الجمهورية الجزائرية الديمقراطية الشعبية

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR
ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE DR. YAHIA FARES DE MEDEA

FACULTE DES SCIENCES ET DE LA TECHNOLOGIE

Département du Génie Électrique et de l'Informatique



Mémoire de Fin d'Études

Présenté pour l'obtention du Diplôme
de Master en Génie Electrique

Spécialité : Système de Télécommunication

THEME

Contribution à l'implémentation sur un circuit FPGA d'une cochlée artificielle en utilisant des filtres numériques

Proposé et dirigé par :

M. ESSAID Billel

Préparé et présenté par:

Melle. REDJIMI Abir

الملخص :

الهدف من هذا المشروع هو دراسة و محاكاة و تطبيق القوقعة الاصطناعية على دائرة المنطق القابلة للبرمجة FPGAباستخدام مرشحات رقمية من نوع ممر الموجة، لهذا العمل استعملنا برامج Matlab و Modelsim من أجل محاكاة شيفرات Matlab و VHDL، و برنامج XILINX ISE من أجل مرحلة الزرع.

كلمات مفتاحية : زراعة القوقعة، مجموعة المرشحات، استراتيجية CIS، VHDL، تطبيق على FPGA.

Résumé:

L'objectif de ce projet est d'étudier, simuler et implémenter une cochlée artificielle sur un circuit logique programmable FPGA à l'aide des filtres numériques de type passe-bande, pour ce travail nous avons utilisé les logiciels Matlab et Modelsim pour la simulation des codes Matlab et VHDL, et le logiciel XILINX ISE pour l'étape de l'implémentation.

Mots clés : Implant cochléaire, Banc de filtres, Stratégie CIS, VHDL, Implémentation sur FPGA.

Abstract :

The objectif of this project is to study, simulate and implement an artificial cochlea on a FPGA programmable logic circuit using digital filters type bandpass, for this work we used Matlab and Modelsim software for simulation of Matlab and VHDL code, and the XILINX ISE software to the stage of implementation.

Key words : Cochlear implant, Filter bank, CIS strategy, VHDL, FPGA Implementation.

SOMMAIRE

LISTE DES FIGURES.....**IV**
LISTE DES TABLEAUX.....**VI**
LISTE DES ACRONYMES..... **VII**
INTRODUCTION GENERALE **1**
CHAPITRE I : L'oreille et les implants cochléaires
1. Introduction **3**
2. Anatomie de l'oreille **3**
2.1. Définition..... **3**
2.2. Les composants de l'oreille..... **4**
2.2.1. L'oreille externe (OE).....**4**
2.2.2. L'oreille moyenne (OM) **4**
2.2.3. L'oreille interne (OI).....**5**
2.3. Les fonctions de l'oreille **6**
2.3.1. L'audition.....**6**
2.3.2. L'équilibration.....**7**
3. La surdité **7**
3.1. Définition..... **7**
3.2. Importance de la surdité.....**7**
3.3. Les différents types de surdité.....**8**
3.3.1. Les surdités de transmission.....**8**
3.3.2. Les surdités neurosensorielles (ou surdités de perception).....**8**
3.3.3. Surdités mixtes.....**8**
3.3.4. Surdités génétiques et surdités acquises.....**8**
3.3.5. Surdités profondes à la naissance.....**9**
3.3.6. Surdités de l'enfant.....**9**
3.3.7. Surdités chez l'adulte.....**9**
3.4. Sévérité de la surdité.....**10**
4. L'implant cochléaire..... **11**
4.1. Définition.....**11**
4.2. Historique.....**11**
4.3. Composition de l'implant.....**12**
4.3.1. La partie externe.....**12**

4.3.2. La partie interne....13
4.4. Choix d'implantation et prise en charge....13
4.5. L'objectif des implants cochléaires....14
4.6. L'implant cochléaire et le téléphone portable....15
4.7. Fonctionnement de l'implant cochléaire....15
4.8. Apprentissage....16
4.9. Les avantages et les inconvénients de l'implantation cochléaire....17
4.9.1. Les avantages....17
4.9.2. Les inconvénients....18
4.10. Limites de l'implant cochléaire....18
5. Les stratégies de stimulation cochléaire (les stratégies basées sur le banc de filtres)....18
5.1. Banc de filtres....19
5.2. Les stratégies CIS et n-de-m....19
6. Conclusion....20
CHAPITRE II : La stratégie de stimulation CIS et les circuits FPGA
1. Introduction....21
2. La stratégie CIS « Continuons Interleaved Sampling »....21
2.1. Définition....21
2.2. Le rôle de la stratégie CIS ?....22
2.3. Principe de fonctionnement de la stratégie CIS....23
2.4. Le traitement des signaux sonores par la stratégie CIS....24
2.5. La stratégie CIS+....24
3. Filtre passe-bande....25
3.1. Définition....25
3.2. Caractéristiques....25
3.2.1. Fréquence centrale ωo....25
3.2.2. Facteur de qualité Q....25
3.2.3. Bande passante....25
3.3. Filtres RIF....26
3.4. Filtres RII....27
3.5. Filtre idéal....27

3.6. Filtre passe-bande analogique **28**
4. Les circuits logiques programmables **28**
4.1. Historique de la logique programmable **28**
4.2. Les cartes FPGAs **29**
4.3. Flot de conception **30**
4.4. Design et conception **31**
4.4.1. Les langages de description de matériel **31**
4.4.2. Traduction et optimisation **33**
4.4.3. Partitionnement, placement et routage **34**
4.4.4. Simulation **34**
4.4.5. Configuration (programmation) **34**
5. Conclusion **35**

CHAPITRE III: Conception, simulation et implémentation du banc de filtres de la stratégie CIS dans FPGA

1. Introduction **36**
2. Réalisation du banc de filtres **36**
2.1. Quantification des filtres **38**
3. Génération du code VHDL du banc de filtres **39**
4. Génération de « test bench » du banc de filtres **39**
5. Simulation du programme VHDL en Modelsim **40**
6. Simulation de la stratégie CIS en utilisant le banc de filtres réalisé **42**
7. Présentation du logiciel XILINX ISE **46**
8. Création d'un nouveau projet sous ISE **46**
9. Chargement du code VHDL du banc de filtres sous ISE **48**
10. Les étapes de l'implémentation du banc de filtres sur FPGA sous ISE **49**
11. Conlusion **53**
CONCLUSION GENERALE **54**
Bibliographie **55**

LISTE DES FIGURES

Figure I.1. *L'oreille humaine*.....3
Figure I.2. *Le pavillon*.....4
Figure I.3. *Schéma de l'oreille moyenne*.....5
Figure I.4. *Schéma de l'oreille interne*.....5
Figure I.5. *Les principales pathologies du système nerveux*.....7
Figure I.6. *La surdité génétique profonde*.....9
Figure I.7. *La surdité d'oreille moyenne chez l'enfant*.....9
Figure I.8. *Surdités chez l'adulte*.....10
Figure I.9. *L'implant cochléaire (modèle Nucléus CI24)*.....11
Figure I.10. *Schéma d'implant cochléaire*.....12
Figure I.11. *Partie interne d'un implant cochléaire Freedom 24 RE*.....13
Figure I.12. *Principe de fonctionnement d'un système d'implant cochléaire*.....16
Figure I.13. *Schéma blocs d'un système d'implant cochléaire utilisant la stratégie CIS*.....19
Figure II.1. *Diagramme de la stratégie CIS*.....22
Figure II.2. *Les impulsions entrelacées de la stratégie CIS*.....23
Figure II.3. *Gabarit passe-bande*.....26
Figure II.4. *La structure d'un PLA avant sa programmation*.....29
Figure II.5. *Physionomie d'un FPGA*.....29
Figure II.6. *Exemple de LUT*.....30
Figure II.7. *Flot de conception*.....31
Figure III.1. *Les caractéristiques d'un filtre passe-bande*.....37
Figure III.2. *La réponse en amplitude du banc de filtres*.....37
Figure III.3. *Réponse en amplitude après quantification du 1er filtre*.....39
Figure III.4. *Le résultat de la simulation du 1er filtre obtenu par Matlab*.....41
Figure III.5. *Le résultat de la simulation du 1er filtre obtenu par Modelsim*.....41
Figure III.6. *Les sorties des filtres 1, 5, 8 et 16*.....43
Figure III.7. *Extraction d'enveloppe des signaux de sortie des filtres 1, 5, 8 et 16*.....43
Figure III.8. *Filtrage des enveloppes extraites par un filtre passe bas*.....44
Figure III.9. *Le signal à modulé*.....44
Figure III.10. *Les signaux modulés de la stratégie CIS des filtres1, 5, 8 et 16*.....45

Figure III.11. *Une partie zoomé des signaux modulés de la stratégie CIS des filtres 1, 5, 8 et 16*.....45
Figure III.12. *Choix d'implémentation*.....47
Figure III.13. *Création d'un nouveau projet*.....47
Figure III.14. *Les bibliothèques et les entités du banc de filtres*.....48
Figure III.15. *Les architectures des filtres*.....49
Figure III.16. *La vue « Hierarchy »* 49
Figure III.17. *Processus de l'implémentation*.....51
Figure III.18. *Circuit de FPGA global*.....51
Figure III.19. *Les différentes slices sur FPGA (une partie zoomée)* 52
Figure III.20. *Le circuit interne du « SLICE_X49Y90 »* 52
Figure III.21. *Une partie zoomée du « SLICE_X49Y90 »* 53

LISTE DES TABLEAUX

Tableau I.1. *Sévérité de la surdité (par rapport au niveau sonore)*.....10
Tableau I.2. *Principaux implants commercialisés, avec leur nombre d'électrodes et la stratégie de codage disponible*.....14
Tableau I.3. *Les avantages des implants cochléaires*.....17
Tableau III.1. *Les bandes fréquentielles des filtres*.....36
Tableau III.2. *Sommaire des ressources utilisées dans le FPGA*.....50
Tableau III.3. *Sommaire des ressources utilisées dans le FPGA (suite)*.....50

LISTE DES ACRONYMES

C

CAE : Conduit Auditif Externe

CAO : Conception Assistée par Ordinateur

CIS : Continuous Interleaved Sampling

CLB : Configurable Logic Block

CPLD : Complex Programmable Logic Device

F

FPB : Filtre Passe Bande

FPGA : Field Programmable Gate Array

G

G : surdité Génétique

H

HDL : Hardware Description Language

I

IC : Implant Cochléaire

IEEE : Institute of Electrical and Electronics Engineers

IOB : In/Out Block

ISE : Integrated Software Environment

L

LC : Logic Cell

LUT : Look Up Table

M

M : maladie de Ménière

O

OE : Oreille Externe

OI : Oreille Interne

OM : Oreille Moyenne

ORL : Oto-Rhino-Laryngologie

P

PLA (PAL) : Programmable Logic Array

PLD : Programmable Logic Device

R

RIF : Réponse Impulsionnelle Finie

RII : Réponse Impulsionnelle Infinie

RTL : Register Transfer Level

T

T : Traumatisme acoustique

Tb : Test bench

V

VHDL : VHSIC Hardware Description Language

VHSIC : Very High Speed Integrated Circuit

INTRODUCTION GENERALE

L'audition est une bénédiction qui ne peut pas être dispensée carrément. Les maladies auditifs sont devenues de nos jours, une grande souffrance qui a beaucoup de difficultés dans la vie des infectés par cette maladie.

L'évolution des besoins de l'homme pour apprivoiser son environnement, et d'améliorer son confort et sa qualité de vie, mène ce dernier à chercher des solutions en fonction des donnés scientifiques et technologiques (L'implantation sur une cochlée réelle).

L'implant cochléaire est une prothèse auditive dont une partie est implantée chirurgicalement. Le premier implant auditif est arrivé sur le marché dans les années quatre-vingts ; il s'agit d'un système électronique stimulant directement les terminaisons du nerf auditif dans la cochlée (l'oreille interne).

La cochlée réelle est modélisée comme un banc de filtres passe bandes, notre objectif est d'implémenter sur circuit FPGA le banc de filtres approximant la cochlée pathologique.

Depuis leur invention par Xilinx en 1984, les FPGA sont partis d'un simple rôle d'interfaçage d'appoint pour arriver à véritablement remplacer les ASIC (circuits intégrés à application spécifique) et les processeurs personnalisés dans des applications de contrôle et de traitement de signaux.

Ce travail est divisé sur trois chapitres :

Dans le premier nous avons défini l'oreille, ses fonctions (l'audition et l'équilibration) et la surdité. Nous avons défini aussi les implants cochléaires, en présentant leur principe de fonctionnement et les avantages et les inconvénients de ce dispositif. Et à la fin du chapitre nous avons cité les stratégies de stimulation basées sur le banc de filtres (notamment la stratégie CIS).

Dans le deuxième chapitre nous avons détaillé la stratégie de stimulation CIS : son but et son principe de fonctionnement, la méthode de traitement des signaux sonores utilisée. Ainsi qu'une description des filtres passe-bandes a été donnée. A la fin nous avons présenté les circuits FPGA en définissant leur flot de conception et leur programmation.

Le troisième chapitre étant consacré à la programmation et la simulation, où nous avons vu comment réaliser un banc de filtres composé de 16 filtres de type passe-bande avec des

bandes fréquentielles différentes, est de générer le code VHDL et le banc d'essai (test bench) du banc de filtres afin de le simuler avec Modelsim pour vérifier si nous pouvons l'implémenter sur FPGA. 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.

A la fin, nous avons terminé par une conclusion générale.

CHAPITRE I :

L’oreille et les implants cochléaires

1. Introduction

Dans ce premier chapitre, nous allons étudier l'organe d'audition qui s'appelle l'oreille, elle joue également un rôle important dans l'équilibre. Ensuite, nous allons passer à la surdité qui est résultante de deux effets : audition moins forte, et perte de la clarté des sons.

Le système d'implant cochléaire est une solution palliative à la surdité mais n'est surtout pas à confondre avec une prothèse auditive traditionnelle, simple amplificateur des sons naturels. Il vise à fournir un certain niveau d'audition pour certaines personnes atteintes d'une surdité profonde. Des électrodes posées chirurgicalement permettent de stimuler directement les terminaisons nerveuses de l'audition situées dans la cochlée. Le système d'implant cochléaire repose sur le principe de stimulation électrique, et il est conseillé aux personnes souffrant d'une surdité de perception sévère à totale.

A la fin de ce chapitre, nous allons présenter les différentes stratégies de stimulation cochléaires basées sur les bancs de filtres.

2. Anatomie de l'oreille

2.1. Définition

L'oreille est un organe pair et symétrique. Elle occupe des cavités creusées dans le rocher (Os du crâne). Elle peut assurer deux fonctions : l'audition et l'équilibration. Elle possède des organes spécifiques pour assurer chacune de ces fonctions. [1]

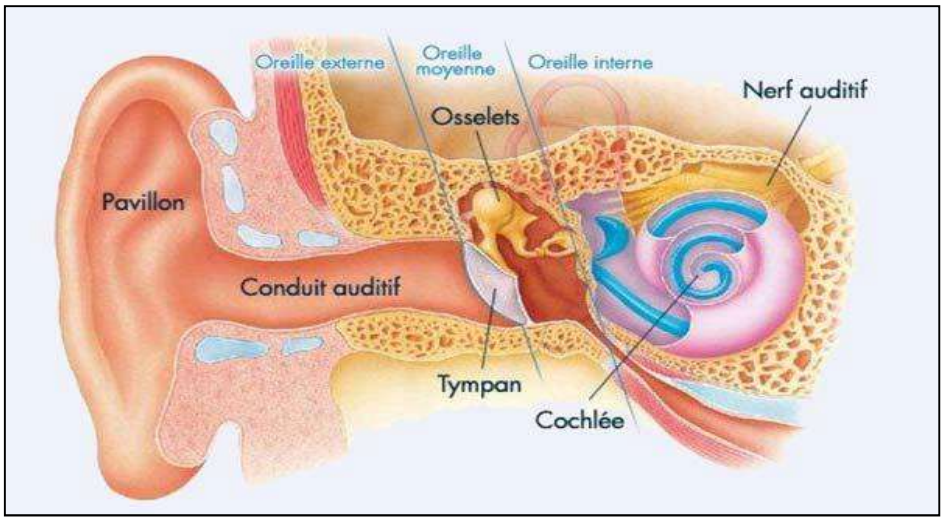


Figure I.1. *L'oreille humaine* [1]

2.2. Les composants de l'oreille

La figure I.1 illustre les composants de l'oreille humaine. Elle se compose de trois parties :

2.2.1. L'oreille externe (OE)

2.2.1.1. Le pavillon

Il est placé latéralement sur le crâne, en arrière de la branche montante du maxillaire inférieur, formé de reliefs et de creux d'origine cartilagineuse. [1]

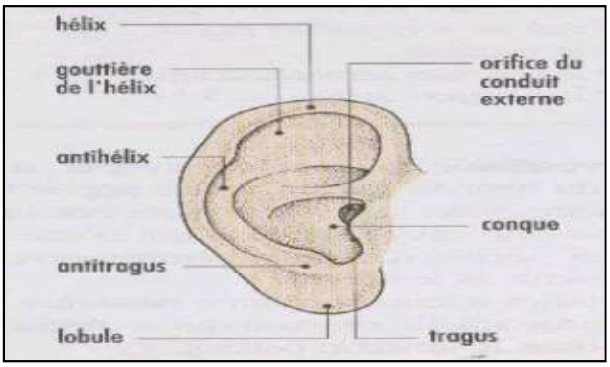


Figure I.2. *Le pavillon* [1]

Dans la figure I.2, nous avons :

- En périphérie, un bourrelet, l'hélix, s'élargissant en bas pour former le lobule de l'oreille.
- Un relief concentrique à l'hélix, l'anthélix.
- La conque, où s'ouvre le conduit auditif externe.
- La conque est bordée du tragus en avant et de l'antitragus en arrière. [1]

2.2.1.2. Le Conduit auditif externe (CAE)

Le CAE est un canal long de 3cm, il s'ouvre au niveau de la conque en dehors. Ses parois contiennent de nombreux follicules pileux, des glandes sudoripares et sébacées (secrètent une matière cireuse : le cérumen). [1]

2.2.2. L'oreille moyenne (OM)

L'OM située dans l'épaisseur du rocher, elle comprend 3 parties communiquant les unes avec les autres :

- La caisse du tympan.

