

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE  
Ministère de l'Enseignement supérieur et de la Recherche Scientifique  
Ecole Nationale Polytechnique



Département d'Electronique

Projet de Fin d'études

Thème

Implémentation sur FPGA d'un filtre adapté à base du procédé de  
compression d'impulsion radar

Dirigé par :

M. Abdelaziz Mohamed EL Mounir

Pr. Adel Belouchrani

Etudié par :

Mahgoun Hichem

Promotion Juin 2008

E.N.P 10 Avenues Hassen BADI EL HARRACH-ALGER

## **Résumé :**

Le filtre adapté est un organe principal du récepteur Radar. Il permet d'extraire l'information noyée dans le bruit. Le but de ce travail est de l'implémenter sur un processeur de type FPGA.

L'étude des performances du filtre adapté passe par deux étapes importantes :

- L'étude des formes d'onde utilisées au niveau de l'émetteur
- L'étude de la réponse du filtre adapté pour chaque forme d'onde.

Le travail réalisé dans ce projet se partage en deux parties :

- La réalisation d'une interface de simulation au niveau de MATLAB qui simule la réponse du filtre
- L'implémentation du filtre adapté sur FPGA.

Mots clés : Radar, FPGA, filtre adapté, compression d'impulsion, forme d'onde.

## **Abstract:**

The matched filter is a main device in a radar receiver, it allows the extraction of information from signal corrupted by noise, the main objective of this work is to implement on a FPGA processor.

The study of this kind of filter performance considers two steps

- The study of common waveforms used by Radar transmitter
- The study of the output of the matched filter

In the present work, we have performed two important tasks

- The realization of an user interface which allows the simulation of the matched filter for different setting
- An implementation of the matched filter on a FPGA processor

Keys words: Radar, FPGA, matched filter, waveforms, compression

## موجز

- المرشح المكيف جهاز رئيسي من أجهزة الاستقبال الرادار عمله استخلاص المعلومات في وجود التشويش الهدف من هذا العمل انجاز المرشح المكيف على معالج من نوع FPGA.
- دراسة عمل المرشح المكيف تتطلب خطوتان
- دراسة أشكال الموجات المستخدمة في الإرسال
  - دراسة استجابة المرشح المكيف لكل شكل من أشكال الموجات المستخدمة في الإرسال الأعمال المنجزة في إطار هذا المشروع تنقسم إلى قسمين
  - دراسة استجابة المرشح المكيف باستعمال ماتلب
  - انجاز المرشح المكيف على معالج من نوع FPGA
- الكلمات المفتاحية: المرشح المكيف, FPGA الرادار الموجات

---

Liste des Figures .....	5
Liste des tableaux.....	8
Liste des symboles .....	9
<b>Introduction générale .....</b>	<b>10</b>
<b>Chapitre 1 : Filtre adapté.....</b>	<b>13</b>
1.1 Radar à état solide .....	14
1.2 Modèle du filtre adapté .....	15
1.3 Lobes secondaires.....	17
1.4 Conclusion.....	18
<b>Chapitre 2 : Fonction d'ambiguïté.....</b>	<b>19</b>
2.1 Introduction .....	20
2.2 Forme d'onde usuelle .....	22
2.3 Insensibilité au Doppler .....	23
2.4 Conclusion.....	24
<b>Chapitre 3 : Compression d'impulsion .....</b>	<b>25</b>
3.1 Introduction .....	26
3.2 Principe .....	26
3.3 Conclusion.....	27
<b>Chapitre 4: Simulation du filtre de compression.....</b>	<b>28</b>
4.1 Introduction .....	29
4.2 Réponse du filtre adapté pour un codage linéaire de fréquence.....	32

---

4.3	Codage non linéaire de fréquence de type 1 et 2.....	36
4.4	Réponse du filtre adapté pour codage linéaire de fréquence+ codage d'amplitude de type HAMMING .....	44
4.5	Réponse du filtre adapté pour un codage de type CHIRPLIKE.....	46
4.6	Réponse du filtre adapté pour un codage de fréquence de type COSTAS.....	52
4.7	Interface de simulation.....	56
4.8	Tableau récapitulatif.....	58
4.9	Conclusion.....	58
<b>Chapitre 5 : Implémentation sur circuit FPGA .....</b>		<b>59</b>
5.1	Introduction .....	60
5.2	Réponse du filtre adapté pour un codage linéaire de fréquence.....	65
5.3	Réponse du filtre adapté pour codage non linéaire de fréquence de type 1 .....	66
5.4	Réponse du filtre adapté pour codage non linéaire de fréquence de type 2 .....	67
5.5	Réponse du filtre adapté pour un codage linéaire de fréquence plus codage d'amplitude de type HAMMING .....	68
5.6	Réponse du filtre adapté pour un codage de type CHIRPLIKE.....	69
5.7	Réponse du filtre adapté pour un codage de type COSTAS .....	70
5.8	Tableau récapitulatif.....	73
<b>Conclusion générale .....</b>		<b>74</b>
<b>Références bibliographiques .....</b>		<b>75</b>

---

1.1	Schéma synoptique des constituants d'un radar à état solide.....	14
1.2	Représentation du filtrage adapté dans le domaine temporel.....	18
1.3	Représentation du filtrage adapté dans le domaine fréquentiel.....	18
2.1	Algorithme de calcul de la fonction d'ambiguïté.....	20
2.2	Algorithme de génération de formes d'onde.....	22
3.1	Procédé de compression d'impulsion .....	27
4.1	Type de codage dans l'interface .....	29
4.2	Type de simulation dans l'interface .....	30
4.3	Type de fonction réalisé par interface .....	31
4.4	Modèle mathématique d'un codage linéaire de fréquence .....	32
4.5	Réponse du filtre adapté pour une LFM.....	33
4.6	Simulation du retard pour une LFM .....	34
4.7	Séparation entre deux cibles pour une LFM.....	34
4.8	Fonction d'ambiguïté pour une LFM .....	35
4.9	Densité de puissance pour une LFM.....	36
4.10	Algorithme de génération d'un codage non linéaire de fréquence de type 1.....	36
4.11	Génération du codage non linéaire de fréquence de type 2 .....	37
4.12	Réponse du filtre adapté pour une NLFM de type 1.....	38
4.13	Fonction d'ambiguïté pour une NLFM de type 1 .....	39
4.14	Séparation entre deux cibles pour une NLFM de type 1 .....	39
4.15	Densité de puissance pour une NLFM de type 1 .....	40
4.16	Réponse du filtre adapté pour une NLFM de type 2.....	41
4.17	Fonction d'ambiguïté pour une NLFM de type 2 .....	42
4.18	Séparation entre deux cibles .....	43

4.19	Spectre d'une NLFM de type 2 .....	43
4.20	Réponse du filtre adapté pour une LFM avec un codage d'amplitude de type (HAMMING) .....	44
4.21	Fonction d'ambiguïté pour une LFM avec un codage d'amplitude de type (HAMMING) .....	45
4.22	Densité de puissance pour une LFM avec pondération de type HAMMING.....	45
4.23	Algorithme de génération des signaux de type CHIRPLIKE .....	46
4.24	Réponse du filtre adapté pour un codage de type P3 .....	47
4.25	Fonction d'ambiguïté pour un codage de type P3 .....	48
4.26	Représentation en deux dimensions de la fonction d'ambiguïté pour le codage P3 .....	49
4.27	Réponse du filtre adapté pour le codage de Frank .....	50
4.28	Fonction d'ambiguïté pour le codage de Frank .....	51
4.29	Séparation entre deux cibles .....	51
4.30	Algorithme de calcul du codage de type COSTAS .....	52
4.31	Réponse du filtre adapté pour un codage de type COSATS .....	52
4.32	Fonction d'ambiguïté pour un codage de type COSTAS .....	53
4.33	Séparation entre deux cibles pour un codage de type COSTAS.....	54
4.34	Représentation du plan temps fréquence pour un codage de type COSTAS.....	55
4.35	Interface de simulation .....	56
5.1	Schéma synoptique du filtre adapté .....	61
5.2	Composant qui réalise la fonction FFT .....	62
5.3	Circuit qui réalise le filtrage adapté .....	63
5.4	Paramètre de simulation au niveau de l'FPGA.....	64

5.5	Réponse du filtre adapté pour une LFM.....	65
5.6	Réponse du filtre adapté pour une LFM en dB.....	65
5.7	Réponse du filtre adapté pour une NLFM de type 1.....	66
5.8	Réponse du filtre adapté pour une NLFM de type 1 en dB.....	66
5.9	Réponse du filtre adapté pour une NLFM de type 2.....	67
5.10	Réponse du filtre adapté pour une NLFM de type 2 en dB.....	67
5.11	Réponse du filtre adapté pour une LFM+HAMMING.....	68
5.12	Réponse du filtre adapté pour une LFM+HAMMING en dB.....	68
5.13	Réponse du filtre adapté pour le code p3.....	69
5.14	Réponse du filtre adapté pour le code p3 en dB.....	69
5.15	Réponse du filtre adapté pour un codage de Frank.....	70
5.16	Réponse du filtre adapté pour un codage de Frank en dB.....	70
5.17	Réponse du filtre adapté pour un codage de COSATS.....	71
5.18	Réponse du filtre adapté pour un codage de COSATS en dB.....	71

4.1	Caractéristiques de la simulation sous MATLAB .....	32
4.2	Caractéristiques de la simulation pour une NLFM de type 1 .....	37
4.3	Constituants de l'interface de simulation .....	57
4.4	Résumé des différents résultats de simulations .....	58
5.1	Signaux d'entrée sortie pour le composant qui réalise la FFT .....	62
5.2	Contraintes temps réel.....	63
5.3	Paramètres de simulation au niveau de l'FPGA.....	64
5.4	Récapitulatif des résultats de la simulation sous FPGA .....	72

$T$	Durée de l'impulsion
$f_d$	L'écart Doppler
$s(t)$	Signal émis
$y(t)$	Signal reçu
$\eta(t)$	Bruit ajouté au signal émis
$\tau$	Le retard
$d$	Distance entre cible et station de réception
$g(t)$	Enveloppe du signal émis
$\varphi(t)$	Phase du signal émis
$g_c(t)$	Composante en phase du signal émis
$g_s(t)$	Composante en quadrature du signal émis
$u(t)$	Signal complexe
$h(t)$	Réponse impulsionnelle du filtre
$\chi(\tau, f_d)$	Fonction d'ambiguïté
$SNR$	Rapport signal sur bruit
$B$	La bande spectrale du signal
$\rho$	Rapport de compression
$\tau_{shift}$	Décalage temporel dû à la forme d'onde
$N$	La longueur du code
$f_m$	Le code fréquentiel au point m
$\tau_{1nul}$	Le premier zéro de la fonction d'auto-corrélation
$BLT$	Coefficient de compression du codage linéaire de fréquence
$BCT$	Coefficient de compression du codage non linéaire de fréquence
$L$	La racine carrée de la taille du code
$\varphi_m$	Le code implanté dans la phase du bit m
LFM	Codage linéaire de fréquence
NLFM type 1	Codage non linéaire de fréquence en utilisant la formule empirique
NLFM type 2	Codage non linéaire de fréquence en utilisant le principe de la phase stationnaire
$\tau_{1sl}$	Location du premier lobe secondaire
$n_{1sl}$	Niveau du premier lobe secondaire
$n_{xsl}$	Niveau du plus grand lobe secondaire
$\Gamma_{uu}(\tau)$	Fonction d'autocorrélation
$H(f)$	Le spectre de la de la réponse impulsionnelle du filtre
$Y(f)$	Le spectre du signal filtré
$K$	Pente de la variation linéaire de la fréquence
$R_{dis}$	Pouvoir de discrimination en distance

Au vu du développement fulgurant dans les différents domaines, le développement de la microélectronique a connu un sursaut conséquent motivé par une concurrence très rude pour répondre à un besoin pointu et accru dans des domaines aussi bien variés que spécifiques, tels que les télécommunications, les réseaux, les communications sans fil, le traitement du signal, le traitement des données...etc.

Pour satisfaire une telle demande, le développement a affecté principalement trois volets ayant trait à la technologie du semi-conducteur, la conception de circuits intégrés avec des architectures et des algorithmes complexes et enfin la conception assistée par ordinateur et cela grâce à la mise en œuvre d'outils logiciels d'aide à la conception de circuits intégrés ayant une forte intégration qui se chiffre avec un ordre de grandeur de quelque milliards de transistors.

Actuellement, le taux d'intégration a rendu l'appellation de circuit obsolète dans la mesure où on parle de système sur puce « System On Chip » dont la gestion n'est possible que par l'utilisation d'outils CAO dédiés qui permettent d'automatiser le processus de conception ou « Design » et ce depuis la formulation des spécifications du besoin jusqu'à l'obtention du masque « layout ». Ces outils d'exécution du processus de conception sont regroupés dans un ensemble ou « package » identifié par l'appellation de cycle de conception « design flow ».

Parmi les cycles de conception, nous trouvons le cycle de « conception FPGA » qui utilise des outils de conception avec des langages évolués qui offrent une aisance et une rapidité pour concrétiser les solutions aux problèmes et plus particulièrement dans le cas où un prototypage est exigé.

C'est à partir de cette approche de prototypage que les travaux inscrits dans le cadre de ce projet de fin d'étude ont été exécutés pour implémenter sur circuit FPGA, de la famille Virtex, un filtre adapté pour un récepteur radar à base du procédé de compression d'impulsions.

L'objectif, visé à travers ce travail, consiste à se prononcer à partir des spécifications radar sur les performances du filtre de compression en termes de séparation de cible, de niveaux des lobes secondaires du filtre adapté et d'insensibilité au Doppler à partir de deux classes de forme d'ondes, à savoir les formes d'ondes à codage de fréquence et les formes d'ondes à codage de phase.

Le profil des travaux exécutés dans ce cadre revêt deux volets. Ces volets sont liés aux outils de travail avec l'environnement de programmations et de simulation du filtre de compression afin de générer des données brutes et ce pour générer des fichiers data d'entrée pour la conception FPGA et comparer ses résultats d'implémentation avec ceux obtenus avec la simulation Matlab.

Ce mémoire se compose de cinq chapitres.

Le premier chapitre décrit brièvement le radar à état solide et traite le filtre adapté en termes de rapport signal sur bruit, les modèles mathématiques du filtre dans le domaine fréquentiel et temporel ainsi que les lobes secondaires de la réponse du filtre.

Le second chapitre traite de la fonction d'ambiguïté comme outil de caractérisation temps-fréquence et plus particulièrement voir ce qui se rapporte à l'effet du Doppler sur les performances du filtre adapté.

Le troisième chapitre décrit la mise en œuvre du filtre adapté d'un récepteur radar à état solide en utilisant le procédé de compression d'impulsions avec deux classes de formes d'ondes : le codage de fréquence et le codage de phase.

Le quatrième chapitre décrit la simulation sous environnement Matlab du filtre de compression avec la fonction d'ambiguïté de la forme d'onde utilisée.

Dans le cinquième chapitre, nous donnerons les résultats d'implémentation sur circuits FPGA de la famille Xilinx avec les ressources FPGA ainsi qu'une évaluation des contraintes temps liées à notre conception.

Enfin, nous clôturons avec une conclusion générale pour résumer les résultats obtenus ainsi que les perspectives de poursuite de ce travail.

## Filtre adapté

## 1.1 Radar à état solide

---

Depuis longtemps l'être humain cherche à copier ses actions et les rendre de plus performantes. La vision ne fait pas défaut à cette règle. Si on pouvait voir à des kilomètres de nous, de nombreux problèmes seraient réglés. Ce rêve resta dans la tête des hommes jusqu'à ce que la technologie le permette. Avec la découverte des ondes électromagnétiques et de leurs applications, on a pu réaliser les premiers radars.

**RADAR** est un acronyme qui associe pour : **R**adio **D**etection **A**nd **R**anging

Donc le radar est un dispositif qui nous permet de détecter la position de la cible à une distance donnée. Pour réaliser cette fonction on doit envoyer une impulsion vers la cible et attendre l'écho. Le signal reçu est traité et on extrait plusieurs paramètres concernant la cible à savoir sa distance radiale, sa vitesse radiale, son angle de site ou son élévation ainsi que son angle de gisement ou sa direction [10].

Le signal informationnel est pris en charge par le biais de deux systèmes radiofréquences. Le premier système d'émission propage l'onde électromagnétique dans l'espace libre par le biais du dispositif d'antenne (aérien et duplexeur) tandis que le système de réception capte l'onde réfléchiée à travers le dispositif d'antenne.

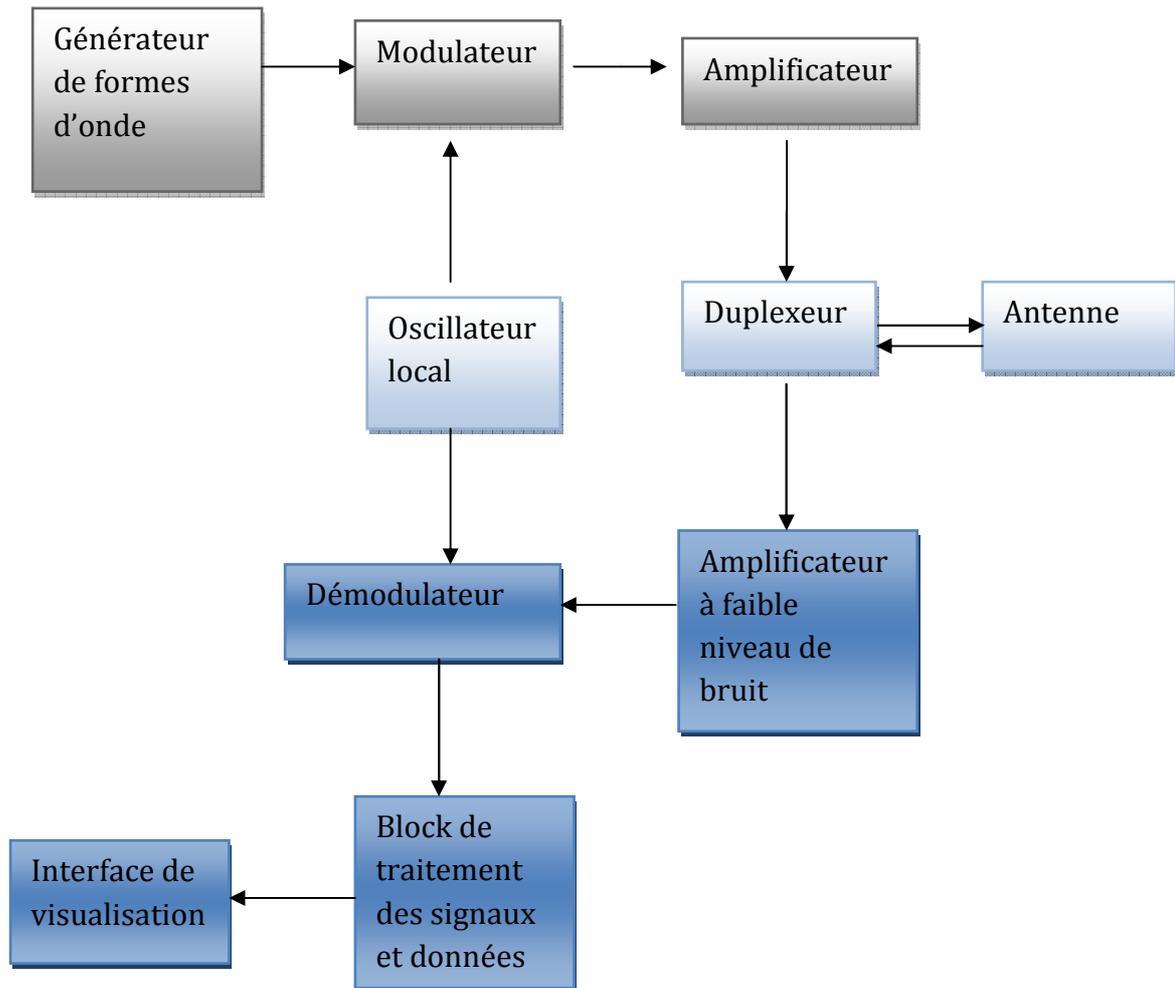
Trois principaux organes caractérisent les systèmes d'émission. Il s'agit du générateur de forme d'onde, du mélangeur à battement supérieur et de l'amplificateur à haute puissance. La forme d'onde est convertie dans le domaine radiofréquence par un ou plusieurs mélangeurs opérant avec une fréquence fixe ou variable de l'oscillateur local pour être amplifiée dans l'amplificateur de puissance.

La fonction du récepteur radar est de :

- Détecter les échos utiles en présence d'échos indésirables de bruit, de clutter et d'interférences.
- Séparer les échos utiles des échos indésirables.
- Amplifier les échos utiles selon une dynamique qui est dictée par les traitements en aval.

Le signal reçu dépend de la forme d'onde émise, de la nature des cibles ainsi que des caractéristiques du bruit, du clutter et des interférences. Un des paramètres déterminant qui permet de se prononcer sur ses performances en présence de signaux indésirables est le rapport signal sur bruit.

Le système de réception comprend principalement un amplificateur à faible bruit ou « Low Noise Amplifier », un mélangeur à battement inférieur et un convertisseur analogique/numérique s'il y'a nécessité de numériser à partir des étages radiofréquences ou de fréquence intermédiaire. Sur la figure 1.1 on visualise les constituants du système radar.



