Introducción a diseño de filtros FIR

Procesamiento de señales

.UBAfiuba FACULTAD DE INGENIERÍA

Ing. Débora P. Copa ● Dra. Cecilia G. Galarza
IIBM LPSC

Estructura de la práctica

- Transformada z
- Filtro ideal
- Filtros prácticos
- Actividad 1
- Filtros de fase lineal generalizada

- Herramientas de diseño mediante Toolbox
- Actividad 2
- Presentación TP1

Sistemas en tiempo discreto

- *Sistemas LTI*
- *Reales: (h(n) es Real para todo n)*
- *Transferencia Racional: H(z)=P(z)/Q(z)*
- *Causales*
- *Estables*

FFT (matlab)

Transformada de Fourier de tiempo discreto (TDFT)

$$X(\omega) = \sum_{n=-\infty}^{\infty} x[n] e^{-j\omega n}$$

Transformada z

$$X(z) = \sum_{n=-\infty}^{\infty} x[n] z^{-n}$$

$$x[n] \longrightarrow \boxed{h[n]} \longrightarrow y[n] = \sum_{k=-\infty}^{\infty} h[k]x[n-k]$$

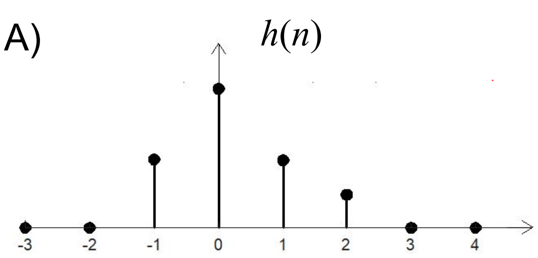
$$X(\omega) \longrightarrow \boxed{H(e^{j\omega})} \longrightarrow Y(\omega) = H(\omega)X(\omega)$$

$$X(z) \longrightarrow \boxed{H(z)} \longrightarrow Y(z) = H(z)X(z)$$

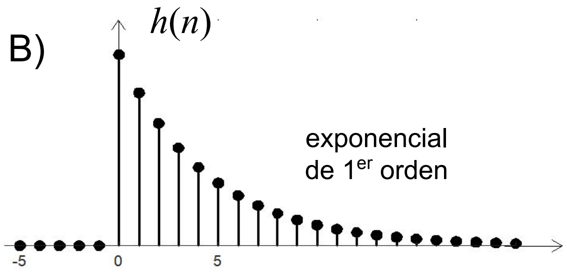
Transformada z   Repaso

Dadas las siguientes Respuestas impulsivas de un sistema LTI, Determine: Longitud de respuesta al impulso (FIR/IIR), Causalidad, Estabilidad, Transferencia H(z), Polos/ceros (en 0 o inf), ROC.

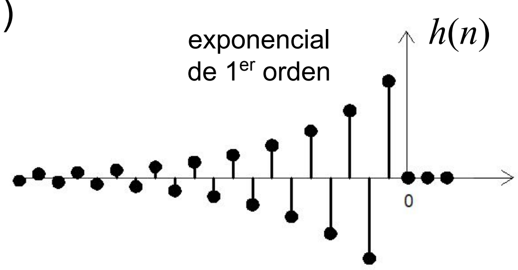
A)



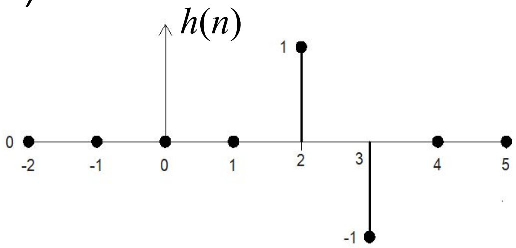
B)



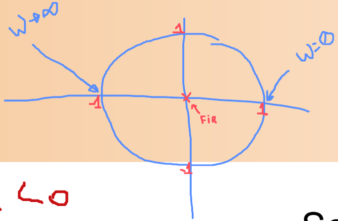
C)



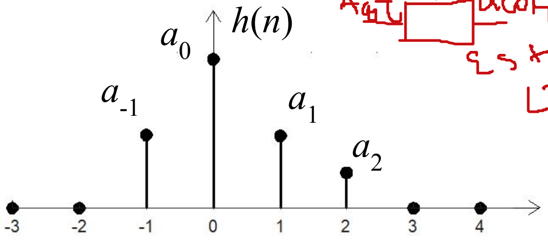
D)



Transformada z   Repaso



A) Respuesta impulsiva



C: $h(n)=0 \quad \forall n<0$

AC: $\forall n \geq 0$

BIBO

Est + LTI: $\sum |h(n)| < \infty$ ABS SUM

FIR ESTABLES {solo polos 0 ó ∞

$H(z) = a_{-1} z + a_0 + a_1 z^{-1} + a_2 z^{-2}$

$\boxed{z=0}$ $= z^{-2}\left(a_{-1} z^3 + a_0 z^2 + a_1 z + a_2\right)$ → a_2

∞ ← 2 Polos

2 Polos

$\boxed{z \to \infty}$ $H(z) = z\left(a_{-1} + a_0 z^{-1} + a_1 z^{-2} + a_2 z^{-3}\right)$ → a_{-1}

1 Polo

\# Polos = # ceros

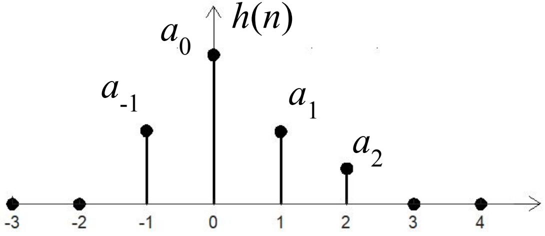
Solución

- FIR/IIR:
- Causalidad: NC (o bilat)
- Estabilidad: Si
- Polos en z=0: 2
- Polos en z→ ∞: 1
- Ceros en z=0:
- Ceros en z→ ∞:
- ROC:

FIR: Caus: Roc incluye ∞; Ac: roc incluye 0

Transformada z Repaso

A) Respuesta impulsiva

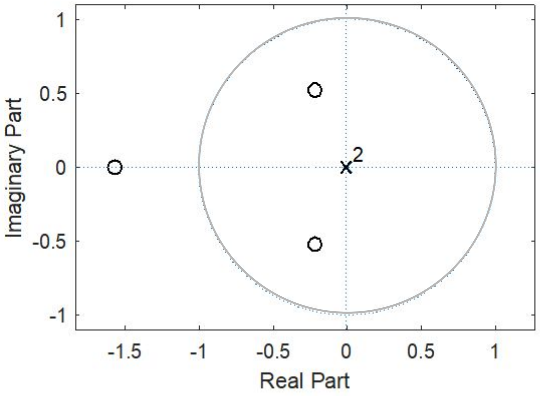


$$h(n) = a_{-1}\,\delta(n+1) + a_0 + a_1\delta(n\text{-}1) + a_2\,\delta(n\text{-}2)$$

Transformada z

$$H(z) = a_{-1}\,z + a_0 + a_1\,z^{-1} + a_2\,z^{-2}$$ **(irrealizable en tiempo real)**

Plano z = r.e^{jw}

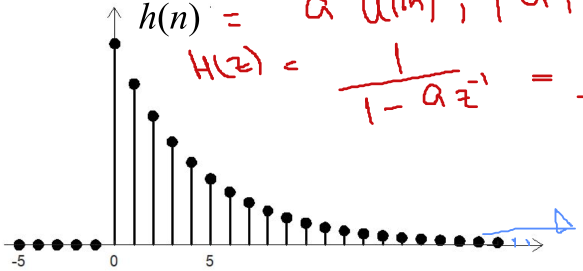


Solución

- **FIR/IIR:** FIR
- **Causalidad:** No Causal
- **Estabilidad:** Estable
- **Polos en z=0:** 2
- **Polos en z→ ∞:** 1
- **Ceros en z=0:** Ninguno
- **Ceros en z→ ∞:** Ninguno
- **ROC:** $\{z \in \mathbb{C} - \{0,\infty\}\}$

Transformada z   Repaso

B) Respuesta impulsiva:
(exponencial de 1er orden)



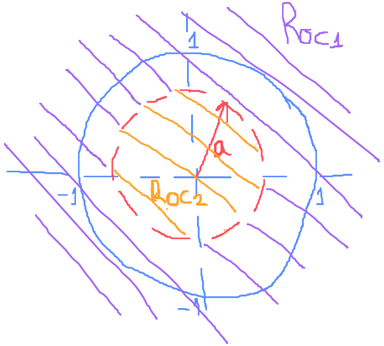
$h(n) = a^n u(n), \ |a| < 1$

$H(z) = \dfrac{1}{1 - a z^{-1}} = \dfrac{z}{z-a}$

$C = 0$.

$P = a < 1$

Causal
$$h(n) \begin{cases} > 0, & n \geq 0 \\ = 0, & n < 0 \end{cases}$$

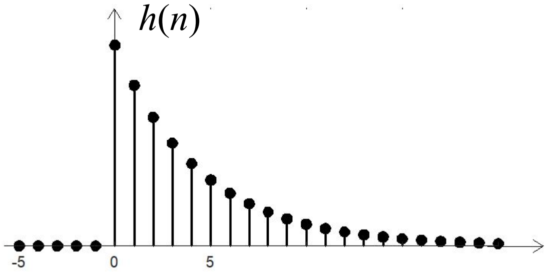


Solución

- **FIR/IIR:**
- **Causalidad:**
- **Estabilidad:**
- **Polos en z=0:**
- **Polos en z→ ∞:**
- **Ceros en z=0:** 1
- **Ceros en z→ ∞:**
- **ROC:** $|z| < a$

Transformada z Repaso

B) Respuesta impulsiva:
(exponencial de 1er orden)



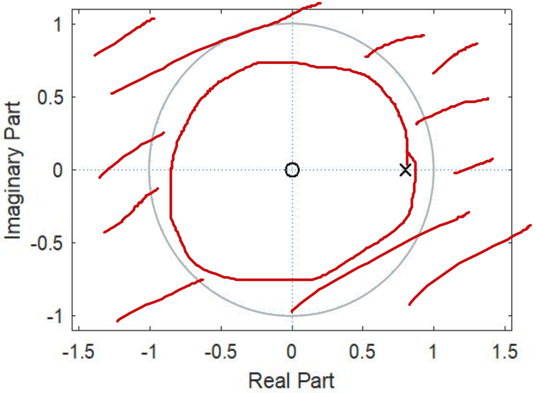
$h(n) = a^n\ u(n),\ a<1$

↓

Transformada z

$H(\mathrm{z}) = 1/(1 - a\mathrm{z}^{-1})$

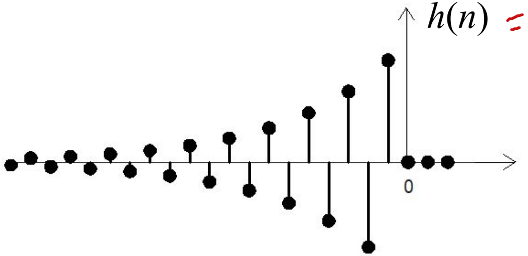
Plano $z = r.e^{jw}$



Solución

- **FIR/IIR:** IIR
- **Causalidad:** Causal
- **Estabilidad:** Estable
- **Polos en z=0:** Ninguno
- **Polos en z→ ∞:** Ninguno
- **Ceros en z=0:** 1
- **Ceros en z→ ∞:** Ninguno
- **ROC:** {z∈ℂ / |z|>|a|}

C) Respuesta impulsiva:
(exponencial de 1er orden)



$= -a^n u(-n-1)$

$a < 0$

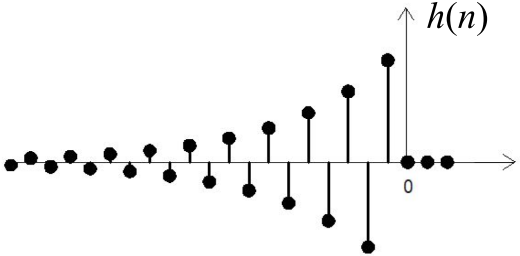
$|a| > 1$

Solución

- **FIR/IIR:**
- **Causalidad:** ×
- **Estabilidad:**
- **Polos en z=0:**
- **Polos en z→ ∞:**
- **Ceros en z=0:**
- **Ceros en z→ ∞:**
- **ROC:**

Transformada z Repaso

C) Respuesta impulsiva:
(exponencial de 1er orden)



