
II. Les filtres numériques: FIR

1)Définition: FIR

les filtres à réponse impulsionnelle finie (RIF) ou FIR (Finite Impulse Response). Dans ces filtres tous les coefficients a_j sont nuls (seulement les coefficients b_i sont présents):

$$H(z) = \frac{Y(z)}{X(z)} = \sum_{i=0}^{i=M} b_i z^{-i}$$

2) L'équation de récurrence d'un FIR

- Équation d'e/s : $y(n)$ est calculée en fonction des entrées (filtre non récurrents)

$$y[n] = \sum_{i=0}^{N-1} b_i x[n-i]$$

$x[n]$ représente les valeurs successives du signal d'entrée,

b_i représente les coefficients de la fonction de transfert du filtre,

$y[n]$ représente les valeurs successives du signal de sortie,

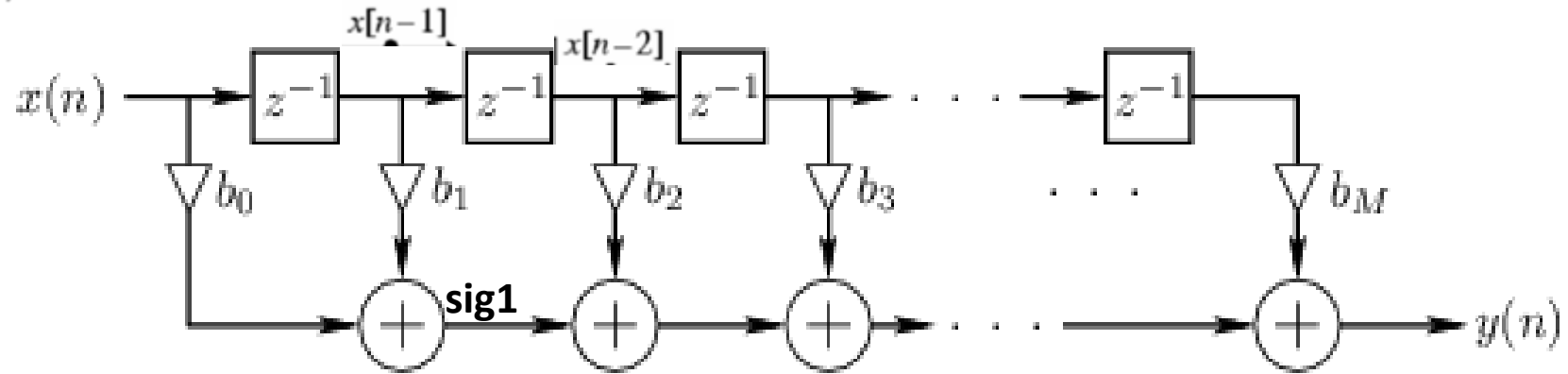
N est le nombre de coefficients du filtre (l'ordre).

3) Propriétés des FIRs

- Réponse stable par défaut
- Réponse en phase linéaire pour un filtre réalisable
- Peuvent demander un temps de calcul excessif

