



HAL
open science

Conditionnement et traitement du Signal Numérique

Christophe Krzeminski

► **To cite this version:**

Christophe Krzeminski. Conditionnement et traitement du Signal Numérique. École d'ingénieur. Lille, France. 2020, pp.115. hal-02988827v1

HAL Id: hal-02988827

<https://hal.science/hal-02988827v1>

Submitted on 4 Nov 2020 (v1), last revised 4 Sep 2023 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - NonCommercial 4.0 International License

*Notes de cours en Conditionnement et Traitement du
Signal Numérique (M1)*

Année scolaire 2020-2021

rédacteur: Christophe Krzeminski

HEI · ISEN · ISA

Siège social : 2, rue Norbert-Ségaré BP 41 290 - 59 014 Lille cedex

Tél. : +33 (0)3 28 38 48 58 / **Web** : junia.com

SIREN : 783 707 003 / **NAF** : 85.42Z

DRAFT

Contents

I	Traitement du signal numérique	5
I.1	Introduction	5
I.2	Définition chaîne de traitement numérique du signal	6
I.3	Exemples de signaux numériques simples	8
I.3.1	Définition de signaux numériques simples	8
a)		8
I.3.2	Représentation de signaux numériques simples	9
I.3.3	Propriétés et classification des signaux numériques	9
a)	Notion d'énergie	9
I.3.4	Notion de périodicité des signaux numériques	10
I.4	Convertisseur domaine analogique/numérique (CAN)	10
I.4.1	Echantillonnage de signaux numériques	12
a)	Notion d'échantillon	12
b)	Echantillonnage idéal mathématique	12
c)	Notion d'échantillonnage naturel	14
d)	Notion d'échantillonnage-blocage (E/B)	14
e)	Définition du temps d'acquisition	15
I.4.2	Echantillonnage et spectre fréquentiel	16
a)	Périodisation du spectre du signal échantillonné	16
b)	Théorème de Shannon	17
c)	Fréquence de Nyquist	17
d)	Limites du théorème de Shannon	17
e)	Estimation performance globale d'un CAN	18
I.4.3	Exercice	19
a)	Systèmes à multiple cadence	19
I.4.4	Quantification de signaux numériques	21
a)	Principe de quantification	21
b)	Caractéristiques de quantification	22
c)	Erreur de quantification	22
I.4.5	Exercice	24
a)	Rapport signal à bruit	24
b)	Rapport signal à bruit d'un signal sinusoïdal	25
c)	Quantification non-uniforme	26
I.4.6	Exemple de quantification non-uniforme	27
I.4.7	Encodage ou représentation binaire	27
I.4.8	Notion de reconstruction	28
a)	Interpolation idéale de Shannon	28
b)	Reconstruction en pratique	29
c)	Caractéristique de transfert d'un CNA	29

I.5	Chaîne complète de traitement du signal	30
I.6	Analyse d'une chaîne complète CAN-CNA	32
I.6.1	Influence d'un filtre anti-recouvrement	33
I.6.2	Influence d'un filtre de reconstruction, lissage	33
I.7	Conclusion	34
II	Analyse Spectrale Signaux Numériques	37
II.1	Introduction	37
II.1.1	Aspect historique	37
II.1.2	Applications	38
II.2	Transformée de Fourier (TF)	38
II.2.1	Définition	38
II.2.2	Vers la transformée discrète	39
II.3	Transformée de Fourier Discrète (TFD)	40
II.3.1	Définition	40
II.3.2	Expression matricielle de la transformée de Fourier discrète	40
II.4	Transformée de Fourier Rapide (TFR)	41
II.4.1	Exercice	44
II.4.2	Exemple de calcul simple sur Octave&Matlab	44
III	Systèmes numériques et Transformée en z	49
III.1	Introduction	49
III.1.1	Aspect historique	49
III.1.2	Introduction	49
III.2	Description des systèmes numériques	50
III.2.1	Définition	50
III.2.2	Exemple de systèmes numériques simples	50
III.2.3	Exercice	50
III.2.4	Critère de stabilité	50
III.3	Diagramme fonctionnel	51
III.4	Propriétés des systèmes numériques	52
III.4.1	Définition des propriétés de systèmes numériques	52
a)	Statique	52
b)	Dynamique	52
c)	Linéaire	52
d)	Stabilité	52
e)	Superposition	52
f)	Invariant	53
g)	Causalité	53
III.4.2	Interconnection des systèmes numériques	54
III.4.3	Notion de système LTI	54
III.5	Définition de la transformée en z (TZ)	54
III.5.1	Définition	54
III.5.2	Table de transformée en z	55
III.5.3	Lien entre DFT et transformée en z	56
III.5.4	Résumé des propriétés de la transformée en z	56
III.5.5	Concept de délai	57
III.5.6	Transformée en z rationnelle	57
III.5.7	Fonction de transfert d'un système numérique	57

III.6	Fonctions Octave&Matlab utiles	58
IV	Introduction aux filtres numériques	61
IV.1	Introduction	61
IV.2	Principes d'un filtre numérique	61
IV.2.1	Pourquoi filtrer ?	61
IV.2.2	Définition	62
IV.2.3	Filtre ARMA	62
IV.2.4	Fonction de transfert	63
IV.3	Catégorie de filtres numériques	63
IV.4	Spécification des filtres numériques	64
IV.4.1	Gabarit d'un filtre passe-bas	64
IV.4.2	Gabarit du filtre	64
IV.4.3	Réponse en fréquence d'un filtre	65
IV.4.4	Caractérisation simple d'un filtre numérique	66
	a) Réponse impulsionnelle	66
	b) Réponse fréquentielle	66
IV.5	Exercice	67
V	Filtre à réponse impulsionnelle finie (RIF)	71
V.1	Introduction	71
V.1.1	Aspect historique	71
V.2	Définition des filtres numériques	71
V.3	Propriétés des filtres RIF	72
V.3.1	Phase linéaire et réponse impulsionnelle	72
V.3.2	Catégories de filtre RIF	72
V.4	Troncature de la réponse impulsionnelle	72
V.4.1	Méthode de conception globale	72
V.4.2	Méthode de la troncature de la réponse impulsionnelle	73
	a) Modèle de filtre à phase linéaire	73
V.4.3	Fenêtrage de réponse	74
V.4.4	Influence de fenêtre rectangulaire	75
V.4.5	Choix de la fenêtre de conception	75
V.4.6	Réponse impulsionnelle idéale des différents filtres	75
V.4.7	Méthode de conception de filtres basés sur Octave&Matlab	76
V.4.8	Exemple implémentation filtre RIF passe-bas	77
V.4.9	Exercice: Séparation des basses et hautes fréquences	77
VI	Filtre à réponse impulsionnelle infinie (RII)	81
VI.1	Introduction	81
VI.1.1	Aspect historique	81
VI.2	Transformée bilinéaire	81
VI.2.1	De l'analogique au numérique ?	81
VI.2.2	La transformée bilinéaire	81
VI.2.3	Exemple	82
VI.3	Gabarit du filtre analogique équivalent	83
VI.4	Conception d'un filtre RII	84
VI.5	Modèle de filtre analogique	84
VI.5.1	Filtre de Butterworth	84

VI.5.2	Filtre de Chebychev	85
VI.5.3	Filtre elliptique dit de Cauer	85
VI.5.4	Exemple	85
VI.6	Conception de filtre RII à l'aide de Matlab	86
VI.6.1	Fonctions	86
VI.6.2	Exemple de conception de filtre	86
VI.7	Implémentation pratique d'un filtre FIR ou RII	87
VI.7.1	Implémentation filtre RIF	87
VI.7.2	Implémentation filtre RII	87
	a) Implémentation forme directe	87
	b) Implémentation sous la forme canonique	88
VI.7.3	Implémentation bloc en série	88
	a) Décomposition sous la forme de bloc du second degré	88
	b) Méthode de réalisation	88
	c) Comparaison filtre RIF vs RII	89
	d) Confusion méthode de conception RIF vs RII	90
VI.8	Exercice	90
VII		95

Nomenclature

AAF *Filtre Anti-Aliasing*

ARMA *Structure de filtre AutoRegressif à Moyenne Adaptée*

CAN *Convertisseur Analogique/Numérique*

CNA *Convertisseur Numérique/Analogique*

DCT *Transformée en Cosinus Discret*

DHT *Transformée de Hartley*

LSB *Bit de poids faible*

LTI *Linéaire et invariant dans le temps*

MSB *Bit de poids fort*

MDCT *Transformée en Cosinus Discret Modifié*

RIF *Filtre à réponse impulsionnelle finie*

RII *Filtre à réponse impulsionnelle infinie*

ROC *Région de convergence (cercle unité)*

RF *Filtre de reconstruction*

SNR *Rapport signal à bruit*

TF *Transformée de Fourier*

TNS *Processeur de traitement du signal numérique*

TFD *Transformée de Fourier discrète*

TFR *Transformée de Fourier rapide*

TFR *Transformée de Fourier rapide*

TZ *Transformée en Z*

Acronyme en anglais

AAF	<i>Anti-aliasing Filter</i>
ARMA	<i>Autoregressive-moving-average Models</i>
A/D	<i>Analog-Digital Converter</i>
D/A	<i>Digital-Analog Converter</i>
DCT	<i>Discrete Cosine Transform</i>
DFT	<i>Discrete Fourier Transform</i>
DSP	<i>Digital Signal Processing</i>
FIR	<i>Finite Impulse Response system</i>
FT	<i>Fourier Transform</i>
FFT	<i>Fast Fourier Transform</i>
HT	<i>Hartley Transform</i>
IIR	<i>Infinite Impulse Response system</i>
LSB	<i>Least Significant Bits</i>
LTI	<i>Linear Time Invariant</i>
MSB	<i>Most Significant Bit</i>
MDCT	<i>Modified Discrete Fourier Transform</i>
RF	<i>Recovery Filter</i>
ROC	<i>Region of Convergence</i>
SNR	<i>Signal-to-Noise Ratio</i>
STFT	<i>Short Term Fourier Transform</i>

Ce document est un première ébauche destinée à être une synthèse de l'ensemble des notions et connaissances exposées lors du cours de *Conditionnement et traitement du signal numérique*. réalisé au sein de ISEN-Lille.

Malgré de nombreuses années d'enseignement, ce document peut comporter des erreurs, des imprécisions indépendant de ma volonté. N'hésitez pas à m'en faire part...

Son écriture permet qu'une présence physique des étudiants en cours ne soit plus obligatoire. En effet, le fait que l'essentiel des notions, connaissances, exercices soit résumé dans ce document fait qu'il peut être utilisé d'une manière avantageuse dans le contexte d'incertitude sanitaire actuel lié au virus sars-cov2 et indépendant de toute plateforme technique numérique. Sans rentrer dans un débat très complexe, l'enseignement du numérique ne passe par forcément par que le numérique...

Par contre, probablement le fait d'étudier de manière autonome mais cadré exige un travail, une rigueur et une concentration soutenue peut-être plus importante qu'en enseignement interactif. Le programme de ce cours et le cadre correspond à la lecture et les exercices liés à un chapitre lui-même faisant la synthèse d'une semaine de cours en présentiel.

Malheureusement, ce document ne remplace pas les interactions, les interrogations explications orales que vous auriez eu dans une séance de cours en présentiel mais rien ne vous empêche d'essayer de me les transmettre et moi d'y répondre.

En vous souhaitant bon courage et une agréable période d'apprentissage par la lecture, la réflexion, le raisonnement et peut-être l'écriture !

UneDes ruptures fondamentales en terme de technologies est l'invention du transistor en 1956. En effet, cette invention permettra le développement fulgurant de circuits intégrés permettant de réaliser des calculs, des fonctions et enfin des systèmes qui avaient été envisagé mais demeurait inaccessible avec les technologies disponible. La technologie numérique marque d'une certaine manière l'avènement de tout un environnement qui se développera d'abord de manière indépendante des autres technologies principales (électricité, mécanique, chimie, agronomie, biologie) avant de coexister à l'interface voire de chercher à en prendre une part non-négligeable de certaines d'entre-elles principalement pour des enjeux économiques ¹.

En français, le numérique se définit par une manière de représenter et d'implémenter l'information sous la forme de nombres. Ces nombres peuvent (ou pas) représenter ou être lié à une grandeur ou un phénomène physique. D'une certaine manière, le numérique propose une manière différente à la fois de définir l'information en utilisant des nombres mais aussi en terme de traitement de l'information celui-ci étant réalisé par des calculs. Il diffère du domaine de l'analogique où l'information est elle représentée par une grandeur, une variable physique. Le processus de numérisation définit la migration du domaine de traitement de l'information du domaine de l'analogique au numérique. Par contre, le monde anglosaxon utilise plutôt le terme *digital* avec une conotation plutôt péjorative pour définir le numérique.

L'objectif de ce support de cours est de décrire la chaîne de traitement de signal numérique ainsi que les problématiques liées aux signaux numériques. Il peut s'agir de caractériser ces propriétés temporelles ou spectrales. La notion de système numérique qui réalise un traitement sur une entrée numérique afin d'obtenir un signal de sortie numérique correspondant à l'application envisagée.

Les systèmes numériques permettent d'implémenter les fonctions suivantes :

- Analyse et le traitement de signaux.
- Système numérique correcteur et systèmes automatiques.
- Système numérique de communication à haut débit.

Le traitement du signal numérique est une discipline scientifique en elle-même dans le sens où elle s'intéresse à un vaste nombre de processus sur un signal comme sa capture, sa reconnaissance, sa détection, son codage, son analyse, son estimation et éventuellement son filtrage. L'ensemble de ces traitements nécessite une chaîne numérique composée de différents blocs ayant chacun des fonctions spécifiques (convertisseur, calculateur, interpolateur, filtre). A l'heure de l'ère "tout-numérique" et de l'instantanéité, les paramètres *physiques* régissant ces systèmes numériques (temps d'acquisition de conversion, la résolution, la compression du signal, l'analyse spectrale, la fréquence d'échantillonnage, la dynamique en amplitude, la plage de conversion, les non-linéarité, la puissance consommée, ...) tendent probablement à être oubliés. Au niveau de la chaîne numérique, les notions exposées dans

¹Le risque probablement actuel est de ne considérer que l'interface numérique en oubliant *in-fine* l'ensemble du socle de connaissances et du savoir-faire sous-jacent.

ce cours concernent les fonctions d'échantillonnage, d'analyse, de blocage, de reconstruction tendent à être confondues. Les contraintes principalement de temps réel font que les analyses spectrales s'orientent probablement vers des signaux non-stationnaires à l'aide de transformée Fourier dépendante du temps, d'ondelette même si la transformée en cosinus discret demeure le standard au niveau du traitement d'image ou du codage audio. Le filtrage demeure une application certes sommaire mais parfois indispensable des chaînes de traitement du signal ou d'information. Enfin, des aspects plus pratique concernant les implémentations de convertisseur analogique/numérique ou de filtres audio sur des processeurs de traitement du signal sont abordés.

DRAFT

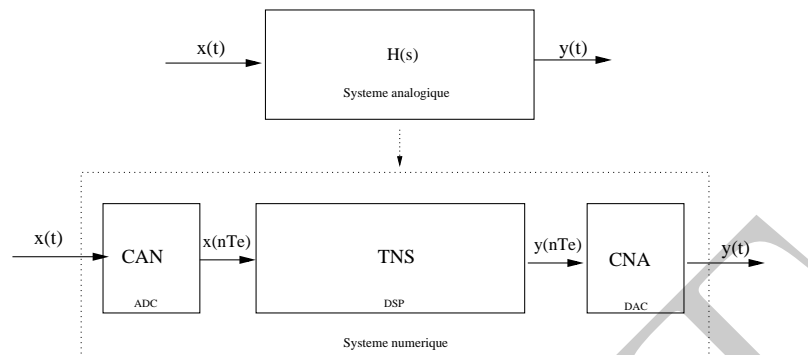


Figure .1: L'opération de numérisation consiste à remplacer un système analogique continu défini par une fonction de transfert $H(s)$ par une chaîne de traitement numérique du signal.

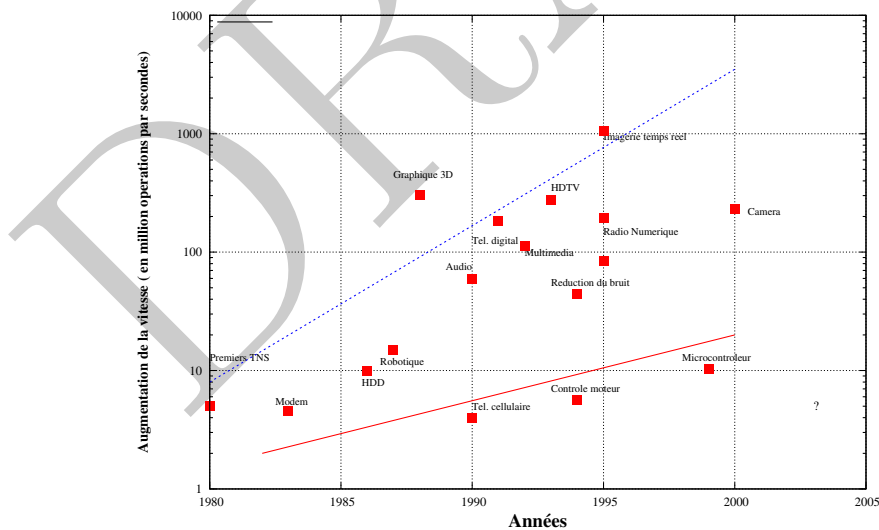


Figure .2: Panorama des applications en traitement du signal en fonction de la vitesse de calcul ².

I – Traitement du Signal Numérique

I.1 Introduction



Harry Nyquist (1889-1976)



Claude Elwood Shannon (1916-2001)

Les notions exposées dans ce cours sont le fruit des travaux et des découvertes de Harry Nyquist (1889-1976) et de Claude Elwood Shannon (1916-2001). Harry Nyquist était un ingénieur au laboratoire Bell aux Etats-Unis. Il a également contribué scientifiquement à la modélisation du bruit thermique, de la régulation et des communications. Claude Shannon était un mathématicien américain. Il est considéré comme le fondateur de la théorie du traitement du signal numérique.

1.2 Définition chaîne de traitement numérique du signal

Une chaîne de traitement du signal numérique simplifiée se compose en pratique de trois blocs principaux implémentant des fonctions différentes :

- en entrée, la chaîne comporte un convertisseur analogique/numérique (CAN). Sa fonction est de transformer le signal analogique $x(t)$ en un signal numérique $x(n)$. Il permet de faire le lien entre le domaine analogique et celui numérique.
- le cœur de la chaîne de traitement du signal numérique est un bloc de calcul, un calculateur appelé (TNS). Celui a pour fonction d'implémenter et de réaliser des calculs uniquement sur des signaux numériques. La sortie du calculateur est un signal numérique $y(n)$.
- en sortie se trouve un autre convertisseur numérique/analogique (CNA) cette fois. Sa fonction est de réaliser l'opération inverse à savoir transformer le signal numérique $y(n)$ en sortie du calculateur en un signal analogique $y(t)$.

Deux paramètres principaux caractérisent principalement la chaîne de traitement numérique du signal à savoir:

- La période ou l'intervalle d'échantillonnage T_e qui permet de définir la durée entre deux échantillons $x(n)$.
- Le pas de quantification q correspond à l'écart entre deux valeurs d'amplitude du signal quantifié.

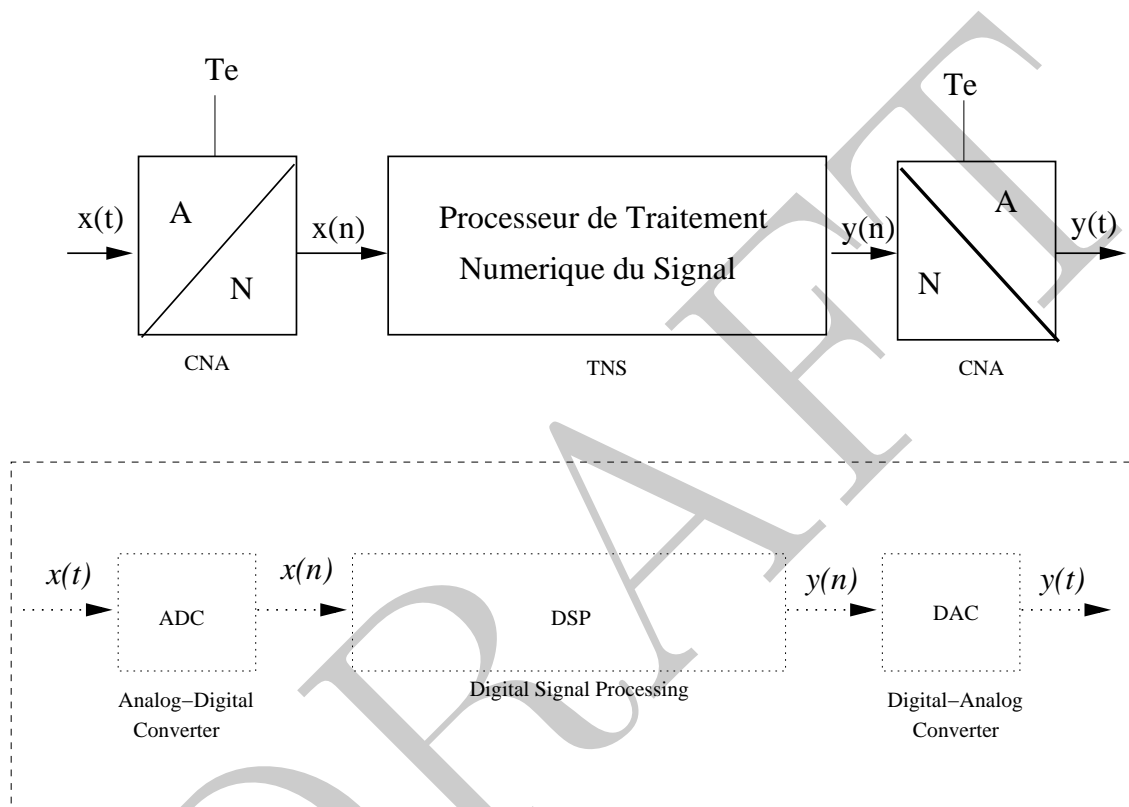


Figure I.1: Chaîne de traitement numérique du signal composé d'un convertisseur analogique/numérique (CAN), d'un processeur de traitement numérique du signal (TNS) et d'un convertisseur numérique/analogique (CNA). La vision anglosaxonne est un peu différente et celle-ci est basée sur le terme *digital* et décrite ci-après.

1.3 Exemples de signaux numériques simples

1.3.1 Définition de signaux numériques simples

1.3.1

Les signaux numériques se distinguent des signaux analogiques par le fait qu'ils ne sont pas continus mais discrets. Souvent en guise d'illustration, c'est le domaine temporel qui est considéré. En pratique, ces signaux ne prennent que des valeurs d'amplitude discrète et à des instants précis.

- Le signal numérique le plus simple est l'échantillon unité. Celui-ci a une valeur d'amplitude unité à un instant donné n . Il est défini d'un point de vue mathématique par l'équation suivante:

$$\delta[n] = \begin{cases} 0, & \neq 0 \\ 1, & = 0 \end{cases} \quad (\text{I.1})$$

- Celui-ci permet de définir un signal numérique correspond à l'échelon numérique. Ce dernier prend des valeurs d'amplitude unité à des instants donné espacé d'une période T_e .

$$u[n] = \begin{cases} 1, & \geq 0 \\ 0, & < 0 \end{cases} \quad (\text{I.2})$$

- Il est possible de définir un signal proche caractérisé par une amplitude décroissante dans le temps. Celui-ci est similaire à une suite géométrique, l'amplitude prenant des valeurs à des instants précis espacé d'une période T_e .

$$x[n] = a^n u(n) \quad (\text{I.3})$$

- Les formes correspondant aux signaux cosinus et sinus sont également définies dans le domaine numérique. Ces signaux prennent des valeurs discrètes caractérisées par des amplitudes variables :

$$x[n] = \sin(n\omega_0 + \phi) \quad \omega_0 = 2\pi f_0 T_e \quad (\text{I.4})$$

où $\omega_0 = 2\pi f_0 T_e$ caractérise une pulsation numérique et ϕ la phase du signal.

- Enfin, la séquence de phase permet d'avoir un signal complexe numérique avec une amplitude constante à des instants précis espacé d'une période T_e , la phase de la séquence est par contre linéaire.

$$x[n] = (a + jb)^n u(n) \quad (\text{I.5})$$

Ces signaux numériques permettent de définir un ensemble simple de signaux numériques modèles.

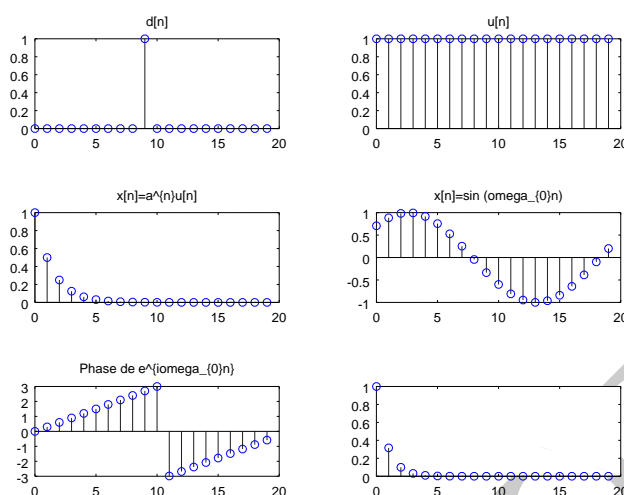


Figure I.2: Exemple de modèles de signaux numériques simples: a)échantillon unité b)échelon unité c)exponentielle numérique d)sinus numérique e) séquence de phase complexe numérique f) séquence complexe numérique

I.3.2 Représentation de signaux numériques simples

La figure suivante I.2 résume graphiquement cette fois la forme de ces différents signaux modèles. En pratique, les signaux numériques obtenus par l'échantillonnage présentent des formes beaucoup plus complexe en lien avec les signaux analogiques ou les variables physiques étudiées.

