

Projet d'électronique numérique  
visualiations musicales sur écran VGA

LESTANDI NATHAN      ECOCHARD FLORENT

11 janvier 2017

## Table des matières

<b>Introduction</b>	<b>2</b>
<b>1 Cahier des charges</b>	<b>2</b>
<b>2 Architecture</b>	<b>2</b>
<b>I DSP : analyse du son</b>	<b>3</b>
<b>1 Filtre passe-bas</b>	<b>3</b>
1.1 modélisation et dimension du filtre . . . . .	3
1.2 simulation . . . . .	4
1.2.1 test bench . . . . .	4
1.2.2 Simulation et interprétation . . . . .	4
<b>2 Filtre passe-haut</b>	<b>5</b>
<b>3 Filtre passe-bande</b>	<b>6</b>
<b>4 Bloc DSP</b>	<b>6</b>
<b>II Visualisations, affichage</b>	<b>8</b>
<b>1 L'"oscilloscope"</b>	<b>8</b>
<b>2 Le "spectrogramme"</b>	<b>8</b>
<b>3 Le bargraph</b>	<b>8</b>
<b>Bilan</b>	<b>8</b>
<b>Bibliographie</b>	<b>9</b>

# Introduction

A la suite de plusieurs séances de TPs d'approfondissement sur le langage de description vhdl dans l'environnement vivado, nous avons eu à imaginer et mener à bien un projet étalé sur 7 semaines à faire en binôme. Le choix étant assez libre et n'ayant que peu de contrainte liées à l'énoncé du projet, nous avons choisi de faire varier l'affichage d'un écran en fonction de la musique. Dans un souci d'efficacité et compte tenu de la charge de travail nous avons décidé d'utiliser le site Trello afin d'optimiser la répartition des tâches et de fixer des deadlines. Ce site nous a permis de suivre l'avancement de notre binôme en sauvegardant de manière ordonnée le travail, ainsi que de garder une structure claire et de répertorier de nombreux éléments nécessaires à la rédaction d'un rapport.

## 1 Cahier des charges

Le cahier des charges imposé implique une machine d'état ainsi qu'une interface utilisateur.

## 2 Architecture

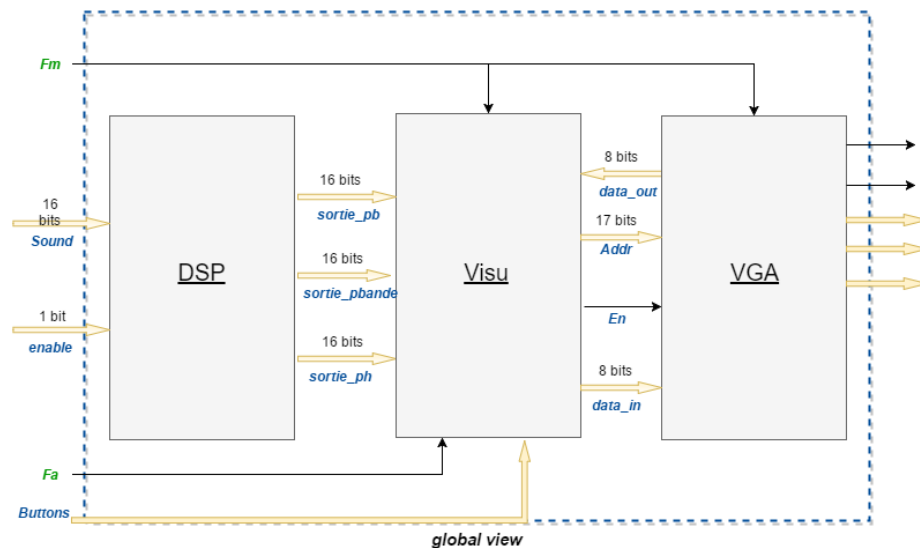


FIGURE 1 – Schéma global du système

Dans la figure 1 page 2, ...