**Figure 1.1 Schéma synoptique des constituants d'un Radar**

## 1.2 Modèle du filtre adapté

---

L'utilisation de la technique du filtre adapté dans les récepteurs radar permet la maximisation de la valeur crête du rapport signal sur bruit (SNR) du signal composite reçu (bruit blanc Gaussien et signal utile). La transformation du filtre adapté est convenable pour obtenir une décision de détection optimale (la présence de cible ou non), ou estimation des paramètres de la cibles (distance, vitesse, etc...), et ce avec un minimum d'erreur. Enfin il faut disposer d'une résolution adéquate dans un scénario à cibles multiples.

La conception du filtre adapté peut être accomplie dans les domaines, temporel ou fréquentiel par le biais d'une transformation de Fourier ou une convolution [radar handbook skolnik] dans la mesure où ces deux opérations sont mathématiquement équivalentes.

Généralement, une réalisation numérique du filtre adapté est fortement recommandée et tend à utiliser principalement une transformation de Fourier qui permet un nombre d'opérations réduit comparativement à la convolution, dans la mesure où la charge de calcul suit en fonction du nombre d'échantillon une loi logarithmique dans le cas de la transformation de Fourier et une loi linéaire dans le cas de la convolution.

Les signaux en bande de base peuvent être écrits sous la forme :

$$g_c(t) = g(t) \cos(\varphi(t)) \quad 1.1$$

$$g_s(t) = g(t) \sin(\varphi(t)) \quad 1.2$$

Avec  $g(t)$  : L'enveloppe du signal et  $\varphi(t)$  la phase du signal

Le fait que le signal soit complexe il nous permet de réduire la bande de fréquence au niveau du récepteur d'un facteur de deux, la limite de NYQUIST devient B au lieu de 2B dans le cas des signaux réels.

$$u(t) = g_c(t) + jg_s(t) \quad 1.3$$

Avec  $u(t)$  : Le signal complexe

$g_c(t)$  : Composante en phase

$g_s(t)$  : Composante en quadrature

Le signal émis est écrit en fonction de sa représentation en bande de base :

$$s(t) = \text{Re}(u(t)e^{j\omega t}) \quad 1.4$$

$$s(t) = \frac{1}{2}(u(t)e^{j\omega t} + u(t)^* e^{-j\omega t}) \quad 1.5$$

Avec  $\omega$  : la porteuse haute fréquence

L'écho comporte l'information sur la distance qui sépare le radar de la cible, cette information est reçue en présence d'un bruit, le filtrage adapté limite le bruit à une bande B là où le signal réside. Dans ce cas la bande spectrale du bruit passe d'une valeur infinie vers une valeur finie B, le choix de la bande B doit maximiser le rapport signal sur bruit au point de détection [1],[9],[10].

La réponse impulsionnelle du filtre adapté suit l'équation suivante :

$$h(t) = u^*(t_0 - t) \quad 1.6$$

Où  $u(t)$  est le signal informationnel émis

Le temps  $t_0$  est important si on veut que le filtre soit causal. Le filtre qui présente une réponse dépendante du temps négative n'est pas implémenté en temps réel, c'est la raison pour laquelle on ajoute un retard pour rendre le filtre causal. En général  $t_0$  est la durée du signal, on peut réaliser l'opération de filtrage pour  $t_0=0$  en temps différé.

On représente la réponse du filtre adapté par la fonction d'auto-corrélation

$$\Gamma_{uu}(\tau) = \int_{-\infty}^{+\infty} u(t)u^*(t - \tau)dt \quad 1.7$$

Pour l'implémentation du filtre on travaille dans le domaine fréquentiel. La première étape est de générer les coefficients du filtre :

$$H(f) = U^*(f) = TF(u^*(-t)) \quad 1.8$$

Le signal de sortie du filtre dans le domaine fréquentiel est donné par :

$$Y(f) = U(f)H(f) \quad 1.9$$

L'équation 1.9 définit le modèle de filtrage dans le domaine fréquentiel

### 1.3 Lobes secondaire

Précédemment on a dit que le filtre adapté se limite à l'implémentation d'un corrélateur au niveau du récepteur.

La réponse du filtre dépend uniquement de la forme d'onde utilisée au niveau de l'émetteur. En général pour des composants à état solide on est limité en puissance crête, [2] ; d'où il y a nécessité d'utilisation des signaux longs codés en fréquence pour avoir une bonne résolution en distance.

Pour apprécier l'effet du codage il faut calculer la fonction d'auto-corrélation. Pour mettre en évidence ces effets, on considère le cas d'un codage linéaire de fréquence.

$$\text{Soit le signal } u(t) = \exp(j2\pi ft) \quad 1.10$$

$$f = kt \quad 1.11$$

$$k = \frac{B}{T} \quad 1.12$$

Avec B : bande de fréquence du codage et T la durée du signal, k : la pente de la droite.

On utilise un signal complexe dont la partie réelle définit la composante en phase, et la partie imaginaire définit la composante en quadrature [1]. Le signal émis dure T et occupe une bande B.

La fonction de corrélation de ce signal présente une allure qui suit un sinus cardinal.

$$|\Gamma_{uu}(\tau)| = \left| \left(1 - \frac{|\tau|}{T}\right) \frac{\sin(\pi B\tau(1 - \frac{|\tau|}{T}))}{\pi B\tau(1 - \frac{|\tau|}{T})} \right| \quad 1.13$$

La réponse du filtre adapté pour un codage linéaire de fréquence, donne un pic de détection  $\tau = 0$ , à cet instant la réponse est maximale.

Le problème qui surgit après la corrélation est la présence de plusieurs lobes secondaires, ce type de lobe peut fausser la détection.

L'équation 1.14 définit la position du premier zéro :

$$\tau = \frac{1}{B} \quad 1.14$$

Ensuite on trouve après plusieurs lobes secondaires dans l'intervalle  $[-T, T]$ , l'équation 1.15 définit la position du premier lobe secondaire :

$$\tau_{1sl} \approx \frac{1.5}{B} \quad 1.15$$

#### 1.4 Conclusion

Dans cette partie du travail on a défini la réponse impulsionnelle du filtre adapté. Cette réponse dépend du signal émis. La figure 1.2 définit le modèle du filtrage dans le domaine temporel et la figure 1.3 définit le modèle de filtrage dans le domaine fréquentiel :

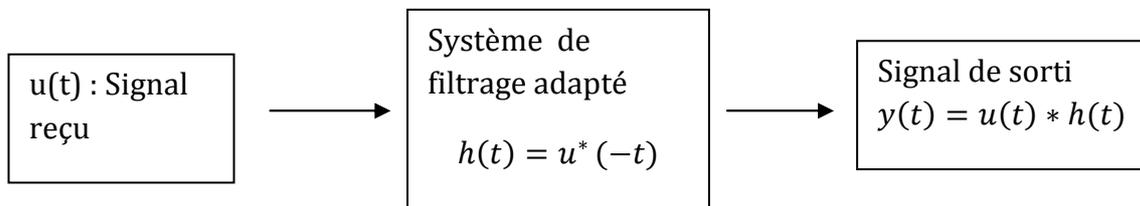


Figure 1.2 Représentation du filtrage adapté dans le domaine temporel

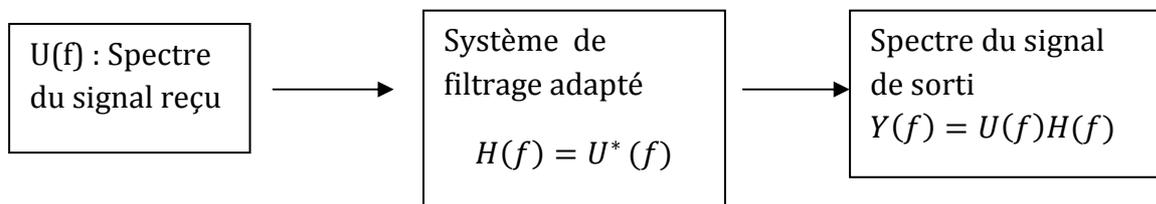


Figure 1.3 : Représentation du filtrage adapté dans le domaine fréquentiel

## Fonction d'ambiguïté

## 2.1 Introduction

L'analyse par la fonction d'ambiguïté est importante pour comprendre les propriétés des formes d'ondes utilisées et d'apprécier leurs effets sur l'efficacité des mesures, la séparation de cibles, l'ambiguïté en distance et la vitesse radiale. Sachant que la réponse du filtre adapté dépend du signal émis, par conséquent la fonction d'ambiguïté est un outil de caractérisation du filtre adapté du point de vu retard  $\tau$  et décalage Doppler  $f_d$ .

L'effet de la désadaptation au niveau du récepteur est étudié par la fonction d'ambiguïté. Elle est définie comme étant la réponse du filtre adapté en fonction de l'écart Doppler de la cible. Cette fonction est un outil mathématique puissant qui permet de faire une classification des formes d'onde selon leurs réponses temps fréquence. Elle permet aussi de mesurer le niveau de la désadaptation des signaux décalés en Doppler [1], [2],[9],[10]. La fonction d'ambiguïté est présentée par l'équation 2.1 et se calcule pour un signal  $u(t)$  donné :

$$|\chi(\tau, f_d)| = \left| \int_{-\infty}^{+\infty} u(t)u^*(t - \tau)e^{j2\pi f_d t} dt \right| \quad 2.1$$

Le schéma suivant illustre les étapes de calcul de la fonction d'ambiguïté.

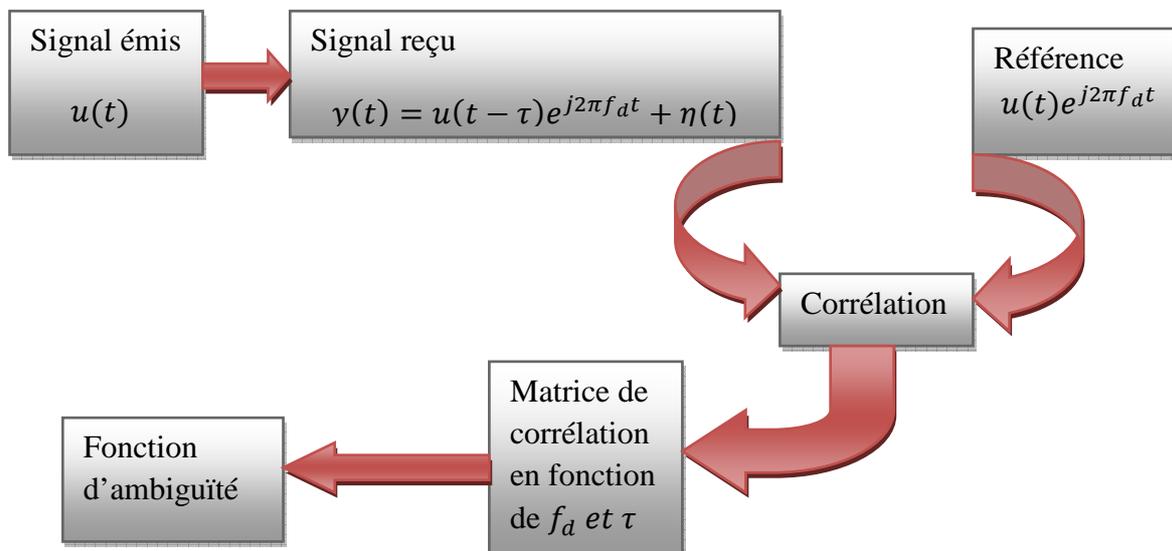


Figure 2.1 Schéma synoptique du calcul de la fonction d'ambiguïté

Quand le signal reçu est identique à la référence, la réponse de la fonction d'ambiguïté coïncide avec la fonction d'auto-corrélation.

$$|\chi(\tau, 0)| = \left| \int_{-\infty}^{+\infty} u(t)u^*(t - \tau)dt \right| \quad 2.2$$

La fonction d'ambiguïté présente une valeur maximale à l'origine, cette valeur donne le point max du SNR, l'équation 2.3 illustre cette définition :

$$|\chi(\tau, f_d)| \leq |\chi(0,0)| = 1 \quad 2.3$$

La fonction d'ambiguïté présente un volume constant. Par cette remarque on peut déduire que la réponse du filtre adapté est fonction de ce volume. Le travail à faire est de pouvoir répartir ce volume d'une manière appropriée [1], pour avoir la réponse voulue, l'équation 2.4 illustre cette définition :

$$\iint_{-\infty}^{+\infty} |\chi(\tau, f_d)|^2 d\tau df_d = 1 \quad 2.4$$

Pour un retard nul ( $\tau = 0$ ), la fonction d'ambiguïté calcule la transformée de Fourier de l'enveloppe du signal, l'équation 2.5 illustre cette définition :

$$|\chi(0, f_d)| = \left| \int_{-\infty}^{+\infty} |u(t)|^2 e^{j2\pi f_d t} dt \right| \quad 2.5$$

La contrainte qui se pose est qu'on ne peut pas définir une fonction d'ambiguïté et trouver le signal qui lui correspond, elle n'a pas de transformée inverse. En conséquence la problématique consiste à construire un signal et valider ses propriétés par le biais de sa fonction d'ambiguïté.

## 2.2 Formes d'ondes

A travers l'étude des formes d'ondes, le travail présenté cerne d'une part le filtrage adapté et d'autre part à l'appréciation de l'insensibilité au phénomène Doppler pour chaque forme d'onde présentée.

Cette partie aborde deux types de codage de forme d'onde. Certains avec un codage de phase et d'autres avec un codage de fréquence. Leur génération est illustrée par l'organigramme suivant :

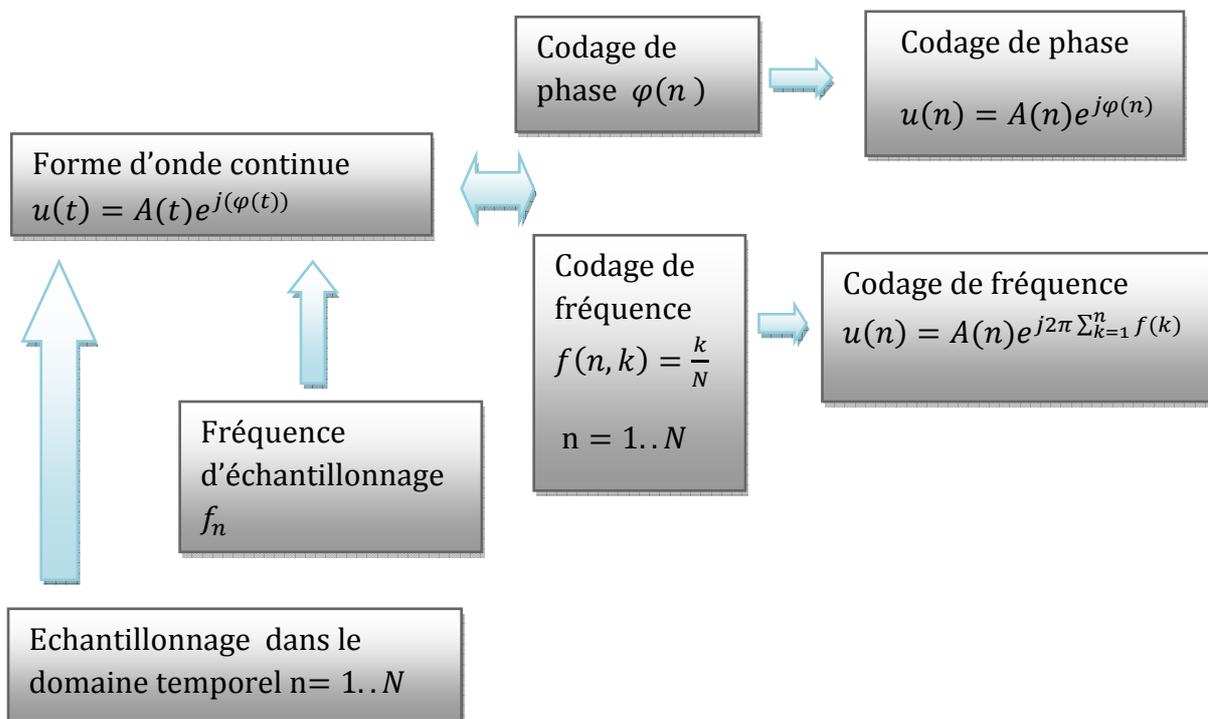


Figure 2.2 Algorithme de génération de formes d'ondes

La technique de génération est simple [1], [2], le plus important est de trouver une forme d'onde qui présente une fonction d'ambiguïté avec des caractéristiques intéressantes.

### 2.3 Insensibilité au Doppler

Pour illustrer analytiquement le concept d'insensibilité au décalage Doppler, on prend une forme d'onde de type CHIRP.

Pour ce type de forme d'onde on peut calculer analytiquement la fonction d'ambiguïté 2.6, et par conséquent on peut mesurer la tolérance au décalage Doppler du filtre adapté.

$$|\chi(\tau, f_d)| = \left| \left(1 - \frac{|\tau|}{T}\right) \frac{\sin(\pi T(f_d + k\tau)(1 - \frac{|\tau|}{T}))}{\pi T(f_d + k\tau)(1 - \frac{|\tau|}{T})} \right| \quad 2.6$$

Les performances du filtre adapté sont maintenues si la phase du signal de référence est fidèlement reproduite au niveau du signal reçu. Le paramètre d'appréciation de cette fidélité est la tolérance au décalage Doppler. Si la cible présente un Doppler différent de celui de la référence, la corrélation donne toujours un pic au point de détection.

Le pic présente un niveau atténué par rapport à la valeur maximale de la fonction d'auto corrélation sans pour autant dégrader l'exploitation du signal reçu [1].

Pour un signal codé linéairement en fréquence, on a une relation linéaire entre le retard  $\tau$  et le Doppler  $f_d$  ; un décalage Doppler crée une réponse décalée en temps, l'équation 2.7 illustre cette définition :

$$\tau_{shift} = \frac{f_d}{k} \quad 2.7$$

On a un pic de détection décalé de  $\tau_{shift}$  dans le cas d'une cible qui présente un doppler  $f_d$ , avec une perte au niveau du SNR défini par l'équation 2.8 :

$$|\chi(\tau_{peak}, f_d)| = \left(1 - \frac{|f_d|}{B}\right) \quad 2.8$$

Dans le cas d'un codage linéaire de fréquence, le filtre est insensible au décalage Doppler dans une certaine bande de fréquence qui se limite théoriquement à  $\pm B$  et dans la pratique à  $\pm \frac{B}{10}$ .

On constate que pour  $|f_d| = B$  le pic de détection présente une valeur nulle :

$$|\chi(\tau_{peak}, f_d)| = 0 \quad 2.9$$

Application numérique pour une cible qui présente les caractéristiques suivantes : Doppler de la cible  $f_d = 2 \text{ kHz}$ ,  $k = 1.28 * 10^{10} \text{ Hz/s}$ , avec  $k$  pente de la variation linéaire de la fréquence, le retard dû au codage de fréquence est  $\tau_{shift} = 156 \text{ ns}$ .

Pour une cible qui se déplace à  $1500 \text{ m/s}$ , on mesure la distance radiale avec un décalage de  $23 \text{ mètres}$  par rapport à la vraie location.

Chaque forme d'onde qui présente un codage linéaire de la fréquence est insensible au décalage Doppler dans les limites de la bande de fréquence. On peut trouver beaucoup de formes d'onde avec la même caractéristique comme les codages de phase de type Frank, Chu, P3, P4, Px.

### 2.4 Conclusion

---

La fonction d'ambiguïté est un outil de mesure des performances du filtre adapté. Elle donne la réponse du filtre en considérant l'écart Doppler.

On peut tirer plusieurs caractéristiques de cette fonction :

- La réponse du filtre adapté pour un Doppler nul,
- Le niveau de la désadaptation lors du filtrage adapté,

## Compression d'impulsion

### 3.1 Introduction

---

Depuis l'existence des applications des techniques radars dans le domaine de la surveillance de l'espace, un intérêt particulier s'est concentré sur l'extension de la distance de détection des objets. Généralement, le besoin d'accroître la portée du radar s'est accompli avec des exigences qui sont en rapport avec l'aptitude des systèmes radars à assurer un minimum de séparation des cibles en distance.

Face à cette situation, les concepteurs des tubes électroniques avaient comme obligation de pousser davantage les limites de la puissance crête des tubes électroniques qui étaient en service et ce du fait que l'amélioration de la puissance avec des émissions à impulsions larges n'étaient pas requises et plus particulièrement dans le domaine de la surveillance de l'espace aérien. A cet effet l'utilisation des tubes électroniques à haute puissance crête s'est avérée inefficace du moment où la puissance moyenne demeurait inchangée.

### 3.2 Principe

---

Pour répondre au besoin d'améliorer la portée de détection moyennant une séparation appréciable des cibles ; une nouvelle technique a permis d'augmenter la puissance moyenne avec des impulsions larges en émission sans pour autant accroître la puissance crête et dégrader la résolution du radar.

Ceci est mis en œuvre par un procédé dit de compression d'impulsion. Il utilise toute l'énergie d'une impulsion longue codée en fréquence ou en phase.

On obtient la résolution représentant la durée d'un code qui matérialise une impulsion courte. Cette réponse est obtenue par filtrage adapté.