I.3.3 Propriétés et classification des signaux numériques

Les signaux numériques peuvent être caractérisés par différentes propriétés en particulier en terme d'amplitude des signaux:

I.3.3 Notion d'énergie

- Une séquence numérique est dite *bornée* si:

$$|x[n]| \leq Max \leq +\infty \quad (I.6)$$

l'amplitude maximale des échantillons.

- Une séquence numérique est dite de *carré sommable* si:

$$\sum_{n=-\infty}^{+\infty} |x(n)|^2 \leq +\infty \quad (I.7)$$

la somme du module des échantillons au carré nest pas infinie.

Si cette propriété est satisfaite, il est possible d'aboutir à la définition de l'énergie E d'un signal numérique $x(n)$ à savoir:

- Energie d'un signal numérique:

$$E = \sum_{n=-\infty}^{n=+\infty} |x(n)|^2 \quad (\text{I.8})$$

I.3.4 Notion de périodicité des signaux numériques

Pour des signaux discrets, la notion de périodicité d'un signal est définie par le fait que pour un nombre d'échantillons définissant la période, donné N l'amplitude du signal sera identique à celle de la période précédente. Ceci se traduit, en terme mathématique, par l'équation toute simple suivante:

$$x(n) = x(n + N) \quad (\text{I.9})$$

En pratique, le passage du domaine analogique au domaine numérique fait que la propriété d'un signal n'est pas forcément conservée. Par exemple, pour un signal sinusoïdal, cette condition se traduit par le fait que l'échantillon n soit égal à celui d'une période :

$$x(n) = \sin(n\omega_0 + \phi) = \sin(n\omega_0 + N\omega_0 + \phi) \quad (\text{I.10})$$

ceci implique par exemple:

$$N\omega_0 = 2k\pi N = \frac{2k\pi}{\omega_0} \quad (\text{I.11})$$

Le fait que N soit nécessairement entier n'est pas évident si l'on fie à l'équation précédente et implique que la pulsation numérique ω_0 doit respecter certaines conditions.

I.4 Convertisseur domaine analogique/numérique (CAN)

Le bloc d'entrée implémentant la première fonction d'une chaîne de traitement numérique du signal est un convertisseur du domaine analogique vers le domaine numérique. En pratique, pour réaliser la conversion d'un signal analogique $x(t)$ en numérique $x(n)$, il est nécessaire de réaliser trois opérations différentes à savoir:

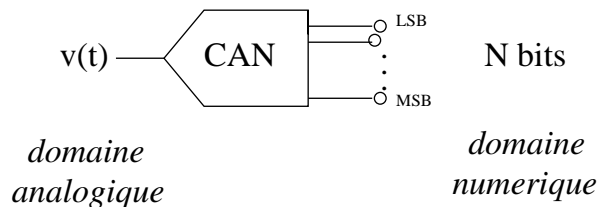


Figure I.3: Principe de fonctionnement d'un composant convertisseur analogique/numérique (CAN) qui consiste à transformer une tension analogique $v(t)$ en une séquence binaire.

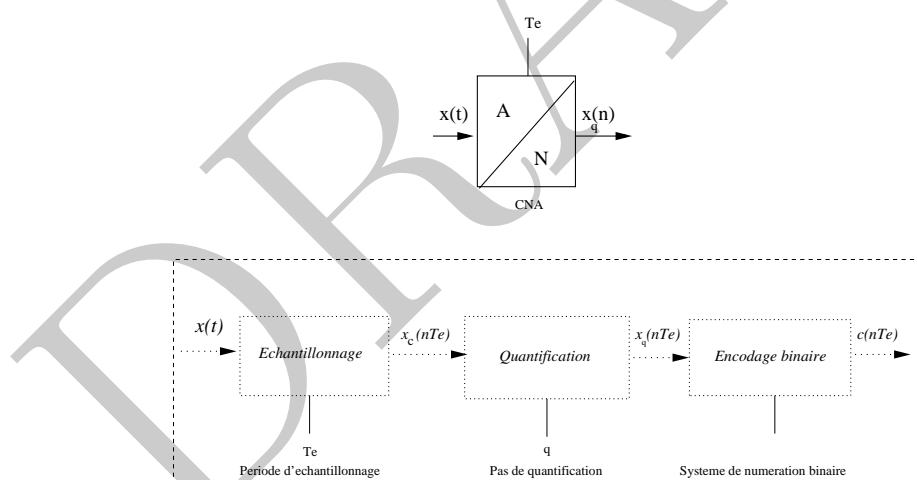


Figure I.4: Les différentes fonctions d'un convertisseur analogique/numérique. La première correspond à l'échantillonnage du signal d'entrée $x(t)$ réalisé selon la période T_e . La seconde est l'opération de quantification qui consiste à discrétiser l'amplitude continue des signaux. Enfin, la dernière consiste à encoder les valeurs quantifiées sous la forme de bits selon une règle spécifiée entre les valeurs et les mots binaires.

- *Echantillonnage*: celui-ci est défini par l'opération consistant à prendre des échantillons discrets, en pratique, sur le domaine étudié, c'est à dire tous les T_e au niveau temporel.
- *Quantification*: le signal analogique d'entrée prend un ensemble de valeurs d'amplitude continu, il est nécessaire de le transformer en un signal comportant des valeurs d'amplitude discrètes défini par un ensemble régi de valeurs quantifiées par un pas de quantification q .
- *Codage*: la dernière opération implémentée par le convertisseur est de réaliser l'encodage des valeurs sur des valeurs de bits précises. Ces valeurs sont définies par l'intermédiaire d'un codage permettant d'implémenter la correspondance entre les valeurs quantifiées et les mots binaires.

1.4.1 Echantillonnage de signaux numériques

1.4.1 Notion d'échantillon

L'échantillonnage des signaux consiste à prendre des valeurs $x(n)$ du signal $x(t)$ selon quelques "instants" discrets sur un domaine continu de valeurs. D'un point de vue représentation graphique, cette opération est ainsi résumée par la figure suivante:

Très souvent, par simplicité, la notion d'échantillon est reliée à un domaine temporel où il n'est pas du tout simple de considérer son aspect continu. Elle est toutefois plus vaste que dans le sens où il est nécessaire d'échantillonner pour un capteur physique le domaine spatial (la surface, le volume) voire le domaine spectral (notion de couleur). Le pas d'échantillonnage permettra ainsi de d'introduire la notion de pixel par exemple pour une image ou un volume élémentaire pour définir un espace.

1.4.1 Echantillonnage idéal mathématique

L'échantillonnage idéal peut s'exprimer aussi d'un point de vue mathématique de la manière suivante:

$$x_{ech}(t) = x(t) \cdot \sum_{n=-\infty}^{+\infty} \delta(t - nT_e) = \sum_{n=-\infty}^{n=+\infty} x(nT_e) \cdot \delta(t - nT_e) \quad (\text{I.12})$$

Il s'agit donc de multiplier le signal analogique $x(t)$ par un train de dirac $\sum_{n=-\infty}^{n=+\infty} \delta(t - nT_e)$ de période T_e . En pratique, un signal idéal proche de la vision mathématicienne à savoir un signal prenant une valeur d'amplitude unité sur un instant infiniment court et de manière périodique n'est pas réalisable. Par ailleurs, la mesure de la valeur physique d'un signal ne peut pas être immédiat mais requiert un intervalle défini.

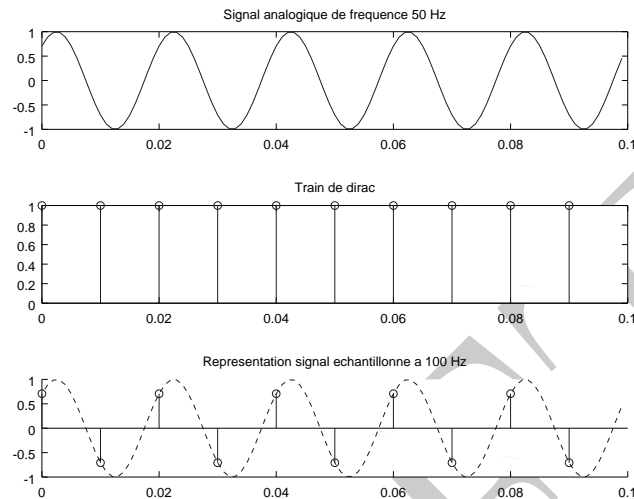


Figure I.5: La figure présente l'échantillonnage correspondant à un signal continu sinus électrique de fréquence 50Hz. La fréquence maximale définie par le théorème de Shannon est respectée, l'échantillonnage discret va se faire à une fréquence deux fois supérieure représentée par un train d'impulsion ou de dirac définissant l'intervalle d'échantillonnage. Le signal échantillonné ne comportera que quelques valeurs discrètes en fonction du pas d'échantillonnage à la différence de son homologue physique.

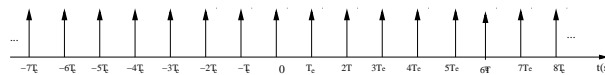


Figure I.6: Echantillonnage idéal d'un signal analogique par un train de Dirac. Les instants discrets d'échantillonnage sont définis par les impulsions du train ou du peigne de Dirac.

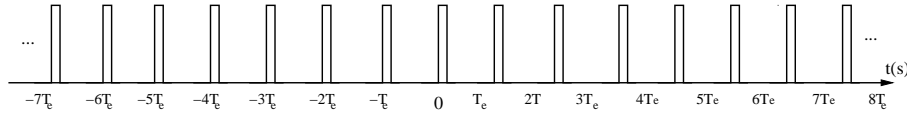


Figure I.7: Echantillonnage naturel d'un signal analogique par un train de porte. Les intervalles d'échantillonnage du signal analogique $x(t)$ sont définis par la durée de la porte T_0 c'est à dire entre $[-\frac{T_0}{2}; \frac{T_0}{2}]$, il peut être remarqué que la périodicité est T_e .

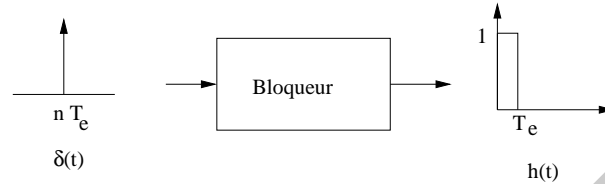


Figure I.8: Principe du bloqueur: de manière à pouvoir quantifier l'amplitude d'un échantillon, l'amplitude est maintenue constante durant la période d'échantillonnage.

I.4.1 Notion d'échantillonnage naturel

L'échantillonnage naturel a été introduit de manière à donner une définition plus réaliste permettant une implémentation pratique. L'échantillonnage naturel est réalisé par un signal rectangulaire périodique de largeur T_0 réduit par rapport à l'intervalle et le pas d'échantillonnage T_e .

$$\sum_{n=-\infty}^{n=+\infty} \Pi_{[-\frac{T_0}{2}; \frac{T_0}{2}]}(t - nT_e) \quad (\text{I.13})$$

Le signal échantillonné est donné par l'expression mathématique suivante:

$$x_{ech}(t) = x(t) \cdot \sum_{n=-\infty}^{n=+\infty} \Pi_{[-\frac{T_0}{2}; \frac{T_0}{2}]}(t - nT_e) = \sum_{n=-\infty}^{n=+\infty} x(nT_e) \cdot \Pi_{[-\frac{T_0}{2}; \frac{T_0}{2}]}(t - nT_e) \quad (\text{I.14})$$

Comme on peut le voir sur la figure, l'échantillonnage naturel fait que le signal analogique est multiplié par un *train de porte*. Cela se traduit au niveau du signal échantillonné qu'il va suivre l'amplitude du signal analogique durant la durée de l'impulsion T_0 .

I.4.1 Notion d'échantillonnage-blocage (E/B)

De manière, à pouvoir réaliser l'opération d'échantillonnage, l'amplitude du signal analogique à convertir varie souvent rapidement et doit demeurer constant durant le processus de numérisation c'est-à-dire durant la période d'échantillonnage T_e :

La fonction bloqueur (E/B) peut être modélisée dans le domaine temporel par un système de

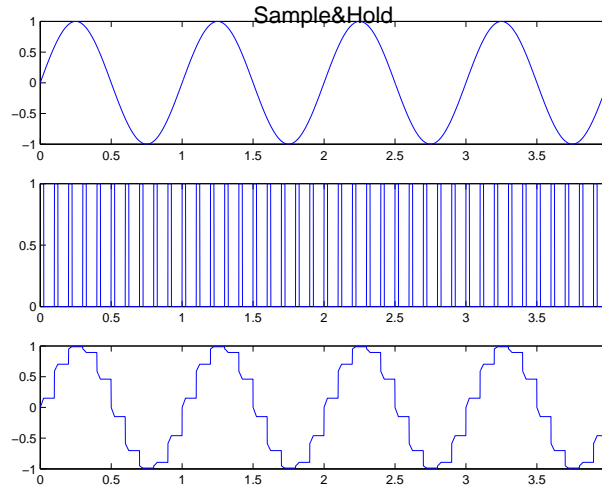


Figure I.9: Exemple d'échantillonnage blocage d'un signal sinusoidal. Le signal sinusoidal est représenté, des échantillons d'une durée courte sont prélevés et enfin l'amplitude est bloquée durant la période entière d'échantillonnage.

réponse impulsionnelle de réponse rectangulaire de durée T_e :

$$x_{E/B}(t) = x(t) * h(t) \quad (\text{I.15})$$

dont la réponse fréquentielle $H(\nu)$ est donc donnée par l'équation suivante:

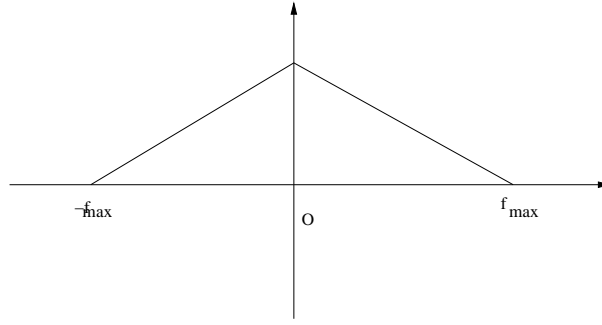
$$H(\nu) = TF(h(t)) = T_e \exp^{-i\pi\nu T_e} \text{sinc}\left(\frac{\nu}{\nu_e}\right) \quad (\text{I.16})$$

Il s'agit principalement de l'opération de préconditionnement du signal analogique avant l'étape de quantification de son amplitude. Un exemple d'échantillonnage bloqueur d'un signal sinusoidal est donné au niveau de la figure suivante:

En pratique, cette opération est implémentée par un circuit basé sur un principe simple à savoir un interrupteur et une capacité C qui va permettre de définir le temps d'acquisition du convertisseur.

I.4.1 Définition du temps d'acquisition

Pour un convertisseur analogique/numérique (CAN), le temps d'acquisition correspond la durée nécessaire afin de capturer la chaîne d'échantillonnage de capturer l'amplitude du signal d'entrée. Celui-ci est défini au niveau physique par la charge de la capacité de l'échantillonneur bloqueur.

Figure I.10: Spectre fréquentiel modèle du signal $x(t)$.

I.4.2 Echantillonnage et spectre fréquentiel

L'opération d'échantillonnage d'un signal a une influence non négligeable sur son spectre. Différents aspects sont abordés ici à savoir sa périodisation lors de l'échantillonnage, l'introduction du théorème de Shannon et le choix d'une fréquence d'échantillonnage adaptée. Certains signaux à spectre non borné peuvent nécessiter l'introduction d'un filtre anti-repliement en entrée de la chaîne de traitement du signal.

I.4.2 Périodisation du spectre du signal échantillonné

Le fait de venir échantillonner un signal analogique n'est pas sans conséquence au niveau de son spectre fréquentiel. En effet, mathématiquement, une des contraintes de la chaîne de traitement numérique du signal est le choix de la période T_e ou de la fréquence d'échantillonnage ν_e .

Considérons un signal analogique $x(t)$ dont le spectre fréquentiel $X(\nu)$ est continu mais borné entre les deux fréquences maximales définies situées entre $-f_{max}$ et $+f_{max}$:

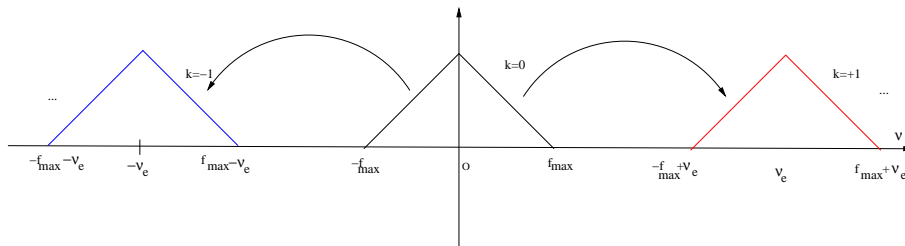
Si l'on considère un échantillonnage idéal du signal $x(t)$:

$$x_{ech}(t) = x(t) \cdot \sum_{n=-\infty}^{+\infty} \delta(t - nT_e) = \sum_{n=-\infty}^{n=+\infty} x(nT_e) \cdot \delta(t - nT_e) \quad (\text{I.17})$$

Le spectre du signal échantillonné peut être estimé simplement à partir de l'équation précédente:

$$X_{ech}(\nu) = X(\nu) * \nu_e \sum_{k=-\infty}^{+\infty} \delta(\nu - k\nu_e) = \nu_e \sum_{k=-\infty}^{+\infty} X(\nu - k\nu_e) \quad (\text{I.18})$$

Il est possible de remarquer que l'opération d'échantillonnage en temporel se traduit dans le domaine fréquentiel par une périodisation du spectre du signal continu. Les répliqués se situent au niveau des multiples de la fréquence d'échantillonnage ν_e .

Figure I.11: Spectre fréquentiel $X_{ech}(\nu)$ du signal échantillonné $x_{ech}(t)$.

I.4.2 Théorème de Shannon

En analysant la figure précédente, il est possible d'expliciter le théorème de Shannon. La condition pour qu'il n'y ait pas de recouvrement entre les répliqués au niveau de la période du spectre est $\nu_e - f_{max} \geq f_{max}$. Le théorème de Shannon spécifie le fait que pour un spectre de fréquence maximale f_{max} , la fréquence d'échantillonnage ν_e du système doit respecter la règle suivante:

$$\boxed{\nu_e \geq 2f_{max}} \quad (\text{I.19})$$

Le théorème de Shannon indique donc que pour échantillonner correctement un signal analogique, il est nécessaire que la fréquence d'échantillonnage soit au moins deux fois égale à la fréquence maximale du spectre f_{max} . Par contre, si cette condition n'est pas respectée, un phénomène de recouvrement et de l'aliasing entre le répliqué du spectre ne peut être exclu et il sera de fait difficile dans les faits de reconstruire le signal analogique à partir de celui numérique.

I.4.2 Fréquence de Nyquist

Dans le cadre du théorème de Shannon, il est intéressant de définir la fréquence de Nyquist $\nu_{nyquist}$:

$$\nu_{nyquist} = \frac{\nu_e}{2} \quad (\text{I.20})$$

Si ce théorème est respecté, la fréquence correspond à la fréquence maximale. Il n'est pas rare que des logiciels utilisent des fréquences normalisées et notamment par rapport à la fréquence de Nyquist. Par exemple, un spectre de fréquence situé entre $[-\frac{\nu_e}{2}, \frac{\nu_e}{2}]$ correspondra à l'intervalle normalisé $[-1; +1]$.

I.4.2 Limites du théorème de Shannon

Le théorème de Shannon est basé sur le postulat qu'il existe une fréquence maximale au niveau du spectre du signal continu $X(\nu)$. La figure suivante illustre la problématique par exemple pour

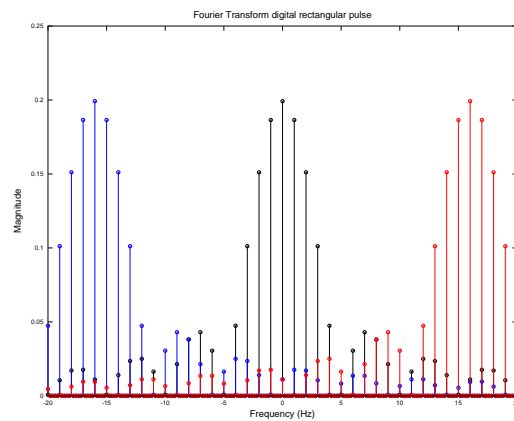


Figure I.12: Exemple du recouvrement des replicats dans le cas d'un spectre d'un signal rectangulaire.

des signaux carrés où il n'existe pas de fréquence maximale et l'échantillonnage va générer un recouvrement de spectres.

En pratique, dans une chaîne de traitement du signal, il est nécessaire de mettre un filtre anti-repliement ou anti-aliasing (AAF) en entrée de la chaîne de manière à pouvoir ensuite appliquer le théorème de Shannon.

I.4.2 Estimation performance globale d'un CAN

Malheureusement, la performance d'un convertisseur analogique/numérique est souvent estimée par sa vitesse de conversion. En effet, celle-ci est donnée par l'équation suivante:

$$P = 2^{N_{bits}} \nu_e \quad (\text{I.21})$$

qui représente la vitesse de conversion P c'est à dire le nombre de bits effectifs multiplié par

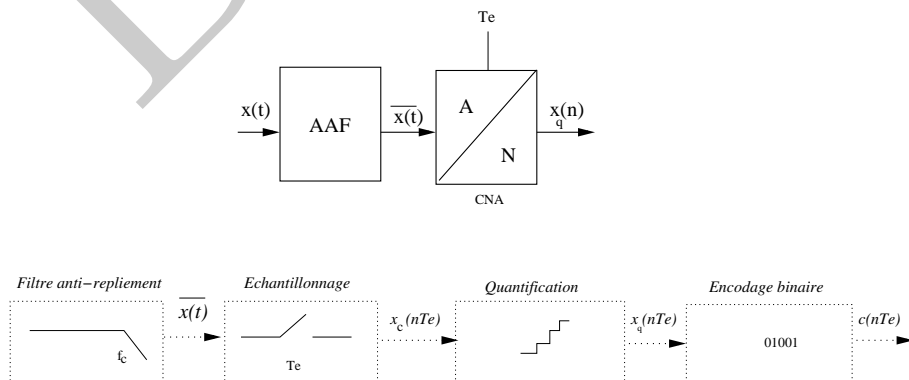


Figure I.13: Principes de base d'un convertisseur analogique/numérique

la fréquence d'échantillonnage. Le nombre de canaux disponible peut également être rajouté dans l'estimation de la vitesse de conversion dans ce cas, $P = 2^{n_{bits}} \nu_e n$. L'unité est exprimée en franç en nombre de bits par secondes (ou en anglais sps ou SPS...).

Il est possible de renormer ce critère en introduisant la puissance moyenne dissipée P_m par le composant, un critère de mérite (FoM) est donné par le rapport de la vitesse divisée par la puissance dissipée:

$$FoM = \frac{2^{N_{bits}} \nu_e}{P_m} \quad (I.22)$$

Ce dernier critère permet de classer les convertisseurs vis-à-vis du compromis en terme de vitesse de conversion, de puissance dissipée P_m voire de surface de circuit S .

I.4.3 Exercice

Considérer un signal analogique cosinusoidal de fréquence 50 Hz et d'amplitude 1V. Estimez la fréquence d'échantillonnage nécessaire. Un filtre anti-aliasing est-il obligatoire ? Ecrire l'expression de la séquence d'échantillons obtenus en considérant une fréquence d'échantillonnage double de celle de Shannon. En calculant les valeurs des premiers échantillons, représenter le signal échantillonné. Le signal numérique est-il toujours périodique ? Estimer finalement l'énergie de la séquence échantillonnée par rapport à celle du signal analogique.

I.4.3 Systèmes à multiple cadence

De nombreux systèmes numériques comportent des signaux avec des fréquences différentes d'échantillonnage pour des raisons applicatives ou liées à la nature des signaux physiques traités. Ainsi, il peut être utile de générer une nouvelle séquence numérique ν_e' avec une fréquence supérieure. Inversement, il est possible de générer une séquence numérique de fréquence avec moins d'échantillons correspondant à l'opération inverse à savoir une décimation.

Selon la variation du taux d'échantillonnage désiré $R \geq 1$ ou $R \leq 1$:

$$R = \frac{\nu_e'}{\nu_e} \quad (I.23)$$

il est possible de distinguer:

l'opération de décimation :

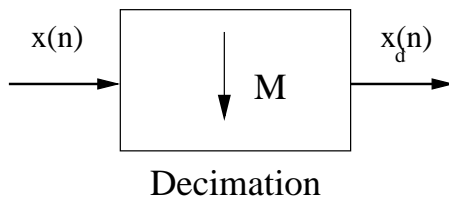


Figure I.14: Opération de décimation correspondant au fait de prendre un échantillon tous les M et ainsi diminuer la fréquence d'échantillonnage d'un facteur M .

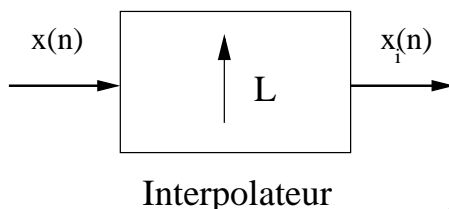


Figure I.15: Opération d'interpolation correspondant au fait d'augmenter le nombre d'échantillons et d'augmenter la fréquence d'échantillonnage d'un facteur L .