4) Implementation du filtre FIR

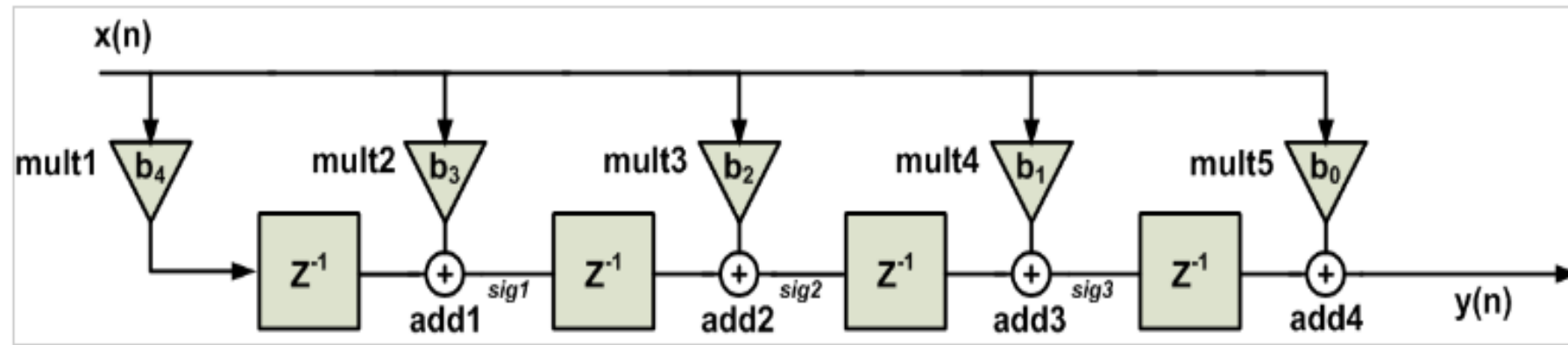
a) Structure directe



$$\text{Sig1} = b_0 x(n) + b_1 x(n-1)$$

4) Implementation du filtre FIR

b) Structure transposée



$$sig1 = b_4x(n-1) + b_3x(n)$$

$$sig2 = b_4x(n-2) + b_3x(n-1) + b_2x(n)$$

$$sig3 = b_4x(n-3) + b_3x(n-2) + b_2x(n-1) + b_1x(n)$$

$$y(n) = b_4x(n-4) + b_3x(n-3) + b_2x(n-2) + b_1x(n-1) + b_0x(n)$$

5) Synthèse des filtres FIR

Concevoir un filtre numérique consiste à déterminer la **fonction de transfert en z** du filtre qui va approcher au mieux les spécifications sur la **réponse fréquentielle en amplitude**

- 1) Méthode de **fenêtrage** (domaine temporel)
- 2) Méthode par **échantillonnage** fréquentielle
- 3) Synthèse par **optimisation** (algo. Remez)

Méthode de fenêtrage (domaine temporel)

Le calcul des coefficients repose sur l'utilisation de la **transformée numérique** et des fenêtres spectrales ((rectangulaire, Hanning, Hamming,...)).