$h(n) = -\,a^n\, u(-n-1),\ |a| > 1$

↓

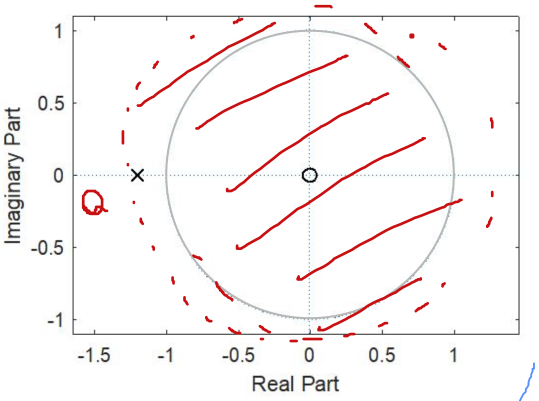
Transformada z

$H(z) = 1/(1 - az\text{-}1)$

$H(z) = \frac{z}{z-a}$

(irrealizable en tiempo real)

Plano z = r.e^{jw}



P = a > 1

C = 0

Solución

- **FIR/IIR:** IIR
- **Causalidad:** Anticausal
- **Estabilidad:** Estable BIBO
- **Polos en z=0:** Ninguno
- **Polos en z→ ∞:** Ninguno
- **Ceros en z=0:** 1
- **Ceros en z→ ∞:** Ninguno
- **ROC:** $\{z \in \mathbb{C}\ /\ |z| < |a|\}$

Transformada z Repaso

D) Respuesta impulsiva

$h(n) = a z^{-2} - a z^{-3} = a_1 z^{-3} (z-1)$

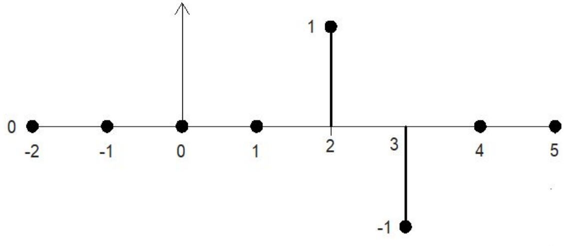
$z=0 \rightarrow$ 3 Polos

$\rightarrow a_1 z^{-2}(1 - z^{-1})$

$z \rightarrow \infty$

2 ceros

1 cero $z=1$

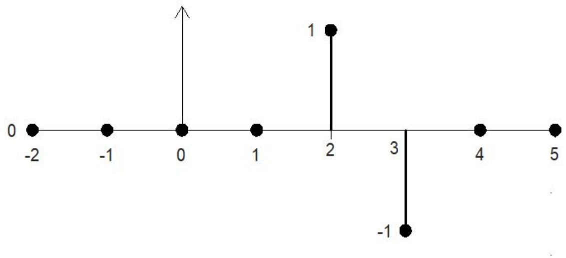


Solución

- **FIR/IIR:**
- **Causalidad:**
- **Estabilidad:**
- **Polos en z=0:**
- **Polos en z→ ∞:**
- **Ceros en z=0:**
- **Ceros en z→ ∞:**
- **ROC:**

Transformada z Repaso

D) Respuesta impulsiva

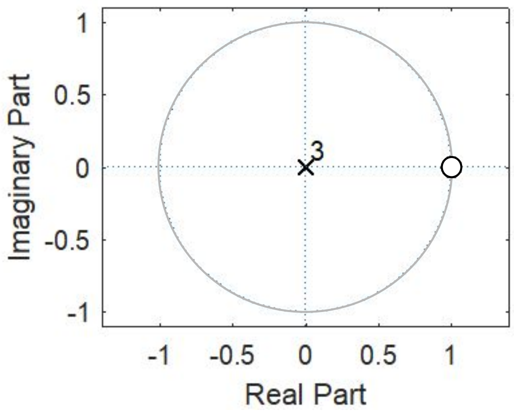


$$h(n) = a_1\,\delta(n\text{-}2) - a_2\,\delta(n\text{-}1)$$

Transformada z

$$H(z) = a_1\, z^{-2} - a_2\, z^{-3}$$

Plano z = $r.e^{jw}$



Solución

- **FIR/IIR:** FIR
- **Causalidad:** Causal
- **Estabilidad:** Estable
- **Polos en z=0:** 3
- **Polos en z→ ∞:** Ninguno
- **Ceros en z=0:** Ninguno
- **Ceros en z→ ∞:** 2
- **ROC:** {z∈ℂ -{0} }

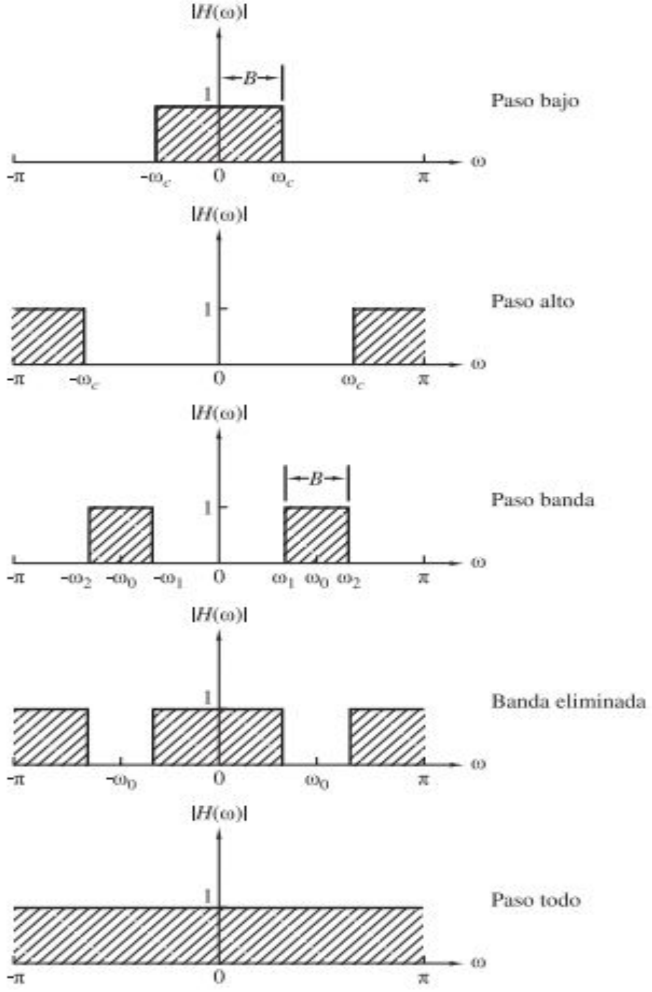
Filtros ideales

Filtros ideales

Idealización matemática de los filtros prácticos

- Ganancia constante (tomada normalmente como ganancia unidad) en la banda de paso.
- Ganancia cero en la banda eliminada.
- Hay saltos discontinuos entre bandas