- Les cavités mastoïdiennes.
- La trompe d'eustache. [1]

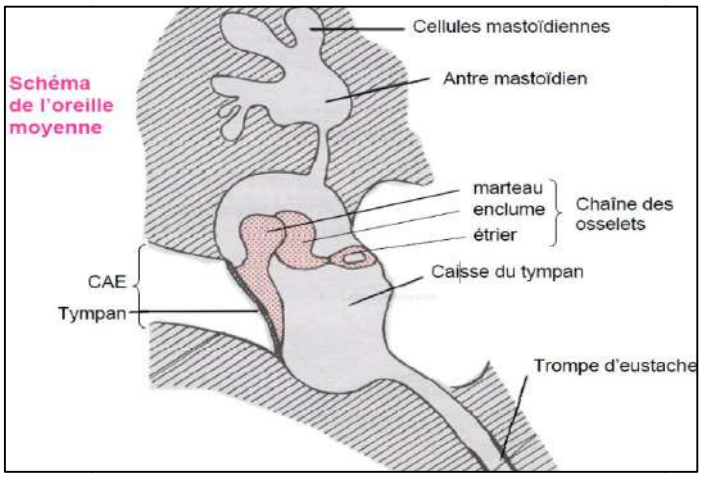


Figure I.3. *Schéma de l'oreille moyenne* [1]

2.2.3. L'oreille interne (OI)

L'OI est située en dedans de la caisse du tympan, elle est encore appelée le labyrinthe :

- Labyrinthe osseux creusé dans le rocher.
- Labyrinthe membraneux contenu dans le labyrinthe osseux.

Ces 2 éléments sont séparés en partie par un espace, l'espace périlymphatique, rempli d'un liquide appelé lapérilymphe. [1]

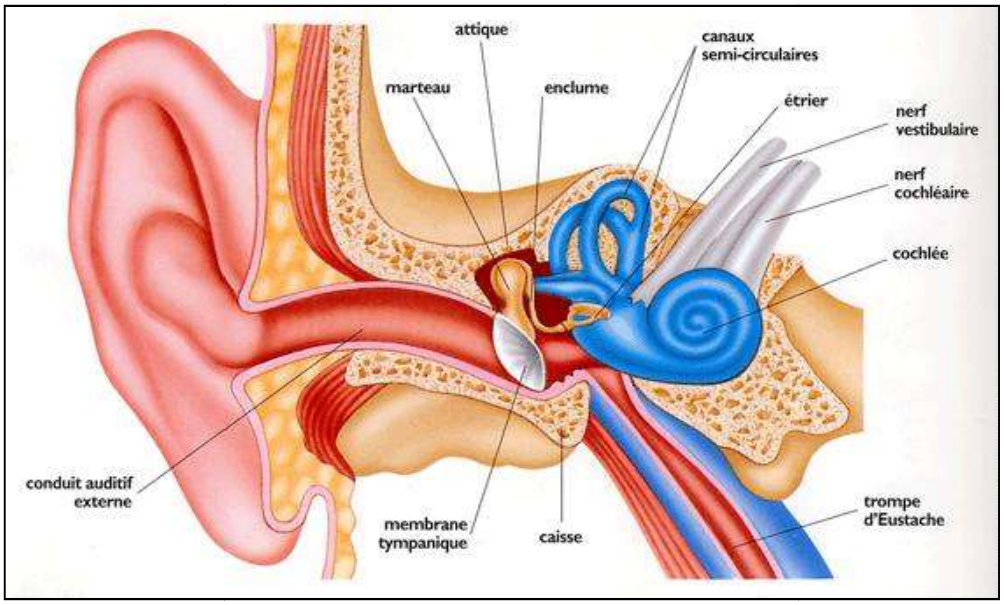


Figure I.4. *Schéma de l'oreille interne* [1]

La cochlée pathologique, est un organe creux en forme d'escargot, rempli d'un liquide appelé endolymphe. Situé au niveau de l'oreille interne, la cochlée constitue la dernière étape de l'intégration du son avant le nerf auditif. [2]

2.3. Les fonctions de l'oreille

2.3.1. L'audition

Le son est le résultat d'un mouvement des molécules d'air qui nous entourent. Il est caractérisé par les éléments suivants :

2.3.1.1. La fréquence

Elle correspond au nombre d'ondes sonores qui passent en un point par unité de temps. Plus un son est aigu, plus la fréquence est élevée. L'oreille humaine perçoit des sons dont la fréquence va de 20 à 20 000 Hz (Hertz = nombre de cycle/s). L'acuité auditive la meilleure se place entre 1000 et 4000 Hz.

2.3.1.2. L'amplitude

Plus un son est puissant, plus son amplitude est grande. L'intensité d'un son se mesure en décibel (dB). L'amplitude d'une conversation normale se situe aux alentours de 50 dB.

2.3.1.3. Le timbre

Il permet de différencier plusieurs instruments qui émettent la même note. L'oreille humaine est capable de discerner environ 400 000 sons différents. [1]

- **Mécanisme de l'audition**

Schématiquement, les ondes sonores sont transportées par les OE et OM vers l'organe de Corti (OI), où elles sont transformées en influx nerveux, qui emprunte ensuite les voies auditives jusqu'au cerveau, qui analyse les ondes sonores. [1]

2.3.2. L'équilibration

Elle est assurée par l'oreille interne, et plus précisément par les canaux semi-circulaires, les utricules et les saccules. Ils contiennent tous une partie sensorielle :

- Macules pour les utricules et les saccules.
- Crêtes ampullaires pour les canaux semi-circulaires. [1]

- **Système auditif central**

Le système auditif central est un ensemble de cellules nerveuses et de zones du cerveau. Il est responsable de la localisation des sons, de la compréhension de la parole dans le bruit et de l'analyse de sons complexes, comme la musique. [3]

3. La surdité

3.1. Définition

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 et les surdités acquises. [4]

3.2. Importance de la surdité

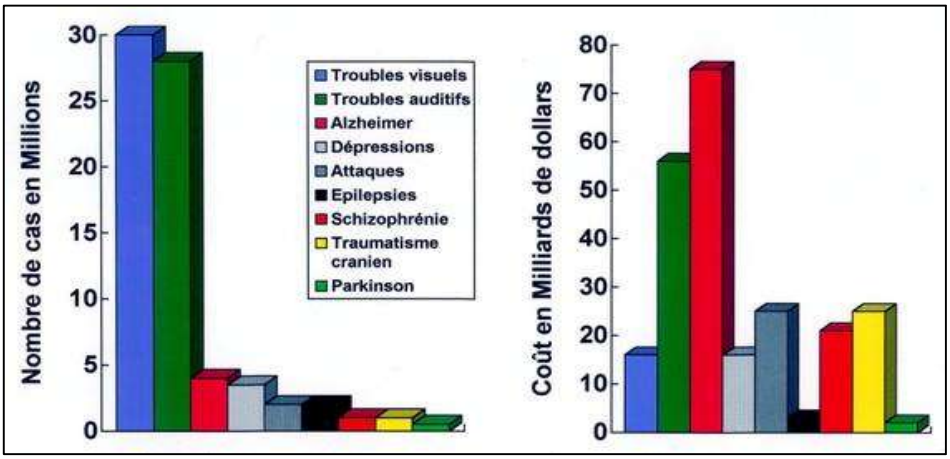


Figure I.5. *Les principales pathologies du système nerveux* [4]

Ces graphes établis il y a déjà plus de 15 ans aux USA pour les principales pathologies du système nerveux et des organes sensoriels, montrent l'incidence des pathologies auditives (vert), qui affectaient à l'époque près de 10% de la population et représentaient un coût très important pour la santé. Ces valeurs ne peuvent qu'être encore accrues à l'heure actuelle. Elles sont similaires dans l'ensemble des pays occidentaux. [4]

3.3. Les différents types de surdité

3.3.1. Les surdités de transmission

Elles sont les surdités liées à une atteinte de l'oreille externe (en général conduit bouché), ou à une atteinte de l'oreille moyenne (otites, lésion des osselets, ...). Le déficit, généralement modéré affecte surtout la perception des sons graves et peu intenses : on n'entend plus (ou difficilement) la voix basse ou chuchotée. La plupart des surdités de transmission peuvent bénéficier d'un traitement chirurgical. Il convient donc de les dépister pour pouvoir les traiter efficacement. [4]

3.3.2. Les surdités neurosensorielles (ou surdités de perception)

Elles sont dues au dysfonctionnement de l'oreille interne (cochlée) et traduisent généralement des lésions des cellules ciliées ou du nerf auditif. Quelques rares surdités ont aussi pour origine les centres auditifs cérébraux. [4]

3.3.3. Surdités mixtes

Une surdité de transmission et une surdité neurosensorielle peuvent se combiner sur la même oreille, constituant ainsi une surdité mixte : par exemple une pathologie de l'oreille moyenne comme l'otospongiose à un stade avancé peut avoir des répercussions sur l'oreille interne et induire une surdité mixte. [4]

3.3.4. Surdités génétiques et surdités acquises

Une surdité peut être d'origine génétique ou acquise au cours de la vie à cause d'une pathologie (traumatisme acoustique, infection, intoxication, vieillissement). Elle peut aussi

être acquise en raison d'une prédisposition génétique : nous ne sommes pas tous égaux devant une infection, un traumatisme. [4]

3.3.5. Surdités profondes à la naissance

Sur 1000 naissances il y a 1 à 1,5 cas de surdité sévère ou profonde. Une cause génétique (g) est responsable des 3/4 de ces surdités. L'autre 1/4 correspond à des surdités acquises au cours de la grossesse ou dans la période périnatale. [4]



Figure I.6. *La surdité génétique profonde* [4]

3.3.6. Surdités de l'enfant

La grande majorité des surdités provient de problèmes d'oreille moyenne (om) comme l'otite chronique. La proportion de surdités génétiques (g) est tombée à environ 10%. Le reste (bleu) est lié à des facteurs variés. [4]

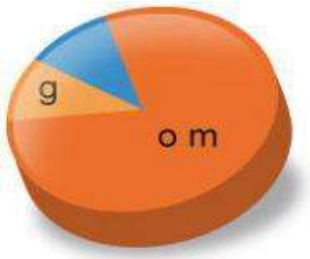


Figure I.7. *La surdité d'oreille moyenne chez l'enfant* [4]

3.3.7. Surdités chez l'adulte

Les facteurs de surdités acquises se multiplient et elles représentent maintenant la très grande majorité des surdités. Les otites chroniques de l'oreille moyenne (om) ne représentent

plus que 20% environ de ces facteurs, un peu moins que la maladie de Ménière (M), qui affecte aussi le système vestibulaire (vertiges). Le traumatisme acoustique (t) dû aux sur-stimulations sonores, occupe à présent le premier rang des facteurs de surdité. D'autres facteurs (bleu) viennent compléter ce tableau : principalement les médicaments toxiques pour l'oreille et les surdités brusques. Les surdités génétiques (g) ne représentent plus qu'un très faible pourcentage, mais il faut cependant noter que dans de nombreux cas (traumatisme sonore, ototoxicité, Ménière, ...) il existe probablement une composante génétique accentuant ou accélérant la surdité acquise. [4]

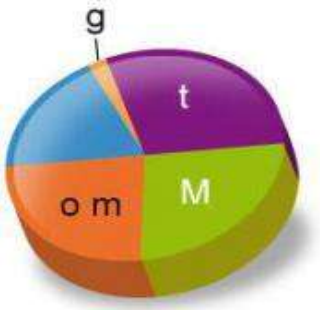


Figure I.8. *Surdités chez l'adulte* [4]

3.4. Sévérité de la surdité

La sévérité de la surdité est appréciée par la combinaison des scores de compréhension et par des critères audiométriques. [5]

Tableau I.1. *Sévérité de la surdité (par rapport au niveau sonore)* [5]

Déficience	Perte tonale (dB)	Perception
Audition normale	< 20	Normale
Déficience auditive légère	21-40	Parole – voix normale
Déficience auditive moyenne	41-70	Parole – voix élevée
Déficience auditive sévère	71-90	Bruits forts
Déficience auditive profonde	91-119	Bruits très puissants
Déficience auditive totale	120	Aucun son

4. L'implant cochléaire

4.1. Définition

L'implant cochléaire est un dispositif qui transforme les informations auditives acoustiques en signal électrique délivré directement au nerf auditif. Il s'adresse essentiellement à des surdités sévères à profondes, bilatérales notamment chez les adultes devenus sourds et chez les enfants sourds congénitaux, leur offrant la possibilité d'une communication orale. [6]

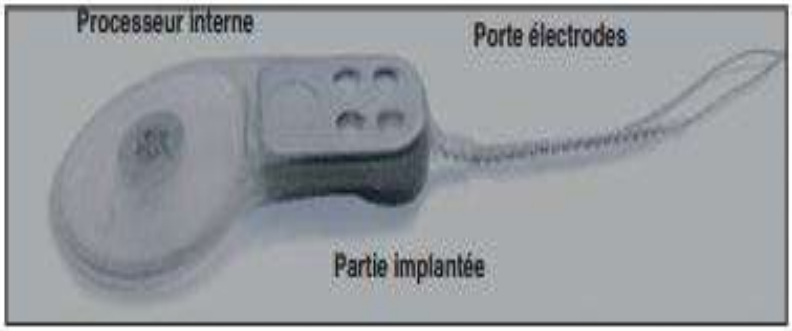


Figure I.9. *L'implant cochléaire (modèle Nucléus CI24)* [6]

4.2. Historique

- En 1961, G von Bekezy (1899-1972) reçoit le Prix Nobel, pour avoir démontré que les sons entrainent une vibration de la membrane basilaire, qui se propage de la base à l'apex de la cochlée sous forme d'ondes longitudinales ; celles-ci présentent des pics en divers endroits répartis selon la fréquence, comme les touches d'un piano : les graves vers l'apex, les aigües vers les fenêtres. On parle à ce propos de tonotopie cochléaire et le terme de « clavier cochléaire » est souvent employé.
- En 1968, H. Davis codifie la physiologie des récepteurs auditifs.
- En 1972, N. Y. Kiang décrit la physiologie de la stimulation électrique de l'oreille interne.
- En 1975, Evans MG montre que les fibres sensorielles afférentes du nerf cochléaire ont chacune une spécificité fréquentielle, qui la rend électivement sensible à une très étroite bande de fréquence.

La stimulation électrique des voies afférentes de l'audition n'existe que depuis la dernière moitié du XX^{e} siècle. La description chronologique des faits conduit à distinguer trois périodes :

- recherches fondamentales et premiers essais chez l'homme (1957-1975).
- évolution de la technologie et des résultats de chacun des implants multi-électrodes proposés par 4 nations (France, Autriche, Australie, États-Unis) jusqu'au consensus né de cette concurrence (1976-1997).
- l'avènement de l'époque contemporaine. [7]

4.3. Composition de l'implant

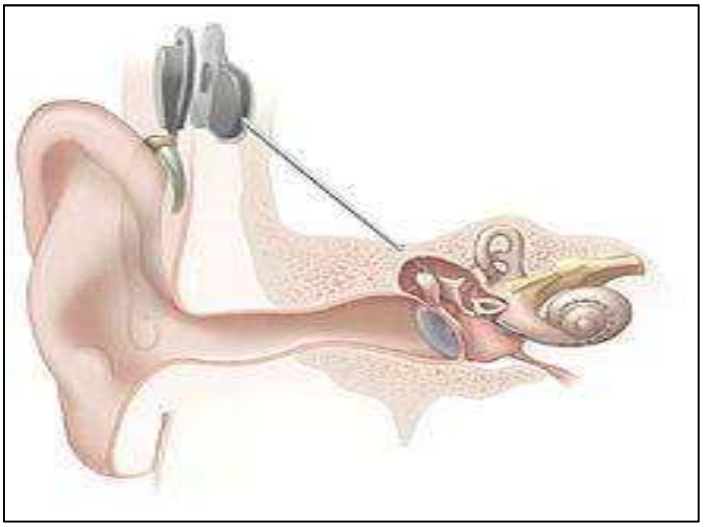


Figure I.10. *Schéma d'implant cochléaire* [7]

L'implant est placé chirurgicalement sous la peau, derrière l'oreille. Il existe plusieurs fabricants et différents modèles, mais ils se composent tous d'une partie externe et d'une partie interne.

4.3.1. La partie externe

Elle comporte :

- Un ou plusieurs microphones qui captent l'environnement sonore et transforment le son en signal électrique.

- Un processeur qui filtre les informations sonores reçues, afin notamment de traiter en priorité la voix humaine, et les répartir sur différents canaux. Les impulsions électriques sont dirigées ensuite vers le transmetteur à travers un fin fil conducteur.
- Un transmetteur par induction électromagnétique, maintenu par un aimant placé derrière l'oreille externe. Le transmetteur fournit à la partie interne de l'appareil l'énergie nécessaire au fonctionnement de l'appareil ainsi que les signaux électriques traités par le processeur.