L'opération de décimation de manière à gérer un signal numérique consiste pour une série d'échantillons $x(n)$ à ne prendre que M valeurs, les autres $M - 1$ valeurs étant considérées comme nulles.

$$x_d(m) = \begin{cases} x(nM) & \text{si } m = nM \\ 0 & \text{autre} \end{cases} \quad (\text{I.24})$$

Cette opération permet ainsi de diminuer la fréquence d'échantillonnage de la série numérique x_d d'un facteur M .

l'opération d'interpolation:

L'opération d'interpolation ou plutôt d'expansion de la série $x_i(n)$ consiste à insérer entre chaque valeur de la série numérique $x(n)$, $L - 1$ valeurs nulles. Toutefois, en pratique, ces valeurs nulles seront remplacées par des valeurs interpolées de la série numérique.

$$x_i(m) = \begin{cases} x(n/L) & \text{si } n = 0, \pm L, \pm 2L, \\ 0 & \text{autre} \end{cases} \quad (\text{I.25})$$

Inversement, l'opération d'interpolation permettra ainsi d'augmenter la fréquence d'échantillonnage de la série numérique x_i d'un facteur L .

I.4.4 Quantification de signaux numériques

Les processeurs de traitement de signal numérique réalisent des calculs sur des nombres. Il est donc nécessaire que les valeurs d'entrée au niveau du processeur de traitement du signal soient codées sur un nombre entier de bits.

I.4.4 Principe de quantification

Le principe de quantification est le suivant:

- une loi de quantification Q va permettre de faire la correspondance entre un domaine continu de valeurs d'amplitude échantillonnée $x(n)$ et un ensemble de N valeurs discrètes $x_q(n) = Q[x(n)]$.
- une plage de conversion définissant les valeurs minimales x_{min} et maximales x_{max} de l'ensemble de quantification. La dynamique D maximale du convertisseur est ainsi définie par $D = x_{max} - x_{min}$.

Dans le cas d'une quantification uniforme, la plage de conversion ou de pleine échelle¹ entre x_{min} et x_{max} est divisée en un pas q égal de quantification l'écart entre deux valeurs quantifiées. Le pas de quantification uniforme est souvent pris égal permettant de définir le rapport entre la dynamique et le nombre de valeurs discrètes possible à savoir 2^{nbits} entre $[0; 2^{nbits-1}]$:

$$q = \frac{D}{N} = \frac{x_{max} - x_{min}}{2^{nbits}} \quad (I.26)$$

La résolution R obtenue lors d'une opération de quantification est définie comme étant le rapport entre le pas de quantification q et la dynamique du convertisseur D :

$$R = \frac{q}{D} = \frac{1}{2^{nbits}} \quad (I.27)$$

En pratique, cette résolution correspond à la valeur du bits de poids ou la position la plus faible (LSB). Ainsi, il existe deux règles de quantification uniforme utilisée à savoir l'arrondi ou la troncature :

- **Arrondi:** $x_q(n) = k \cdot q$ si $(k - \frac{1}{2}) \leq x(n) \leq (k + \frac{1}{2})$
- **Troncature:** $x_q(n) = (k + \frac{1}{2}) \cdot q$ si $(k \cdot q) \leq x(n) \leq (k + 1) \cdot q$

L'adoption de l'une ou l'autre de ces règles a une influence sur la forme de la caractéristique de transfert ou la valeur maximale de l'erreur quantifiée.

¹nota bene: la notation anglosaxone est plutôt FS pour Full Scale.

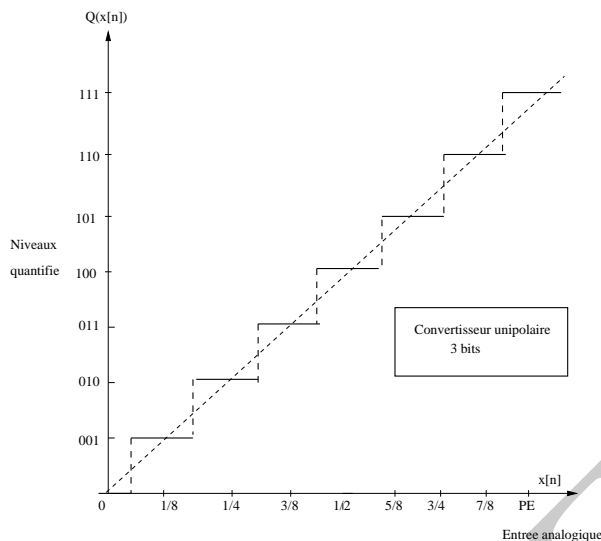


Figure I.16: Caractéristique de transfert modèle d'un convertisseur analogique/numérique centré sur 3 bits. La pente de la caractéristique du convertisseur analogique/numérique (CAN) théorique illustrée en pointillé est unitaire.

I.4.4 Caractéristiques de quantification

La caractéristique de quantification résume d'une manière synthétique l'opération de quantification implémentée par le convertisseur analogique numérique. Cette caractéristique est décomposée sous la forme de niveaux qui définissent la forme de la caractéristique de transfert à savoir un escalier. A chaque quantum, valeur quantifiée est associée un code qui peut-être défini par contre de manière arbitraire à l'aide d'un dictionnaire. Par exemple, si l'on analyse la caractéristique de transfert I.17, l'entrée analogique normalisée est divisée en différents niveaux ici (2^3) correspondant au centre de chacun des niveaux de l'escalier. A chaque niveau est associé un code permettant de décrire le quantum quantifié. L'étape de quantification permet de convertir un gamme d'amplitude continue en des niveaux discrets codé par des niveaux unipolaires

I.4.4 Erreur de quantification

L'erreur de quantification $\epsilon(n)$ est définie par la différence entre la valeur discrète de l'échantillon $x(n)$ et celle obtenue par l'opération de quantification $x_q(n)$:

$$\epsilon(n) = x(n) - x_q(n) \quad (\text{I.28})$$

La définition de l'erreur de quantification est une notion importante car elle décrit d'une certaine manière la distortion introduite lors de l'étape de quantification du signal et peut avoir une influence significative selon le type d'applications. Lorsque le signal d'entrée demeure dans la dynamique de la plage de conversion $x_{max} - x_{min}$, l'erreur de quantification demeure bornée. Par contre, si le

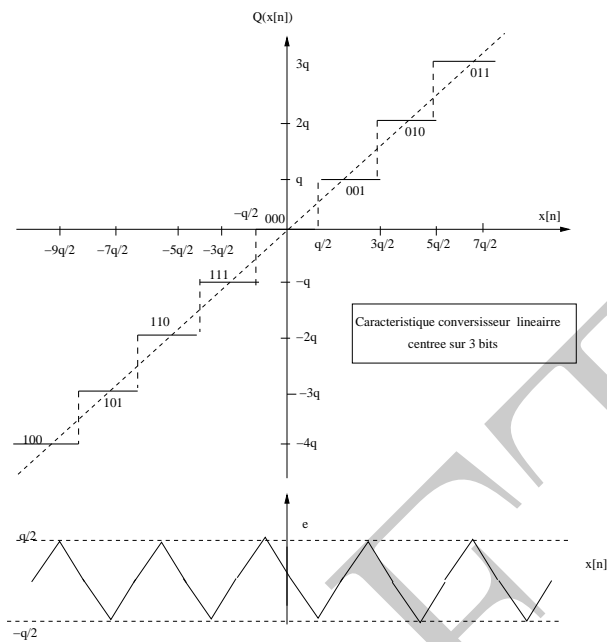


Figure I.17: Caractéristique de transfert modèle d'un convertisseur analogique/numérique centrée sur 3 bits. L'erreur de quantification générée par le convertisseur est illustrée au niveau de la figure ci-dessous .

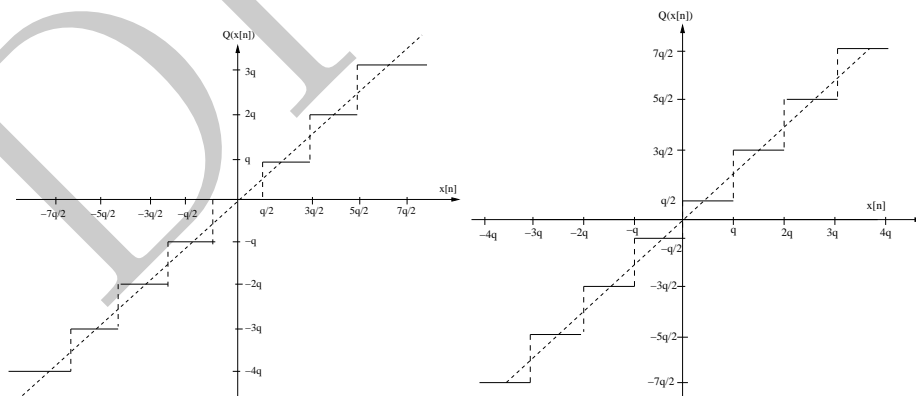


Figure I.18: Caractéristique linéaire centrée ou décalée de transfert d'un convertisseur analogique/numérique. Dans le cas de la caractéristique centrée (méthode arrondi), le niveau 0 est un état possible de la sortie du convertisseur analogique/numérique.

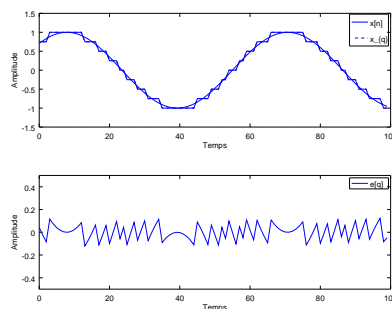


Figure I.19: Exemple de quantification d'un signal sinusoïdal sur 5 bits par la méthode de l'arrondi. L'erreur de quantification du signal est extraite juste après.

signal d'entrée n'est pas dans la plage de conversion $x_{max} - x_{min}$, l'erreur peut-être beaucoup plus importante dans la plage de saturation du convertisseur.

I.4.5 Exercice

Un convertisseur analogique/numérique fonctionnant sur 3 bits est caractérisé par une tension d'alimentation entre $[-5V, +5V]$. En vous aidant des notions abordées dans ce document, définissez l'ensemble des paramètres caractéristiques du convertisseur analogique/numérique (CAN).

I.4.5 Rapport signal à bruit

Le rapport signal à bruit (SNR) d'un convertisseur en échelle linéaire est défini par le rapport statistique de la variance du signal x_n par rapport à celle de l'erreur de quantification souvent assimilée à un bruit:

$$SNR_{A/D} = \frac{\sigma_{signal}^2}{\sigma_{bruit}^2} = \frac{E(x_n^2)}{E(\epsilon_n^2)} = \frac{\sum_{n=0}^{N-1} x_n^2}{\sum_{n=0}^{N-1} \epsilon_n^2} \quad (I.29)$$

Celui-ci peut être défini en décibel (dB) en utilisant l'échelle logarithmique suivante:

$$SNR_{A/D}(dB) = 10 \log(SNR_{A/D}) \quad (I.30)$$

Ce rapport qui est exprimé (en dB) permet de caractériser le composant en terme de niveau de bruit introduit par l'opération de quantification.

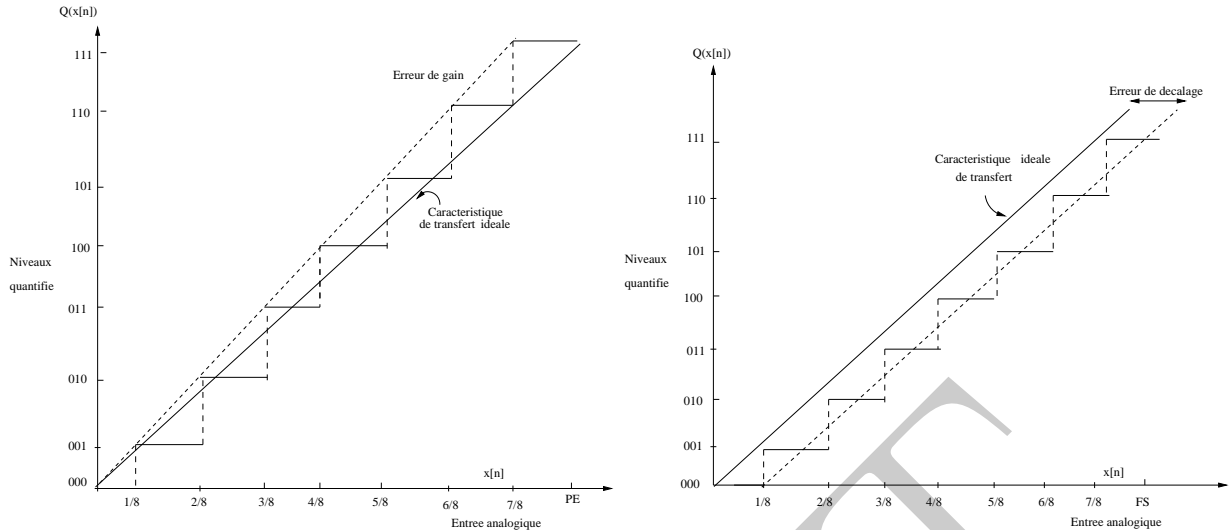


Figure I.20: a) Non-linéarité et caractéristique de transfert d'un convertisseur analogique/numérique. Ces non-linéarités peuvent être a) un gain qui dévie légèrement du gain théorique unité aboutissant à la fin de la dynamique à une erreur importante. b) Seconde non-linéarité illustrée est une erreur de décalage (offset) amenant à un décalage de l'ensemble de la caractéristique de transfert.

1.4.5 Rapport signal à bruit d'un signal sinusoïdal

Le rapport signal à bruit (SNR) peut être déterminé analytiquement dans le cas d'un signal sinusoïdal. En effet, pour la partie concernant le signal sinusoïdal quantifié, la variance est définie en fonction de l'amplitude du signal sinusoïdal V_{ref} :

$$\sigma_{signal}^2 = \frac{V_{ref}^2}{2} \quad (I.31)$$

Il peut être montré que la variance du bruit de l'erreur est égale à $\frac{q^2}{12}$ en estimant l'erreur de quantification d'un signal sinusoïdal. Ceci permet d'exprimer la variance du bruit de quantification en fonction de l'amplitude du signal et du nombre de n_{bits} :

$$\sigma_{bruit}^2 = \frac{q^2}{12} = \frac{V_{ref}^2}{2^{n-1} \cdot 12} \quad (I.32)$$

Le rapport des deux variances permet de simplifier l'expression qui se réduit à:

$$SNR^2 = \left(\frac{\sigma_{signal}}{\sigma_{bruit}} \right)^2 = 6 * 2^{2n-2} \quad (I.33)$$

Dans le cadre d'un signal sinusoïdal, il est possible d'aboutir à l'expression du rapport signal à

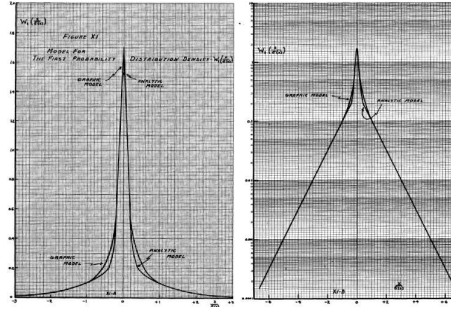


Figure I.21: Illustration de la densité de probabilité au niveau de la parole. Celle-ci est caractérisée par une distribution très pentue, la majeure partie des événements étant d'amplitude faible similaire à une distribution de Laplace .

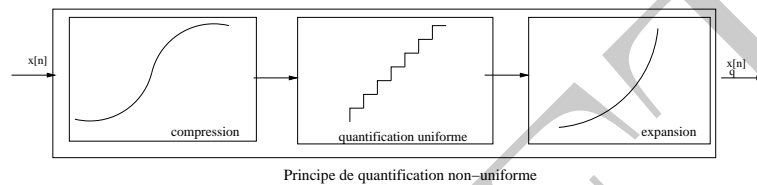


Figure I.22: La quantification non-uniforme est implémentée de la manière suivante: une étape de compression est réalisée permettant de définir des intervalles non uniformes, une étape de quantification uniforme classique et enfin une étape d'expansion permettant de décompresser en amplitude le signal quantifié.

bruit (SNR) en dB pour un quantifier:

$$SNR(dB) = 10 \log_{10}(6) + 20(n - 1) \log_{10}(2) = 6n_{bits} + 1.8 \quad (I.34)$$

Cette formule est souvent étendue et sert au niveau de l'ingénierie en pratique à estimer le rapport signal à bruit (SNR) théorique pour un nombre de bits donné.

I.4.5 Quantification non-uniforme

L'une des principales problématiques rencontrées au niveau de la quantification est que de nombreux signaux physiques se prêtent mal à une quantification uniforme. Si l'on prend l'exemple de la parole, les variations de valeur d'amplitude peuvent prendre des variations importantes. Néanmoins, la figure suivante présente la densité de probabilité expérimentale pour des enregistrements de parole et montre que la majorité des événements correspondent à des niveaux d'amplitude relativement faible I.21.

Une quantification de type uniforme n'est donc pas adaptée. Le principe de la quantification non-uniforme consiste à utiliser une loi logarithmique permettant d'amplifier les variations v d'amplitude les plus faibles et à l'inverse minimiser les amplitudes les plus élevées. En pratique, le pas de quantification est de plus en plus grand pour des valeurs de q_{signal} d'entrée de plus en plus importantes.

I.4.6 Exemple de quantification non-uniforme

La quantification non-uniforme est utilisée principalement dans des applications en téléphonie... Par exemple, une recommandation ancienne (G.711) utilise la loi μ permettant de compresser l'amplitude des différents échantillons par une loi pseudo-logarithmique sur 8 bits:

$$x_c(n) = \text{sgn}(x) \frac{\log(1 + \mu \frac{x(n)}{D})}{\log(1 + \mu)} \quad \text{où} \quad -1 \leq x \leq 1 \quad (\text{I.35})$$

où D est la dynamique de l'amplitude permettant de normaliser l'amplitude entre ± 1 et μ est un nombre entier en pratique $\mu = 255$ ($2^8 - 1$).

I.4.7 Encodage ou représentation binaire

L'opération de codage réalisée par le convertisseur analogique/numérique (CAN) permet à l'aide de la caractéristique de transfert de transformer une tension physique analogique en un code permettant de regrouper en série des niveaux binaires. Si l'on considère le codage binaire unipolaire, la relation la plus simple entre la tension du convertisseur V et la valeur des différents bits :

$$V = q \sum_{i=1}^N a_n 2^n \quad \text{où} \quad a_n = 1 \quad \text{ou} \quad 0 \quad (\text{I.36})$$

Le plus simple en terme de codage pour un convertisseur analogique/numérique est le codage binaire naturel ($a_0 a_1 a_2 a_{N-1}$) où a_0 correspond au bit de poids faible 2^0 (LSB) et a_{N-1} est celui de poids fort (MSB):

$$\text{nombre} = a_{N-1} 2^{N-1} + a_{N-2} 2^{N-2} + \dots + a_1 2^1 + a_0 2^0 \quad (\text{I.37})$$

Il est possible de définir les différents types de représentation binaire pour des convertisseurs analogiques/numériques :

- Signe et valeur absolue
- Binaire décentré ou décalé
- Complément à 1
- Complément à 2

Niveaux	Signé	Binaire décentré	Complément à 1	Complément à 2
+3	011	111	011	011
+2	010	111	010	010
+1	001	101	001	001
+0	000	100	000	000
-0	100	-	111	-
-1	101	011	111	-
-2	110	010	110	111
-3	111	001	101	110
-4	-	(000)		101

Table I.1: Niveaux de quantification et représentations binaires pour des convertisseurs linéaires.

Le tableau résume les différents niveaux de quantification pour un convertisseur 3 bits et les différents encodage possible.

I.4.8 Notion de reconstruction

L'étape de reconstruction d'une séquence numérique $x(n)$ consiste à réaliser l'opération inverse à savoir transformer une séquence numérique en une séquence analogique voire en un signal physique interprétable par l'humain.

I.4.8 Interpolation idéale de Shannon

La clef du théorème de Shannon est probablement l'aspect reconstruction. Sur le plan théorique, il est possible de retrouver le signal analogique original $x_a(t)$ à partir des échantillons numériques $x(k)$ selon la formule d'interpolation de Whittaker-Shannon:

$$x_a(t) = \sum_{k=-\infty}^{+\infty} x(k)h(t - kT_e) \quad \text{où} \quad h(t) = \frac{\sin(\pi\nu_e t)}{\pi\nu_e t} \quad (\text{I.38})$$

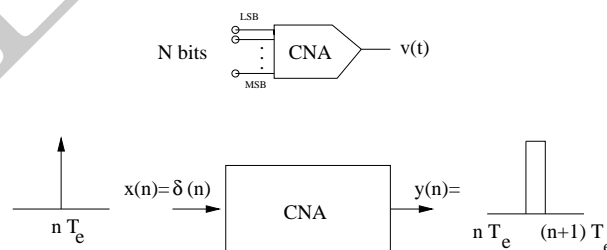


Figure I.23: Principe de fonctionnement d'un convertisseur numérique/analogique. En entrée de celui-ci, une impulsion unité représentant une valeur du signal numérique, ce dernier va transformer le signal numérique en un signal porte ayant avec une amplitude identique à celle du signal d'entrée mais constant durant la période T_e .

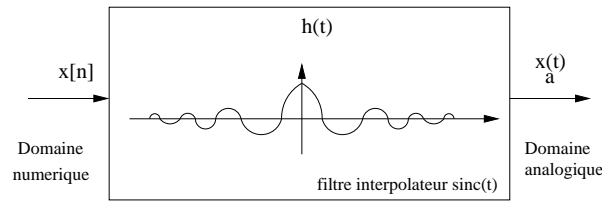


Figure I.24: Opération de reconstruction par un filtre idéal en sinus cardinal permettant d'implémenter la formule d'interpolation de Whittaker-Shannon.

Par contre, il existe deux problématiques de manière à appliquer cette interpolation à savoir que le signal analogique d'entrée $x_a(t)$ doit être caractérisé par un spectre borné en fréquence. En outre, l'implémentation du filtre en *sinc* n'est pas forcément simple, comme pour le calcul de la transformée de Fourier il est en effet nécessaire de faire une somme infinie d'échantillons.

I.4.8 Reconstruction en pratique

En pratique, l'étape de reconstruction peut-être implémenté par des interpolations. Le plus simple est celui adopté dans la description du convertisseur numérique/analogique (CNA), il s'agit du bloqueur d'ordre zero $h_0(t)$ qui permet de maintenir une amplitude constante entre deux valeurs numériques:

$$x_a(t) = \sum_{k=-\infty}^{+\infty} x(k)h_0(t - kT_e) \quad \text{où} \quad h_0(t) = \Pi_{[k \cdot T_e, (k+1) \cdot T_e]}(t) \quad (\text{I.39})$$

- bloqueur d'ordre un: une interpolation linéaire entre deux valeurs numériques.
- bloqueur d'ordre deux: une interpolation parabolique entre plusieurs valeurs numériques.
- ...
- Interpolateur idéal de Shannon : génère une séquence d'amplitude continue à l'aide d'un filtre en sinus cardinal (*sinc*) partir des différentes valeurs numériques.

Il existe donc toute une large gamme possible afin d'implémenter la reconstruction d'un signal.

I.4.8 Caractéristique de transfert d'un CNA

La caractéristique de transfert d'un convertisseur numérique/analogique sur 3 bits est illustrée au niveau de la figure suivante. L'entrée numérique correspond à l'ensemble des valeurs discrètes, quantum possible.

A chaque mot binaire formé par $(a_{N-1}a_{N-2}...a_0)$ du convertisseur numérique/analogique est associé une grandeur physique en sortie en terme de courant ou de tension:

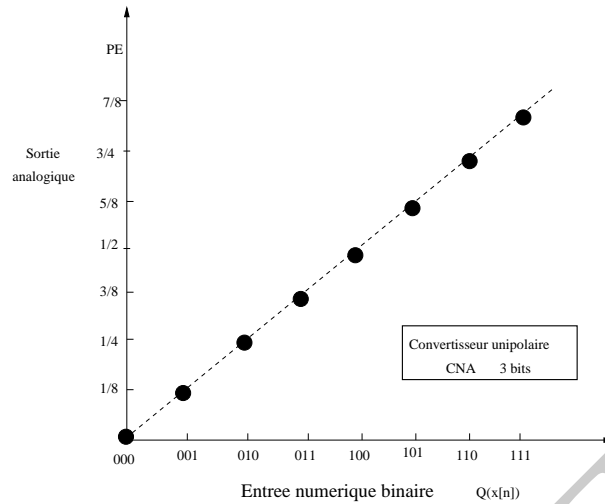


Figure I.25: Caractéristique de transfert d'un CNA sur 3 bits.