No se pueden implementar físicamente



Filtro ideal

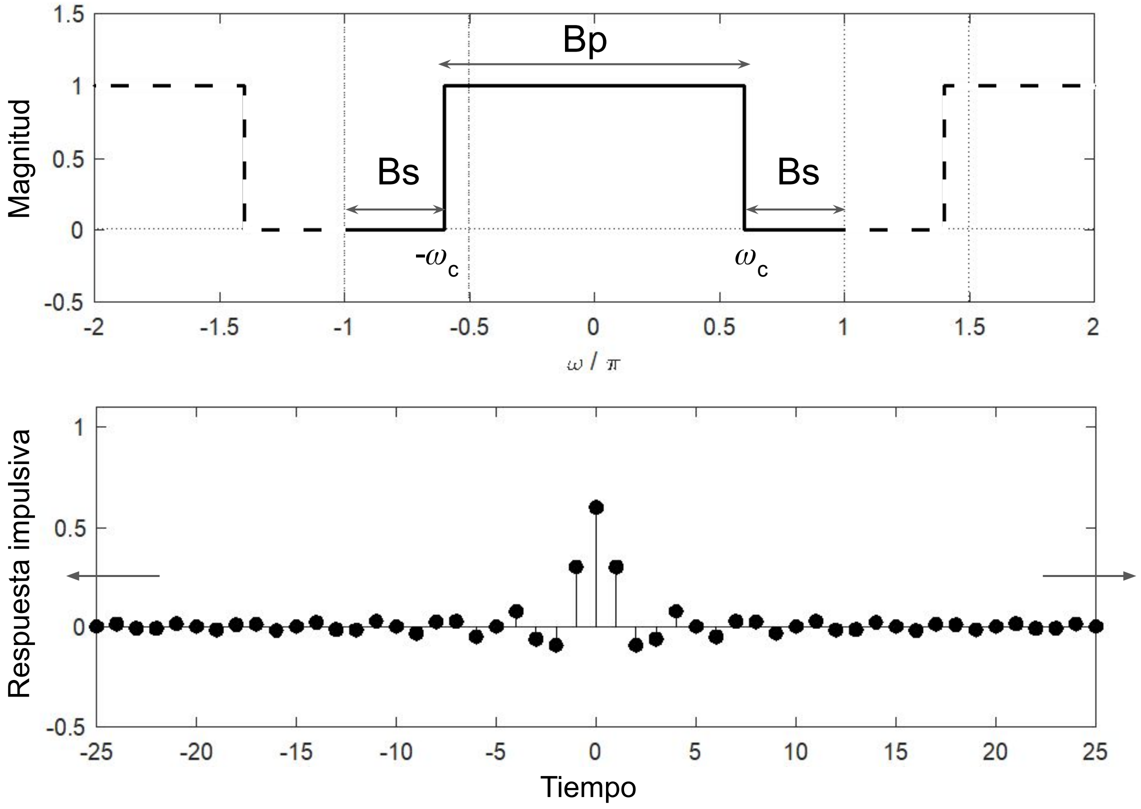
Respuesta temporal y en frecuencia

$$H_0(\omega) = \begin{cases} 1 & |\omega| \leq \omega_c \\ 0 & \forall \text{ otro } \omega \end{cases}$$

$\downarrow$ TDFT $^{-1}$

Respuesta impulsiva ideal (**no causal**)

$$h_0[n] = \frac{\sin(\omega_c n)}{\pi n} = \frac{\omega_c}{\pi} \operatorname{sinc}\left(\frac{\omega_c}{\pi} n\right)$$



Filtro ideal

Respuesta temporal y en frecuencia

muevo en $N/2$

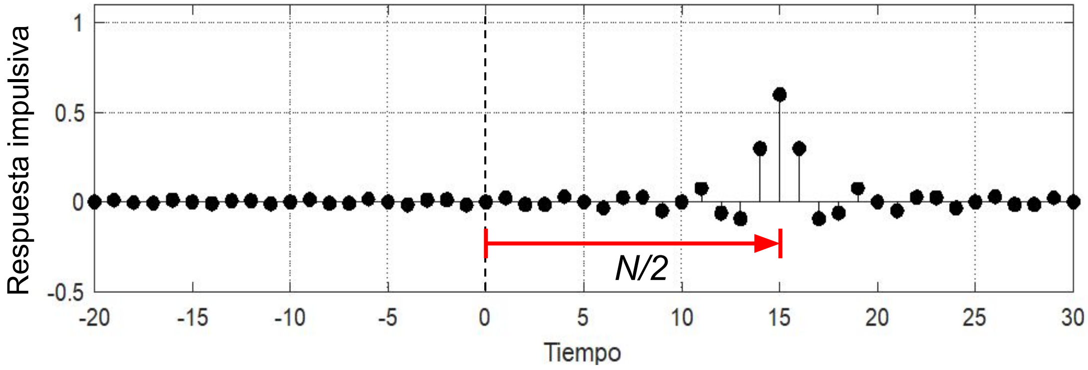
$$h_d[n] = \frac{\sin(\omega_c (n - N/2))}{\pi (n - N/2)}$$

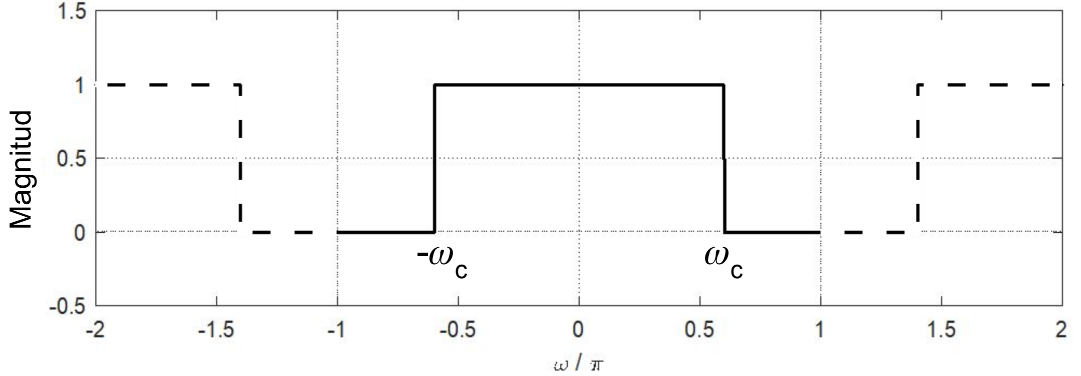
$$= \frac{\omega_c}{\pi} \operatorname{sinc}\left(\frac{\omega_c}{\pi}(n - N/2)\right)$$

