République Algérienne Démocratique et Populaire
Ministère de l'enseignement Supérieur et de la Recherche Scientifique

Université Mouloud Mammeri Tizi-Ouzou

Faculté de Génie électrique et d'Informatique

Mémoire de master

Domaine : sciences et Technologie
Spécialité : Génie Electrique
Option : télécommunication et réseaux

Thème

Implémentation VHDL d'une chaîne de communication numérique : application sur la parole

Proposé et dirigé par :
Mr. LAGHROUCHE.M

Présenté par :
Mr. FETTAL WASSIM
Mr. BOUKHENFRA RIYAD

C . promoteur :
BENLMAKHLOUF. T

Promotion 2010-2011

Remerciement

Tout d'abord, Nous remercions le Dieu tout puissant de nous avoir donné le savoir et l'opportunité de pouvoir poursuivre nos études et de choisir un métier aussi noble.

Nous tenons à exprimer tous nos vifs Remerciements et notre profonde Gratitude à notre

Promoteur Mr LAROUCHE.Mourad ainsi qu'a Mr BENLMAKHLOUF.Tahar pour leurs disponibilités, aide et bonne humeur durant tout le déroulement de tout ce projet. Grâce à leurs dévouements, conseils scientifiques et suivis, nous avons pu mener à terme notre travail. Encore une autre fois, MERCI.

Nous adressons également nos vifs remerciements et notre profonde gratitude :

Au président et aux membres du jury pour l'honneur qu'ils nous font, en acceptant de

Juger notre travail.

A l'ensemble du personnel de l'université U.M.M.T.O qui ont participés à notre formation de prés ou de loin.

Enfin, nous tenons à remercier tous nos enseignants et tous ceux qui ont contribué, de prés ou de loin, à la réalisation de ce projet.

Dédicaces

Je dédie ce modéste travail en premier lieu

A mes chére parents les quelles je serai toujours très reconnaissant.

A mes trés chérs fréres et sœurs.

A notre promotre.

A tous mes amis :Ryad, Hossin, Mohamed, Karim, Said, Sofiane,Hamza,takfrines,fouzi,Lilya….

A toute la promotion 2010/2011.

A tous cuex qui me connaissent.

Ainsi qu'à tous les enseignants qui ont assuré mon cursus scolaire du primaire à l'université.

FETTAL WASSIM

Dédicaces

Je dédie ce modéste travail en premier lieu

A mes chére parents les quelles je serai toujours très reconnaissant.

A mes trés chérs fréres.

A tous mes oncles et tantes,

A notre promotre.

A tous mes amis :Wassim, Karim, Said, Mohamed,Hamza, Islam, Smail, Kahina,takfrines,fouzi.

A toute la promotion 2010/2011.

A tous cuex qui me connaissent.

Ainsi qu'à tous les enseignants qui ont assuré mon cursus scolaire du primaire à l'université.

BOUKHENFRA RIYAD

Sommaire

Liste Des Figures

Liste Des Tableaux

Liste Des Abréviations

AES	: Advenced Encryption Standard
ADPCM	: Adaptative Differentiel Pulse-Code Modulation
AR	: AutoRegressif
ARMA	: Modele Regressif a Moyenne Ajusté
ASIC	: Application Spécific Integrated Circuit
CELP	: Code Excited Linear Prediction
CLB	: Configurable Logic Bloc
CPLD	: Complex Programmable Logic Device
DES	: Data Encryption Standard
FFT	: Fast Fourier Transform
PCM	: Pulse Code Modulation
FPGA	: Fieled Programmable Gate Array
IOB	: Input Output Bloc
LUT	: Look Up Table
LPC	: Linear Prediction Coding
NIST	: National Institute of Standard and Technology
PLA	: Programmable Logic Array
PLD	: Programmable Logic Device
RSA	: Rivest-Shamir-Adleman
SIFT	: Simplified Inverse Tracking Algorithme
STFT	: Short Time Fourier Transform
TFD	: Transformé de Fourier Discrète
VHDL	: Very High Description Language

Introduction Générale

Un besoin souvent invoqué dans les échanges d'information est l'augmentation des débits dans le souci de l'authentification des entités communicants et celui d'assurer une certaine confidentialité des messages échangés.

Beaucoup de méthode de codage et de cryptage dans le domaine de la parole ont été introduits pour assurer la rapidité et la sécurisation dans un tel échange. Or coder numériquement des informations complexes exige des temps de traitement pénalisant.

L'objet de notre étude se situe dans le domaine (particulièrement important pour les systèmes de télécommunication modernes) du codage et le cryptage du signal parole.

Pour transmettre un signal de parole selon des normes bien définie dans notre mémoire des chaînes de communication analogiques ont été utilises presque exclusivement jusqu' à une époque récent. A cause des perturbations et des bruits apparaissant inévitablement dans le canal de transmission, le signal reconstruit au récepteur n'être qu'une réplique exacte du signal émis.

Mais avec l'agrandissement de la taille des réseaux de télécommunication, des problèmes d'interférence et de saturation de bande passante sont apparus ce qui a permet de mettre des nouveaux systèmes de transmission efficace capable de réduire la largeur de bande occupée et surtout la possibilité de correction d'erreur.

Mais lors de la transmission des signaux très redondent (riche d'information) telle que la parole, si on le numérise la séquence binaire obtenue a une taille énorme et comme le traitement se fait avec des calculateurs numérique ou le paramètre « temps d'exécution » joue un rôle très important pour une transmission en temps réelle, pour cela plusieurs norme de compression et de codage sont mis pour réduire le flux de donnée à traiter.

Un autre paramètre qu'on doit prendre en considération dans le cas de la réalisation pratique est l'échauffement de calculateur, ce dernier est capable de modifié la structure physique de calculateur par conséquence sa réponse au signal d'entrée, Ce qui augmente les erreurs et dégrade la fiabilité de système.

De plus, les échanges de l'information doivent rester secrètes ou confidentielles, ceci conduit inévitablement à la cryptographie, alors qu'avant elle n'avait pour rôle que de protéger un texte écrit

Aujourd'hui, la cryptographie étend dans différents domaines tels que la téléphonie, le stockage des donnes ….etc. La méthode de cryptage été introduite après le codage pour rendre la parole totalement inintelligible et délicat.

La possibilité de réduire le débit et supprimer les redondances dans le signal parole par le codage LPC permet d'augmenter la robustesse et la sécurité.

Les puissances de calcule disponibles étant importants, grâces au développement du circuit FPGA, des modèles mathématique très proches du système phonatoire humain ont pu être mis au point et utilisés dans des algorithmes de codage de plus en plus complexes. Bien

que les progrès technologiques jusqu' a aujourd'hui aient été considérables, la complexité algorithmique a encore un impact direct sur le coût d'un codeur/décodeur

Ainsi, toutes les opérations de codage du signal de parole doivent se faire respectivement un certain nombre de contraintes la plus importantes étant la qualité de restitution du signal vocal.

Pour présenter ce travail des généralités et des définitions sont présentées dans le premier chapitre, le second porte sur le outil de codage LPC, le troisième chapitre consacré pour la définition de cryptage symétrique AES-RIJINDAEL, le quatrième pour les résultats de simulation et implémentation .

Enfin nous terminerons par une conclusion générale et Annexe.

Chapitre I : GENERALITES SUR LA PAROLE ET LA CHAINE DE COMMUNICATION

I.1. Introduction

Le terme (télécommunications) fut inventé en 1904, et signifie communiquer à distance, le but des Télécommunications est donc de transmettre un signal, porteur d'une information (voix, musique, Image, donnes) d'un lieu a un autre lieu situé à distance.

La possibilité de transmettre un message a toujours été d'une importance essentielle. L'idée n'est pas récente : nuages de fumés (autochtones), feux de bois (Grecs-Perses), torches enflammées (Romains),…etc.

Un grand pas a été effectué durant les deux derniers siècles avec le développement des systèmes De transmission sur câbles, sur ondes hertziennes et satellites et sur fibres optique.

Et grâce a la technologie microélectronique et la puissance de calculs actuel on arrive a réalise des systèmes de communication numérique a une tel performance sur des circuits intégré miniature. Utilise

I.2. La chaîne de communication numérique

Une chaîne de transmission numérique permet de réalise un système de communication qui est chargé de transmettre les informations d'une source vers un récepteur a travers un canal de transmission, La source et le récepteur peuvent être très proches ou éloignés. Par fois ces informations sont privé ou très important à un niveau qu'impose une sécurisation avant la transmission, cote performance, la chaîne de communication numérique est plus performant que la chaîne analogique car elle offre non seulement une protection contre les bruits mais aussi une correction d'erreur tout cela est possible grâce a un étage « codeur canal » qu'ont trouve seulement dans les chaînes de communication numérique. Cependant, tout système de communication dit numérique sont calqués a partir de schéma général de base donne par la figure suivant :

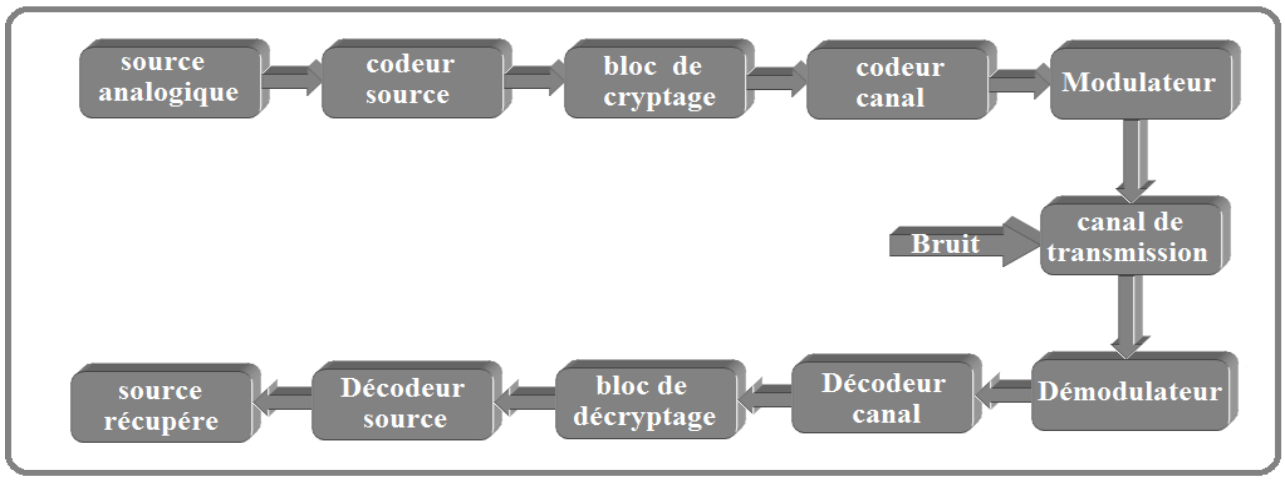


Figure I-1: Schéma synoptique de la chaîne de transmission numérique.

Le schéma de principe d'une chaîne de transmission numérique est représenté sur la Figure I-1.

On peut distinguer : la source de message, le milieu de transmission et le destinataire qui sont des données du problème le codage et le décodage de source, le codage et le décodage de canal, l'émetteur et le récepteur représentent les degrés de liberté du concepteur pour réaliser le système de transmission. Nous allons maintenant décrire de façon succincte les différents éléments qui constituent une chaîne de transmission en partant de la source vers le destinataire.

I.2.1. La source de message

Le message fournie par la source en générale est analogique donc pour le transmettre, il doit être convertie de sa forme analogique vers le numérique en passent par deux étape essentiel :

- l'échantillonnage (discrétisation en temps).
- La quantification ou bien le codage (discrétisation en amplitude).

Par exemple pour numérise le signal de parole (obtenus a la sortie d'un microphone) ont échantillonnant le message analogique puis en quantifiant les échantillons obtenus. Chaque échantillon quantifié est ensuite codé sur 'm' bits.

I.2.2. Le codage et décodage de source

Consiste à supprimer la redondance contenue dans les messages de la source d'information. Il peut être avec ou sans pertes d'information. La compression avec pertes vise les signaux numérisés (image, audio ou vidéo). Après numérisation et codage, la source de message numérique est caractérisée par son débit binaire D. défini comme le nombre d'éléments binaires émet par unité de temps.

Donnons quelques exemples numériques de débit binaire en sortie de sources numérisées et codées. La numérisation du signal de parole, préalablement limité à la bande 300-3400 Hz en téléphonie, est réalisée en échantillonnant ce signal à la fréquence de 8 kHz, puis en codant les échantillons quantifiés sur m =8 bits, Ainsi après numérisation, le signal de parole est transformé en une source numérique ayant un débit binaire de 64 k bit/s ; ce codage de la parole a pris (improprement) le nom de « codage MIC » (modulation par impulsion codée). Avec un codage de source plus élaboré, ce débit de 64 k bit/s peut être réduit à 32kbit/s sans dégradation de la qualité subjective de la parole, des algorithmes permettant d'atteindre des débits de 16 et 8 k bit/s ont même été adoptés récemment par les organismes internationaux normalisation .Pour le radiotéléphone cellulaire numérique européen (GSM), ce débit a été ramené à 13 k bit/s. [1]

Codage Cependant, le décodeur de source effectue la tâche inverse du codeur de source en Convertissant la sortie binaire du décodeur de canal en une séquence de symboles. Dans le cas idéal, cette séquence est identique à la sortie de la source (canal idéal sans perturbation).

I.2.3. Codage et décodage canal

Le codage canal est appelle codage détecteur d'erreur ou bien code correcteur d'erreur est une fonction spécifique des transmissions numériques n'a pas son équivalent en transmission analogique des transmissions numériques, leur rôle principale dans une chaîne de communication numérique est consiste à insérer dans le message des éléments binaires dits de redondance suivant une loi donnée (convolutif……..). Cette opération conduite donc à une augmentation du débit

binaire de la transmission; Le décodeur de canal, qui connaît la loi de codage utilisée à l'émission, vient vérifier si cette loi est toujours respectée en réception. Si ce n'est pas le cas, il détecte la présence d'erreurs de transmission qu'il peut corriger sous certaines conditions. Donc l'objectif d'un codeur de canal est d'établir un système de contrôle des erreurs par un nouveau codage du message. Une chose très important il faut dire que Le codage de canal n'est possible que si le débit de la source binaire est inférieur à la capacité du canal de transmission. [1]

Pour le décodage de canal en fait le processus inverse du codage via plusieurs algorithmes

(Viterbi……..)

I.2.4. Cryptage et décryptage

Ce bloc la, est le plus important dans une chaîne de transmission sécurise.

Le but d'un système cryptographique est de chiffrer un texte en clair (message) en un cryptogramme au moyen d'une clé, se cryptogramme est en suite transmis à son destinataire sur le canal. Le destinataire légitime doit pouvoir déchiffrer le cryptogramme à l'aide de la clé pour obtenir le texte en claire. [1]

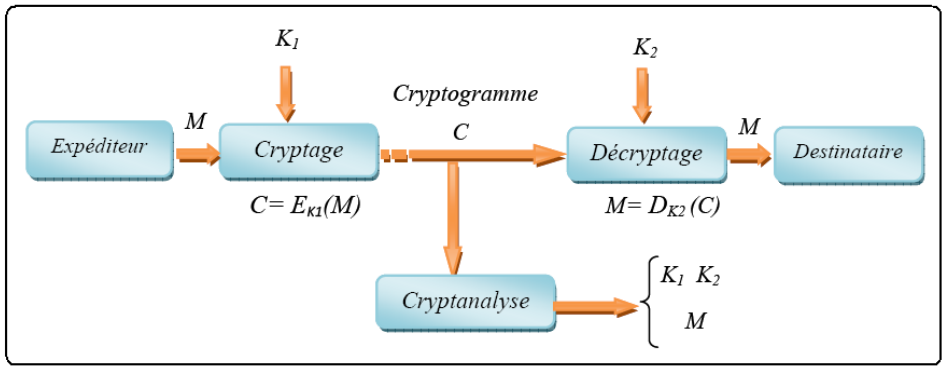


Figure I-2: Modèle de Shannon pour le secret.

Ils existent plusieurs types de cryptage

Un système de chiffrement est dit :

- symétrique quand il utilise la même clé pour chiffrer et déchiffrer.
- asymétrique quand il utilise des clés différentes : une paire composée d'une clé publique, servant au chiffrement, et d'une clé privée, servant à déchiffrer. Le point fondamental soutenant cette décomposition publique/privée est l'impossibilité calculatoire de déduire la clé privée de la clé publique.

Les méthodes les plus connues sont le DES, le Triple DES et l'AES pour la cryptographie symétrique, et le RSA pour la cryptographie asymétrique, aussi appelée cryptographie à clé publique.

L'utilisation d'un système symétrique ou asymétrique dépend des tâches à accomplir. La cryptographie asymétrique présente deux intérêts majeurs : elle supprime le problème de transmission sécurisée de la clé, et elle permet la signature électronique. Elle ne remplace cependant pas les systèmes symétriques car ses temps de calcul sont nettement plus longs

I.2.5. Modulation numérique

La modulation numérique pour fonction d'adapter le signal à transmettre au canal de transmission. Elle consiste a moduler la phase, la fréquence, l'amplitude, d'un ou plusieurs porteuses centre sur la bande de fréquence de canal on appelle une modulation linéaire les modulations qui translate le spectre en bande de base vers la fréquence de la porteuse sans modifie l'allure de ce spectre.

On appelle une modulation non linéaire tout celles qui change la forme de spectre.

La modulation numérique permet également de partager d'un même canal allouent des bandes de fréquence au différant utilisateurs.

Les types de modulation numérique:

Il est possible de classifie les modulations numérique de différant façon ;

Les modulations linéaire : modulation d'amplitude (MDA, ASK), de phase (MDP,PSK)

Les modulations non linéaire : modulation de fréquence (MDF, FSK)

I.2.6. Canal de transmission

C'est le support physique dont la quelle on peut véhiculer l'information que l'on souhaite de transmettre, l'inconvénient major de cette canal c'est le bruit quelle que soit additif, blanc, gaussien …..Etc. car il introduit toujours des modifications qui peuvent dégrader la qualité de système de communication.

Selon la nature du canal, les signaux sont de nature différente :

Atmosphère : onde électromagnétique

Câble coaxial : signaux électrique (tensions, courant)

Fibre optique : ondes électromagnétiques optique (lumière visible infrarouge)

Les canaux de transmission peuvent être classés selon l'effet qu'ils ont sur le signal on distingue :

- Canal a bruit additif blanc
- Canal a évanouissement

I.3. Introduction au traitement de la parole

Dés le début de l'informatique, la nécessité d'une communication homme- machine s'est faite sentir. Contrairement à l'homme, la machine apprend à "parler" et à "comprendre" après avoir appris à "lire" et à "écrire".

La parole qui est le support le plus naturel de communication de la communauté humaine fait appel à des processus d'analyse et de synthèse fort complexes. L'information du message parlé réside dans les fluctuations de la pression de l'air engendré, puis émis par l'appareil phonatoire, ces fluctuations constituent le signal vocal.

I.3.1. Aspect physiologique de la phonation

L'organe phonatoire humain se compose essentiellement de trois parties [2]:

- La partie subglottique: constituée les poumons et de la trachée artère, elle assure le flux d'air nécessaire à la phonation.
- La partie glottique ou larynx: est un ensemble très complexe de cartilages, de ligaments et de muscles à la base duquel sont attachées les cordes vocales qui jouent un rôle de valves vis à vis à l'air issu des poumons, libérant ainsi un certain flux d'air représentant l'excitation du conduit vocal.
- La partie supraglotique ou conduit vocal: elle est constituée essentiellement du pharynx et des cavités buccale et nasale.

La figure suivante représente le système phonatoire humain :

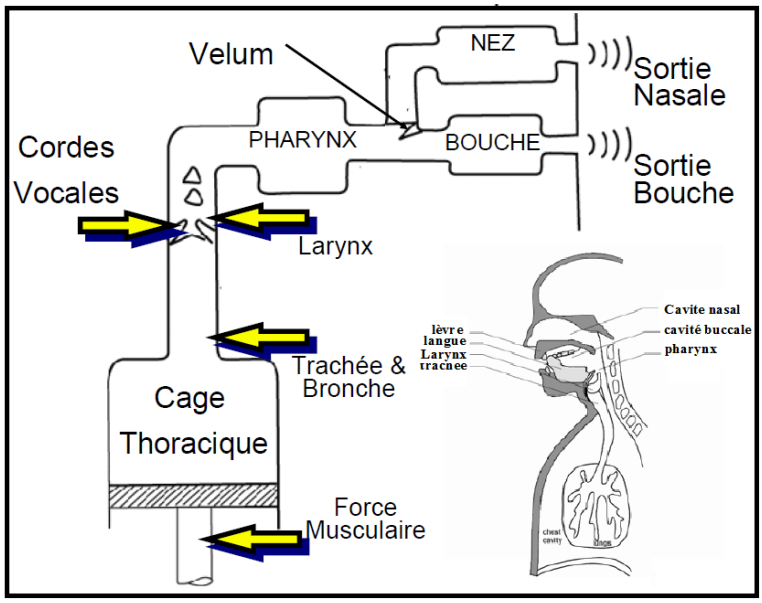


Figure I-3: appareille phonatoire humains.

I.3.2. Production des Sons de parole

Décrire le processus de production de la parole en vue de spécifier le signal ainsi produit nécessite l'acquisition d'un certain nombre de connaissances liées à la complexité du processus de génération. Les sons de parole sont produits soit par la vibration des cordes vocales " une source de voisement", soit par l'écoulement turbulent de l'air dans le conduit

vocal, ou lors du relâchement d'une occlusion de ce conduit "source de bruit". Cette production se fait lors de l'excitation du conduit vocal par l'air expiré contenu dans les poumons, cette excitation se compose de deux sortes:

- Une excitation quasi-périodique : où les cordes vocales entrent en vibration sous l'effet de l'air et l'action des différents muscles du larynx.
- Une excitation apériodique: les cordes vocales n'entrent pas en vibration, le passage de l'air se fait librement dans le conduit "constriction, occlusion " ce qui donne naissance à un bruit se propageant dans ce conduit.

La figure suivant représente ce quand appelle le model de production de sons de parole (modèle source filtre).