Figure I.11. *Partie interne d'un implant cochléaire Freedom 24 RE* [7]

4.3.2. La partie interne

Elle se compose de :

- Un récepteur et stimulateur, placé sous la peau, il est relié à l'os afin d'être maintenu en place. Il convertit les signaux sous formes d'impulsions électriques et les envoie à travers un câble jusqu'aux électrodes.
- Un groupe d'électrodes pouvant allé jusqu'à 22 électrodes pour les modèles les plus récents, traversent la cochlée et transmettent les impulsions électriques aux nerfs de la rampe tympanique qui les relaient jusqu'au cerveau. [7]

4.4. Choix d'implantation et prise en charge

La décision de l'implantation, l'intervention et le suivi du patient sont assurés par une équipe pluridisciplinaire le plus souvent composée d'ORL, d'orthophonistes, d'audiologistes et de psychologues. Les caractéristiques de la surdité, l'âge, l'envie de communiquer et avec quels moyens, d'éventuels troubles associés (autisme, hyperactivité), … sont autant de critères étudiés avant la décision d'une implantation. [6]

Tableau I.2. *Principaux implants commercialisés, avec leur nombre d'électrodes et la stratégie de codage disponible* [6]

Implants	Nombre d'électrodes	Stratégie de codage
Nucleus	24	Speak
Clarion	16	CIS / CA
Med-El	24	CIS
MXM	15	Proche du CIS

4.5. L'objectif des implants cochléaires

- Ils peuvent aider à fournir une audition partielle chez les patients dont la surdité est due à des dommages ou à l'absence totale de cils sensoriels dans la cochlée.
- L'implant cochléaire ne fonctionne pas dans les cas de surdités dues à l'absence de nerf vestibulo-cochléaire, mais il peut permettre parfois la transmission parcellaire de quelques informations auditives dans les cas où ce nerf est sévèrement endommagé.
- Les implants les plus récents peuvent permettre une audition améliorée dans les environnements bruyants, d'écouter de la musique, tout en autorisant par exemple la baignade en eau peu profonde.
- Un suivi et un apprentissage, auprès d'un orthophoniste, sont nécessaires après la pose de l'implant. Le cerveau demande une période d'adaptation pour apprendre à identifier les informations reçues.
- Différentes sensations sonores temporaires peuvent être perçues juste après l'activation de l'implant : une succession de bips et sifflements sans signification, une impression de voix déformées comme celle d'un rebot ou aiguës comme celle d'un personnage de dessin animé. Le niveau sonore de l'implant est lentement augmenté, pour donner le temps au cerveau de s'habituer, et il est possible dans les meilleurs cas de récupérer plus de 80 % de l'audition.
- La psychologie du patient et son vécu de sa surdité entrent pour beaucoup dans l'impression des résultats obtenus. Les adultes sourds profonds de naissance et depuis une longue durée tendent à rencontrer plus de difficultés, alors que les jeunes enfants présentent une capacité d'adaptation très élevée. Ceci peut motiver une implantation précoce chez les enfants, alors qu'une partie des sourds et malentendants militent pour

que ces enfants puissent s'ouvrir à la langue des signes, à la culture et à la communauté sourde, et que le choix d'une implantation soit reportée à l'âge adulte. [7]

4.6. L'implant cochléaire et le téléphone portable

Certains téléphones portables numériques (par exemple : global system for mobile communications (GSM) dans certains pays) peuvent parasiter la partie externe de l'équipement. Cela peut entrainer, pour le porteur d'implant, une perception déformée des sons à moins de 1-4 m d'un téléphone portable numérique en marche. [8]

4.7. Fonctionnement de l'implant cochléaire

Le système d'implant cochléaire est composé d'un processeur de son placé derrière l'oreille et d'un implant fixé chirurgicalement sous la peau. Il permet l'insertion d'un réseau d'électrodes dans la cochlée. Les sons sont captés par le microphone du processeur qui les numérisés et traités. Les informations codées par le processeur sont envoyées à l'implant via l'antenne. L'implant transforme ces informations en signaux électriques et les envoie aux électrodes. Chaque électrode correspond à une bande de fréquence du signal sonore (son aigu ou grave), et va délivrer des impulsions électriques au nerf auditif comme le feraient les cellules ciliées. Le nerf auditif transmet ces informations jusqu'au cerveau qui interprète ces signaux en sons. A la fin l'on perçoit le son. [7]

L'implant cochléaire fonctionne selon le principe suivant:

- *La partie externe :* un processeur transforme les sons de l'environnement captés par un microphone, en signal électrique. Ensuite, une antenne le transmet à travers la peau à la partie interne implantée.
- *La partie interne :* implantée chirurgicalement, elle réceptionne puis véhicule le signal électrique jusqu'à un faisceau d'électrodes placé dans la cochlée. Le rôle de ce dernier est d'émettre des impulsions électriques qui stimulent les fibres du nerf auditif. L'influx nerveux créé est envoyé vers le cerveau où il est analysé et interprété. [5]

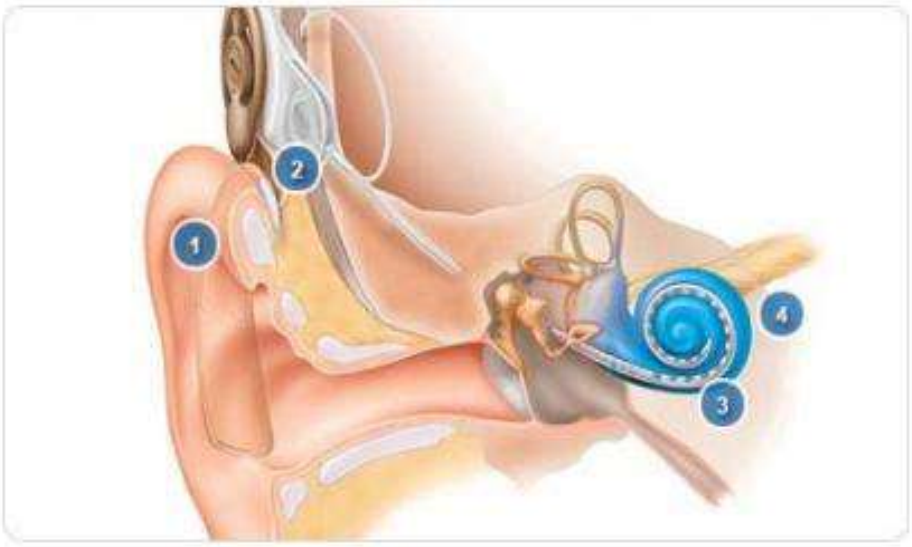


Figure I.12. *Principe de fonctionnement d'un système d'implant cochléaire* [5]

Tel que :

(1) Processeur vocal.

(2) Signaux numériques.

(3) Electrodes.

(4) Nerf auditif.

- Les sons sont captés par un microphone et transformés en signaux numériques.
- Ce signal est traité par le processeur vocal qui le convertit en impulsions électriques selon un codage spécial.
- Ces impulsions sont envoyées à l'émetteur (ou antenne) qui les transmet au récepteur implanté à travers la peau intacte au moyen d'ondes électromagnétiques.
- Le récepteur produit une série d'impulsions électriques pour les électrodes placées dans la cochlée.
- Les contacts de l'électrode stimulent directement les fibres nerveuses dans la cochlée sans utiliser les cellules ciliées endommagées. [5]

4.8. Apprentissage

La compréhension avec l'implant cochléaire requiert du temps et une rééducation primordiale pour intégrer les informations auditives reçues. Cet apprentissage se fait avec

l'aide d'une orthophoniste, d'un audiologiste, mais également au quotidien par des exercices personnels (CD, internet, écoute de l'environnement sonore,...). [6]

4.9. Les avantages et les inconvénients de l'implantation cochléaire

4.9.1. Les avantages

Tableau I.3. *Les avantages des implants cochléaires* [7]

	Sans appareil auditif	**Avec appareil auditif**	**Avec implant cochléaire**
Bruits entendus 	Seuls les bruits forts et la voix criée proche 	Bruits d'intensité moyenne (casseroles, moteur de voiture, etc.) 	Même les sons faibles (sonneries, écoulement d'eau, chants des oiseaux, etc.) 
Compréhension de la parole 	Quasiment aucune 	Incomplète, seulement avec lecture sur les lèvres 	Complète pour un dialogue à 1 m dans une pièce calme, même au téléphone pour certains patients 

L'oreille perçoit et interprète les sons, elle permet à chacun de communiquer avec son entourage, de participer à des conversations. Les personnes atteintes de surdité sévère à profonde sont malheureusement souvent privées de cette faculté. L'implant cochléaire peut être une solution efficace. En effet, lorsque les prothèses auditives classiques n'apportent plus suffisamment de bénéfices pour la compréhension de la parole, l'implant cochléaire permet aux personnes atteintes de surdité sévère à profonde d'améliorer leur qualité de vie en leur

donnant (ou en leur redonnant) une meilleure intelligibilité dans le calme et dans le bruit, et en les aidant à affronter les défis du quotidien, de la vie professionnelle ou scolaire. Les patients implantés déclarent se sentir davantage en sécurité (en pouvant être alertés en cas d'urgence par exemple), ils sont aussi plus indépendants et retrouvent une vie sociale. Pouvoir utiliser le téléphone, prendre rendez-vous chez le médecin, regarder la télévision, entendre une alarme… [7]

4.9.2. Les inconvénients

Les inconvénients sont le surcoût lié à l'implantation de la deuxième oreille et la difficulté voire l'impossibilité de recourir ultérieurement à d'autres techniques.

- Résultats très dépendants de l'audition résiduelle, avec la difficulté particulière chez l'enfant de déterminer le statut auditif de l'oreille non implantée.
- Possibilité que le patient refuse d'utiliser l'audioprothèse si les performances sont limitées.
- Diminution possible des performances de l'IC.
- Retard de bénéfice à l'utilisation de l'implant cochléaire si l'audioprothèse donne de bons résultats (il est recommande de ne pas utiliser de prothèse pendant quelques semaines après l'implantation). [5]

4.10. Limites de l'implant cochléaire

Au-delà des critères d'implantation, l'implant présente des limites perceptives. Globalement, la compréhension dans le silence est acquise par la grande majorité des patients implantés. En revanche, une conversation dans le bruit ou la perception de la musique sont des situations sonores difficiles. Par exemple, l'écoute de la musique laisse un sentiment partagé : certains patients implantés l'apprécieront, reconnaitrons quelques paroles et mélodies ; d'autres au contraire, rencontreront de grandes difficultés. [6]

5. Les stratégies de stimulation cochléaire (les stratégies basées sur le banc de filtres)

Il existe des implants cochléaires basés sur un banc de filtres, ayant pour objectif de remplacer une cochlée pathologique défaillante. Parmi ces implants, ceux qui utilisent la stratégie de stimulation CIS (Continuous Interleaved Sampling).

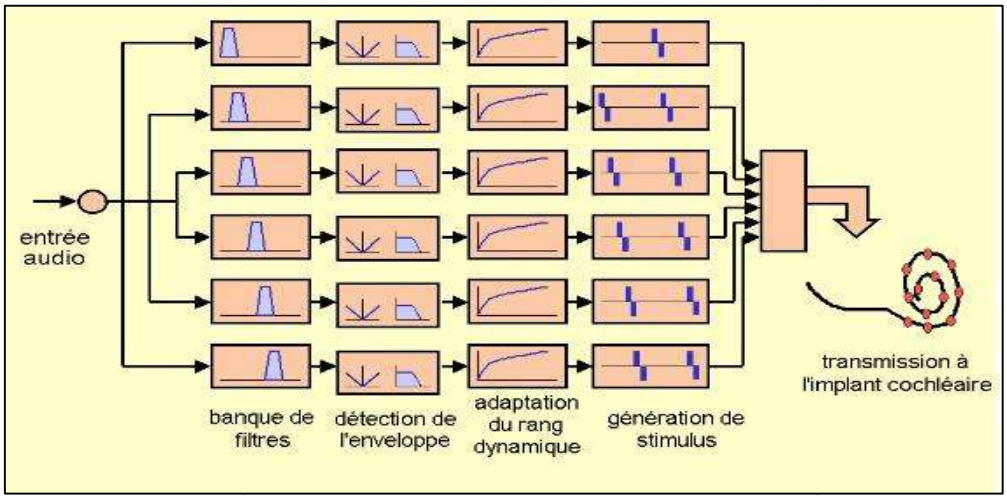


Figure I.13. *Schéma blocs d'un système d'implant cochléaire utilisant la stratégie CIS* [9]

La stratégie CIS (Continuous Interleaved Sampling ou "échantillonnage continu entrelacé") a été initialement décrite par Wilson. Dans la stratégie CIS, l'information est codée en impulsions carrées biphasées d'amplitude variable. Les électrodes sont stimulées de façon indépendante, séquentielle plutôt que simultanée ; deux électrodes ne sont jamais stimulées de façon synchrone pour éviter les interférences.

La stratégie CIS correspond à une représentation spectrale sur plusieurs canaux du signal de parole combinée à une haute résolution temporelle, étant donnée la cadence élevée d'impulsions. Les possibilités d'interactions entre canaux sont réduites en ne stimulant qu'un canal à la fois. [9]

5.1. Banc de filtres

Le signal audio passe par une série de filtres passe bande se chevauchant partiellement (chacun passant une gamme de fréquences différentes) où le signal est analysé en termes de fréquence et d'amplitude. Le banc de filtre divise le signal audio en plusieurs bandes de fréquences simulant le mécanisme de filtrage auditif dans une audition normal. [9]

5.2. Les stratégies CIS et 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. Cet accroissement de la fréquence de stimulation est fait au dépend d'une réduction de la qualité, puisque l'information correspondante aux canaux non sélectionnés est perdue.

De cette façon, l'information correspondante aux canaux non sélectionnés est éliminée du signal synthétisé. [9]

6. Conclusion

La stratégie de stimulation cochléaire CIS qui est basée sur le banc de filtres passe bandes, est la plus utilisée dans les implants cochléaires de nos jours. Dans le chapitre suivant nous donnerons plus de détail sur cette stratégie, afin de pouvoir réaliser le banc de filtres qui est l'objectif de ce travail.

CHAPITRE II :

La stratégie de stimulation CIS et les circuits FPGA

1. Introduction

L'application des implants cochléaires dans la correction de l'audition et troubles de la parole chez les personnes sourdes est connu plus de 25 ans. Bien qu'il y ait de nombreuses stratégies de traitement de la parole, il est nécessaire d'étudier la stratégie d'échantillonnage entrelacé continu (CIS).

La parole sera filtrée par un banc de filtres numériques ayant des bandes passantes réparties sur le spectre audible. L'énergie de ce signal va être extraite des différentes bandes, et sera ensuite codée à la partie interne pour générer des stimuli. Pour réaliser cette approche temporelle, il est nécessaire d'étudier les méthodes de synthèse des filtres numériques, afin de pouvoir réaliser le banc de filtres de la stratégie de stimulation.

Ce chapitre est consacré à l'étude de la stratégie de stimulation cochléaire CIS, ainsi que les caractéristiques des filtres passe bande qui servent à la répartition spectral de l'énergie du signal audio sur les électrodes de stimulation.

La deuxième partie de ce chapitre sera une présentation sur les circuits logiques programmables, notamment les FPGA, et le principe de l'implémentation de ces circuits.

2. La stratégie CIS « Continuons Interleaved Sampling »

2.1. Définition

La stratégie CIS est l'une des techniques de traitement du signal en préservant les informations sur la forme d'onde pour les implants cochléaires. La figure II.1 montre le diagramme de l'ensemble du processus de la stratégie CIS. [10]

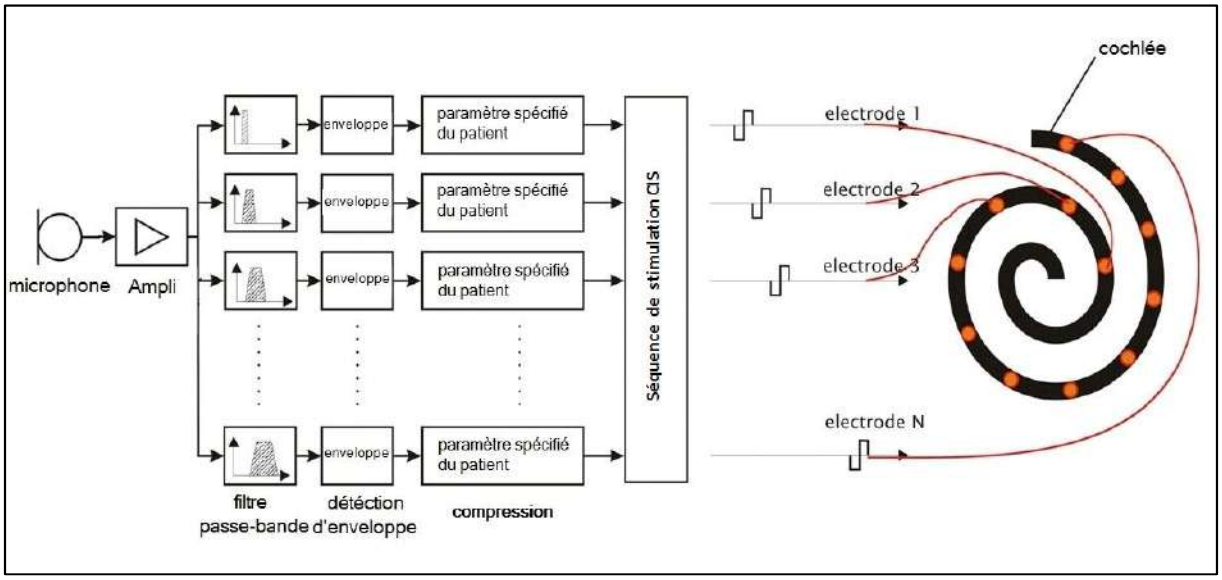


Figure II.1. *Diagramme de la stratégie CIS* [11]

La stratégie CIS de la figure II.1 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. [11]

2.2. Le rôle de la stratégie CIS

La stratégie CIS a été conçue pour combiner les avantages des impulsions non simultanées sur plusieurs électrodes avec la préservation de la structure temporelle fine du signal analogique original en utilisant de brèves impulsions à haute fréquence. Avec la stratégie CIS, le signal est passé à travers un banc de filtre numérique (des filtres de fréquences spécifiques). L'amplitude de chaque impulsion est dérivée à partir des enveloppes des signaux de sortie du filtre, qui sont formés par redressement et filtrage passe-bas. Avec le taux élevé de la stratégie

CIS, les canaux sont stimulés à 1500 impulsions/s. Quels que soient les avantages théoriques et les inconvénients de chaque stratégie de codage, ses intérêt et défauts sont finalement déterminés par la performance empirique des patients. Le noyau du 22éme canal (et récemment 24) de l'implant cochléaire est actuellement le système multicanal le plus couramment utilisé. Le noyau implantable rangé d'électrodes est constitué d'électrodes de la bande de platine-iridium placées dans un support de silastic.

