REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITEYAHIA FARES DE MEDEA



Faculté de **T**echnologie

Mémoire de fin d'études de Master

Présenté pour l'obtention du Diplôme
de Master en Génie Électrique
Spécialité : Système de Télécommunication

Implémentation d'un banc de filtre dédié à un implant cochléaire sur circuit FPGA

Présenté par :Proposé et Dirigé par :

HACHEMI AsmaaM. ESSAID Billel

HARABI Khaoula

Promotion 2019

Dédicaces

Je dédie ce modeste travail à :

Ma mère «L'avenir d'un enfant est l'œuvre de sa mère», pour son amour, son soutien, et ses précieux conseils, Tu as toujours été pour moi une source d'inspiration et de bonheur.

Mon père pour ses conseils, sa rigueur dans mon éducation, son amour du travail bien fait, son honnêteté, et tous les sacrifices consentis pour mon éducation.Merci pour ce que vous avez fait pour moi, Que dieu vous procure bonne santé et longue vie.

Mon cher frère AbdElhalim qui s'est sacrifié pour que je puisse achever mes études.

A mes chers sœurs ; FtimaAlzahra et Hadjira.

Ma sœur et mon cher amie Wafaa

Mes oncles et toute la famille HACHEMI et BEN KAIDA.

A Notre respectueux promoteur Mr ELSSAID Bilel. Et a Mon Binôme Khaoula.

Mes chers amis «khaoula, asmaa, sihem, karima, lamia, rachida, sara» que j'aime tant.

Amis d'étude préférés «amira, yasmine, sara»

A tous ceux qui m'aiment, m'ont encouragé et ont cru en moi.

HACHEMI Asmaa

Dédicaces

Ma Mére, Mon Pére

Vous représentes pour moi le symbole de la bonté par excellence, la source de tendresse

Et l'exemple du dévouement qui n'a pas cessé de m'encourages et de prier pour moi.

Soyez surs que je continuerai mon chemin.

Je vous dédie ce travail en témoignage de mon profond amour.

Puisse dieu, le tout puissant, vous préserver et vous accorder santé , longue vie et bonheur.

A messœurs : Sihem, Sara, Ikram ,hassiba

A mes frére : Zohire, Ilyas

En témoignage de l'attachement, de lamour et de l'affection que jeporte pour vous.

A tout les membres de ma famille «HARABI et AMRAOUI » petits et grands

A Notre respectueux promoteur Mr ELSSAID Bilel. Et a Mon Binôme Asmaa.

A tous mes amis de proches : Asmaa, Asmaa, Rachida ,Karima , Lamia,Aamira, Yasmine, Sara

Dieu merci à nous Donner force et volonté pour achever cet humble travail.

Nous remercions aussi, de tout cœur, tous ceux qui ont

contribué, de prés ou de loin, à la réalisation de ce travail.

Et, nous n'oublions pas de remercier tous les enseignants

qui ont suivi notre formation durant ces cinq années.

Enfin, nous tenons à remercier les membres de jury.

HARABI Khaoula

REMERCIEMENT

Nous remercions tous d'abord le bon Dieu, le tout puissant de nous avoir armé de force et de courage pour mener à terme ce projet de fin d'études.

Notre profonde gratitude s'adresse à Monsieur « ESSAID BILLEL » encadreur de ce mémoire pour ses conseils fructueux et pour son aide précieuse qui nous a conduits à concrétiser ce travail.

Nos remerciements à tous les enseignants qui ont contribués à notre formation, et ainsi tous les gens de prés et de loin, surtout l'étudiant MAHDJOUBI Abderrahman, qui ont aidé à l'élaboration de ce sujet et à tous nos professeurs et travailleurs de l'université YAHIA FARES -MEDEA- Ainsi que Tous les membres de jury qui nous ont fait l'honneur d'examiner notre travail.

Et enfin, que nos chers parents et familles, et bien avant tout, trouvent ici l'expression de nos remerciements les plus sincères et les plus profonds en reconnaissance de leurs sacrifices, aides, soutien et encouragement afin de nous assurer cette formation de Master dans les meilleures conditions.

الملخص:

الهدفمنهذاالمشروعهودراستومحاكاةإستراتيجيةتحفيزقوقعةالأذنACEباستخدامبنكترشيحيتكونمن 22 مرشحًا،منأجلتنفيذبنكالترشيحعلىدارةمنطققابلةللبرمجةFPGA. لتنفيذهذاالعمل،استخدمنابرامجMatlabوModelsimو XILINX ISE.

الكلمات المفتاحية:زراعةالقوقعة, إستراتيجيةACE,مجموعةالمرشحات, تطبيق على FPGA.

Résumé:

L'objectif de ce projet est d'étudier et simuler la stratégie de stimulation cochléaire ACE en utilisant un banc de filtres constitués de 22 filtres, afin d'implémenter le banc de filtres sur un circuit logique programmableFPGA.Pour réaliser ce travail nous avons utilisé les logiciels Matlab, Modelsim et XILINX ISE.

Mots-clés: implant cochléaire, stratégie ACE, banc de filtres, Implémentation sur FPGA.

Summary:

The objective of this project is to study and simulate the ACE cochlear strategy using a filter bank consisting of 22 filters, to implement the filter bank on a programmable logic circuit FPGA.For this work we used the software Matlab, Modelsim and XILINX ISE.

Keywords: cochlear implant, ACE strategy, filter bank, FPGA implementation.

Sommaire

LISTE DES FIGURES ……..I
LISTE DES TABLEAUX ……...III
LISTE DES ACRONYMES ……...V
INTRODUCTION GENERALE 1
CHAPITRE I : L'OREILLE ET LES IMPLANTS COCHLEAIRES
1. INTRODUCTION..... 2
2. ANATOMIE DE L'OREILLE..... 2
2.1. L'oreille externe ……..... 3
2.2. L'oreille moyenne ……..... 3
2.3. L'oreille interne ……..... 4
3. LA SURDITE 5
3.1. Types de surdité 5
3.1.1. Les surdités de transmission..... 5
3.1.2. Les surdités de perception……..... 5
3.1.3. Surdités mixtes..... 5
3.2. Degrés de surdité..... 6
4.L'IMPLANT COCHLEAIRE..... 6
4.1.Historique 6
4.2.Composition de l'implant cochléaire 7
4.2.1.La partie externe..... 8
4.2.2.La partie interne..... 8
4.3.Fonctionnement de l'implant cochléaire …..... 8
4.3.1. La partie externe..... 9
4.3.2.La partie interne..... 9
4.4.Avantages et inconvénients des appareils auditifs..... 10
5. LES STRATEGIES DE STIMULATION COCHLEAIRE..... 10
5.1.La stratégie SPEAK..... 10
5.2.La stratégie ACE..... 11
5.3.La stratégie CIS (Continuons InterleavedSampling)..... 12
5.4. La stratégie HiRes120..... 13
5.5. Les stratégies n-de-m..... 14
6. BANC DE FILTRES..... 14
7. LES CIRCUITS LOGIQUES PROGRAMMABLES.....16

7.1. Historique des circuits logiques programmables.....17
7.2. Ressources matérielles de circuits logiques programmables..... 17
7.3. La technologie FPGA.....17
7.3.1. Les étapes du flot de conception.....19
7.3.2. Le langage VHDL..... 19
8. CONCLUSION..... 21
CHAPITRE II : ETUDE ET SIMULATION DE LA STRATEGIE ACE
1. INTRODUCTION..... 22
2. LA STRATÉGIE ACE (ADVANCED COMBINATION ENCODER)..... 22
2.1.Principe de fonctionnement de la stratégie ACE..... 22
2.2. Le rôle de la stratégie ACE.....23
3.DIFFERENTS TYPES DES FILTRES..... 23
3.1.Les filtres numériques..... 23
3.1.1.Filtres à réponse impulsionnelle finie (filtres RIF)..... 24
3.1.2.Filtres à réponse impulsionnelle infinie (filtres RII)..... 24
3.2.Les filters analogiques..... 24
4.FILTRE PASSE-BANDE..... 25
Caractéristiques des filtres passe-bande..... 25
5.FILTREIDÉAL..... 26
6.REALISATION DU BANC DE FILTRES..... 26
7.SIMULATION DE LA STRATEGIE ACE EN UTILISANT LE BANC DE FILTRES REALISE..... 28
8.CONCLUSION..... 33
CHAPITRE III: SIMULATION ET IMPLEMENTATION DU BANC DE FILTRES EN MODELSIM
1. INTRODUCTION.....34
2. QUANTIFICATION DES FILTRES..... 34
3. GENERATION DU CODE VHDL DU BANC DE FILTRES..... 36
4. GENERATION DE « TEST BENCH » DU BANC DE FILTRES..... 36
5. SIMULATION DU PROGRAMME VHDL PAR MODELSIM..... 37
6. LOGICIEL XILINX ISE..... 44
6.1. Création d'un projet..... 44
6.2. Chargement du code VHDL du banc de filtres sous ISE..... 45
6.3. Synthèse et implémentation du projetISE..... 46
6.4. Verification du circuit FPGA.....53

7. CONCLUSION..... 56
CONCLUSION GENERALE..... 57
BIBLIOGRAPHIE..... 58

LISTE DES FIGURES

Figure I.1. Vue d'ensemble des 3 parties de l'oreille..... 2

Figure I.2. Schéma de l'oreille externe..... 3

Figure I.3. Schéma de l'oreille moyenne..... 4

Figure I.4. Schéma de l'oreille interne..... 4

Figure I.5. Implant cochléaire..... 6

Figure I.6. Représentation des parties externe et interne de l'implant cochléaire..... 8

Figure I.7. Partie interne d'un implant cochléaire..... 8

Figure I.8. Principe de fonctionnement d'un système d'implant cochléaire..... 9

Figure I.9. Le diagramme de stratégie de parler..... 11

Figure I.10. Schéma de la stratégie ACE..... 12

Figure I.11. Diagramme de la stratégie CIS..... 13

Figure I.12. Schéma fonctionnel de la stratégie de stimulation HiRes120..... 14

Figure I.13. Découpage en 4 bandes..... 15

Figure I.14. Banc de filtre à 4 voies..... 15

Figure I.15. Banc de filtres à M canaux (a) banc d'analyse (b) banc de synthèse..... 16

Figure I.16. Architecture générale d'un FPGA..... 18

Figure I.17. Les étapes du flot de la conception..... 19

Figure II.1. Les caractéristiques d'un filtre passe-bande..... 23

Figure II.2. La réponse en amplitude du banc de filtres..... 23

Figure II.3. Les sorties des filtres 1, 2, 3 et 4 sélectionné..... 24

Figure II.4. Les sorties des filtres 5, 6, 7 et 8 sélectionné..... 25

Figure II.5. Enveloppes des filtres 1, 2, 3 et 4 sélectionné..... 25

Figure II.6. Enveloppes des filtres 5, 6, 7 et 8 sélectionné..... 26

Figure II.7. Le signal à modulé..... 26

Figure II.8. Les signaux modulés de la stratégie ACE des filtres1, 2, 3 et 4..... 27

Figure II.9. Les signaux modulés de la stratégie ACE des filtres 5, 6, 7 et 8..... 27

Figure II.10. Une partie zoomé des signaux modulés de la stratégie ACE des filtres 1, 2, 3 et 4.. 28

Figure II.11. Une partie zoomé des signaux modulés de la stratégie ACE des filtres 5, 6, 7 et 8.. 28

Figure III.1. Réponse en amplitude après quantification du 1er filtre..... 32

Figure III.2. Le résultat de la simulation du 1er filtre obtenu par Matlab..... 33

Figure III.3. Le résultat de la simulation du 1er filtre obtenu par Modelsim..... 34

Figure III.4. Le résultat de la simulation du 2eme filtre obtenu par Matlab..... 34

Figure III.5. Le résultat de la simulation du 2eme filtre obtenu par Modelsim..... 34

Figure III.6. Le résultat de la simulation du 3eme filtre obtenu par Matlab..... 34

Figure III.7. Le résultat de la simulation du 3eme filtre obtenu par Modelsim..... 35

Figure III.8. Le résultat de la simulation du 4eme filtre obtenu par Matlab..... 35

Figure III.9. Le résultat de la simulation du 4eme filtre obtenu par Modelsim..... 35

Figure III.10. Le résultat de la simulation du 5eme filtre obtenu par Matlab..... 35

Figure III.11. Le résultat de la simulation du 5eme filtre obtenu par Modelsim..... 36

Figure III.12. Le résultat de la simulation du 6eme filtre obtenu par Matlab..... 36

Figure III.13. Le résultat de la simulation du 6eme filtre obtenu par Modelsim..... 36

Figure III.14. Le résultat de la simulation du 7eme filtre obtenu par Matlab..... 36

Figure III.15. Le résultat de la simulation du 7eme filtre obtenu par Modelsim..... 37

Figure III.16. Le résultat de la simulation du 8eme filtre obtenu par Matlab..... 37

Figure III.17. Le résultat de la simulation du 8eme filtre obtenu par Modelsim..... 37

Figure III.18. Le résultat de la simulation du 1er filtre obtenu par Modelsim..... 38

Figure III.19. Le résultat de la simulation du 2eme filtre obtenu par Modelsim..... 38

Figure III.20. Le résultat de la simulation du 3eme filtre obtenu par Modelsim..... 38

Figure III.21. Le résultat de la simulation du 4eme filtre obtenu par Modelsim..... 38

Figure III.22. Le résultat de la simulation du 5eme filtre obtenu par Modelsim..... 39

Figure III.23. Le résultat de la simulation du 6eme filtre obtenu par Modelsim..... 39

Figure III.24. Le résultat de la simulation du 7eme filtre obtenu par Modelsim..... 39

Figure III.25. Le résultat de la simulation du 8eme filtre obtenu par Modelsim..... 39

Figure III.26. Choix d'implémentation..... 44

Figure III.27. Création d'un nouveau projet..... 44

Figure III.28. Les bibliothèques et les entités et les architectures des filtres..... 45

Figure III.29. Ports d'entrée et de sortie..... 46

Figure III.30. La vue « Hierarchy »..... 46

Figure III.31. Processus de l'implémentation..... 47

Figure III.32. Ports d'entrée et de sortie dans le circuit FPGA.....48

Figure III.33. Les ports d'entrée (une partie zoomée)..... 49

Figure III.34. Les ports de sortie (une partie zoomée).....49

Figure III.35. Circuit FPGA de 1ere filtre..... 51

Figure III.36. Circuit de FPGA global..... 51

Figure III.37. Les différentes slices sur FPGA (une partie zoomée)..... 52

Figure III.38. Le circuit interne du « SLICE_X48Y105»..... 52

Figure III.39. Carte De Développement XILINX FPGA Spartan-3..... 53

Figure III.40. Processus de l'implémentation..... 53

Figure III.41.L'outil iMPACT..... 54

Figure III.42. Schéma de vérification le circuit FPGA..... 54

LISTE DES TABLEAUX

Tableau II.1. Les bandes fréquentielles des filtres**.....** 21

Tableau III.1. Sommaire des ressources utilisées dans le FPGA**.....** 47

Tableau III.2. Sommaire des ressources utilisées dans le FPGA (suite)**.....** 48

LISTE DES ACRONYMES

A

ACE: Advanced Combinational Encoder

AGC: Automatic Gain Control

C

CIS: Continuous Interleaved Sampling

D

DSP: Digital Signal Processor

DF: forme directe

E

EPLD: Electrically Programmable Logic Device

F

FIFO: First In First Out

FFT : transformée de Fourier rapide

FBMC: Filtre Bank based Multi Carrier

FPGA: Field Programmable Gate Array

G

GE: General Electric

I

ISE: Integrated Software Environment

IBM: International Business Machines Corporation

L

LGF : fonction de gain logarithmique

LB: blocs logiques

LUT: Look Up Table

M

MOS: Metal oxide semiconductor

N

NVMem: mémoire vive avec batterie de secours

P

PLD: Programmable LogicDevice

R

RIF : réponse impulsionnelle finie

RII : réponse impulsionnelle infinie

RAM: Read Only Memory

T

Tb: test bench

TI: Texas Instruments

U

Ucf: user constraints file

V

VHDL: VHSIC Hardware Description Language

INTRODUCTION GENERALE

Depuis plusieurs années L'implant cochléaire a été développé et commercialisé avec un succès indéniable, Où il a changé la vie de milliers de personnes souffrant de surdité sévère.

L'implant cochléaire est composé de deux parties: une partie interne placée chirurgicalement et une partie externe appelée processeur vocal, Ces deux parties sont indépendantes mais indissociables et le système ne sera fonctionnel qu'après les premiers réglages de la partie externe.

Le principe du fonctionnement de l'implant consiste à capter un signal sonore et à le transmettre sous forme d'impulsions au nerf auditif par l'intermédiaire des électrodes, suite à une stimulation électrique suivant la stratégie de codage.

Plusieurs stratégies de codage existent, qui sont différer par des cadences de stimulation, extraction de pics.... où les stratégies les plus utilisées dans les implants commercialisés utilisent un banc de filtres pour l'extraction des signaux de stimulation.

Ce travail, consiste à réaliser un banc de filtres dédié à un implant cochléaire, ensuite de l'implémenter sur un circuit numérique de type FPGA.

Cemémoire sera organisé en trois chapitres :

Dans le premier chapitre, nous présentons l'oreille et le problème de la surdité qui est résolus par l'implant cochléaire, Nous allons aussi parler de l'implant cochléaire et de ses composants ainsi que du principe de son fonctionnement, en se basant sur les stratégies de stimulation utilisées.

Dans le deuxième chapitre, nous parlons de la stratégie ACE et de son principe de fonctionnement, Ainsi qu'une description sur les filtres RIF et RII. Ensuite nous utilisons le logiciel Matlab pour simuler la stratégie ACE, en utilisant une banque de filtres composé de 22 filtres.

Dans le dernier chapitre nous commençons à générer le code VHDL et le« test-bench » pour la simulation du banc de filtres avec Modelsim. Après la récolte des résultats de simulation du programme VHDL nous passons à l'étape de l'implémentation sur un circuit FPGA en utilisant le logiciel ISE de Xilinx.

Enfin, nous terminerons par une conclusion générale.

CHAPITRE I

L'OREILLE ET LES IMPLANTS COCHLEAIRES

1. INTRODUCTION

Dans ce premier chapitre, nous présenterons l'oreille, puis nous découvrirons la surdité, qui a deux types : une perte de clarté de la voix et une déficience auditive.

Les scientifiques ont résolu le problème de la surdité grâce à un système d'implant cochléaire artificiel, qui amplifie les sons naturels, procurant un certain niveau d'audition à certaines personnes atteintes de surdité profonde. Cela se fait par des électrodes placées chirurgicalement pour stimuler les terminaisons nerveuses de l'audition dans la cochlée.

Enfin, nous discutons dans ce chapitre sur les différentes stratégies de stimulation cochléaire basées sur des bancs de filtres.

2. ANATOMIE DE L'OREILLE

L'oreille est un organe pair et symétrique. Elle peut assurer deux fonctions : l'audition et l'équilibration, est constituée de trois grandes régions :[1]

- L'oreille externe.
- L'oreille moyenne.
- L'oreille interne.

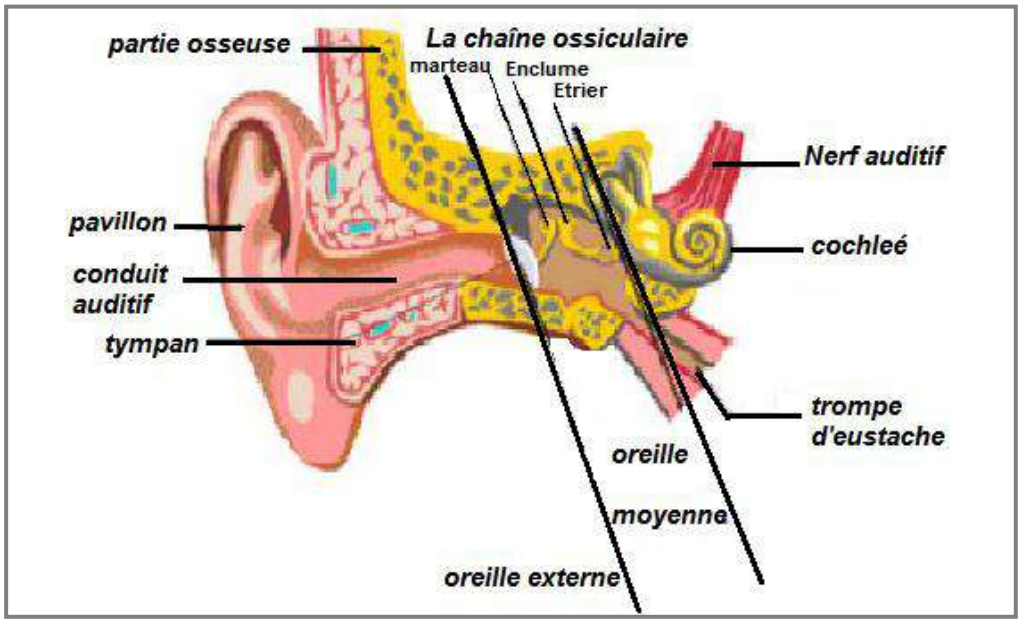


Figure I.1. Vue d'ensemble des 3 parties de l'oreille. [2]

2.1.L'oreilleexterne

Elle se compose du pavillon auriculaire, et du conduit auditif qui mène à la paroi extérieure du tympan. [2]

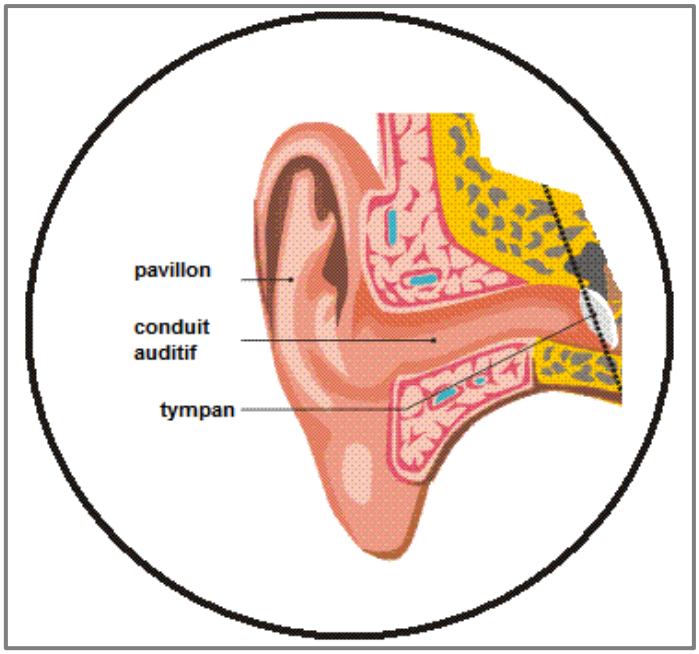


Figure I.2. Schéma de l'oreille externe. [3]

- L'oreille externe se comporte comme une antenne acoustique.
- Le pavillon dirige les ondes sonores vers le conduit auditif externe qui guide les sons vers le tympan.
- Le tympan est la terminaison acoustique de l'oreille externe.
- Elle assure la protection de l'oreille moyenne des agressions et chocs extérieurs. [4]

2.2.L'oreillemoyenne

L'oreille moyenne transmet l'énergie acoustique du tympan àl'oreille interne, en réalisant une adaptation entre le milieu aérien et le milieu liquidien. [4]

Elle comprend 3 parties communiquant les unes avec les autres :

- La caisse du tympan.
- Les cavitésmastoïdiennes.
- La trompe d'eustache. [1]

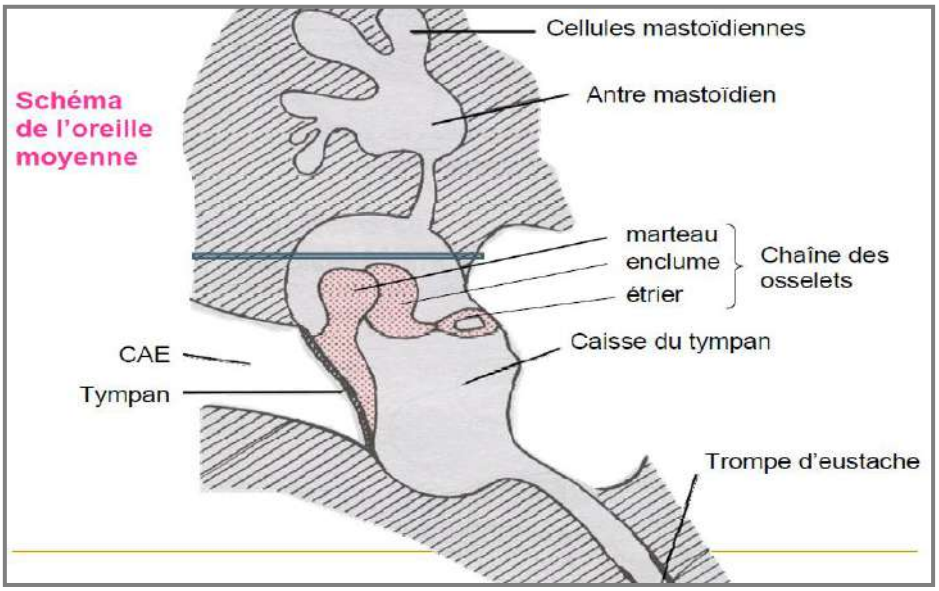


Figure I.3. Schéma de l'oreille moyenne. [1]

2.3.L'oreille interne

L'oreille interne est un labyrinthe rempli d'un liquide, lapérilymphe, et contient :

- La cochlée (avec la fenêtre ovale et la fenêtre ronde) organe de l'audition d'où part le nerf cochléaire.
- Le vestibule et les canaux semi circulaires, organes jouant un rôle dans l'équilibre, d'où par le nerf vestibulaire.