TDFT

Filtro deseado (ideal) y desplazado en *N*/2

$$H_d(\omega) = \begin{cases} e^{-j\omega\frac{N}{2}} & |\omega| \leq \omega_c \\ 0 & \forall \text{ otro } \omega \end{cases}$$





Filtros Prácticos

Filtros prácticos

Truncado en el tiempo de un filtro ideal

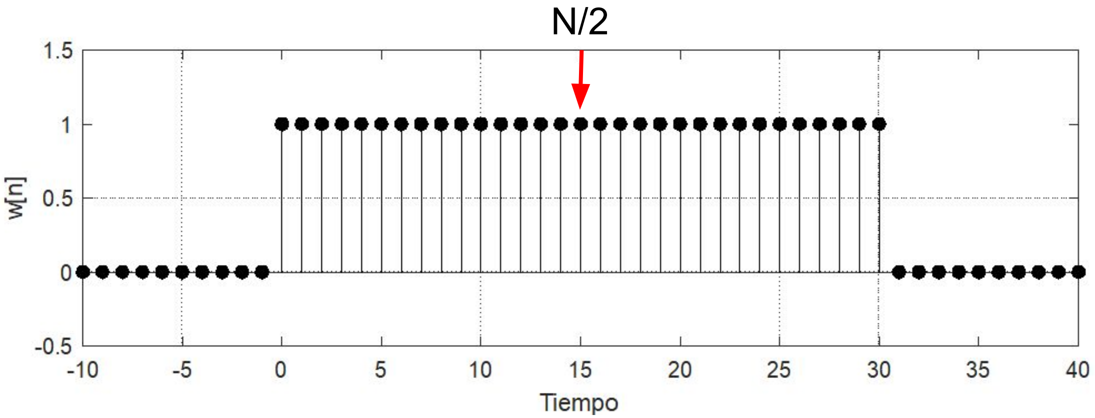
Ventana cuadrada causal

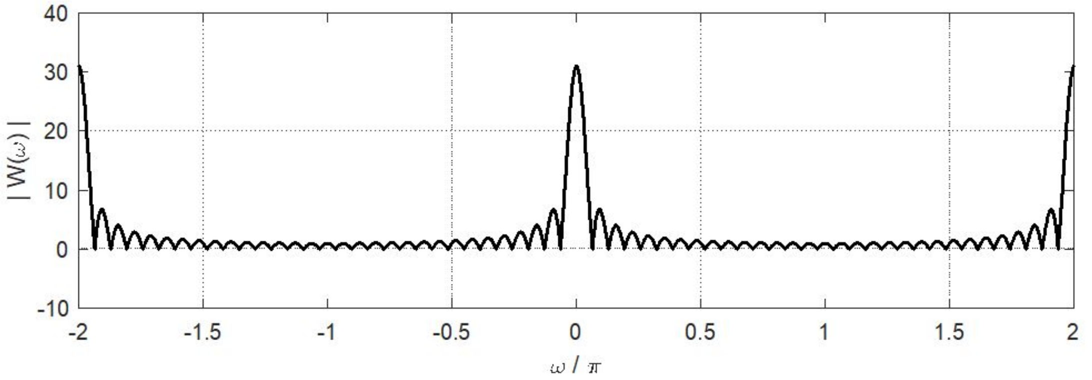
$$w[n] = \begin{cases} 1 & n \in [0, N] \\ 0 & \forall \text{ otro } n \end{cases}$$

TDFT

Respuesta de la ventana en frecuencia

$$W(\omega) = \frac{\sin\left(\omega\left(\frac{N+1}{2}\right)\right)}{\sin\left(\frac{\omega}{2}\right)} e^{-j\omega\frac{N}{2}}$$



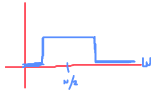


Filtros prácticos

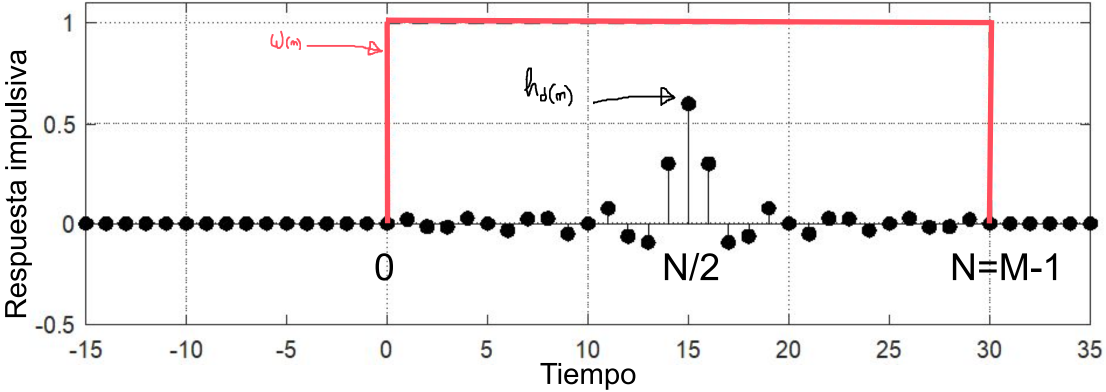
Truncado en el tiempo de un filtro ideal

Ventaneo de la respuesta impulsiva deseada

$$h_d[n] = \frac{\omega_c}{\pi}\,\mathrm{sinc}\left(\frac{\omega_c}{\pi}(n - N/2)\right) \xRightarrow{fft}$$