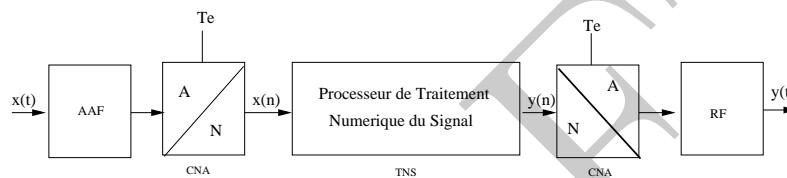


Figure I.26: Chaîne complète de traitement du signal numérique incluant en entrée un filtre anti-recouvrement (AAF) et en sortie un filtre de reconstruction ou lissage (RF).

$$v(t) = q \cdot (a_{N-1}2^{N-1} + a_{N-2}2^{N-2} + \dots + a_12^1 + a_02^0) \quad (\text{I.40})$$

I.5 Chaîne complète de traitement du signal

Il est possible de résumer l'ensemble des opérations (échantillonnage, quantification, filtre anti-recouvrement et filtre de reconstruction) de la chaîne complète de traitement du signal numérique:

Deux limites principales ont été identifiées dans la chaîne de traitement du signal numérique complète:

- Le recouvrement résiduel $R_{aliasing}$ généré par un filtrage non-idéal
- Le bruit de quantification lié à une résolution limitée du convertisseur analogique/numérique

La conception d'une chaîne de traitement du signal numérique quelque soit le signal physique à

traiter va concerner le choix des différents paramètres à savoir la résolution, la fréquence d'échantillonnage (ν_c) et la bande passante du système et enfin l'ordre du filtre anti-recouvrement. Les deux paramètres à savoir la résolution (CAN) demandée et la bande passante (ν_c):

- **Applications:** Résolution CAN ($nbits$) & bande passante (ν_c)
- **Compromis:** Fréquence d'échantillonnage (ν_e) et l'ordre du filtre anti-recouvrement (m)

Le recouvrement résiduel $R_{aliasing}$ au limite de la bande passante ν_c peut être estimé par:

$$R_{aliasing} = |H(\nu - \nu_e)|_{\nu=\nu_c} = \frac{1}{\sqrt{1 + \left(\frac{\nu_c - \nu_e}{\nu_c}\right)^{2m}}} \quad (\text{I.41})$$

La résolution du convertisseur analogique/numérique (CAN) est donnée par le nombre de bits: $R_{CAN} \cong \frac{1}{2^{nbits-1}}$.

D'une manière pratique, on peut supposer que la distortion introduite par le recouvrement de spectre soit inférieure à la distortion introduite par le convertisseur analogique/numérique:

$$R_{aliasing} < R_{CAN} \iff \sqrt{1 + \left(\frac{\nu_c - \nu_e}{\nu_c}\right)^{2m}} < 2^{nbits-1} \quad (\text{I.42})$$

Le rapport entre la fréquence d'échantillonnage ν_e et la fréquence de coupure ν_c est donnée par:

$$\frac{\nu_e}{\nu_c} > (1 + (2^{nbits-1})^{1/m}) \quad (\text{I.43})$$

Il est possible de décomposer cette équation sous la forme d'un tableau avec le nombre de bits du convertisseur analogique/numérique (CAN) en entrée ainsi que l'ordre du filtre (m):

	Résolution du CAN (N_{bits}):				
Ordre du filtre m :	8	10	12	14	16
4	4.4	5.8	7.7	10.6	14.5
6	3.3	3.9	4.6	5.5	6.7
8	2.9	3.2	3.6	4.1	4.7

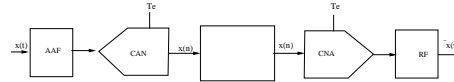


Figure I.27: Principe d'analyse d'une chaîne convertisseur analogique/numérique et numérique/analogique et estimation de la distortion introduite.

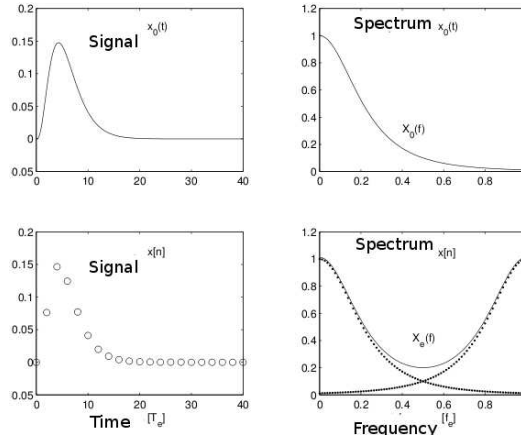


Figure I.28: Recouvrement spectral et absence de filtre anti-aliasing.

Si l'on fait l'hypothèse de l'utilisation d'un filtre anti-recouvrement d'ordre 8, la dernière ligne du tableau permet d'estimer un ordre de grandeur situé entre 3 et 5 pour le rapport entre ν_e et la fréquence de coupure ν_c du filtre définissant la bande passante:

$$\nu_e \cong 3...5\nu_c \quad (\text{I.44})$$

I.6 Analyse d'une chaîne complète CAN-CNA

L'objectif de cette section est de modéliser quasiment une chaîne complète CAN-CNA mais aussi de représenter l'influence ou l'absence d'un des composants clés de la chaîne de traitement numérique (filtre anti-aliasing pour le convertisseur analogique/numérique) ou d'un filtre de reconstruction ou lissage pour le convertisseur numérique/analogique.

Il s'agit d'envoyer dans la chaîne de traitement un signal analogique bref mais de forme et d'amplitude différente des signaux usuels. La figure suivante présente la forme du signal considéré, sa forme échantillonnée et le recouvrement de spectre non négligeable dû à l'échantillonnage au centre de la bande passante. Il peut être observé que celui-ci est maximum à la moitié de la fréquence d'échantillonnage à $0.5 \nu_e$.

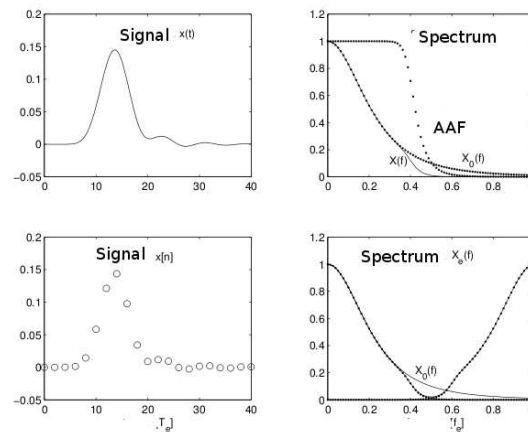


Figure I.29: Attenuation du recouvrement spectral par une opération de filtrage.

I.6.1 Influence d'un filtre anti-recouvrement

La figure suivante illustre l'influence de la présence d'un filtre anti-recouvrement (AAF). En coupant les composantes à hautes fréquences, il est possible d'observer que le recouvrement de spectre est alors effectivement fortement diminué. Par contre, l'influence du filtre se traduit au niveau temporel par une légère distorsion et l'introduction d'un délai.

I.6.2 Influence d'un filtre de reconstruction, lissage

La figure suivante présente le signal de la chaîne de traitement numérique, le recouvrement de spectre fortement atténué. La partie suivante présente le bloqueur d'ordre zero ainsi que sa réponse fréquentielle.

L'influence des étapes d'échantillonnage et de quantification se retrouve au niveau du signal analogique d'entrée mais avec une distorsion importante introduite par les paliers du bloqueur. Cela

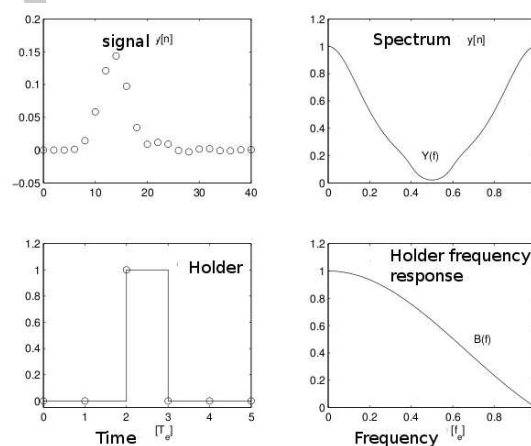


Figure I.30: Influence d'un bloqueur d'ordre zero et réponse en fréquence.

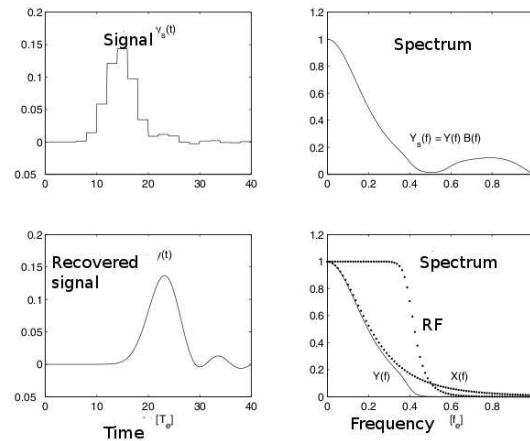


Figure I.31: Influence d'un filtre de reconstruction sur le signal extrait d'un bloqueur d'ordre zero.

se traduit au niveau du spectre par la présence d'une composante à haute fréquence absente dans le signal $x(t)$ initial. L'utilisation d'un filtre de reconstruction ou de lissage (RF) permet effectivement de filtrer cette partie à plus haute fréquence, de lisser le signal extrait de la sortie du bloqueur et enfin de retrouver un signal similaire proche de celui initial. Il est intéressant de noter que l'ensemble de la chaîne se traduit par la présence d'un délai et d'une distorsion significative du signal dans le signal échantillonné puis reconstruit.

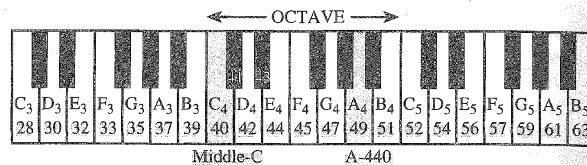
I.7 Conclusion

Ce cours a présenté la structure simple d'une chaîne de traitement du signal numérique principe qui est la base de très nombreuses applications. Il insiste sur le rôle clef de différents composants des convertisseurs analogiques/numériques et numériques/analogiques (CNA) dans les fonctions d'échantillonnage, de quantification et de reconstruction. Au delà de la simple fréquence d'échantillonnage et du nombre de bits utilisés, d'autres paramètres clefs permettent de définir finement la performance et le domaine d'application du système numérique comme l'architecture (non présenté), le temps d'acquisition, de conversion, l'ordre des filtres anti-recouvrement, le critère de mérite estimant la vitesse de conversion en lien avec l'énergie consommée par le système notamment pour des applications embarquées où celle-ci est précieuse...

Applications	Paramètres importants de la chaîne numérique
Audio	Résolution (ex: 16 bits, 96 dB)
Contrôle, automatique	Monotonie, stabilité
Analyse image	Résolution, rapidité de conversion
Mesures physiques	Résolution, temps d'acquisition (ex: 18 bits, SNR \geq 100 dB)
Analyse spectrale	Rapport signal à bruit
Télécommunication	Compression signal, énergie (ex: 8 bits, 50 dB)
Vidéo	Réponse en fréquence, rapidité de conversion
...	

Tutoriel 1: Piano numérique

Le principal objectif de ce tutoriel est d'illustrer les opérations d'échantillonnage, de quantification et de recouvrement d'une sinusoïde numérique générée par un piano numérique. Il s'agit de synthétiser une sinusoïde à l'aide d'Octave ou de Matlab et de générer la note au travers d'une carte son. Le piano suivant est considéré :



Le synthétiseur numérique est illustré au niveau de la figure suivante. Celui-ci est divisé en octave, chaque note musicale dans chaque octave correspondant au double de la fréquence pour l'octave précédente. Chaque octave contient douze notes de piano comme le montre la figure précédente c'est-à-dire 5 notes noires et 7 blanches. Le ratio fréquentiel entre chaque note est constant et correspond à $2^{1/12}$. La note A₄, correspondant à la note musicale de référence La correspond à une fréquence de 440 Hz.

1. Calculer la fréquence correspondant à la note 41 du piano.
2. Estimer la fréquence de Nyquist nécessaire pour générer un signal sinusoïdal propre et ainsi en déduire la fréquence minimale d'échantillonnage.

Le signal numérique est généré en utilisant un rythme de 500 échantillons par seconde.

3. Exprimer la forme mathématique de la séquence numérique générée.
4. En utilisant le logiciel Octave ou Matlab, tracer la forme discrète de forme du signal durant trois secondes. Ecouter la note.
5. La séquence du sinus numérique est-il toujours périodique ?
6. Existe des fréquences dites apparentes ?

Générer la note avec maintenant une fréquence d'échantillonnage plus réaliste par exemple 11025 Hz soit le quart de la norme CD.

7. Si l'on envisage une qualité audio-numérique au niveau du piano. Estimer la résolution nécessaire au niveau du convertisseur analogique/numérique CAN de quantification uniforme pour le piano numérique. Prener en compte une non-idéalité possible se traduisant par erreru estimée un demi LSB.

8. En utilisant Octave ou Matlab et une résolution en guise d'illustration de 3 bits, tracer alors le forme de la sinusoïde quantifiée en utilisant la règle de l'arrondi.
9. En utilisant Octave/Matlab, réaliser une quantification non-linéaire en considérant la loi μ et une valeur de 255.
10. Enfin tracer la forme du signal reconstruit en utilisant un bloqueur d'ordre zero.

DRAFT

II – Analyse Spectrale de Signaux Numériques

II.1 Introduction

II.1.1 Aspect historique



John Tukey (1915-2000)



James Cooley (1926-2016)



Nasir Ahmed (1940-)

Les notions abordées dans ce cours et principalement la transformée de Fourier rapide sont principalement le résultat des travaux de John Tukey et James Cooley. John Tukey était un statisticien américain, professeur de statistiques à l'université de Princeton et chercheur au laboratoire Bell. James Cooley était un mathématicien américain et l'un des pionniers dans le domaine du traitement du signal numérique. Il a travaillé à IBM Watson et a co-développé l'algorithme de calcul de transformée de Fourier rapide (FFT) avec John Tukey dans les années 1965. La transformée en cosinus discret est plus récente et concerne le travail de recherche de Nasir Ahmed (1940-) en 1972 pour la compression d'image.

II.1.2 Applications

De nombreuses applications de la vie de tous les jours (son, image, vidéos) requièrent la connaissance ou tout du moins une estimation du spectre fréquentiel du signal:

- Son (MP3) : encodage et compression des données.
- Image (format Jpeg): encodage et compression des données.
- Vidéos : encodage, compression au niveau des codecs.
- Télécommunication: création et décodage d'un signal ADSL.
- Mesure : analyse fréquentielle de signaux complexes.

II.2 Transformée de Fourier (TF)

II.2.1 Définition

La définition mathématique de la transformée de Fourier permet de calculer le spectre $X(\nu)$ d'un signal temporel continu $x(t)$:

$$X(\nu) = \int_{-\infty}^{+\infty} x(t) \exp(-2i\pi\nu t) dt \quad (\text{II.1})$$

Sur le plan applicatif, les différentes contraintes (contraintes mémoires, temps de calcul) ne permettent pas directement une implémentation de la transformée de Fourier classique (TF) sur un dispositif de calcul:

- le signal $x(t)$ est un signal temporel continu
- le domaine d'intégration est infini
- le spectre $X(\nu)$ est également continu

Il est donc nécessaire de simplifier l'équation II.1 ou tout du moins d'essayer de formuler une version implémentable.

II.2.2 Vers la transformée discrète

L'un des premiers verrous est que sur le plan de la définition mathématique, le signal $x(t)$ est continu à la fois en temporel et en amplitude. La première approximation va donc consister à échantillonner le signal continu afin de générer un signal discret $x(nT_e)$. Si l'on introduit la période d'échantillonnage comme étant T_e , n correspond au numéro d'échantillons, il est possible de définir une transformée de Fourier numérique, le spectre $X(\nu)$ est pour l'instant continu:

$$X(\nu) = \sum_{n=-\infty}^{+\infty} x(nT_e) \exp(-2i\pi\nu nT_e) \quad (\text{II.2})$$

Une seconde approximation consiste à réduire l'intervalle d'intégration qui est supposé infini. Pour cela, l'intégration va se limiter à une somme de N valeurs.

$$X(\nu) = \sum_{n=0}^N x(nT_e) \exp(-2i\pi\nu nT_e) \quad (\text{II.3})$$

Cette approximation de fait correspond à l'opération de fenêtrage du signal $x(nT_e)$ par une fenêtre rectangulaire, ce dernier étant considéré comme causal, les échantillons négatifs supposés de valeur nuls. Enfin, la dernière limitation vient du fait que le spectre calculé $X(\nu)$ demeure pour l'instant continu. Il est nécessaire de venir également échantillonner le spectre signal c'est à dire qu'il ne sera évalué que sur quelques fréquences spécifiques $X(\nu = k\nu_s)$. Afin d'éviter un recouvrement, cette fois-ci, dans le domaine temporel d'imposer le fait:

$$(N - 1)T_e < \frac{1}{\nu_s} \quad (\text{II.4})$$

La durée d'observation de la série temporelle $D = NT_e$ correspondant aux différents N échantillons doit être inférieure à l'inverse de la période des échantillons fréquentiels (ν_s). Classiquement, le choix de la résolution $\Delta\nu$ de la transformée de Fourier discrète est fait de telle manière que les échantillons spectraux ν_s soient réparties de manière uniforme entre $[0 - \nu_e]$:

$$\nu_s = \frac{k}{NT_e} \quad \text{où} \quad 0 \leq k \leq N - 1 \quad (\text{II.5})$$

Ceci aboutit à la définition de la transformée de Fourier Discrète (TFD):

$$X(k) = \sum_{n=0}^{N-1} x(n) \cdot \exp -\frac{2i\pi kn}{N} \quad \text{pour} \quad 0 \leq k < N - 1 \quad (\text{II.6})$$

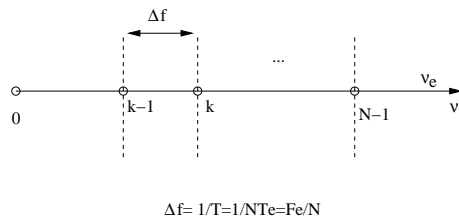


Figure II.1: Résolution $\Delta\nu$ des échantillons spectraux. Chaque échantillon spectral k de $[0; \nu_e]$ obtenu par la transformée de Fourier Discrète (DFT) est espacé de la résolution $\Delta f = \frac{\nu_e}{N}$.

Contrairement à la transformée de Fourier continue II.1, cette transformée similaire à la définition de Fourier continue est désormais implémentable sur un processeur de calcul. Par contre, il est important d'avoir à l'esprit que l'ensemble des différentes approximations qui ont été effectuées à savoir l'échantillonnage temporel, un fenêtrage du signal temporel sur N échantillons, échantillonnage fréquentiel pour aboutir à la définition d'une transformée implémentable.

II.3 Transformée de Fourier Discrète (TFD)

II.3.1 Définition

La transformée discrète (**TFD**) d'une séquence numérique $x(n)$ est définie par l'équation suivante :

$$X(k) = \sum_{n=0}^{N-1} x(n) \cdot W_N^{kn} \quad \text{pour} \quad 0 \leq k < N - 1 \quad (\text{II.7})$$

avec W l'opérateur $W_N^{nk} = \exp -\frac{2i\pi kn}{N}$

D'une manière équivalente, la transformée discrète inverse (**IDFT**) peut être également définie par l'équation suivante:

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) \exp \frac{2i\pi kn}{N} \quad n = 0, \dots, N - 1 \quad (\text{II.8})$$

II.3.2 Expression matricielle de la transformée de Fourier discrète

Il est possible d'exprimer la suite d'échantillons temporels sous la forme d'un vecteur :

$$x(n) = [x(0) \cdots x(n) \cdots x(N - 1)]^T \quad (\text{II.9})$$

La matrice de transformée discrète (TFD) est alors donnée par:

$$T_n = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \exp\left(\frac{-2i\pi nk}{N}\right) & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \cdots & \cdots & \cdots & \cdots \end{bmatrix} \quad (\text{II.10})$$

Le vecteur d'échantillons spectraux résultant est alors donné simplement par le produit de la matrice de transformée discrète (TFD) et les échantillons temporels $x(n)$ de la suite numérique:

$$X = T_n \cdot x \quad (\text{II.11})$$

ou plus précisément en décomposant:

$$\begin{bmatrix} X(0) \\ \vdots \\ X(k) \\ \vdots \\ X(N-1) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \exp\left(\frac{-2i\pi nk}{N}\right) & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \cdots & \cdots & \cdots & \cdots \end{bmatrix} \begin{bmatrix} x(0) \\ \vdots \\ x(n) \\ \vdots \\ x(N-1) \end{bmatrix} \quad (\text{II.12})$$

L'implémentation directe de la transformée de Fourier discrète nécessite donc $N \cdot (N-1)$ additions, N^2 multiplications ce qui amène à une complexité de l'algorithme en $\mathcal{O}(N^2)$.

Considérer la transformée discrète (TFD) sur 4 points de la séquence numérique $x(n)$ suivante:

$$x(n) = [1 \ 0 \ 1 \ 0] \quad (\text{II.13})$$

Ecrire la matrice de TFD permettant le calcul de la transformée de Fourier Discrète. Calculer alors les échantillons spectraux.

II.4 Transformée de Fourier Rapide (TFR)

Il existe différentes implémentations d'algorithme cherchant à simplifier la transformée de Fourier discrète mais avec moins d'opérations implémentant de réaliser des calculs:

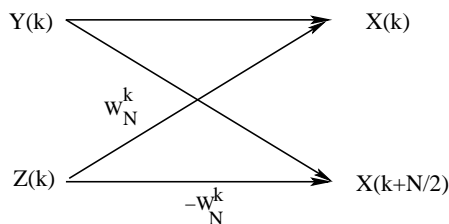


Figure II.2: Définition de l'opérateur papillon permettant d'implémenter l'opération d'entrelacement pour la définition de la transformée rapide.

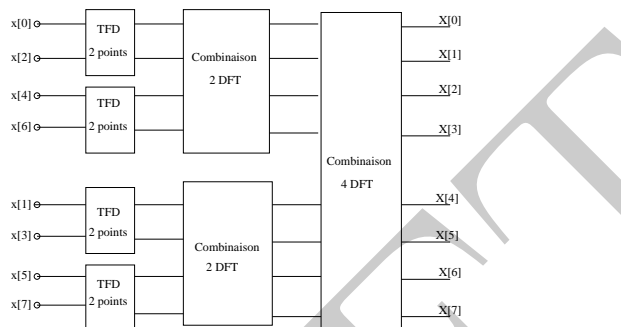


Figure II.3: Principe de décimation fréquentielle d'une transformée de discrète (TFD).

$$X(k) = \sum_{n=0}^{N-1} x(n) \cdot W_N^{nk} \quad (\text{II.14})$$

Ces implémentations utilisent des propriétés de l'opérateur W complexe W_N^{k+N} à savoir que :

$$\begin{cases} W_N^{k+N/2} = -W_N^k & (\text{Notion de symétrie}) \\ W_N^{k+N} = W_N^k & (\text{Notion de périodicité}) \end{cases} \quad (\text{II.15})$$

L'approche afin de simplifier le calcul de cette expression adoptée à l'époque est de supposer que le signal soit de longueur composite c'est à dire décomposer la série de N échantillons en deux sous-séries $N = ab$ ou en puissance de 2. Le plus simple est par exemple de séparer la somme en deux parties $N/2$ c'est à dire les termes pairs ($2n$) et impairs ($2n+1$):

$$X(k) = \sum_{n=0}^{N/2-1} x(2n)W_N^{2nk} + \sum_{n=0}^{N/2-1} x(2n+1)W_N^{(2n+1)k} \quad (\text{II.16})$$

L'introduction de l'opérateur papillon permet de décrire l'opération d'entrelacement en temporel:

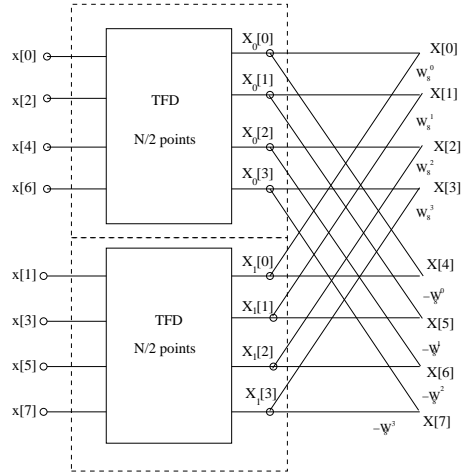


Figure II.4: Développement sous la forme de l'opérateur papillon du dernier bloc de transformée discrète (TFD).

$$\begin{cases} X(k) = X_0(k) + W_N^k X_1(k) & k = 0, 1, \dots, N/2 - 1 \\ X(k + N/2) = X_0(k) - W_N^k X_1(k) & k = 0, 1, \dots, N/2 - 1 \end{cases} \quad (\text{II.17})$$

Il est possible de ré-itérer la même opération pour le bloc intermédiaire pour chacune des deux séries d'échantillons fréquentiels $X_0(k)$ et $X_1(k + N/2)$ afin de générer:

$$\begin{cases} X_0(k) = X_{00}(k) + W_N^k X_{01}(k) & k = 0, 1, \dots, N/4 - 1 \\ X_0(k + N/4) = X_{00}(k) - W_N^k X_{01}(k) & k = N/4, \dots, N/2 - 1 \\ X_1(k + N/2) = X_{10}(k) + W_N^k X_{11}(k) & k = N/2, \dots, 3N/4 - 1 \\ X_1(k + N/2) = X_{10}(k) - W_N^k X_{11}(k) & k = 3N/4, \dots, N \end{cases} \quad (\text{II.18})$$

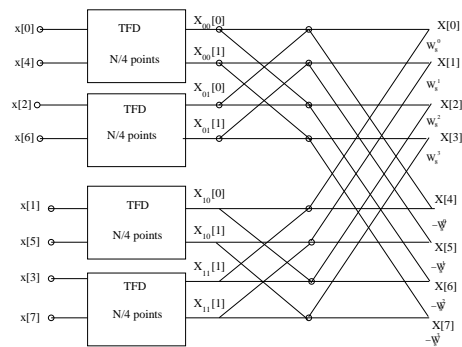


Figure II.5: Développement de la partie intermédiaire de la DFT sous la forme d'opérateur papillon.