6) Conception de FIR par fenêtrage

Spécifications

w_p : Pulsation de la bande passante

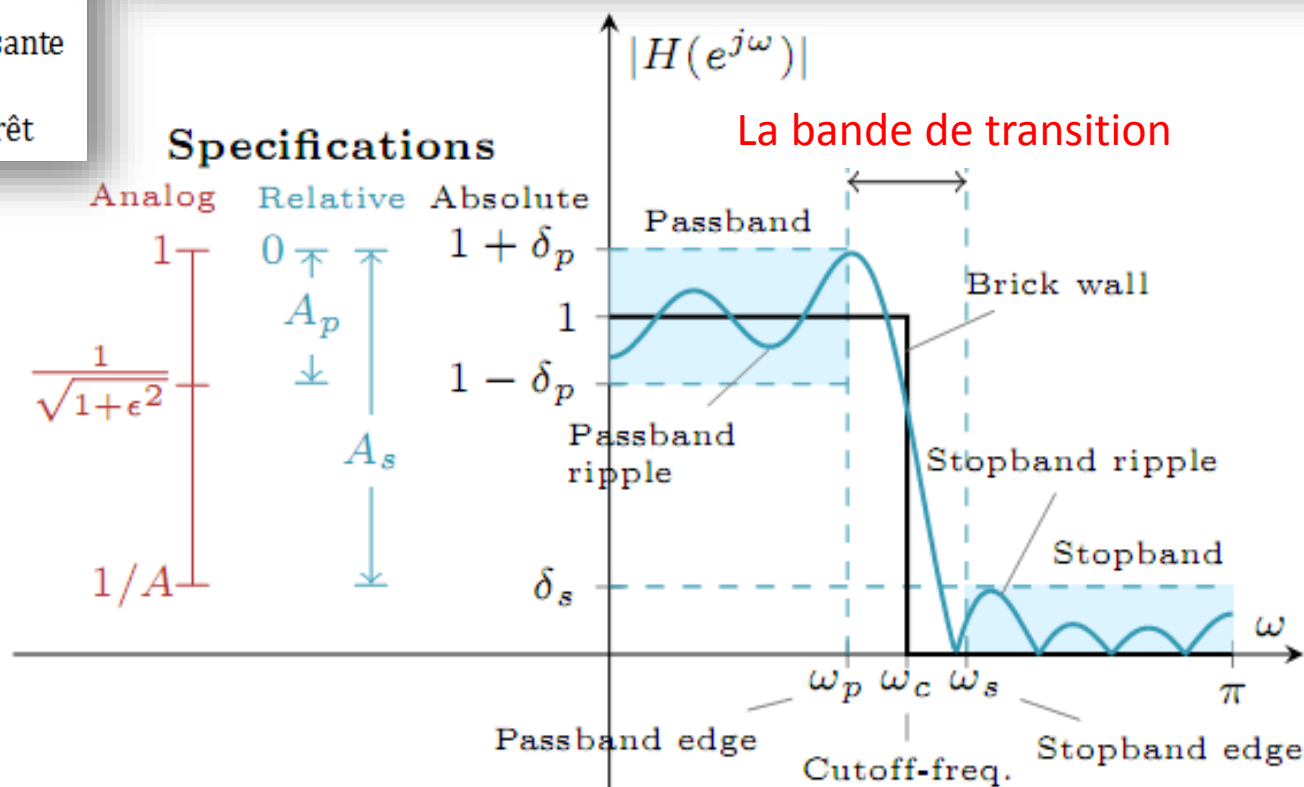
w_s : Pulsation de la bande d'arrêt

δ_p : Déviation de la bande passante

δ_s : Déviation de la bande d'arrêt

Paramètres à considérer

- La **bande passante** est définie par : $0 \leq w \leq w_p$, on a besoin $|G(e^{jw})| \cong 1$ avec une erreur $\mp \delta_p$, c'est-à-dire $1 - \delta_p \leq |G(e^{jw})| \leq 1 + \delta_p, |w| \leq w_p$
- La **bande d'arrêt**, définie par $w_s \leq w \leq \pi$, on a besoin $|G(e^{jw})| \cong 0$, avec une erreur δ_s , c'est-à-dire $|G(e^{jw})| \leq \delta_s, w_s \leq |w| \leq \pi$



6) Conception de FIR par fenêtrage

- Dans la méthode de la fenêtre, nous développons un filtre FIR de phase linéaire causal en **multipliant** un **filtre idéal ($h_d(n)$)** par une **fonction fenêtre ($w(n)$)** de durée finie:

$$h[n] = h_d(n)w[n]$$

7) Réponse impulsionnelle des filtres idéals

A. Lowpass filters

$$H_d(e^{j\omega}) = \begin{cases} G e^{-j\omega n_d}, & 0 \leq |\omega| \leq \omega_c \\ 0, & \omega_c < |\omega| \leq \pi \end{cases}$$

$$h_d[n] = G \frac{\sin(\omega_c(n - n_d))}{\pi(n - n_d)}$$

B. Highpass filters

$$H_d(e^{j\omega}) = \begin{cases} G e^{-j\omega n_d}, & \omega_c \leq |\omega| \leq \pi \\ 0, & |\omega| < \omega_c \end{cases}$$

$$h_d[n] = G \left(\delta[n - n_d] - \frac{\sin(\omega_c(n - n_d))}{\pi(n - n_d)} \right)$$

C. Bandpass filters

$$H_d(e^{j\omega}) = \begin{cases} G e^{-j\omega n_d}, & \omega_1 \leq |\omega| \leq \omega_2 \\ 0, & |\omega| \leq \omega_1 \\ 0, & \omega_2 \leq |\omega| \leq \pi \end{cases}$$

$$h_d[n] = G \left(\frac{\sin(\omega_2(n - n_d))}{\pi(n - n_d)} - \frac{\sin(\omega_1(n - n_d))}{\pi(n - n_d)} \right) \text{ or, equivalently,}$$

$$h_d[n] = 2G \cos\left(\left(\frac{\omega_2 + \omega_1}{2}\right)(n - n_d)\right) \frac{\sin\left(\left(\frac{\omega_2 - \omega_1}{2}\right)(n - n_d)\right)}{\pi(n - n_d)}$$

8)Implementation en Matlab

- `w=boxcar(M)` returns the M-point rectangular window function in array `w`.
- `w=bartlett(M)` returns the M-point Bartlett window function in array `w`.
- `w=hann(M)` returns the M-point Hann window function in array `w`.
- `w=hamming(M)` returns the M-point Hamming window function in array `w`.
- `w=blackman(M)` returns the M-point Blackman window function in array `w`.
- `w=kaiser(M,beta)` returns the `beta`-valued M-point rectangular window function in array `w`.

9)Exemple de fenêtre:wvtool

M: ordre du filtre

Rectangulaire

$$w[n] = \begin{cases} 1, & 0 \leq n \leq M \\ 0, & \text{otherwise} \end{cases}$$

Bartlett :

$$w(n) = \begin{cases} \frac{2n}{M} & 0 \leq n \leq \frac{M}{2} \\ 2 - \frac{2n}{M} & \frac{M}{2} \leq n \leq M \end{cases}$$

Hanning :

$$w(n) = \frac{1}{2} \left[1 - \cos\left(\frac{2\pi n}{M}\right) \right] \quad 0 \leq n \leq M$$

Hamming :