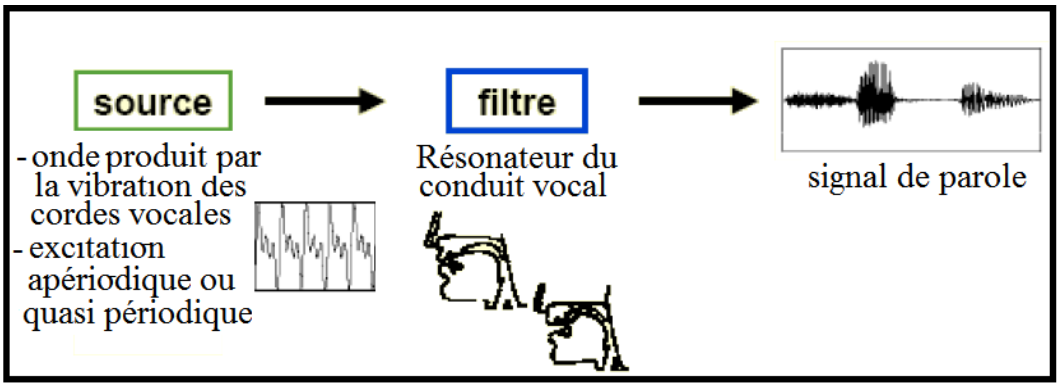


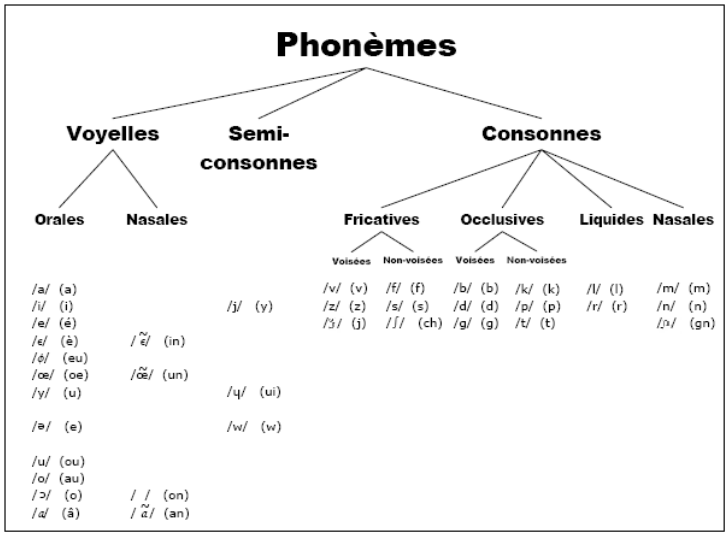
Figure I-4: Modèle source filtre

I.3.3. Aspect fréquentiel et Propriétés des sons de parole

Il existe plusieurs propriétés parmi lesquelles on peut citer l'essentiel :

I.3.3.1. Le phonème

La plupart des langues naturelles sont composées à partir de Sons distincts: les phonèmes. Un phonème est la plus petite unité présente dans la parole et susceptible par sa présence de changer la signification d'un mot. Par exemple 'pari' et 'mari', 'riz' et 'rat'. [3]



I.3.3.2. Le pitch

C'est la fréquence fondamentale de la source d'excitation lorsque le son est voisé. C'est un paramètre très important pour la synthèse de la parole; en effet, l'oreille est très sensible à ses variations. [3]

Le pitch peut varier:

de 80 à 200 Hz pour une voix masculine,

de 150 à 450 Hz pour une voix féminine,

de 200 à 600 Hz pour une voix d'enfant.

I.3.3.3. Les formants

Les formants sont des zones fréquentielles dont l'intensité est renforcée. Chaque voyelle est reconnaissable par l'amplification d'harmoniques déterminés du son laryngé, appelés formants. La composition formantique de chaque voyelle est indépendante de la hauteur de son fondamental. Ainsi, que l'on soit un homme, une femme ou un enfant, on prononce les mêmes voyelles.[4]

- Le 1er formant (F1): La zone formantique de F1 est située entre 250 et 750Hz. Le premier formant F1 correspond à l'aperture de la voyelle (ouverture de la mandibule).
- Le 2ème formant (F2) : La zone formantique de F2 est située entre 750 et 2500Hz. C'est surtout ce deuxième formant qui est nécessaire pour l'intelligibilité du langage, et en particulier dans la zone située autour de 2KHz. Il exprime la position plus ou moins avancée de la langue.
- Le 3ème formant (F3) : Le troisième formant est beaucoup moins caractéristique de la voyelle que le premier et le deuxième, car sa hauteur fréquentielle varie peu pour la majorité des voyelles.

Il est aussi à noter que le troisième formant donne de l'information sur l'arrondissement des lèvres.

La Figure I-5 illustre la fonction de transfert en fréquence du conduit vocal.

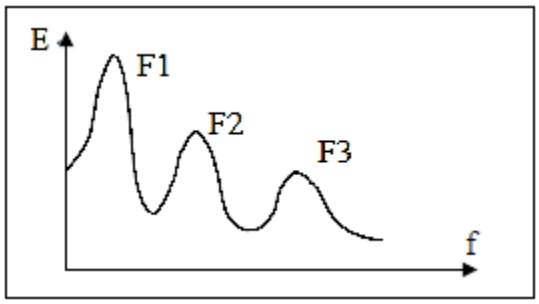


Figure I-5: Représentation de la fonction de transfert de conduit vocal

Les valeurs des premiers et deuxièmes formants permettraient aux auditeurs d'identifier les voyelles orales. Leurs valeurs respectives rendent compte des propriétés du résonateur buccal et du résonateur pharyngal. Ce sont les formants les plus graves et il arrive que le premier formant se confonde avec le fondamental, particulièrement lorsqu'il s'agit de voix de femmes ou d'enfants dont la fréquence naturelle de la voix est plus élevée.

Les voyelles sont souvent représentées positionnées sur un plan, dont les axes sont les formants F1 et F2. Elles tracent alors un triangle dont les extrémités sont occupées par les voyelles "extrêmes", c'est-à-dire [a], [u], [i]. Ce triangle représente également, de manière assez grossière, les positions de la langue dans la bouche selon deux axes [4] (voir Figure I-6) :

- Antérieur à postérieur.
- Fermé à ouvert.

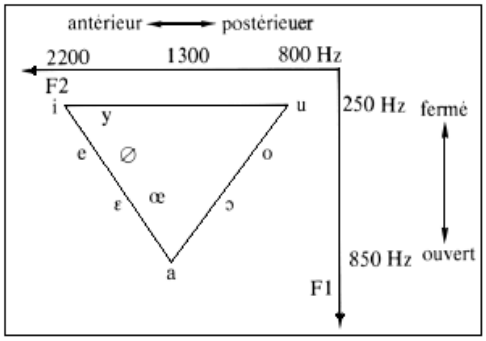


Figure I-6: Représentation des voyelles dans le plan F1 - F2.

- La hauteur: qualité liée à la sensation aigue ou grave que donne un son. Elle dépend de la fréquence du son perçu ainsi que de son intensité. Pour une fréquence donnée, le son le plus intense masque celui de faible intensité. [5]
- Le timbre : c'est la caractéristique de deux sons de même fréquence. Il se caractérise par l'amplitude harmonique qui est propre à chaque son. [5]
- L'intensité : elle est liée à la puissance acoustique. Elle permet de distinguer un son plus ou moins fort, deux Sons de même puissance acoustique et de fréquences différentes ont une intensité différente. Celui de haute fréquence étant plus intense. Ainsi elle donne des informations sur l'amplitude de la voix, celle-ci pouvant être normale chuchotée ou criée. [5]
- La prosodie: évolution de la hauteur, l'intensité et la durée syllabique représente l'évolution de la prosodie. [5]
- La mélodie : évolution de la fréquence fondamentale au cours du temps.[5]
- La dynamique: la variation de l'énergie phonétique d'un individu s'exprime en décibel. La dynamique d'un signal est l'écart entre son maximum et son minimum (DS = Nmax —Nmin) dB. En général l'énergie de la parole varie entre 1/10 et 1/100, soit une dynamique de 40 db. [5]

I.4. Les différents types des Sons de parole

Le conduit vocal est assimilé à un filtre que l'on représente par une fonction de transfert, attaqué en entrée par une source d'excitation périodique ou non, représentant l'alimentation en air phonatoire. La séparation entre la source d'excitation et le filtre simulant le conduit vocal, n'est cependant valable qu'en première approximation, car dans la réalité l'air phonatoire agit aussi sur la forme du conduit vocal et inversement.

- Sons voises : si le mouvement de l'air phonatoire est de nature périodique, le conduit vocal comportera des fréquences de résonance: ce sont les "formants ". Le son résultant sera dit voisé ou sonore, comme c'est le cas des voyelles.
- Sons non voisés: si le signal d'excitation est non périodique, le conduit vocal peut, suivant sa configuration, présenter une réponse faisant apparaître certaines bandes fréquentielles appelées formants de bruit. Le son résultant sera dit non voisé ou sourd. C'est le cas des "consonnes sourdes ".

Une configuration particulière du conduit vocal, attaqué par une source voisée, peut mener à la production de “consonnes voisées “.

I.5. Représentations spectrales du signal de parole

Il existe plusieurs transformé qui permet la représentation spectrale du signal de parole, parmi elles nous citerons :

I.5.1. Spectre obtenu par FFT

Tout son est la superposition de plusieurs ondes sinusoïdales. Grâce à la FFT, on peut isoler les différentes fréquences qui le composent. On obtient ainsi une répartition spectrale du signal

Les valeurs des formants sont calculées automatiquement dans le signal de parole au moyen d'un lissage spectral. [5]

On définie la transformé de Fourier discret ou bien ça version rapide obtenus par l’entrelacement La transformé de Fourier rapide (Faste Fourier transformer) par la relation suivant :

$$X[k]=\sum_{n=0}^{N-1} x(n)e^{-j2\pi\frac{kn}{N}}$$

Avec : k et n=0……N-1

I.5.2. Spectre obtenu par la méthode de prédiction linéaire (LPC)

Le spectre obtenu par LPC est plus lisse et permet ainsi de repérer plus facilement les formants Pour estimer les fréquences des formants, on calcule le spectre d'amplitude correspondant au modèle LPC et on cherche les fréquences correspondant aux pics spectraux.[5]

La Figure I-7 représente le spectre obtenu par la FFT et le spectre obtenu par la prédiction linéaire :

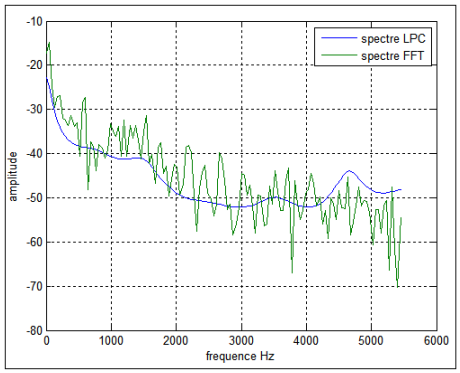


Figure I-7: Le spectre obtenu par la FFT et LPC

I.5.3. Le Spectrogramme

Le spectrogramme est un outil de visualisation utilisant la technique de la transformée de Fourier et donc du calcul de spectres. Il a commencé à être largement utilisé en 1947, à l’apparition du sonagraphe, et est devenu l’outil incontournable des études en phonétique pendant de nombreuses années.

L’apparition de l’informatique puis d’écrans graphiques de bonne qualité a permis d’abandonner tout matériel comme le sonagraphe mais la technique du spectrogramme est encore

aujourd'hui largement utilisée dans de nombreux domaines, du fait de sa simplicité de mise en œuvre et des résultats intéressants qu'elle procure.

On parle de spectrogramme à larges bandes ou à bandes étroites selon la durée de la fenêtre de pondération. Les spectrogrammes à bandes larges sont obtenus avec des fenêtres de pondération de faible durée (typiquement 10 ms); ils mettent en évidence l'enveloppe spectrale du signal, et permettent par conséquent de visualiser l'évolution temporelle des formants. Les périodes voisées y apparaissent sous la forme de bandes verticales plus sombres. Les spectrogrammes à bandes étroites sont moins utilisés. Ils mettent plutôt la structure fine du spectre en évidence : les harmoniques du signal dans les zones voisées y apparaissent sous la forme de bandes horizontales.

Le spectrogramme permet de mettre en évidence les différentes composantes fréquentielles du signal à tout instant.

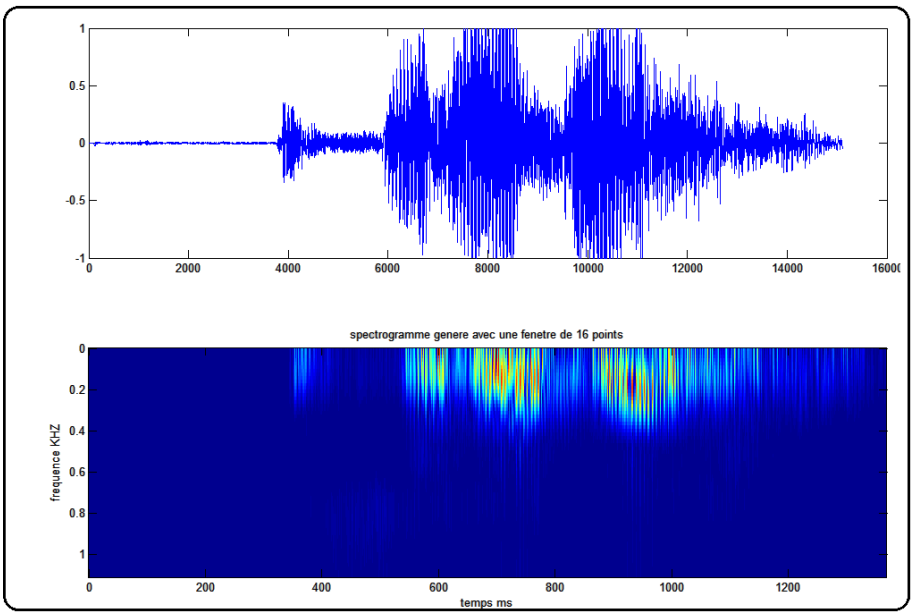


Figure I-8: Spectrogramme et signal temporel de la phrase "السلام عليكم"

I.5.4. Intérêts de la représentation fréquentielle du signal de parole

La représentation fréquentielle de la parole est d'une très grande importance dans le domaine de la communication parlée. Elle a permis l'extraction des paramètres pertinents du signal de parole comme la fréquence fondamentale et les formants. Ces paramètres sont d'une importance capitale dans de nombreux domaines comme :

- Les différentes méthodes d'analyse et synthèse
- La reconnaissance automatique de la parole et du locuteur
- l'identification automatique des langues.

Et bien d'autres domaines.

I.6. Conclusion

Dans ce chapitre on a définie les principaux blocs qui constitue la chaîne de transmission numérique dit sécuriser, et on a aussi vus que la qualité obtenus par cette dernier est bien meilleur que la chaîne de transmission analogique et ce si grâce a un étage qui assure la correction d'erreur non seulement ça mais aussi la possibilité d'implémenté un tel système dans un seul circuit intégré.

Et car ce travail est sur la transmission de la parole donc on a cite a un minimum a savoir sur la parole tel que les aspects physiologique de la phonation, la production de sons de parole, les différentes méthodes de représentation spectrale, et on a termine par cite l'intérêt de la représentation spectrale de signal parole.

Chapitre II : CODAGE SOURCE

I.1. Introduction

L'analyse de la parole est une étape indispensable à toute application de synthèse, de Codage ou de reconnaissance. Elle repose en général sur un modèle. Celui-ci possède un ensemble de paramètres numériques dont les plages de variation définissent l'ensemble des signaux couverts par le modèle.

I.2. Généralité sur les codeurs de parole

Dans les systèmes numérique l'information doit être échantillonne et quantifie, mais les signaux ne sont pas toujours de même forme ni de même fréquence ni de même nature (stationnaire, non stationnaire) d'une part, et aussi la taille ou la longueur de signal a traité après la numérisation peut augmenter le temps de calcule ainsi que les erreurs que vont forcement dégradé la performance la performance de système.

Pour cela en trouve plusieurs normes de codage qui sont mis pour assuré une dynamique et une rapidité de traitement.

Pour notre travail nous intéressant seulement au norme de codage mis pour la parole.

- ***Classification des codeurs et norme de codage***

Les méthodes pour le codage de la parole peuvent être classées en trois classes :

- Les codeurs par forme d'onde.
- Les vocodeurs (technique d'analyse par synthèse).
- Les codeurs hybrides.

Typiquement les codeurs par forme d'onde sont conçus pour être capable de coder n'importe quelle signale et quelque soit sa nature, ils sont utilisée pour des débits élevé et donnent une très bon qualité.

Les vocodeurs utilisent une méthode dit analyse par synthèse ou l'on essaye d'extraire du signal de parole un ensemble de paramètre lie a un modèle simplifie (enveloppé spectrale, pitch, voisement, énergie). Ces codeurs opérant a des très faibles débits mais la qualité de la parole est limite.

Les codeurs hybrides font intervenir les techniques de codage par forme d'onde et les techniques d'analyse par synthèse, et donnent une bonne qualité de la parole à des débits moyens.[6]

Le tableau suivant représente les classes des codeurs ainsi que les techniques utilisent et aussi quelques exemples d'algorithme :

TableauII-1: Les classes des codeurs de la parole

<table>
<tr><th>Classe</th><th colspan="2">Techniques</th><th colspan="2">Exemples d'Algorithmes</th></tr>
<tr><td rowspan="9">Codeurs par forme d'onde
(Représentation directe)</td><td rowspan="7">Domaine temporel</td><td rowspan="2">Quantification scalaire non linéaire</td><td rowspan="2">PCM, 64-56 kbps</td><td>A-Law</td></tr>
<tr><td>μ-Law</td></tr>
<tr><td>Quantification scalaire différentielle</td><td colspan="2">DPCM, 24 kbps</td></tr>
<tr><td></td><td colspan="2">DM,</td></tr>
<tr><td>Quantification scalaire différentielle adaptative</td><td rowspan="2">ADM, 16 kbps</td><td>CFDM (Jayant)</td></tr>
<tr><td>CVSDM (Greekles)</td></tr>
<tr><td rowspan="3">Quantification scalaire Prédictive adaptative</td><td rowspan="3">ADPCM, 16-24-32-40 kbps</td><td>ITU-T ADPCM</td></tr>
<tr><td>IMA ADPCM</td></tr>
<tr><td>Dialogic ADPCM</td></tr>
<tr><td rowspan="2">Domaine fréquentiel</td><td>Division fréquentielle</td><td colspan="2">SBC, 16 à 24 kbps</td></tr>
<tr><td>Codage par Transformée</td><td colspan="2">ATC, 9.6 kbps</td></tr>
<tr><td>Vocodeurs</td><td colspan="2">Analyse par synthèse</td><td colspan="2">LPC, 2,4 à 16 kbps</td></tr>
<tr><td rowspan="3">Codeurs hybrides</td><td colspan="2" rowspan="3">- Quantification vectorielle adaptative
- Analyse par synthèse</td><td colspan="2">CELP, 2-8 kbps</td></tr>
<tr><td colspan="2">LD-CELP, 16 kbps</td></tr>
<tr><td colspan="2">GSM, 13 kbps</td></tr>
</table>

Pour les applications de parole les vocodeurs sont souvent utilise grâce a leur simplicité de mise en œuvre.

I.3. Principe de la prédiction linéaire

Un codeurs LPC (linéaire prédiction coding) propose un modèle simple de la source et du conduit vocal puis il code le modèle le plus proche possible de l'origine, c'est un système d'analyse par synthèse, ou la parole est synthétisé en utilisent les paramètres extraire du modèle les paramètres extraits du modèle de production de la parole. [6]

Le codeur utilise un filtre LPC (filtre de synthèse) dont les coefficients sont modifie périodiquement (10ms) pour modélise le conduit vocal. La première étape qui est l'analyse consiste le voisement ou le nom voisement du signal de la parole a code, si il est voisé la fréquence fondamentale est extraite. [6]

Donc les informations transmises par trame vers le récepteur sont :

- type d'excitation (voise, non voise).
- La période de pitch (p) pour les excitations voise.
- Le gain G (l'énergie de signal).
- Les coefficients a_i de filtre LPC.

Donc en donné la représentation générale d'un codeur/décodeur LPC :

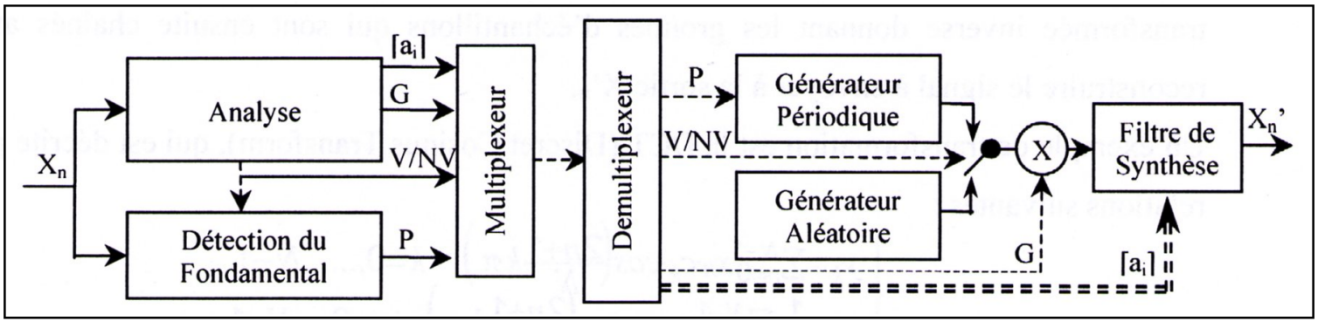


Figure II-1: Représentation général d'analyse/synthèse LPC

I.4. Codage de la parole (analyse)

I.4.1. Etage prétraitement

Avant d'être analysé, le signal analogique de parole doit subir un pré traitement qui se présente suivant trois étapes : échantillonnage, pré accentuation et fenêtrage.