Plusieurs générations de processeurs vocaux ont été utilisées avec le noyau de l'implant cochléaire multicanal. Les noyaux initiaux des processeurs de la parole utilisent un schéma caractéristique d'extraction dans lequel les principales caractéristiques de la parole sélectionnées ont été présentées à travers le réseau d'électrodes implantées. [12]

2.3. Principe de fonctionnement de la stratégie CIS

La stratégie CIS est mise en œuvre par tous les fabricants. Le son enregistré par un microphone pénètre dans les filtres passe-bande (5-20, le nombre de FPB identique au nombre d'électrodes) avec différentes fréquences de coupure. L'enveloppe temporelle de chaque bande est extraite par rectification suivi par un filtre passe-bas. La fréquence de coupure du filtre d'enveloppe passe-bas est d'environ 160-320 Hz.

Après cela, l'enveloppe est logarithmiquement compressée pour faire correspondre les amplitudes acoustiques très variables à la dynamique électrique étroite. Cette amplitude d'enveloppe compressée module une porteuse biphasée à taux fixe (100-1000 Hz/s). Les transporteurs sont entrelacés entre les bandes pour éviter les interférences du champ électrique simultané (pas de stimulation simultanée produite entre les bandes à tout moment). Cette stimulation non simultanée (figure II.2) ne nécessite qu'une seule source de courant avec un multiplexeur.

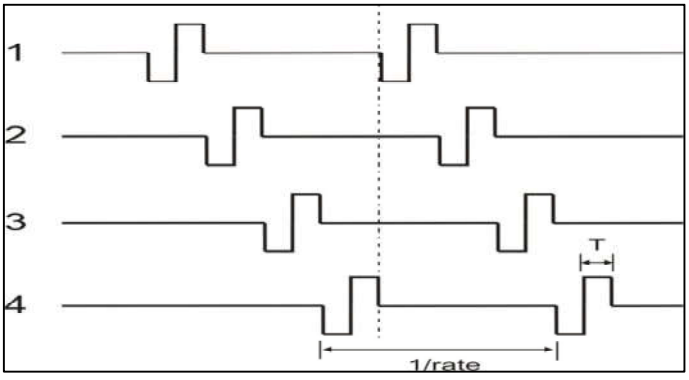


Figure II.2. *Les impulsions entrelacées de la stratégie CIS* [12]

Les impulsions entrelacées utilisées sur quatre électrodes dans le cas de la stratégie d'échantillonnage entrelacé continu (CIS). T est la durée d'impulsion et 1/rate est la durée entre deux impulsions sur chaque canal. [12]

2.4. Le traitement des signaux sonores par la stratégie CIS

Quand une personne parle, sa voie est captée par le microphone qui convertit les ondes sonores en des signaux électriques. Le signal est d'abord amplifié avant d'aller dans une « boite » de filtres passe-bande qui séparent les signaux sonores en différentes fréquences.

Après, des redresseurs à ondes pleines et des filtres passe-bas (typiquement avec une fréquence de coupure de 200 ou 400 Hz) extraient les formes d'ondes filtrées en « enveloppe ». Ensuite, les sorties de l'enveloppe sont compressées où la compression est une fonction non-linéaire. Elle est utilisée pour vérifier que les sorties d'enveloppe correspondent à la plage dynamique de l'audition électrique évoquée et basée sur les préférences du patient. Après la compression, les signaux sont utilisés pour moduler les impulsions biphasées. Le schéma de ces impulsions est représenté dans la figure II.2. Les amplitudes des impulsions sont proportionnelles aux enveloppes.

Elles sont livrées à des électrodes d'une manière non chevauchante. Dans de nombreuses études, la stimulation d'un taux d'impulsion élevé donne de meilleures performances que celle d'un taux d'impulsion faible. [10]

2.5. La stratégie CIS+

Une version modifiée (CIS+) de la stratégie CIS est utilisée de nos jours, plus typiquement dans les implants Med-El. CIS+ utilise une transformation de Hilbert pour représenter l'enveloppe d'amplitude des sorties des filtres. Cette transformation permet de suivre plus étroitement le signal acoustique pour une représentation plus précise de ses dynamiques temporelles par rapport aux autres techniques utilisées pour représenter l'enveloppe d'amplitude dans d'autres systèmes d'implants comme : « rectification d'onde », « filtrage passe-bas » ou « transformée de Fourier rapide ». [11]

3. Filtre passe-bande

3.1. Définition

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 (un signal). 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. [13]

3.2. Caractéristiques

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

3.2.1. Fréquence centrale ωo

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.

3.2.2. 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.

3.2.3. Bande passante

La bande passante *BP* d'un filtre passe-bande est l'intervalle de pulsations [ω_{c1}, ω_{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

et : $BP = [\omega_{c1}, \omega_{c2}]$

Ces pulsations de coupure sont telles que : [14]

$$\left|h(j\omega_{c1})\right| = \left|h(j\omega_{c2})\right| = \frac{hmax}{\sqrt{2}} \quad ; \quad \left(Car\ 20log\frac{1}{\sqrt{2}} = -3\right) \tag{II.1}$$

3.3. Filtres RIF

Le filtre à réponse impulsionnelle finie (RIF) est un filtre numérique qui est caractérisé par une réponse uniquement basée sur un nombre fini de valeurs du signal d'entrée. Ce filtre est l'un des deux types de filtre numérique linéaire. [15]

Un filtre RIF possède une fonction de transfert polynomiale de la forme :

$$H(z) = \sum_{n=0}^{N-1} h_n z^{-n} \tag{II.2}$$

La transformée de Fourier correspondante est : [16]

$$H\left(e^{j\omega}\right) = \sum_{n=0}^{N-1} h_n e^{-jn\omega} = \pm\left|H\left(e^{j\omega}\right)\right| e^{j\varphi(\omega)} \tag{II.3}$$

Pour obtenir un filtre passe-bande ou coupe-bande la procédure est quelque peu modifiée si les bandes de transition ne sont pas identiques (figure II.3).

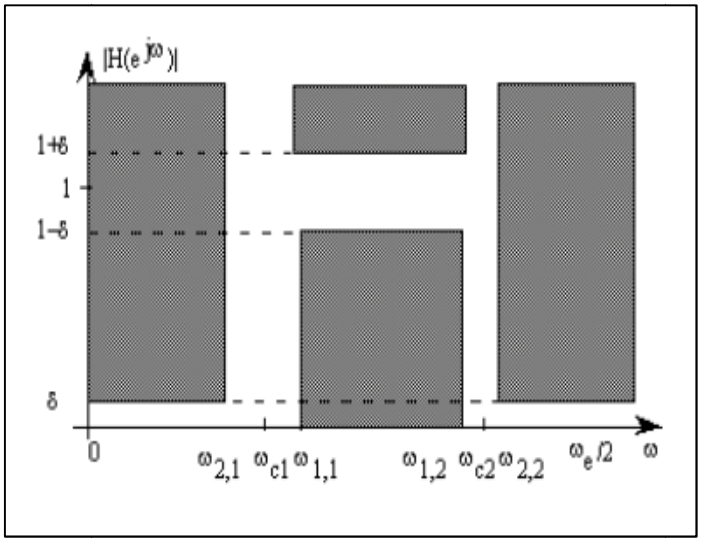


Figure II.3. *Gabarit passe-bande* [17]

On choisit pour bande de transition, la plus étroite :

$$\Delta\omega = min[(\omega_{1.1} - \omega_{2.1}), (\omega_{2.2} - \omega_{1.2})] \tag{II.4}$$

On définit H (ejω) par :

$$H\left(e^{j\omega}\right) = \begin{cases} 0\ pour\ 0 \leq |\omega| < \omega_{c1} \\ 1\ pour\ \omega_{c1} \leq |\omega| < \omega_{c2} \\ 0\ pour\ \omega_{c2} \leq |\omega| < \pi \end{cases} \quad \text{(II.5)}$$

Où ωc1 et ωc2 sont définies par : [17]

$$\omega_{c1} = \omega_{1.1} - \frac{\Delta\omega}{2} \quad \text{(II.6)}$$

$$\omega_{c2} = \omega_{1.2} + \frac{\Delta\omega}{2} \quad \text{(II.7)}$$

3.4. Filtres RII

Un filtre à réponse impulsionnelle infinie (RII) est un type de filtre numérique caractérisé par une réponse basée sur les valeurs du signal d'entrée ainsi que les valeurs antérieures de cette même réponse. Il est nommé ainsi parce que dans la majorité des cas la réponse impulsionnelle de ce type de filtre est de durée théoriquement infinie. Ce filtre est l'un des deux types de filtre numérique linéaire. [18]

La fonction de transfert des filtres RII s'écrit :

$$H(Z) = \frac{\sum_{i=0}^{M} b_i z^i}{1 - \sum_{j=1}^{N} b_j z^j} \quad \text{(II.8)}$$

Comme la réponse impulsionnelle de ces filtres est infinie, donc ils peuvent devenir instables. [19]

3.5. 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 G_{max} −3 dB et ensuite ce gain décroît de −20 dB par décade (soit −6 dB par octave) (filtre de 1er ordre). [13]

3.6. Filtre passe-bande analogique

Un filtre passe-bande peut être implémenté de façon analogique avec des composantes électroniques. Par conséquent, ce genre de filtre s'applique sur des signaux continus en temps réel. Les composantes et la configuration du circuit fixeront les différentes caractéristiques du filtre, telles que l'ordre, les fréquences de coupure et son diagramme de Bode. Les filtres analogiques classiques sont du premier ou du second ordre. Il existe plusieurs familles de filtres analogiques : Butterworth, Tchebychev, Bessel, elliptique, etc. L'implémentation des filtres de même famille se fait généralement en utilisant la même configuration de circuit, et ceux-ci possèdent la même forme de fonction de transfert, mais ce sont les paramètres de celle-ci qui changent, donc la valeur des composantes du circuit électrique. [13]

4. Les circuits logiques programmables

4.1. Historique de la logique programmable

Un circuit logique programmable se définit comme un composant discret contenant des modules de logique combinatoires et séquentiels dont les interconnexions sont établies par programmation. Il peut être configuré et reconfiguré par l'utilisateur pour la réalisation de fonctions logiques. Les premiers circuits programmables, les réseaux logiques programmables (PLA, pour Programmable Logic Array), inventés par Monolithic Memories Inc.(MMI) en 1975 utilisent le fait que toute fonction logique combinatoire peut se mettre sous forme d'une somme (OU logique) de produits (ET logique), c'est ce que l'on appelle classiquement la première forme canonique, ou forme disjonctive ; ou bien sous forme d'un produit des sommes en utilisant la loi de « De Morgan », la structure du PAL est plutôt intuitive. Elle se compose généralement des entrées avec des inverseurs menant dans une série de portes ET, leurs sorties mènent vers une série des portes OU. Le schéma de la figure II.4 représente une structure de PLA simple vierge avant programmation, avec toutes les connections possibles qui sont réalisées typiquement pour ce genre de composant par des fusibles. La programmation du circuit, pour mettre en application une conception envisagée, consiste à supprimer certaines des connexions marquées d'une croix, un programme est employé pour « griller » les fusibles avec un courant fort afin de supprimer les raccordements non désirés. Si une connexion est supprimée, une valeur constante « 1 » est appliquée à l'entrée

correspondante de la porte ET, c'est ce qui est symbolisé par le réseau de résistances reliées à cette valeur constante. [20]

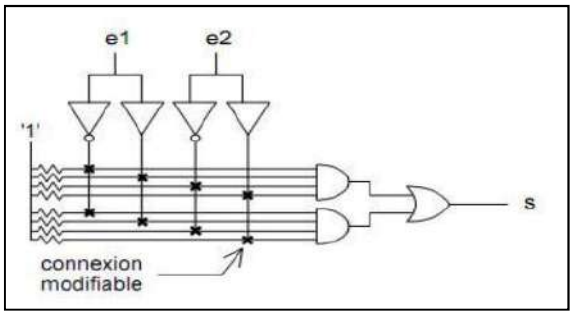


Figure II.4. *La structure d'un PLA avant sa programmation* [20]

4.2. Les cartes FPGAs

En 1984, Xilinx a présenté le premier FPGA (Field Programmable Gate Array). C'est un type alternatif du PLD qui s'est développé plus récemment. Ces circuits en silicium programmables sont devenus au milieu des années 90 des circuits plus complets grâce à l'intégration de ressources spécifiques dédiées, associées à de la mémoire interne et à des ports d'entrées/sorties plus flexibles.

Les FPGAs se composent d'une matrice de blocs logiques élémentaires (CLB) permettant de réaliser des fonctions combinatoires et des fonctions séquentielles. Tout autour de ces blocs logiques configurables, nous trouvons des blocs entrées/sorties IOB dont le rôle est de gérer les entrées-sorties réalisant l'interface avec les modules extérieurs et de ressources d'interconnexion (programmable interconnect) totalement flexibles contrairement au CPLDs ; la figure II.5 montre cette structure des FPGAs dont le programme, appelé « bitstream » n'a pas vocation à être exécuté par un microprocesseur, mais à configurer des blocs logiques et une logique d'interconnexions permettant de relier ces blocs logiques entre eux.

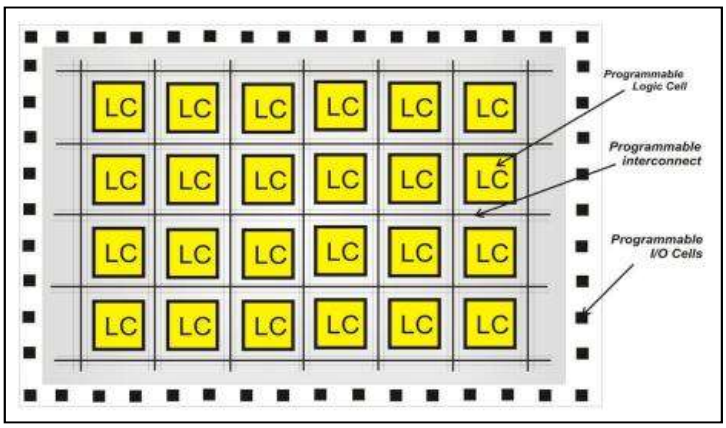


Figure II.5. *Physionomie d'un FPGA* [20]

La partie fondamentale du FPGA est appelée « look up table », ou LUT elle peut être considérée comme générateur de fonction combinatoire. Historiquement chez la plupart des fabricants de FPGA une LUT contient 16 bits et permettait donc de réaliser n'importe quelle fonction combinatoire à 4 variables. A chaque LUT peur être adjoint un registre élémentaire (bascule D) piloté par une horloge et un multiplexeur permettant d'utiliser le registre ou non. Chaque fabriquant de FPGA a par la suite réalisé des regroupements de LUTs et d'ajout de capacité supplémentaire. Aujourd'hui, la structure la plus utilisée est basée sur une look-up-Table de type RAM pour implémenter une fonction combinatoire plus une bascule D. La fonction de la LUT est de stocker la table de vérité de la fonction combinatoire à implémenter dans la cellule comme il est montré sur la figure II.6 : [20]

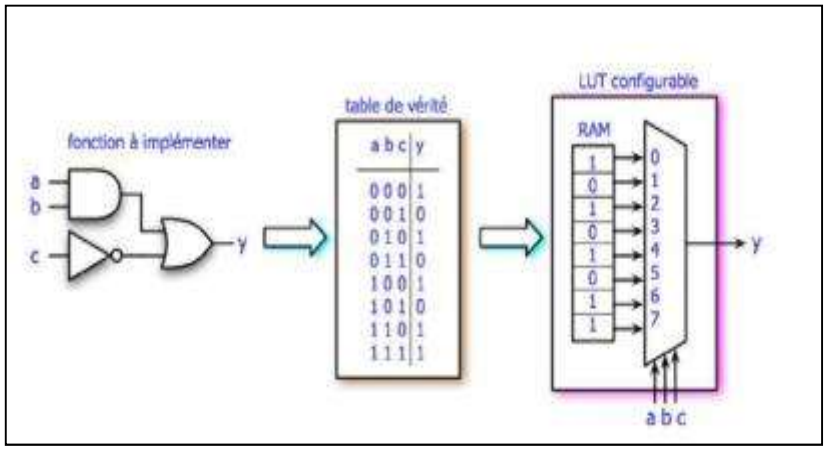


Figure II.6 : *Exemple de LUT* [20]

4.3. Flot de conception

Pour configurer un FPGA on utilise principalement le langage VHDL ou bien la conception « Schématique ». Un environnement software intégré est généralement fourni par le constructeur de FPGA. La configuration passe par un certain nombre d'étapes appelé flot de conception. Les étapes de conception sont décrites dans la figure II.7 : [21]

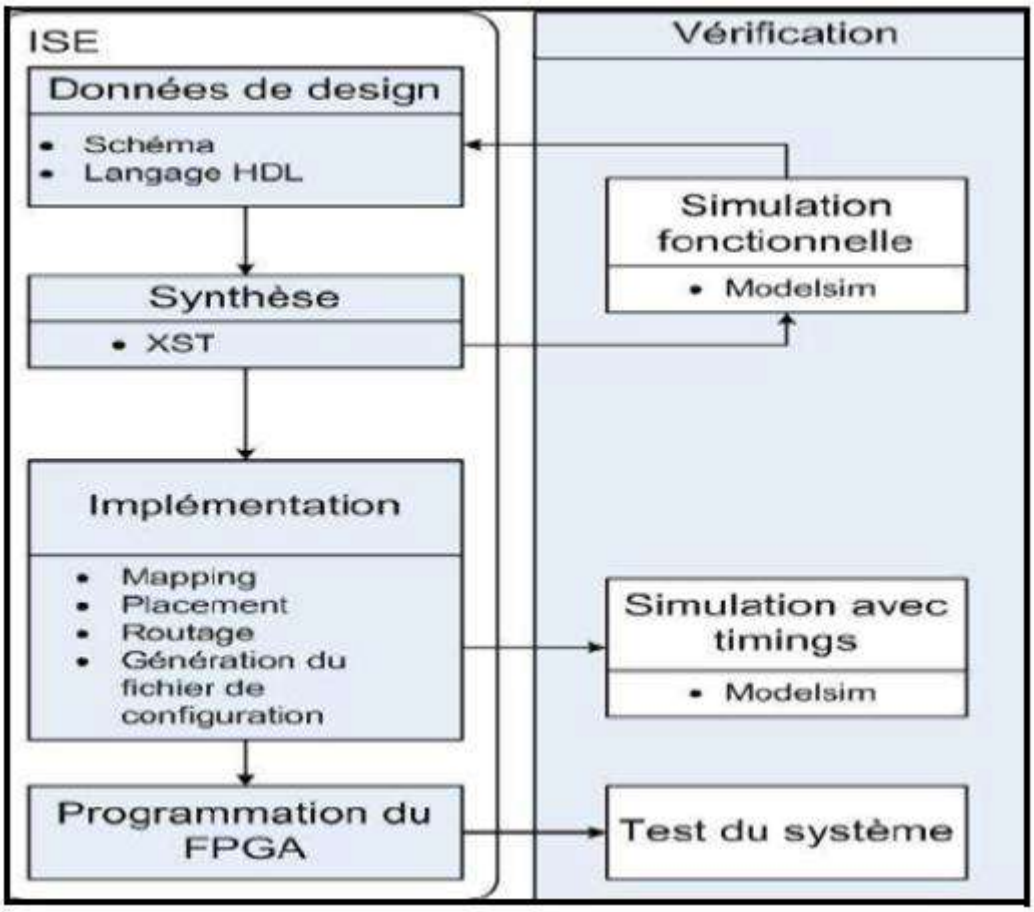


Figure II.7 : *Flot de conception* [21]

4.4. Design et conception

4.4.1. Les langages de description de matériel

Actuellement la densité de fonctions logiques (portes et bascules) intégrée dans les FPGA est telle (plusieurs milliers de portes voire millions de portes) qu'il n'est plus possible d'utiliser les outils de saisie d'un schéma pour développer les circuits. Les sociétés de développement et les ingénieurs ont voulu s'affranchir des contraintes technologiques des circuits. Ils ont donc créé des langages dits de haut niveau à savoir aux langages de description matérielle de haut niveau (les HDL, Hardware Description Langage). Deux d'entre eux ont émergé et sont couramment utilisés : VHDL (VHSIC Hardware Description Langage) et Verilog. Ces deux langages bénéficient du support de la quasi-totalité des logiciels. Tous les deux sont basés sur des descriptions écrite (texte) de la structure électroniques d'un circuit numérique, Ils permettent au code écrit d'être portable, c'est à dire qu'une description écrite pour un circuit peut être facilement utilisée pour un autre circuit. Il

faut avoir à l'esprit que ces langages dits de haut niveau permettent de matérialiser les structures électroniques d'un circuit.