# Première partie

## DSP : analyse du son

### 1 Filtre passe-bas

#### 1.1 modélisation et dimension du filtre

Afin de réaliser un filtre passe-bas le moins encombrant possible et le plus facile à paramétrer, nous avons utilisé un filtre exponentielle du première ordre qui s'écrit de la manière suivante :

$$y(n) = (1 - \epsilon)y(n - 1) + x(n) \times \epsilon, \epsilon \in [0, 1] \quad (1)$$

ce filtre correspond à un filtre RC, il ne possède qu'une seule variable :  $\epsilon$ , la fréquence de coupure est définie par :

$$fc = \frac{-\ln(1 - \epsilon)}{2\pi} \times f_{\text{fonctionnement}} \quad (2)$$

Cependant sous cette forme un déphasage est induit dans les hautes fréquences. Pour remédier à ce problème, nous allons moyenner l'entrée avec l'échantillon précédent soit :

$$y(n) = (1 - \epsilon)y(n - 1) + \frac{(x(n) + x(n - 1)) \times \epsilon}{2} \quad (3)$$

Sachant que nous travaillons sur une architecture numérique, il faut éviter les multiplications/divisions. Pour  $\epsilon$  nous avons choisi de prendre des valeurs multiples de deux afin de limiter les opérations à des décalages. Pour déterminer les  $\epsilon$  de tous les filtres on prends comme fréquence de référence celle du enable qui dépend du module de gestion de la carte sd, soit 44100Khz. Un epsilon de  $2^{-4} = 0.625$  permettra de couper à environ 450Hz tout en laissant les opérations faites dans le filtre être de simple décalage.

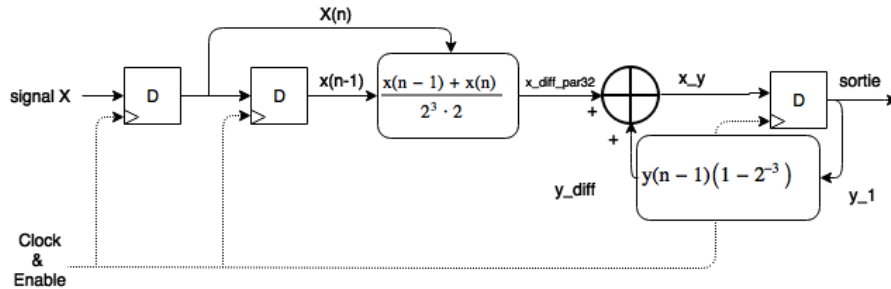


FIGURE 2 – schéma bloc passe-bas

On commence par synchronisé le circuit avec une bascule D, puis celle-ci serviront à obtenir les valeurs n-1 nécessaire aux calculs. les autres calculs seront fais dans des processus implicite.

Dans l'équation 3 on identifie  $f_{fonctionnement}$ . Dans notre modélisation du filtre, sans un signal enable, ce serait la clock qui fournirait cette fréquence. Hors on scrute aussi le signal enable pour activer les bascules, on a donc :

$f_{fonctionnement} = f_{enable} = 44100Khz$ , valeur qui est fournit par le module SD.

## 1.2 simulation

Nous avons décidé de préciser la simulation du passe-bas car c'est le bloc de base des autres filtres et comprendre son fonctionnement est indispensable.

### 1.2.1 test bench

Pour simuler ce bloc, nous avons créer un testbench basique. Pour rédiger la trame principale nous avons utiliser [?] puis nous avons complété le testbench de manière a testé le signal d'entrée , le reset, le signal enable et la sortie. Pour ça nous avons imposé des valeurs en entrée, puis nous avons confronté les résultat aux valeurs théoriques calculés à partir de la formule (3).