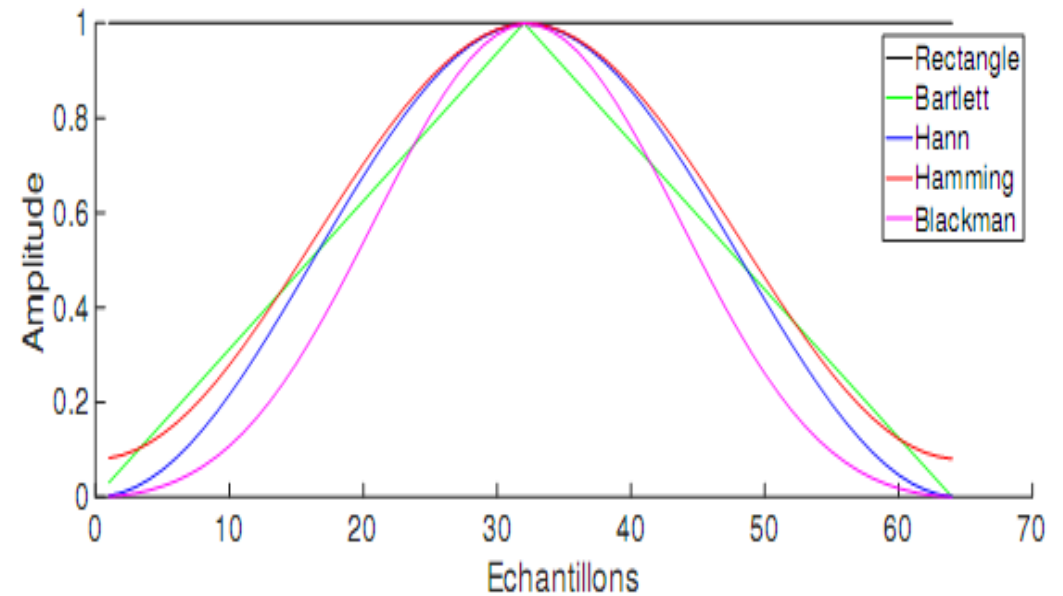
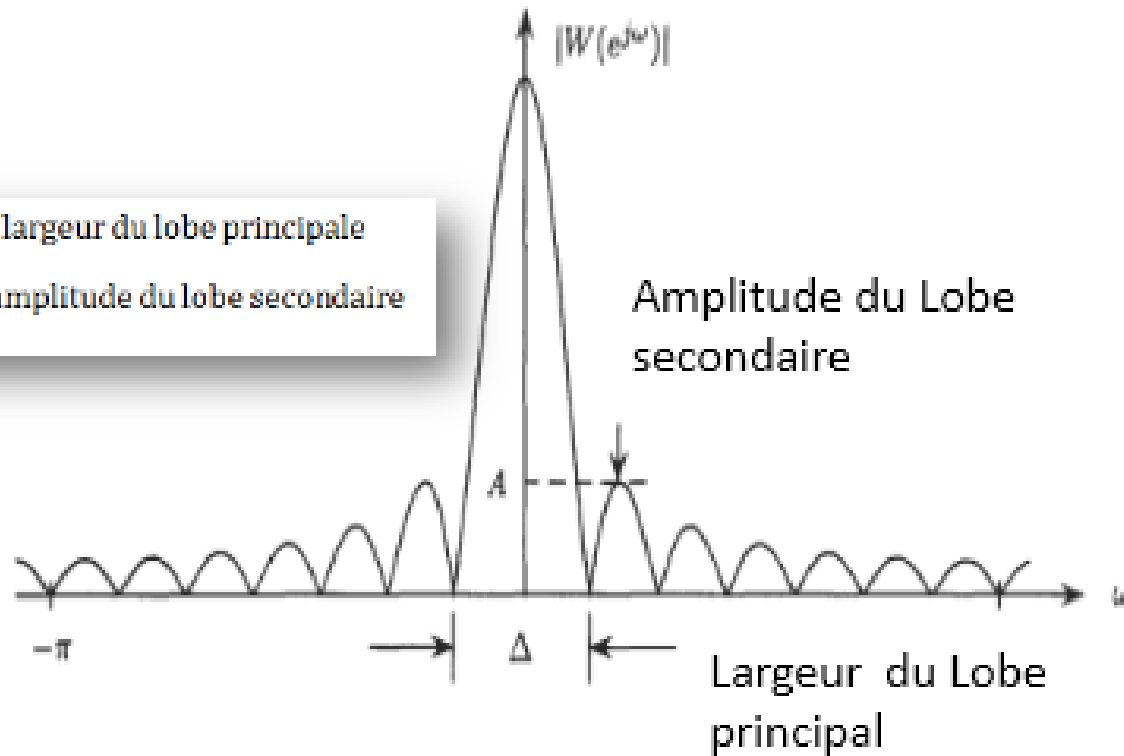
$$w(n) = 0.54 - 0.46 \cos\left(\frac{2\pi n}{M}\right) \quad 0 \leq n \leq M$$

Blackman :

$$w(n) = 0.42 - 0.5 \cos\left(\frac{2\pi n}{M}\right) + 0.08 \cos\left(\frac{4\pi n}{M}\right) \quad 0 \leq n \leq M$$

9)Exemple de fenêtre:temporelle

- La largeur du lobe principale
- L'amplitude du lobe secondaire



Fenêtres populaires : représentation temporel

11)La méthode des fenêtres

On trouve l'ordre du filtre selon la largeur de la bande de transition, une fois l'atténuation est déterminée dans la bande d'arrêt.