Le VHDL est un langage de description matériel, Dans les années 80, le département de la défense aux Etats-Unis fait un appel d'offre pour avoir un langage de description matérielle numérique unique. Le langage VHDL (VHSIC Hardware Description Langage) est inventé pour répondre à ces critères. Ce langage se rapporte aux VHSIC (Very High Speed Integrated Circuit). Ce langage est ouvert au domaine public en 1985 et deviendra une norme en 1987 sous la dénomination d'IEEE 1076-1987. Des changements minimes se feront pour la seconde normalisation en 1993 et il portera le nom VHDL93. La dernière évolution de la norme est la norme IEEE 1076-2001. Les avantages qui se dégagent de ce langage sont la réutilisation de fichiers grâce à la notion de « paquetage » et la généricité individuelle des modèles. Cependant, la norme IEEE 1076-2001 ne permet pas seule la description mixte (numérique et analogique). Le Verilog et le VHDL ont des capacités techniques équivalentes. Le choix du langage est souvent dicté par la « culture » de l'équipe de développement ou de l'équipe de recherche. De plus, ce choix est parfois imposé par les outils disponibles, les logiciels de simulation et de synthèse, dont les langages associés sont fixés par des aspects économiques. Nous pouvons noter toutefois que les outils actuels permettent de mélanger ces deux langages.

VHDL a continué son évolution et la description analogique vient d'y être intégrée sous la référence IEEE 1076.6 (VHDL-AMS). L'extension vise l'analogique. L'électronicien a toujours utilisé des outils de description pour représenter des structures logiques ou analogiques. Le schéma structurel que l'on utilise depuis si longtemps et si souvent n'est en fait qu'un outil de description graphique. Aujourd'hui, l'électronique numérique est de plus en plus présente et tend bien souvent à remplacer les structures analogiques utilisées jusqu'à présent. Ainsi, l'ampleur des fonctions numériques à réaliser nous impose l'utilisation d'un autre outil de description. Il est en effet plus aisé de décrire un compteur ou un additionneur 64 bits en utilisant l'outil de description VHDL plutôt qu'un schéma.

Le deuxième point fort du VHDL est d'être "un langage de description de haut niveau". D'autres types de langage de description, comme ABEL dont est issu le VHDL par exemple, ne possèdent pas cette appellation. En fait, un langage est dit de haut niveau lorsqu'il fait la plus possible abstraction de l'objet auquel ou pour lequel il est écrit. Dans le cas du langage VHDL, il n'est jamais fait référence au composant ou à la structure pour lesquels on l'utilise.

Ainsi, il apparaît deux notions très importantes : portabilité des descriptions VHDL, c'est-à-dire, possibilité de cibler une description VHDL dans le composant ou la structure que l'on souhaite en utilisant l'outil que l'on veut. [21]

4.4.2. Traduction et optimisation

La description VHDL définit la fonctionnalité du circuit en termes de blocs définis " haut niveau ". Progressivement, les blocs sont détaillés précisément jusqu'à une description proche des ressources matérielles. En effet, le langage VHDL autorise trois niveaux de description.

- le niveau structurel décrit le câblage des composants élémentaires,
- le niveau flot de données décrit les transformations d'un flot de données de l'entrée à la sortie,
- le niveau comportemental décrit le fonctionnement par des blocs programmes appelés « Process » qui échangent des données au moyen de signaux et comprenant des instructions séquentielles.

Une description VHDL d'un circuit subit toute une suite de transformations avant que ce circuit soit effectivement configuré sur un FPGA.

L'étape de synthèse consiste à lire la description VHDL du circuit pour en extraire toute la structure logique. Lorsque des composants de ce circuit sont décrits de manière comportementale, c'est le rôle du synthétiseur d'inférer la logique nécessaire à la réalisation des comportements spécifiés. Cette tâche est la modélisation au niveau Transfert de registres (Register Transfer Level). Cette modélisation revient à décrire l'implémentation sous forme d'éléments séquentiels (registres ou bascules) et de combinaisons logiques entre les différentes entrées/sorties des éléments séquentiels et des entrées/sorties primaires du circuit. Cette modélisation est codée à l'aide d'un langage de programmation. Cela se fait soit en VHDL, soit en Verilog. Le propre d'une description RTL est d'être automatiquement synthétisable en portes logiques combinatoires (portes ET, OU, multiplexeur, etc.) et séquentiels (comme les bascules D synchrones) issues d'une bibliothèque de cellules standard (Standard Cell Library). Pour cela, le codage RTL en VHDL ou Verilog doit suivre certaines règles précises afin que le code soit dit « synthétisable ». Il est tout à fait possible de décrire une fonctionnalité équivalente en VHDL ou Verilog qui ne soit pas synthétisable. On appelle cela une description comportementale de plus haut niveau que le niveau RTL. Un autre rôle du synthétiseur est d'appliquer diverses optimisations logiques au circuit. Il cherche ainsi à

minimiser toutes les expressions logiques utilisées par ce circuit. Enfin, le synthétiseur après compilation produit une netlist c'est-à-dire une description de haut niveau (liste de fils et de porte logiques) à partir de la description RTL du circuit. [21]

4.4.3. Partitionnement, placement et routage

La netlist du circuit est ensuite adaptée aux primitives logiques du FPGA cible. Cette étape de ciblage technologique permet ainsi de passer d'une description structurelle générique à une description spécialisée pour une architecture précise. Les éléments de base ne sont alors plus de simples portes logiques mais des cellules programmables du FPGA.

La phase de « mapping » ayant identifié les ressources logiques nécessaires à l'implémentation d'un circuit, la tâche de placement/routage consiste alors à disposer et connecter ces ressources sur la surface du FPGA. Il est possible de donner au placeur/routeur des contraintes spatiales ou temporelles, pour fixer par exemple la position des ports d'entrée/sortie du circuit ou bien une période d'horloge maximale.

A l'issue de l'étape de placement/routage, le circuit obtenu est tel qu'il sera programmé sur le FPGA. C'est sur ce modèle que sont calculées les estimations de surface et de latence du circuit. [21]

4.4.4. Simulation

La simulation logicielle d'un circuit est une étape essentielle de sa conception. Il faut en effet s'assurer que celui-ci fonctionne correctement avant de le configurer sur le FPGA. La simulation permet ainsi de fournir au circuit des vecteurs de test en entrée, puis de vérifier que le circuit se comporte de la manière prévue. Bien entendu, la simulation ne garantit pas la correction du circuit. Mis à part pour de petits opérateurs, il est en effet impossible de réaliser une simulation exhaustive à cause de la lenteur de la simulation logicielle. Il faut donc essayer de choisir des vecteurs de test adaptés à l'opérateur pour couvrir le mieux possible les différents cas de figure. Le simulateur logiciel utilisé pour nos architectures est le simulateur ModelSim, développée par Xilinx. [21]

4.4.5. Configuration (programmation)

Enfin, la dernière étape avant la programmation effective du FPGA consiste à générer un bitstream. Ce bitstream est en fait un fichier contenant tous les bits de configuration du

FPGA, construit pour correspondre parfaitement au circuit décrit lors du placement/routage. Une fois le bitstream créé, il ne reste plus qu'à le télécharger sur le FPGA grâce à une interface dédiée. Le FPGA restera ainsi configuré tant qu'il sera sous tension, ou bien lorsqu'un autre circuit y sera programmé suivant le même procédé.

ISE, est l'environnement de développement fourni par Xilinx. Cette suite logicielle inclut notamment le synthétiseur XST, ainsi que tous les outils évoqués précédemment. [21]

5. Conclusion

Dans ce chapitre, nous avons présentés la stratégie de stimulation cochléaire CIS, qui est la plus utilisé dans les implants commercialisés ; nous avons donné aussi un aperçu sur les caractéristiques des filtres passe bandes qui jouent un rôle important dans la stratégie CIS.

Et dans le but d'implémenter le banc de filtres de la stratégie CIS dans un circuit FPGA, nous avons expliqué le principe de l'implémentation dans les circuits logiques programmables, afin de pouvoir réaliser cette tâche dans le chapitre suivant.

CHAPITRE III :

Conception, simulation et implémentation du banc de filtres de la stratégie CIS dans FPGA

1. Introduction :

Le filtrage et l'analyse spectrale sont des techniques de base dans le traitement numérique du signal. Un filtre numérique est une combinaison linéaire d'échantillons. La réalisation d'un filtre sous Matlab doit comporter plusieurs étapes successives afin de le générer en spécifiant les paramètres caractérisant ce dernier.

Dans ce chapitre, nous présentons la méthode de génération du programme VHDL et du "Tb" (test bench) pour le banc de 16 filtres qui approxime la fonction réalisée par la cochlée pathologique de l'être humain, ensuite une co-simulation avec le logiciel Modelsim et avec Matlab est effectuée afin de confirmer la justesse du programme VHDL généré.

2. Réalisation du banc de filtres :

Le tableau suivant présente les bandes fréquentielles du banc de filtres a réalisé.

Tableau III.1. *Les bandes fréquentielles des filtres*

Numéro du filtre	La bande passante
1er filtre	[125 175]
2eme filtre	[225 275]
3eme filtre	[325 375]
4eme filtre	[420 480]
5eme filtre	[530 605]
6eme filtre	[655 745]
7eme filtre	[790 890]
8eme filtre	[940 1060]
9eme filtre	[1105 1235]
10eme filtre	[1285 1455]
11eme filtre	[1505 1695]
12eme filtre	[1745 1955]
13eme filtre	[2005 2295]
14eme filtre	[2345 2655]
15eme filtre	[2705 3095]
16eme filtre	[3145 3655]

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

```
Fs=7320; % Fréquence d'échantillonnage
N=2;     % l'ordre du filtre
Ast1=60;
Ast2=80;
Ap=1;
Fp1=125; Fp2=175;
filtre_1=fdesign.bandpass('n,Fp1,Fp2,Ast1,Ap,Ast2',2,Fp1,Fp2,Ast1,Ap,Ast2,Fs);
```

Sachant que :

- Ast1, Ast2 : 1ere et 2eme stop-band atténuation (dB).
- Ap : ondulation passe-bande (dB).
- Fp1, Fp2 : 1ere et 2eme fréquence passe-bande (Hz).

Ces paramètres sont présentés sur la figure III.1 représentant le Gabarit d'un tel filtre :

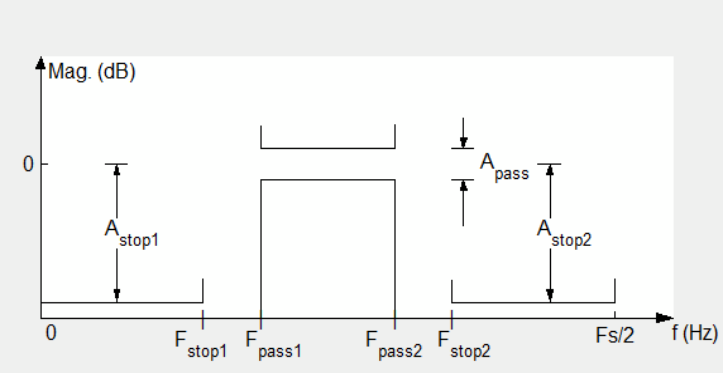


Figure III.1. *Les caractéristiques d'un filtre passe-bande*

La génération du filtre doit suivre une structure avant de passer à la simulation de ce filtre, pour notre cas nous avons opté pour la structure elliptique. La commande suivante permet de réaliser cette tâche :

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

Après la génération du banc de 16 filtres passe-bandes, selon les bandes du tableau III.1, nous avons représenté la réponse en amplitude de ce dernier grâce à la commande « fvtool » :

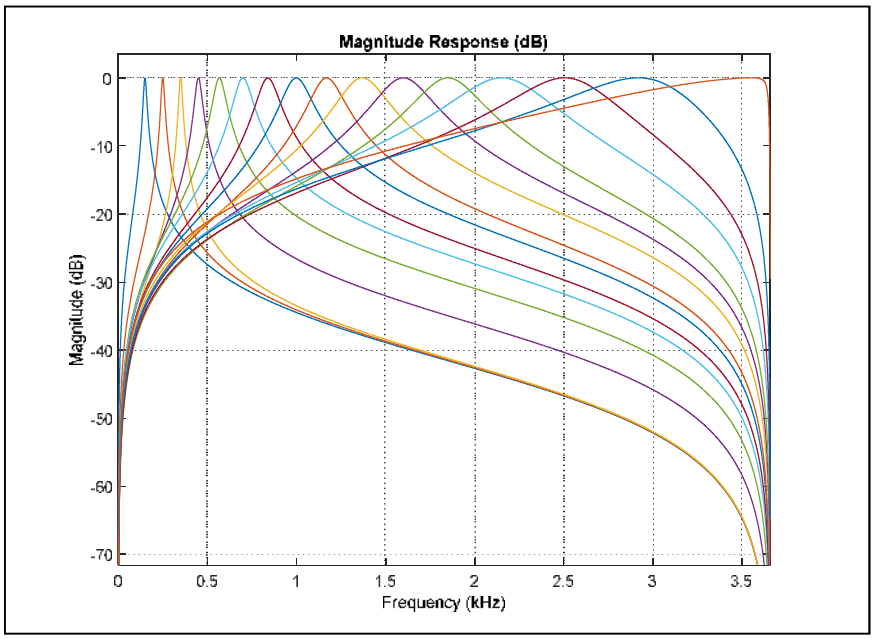


Figure III.2. *La réponse en amplitude du banc de filtres*

2.1. Quantification des filtres :

Avant de passer à l'implémentation du filtre dans un circuit numérique, une étape de quantification est nécessaire. Les données du filtre sont définies au mode à point fixe. Nous avons fixé 8 bits pour le mot du signal d'entrée du filtre, 16 bits pour les coefficients de la réponse impulsionnelle et 20 bits pour le numérateur et le dénominateur de la fonction de transfert du filtre, mais avant de spécifier ça, il est nécessaire de convertir la structure de filtre par défaut (forme directe 2 : DF2SOS) à la structure désirée (forme directe 1 : DF1SOS), cette dernière c'est la seule structure qui permet la quantification en Matlab.

Le script suivant montre l'étape de quantification réalisée :

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

Nous vérifions la réponse en amplitude d'un seul filtre du banc (1er filtre) avec la commande fvtool du Matlab sous une échelle logarithmique par la commande suivante :

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

La figure III.3 nous montre qu'il n'y avait pas de changement entre la réponse en amplitude du filtre réel et le filtre quantifié (1er filtre).

La même étape sera répétée sur tous les filtres du banc.