I.4.1.1. Echantillonnage

Le signal vocal issu de la source 'microphone ou appareil d'enregistrement' est un signal analogique. Pour qu'il soit traité par les calculateurs, il est nécessaire de le discrétiser en temps. Donc l'échantillonnage c'est l'opération qui fait correspondre au signal s(t) continue une suite numérique qui constitue le signal discret s[n]. [5]

D'après le théorème de Shannon, la perte d'information entre le signal continu et le signal

Discret correspondant est quasiment nulle si et seulement si la fréquence d'échantillonnage :

$$Fe \geq 2 \cdot F_{\max} \text{.....(1)}$$

Avec F max est la fréquence maximale du spectre du signal

Pour le signal de parole, le choix de la fréquence d'échantillonnage résulte d'un compromis. Son spectre peut s'étendre jusque 12khz ; il faut donc en principe choisir une fréquence Fe égale à 24khz au moins. Cependant le coût d'un traitement numérique, filtrage, transmission, ou simplement enregistrement peut être réduit d'une façon notable si l'on accepte une limitation du spectre par un filtre préalable.

Cependant, Fe peut être choisie suivant la technique utilisée. Fe peut varier de 6khz à 16khz pour les techniques d'analyse, de synthèse ou de reconnaissance de la parole. Par contre pour le signal audio (parole et musique), on exige une bonne représentation du signal jusque 20khz.

L'opération inverse d'échantillonnage est dite interpolation. [5]

I.4.1.2. Préaccentuation

Le spectre du signal en sortie présente une atténuation de –6db/oct due aux influences de la source d'excitation et des lèvres. Pour compenser cette atténuation, on introduit le signal vocal dans un filtre de pré accentuation afin d'égaliser les aigus toujours plus faibles que les graves. Ce filtre est défini par une fonction de transfert dont la transformée en Z est :

$$X(z) = 1 - a \cdot z^{-1} \text{..... ....(2)}$$

Le paramètre d'accentuation 'a' est tel que : 0.90< a<0.98

En pratique en choisie a = 0.95

I.4.1.3. Fenêtrage

La parole est un phénomène non stationnaire, c'est à dire, ses propriétés statistiques changent continuellement dans le temps. Cependant, l'observation du signal de la parole indique qu'il n'évolue pas ou peu sur des durées de quelques millièmes de secondes. On peut donc considérer ce signal comme étant stationnaire durant ce temps qu'on appellera fenêtre (stationnarité locale).

Le fenêtrage consiste à délimiter la durée de ce dernier en le multipliant par une fenêtre allant de 20 à 30ms. En effet, sur cette durée, on estime que le signal n'a pas le temps de varier et conserve au moins une durée de la période du fondamental.

Ces fenêtres doivent être glissantes de manière à conserver les échantillons importants à traiter et se recouvrir afin de diminuer la perte d'informations aux bords de ces dernières. Il existe plusieurs sortes de fenêtres: Hamming, Hanning, rectangulaire et autres.

La fonction générale de la fenêtre, peut s'écrire de la forme suivante :

$$W_H = \begin{cases} \alpha + (1-\alpha).\cos(2\pi n / N) : pour : |n| \le N/2 \\ 0 \cdots ailleur \end{cases} \quad \text{.....(3)}$$

n : le néme échantillon

N : le nombre d'échantillons

Pour :

- α=0.5 on obtient la fenêtre de Hanning.
- α=0.54 on obtient la fenêtre de Hamming.
- α=1 on obtient la fenêtre de rectangulaire.

Le fenêtrage de type Hamming est le plus utilisée en parole, car les lobes latéraux de son spectre sont plus faibles que ceux des autres types de fenêtres et le lobe principal contient le max d'énergie.

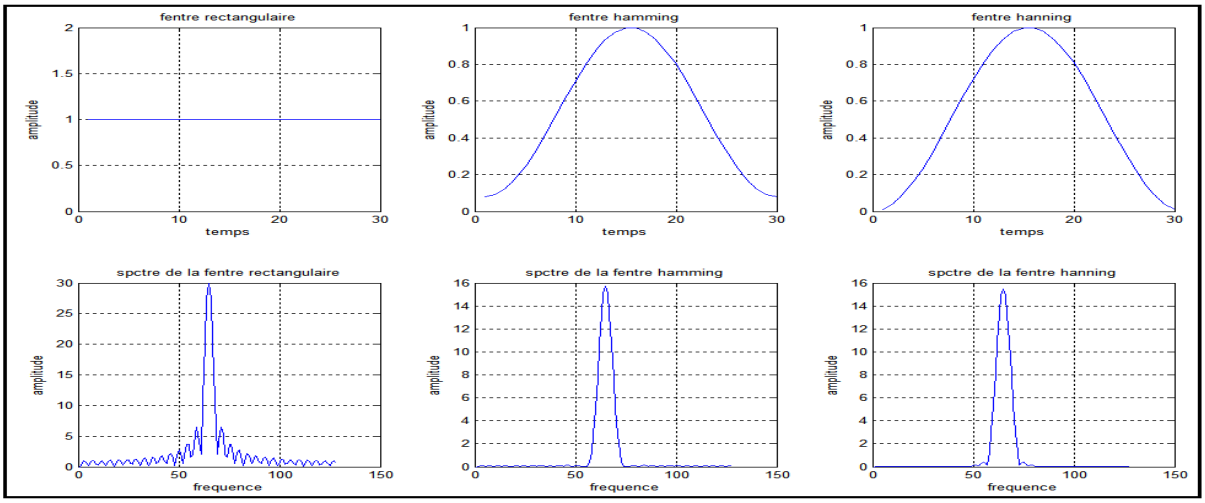


Figure II-2: Fenêtres et leurs spectres

L'inconvénient du fenêtrage est la perte d'information sur les échantillons proches des extrémités de ces fenêtres, il est donc nécessaire d'effectuer une opération de recouvrement. En générale on effectue un recouvrement de moitié.

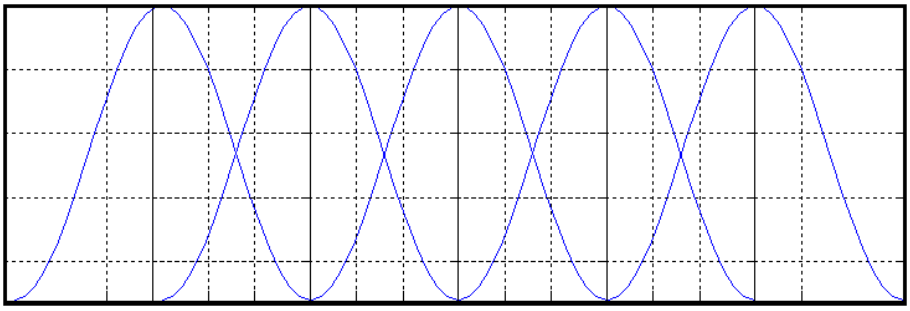


Figure II-3: Recouvrement des fenêtres type Hamming

I.4.2. Calcul des coefficients de prédiction

Le codage LPC consiste a estimer la valeur de d'échantillon a venir sur la base de quelques valeurs mesurées précédemment S[n-k]. [7]

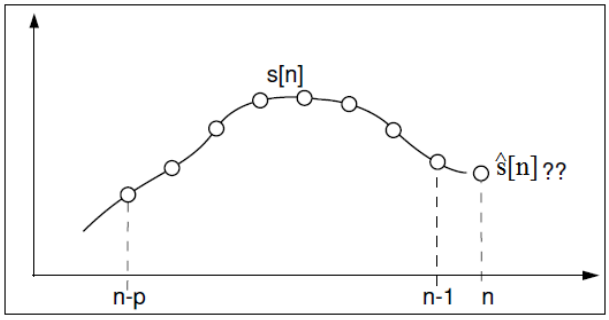


Figure II-4: Les échantillons S[n-p] a S[n-1] sont utilisés pour estimer la valeur a venir

Donc on définie la valeur estimée $S^{\wedge}[n]$ à partir échantillons précédents pondérés par les coefficients de prédiction a_k avec un ordre de prédiction P généralement de 8 a12 :

$$S^{\wedge}[n] = \sum_{K=1}^{P} a_K S(n-K) \quad \text{.....(4)}$$

On définie l'erreur de prédiction [1] a l'aide de signal original et le signal prédit par la relation suivant :

$$e(n) = s(n) - \hat{s}(n) \quad \text{.....(5)}$$

$$e(n) = s(n) - \sum_{K=1}^{P} a_K s(n-k) \quad \text{.....(6)}$$

Pour obtenir les coefficients de prédiction en applique le critère de moindre carré, en minimise l'énergie de l'erreur donnée par relation suivant :

$$J = \sum_{n} (e(n))^2 \quad \text{.....(7)}$$

$$J = \sum_{n} (s(n) - \sum_{K=1}^{P} a_K s(n-K))^2 \quad \text{.....(8)}$$

Ce critère sera minimisé en annulent la dérivé de cette énergie par rapport au coefficient de prédiction :

Pour $1 \le i \le P$

$$\frac{\delta J}{\delta a_K} = \frac{\delta[\sum_n (s(n) - \sum_{K=1}^{P} a_K s(n-K))^2]}{\delta a_K} = 0 \text{.....(9)}$$

Ce qui conduit a l'équation suivant en supposant la stationnarité de signal sur un intervalle de durée de 15 à 25 ms:

Pour $1 \le i \le P$

$$\sum_{n1}^{n2} \sum_{K=1}^{P} a_K s(n-k)s(n-i) = \sum_{n1}^{n2} s(n)s(n-i) \text{.....(10)}$$

Pour résoudre cette équation (système de Yule Walker) il existe plusieurs méthode de résolution tel que :

- La méthode d'autocorrélation.
- La méthode de covariance.

o ***Méthode d'autocorrélation***

Dans cette méthode l'autocorrélation est calculée sur une longueur de signal choisie par une fenêtre de durée N.

La fonction dit autocorrélation a court terme sera définie par :

$$R(i) = \sum_{n=0}^{N-1} s(n)s(n-i) \text{.....(11)}$$

Alors le système d'équation (A) s'écrira alors sous la forme :

$$R(i) = \sum_{K=1}^{P} a_K R(i-K) \text{.....(12)}$$

Ou bien sous la forme matricielle, tel que :

$$\begin{bmatrix} R_0 & R_1 & \cdots & R_p \\ R_1 & R_0 & \cdots & R_{p-1} \\ \vdots & \vdots & \vdots & \vdots \\ R_N & R_{p-1} & \cdots & R_0 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_p \end{bmatrix} = \begin{bmatrix} R_0 \\ R_1 \\ \vdots \\ R_p \end{bmatrix}$$

Donc nous avons une matrice dite de Toeplitz symétrique ou les éléments diagonaux sont identiques, ces propriétés sont exploitées pour développer un algorithme de résolution efficace. [8]

En exploitant les équations (8),(10) et (11) l'erreur de prédiction minimale sera donnée par :

$$E(p) = R(0) + \sum_{K=1}^{P} a_K R(k) \text{.....(13)}$$

I.4.3. Algorithme de résolution

Il s'agit de résoudre le système linéaire donne par la matrice précèdent, en inversant la matrice d'ordre P de ce système. La méthode algébrique classique exige pour cela un nombre d'opération (multiplication, addition) de l'ordre p^3 .l'algorithme de Levinson profite de la structure particulière Tœplitz symétrique de la matrice d'autocréation pour résoudre le système matriciel avec un nombre d'opérations seulement p^2 . [8]

- ***Algorithme de Levinson***

Pour fixe les idées on va résoudre un système de Yule Walker d'ordre 2 suivant :

$$\begin{bmatrix} R(0) & R(1) & R(2) \\ R(1) & R(0) & R(1) \\ R(2) & R(1) & R(0) \end{bmatrix} \cdot \begin{bmatrix} 1 \\ a_2(1) \\ a_2(2) \end{bmatrix} = \begin{bmatrix} E(2) \\ 0 \\ 0 \end{bmatrix}$$

Soit un signal x(n) que l'on cherche a modélise. Sa fonction de d'auto corrélation est donne par :

R(m) avec m=0.1...P

Ordre 0 :

$$[R(0)] \cdot [a_0(0)] = [E(0)]$$

$$a_0(0) = 1$$

$$E(0) = R(0)$$

Ordre 1 :

On ajoute une (ligne, colonne) au système (0)

$$\begin{bmatrix} R(0) & R(1) \\ R(1) & R(0) \end{bmatrix} \cdot \begin{bmatrix} 1 \\ a_1(1) \end{bmatrix} = \begin{bmatrix} E(1) \\ 0 \end{bmatrix}$$

On dévlope les équations :

$$\begin{cases} R(1) + R(0) \cdot a_1(1) = 0 \\ R(0) + R(1) \cdot a(1) = E(1) \end{cases}$$

Ce qui donne :

$$\begin{cases} a_1(1) = -R(1)/R(0) \\ E(1) = R(0) \cdot (1 - (R(1)/R(0))^2) \end{cases}$$

$$\begin{bmatrix} R(0) & R(1) \\ R(1) & R(0) \end{bmatrix} \cdot \left(\begin{bmatrix} 1 \\ 0 \end{bmatrix} + K_1 \begin{bmatrix} 0 \\ 1 \end{bmatrix} \right) = \begin{bmatrix} E(0) \\ 0 \end{bmatrix} + K_1 \begin{bmatrix} 0 \\ E(0) \end{bmatrix} = \begin{bmatrix}$$

Donc la valeur de K(1) est :

Ordre 2 :

$$\begin{bmatrix} R(0) & R(1) & R(2) \\ R(1) & R(0) & R(1) \\ R(2) & R(1) & R(0) \end{bmatrix} \cdot \begin{bmatrix} 1 \\ a_2(1) \\ a_2(2) \end{bmatrix} = \begin{bmatrix} E(2) \\ 0 \\ 0 \end{bmatrix}$$

On ajoute une (ligne, colonne) au système d'ordre (1) :

$$\begin{bmatrix} R(0) & R(1) & R(2) \\ R(1) & R(0) & R(1) \\ R(2) & R(1) & R(0) \end{bmatrix} \cdot \begin{bmatrix} 1 \\ a_1(1) \\ 0 \end{bmatrix} = \begin{bmatrix} E(1) \\ 0 \\ Y(2) \end{bmatrix}$$

Donc : $Y(2) = R(2) + R(1) \cdot a_1(1)$ Avec :

$$\begin{bmatrix} 1 \\ a_2(1) \\ a_2(2) \end{bmatrix} = \begin{bmatrix} 1 \\ a_1(1) \\ 0 \end{bmatrix} + K(2) \begin{bmatrix} 0 \\ a_1(1) \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} E(2) \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} E(1) \\ 0 \\ Y(1) \end{bmatrix} + K(2) \begin{bmatrix} Y(1) \\ 0 \\ E(1) \end{bmatrix}$$

A partir de la on peut calculer :

$Y(1)+K(1)\cdot E(0)=0 \Rightarrow K(1)=-\frac{Y(1)}{E(0)}=-\frac{R(1)}{E(0}$	$\begin{cases} Y(2)+K(2)\cdot E(1)=0 \Rightarrow K(2)=-{Y(2)}/{E(1)} \\ a_2(1)=a_1(1)+K(2)\cdot a_1(1) \\ a_2(2)=K(2) \\ E(2)=E(1)+K(2)\cdot Y(2)=E(1)\cdot(1-K(2)^2) \end{cases}$

La méthode de Levinson-Durbin permet de résoudre le système par une récursion sur l'ordre de prédiction P. L'algorithme fait apparaître trois ensembles de paramètre intéressants :

- les coefficients de prédiction (ai).
- l'énergie de prédiction (Ei).
- les coefficients de prédiction (Ki).

1. initialisation L'algorithme de Levinson avec :

$a_0(0)=1$ et $E(0)=R(0)$

2. le premier coefficient de reflexion K(1) est donne par :

$$K(1)=a_1(1)=-{R(1)}/{R(0)}$$

3. les autres coefficients de reflexion K(m)…….K(P) proviennent de :

$$Y(m)=R(m)+a_{m-1}(1)\cdot R(m-1)+.....+a_{m-1}(m-1)\cdot R(1)$$

$$K(m)=-{Y(m)}/{E(m-1)} \quad \text{avec} \quad m=2,.....,P$$

4. les coefficients de prediction $a_m(i)$:

$$a_m(0)=1$$
$$a_m(i)=a_{m-1}(i)+K(m)\cdot a_{m-1}(m-i)$$
$$a_m(m)=K(m)$$

5.l'energie residuelle de prédiction E(m) :

$$E(m)=E(m-1)(1-K^2(m)) \quad \text{avec} \quad m=1,.....,P$$

6. retour au 2 jusqu'à que m=P

I.4.4. Recherche de la fréquence fondamentale et le type d'excitation

La période du fondamental est un paramètre très important pour la synthèse de la parole, l'oreille est en effet très sensible à ses variations, lesquelles constituent la prosodie. L'estimation du pitch des bien sur liée a la localisation des tranches voisées. Cette estimation est très délicate d'où le grand nombre de méthode proposés pour la détection de la fréquence fondamentale parmi lesquelles on peut cite [4] :

- Méthode d'estimation du pitch par l'autocorrélation (algorithme de SONDHI, DUBNOWSKI).

- Méthode d'estimation du pitch dans le domaine temporel (technique de traitement parallèle).
- Méthode d'estimation du pitch par analyse LPC (l'algorithme SIFT).

o ***Algorithme de Sift***

En 1972 Markel a proposé une méthode que consiste a estimer F0 en se basant sur le filtre inverse dont la fonction de transfert A (z) est donne par :

$$A(z) = 1 - \sum_{k=1}^{P} a_k z^{-k} = {}^{1}\!/_{H(z)} \text{.....(14)}$$

Elle est basée sur l'examen de la fonction d'autocorrélation du résidu LPC e(n), ce qui fait la particularité de cette méthode, c'est on effectue l'analyse sur la source directement, évitant ainsi toute interaction source-conduit vocal. [4]

Les étapes de cette méthode sont :

a) Filtrage de signal

On a vue précédemment que la fréquence fondamental compris entre (80-600Hz) en respectant la gamme de variation de pitch pour les voix masculines, féminine et enfantines, donc il est très intéressent d'utilise un filtre passe bas dont la fréquence de coupure Fc=600Hz pour limite le spectre de signal pour réduire le temps de calcul. [4]

b) Recherche de signal d'excitation e(n)

Nous avons vus que le résidu e(n) de la prédiction linéaire peut être considéré comme le

Signal d'excitation servant a crées le signal s(n) en passant a travers le filtre récursif [4] :

$$H(z) = \frac{S(z)}{E(z)} = \frac{1}{A(z)}$$

Puisque, dans notre cas, le signal s(n) est connu, on peut par filtrage inverse obtenir le résidu e(n) :

$$E(z) = S(z) \cdot A(z)$$

Ce qui revient a convoluer les coefficients $a_i \equiv a(n)$ avec le signal s (n) :

$$e(n) = a(n) \otimes s(n)$$

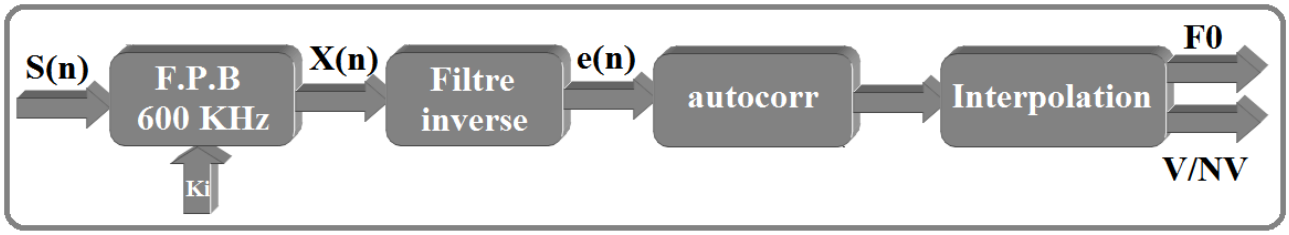
c) calcule d'autocorrélation de e(n)

Comme le signal s(n) est passablement bruité, la recherche de la période est grandement facilitée si on l'effectue le calcule d'autocorrélation sur la e(n). Le résultat de cette dernier est un vecteur de longueur 2N-1 avec un maximum en son milieu, si le signal est périodique d'autres pics distants de la valeur du pitch seront présents. Pour trouve ce dernier il suffit donc de mesure cette distance. [4]

d) interpolation (critère de décision)

L'auto corrélation e(n) peut faire apparaître plusieurs pics, un à l'origine, et un autre pic si le signal est voisé. L'amplitude minimum pour considérer le deuxième pic comme valide est 40% de l'amplitude à L'origine. La fréquence ainsi obtenue est la fréquence d'excitation. En l'absence d'un deuxième pic, le signal est considéré non voisé.[4]

La figure suivant donne la représentation générale de structure de l'algorithme SIFT



FigureII-5: Schéma synoptique de l'algorithme SIFT.

I.4.5. Calcule de gain

L'énergie totale contenue dans la séquence du signal de synthèse doit être égale à l'énergie totale de la séquence correspondante du signal analyse, soit pour un signal voise ou non voise l'énergie est calculer de la même façon par la relation suivant :

$$G^2 = E_p = R_0 - \sum_{i=1}^{p} a_i R_i$$

Par conséquent le gain représente la racine carrée de l'erreur quadratique totale minimale.

Donc on peut représente le codage LPC par le schéma synoptique suivants :

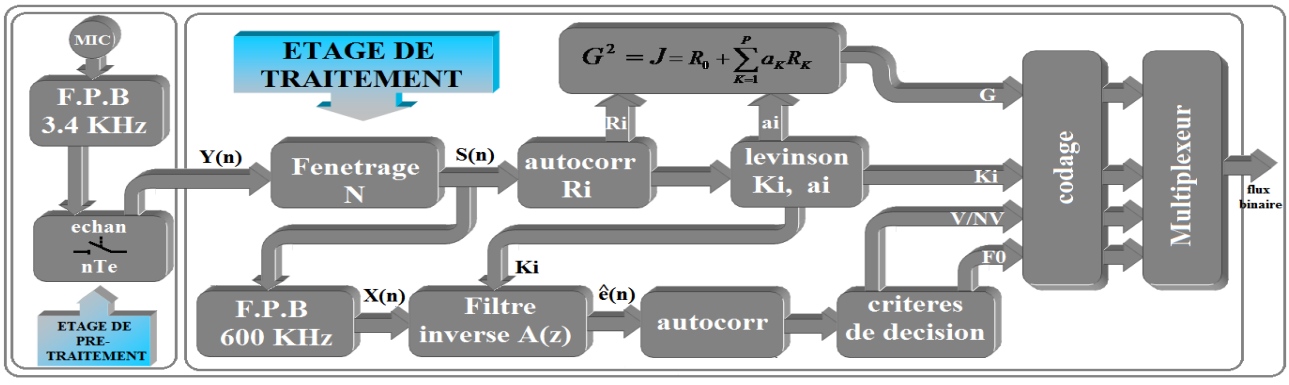


Figure II-6: Schéma synoptique de codeur LPC.