La réunion des deux nerfs constitue le nerf auditif. [2]

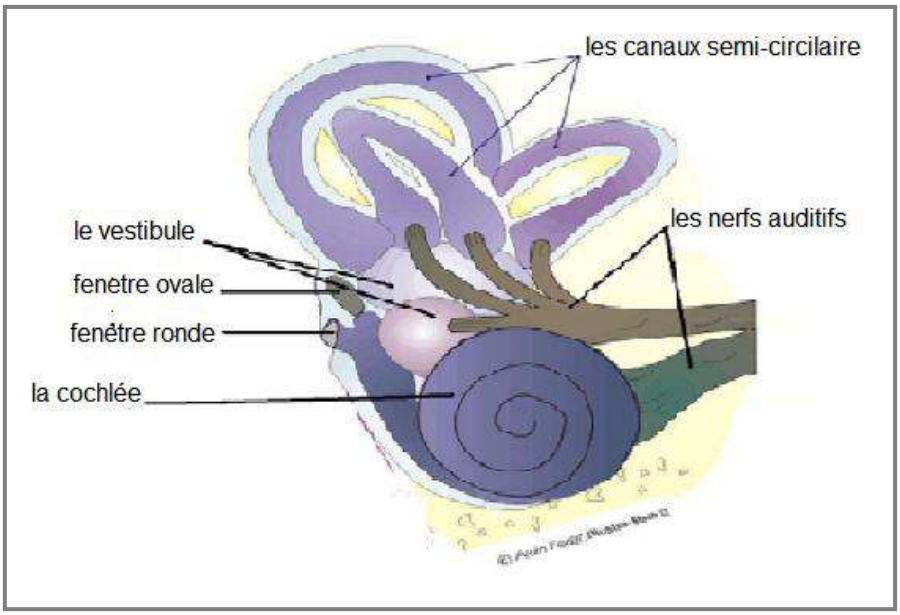


Figure I.4. Schéma de l'oreille interne. [4]

3. LA SURDITE

La surdité (ou hypoacousie) est un terme générique désignant la diminution de l'acuité auditive. Même s'il existe quelques rares surdités centrales (impliquant le cerveau), la grande majorité des surdités est liée à un problème au niveau de l'oreille. Les surdités de transmission ont pour origine l'oreille externe et moyenne. Les surdités neurosensorielles (encore appelées de perception) ont essentiellement pour origine l'oreille interne. Dans ces deux catégories, on distingue les surdités génétiques des surdités acquises. [5]

3.1.Types de surdité

3.1.1. La surdité de transmission

Est due à un défaut dans le système qui transmet le son, c'est-à-dire dans l'oreille externe et/ou moyenne. Elle se manifeste par une diminution de l'audition par la voie aérienne et par une audition conservée par la voie osseuse. L'otoscopie est souvent anormale. L'acoumétrie montre un test de Rinne négatif et un Weber latéralisé du côté de l'oreille malade. L'audiométrie tonale montre une diminution de la courbe aérienne. L'espace entre la courbe aérienne et la courbe osseuse s'appelle le gap de transmission. Le plus souvent, la surdité de transmission peut se traiter chirurgicalement, soit par reconstruction du système de transmission, soit par la pause d'un implant auditif à ancrage osseux. [6]

3.1.2. La surdité de perception

La **surdité de perception** concerne la partie qui réceptionne le son et l'envoie au cerveau, c'est-à-dire l'oreille interne et les voies nerveuses auditives. Elle se manifeste par une diminution de l'audition au niveau de la voie aérienne et de la voie osseuse. L'otoscopie est généralement normale. L'acoumétrie montre un Rinne positif et un Weber latéralisé du côté de l'oreille saine. L'audiométrie tonale montre une diminution conjointe des courbes aérienne et osseuse. La surdité de perception ne se traite pas chirurgicalement mais peut avoir recours à des implants auditifs, notamment l'implant cochléaire pour les surdités profondes. [6]

3.1.3. La surdité mixte

On parle finalement de **surdité mixte** lorsque les deux systèmes sont touchés. Les tests acoumétrique et audiométrique sont très variables, en fonction de la prépondérance de l'aspect de transmission ou de perception dans la surdité présentée.[6]

3.2.Degrés de surdité

Pour apprécier l'importance du handicap provoqué par une diminution de l'audition, les surdités sont classées, en fonction de la perte moyenne, en cinq degrés de surdité :

- surdité légère : la perte se situe entre 20 et 40 dB
- surdité modérée : la perte se situe entre 40 et 70 dB
- surdité sévère : la perte se situe entre 70 et 90 dB
- surdité profonde : la perte est supérieure à 90 dB.
- surdité totale : pas d'audition mesurable.[6]

4. L'IMPLANT COCHLEAIRE

L'implant cochléaire permet de réhabiliter comme l'aide auditive les patients atteints de surdité profonde a totale et qui ne tirent pas d'effet d'une amplification classique. Il permet de compenser les impulsions électriques délivrées par les cellules ciliées par l'intermédiaire d'électrodes placées dans la cochlée pour que les impulsions du nerf auditif soient interpréter par le cerveau.[7]

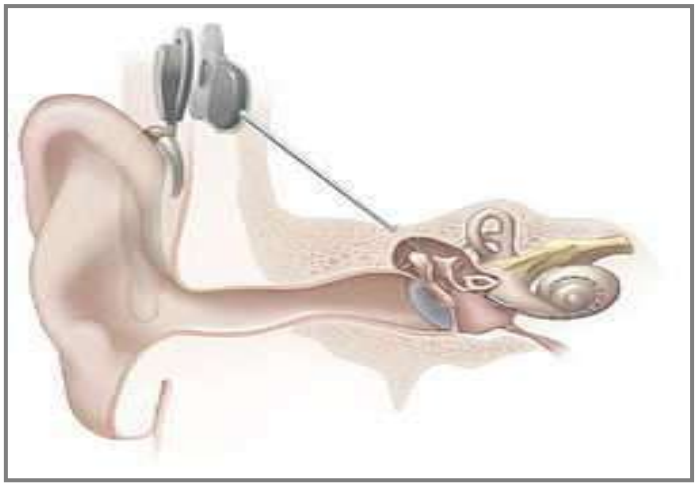


Figure I.5. Implant cochléaire.[7]

4.1.Historique

- Années 1790 A. Volta découvre que la stimulation électrique des oreilles crée une sensation auditive.
- 1868 R. Brenner utilise une stimulation alternative avec une vitesse, une polarité et une intensité variées.
- 1950 N. Lundberg stimule directement le nerf auditif avec un courant électrique.
- 1957 A. Djourno et C. Eyries réalisent une étude plus détaillée sur la stimulation des nerfs auditifs à l'aide de bobines inductives. Ils effectuent également une rééducation

postopératoire pour évaluer le développement de la parole chez une personne précédemment sourde.

- 1961 W. House, John Doyle et James Doyle insèrent des implants à une électrode dans la scala tympani de trois patients profondément sourds pendant une période d'essai d'environ trois semaines.
- 1966 F. Simmons implante un faisceau de six électrodes dans le modiolus (axe central) de la cochlée et tente pour la première fois de séparer les composants spectraux en bandes de fréquence.
- 1972 W. House construit le premier processeur de signal portable pour les implants à une électrode.
- 1978 G. Clark et B. Pyman réalisent la première opération d'implant cochléaire à électrodes multiples.
- 1984 La FDA4 approuve le premier CI (de House / 3M / Cochlear) pour implantation chez l'adulte.
- 1985 La FDA approuve l'utilisation de l'implant multicanal Nucleus de Cochlearchez l'adulte.
- 1990 La FDA approuve l'utilisation de l'implant Nucleus à partir de 2 ans.
- 1996 La FDA approuve l'implant multi canal Clarion de Advanced Bionics.
- 2001 La FDA approuve l'implant multi canal Combi40 + de MED-EL.
- 2004 Première combinaison réussie d'un appareil auditif et d'un implant cochléaire dans la même oreille. [8]

4.2.Composition de l'implantcochléaire

Tous les types d'implant cochléaire possèdent deux parties : une partie externe qui permet de capter le son à l'aide d'un microphone, puis de traiter via le microprocesseur les signaux. Ensuite une partie interne va simuler le nerf auditif grâce aux électrodes. Figure I.6 résume les différentes parties.[7]

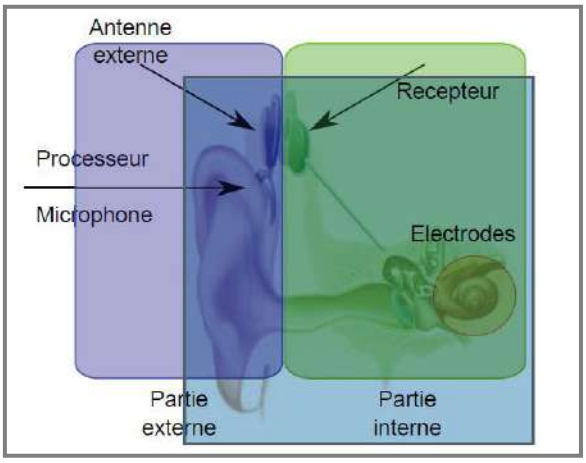


Figure I.6. Représentation des parties externe et interne de l'implant cochléaire.[7]

4.2.1. La partieexterne

Les accessoires principaux de la parie externe sont le microphone, le processeur et l'antenne. Le son est capte par le microphone, puis par une FFT, traite au niveau fréquentiel pour sélectionner les électrodes à simuler, la compression et le codage sont ensuite appliques. [7]

4.2.2. La partie interne

La partie interne contient une bobine qui capte le signal émis par l'antenne externe. Puis elle va stimuler les différentes électrodes situées dans la cochlée.[7]

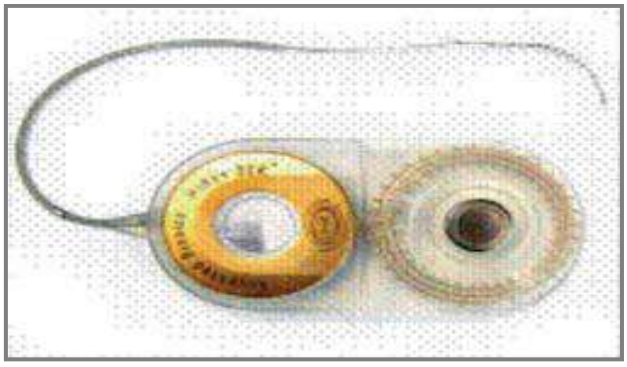


Figure I.7. Partie interne d'un implant cochléaire.

4.3.Fonctionnement de l'implantcochléaire

Dans l'audition normale, le rôle de la cochlée est de convertir les sons transmis par le tympan et la chaîne des osselets, en signaux électriques acheminés au cerveau par le nerf auditif. L'oreille interne a pour propriété d'analyser l'onde acoustique et de la distribuer le long de la cochlée, en fonction de la fréquence qui est détectée par la membrane basilaire.[9]

L'implant cochléaire fonctionne selon le principe suivant :

4.3.1. Fonctionnement de la partie externe

L'explant ou encore le processeur est un contour d'oreille ou un petit boîtier déporté qui traite le son ambiant capté par un microphone placé sur le pavillon de l'oreille. Le traitement du signal sonore réalisé par le microprocesseur est entièrement numérique. La stimulation est alors transmise à l'implant par radiofréquence qui franchit la barrière cutanée par induction électromagnétique via une antenne dissimulée dans les cheveux. Le processeur externe est alimenté par un bloc accumulateur rechargeable ou des piles zinc-air suivant le modèle. L'antenne externe émettrice est maintenue en regard du récepteur implanté grâce à un aimant à force réglable. [9]

4.3.2. Fonctionnement de la partie interne

L'implant à proprement parler est un petit boîtier mis en place chirurgicalement derrière l'oreille et sous la peau. Il reçoit les informations du processeur externe et les distribue sur un ensemble d'électrodes insérées dans la cochlée. Comme sur le clavier d'un piano, chaque électrode correspond à une bande de fréquences du signal sonore (des fréquences graves aux fréquences aiguës).

L'implant est constitué d'un boîtier étanche et biocompatible en céramique et/ou en titane qui contient l'électronique de stimulation, une antenne de réception des informations externes ainsi qu'un aimant de maintien de l'antenne externe.[9]

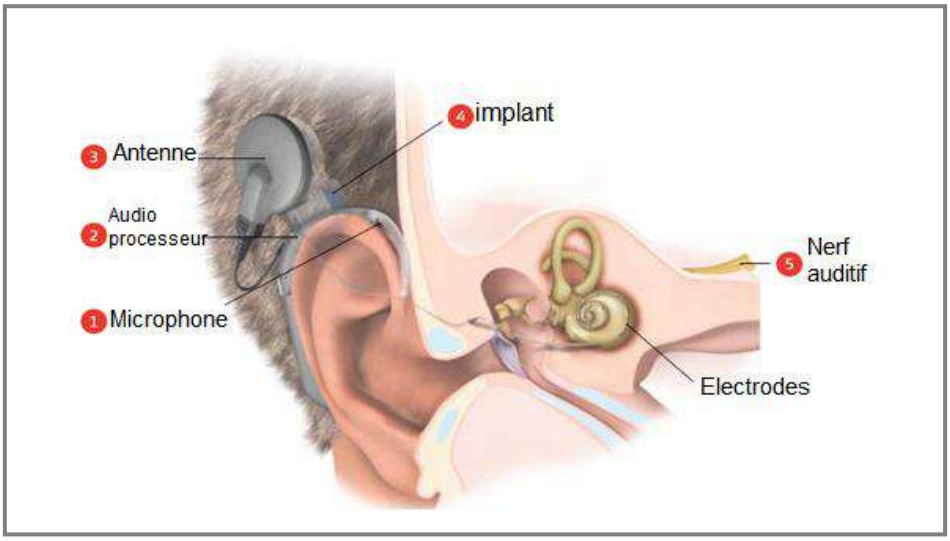


Figure I.8. Principe de fonctionnement d'un système d'implant cochléaire. [10]

1. Les sons sont captés par le microphone de l'audio processeur.
2. L'audio processeur analyse et code les sons en séquences spécifiques d'impulsions électriques.
3. Ces impulsions sont envoyées vers l'antenne et transmises à travers la peau vers l'implant.
4. L'implant envoie les impulsions aux électrodes dans la cochlée.
5. Le nerf auditif achemine les signaux électriques vers les centres auditifs du cerveau qui les interprète comme des sons.[10]

4.4.Avantages et inconvénients des appareils auditifs

Les appareils auditifs présentent de nombreux avantages et inconvénients Nous mentionnons ce qui suit :

Les avantages

- Permettre aux personnes sourdes profondes et sévères d'entendre, on peut considérer cet appareil comme un miracle pour certaines personnes.
- Associer des accessoires en même temps sur un implant (mini-micro, émetteur audio Tv,...).
- Disposer d'un système Bluetooth integer.
- Permettre d'aller sous l'eau avec l'implant grâce à un accessoire spécifique.

Les inconvénients

- La personne implantée doit revenir faire des réglages de ses implants fréquemment au début puis moins souvent par la suite ce qui reste une contrainte.
- Audition différente.
- Coûtélevé.
- Eviter les chocs pour que la partie interne ne bouge pas.[11]

5. LES STRATEGIES DE STIMULATION COCHLEAIRE

Dans le système d'implant cochléaire, la stratégie de stimulation joue un rôle extrêmement importantpour générer les sons entendus par les utilisateurs.

De nombreuses stratégies de stimulation ont été développées au cours des deux dernières décennies. La stratégie de stimulation reproduit fidèlement le spectre sonore original et permet à l'utilisateur de l'implant cochléaire d'entendre des sons clairs. [12]

Dans ce qui suit, nous décrivons brièvement les stratégies :spectral peak (SPEAK), ContinuousInterleavedSampling (CIS), Advanced Combinational Encoder (ACE), et HiResWithFidelity 120 option (HiRes 120), et la stratégie n-de-m.

5.1.La stratégie SPEAK

Le schéma fonctionnel de la stratégie SPEAK est décrit à laFigure I.9. Le signal d'entrée est de la parole segmentée (longueur de 128 échantillons et chevauchement de 75%) et le spectre est calculé. En outre, une banque de filtres avec 20 filtres passe-bande avec une largeur de bande croissante est utilisée. Chaque filtre correspond à une électrode intracochléaire. Les électrodes 1 et 2 ne sont pas utilisées (le Nucleus 24 comprend 22 électrodes intracochléaires, la stratégie SPEAK ne

fonctionne qu'avec 20 bandes). L'électrode apicale correspond à la bande 1 (la bande passante la plus étroite). Le bloc de calcul d'énergie calcule l'énergie dans chaque bande traitée. Le bloc de sélection de bande sélectionne M bandes avec une énergie maximale. Le M peut varier entre 6 et 10. L'énergie de toutes les bandes non sélectionnées n'est pas utilisée pour la stimulation. Le bloc LGF (fonction de gain logarithmique) implémente la relation logarithmique entre l'amplitude actuelle et la sonie de la perception auditive. Le bloc de mappage de canal convertit les informations du bloc LGF en impulsions de courant et définit le taux de stimulation. Le bloc NVMem (mémoire vive avec batterie de secours) stocke les paramètres psychophysiques de base tels que niveau seuil, niveau de confort, taux de stimulation, etc. La stratégie SPEAK peut être caractérisée comme une stratégie avec un nombre élevé de bandes de traitement mais avec un taux de stimulation faible 250 Hz. [13]

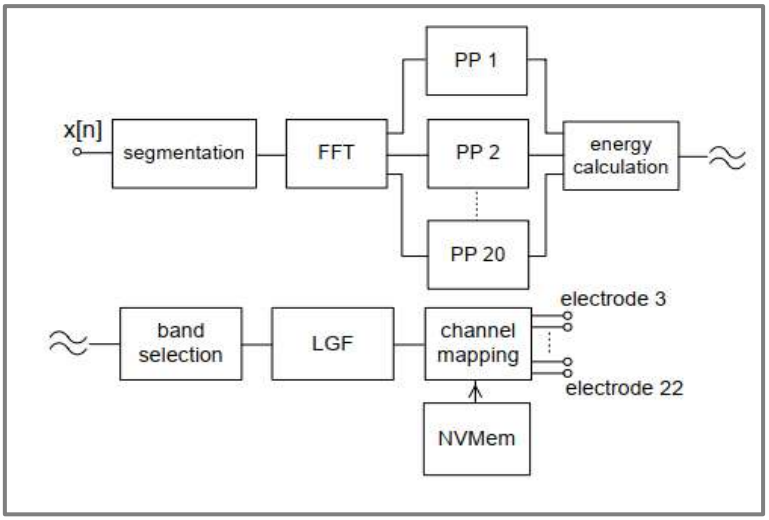


Figure I.9. Le diagramme de stratégie SPEAK.[13]

5.2.La stratégie ACE

La stratégie ACE, utilisée dans l'implant Nucleus, repose sur le principe dit de N sur M. Ce système utilise 22 électrodes implantées qui peuvent être activées pour générer 22 canaux fixes. Le signal est traité dans 22 bandes de fréquences pour chaque image de son enregistré.

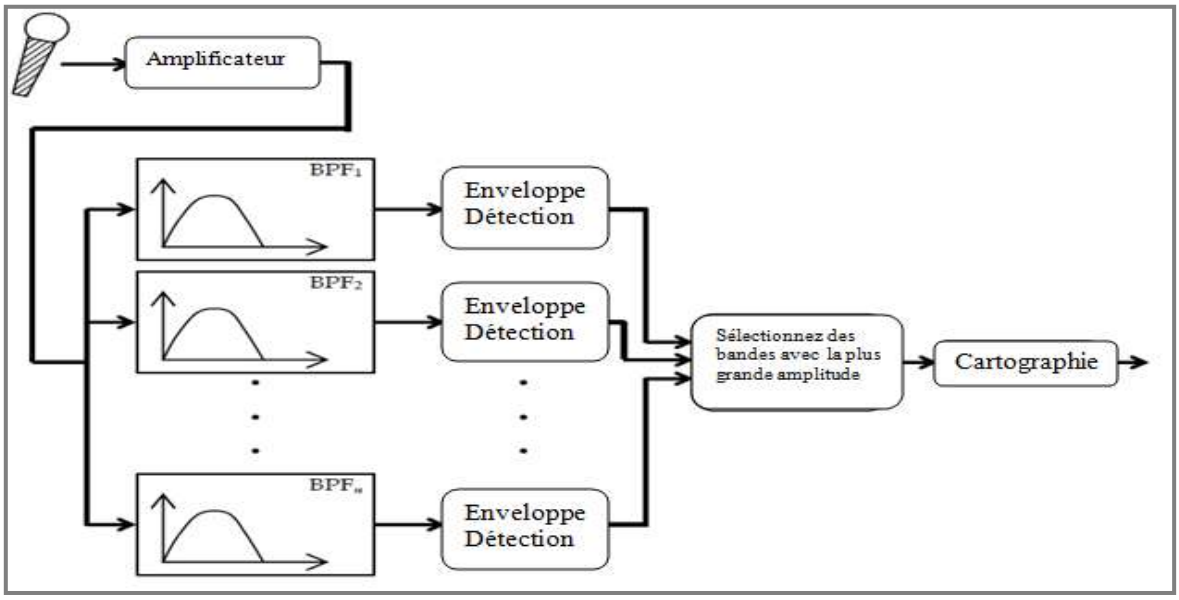


Figure I.10. Schéma de la stratégie ACE. [12]

Une fois les informations d'enveloppe extraites pour chaque bande de fréquences, 8 à 10 bandes (définies par l'audiologiste) ayant les plus grandes amplitudes seront stimulées. Les électrodes correspondant aux canaux sélectionnés sont ensuite activées. Ainsi, dans la stratégie ACE, un canal est généré par une électrode implantée et le spectre initial est reproduit par 8 à 10 canaux fixes.[12]

5.3.La stratégie CIS (Continuons InterleavedSampling)

La stratégie CIS de la Figure I.11 a été développée pour l'implant « Ineraid ». Dans cette stratégie, le banc de filtres comprend six bandes de fréquences. L'enveloppe de la forme d'onde est estimée à la sortie de chaque filtre. Ces signaux d'enveloppe sont échantillonnés à un taux fixe. Les sorties de l'enveloppe sont finalement compressées pour correspondre à la plage dynamique de l'audition électrique et ensuite utilisés pour moduler les impulsions biphasées. La sortie du filtre en fonction d'impulsions électriques est délivrée à six électrodes intracochléaires fixes. La principale caractéristique de la stratégie CIS est l'utilisation des taux de stimulation plus élevés pour assurer une meilleure représentation de l'information temporelle. Les variations de l'amplitude d'impulsion peuvent suivre les changements rapides dans le signal de la parole. Ceci est possible en raison des impulsions plus courtes avec des retards minimes par rapport à l'intervalle entre les impulsions. Le taux de stimulation utilisé dans la stratégie CIS de l'implant Ineraid est généralement 800 pps/ch ou plus. [14]