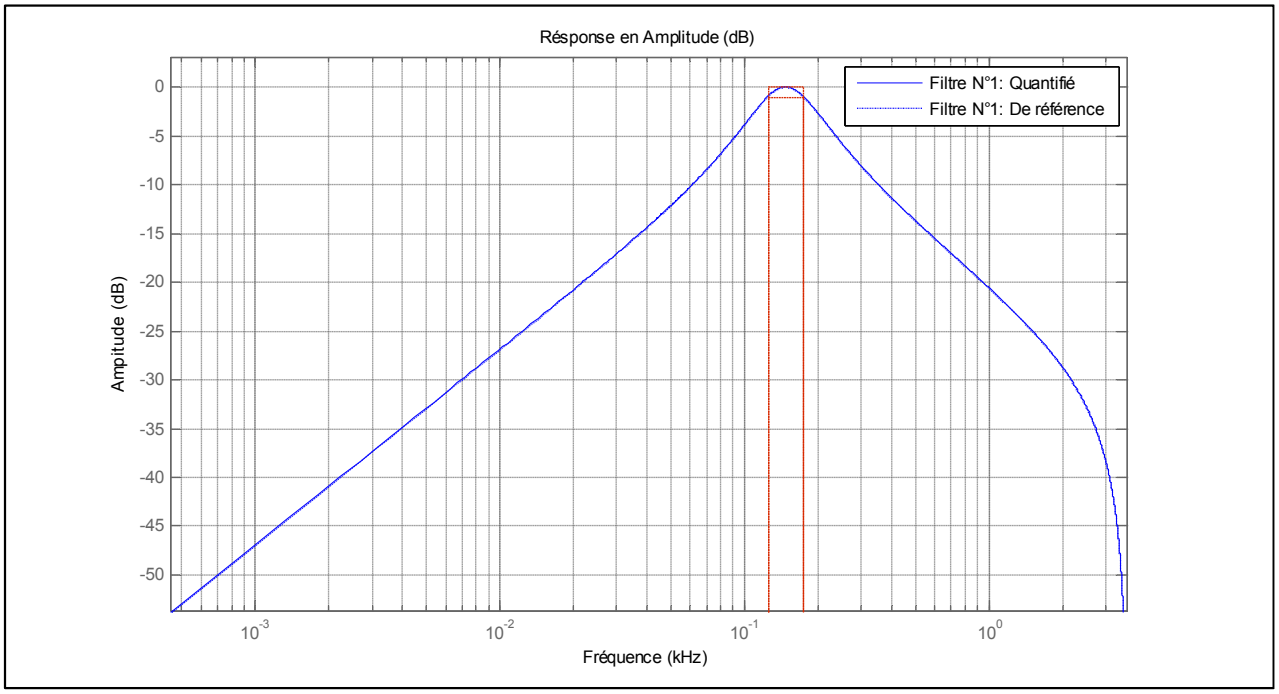


Figure III.3. *Réponse en amplitude après quantification du 1er filtre*

Une étape principale pour la conception matérielle (sous FPGA) d'un filtre est de vérifier si les valeurs caractéristiques du filtre sont à une échelle raisonnable, et d'ajuster l'échelle si nécessaire. La commande suivante permet de réaliser cette tâche.

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

3. Génération du code VHDL du banc de filtres :

En commençant par le filtre correctement quantifié, nous pouvons générer le code VHDL de ce dernier.

Nous créons un espace de travail provisoire (temporaire), et générons le code VHDL pour ce filtre qui est met dans l'éditeur sous l'extension « .vhd ».

Le script suivant montre cette étape de génération :

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

4. Génération de « test bench » du banc de filtres :

Puisque la bande passante de ce filtre est très faible par rapport à la fréquence d'échantillonnage, un stimulus d'entrée du filtre est construit afin d'examiner l'implémentation

du filtre en VHDL. Nous construisons le signal d'entrée de test avec un cycle de 50 à 300 Hz avec un pas de 50 Hz, et nous appliquons ce signal sur la structure de filtre en VHDL dans le but de vérifier que les résultats correspondent exactement aux résultats obtenus sous Matlab.

Cette dernière tâche est réalisée par le script suivant :

```
stimulus = [];
for n = [50, 100, 150, 200, 250, 300]
  stimulus = [stimulus, sin(2*pi*n/Fs*(0:Fs/n))];
end
generatehdl(Hd, '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 en Modelsim :

Dans un premier temps, nous commençons par la visualisation du signal de sortie de ce filtre en Matlab, en exécutant les commandes suivantes :

```
Temps = (0:length(stimulus) - 1);
y = filter(Hd, stimulus);
subplot(2,1,1); plot(Temps, stimulus);
axis([0 length(stimulus) -1.1 1.1]);
title('Le signal d''entrée du filtre N°1 a implémenter sur FPGA');
xlabel('Echantillons');grid
subplot(2,1,2); plot(Temps, y);
axis([0 length(stimulus) -1.1 1.1]);
title('la réponse du filtre au signal d''entrée');
xlabel('Echantillons');grid
```

Le résultat est donné par la figure III.4.

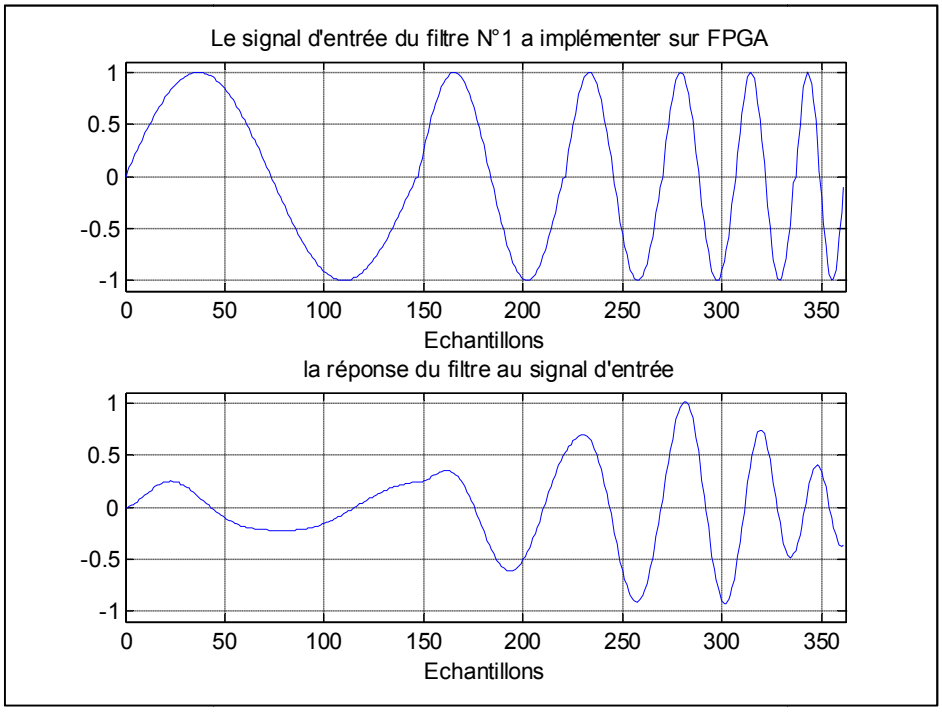


Figure III.4. *Le résultat de la simulation du 1^er^ filtre obtenu par Matlab*

Afin de comparer le résultat obtenu avec celui obtenu par programme VHDL, nous devons passer à la simulation du programme « filtre1.vhd » par Modelsim sous la même excitation (c.à.d. par le « test bench, filtre1_tb.vhd»).

Le résultat obtenu avec le logiciel Modelsim est représenté sur la figure III.5.

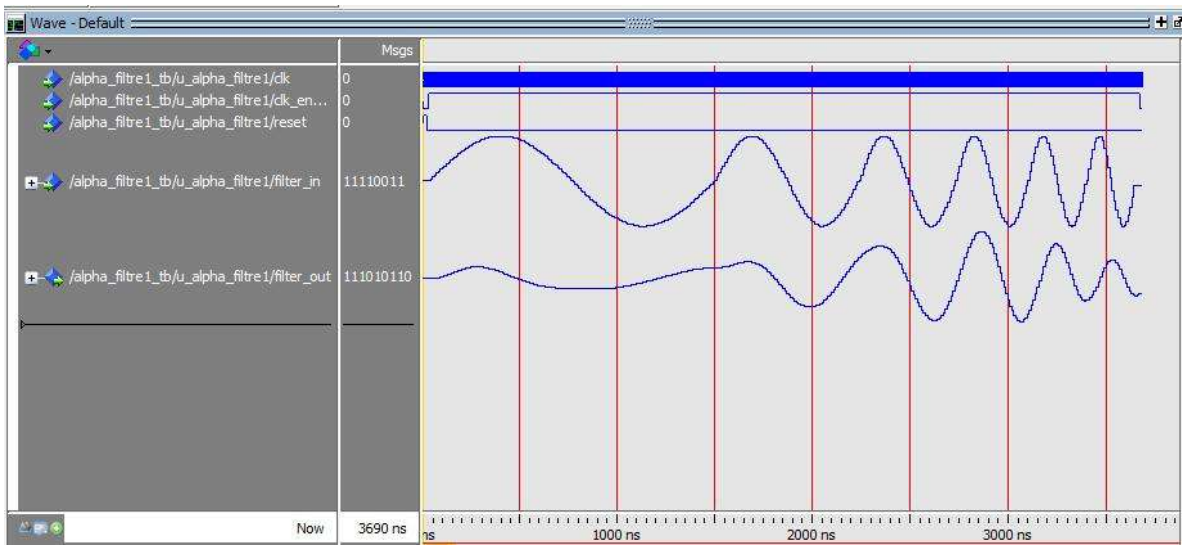


Figure III.5. *Le résultat de la simulation du 1^er^ filtre obtenu par Modelsim*

Tous ces opérations seront répéter pour les 15 autres filtres afin d'obtenir un banc de filtres composer de 16 filtres numériques de type passe-bande qui approxime la cochlée pathologique.

6. Simulation de la stratégie CIS en utilisant le banc de filtres réalisé :

Avant de commencer, il faut exporter un fichier audio pour faire les essais de la stratégie CIS.

```
[y,Fs]=wavread('clean_2.wav');
```

Après nous devons filtrer ce signal par le banc de filtres, en utilisant le script suivant :

```
% filtrage du signal audio par le banc de filtres
sortie_filtre1=filter(Hd1,y);
sortie_filtre2=filter(Hd2,y);
sortie_filtre3=filter(Hd3,y);
sortie_filtre4=filter(Hd4,y);
sortie_filtre5=filter(Hd5,y);
sortie_filtre6=filter(Hd6,y);
sortie_filtre7=filter(Hd7,y);
sortie_filtre8=filter(Hd8,y);
sortie_filtre9=filter(Hd9,y);
sortie_filtre10=filter(Hd10,y);
sortie_filtre11=filter(Hd11,y);
sortie_filtre12=filter(Hd12,y);
sortie_filtre13=filter(Hd13,y);
sortie_filtre14=filter(Hd14,y);
sortie_filtre15=filter(Hd15,y);
sortie_filtre16=filter(Hd16,y);
```

Le script suivant nous permet de visualiser la réponse du 1er, 5ème, 8ème et 16ème filtre du banc :