I.5. Le décodage (synthèse LPC)

Fant [9] a proposé en 1960 un modèle de production qui spécifie qu'un signal voisé peut être modélisé par le passage d'un train d'impulsions u(n) à travers un filtre numérique récursif de type tous pôles. On montre que cette modélisation reste valable dans le cas des sons non voisés, à condition que u(n) soit cette fois un bruit blanc. Le modèle final (sauf l'étage décodage et de multiplexage qui sont utilise dans le cas d'une transmission) est illustré à la Fig. (2.7). Il est souvent

appelé modèle auto régressif (AR), parce qu'il correspond dans le domaine temporel à une régression linéaire de la forme :

$$X(n) = G \cdot u(n) + \sum_{i=1}^{P} - a_i X(n-1)$$

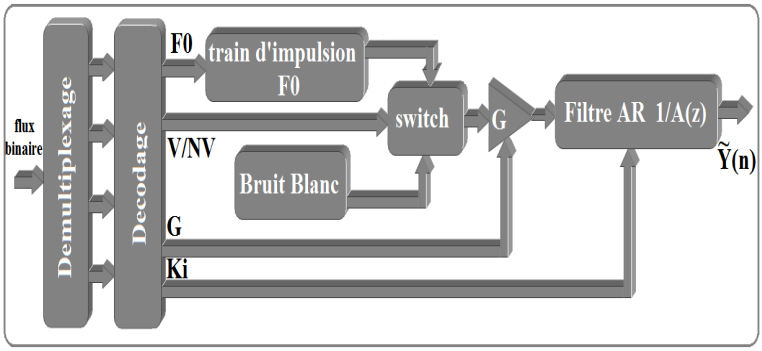


Figure II-7: Schéma synoptique de décodeur LPC ainsi que le modèle de synthétiseur.

Donc la synthèse consiste à reconstituer le signal d'origine à partir des paramètres réduits issus de l'analyse.

Après modélisation de la fonction de transfert, dont les paramètres sont estimés de telle façon que l'erreur entre le signal et la sortie du modèle soit minimale, le synthétiseur sera constitué essentiellement d'un filtre numérique récursif se basant sur les paramètres fournis par l'analyse.

Un commutateur faire sélectionne :

Pour la synthèse d'un son voisé de fréquence fondamentale F0, l'excitation sera un train d'impulsions d'amplitudes unité espacées de l/F0.

Pour la synthèse d'un son non voisé, l'excitation sera un bruit blanc de moyenne nulle et de variance unité.

Bien sur ce signal d'excitation doit être multiplie par un gain proportionnel a la valeur efficace du signal s(n).

Les relations d'équivalences entre le modèle physique et le modèle mathématique peuvent être données comme suit : modélisation

Tableau II-2 : Equivalences modèle physique/modèle mathématique.

modèle physique	modèle mathématique
Conduit vocal	1 A(z), le filtre LPC
Le flux d'air	Le signal d'excitation u(n)
Vibration des cordes vocales	Voisé
Période de vibration des cordes vocales	0 T = 1 F , période du pitch
Fricatives et plosives	non voisé/voisé
Volume d'air	G , le gain

I.5.1. Les paramètres qui control le synthétiseur

La qualité de la synthèse dépend de plusieurs facteurs qui influencent sur les échantillons synthétisés, ces paramètres sont classés en deux catégories [4]:

Paramètres fixes pour tout le signal

- Fréquence d'échantillonnage
- La largeur de la trame
- Le facteur de préaccentuation

Paramètres variables à chaque trame

- Les coefficients de prédiction et les coefficients de réflexion
- La décision de voisement V
- La fréquence fondamentale F0
- Le gain

I.5.2. Stabilité de filtre

Elle est entièrement déterminée par le module des pôles de la fonction de transfert, si ceux ci sont à l'intérieur du cercle unité la stabilité est obtenue. La condition nécessaire et suffisante pour garantir la stabilité [9] est:

$$|K_i| < 1$$

K(i) : coefficients de réflexion

En pratique des instabilités peuvent apparaître, elles sont dues essentiellement à l'accumulation des imprécisions sur le calcul, par exemple des erreurs dues aux arrondis au cours des calculs, aux troncatures etc. [10]

I.5.3. Les structures des filtres

La structure du synthétiseur, comme a été déjà dit auparavant, dépend des paramètres transmis, nous proposons dans le paragraphe suivant de comparer entre deux cas de transmission possibles utilisant les coefficients ai ou ki. [10]

I.5.3.1. Structure transverse

Cette structure, appelée aussi directe, utilise les coefficients ai l'intérêt majeur de cette structure est qu'elle permet d'obtenir un nombre d'opérations très réduit P multiplications et P additions, mais présente plusieurs inconvénients du fait que les coefficients ai de prédiction ne sont pas tous bornés elle présente un grand risque d'instabilité du filtre, il en résulte un signal parole qui oscille fortement. Pour y remédier, on fait appel à une autre structure qui utilise les coefficients parcours qui eux sont borné et assurent la stabilité du filtre. Cette structure est appelée structure en treillis. [10]

I.5.3.2. Structure en treillis

Selon cette structure e filtre inverse engendre simultanément les erreurs progressives et rétrogrades.

Si Em (Z) et E'm(Z) sont les transformées des erreurs forwards et backwards la récurrence nous permettent d'écrire

Cette double récurrence définit la structure en treillis illustrée à la figure (II-8). [4]

On passe du filtre inverse en treillis à celle du filtre de synthèse

- En inversant le sens de propagation du signal dans la ligne supérieure
- En permutant entrée et sortie
- En changeant le signe du multiplieur de la branche ascendante dans chaque cellule.

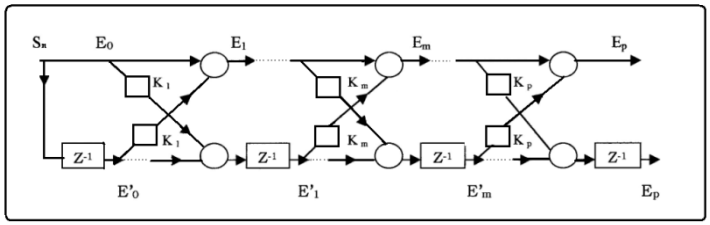


Figure II-8: Représentation d'un filtre en treillis

I.5.4. Désaccentuation

Le signal vocal est souvent soumis à une préaccentuation afin d'assurer un meilleur conditionnement numérique lors de l'analyse. Pour retrouver le signal original, il faut donc faire suivre le filtre de synthèse par un filtre de désaccentuation.

Le principe de la désaccentuation est le même que celui de la préaccentuation, elle consiste en un passage dans un filtre de transmitance :

$$1 \big/ 1 - \eta Z^{-1}$$

I.6. La quantification

La représentation numérique d'un signal implique la quantification de chaque échantillon selon un nombre fini de valeur discrète, l'objectif technique visé est la transmission dans le premier cas chaque paramètre LPC est quantifié, codé puis transmis à la réception, il est décodé puis converti en amplitude continue, après interpolation on souhaite retrouver l'image la plus fidèle possible du signal original.

Dans le seconde cas, la loi de quantification est imposée par le système de traitement ; il peut s'agir d'une représentation en virgule fixe ou en virgule flottante, une contrainte importante pour un système de traitement numérique consiste a commettre des erreurs de calcul qui soient négligeables vis-à-vis de l'incertitude sur le signal lui-même, cet objectif doit être atteint malgré le caractère non stationnaire du signal vocal. [5]

Un traitement en virgule flottante autorise une dynamique beaucoup plus importante pour le signal, mais au prix d'une plus grande complexité des opérateurs arithmétiques. Un traitement en virgule fixe est beaucoup plus simple mais il exige un conditionnement adéquat des algorithmes de traitement. [5]

I.6.1. Quantification uniforme

La figure représente une loi de quantification uniforme de pas δ la valeur quantifié y(i) est choisie au milieu de l'intervalle.

$$y(i) = (1/2)[x(i-1) + x(i)]$$

L'erreur de quantification est aussi représenté, elle est comprise entre $-\delta/2$ et $+\delta/2$ pour $|x| \le x_s = L\delta/2$

$$-x_s \le x \le x_s \rightarrow |e| \le \delta/2$$

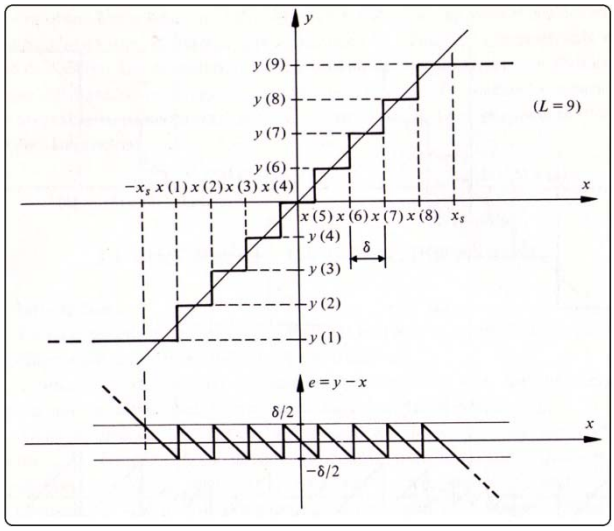


Figure II-9: Quantification uniforme (L=9).

On parle dans ce cas d'erreur (ou de bruit) de granulation. Lorsque $|x| > x_s$ il y a dépassement, on parle alors d'erreur (ou de bruit) de saturation ou de dépassement.

Une loi uniforme et symétrique est donc caractérisée par :

Les niveaux de saturation $\pm x_s$.

Le nombre de niveaux quantifiés L, on choisit normalement $L(ouL+1) = 2^b$

Le pas de quantification vaut :

$$\delta = 2x_s/L$$

Le facteur de charge Γ de quantificateur est défini par le rapport :

$\Gamma = x_s/\sigma_x$ Avec σ_x est l'écart type du signal

I.6.2. Rapport signal a bruit pour une quantification uniforme

Le rapport signal a bruit, noté RSB est défini par le logarithme du rapport de la variance du signal a celle du bruit :

$$RSB = 10 \cdot \log(\sigma^2{}_x / \sigma_e^2)[dB]$$

En l'absence de dépassement σ_e^2 est donnée par :

$$\sigma_e^2 = \delta^2 / 12$$

Par conséquent le RSB vaut donc

$$RSB = 4.77 + 6.025 \cdot b - 20 \cdot \log(\Gamma)$$

En cas de dépassement, le RSB est dégrade :

$$RSB = 10 \cdot 10 \log(\frac{\sigma_x^2}{\sigma_e^2 + \sigma_D^2})$$

Cette dégradation du RSB dépend essentiellement de la loi de répartition du signal x, elle de manifeste dans le cas d'un signal aléatoire uniforme. [5]

I.7. Multiplexeur démultiplexeur

Un multiplexeur est un système combinatoire qui met sur sa sortie unique la valeur d'une de ses 2^n entrées de données, le numéro de l'entrée sélectionnée étant fourni sur les n entrées de commande, ce qui a permet de réduire le nombre des canaux a utilise pour la transmission d'information.

Le démultiplexeur est le circuit qui permet de récupérée les informations envoyées par le multiplexeur sur un seul canal a l'aide a des entres d'adressage qui vont choisir sur quel sortie on doit avoir ces mots logique. [11]

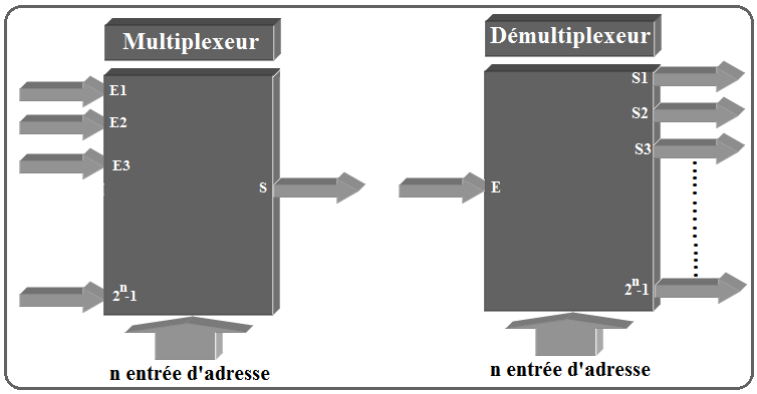


Figure II-10: Schéma synoptique du multiplexeur et le démultiplexeur

I.8. Conclusion

Pour réduire encore le débit binaire et atteindre des valeurs situées entre 2000 et 16000 b/s, il faut exploiter une connaissance a priori du mécanisme qui a produit le signal vocal. Certaines méthodes d'analyse permettent d'estimer les paramètres d'un modèle de production ; il s'agit d'une part, de la transmitance du conduit vocal et d'autre part, de son mode d'excitation.

Cependant, le signal vocal ne peut être considère comme quasi stationnaire que sur des intervalles de temps de l'ordre de 20 ms ; les paramètres du modèle doivent donc être rafraîchis toutes les 20ms environ, ce qui implique des possibilités adéquates de calcul en temps réel .On verra que la transmission de ces paramètres exige un débit de l'ordre de 4500b/s .On parle dans ce cas de codage source

Un ensemble de paramètres ainsi transmis caractérise un spectre court terme c'est-à-dire valable pendant un temps limité. L'oreille est sensible à spectre court terme ; on peut alors pousser l'idée à son terme si l'on remarque qu'elle ne peut distinguer qu'on nombre fini de spectres distincts.

On peut associer à chacun de ces spectres un mot de B bits, qui sera transmis, l'expression montre que l'on peut ainsi ramener le débit à une valeur inférieure à 1000b/s

Le signal synthétique qui résulte d'un codage paramétrique du signal original ne pressente évidemment pas une qualité suffisante pour la téléphonie commerciale ; en particulier, il devient très malaisé de reconnaître le locuteur. Ces techniques de codage concernent donc des applications très particulières (application militaires)

CHAPITRE III : CRYPTAGE AES

I.1. Introduction

L'homme a toujours le besoin de dissimuler des informations confidentielles de leurs communication .ce procède s'appelle la cryptographie, il existe depuis très long temps le mots de cryptographie est un terme générique désignant l'ensemble des technologie permettant de chiffrer des messages, c'est-à-dire permettant de le rendre inintelligibles sans une action a fin qu'aucune personne autre que le destinataire ne puisse les lire.

Les messages cryptés sont la plupart du temps échangés entre un émetteur et un récepteur (Pas forcément humains d'ailleurs) communiquant à travers un certain canal de communication courrier, réseau informatique …etc.).

I.2. Historique

Depuis le cassement de cryptage DES le NIST à lancé en 1997 un appel d'offres pour remplaces le DES et élire un algorithme de cryptage à clé secrète capable de protéger la confidentialité des informations pour le 21emme siècle.

En 1998, le NIST avait reçu quinze propositions parmi ces propositions RIJNDAEL Celles-ci proviennent de la cryptographique mondiale et ont été présentées au cours de la conférence First AES[1] candidate conférence (AES 1).

Au cours de cette conférence, le NIST a demandé des avis publics sur les différents candidats. Les résultats des analyses de ces différents chiffres ont été présentés au cours de la conférence second AES Candidate Conférence (AES 2) qui s'est tenue en 1999. A l'issue de cette seconde conférence et en utilisant les analyses et les commentaires sur les différents candidats, cinq chiffres finalistes ont êtes retenus : MARS, RC6, Rijndael, Serpent, Towfish.

L'algorithme de RIJINDAEL, conçu par deux chercheurs belges Vincent Rijmen et Joan Daemen, fut nommé nouveau standard le 02 octobre 2000.Ce dernier adopte une structure très différente de celle du DES. [1]

En effet, toutes ses transformations sont effectuées au niveau octet et non pas au niveau bit. De plus, il réalise toutes les opérations arithmétiques dans le champ de Galois GF (2^8).

Dans notre cadre PFE on intéresse par ce cryptage AES de telle manière en traite les signaux avant et âpres le cryptage. Pour cela on commencera par introduire quelques concepts mathématiques puis en entame le plan général de l'algorithme leur simulation et le résultat d'implémentation de cryptage.

I.3. Rappels mathématiques

Toutes les opérations de multiplication et addition dans l'algorithme de RIJINDAEL sont réalises dans le champ de GALOIS [12] GF (2^8).On désigne par GF (2^8), l'ensemble des polynômes a coefficient binaires inférieur a 8 noté (2^8).

I.3.1. L'addition

Pour la représentation polynomiale, la somme de deux éléments est polynôme dont les coefficient sont donnés par la somme terme a terme modulo 2 de ce dernier. [12]

Exemple : soit B1 et B2 deux élément de GF (2^8).

B1 = (57)10= $x^5 + x^4 + x^3 + 1$

B2 = (83)10= $x^6 + x^4 + x + 1$

Les résultats de la somme de B1 et B2 dans GF (2^8) est donne par :

B = B1+ B2= $(x^6 + x^4 + x + 1) + (x^5 + x^4 + x^3 + 1)$

= $x^6 + x^5 + x^3 + x + 1$= (106)10

Avec la notation binaire, ceci automatiquement correspond a un simple XOR entre les 2 polynômes

B = B1+ B2= $(111001)_2 + (1010011)_2 = (1101010)_2$

L'addition ainsi définie remplit toutes les conditions pour donner une structure algébrique

- c'est une opération interne.
- Associativité et commutativité
- Existence de l'élément neutre (0)10= (00000000)2 = (00) hex
- Existence d'un inverse pour chaque élément qui est l'élément lui-même

I.3.2. La multiplication

Avec la notion polynomiale, la multiplication dans le champ de GALOIS correspond a une multiplication polynomiale modulo un polynôme irréductible de degré 8.pour RIJINDAEL, ce polynôme, noté m (x), est donné par [12] :

m (x)=(283)dec= $x^8 + x^4 + x^3 + x + 1$=(11B)hex

Prenons comme exemple le produit B= B1*B2

B1 (57)hex = $x^6 + x^4 + x^2 + x + 1$

B2 (83)hex = $x^7 + x + 1$

B (C1)hex = $x^7 + x^6 + 1$

On aura

$(x^6 + x^4 + x^2 + x + 1).(x^7 + x^6 + 1) =$

$x^{13} + x^{11} + x^9 + x^8 + x^7 + x^7 + x^5 + x^3 + x^2 + x + x^6 + x^4 + x^2 + x + 1$

B= $(x^{13}+x^{11}+x^9+x^8+x^6+x^5+x^4+x^3+1)$ mod $(x^8+x^4+x^3+x+1)= x^7+x^6+1$

Le résultat de multiplication est un polynôme à coefficients binaires de degré inférieur à 8. Cette opération est associative, commutative, distributive par rapport à l'addition et possède un élément neutre (01) hex .De plus, pour chaque élément B de GF (2^8),il existe un inverse multiplicatif B^{-1} vérifiant B$\times B^{-1}$ = (01)hex.

I.3.3. La multiplication par « x »

Un cas particulier de la multiplication est celui de multiplier un polynôme B1 par le polynôme de degré « 1» : B2=x. ce cas est très intéressant dans la mesure ou il peut être généralise pour la multiplication de n'importe quels deux polynômes. [12]

Pour illustrer ceci, soit B1=$b_7x^7+b_6x^6+b_5x^5+b_4x^4+b_3x^3+b_2x^2+b_1x+b_0$

B' =B1.x=$b_7x^8+b_6x^7+b_5x^6+b_4x^5+b_3x^4+b_2x^3+b_1x^2+b_0x$

Le résultat de la multiplication de B par « x » est obtenu par la réduction de B modulo m(x).

Si b7=0, le résultat de l'opération est B=B'.et si b7=1, le résultat est obtenu en réalisant un XOR entre B' et (11B)hex.

A noter que la multiplication par des puissances supérieures de « x » n'est autre que des multiplications successives par « x » et l'addition des résultats intermédiaires si nécessaire

Pour illustrer ceci, on prend l'exemple suivant :

B1=(57)hex=(1010111)2

B2=(13)hex=(10011)2

B=B1.B2=(FE)hex

(57)hex (02) hex =(AE) hex

(57) hex (04) hex = (AE) hex (02) hex =(47)

(57) hex (08) hex = (AE) hex (02) hex = (47) hex (02) hex = (8E) hex

(57) hex (10) hex = (AE) hex (08) hex = (47) hex (04) hex =(8E) hex (02) hex = (07) hex

Donc

B =(57) hex (13) hex = (57) hex [(01) hex +(02) hex +(10) hex]

= (57) hex + (AE) hex + (07) hex =(FE) hex

I.3.4. polynômes à coefficient dans GF (2^8)

Un mot de 32 bits (4 octets) peut être représenté par un polynôme de degré inférieur à 4 et à cœfficients dans GF (2^8). (Polynômes à coefficient binaire de degré inférieur à 8)

Chacun de ses coefficients représente un octet. [12]

Par exemple, soit :

B $= (5B99B3E7)_{hex} = (10110111001100110110001111100111)_2$

$= (5B)_{hex}.x^3 + (99)_{hex}.x^2 + (B3)_{hex}\,x + (E7)_{hex}$

L'addition de deux mots est alors égale à l'addition des polynômes les représentants soit un XOR entre les coefficients de même degré.

