

Filière Systèmes industriels

Orientation Infotronics

Travail de bachelor Diplôme 2023

Cesalli Maxime

Audiophile 2-way splitter for active loudspeakers



Professeur
François Corthay



Expert
Dario Biner



Date de la remise du rapport
18.08.2023

Table des matières

Introduction	3
Description	3
Analyse.....	4
Protocole i2s.....	4
Décodeur i2s.....	5
Encodeur i2s.....	5
Architectures des filtres.....	6
Analyse de consommation ressources FPGA.....	8
Analyse se des résultats.....	8

Introduction

La plupart des haut-parleurs du marché sont passifs, cela signifie qu'un signal audio amplifié leur arrive directement et est ensuite divisé en fonction des fréquences par des filtres LC analogiques : un filtre passe bas pour les basses et un filtre passe haut pour les aigus. Les haut-parleurs actifs eux reçoivent individuellement chaque bande de fréquence préalablement filtrés puis amplifiés. Le fait de filtrer le signal avant l'amplification rend possible l'utilisation de filtres plus performants et complexes. Cela permet aussi de se débarrasser des distorsions produites par un filtre analogique. L'objectif de ce projet est de concevoir un diviseur audio à 2 voies (passe-bas et passe-haut complémentaire) basé sur des filtres FIR. Le circuit doit être implémenté à l'intérieur d'une FPGA, avec une entrée/sortie audio numérique I2S. Le système vas d'abord être implémenté avec des coefficients de filtre fixes, puis avec la possibilité de télécharger les coefficients dans la FPGA.

Description

Cahier des charges

- Implémenter un encodeur/décodeur i2s dans la FPGA et le tester sur le channel des basses du haut-parleur
- Implémenter un filtre passe bas digital utilisant différentes méthodes pour sa réalisation
- Implémenter le filtre passe haut complémentaire et faire le test avec les deux channels
- Proposer une méthode pour importer les coefficients du filtre dans la FPGA

Schéma bloc du système

Le signal audio analogique arrive à travers 2 connecteurs RCA. Un convertisseur analogique digital va ensuite échantillonner le signal a 96kHz (48kHz par channel), puis l'envoyer à la FPGA au travers du protocole i2s. La FPGA vas ensuite séparer ce signal en fonction des fréquences qui le compose pour créer un signal i2s contenant un channel aigu et un channel grave. Les données vont ensuite être reconverties en un deux signaux analogiques qui pourront ensuite être amplifiés séparément.

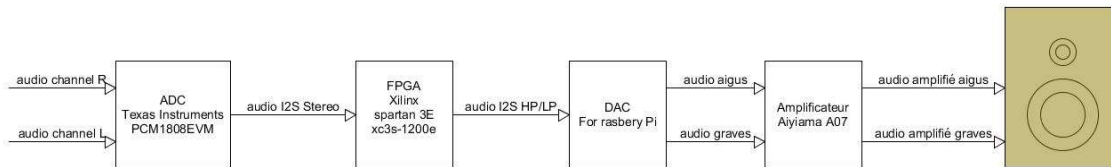


Figure 1: schéma block du système.

Schéma block architecture FPGA

Au sein de la FPGA le signal commence par être dés-sérialiser par le bloc i2s décodeur. Les deux channels gauche et droite sont ensuite moyennés pour former une donnée unique à filtrer. La donnée mono passe ensuite dans deux filtres un passe bas et une passe haut. Les données qui en sortent sont ensuite sérialisées et envoyées a un convertisseur digital/analogique.