N: la longueur de la fenêtre

$\Delta f = f_s - f_p$

Ordre $M = N - 1$

Fenêtre	Bande de transition normalisée ($\Delta f(\text{Hz})$)	Ondulation dans la bande passante (dB)	Atténuation dans la bande d'arrêt (dB)	Amplitude du lobe secondaire (dB)	β de la fenêtre de Kaiser équivalente
Rectangulaire	$\frac{0.9}{N}$	0.7416	-21	-13	0
Hanning	$\frac{3.1}{N}$	0.0546	-44	-31	3.86
Hamming	$\frac{3.3}{N}$	0.0194	-53	-41	4.86
Blackman	$\frac{5.5}{N}$	0.0017	-74	-57	7.04
Kaiser	$\frac{2.93}{N} \rightarrow \beta = 4.54$	0.0274	-50		
	$\frac{5.71}{N} \rightarrow \beta = 8.96$	0.000275	-90		

12) Méthode de la conception

Soit les spécifications suivantes : $\{w_p, w_s, \delta_p, \delta_s\}$

1- $\delta = \min \{ \delta_p, \delta_s \}$

2- $w_c = \frac{w_p + w_s}{2}$

3- Calcul de l'atténuation $A_s = -20 \log_{10} \delta$ et $\Delta w = w_s - w_p$ (A_s atténuation minimale de la bande coupée)

4- Choisir une fenêtre (à partir de la table) :

- Sélectionner la forme de fenêtre qui possède le petit A qu'est plus grand que A_s
- Sélectionner une longueur de fenêtre minimale ($N = ?$)

5- Calculer la réponse impulsionnelle par $h_d[n] = \frac{\sin [w_c(n - \frac{M}{2})]}{\pi(n - \frac{M}{2})}$ (pour un filtre Passe-bas)

6- Calculer la réponse impulsionnelle fenêtrée par (multiplier les coefficients par la fenêtre) $h[n] = h_d[n]w[n]$

Exercice

Exercice 1

Concevoir un filtre passe-bas à phase linéaire FIR selon les spécifications suivantes :

$$0.99 \leq |H(e^{jw})| \leq 1.01 \quad 0 \leq |w| \leq 0.19\pi$$

$$|H(e^{jw})| \leq 0.01 \quad 0.21\pi \leq |w| \leq \pi$$

III. Les filtres numériques IIR

1)Introduction

- Les FIR n'a pas d'équivalence analogique
- IIR peut être dérivé d'un filtre analogique équivalent
- L'introduction des pôles dans $H(z)$ de IIR réduit considérablement le nbre de coeff par rapport à un filtre équivalent FIR
- Inconvénient: la stabilité des IIR doivent être vérifiées lors de sa conception

2) Définition :filtres récursif IIR

- ✓ Filtres à réponse impulsionnelle infinie (RII) ou IIR (Infinite Impulse Response), ces filtres sont encore appelés filtres récursifs.
- ✓ Dans ces filtres les coefficients **a_j** sont différents de zéro, en conséquence un échantillon y(n) dépend de tous les échantillons x(n) présents et passés, et aussi en fonction des valeurs passées de la sortie y(n-1).
- ✓ La transformée H(z) de ces filtres s'écrit :

$$H(Z) = \frac{Y(z)}{X(z)} = \frac{\sum_{i=0}^M b_i Z^{-i}}{1 - \sum_{j=1}^N a_j Z^{-j}}$$

3)Equation d'un IIR

- Équation d'e/s :

$$Y[n] = \sum_{i=0}^{N-1} b[i] x[n-i] + \sum_{j=1}^M a[j] y[n-j]$$

- $x[n]$ représente les valeurs successives du signal d'entrée,
 a_j, b_i représentent les coefficients de la fonction de transfert du filtre,
 $y[n]$ représente les valeurs successives du signal de sortie,
 N, M représentent les ordres du numérateur et du dénominateur de $H(Z)$ (M est souvent appelé l'ordre du filtre).