Si on considère B_1, B_2 deux mots de 32 bits avec cette représentation :

$B_1 = a_3.x^3 + a_2.x^2 + a_1.x + a_0$

$B_2 = b_3.x^3 + b_2.x^2 + b_1.x + b_0$

On aura : $B_1 + B_2 = (a_3 + b_3).x^3 + (a_2 + b_2).x^2 + (a_1 + b_1).x + (a_0 + b_0)$

La multiplication de deux mots donne un polynôme de degré inférieur à 7 calculé par la définition précédente et la définition du produit de deux polynômes en générale, ainsi :

$$B = B_1 \times B_2 = c_6.x^6 + c_5.x^5 + c_4.x^4 + c_3.x^3 + c_2.x^2 + c_1 x + c_0$$

$$\begin{aligned}
c_0 &= (a_0 \times b_0)\\
c_1 &= (a_1 \times b_0) + (a_0 \times b_1)\\
c_2 &= (a_2 \times b_0) + (a_1 \times b_1) + (a_0 \times b_2)\\
c_3 &= (a_3 \times b_0) + (a_2 \times b_1) + (a_1 \times b_2) + (a_0 \times b_3)\\
c_4 &= (a_3 \times b_1) + (a_2 \times b_2) + (a_1 \times b_3)\\
c_5 &= (a_3 \times b_2) + (a_2 \times b_3)\\
c_6 &= (a_3 \times b_3)
\end{aligned}$$

A fin de rester dans le domaine des mots de 32 bits, on définit un polynôme B'par

B'= B modulo $(x^4 + 1) = d_3.x^3 + d_2.x^2 + d_1.x + d_0$

Les nouveaux cœfficients donnant B'seront donnés par :

$$\begin{aligned}
d_0 &= (a_0 \times b_0) + (a_3 \times b_1) + (a_2 \times b_2) + (a_1 \times b_3)\\
d_1 &= (a_1 \times b_0) + (a_0 \times b_1) + (a_3 \times b_2) + (a_2 \times b_3)\\
d_2 &= (a_2 \times b_0) + (a_1 \times b_1) + (a_0 \times b_2) + (a_3 \times b_3)\\
d_3 &= (a_3 \times b_0) + (a_2 \times b_1) + (a_1 \times b_2) + (a_0 \times b_3)
\end{aligned}$$

Sous forme matricielle, on écrira :

$$\begin{pmatrix} d_0 \\ d_1 \\ d_2 \\ d_3 \end{pmatrix} = \begin{pmatrix} a_0 & a_3 & a_2 & a_1 \\ a_1 & a_0 & a_3 & a_2 \\ a_2 & a_1 & a_0 & a_3 \\ a_3 & a_2 & a_1 & a_0 \end{pmatrix} \times \begin{pmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{pmatrix}$$

Comme le polynôme (x^4+1) n'est pas irréductible, la multiplication ainsi définie ne possède pas d'élément inverse pour chaque polynome.Pour cette raison, l'AES utilise un polynôme multiplicatif inversible particulier A (x) donnée par :

$$A(x) = (03)_{hex}.x^3 + (01)_{hex}.x^2 + (02)_{hex}.x + (02)_{hex}$$

L'inverse :

$$A^{-1}(x) = (0B)_{hex}.x^3 + (0D)_{hex}.x^2 + (09)_{hex}.x + (0E)_{hex}$$

$$A(x) \cdot A^{-1}(x) = (01)\text{hex}$$

I.4. Déroulement d'algorithme de cryptage :

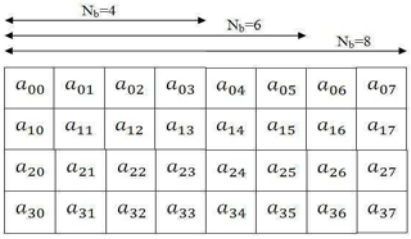
L'AES [13] est un algorithme de chiffrement par bloc symétrique et itératif (à plusieurs rondes) avec une taille de bloc d'entrée et une taille de clef variables. En effet, l'algorithme supporte des tailles de bloc et des longueurs de clefs de 128 bits, 192 bits, 256 bits indépendamment.

Matrice d'état :

Le bloc de données ou bien l'information binaire a crypté est appelé aussi matrice d'état contient 4 lignes et Nb colonnes dont les élément sont des octets .donc Nb prend les valeurs 4,6,8 pour les tailles de bloc 128,192,256. [14]

NB: pour simplifier les chose on convertis ces octets en Hexa

Tableau III-1 : Matrice d'états et le Nb qui prend chaque fois



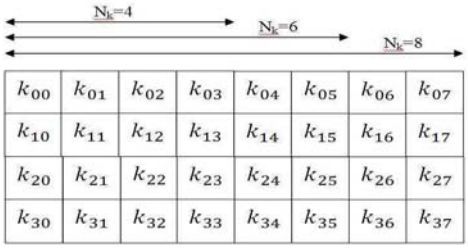
Par exemple : dans notre cas en utilise l'AES 128 donc matrice d'état devient :

19	a0	9a	e9
3d	f4	c6	f8
e3	e2	8d	48
be	2b	2a	08

La clé initiale :

La clef est aussi une matrice de 4 lignes et de Nk colonnes qui prend des valeurs 4, 6, 8 pour des longueurs 128bits, 192bits, 256bits. [14]

Tableau III-2 : la clé initial et le Nk qui peut prend dans chaque cas



k_{00}	k_{01}	k_{02}	k_{03}	k_{04}	k_{05}	k_{06}	k_{07}
k_{10}	k_{11}	k_{12}	k_{13}	k_{14}	k_{15}	k_{16}	k_{17}
k_{20}	k_{21}	k_{22}	k_{23}	k_{24}	k_{25}	k_{26}	k_{27}
k_{30}	k_{31}	k_{32}	k_{33}	k_{34}	k_{35}	k_{36}	k_{37}

Par exemple : dans notre cas en utilise l'AES 128 le clef initial sera :

2b	28	ab	09
7e	ae	f7	cf
15	d2	15	4f
16	a6	88	3c

Le nombre de rounds est très important pour le quel on peut réaliser un chiffrement, ce nombre noté Nr est en fonction de Nb et Nk comme le tableau indique

Tableau III-3 : Nombre de rounds utilise dans l'algorithme

N_r	N_b=4	N_b=6	N_b=8
N_k=4	10	12	14
N_k=6	12	12	14
N_k=8	14	14	14

I.4.1. Le cryptage

Le cryptage de l'algorithme de RIJINDAEL consiste à appliquer 4 transformations sur les octets de la matrice d'état en itérant Nr-1.[14]

- Sub byte
- Shift Rows
- Mix columns
- Add Round Key

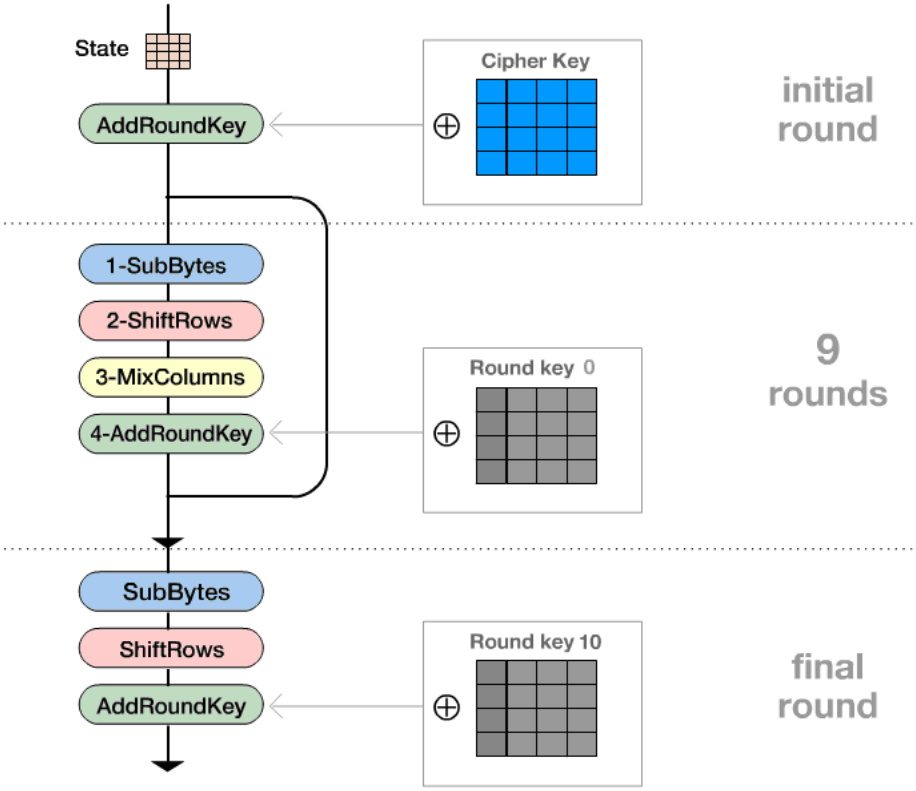


Figure III-1: Algorithme de chiffrement pour AES-RIJINDAEL

I.4.1.1. Sub byte

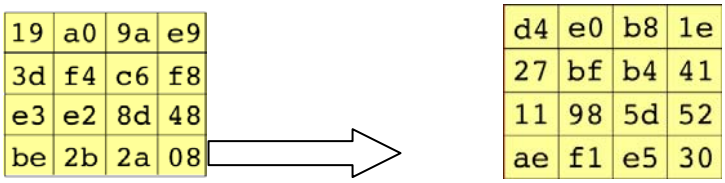
C'est une substitution non linéaire d'octets appliquée a la matrice d'états qui déjà sommé avec un clé initial suivants une table S-box en Hexa donc cette transformation consiste à lire les valeurs d'un octet d'entrée B=b7b6b5b4b3b2b1b0 le poids le plus fort concerne la lecture par rapport a la ligne de S-box et le plus faible concerne la lecture au colonnes.

Tableau 0-1 : Matrice S-box

S	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
8	CD	0C	13	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

Par exemple :

(19) hex= (D4) hex la substitution aura lieu avec la valeur situé à l'intersection de la ligne d'indice 1 et le colonnes d'indice 9.



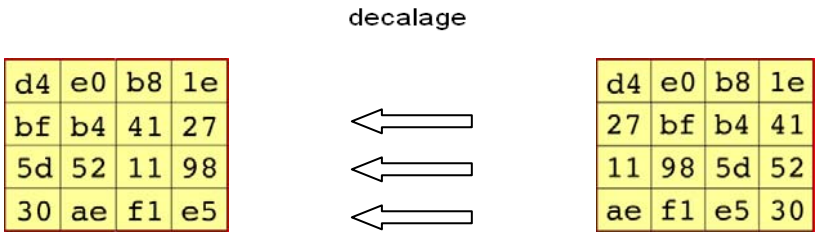
I.4.1.2. Shift Rows

Ce qui concerne cette partie, touts les dernières lignes de la matrice résultant de Sub byte subissent des rotations circulaires vers la gauche

Tableau III-5 : Nombre de décalage par rapport a chaque taille de matrice d'état

N_b	4	6	8
Ligne « 1 »	*1*	*1*	*1*
Ligne « 2 »	*2*	*2*	*3*
Ligne « 3 »	*3*	*3*	*4*

Ce dessous montre l'effet de cette transformation sur la matrice Nb=4.



I.4.1.3. Mix columns

Cette transformation traite chaque colonne de la matrice résultant aussi de transformation Shift Rows comme un polynôme de degré 3 à coefficient dans le champ de GALOIS GF (2^8) et réalise le produit de celle-ci avec un polynômes constant

A(x)= (03) hex. x3+(01)hex.x2 +(02)hex.x+(02)hex

Pour une colonne d'entrée (4 octets) B = (B3B2B1B0)hex, on peut écrire cette opération sous forme matricielle

$$B' = \begin{pmatrix} B'_0 \\ B'_1 \\ B'_2 \\ B'_3 \end{pmatrix} = \begin{pmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{pmatrix} \times \begin{pmatrix} B_0 \\ B_1 \\ B_2 \\ B_3 \end{pmatrix}$$

I.4.1.4. Add round Key

C'est la partie finale ou en somme la matrice avec la clé terme a terme. Pour chaque ronde, il y a une clé extraire a partir de la clé originale dont la dimension est de 4*Nb octets.

I.4.1.5. Génération des clés

Dans cette partie on désire créer des matrices de sous-clé pour additionner dans chaque round (sauf la dernier) dont la taille est de 4*(Nb*(Nr+1)) octets.

Il s'agit ici de voir comment Rjindael opéré pour déduire 10 clés secondaires dont il a besoin pour aboutir au texte chiffré. On considère les deux matrices suivants (Rcon est constante)

Tableau III-6 : Matrice de Rcon

01	02	04	08	10	20	40	80	1b	36
00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00

Tableau III-7 : Clé initiale

2b	28	ab	09
7e	ae	f7	cf
15	d2	15	4f
16	a6	88	3c

Le tableau W est une matrice de 4*(Nb*(Nr+1)) octets qui contient tout les sous clés et la notation W[i] indique la ieme colonne, la transformation subbyte est la même que celle définie précdement par contre Rotword est une nouvelle colonne W' [i] qui n'est d'autre qu'une rotation circulaire vers le haut.

W' [i]=Rotword (W[i])=Rotword $\left(\begin{Bmatrix} W_{0i} \\ W_{1i} \\ W_{2i} \\ W_{3i} \end{Bmatrix}\right) = \left(\begin{Bmatrix} W_{3i} \\ W_{1i} \\ W_{2i} \\ W_{0i} \end{Bmatrix}\right)$

En ce qui concerne Rcon, dite Round constantes,

Pour mieux comprendre l'algorithme de génération des clefs on propose ce schéma de déroulement :

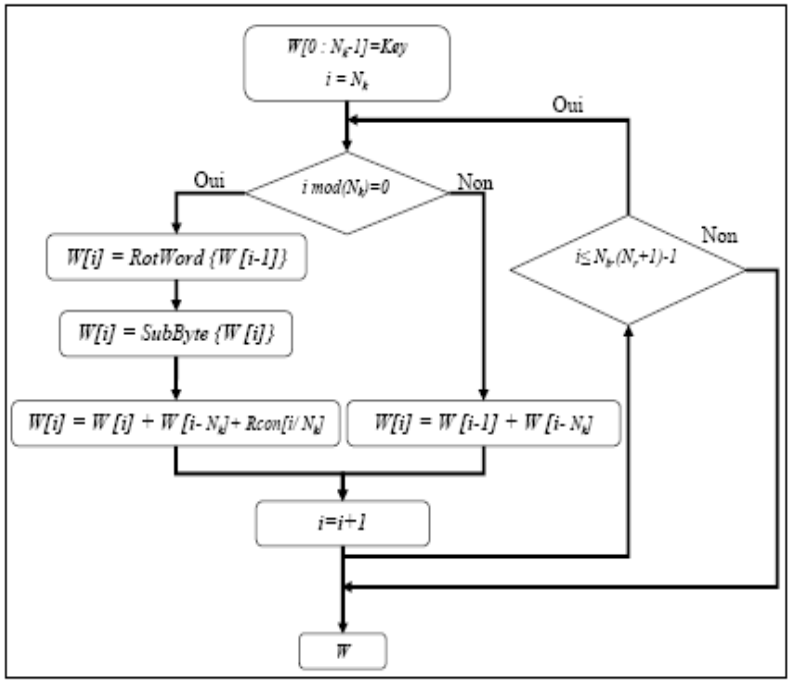


Figure III-2: Plan de génération de sous-clés

I.4.2. Le décryptage

L'idée de décryptage et la même que le cryptage sauf en inverse tout les instructions Matrice, décalage…..Etc. ce qui implique a inversé les transformation invSubByte, InvShiftRows, InvMixcolumns

I.4.2.1. Transformation invSubByte

Elle réalise la substitution inverse introduite par la table S-box suivant une notée InvS-box. Cette table peut être déduite a partir de S-box en lisant pour chaque valeur de cette dernière la ligne et la colonne correspondantes.

Par exemple :

(D4) hex = (19) hex

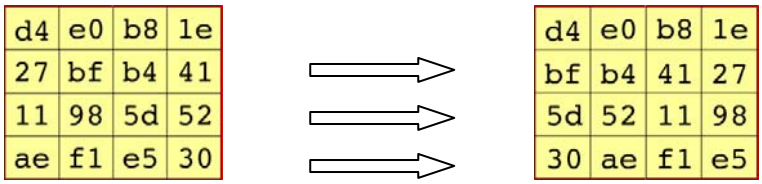
NB : pour vérifier que notre substitution est juste, il faut que le résultat de sub byte lise être l'inverse du résultat obtenant dans l' invSubByte :

Tableau III-8 : Matrice InvS-box

Inv	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	52	09	6A	D5	30	36	A5	38	BF	40	A3	9E	81	F3	D7	FB
1	7C	E3	39	82	9B	2F	FF	87	34	8E	43	44	C4	DE	E9	CB
2	54	7B	94	32	A6	C2	23	3D	EE	4C	95	0B	42	FA	C3	4E
3	08	2E	A1	66	28	D9	24	B2	76	5B	A2	49	6D	8B	D1	25
4	72	F8	F6	64	86	68	98	16	D4	A4	5C	CC	5D	65	B6	92
5	6C	70	48	50	FD	ED	B9	DA	5E	15	46	57	A7	8D	9D	84
6	90	D8	AB	00	8C	BC	D3	0A	F7	E4	58	05	B8	B3	45	06
7	D0	2C	1E	8F	CA	3F	0F	02	C1	AF	BD	03	01	13	8A	6B
8	3A	91	11	41	4F	67	DC	EA	97	F2	CF	CE	F0	B4	E6	73
9	96	AC	74	22	E7	AD	35	85	E2	F9	37	E8	1C	75	DF	6E
A	47	F1	1A	71	1D	29	C5	89	6F	B7	62	0E	AA	18	BE	1B
B	FC	56	3E	4B	C6	D2	79	20	9A	DB	C0	FE	78	CD	5A	F4
C	1F	DD	A8	33	88	07	C7	31	B1	12	10	59	27	80	EC	5F
D	60	51	7F	A9	19	B5	4A	0D	2D	E5	7A	9F	93	C9	9C	EF
E	A0	E0	3B	4D	AE	2A	F5	B0	C8	EB	BB	3C	83	53	99	61
F	17	2B	04	7E	BA	77	D6	26	E1	69	14	63	55	21	0C	7D

I.4.2.2. Transformation InvShiftRows

Elle consiste à appliquer des rotations circulaires vers la droite pour les trois dernières lignes inverses à ceux introduites par l'opération ShiftRows



I.4.2.3. La transformation InvMixcolumns

La transformation InvMixcolumns, en traitant chaque colonne de la matrice comme étant un polynôme de degré 3 à coefficient dans GF (2^8), réalise un produit polynomial :

$$A^{-1}(x) = (0B)_{hex}.x_3 + (0D)_{hex}.x_2 + (09)_{hex}.x + (0E)_{hex}$$

Sous forme matricielle, pour un mot d'entré de 32 bits B= (B3B2B1B0)

$$B' = \begin{pmatrix} B'_0 \\ B'_1 \\ B'_2 \\ B'_3 \end{pmatrix} = \begin{pmatrix} 0E & 0B & 0D & 09 \\ 09 & 0E & 0B & 0D \\ 0D & 09 & 0E & 0B \\ 0B & 0D & 09 & 0E \end{pmatrix} \times \begin{pmatrix} B_0 \\ B_1 \\ B_2 \\ B_3 \end{pmatrix}$$

le plan général de cryptage et décryptage est représenté à la Figure .

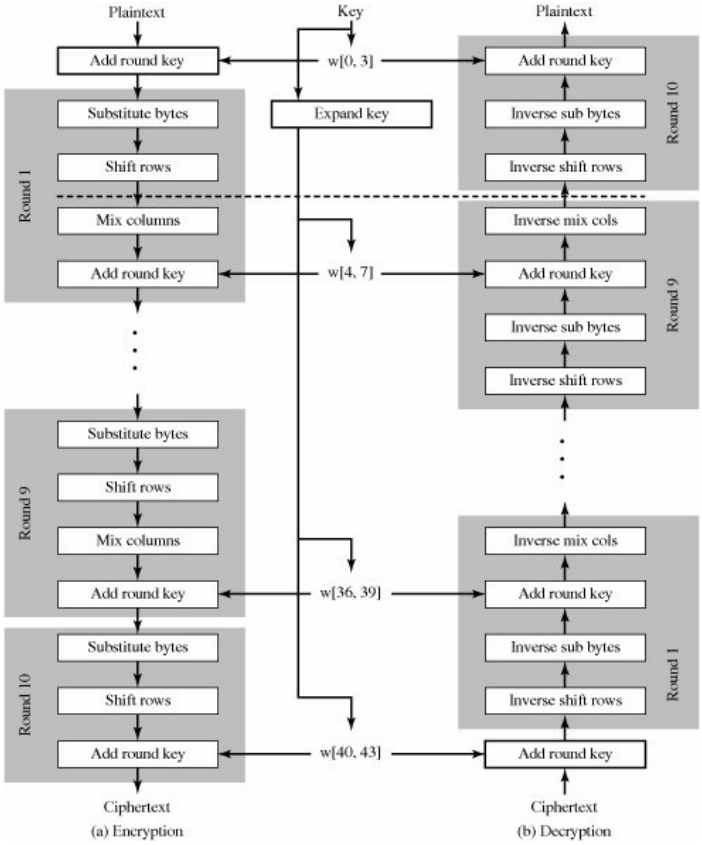


Figure III-3: Plan général de cryptage et décryptage

I.5. Sécurité de l'AES

La structure de l'algorithme RIJINDAEL est très différente de celle adopté par le DES. En effet, L'AES est basé sur la théorie de champs de gallois.ça conception a pris en compte les différentes attaques connues jusqu'à l'année 1997. [13]

Parmi les faiblesses principales du DES, on trouve bien la symétrie pour le traitement du bloc des données à travers une ronde.Cette symétrie est a l'origine de l'existence des clefs faibles, Semi –faibles et potentiellement faibles. Pour l'AES, un soin particulier était attribué pour éliminer toute symétrie .En effet, le processus de déchiffrement utilise des opération inverses calculées dans le GF(2^8) .de plus ,la génération des clefs utilise une table de substitution S-box de caractère non linéaire ce qui exclut toute possibilité d'avoir des clefs faibles. Elle utilise aussi la matrice Round constantes (Rcon), pour renforcer l'asymétrie des sous-cles.De ce fait, les deux concepteurs de RIJINDAEL estiment qu'une attaque à clefs corrélées est peu efficace. [13]

Un autre point fort dans la conception de RIJINDAEL est la S-box .En effet, elle est calculée à partir d'une transformation non linéaire <<x^{-1}>> dans GF(2^8).Cette expression simple peut suggérer un type d'attaque particulier qui est l'interpolation polynomial .cryptanalyse essaye d'approcher la relation entre texte chiffré /texte en clair avec une relation polynomiale .pour renforcer la résistance contre cette attaque, la transformation affine est ajoutée .D'après joan Deamen et vincent Rijmen, l'interpolation polynomiale ne peut fonctionner que pour un nombre réduit de rondes . [13]

Pour l'opération ShiftRows, elle était ajoutée pour contrer deux attaques qui sont l'attaque des différences tronquées et l'attaque square.