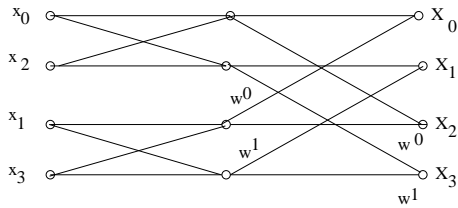


Figure II.6: Transformée de Fourier rapide (TFR) à entrelacement temporel d'ordre 4.

II.4.1 Exercice

Ci-joint dessus se trouve un exemple de représentation de transformée de Fourier rapide d'ordre 4 à entrelacement temporel. Développer les relations entre échantillons temporels et spectraux. Retrouvez-vous l'expression de la matrice de TFD d'ordre 4 ?

II.4.2 Exemple de calcul simple sur Octave&Matlab

Afin de réaliser une implémentation de calcul simple sur Octave, il est nécessaire de définir un tableau d'échantillons à la période d'échantillonnage fixée ainsi que la séquence numérique :

```
f0=200;f1=400;fe=4000;Te=1/fe;n=[0:1:100];t=n*Te;
x=sin(2*pi*f0*t)+4*sin((2*pi*f1*t+pi/6));
```

Les logiciels octave ou matlab ne définissent pas par défaut l'échelle des fréquences discrètes afin de représenter la transformée de Fourier discrète calculées ici sur dizaines échantillons.

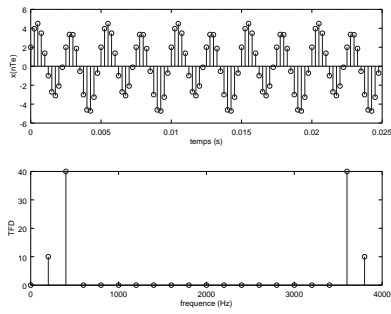
```
N=10;n2=[0:1:N-1];
vk=(1/(N*Te))*n2;
```

La fonction `fft.m` par contre implémente un algorithme de calcul rapide de transformée de Fourier discrète:

```
y=fft(x,N);
```

```
figure(1)
subplot(211)
stem(t,x)
xlabel('temps (s)')
ylabel('x(nTe)')
```

```
subplot(212)
```

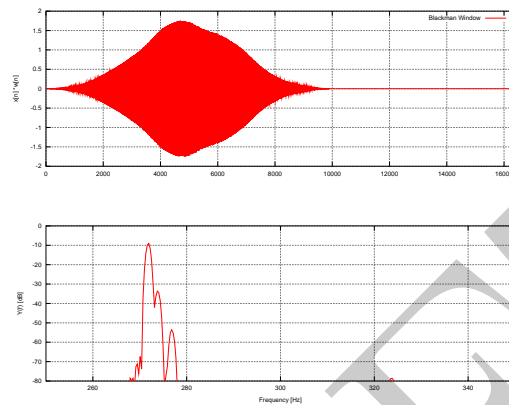


```
stem(vk,abs(y));
xlabel('frequency (Hz)')
ylabel('TFD')
```

La fonction `fftshift.m` permet de réordonner les échantillons de transformée discrète $[-\nu_{nyquist}; \nu_{nyquist}]$. Les échantillons spectraux de transformée de Fourier Discrète sont souvent obtenus entre $[0;$

Tutoriel : transformée de Fourier rapide et spectre numérique

L'objectif principal de ce tutoriel est de réaliser l'analyse du spectre d'un signal numérique en utilisant les informations contenues dans le signal et également d'identifier une fréquence pas forcément visible dans le signal original observé.



1. Télécharger les deux fichiers texte contenant la forme analogique et numérique du signal en cours d'étude et tracer ses évolutions temporelles en utilisant Octave ou Matlab. Pouvez vous faire un commentaire sur les propriétés du signal ?
2. A l'aide des formes temporelles analogique et échantillonnée du signal donner une estimation de la période du signal et des deux fréquences à -priori contenues

Supposons que la fréquence d'échantillonnage soit de 5 kHz et que le signal ait été observé durant deux secondes.

3. Estimer le nombre d'échantillons étudié.

Certaines implémentations de l'algorithme de transformée rapide nécessite le fait qu'elle soit caractérisée par un ordre de puissance 2^n

4. Donner l'ordre de la TFR et estimer la résolution fréquentielle et la finesse d'analyse. Discuter la différence entre les deux notions.
5. A l'aide d'octave ou matlab en utilisant une fenêtre rectangulaire, calculer la TFR du signal. Etes-vous capable d'identifier les fréquences initiales contenues dans le signal ? (nota bene: il peut être intéressant de tracer le spectre d'amplitude en échelle logarithmique)
6. Réaliser la même opération mais en utilisant une fenêtre plus évoluée moins sensible au phénomène de Gibbs.

7. Estimer les amplitudes relatives des différentes composantes fréquentielles en échelle logarithmique. On normalisera la première composante à une amplitude unité.

DRAFT

DRAFT

III – Systèmes numériques et Transformée en z

III.1 Introduction

III.1.1 Aspect historique



Witold Hurewicz (1904-1956) était un mathématicien polonais et émigré aux Etats-Unis d'Amérique. Il était un expert en topologie et dans le contexte de la seconde guerre mondiale, il a contribué à la modélisation du système radar permettant ainsi de contribuer au domaine des mathématiques appliquées avec la découverte de la transformée en z en 1947.

III.1.2 Introduction

Les notions abordées dans ce cours s'inscrivent dans le cadre de la transformée en z et sont celles développés par Witold Hurewicz (1904-1956). Wiltold Hurewicz était un mathématicien polonais et émigré aux Etats-Unis d'Amérique. Il était un expert en topologie lorsque durant le seconde guerre mondiale, il a contribué à la modélisation du système radar permettant ainsi de contribuer au domaine des mathématiques appliquées et la découverte de la transformée en z en 1947. Cette dernière est un outil mathématique indispensable et similaire à la transformée de Laplace pour la description des systèmes discrets permettant également la définition des systèmes simples, l'analyse de leurs propriétés et enfin les calculs des fonctions de transfert.

III.2 Description des systèmes numériques

III.2.1 Définition

Un système numérique est une fonction matérielle ou un algorithme permettant d'implémenter une transformée Z et de transformer le signal numérique d'entrée $x(n)$ en une sortie également numérique $y(n)$. Il est possible de décrire mathématiquement un système numérique par l'équation suivante:

$$y[n] = Z\{x[n]\} \quad (\text{III.1})$$

où $x[n]$ est l'entrée du signal, $y[n]$ la sortie du système et T la transformée implémentée par le système. D'une manière équivalente au système continu, la représentation sous forme de diagramme fonctionnel d'un système numérique est souvent préférée.

III.2.2 Exemple de systèmes numériques simples

Il est possible de représenter des systèmes numériques par les équations suivantes:

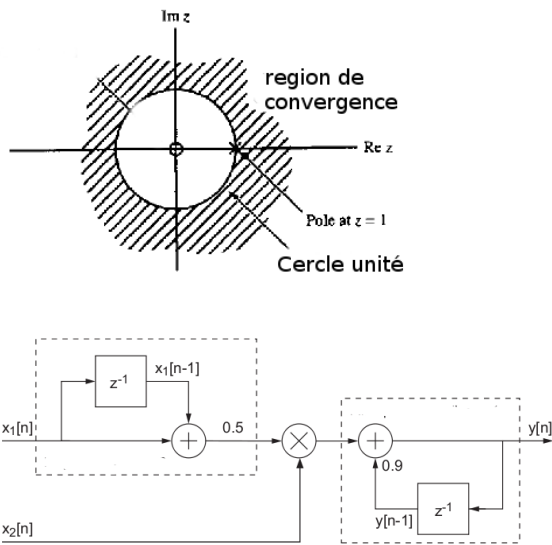
- $y[n] = x[n]$
- $y[n] = x[n - 1]$
- $y[n] = x[n + 1]$
- $y[n] = \max\{x[n - 1], x[n], x[n + 1]\}$
- $y[n] = \sum_{-\infty}^n x[k]$
- $y[n] = x[n] - x[n - 1]$

III.2.3 Exercice

Qualifier c'est dire (donner un nom) aux systèmes précédents. En considérant la séquence numérique suivante: $\{0, 1, 2, 3, 4, 5, 0, 0, 0, 0, \dots\}$, estimer la réponse de ces systèmes numériques simples.

III.2.4 Critère de stabilité

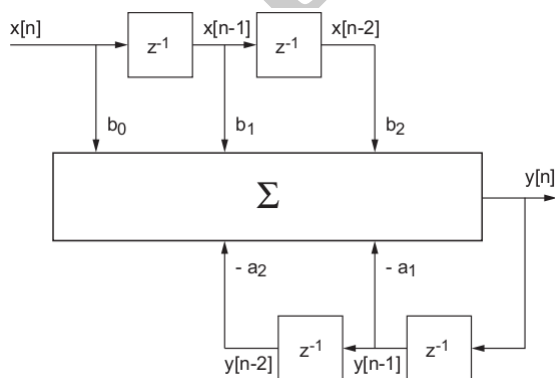
Un système est stable si les modules des pôles ($p_i \leq |1|$) de sa fonction de transfert sont inclus dans le cercle unité (ROC).



III.3 Diagramme fonctionnel

La description d'un système numérique peut également être décrit à l'aide d'un diagramme fonctionnel. Ce dernier est alors composé des opérations suivantes:

- Addition: \oplus
- Multiplication entre deux signaux: \otimes
- Multiplication par un coefficient: \rightarrow
- Retard: z^{-1}
- Avance: z



III.4 Propriétés des systèmes numériques

Les systèmes numériques sont caractérisés par des propriétés

III.4.1 Définition des propriétés de systèmes numériques

III.4.1 Statique

Un système statique est un système où la sortie est dépendante seulement de l'entrée du signal à un instant ($t = nT_e$) ou l'échantillon (n) est donné par nT_e : $y(n) = ax(n) + bx(n)^2$.

III.4.1 Dynamique

Un système dynamique est un système où la sortie dépend des états précédents ou futurs du signal d'entrée: $y(n) = ax(n) + bx(n-1) + cx(n+1)$. Par exemple, la sortie de ce système dépend de l'instant présent, de celui précédent mais également de l'instant suivant.

III.4.1 Linéaire

Un système linéaire satisfait au principe de superposition c'est-à-dire que si l'on considère deux signaux $x_1(n)$ et $x_2(n)$:

$$\begin{aligned} y[n] &= T\{ax_1[n]\} + b\{x_2[n]\} \\ &= aT\{x_1[n]\} + bT\{x_2[n]\} \\ &= y_1[n] + y_2[n] \end{aligned}$$

III.4.1 Stabilité

Un système est dit inconditionnellement stable si la sortie ne diverge pas quelle que soit la séquence numérique ou digitale qui est en entrée.

III.4.1 Superposition

Un système linéaire satisfait au principe de superposition:

$$\begin{aligned}
 y[n] &= T\{ax_1[n]\} + b\{x_2[n]\} \\
 &= aT\{x_1[n]\} + bT\{x_2[n]\} \\
 &= y_1[n] + y_2[n]
 \end{aligned}$$

III.4.1 Invariant

Un système invariant est un système où le retard à l'entrée se traduit simplement par le retard au niveau de la sortie d'amplitude équivalente.

$$T\{x(n+d)\} = y(n+d) \quad (\text{III.2})$$

III.4.1 Causalité

Un système causal est un système où la sortie ne dépend que des états actuels ou précédents de la séquence d'entrée. Il ne dépend pas des états futurs du signal.

Quelles propriétés sont satisfaites par les systèmes suivants:

- un système accumulateur numérique implémentant une fonction équivalente à celle d'un intégrateur:

$$y(n) = \sum_{k=-\infty}^n x(k)$$

- un dérivateur numérique:

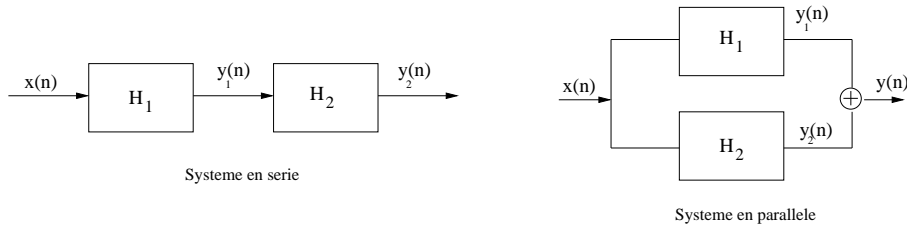
$$y(n) = x(n+1) - x(n)$$

- un système invariant et quadratique:

$$y(n) = x(n)^2$$

- un système invariant et sous-échantillonnage:

$$y(n) = x(2n)$$



III.4.2 Interconnection des systèmes numériques

Au niveau de l'interconnection des systèmes numériques, il y a deux manières de les connecter à savoir en série ou en parallèle:

Dans un système connecté en série, les fonctions de transfert sont reliées de manière consécutive entre elles, l'entrée du premier système correspondant à la sortie du second système:

$$y[n] = H_2\{H_1\{x[n]\}\} \quad (\text{III.3})$$

Par contre, pour un système en parallèle, la connection se fait au niveau de la sortie du système, la sortie globale étant l'addition des deux signaux de sortie.

$$y[n] = H_2\{x[n]\} + H_1\{x[n]\} \quad (\text{III.4})$$

III.4.3 Notion de système LTI

Les systèmes numériques ne sont pas forcément simples à décrire et seuls les systèmes linéaire et invariant dans le temps (LTI), les concepts introduits par pour les systèmes continus demeurent valides :

- Réponse impulsionnelle
- Produit de convolution
- Fonction de transfert

III.5 Définition de la transformée en z (TZ)

III.5.1 Définition

La transformée en z (TZ) est une généralisation de la transformée discrète DFT. D'une manière similaire, l'utilisation de la transformée de Laplace, l'utilisation de la transformée en z permet de

décrire des opérations simples pour des systèmes numériques. Cette transformée en z est également un outil d'analyse et de conception de systèmes numériques au sens large comme on le verra dans les chapitres suivants. Si l'on considère une séquence discrète numérique $x[n]$, sa transformée en z est définie par l'équation suivante:

$$X(z) = \sum_{n=-\infty}^{+\infty} x[n]z^{-n} \quad (\text{III.5})$$

où z est une variable complexe continue dans l'espace complexe \mathbb{C} . Elle s'exprime souvent sous la forme d'un opérateur Z qui permet de faire le lien entre le domaine temporel discret $x(n)$ et le domaine en z .

$$Z\{x[n]\} = X(z) = \sum_{n=-\infty}^{+\infty} x[n]z^{-n} \quad (\text{III.6})$$

Cette relation peut s'exprimer de la manière suivante:

$$x[n] \leftrightarrow X(z) \quad (\text{III.7})$$

En considérant un dirac ou un échelon unitaire, calculer la transformée en z du signal numérique.

III.5.2 Table de transformée en z

Le calcul de la transformée en z de la séquence numérique plus complexe peut nécessiter des calculs fastidieux notamment par l'intermédiaire des résidus. La table suivante résume les transformées en z principales utiles.

Signaux numériques	Transformée en z	Domaine de convergence
$\delta(n)$	1	\mathbb{C}
$u(n)$	$\frac{1}{1-z^{-1}}$	$ z \geq 1$
$a^n u(n)$	$\frac{1}{1-az^{-1}}$	$ z \geq a $
$na^n u(n)$	$\frac{az^{-1}}{(1-az^{-1})^2}$	$ z \geq a $
$\cos(\omega_0 n)u(n)$	$\frac{1-z^{-1}\cos(\omega_0)}{1-2z^{-1}\cos(\omega_0)+z^{-2}}$	$ z \geq 1$
$\sin(\omega_0 n)u(n)$	$\frac{1-z^{-1}\cos(\omega_0)}{1-2z^{-1}\cos(\omega_0)+z^{-2}}$	$ z \geq 1$
$a^n \cos(\omega_0 n)u(n)$	$\frac{1-az^{-1}\cos(\omega_0)}{1-2z^{-1}\cos(\omega_0)+a^2z^{-2}}$	$ z \geq 1$
$a^n \sin(\omega_0 n)u(n)$	$\frac{az^{-1}\sin(\omega_0)}{1-2z^{-1}\cos(\omega_0)+a^2z^{-2}}$	$ z \geq 1$

III.5.3 Lien entre DFT et transformée en z

Si l'on pose $z = re^{j\omega}$, la définition de la transformée en z aboutit à :

$$X(e^{j\omega}) = \sum_{-\infty}^{+\infty} x(n)e^{-j\omega n} \quad (\text{III.8})$$

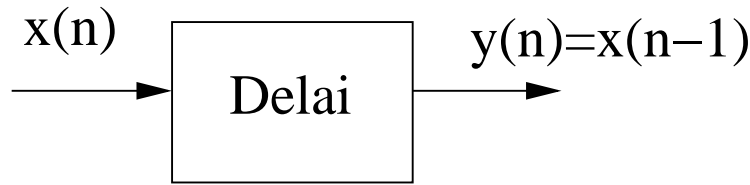
En comparant avec la transformée de la DFT $X(e^{j\omega})$ de la séquence $x(n)$:

$$X(e^{j\omega}) = \sum_{-\infty}^{+\infty} x[n]e^{-j\omega n} \quad (\text{III.9})$$

c'est-à-dire $X(re^{j\omega})$ est la transformée de la Fourier discrète de la séquence $\{x[n]r^{-n}\}$.

III.5.4 Résumé des propriétés de la transformée en z

Propriété	Séquence numérique	Transformée en z
	$g(n)$	$G(z)$
	$h(n)$	$H(z)$
Conjugaison	$g^*(n)$	$G^*(z^*)$
Renversement temporel	$g(-n)$	$G(1/z)$
Linearité	$ag(n) + bh(n)$	$aG(z) + bH(z)$
Décalage temporel	$g(n - n_0)$	$z^{-n_0}G(z)$
Multiplication $a^n u(n)$	$a^n g(n)$	$G(z/a)$
Convolution	$g(n) \otimes h(n)$	$G(z)H(z)$



III.5.5 Concept de délai

La notion de délai temporel est importante dans les systèmes numériques. L'introduction volontaire de délai permet d'utiliser les états précédents de l'entrée. Celle-ci est réalisée facilement à l'aide d'une porte logique qui se traduit dans le domaine temporel par un retard $n - 1$:

Dans le domaine en z , ce délai se traduit de la manière suivante :

$$Y(z) = z^{-1}X(z)$$

. Sa notation est variable dans le sens où il s'écrit sous la forme de l'opérateur z^{-1} ou T .

III.5.6 Transformée en z rationnelle

Dans le cadre des systèmes linéaire et invariant (LTI) c'est l'attribut principal de la transformée en z , les systèmes à temps discrets sont des fonctions rationnelles en z^{-1} :

$$H(z) = \frac{N(z)}{D(z)} = \frac{b_0 + b_1z^{-1} + \dots + b_{M-1}z^{M-1} + b_Mz^{-M}}{a_0 + a_1z^{-1} + \dots + a_{N-1}z^{N-1} + a_Nz^{-N}} \quad (\text{III.10})$$

où le degré du numérateur $N(z)$ et $D(z)$ et M est le degré du dénominateur est N . Le degré de la fonction rationnelle $H(z)$ est le maximum du degré du numérateur $N(z)$ ou du dénominateur $D(z)$.

La fonction $H(z)$ peut être factorisée en utilisant les pôles (p_i) et les zéros (λ_i):

$$H(z) = \frac{p_0 \prod_{i=1}^M (1 - p_i z^{-1})}{d_0 \prod_{i=1}^N (1 - \lambda_i z^{-1})} \quad (\text{III.11})$$

III.5.7 Fonction de transfert d'un système numérique

Un système numérique est stable si le module de l'ensemble des pôles sont inclus dans le cercle unité $p_i \leq |1|$. Un système numérique est souvent décrit par une équation récursive définie pour des échantillons discrets:

$$y(n) + \sum_{k=1}^N a_k y(n-k) = \sum_{k=0}^N b_k x(n-k) \quad (\text{III.12})$$

En utilisant les différentes propriétés (linéarité, décalage temporel) de la transformée en z , l'équation aux différences finies est:

$$y(n) + \sum_{k=1}^N a_k y(n-k) = \sum_{k=0}^N b_k x(n-k) \quad (\text{III.13})$$

III.6 Fonctions Octave&Matlab utiles

Différentes fonctions disponibles sous Octave ou Matlab permettent d'analyser les différents systèmes numériques :

- Il est possible en définissant les numérateurs et dénominateurs sous la forme de tableau, de définir la fonction de transfert d'un système numérique à l'aide de la fonction : `tf`. Contrairement au système analogique, il est nécessaire de préciser la fréquence d'échantillonnage F_e .

```
A=[1 1];B=[1 1 1];tf(A,B,1/Fe);
```

- Pour des fonctions de transfert complexe, le calcul des pôles et zéros à la main peut s'avérer fastidieux. Il est possible d'estimer numériquement la stabilité des pôles : `TF2ZP,ZPLANE`.

```
[zero,pole]=tf2zp(A,B);zplane(A,B);
```

- Réponse impulsionnelle et fréquentielle `IMPULSE` et `FREQZ`

```
[y,t]=impulse(H,1,10,100);
```

```
[G2,f]=freqz(A2,B,500,Fe);
```

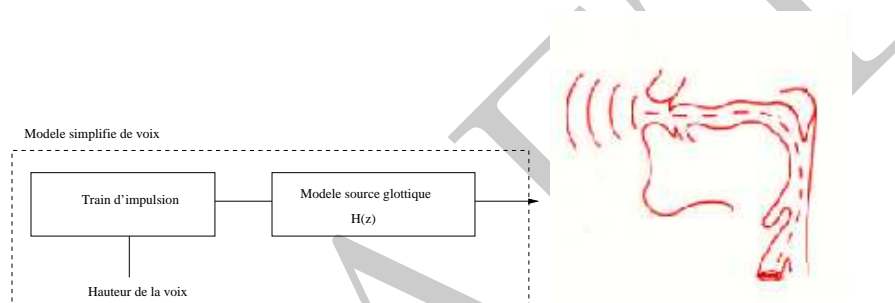
```
plot(t,y);plot(f,G2)
```

Tutoriel : transformée en z et modèle de vocodeur

Le mécanisme phonatoire physique est composé grossièrement de trois parties:

- L'appareil respiratoire génère le souffle d'air.
- Le larynx où se trouve les cordes vocales transforme le souffle en un son glottique.
- Les organes de la cavité bucco-pharyngale mettent en forme le son de la parole.

Il peut être modélisé sous une forme numérique très simplifiée permettant la génération de parole synthétique (vocodeur) schématisée par le diagramme suivant:



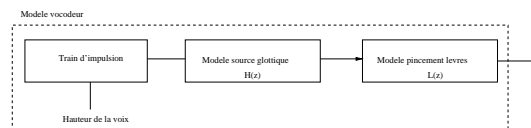
Le train d'impulsion correspond au souffle d'air de l'appareil respiratoire et génère une séquence périodique correspondant au rythme de la partie voisée de la voix, la génération de l'onde acoustique par le larynx étant modélisé par un système discret $H(z)$:

$$H(z) = \frac{\alpha z^{-1}}{(1 - \alpha z^{-1})^2}, \quad 0 < \alpha < 1 \quad (\text{III.14})$$

Le nombre de cycle normal respiratoire est estimée entre 12-20 cycles par minute pour l'adulte.

1. Définir une fréquence d'échantillonnage réaliste correspondant au système vocal.
2. Le système discret $H(z)$ est-il stable ?
3. En utilisant Octave ou Matlab, estimer les pôles et les zéros de la fonction de transfert $H(z)$.
4. En considérant le signal numérique discret $X(z)$, donner la sortie du système numérique $Y(z)$.
5. Exprimer l'équation aux différences finies gouvernant les échantillons temporels de sortie $y(n)$.
6. En utilisant la table de transformée en z , exprimer l'expression de la fonction de transfert dans le domaine temporel en utilisant l'inverse de la transformée en z .

L'influence du pincement des lèvres sur la génération de la parole peut être modélisé par la fonction de transfert $L(z) = (1 - z^{-1})$



7. Quelle est la fonction numérique implémentée par la fonction $L(z)$?
8. Estimer et tracer à l'aide d'Octave ou de Matlab, la réponse impulsionnelle de l'ensemble du modèle simple de vocodeur.
9. Estimer et tracer à l'aide d' Octave ou de Matlab, la réponse en fréquence du modèle simple de vocodeur.

Si plusieurs impulsions correspondant aux respirations sont considérées, existe-t-il un accord entre le modèle de vocodeur et l'exemple de parole voicée extrait de la littérature ?