### 1.2.2 Simulation et interprétation

Voici une capture de la simulation obtenue : Sur celle-ci nous pouvons ob-

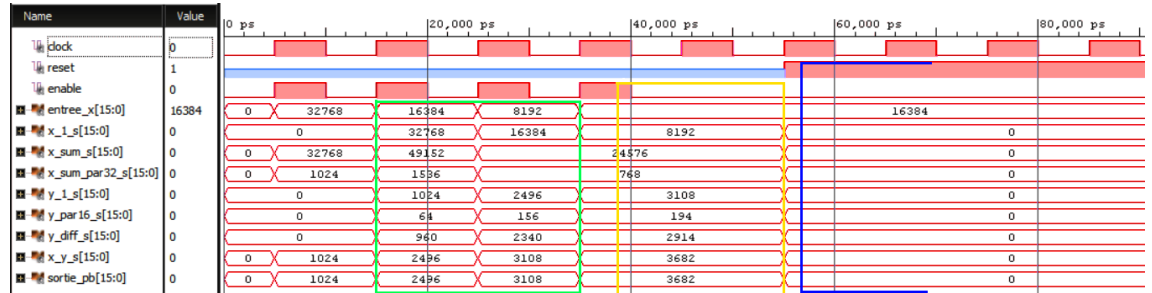


FIGURE 3 – schéma bloc passe-bas

server plusieurs chose, mais nous allons d'abords scruté les effets du reset et d'enable pour voir si ils répondent aux attentes.

Le signal reset est testé dans la partie la plus à droite délimité par le carré, on voit qu'il remet bien à 0 toute les sorties.

Le signal enable est lui testé dans le carré jaune, on voit que lorsqu'il est à zéro, le système ne varie pas. Ce qui est l'effet escompté puisque ce signal correspond à l'envoi en entrée d'un échantillon audio. Sans échantillon le filtre ne doit pas fonctionner.

Vérifions maintenant la fonction effectuée par le bloc, pour cela nous allons nous concentrer sur les 2 coups d'horloge dans le carré vert. Pour savoir la correspondance des signaux, se référer à la figure 2. On observe bien sur le premier coup d'horloge encadré que x-sum-par32 correspond bien à  $\frac{x(n)+x(n-1)}{2-5}$ , avec les valeurs binaire, un décalage de 5 vers la droite est bien effectué. De même y-diff est bien correspond à  $y(n-1) \times (1-\epsilon)$ . La sortie est bien égale à xy qui est la somme de x-sum-par32 et y-diff. Les décalages sont bien effectués entre deux coups d'horloge, ce qu'on peut vérifier en comparant avec le deuxième coup d'horloge dans le cadre vert : x-1 et y-1 sont bien x(n-1) et y(n-1).

## 2 Filtre passe-haut

Pour le passe-haut nous allons utiliser un principe instinctif suivant lequel, modéliser un passe-haut revient à soustraire au signal d'entrée sa valeur filtrée dans un passe-bas soit :

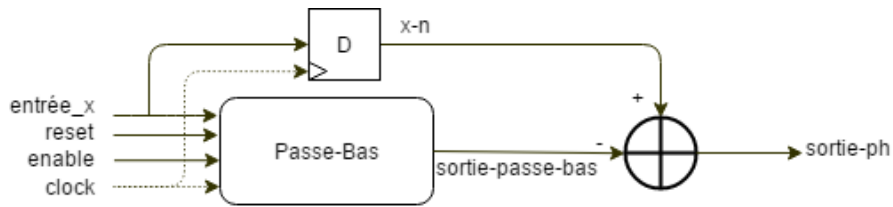


FIGURE 4 – schéma bloc passe-haut

On utilise une bascule D sur entrée-x afin de synchroniser celle-ci avec le bloc passe-bas. Nous avons fixé  $\epsilon=2^{-1}$  de manière à avoir une fréquence de coupure plus haute que le passe-bas soit environ 5Khz. Pour ce faire nous avons utilisé un generic : n qui correspond à la puissance d'  $\epsilon$  dans le filtre passe-bas. Nous vérifierons que le bloc a une réponse impulsionnelle correspondant à un passe- haut dans la figure [?]

### 3 Filtre passe-bande

Pour créer le passe-bande, nous avons mis en série le passe haut et le passe-bas soit :



FIGURE 5 – schéma bloc passe-bande

On utilise les blocs précédents en branchant la sortie directement sur celle du passe haut on garde ici le même epsilon dans le passe bas que dans sa version original, de même pour le passe haut, de cette manière on couvre l'intégralité du spectre du signal.