Pour l'opération MixColumns, elle produit un grand effet de diffusion à travers la matrice d'état.un changement d'un seul octet induit un changement de quatre octets en sortie

Comme résultat final, il n'existe pas d'attaque dont la complexité est inférieure à celle de l'attaque à force brute avec un nombre de rondes supérieur a six.Ainsi la sécurité estimée de l'AES est de :

2^{128} Pour une taille de clé de 128 bits

2^{192} Pour une taille de clé de 192 bits

2^{255} Pour une taille de clé de 256 bits

Ceci offre une grande marge de sécurité vis-à-vis toutes les attaques connues mais n'exclut pas l'apparition de nouvelles techniques cryptanalytiques.

I.6. Conclusion

Dans ce chapitre on a vu le déroulement de l'algorithme de chiffrement standard AES.

L'AES remplace le DES même le Triple DES dans le chiffrement standard symétrique grâce a son niveau de sécurité très élevé par rapport a DES pour cette raison on a choisi l'AES comme bloc de chiffrement dans le PFE pour crypter le signal vocale résultant de l'analyse.

CHAPITRE IV : RESULTATS DE SIMULATION

I.1. Introduction

Dans ce chapitre nous allons expose les résultats des travaux effectué dans le but de simulé les blocs de la chaîne (codage, cryptage).

Pour cela on a servi des outils suivants :

- le programme GOLDWAVE avec lequel on a enregistré le mot "السلام عليكم" sous un fichier (*.txt).
- le programme MATLAB pour réalise les deux bloc sous SIMULIK.

I.2. Plan général de déroulement de la simulation

Pour simuler l'analyse/synthèse LPC on a programmé sous Matlab les fonctions suivantes :

I.2.1. lpc_demo

C'est le programme principal qui représente notre chaîne de simulation de codeur/décodeur source, il fait appel aux fonctions d'analyse/synthèse de la parole surnom « analyse_lpcS/synthese_lpcS » ainsi que la fonction de codage/décodage surnom « codage_S/ decodage_S ».

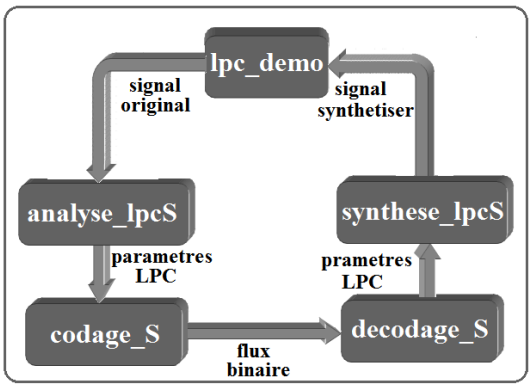


Figure IV-1: Plan général de simulation

I.2.1.1. Analyse_lpcS

C'est la fonction charger d'extraire les paramètres LPC a l'aide de ces trois programmes :

- pitch_S : Elle représente l'algorithme de SIFT qui sert au calcul de la fréquence fondamentale.
- LPC : c'est une fonction qui existe déjà dans Matlab, donc on a utilise directement pour calculer les cœfficients de réflexion et l'erreur de prédiction.
- Gains_S : cette fonction compris l'algorithme de calcule de gain on utilisant la même relation cite au chapitre II.

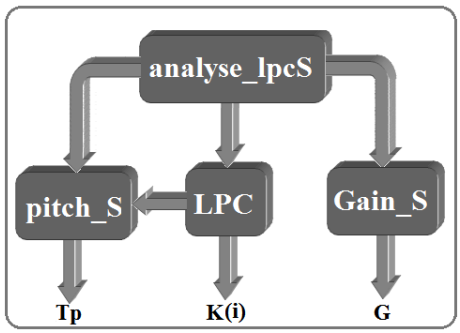


Figure IV-2: Plan général de l'analyse LPC

I.2.1.2. Codage_S

Pour envoyer les paramètres LPC obtenus par l'analyse vers l'étage de cryptage on doit les convertir en binaire, mais d'après ce qu'on a remarqué les valeurs des ces trois paramètres

Ne varie pas de la même façon ce qui nous a ramené a faire ces trois programmes de logique différent :

- COD_SIG1 : fonction compris l'algorithme de codage de gain.
- COD_SIG2 : fonction compris l'algorithme de codage de la fréquence fondamental.
- COD_SIG3 : fonction compris l'algorithme de codage des coefficients de réflexion.

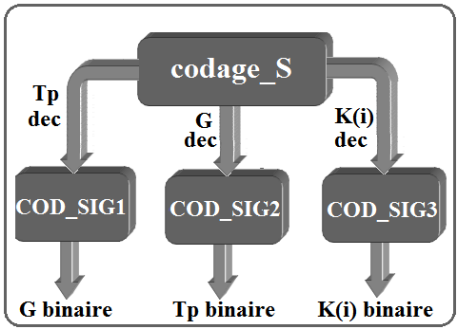


Figure IV-3: Plan de la fonction codage des paramètres

I.2.1.3. Décodage_S

Les paramètres binaire issu de l'étage de décryptage sont convertie au décimal grâce a cette fonction qui fait appel aux fonctions suivants :

- DEC_SIG1 : fonction compris algorithme de décodage de gain.
- DEC_SIG2 : fonction compris algorithme de décodage de la fréquence fondamental.
- DEC_SIG3 : fonction compris algorithme de décodage des coefficients de réflexion.

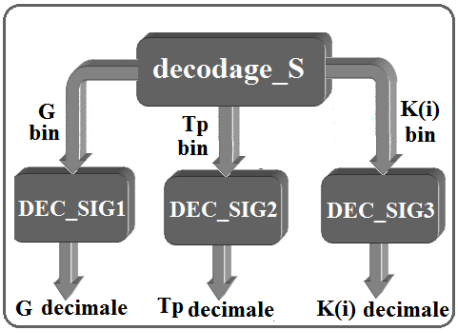


Figure IV-4: Plan de la fonction de décodage

I.2.1.4. synthese_lpcS

C'est le programme principale qui permet d'avoir un signal synthétise il fait appel au fonction suivants :

- generate_imp1 : programme de génération des impulsions périodiques.
- generate_bruit : programme de génération de bruit.
- switch : programme qui choisie le type d'excitation selon la trame reçus (voise ou non).
- filter : on a utilise cette fonction qui existe dans Matlab pour générer le filtre de synthèse.

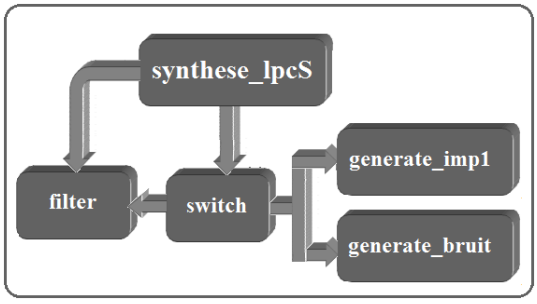


Figure IV-5: Plan de la fonction synthèse

I.3. Résultat de la simulation

I.3.1. Résultat de calcul des coefficients de réflexion

Le Tableau illustre quelque valeur calculer des cœfficients de PARCORR ainsi que leur valeur codé (sur 8 bit compris 1 bit pour le signe), les valeurs obtenus après le décodage et l'erreur due au algorithme de codage.

Dans notre cas si la valeur de cette erreur est important l'écart entre les coefficients codé et décodé devient non négligeable ce qui agir directement sur la qualité de signal synthétisé.

Le Tableau illustre les plages de variation des valeurs de coefficient en % pou 59 trames de mot "السلام عليكم" en faite quelque soit le signal de parole a traité ces coefficient prendre une valeur entre 1 et -1, dans notre cas on a utilise cette table surtout pour la réalisation des programmes de codage et de décodage des coefficients.

Tableau IV-1 : Résultat de simulation concerne les coefficients

	Ki	Ki(BIN)	Ki'	ΔK_i		Ki	Ki(BIN)	Ki'	ΔK_i
Trame : 1	1.0000	01100100	1.0081	-0.0085	Trame : 2	1.0000	01100100	1.0097	-0.0005
	-1.0883	11101100	-1.0891	0.0010		-1.1946	11110111	-1.1996	-0.0036
	0.3906	00100111	0.3913	-0.0062		0.7561	01001011	0.7549	-0.0021
	-0.2323	10010111	-0.2391	0.0053		-0.5471	10110110	-0.5480	-0.0002
	0.2058	00010100	0.2063	-0.0016		0.5515	00110111	0.5514	-0.0017
	-0.3454	10100010	-0.3410	-0.0015		-0.5310	10110101	-0.5342	0.0085
	0.5554	00110111	0.5528	-0.0012		0.4972	00110001	0.4992	0.0069
	-0.1898	10010010	-0.1855	-0.0081		-0.3847	10100110	-0.3879	-0.0003
	-0.0932	10001001	-0.0996	0.0039		0.2997	00011101	0.2996	0.0059
	-0.0662	10000110	-0.0696	-0.0059		-0.2012	10010100	-0.2066	0.0065
	0.0954	00001001	0.0916	0.0026		-0.0169	10000001	-0.0104	0.0011
	Ki	Ki(BIN)	Ki'	ΔK_i		Ki	Ki(BIN)	Ki'	ΔK_i
Trame : 3	1.0000	01100100	1.0019	-0.0019	Trame : 4	1.0000	01100100	1.0055	-0.0055
	-1.0213	11100110	-1.0249	0.0036		-1.0090	11100100	-1.0014	-0.0076
	0.6777	01000011	0.6745	0.0032		0.8479	01010100	0.8415	0.0064
	-0.4799	10101111	-0.4765	-0.0034		-0.7481	11001010	-0.7426	-0.0055
	0.4171	00101001	0.4171	0.0000		0.6172	00111101	0.6184	-0.0012
	-0.4267	10101010	-0.4275	0.0008		-0.5944	10111011	-0.5925	-0.0019
	0.2470	00011000	0.2428	0.0042		0.5015	00110010	0.5081	-0.0066
	-0.1466	10001110	-0.1468	0.0002		-0.3909	10100111	-0.3924	0.0015
	0.2396	00010111	0.2366	0.0030		0.2799	00011011	0.2793	0.0006
	-0.1878	10010010	-0.1816	-0.0062		-0.1173	10001011	-0.1135	-0.0038
	-0.0061	10000000	-0.0012	-0.0049		-0.0467	10000100	-0.0420	-0.0047

Ki : les coefficients de réflexion a transmettre

Ki' : les coefficients de réflexion obtenus a la réception

$\Delta K_i : represent\_l'erreur\_du\_au\_quantification$

Tableau IV-2 : Table de plage de variation pour les Ki

	K1	K2	K3	K4	K5	K6	K7	K8	K9	K10	*K11*
0.xxxx	0%	100%	65%	61%	40%	40%	38%	42%	29%	25%	*21%*
0.0xxx	0%	0%	31%	34%	51%	46%	53%	48%	59%	65%	*70%*
0.00xx	0%	0%	2.12%	4%	8%	12%	8%	8%	8%	8%	*6%*
0.000x	*0%*	*0%*	*0%*	*0%*	*0%*	*0%*	*0%*	*0%*	*2%*	*0%*	*2%*

I.3.2. Vérification de la stabilité de filtre de synthèse

La stabilité du filtre est déduite a partir des coefficients parcours Ki, celui-ci est dit stable si est seulement si les Ki sont inférieurs a l'unité.

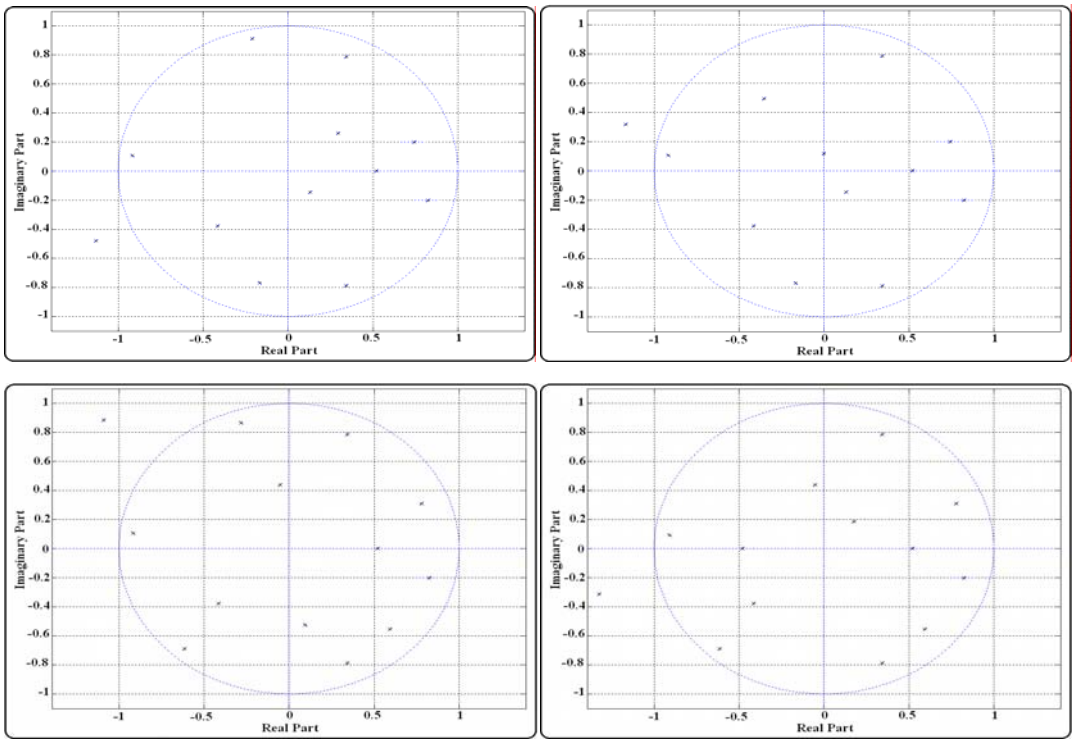


Figure IV-7: Stabilité de filtre de synthèse

I.3.3. Résultat de calcule de gain et la fréquence fondamental

La fréquence fondamentale a été détectée pour chaque trame par l'algorithme de SIFT

(voir Chapitre II).

On donne tous les résultats sous forme de tableau pour le temps de pitch (ms), la fréquence fondamentale ainsi que la valeur codé et décodé puis l'erreur de codage.

Tableau IV-3 : Résultat de calcul de fréquence fondamental

	Trame1	Trame2	Trame3	Tram4	Trame5	Trame6	Trame7	Trame8
Tp(sec)	0.0054	0.0026	0.0024	0.0028	0.0036	0.0025	0.0042	0.0073
Tp(BIN)	10001010	01000010	00111101	01000111	01011100	01000000	01101011	10111010
Tp'(sec)	0.0054	0.0026	0.0024	0.0028	0.0036	0.0025	0.0042	0.0073
$\Delta Tp(10^{-4})$	0.0938	0.2187	0.1719	0.2656	0.0625	0	0.2031	0.3437
Fp(Hz)	185.18	384.61	416.66	357.14	277.77	400	238.095	136.98

Le gain est calculé à partir des erreurs déduites pour chaque trame, les résultats de quelque trame sont donnés sur la table suivant.

Tableau IV-4 : Résultat de calcul de Gain

	Trame1	Trame2	Trame3	Tram4	Trame5	Trame6	Trame7	Trame8
Gain	0.0309	0.0328	0.0323	0.0442	0.0629	0.0368	0.0323	0.0324
G(BIN)	00000111	00001000	00001000	00001011	00010000	00001001	00001000	00001000
Gain’	0.0273	0.0313	0.0313	0.0430	0.0625	0.0352	0.0313	0.0313
ΔG	0.0036	0.0016	0.0011	0.0012	0.0004	0.0016	0.0011	0.0011

La figure suivante représente la valeur codé et décode de gain (1), les séquences binaires de chaque valeur de gain a envoyé pour ces trames (2) ainsi que la représentation de l’erreur due au algorithme de codage ,on peut dire la même chose pour la fréquence fondamental(4)(5)(6).

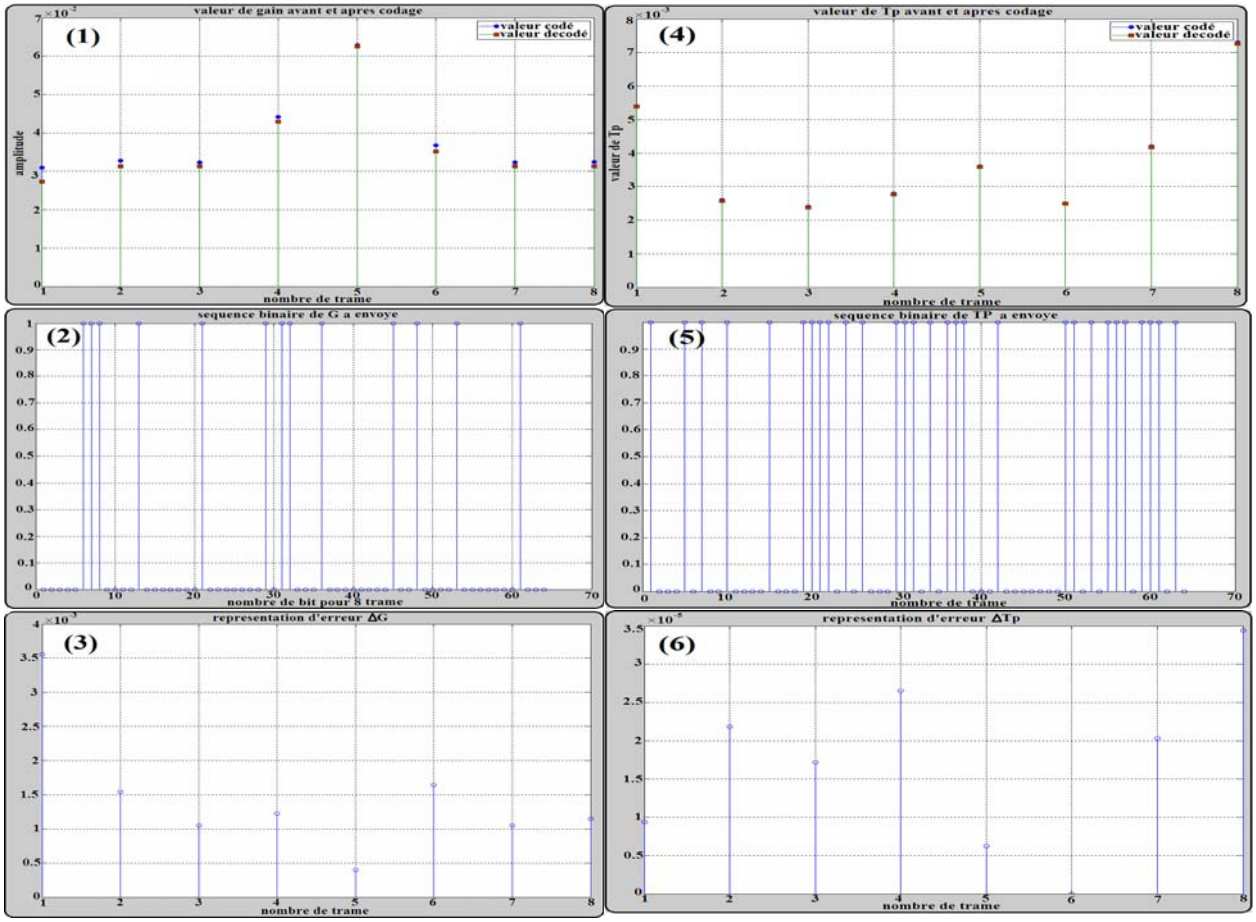


Figure IV-8: Valeur codé et décode de fréquence fondamental et de gain

I.3.4. Simulation sous SIMULINK

Pour cette partie on utilise le bloc EMBEDDED FUNCTION MATLAB (USER-DEFINDE-FUNCTION) pour réalise la chaîne analyse/synthèse sous SIMULINK

Les différents blocs utilisent dans cette partie sont :

- Constant : bloc utilise pour charge le signal parole
- Analyse LPC : bloc qui effectue l'analyse LPC
- Synthèse LPC : bloc qui effectue la synthèse LPC
- Audio device : bloc utilise pour l'écoute de signal
- Spectrogramme : bloc utilise pour trace les spectrogrammes des signaux
- DES_SIGNAL : bloc utilise pour trace la densité spectrale d'énergie des signaux

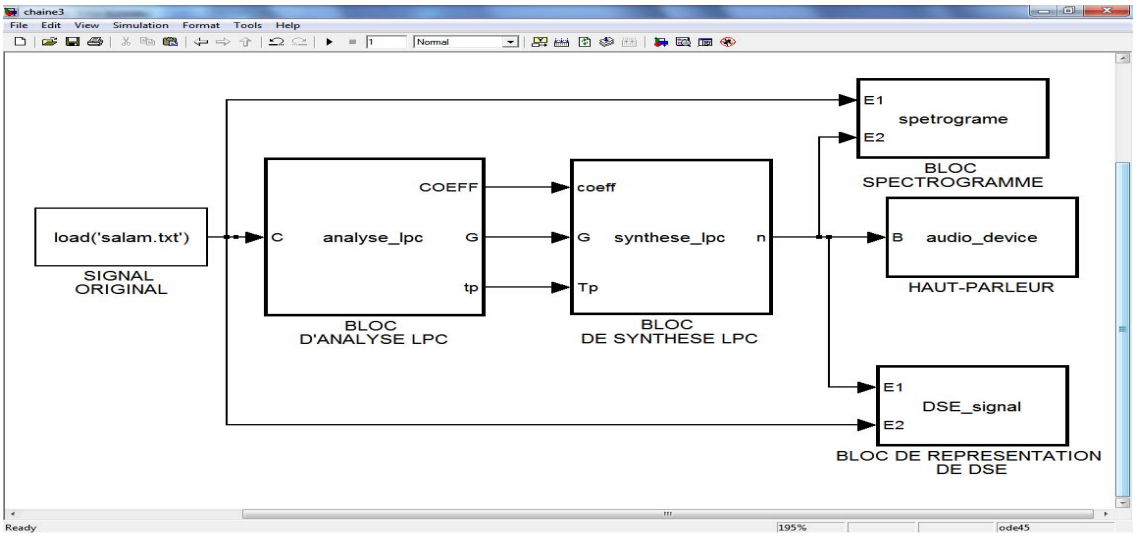


Figure IV-9: Simulation de les bloc LPC sous SIMULINK

On ce qui concerne les résultats obtiennent dans cette partie et après l'écoute de signal synthétise on a constate que la qualité n'est pas bonne mais elle reste acceptable et dans les normes.