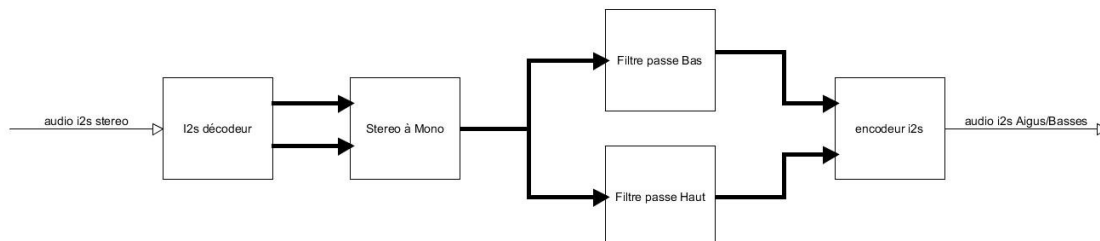


Figure 2 : Schéma block architecture FPGA

Analyse

Protocole i2s

La première étape afin de pouvoir manipuler les signaux audio transmis est de comprendre le Protocole qui les transporte. Le protocole I2S est une interface série synchrone utilisée principalement pour la transmission des données audio numériques. Il a été développé par Philips Semiconductor (maintenant NXP Semiconductors) et est largement adopté dans l'industrie audio. Le protocole I2S permet la transmission simultanée des données audio et des signaux de synchronisation, garantissant ainsi une reproduction précise du son. Le protocole I2S se compose de trois signaux principaux : la ligne de données (Data line), la ligne de synchronisation des horloges (Bit Clock line) et la ligne de synchronisation des canaux (Word Select line).

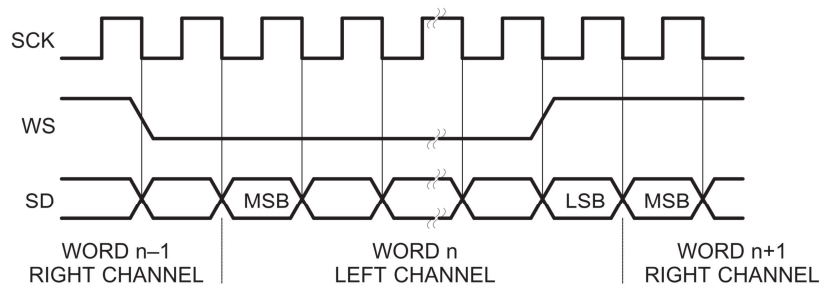


Figure 3: chronogramme d'un transfert de données i2s

Comme le montre la figure X, une des particularités du transfert i2s est que les données binaires ne changent pas synchronisées avec le Word Select mais décalées d'un coup de clock, ce qui a posé un certain nombre de problèmes lors du développement de ce projet.

Décodeur i2s

Le décodeur i2s parallélise les données sérielles qu'il reçoit. Pour ce faire un compteur se décrémente à chaque flanc montant du bit clock. La valeur de ce compteur va ensuite se faire multiplexer en fonction de la valeur du word select, décalé de 1 coup de clock pour correspondre à la spécification i2s. La valeur comptée va ensuite déterminer à quelle place la valeur de la data line va être enregistrée, par exemple si le compteur est à 24, la valeur de la data line va être lue et enregistré comme étant le bit 24 de notre donnée. Une fois nos deux données complétées ce qui correspond dans ce cas au flanc montant de notre Word Select décalé une impulsion est créée pour indiquer aux blocs suivants que des nouvelles données sont arrivées et qu'il est possible de les lire.

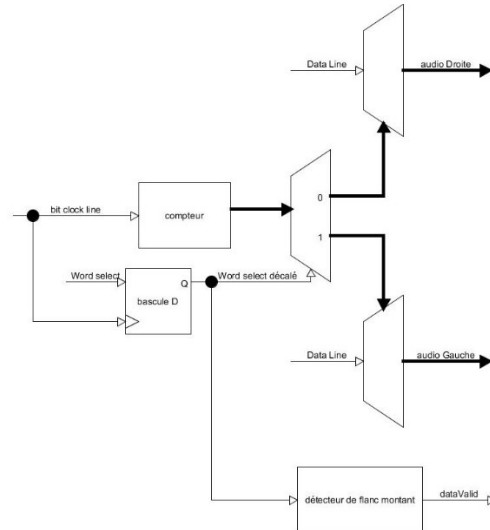


Figure 4: schéma interne du décodeur i2s

Pour compléter ce bloc de désérialisation il faut encore ajouter un registre qui met les données à sa sortie que lorsque dataValid est à 1, car tout le reste du temps les données de font mettre à jour et ont donc des valeurs transitoires.

Encodeur i2s

L'encodeur i2s prend des données parallèle et les convertis en données sérielles. Il reçoit les informations de timing depuis l'encodeur, cela permet de pouvoir s'adapter à n'importe quelle fréquence de transmission, tant qu'elle ne s'approche pas trop de la fréquence du clock interne de la FPGA. Le principe de fonctionnement est l'exact inverse du décodeur. On y retrouve un décompteur qui va sélectivement parcourir chaque bit de notre donnée afin de le transmettre sur la data line. Le Word Select lui va se faire décaler à l'aide d'une bascule D puis, venir sélectionner laquelle des données, droite ou gauche ou, dans notre cas aigus ou grave vas se faire sérialiser.