```
figure(1)
subplot(2,2,1),plot(sortie_filtre1),title('sortie du 1er filtre')
subplot(2,2,2),plot(sortie_filtre5),title('sortie du 5eme filtre')
subplot(2,2,3),plot(sortie_filtre8),title('sortie du 8eme filtre')
subplot(2,2,4),plot(sortie_filtre16),title('sortie du 16eme filtre')
```

La figure III.6 présente le résultat obtenu.

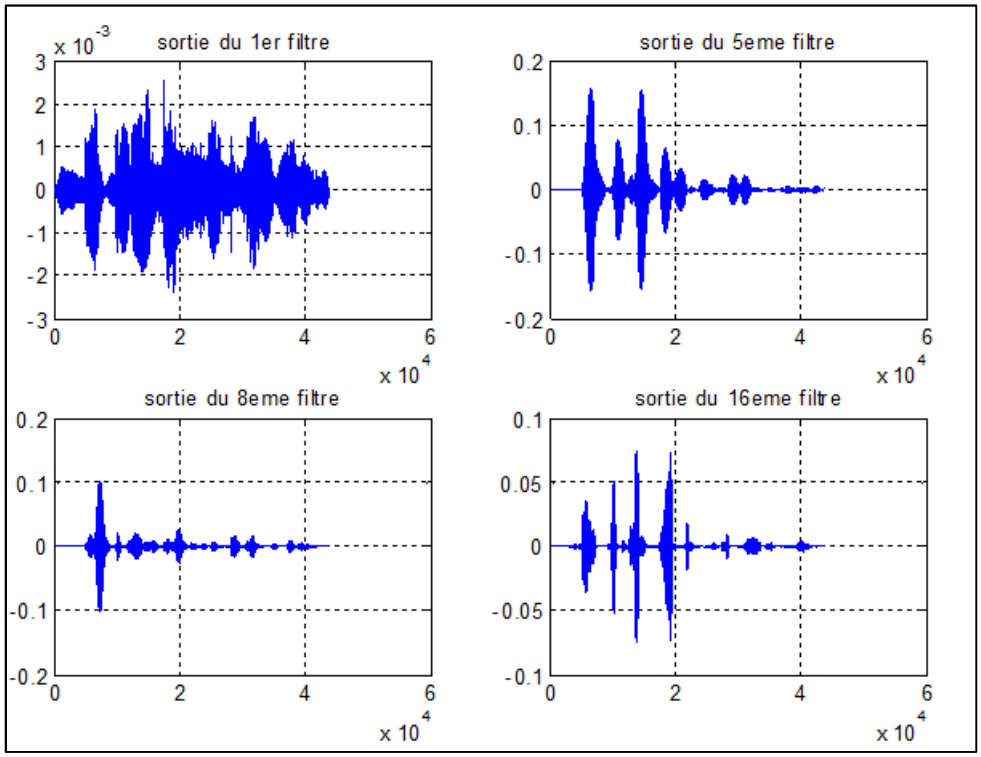


Figure III.6. *Les sorties des filtres 1, 5, 8 et 16*

L'opération suivante sert à extraire l'enveloppe de la sortie de chaque filtre du banc, le résultat est obtenu sur la figure III.7.

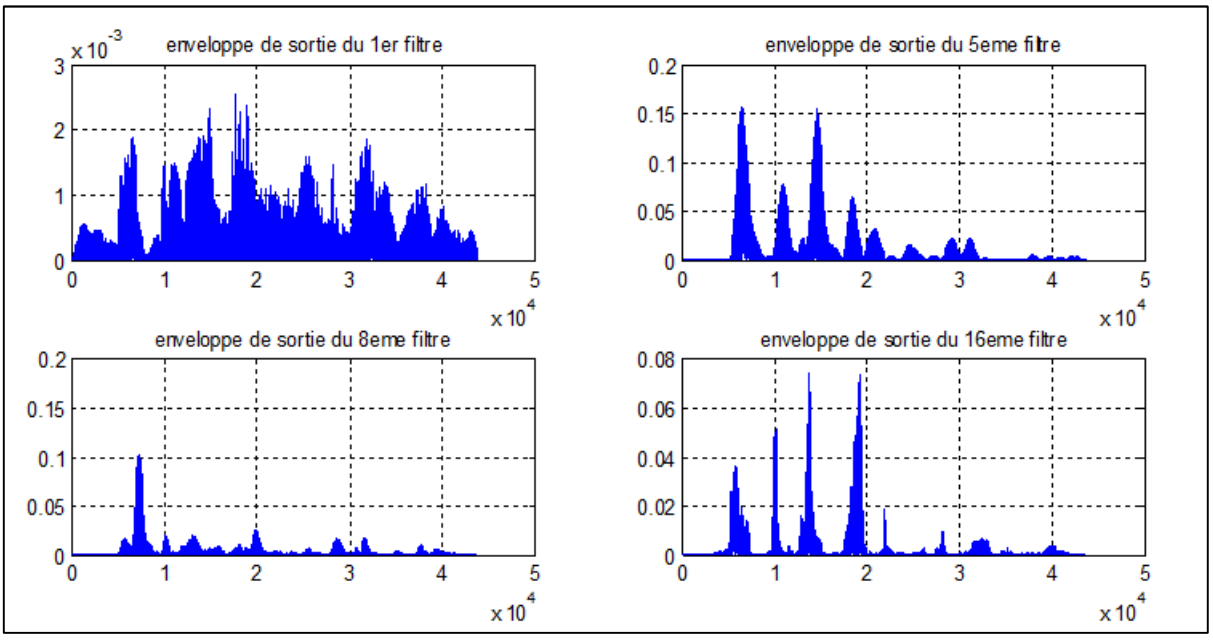


Figure III.7. *Extraction d'enveloppe des signaux de sortie des filtres 1, 5, 8 et 16*

Ensuite un filtre passe bas est appliqué aux signaux d'enveloppe pour enlever les hautes fréquences du signal, généralement la fréquence de coupure de ce filtre est de 200Hz.

Le résultat est présenté par la figure III.8.

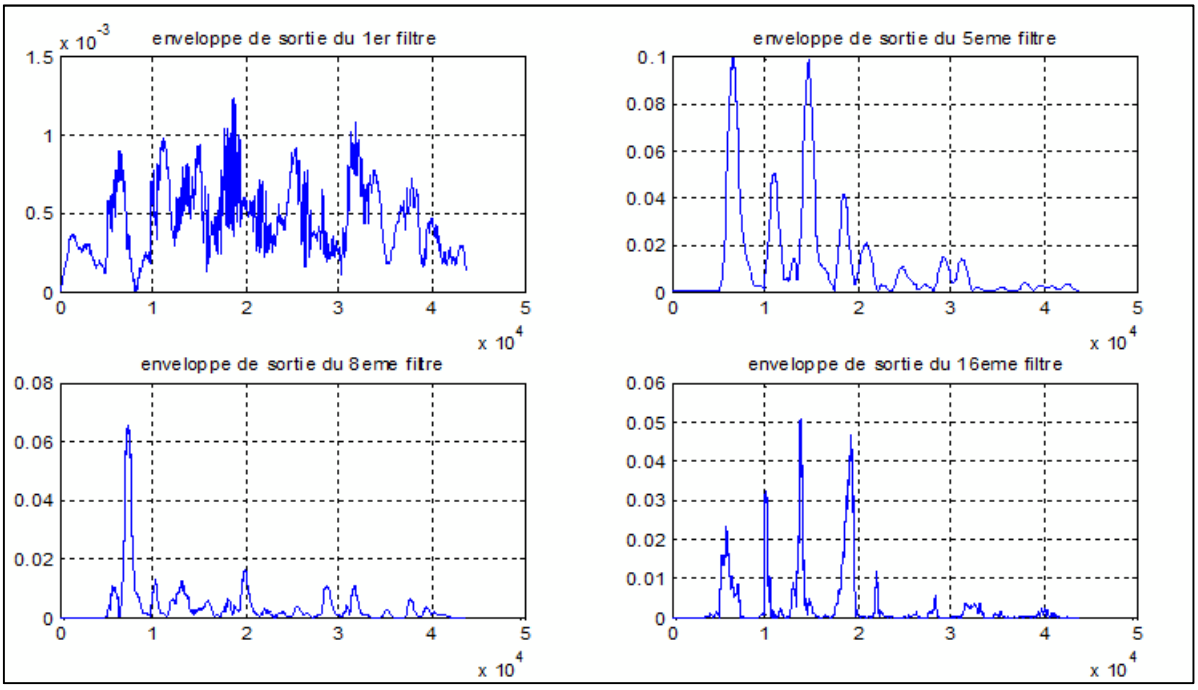


Figure III.8. *Filtrage des enveloppes extraites par un filtre passe bas*

La modulation des signaux d'enveloppes, nous oblige à créer un signal carré bi-phasique comme il est montré dans la figure III.9.

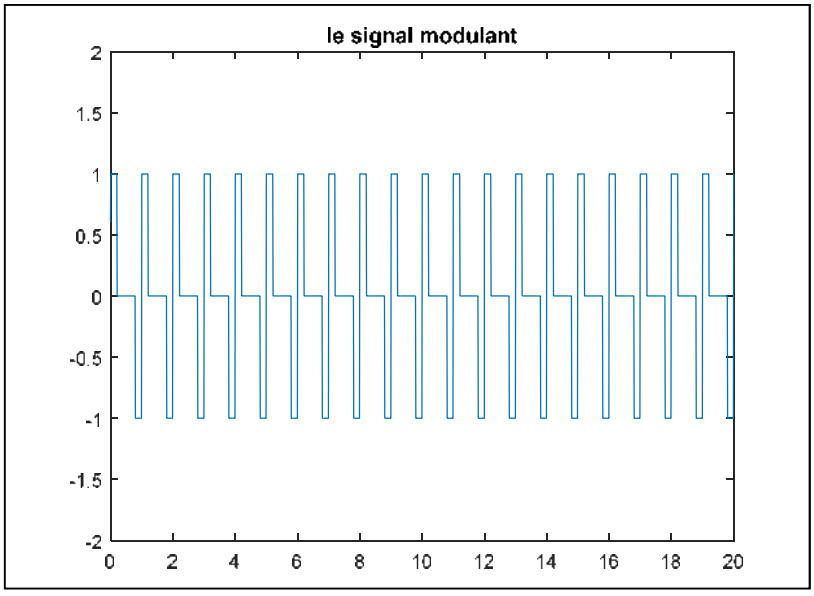


Figure III.9. *Le signal à modulé*

Après la modulation nous obtenons les résultats présentés par la figure III.10.

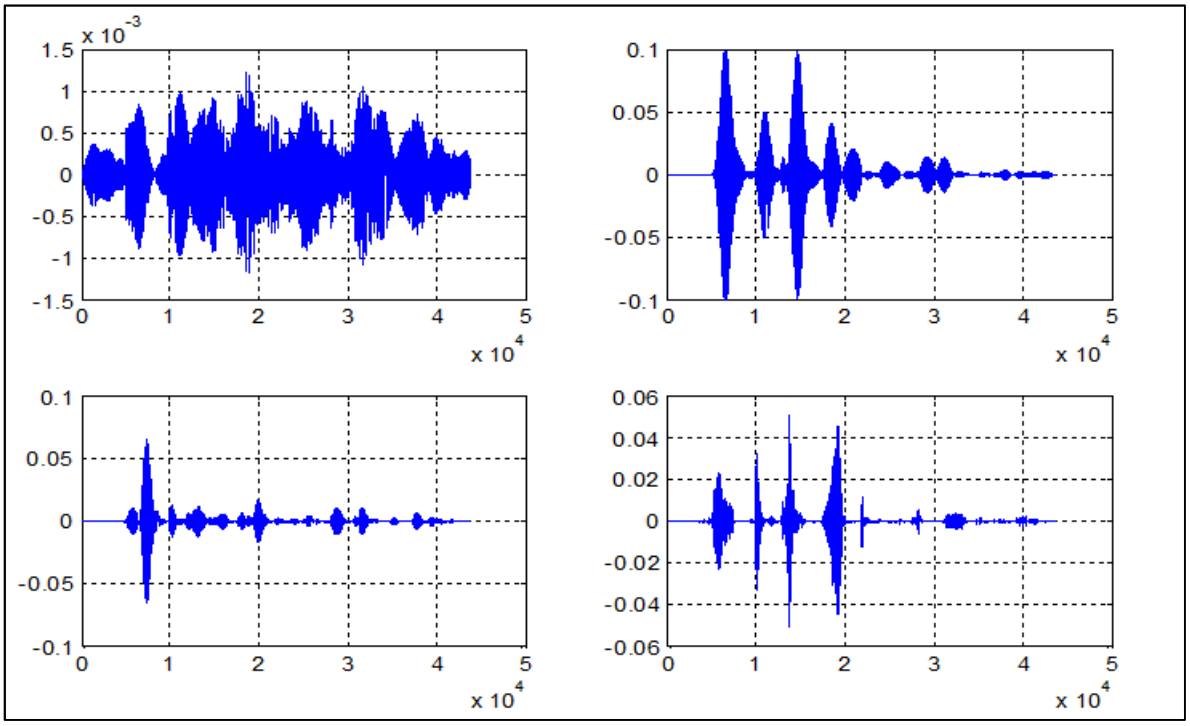


Figure III.10. *Les signaux modulés de la stratégie CIS des filtres1, 5, 8 et 16*

Une partie zoomée du signal de la stratégie CIS du 1er, 5ème, 8ème et 16ème filtre est présentée dans la figure III.11 :

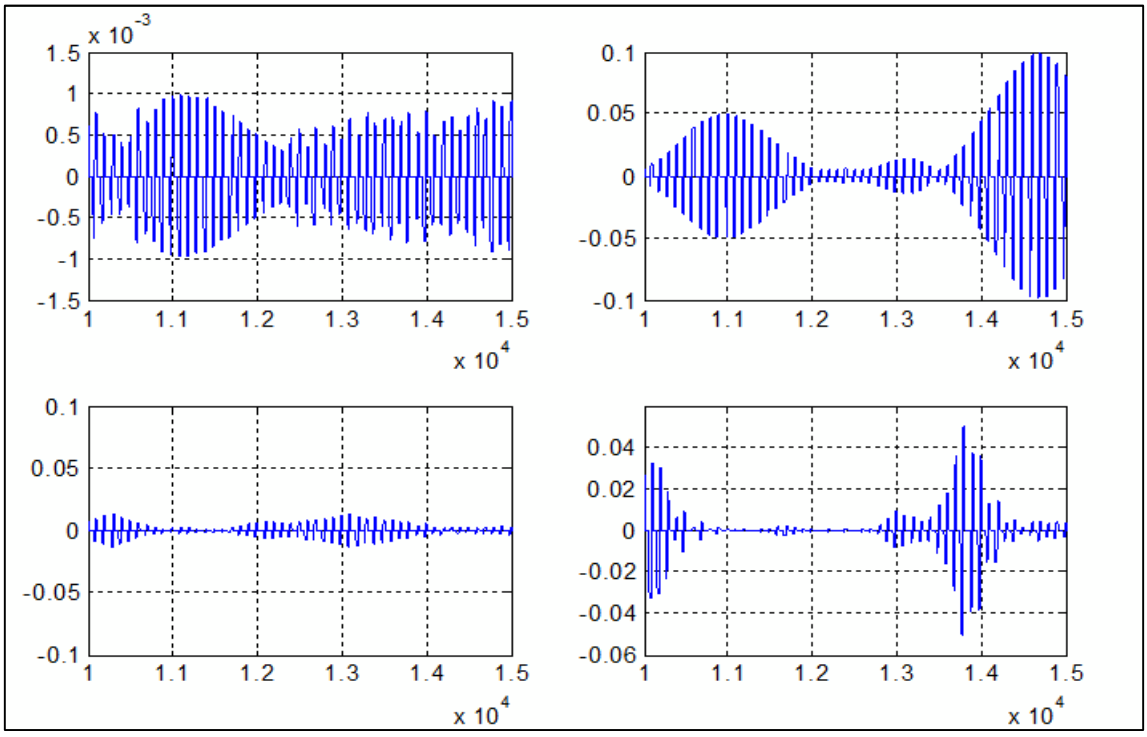


Figure III.11. *Une partie zoomé des signaux modulés de la stratégie CIS des filtres 1, 5, 8 et 16*

Nous pouvons bien remarquer les signaux de stimulation résultants.

7. Présentation du logiciel XILINX ISE :

L'environnement que nous allons exploiter au cours de notre travail est XILINX ISE (Integrated Software Environment). C'est un logiciel multitâche qui possède différents outils permettant la création de systèmes ou circuits numériques.

Le navigateur de projet ISE sera utilisé comme outil de conception par excellence, lors de cette réalisation. Ce logiciel XILINX ISE permet la simulation de la description, et la synthèse du circuit logique équivalent, puis le placement et le routage du circuit sur un prototype correspondant à une technologie FPGA bien précise, et enfin lorsque toute les vérifications sont effectuées vient l'implantation sur un FPGA réel ce qui correspond à générer le fichier de configuration du circuit cible choisi, afin d'établir les interconnections des cellules logiques correspondantes au circuit logique conçu avec optimisation de ressources disponibles au niveau de circuit programmable FPGA.

ISE offre un environnement de conception centralisé extrêmement efficace qui regroupe tous les outils nécessaires la conception, la simulation et l'implémentation d'un projet d'envergure ainsi que la configuration de la carte.

La conception de circuits sur XILINX ISE met en œuvre quatre outils : un éditeur de texte ou entrée graphique, un simulateur, un synthétiseur et un placeur-routeur. L'éditeur de texte ou entrée graphique permet l'introduction de la description sur un outil CAO c'est dire de dessiner ou décrire le circuit avec une interface graphique ou textuelle. La simulation du système permet la vérification de la validité du code avant-synthèse, après-synthèse et même après placement-routage. Les deux étapes de synthèse et de routage qui suivent permettent pour de réaliser la transcription de la description d'une forme texte vers la forme graphique(RTL) basée sur des blocs logiques, et pour la deuxième étape nommée routage, d'élaborer une adaptation du circuit logique synthétisé sur les ressource disponible dans le circuit FPGA ciblé.

8. Création d'un nouveau projet sous ISE :

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 »).

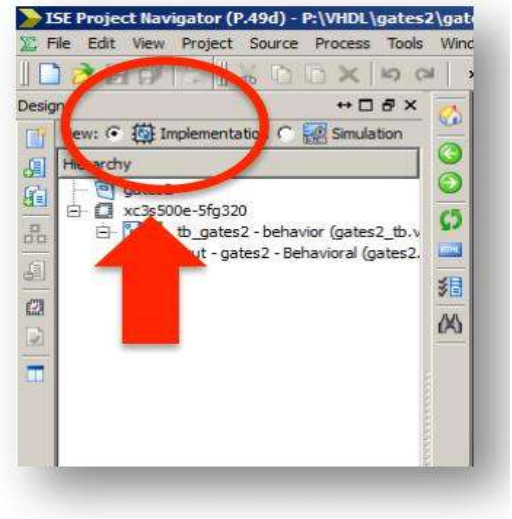


Figure III.12. *Choix d'implémentation*

Tout d'abord, avant de commencer l'implémentation d'un circuit sur FPGA sous ISE avec les différentes étapes, nous devons créer un nouveau projet en utilisant la structure suivante :

File ⟶ New Project.

Ensuite, nous obtenons la fenêtre suivante où nous devons nommer ce nouveau projet et préciser son type et le mettre dans sa location.

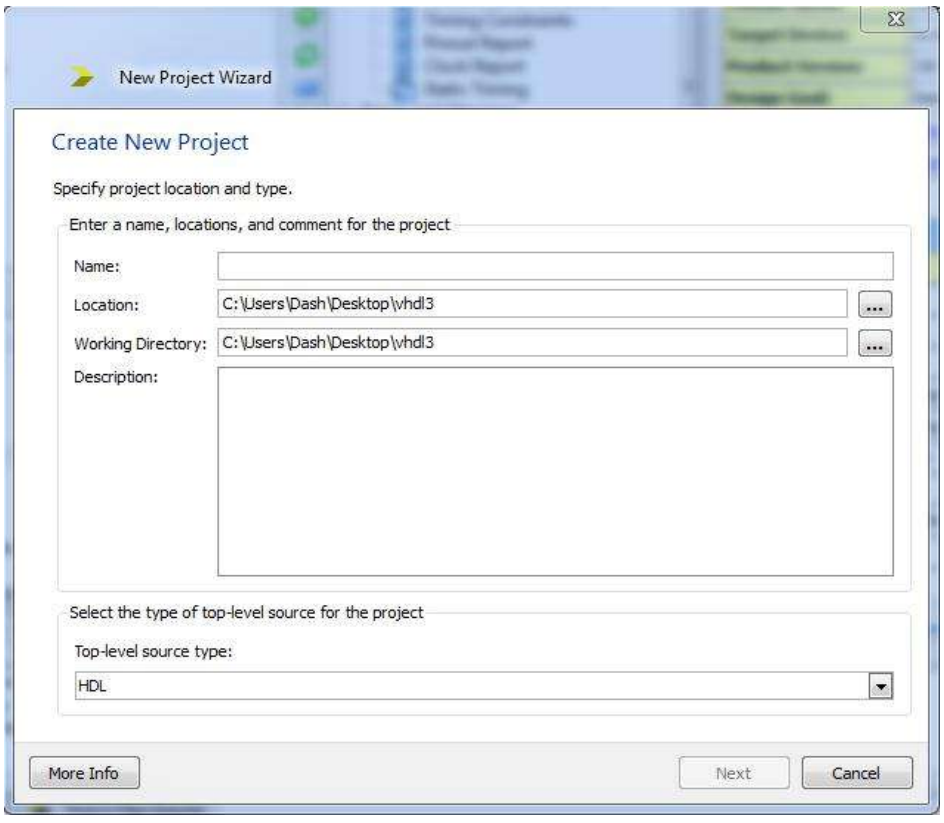


Figure III.13. *Création d'un nouveau projet*

9. Chargement du code VHDL du banc de filtres sous ISE :

A l'aide du logiciel ISE, nous pouvons créer un projet, créer des fichiers VHDL, les compiler, créer des testbench, et simuler. Nous allons maintenant nous focaliser sur un deuxième aspect de l'outil ISE et qui concerne l'implémentation. En fait, il y a de nombreuses étapes avant d'arriver à un bitstream, et l'objectif de ce travail est aussi de comprendre ces différentes étapes.

Tout d'abord, nous avons les bibliothèques et les entités de chaque filtre :

```
1  LIBRARY IEEE;
2  USE IEEE.std_logic_1164.all;
3  USE IEEE.numeric_std.ALL;
4
5  ENTITY filtre1 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_out                  :   OUT   std_logic_vector(8 DOWNTO 0)  -- sfix9_En7
11          );
12
13 END filtre1;
14
15
16 LIBRARY IEEE;
17 USE IEEE.std_logic_1164.all;
18 USE IEEE.numeric_std.ALL;
19
20 ENTITY filtre2 IS
21    PORT( clk                         :   IN    std_logic;
22          clk_enable                  :   IN    std_logic;
23          reset                       :   IN    std_logic;
24          filter_in                   :   IN    std_logic_vector(7 DOWNTO 0); -- sfix8_En7
25          filter_out                  :   OUT   std_logic_vector(8 DOWNTO 0)  -- sfix9_En7
26          );
27
28 END filtre2;
```

filters.vhd | Design Summary (Implemented)

Figure III.14. *Les bibliothèques et les entités du banc de filtres*

Et après, les architectures du banc de filtres (16 filtres) :