4) Implémentation d'un filtre IIR

1. Forme directe 1
2. Forme directe 2
3. Forme directe transposée

4) Implémentation d'un filtre IIR

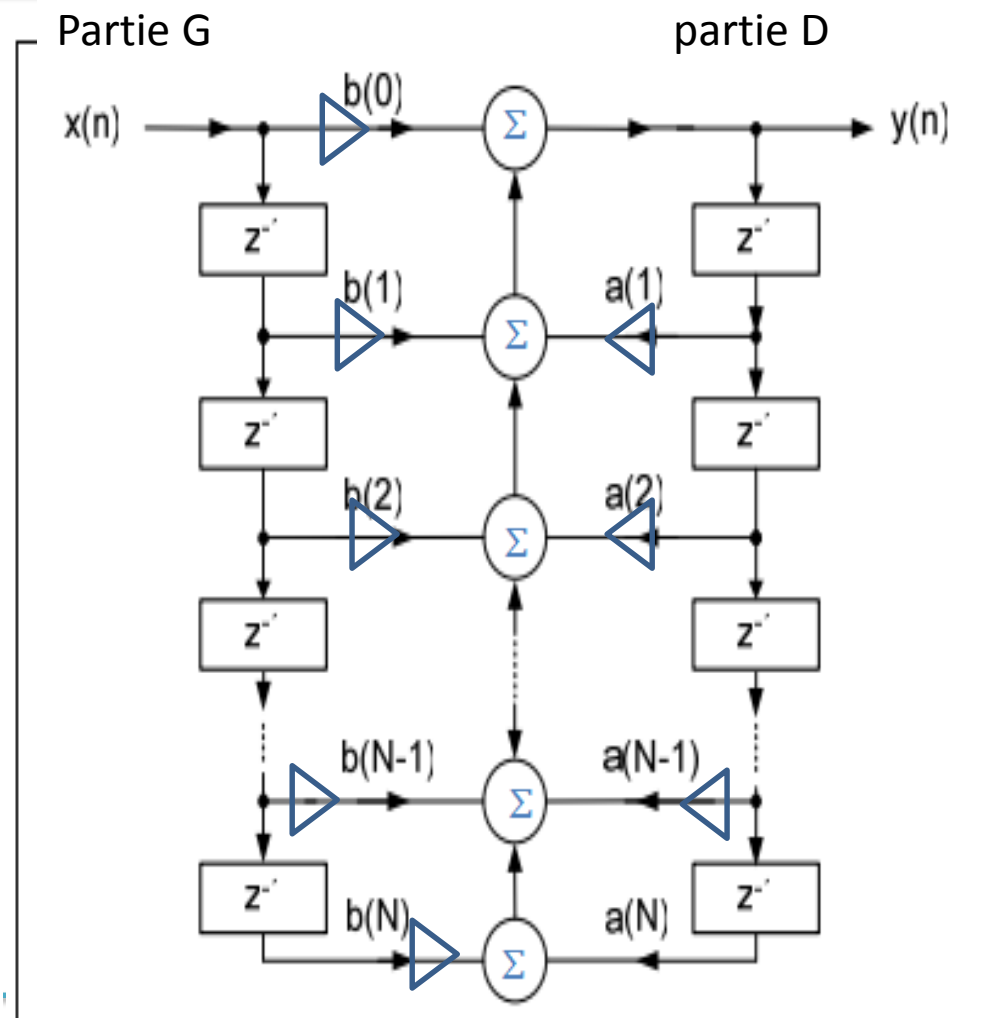
IIR: Forme directe

1 La partie **gauche** du flux de signaux implémente des **zéros** et la partie **droite** du flux de signaux réalise des **pôles**.

2 Nécessite **2N** unités de mémoire pour sauvegarder les variables d'état.

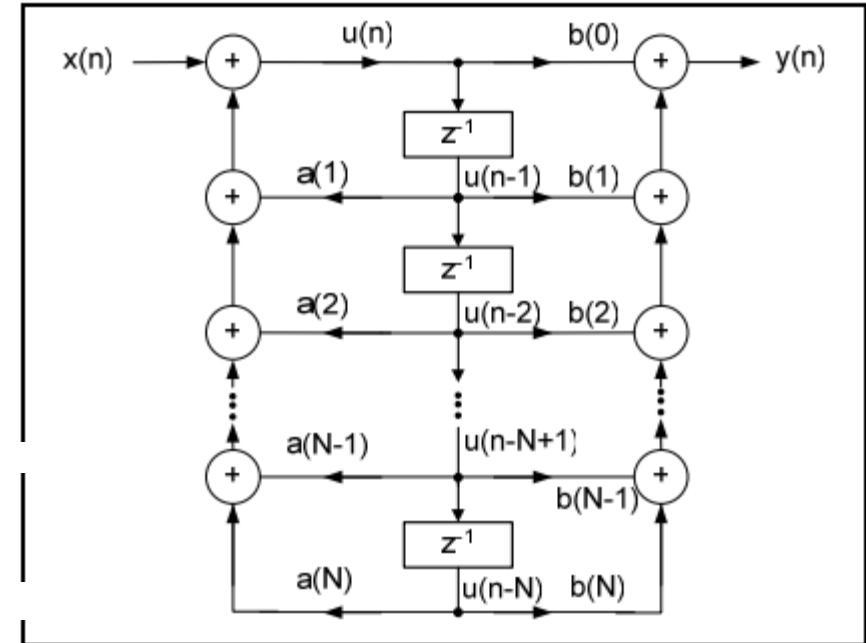
3 Partie G: non réursive

4 Partie D: réursive



4) Implémentation d'un filtre IIR

IIR: Forme directe 2



- ✚ La structure canonique directe de type 2 nécessite moins de cellules mémoire que la structure canonique directe de type 1.