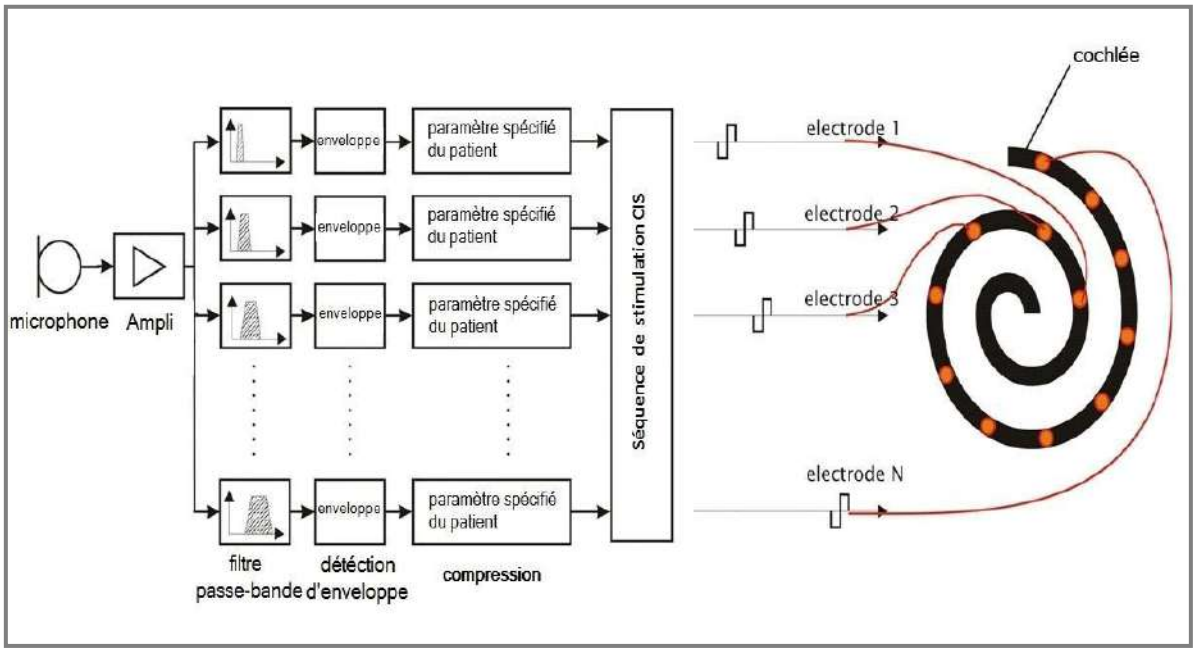


Figure I.11. Diagramme de la stratégie CIS. [14]

5.4. La stratégie HiRes120

La stratégie HiRes120, utilisée dans l'implant Bionique avancée, est la première stratégie de stimulation utilisant la technique du canal virtuel. Des canaux virtuels sont créés en ajustant le niveau énergétique de deux électrodes voisines. Depuis l'implant bionique avancée comporte 16 électrodes implantées, il existe 15 paires d'électrodes pouvant être utilisé pour diriger le centre de la stimulation électrique. La Figure I.12 représenté le schéma bloc de la stratégie HiRes120. Pour chaque signal sonore, le signal est divisé par 15filtres passe-bande et l'enveloppe est extraite pour chaque bande de fréquence. En outre, les 15 pics spectraux, qui indiquent la fréquence la plus importante dans chaque bande fréquencielle, sont également dérivés en utilisant la transformée de Fourier rapide (FFT). Ces pics spectraux sont ensuite dirigés par paires d'électrodes correspondantes basées sur la technique du canal virtuel. La stratégie HiRes120 fournit également des canaux en séquence avec un taux de stimulation élevé similaire à celui de la CIS. Ainsi, dans la stratégie HiRes120, un canal est généré par deux électrodes voisines. Le spectre original est reproduit par 15 canaux virtuels et toutes les paires d'électrodes sont activé dans une séquence prédéfinie dans les 15 cycles d'horloge consécutifs. [12]

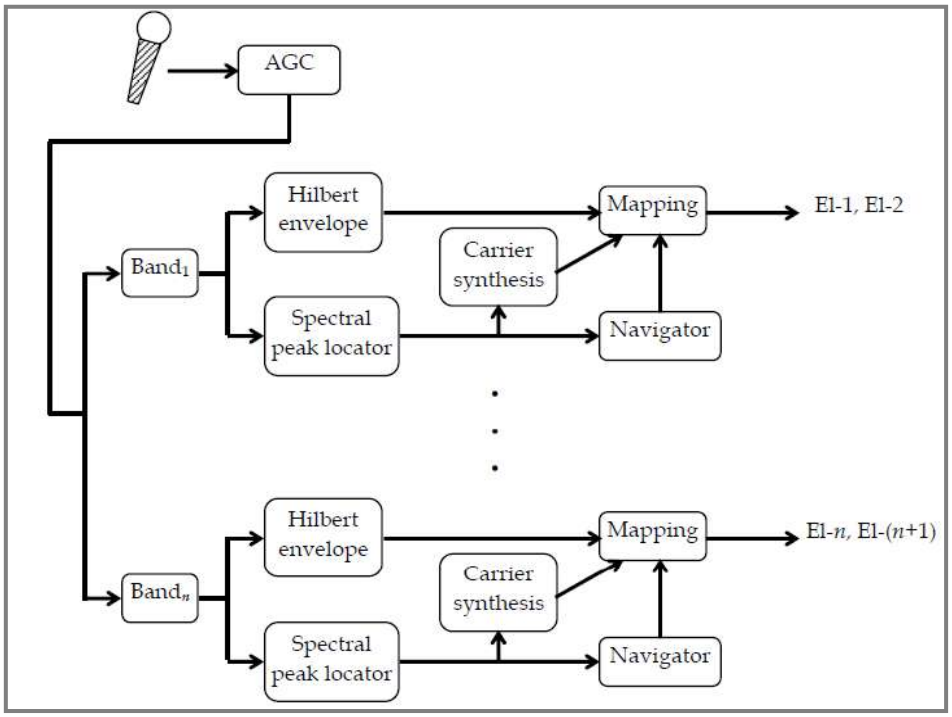


Figure I.12. Schéma fonctionnel de la stratégie de stimulation HiRes120. [12]

5.5. Les stratégies n-de-m

Le paramètre n-de-m permet de sélectionner des stratégies CIS (lorsque n est égal à m, c'est-à-dire, au nombre de canaux insérés) ou des stratégies n-de-m (lorsque n est inférieur à m). Les stratégies n-de-m fonctionnent en activant, dans chaque cycle de traitement, uniquement les n canaux qui présentent le plus d'énergie parmi les m disponibles. Le but des stratégies n-de-m est de permettre un accroissement de la fréquence de stimulation grâce à la réduction du nombre de canaux activés dans chaque cycle, ce qui réduit la durée totale du cycle de traitement. [15]

6. BANC DE FILTRES

Les bancs de filtres sont des outils du traitement de signal qui permet d'obtenir une représentation particulière de l'information contenue dans un signal ; la représentation donnée par les bancs de filtres reste du domaine temporel. L'idée de base est d'obtenir une série de signaux représentatifs d'une bande de fréquences du signal original. Le signal est filtré par un ensemble de M filtres passe-bandes à supports disjoints Figure I.13, ce qui permet d'obtenir M signaux correspondant chacun à une partie du spectre du signal original. [16]

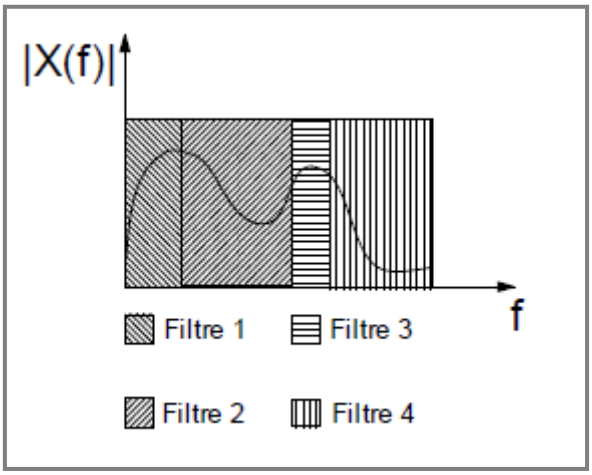


Figure I.13. Découpage en 4 bandes.

Dans une telle représentation, à un échantillon de l'entrée, correspondent 4 Échantillons en sortie. Il y a redondance de l'information car, chaque X_n^kn'occupe qu'une partie du spectre. Les X_n^ksont des signaux à bande Étroite ; on doit, donc, pouvoir les échantillonner à une fréquence plus basse que la fréquence d'échantillonnage initiale sans perte d'information.

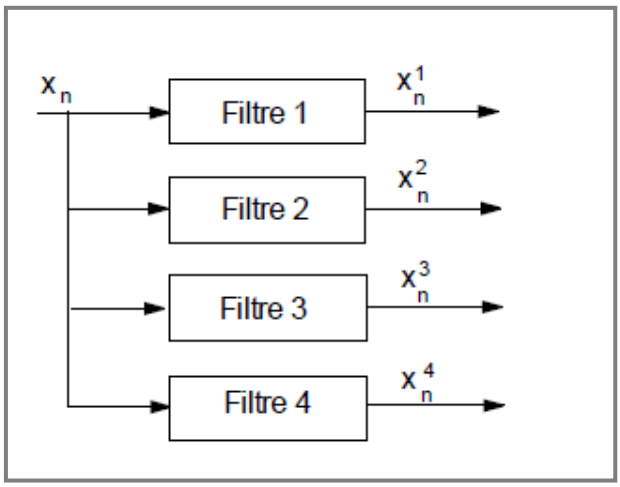


Figure I.14. Banc de filtre à 4 voies.

$\boldsymbol{X_n^k}$ Est le signal FBMC(Filtre Bank BasedMulti Carrier) avec **k** : le nombre de sous porteuses et **n** :est le flux en temps.

Un banc de filtres est, par définition, une série de filtres ayant, soit une entrée, soit une sortie, en commun.

Le système de la Figure I.15 (a) est appelé banc de filtres d'analyse, et les filtres H_k(z) sont les filtres d'analyse. Ce banc de filtres à reconstruction parfaite décompose le signal x(n) en M signaux$v_i(n)$ appelés signaux de sous-bandes.

Le système de la Figure I.15 (b) est appelé banc de filtres de synthèse et les filtres F_k(z) sont les filtres de synthèse. Il combine les M signaux w_k(n) en un seul signal y(n).Les bancs de filtres d'analyse et de synthèse sont, généralement, associés; le premier décompose un signal pour appliquer un traitement à chaque signal de sous-bande,et le second recombine les signaux de sous-bandes traités pour construire le signal modifié.

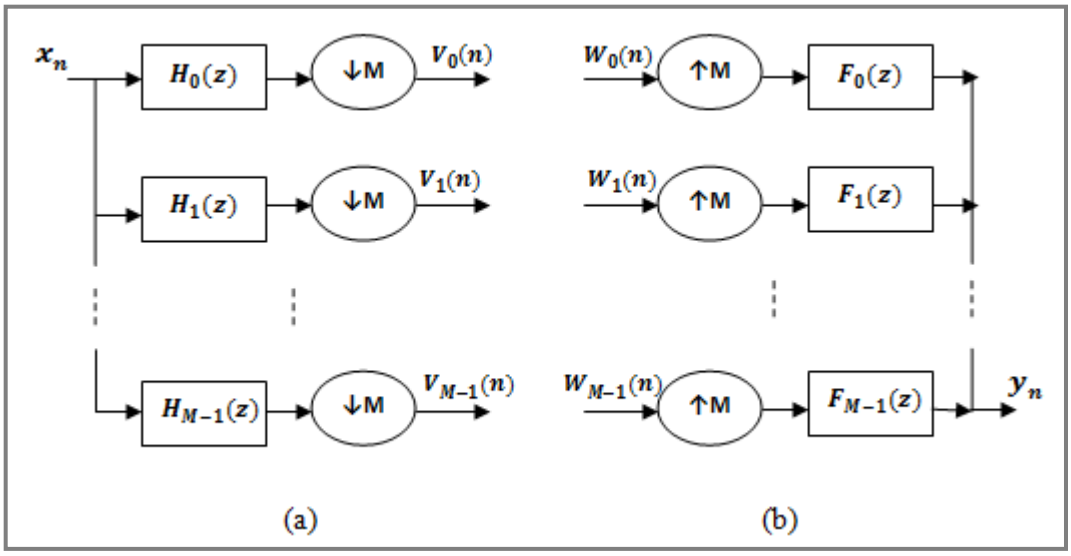


Figure I.15. Banc de filtres à M canaux (a) banc d'analyse (b) banc de synthèse. [16]

7. LES CIRCUITS LOGIQUES PROGRAMMABLES

Il y a quelques années la réalisation d'un montage en électronique numérique impliquait l'utilisation d'un nombre important de circuits intégrés logiques. Ceci avait pour conséquences un prix de revient élevé, une mise en œuvre complexe et un circuit imprimé de taille importante.

Le développement des mémoires utilisées en informatique fut à l'origine des premiers circuits logiques programmables (PLD : Programmable LogicDevice).

Les circuits logiques programmables sont utilisés pour remplacer l'association de plusieurs boîtiers logiques. Le câblage est simplifié, l'encombrement et le risque de pannes est réduit. Certains PLD ne permettent pas la relecture de la fonction logique programmée, c'est pratique lorsque le programme doit rester confidentiel.

Ces circuits disposent d'un certain nombre de broches d'entrées et de sorties. L'utilisateur associe ces broches aux équations logiques (plus ou moins complexes) qu'il programme dans le circuit.Sa mise en œuvre se fait très facilement à l'aide d'un programmateur, d'un micro-ordinateur et d'un logiciel adapté.[25]

7.1.Historique des circuits logiques programmables

- Le premier circuit logique programmable a été mis au point par Harris Semiconductors et consistait en une matrice de diodes très simple. Ceci ne représentequ'un17rimiti17rimitive des premiers PLD.
- Les PLD actuels sont plutôt basés sur des réseaux logiques programmables (PLA). Ce dernier type de PLD a été créé en 1969 par IBM et amélioré par Texas Instruments (TI), Collins Radio et National Semiconductor.

Ensuite,

- Intersil, Signetics et Data I/O ont été parmi les premiers à lancer la vague des PLD modernes.
- Le premier PLD en technologie des transistors MOS a été complété en 1971 par collaboration avec GE ont commencé une nouvelle série de PLD qu'on appelait réseau General Electric (GE). Ce circuit effaçable à l'ultraviolet représentait le premier EPLD (Electrically PLD) disponible sur le marché. [26]

7.2.Ressources matérielles de circuits logiques programmables

Un circuit logique programmable contient généralement trois types de ressources ou de fonctions de base:

1) Fonction combinatoire ou séquentielle: la reconfiguration de cette fonction peut être basée sur une table de vérité (LUT)(Look Up Table), un jeu de multiplexeurs, un plan ET-OU, différentes portes logiques, ou un ensemble de transistors.
2) Bloc d'entrée/sortie: ce type de fonction réalise un circuit logique d'entrée/sortie bidirectionnel. Il est basé sur des bascules, des bistables, contrôle du « slew-rate », «pullup» et « pulldown ».
3) Interconnexions : Circuiterie servant à accommoder les deux types d'interconnexions: locale et globale. L'importance de ce type de fonction tient au fait qu'il faut assurer une flexibilité maximale. [26]

7.3.La technologie FPGA

Les FPGAs(Field Programmable GateArray), ou bien les réseaux logiques programmables, sont des composants électroniques programmables de la famille des PLDs. Un FPGA est un ensemble de blocs logiques élémentaires que l'utilisateur peut interconnecter pour réaliser les fonctions logiques de son choix. La densité des portes est importante et sans cesse en évolution. L'avantage d'un

FPGA est leur grande souplesse dans leur technologie permettant une réutilisation à volonté et en un temps très court (quelques millisecondes) dans des algorithmes différents. Le progrès technologique permet de faire des composants toujours plus rapides et à plus haute intégration, autorisant la programmation d'applications importantes.

Grâce à l'évolution des procédés de fabrication, ces composants peuvent actuellement supporter des applications complexes. Les FPGAs sont constitués d'un réseau de blocs logiques, de blocs mémoires, de blocs dédiés et d'entrées/sorties (l'exemple des Virtex5 de Xilinx et Cyclone-II de Altera). L'ensemble est relié par un réseau d'interconnexions programmable. Les blocs logiques permettent de réaliser des opérations avec quelques variables à travers une LUT (Look Up Table). Le résultat peut être éventuellement stocké dans un registre, les blocs RAM permettent d'implanter des mémoires adressables et des FIFO, les blocs dédiés permettent de réaliser facilement de nombreuses opérations de traitement (blocs DSP), de gérer l'horloge, ou des interfaces de communication. Les nombreux ports permettent également de connecter des périphériques de la plateforme matérielle à base de FPGA.

La programmation des FPGA se fait avec un langage de programmation de hardware tel que le VHDL et le Verilog. L'outil de développement transforme cette description en un fichier de configuration du FPGA en plusieurs étapes : le HDL doit d'abord être synthétisé (transformé en éléments logiques de base), ensuite les éléments doivent être placés sur le composant (placement) et enfin interconnectés (routage). [28]

Tous les FPGA sont constitués de trois composants principaux, à savoir blocs logiques (LB), Les blocs d'entrée sortie I/O, et le routage programmable ou interconnexions comme le montre la figure III.26.

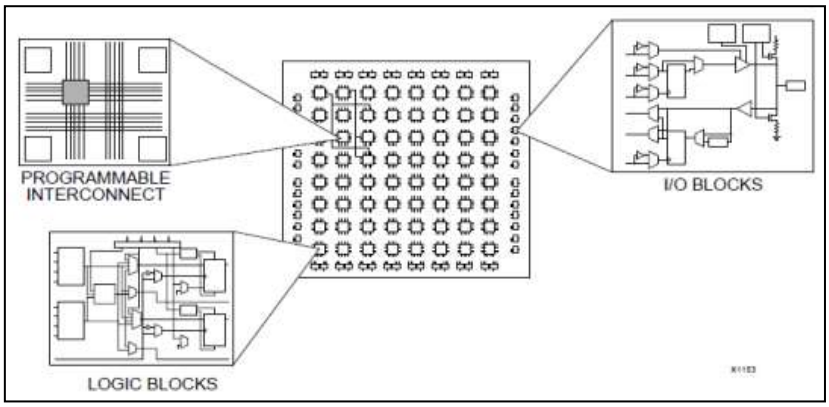


Figure I.16. Architecture générale d'un FPGA. [28]

7.3.1. Les étapes du flot de conception

Afin de mettre en œuvre un circuit sur un FPGA, plusieurs étapes sont parcourues (Figure III.27) qui consistent à concevoir et affiner la description du circuit du haut niveau (algorithme) jusqu'au bas niveau (dessin géométrique ou layout).

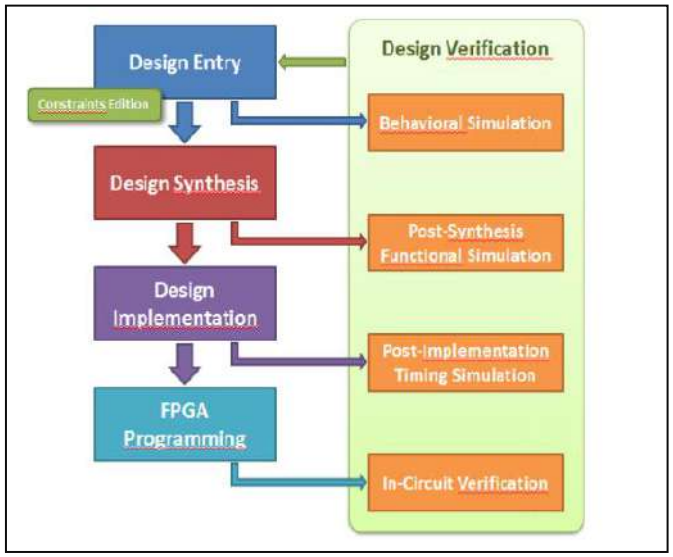


Figure I.17. Les étapes du flot de la conception.

Passer d'un niveau au suivant et vérifier le résultat obtenu se fait de façon semi-automatique en utilisant des outils de conception tels queISE de Xilinx. Ainsi, les étapes de flot de conception se résument comme suit :

- Entrée de conception (Design entry).
- Synthèse (Synthesis).
- Implémentation (Implementation).
- Vérification (Verification).
- La configuration du circuit (Device configuration). [28]

7.3.2. Le langage VHDL

C'est un langage de description de matériel qui est utilisé pour la spécification, la simulation et la preuve formelle d'équivalence de circuits. Ensuite il a été utilisé pour la synthèse automatique. C'est un langage de description de matériel pour circuit à très haute vitesse d'intégration.

Aussi le langage VHDL permet la description des aspects les plus importants d'un système matériel (hardware system), à savoir son comportement, sa structure et ses caractéristiques temporelles. Par système matériel, on entend un système électronique arbitrairement complexe réalisé sous la forme d'un circuit intégré ou d'un ensemble de cartes.

- **Quelquecaractéristique du langage VHDL**

- **La portabilité :** La portabilité constituait un des objectifs principaux pendant la phase de définition du langage VHDL. L'utilisation largement répandue de ce langage est essentiellement due au fait qu'il soit un standard.
- **La lisibilité :** La représentation schématique n'est pas toujours d'une grande lisibilité. Il est difficile de saisir rapidement le fonctionnement d'une machine d'état parmi un enchevêtrement de registres et de portes logiques.
- **La modularité :** La modularité est une des caractéristiques essentielles de VHDL. Une description partitionnée en plusieurs sous-ensembles est dite modulaire.
- **Le couple entité architecture :** L'élément essentiel de toute description en VHDL est formé par le couple entité/architecture qui décrit l'apparence externe d'un module et son fonctionnement interne.

➢ L'entité décrit la vue externe du modèle : elle permet de définir les ports par où sont véhiculés les informations (signaux) et les paramètres génériques.

➢ L'architecture définit la vue interne du modèle. Cette description peut être de type structurel, flot de données, comportemental ou une combinaison des trois.

On distingue 3 grandes classes de descriptions dans ce langage :

✓ **La description comportementale**: Elles spécifient le comportement au moyen de formules ou de programmes d'actions. Cela signifie que la sortie est en permanence donnée en fonction des entrées. Une telle description n'utilise aucun composantinterne : on considère que le composant est atomique, entièrement défini par sa spécification logique.

✓ **La description structurelle**: Elles énoncent une interconnexion de composants. Une description structurelle est la transcription directe d'un schéma. Dans cette description le comportement résulte du comportement des composants internes et du câblage réalisé, selon les règles usuelles de causalité physique.

✓ **La description flot de données:** Une description flot de données spécifie comment la donnée est transférée de signal à signal sans utiliser d'affectations séquentielles (comme dans la description comportementale). Cette description ne nécessite pas de déclaration de processus, et toutes les déclarations s'exécutent en même temps. [28]

8. CONCLUSION

La stratégie de stimulation est importante dans l'implantation cochléaire,Dans ce chapitre,Certaines stratégies de stimulation de base utilisées dans les systèmes commerciaux ont été passées en revue. Parmi eux,La stratégie de stimulation cochléaire ACE la plus couramment utilisée dans les implants cochléaires industriels qui est basée sur le banc de filtres passe bandes, ce que nous expliquerons en détails dans le deuxième chapitre.

CHAPITRE II

ETUDE ET SIMULATION DE LA STRATEGIE ACE

1. INTRODUCTION

Diverses stratégies de traitement de la parole ont été développées au cours des 30 dernières années pour simuler le plus naturellement possible les profils de résonance au sein de la cochlée, car ils jouent un rôle très important dans la maximisation du potentiel de communication global de l'utilisateur.

Dans ce chapitre, nous parlerons de la stratégie de stimulation cochléaire ACE. Nous entamons ensuite la simulation en Matlab de cette stratégie en se basant sur le banc de filtres, qui nous intéresse le plus dans le cadre de notre travail.

2. LA STRATÉGIE ACE (ADVANCED COMBINATION ENCODER)

Dans le système d'implant cochléaire, la stratégie de stimulation joue un rôle extrêmement important dans la génération des sons entendus par les utilisateurs.

2.1. Principe de fonctionnement de la stratégie ACE

Le mécanisme principal de l'ACE est la mise en correspondance du spectre de puissance du signal avec des électrodes, où seules N des M (N <M) électrodes d'amplitude la plus élevée sont activées.