Contrairement à la technologie analogique de réalisation du procédé de compression qui ne permettait pas une flexibilité de configuration des paramètres de la forme d'onde émise et ce du fait que chaque configuration sollicitait des ressources à part. L'évolution de la technologie numérique a permis le développement de moyens plus flexibles pour réaliser une compression d'impulsion

pour les radars au sol dans un premier temps et les radars embarqués ultérieurement, et ce pour des contraintes d'encombrement, de poids et de puissance. Afin de satisfaire les exigences accrues en performances, une implémentation numérique programmable de la fonction de convolution du filtre de compression est nécessaire pour permettre aux ressources de traitement de répondre à la contrainte en temps réel du système radar.

Pour répondre à cette contrainte de temps réel, le débit (throughput) requis par le système de compression d'impulsion dépend du nombre d'opérations requises pour effectuer l'opération de convolution.

Cette opération s'effectue entre le signal reçu et une réplique du signal émis [1]. La réalisation de la compression d'impulsion dans le domaine temporel, par le biais d'une convolution, demande un nombre d'opération proportionnel à  $N^2$  alors que dans le domaine fréquentiel par transformation de Fourier rapide, le nombre d'opération est proportionnel à  $N \log_r N$ .

Avec  $N$  le nombre de points de l'impulsion émise et  $r$  le radix de l'opération FFT [16].

#### 3.3 Conclusion

---

Pour répondre aux contraintes de ressources hardware en fonction du nombre de point de l'impulsion émise, l'implémentation du procédé de compression dans le domaine fréquentiel est requise en fonction du nombre de points  $N$  utilisé dans notre travail (128 et 256).

## Simulation du filtre de compression

## 4.1 Introduction

Pour Simuler le fonctionnement du filtre de compression, il faut étudier les diverses formes d'ondes, leurs filtres adaptés et la fonction d'ambiguïté associée à ce code.

Toute la programmation des formes d'onde a été regroupée sous une interface graphique qui permet une plus grande facilité de simulation.

Pour générer les formes d'ondes, on doit choisir un type de codage du menu principal :

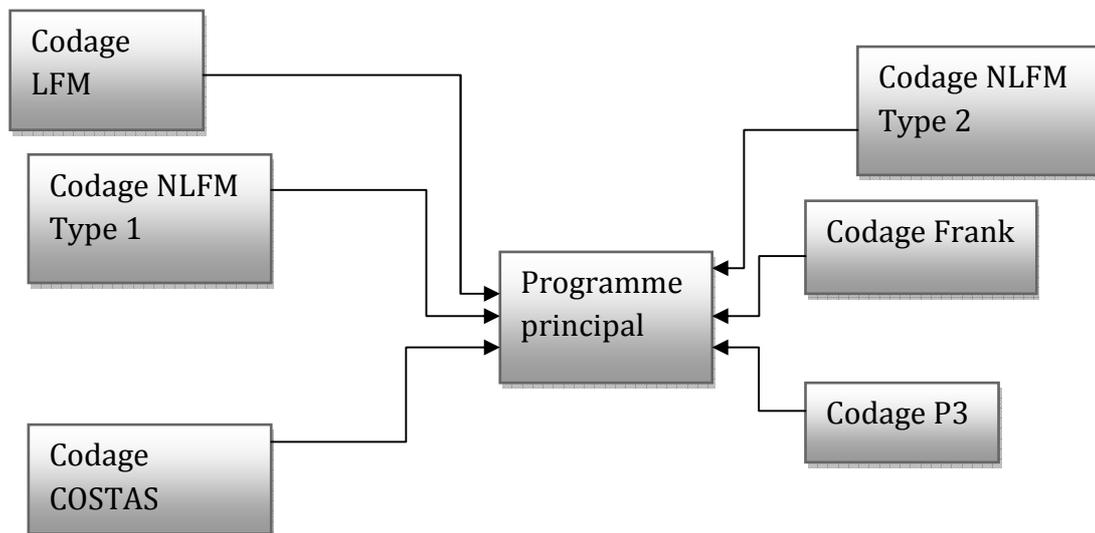


Figure 4.1 Formes d'ondes utilisées dans la simulation

Dans la figure 4.1 on présente deux classes de codage, la première classe porte sur le codage de fréquence tandis que la seconde sur le codage de phase. Les codes traités dans ces deux classes représentent les formes usuelles utilisées pour mettre en œuvre le procédé de compression d'impulsion dans un radar. A cet effet nous considérons les formes d'ondes suivantes :

### Classe à codage de fréquence :

- Codage linéaire de fréquence CHIRP ;
- Codage non linéaire de fréquence ;
- Codage de Costas.

Classe à codage de phase :

- Codage de type Frank ;
- Codage de type P3.

L'organigramme de simulation de la compression d'impulsion avec l'introduction de position d'échos de cibles avec les différents types de codes est illustré suivant le schéma de la figure 4.2.

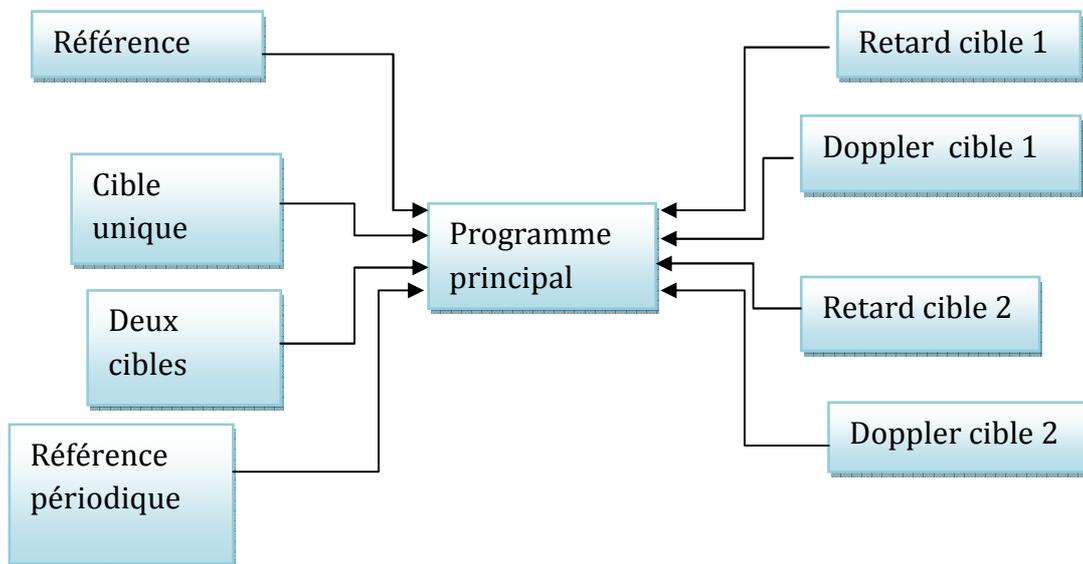


Figure 4.2 Types de simulation

Pour une simulation des caractéristiques des cibles en termes de position et de vitesse, l'interface peut générer un écho avec le retard et le Doppler.

Il y a deux scénarios de simulation, l'un avec une cible et l'autre avec deux cibles et ce pour apprécier la séparation après filtrage adapté.

Le diagramme suivant illustre les diverses fonctionnalités du simulateur :

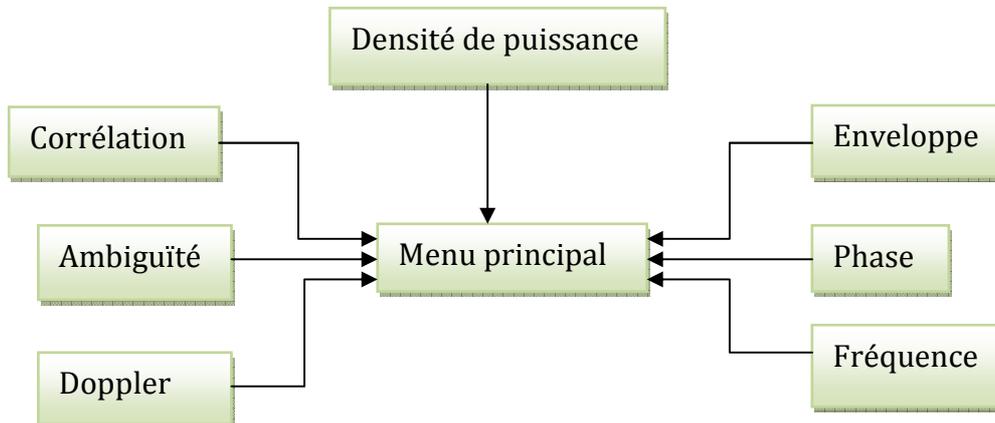


Figure 4.3 Fonctionnalité de l'interface de simulation.

L'interface de simulation permet le tracé de plusieurs fonctions nécessaires pour l'étude des performances de la forme d'onde. Les fonctions de base sont l'ambiguïté, la corrélation, la réponse Doppler pour un retard nul, la densité de puissance, l'enveloppe du signal, la phase et la fréquence.

Grâce aux fonctionnalités élaborées dans l'interface, on peut générer une forme d'onde, puis étudier ses caractéristiques dans la partie suivante, on réalise la simulation de quelques formes d'ondes, ensuite on extrait leurs caractéristiques du point de vue résolution en distance et lobes secondaires.

## 4.2 Réponse du filtre adapté pour un codage linéaire de fréquence

Pour générer un signal codé linéairement en fréquence, on doit affecter pour chaque échantillon une phase ou une fréquence décrite par le modèle suivant [1] :

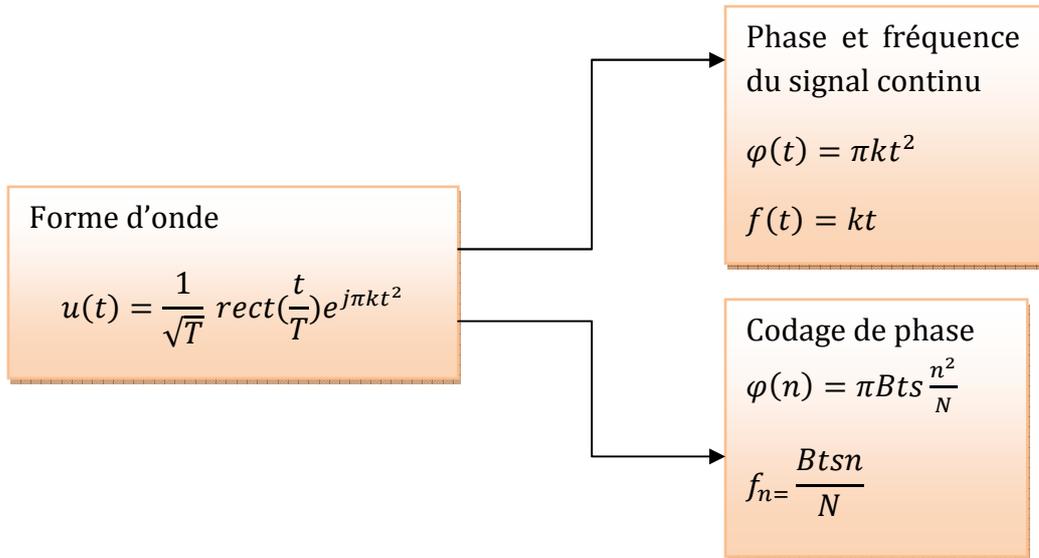


Figure 4.4 : Modèle mathématique d'un codage linéaire de fréquence

On prend les caractéristiques du tableau 4.1 Pour un codage linéaire de fréquence.

GRANDEUR	VALEUR
T : Durée du signal	100 us
fs : Fréquence d'échantillonnage	1.28 MHz
$\rho$ : coefficient de compression	$\rho = TB$ donc 128
B : occupation spectrale du signal	1.28 MHz
$\tau$ : résolution en distance	$\tau = \frac{T}{\rho}$ donc 0.78 $\mu$ s
x : Résolution en mètre	$x = c * \frac{\tau}{2}$ donc 117 m
d_1 : distance de la première cible	30 km
d_2 : distance de la deuxième cible	31 km
fd : tolérance au décalage Doppler	$\frac{B}{10}$ donc 128 kHz
$\tau_1$ : Retard de la cible 1	200 $\mu$ s
$\tau_2$ : Retard de la cible 2	206.66 $\mu$ s

Tableau 4.1 Caractéristiques de la simulation

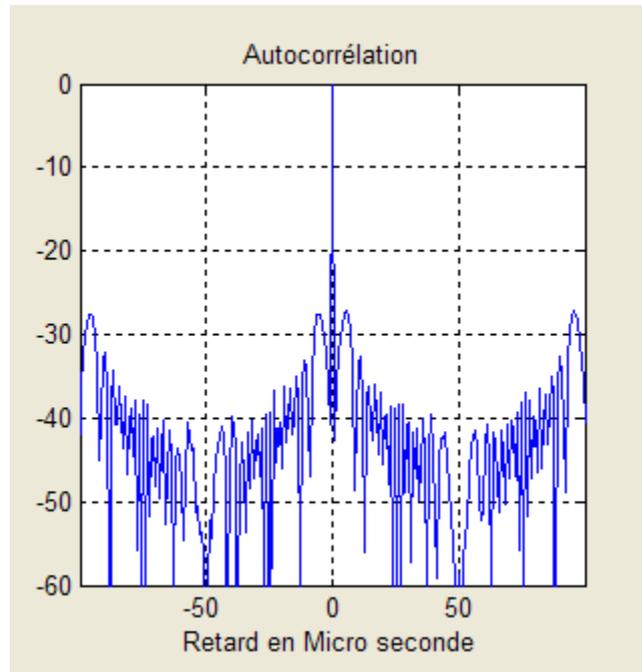


Figure 4.5 Réponse du filtre adapté pour une LFM

Sur la figure 4.5 on représente la réponse du filtre adapté en dB pour un codage linéaire de fréquence.

Caractéristiques de l'impulsion compressée :

Grandeur	Forme d'onde
Résolution en distance	$\tau_{1\text{ nul}} = 0.78 \mu\text{s}$
Premier lobe secondaire	$n_{\text{lobe } s1} = -27.4 \text{ dB}$
Le plus grand lobe	$n_{\text{lobe } sx} = -27.4 \text{ dB}$
Location du premier lobe	$\tau_{1\text{ sl}} = 5.5 \mu\text{s}$

Le premier zéro se trouve à  $\tau_{1\text{ nul}} = 0.78 \mu\text{s}$ . Le pouvoir de discrimination entre deux cibles est illustre par l'équation 4.1, avec c : vitesse de la lumière :

$$R_{dis} = \frac{c\tau_{1\text{ nul}}}{2} = 117 \text{ m} \quad 4.1$$

Grâce au codage de fréquence, on peut discriminer entre deux cibles à une distance minimale de 117 m.

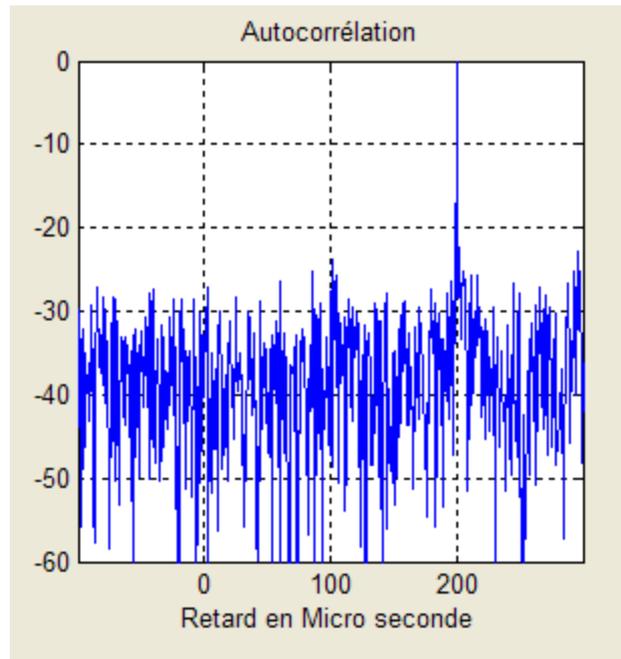


Figure 4.6 Simulation du retard

La figure 4.6 représente la réponse du filtre adapté dans le cas d'une cible qui se trouve à une distance  $d_1$  avec présence du bruit.

Pour mesurer l'effet de la compression, la figure 4.7 montre la séparation entre deux cibles distancées de 1 Km.

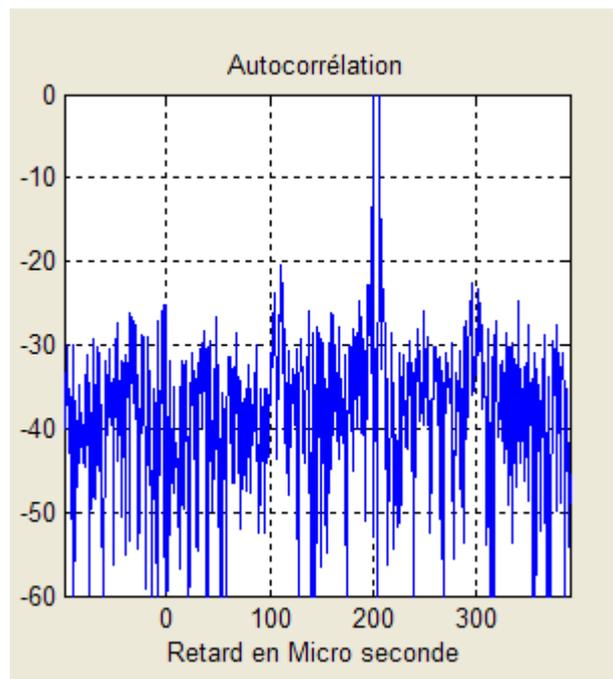


Figure 4.7 : Séparation entre deux cibles

Le signal modulé linéairement en fréquence est tolérant au décalage Doppler, la fonction d'ambiguïté présente une forme de CHIRP. En cas de décalage Doppler, le filtre donne toujours une réponse.

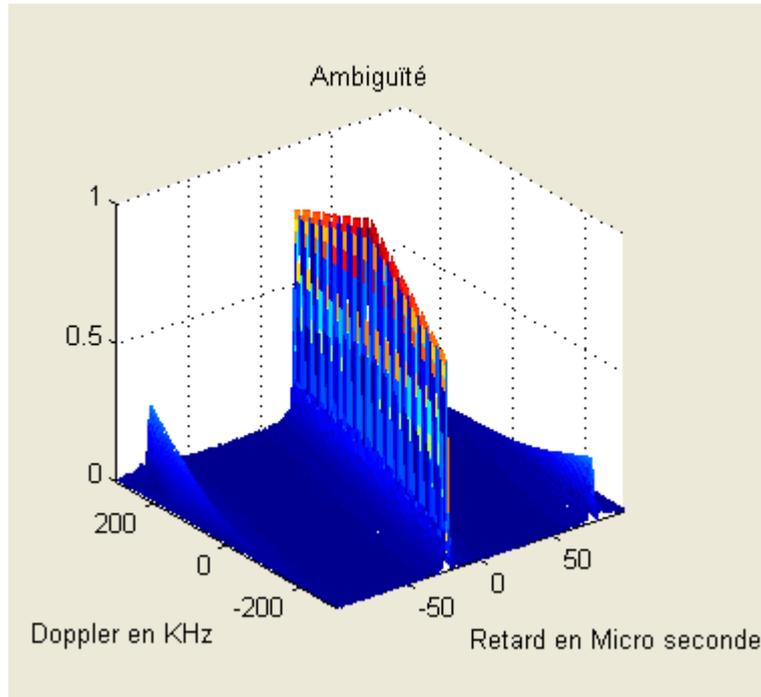


Figure 4.8 Fonction d'ambiguïté pour une LFM

La figure suivante illustre le spectre de ce type de codage

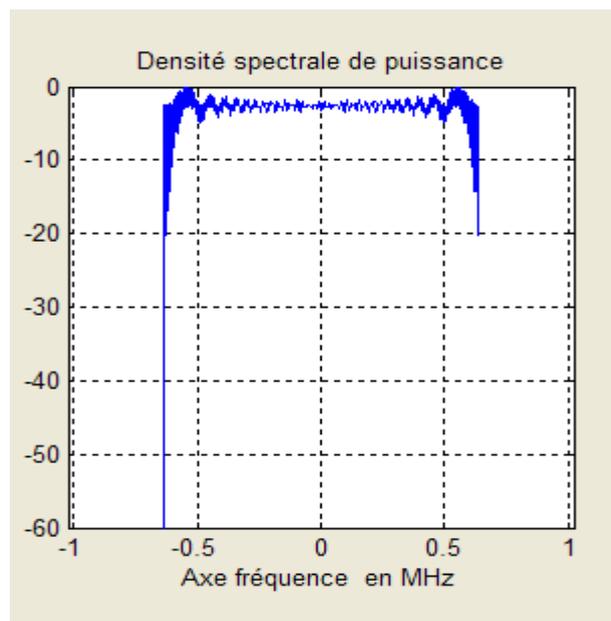


Figure 4.9 Densité de puissance pour une LFM

### 4.3 Codage non linéaire de fréquence de type 1 et 2

Un codage non linéaire de fréquence nous permet de réduire le niveau des lobes secondaires avec une perte en résolution.

Pour générer une forme d'onde avec un codage non linéaire de fréquence, on a deux choix : soit en utilisant une formule empirique, soit par l'utilisation du principe de la phase stationnaire [1].