La cause de cette diminution se voit clairement sur la représentation spectrale des deux signaux qu'est principalement due à la modification d'énergie pour certaine fréquence.

Cependant, en réalité pour mieux connaître les régions de repartitionner d'énergie par rapport au temps et fréquence d'un signal non stationnaire (évolution de fréquence en fonction de temps) Sa serai bien d'utilise ce qu'on appelle le spectrogramme obtenus par une transforme appelle transforme temp-fréquence (STFT), mais dans ce cas un choix de fenêtre de pondération s'impose pour avoir une bonne résolution, cette différence apparaître claire sur (3) spectrogramme généré par une fenêtre 128 point et (4) spectrogramme généré par une fenêtre 16 point , alors on peut dire que pour notre signal on est capable de repère précisément les régions d'énergie modifie qui agir directement sur la qualité de signal synthétise par rapport au temps et au fréquence

Par contre n'en vois aucune modification de fréquence par rapport au signal original.

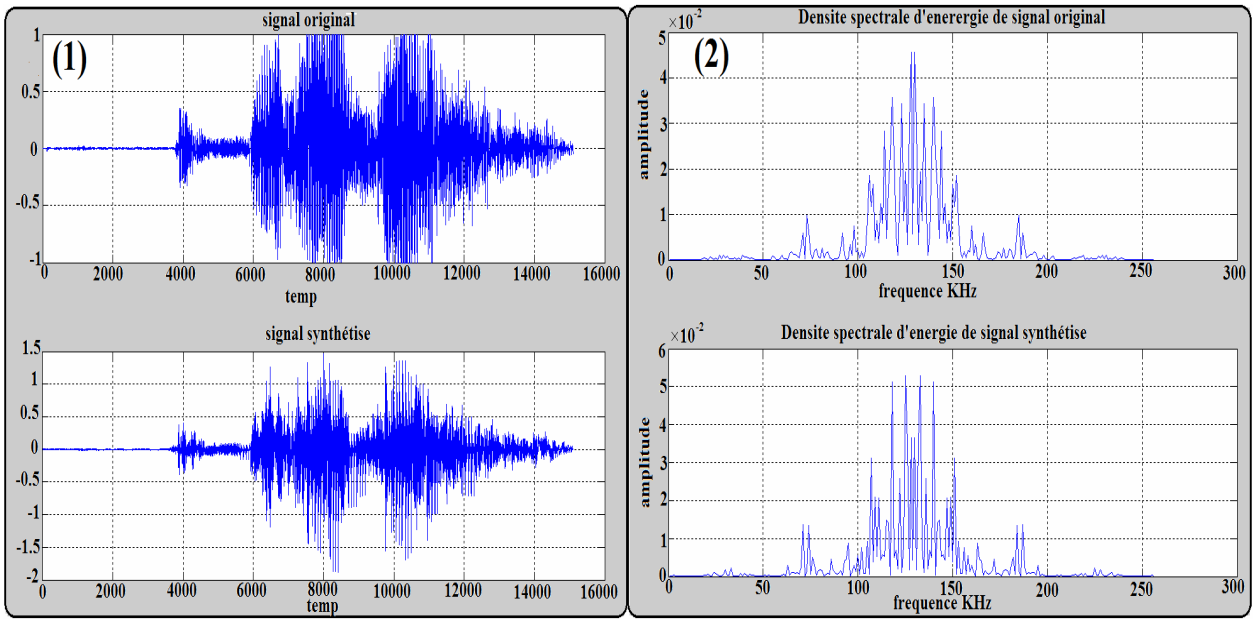


Figure IV-10: Signal original et synthétise ainsi leur DSE

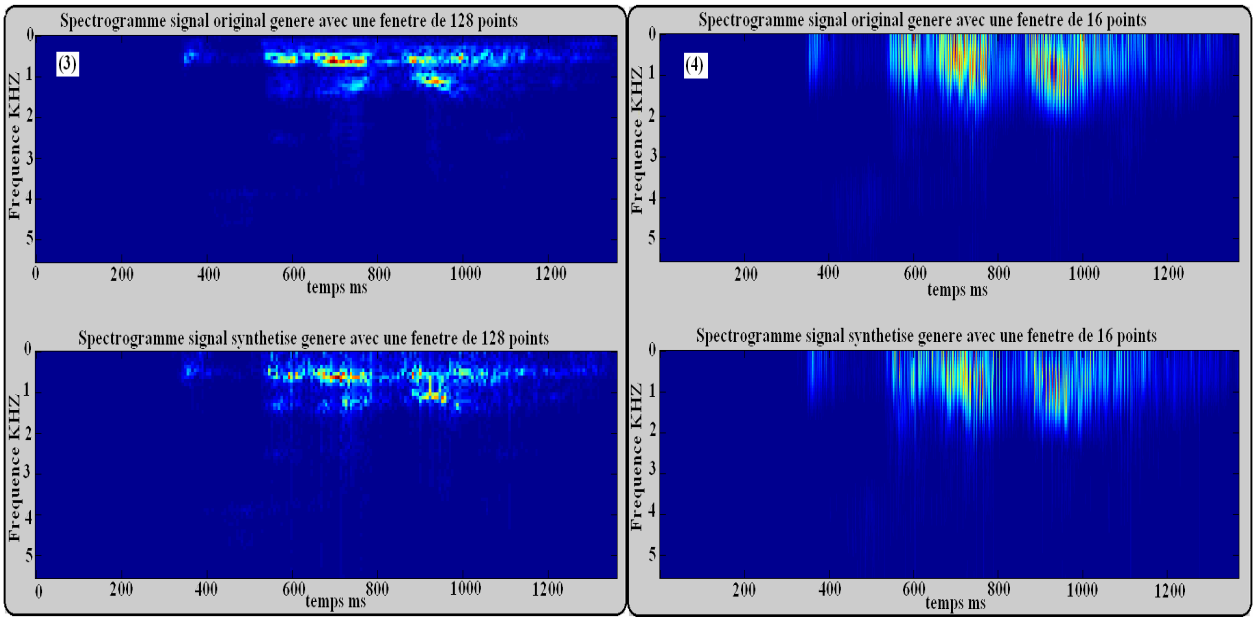


Figure IV-11: Spectrogramme signal original et synthétise

I.4. Simulation de cryptage /décryptage

Dans cette partie on va montrer tout les résultat obtenu par MATLAB et SIMULINK avec des commentaires sur les signaux et les résultats.

Le déroulement de programme cryptage et décryptage sous MATLAB :

I.4.1. Aes_main

C'est le programme principal. Il fait appel à la fonction aes_init pour initialiser les paramètres nécessaires aux fonctions de cryptage cipher et de décryptage inv_cipher.

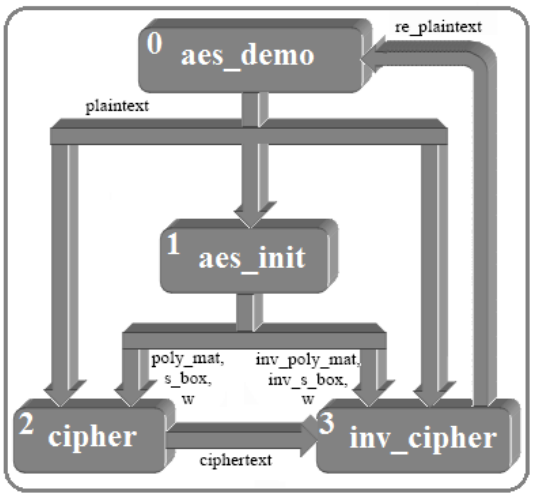


Figure IV-12: Schéma général de simulation de cryptage en MATLAB

I.4.2. Aes_init

Génère les s_box et inv_s_box par appelle à la fonction s-box_gen. Définit le vecteur constant de boucle rcon qui contribue à la génération de la clé interne.

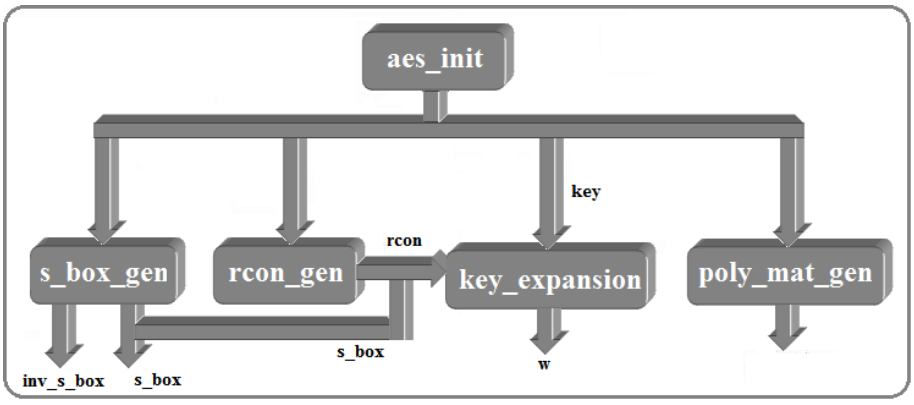


Figure IV-13: Schéma général de génération les constantes

Rcon_gen :

Rcon est utilisé par la fonction key_expansion avec la s_box et la clé de l'utilisateur pour générer une clé interne image w.

S_box_gen :

Les s_box sont utilisées par les fonctions key_expansion, cipher et inv_cipher pour réaliser une substitution de bits dans le corps de Galois GF (2^8) par des bits du même corps.

Poly_Mat_Gen:

Les deux matrices poly_mat et inv_poly_mat générées par cette fonction sont utilisées par la fonction mix_comlumns lorsqu'elle est appelée par cipher et inv_cipher. Ces deux matrices sont des matrices 4X4 qui ont chaque case d'élément comme le résultat

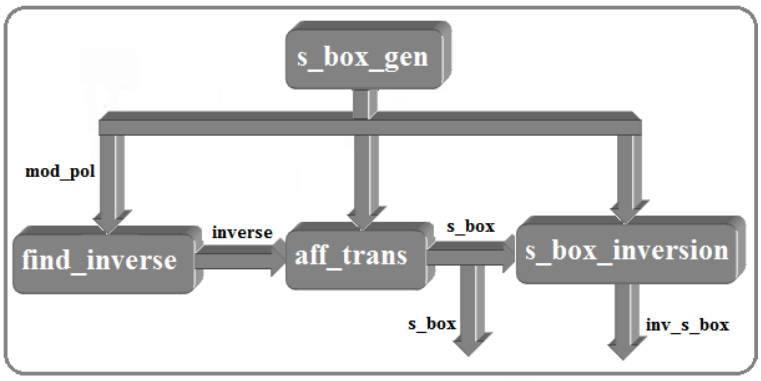


Figure IV-14: Génération des s_box

Fonction Find_Inverse :

La première étape dans la génération des s_box est la recherche des inverses de tous les éléments du corps de Galois GF (28). Pour les 256 bits possibles b, on cherche les bites b-1 satisfaisant à b*b-1=1

Fonction Aff_Trans:

Apres l'obtention des inverses de tous les octets, l'étape suivante dans le processus de génération des s_box est une transformation affine consistant en une multiplication par une constante de spécification (31d=00011111b), modulo une autre constante (257d=100000001) et un xor avec (99d=01100011b). bout = bin ∗ 31d mod 257d xor(99d)

Fonction S_box_inversion:

La table de substitution inverse est utilisée dans la fonction de décryptage pour obtenir le bit en claire correspondant au texte crypté

I.4.3. Key_expansion (Expansion de la clé)

Elle prend la clé fournie par l'utilisateur Key (128 bits), utilise la matrice rcon et la s_box (16*16) pour générer une nouvelle clé w de longueur 167 octets (1408 bits) qui sera utilisée dans le processus.

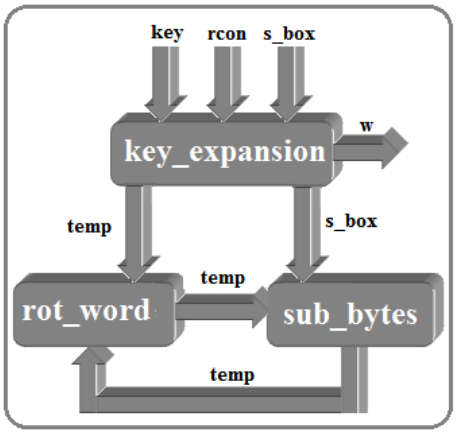


Figure IV-15: Expansion de la clé

Fonction Rot_Word:

La fonction Rot_Word réalise une permutation cyclique entre les éléments d'un mot.

I.4.4. Le chiffrement par Cipher

La fonction Cipher réalise le chiffrement d'un bloc de 128, 192 ou de 256 bits selon l'utilisation

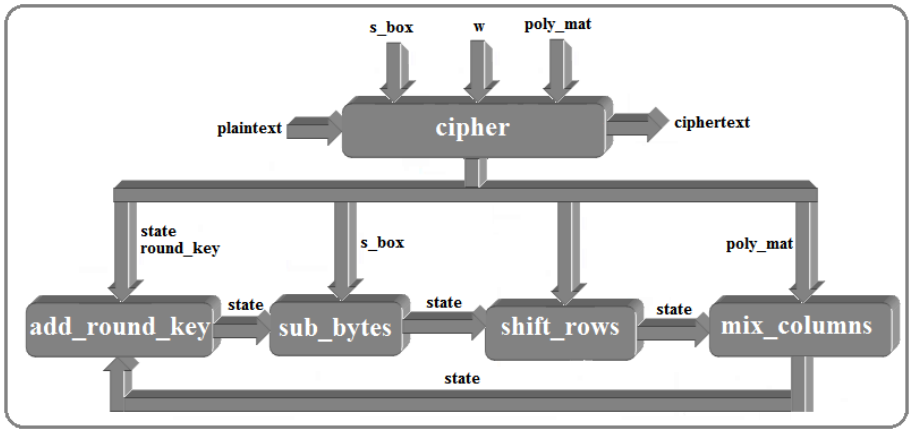


Figure IV-16: La fonction Cipher qui donne le chiffrement

I.4.5. Le déchiffrement par Inv_Cipher

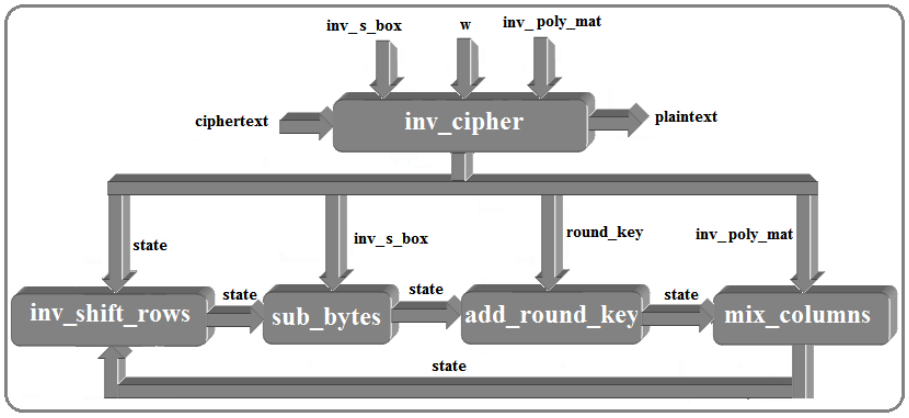


Figure IV-17: La fonction Inv_Cipher qui donne le déchiffrement

La fonction cipher inverse les transformations opérées dans le processus de cryptage.Les paramètres d'entée sont le texte crypté (à décrypter), la table de substitution inverse (inv_s_box), la clé (w) et la matrice polynomiale inverse inv_poly_mat. Etant donné que shift_rows réalise des décalages à gauche, inv_shift_rows réalise simplement des décalages à droite de la matrice d'état STATE.

I.4.6. Résultat de simulation sous MATLAB

Chiffrement (128bit) :

Key	texte en clair	Texte chiffré :
2b 7e 15 16	00 44 88 cc	e9 b8 91 e9
28 ae d2 a6	11 55 99 dd	38 53 5f 59
ab f7 15 88	22 66 aa ee	63 f2 7b f6
09 cf 4f 3c	33 77 bb ff	e8 60 38 aa

Déchiffrement (128bit) :

Key	texte en clair	Texte déchiffré :
2b 7e 15 16	e9 b8 91 e9	00 44 88 cc
28 ae d2 a6	38 53 5f 59	11 55 99 dd
ab f7 15 88	63 f2 7b f6	22 66 aa ee
09 cf 4f 3c	e8 60 38 aa	33 77 bb ff

Les matrices au dessus montre qu'on faire un algorithme de cryptage juste, car l'information est chiffrée et on reçois le texte chiffré puis pour le déchiffrement on injecte l'information chiffré dans le programme de déchiffrement après les opération de déchiffrement on trouve notre information ou bien texte en clair sans perte d'information.

I.4.7. Simulation sous SIMULINK

Pour simuler la chaîne cryptage/décryptage on utilise le même bloc cité a la partie simulation Analyse synthèse LPC, on utilise directement signal parole charger a l'aide de bloc source « constante » et qu'il a été convertir en binaire grâce au bloc de codage, une fois séquence binaire est obtenu le signal peuvent être crypte puis décrypte

A la fin on récupère notre signal à l'aide de bloc de décodage (conversion binaire-decimal)

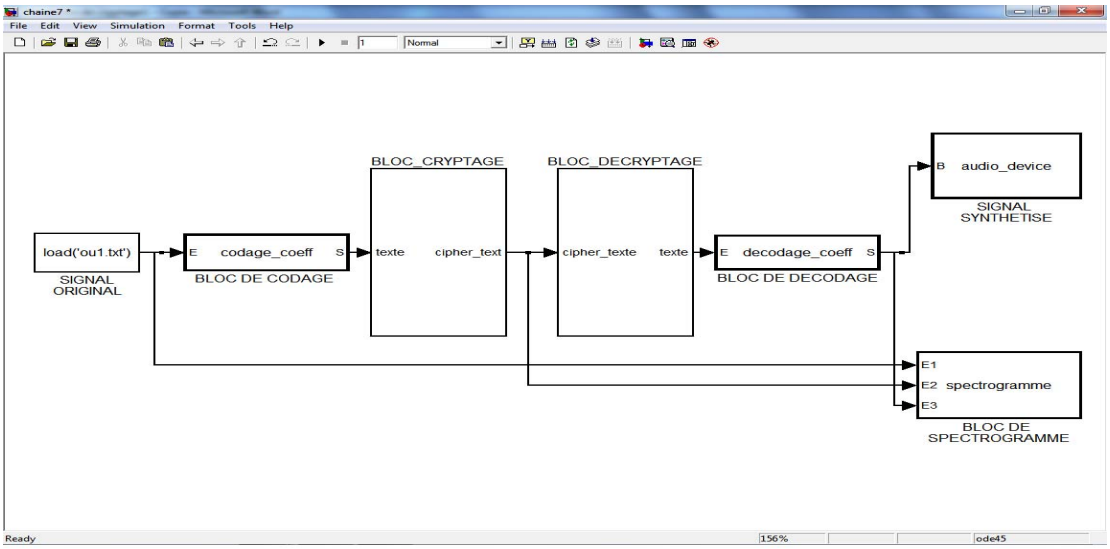


Figure IV-18: Simulation du bloc de chiffrement avec un signal parole

Dans cette partie on confirme l'efficacité du bloc de cryptage qu'est destiné a sécuriser le signal parole a transmettre de telle façon on dissimule l'énergie total ainsi que son contenu fréquentiel.

On remarque que le signal crypté relativement constante ceci est similaire au comportement d'un bruit blanc donc le signal crypté n'est autre que le signal d'origine noyé dans un bruit.

C'est pour ça il sera impossible d'extraire une information sur le signal original, le tracé de spectrogramme confirme que l'énergie le plus important de signal crypté existe sur tout la bande de fréquence qui s'étale de 0 a 1 KHZ le long de domaine temporelle.