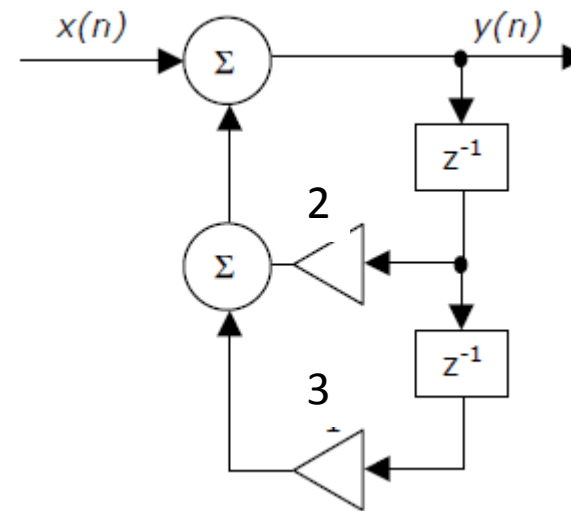
Activer Wi

4) Implémentation d'un filtre IIR

Exemple

Soit l'équation aux différences suivante:
 $y(n) = 2y(n-1) + 3y(n-2) + x(n)$

1) Représenter son schéma bloc.



5) Conception d'un IIR

Contrairement aux filtres RIF on ne part pas de la réponse impulsionnelle théorique d'un filtre analogique que l'on tronque mais directement de la fonction de transfert en (P) du filtre analogique. Cette fonction de transfert analogique n'étant pas tronquée, sa réponse impulsionnelle est entièrement conservée, c'est pourquoi on parle de filtre à réponse impulsionnelle infinie.

6) Méthodes de synthèse de filtres récurrents (IIR)

Il existe de nombreuses méthodes dont :

- La méthode de l'invariance impulsionnelle
- La méthode de l'invariance indicielle
- La méthode des pôles et des zéros
- La transformation bilinéaire

7) Synthèse de filtres RII par transformation bilinéaire

- **Principe**

On dispose d'un filtre continu ayant un gabarit fréquentiel répondant au cahier des charges demandées (synthèse de filtres analogiques de type Butterworth, Cauer, elliptique,...).

- **Objectif :**

Trouver le filtre numérique récursif ayant une réponse fréquentielle équivalente à celle du filtre analogique.

- **Méthode de la transformation bilinéaire**

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

8) Conception d'un filtre RII

Concevoir un filtre numérique consiste à déterminer la fonction de transfert en z du filtre qui va approcher au mieux les spécifications sur la réponse fréquentielle en amplitude

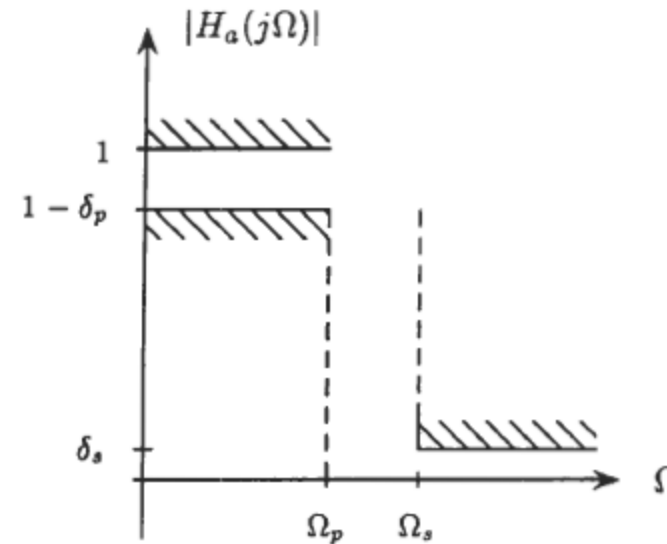
Toujours cinq étapes requises :

- 1. Spécification du filtre
- 2. Calcul des coefficients
- 3. Choix d'une architecture de mise en œuvre.
- 4. Simulation (conseillé)
- 5. Implémentation.