La figure suivante illustre l'algorithme de génération du codage de fréquence de type 1 basé sur la formule empirique.

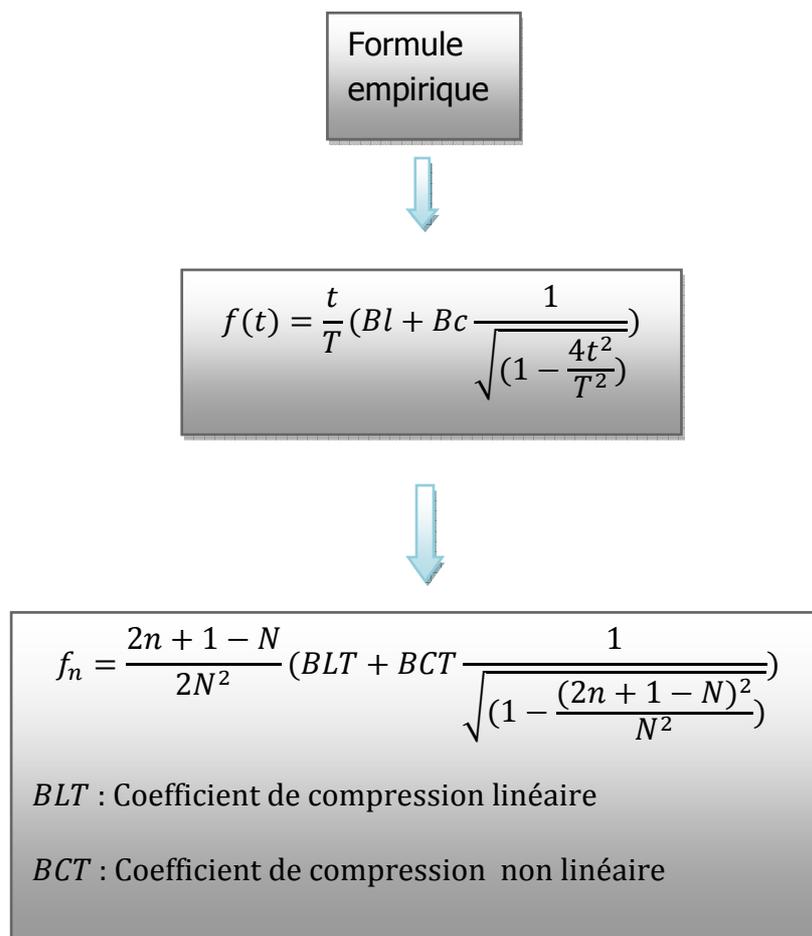


Figure 4.10 : Algorithme de génération pour un codage non linéaire de fréquence

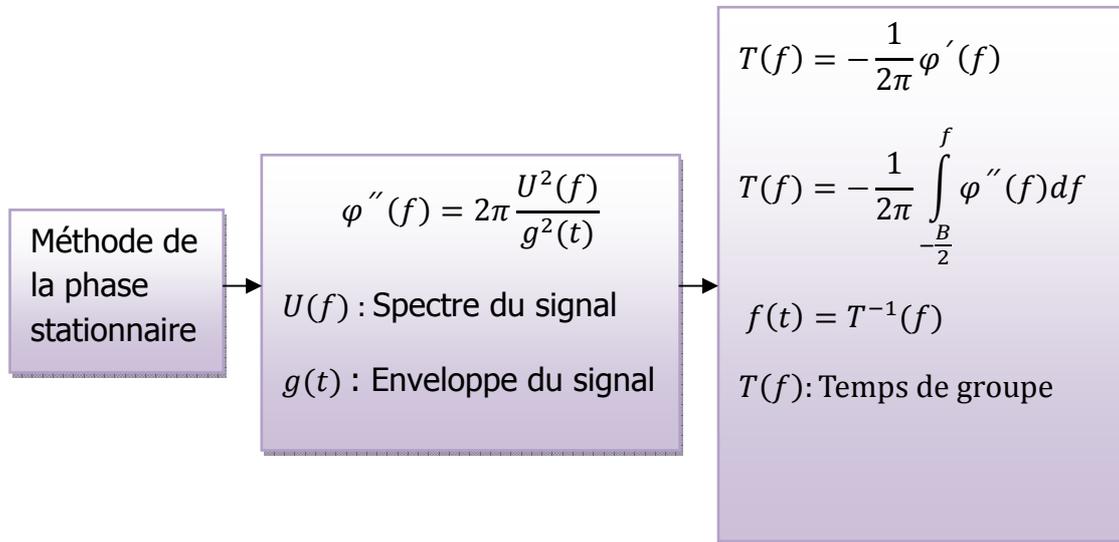


Figure 4.11 : Génération du codage non linéaire de type 2 par le principe de la phase stationnaire.

### Codage non linéaire de fréquence de type 1

Pour le codage non linéaire de fréquence de la figure 4.10, on prend l'exemple suivant :

On se fixe l'objectif que tous les lobes secondaires soient inférieurs à -40 dB, la conséquence de ce changement de codage est la diminution de la résolution.

Grandeur	Valeur
T : Durée du signal	100 $\mu$ s
B : Bande de fréquence	1.28 MHz
fs_1 : Fréquence d'échantillonnage	1.28 MHz
M_1 : Nombre de point	128
BLT : Rapport de compression linéaire	40
BCT : Rapport de compression non linéaire	12

Tableau 4.2 : Caractéristiques de la simulation pour une NLFM de type 1

La conséquence du codage non linéaire est une diminution de la résolution, le gain réel est constaté sur le niveau des lobes secondaires soit la figure 4.12 qui illustre la réponse du filtre adapté :

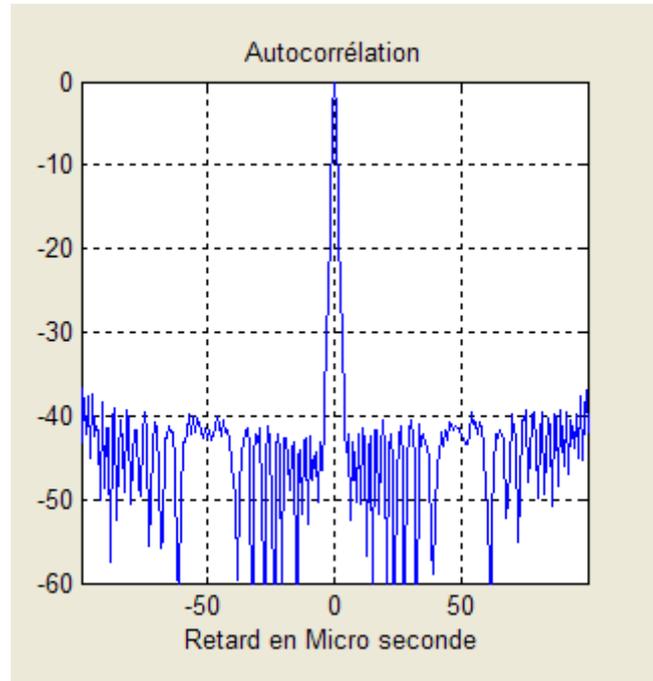


Figure 4.12 Réponse du filtre adapté pour une NLFM de type 1

Caractéristiques de l'impulsion compressée :

Grandeur	Forme d'onde
Résolution en distance	$\tau_{1\ nul} = 3.5\ \mu s$
Premier lobe secondaire	$n_{lobe\ s1} = -42\ dB$
Le plus grand lobe	$n_{lobe\ sx} = -36.6\ dB$
La location du premier lobe	$\tau_{1\ sl} = 5.5\ \mu s$

Le premier zéro se trouve à  $\tau_{1\ nul} = 3.5\ \mu s$ . Le pouvoir de discrimination entre deux cibles est  $R_{dis} = 525\ m$ .

Le codage non linéaire de fréquence augmente la distance minimale de discrimination avec une réduction du niveau des lobes secondaires.

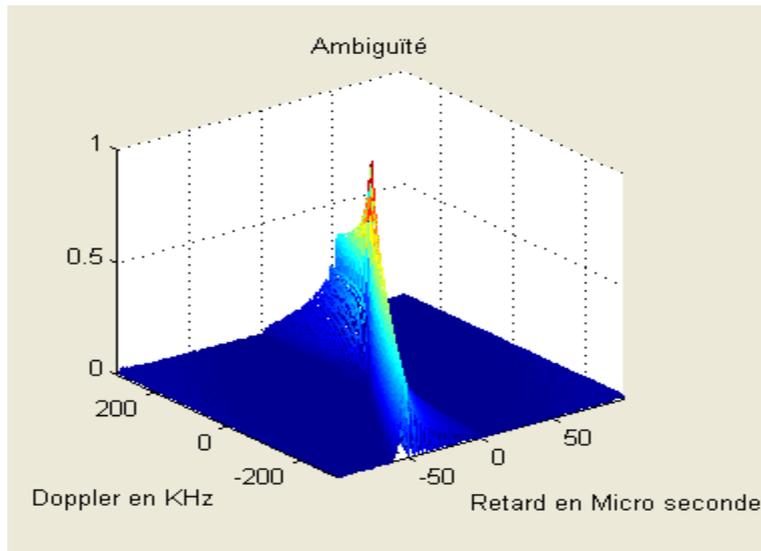


Figure 4.13 Fonction d'ambiguïté pour une NLFM de type 1

L'effet de la non linéarité sur la réponse du filtre peut-être observé grâce à la fonction d'ambiguïté, le volume de la réponse du filtre est réparti sur l'axe Doppler d'une manière qui suit la variation non linéaire de la fréquence. On constate le gain sur le niveau des lobes secondaires et la perte au niveau de la résolution.

Pour le même exemple de séparation entre deux cibles, la réponse du filtre adapté est meilleure du point de vue lobes secondaires voir figure 4.14

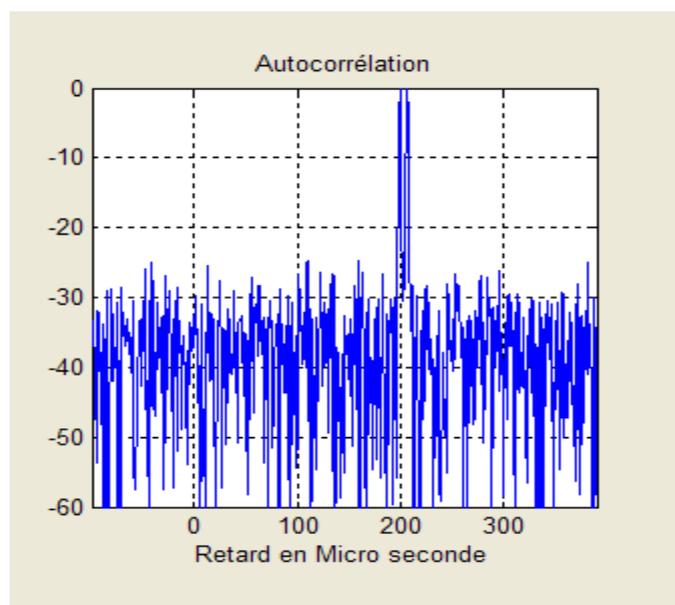


Figure 4.14 Séparation entre deux cibles

De la figure 4.14 on peut constater que le niveau des lobes secondaires a diminué d'une manière significative.

Le codage non linéaire de fréquence tire son principe de la relation qui lie la fonction d'auto-corrélation avec la densité de puissance. Le changement de la forme du spectre du signal permet le changement de la réponse du filtre adapté.

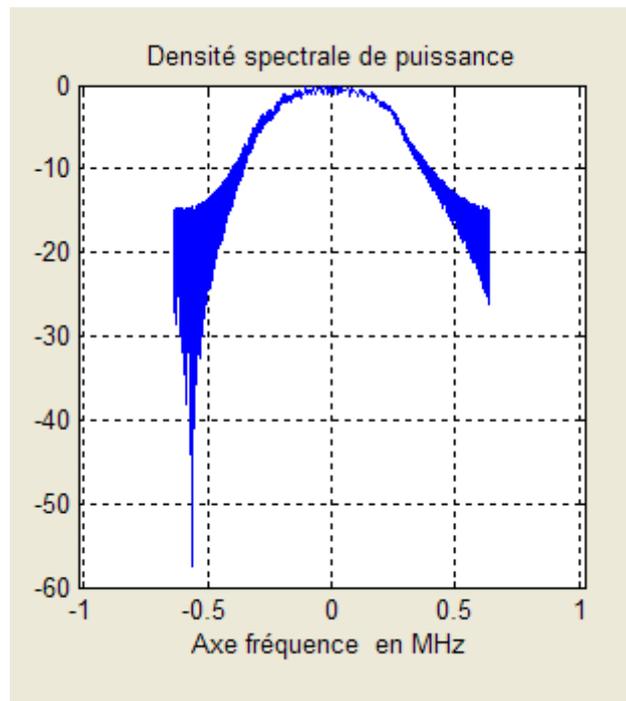


Figure 4.15 Densité de puissance pour une NLFM de type 1

Sur la figure 4.15 on remarque que la forme du spectre a changé, ce type de changement a entraîné la réduction du niveau des lobes secondaires.

Un niveau élevé de lobe secondaire peut fausser la détection dans le cas de la présence de deux cibles différentes. En termes de surface équivalente, les lobes secondaires de la plus grande cible peuvent cacher la plus petite cible.

L'utilisation du codage non linéaire de fréquence permet de réduire le niveau des lobes pour une meilleure détection.

### Codage non linéaire de fréquence de type 2

Soit l'exemple d'un codage non linéaire de fréquence de type 2 qui utilise le principe de la phase stationnaire

En fixant la forme du spectre, on peut déduire un codage non linéaire de fréquence.

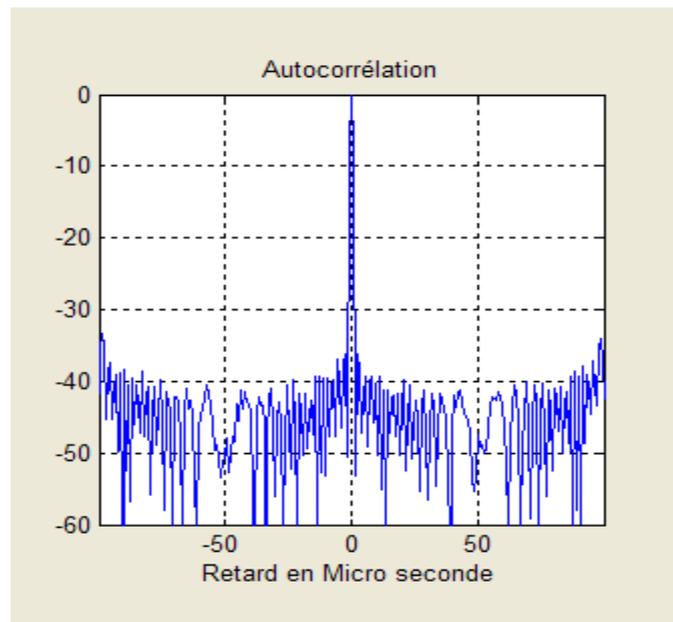


Figure 4.16 : Réponse du filtre adapté pour une NLFM de type 2

La figure 4.16 montre une atténuation exceptionnelle des lobes secondaires.

Caractéristiques de l'impulsion compressée :

Grandeur	Forme d'onde
Résolution en distance	$\tau_{1\text{ nul}} = 2 \mu\text{s}$
Premier lobe secondaire	$n_{\text{lobe } s1} = -36.2 \text{ dB}$
Le plus grand lobe	$n_{\text{lobe } sx} = -34 \text{ dB}$
La location du premier lobe	$\tau_{1\text{ sl}} = 2.35 \mu\text{s}$

Pour ce type de codage, on peut discriminer entre deux cibles à une distance minimale  $R_{dis} = 300 \text{ m}$ .

Pour un codage non linéaire de fréquence basé sur le principe de la phase stationnaire, la réponse du filtre adapté présente les mêmes caractéristiques qu'un codage non linéaire qui utilise la formule empirique et ce avec un meilleur contrôle sur la génération du code.

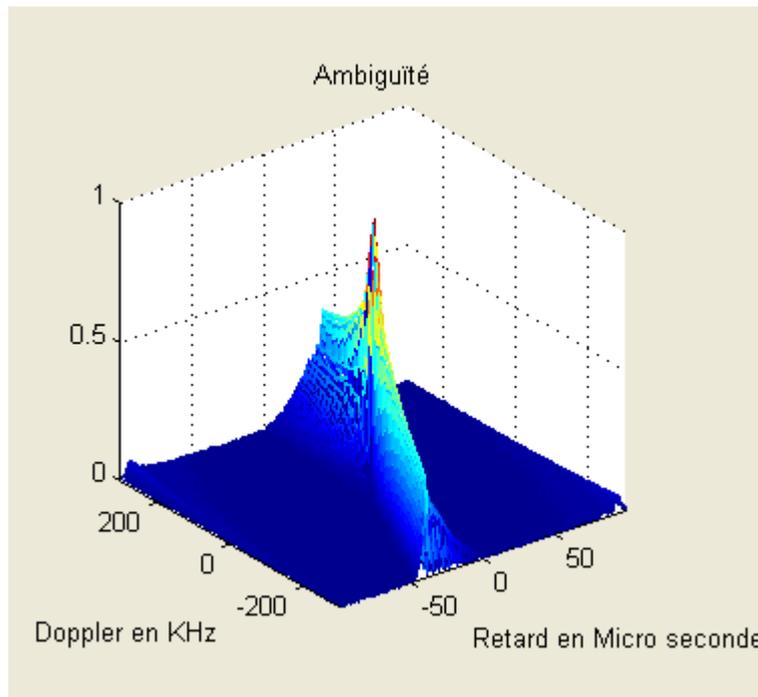


Figure 4.17 Fonction d'ambiguïté pour une NLFM de type 2

La figure 4.17 montre une grande tolérance au décalage Doppler dans le cas du codage non linéaire de fréquence

La figure suivante illustre la séparation entre deux cibles dans le cas du codage non linéaire de fréquence basé sur le principe de la phase stationnaire.

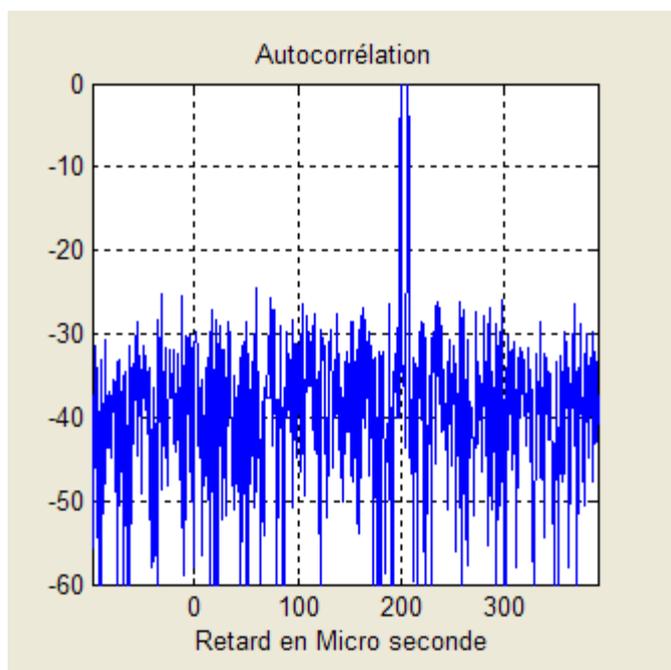


Figure 4.18 Séparation entre deux cibles

Le spectre de cette forme d'onde est illustré sur la figure suivante

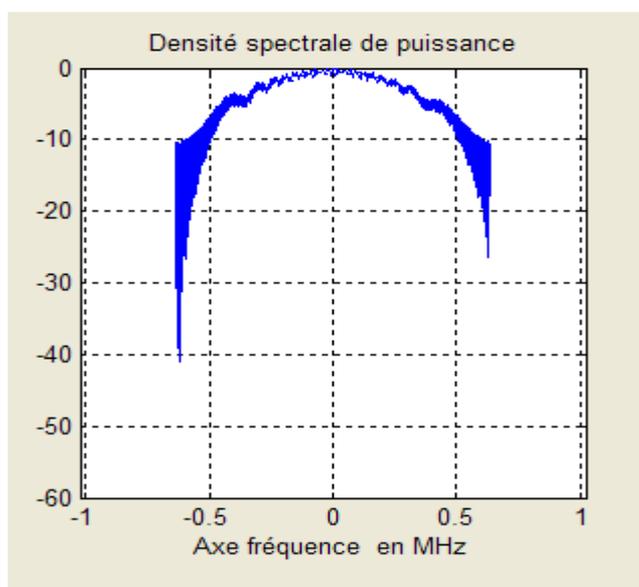


Figure 4.19 Spectre d'une NLFM de type 2

Le spectre de la figure 4.19 suit la forme utilisée au niveau du codage de fréquence.

#### 4.4 Codage linéaire de fréquence avec fenêtrage de type HAMMING

Dans les précédentes simulations on a essayé de réduire le niveau des lobes secondaires en utilisant un codage non linéaire fréquence qui ne crée pas de désadaptation. Mais si on accepte une petite perte au niveau du SNR, on peut réduire le niveau des lobes en utilisant une forme d'onde codée linéairement en fréquence et codée en amplitude par une fenêtre de pondération de type HAMMING ou autre.