IV – Introduction aux filtres numériques

IV.1 Introduction

L'opération de filtrage est un dispositif permettant de filtrer un signal d'entrée de manière à générer un signal de sortie dont une partie du spectre est filtrée. Initialement, les filtres ont été implémentés de manière analogique par des composants électroniques passifs et des amplificateurs. Ils présentent néanmoins les inconvénients suivants:

- La fréquence de la bande passante est limitée
- La sensibilité au bruit est importante ainsi que la présence de non-linéarité
- La stabilité et la flexibilité sont limitées

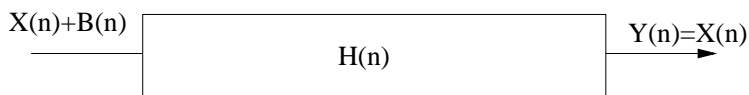
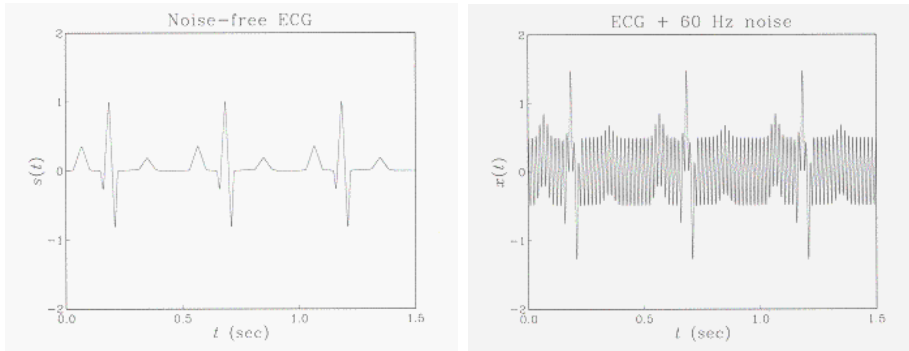
Les implémentations des filtres numériques sur le plan technologique ont souvent été préférées parce qu'il est communément admis comme étant plus fiable, stable et adaptable. Par contre, les implémentations analogiques demeurent en terme de coût ou pour des gammes de fréquences élevées moins fiables.

IV.2 Principes d'un filtre numérique

IV.2.1 Pourquoi filtrer ?

L'exemple suivant illustre le fait qu'un signal extrait d'un dispositif médical comme un électrocardiogramme (ECG) peut-être perturbé par la présence d'un bruit:

Un filtre est nécessaire afin d'extraire le signal utile du bruit.



IV.2.2 Définition

Un filtre numérique est un système permettant sur le plan fonctionnel de transformer un signal numérique $X(n)$ à l'entrée en un signal $Y(n)$ en enlevant le bruit $B(n)$ éSENT 'a l'entrée et éventuellement une partie du spectre non désiré:

Un filtre numérique respecte le principe de causalité dans le sens où la sortie du filtre n'est pas définie et est considérée nulle pour des instants négatifs $Y(n) = 0 \quad \forall n \leq 0$.

IV.2.3 Filtre ARMA

En pratique, le filtre appartient à la catégorie des systèmes numériques, il est défini par une équation aux différences finies d'ordre N avec les coefficients constants a_k et b_k :

$$y(n) = \sum_{k=0}^M b_k x(n-k) - \sum_{k=1}^N a_k y(n-k) \quad (\text{IV.1})$$

La première partie de l'équation du filtre IV.1 correspond aux coefficients $b_k x(n-k)$ est une moyenne mobile (MA) des échantillons de l'entrée tandis que la seconde partie est auto-regressive $\sum_{k=1}^N a_k y(n-k)$ de l'équation (AR). Cette structure est souvent qualifiée de structure de filtre ARMA et est un modèle dit à "autoregressif à moyenne adaptée". Cette structure de filtre permet d'implémenter des filtres RII. L'hypothèse faite est de poser une différence au lieu d'une addition au niveau de l'équation aux différences finies permettent d'obtenir une expression plus simple au niveau des fonctions de transfert par contre il faut faire attention au signe des coefficients obtenus par les méthodes de conception qui peuvent adopter une autre définition au niveau du signe.

IV.2.4 Fonction de transfert

A l'aide de la transformée en z , il est possible de transformer les retards sous la forme d'un opérateur z^{-1} et de mettre l'équation aux différences finies sous la forme d'une fonction de transfert $H(z)$ définie de la manière suivante :

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^M b_k z^{-k}}{1 + \sum_{k=1}^N a_k z^{-k}} \quad (\text{IV.2})$$

Au niveau d'un filtre numérique, sa réponse en fréquence est définie par les coefficients a_k , b_k et tout le problème va consister à définir ces coefficients de manière à avoir la réponse du filtre désirée.

IV.3 Catégorie de filtres numériques

Il existe principalement deux catégories différentes de filtres numériques à savoir des filtres dits RIF ou RII. Les filtres non-récurrents RIF sont caractérisés par :

- Un filtre RIF est un filtre non récurrent avec une réponse impulsionnelle infinie

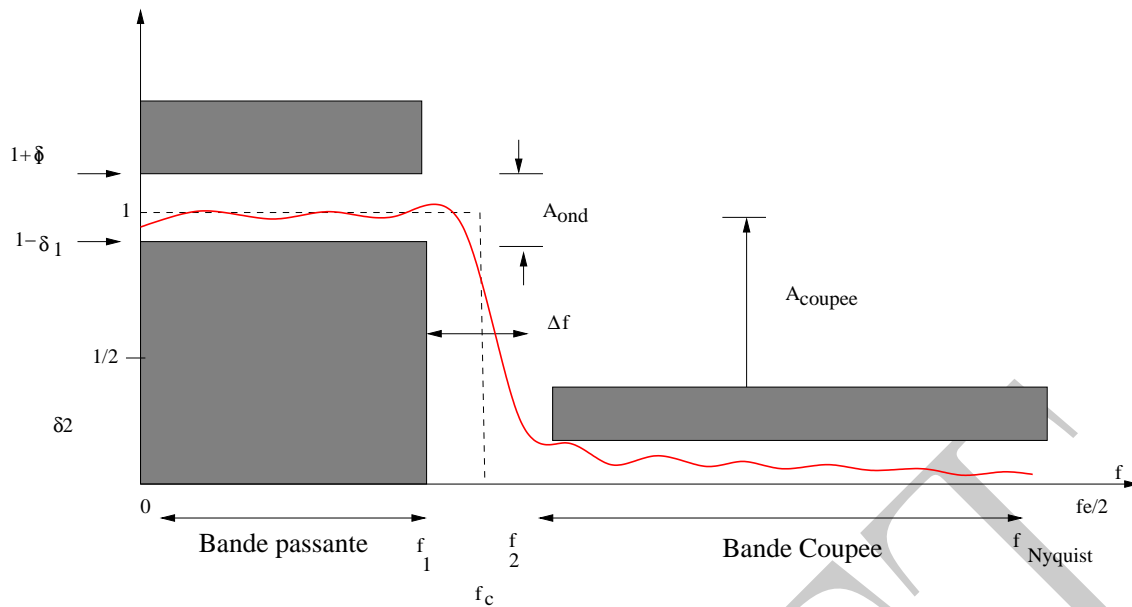
$$y(n) = \sum_{k=0}^M b_k x(n-k) \quad (\text{IV.3})$$

Le filtre *RIF* est caractérisé par une relation directe entre les échantillons de la séquence numérique de l'entrée $x(n)$ et ceux de la sortie $y(n)$. Il est caractérisé par une réponse impulsionnelle finie dans le sens où l'entrée d'un signal discret va provoquer une réponse en sortie qui est fonction des coefficients b_k .

- Un filtre RII est un filtre récurrent avec une réponse impulsionnelle infinie:

$$y(n) = \sum_{k=0}^M b_k x(n-k) - \sum_{k=1}^N a_k y(n-k) \quad (\text{IV.4})$$

Le filtre *RII* est caractérisé par une relation entre les échantillons de l'entrée et des états précédents de la sortie.



IV.4 Spécification des filtres numériques

Un filtre numérique est défini par un gabarit. Ce dernier est souvent précisé par un cahier des charges en fonction de l'application et des différentes spécificités.

IV.4.1 Gabarit d'un filtre passe-bas

La figure ci-dessous montre la définition d'un exemple de filtre passe-bas défini par la présence de deux régions bien spécifiques. La première correspond à la région où la réponse idéale du filtre est caractérisée par un gain unitaire dans la bande passante et un gain en théorie nul dans la bande coupée. La région du gabarit entre la partie passante et la partie coupée est caractérisée par une largeur de la zone de transition Δf . La réponse idéale du filtre correspond à la courbe en pointillée.

IV.4.2 Gabarit du filtre

En pratique, il n'est pas possible d'implémenter cette réponse ce filtre idéal, sa réponse impulsionnelle n'étant malheureusement pas causale. Il est nécessaire d'autoriser de l'ondulation en bande passante définie par le paramètre δ_1 dans la bande passante au niveau du gain unitaire et la variation tolérée se situe entre $1 + \delta_1$ et $1 - \delta_1$. La transition entre la région de gain unité et celle de gain nul ne peut être immédiate et en pratique, il est nécessaire de définir la largeur fréquentielle de la zone de transition:

Ce paramètre est appelé $\Delta f = f_2 - f_1$.

La raideur de la pente de transition (raideur de coupure) entre la fréquence de bande passante f_1 et f_2 permet de définir la selectivité k du filtre:

$$k = f_1/f_2 \quad (\text{IV.5})$$

Le temps de propagation de groupe $\tau(\omega)$ est défini comme la dérivée de la phase en fonction de la pulsation:

$$\tau(\omega) = -\frac{\delta\phi(\omega)}{\delta\omega} \quad (\text{IV.6})$$

IV.4.3 Réponse en fréquence d'un filtre

La réponse en fréquence $H(\nu)$ d'un filtre est définie par le système d'équations suivant:

$$\begin{cases} 1 - \delta_1 \leq |H(\nu)| \leq 1 + \delta_1 & \forall \nu \in [0; \nu_1] \\ 0 \leq |H(\nu)| \leq \delta_2 & \forall \nu \in [\nu_2; +\infty] \end{cases} \quad (\text{IV.7})$$

o les différents paramètres peuvent être également exprimés en dB. L'ondulation A_{ond} en bande passante vaut:

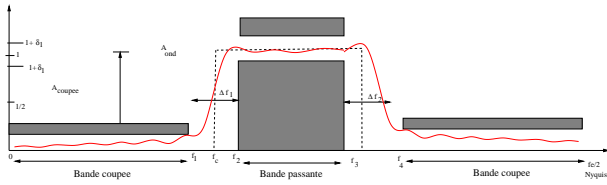
$$A_{ond} = 20 \log\left(\frac{1 + \delta_1}{1 - \delta_1}\right) \text{ en dB} \quad (\text{IV.8})$$

Dans le cas de l'atténuation en bande coupée A_{coupee} , celle-ci vaut:

$$A_{coupee} = 0 - 20 \log(\delta_2) \quad (\text{IV.9})$$

$$A_{coupee} = -20 \log(\delta_2) \quad (\text{IV.10})$$

Il est possible d'étendre la définition d'un gabarit de filtre passe-bas afin de définir le gabarit d'un filtre passe-bande. Ce filtre est caractérisé par deux bandes coupées à la fois pour les basses et les hautes fréquences. Ce filtre va couper les fréquences de $[0 - f_1]$ et de $[f_4 - f_{Nyquist}]$ et par contre va conserver la gamme de fréquences entre $[f_3 - f_4]$.



IV.4.4 Caractérisation simple d'un filtre numérique

IV.4.4 Réponse impulsionnelle

La caractérisation primaire d'un filtre numérique se fait par sa réponse impulsionnelle c'est-à-dire par sa réponse à une réponse impulsionnelle:

$$X(n) = \delta(n) = \begin{cases} X(0) = 1 \\ X(n \neq 0) = 0 \end{cases} \quad (\text{IV.11})$$

Cette réponse impulsionnelle $h(n)$ peut être utilisée pour calculer d'autres réponses en utilisant la propriété de convolution. En effet,

$$y(n) = (h * x)(n) = \sum_{-\infty}^{+\infty} h(k-n)x(n) \quad (\text{IV.12})$$

IV.4.4 Réponse fréquentielle

La réponse fréquentielle d'un filtre est donnée par la transformée de Fourier discrète (TFD) de la réponse impulsionnelle:

$$H(\nu) = \sum_{n=-\infty}^{+\infty} h(n)e^{-2i\pi\nu nT_e} \quad (\text{IV.13})$$

où plus simplement en utilisant la fonction de transfert en z :

$$H(\nu) = H(z = e^{i\omega} \text{ avec } \omega = \frac{2\pi\nu}{\nu_e}) \quad (\text{IV.14})$$

Cette réponse peut être complexe dans le sens où la réponse du filtre est un nombre complexe caractérisé par un module $H(\nu)$ et une phase $\phi(\nu)$.

IV.5 Exercice

Exercice: un filtre est défini par la fonction de transfert suivante

$$H(z) = 0.25 + 0.05z^{-1} + z^{-2} + 0.05z^{-3} + 0.25z^{-4} \quad (\text{IV.15})$$

1. Déterminer l'équation aux différences
2. S'agit-il d'un filtre FIR ou IIR ?
3. Donner la réponse impulsionnelle
4. Evaluer la stabilité du filtre
5. Déterminer la réponse fréquentielle
6. Déterminer le module et la phase du filtre
7. Dessiner la réponse en fréquence

Tutoriel : filtrage et notion de phase linéaire

Considérons la fonction de transfert analogique suivante du premier ordre :

$$H(s) = \frac{1}{s + 2} \quad (\text{IV.16})$$

1. Calculer la réponse impulsionnelle du filtre analogique.
2. Exprimer la réponse cette fois-ci en fréquence d'un filtre analogique. Est-il possible de définir une fréquence de coupure à -3 dB ?

Le principal objectif de ce tutoriel est d'illustrer le fait que la réponse c'est à dire le gain du filtre dans le domaine fréquentiel ait une large importance mais que la phase ne doit pas être négligée. En particulier, un filtre avec une phase non-linéaire peut induire des distorsions importantes en filtrant le signal.

3. La phase du filtre $H(s)$ analogique est-elle linéaire ?
4. Estimer la sortie du filtre en considérant la forme du signal suivant:

$$x(t) = A_1 \cos(2\pi f_1 t) + A_2 \cos(2\pi f_2 t) \quad (\text{IV.17})$$

En utilisant le fait que :

$$G(\nu)\delta(\nu - \nu_0) = G(\nu_0)\delta(\nu - \nu_0) \quad (\text{IV.18})$$

montrer que le signal de sortie $Y(\nu)$ est égal à :

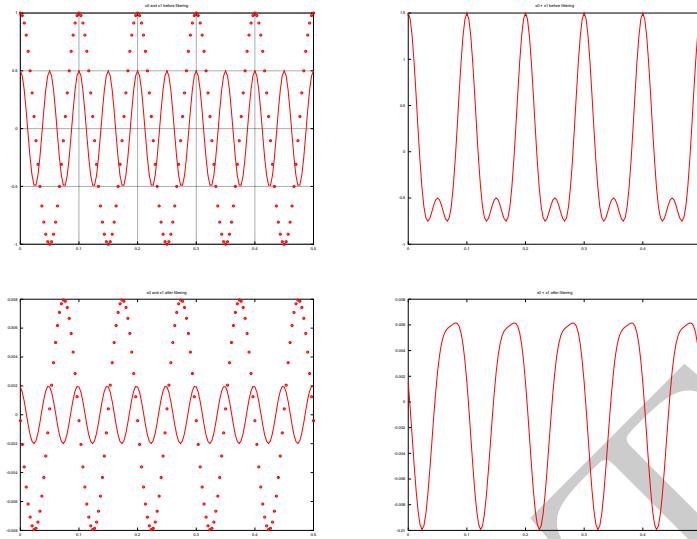
$$Y(\nu) = \sum_{i=1}^2 \frac{A_i}{2\sqrt{(1 + \pi^2 f_i^2)}} [e^{(-i \arctan(\pi f_i))} \delta(\nu - f_i) + e^{(i \arctan(\pi f_i))} \delta(\nu + f_i)] \quad (\text{IV.19})$$

5. En rappelant la propriété du délai au niveau de la transformée de Fourier :

$$\int_{-\infty}^{+\infty} f(t - t_r) e^{(-2i\pi\nu t)} dt = F(\nu) e^{(-2i\pi\nu t_r)} \quad (\text{IV.20})$$

avec $F(\nu)$ la transformée de Fourier de $f(t)$. Démontrer que le signal $y(t)$ est égal à :

$$y(t) = \sum_{i=1}^2 \frac{A_i}{2\sqrt{(1 + 4\pi^2 f_i^2)}} \cos(2\pi f_i(t - t_r(f_i))) \quad (\text{IV.21})$$



Exemple: les figures suivantes illustrent l'influence de la réponse d'un filtre à phase non-linéaire pour un signal comportant deux sinusoides légèrement décalées au niveau temporel.

DRAFT

V – Filtre à réponse impulsionnelle finie (RIF)

V.1 Introduction

V.1.1 Aspect historique



Ronald Schafer est un ingénieur en génie électrique qui a contribué de manière significative au traitement du signal numérique. Titulaire d'un doctorat au MIT en 1968, il a mené l'ensemble de ces recherches au département d'acoustique des laboratoires Bell sur le traitement du signal et l'encodage de la voix. Il a rejoint Hewlett Packard en mars 2005.



James F. Kaiser est un professeur de l'école d'ingénieur de Duke. Sa contribution principale est l'invention de la fenêtre de Kaiser pour de l'analyse spectrale en 1980.

V.2 Définition des filtres numériques

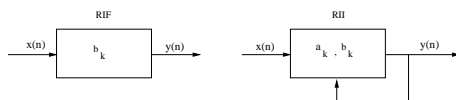
Il existe deux catégories de filtre numérique différentes:

- RIF: filtre non récursif avec une réponse impulsionnelle finie

$$y(n) = \sum_{k=0}^M b_k x(n-k) \quad (\text{V.1})$$

- RII: filtre non-récursif avec une réponse impulsionnelle infinie

$$y(n) = \sum_{k=0}^M b_k x(n-k) - \sum_{k=1}^N a_k y(n-k) \quad (\text{V.2})$$



La méthode de conception afin de générer un filtre (RIF ou RII) est différent selon qu'ils soient à réponse impulsionnelle finie ou infinie.

V.3 Propriétés des filtres RIF

V.3.1 Phase linéaire et réponse impulsionnelle

Dans le cadre d'applications sonore, les filtres utilisés doivent être à phase linéaire de manière à ne pas introduire de distorsion. C'est-à-dire, si la réponse en fréquence d'un filtre est définie comme étant le produit $|H(\nu)|\phi(\nu)$, la phase doit donc être linéaire en fonction de la fréquence ν :

$$\phi(\nu) = \alpha\nu + \beta \quad (\text{V.3})$$

Un filtre RIF est à phase linéaire si et seulement si sa réponse impulsionnelle est symétrique ou anti-symétrique.

V.3.2 Catégories de filtre RIF

Type	I	II	III	IV
Ordre	Pair	Impair	Pair	Impair
$h(n)$	symétrique	symétrique	anti-symétrique	anti-symétrique
$ H(\nu) $	symétrique	symétrique	anti-symétrique	anti-symétrique
$ H(0) $	$\sum_{n=0}^M h(n)$	$\sum_{n=0}^M h(n)$	0	0
$ H(\frac{\nu_c}{2}) $	arbitraire	0	0	arbitraire
Applications	Passe-bas Passe-haut Passe-bande Coupé	Passe-bas Passe-bande Derivé	Dérivé Derivé	Passe-haut

V.4 Troncature de la réponse impulsionnelle

V.4.1 Méthode de conception globale

La méthode de conception globale d'un filtre est la suivante:

1. Déterminer les spécifications du gabarit du filtre.

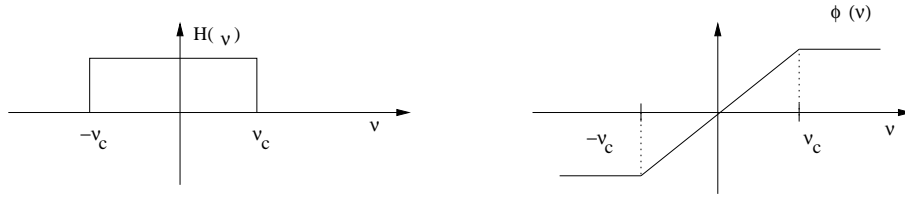


Figure V.1: Réponse d'un filtre idéal passe-bas de phase linéaire de fréquence ν_c . Le gain vaut un dans la gamme $([-\nu_c, +\nu_c])$ et la variation de phase est également linéaire avec la fréquence.

2. Déterminer les coefficients de la fonction de transfert a_k et b_k .
3. Implémenter la structure du filtre qui permet d'organiser les calculs.
4. Estimer les effets d'encodage.
5. Implémentation du filtre: matériel ou logiciel.

V.4.2 Méthode de la troncature de la réponse impulsionnelle

V.4.2 Modèle de filtre à phase linéaire

De manière à concevoir un filtre numérique RIF, il est nécessaire de partir d'un modèle de filtre à phase linéaire. Cette réponse de filtre idéale est définie de la manière suivante à savoir. Le gain est unitaire $|H(\nu)|$ dans la bande coupée $[-\nu_c; \nu_c]$ et à phase linéaire $\phi(\nu)$:

$$\begin{cases} |H(\nu)| = 1 & \forall \nu \in [-\nu_c; \nu_c] \\ \phi(\nu) = \exp(-2i\pi\nu T_e) & \forall \nu \in [-\nu_c; \nu_c] \end{cases} \quad (\text{V.4})$$

La réponse impulsionnelle du filtre peut être estimée en prenant la transformée de Fourier inverse de la réponse en fréquence désirée:

$$h(n) = \frac{1}{\nu_e} \int_{-\infty}^{\infty} H(\nu) \exp^{2i\pi\nu n T_e} (\text{V.5})$$

$$h(n) = \frac{1}{\nu_e} \int_{-\nu_c}^{\nu_c} H(\nu) \exp^{2i\pi\nu(n-k)T_e} (\text{V.6})$$

Ce qui aboutit à la réponse impulsionnelle idéale d'un filtre passe-bas en sinus cardinal, en définissant la pulsation numérique ω_c :

$$h(n) = \frac{\sin \omega_c(n-k)}{\pi(n-k)} \quad (\text{V.7})$$

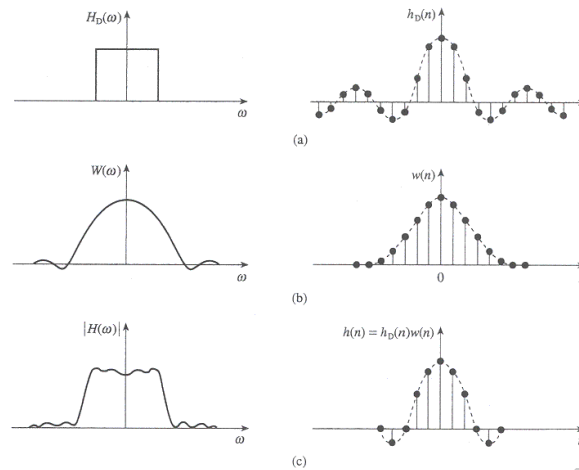


Figure V.2: Conception par fenêtrage. Des fenêtres idfférentes de la fenêtre rectangulaire peuvent être utilisées notamment celle cosinusoidale afin de limiter les oscillations de la réponse en *sinc*.

Cette réponse impulsionnelle est non-causale, infinie et symétrique par rapport à k . Il est nécessaire de la fenêtrer, c'est-à-dire de limiter le nombre d'échantillons.

V.4.3 Fenêtrage de réponse

Un fenêtrage de la réponse impulsionnelle idéale est réalisée de manière à obtenir une réponse finie et causale par une fenêtre rectangulaire:

$$\begin{cases} h_r(n) = h(n) & n \in \{0, 1, 2, \dots, 2k\} \\ h_r = 0 & \text{autre} \end{cases} \quad (\text{V.8})$$

c'est-à-dire que la réponse impulsionnelle réelle h_r est donnée par le produit de $h(n)$ par la fenêtre:

$$\boxed{h_r = h(n) \cdot w(n)} \quad (\text{V.9})$$

La réponse en fréquence est donnée par l'équation suivante:

$$H_r(\nu) = H(\nu) \otimes R(\nu) \quad (\text{V.10})$$

où $R(\nu)$ est le spectre d'une fenêtre rectangulaire.

Fenêtre	No	Expression mathématique
Rectangulaire	1	$w(n) = 1$
Barlett	2	$w(n) = 1 - \frac{ n }{M+1}$
Hann	3	$w(n) = \frac{1}{2}(1 + \cos(\frac{\pi n}{M}))$
Hamming	4	$w(n) = 0.54 + 0.46 \cos(\frac{\pi n}{M})$
Blackmann	5	$w(n) = 0.42 + 0.5 \cos(\frac{\pi n}{M}) + 0.08 \cos(\frac{2\pi n}{M})$
Kaiser	6	$w(n) = \frac{\beta \sqrt{1 - (\frac{n}{M})^2}}{I_0(\beta)}$

Table V.1: Expression mathématique des différentes fenêtres

V.4.4 Influence de fenêtre rectangulaire

V.4.5 Choix de la fenêtre de conception

Il est possible de choisir d'autres fenêtres de manière à obtenir des réponses en fréquences différentes et ainsi de diminuer les lobes secondaires. Le choix de la fenêtre du filtre est principalement gouverné par l'atténuation en bande coupée souhaitée (A_{coupee}). Le tableau ci-dessous illustre les caractéristiques attendues du filtre au niveau de l'atténuation en bande coupée (A_{coupee}) et de l'ondulation en bande passante. Les expressions mathématiques des différentes fenêtres possible sont les suivantes où $-M \leq n \leq M$:

Contrairement aux autres fenêtres où les performances attendues en terme d'atténuation ou de largeur de transition sont fixes, la fenêtre de Kaiser (β) permet d'avoir des performances variables à l'aide d'un paramètre additionnel entrant au niveau de la définition de la fenêtre.

V.4.6 Réponse impulsionnelle idéale des différents filtres

L'expression des différentes réponses impulsionnelles idéales sont nécessaire afin de définir pouvoir définir le filtre:

Fenêtre	δ	A_{coupee}	A_{ond}	Largeur du lobe principal
Rectangulaire	8.9 %	20.9 dB	1.55 dB	0
Bartlett		43.9 dB		
Hann		54.5 dB	0.03 dB	3.21
Hamming	0.2 %	75.3 dB		
Blackman				
Kaiser	variable	$-20 \log(\delta)$	17.372δ	$\frac{A-7.95}{14.36}$

Table V.2: Propriétés principales en terme d'atténuation et d'ondulation des fenêtres des filtres numériques.