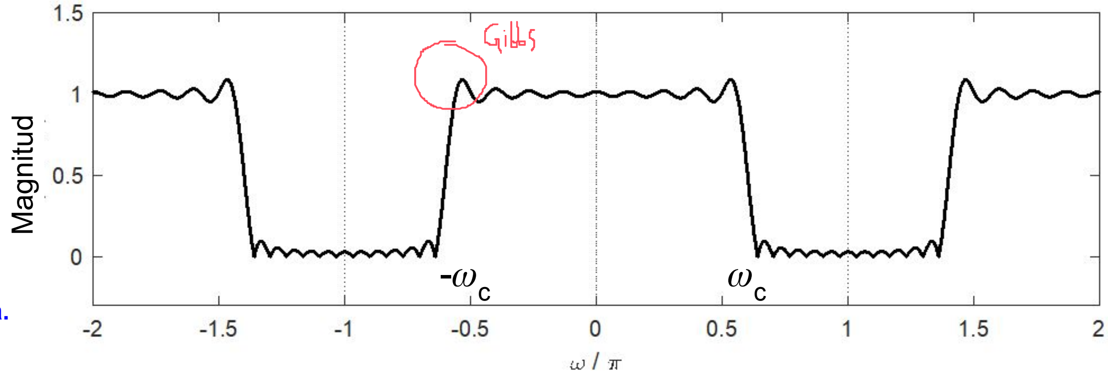
$$h[n] = h_d[n].w[n] = \begin{cases} h_d[n] & n \in [0, N] \\ 0 & \forall \text{ otro } n \end{cases}$$



TDFT

Respuesta en frecuencia del filtro ventaneado y causal

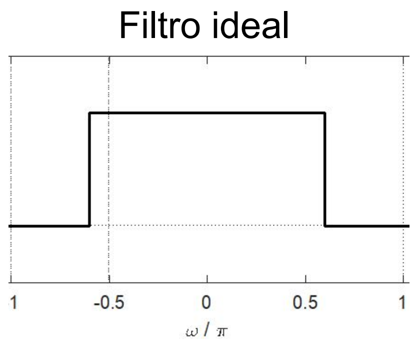
$$H(\omega) = \frac{1}{2\pi} H_d(\omega) * W(\omega)$$



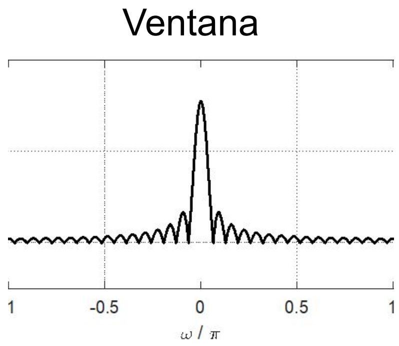
- ***N*: orden del polinomio de la función de transferencia.**
- ***M* = *N* + 1: Longitud del filtro.**

Filtros prácticos

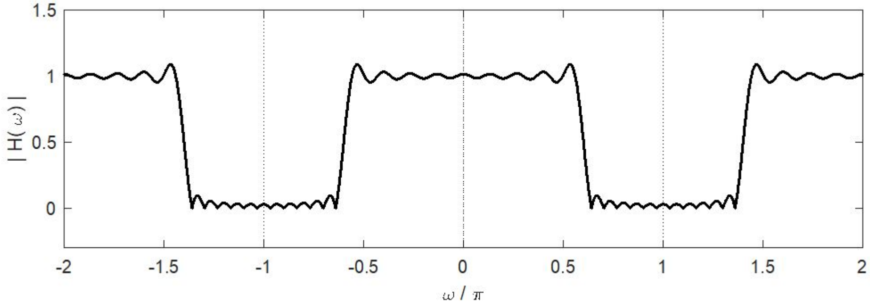
Truncado en el tiempo de un filtro ideal



*

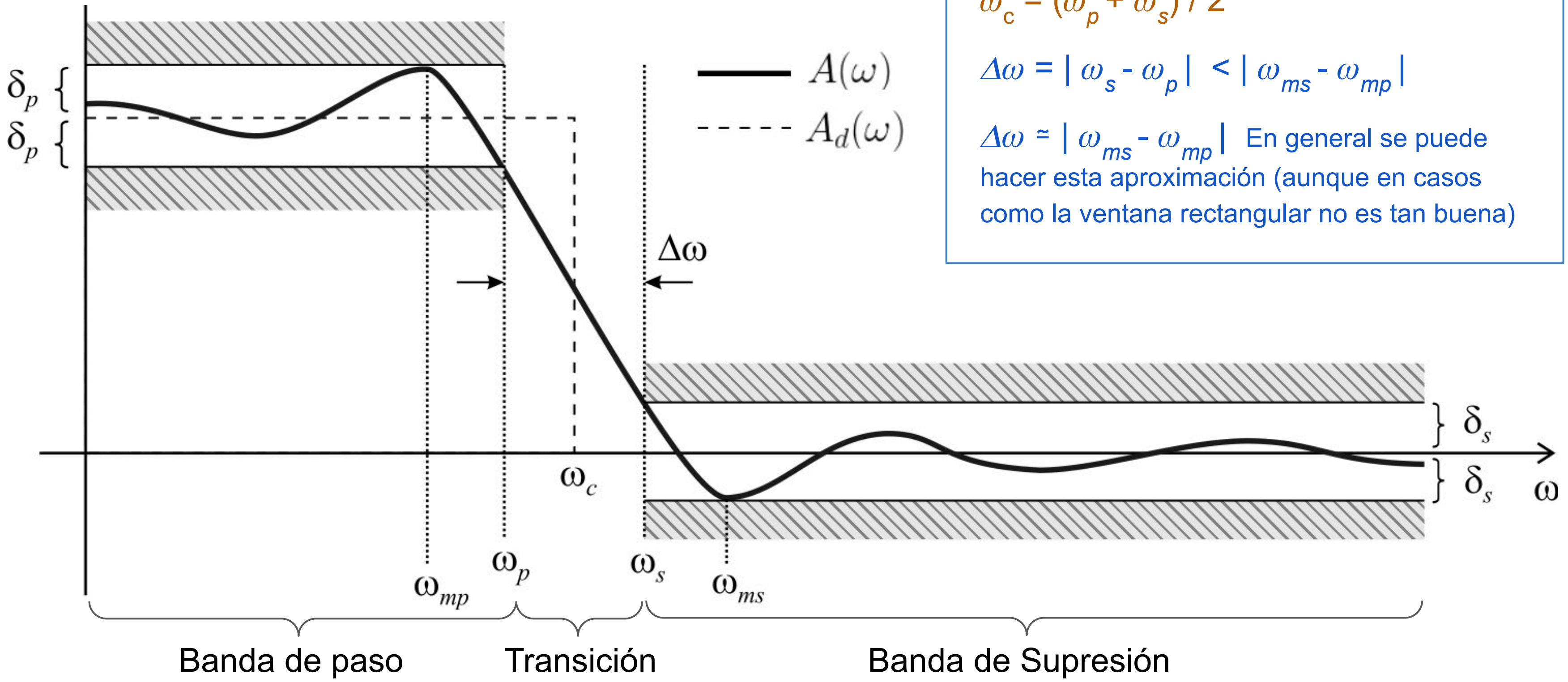


$$H(\omega) = \frac{1}{2\pi} H_d(\omega) * W(\omega)$$



Especificaciones de un filtro

Método de ventanas



$\omega_c = (\omega_p + \omega_s) / 2$

$\Delta\omega = |\omega_s - \omega_p| < |\omega_{ms} - \omega_{mp}|$

$\Delta\omega \simeq |\omega_{ms} - \omega_{mp}|$ En general se puede hacer esta aproximación (aunque en casos como la ventana rectangular no es tan buena)