La réponse du filtre de compression présente un élargissement du lobe principal et une atténuation des lobes secondaires à plus de -50 dB.

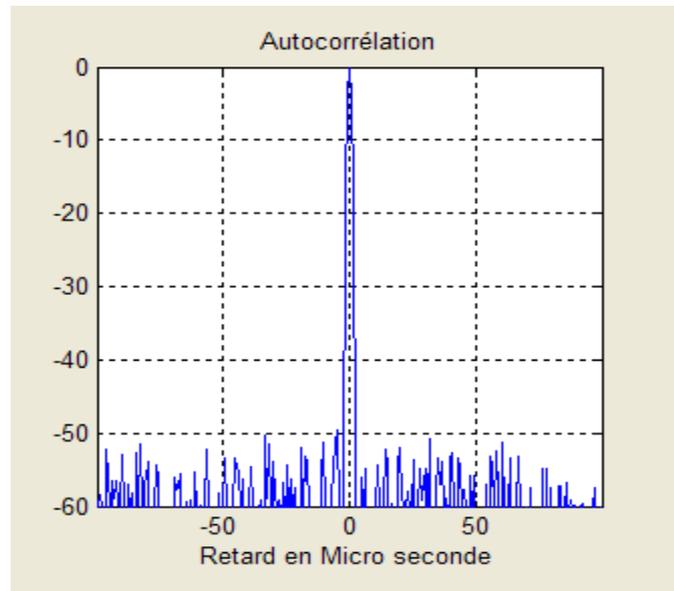


Figure 4.20: Réponse du filtre adapté pour une LFM avec un codage d'amplitude de type (HAMMING)

##### Caractéristiques du filtre adapté

Grandeur	Forme d'onde
Résolution en distance	$\tau_{1\text{ nul}} = 2.1 \mu\text{s}$
Premier lobe secondaire	$n_{\text{lobe } s1} = -55 \text{ dB}$
Le plus grand lobe	$n_{\text{lobe } sx} = -50 \text{ dB}$
La location du premier lobe	$\tau_{1\text{ sl}} = 4.5 \mu\text{s}$

Pour ce type de codage, on peut discriminer entre deux cibles à une distance minimale  $R_{dis} = 315 \text{ m}$ .

L'utilisation d'un codage au niveau de l'amplitude diminue le niveau des lobes secondaires avec une perte en résolution.

Sur la fonction d'ambiguïté, on peut voir l'effet du codage au niveau d'amplitude.

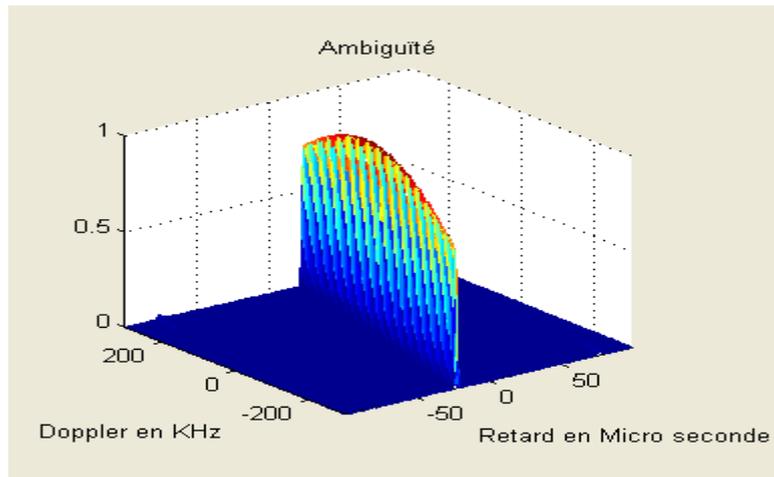


Figure 4.21 Fonction d'ambiguïté pour une LFM avec un codage d'amplitude de type (HAMMING)

La forme du spectre pour ce type de forme d'onde est illustrée sur la figure suivante

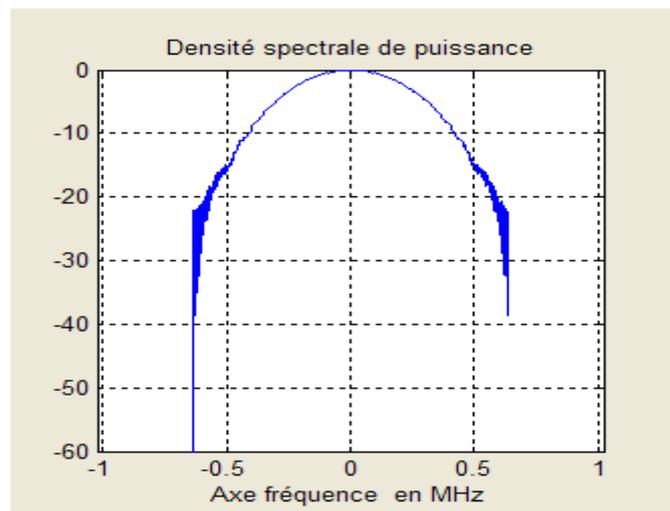


Figure 4.22 Densité de puissance pour une LFM avec pondération de type HAMMING

#### 4.5 Réponse du filtre adapté pour un codage de phase de type CHIRPLIKE

Pour un codage de phase de type CHIRPLIKE, la réponse du filtre adapté présente une tolérance au décalage Doppler. On prend deux exemples, le codage de Frank et celui de type P3 pour illustrer la réponse du filtre adapté [1].

Modèle mathématique pour les deux type de codage est :

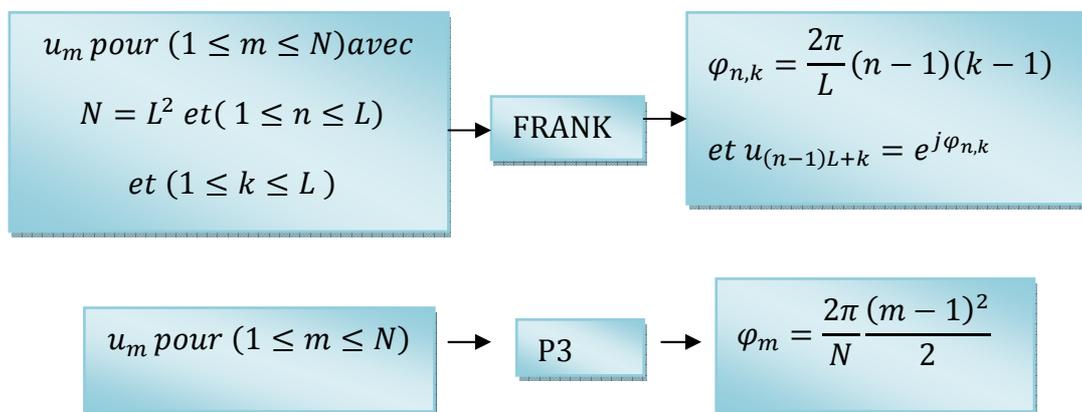


Figure 4.23 : Algorithme de génération des signaux de type CHIRPLIKE

#### Simulation du codage de type P3

Grandeur	Valeur
Taille du code	M=128
Fréquence d'échantillonnage	fs=1.28 M Hz

Si on utilise un codage de phase de type P 3, on obtient la réponse du filtre adapté de la figure 4.24 :

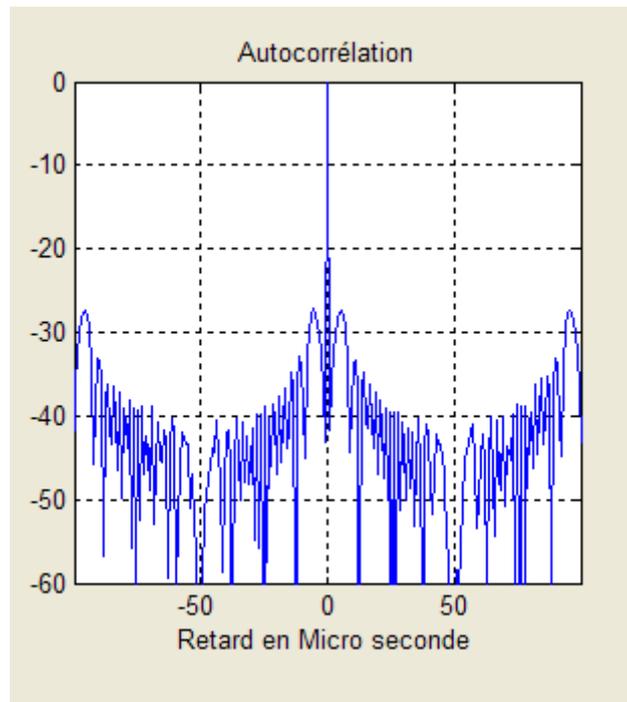


Figure 4.24 réponse du filtre adapté pour un codage de type P3

Caractéristiques de l'impulsion compressée :

Grandeur	Forme d'onde
Résolution en distance	$\tau_{1\text{ nul}} = 0.78 \mu\text{s}$
Premier lobe secondaire	$n_{\text{lobe } s1} = -27.4 \text{ dB}$
Le plus grand lobe	$n_{\text{lobe } sx} = -27.4 \text{ dB}$
La location du premier lobe	$\tau_{1\text{ sl}} = 5.5 \mu\text{s}$

Pour ce type de codage, on peut discriminer entre deux cibles à une distance minimale  $R_{dis} = 117 \text{ m}$ .

Caractéristique de la fonction d'ambiguïté pour un codage de type P3

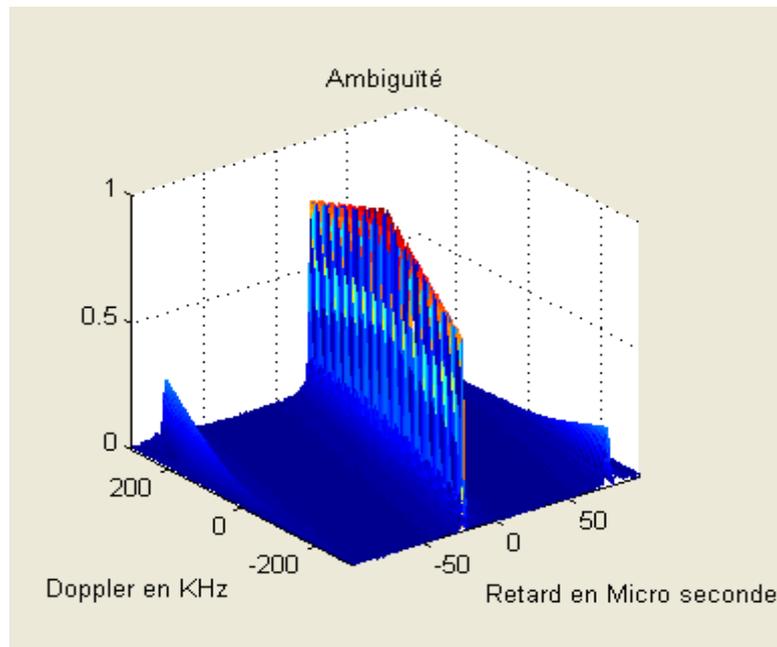


Figure 4.25 Fonction d'ambiguïté pour un codage de type P3

On voit que le codage de type P3 donne une même évolution de la forme de la fonction d'ambiguïté que celle d'un codage linéaire de fréquence par conséquent le codage P3 présente une tolérance au décalage Doppler.

Pour ce codage une représentation sur deux dimensions illustre la tolérance au décalage Doppler.

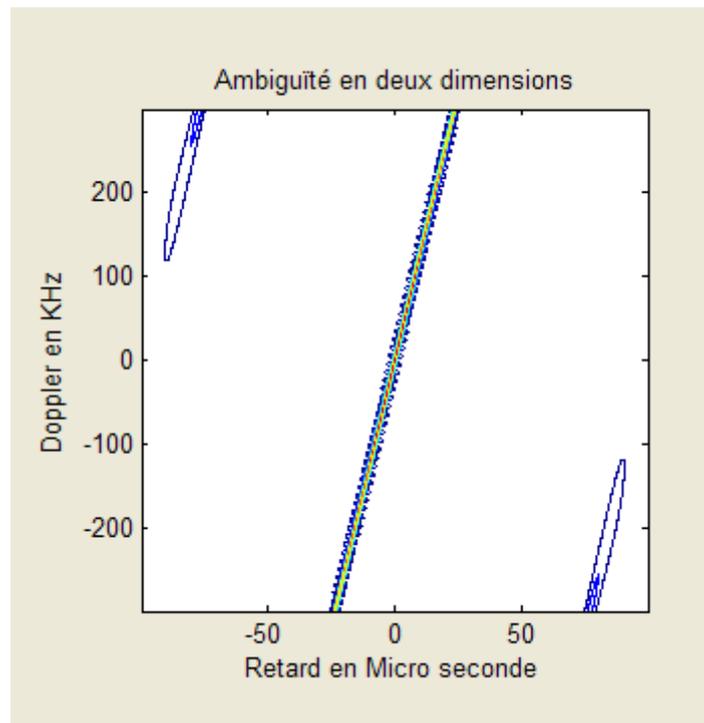


Figure 4.26: Représentation en deux dimensions de la fonction d'ambiguïté pour le codage P3

Sur la figure 4.26 on peut observer la relation de linéarité qui existe entre le domaine temporel et le domaine fréquentiel dans le cas du codage de type P3.

### Simulation du codage de Frank

Si on prend Le codage de Frank, il se calcule pour un code qui présente un nombre d'éléments dans la racine carrée est un nombre entier. La figure 4.27 présente la réponse du filtre adapté, on remarque des caractéristiques très intéressantes du point de vue résolution et lobes secondaires. De plus il tolère le décalage Doppler. On le constate sur la fonction d'ambiguïté de la figure 4.28 :

Grandeur	Valeur
Taille du code	N=256
Fréquence d'échantillonnage	fs=2.56 M Hz

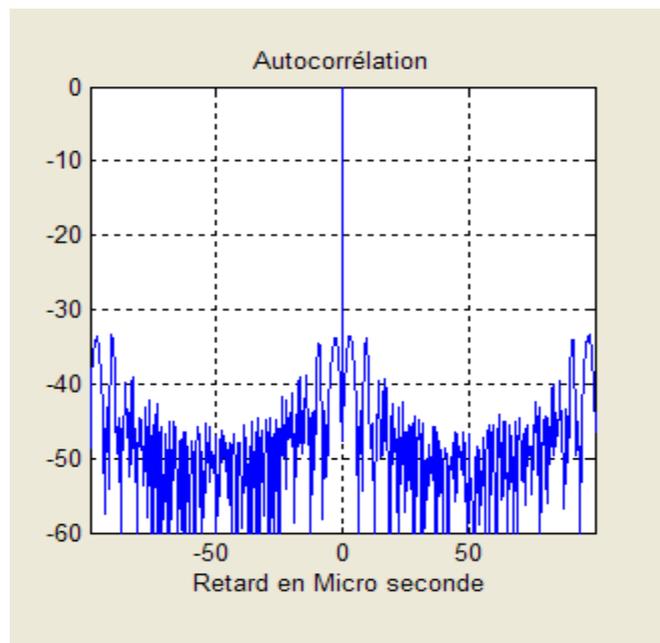


Figure 4.27 : Réponse du filtre adapté pour le codage de Frank

Grandeur	Forme d'onde
Résolution en distance	$\tau_{1\text{ nul}} = 0.4 \mu s$
Premier lobe secondaire	$n_{lobe\ s1} = -34\ dB$
Le plus grand lobe	$n_{lobe\ sx} = -34\ dB$
La location du premier lobe	$\tau_{1\text{ sl}} = 3.1 \mu s$

Pour ce type de codage, on peut discriminer entre deux cibles à une distance minimale  $R_{dis} = 60\ m$ .

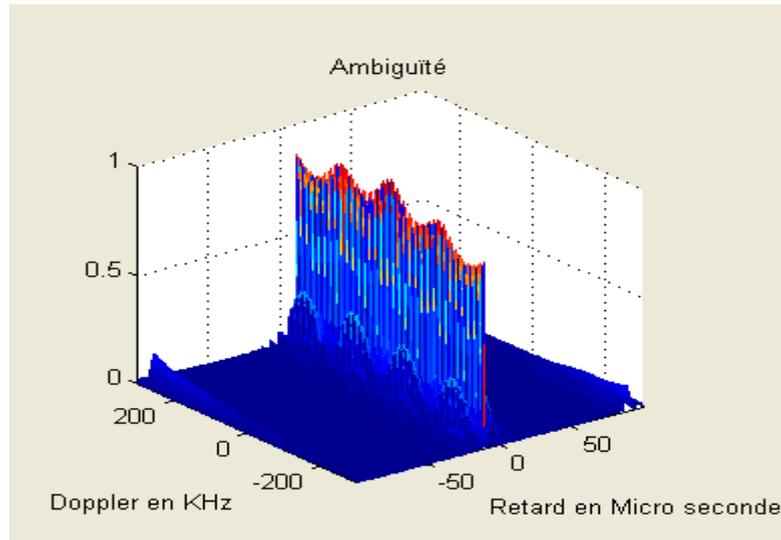


Figure 4.28 : Fonction d'ambiguïté pour le codage de Frank

Si on prend l'exemple de la séparation entre deux cibles la réponse du filtre adapté aura la forme suivante :

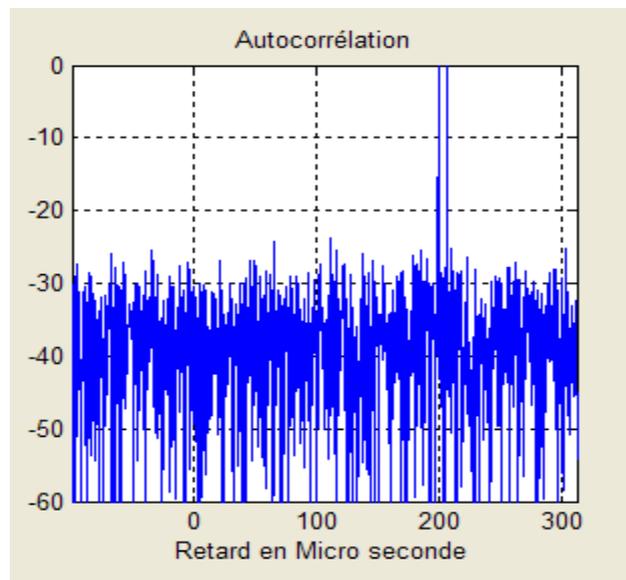


Figure 4.29 : Séparation entre deux cibles

Le codage de Frank est le meilleur code du point de vue résolution et lobes secondaires. Le seul problème est qu'il ne se calcule que pour des vecteurs avec une valeur entière de la racine carrée du nombre de points du code.

#### 4.6 Réponse du filtre adapté pour un codage de fréquence de type COSTAS

Modèle mathématique de la génération de signaux de type COSTAS par l'algorithme de type WELCH [1].

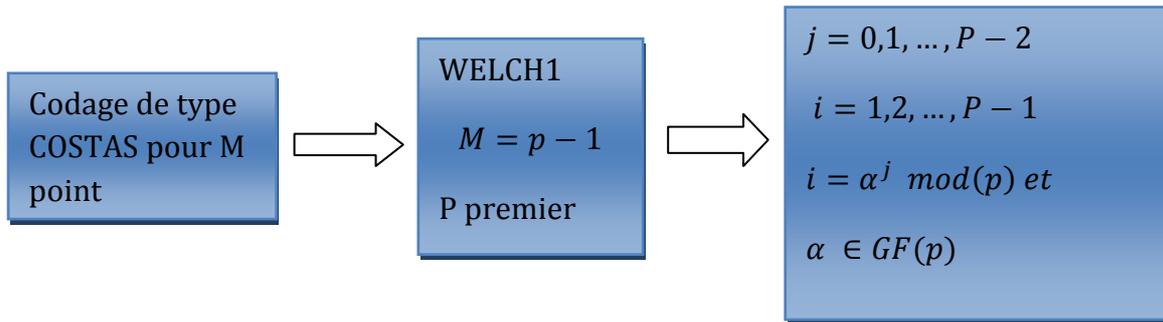


Figure 4.30 : Algorithme de calcul du codage de type COSTAS

Dans le cas où  $p$  n'est pas premier, on utilise la méthode WELCH2 et WELCH3

Caractéristiques de la simulation suivante :

Grandeur	Valeur
Taille du code	M=16
Fréquence d'échantillonnage	fs=2.56 M Hz
Coefficient de compression	256

La figure 4.30 montre la réponse du filtre adapté. On a changé la taille du code à cause de la contrainte liée au modèle de génération de la forme d'onde. Il faut que la racine du rapport de compression soit entière, donc on est passé de 128 à 256.