- filtre passe-bas: $h(n) = \frac{\sin \omega_c(n-k)}{\pi(n-k)}$
- filtre passe-haut: $h(n) = \delta(n-k) - \frac{\sin \omega_c(n-k)}{\pi(n-k)}$
- filtre passe-bande: $h(n) = \frac{\sin \omega_b(n-k)}{\pi(n-k)} - \frac{\sin \omega_a(n-k)}{\pi(n-k)}$
- filtre rejeteur: $h(n) = \delta(n-k) - \frac{\sin \omega_b(n-k)}{\pi(n-k)} - \frac{\sin \omega_a(n-k)}{\pi(n-k)}$ $n \in \{0, 1, \dots, 2k\}$

V.4.7 Méthode de conception de filtres basés sur Octave&Matlab

Comme nous l'avons vu, la méthode de conception d'un filtre standard est la troncature de la réponse impulsionnelle, où la réponse impulsionnelle d'un filtre est fenêtrée en fonction des performances désirées :

1. fir1.m: méthode basée sur le fenêtrage pour des filtres passe-bas, passe-haut, passe-bande et bande coupée

Il existe également des méthodes permettant d'optimiser la réponse fréquentielle d'un filtre:

1. fir2.m: Conception de filtre FIR fenêtrage optimale
2. remez.m: Parks-McClellan algorithme de minimisation

Il est possible de définir les fenêtres suivantes:

1. Hamming.m: fenêtre de Hamming
2. Hanning.m: fenêtre de Hanning
3. Kaiser.m: fenêtre de Kaiser
4. Triang.m: fenêtre Triangulaire

V.4.8 Exemple implémentation filtre RIF passe-bas

```
%Paramètres spécifiques (fréquence de coupure normalisée)
%
Fc=1000;Fe=10000;N=10;Wstop=Fc/(Fe/2);
%
%Fenêtre rectangulaire (Filtre Ordre N et correspond à N+1 coefficients)
w=window('rectwin',N+1);
%
%Conception d'un filtre passe-bas de N coefficients
B=fir1(N,Wstop,w);
stem(B);
```

V.4.9 Exercice: Séparation des basses et hautes fréquences

La norme CD considère une fréquence d'échantillonnage de 44.1 kHz, une atténuation en bande coupée importante de 65 dB et une largeur de transition de 2 kHz. Concevoir un filtre permettant de séparer la partie à haute fréquence ("high-pitched") de celle à basse fréquence c'est à dire $[0 - 2\text{kHz}]$. Proposer une implémentation efficace du filtre.

Pour cela, il sera nécessaire de se poser les questions suivantes:

Écrire la réponse impulsionnelle d'un filtre passe-bas. Le choix de la fenêtre est importante pour la définition des caractéristiques du filtre.

Quelle fenêtre peut-on utiliser si l'on réfère au tableau une telle atténuation en bande coupée importante. Justifier

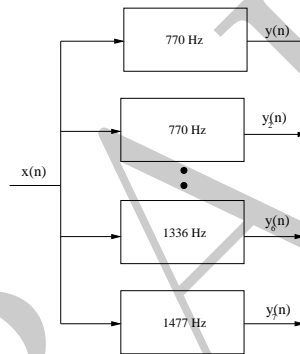
Calculer les coefficients du filtre RIF. il s'agit de multiplier les coefficients de la réponse impulsionnelle avec la fenêtre. Dans un premier temps calculer avec une simple fenêtre rectangulaire. Il est possible éventuellement chercher à calculer les coefficients à l'aide d'octave ou de matlab pour des fenêtres plus compliquées comme dans l'exemple précédent. Il est par contre nécessaire d'installer la boîte outil (toolbox en anglais) traitement du signal notamment pour le calcul des fenêtres.

Tutoriel : conception d'un filtre à réponse impulsionnelle finie (RIF)

Un clavier téléphonique permet de générer des signaux sinusoidaux à deux tons afin de transmettre les informations permettant l'établissement d'une communication téléphonique. Le tableau suivant résume l'encodage réalisé au niveau des touches du clavier téléphonique :

Fréquences	1209 Hz	1336 Hz	1477 Hz
697 Hz	1	2	3
770 Hz	4	5	6
852 Hz	7	8	9
941 Hz	*	0	#

De manière à décoder un signal DMTF, un ensemble de filtres à réponse impulsionnelle finie (RIF) est utilisé:



L'objectif principal de ce tutoriel est de concevoir un filtre RIF avec les spécifications techniques suivantes: séparation de la fréquence la plus basse (697 Hz) par rapport aux autres plus élevées (770-1477 Hz). La fréquence d'échantillonnage utilisée est de 8 kHz et une ondulation en bande passante maximale de 3 dB est considérée. Pour la conception de filtre RIF, à l'aide d'Octave ou de Matlab, il est nécessaire de considérer des fréquences normalisées c'est-à-dire que la fréquence de Nyquist normalisée $f_{Nyquist}$ est égale à un.

1. Définir le gabarit du filtre et normaliser les différents paramètres:

- Quel est le type de filtre utilisé (passe-bas, passe-haut, passe-bande, bande coupée: lowpass/highpass/bandpass/bandstop)?
- Quelle est la fréquence d'échantillonnage $\nu_e = \text{Hz}$, au niveau Octave ou Matlab, celle-ci correspond dans le système normalisé à deux.
- La fréquence de bande passante $\nu_1 = \text{Hz}$, $\Omega_1 = []$, $(\Omega_1/2 = (\nu_1/\nu_e))$
- La fréquence de bande coupée $\nu_2 = \text{Hz}$, $\Omega_2 = []$, $(\Omega_2/2 = (\nu_2/\nu_e))$
- L'ondulation maximale en bande passante en dB vaut $\alpha_{max} = \text{dB}$ $R_p = []$.

- L'atténuation minimale $\alpha_s =$ en dB vaut $R_s = []$.
2. Déterminer le nombre minimum de coefficients nécessaire au niveau de la méthode de la troncature de la réponse impulsionnelle.
 3. Quelle fenêtre est la mieux adaptée afin de définir le gabarit du filtre? Commenter votre choix.
En utilisant Octave ou Matlab:
 4. Représenter les échantillons de la fenêtre.
 5. Calculer les premiers échantillons de la réponse impulsionnelle du filtre.
 6. Comparer les coefficients du filtre RIF en utilisant la fonction `fir1.m`
 7. Exprimer l'équation aux différences finies qui est implémentée au niveau du processeur de calcul de traitement du signal.
 8. Dessiner la réponse fréquentielle du filtre numérique et comparer avec les spécifications initiales du gabarit désiré. Commenter les résultats obtenus.

DRAFT

VI – Filtre à réponse impulsionnelle infinie (RII)

VI.1 Introduction

VI.1.1 Aspect historique

La transformée bilinéaire est la découverte d' Arnold Tustin (1899-1994). Il a été un ingénieur britannique et professeur d'ingénierie à l'université de Birmingham et à l'Impérial College. Il a également contribué à l'automatique et à ses applications en terme de machine électrique.

VI.2 Transformée bilinéaire

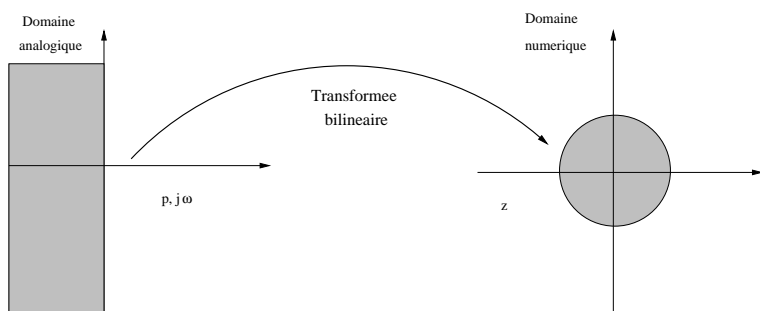
VI.2.1 De l'analogique au numérique ?

Un problème classique est de transformer, remplacer un système analogique en un système numérique caractérisé par des propriétés identiques.

- De nombreuses méthodes ont été étudiées et sont basées sur une fonction de transfert.
- Dans de nombreuses applications, un filtre numérique est utilisé afin de reproduire le comportement similaire d'un filtre analogique.

VI.2.2 La transformée bilinéaire

Le problème est de trouver une transformation de manière à transformer un filtre analogique stable $H_a(s)$ en un filtre numérique également stable $G(z)$. Cette transformée est la suivante:



$$T = 2\nu_e \frac{1 - z^{-1}}{1 + z^{-1}} \quad (\text{VI.1})$$

Celle-ci correspond à une bijection entre le domaine complexe en s où $(\Re(s) < 0)$ (Laplace) et le plan complexe en z où $|z| < 1$. Par contre, il en existe différentes définitions toutefois très proche. Cette dernière considérant la fréquence d'échantillonnage ν_e est compatible avec le logiciel Matlab.

VI.2.3 Exemple

Considérons que le filtre analogique passe-bas $H_a(s)$ est le suivant:

$$H_a(s) = \frac{1}{(2 + 2s)} \quad (\text{VI.2})$$

La transformée bilinéaire permet de transformer l'expression de la fonction de transfert du filtre analogique en un filtre numérique. Le filtre numérique de la fonction de transfert $G(z)$ et de fréquence d'échantillonnage ν_e est le suivant:

$$G(z) = \frac{1}{2 + 4\nu_e \frac{1 - z^{-1}}{1 + z^{-1}}} \quad (\text{VI.3})$$

En développant le numérateur, il est possible de simplifier la fonction de transfert du filtre et d'aboutir à l'expression si on considère un échantillonnage à une fréquence de 1 Hz:

$$G(z) = \frac{1 + z^{-1}}{6 - 2z^{-1}} \quad (\text{VI.4})$$

VI.3 Gabarit du filtre analogique équivalent

Les spécifications d'un filtre numérique sont souvent donnés non pas sous forme de fonction de transfert mais plutôt en terme de contraintes au niveau du gabarit du filtre :

$$|H_{numerique}(\nu)| = |H_{numerique}(z = \exp(i\omega))| \quad (\text{VI.5})$$

L'une des problématiques est donc de trouver le lien existant entre le gain du filtre numérique et son équivalent analogique. Comme nous l'avons vu précédemment, la relation existante entre le domaine analogique s et celui numérique z est la suivante:

$$s = 2\nu_e \frac{1 - z^{-1}}{1 + z^{-1}} \quad (\text{VI.6})$$

Le passage dans le domaine fréquentiel se fait de la manière suivante en se positionnant sur le cercle unité $s = j\omega$ en analogique et $z = \exp \frac{j2\pi\nu}{\nu_e}$ en numérique:

$$j\omega = 2\nu_e \frac{1 - \exp(-j\omega T_e)}{1 + \exp(-j\omega T_e)} \quad (\text{VI.7})$$

En notant Ω , la pulsation numérique:

$$j\omega = 2\nu_e \frac{1 - \exp(-j\Omega)}{1 + \exp(-j\Omega)} \quad (\text{VI.8})$$

Si l'on divise le numérateur et le dénominateur par $\frac{j\Omega}{2}$, ceci amène à l'expression suivante:

$$j\omega = 2\nu_e \frac{\exp(j\Omega/2) - \exp(-j\Omega/2)}{\exp(j\Omega/2) + \exp(-j\Omega/2)} = 2\nu_e \frac{2j \sin(\Omega/2)}{2 \cos(\Omega/2)} \quad (\text{VI.9})$$

Ainsi, la relation entre la pulsation du domaine analogique équivalent (ω) et celle numérique (Ω) peut être déduite de la manière suivante :

$$\boxed{\omega = 2\nu_e \tan\left(\frac{\Omega}{2}\right)} \quad (\text{VI.10})$$

Pour des fréquences faibles, il est possible de simplifier cette relation qui est linéaire mais dans un régime de fréquences réduites $\omega = 2\nu_e \tan \frac{\pi\nu}{\nu_e} \sim 2\pi\nu$.

VI.4 Conception d'un filtre RII

La méthode standard pour concevoir un filtre RII à réponse impulsionnelle infinie est basée sur trois étapes différentes :

- Le gabarit du filtre analogique équivalent doit être estimé à partir du filtre numérique désiré. Il s'agit principalement de déterminer les pulsations de coupure équivalente).
- La conception du filtre analogique doit être réalisée $H(s)$ dans le domaine analogique. En pratique, des filtres modèles sont utilisés.
- Enfin, la transformée bilinéaire est utilisée pour repasser dans le domaine numérique:

$$H(z) = H(s) \Big|_{s = 2\nu_e \frac{1 - z^{-1}}{1 + z^{-1}}}$$

VI.5 Modèle de filtre analogique

De manière à concevoir des filtres à réponse impulsionnelle infinie, des gabarits de filtre analogique sont nécessaires. Ainsi, il peut-être utile d'utiliser les différents types de filtre en tenant compte que ces filtres sont caractérisés par des propriétés fort différentes. Les propriétés des filtres de Butterworth, Tchebychev ou elliptique sont présentées juste après.

VI.5.1 Filtre de Butterworth

La fonction de transfert d'un filtre de Butterworth est décrite par l'équation suivante:

$$|H_{analogique}(\Omega)|^2 = \frac{1}{(1 + \frac{\Omega}{\Omega_c})^2} \quad (\text{VI.11})$$

La réponse de ce filtre en fréquence est caractérisée par une réponse monotone dans la bande passante ou coupée. La zone de transition entre la bande passante et la bande coupée est importante et décroît en fonction de l'ordre du filtre.

VI.5.2 Filtre de Chebychev

La réponse d'un filtre de Tchebychev est définie à l'aide d'un polynôme de Chebychev:

$$|H_{analogique}(\Omega)|^2 = \frac{1}{1 + \epsilon^2 T_N^2\left(\frac{\Omega}{\Omega_p}\right)} \quad (\text{VI.12})$$

où T_N est l'ordre N du polynôme de Tchebychev :

$$T_0(x) = 1, T_1(x) = x, T_{N+1}(x) = 2xT_N(x) - T_{N-1}(x) \quad (\text{VI.13})$$

La réponse des filtre de Tchebychev en fréquence est caractérisée par une ondulation élevée en bande passante par contre celle-ci demeure monotone dans la bande de fréquence coupée.

VI.5.3 Filtre elliptique dit de Cauer

Les filtres de Cauer sont caractérisés par une ondulation à la fois dans la bande passante et coupée. La largeur de la région de transition entre les deux bandes est réduite.

$$|H_{analogique}(s)|^2 = \frac{1}{1 + \epsilon^2 R_n(s, L)} \quad (\text{VI.14})$$

où R_n est la fonction rationnelle de Tchebychev.

La fonction de transfert de ces filtres de Cauer est basée sur des fonctions rationnelles de Tchebychev.

VI.5.4 Exemple

Concevoir un filtre numérique RII basée sur une fonction de transfert d'un filtre de Butterworth d'ordre deux avec une fréquence de bande passante $\nu_1 = 3$ kHz et une fréquence d'échantillonnage $\nu_e = 10$ kHz:

- La pulsation numérique $\Omega_1 = 2\pi \frac{\nu_1}{\nu_e} = 2\pi \frac{3000}{10000} = 0.6\pi$ [rad]
- Calcul de la pulsation analogique équivalente $\omega_1 = 2\nu_e \tan\left(\frac{\Omega_1}{2}\right) = 27528$ [rad/s]
- Fonction de transfert de Butterworth normalisée : $H_{normalisee}(s) = \frac{1}{s^2 + 1.4142s + 1}$

- Filtre analogique passe-bas: $H(s) = H_{normalisee}(s) \Big|_{\frac{s}{\omega_1}} = \frac{7.5779e+08}{s^2+3.893e04s+7.5779e+08}$
- Fonction de transfert filtre RII par la transformée bilinéaire: $H(z) = \frac{0.3913z^2+0.7827z+0.3913}{z^2+0.3965z+0.1958}$

VI.6 Conception de filtre RII à l'aide de Matlab

La section précédente illustre le fait que la conception de filtre RII exige des calculs numériques qui peuvent devenir important pour des filtres d'ordre élevé. Il existe un certain nombre de fonctions prédéfinies permettant d'aider à la conception.

VI.6.1 Fonctions

- RII: conception de filtre analogique sur Matlab
 1. butter.m: définition de fonction de transfert de filtre Butterworth
 2. cheby1.m: définition de fonction de transfert de filtre de Chebyshev type I
 3. cheby2.m: définition de fonction de transfert de filtre de Chebyshev type II
 4. ellip.m: définition de fonction de transfert de filtre de Caue
 5. maxflat.m: filtre généralisé de Butterworth
- RII: estimation de l'ordre du filtre
 1. buttord.m: estimation ordre du filtre de Butterworth
 2. cheb1ord: estimation ordre du filtre de Chebyshev type I
 3. cheb2ord: estimation ordre du filtre de Chebyshev type II
- bilinear.m: implémentation de la transformée bilinéaire

VI.6.2 Exemple de conception de filtre

```
%D\`efinition de la frequence de coupure du filtre de frequence fc
[b,a]=butter(N,Fc,'low','s');
%
%Transformee bilineaire:
[bd,ad]=bilinear(b,a,Fe);
%
%Affichage de la fonction de transfert du filtre RII
tf(bd,ad,1/Fe);
```

VI.7 Implémentation pratique d'un filtre FIR ou RII

Une fois, les coefficients b_k et a_k déterminés, la question qui est posée est comment implémenter l'équation suivante correspondant à l'équation du filtre numérique ?

$$H(z) = \frac{P(z)}{D(z)} = \frac{b_0 + b_1z^{-1} + b_2z^{-2} + b_3z^{-3} + \dots}{a_0 + a_1z^{-1} + a_2z^{-2} + a_3z^{-3} + \dots} \quad (\text{VI.15})$$

Il existe principalement deux approches pour implémenter et concevoir un filtre numérique:

- Réalisation sous forme directe: la fonction de transfert estimée est implémentée directement.
- Réalisation en cascade: la fonction de transfert est décomposée sous la forme de pôles et de zéros ou de blocs élémentaire du second ordre.

En terme d'implémentation, qu'elle soit matérielle ou logicielle, les différent éléments suivants sont utilisés:

- Additionneur: \oplus
- Multiplieur: \otimes
- Délai: z^{-1} ou T
- Lien: \rightarrow

VI.7.1 Implémentation filtre RIF

Un filtre à réponse impulsionnelle finie (RIF) est définie par la fonction de transfert suivant. En pratique, les coefficients b_k sont souvent confondus avec ceux de sa réponse impulsionnelle h_k .

$$H(z) = b_0 + b_1z^{-1} + b_2z^{-2} + b_3z^{-3} + \dots \quad (\text{VI.16})$$

Il est possible de schématiser la structure directe d'implémentation du filtre (RIF) à l'aide d'opérateurs retard et de multiplication par des coefficients b_k par la figure suivante:

VI.7.2 Implémentation filtre RII

VI.7.2 Implémentation forme directe

$$H(z) = \frac{P(z)}{D(z)} = \frac{b_0 + b_1z^{-1} + b_2z^{-2} + b_3z^{-3} + \dots}{a_0 + a_1z^{-1} + a_2z^{-2} + a_3z^{-3} + \dots} \quad (\text{VI.17})$$

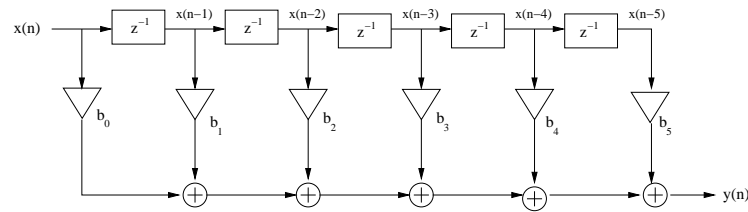


Figure VI.1: Schéma bloc de l'implémentation directe d'un filtre RIF

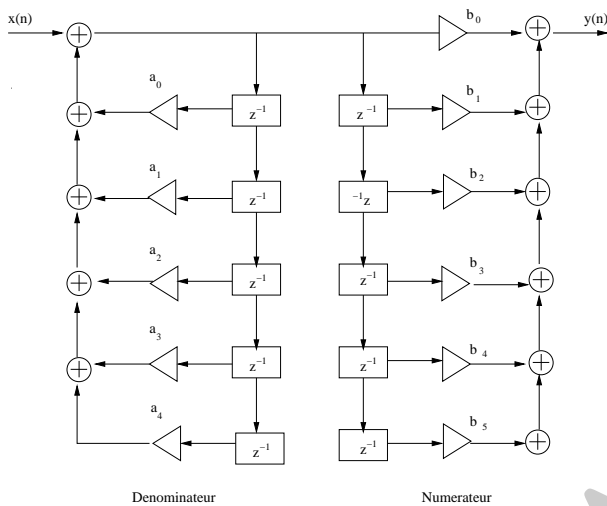


Figure VI.2: Schéma bloc de l'implémentation directe d'un filtre RII

VI.7.2 Implémentation sous la forme canonique

VI.7.3 Implémentation bloc en série

VI.7.3 Décomposition sous la forme de bloc du second degré

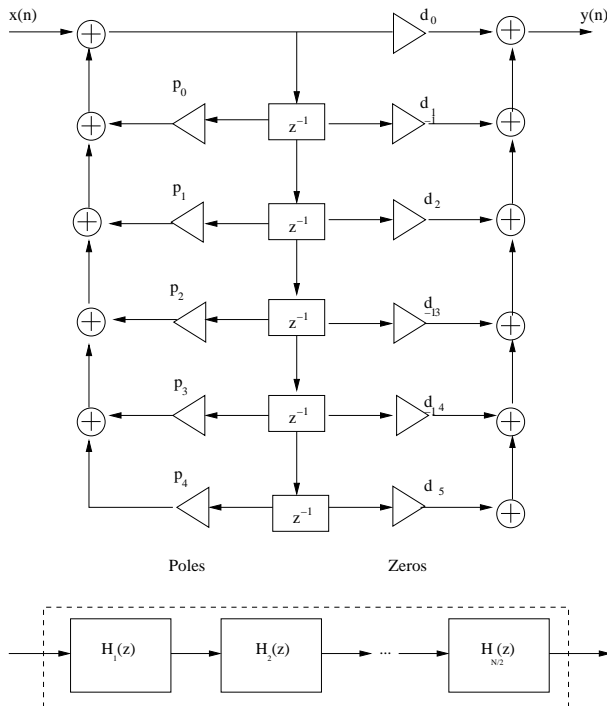
La décomposition sous la forme de fonction de transfert du second ordre:

$$H(z) = \prod_{i=1}^{N/2} H_i(z) = \prod_{i=1}^{N/2} \frac{b_{i,0} + b_{i,1}z^{-1} + b_{i,2}z^{-2}}{1 + a_{i,1}z^{-1} + a_{i,2}z^{-2}} \quad (\text{VI.18})$$

Le principal avantage d'une décomposition sous la forme de blocs du second degré est une moindre sensibilité du filtre RII au bruit de quantification ou la propagation d'erreur de calcul numérique.

VI.7.3 Méthode de réalisation

La méthode où l'algorithme pour l'implémentation de la fonction de transfert d'un bloc RII en cascade est:



- Les pôles et zéros de la fonction de transfert sont calculés.
- Etape i :
 1. Trouver les deux pôles p_{i1} et p_{i2} dont les modules sont le plus proche du cercle de stabilité (ROC).
 2. Trouver les deux zéros λ_{i1} et λ_{i2} , les plus proches des pôles p_{i1} et p_{i2} .
 3. Construire la fonction de transfert du bloc de second ordre.
 4. Enfin, enlever les pôles p_{i1} , p_{i2} , λ_{i1} , λ_{i2} comme un choix possible et réitérer ensuite l'étape i de la procédure pour la création du bloc du second degré suivant $i + 1$.

VI.7.3 Comparaison filtre RIF vs RII

La problématique du choix d'un filtre à réponse impulsionnelle finie (RIF) versus un filtre à réponse impulsionnelle infinie (RII) se pose souvent. Il est possible de résumer les avantages ou les inconvénients de l'une ou l'autre de ces catégories de filtre par les arguments suivants:

- Les filtres RII nécessitent souvent moins de ressources en terme de calculs.
- Les filtres RIF sont souvent caractérisés par une phase linéaire.
- Les filtres RIF sont inconditionnellement stables. Par contre, les filtres RII peuvent être instables.
- Les filtres RIF sont caractérisés par une fenêtre temporelle finie. Ce qui fait que l'influence d'un bruit peut influencer le filtre RIF uniquement sur un nombre limité d'échantillons. Ce n'est

pas le cas des filtres RII où la rétroaction se fait et son influence n'est pas limitée sur le plan temporel.

VI.7.3 Confusion méthode de conception RIF vs RII

Il est souvent observé une confusion entre les méthodes de conception de filtre à réponse impulsionnelle finie (RIF) ou RII:

- La conception des filtres RIF est principalement basée sur la méthode troncature de la réponse impulsionnelle. Celle-ci est également implémentée au niveau de la fonction `fir1.m` de Matlab. L'ensemble du procédé de conception se place dans le domaine **numérique**.
- Pour un filtre RII, la méthode est basée sur la conception d'un filtre analogique classique et ensuite la transformée bilinéaire. La conception est réalisée dans le domaine **analogique**.

Les filtres numériques RIF ou RII sont conçus par des méthodes (troncature de la réponse impulsionnelle versus transformée bilinéaire) et sur des domaines (numériques ou analogique) différents.