Le signal du microphone est d'abord mis en évidence par un filtre qui amplifie notamment les composants hauts fréquence. Le contrôle de gain adaptatif (AGC) est ensuite utilisé pour limiter la distorsion des sons forts en réduisant l'amplification au bon moment.

Ensuite, le signal est numérisé et envoyé par un banc de filtres. ACE ne définit pas explicitement une certaine approche de banc de filtres. Les limites de fréquence du groupe de filtres sont linéairement espacées au-dessous de 1000 Hz et logarithmiquement au-dessus de 1000 Hz.

Une estimation de l'enveloppe est calculée pour chaque bande spectrale du signal audio. Les enveloppes sont obtenues en calculant la magnitude de la sortie complexe. Chaque filtre passe-bande est attribué à une électrode et représente un canal. Pour chaque trame du signal audio, N électrodes sont stimulées séquentiellement et un cycle de stimulation est terminé. Le nombre de cycles / seconde détermine donc le taux de stimulation sur un seul canal, également appelé taux de stimulation du canal.

La largeur de bande d'un implant cochléaire est limitée par le nombre de canaux (électrodes) et le taux de stimulation global. Le taux de stimulation du canal représente la résolution temporelle de l'implant, tandis que le nombre total d'électrodes M représente la résolution en fréquence.

Cependant, seules N électrodes sur M (N <M) sont stimulées à chaque cycle. Par conséquent, un sous-ensemble d'échantillons de sortie de la banque de filtres ayant la plus grande amplitude est sélectionné. Si N est diminué, la représentation spectrale du signal audio devient plus mauvaise, mais le taux de stimulation du canal peut être augmenté, donnant une meilleure représentation temporelle du signal audio. Inversement, si le taux de stimulation du canal est diminué, N peut être augmenté, ce qui donne une meilleure représentation spectrale du signal audio.

Enfin, la dernière étape du processus mappe les amplitudes sur les électrodes correspondantes, en comprimant les amplitudes acoustiques dans la plage dynamique du sujet entre le seuil mesuré et le niveau de volume maximal confortable pour la stimulation électrique. [17]

2.1.Le rôle de la stratégie ACE

Le rôle de la stratégie de codage est donc fondamental dans le succès de l'implant cochléaire. Elle consiste à gérer la stimulation électrique des électrodes en fonction du signal capté par le microphone tout en tenant compte de la sensibilité auditive du patient. Ensuite, il a été montré que la parole restait intelligible, même en étant très appauvrie fréquentiellement. En effet, la parole vocodée par très peu (huit) bandes de fréquences modulées à l'enveloppe du signal reste intelligible. Les électrodes réparties le long de la cochlée sont affectées d'une bande de fréquence allant de la plus élevée à la plus basse de la base vers l'apex. L'activité des électrodes est par conséquent modulée par les caractéristiques spectrales du son capté par le microphone La stratégie de codage la plus répandue ACE fonctionne de la façon suivante : le signal sonore est décomposé par transformée de Fourier, et le contenu spectral est réparti dans les électrodes en fonction de leur bande de fréquence. Les N bandes les plus énergétiques sont retenus. Enfin, la stimulation est délivrée séquentiellement, une électrode après l'autre avec une cadence de R impulsions par seconde. [18]

3. DIFFERENTS TYPES DES FILTRES

On classe les filtres en deux grandes familles : analogique et numérique.

3.1.Les filtres numériques

Un filtre numérique est une combinaison linéaire d'échantillons. Le filtrage et l'analyse spectrale sont des techniques de base dans le traitement numérique du signal.

Quelques domaines d'application du filtrage (liste non-exhaustive):

- Communications : téléphone, radio, télévision, etc.
- Musique.

- Radar.
- Reconnaissance de Parole.
- Traitement d'image (ex : satellite, médicale, inspection industrielle).
- Vision par ordinateur.

Il existe deux sortes de filtres numériques : RIF et RII. [19]

3.1.1. Filtres à réponse impulsionnelle finie (filtres RIF)

Un filtre numérique à réponse impulsionnelle finie réalise le calcul d'un signal de sortie y_n en fonction du signal d'entrée x_n de la manière suivante :

$$y_n = \sum_{k=0}^{N-1} b_k \, x_{n-k} \tag{II.1}$$

La valeur de la sortie à l'instant n'est donc une combinaison linéaire des valeurs de l'entrée à l'instant n et aux N − 1 instants antérieurs. Si les valeurs de l'entrée ne sont disponibles qu'à partir de l'indice 0, la première valeur de la sortie est y_{N-1}, puisqu'il faut au moins N éléments en entrée pour faire le calcul. Pour voir la signification des coefficients h_k, considérons en entrée une impulsion. [20]

3.1.2. Filtres à réponse impulsionnelle infinie (filtres RII)

Filtre pour lequel la réponse impulsionnelle n'est pas finit. Typiquement réalisé par une opération linéaire "récursive". Ces filtres sont aussi connu par nom filtres récursif. Pour un filtre RII, la réponse impulsionnelle ne peut pas être calculée (par définition). Le filtre est spécifié par deux jeux de coefficients a(n), $1 \leq n < N$ et b(n), $0 \leq n < M$:

$$y_n = \sum_{k=0}^{N-1} b_k x_{n-k} - \sum_{k=1}^{M-1} a_k \, y_{n-k-1} \tag{II.2}$$

Le filtre RII peut être conçu par des méthodes semblables à ceux utilisé pour les filtres analogiques. Ceci n'est pas vraie pour les filtres RIF. [19]

3.2.Les filters analogiques

Les filtres analogiques se divisent eux-mêmes en plusieurs catégories :

Les filtres passifs sont presque toujours constitués d'inductances de haute qualité et de condensateurs.

Les filtres actifs sont constitués de condensateurs et de résistances, et d'éléments actifs qui sont presque toujours des amplificateurs opérationnels. Ils sont moins encombrants, plus faciles à

réaliser et donc moins coûteux. Par contre ils ne sont pas utilisables avec des signaux de fréquences trop élevées le maximum pratique étant de quelques mégahertz. [23]

4. FILTRE PASSE-BANDE

Un filtre passe-bande est un filtre ne laissant passer qu'une bande ou intervalle de fréquences compris entre une fréquence de coupure basse et une fréquence de coupure haute du filtre. Le concept de filtre passe-bande est une transformation mathématique appliquée à des données. L'implémentation d'un filtre passe-bande peut se faire numériquement ou avec des composantes électroniques. Cette transformation a pour fonction d'atténuer les fréquences à l'extérieur de la bande passante, l'intervalle de fréquences compris entre les fréquences de coupure. Ainsi, uniquement les fréquences comprises dans cet intervalle sont conservées intactes ou avec une faible atténuation. [21]

En traitement du signal audio, ils sont utilisés dans les égaliseurs, qui permettent par exemple d'équilibrer les signaux issus des différents microphones lors d'une prise de son.

Les filtres passe-bandes ont pendant longtemps été réalisés sous forme analogique. Aujourd'hui, ces filtres sont supplantés par leur équivalent numérique, bien plus facile à configurer et offrant une sélectivité supérieure. [24]

Caractéristiques des filtres passe-bande

Les filtres passe-bande ont quelques caractéristiques additionnelles comparativement aux Filtres passe-bas et passe-haut. Ces paramètres sont :

- **Fréquence centrale ω_0**

C'est la fréquence à laquelle la fonction de transfert du filtre est purement réelle. On l'appelle aussi la fréquence de résonance. La fréquence centrale est la moyenne géométrique des fréquences de coupure : $\omega_0 = \sqrt{\omega_{c1}\omega_{c2}}$. Pour un filtre passe-bande, l'amplitude de la fonction de transfert est maximale à la fréquence centrale.

- **Largeur de bande BP**

La bande passante BP d'un filtre passe-bande est l'intervalle de pulsations $[\omega_{c1},\omega_{c2}]$ qui correspond aux pulsations telles que le gain soit au plus à 3 décibels en dessous du gain maximum (ici 0 dB) :

$$G(\omega_{ci}) = G(\omega_0) - 3 \qquad \text{(II.3)}$$

Ces pulsations de coupure sont telles que :

$$|h(j\omega_{c1})| = |h(j\omega_{c2})| = \frac{hmax}{\sqrt{2}} \text{ ; (car } 20\log 1/\sqrt{2} = -3) \quad \text{(II.4)}$$

- **Facteur de qualité Q**

C'est le rapport entre la fréquence centrale et la largeur de bande. Le facteur de qualité est une mesure de la largeur de la bande passante, indépendamment de la fréquence centrale. [22]

5. FILTRE IDÉAL

Un filtre passe-bande idéal a un gain constant dans sa bande passante et un gain nul dans la bande coupée. La transition entre les deux états est instantanée. Dans la réalité, un filtre possède à sa fréquence de coupure un gain *Gmax* −3 dB et ensuite ce gain décroît de −20 dB par décade (soit −6 dB par octave) (filtre de 1er ordre). [20]

6. REALISATION DU BANC DE FILTRES

Le tableau suivant, montres les bandes fréquentielles du banc de filtres de la stratégie de stimulation cochléaire ACE.

Tableau II.1. Les bandes fréquentielles des filtres.

Numéro du filtre	La bande passante
1er filtre	[224 275]
2eme filtre	[342 408]
3eme filtre	[461 539]
4eme filtre	[579 671]
5eme filtre	[697 803]
6eme filtre	[815 935]
7eme filtre	[934 1066]
8eme filtre	[1051 1198]
9eme filtre	[1170 1330]
10eme filtre	[1347 1527]
11eme filtre	[1584 1790]
12eme filtre	[1800 2054]
13eme filtre	[2057 2317]
14eme filtre	[2353 2647]
15eme filtre	[2707 3043]
16eme filtre	[3121 3503]
17eme filtre	[3594 4030]
18eme filtre	[4129 4627]
19eme filtre	[4718 5282]
20eme filtre	[5368 6006]
21eme filtre	[6137 6863]
22eme filtre	[7023 7851]

Le script suivant montre les instructions Matlab utilisées pour générer un filtre passe bande :

```
Fs=16000; % Fréquence d'échantillonnage
N=2; % l'ordre du filtre
Ast1=60; Ast2=80; % 1ere et 2eme stop-band atténuation (dB)
Ap=1; % ondulation passe-bande (dB)
Fp1=224; Fp2=275; % 1ere et 2eme fréquence passe-bande (Hz)
filtre_1=fdesign.bandpass('n,Fp1,Fp2,Ast1,Ap,Ast2',N,Fp1,Fp2,Ast1,Ap,...
Ast2,Fs);
```

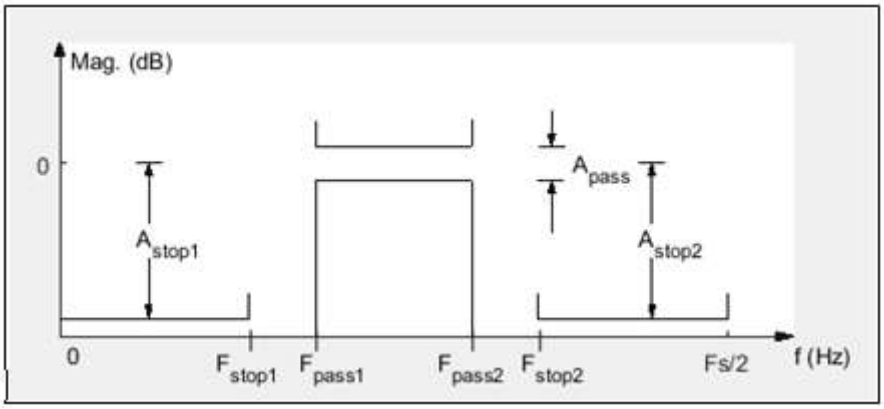


Figure II.1. Les caractéristiques d'un filtre passe-bande.

Pour passer à la simulation du filtre, nous avons choisi la structure elliptique correspondent à la fois à la bande passante et à la bande d'arrêt.

La commande suivante permet de réaliser cette tâche :

```
Hd=design(filtre_1,'ellip');
```

Après la génération du banc de 22 filtres passe-bandes, nous avons représenté la réponse en amplitude de ce dernier grâce à la commande « fvtool » :

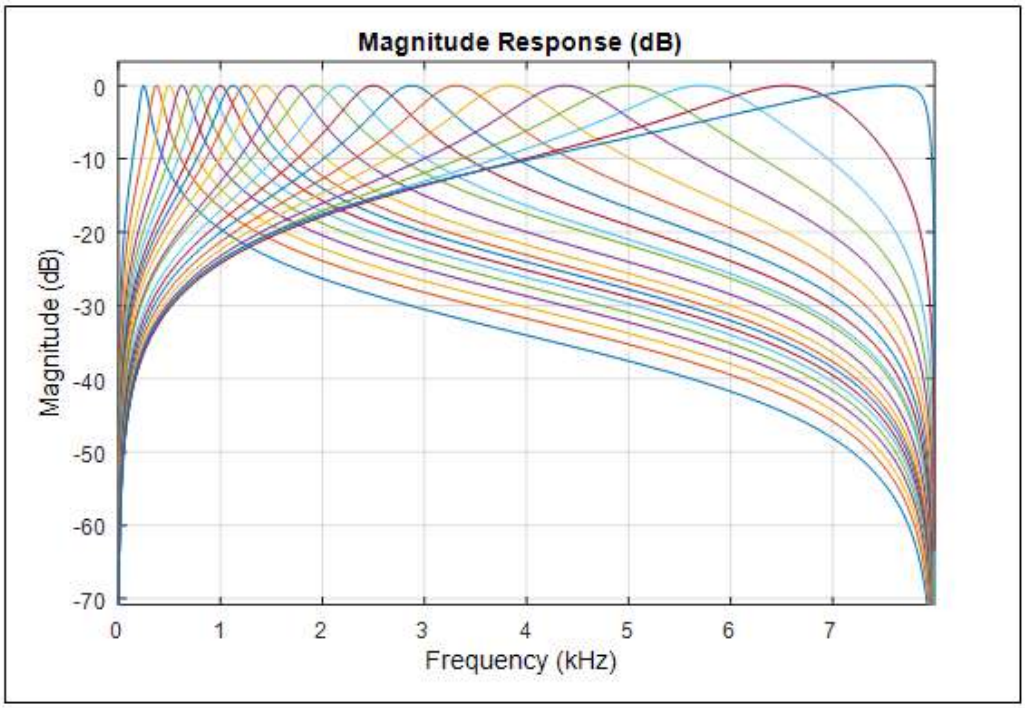


Figure II.2. La réponse en amplitude du banc de filtres.

7. SIMULATION DE LA STRATEGIE ACE EN UTILISANT LE BANC DE FILTRES REALISE

Dans cette étape, nous avons on porté un fichier audio pour tester la stratégie ACE.

```
[s,Fs]=wavread('clean_S.wav');
```

Ensuite, nous devons filtrer le signal audio par le banc de filtre en suivant la commande qui suit :

```
sortie_1=filter(Hd1,s);
```

Pour extraire l'enveloppe de la sortie de chaque filtre du banc, un filtre passe-bas est appliqué à l'énergie de la sortie de chaque filtre du banc. La fréquence de coupure de ce filtre est comprise entre 200 et 400 Hz. [12]

Le script suivant montre comment nous avons réalisé le filtre passe bas.

```
lpFilt = designfilt('lowpassfir', 'PassbandFrequency', 200,...
'StopbandFrequency',400,'PassbandRipple', 9,'StopbandAttenuation',...
65, 'SampleRate', 16e3);
```

La stratégie ACE stimule souvent 8 bandes de fréquences ayant les plus grandes amplitudes. Après le calcul l'énergie pour les 22 filtres, les 8 filtres sélectionnés sont représentés comme suit :

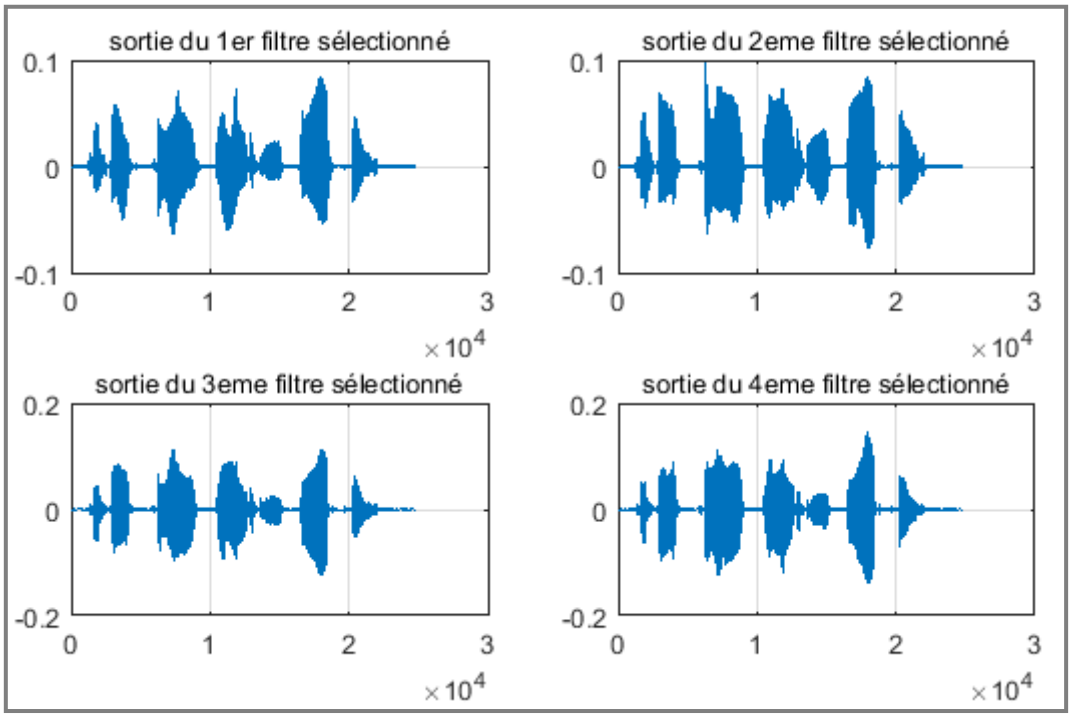


Figure II.3. Les sorties des filtres 1, 2, 3 et 4 sélectionné.

.

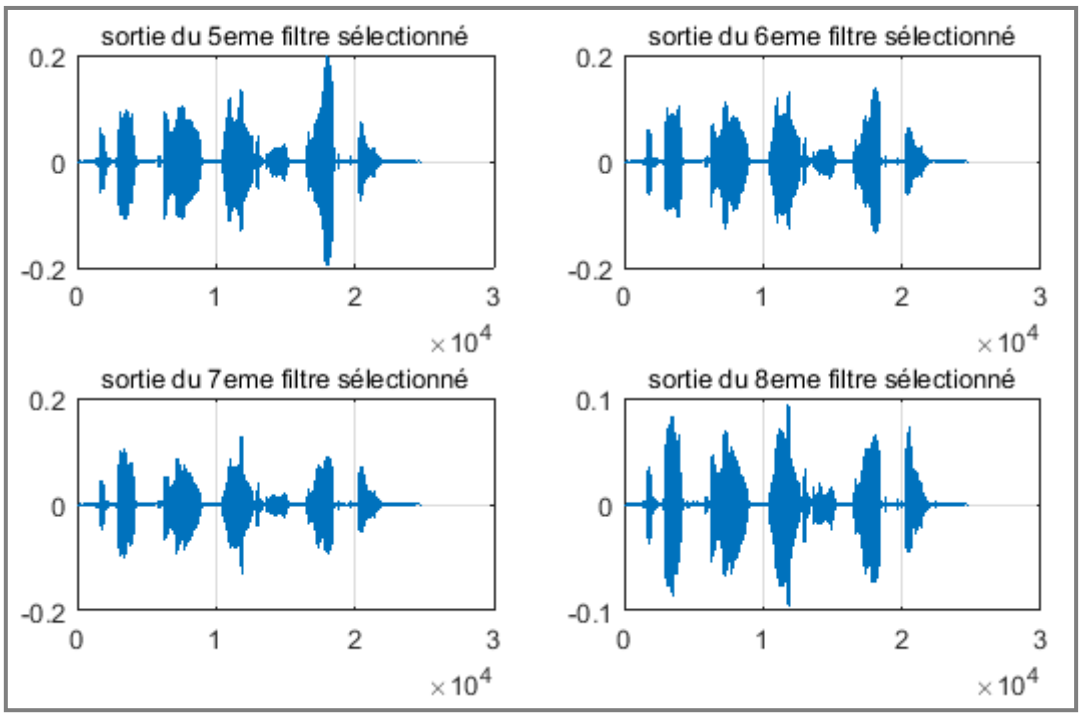


Figure II.4. Les sorties des filtres 5, 6, 7 et 8 sélectionné.

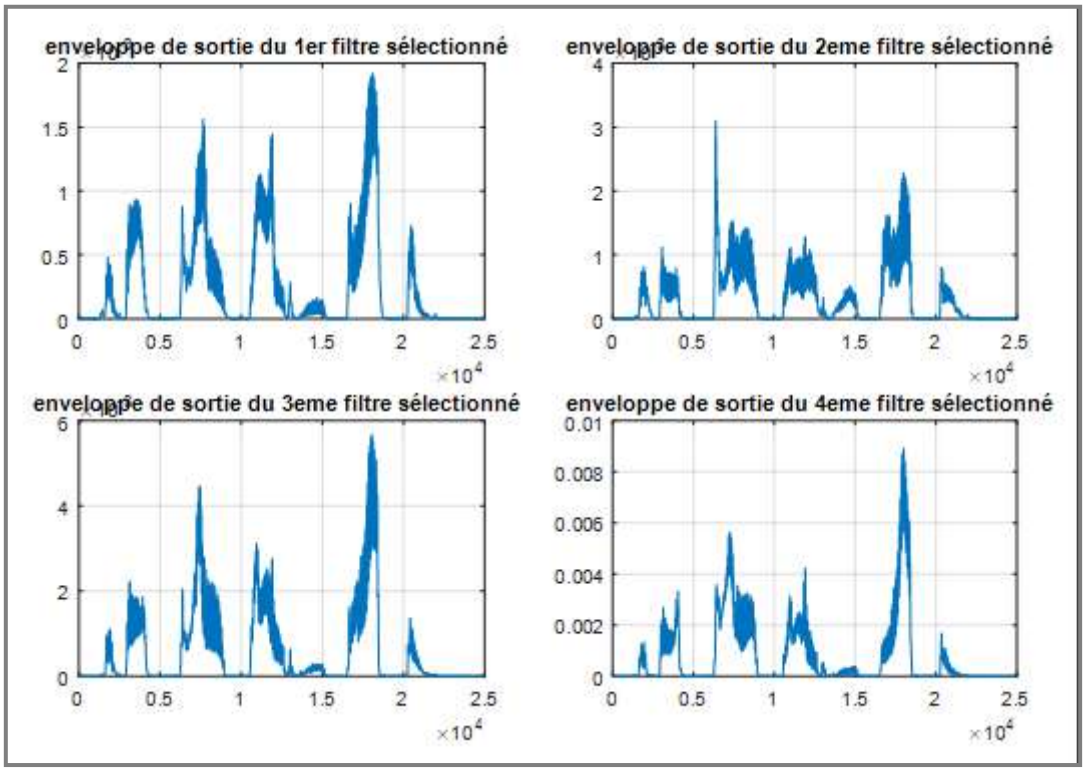


Figure II.5. Enveloppes des filtres 1, 2, 3 et 4 sélectionné.