En fin on peut dire que le bloc de décryptage n'introduit pas des modifications qui peuvent altérer les caractéristiques de signal original (Amplitude, fréquence) et ceci confirmer aussi par (3) et les spectrogrammes (4), (6)

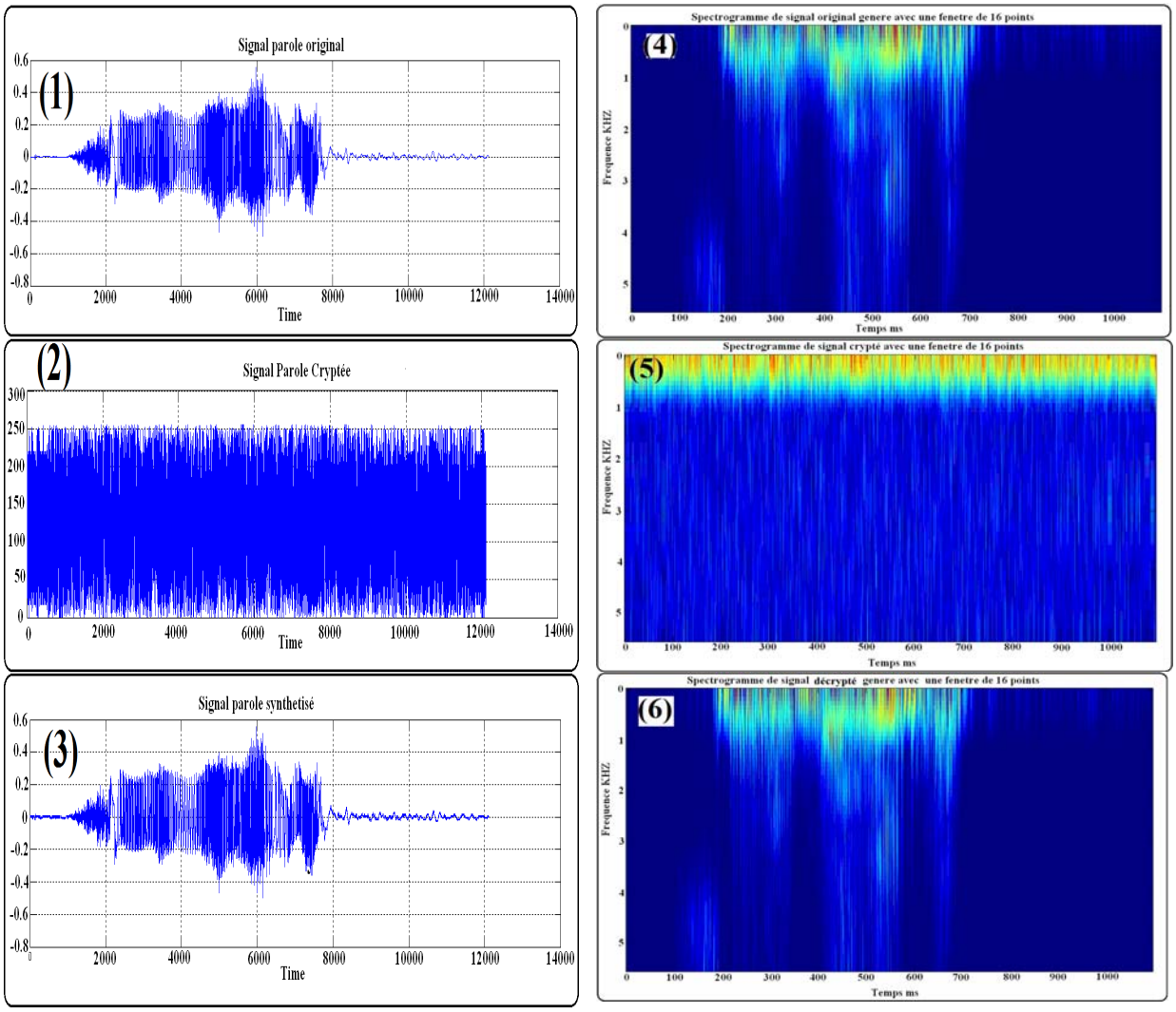


Figure IV-19: Résultat de simulation de bloc de cryptage

Le but de la simulation intégré en premier temps est de confirmer le fonctionnement des bloc et aussi voir est ce qu'il y a des problème d'adaptation entre ces bloc et deuxièmement on cherche a localisé l'étage qui agir sur la qualité de signal parole.Alors après la simulation intégré on constate que cette dégradation est due surtout au méthode et algorithme d'analyse et aussi les erreurs de quantification.

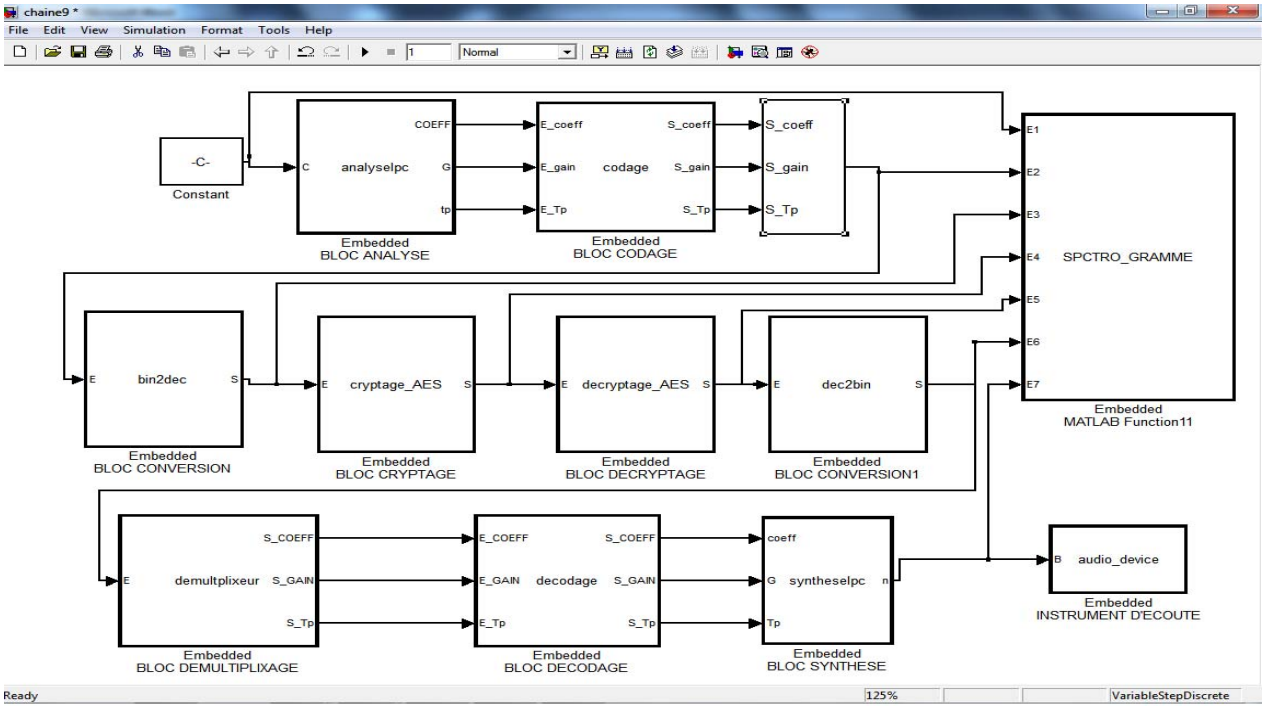
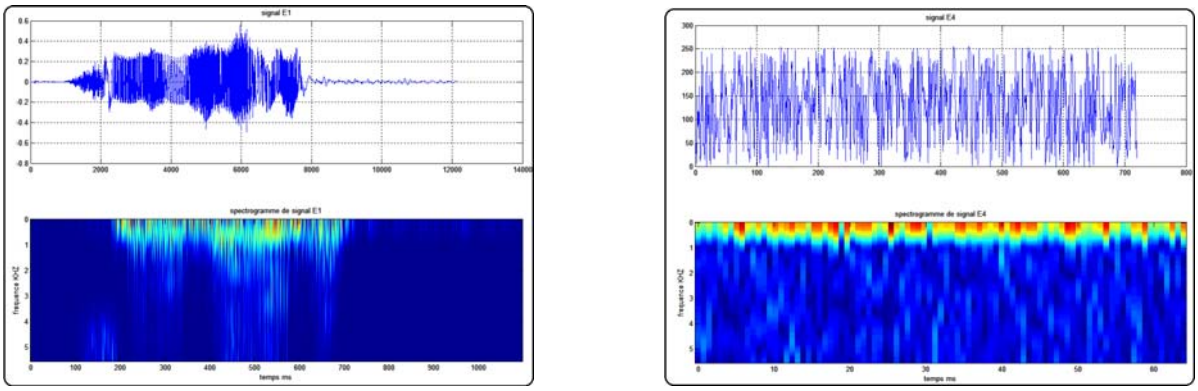


Figure IV-20: Représentation de la chaîne de communication numérique sous SIMULINK



(a) **(b)**

Figure IV-22: (a) Signal original a transmettre, **(b)** cryptage des paramètres a transmettre

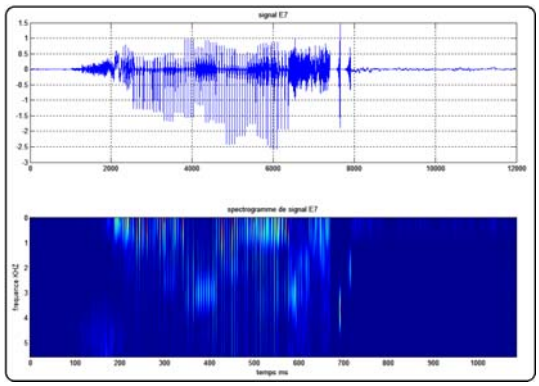


Figure IV-22: Signal parole synthétise

I.5. Conclusion

Dans ce chapitre nous avons rapporté les résultats des différentes phases de validation de l'implémentation software d'un algorithme de codage et de cryptage, la qualité du signal synthétisé est acceptable mais il y aura des déformation dans ce signal a cause de l'effet tuyau et parce que on fabriqué un signal robot c'est pas un signal vocal humaine

Pour le cryptage,on crypte le signal parole a l'aide d'un clefs de 128 bits et on peut dire que le signal crypté n'est autre que le signal parole noyé dans un bruit pour cela le système de chiffrement AES-RIJINDAEL est efficient pour enfouir le signal parole lors de transmission

Conclusion Générale

Le projet qui nous a été proposé a pour objectif d'implémenter en temps réel une chaîne de communication numérique (le codage LPC et le cryptage de signal parole).

On été intéressés principalement au codeur de parole analyse par synthèse LPC qui assure un débit étalé sur une bande de 2.4à16 Kbits/s, comme le modèle LPC est l'un des modèles de base dans le traitement de la parole, la qualité de signal synthétisée reste acceptable par rapport aux taux de compression présentée (13 paramètre par trame au lieu de 256 échantillons) et un temps de calcul réduit.

Le débit de notre codeur source est de 4,3 Kbit/s obtenus grâce à une quantification uniforme et un codage sur 8 bits (pour chaque paramètre) puis le bloc de cryptage qui va assurer la confidentialité de notre signal parole à transmettre.

A travers ce projet, nous avons eu l'opportunité de nous familiariser avec des domaines très variés, tel que le traitement de la parole qui nous a permis d'approfondir nos connaissances dans le domaine du traitement numérique du signal, la cryptographie dont son étude et sa pratique concrète nous a convaincu de son importance dans les sociétés qui ont des bureau d'étude et les centres de recherche qui s'intéressent sur les chaînes de communication numérique sur puce S.O.C

Nous avons pu aussi de familiariser avec les outils de développement de programmes pour FPGA.

En perspectives nous proposons une implémentation de ces deux blocs sur un FPGA

Pour améliorer la qualité de signal synthétise en se basant sur ce qu'on a fait :

- ✓ Amélioration l'algorithme d'analyse ainsi que de synthèse.
- ✓ Ajoutons des algorithmes de lissage des coefficients.
- ✓ Amélioration d'algorithme de détection de pitch.
- ✓ Un algorithme qui minimise l'effet tuyau

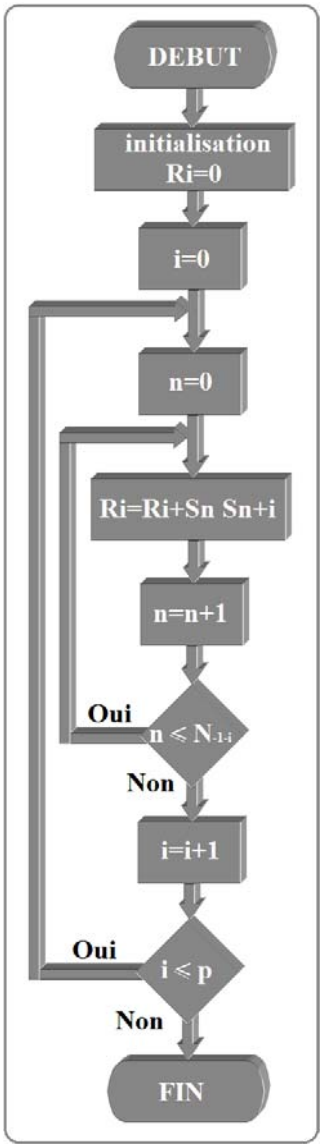
Pour compléter l'implémentation de cette chaîne de transmission numérique :

- ✓ implémentation de bloc codage canal (codage convolutif, décodage viterbi) sur FPGA.
- ✓ implémentation de bloc de modulation (modulation OFDM) sur FPGA.
- ✓ Ajoutons un étage HF pour réalise une transmission sans fil.

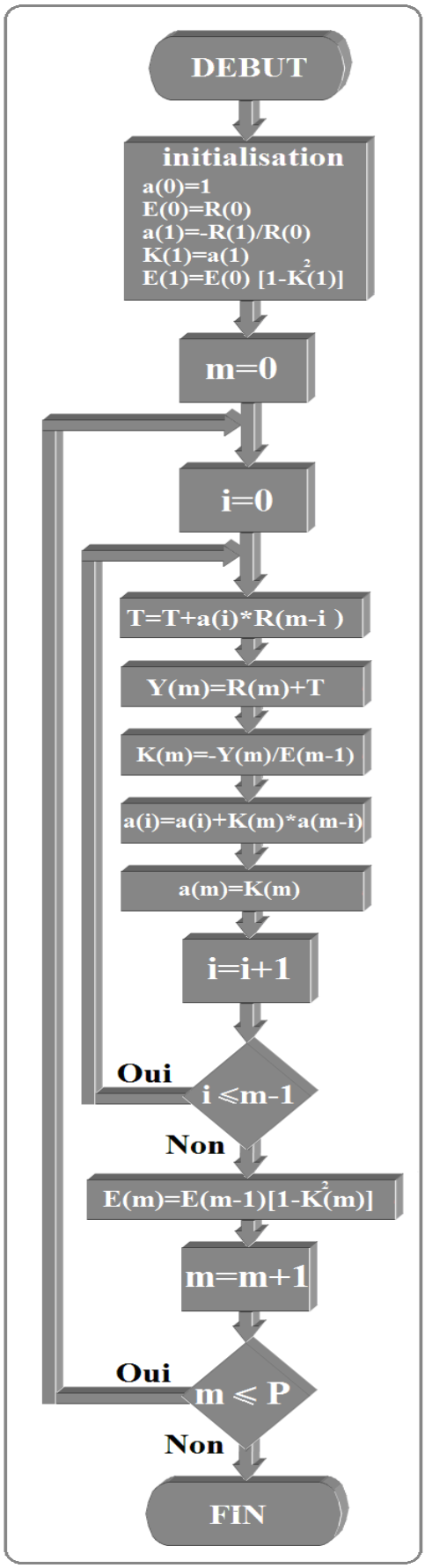
Par defaut de temps la partie propre à l'implementation sur FPGA n'a pas été réalisée

Annexe

A.1 L'organigramme de l'autocorrection



A.2 Organigramme d'algorithme de Levinson :



A.2 Le D.E.S

D.E.S, pour Data Encryptions Standard « standard de cryptage de données », est un algorithme très répandu à clé privée créé à l'origine par IBM en 1977. Il sert à la confidentialité des données et conçu pour être implémenté directement en machine. Il a été jugé si difficile à percer par le gouvernement des Etats-Unis qu'il a été adopté par le ministère de la défense. DES a été développé par les chercheurs d'IBM pour satisfaire la demande des banques.

L'algorithme DES[18] est un algorithme de cryptographie par blocs. Il sert à crypter une série de blocs de 64 bits (8 octets) en utilisant une clé de 56 bits. Le DES est à 16 rondes c'est-à-dire qu'il applique la même combinaison de technique sur le bloc de texte en clair 16 fois.

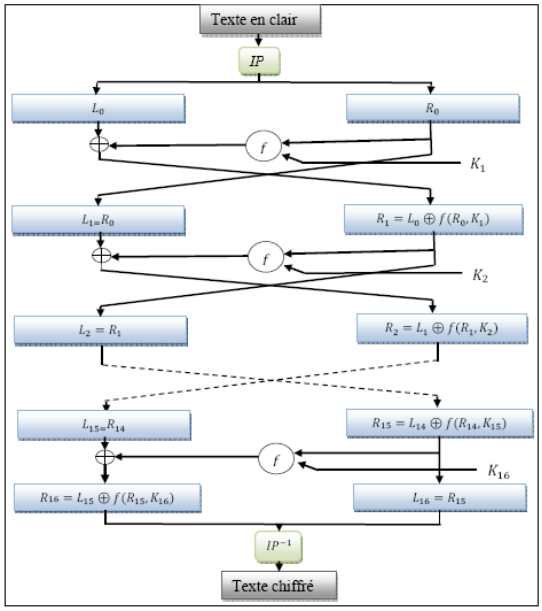


Figure A- 1: Plan générale de l'algorithme DES

A.3 LE TRIPLE DES

Le Triple DES est un algorithme de chiffrement symétrique enchaînant 3 applications successives de l'algorithme DES [18]sur le même bloc de données de 64 bits, avec 2 ou 3 clés différente.

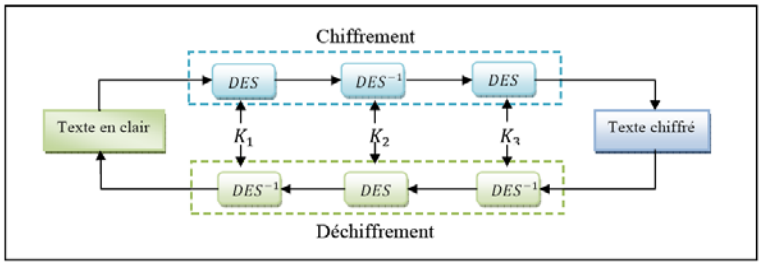


Figure A- 2: Plan générale de l'algorithme DES

Le Triple DES[18] est généralement utilisé avec seulement deux clés différentes. Le mode d'usage standard est de l'utiliser en mode EDE (Encryption, Decryption, Encryption) ce qui le rend compatible avec DES quand on utilise trois fois la même clé.

A.4 Théorème de Shannon

La méthode la plus utilisée pour engendrer un signal numérique est l'échantillonnage d'un signal analogique $x_a(t)$ .En général, les échantillons sont prélevés périodiquement avec une période t'appelée période d'échantillonnage.

La théorie :

La perte d'information entre le signal continu et le signal discret correspondant est nulle si et seulement si la fréquence d'échantillonnage Fe est au moins supérieure ou égale au

double de la plus haute fm contenue dans ce signal :fm $\leq$ Fe/2.la figure ci_dessous illustre ce théorème.

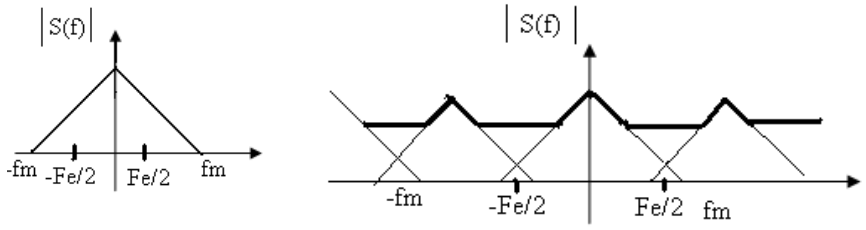


Figure A- 3: Repliement spectral

Il est facile de remarquer qu'une distorsion du spectre d'amplitude apparaît dés que fmax>Fe/2 dans la bande [-Fe/2,Fe/2].Ce phénomène porte le nom de « repliement spectrale » autour de la fréquence Fe. Ce théorème reste valable dans le cas des signaux aléatoires.

A.5 Démonstration de quantification

$$\delta = 2 \cdot X_s \cdot 2^{-b}$$

$$\sigma_e^2 = \frac{\delta^2}{12} = \frac{4 \cdot X_s^2}{12 \cdot 2^{2b}} = \frac{X_s^2}{3 \cdot 2^{2b}}$$

$$RSB = 10 \cdot \log(\sigma_x^2 / \sigma_e^2) = 10 \cdot \log(\sigma_x^2 \cdot 3 \cdot 2^{2b} / X_s^2)$$

$$RSB = 10 \cdot \log(\sigma_x^2 \cdot 3 \cdot 2^{2b}) - 10 \cdot \log(X_s^2)$$

$$RSB = 10 \cdot (\log(3 \cdot 2^{2b}) + \log(\sigma_x^2) - \log(X_s^2))$$

$$RSB = 10 \cdot (\log(3) + b \cdot \log(4) - 2 \cdot \log(\frac{\sigma_s}{\sigma_x}))$$

$$\Gamma = \frac{\sigma_s}{\sigma_x}$$

$$RSB = 4.77 + 6.025 \cdot b - 20 \cdot \log(\Gamma)$$

Bibliographie

[1] AIT-HADDA.Mounira, IHADADENE.Dalila, « Synthèse Paramétrique des signaux de parole » USTHB.2001. Dirigé par Mr.A.HOUACINE.

[2] BENAKMOUNE Yacine, « Optimisation des paramètres de codage de la M.LPC » USTHB.Juillet 1994. Dirigé par Mr.B.BOUDRAA.

[3] BRUNO Martin, « Codage, Cryptographie et Application ». Presses Polytechniques Universitaires Romandes.

[4] Federal Information Processing Standards Publication 197, «Announcing the ADVANCED ENCRYPTION STANDARD (AES) ». November 26, 2001.

[5] GRASSI Sara, «Optimized Implementation of Speech Processing Algorithms», Thèse soumise à la faculté des sciences de l'université de neuchâtel pour l'obtention du grade de docteur ès sciences

[6] G.Fant, «Acoustic Theory of speech production », Mounton and Co, Gravenhage, The Netherlands.1960.

[7] Joan Daemen and Vincent Rijmen, AES submission document on Rijndael, June 1998.

[8] Joan Daemen and Vincent Rijmen, the Design of Rijndael, AES The Advanced Encryption Standard, Springer-Verlag 2002 (238 pp.).

[9] R.Viswanthan and J.Makhoul, «Quantization proprities of transmission paramteres in Linear predictive systems». IEEE Trans Acous.Speech and Signal Process.Vol ASSP-23.NO 3(June 1975).

[10] René Boite et Murat Kunt, «Traitement de la parole», Presses Polytechniques Romandes.

[11] SHNEIER Bruce, « Cryptographie Appliquée », Algorithmes, Protocoles, Codes Sources.Traduction par Marc Vauclair.

[12] Federal Information Processing Standards Publication 197, «Announcing the ADVANCED ENCRYPTION STANDARD (AES) ». November 26, 2001.

[13] Joan DAEMEN and VICNCENT RIJMEN, AES submission document on Rijndael, June 1998.

[14] B.REKIOUA et Y.GUEMANA
« Etude et implémentation d'un algorithme de cryptage de la parole sur une plateforme DSP de type TMS 320C6713 » EMP 2008.
Encadré par S.HALILOU

[16] M.ELKERIA et S.HADJ-KACI
« *Conception d'un binomieur en ligne pour l'évaluation des séries entières* » USTHB 2000. .

[18] SHNEIER Bruce, « Cryptographie Appliquée », Algorithmes, Protocoles, Codes Sources.Traduction par Marc Vauclair. International Thomson publiching France 1995.