### 4 Bloc DSP

Le bloc DSP regroupe les parties précédentes , il envoie en continue les sortie des 3 filtres aux étages de visualisation.

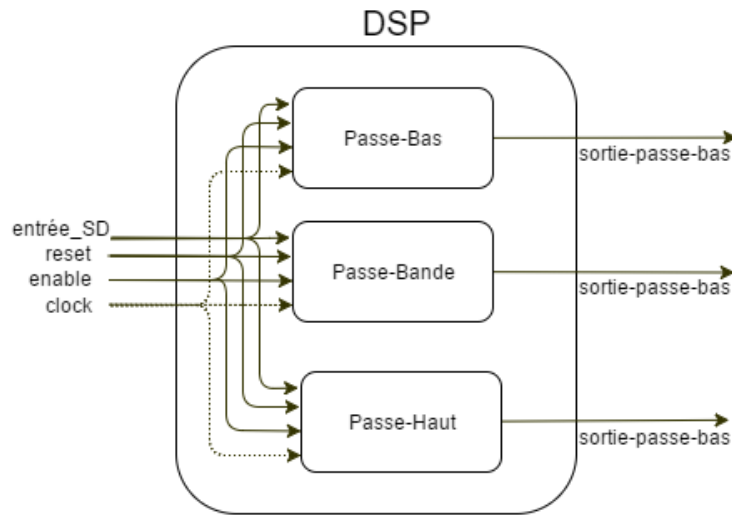


FIGURE 6 – schéma DSP

Afin de prouver le bon fonctionnement nous avons tester la réponse impulsionnelle de l'ensemble du bloc DSP. Pour ce faire nous avons fixé l'entrée à une valeur, ici 32768 (en binaire 1000000000000000) et nous avons regarder la sortie des filtres de manière analogique( waveform style -> analog) et on a :

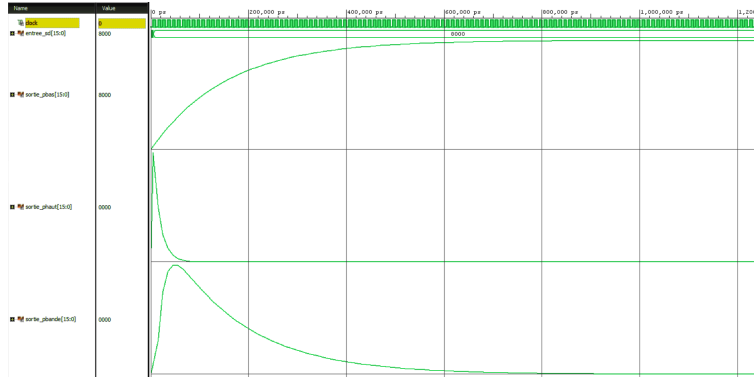


FIGURE 7 – réponse impulsionnelle bloc DSP

La sortie du passe-bas tend bien vers la valeur du signal lorsque celui-ci reste longtemps en entrée et inversement pour le passe-haut, il coupe lorsque le signal reste longtemps identique. Le passe bande est bien un compromis des deux en filtrant lorsque la fréquence est trop haute ou trop basse.

Nous avons mesuré les fréquences de coupure pour chacun de ces filtres en prenant pour fréquence de référence celle de la carte soit 100Mhz, on a alors  $\tau_{p-b}=150\text{ns}$  pour le passe-bas et  $\tau_{p-h}=15\text{ ns}$  pour le passe haut d'après la formule 3. Nous avons bien des valeurs constantes en sortie des filtres regardant leur réponses qui est constante à  $5\tau$  (75ns et 750ns).



Deuxième partie

## Visualisations, affichage

1 L'"oscilloscope"

2 Le "spectrogramme"

3 Le bargraph

Bilan

## Références

<https://tomroelandts.com/articles/low-pass-single-pole-iir-filter> <https://fiiir.com/>  
<http://www.dspguide.com/ch19/2.htm>