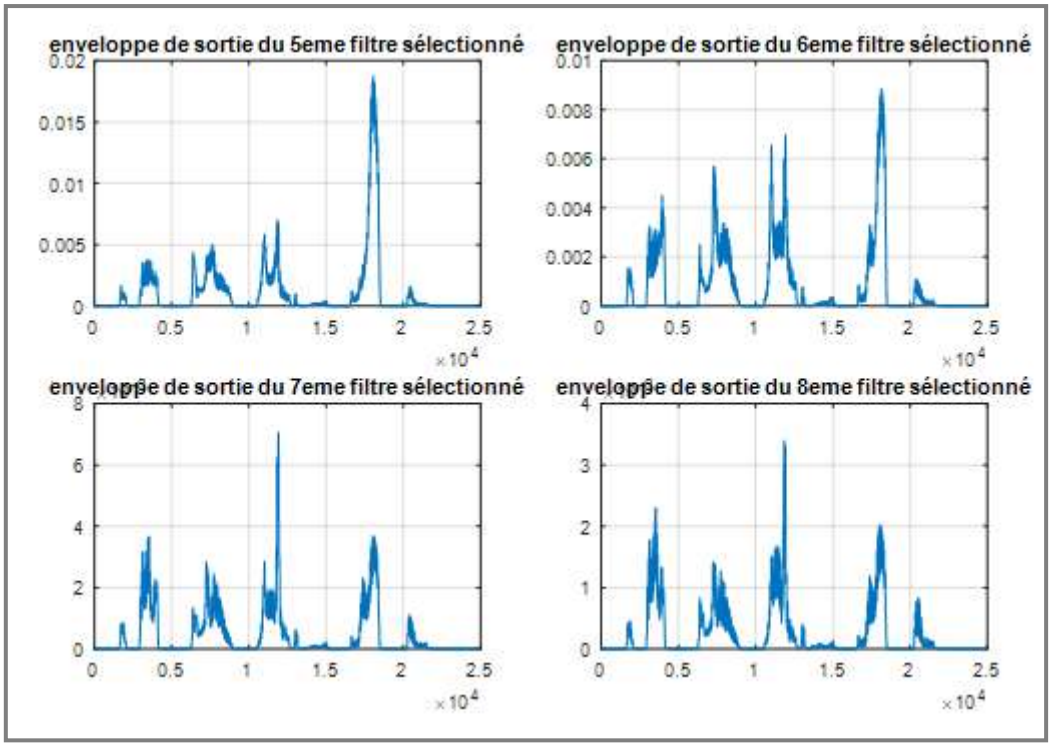


Figure II.6. Enveloppes des filtres 5, 6, 7 et 8 sélectionné.

La modulation des signaux d'enveloppes, nous oblige à créer un signal carré bi-phasique.

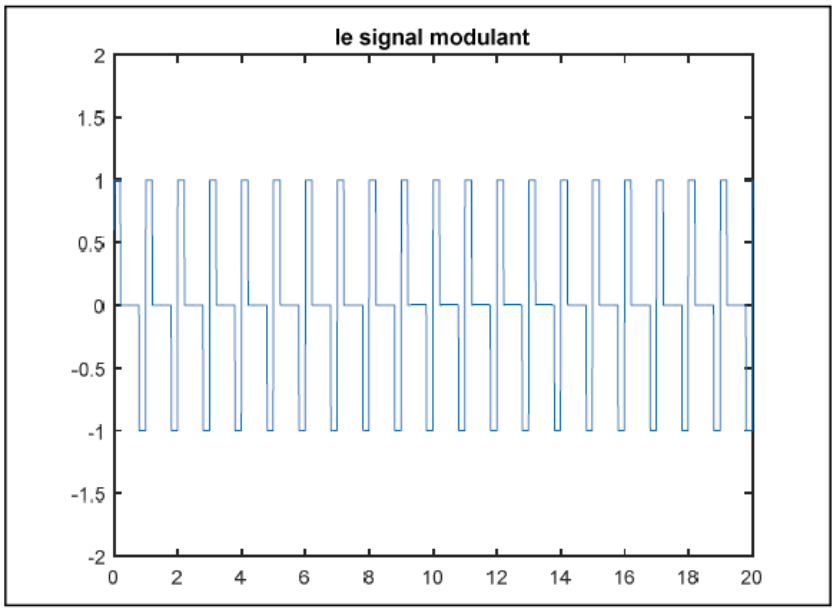


Figure II.7. Le signal à modulé.

Après l'étape de modulation nous obtenons les résultats suivants :

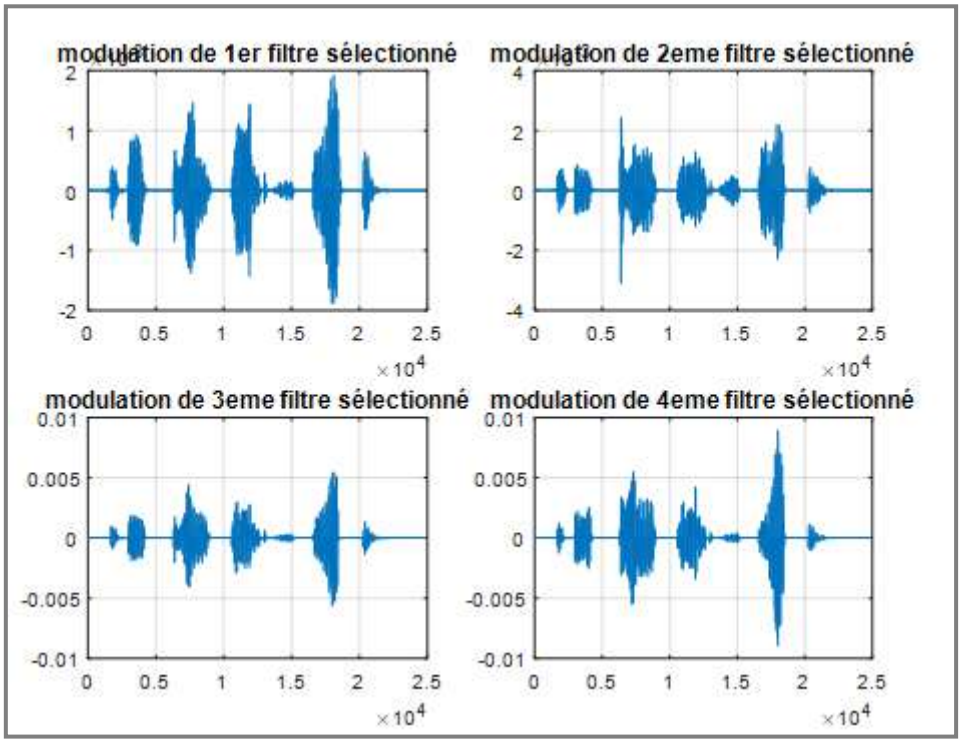


Figure II.8. Les signaux modulés de la stratégie ACE des filtres1, 2, 3 et 4.

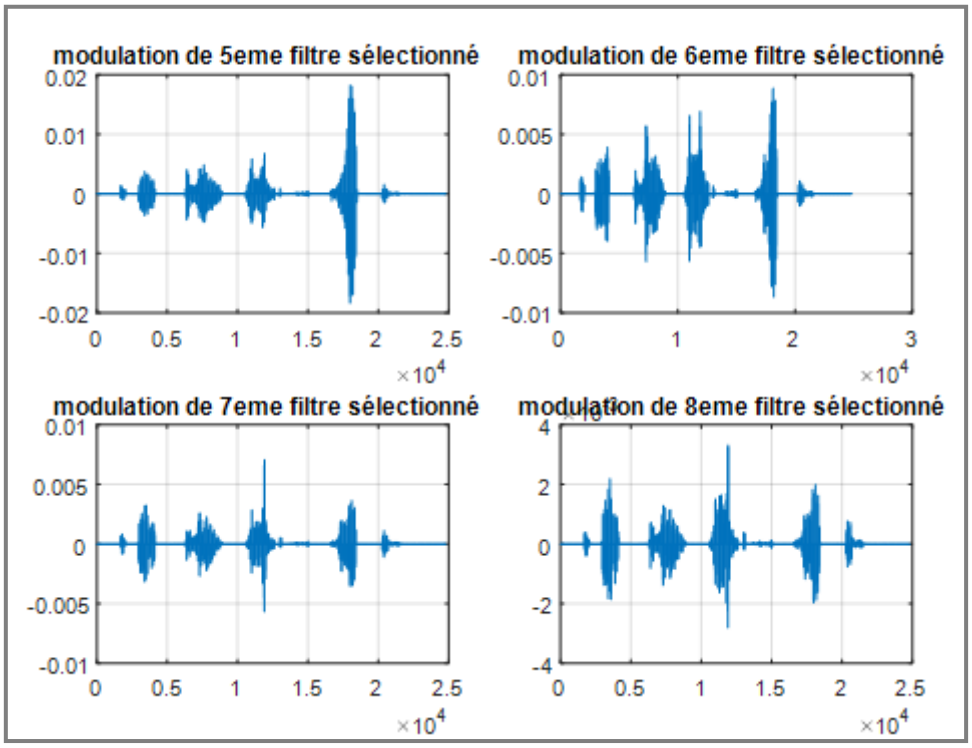


Figure II.9. Les signaux modulés de la stratégie ACE des filtres 5, 6, 7 et 8.

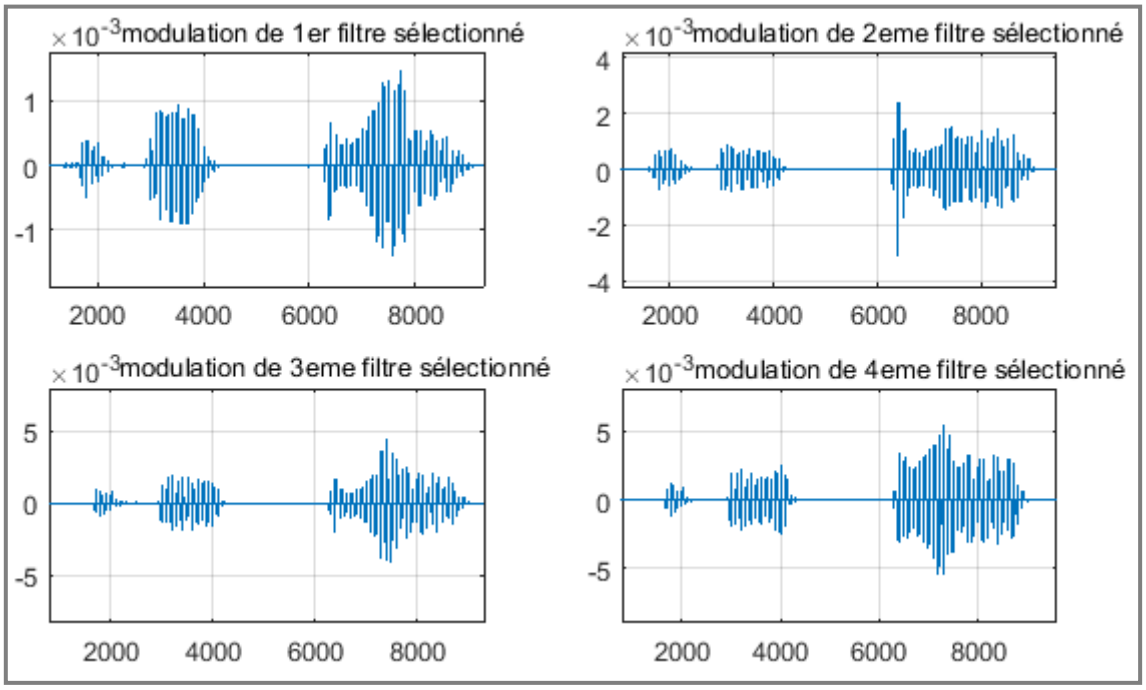


Figure II.10. Une partie zoomé des signaux modulés de la stratégie ACE des filtres 1, 2, 3 et 4.

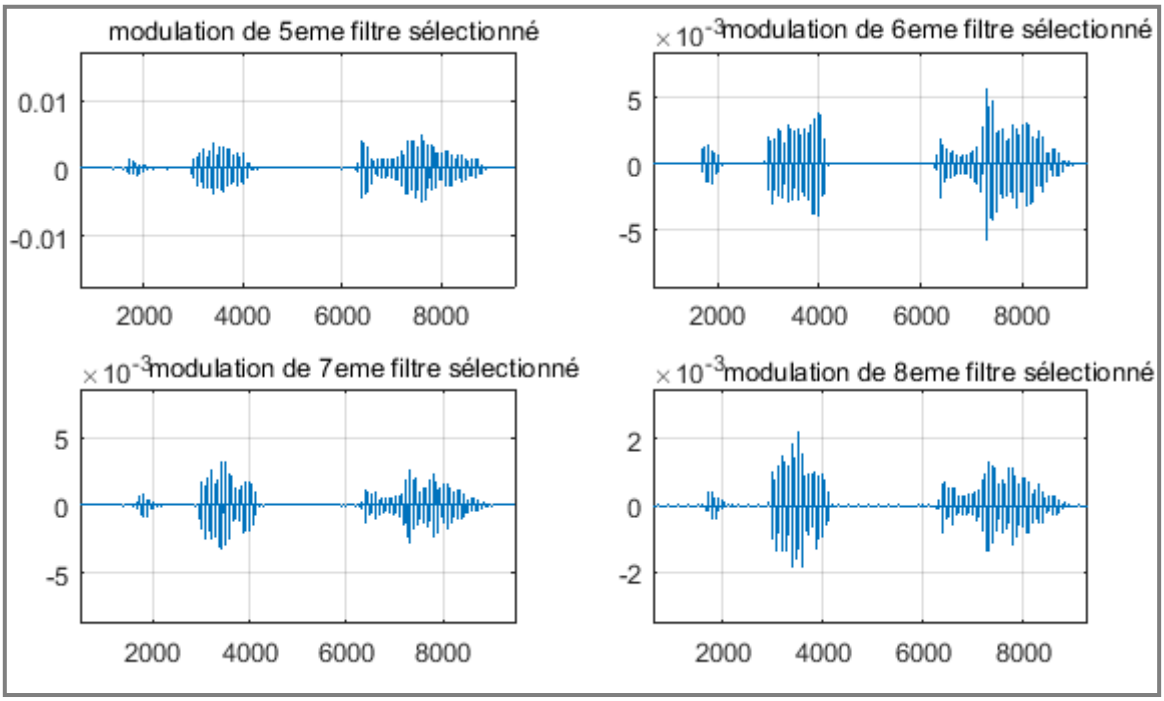


Figure II.11. Une partie zoomé des signaux modulés de la stratégie ACE des filtres 5, 6, 7 et 8.

8. CONCLUSION

Dans ce chapitre, nous avons présenté la stratégie de stimulation cochléaire, la plus largement utilisée dans les implants commercialisés, ainsi que les caractéristiques des filtres, qui jouent un rôle important dans la stratégie du codage avancé (ACE), ensuit nous avons passé à la simulation Matlab de cette dernière stratégie.

Nous présenterons dans le troisième chapitre la méthode de génération du programme VHDL et du «Tb» (test-bench) dans Matlab et l'implémentation sur un circuit FPGA en utilisant le logiciel ISE de Xilinx.

CHAPITRE III

SIMULATION ET IMPLEMENTATION DU BANC DE FILTRES EN MODELSIM

1. INTRODUCTION

Nous présenterons dans ce troisième chapitre la méthode de génération du programme VHDL et du "Tb" (test-bench) pour le banc de 22 filtres, ensuite, nous faisons une simulation commune avec le logiciel Modelsim et avec Matlab et la comparaison entre eux afin de confirmer la justesse du programme VHDL généré.

Après la récolte des résultats de simulation du programme VHDL, nous passons à l'étape de l'implémentation sur un circuit FPGA en utilisant le logiciel ISE de Xilinx.

2. QUANTIFICATION DES FILTRES

Dans cette étape, nous expliquons l'étape de la quantification, le format des données utilisées par chacun des filtres est en point fixe. Nous définissons 8 bits pour la longueur du mot et 7 bits pour la longueur de fraction du signal d'entrée du filtre, 9 bits pour la longueur de mot utilisée pour les données de sortie, et 20 bits pour le numérateur et le dénominateur de la fonction de transfert du chaque filtre, 20 bit pour la longueur de mot utilisée pour stocker les données dans l'accumulateur, et 16 bits pour la longueur de mot à appliquer aux coefficients des filtres.

Pour le mode utilisé par le filtre pour mesurer la taille des données filtrées pour la sortie, nous avons sélectionné **'SpecifyPrecision'** car il permet de spécifier la longueur des mots et des fractions utilisés par les données de sortie du filtre, que ce soit pour le mode utilisé pour déterminer les valeurs numériques, nous avons choisi **'nearest'** Arrondir vers le plus proche. Où les liens tournent vers l'infini positif, et le mode utilisé pour répondre aux conditions de débordement dans l'arithmétique en virgule fixe nous avons choisi **'wrap'** (définir les valeurs en débordement sur la valeur représentable la plus proche en utilisant l'arithmétique modulaire).

La propriété de format « CastBeforeSum » indique s'il faut convertir les données numériques dans le format d'accumulateur approprié et « OutputFracLength » détermine comment le filtre interprète les données de sortie du filtre.

Mais avant de spécifier ça, la structure de filtre par défaut (forme directe 2: DF2SOS) doit être convertie en la structure requise (forme directe 1: DF1SOS), cette dernière est la forme directe la plus courante, car elle correspond tout simplement à l'équation aux différences, et est donnée par l'équation :

$$y_k = \sum_{i=0}^{n} b_i u_{k-i} - \sum_{i=1}^{n} a_i y_{k-i} \qquad \text{(III.1)}$$

Où *n* est l'ordre du filtre.

Et la forme directe 2 (DF2) permet d'écrire un filtre sous une forme plus compacte que la DF1 en mutualisant les opérateurs retard.

La DF1 est décrite par l'équation suivant :

$$\begin{cases} e_k = u_k - \sum_{i=1}^{n} a_i \, e_{k-i} \\ y_k = \sum_{i=0}^{n} b_i \, e_{k-i} \end{cases} \quad \text{(III.2)}$$

Nous avons choisi (forme direct 1: DF1SOS) car c'est la seule structure qui permet la quantification dans Matlab.

```
Hd1=convert(Hd1,'df1sos');
Hd1.arithmetic = 'fixed';
Hd1.InputWordLength = 8;
Hd1.InputFracLength = 7;
Hd1.OutputWordLength = 9;
Hd1.OutputMode = 'SpecifyPrecision';
Hd1.OutputFracLength = 7;
Hd1.CoeffWordLength = 16;
Hd1.AccumWordLength = 20;
Hd1.NumStateWordLength = 20;
Hd1.DenStateWordLength = 20;
Hd1.CastBeforeSum = false;
Hd1.RoundMode = 'nearest';
Hd1.OverflowMode = 'wrap';
```

Par la commande fvtool du Matlab nous avons vérifié la réponse en amplitude pour un seul filtre du banc (premier filtre). La même étape sera répétée sur tous les filtres du banc.

```
fvtool(Hd1, 'Fs', Fs, 'FrequencyScale', 'log');
```

Après avoir la quantification, la figure III.1 montre qu'il n'y avait pas de changement entre la réponse en amplitude du filtre réel (premier filtre) et le filtre quantifié.

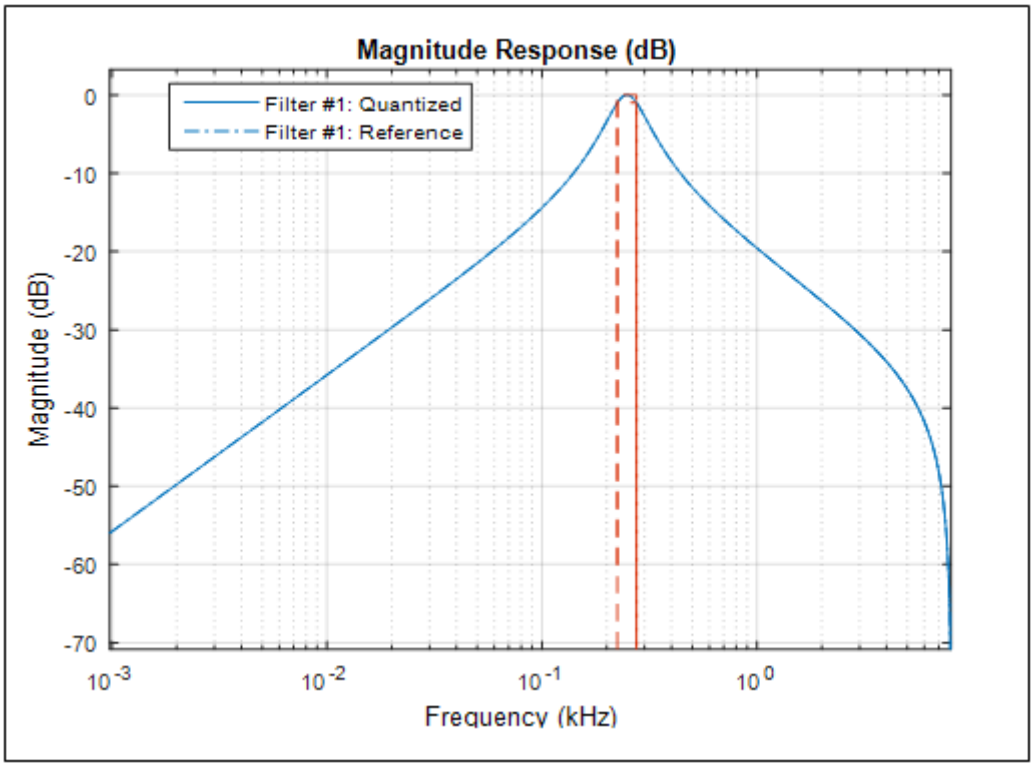


Figure III.1. Réponse en amplitude après quantification du 1er filtre.

Pour vérifier si les valeurs caractéristiques du filtre sont à une échelle raisonnable ou ajuster l'échelle, nous faisons la commande suivante :

```
scale(Hd1,'Linf');
```

3. GENERATION DU CODE VHDL DU BANC DE FILTRES

Le texte suivant montre comment créer un espace de travail temporaire (temporaire), et générons le code VHDL pour ce filtre.

```
workingdir = tempname;
generatehdl(Hd1,'Name','filtre1','TargetLanguage','VHDL',...
'TargetDirectory',workingdir);
edit(fullfile(workingdir, 'filtre1.vhd'));
```

Le fichier généré porte le nom de "filtre1.vhd"

4. GENERATION DE « TEST BENCH » DU BANC DE FILTRES

Afin de vérifier que les résultats correspondent exactement aux résultats obtenus sous Matlab, Nous construisons le signal d'entrée de test avec un cycle de 50 à 300 Hz avec un pas de 50 Hz, et l'appliquons à la structure du filtre VHDL , Ceci est dû au fait que la bande passante de ce filtre est très faible par rapport à la fréquence d'échantillonnage.

```
stimulus = [];
for n = [50, 100, 150, 200, 250, 300]
stimulus = [stimulus, sin(2*pi*n/Fs*(0:Fs/n))];
end
generatehdl(Hd1,'Name','filtre1','TargetLanguage','VHDL',...
'TargetDirectory',workingdir,'GenerateHDLTestbench','on',...
'TestBenchUserStimulus', stimulus);
edit(fullfile(workingdir, ' filtre1_tb.vhd'));
```

Le fichier généré porte le nom de “filtre1_tb.vhd”

5. SIMULATION DU PROGRAMME VHDL PAR MODELSIM

Nous commençons par la visualisation du signal de sortie des premières 8 filtres en Matlab. Les mêmes étapes seront répétées sur tous les filtres du banc.

Nous devons passer à la simulation du programme «filtre1.vhd» par Modelsim, et le comparer avec Le résultat de la simulation des premières 8 filtres obtenu par Matlab.

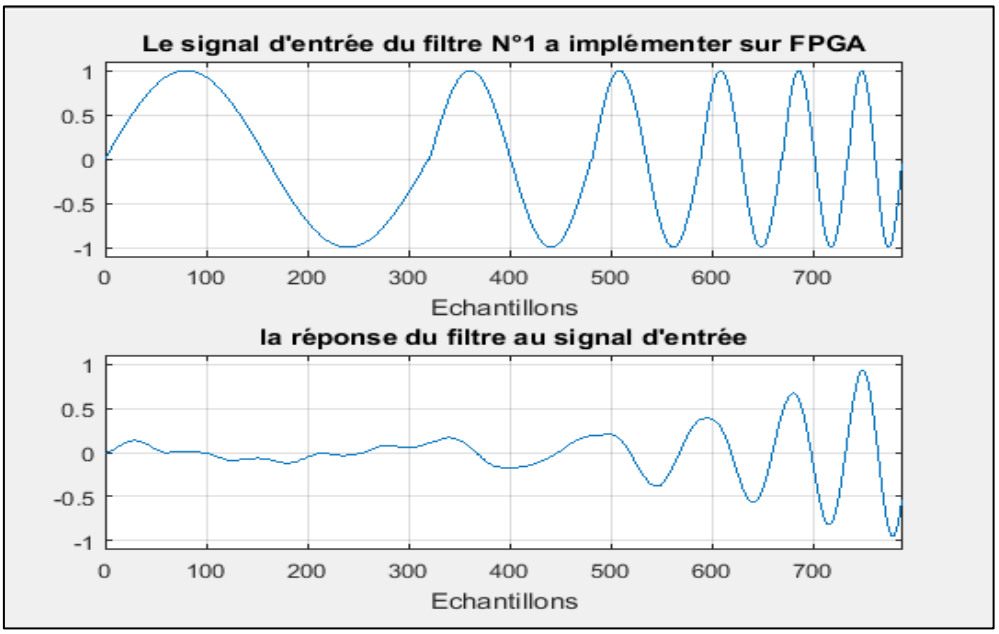


Figure III.2. Le résultat de la simulation du 1er filtre obtenu par Matlab.