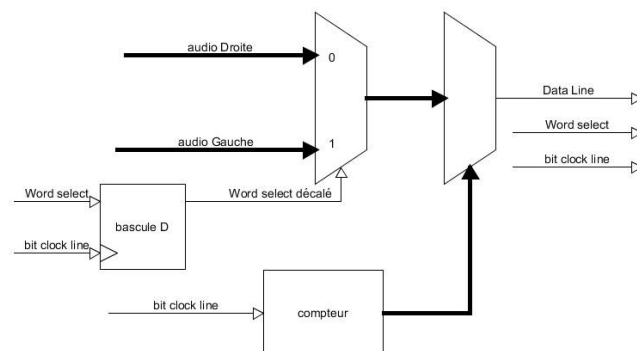


Figure 5: Schéma block encodeur i2s

Architectures des filtres

Parallèle

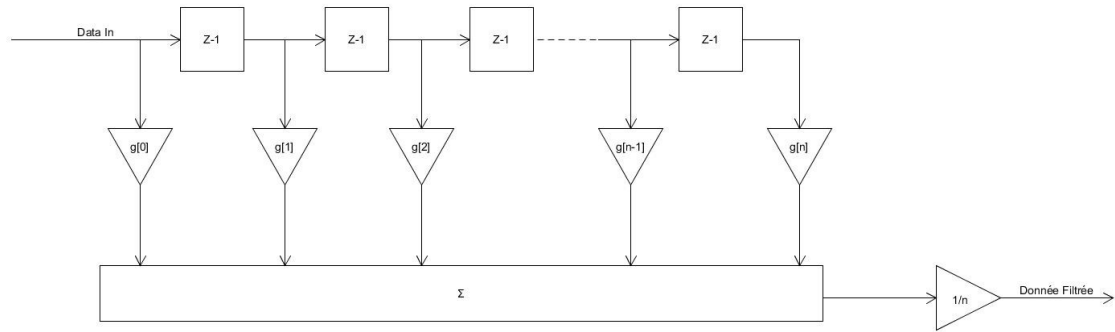


Figure 6: schéma de principe filtre FIR parallèle

Parallèle avec symétrie

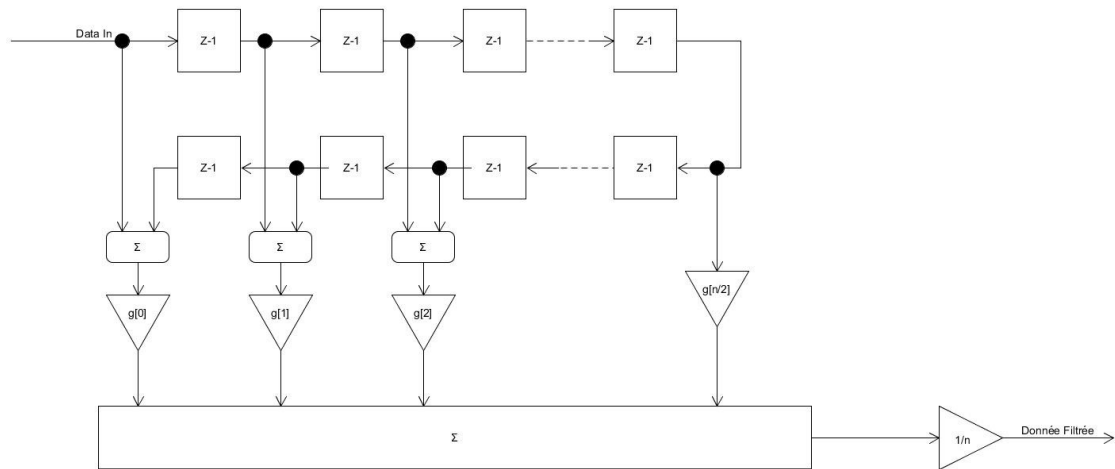