Actividad 1

Actividad 1

Entorno de trabajo

MATLAB

GNU Octave

OCTAVE ONLINE

PYTHON

Actividad 1

Filtros FIR de fase lineal con ventana rectangular

1) Implementar en un filtro FIR pasa bajos (ejemplo anterior) de largo *M*=51 (es decir orden *N*=50), frecuencia de corte fc=2400 Hz y frecuencia de muestreo fs=7200 Hz.

   - Computar la respuesta impulsiva *h*(*n*)=*hd*(*n*).*w*(*n*) (usando la función *sinc()*)
   - Computar la respuesta en frecuencia (usando FFT de 8192 puntos)
   - Graficar *h*(*n*) (con *stem()*) y *H*(ω) (en módulo y con los ejes de frecuencia ω en [0,2π), o bien ω/π en [0,2)).

2) Graficar en un mismo *plot()* el módulo de H(ω) para los filtros definidos con los siguientes órdenes: *N*={50, 100, 200, 500}. Medir ripple y transición.

$$h(n) = h_d(n).w(n) = \begin{cases} \frac{\omega_c}{\pi} \operatorname{sinc}\left(\frac{\omega_c}{\pi}\left(n - \frac{N}{2}\right)\right) & n \in [0, N] \\ 0 & \forall \text{ otro } n \end{cases}$$

Actividad 1

Filtros FIR de fase lineal con ventana rectangular

Resumen de mediciones sobre los gráficos simulados

N	δ_p	δ_s	ω_{mp}	ω_{ms}	$\lvert \omega_{ms} - \omega_{mp} \rvert$
50					
100					
200					
500					

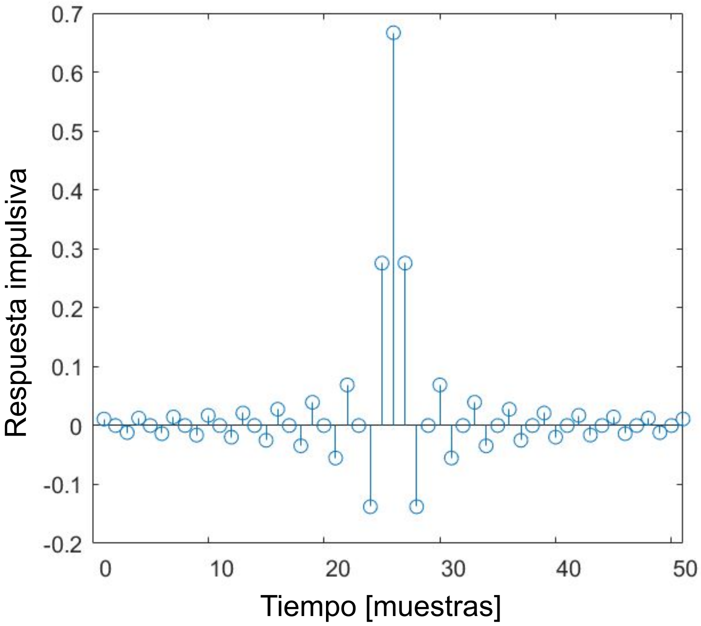
↑ Orden (N): $\underbrace{\delta_p \quad \delta_s}_{?}$ $\underbrace{\lvert \omega_{ms} - \omega_{mp} \rvert}_{?}$

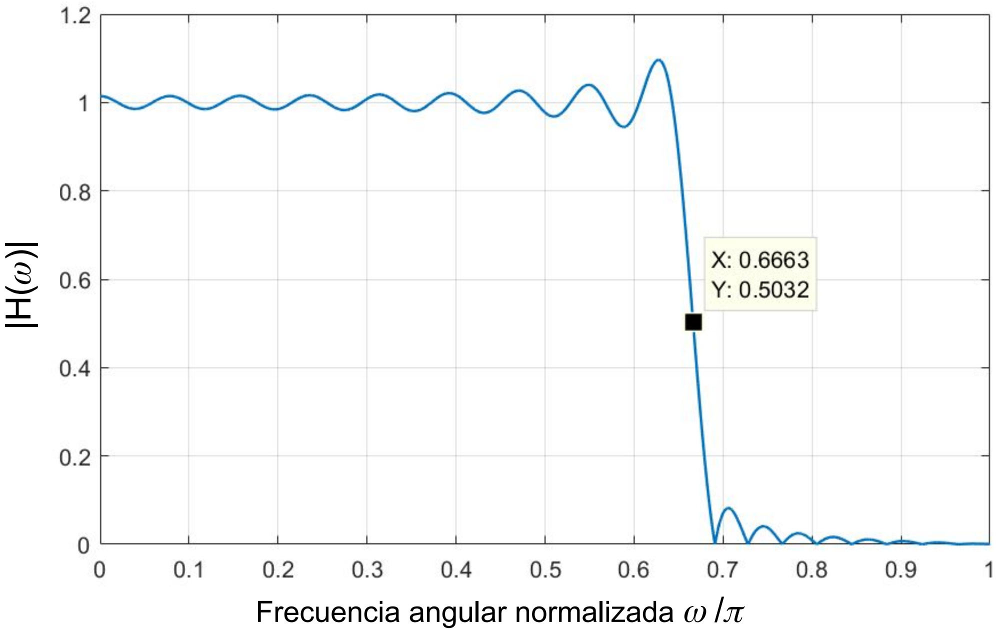
Actividad 1

Filtros FIR de fase lineal con ventana rectangular

1)

$$fc = \omega_c/(2\pi Ts) = 0{,}6663/(2Ts) = 2398{,}7 \text{ Hz}$$





Actividad 1

Filtros FIR de fase lineal con ventana rectangular

```
% * Implementar en Matlab un filtro FIR pasa bajos (ejemplo anterior) de largo
% L=51 (es decir M=50) y frecuencia de corte fc=2400 Hz.
clear all
close all

nfft=8192;% Largo de la fft
w=linspace(0,2*pi,nfft);% frecuencia angular: w en [0,2pi)

fc=2400; % Frecuencia de corte en[Hz] // Omega_c = 2*pi*fc;
fs=7200; % Frecuencia de muestreo en[Hz]
Ts=1/fs; % Tiempo de muestreo

wc=(2 * pi * fc)* Ts;  % Frec. de corte de tiempo discreto
M=51;                  % Largo del filtro
N=M-1;                 % Orden
```