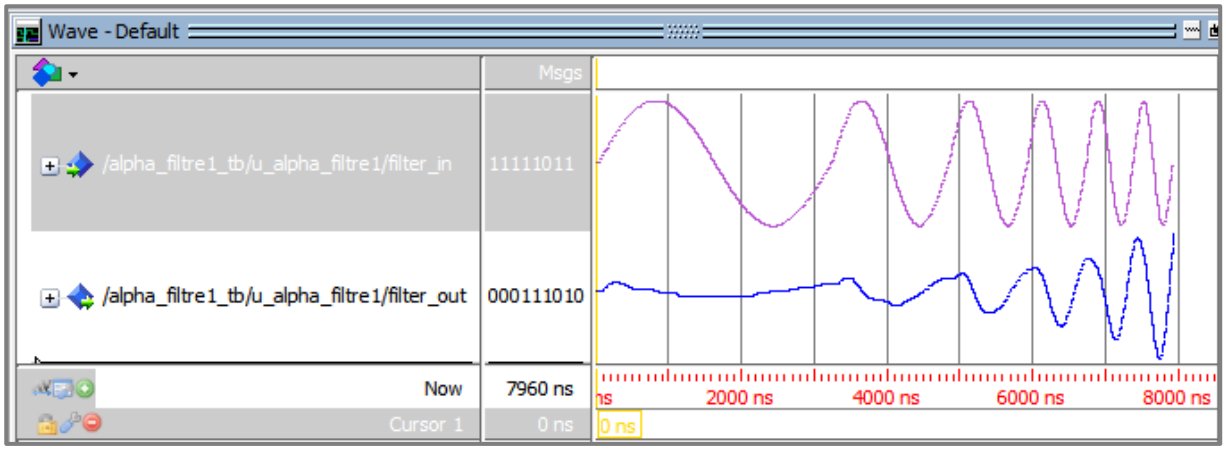


Figure III.3. Le résultat de la simulation du 1er filtre obtenu par Modelsim.

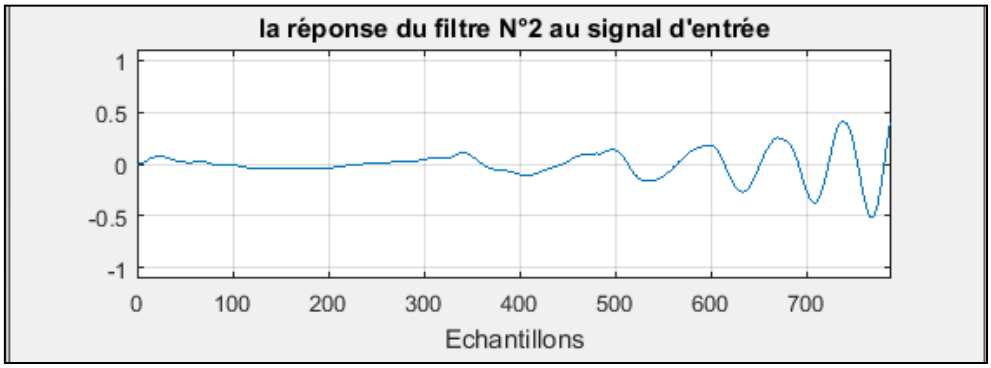


Figure III.4. Le résultat de la simulation du 2eme filtre obtenu par Matlab.

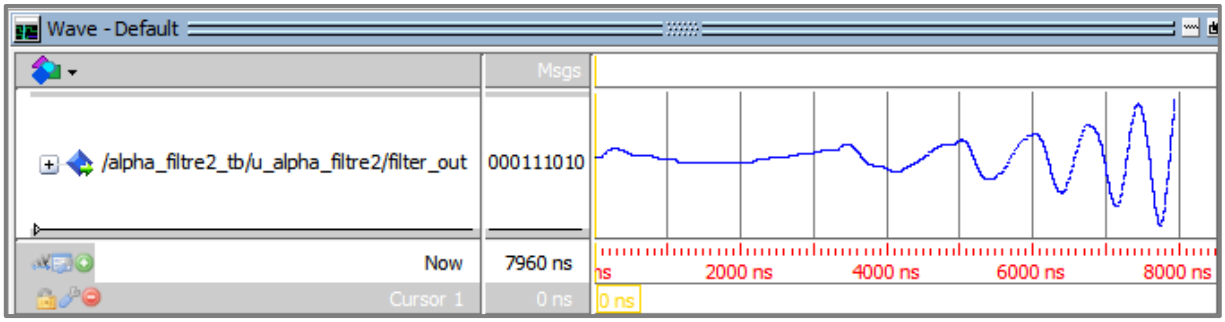


Figure III.5. Le résultat de la simulation du 2eme filtre obtenu par Modelsim.

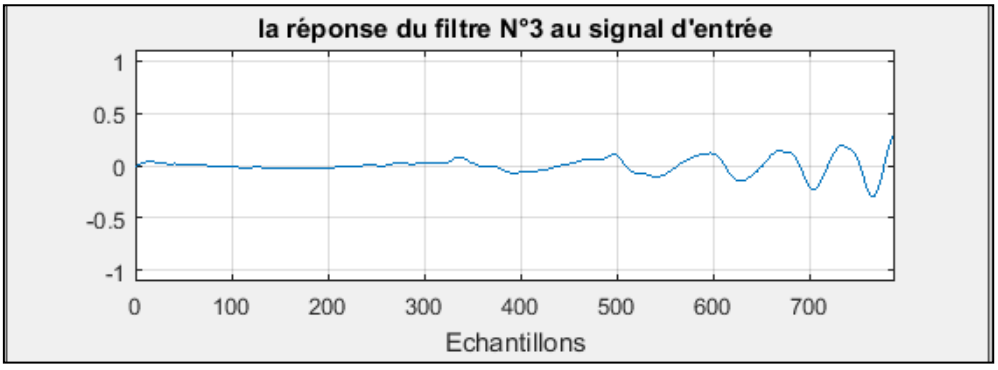


Figure III.6. Le résultat de la simulation du 3eme filtre obtenu par Matlab.

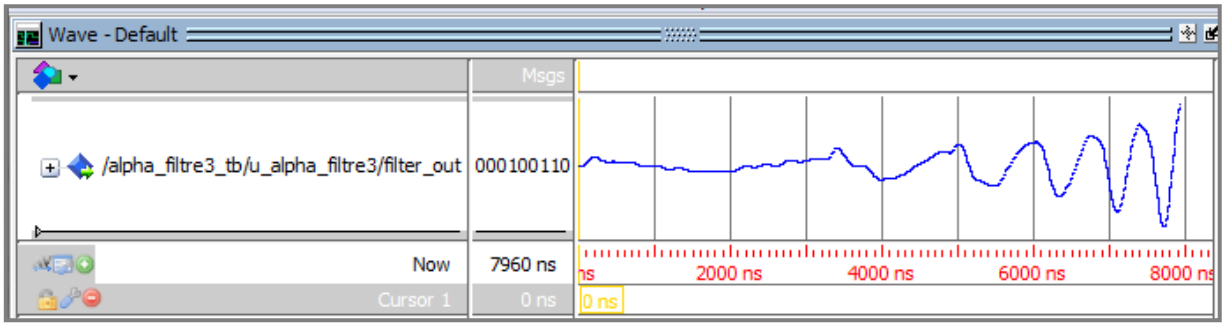


Figure III.7. Le résultat de la simulation du 3eme filtre obtenu par Modelsim.

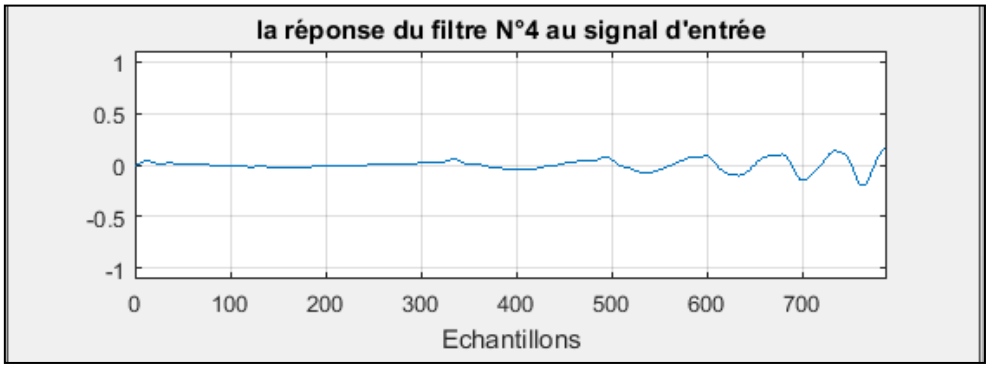


Figure III.8. Le résultat de la simulation du 4eme filtre obtenu par Matlab.

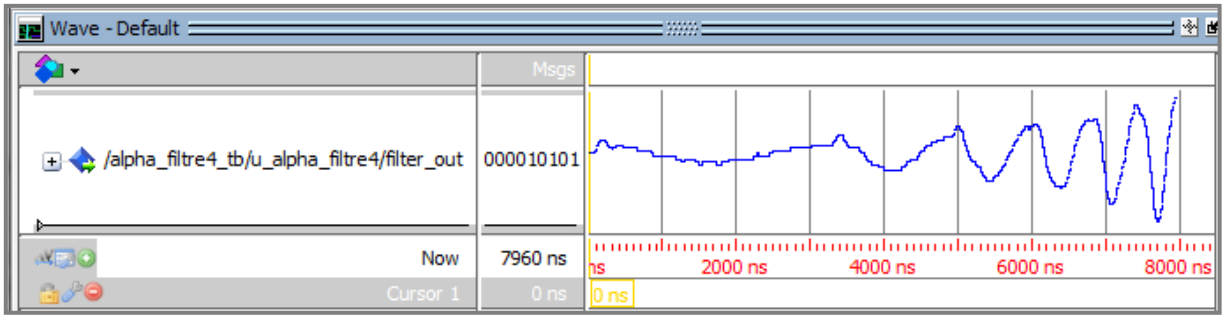


Figure III.9. Le résultat de la simulation du 4eme filtre obtenu par Modelsim.

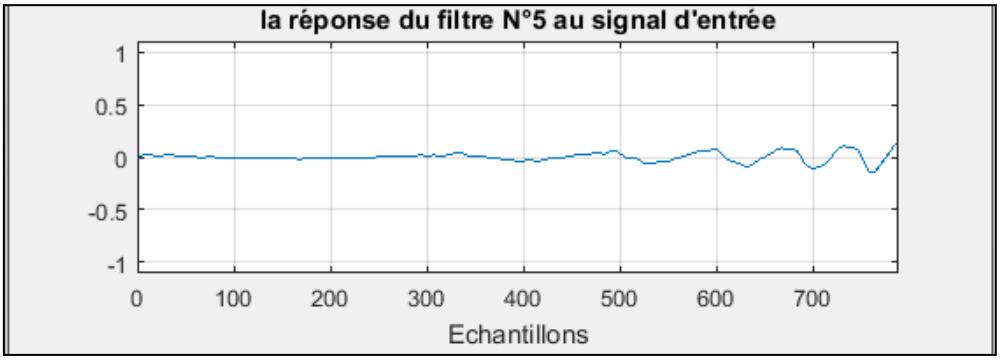


Figure III.10. Le résultat de la simulation du 5eme filtre obtenu par Matlab.

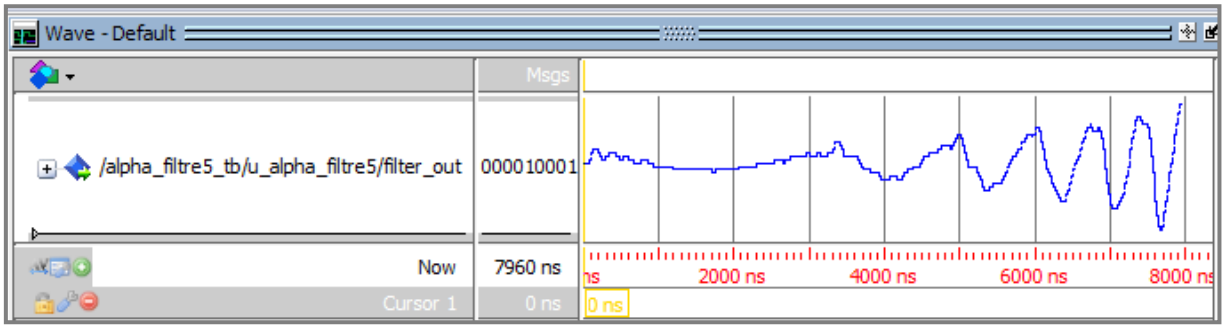


Figure III.11. Le résultat de la simulation du 5eme filtre obtenu par Modelsim.

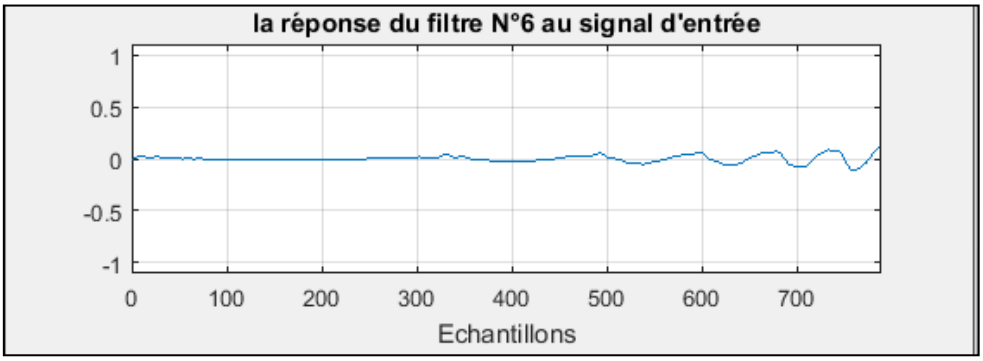


Figure III.12. Le résultat de la simulation du 6eme filtre obtenu par Matlab.

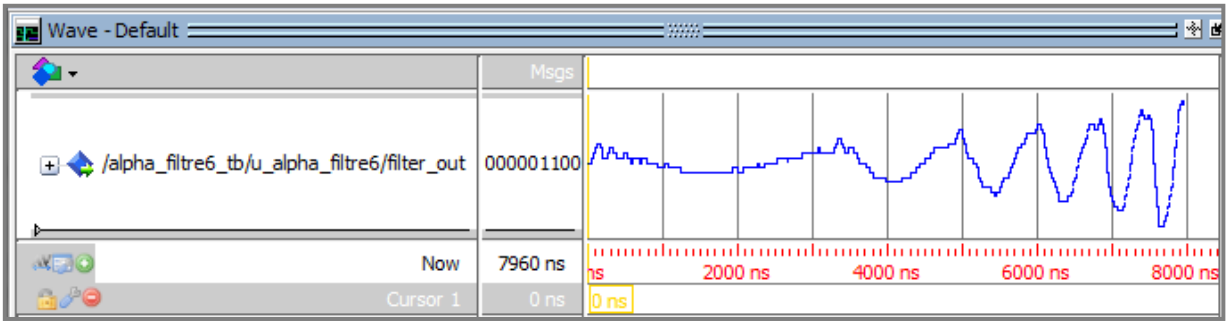


Figure III.13. Le résultat de la simulation du 6eme filtre obtenu par Modelsim.

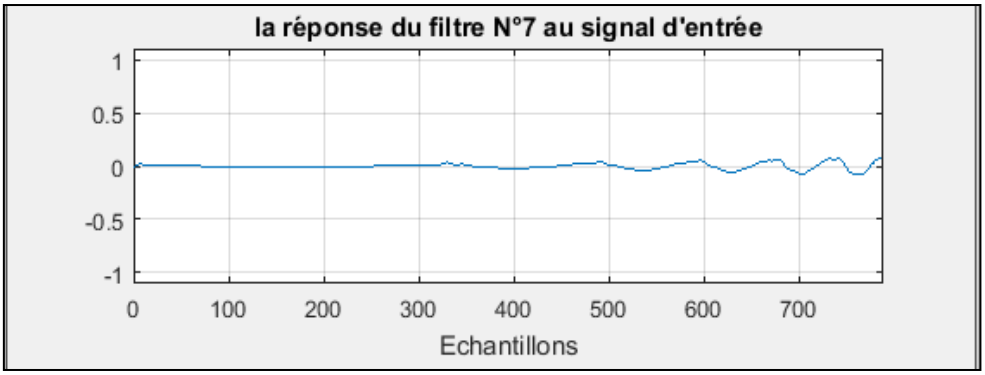


Figure III.14. Le résultat de la simulation du 7eme filtre obtenu par Matlab.

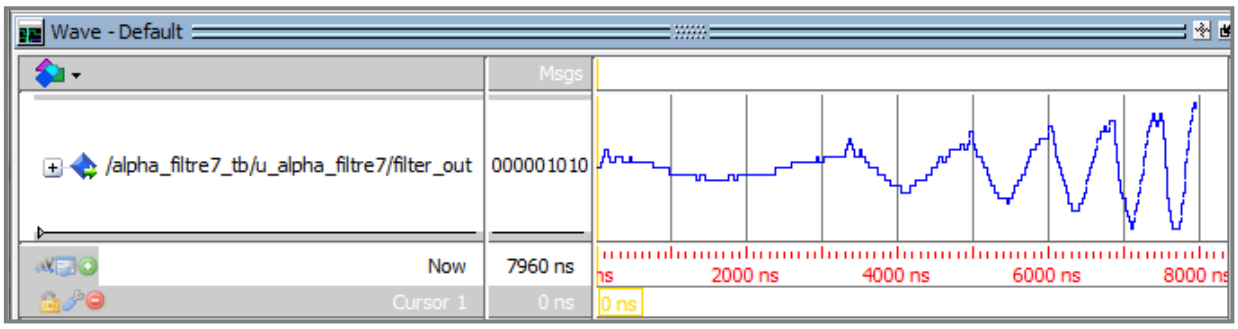


Figure III.15. Le résultat de la simulation du 7eme filtre obtenu par Modelsim.

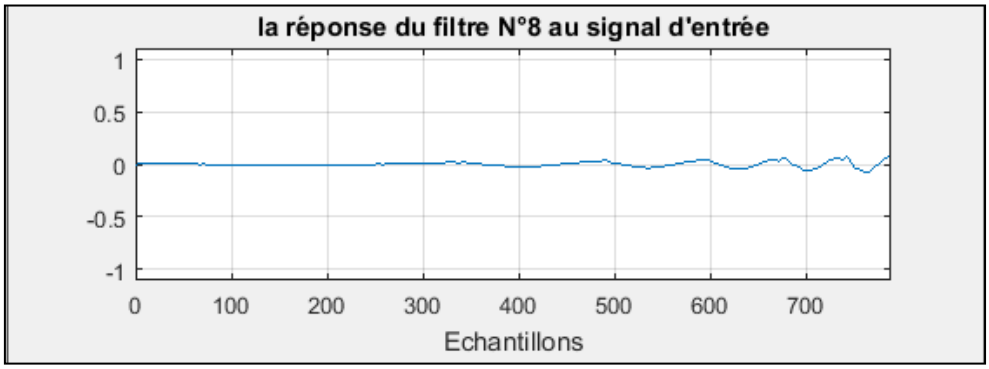


Figure III.16. Le résultat de la simulation du 8eme filtre obtenu par Matlab.

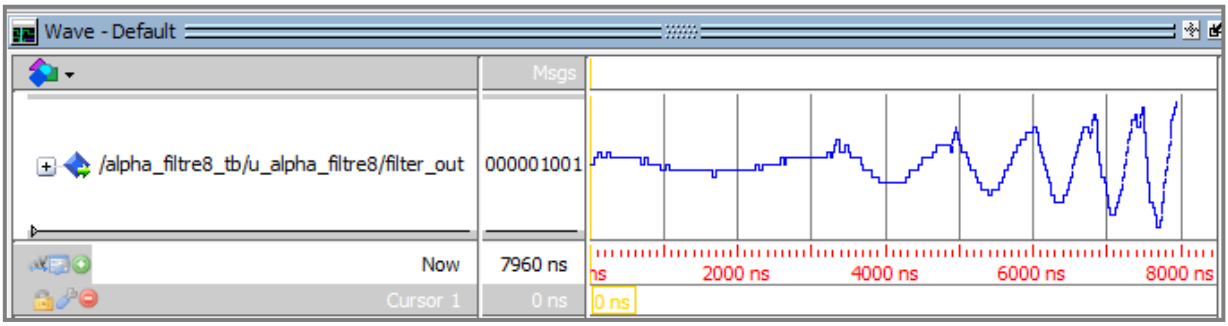


Figure III.17. Le résultat de la simulation du 8eme filtre obtenu par Modelsim.

Après avoir vérifié que les résultats sont entièrement compatibles entre Modelsim et Matlab, nous modifions le signal d'entrée de test par le signal audio et Nous visualisons le signal de sortie des 8 filtres sélectionnées.

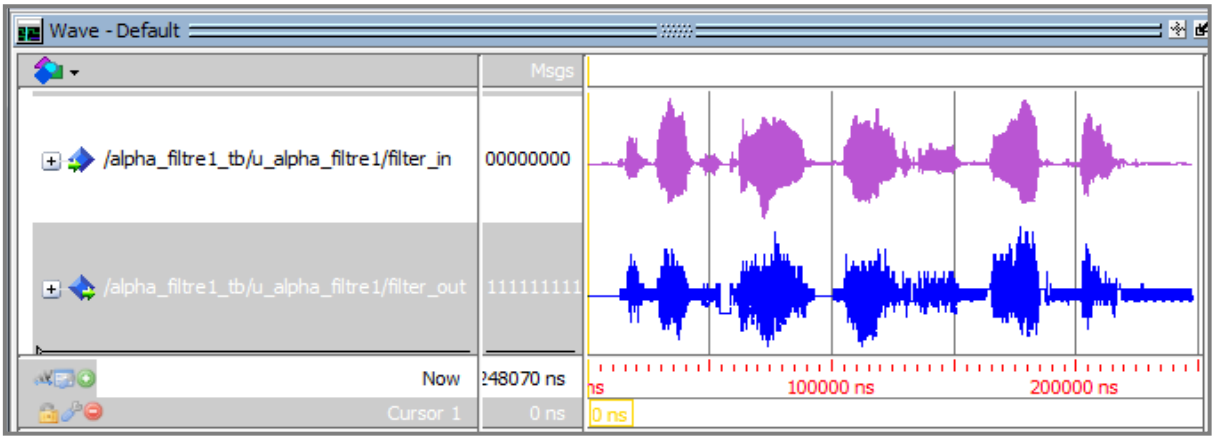


Figure III.18. Le résultat de la simulation du 1er filtre obtenu par Modelsim.

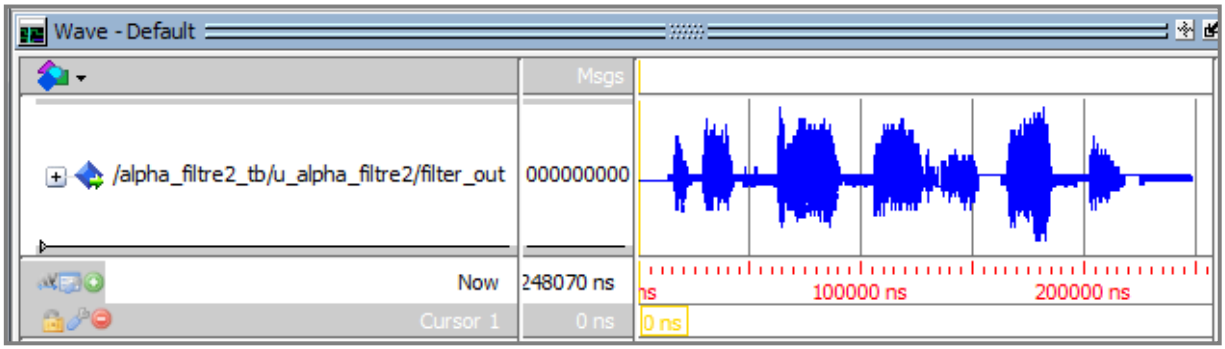


Figure III.19. Le résultat de la simulation du 2eme filtre obtenu par Modelsim.