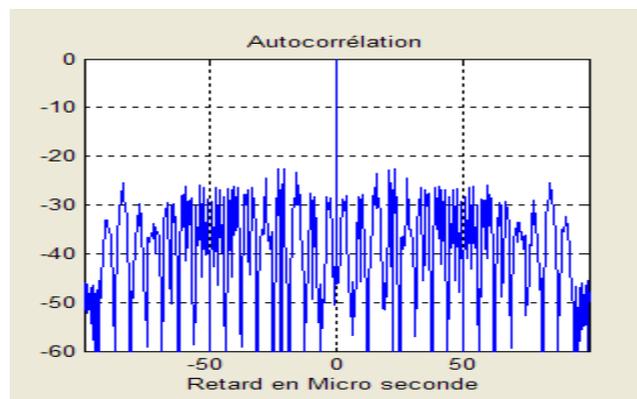


Figure 4.31 : Réponse du filtre adapté pour un codage de type COSATS

Caractéristiques de l'impulsion compressée :

Grandeur	Forme d'onde
Résolution en distance	$\tau_{1\text{ nul}} = 0.4 \mu\text{s}$
Premier lobe secondaire	$n_{\text{lobe } s1} = -28.5 \text{ dB}$
Le plus grand lobe	$n_{\text{lobe } sx} = -22.6 \text{ dB}$
La location du premier lobe	$\tau_{1\text{ sl}} = 3.1 \mu\text{s}$

Pour ce type de codage, on peut discriminer entre deux cibles à une distance minimale  $R_{dis} = 60 \text{ m}$ .

Si on compare les résultats de la simulation avec les prévisions théoriques [1], on trouve que  $\rho = 256$  et les lobes entre  $[-22.6, -25.5]$  et théoriquement on a  $\rho = 256$  et les lobes entre  $[-21, -24]$ .

La figure 4.32 nous montre l'intolérance au décalage Doppler du codage de type COSTAS.

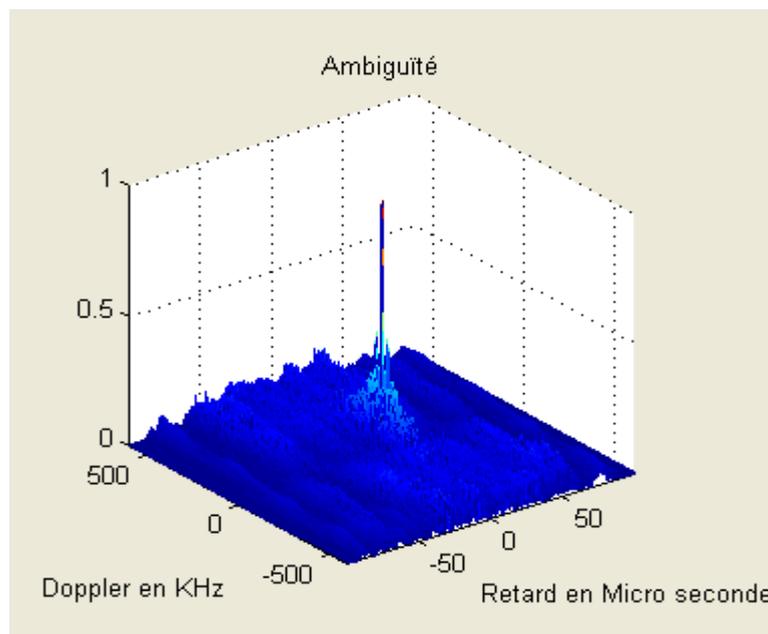


Figure 4.32 : Fonction d'ambiguïté pour un codage de type COSTAS

Pour le même exemple de séparation entre deux cibles la réponse du filtre adapté est illustrée sur la figure 4.33 :

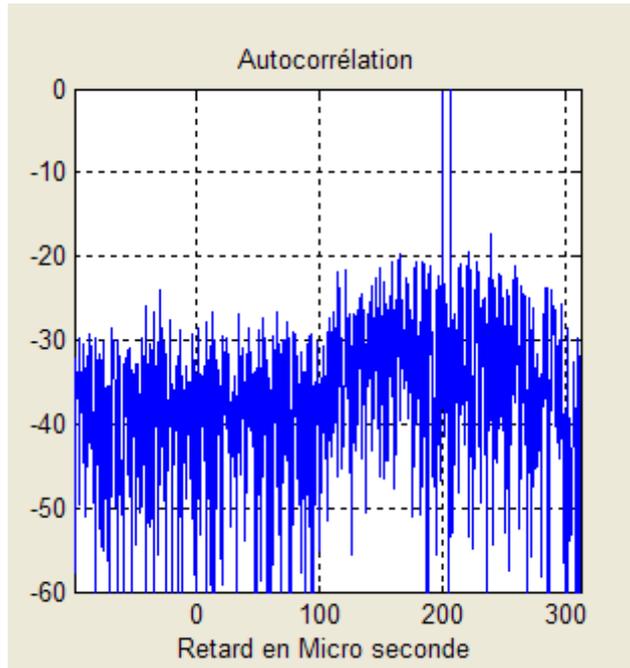


Figure 4.33

Le codage de type COSTAS nous permet d'avoir un contrôle sur le niveau des lobes secondaires, sachant qu'on peut générer le code ensuite trouver la distribution des lobes secondaires sur l'axe temps fréquence.

Pour une application donnée, si on se fixe que tous les lobes secondaires sont inférieurs à -40 dB, théoriquement si on génère un code de COSATS avec 100 éléments tous les lobes secondaires seront à -40 dB et pratiquement entre [-34,-40 dB].

Pour trouver la distribution des lobes secondaires, on doit calculer la matrice de différence puis déduire la matrice des lobes secondaires [1]. Ce qui nous donne la distribution des lobes ; dans cet exemple on se limite à une représentation numérique de la projection sur deux dimensions de la fonction d'ambiguïté.

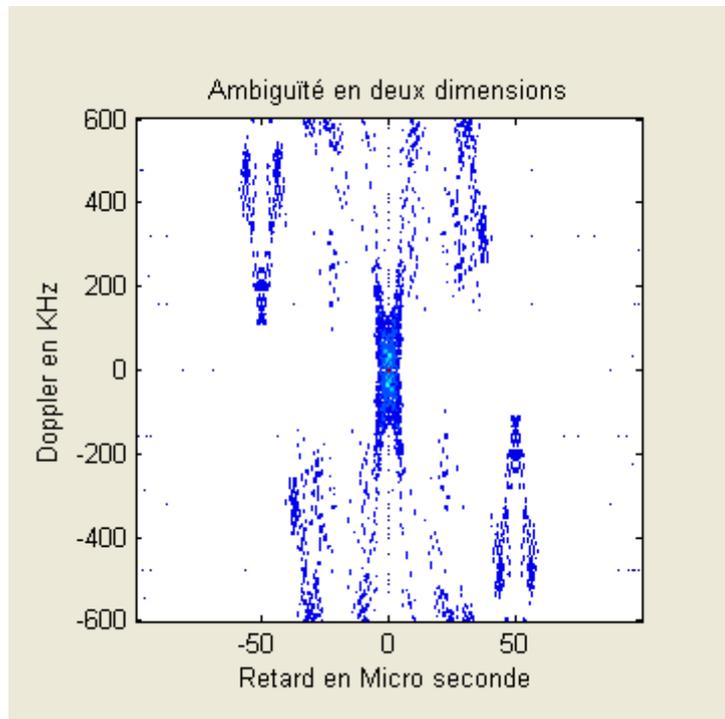


Figure 4.34 : Représentation du plan temps fréquence

La distribution des lobes secondaires qu'on voit sur la figure 4.34 peut se calculer à l'aide d'un algorithme traité dans la référence [1].

## 4.7 Interface de simulation

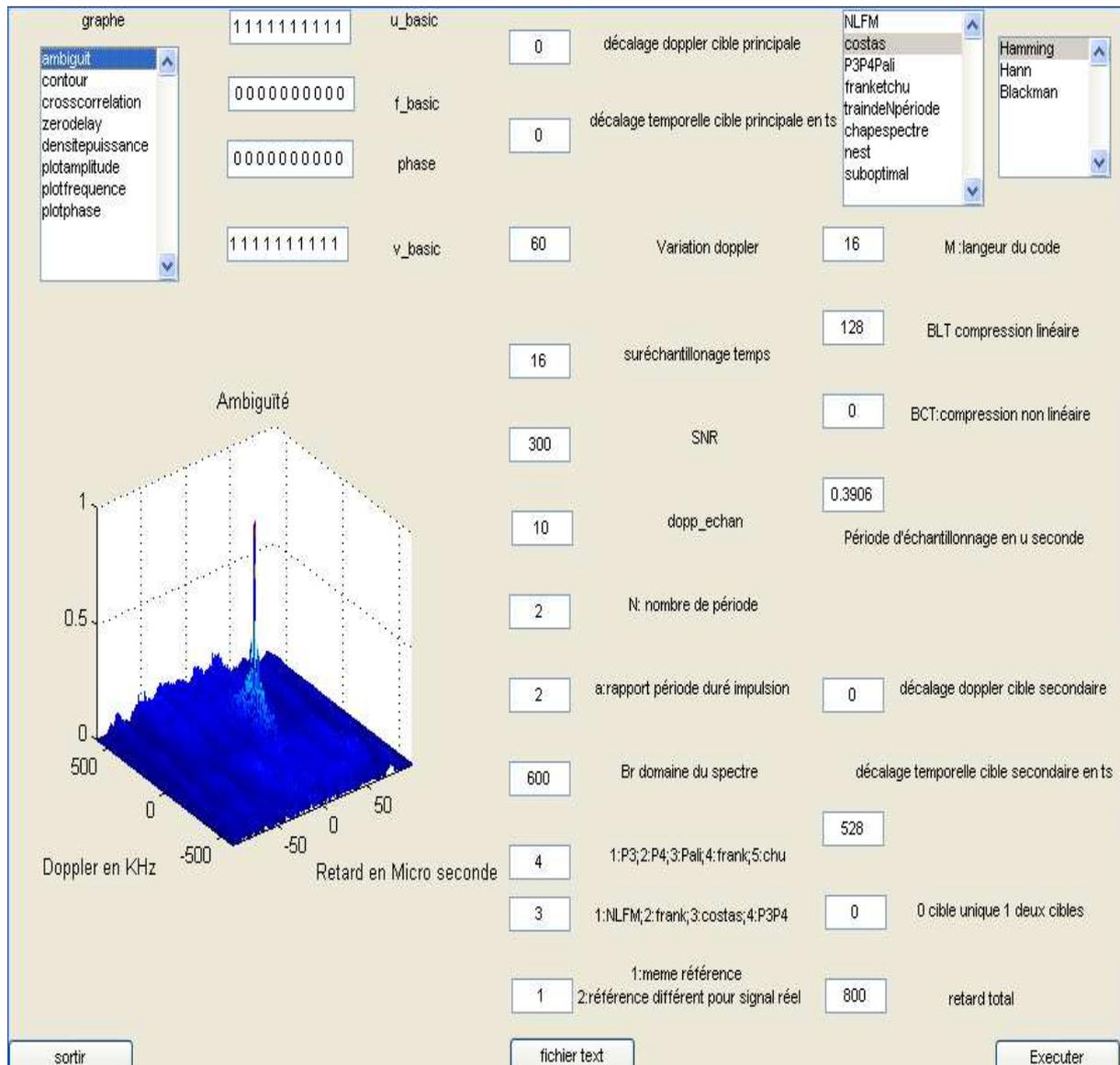


Figure 4.35 : Interface de simulation

Pour réaliser les diverses simulations, on doit en premier lieu générer tous les types de forme d'onde énumérées dans ce document et d'autres non traitées, ensuite calculer et tracer la réponse du filtre adapté pour visualiser l'effet de la compression, la résolution temps fréquence et le niveau des lobes secondaires.

<b>Symbole</b>	<b>Définition</b>
u_basic	L'enveloppe de la référence utilisée lors du filtrage
f_basic	Fréquence de la référence utilisée lors du filtrage
phase	Phase de la référence utilisée lors du filtrage
V_basic	Référence optimisée pour le niveau des lobes secondaires
Ambiguïté	Fonction qui trace la réponse temps fréquence du filtre
Cross_corrélation	Réponse du filtre adapté
Zero_delay	La réponse en Doppler de la cible
Densite_puissance	Le tracé du spectre du signal
Plot_amplitude	Il trace l'enveloppe du signal
Plot_phase	Il trace la phase du signal
Plot_fréquence	Il trace la fréquence du signal
NLFM	Codage non linéaire de fréquence par une équation empirique type 1
COSTAS	Codage fréquentiel de type COSTAS
P3P4PALI	Codage de phase de type P3 P4 ...
Franketchy	Codage de phase de type frank et chy
TRAINdeNperiode	Train d'impulsion codé par toutes les formes simples
Chapspectre	Codage de fréquence par fenêtrage spectral type 2
Nest	Combinaison entre divers code de Barker
Suboptimal	Référence optimisée dans le cas des signaux biphasés
Hamming	Variation de l'enveloppe de la référence selon Hamming
Hann	Variation de l'enveloppe de la référence selon Hann
Blackman	Variation de l'enveloppe de la référence selon Blacman
M	Nombre de bits élémentaires utilisés dans le signal
N	Nombre de périodes de répétition
A	Rapport période impulsion
BLT	Coefficient de compression de la partie linéaire
BCT	Coefficient de compression de la partie non linéaire
Ts	Période d'échantillonnage en us

Tableau 4.3 Constituants de l'interface de simulation

Grâce à une programmation modulaire, on a pu rassembler toutes les formes d'onde étudiées dans le document et les mettre sous une interface créée grâce à MATLAB, cette interface donne la réponse du filtre adapté et beaucoup d'autres informations pour le dessin de la forme d'onde et le choix du type de codage.

#### 4.8 Récapitulatif des résultats de simulations

<b>Modulation</b>	<b>Résolution</b>	<b>Lobes secondaire</b>
LFM (128)	$\tau_{1\text{ nul}} = 0.78 \mu\text{s}$	$n_{\text{lobe } s1} = -27.5 \text{ dB}$ $n_{\text{lobe } sx} = -27.5 \text{ dB}$ $\tau_{1\text{ sl}} = 5.5 \mu\text{s}$
NLFM type 1 (128)	$\tau_{1\text{ nul}} = 3.5 \mu\text{s}$	$n_{\text{lobe } s1} = -42 \text{ dB}$ $n_{\text{lobe } sx} = -36.6 \text{ dB}$ $\tau_{1\text{ sl}} = 5.5 \mu\text{s}$
COSTAS (256)	$\tau_{1\text{ nul}} = 0.4 \mu\text{s}$	$n_{\text{lobe } s1} = -28.5 \text{ dB}$ $n_{\text{lobe } sx} = -22.6 \text{ dB}$ $\tau_{1\text{ sl}} = 3.1 \mu\text{s}$
P3 (128)	$\tau_{1\text{ nul}} = 0.78 \mu\text{s}$	$n_{\text{lobe } s1} = -27.4 \text{ dB}$ $n_{\text{lobe } sx} = -27.4 \text{ dB}$ $\tau_{1\text{ sl}} = 5.5 \mu\text{s}$
NLFM type 2 (128)	$\tau_{1\text{ nul}} = 2 \mu\text{s}$	$n_{\text{lobe } s1} = -36.2 \text{ dB}$ $n_{\text{lobe } sx} = -34 \text{ dB}$ $\tau_{1\text{ sl}} = 2.35 \mu\text{s}$
Frank (256)	$\tau_{1\text{ nul}} = 0.4 \mu\text{s}$	$n_{\text{lobe } s1} = -34 \text{ dB}$ $n_{\text{lobe } sx} = -34 \text{ dB}$ $\tau_{1\text{ sl}} = 3.1 \mu\text{s}$
LFM + HAMMING (128)	$\tau_{1\text{ nul}} = 2.5 \mu\text{s}$	$n_{\text{lobe } s1} = -55 \text{ dB}$ $n_{\text{lobe } sx} = -50 \text{ dB}$ $\tau_{1\text{ sl}} = 4.5 \mu\text{s}$

Tableau 4.4 : Résumé des différentes simulations

#### 4.9 Conclusion

Si on avait à choisir un code qui réalise la meilleure compression d'impulsion, on doit s'orienter vers un codage de phase. Parmi les différents codages de phase celui de Frank présente le bon compromis entre le niveau des lobes secondaires et la résolution comparativement aux différents codes simulés.

## Implémentation sur circuit FPGA

## 5.1 Introduction

Le but de ce travail est d'implémenter sur FPGA un filtre adapté à base de procédé de compression d'impulsion radar.

Le choix s'est porté sur l'FPGA pour sa structure configurable. Le fait qu'il réalise une abstraction des contraintes matérielles, nous permet une plus grande flexibilité dans la programmation. De ce fait on doit se focaliser sur la description correcte de l'application. La famille XILINX qui réalise ce type de circuits fournit une bibliothèque orientée pour le traitement du signal, ce qui accélère la phase de synthèse et de prototypage.

La description modulaire de l'application utilise le langage VHDL pour l'implémentation sur FPGA et l'estimation des contraintes temps réel et des ressources.

Le processeur appartient à la famille des FPL (field programming logic), ce sont des composants programmables avec des cellules élémentaires qui se répètent sur toute la surface du processeur constituant une matrice de bloc. On trouve deux types de bloc, les CLB (configurable logic block) et les IOB (input output block). On peut aussi associer l'FPGA à l'ASIC (application specific integrated circuit), mais l'FPGA a l'avantage de la reprogrammation, ce qui le différencie des ASIC. [5],[14],[15].

L'application à implémenter consiste à réaliser un filtre adapté pour un récepteur radar. On définit le signal d'entrée comme étant un vecteur qui contient N échantillons codés sur un certain nombre de bits selon l'application.

$$y(n) = \begin{cases} x(n) * h(n) & 5.1 \\ \sum_k x(k)h(n-k) & 5.2 \end{cases}$$

La fonction de convolution définit le modèle mathématique du filtrage (5.1), son implémentation sous cette forme présente une grande complexité du fait qu'on utilise un nombre d'échantillons important. Pour y remédier on utilise une transformation Fourier.

On a besoin de trois composants, le premier est dédié à la transformation du signal reçu, le deuxième pour faire une multiplication complexe puis le dernier pour réaliser la transformation inverse, on ajoute un bloc qui réalise le calcul du module.

Cette opération est définie comme étant une convolution rapide et le but du présent travail est de l'implémenter sur FPGA :

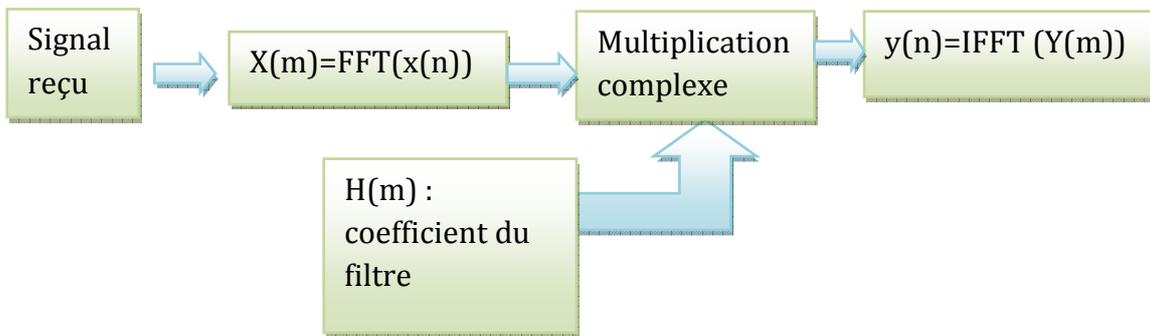


Figure 5.1 Schéma synoptique du filtre adapté

Sur la figure 5.1, on représente les différentes phases à réaliser pour l'implémentation de l'opération de compression qui se base principalement sur la mise en œuvre au niveau du circuit FPGA de deux opérations FFT. Pour avoir les spectres du signal reçu et de la réplique de l'impulsion émise. Les deux spectres sont multipliés point à point ce qui équivaut à une opération de corrélation. Enfin on accomplit une FFT inverse du produit complexe des deux spectres pour avoir la sortie du filtre dans le domaine temporel.

La conception du filtre de compression est transcrite en langage VHDL et synthétisée avec le logiciel Xilinx ISE pour être supportée au niveau du circuit FPGA VIRTEX 4vfx100ff1152 avec une contrainte temps de 0.8 ms ( $1024 \times 0.78 \mu\text{s}$ ).

Les simulations des résultats d'implémentations du filtre de compression avec différentes formes. Nous utilisons un logiciel dédié en l'occurrence le modelsim. A partir du projet VHDL du filtre de compression nous générons des fichiers qui sont liés aux contraintes temps et ressources à partir desquels nous assurons la transcription du projet VHDL du logiciels ISE vers le logiciels Modelsim.

Pour donner un aperçu sur la démarche d'implémentation du filtre de compression, nous décrivons principalement la réalisation d'une opération FFT sur un circuit FPGA de type Virtex grâce à la figure 5.2 qui donne un aperçu sur les entrées sorties du CORE FFT avec ses entrées/sorties en terme de signaux de données brutes et filtrées ainsi que ses signaux de synchronisation.