```
241  ----------------------------------------------------------------
242  --Module Architecture: filtre1
243  ----------------------------------------------------------------
244  ARCHITECTURE rtl OF filtre1 IS
245    -- Local Functions
246    -- Type Definitions
247    TYPE numdelay_pipeline_type IS ARRAY (NATURAL range <>) OF signed(19 DOWNTO 0); -- sfix20_En15
248    TYPE dendelay_pipeline_type IS ARRAY (NATURAL range <>) OF signed(19 DOWNTO 0); -- sfix20_En15
249    -- Constants
250    CONSTANT coeff_b1_section1             : signed(15 DOWNTO 0) := to_signed(17544, 16); -- sfix16_En21
251    CONSTANT coeff_b2_section1             : signed(15 DOWNTO 0) := to_signed(0, 16); -- sfix16_En21
252    CONSTANT coeff_b3_section1             : signed(15 DOWNTO 0) := to_signed(-17544, 16); -- sfix16_En21
253    CONSTANT coeff_a2_section1             : signed(15 DOWNTO 0) := to_signed(-32225, 16); -- sfix16_En14
254    CONSTANT coeff_a3_section1             : signed(15 DOWNTO 0) := to_signed(16110, 16); -- sfix16_En14
255    CONSTANT scaleconst2                   : signed(15 DOWNTO 0) := to_signed(32761, 16); -- sfix16_En15
256    -- Signals
257    SIGNAL input_register                  : signed(7 DOWNTO 0); -- sfix8_En7
258    -- Section 1 Signals
259    SIGNAL numtypeconvert1                 : signed(19 DOWNTO 0); -- sfix20_En15
260    SIGNAL a1sum1                          : signed(19 DOWNTO 0); -- sfix20_En11
261    SIGNAL dentypeconvert1                 : signed(19 DOWNTO 0); -- sfix20_En15
262    SIGNAL numdelay_section1               : numdelay_pipeline_type(0 TO 1); -- sfix20_En15
263    SIGNAL dendelay_section1               : dendelay_pipeline_type(0 TO 1); -- sfix20_En15
264    SIGNAL a2mul1                          : signed(35 DOWNTO 0); -- sfix36_En29
265    SIGNAL a3mul1                          : signed(35 DOWNTO 0); -- sfix36_En29
266    SIGNAL b1mul1                          : signed(35 DOWNTO 0); -- sfix36_En36
267    SIGNAL b3mul1                          : signed(35 DOWNTO 0); -- sfix36_En36
268    SIGNAL b1sum1                          : signed(19 DOWNTO 0); -- sfix20_En18
```

filters.vhd | Design Summary (Implemented)

Figure III.15. *Les architectures des filtres*

10. Les étapes de l'implémentation du banc de filtres sur FPGA sous ISE :

Pour implémenter un circuit sur un FPGA, il y a trois éléments indispensables pour y parvenir :

1. Disposer de la description logique du circuit (schématique, diagrammes d'états ou code VHDL).
2. L'environnement de développement est choisi en fonction du composant sur lequel le circuit sera implémenté, spécialement le logiciel du fabricant du composant (ISE XILINX : Integrated Software Environment).
3. Une carte de développement pour implémenter le circuit et vérifier son comportement.

La figure III.16 montre le répertoire du projet crée sous ISE.

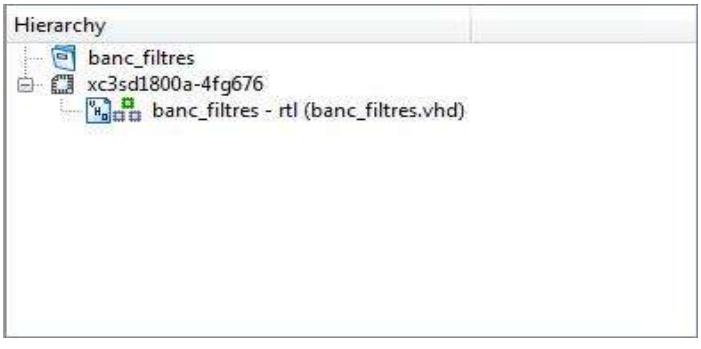


Figure III.16. *La vue « Hierarchy »*

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

Tableau III.2. *Sommaire des ressources utilisées dans le FPGA*

banc_filtres Project Status (06/01/2016 - 13:10:10)			
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:**	35 Warnings (0 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	17	33,280	1%	
Number of 4 input LUTs	139	33,280	1%	
Number of occupied Slices	118	16,640	1%	
Number of Slices containing only related logic	118	118	100%	
Number of Slices containing unrelated logic	0	118	0%	
Total Number of 4 input LUTs	223	33,280	1%	
Number used as logic	139			
Number used as a route-thru	84			
Number of bonded IOBs	155	519	29%	
Number of BUFGMUXs	1	24	4%	

banc_filtres_2016-6-1-14-4-42.twx | Design Summary

Tableau III.3. *Sommaire des ressources utilisées dans le FPGA (suite)*

Number of DSP48As	8	84	9%	
Average Fanout of Non-Clock Nets	1.61			

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	mer. 1. juin 14:21:31 2016	0	35 Warnings (0 new)	4 Infos (0 new)
Translation Report	Current	mer. 1. juin 14:21:46 2016	0	0	0
Map Report	Current	mer. 1. juin 14:21:56 2016	0	0	1 Info (0 new)
Place and Route Report	Current	mer. 1. juin 14:24:11 2016	0	0	2 Infos (0 new)
Power Report					
Post-PAR Static Timing Report	Current	mer. 1. juin 14:24:19 2016	0	0	6 Infos (0 new)
Bitgen Report					

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

banc_filtres_2016-6-1-14-4-42.twx | Design Summary

Après avoir synthétisé le programme et généré le fichier (.bit) nous procédons à la programmation du FPGA. La figure suivante nous démontre la synthétisation, l'implémentation et la programmation du FPGA.

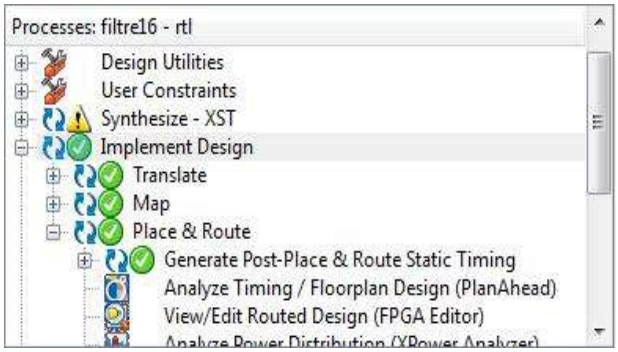


Figure III.17. *Processus de l'implémentation*

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

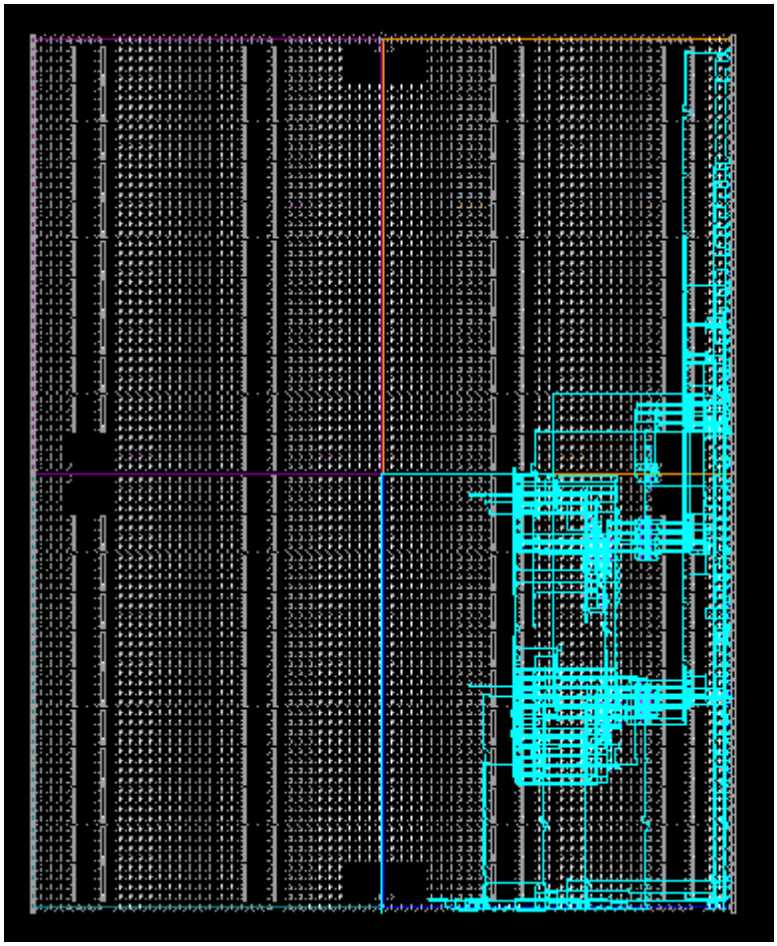


Figure III.18. *Circuit de FPGA global*

Les différentes slices utilisées dans le circuit réalisé sont présentées dans la figure III.19 :

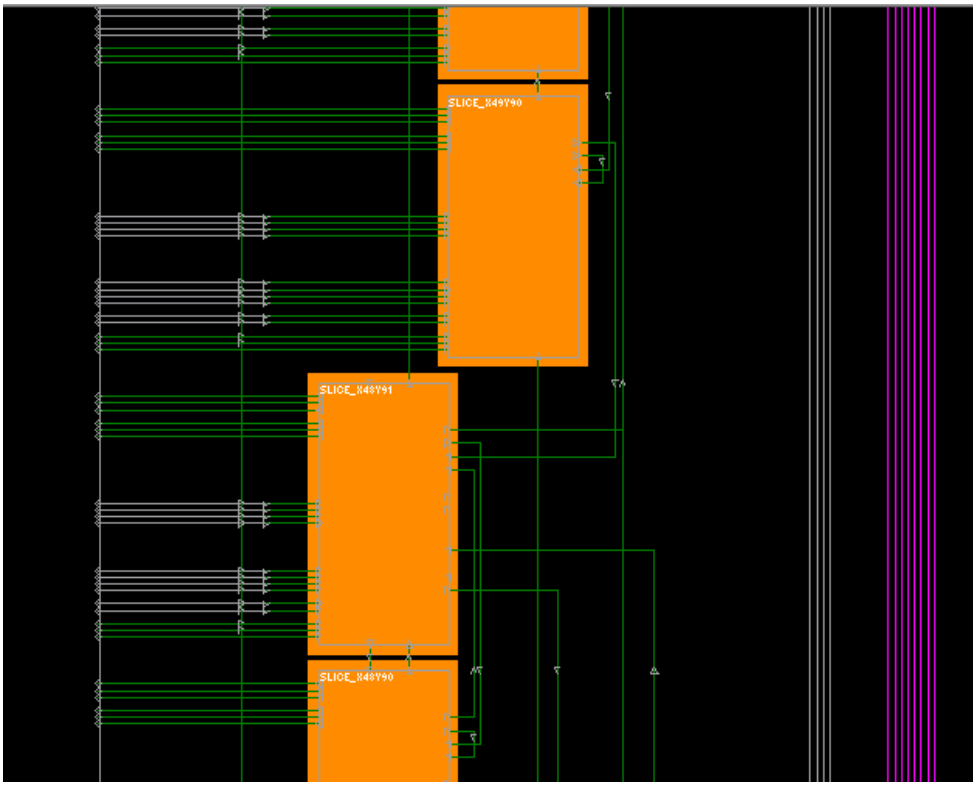


Figure III.19. *Les différentes slices sur FPGA (une partie zoomée)*

La figure III.20 représente le circuit interne du slice « SLICE_X49Y90 » conçu en utilisant les bascules et les portes logiques:

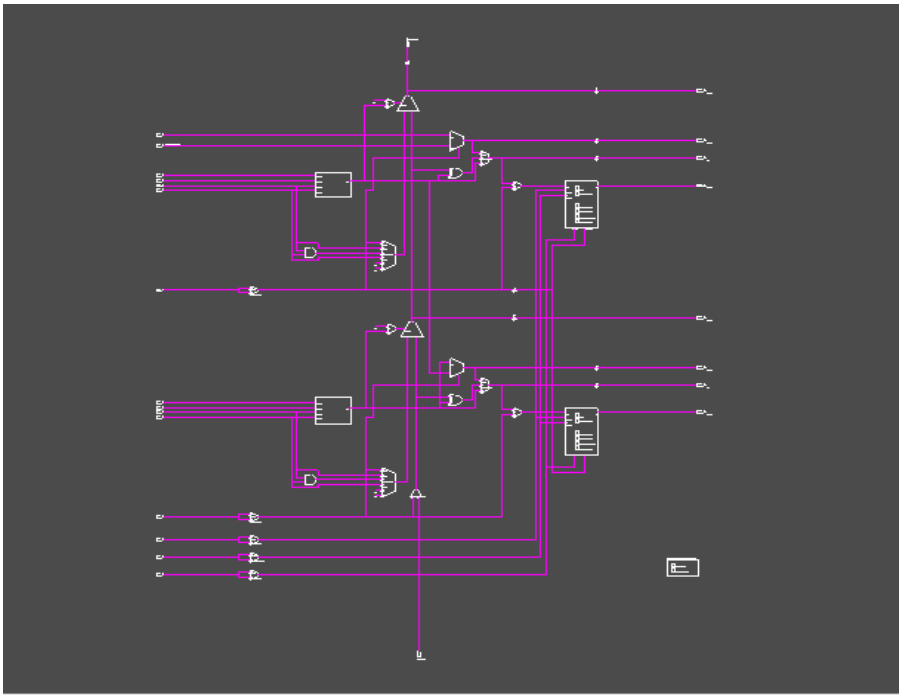


Figure III.20. *Le circuit interne du « SLICE_X49Y90 »*

Une partie zoomée de la figure précédente est montrée dans la figure III.21 :

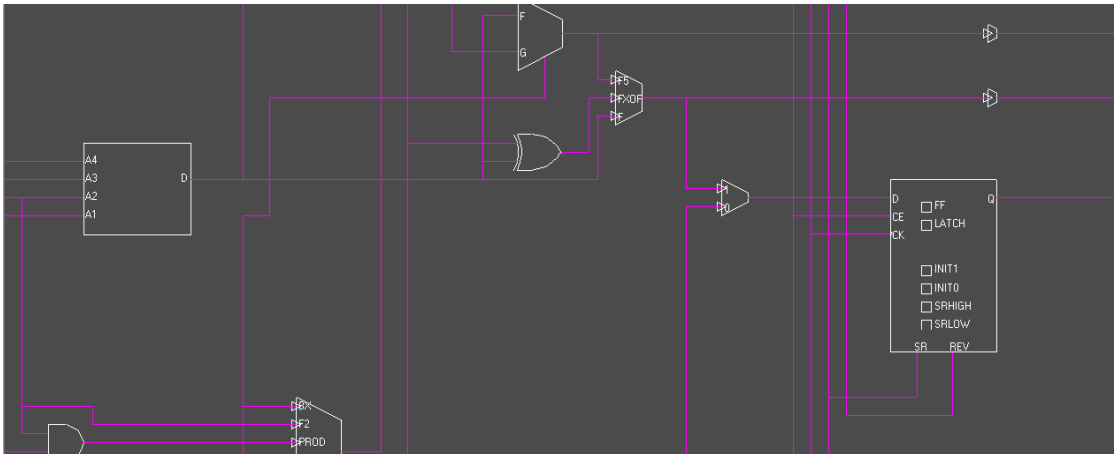


Figure III.21. *Une partie zoomée du « SLICE_X49Y90 »*

11. Conclusion :

Dans ce chapitre, nous avons présenté la méthodologie de l'implémentation du banc de filtres simulant une cochlée pathologique de l'être humain sur un circuit FPGA. La simulation en Modelsim des codes VHDL réalisés sous Matlab donne des bonnes résultats en les comparant avec les résultats obtenus par simulation Matlab.

CONCLUSION GENERALE

Le présent travail avait pour but l'implémentation sur un circuit FPGA d'une cochlée artificielle en utilisant des filtres numériques de type passe-bande.

Dans notre travail, nous avons vu que les implants cochléaires ont une importance très importante chez les personnes sourds, comme ils ont des inconvénients, aussi nous avons cité les stratégies de stimulation cochléaires qui sont basées sur le banc de filtres en détaillant la stratégie CIS et en précisant son fonctionnement et le banc de filtres utilisés dans notre étude.

La réalisation de ce projet nous a imposé d'atteindre différents objectifs tels que : La connaissance des circuits logiques programmables et leur configuration, la simulation d'un banc de filtres d'une stratégie précise, la conception d'une application et son implémentation dans FPGA et la génération du code VHDL et son utilité sous XILINX ISE.

La méthodologie adoptée a eu des résultats considérables comme étant concluants. La plateforme utilisée après sa mise au point pourrait constituer un outil de test temps-réel appréciable.

Les résultats de notre étude nous permettent de conclure que la simulation dans Matlab et Modelsim a été réussite ce qui nous laisse passer à la dernière partie concernant l'implémentation sur FPGA sous XILINX ISE.

Finalement, nous pouvons dire que notre but de cette réalisation pour terminer l'implémentation en succès a été achevé.

Bibliographie

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

[2] Chouard CH, Josset P, Meyer B, Buche JF. Ann Otolaryngol Chir Cervicofac, "Implant cochléaire", Richard T. Miyamoto (dir.), 2015. [En ligne]. Disponible : http: // www.bium.univ-paris5.fr/histmed/medica/orld.htm.

[3] 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.

[4] Rémy Pujol Frédéric Venail, "Surdité : généralités", *La surdité*, 2016. [En ligne]. Disponible : http://www.cochlea.eu/pathologie. [Consulté le 3 janvier 2016].

[5] Haute Autorité de Santé, *TRAITEMENT DE LA SURDITÉ PAR POSE D'IMPLANTS COCHLÉAIRES OU D'IMPLANTS DU TRONC CÉRÉBRAL*, Classement CCAM : 03.04.02 – Codes : CCGA001, CDLA002, CDLA003, CDMP002, France, mai 2007.

[6] Diane Lazard, Frédéric Venail, Antoine Lorenzi, "Implants cochléaires", *Principe des implants cochléaires*, 2013. [En ligne]. Disponible : http://www.cochlea.org/quels-traitements-aujourd-hui/implants-cochleaires.

[7] neurelec See.Feel.Hear.Live, "comment fonctionne l'implant cochléaire", *Le système d'implant*, 2013. [En ligne]. Disponible : http://www.neurelec.com/fr/implant-cochleaire.

[8] Peter Merian, *Implants cochléaires Nucleus, informations importantes*, Switzerland, 2010.

[9] Angel de la Torre Vega, Marta Bastarrica Martí, Rafael de la Torre Vega, Manuel Sainz Quevedo, "Description et utilisation du programme", in *Cochlear Implant Simulation version 2.0*, Grenade (Espagne), Décembre 2004. [En ligne].

[10] Mr. Y. Srinivas, Mr. P. Darwin, Dr. V. Sailja, "Continuous Interleaved Sampled (CIS) Signal Processing Strategy for Cochlear Implants MATLAB Simulation Program", *International Journal of Scientific & Engineering Research*, Vol. 3, Issue 12, pp. 1-7, December-2012.

[11] Komal Arora, Richard Dowell and Pam Dawson, "Cochlear Implant Stimulation Rates and Speech Perception", in *Modern Speech Recognition Approaches with Case Studies*, 2007. [En ligne]. Disponible : http://dx.doi.org/10.5772/49992.

[12] Branko Somek, Sinisa Fajt, Ana Dembitz, Mladen Ivkovic, Jasmina Ostojic, *Coding Strategies for Cochlear Implants*, ISSN 0005–1144, ATKAAF 47(1–2), 69–74 (2006). [Livre électronique]. AUTOMATIKA 47(2006) 1–2, 69–74.

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

[14] "Filtre passe-bande", *licence Creative Commons attribution*. [Livre électronique]. 501 ©(3) Etats-Unis.

[15] Ashock Ambardar, "Filtre à réponse impulsionnelle finie", *Analog and Digital Signal Processing*, Brooks/Cole, 1999**.**

[16] J. Prado, *Bancs de filtres*. [Livre électronique]. 8 juin 2006.

[17] J. Prado, *Introduction aux bancs de filtres*. [Livre électronique]. 25 novembre 2008.

[18] Ashock Ambardar, "Filtre à réponse impulsionnelle infinie", *Analog and Digital Signal Processing*, Brooks/Cole, 1999**.**

[19] "Mathématique pour le signal, Projet en Filtrage Numérique avec Simulation Matlab", mémoire de Master en Systèmes de Communication & Informatique, Université Abdelmalek Essaadi, Département de Télécoms, Ensa de Tanger, Maroc, mars 2013. [En ligne].

[20] Abdelhakim Sahour, "Implémentation sur FPGA d'un algorithme de débruitage", Département d'électronique BADJI MOKHTAR-ANNABA UNIVERSITY, 2013.

[21] SIFI Med Lamine, KELILI Walid, "Interfaçage d'un circuit FPGA pour l'implantation d'algorithmes de traitement de la parole", Université des Sciences et de la Technologie Houari Boumediene, Alger, juin 2015.