VI.8 Exercice

Le gabarit d'un filtre numérique RII passe-bas est défini par les paramètres suivant:

- La fréquence d'échantillonnage est de 1000 Hz.
- La fréquence de bande passante est de 50 Hz.
- La fréquence de coupure est de 200 Hz.
- L'ondulation en bande passante est de 1.74 dB.
- L'atténuation en bande coupée est de 20 dB.

En reprenant la méthode de synthèse d'un filtre RII basée sur la transformé bilinéaire, concevoir le filtre et proposer un schéma d'implémentation.

Tutoriel : implant cochléaire et filtre à réponse impulsionnelle infinie (RII)

Ouïe et appareil auditif humain

La figure suivante présente une description sommaire de l'oreille humaine avec ces différentes parties (interne, moyenne, externe). L'onde acoustique sonore permet le déplacement de la membrane basilaire, celle-ci étant attachée aux cils internes qui permettent de transformer le déplacement de celle-ci en signaux électrochimiques au niveau des nerfs. Ceci correspond au phénomène de l'ouïe. La seconde figure présente un diagramme simplifié de la membrane basilaire et de la position en fréquence des maximums de résonance.

Prothèse cochleaire De manière à concevoir une prothèse cochléaire possédant les propriétés de l'oreille humaine, les différentes opérations en terme de traitement numérique du signal sont nécessaires. Le principe afin de fabriquer une prothèse auditive ayant des propriétés similaires à celle de l'oreille. Le son collecté par un microphone est le suivant, un son est collecté par un microphone (1) et envoyé à un processeur de traitement du signal (2). Cette information est ensuite envoyée par un câble électrique (3) au niveau de la peau du patient via un lien téléométrique (4). Dans l'oreille du patient, un récepteur interne (6) utilise les informations de manière à délivrer des impulsions à un réseau d'électrodes au nerfs auditifs permettant la perception du son au patient. La figure suivante présente le principe d'un implant cochleaire.

Filtrage de la bande passante

L'objectif principal de ce tutoriel est de réaliser un filtre RII permettant d'extraire le son ou la parole autour de la fréquence de 5100 Hz. L'ondulation en bande passante est de 1.74 dB et l'atténuation en bande coupée est de 20 dB.

- L'ondulation maximale en bande passante (dB) est de 1.74 dB.
- L'atténuation en bande coupée est de 20 dB.

Certains paramètres du filtre RII nécessitent d'être déterminés comme le gabarit du filtre et les fréquences de bande coupée.

1. En analysant la figure, trouver la fréquence de bande coupée qui doit être considérée au niveau du cahier des charges de manière à extraire la bande de fréquence autour de 5100 Hz. Estimer la fréquence d'échantillonnage possible.

La fréquence d'échantillonnage est fixée à 40 kHz. De manière à estimer la fréquence de bande passante, la largeur de la bande de transition Δf est définie comme étant 2 % de la fréquence la plus élevée de bande coupée.

2. Représenter le gabarit du filtre analogique équivalent en utilisant les spécifications précédentes du filtre RII.

3. Déterminer le cahier des charges du filtre analogique équivalent.

Un filtre de Butterworth est utilisé de manière à implémenter le filtre analogique

4. Déterminer l'ordre du filtre nécessaire

5. Déterminer les coefficients du filtre de Butterworth dans le domaine analogique

6. En utilisant la transformée bilinéaire, déterminer la fonction de transfert du filtre numérique

7. Représenter la réponse en fréquence du filtre numérique et la comparer avec les spécifications initiales du cahier des charges. Tracer également la réponse en phase du filtre en fréquence. Quel ordre est-il nécessaire de prendre afin d'avoir un filtre RII stable ?

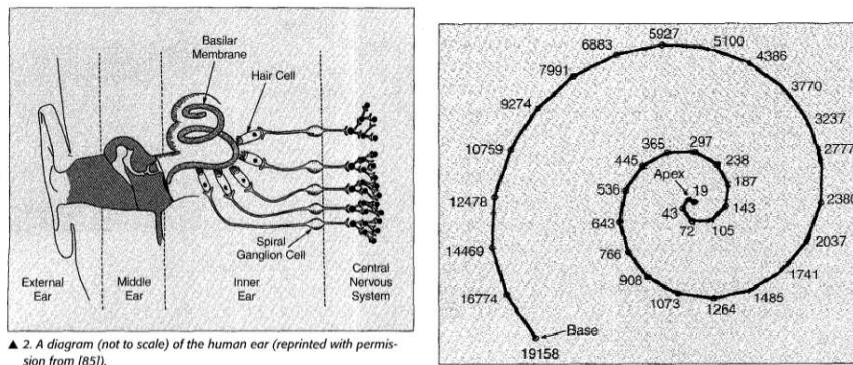


Figure VI.3: Description de l'oreille humaine et ces différentes parties (interne, moyenné et externe) et la membrane basilaire. Diagramme simplifié de la membrane basilaire indiquant la fréquence maximale de déplacement.

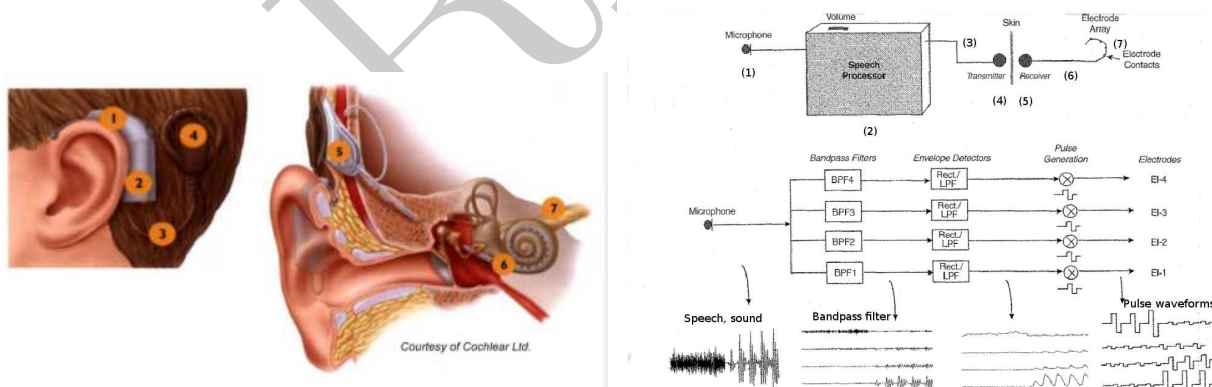


Figure VI.4: Principe de la prothèse cochleaire et les différentes étapes de traitement du signal nécessaire afin d'avoir des propriétés auditives similaires à l'oreille humaine et délivrer le signal approprié.

DRAFT

VII –



DRAFT

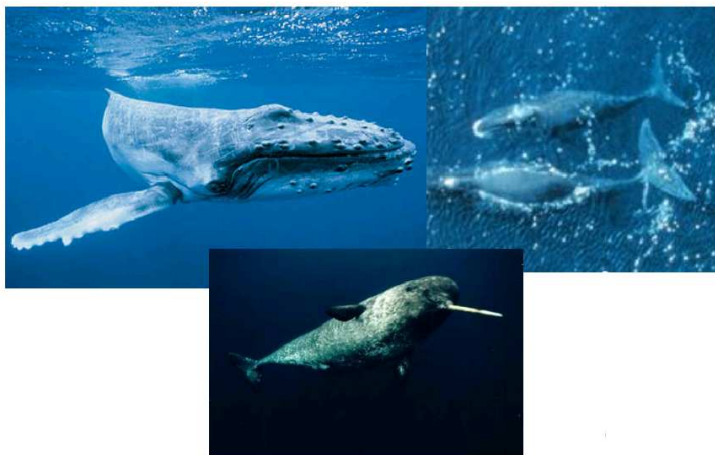
DRAFT

VIII – Practiciel : conception et implémentation d'un filtre numérique

Auteur: Alain Fruleux/Christophe Krzeminski

VIII.1 Introduction

Le filtrage numérique est souvent utilisé afin de sélectionner la partie intéressante d'un signal. Par exemple, si l'on veut étudier le comportement de différents poissons ou de mammifères marins, *l'ambiance acoustique provenant de l'océan doit être filtrée* de manière à extraire précisément le son des animaux. Dans ce praticiel, l'objectif est de séparer le son de la baleine à bosse, de la baleine boréale et enfin de l'espadon.



Dans le cadre des cours de traitement du signal numérique, l'implémentation d'un filtre numérique est proposé. Ce travail pratique est suivi d'une évaluation pratique utilisant une carte électronique basée sur un processeur de traitement du signal (TNS). Le principal objectif de cette séance est dans un premier temps de comparer la réponse d'un filtre analogique à celui numérique, ensuite d'implémenter un filtre numérique RIF. La méthode de conception considérée est le fenêtrage de la

réponse impulsionnelle et l'utilisation de la transformée bilinéaire pour la conception du gabarit du filtre. Vous aurez également besoin d'Octave ou Matlab et probablement de l'une ou l'autre de ces fonctions :

- `window` : permet la conception d'une fenêtre.
- `fir1` : génère un filtre RIF d'ordre N par la méthode de la troncature de la réponse impulsionnelle.
- `freqz` : calcule la réponse en fréquence du filtre.
- `buttord` : estime l'ordre du filtre de Butterworth nécessaire.
- `butter` : génère un filtre Butterworth d'ordre N .
- `bilinear` : implémente la transformée bilinéaire.
- `grpdelay` : calcule le délai de groupe.

Au préalable de la réalisation de ce praticiel, il est recommandé des connaissances théoriques au niveau des fonctions Octave ou Matlab sur la conception de filtres. De plus, il est préférable d'avoir une connaissance du langage C. Durant cette séance, le principal objectif sera l'analyse d'un programme écrit dans ce langage pour l'implémentation d'un filtre numérique. Les calculs se feront à virgule fixe. Il peut être intéressant pour vous de prendre des notes des résultats de la séance.

VIII.2 Description sommaire de l'implémentation d'un filtre

Durant cette séance de travail pratique, l'objectif est d'implémenter le filtre numérique et analogique et ensuite implémenter un filtre à réponse impulsionnelle finie (RIF). Le filtre sera implémenté sur la carte de traitement de signal, ceci n'est pas une simulation Octave ou Matlab. La carte USBTK5515 de la société Texas instrument sera utilisée pour une implémentation en temps réel.

- lire les différents sons du praticiel sur votre portable.
- l'entrée du signal audionumérique se fait au niveau de la connection jack de la carte.
- le signal est convertit à l'aide d'un convertisseur analogique/numérique en une série de bits.
- ces échantillons sont sauvegardés dans le registre mémoire de la carte.
- le programme lit les échantillons du registre mémoire d'entrée et réordonne les résultats de sortie.
- le signal est en théorie filtré et vous pouvez entendre le résultat.

Propriété intellectuelle de Texas Instrument

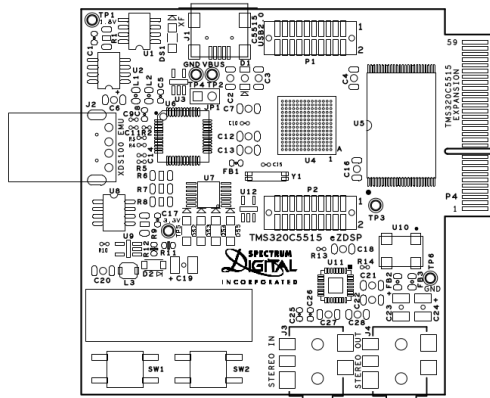
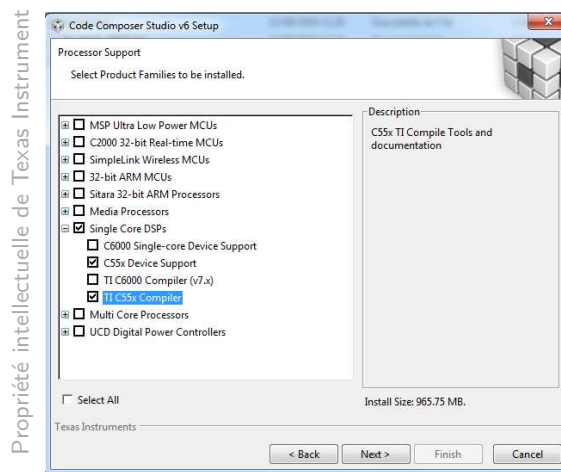


Figure VIII.1: Schéma de conception de la carte électronique Digital Spectral USBTK5515 de Texas Instrument

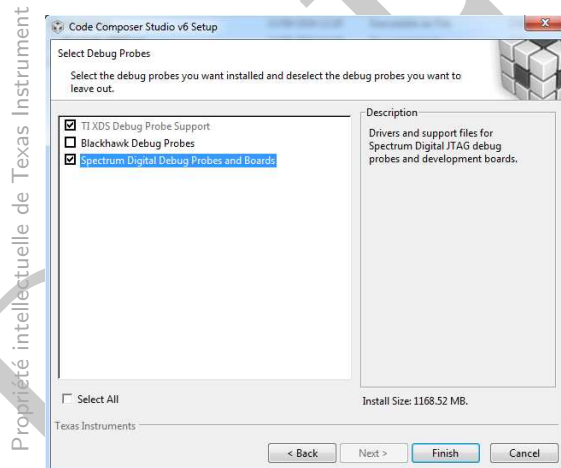
VIII.3 Installation du code composer studio

L'ensemble des logiciels code composer studio permet d'écrire, de compiler et de transférer le code exécutable sur un processeur de traitement du signal (TNS). Il est possible de visualiser les registres d'entrée et de sortie. De manière à installer code composer studio et de réaliser ces opérations :

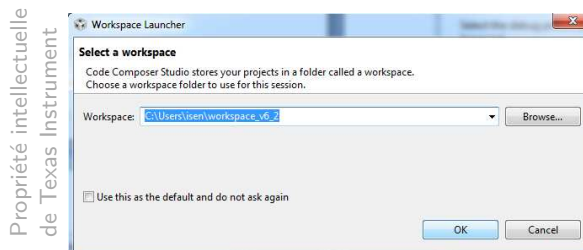
- Il est nécessaire de pas connecter la carte TNS à votre portable avant l'installation du logiciel.
- Démarrer l'installation de Code Composer Studio (CCSV6.X) sur votre pc en utilisant l'installateur se trouvant dans la clef usb.
- L'installation dure normalement environ une trentaine de minutes si votre disque n'est pas trop chargé ou fragmenté. Il est préférable de réaliser l'installation du logiciel sur le disque C.
- Sélectionner le processeur de la carte DSPs C55xx utilisé par le compilateur.



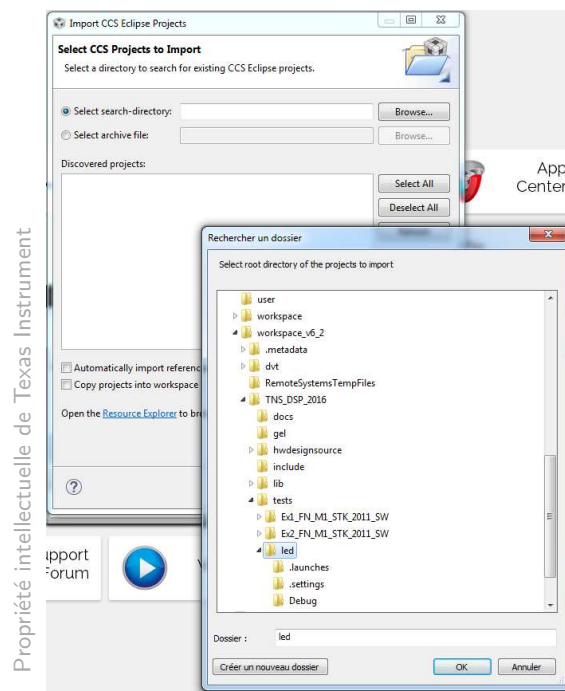
- Sélectionner l'environnement de debug et la carte de debugage Ti XDS Spectrum Digital.



- Choisir l'environnement par défaut C:\Users (ou utilisateurs)\isen\workspace_v6_2. Il s'agit normalement du répertoire où les fichiers doivent être copiés.



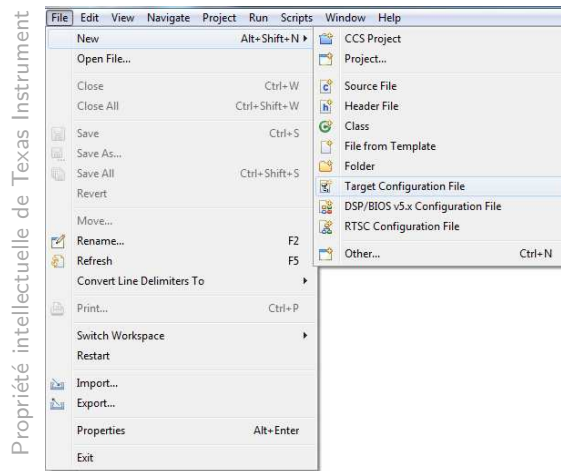
- Après l'installation du logiciel Code Composer Studio v6.2 et avant de lancer Code Composer, copier le répertoire suivant (TNS_DSP_2016) dans le répertoire préalablement défini.



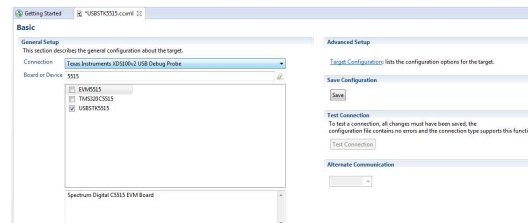
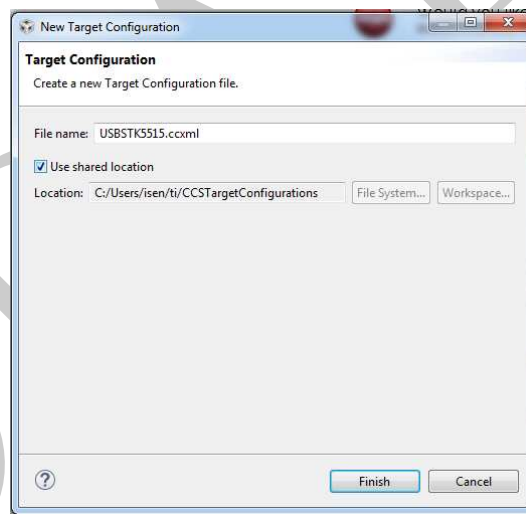
VIII.4 Projet led et test primaire de la carte TNS

Il est possible de démarrer Code Composer Studio qui peut être lancé à l'aide d'un simple clic de souris sur l'icône du logiciel code composer studio. Il est possible de définir la carte sur laquelle vous allez pouvoir travailler. La carte "Texas Instruments XDS100v2 USB Emulator/USBSTK5515" doit être sélectionné en définissant "Target/New target configuration". La création de la cible est expliquée juste après. Vous devez ensuite importer le projet de la manière suivante:

- Lancer la commande Project/import CCS...importer CCS et sélectionner le projet led



- Lancer le projet led project dans le répertoire `project_explorer`, compiler le projet (build project)
- Créer un fichier cible de configuration USBSTK5515.ccxml et sélectionner le répertoire partagé.



Une fois, la cible de la configuration créé, l'ensemble des opérations suivantes peuvent maintenant être réalisées :

- Project/rebuild projet actif
- Run/debug projet actif
- Target/run

La led de la carte doit clignoter une dizaine de fois et s'éteindre. Il est ainsi possible de passer à la seconde partie concernant la conception du filtre.

VIII.5 Test de l'implémentation d'un filtre numérique

Cette section a deux objectifs principaux à savoir la prise en main de l'environnement numérique gérant la carte de traitement du signal et ensuite d'étudier les propriétés des filtres numériques.

VIII.5.1 Test de fonctionnalité du filtre numérique

- Ouvrir l'environnement de travail indiqué dans la section précédente.
- Charger le projet dans la carte DSP : Ex1_FN_M1_stk_SW
- Rendre le projet actif: dans la liste des différents projets, sélectionner le projet Ex1_FN_M1_stk_SW et le rendre actif par un click droit.
- Le projet actif (cette opération doit être réalisée pour chaque filtre).
- Ne pas modifier les coefficients du filtres pour ce test de fonctionnalité.
- Target/debug du projet actif

- Lancer sur votre portable une forme sinusoidale et connecter votre casque sur l'entrée de la carte TNS
- Target/Run
 - Le bouton SW1 sur la carte permet de passer du mode filtré au mode non filtré.
 - Il est souvent intéressant de comparer les deux modes (filtré ou pas) de manière à évaluer l'influence du filtre.
 - Faire attention que le programme de lancement du filtre ne dure qu'une trentaine de secondes.

VIII.5.2 Conception et estimation des coefficients du filtre numérique

Le filtre numérique dont la fréquence d'échantillonnage est de 8 kHz est un filtre analogique passe-bas de fréquence de coupure 800 Hz. En analysant le programme utilisé retrouver les caractéristiques

suivantes:

- Structure du filtre numérique
- Fonction de transfert
- Type de filtre : RII ? RIF ?

En utilisant la transformée bilinéaire, calculer les coefficients du filtre numérique en faisant l'hypothèse d'un simple filtre analogique passe-bas. La procédure principale pour définir le filtre est la suivante :

```
// C program for the first exercice
// Definition of the digital filter
#define ordre 2 // Correspond aux nombres d'\`echantillons utilis\`es durant le filtrage
#define PASFILTRE 1 // Permet d'appliquer ou non un filtre
#define FILTRE 0

/* Coefficients du Filtre */
int coeff_a[ordre]={0x0086,0x0000} ;
    int coeff_b[ordre]={0x003C,0x003C} ;
    int bypass= FILTRE ;

void FiltreC (Int16 *data, Int16 *results, Int16 blksize)
{
    int i;
    int j;
    long addsamp;
/* definition de la sortie du filtre */
    long y;
    for (i = 0; i < blksize; i++)
    {
        if (bypass==0)
        {
/* Definition de l'equation aux differences finies */
y=0 ;
/* Calcul de la partie auto-regressive*/
for (j = 0; j<ordre; j++)
    {
        addsamp=results[i-j-1];
        y =y+coeff_a[j]*addsamp;
    }

```

```

/* Calcul de la partie mobile*/
for (j = 0; j<ordre; j++)
    {
        addsamp=data[i+ordre-1-j];
        y =y + coeff_b[j]*addsamp;
    }
    y=y/256      ;
//les coefficients sont cod\`es en virgule fixe et la sortie normalis\`ee par
//rapport \`a
    results[i]= y;
}
else
results[i]=data[i] ;
}
}

```

VIII.6 Conception et Implémentation d'un filtre RIF

Cet exercice consiste à choisir le spectre d'un des animaux et à extraire un signal complexe de manière dans une gamme de fréquence. Il sera possible de trouver un exemple de gabarit de filtre qui indique le spectre fréquentiel des différents signaux.

- Définir le gabarit du filtre et isoler le son de chaque animal.
- En utilisant Octave ou Matlab, calculer les coefficients et la réponse en fréquence du filtre numérique RIF ou RII. Vérifier que la réponse obtenue correspond au cahier des charges.
- Implémenter les coefficients du filtre de la carte DSP.

- Sélectionner Ex2_FN_M1_stk_SW comme un projet actif.
- Entrer les coefficients du filtre RIF conçu au niveau de la table suivante: int coeff_a[ordre] et int coeff_b[ordre]

Il doit être fait attention au fait que:

1. Les coefficients sont codés en virgule fixe ($1=2^{12}$).
 2. Le signe des coefficients a_i peuvent dépendre des définitions des structures d'implémentation ou de l'équation aux différences.
- Compiler (Project/Rebuild_all) à chaque fois que les valeurs des coefficients sont modifiés.
 - Recharger le programme sur la carte à chaque modification.

Il est important de vérifier que chaque filtre testé sur de simples formes sinusoidales, le filtre pouvant être désactivé en activant l'interrupteur sur la carte. Comparer la réponse fréquentielle avec celle du spectre étudié. En théorie, il est possible d'extraire le son de la baleine à bosse, de la baleine boréale ou de l'espadon. Néanmoins, il peut y avoir des limitations par des contraintes matérielles.

VIII.7 Sommaire

Nom	Description	Objectif
Led	Test	Test de fonctionnalité
Ex1_FN_M1_stk_SW	Filtre RII	Echantillonnage Filtre RII
Ex2_FN_M1_stk_SW	Filtre RIF (Ordre max. 32)	Exemple de filtre RIF Conception filtre Calcul des coefficients Test fonctionnalité

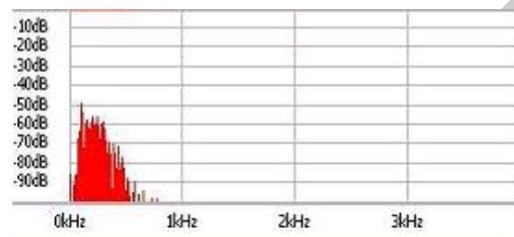


Figure VIII.2: Spectre fréquentiel de la baleine à bosse.

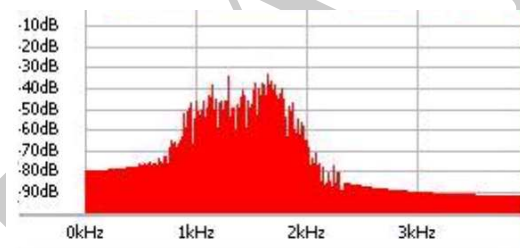


Figure VIII.3: Spectre fréquentiel de la baleine boréale.

DRAFT

Figure VIII.4: Spectre fréquentiel pour le narval