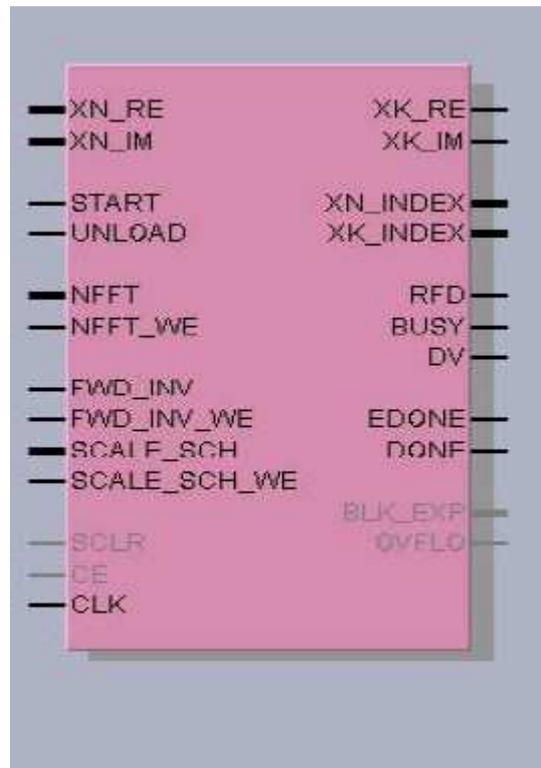


Figure 5.2 Composant qui réalise la fonction FFT

On peut compter sur la figure (5.2) 13 signaux d'entrée et 11 signaux de sortie, le tableau 5.1 illustre la définition des signaux importants

<b>Signal</b>	<b>Définition</b>
Xn_re	Composante réelle du signal d'entrée
Xn_im	Composante imaginaire du signal d'entrée
Xk_re	Composante réelle du signal de sortie
Xk_im	Composante imaginaire du signal de sortie
Clk	Horloge du microprocesseur
NFFT	Le nombre de point de la FFT
FWD_IN	Type de la transformation (0 IFFT ou 1 FFT)
Scale_sch	Division pour ne pas dépasser le nombre de bit max
RFD	Il peut recevoir des données
BUSY	Il calcule la transformation
Dv	Le résultat est disponible à la sortie

Tableau 5.1

La figure 5.3 représente le schéma électronique du filtre de compression après synthèse du projet VHDL du filtre de compression, où nous présentons les fonctionnalités du filtre de compression implémenté qui comporte :

- Une FFT pour le signal reçu ;
- Un produit complexe entre le spectre du signal reçu et la réplique du signal émis ;
- Une FFT inverse ;
- Un calcul du module du signal complexe compressé.

Il y'a lieu de noter, la non nécessité de calculer à chaque période de répétition radar le spectre de la réplique du fait que celle-ci est inchangée d'une émission à la suivante. Ce qui permet de charger les points du spectre de la réplique à partir d'une table externe (EPROM) ou la lecture d'un fichier dans notre cas.

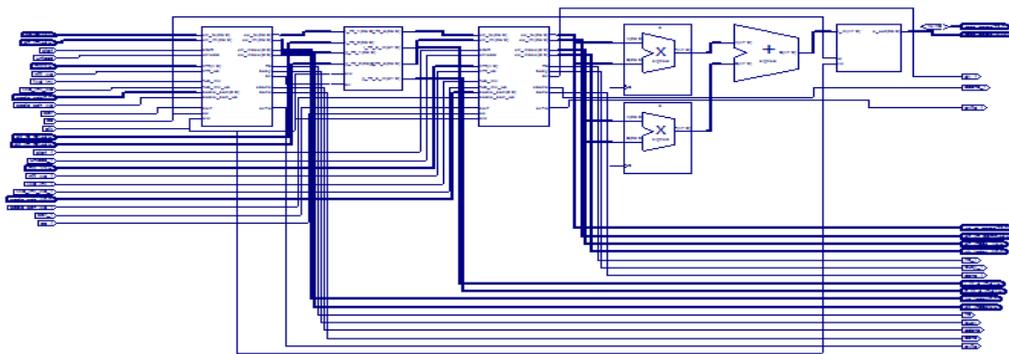


Figure 5.3 : Circuit qui réalise le filtrage adapté

Le filtre de compression implémenté est cadencé avec une fréquence d'horloge maximale de 70.242 MHz. Le tableau 5.2 résume les contraintes temps liées à l'implémentation.

GRANDEUR	DEFINITION
Circuit FPGA	VIRTEX 4vfx100ff1152
Période minimale	14.237 ns
Fréquence maximale	70.242 MHz
Temps nécessaire après le top d'horloge	16.432 ns
Temps nécessaire avant top d'horloge	15.151 ns

Tableau 5.2 : Contraintes temps réel

Paramètres de simulation au niveau de l’FPGA

GRANDEUR	DEFINITION
Fréquence d’échantillonnage	1.28 MHz
Nombre de point de la FFT	Nt=1024
Retard lié à la cible	400 $\mu$ s
Durée du signal avant compression	T=100 $\mu$ s
Codage Frank + COSTAS	fs=2.56 MHz retard 200 $\mu$ s

Tableau 5.3 : Paramètres de simulation au niveau de l’FPGA

Après génération et simulation de la forme d’onde sous MATLAB, on exporte le signal vers le logiciel MODELSIM. On réalise l’opération de filtrage. On fait l’acquisition du résultat, on l’exporte sous MATLAB et on trace la réponse en dB.

Dans la partie qui suit, nous présentons les tracés des sorties du filtre de compression implanté avec différentes formes d’ondes. Le digramme qui suit illustre les formes d’onde utilisées pour la simulation.

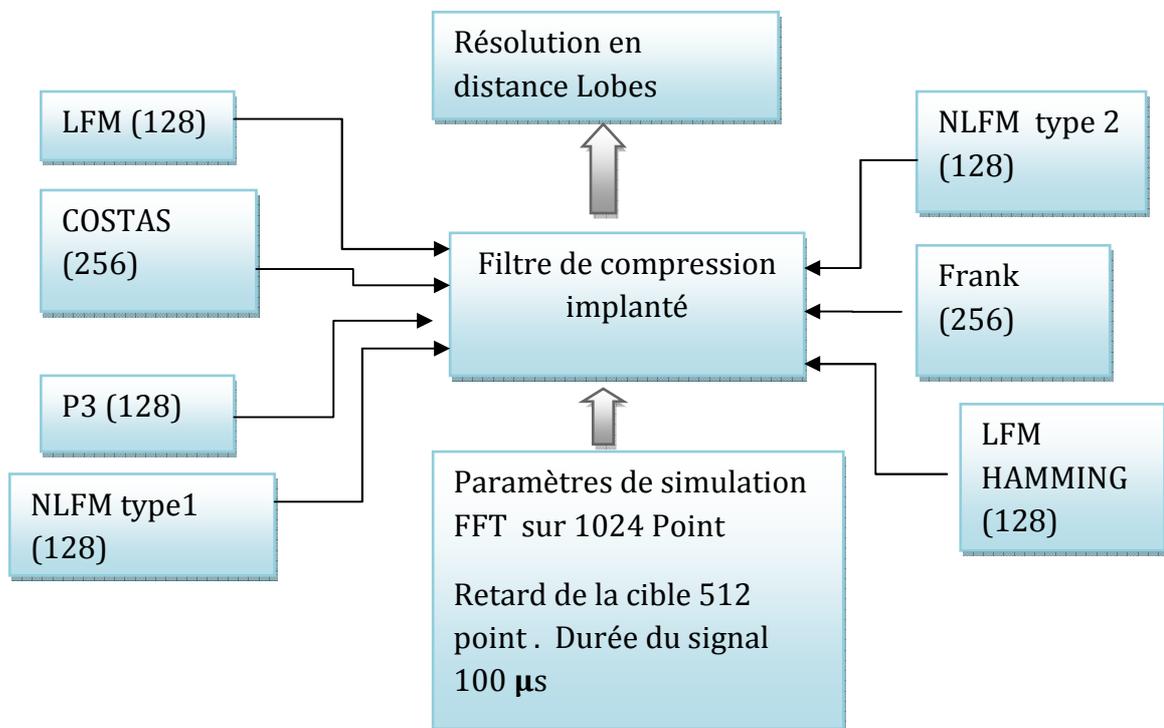


Figure 5.4 Paramètre de simulation au niveau de l’FPGA

## 5.2 Réponse du filtre adapté pour un codage linéaire de fréquence

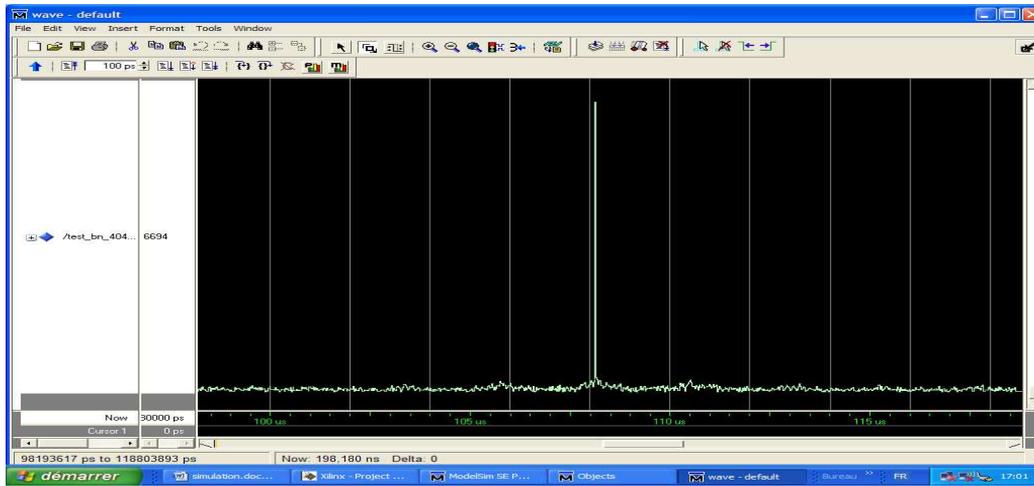


Figure 5.5 Réponse du filtre adapté pour une LFM

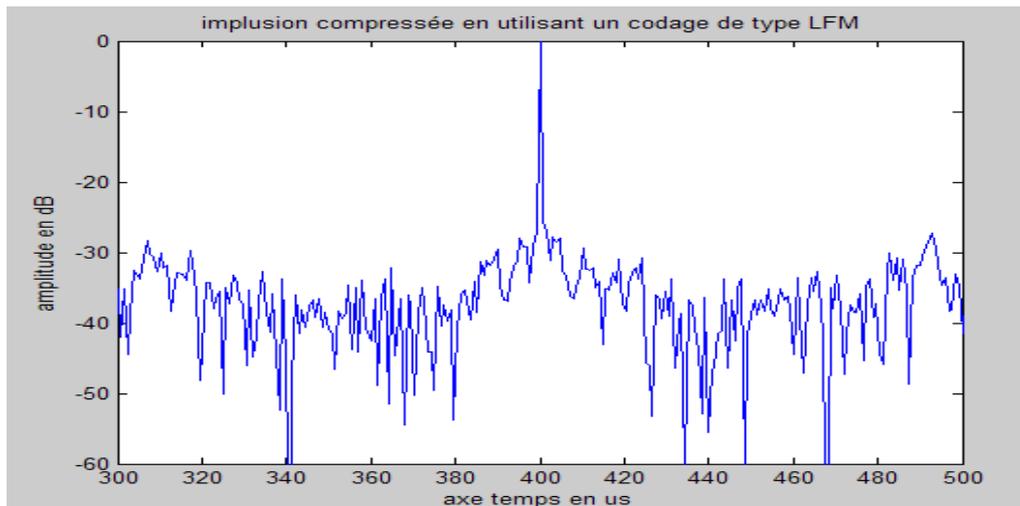


Figure 5.6 Réponse du filtre adapté pour une LFM en dB

Grandeur	Forme d'onde
Résolution en distance	$\tau_{1\text{ nul}} = 0.78 \mu\text{s}$
Premier lobe secondaire	$n_{\text{lobe } s1} = -28 \text{ dB}$
Le plus grand lobe	$n_{\text{lobe } sx} = -28 \text{ dB}$
Location du premier lobe	$\tau_{1\text{ sl}} = 3.1 \mu\text{s}$

La figure 5.5 montre le résultat de la compression sous MODELSIM et la figure 5.6 représente le même résultat après transfert sous MATLAB.

## 5.3 Codage non linéaire de fréquence de type 1

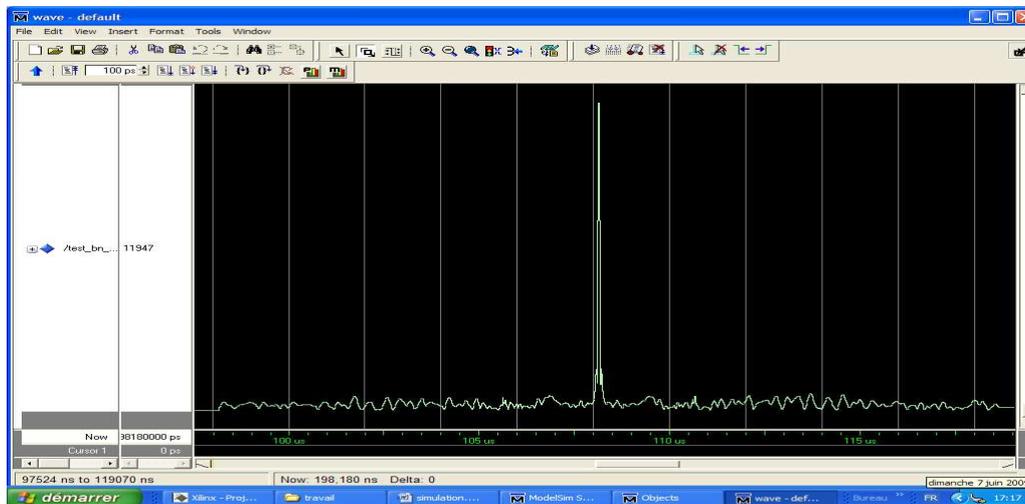


Figure 5.7 Réponse du filtre adapté pour une NLFM de type 1

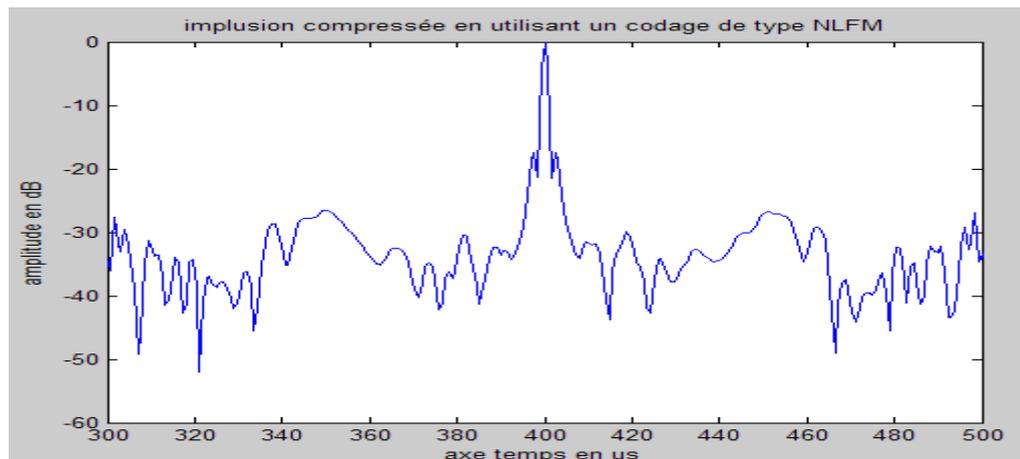


Figure 5.8 Réponse du filtre adapté pour une NLFM de type 1 en dB

Grandeur	Forme d'onde
Résolution en distance	$\tau_{1\text{ nul}} = 5.87 \mu\text{s}$
Premier lobe secondaire	$n_{\text{lobe } s1} = -17.5 \text{ dB}$
Le plus grand lobe	$n_{\text{lobe } sx} = -17.5 \text{ dB}$
Location du premier lobe	$\tau_{1\text{ sl}} = 2.5 \mu\text{s}$

La figure 5.7 montre le résultat de la compression sous MODELSIM et la figure 5.8 représente le même résultat après transfert sous MATLAB.

## 5.4 codage non linéaire de fréquence type 2

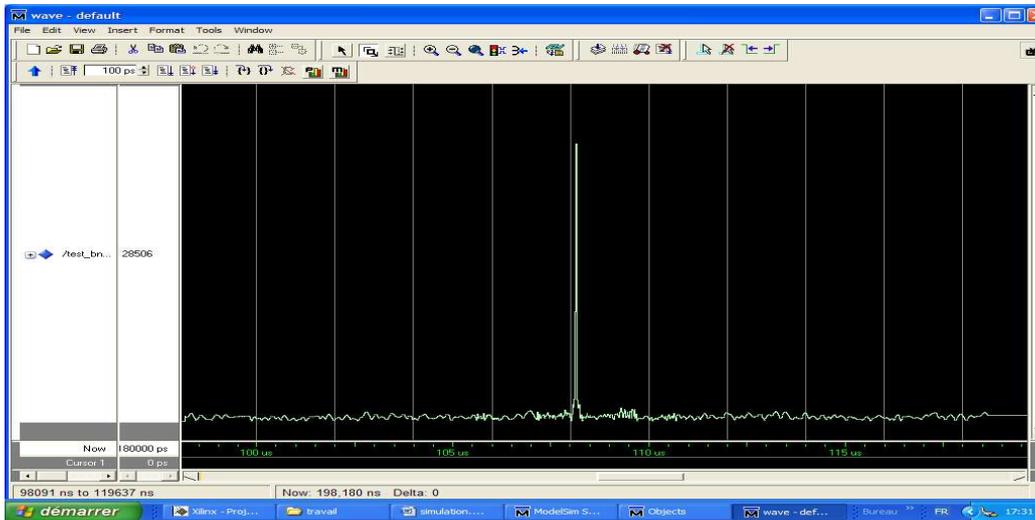


Figure 5.9 Réponse du filtre adapté pour une NLFM de type 2

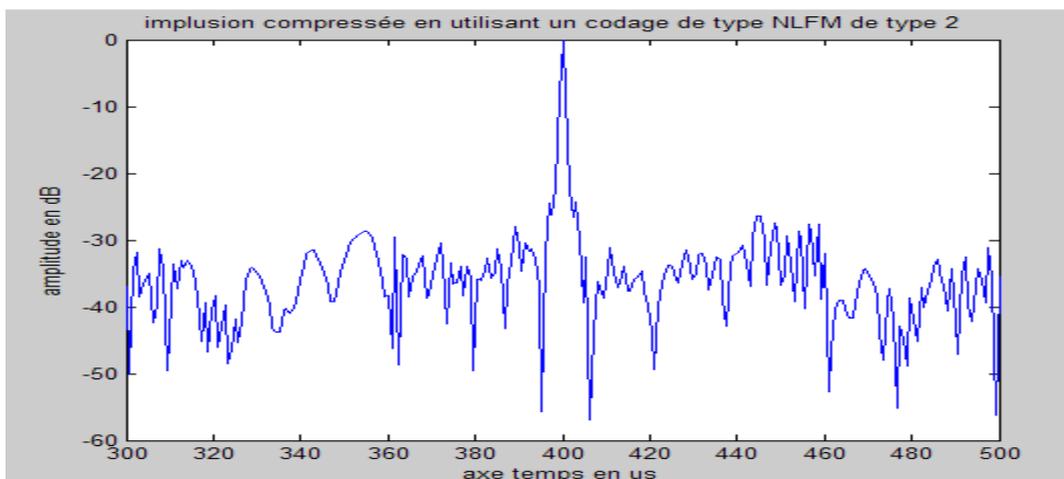


Figure 5.10 Réponse du filtre adapté pour une NLFM de type 2 en dB

Grandeur	Forme d'onde
Résolution en distance	$\tau_{1\text{ nul}} = 3.9 \mu\text{s}$
Premier lobe secondaire	$n_{\text{lobe } s1} = -24.3 \text{ dB}$
Le plus grand lobe	$n_{\text{lobe } sx} = -24.3 \text{ dB}$
Location du premier lobe	$\tau_{1sl} = 3.1 \mu\text{s}$

La figure 5.9 montre le résultat de la compression sous MODELSIM et la figure 5.10 représente le même résultat après transfert sous MATLAB.