Figure 7: schéma de principe filtre FIR parallèle symétrique

Série

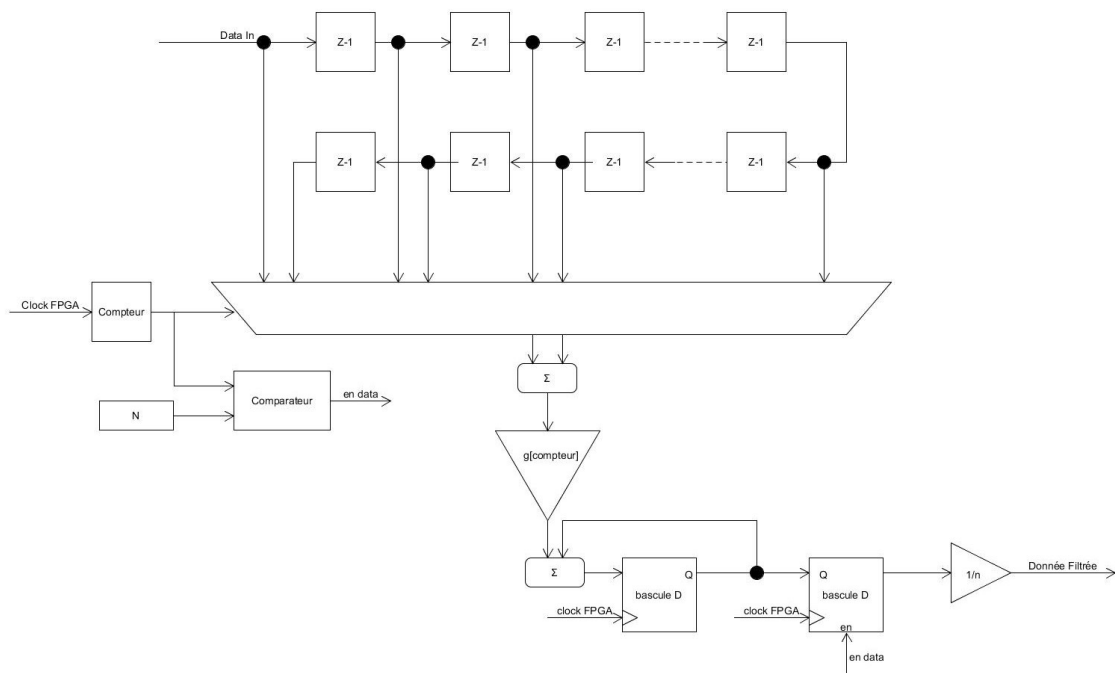


Figure 8 : schéma de principe filtre FIR série

Ce rapport est l'original remis par l'étudiant.
Il n'a pas été corrigé et peut donc contenir des inexactitudes ou des erreurs.

