIO29	4	MUX_GPIO28
5	GPIO14	6	GPIO20
7	GPIO21	8	GPIO23
9	GPIO0	10	GPIO1
11	GPIO2	12	GPIO3
13	GPIO4	14	GPIO5
15	GPIO27	16	GPIO6
17	GPIO13	18	GPIO34
19	GND	20	GND
21	GPIO7	22	GPIO15
23	GPIO16	24	GPIO17
25	GPIO18	26	GPIO19
27	MUX_GPIO31	28	MUX_GPIO30
29	MUX_GPIO11	30	MUX_GPIO8
31	MUX_GPIO9	32	MUX_GPIO10
33	GPIO22/GPIO24	34	GPIO25
35	GPIO26	36	GPIO32
37	GPIO12	38	GPIO33
39	GND	40	GND

Figure 5: Définition de l'interface numérique

## ANNEXE 1

### FILTRE PASSE-BAS :

Fonction de transfert :

$$H(p) = \frac{Y(p)}{X(p)} = \frac{1}{1 + \tau p}$$

Avec,

$$\tau = \frac{1}{\omega_c} = \frac{1}{2\pi f_c} = \frac{1}{2\pi \times 100}$$

Donnez la transformée en  $z$  de la fonction de transfert  $H(p)$  en utilisant la transformée bilinéaire

$$p = \frac{2}{T_e} \frac{1 - z^{-1}}{1 + z^{-1}} \text{ avec } T_e = 5\mu s$$

Donnez l'équation de récurrence correspondante sous la forme :

$$y(n) = C_1 x(n) + C_2 x(n-1) + C_3 y(n-1)$$

En appliquant ces filtres au signal "Buffer" ( $= x(n)$ ) on obtient trois signaux filtrés "Buffer\_Filtre\_n" ( $\equiv y(n)$ ) qui correspond à chacune des composantes du signal d'entrée.

### FILTRE PASSE-HAUT :

Fonction de transfert :

$$H(P) = \frac{Y(p)}{X(p)} = \frac{\tau p}{1 + \tau p}$$

Avec,

$$\tau = \frac{1}{\omega_c} = \frac{1}{2\pi f_c} = \frac{1}{2\pi \times 4500}$$

Donnez la transformée en  $z$  de la fonction de transfert  $H(p)$  en utilisant la transformée bilinéaire

$$p = \frac{2}{T_e} \frac{1 - z^{-1}}{1 + z^{-1}} \text{ avec } T_e = 5\mu s$$

Donnez l'équation de récurrence correspondante.

### FILTRE PASSE-BANDE :

Le filtre passe-bande sera réalisé par l'association de deux filtres en série. Un premier filtre passe-haut de fréquence de coupure 1 kHz suivi d'un passe-bas de 3 kHz dont vous donnerez les transformées en  $z$  ainsi que les équations de récurrence.

La sortie du filtre passe-haut sera injectée en entrée du filtre passe-bas.

## ANNEXE 2

### CHECK LISTE DE RAPPORT

Calculs PWM : horloges & PRD		
Relevé oscillo PWM		
Code ADC		
Graphe ADC		
Calcul Coef Pbas		
Code Coef Pbas		
Graphe Pbas (*CCS)		
Calcul Coef Phaut		
Code Coef Phaut		
Graphe Phaut (*CCS)		
Calcul Coef Pbande		
Code Coef Pbande		
Graphe Pbande (*CCS)		
Signaux filtrés sur oscillo (Pb, Ph, PB)		

Les relevés difficiles à incorporer (ex: oscilloscope) peuvent être remplacés par une signature de l'enseignant sur le code source attestant de leur existence.

CCS : Code Composer Studio

Note : Afficher un minimum de graphes afin d'optimiser leur rendu.