## 5.5 Codage linéaire de fréquence avec fenêtrage de type (HAMMING)

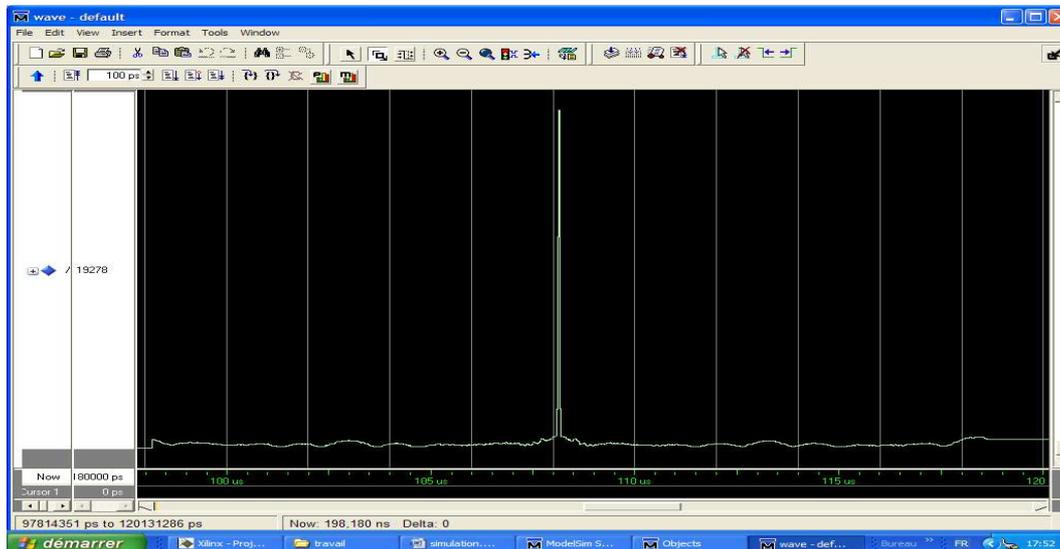


Figure 5.11 Réponse du filtre adapté pour une LFM+HAMMING

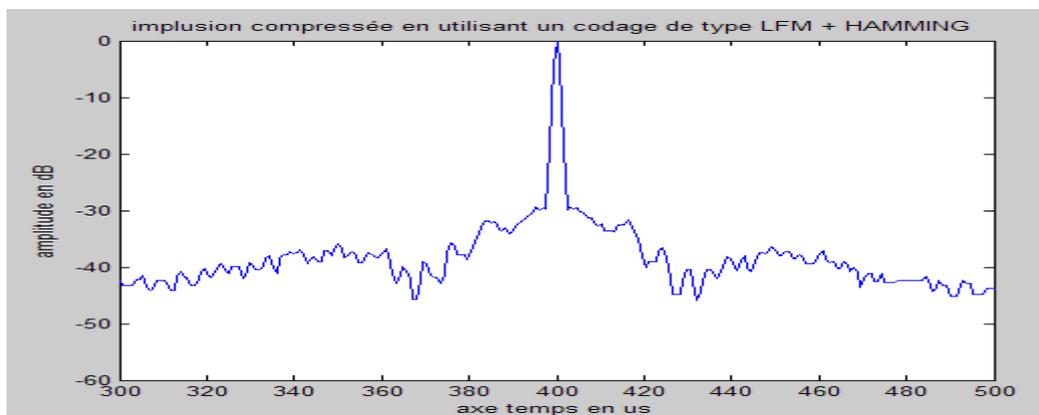


Figure 5.12 : Réponse du filtre adapté pour une LFM+HAMMING en dB

Grandeur	Forme d'onde
Résolution en distance	$\tau_{1\text{ nul}} = 2.5 \mu\text{s}$
Premier lobe secondaire	$n_{\text{lobe } s1} = -29.8 \text{ dB}$
Le plus grand lobe	$n_{\text{lobe } sx} = -29.8 \text{ dB}$
Location du premier lobe	$\tau_{1\text{ sl}} = 3.1 \mu\text{s}$

La figure 5.11 montre le résultat de la compression sous MODELSIM et la figure 5.12 représente le même résultat après transfert sous MATLAB.

## 5.6 Réponse du filtre adapté pour un codage de type CHIRPLIKE

## Simulation du codage de type P3

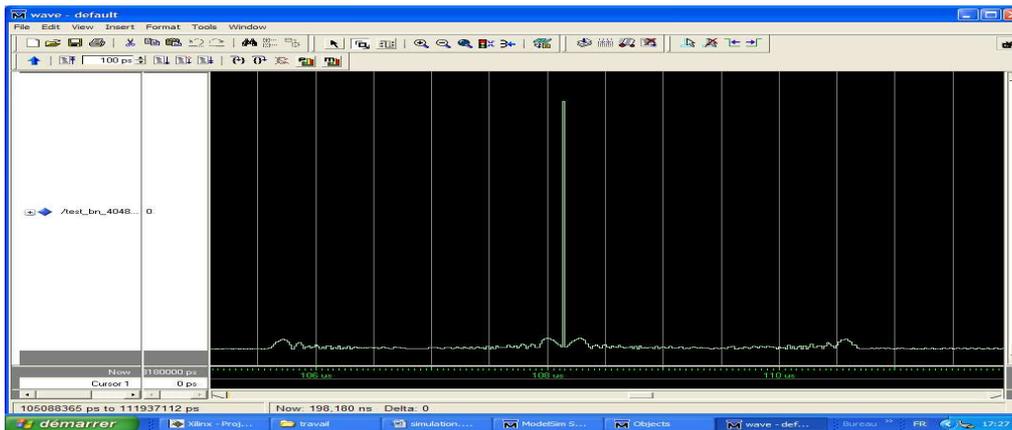


Figure 5.13 Réponse du filtre adapté pour le code p3

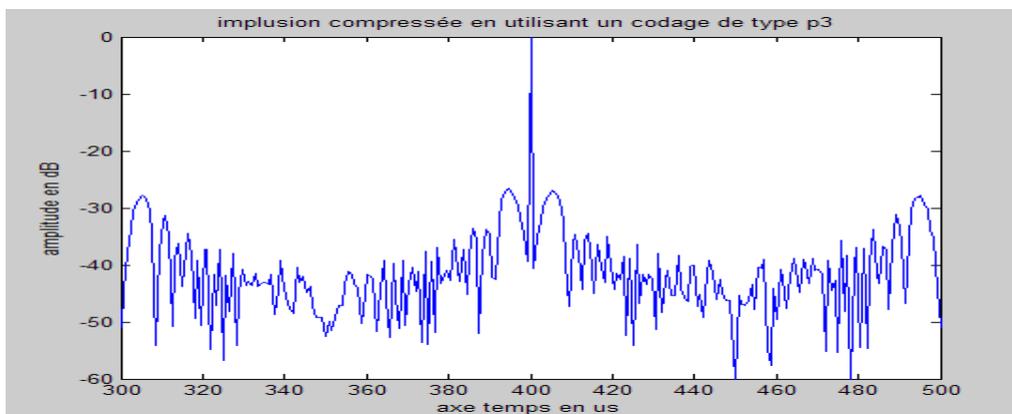


Figure 5.14 Réponse du filtre adapté pour le code p3 en dB

Grandeur	Forme d'onde
Résolution en distance	$\tau_{1\text{ nul}} = 0.78 \mu\text{s}$
Premier lobe secondaire	$n_{\text{lobe } s1} = -26.6 \text{ dB}$
Le plus grand lobe	$n_{\text{lobe } sx} = -26.6 \text{ dB}$
La location du premier lobe	$\tau_{1\text{ sl}} = 5.5 \mu\text{s}$

La figure 5.13 montre le résultat de la compression sous MODELSIM et la figure 5.14 représente le même résultat après transfert sous MATLAB.

## Simulation du codage de type Frank

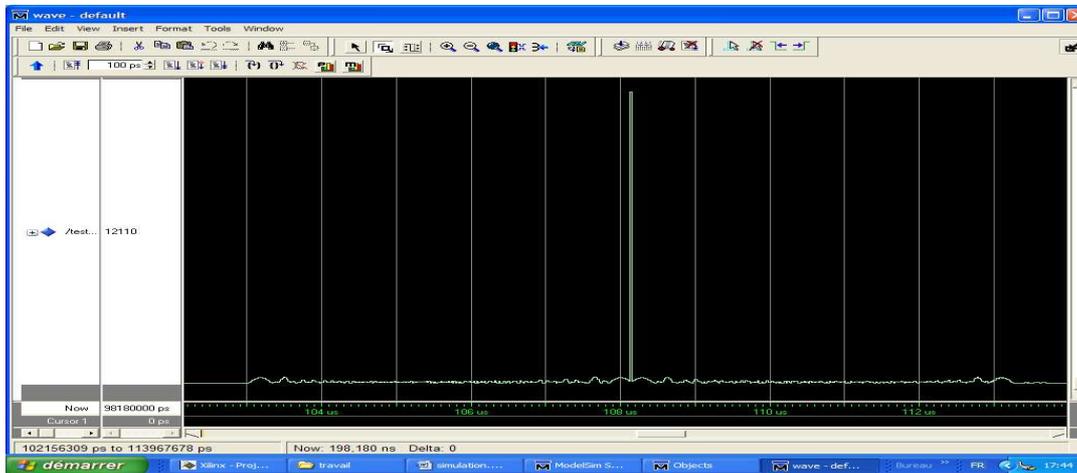


Figure 5.15 Réponse du filtre adapté pour un codage de Frank

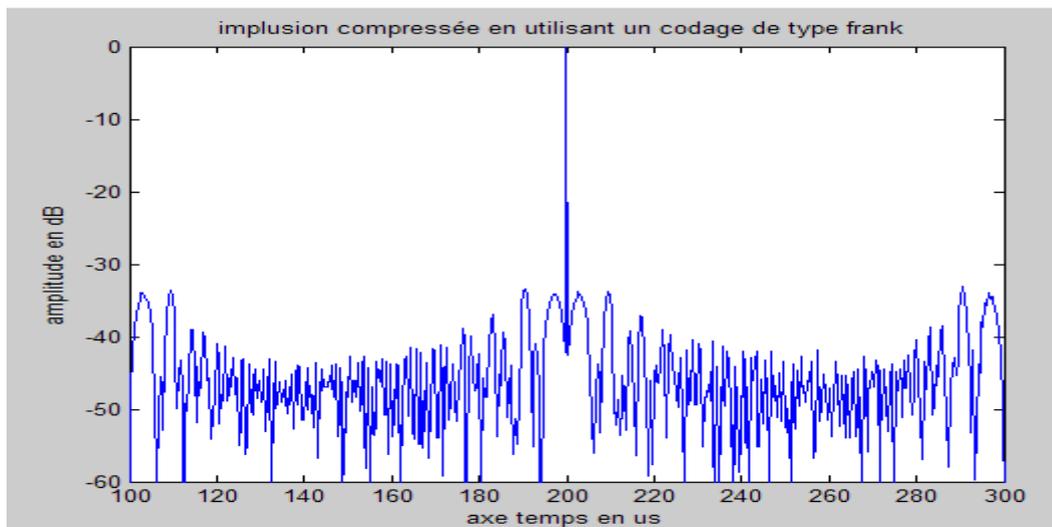


Figure 5.16 Réponse du filtre adapté pour un codage de Frank en dB

Grandeur	Forme d'onde
Résolution en distance	$\tau_{1\text{ nul}} = 0.40 \mu\text{s}$
Premier lobe secondaire	$n_{\text{lobe } s1} = -34 \text{ dB}$
Le plus grand lobe	$n_{\text{lobe } sx} = -33.5 \text{ dB}$
La location du premier lobe	$\tau_{1\text{ sl}} = 2.40 \mu\text{s}$

La figure 5.15 montre le résultat de la compression sous MODELSIM et la figure 5.16 représente le même résultat après transfert sous MATLAB.

## 5.7 Réponse du filtre adapté pour un codage de type COSTAS

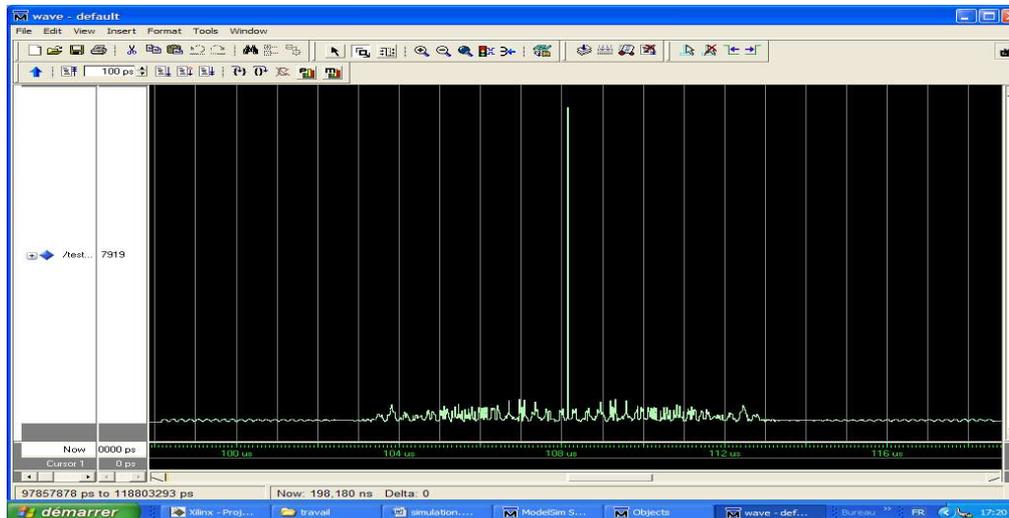


Figure 5.17 Réponse du filtre adapté pour un codage de COSATS

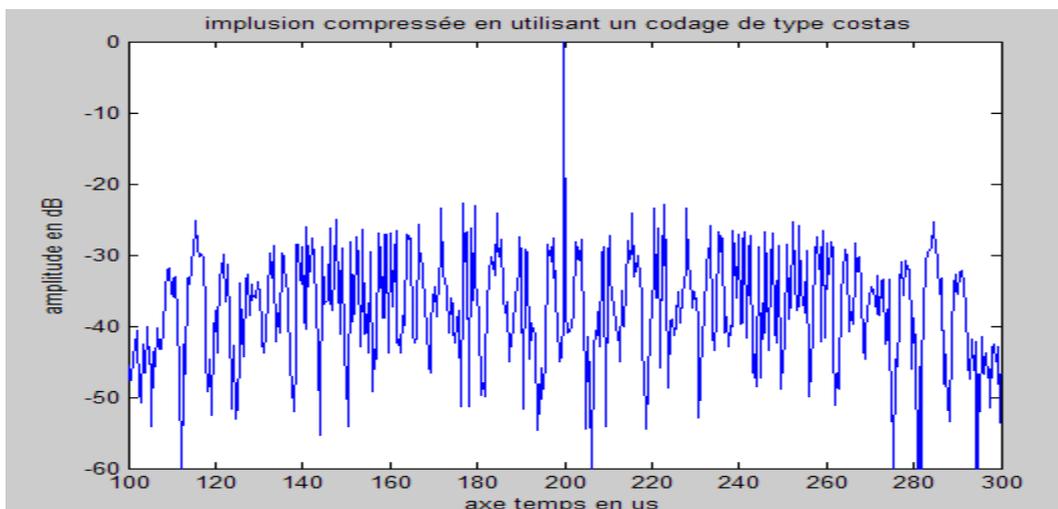


Figure 5.18 Réponse du filtre adapté pour un codage de COSATS en dB

Grandeur	Forme d'onde
Résolution en distance	$\tau_{1\text{ nul}} = 0.487 \mu\text{s}$
Premier lobe secondaire	$n_{\text{lobe } s1} = -27.6 \text{ dB}$
Le plus grand lobe	$n_{\text{lobe } sx} = -22.8 \text{ dB}$
Location du premier lobe secondaire	$\tau_{1\text{ sl}} = 2.2 \mu\text{s}$

La figure 5.17 montre le résultat de la compression sous MODELSIM et la figure 5.18 représente le même résultat après transfert sous MATLAB.

### 5.8 Récapitulatif des résultats de simulations

<b>Modulation</b>	<b>Résolution en distance</b>	<b>Lobes secondaire</b>
LFM (128)	$\tau_{1\text{ nul}} = 0.78 \mu\text{s}$	$n_{\text{lobe } s1} = -28 \text{ dB}$ $n_{\text{lobe } sx} = -28 \text{ dB}$
NLFM type 1 (128)	$\tau_{1\text{ nul}} = 5.87 \mu\text{s}$	$n_{\text{lobe } s1} = -17.5 \text{ dB}$ $n_{\text{lobe } sx} = -17.5 \text{ dB}$
NLFM type 2(128)	$\tau_{1\text{ nul}} = 3.9 \mu\text{s}$	$n_{\text{lobe } s1} = -24.3 \text{ dB}$ $n_{\text{lobe } sx} = -24.3 \text{ dB}$
LFM + HAMMING	$\tau_{1\text{ nul}} = 2.5 \mu\text{s}$	$n_{\text{lobe } s1} = -29.8 \text{ dB}$ $n_{\text{lobe } sx} = -29.8 \text{ dB}$
COSTAS (256)	$\tau_{1\text{ nul}} = 0.487 \mu\text{s}$	$n_{\text{lobe } s1} = -27.6 \text{ dB}$ $n_{\text{lobe } sx} = -22.8 \text{ dB}$
P3 (128)	$\tau_{1\text{ nul}} = 0.78 \mu\text{s}$	$n_{\text{lobe } s1} = -26.6 \text{ dB}$ $n_{\text{lobe } sx} = -26.6 \text{ dB}$
Frank (256)	$\tau_{1\text{ nul}} = 0.4 \mu\text{s}$	$n_{\text{lobe } s1} = -34 \text{ dB}$ $n_{\text{lobe } sx} = -33.5 \text{ dB}$

Tableau 5.4 : Récapitulatif des résultats de la simulation sous FPGA

### Conclusion

La simulation des résultats d'implémentation sur FPGA du filtre adapté fait ressortir les mêmes conclusions obtenues avec la simulation sous MATLAB. Le codage de phase présente la meilleure résolution et si on avait à choisir un codage de phase, le code de Frank présente le plus faible niveau de lobes secondaires. On peut expliquer cette caractéristique par sa propre nature, le code de Frank appartient à la famille des codes parfaits, sa fonction de corrélation périodique ne présente aucun lobe secondaire.

L'étude du codage de forme d'onde présente une importance capitale pour le bon fonctionnement du système Radar, il nous permet d'atteindre les exigences de tous les types de radar, du radar de route jusqu'au radar spatial. Grâce au codage on peut avoir n'importe quelle résolution exigée par l'application dans la mesure où la technologie le permet.

Pour vérifier l'exactitude de cette résolution, on utilise la fonction d'ambiguïté, elle nous permet de voir l'effet de la compression d'impulsion.

On a remarqué que le problème de la compression est la présence de lobes secondaires sur le plan temps fréquence, ce type de lobes peut fausser la détection.

Pour résoudre ce problème, on a deux solutions :

- Utilisation des fenêtres de pondération (HAMMING)
- Intégration d'un codage non linéaire de fréquence

La première solution engendre une désadaptation au niveau du récepteur et une perte au niveau du rapport signal sur bruit.

La deuxième solution ne crée pas de désadaptation et présente la solution optimale pour réduire le niveau des lobes sans atténuer le rapport signal sur bruit.

L'implémentation sur FPGA a ajouté à ce travail une touche pratique ainsi qu'une mesure des contraintes temps réel.

Les perspectives dans le domaine du codage de forme d'onde sont illimitées. L'intégration de forme d'onde périodique ajoute une diversité importante et définit les futures applications.

L'outil qui permet d'étudier les caractéristiques de ce type de codage est la fonction d'ambiguïté périodique.

L'interface sous MATLAB permet la simulation de la fonction d'ambiguïté périodique et la mise en pratique de la simulation de ce type de forme d'onde.

L'étude des performances des différents codes nous a permis de mieux cerner l'influence des codes utilisés sur le comportement du filtre de compression par rapport à la résolution (largeur du lobe principal), le niveau des lobes secondaires et l'insensibilité au Doppler. On tient à noter qu'un code tire son utilité par rapport à un autre code en fonction de l'application ou du besoin. Ainsi ce travail a permis d'avoir une approche élaborée quant aux choix de formes d'ondes pour la réalisation

du filtre de compression. Néanmoins on tient à préciser que ce travail a pour objectif de faire une synthèse sur les formes d'ondes usuelles mises en œuvre pour la réalisation de filtre de compression radar et la comparaison des résultats d'implémentation de filtre de compression sur circuit FPGA avec ceux obtenus par le logiciel Matlab. Ce qui constitue une étape préliminaire pour entamer dans le même contexte des travaux futurs qui permettent de quantifier les performances du filtre de compression en terme de dérive due au Doppler moyennant l'utilisation des outils temps-fréquence plus élaborés et de réaliser une implémentation sur circuit FPGA d'un package de filtre de compression avec une diversité de forme d'onde.

- [1] Nadav Levanon ,and Eli Mozeson , Radar signals , John Wily & Sons Hoboken – New Jersey, 2004
- [2] Merrile Skolnik, Introduction to Radar systems, Mc Grow-Hill Book Company Second Edition, 1980-1962
- [3] Pr Adel Belouchrani, Wave form radars, Ecole nationale polytechnique, 2006-2007
- [4] Alan v Oppenheim, Digital time signal processing, Prentice-Hall, 1999-1989
- [5] U.Mayer-Baese , Digital signal processing with field programmable Gate Arrays, Springer, 2001
- [6] Kunt , Traitement numérique des signaux, Dunod, 1980
- [7] R.Bente Jac , Techniques du Radar classique , Masson Edition, 1992
- [9] L.thourel, Initiation aux techniques modernes des radars , Cepadues Edition, mai 1982
- [10] Michel H. Carpentier , Radars concepts nouveaux, par Dunod Paris 1966
- [11] John G.Proakis, Masoud Salehi , Communication systems Engineering, Prentice èHall, New Jersey ,1994
- [12] A.BRUCE Carlson , Paul B.Crilly , Janet C.Rutledge , Communication Systems, McGraw-Hill ,2002
- [13] André Quiniquis , Le traitement du signal sous Matlab, HERMES Science Publication Paris 2000
- [14] R.AIRIAU –J.-M.BERGE, VHDL, Langage , Modélisation ,Synthèse , Presses polytechniques 1990,1998
- [15] Alain Vachoux , Xemics , Modélisation des systèmes Numériques Intégrés introduction à VHDL , Ecole polytechnique fédérale de Lausanne , 1989,1999
- [16] Joseph W. Ianniello ,Frank D.Stroili , AN ADVACED PULSE COMPRESSION PROCESSOR , 1990 IEEE