Série parallèle

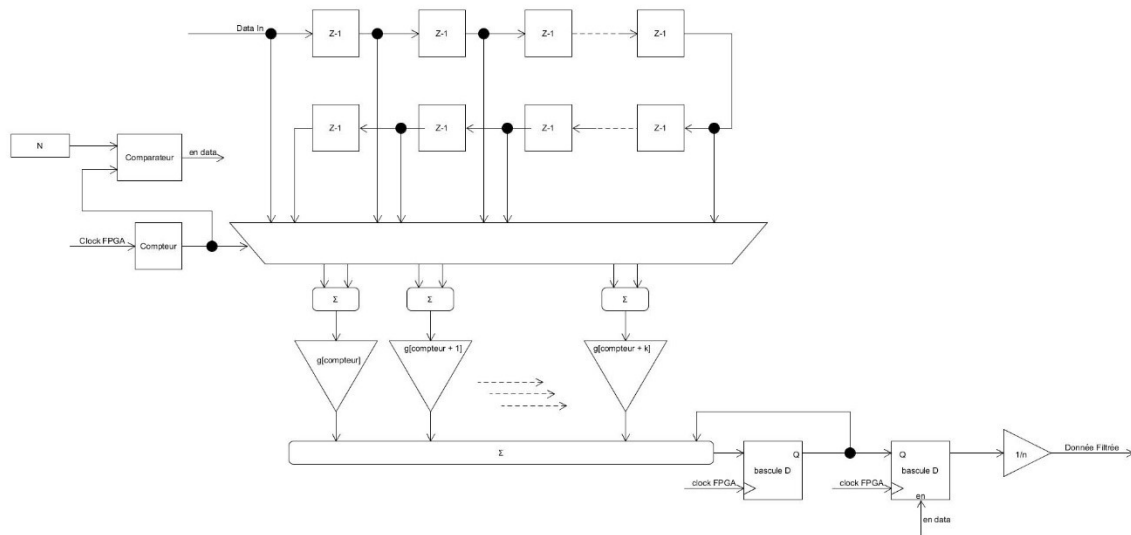


Figure 9: schéma de principe filtre FIR parallèle / série

Regroupement passe haut passe bas avec le même registre a décalage

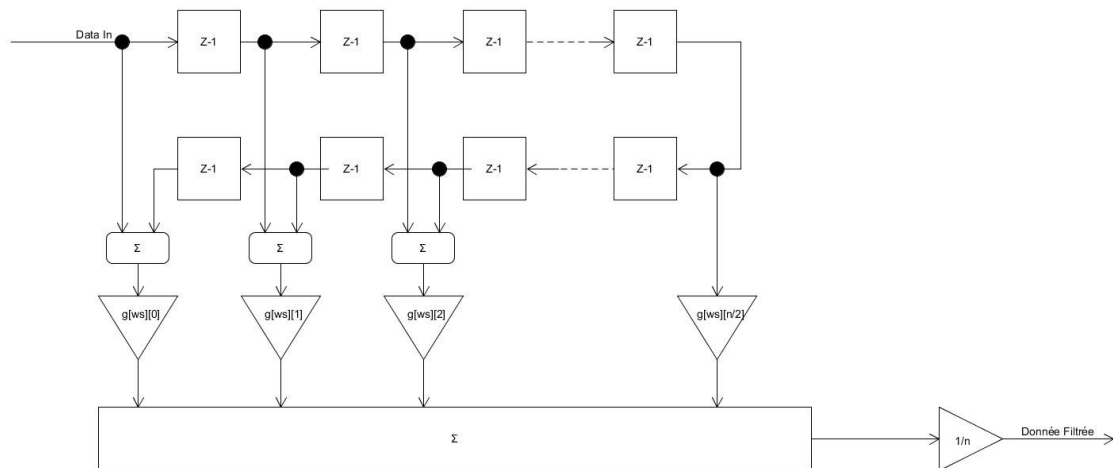


Figure 10: schéma de principe filtre FIR passe haut/passe bas

Analyse de consommation ressources FPGA

xc3s-500e

Architecture :	Parallèle		Parallèle avec symétrie	
	ordre 30		ordre 55	
	utilisation :		utilisation :	
	flip flops	12%	flip flops	21%
	LUTs	61%	LUTs	66%
	Sclices	81%	Sclices	94%
	ordre 38		ordre 63	
	utilisation :		utilisation :	
	flip flops	15%	flip flops	24%
	LUTs	83%	LUTs	79%
	Sclices	102%(4752/4656)	Sclices	104%(4855/4656)

xc3s-1200e

Architecture :	Parallèle		Parallèle avec symétrie	
	ordre 33		ordre 104	
	utilisation :		utilisation :	
	flip flops	7%	flip flops	20%
	LUTs	34%	LUTs	67%
	Sclices	45%	Sclices	94%
	ordre	63	ordre 116	
	utilisation :		utilisation :	
	flip flops	12%	flip flops	22%
	LUTs	74%	LUTs	74%
	Sclices	97%	Sclices	99%(8670/8672)
	ordre	69		
	utilisation :			
	flip flops	14%		
LUTs	83%			
Sclices	100%(8702/8672)			

Analyse se des résultats

Les premiers résultats de crossover obtenus sont les suivants, ils ont été réalisés avec deux filtres parallèle symétriques de 55 taps chacun, ne partageant pas leur registres a décalage.



Figure 11: diagramme de baude filtre passe bas 55 taps

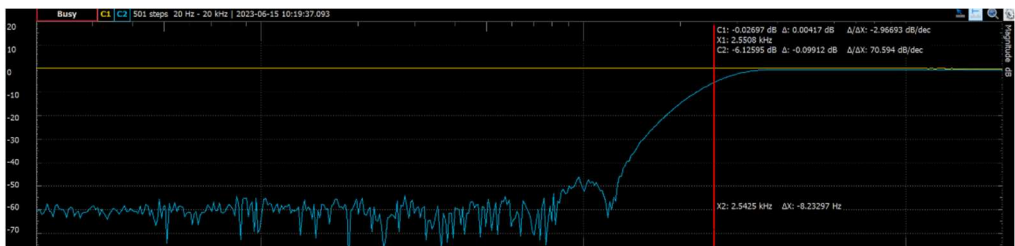


Figure 12: diagramme de baude filtre passe haut 55 taps

Ce rapport est l'original remis par l'étudiant.
Il n'a pas été corrigé et peut donc contenir des inexactitudes ou des erreurs.