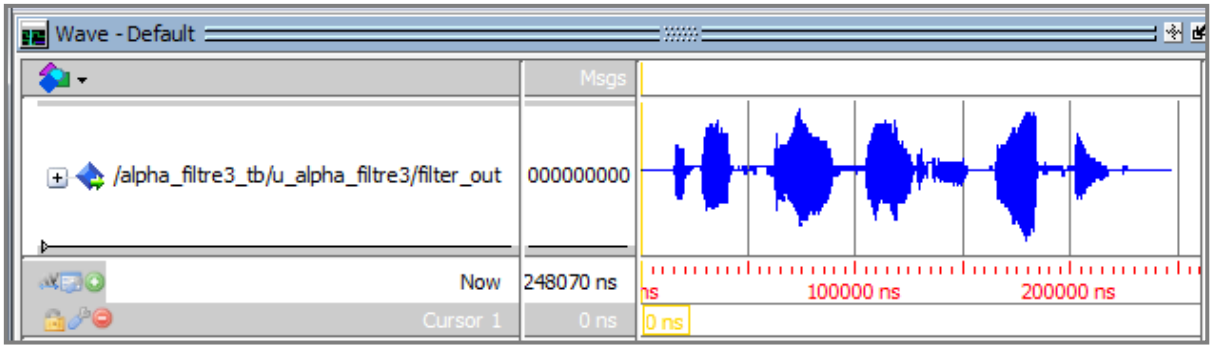


Figure III.20. Le résultat de la simulation du 3eme filtre obtenu par Modelsim.

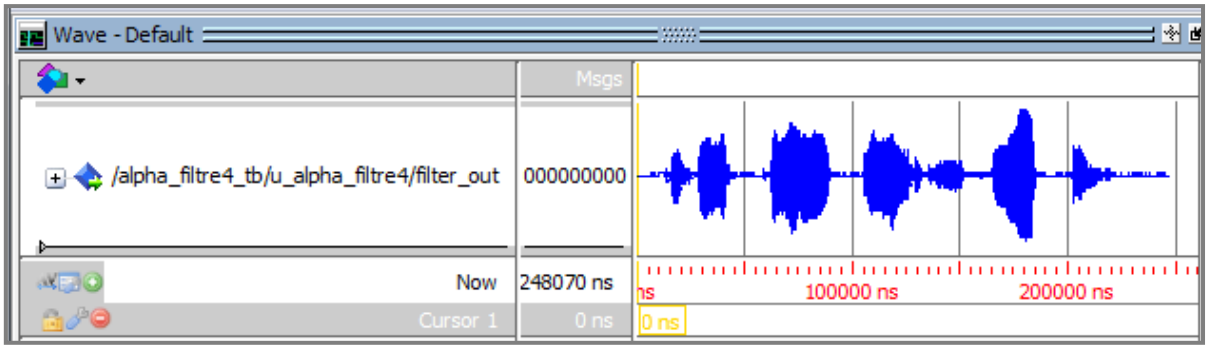


Figure III.21. Le résultat de la simulation du 4eme filtre obtenu par Modelsim.

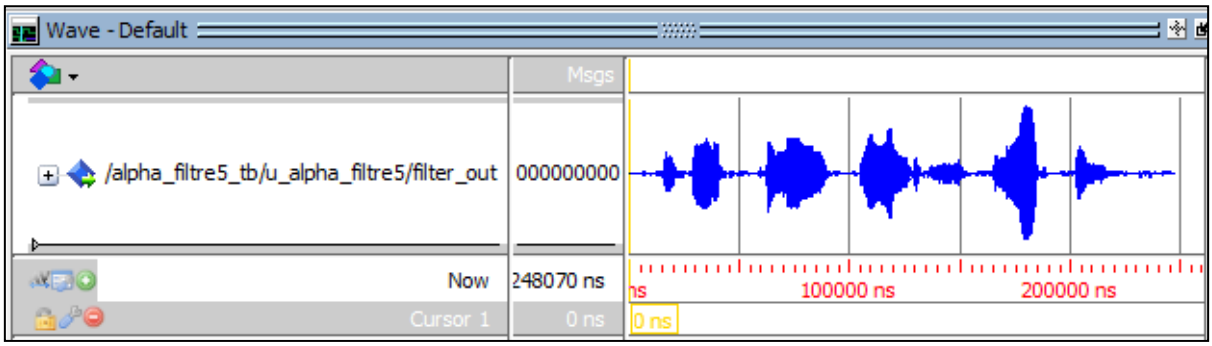


Figure III.22. Le résultat de la simulation du 5eme filtre obtenu par Modelsim.

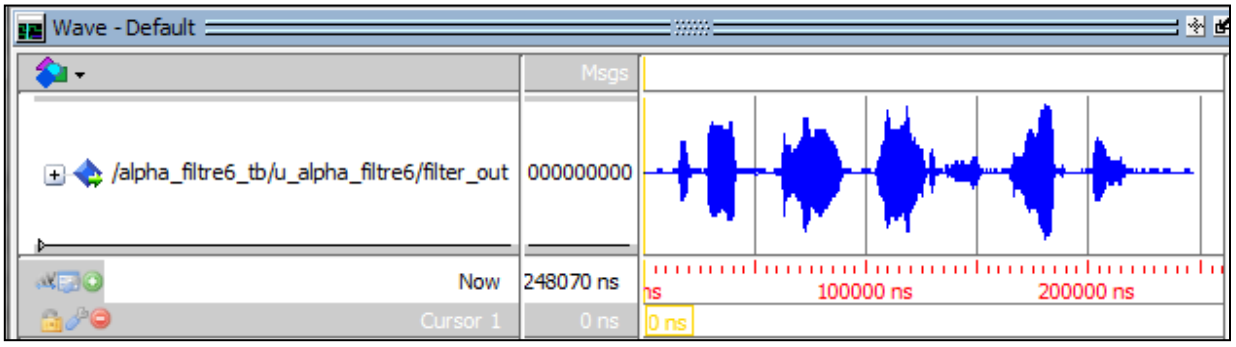


Figure III.23. Le résultat de la simulation du 6eme filtre obtenu par Modelsim.

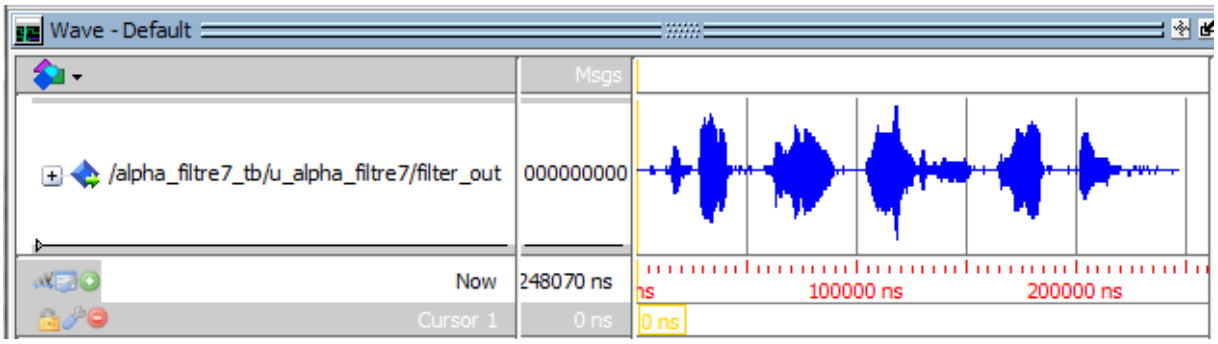


Figure III.24. Le résultat de la simulation du 7eme filtre obtenu par Modelsim.

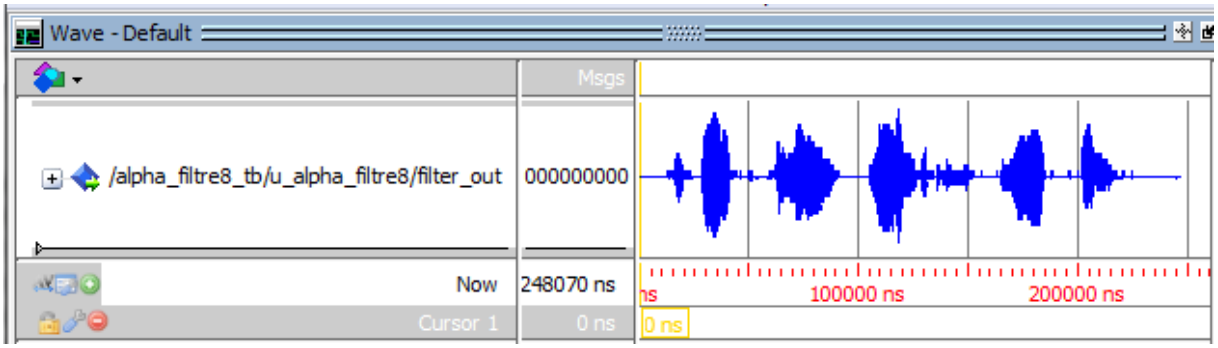


Figure III.25. Le résultat de la simulation du 8eme filtre obtenu par Modelsim.

Tous ces opérations seront répéter pour tous les filtres afin d'obtenir un banc de filtres composer de 22 filtres numériques de type passe-bande qui approxime la cochlée pathologique.

6. LOGICIEL XILINX ISE

Le logiciel Xilinx ISE (Integrated Software Environment) est un outil de conception de circuit pour les FPGAs de Xilinx. Ce logiciel permet essentiellement d'effectuer les différentes étapes propres à la synthèse de circuits numériques sur FPGA. Il est alors possible d'en faire l'implémentation sur les différentes familles de puces fournies par Xilinx.

6.1.Création d'un projet

Avant de créer un nouveau projet sur l'interface d'ISE, il faut sélectionner la vue « implémentation » dans « View » (sur la gauche de l'interface, au-dessus de la petite fenêtre « Hierarchy »).

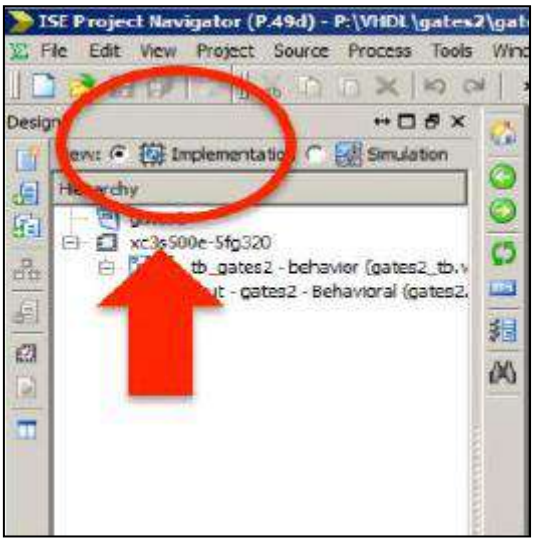


Figure III.26. Choix d'implémentation.

Cliquez sur le bouton **New Project** qui se situe à gauche (ou faites File >**New Project**).



Figure III.27. Création d'un nouveau projet.

Ensuite, nous obtenons une fenêtre où devons Choisissez un nom représentatif pour le projet. Il faut aussi spécifier un répertoire où le projet sera sauvegardé.

6.2.Chargement du code VHDL du banc de filtres sous ISE

Dans la fenêtre de Design, afin d'y ajouter des fichiers, faisons un clic droit sur le projet et cliquons sur **Add Source**, accédons au fichier source et sélectionnons-le, Cliquons sur Ouvrir.

Nous verrons alors un nouveau fichier VHDL dans notre projet.

```
1  LIBRARY IEEE;
2  USE IEEE.std_logic_1164.all;
3  USE IEEE.numeric_std.ALL;
4
5  ENTITY banc_filtres IS
6     PORT( clk                          :   IN    std_logic;
7           clk_enable                   :   IN    std_logic;
8           reset                        :   IN    std_logic;
9           filter_in                    :   IN    std_logic_vector(7 DOWNTO 0); -- sfix8_En7
10          filter_out1,filter_out2,filter_out3, filter_out4,filter_out5,filter_out6,filter_out7,filter_out8,filter_out!
11          );
12 END banc_filtres;
13
14 -------------------------------------------------------------
15 --Module Architecture: banc_filtres
16 -------------------------------------------------------------
17 ARCHITECTURE rtl OF banc_filtres IS
18   -- Local Functions
19   -- Type Definitions
20   TYPE numdelay_pipeline_type IS ARRAY (NATURAL range <>) OF signed(19 DOWNTO 0); -- sfix20_En15
21   TYPE dendelay_pipeline_type IS ARRAY (NATURAL range <>) OF signed(19 DOWNTO 0); -- sfix20_En15
22   -- Constants
23   CONSTANT coeff_b1_section1              : signed(15 DOWNTO 0) := to_signed(26044, 16); -- sfix16_En20
24   CONSTANT coeff_b2_section1              : signed(15 DOWNTO 0) := to_signed(0, 16); -- sfix16_En20
25   CONSTANT coeff_b3_section1              : signed(15 DOWNTO 0) := to_signed(-26044, 16); -- sfix16_En20
26   CONSTANT coeff_a2_section1              : signed(15 DOWNTO 0) := to_signed(-31611, 16); -- sfix16_En14
27   CONSTANT coeff_a3_section1              : signed(15 DOWNTO 0) := to_signed(15570, 16); -- sfix16_En14
28 -- Constants2
29  CONSTANT coeff_b1_section2              : signed(15 DOWNTO 0) := to_signed(26044, 16); -- sfix16_En20
30   CONSTANT coeff_b2_section2              : signed(15 DOWNTO 0) := to_signed(0, 16); -- sfix16_En20
31   CONSTANT coeff_b3_section2              : signed(15 DOWNTO 0) := to_signed(-26044, 16); -- sfix16_En20
32   CONSTANT coeff_a2_section2              : signed(15 DOWNTO 0) := to_signed(-31611, 16); -- sfix16_En14
33   CONSTANT coeff_a3_section2              : signed(15 DOWNTO 0) := to_signed(15570, 16); -- sfix16_En14
```

Figure III.28. Les bibliothèques et les entités et les architectures des filtres.

6.3.Synthèse et implémentation du projet

La synthèse d'un circuit consiste à traduire la description du circuit en blocs disponibles dans la technologie utilisée.

L'implémentation du circuit est divisée en quatre sous étapes :

- la transformation (mapping) : regrouper les composantes obtenues lors de la synthèse dans des blocs spécifiques du FPGA.
- la disposition (placement) : choisir des endroits spécifiques sur le FPGA où disposer les blocs utilisés, et choisir les pattes du FPGA correspondant aux ports d'entrée et de sortie.
- le routage (routin): établir des connexions électriques entre les blocs utilisés.
- la configuration (configuration) : convertir toute cette information en un fichier pouvant être téléchargé sur le FPGA pour le programmer.

Pendant l'étape de disposition de l'implémentation, il faut assigner des pattes spécifiques du FPGA à des ports d'entrée et de sortie de son design.

L'assignation des ports se fait par l'entremise d'un fichier de contraintes avec l'extension «.ucf» (pour user constraints file).

```
# PlanAhead Generated physical constraints

NET "filter_in[0]" LOC = A3;
NET "filter_in[1]" LOC = A4;
NET "filter_in[2]" LOC = A5;
NET "filter_in[3]" LOC = A7;
NET "filter_in[4]" LOC = A8;
NET "filter_in[5]" LOC = A9;
NET "filter_in[6]" LOC = A10;
NET "filter_in[7]" LOC = A12;
NET "filter_out1[0]" LOC = A15;
NET "filter_out1[1]" LOC = A18;
NET "filter_out1[2]" LOC = A19;
NET "filter_out1[3]" LOC = A20;
NET "filter_out1[4]" LOC = A22;
NET "filter_out1[5]" LOC = B3;
NET "filter_out1[6]" LOC = B4;
NET "filter_out1[7]" LOC = B6;
NET "filter_out1[8]" LOC = B7;
NET "filter_out2[0]" LOC = B8;
NET "filter_out2[1]" LOC = B9;
NET "filter_out2[2]" LOC = B10;
NET "filter_out2[3]" LOC = B12;
NET "filter_out2[4]" LOC = B15;
NET "filter_out2[5]" LOC = B17;
NET "filter_out2[6]" LOC = B18;
NET "filter_out2[7]" LOC = B19;
NET "filter_out2[8]" LOC = B20;
NET "filter_out3[0]" LOC = B21;
NET "filter_out3[1]" LOC = B23;
```

Figure III.29. Ports d'entrée et de sortie.

On note que chaque port est listé dans le fichier, exactement comme il apparaît dans le code. Le symbole du dièse (#) indique que le reste de la ligne est un commentaire (ne s'applique pas dans un fichier VHDL).

Sauvegardez le fichier dans le même répertoire que les autres fichiers source de votre design, sous le nom de « banc_filtres.ucf » Puis, dans ISE, faisons un clic-droit sur votre projet, cliquez sur **Add Source**. Nous pouvons alors choisir le fichier que nous venons de créer.

Nous devrions avoir la vue suivante dans ISE :

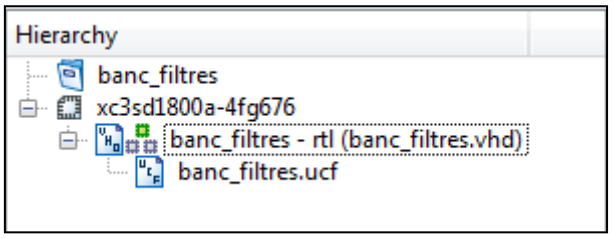


Figure III.30. La vue « Hierarchy ».

Il est maintenant possible de faire la synthèse de notre circuit en vue d'une implémentation sur le FPGA. Pour ce faire, il faudra utiliser la partie inférieure de l'onglet Design disponible dans ISE, que voici :

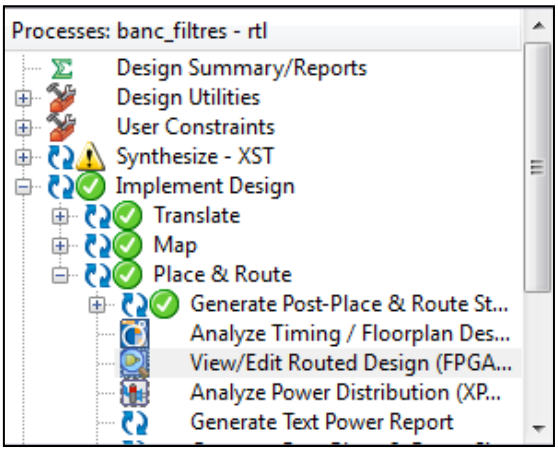


Figure III.31. Processus de l'implémentation.

Pour réaliser les options **Synthesize –XST**, **Implement Design** et **Generate Post & Route Static Timing**, il faut faire un clic-droit sur chacun, puis **Run**.

Le tableau suivant montre les différentes ressources utilisées dans notre implémentation sur FPGA avec leurs pourcentages :

Tableau III.1. Sommaire des ressources utilisées dans le FPGA.

banc_filtres Project Status (06/09/2019 - 23:59:28)			
Project File:	banc_filtres.xise	**Parser Errors:**	No Errors
Module Name:	banc_filtres	**Implementation State:**	Placed and Routed
Target Device:	xc3sd1800a-4fg676	**•Errors:**	No Errors
Product Version:	ISE 14.1	**•Warnings:**	804 Warnings (804 new)
Design Goal:	Balanced	**•Routing Results:**	All Signals Completely Routed
Design Strategy:	Xilinx Default (unlocked)	**•Timing Constraints:**	All Constraints Met
Environment:	System Settings	**•Final Timing Score:**	0 (Timing Report)

Device Utilization Summary [-]				
Logic Utilization	**Used**	**Available**	**Utilization**	**Note(s)**
Number of Slice Flip Flops	227	33,280	1%	
Number of 4 input LUTs	5,199	33,280	15%	
Number of occupied Slices	3,988	16,640	23%	
Number of Slices containing only related logic	3,988	3,988	100%	
Number of Slices containing unrelated logic	0	3,988	0%	
Total Number of 4 input LUTs	7,365	33,280	22%	
Number used as logic	5,199			
Number used as a route-thru	2,166			
Number of bonded IOBs	209	519	40%	
Number of BUFGMUXs	1	24	4%	
Number of DSP48As	84	84	100%	
Average Fanout of Non-Clock Nets	1.59			

Tableau III.2. Sommaire des ressources utilisées dans le FPGA (suite).

Performance Summary				[-]
Final Timing Score:	0 (Setup: 0, Hold: 0)	**Pinout Data:**	Pinout Report	
Routing Results:	All Signals Completely Routed	**Clock Data:**	Clock Report	
Timing Constraints:	All Constraints Met			

Detailed Reports					[-]
Report Name	**Status**	**Generated**	**Errors**	**Warnings**	**Infos**
Synthesis Report	Current	dim. 9. juin 23:33:36 2019	0	804 Warnings (804 new)	108 Infos (108 new)
Translation Report	Current	dim. 9. juin 23:44:04 2019	0	0	0
Map Report	Current	dim. 9. juin 23:44:35 2019	0	0	1 Info (0 new)
Place and Route Report	Current	dim. 9. juin 23:58:55 2019	0	0	2 Infos (2 new)
Power Report					
Post-PAR Static Timing Report	Current	dim. 9. juin 23:59:27 2019	0	0	6 Infos (6 new)
Bitgen Report					

Secondary Reports			[-]
Report Name	**Status**	**Generated**	

La figure suivante représente le schéma des ports d'entrée et de sortie dans le circuit FPGA.

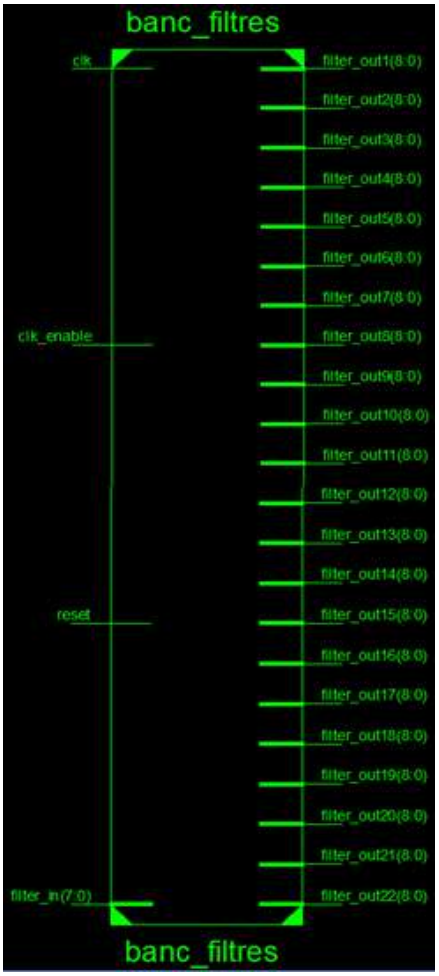


Figure III.32. Ports d'entrée et de sortie dans le circuit FPGA.