8) Conception d'un filtre RII

Étape 1 : spécification du filtre

- On utilise souvent un filtre passe bas de référence que l'on adapte
- Le filtre de référence doit être réalisable



8) Conception d'un filtre RII

Étape 2 : Calcul des coefficients

- Transformation bilinéaire

La conception d'un filtre digital à partir d'un filtre analogique nécessite la transformation

$$H(z) = H_a(s) \bigg|_{s = \frac{2}{T_s} \frac{1-z^{-1}}{1+z^{-1}}}$$

8) Conception d'un filtre RII

Liste des polynômes de Buterworth

N	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8
1	1.0000							
2	1.4142	1.0000						
3	2.0000	2.0000	1.0000					
4	2.6131	3.4142	2.6131	1.0000				
5	3.2361	5.2361	5.2361	3.2361	1.0000			
6	3.8637	7.4641	9.1416	7.4641	3.8637	1.0000		
7	4.4940	10.0978	14.5918	14.5918	10.0978	4.4940	1.0000	
8	5.1258	13.1371	21.8462	25.6884	21.8462	13.1372	5.1258	1.0000

Exemple: La fonction de transfert de Butterworth:

$$|H(j\Omega)|^2 = \frac{1}{1 + \left(\frac{\Omega}{\Omega_c}\right)^{2N}}$$

Conception de filtre pass bas de Buterworth : Méthodologie

1. Calculer la valeur des paramètres **k** (facteur de discrimination) et **d** (facteur de sélectivité)

$$K = \frac{\Omega_p}{\Omega_c}$$

$$d = \left[\frac{(1 - \delta_p)^{-2} - 1}{\delta_s^{-2} - 1} \right]^{1/2}$$

2. Déterminer l'ordre du filtre $N=?$

$$N \geq \frac{\log d}{\log k}$$

3. Déterminer la valeur de $\overline{\Omega_c}$ dans le rang

$$\Omega_p [(1 - \delta_p)^{-2} - 1]^{-1/2N} \geq \Omega_c \geq \Omega_s [\delta_s^{-2} - 1]^{-1/2N}$$

8) Conception d'un filtre RII

Conception de filtre pass bas de Buterworth

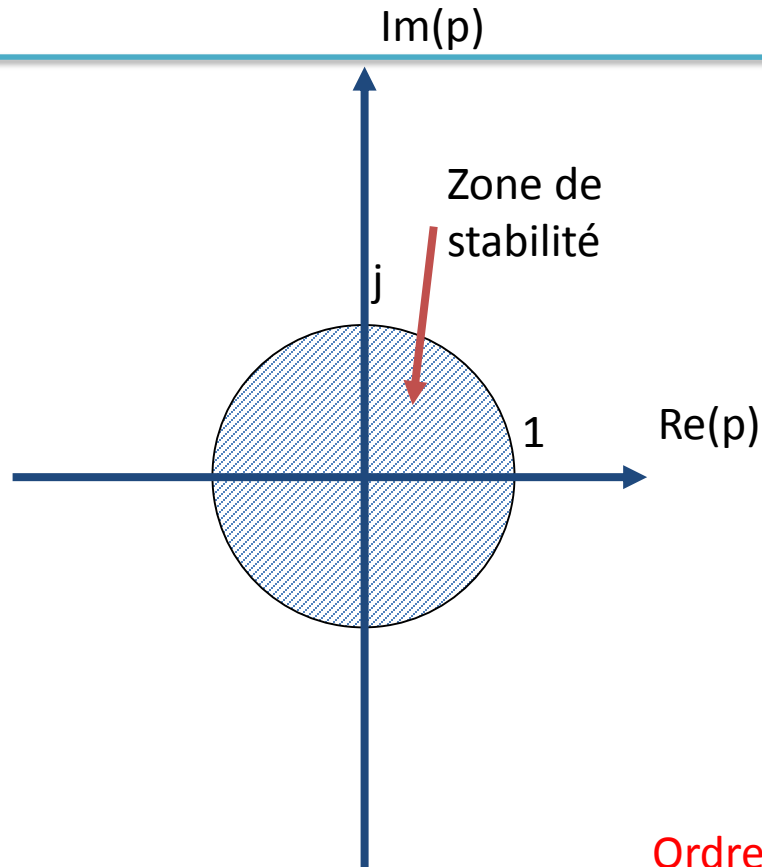
4. Avec la valeur de l'ordre, trouver la fonction de transfert $H_a(s)$, en substituant s par $\frac{s}{\Omega_c}$
5. Convertir la fonction analogique $H_a(s)$ à la fonction numérique en utilisant la méthode bilinéaire

8) Conception d'un filtre RII

Exemple: Soit à concevoir un filtre passe bas de type Butterworth par la méthode bilinéaire qui vérifie les spécifications suivantes :

$$f_p = 6\text{KHz} ; f_s = 10\text{KHz} ; \delta_p = \delta_s = 0.1$$

9) Filtres à minimum de phase



Le filtre est stable si:

$$\forall i, |p_i| < 1$$

Le filtre est à minimum de phase si:

$$\forall i, |z_i| < 1$$

Ordre du filtre = $\max(\text{nb pôles}, \text{nb zéros})$