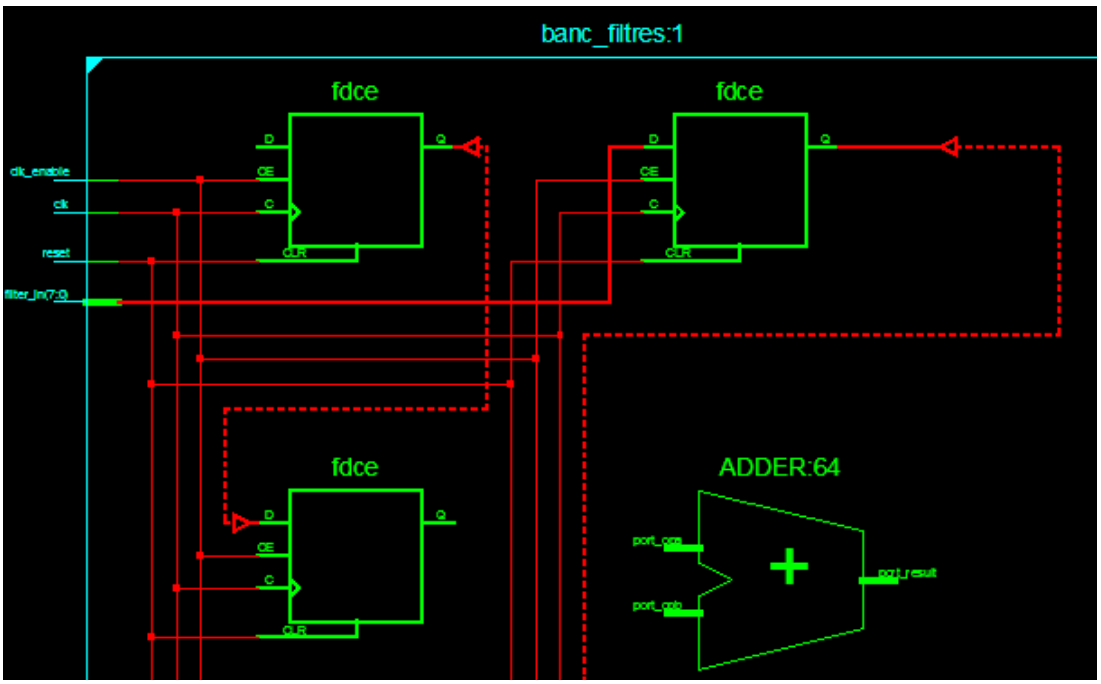


Figure III.33. Les ports d'entrée (une partie zoomée).

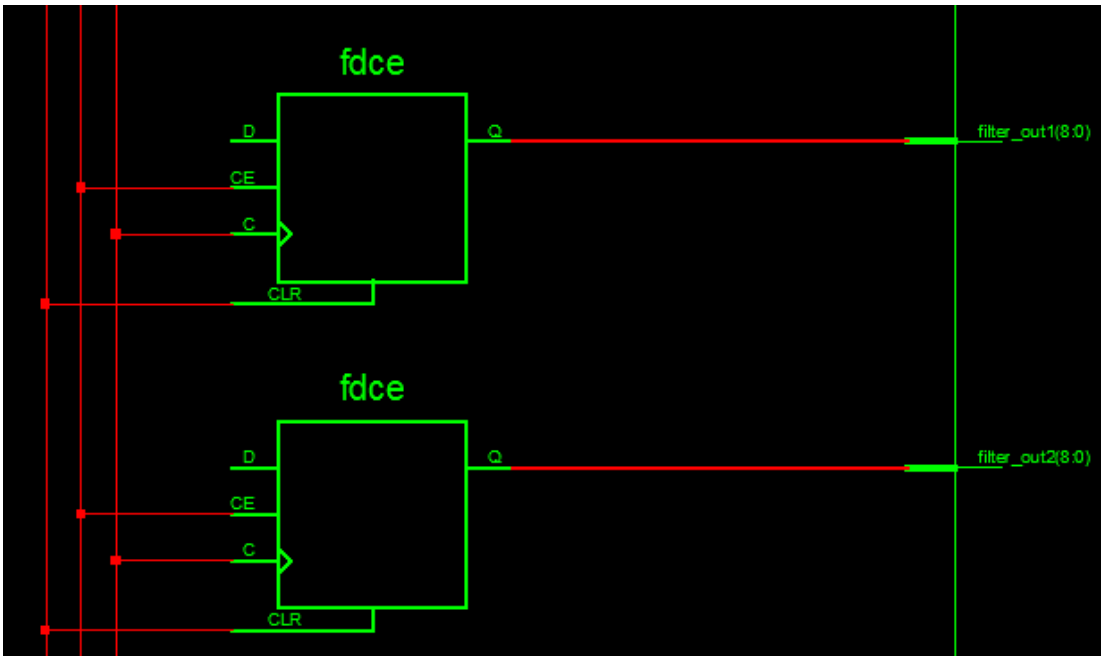


Figure III.34. Les ports de sortie (une partie zoomée).

La figure suivante illustre le circuit FPGA de 1ere filtre.

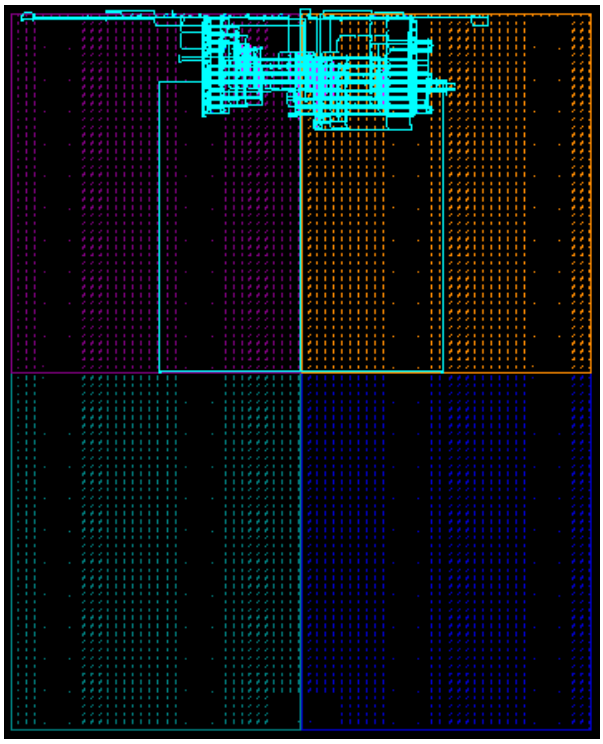


Figure III.35. Circuit FPGA de 1ere filtre.

La figure suivante illustre le circuit FPGA global et son occupation par les interconnexions et les slices et bascules (en couleur bleu)

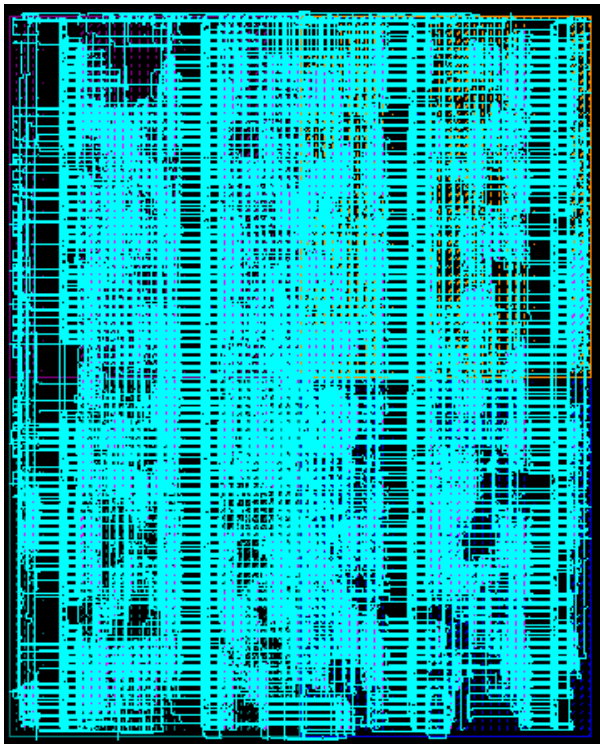


Figure III.36. Circuit de FPGA global.

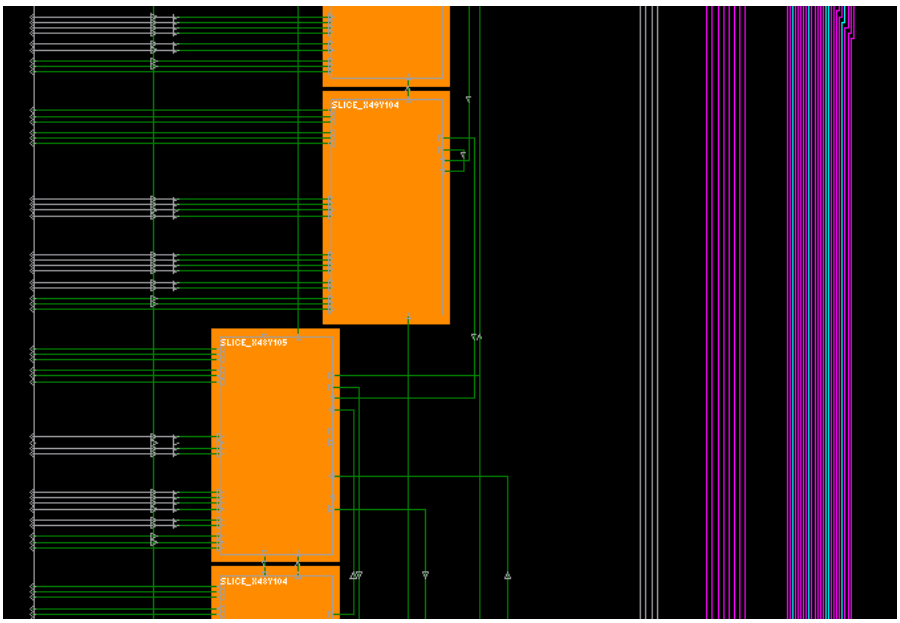


Figure III.37. Les différentes slices sur FPGA (une partie zoomée).

La figure III.41.Représente le circuit interne du slice « SLICE_X48Y105»conçu en utilisant les bascules et les portes logiques.

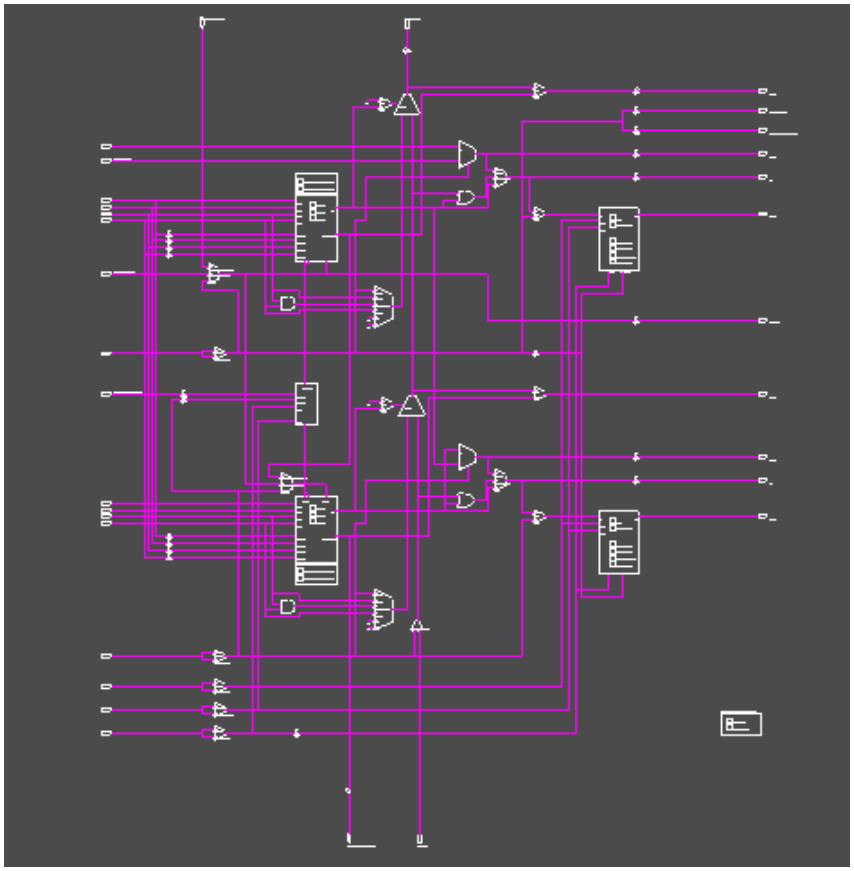


Figure III.38. Le circuit interne du « SLICE_X48Y105».

6.4.Verification du circuit FPGA

Si tout s'est bien passé lors de la synthèse, un fichier banc_filtres.bit est placé dans le projet et est prêt à être chargé sur le kit de développement FPGA.

Au laboratoire, le kit FPGA devrait déjà être allumée et reliée au poste de travail par le câble USB2.

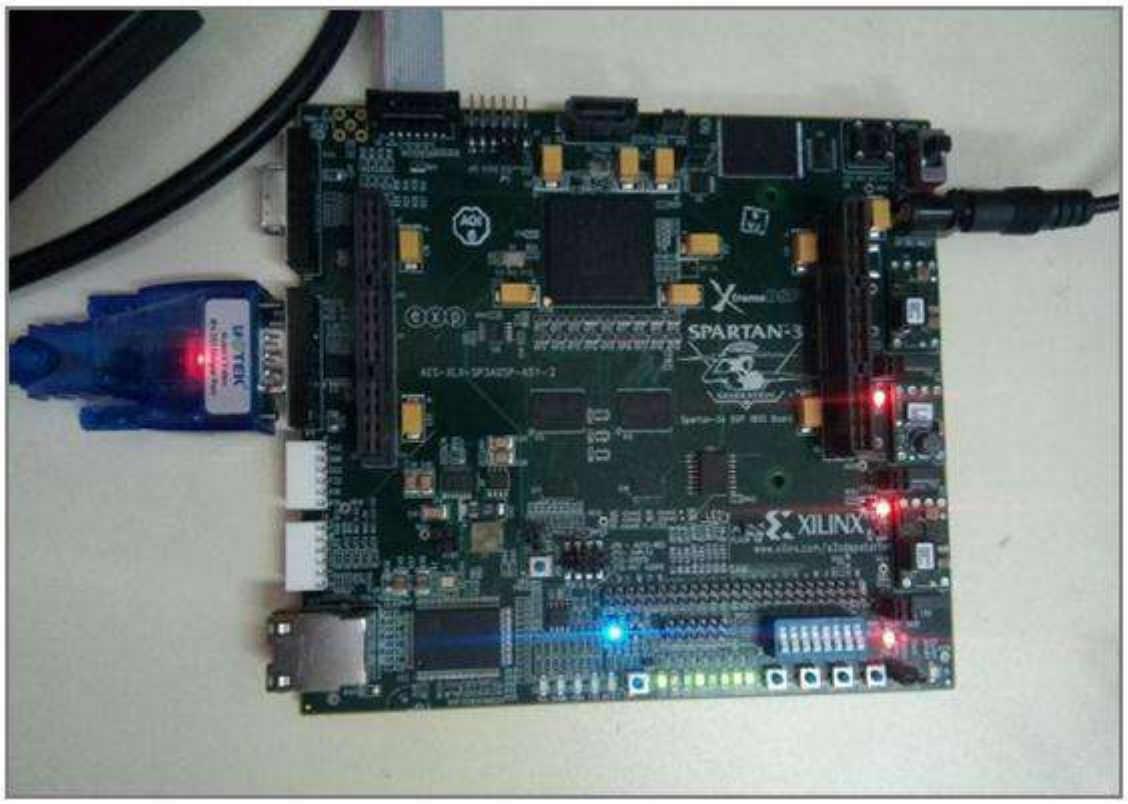


Figure III.39. Carte De Développement XILINX FPGA Spartan-3.

Dans Xilinx ISE, cliquons deux fois sur **Configure Target Device**.

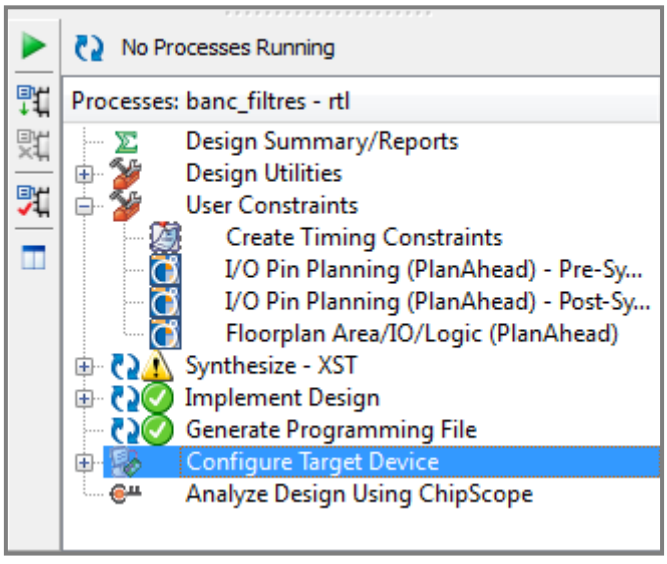


Figure III.40. Processus de l'implémentation.

L'outil iMPACT s'ouvre, C'est l'outil dans lequel on vérifie le circuit FPGA.

Nous assurons de mettre l'option de branchement automatique avant de cliquer sur OK.

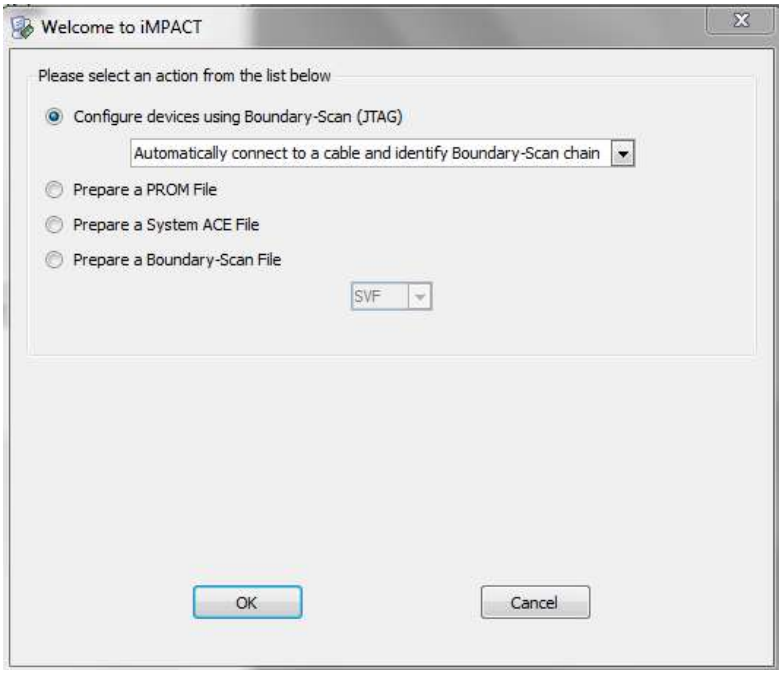


Figure III.41. L'outil iMPACT.

Sélectionner le fichier de programmation créé (banc_filtres.bit pour un design basé sur un schéma).

Nous obtenons le schéma suivant indiquant que la programmation du FPGA s'est bien déroulée.

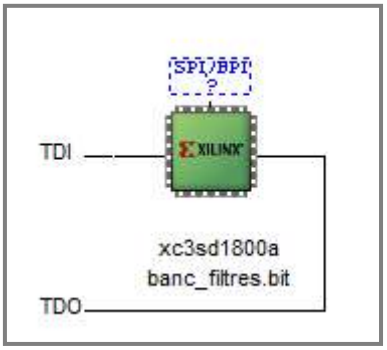


Figure III.42. Schéma de vérification le circuit FPGA.

7. CONCLUSION

Dans ce chapitre, nous avons généré le code VHDL et le fichier de teste "Tb" (test bench), ensuite nous avons simulé ce dernier en utilisant Modelsim afin de faire une comparaison avec les résultats obtenus en utilisant Matlab. La bonne correspondance entre les résultats de comparaison, nous laisse passer à la dernière étape qui est l'implémentation sur un circuit FPGA sous XILINX ISE.

Conclusion générale

Ce mémoire a pour objectif la mise en œuvre, et l'implémentation sur un circuit FPGA d'un banc de filtres numériques, dédié à l'utilisation sur un implant cochléaire.

Dans notre travail, nous avons vu que les implants cochléaires ont très importante chez les personnes sourds, nous avons présenté la stratégie de stimulation cochléaire ACE en précisant son principe de fonctionnement basé sur le banc de filtres.

À travers ce projet, Nous sommes arrivés aux objectifs fixés par le cahier de charge, tels que La connaissance des circuits logiques programmables et leur configuration, la simulation d'un banc de filtres d'une stratégie de stimulation cochléaire, et l'implémentation dans un circuit numérique de type FPGA par la génération du code VHDL du banc de filtres en utilisant le logiciel XILINX ISE.

Les résultats des simulations obtenus dans ModelSim et Matlab sont presque identiques, ce qui confirme la justesse de nos codes VHDL, et que l'objectif de l'implémentation a été réalisé avec succès.

BIBLIOGRAPHIE

[1] STEPHANE Dubus, "L'oreille", Equipe enseignante de l'IFSI du Centre Hospitalier de ROUBAIX, France, rapport technique, octobre 2003.

[2] Entendre+Hearing, "Audition, anatomie et fonctionnement de l'oreille", *Anatomie de l'oreille*, 2015. [En ligne]. Disponible : http://www.entendreplushearing.com /html/audition/comment-on-entend.html#central.

[3] "Des nuisances sonores à la musicothérapie: le son est-il un danger ou un bienfait? " janvier 2011

[4] HELLERRemy" Anatomie et physiologie de l'oreille"Cours Guide dePalanquee, Janvier 2013

[5] PUJOL Remy VENAIL Frederic, "Surdité : généralités", 2016.

[6] MUDRY Albert "L'oreille: ses maladies et ses traitements", 2009

[7] ARNAUD JEANVOINE. "Intérêt des algorithmes de réduction de bruit dans l'implant cochléaire : Application à la binauralité", Université Claude Bernard - Lyon I, 2012.

[8] TAMÁSHarczos "cochlear implant electrode stimulation strategy based on a human auditory model" 12. Januar 2015.

[9] GALLEGO Stéphane, COLLET Lionel " L'implant cochléaire : comment et pourquoi cela marche?", Hôpital Edouard Herriot, pavillon U, service O.R.L

[10]INNSBRUCK, "Comprendre les implants cochléaires",Austria.

[11]http://auditionimplantscontour.e-monsite.com/pages/les-avantages-et-inconvenients-des-appareils.html.

[12] CHARLES T. M. Choi and YI-HSUAN Lee "A Review of Stimulating Strategies for Cochlear Implants" National Chiao Tung University, National Taichung University of Education, Taiwan.

[13]. MARTIN Vondrášek, PAVEL Sovka and TOMÁŠTichý , "ACE Strategy with Virtual Channels", Dept. of Circuit Theory, Czech Technical University in Prague, Technická 2, 166 00 Praha 6, Czech Republic.

[14] KOMALArora, RICHARD Dowell and PAM Dawson, "Cochlear Implant Stimulation Rates and Speech Perception", in *Modern Speech Recognition Approaches with Case Studies*, 2007.

[15] ANGEL de la Torre Vega, MARTA Bastarrica Martí, RAFAEL de la Torre Vega, MANUEL SainzQuevedo, "Description et utilisation du programme", in *Cochlear Implant Simulation version2.0*, Grenade (Espagne), Décembre 2004.

[16] J. PRADO, « *Bancs de filtres* ». Livre électronique. 8 juin 2006.

[17] ONUR BABACAN " Implementation of a Neurophsiology-Based Coding Strategy for the Cochlear Implant" A report submitted to the swiss federal institute of technology zurich, 2010.

[18] IDRICKAKHOUN, COLETTE MCKAY « Vers un réglage automatisé de l'implant cochléaire: quelle utilisation des potentiels composites évoqués électriquement sur le nerf auditif (ECAPs)? ».

[19] JAMES L. Crowley « Le Filtrage Numérique » 31 mars 2000.

[20] FREDERIC LEGRAND « Introduction aux filtres numériques » Licence Creative Commons.

[21] "Filtre passe-bande", *licence Creative Commons attribution*. [Livre électronique]. 501

©(3) Etats-Unis.

[22] "GELE3132", chapitre 3. Filtres et analyse fréquentielle, pp. 11 et 12.

[23] JEAN-PHILIPPE Muller « Le filtrage analogique » version juillet 2002.

[24] FREDERIC Legrand, « Filtres passe-bande analogique et numérique ».

[25]Chapitre 5. « Introduction aux circuits logiques programmables ».

[26] GEN1243, Conception de Systèmes Digitaux. « logique programmable et ses outils de développement ».

[27] K. MESSAOUDI, A. SAADI, L. LOUZE LAMRI, A. KHEZZAR, S. TOUMI, EL-BAY BOURENNANE « Implémentation Matérielle sur FPGA pour la Commande d'un Onduleur Triphasé » Octobre 2014.

[28] BENMAMMAR Lydia, YACEF Soumeya « Implémentation d'un filtre orientable pour l'extraction des attributs des images cérébrales sur la carte ZedBoard » Université Mohamed Bougara Boumerdes 2016.