Actividad 1

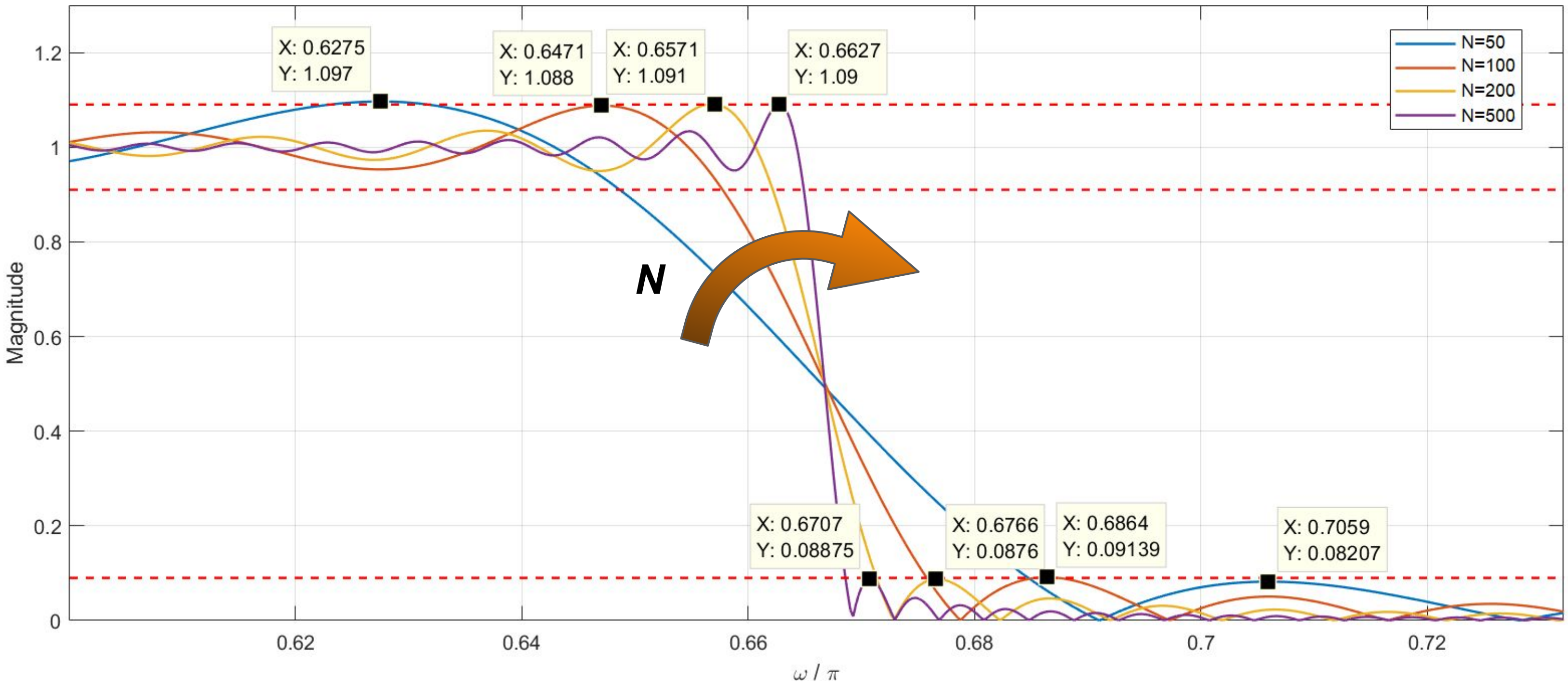
Filtros FIR de fase lineal con ventana rectangular

```
% * Computar la respuesta impulsiva h(n)=hd(n).w(n) (usando la función sinc() )
n=0:N; % M elementos de la ventana rectangular w(n)
h=wc/pi*sinc(wc/pi*(n-N/2));
stem(n,h);
xlabel('Tiempo [muestras]')
ylabel('Respuesta impulsiva')
grid on

figure
% * Computar la respuesta en frecuencia (usando FFT de 2048 puntos)
H=fft(h,nfft);
plot(w/pi,abs(H)); % con frecuencia angular normalizada: w en [0,2)
xlabel('Frecuencia angular normalizada \omega/\pi')
ylabel('|H(\omega)|')
xlim([0,1])
grid on
```

Actividad 1
Filtros FIR de fase lineal con ventana rectangular

2)



Actividad 1
Filtros FIR de fase lineal con ventana rectangular

2)

Resumen de mediciones sobre los gráficos simulados

N (orden)	δ_p	δ_s	ω_{mp}	ω_{ms}	$\lvert \omega_{ms} - \omega_{mp} \rvert$
50	0.097	0.082	0.627π	0.706π	0.078π
100	0.088	0.091	0.647π	0.686π	0.039π
200	0.091	0.088	0.657π	0.677π	0.020π
500	0.090	0.089	0.663π	0.671π	0.008π

$\approx$

$\Delta\omega$

$\uparrow$ Orden (N):

$\Delta\omega \propto \frac{1}{N}$

Actividad 1

Filtros FIR de fase lineal con ventana rectangular

2)

Resumen de mediciones sobre los gráficos simulados

N (orden)	δ_p	δ_s	ω_{mp}	ω_{ms}	$\lvert \omega_{ms} - \omega_{mp} \rvert$
50	0.097	0.082	0.627π	0.706π	0.078π
100	0.088	0.091	0.647π	0.686π	0.039π
200	0.091	0.088	0.657π	0.677π	0.020π
500	0.090	0.089	0.663π	0.671π	0.008π

↑ Orden (N):

Break

Filtros de fase lineal generalizada (FLG)

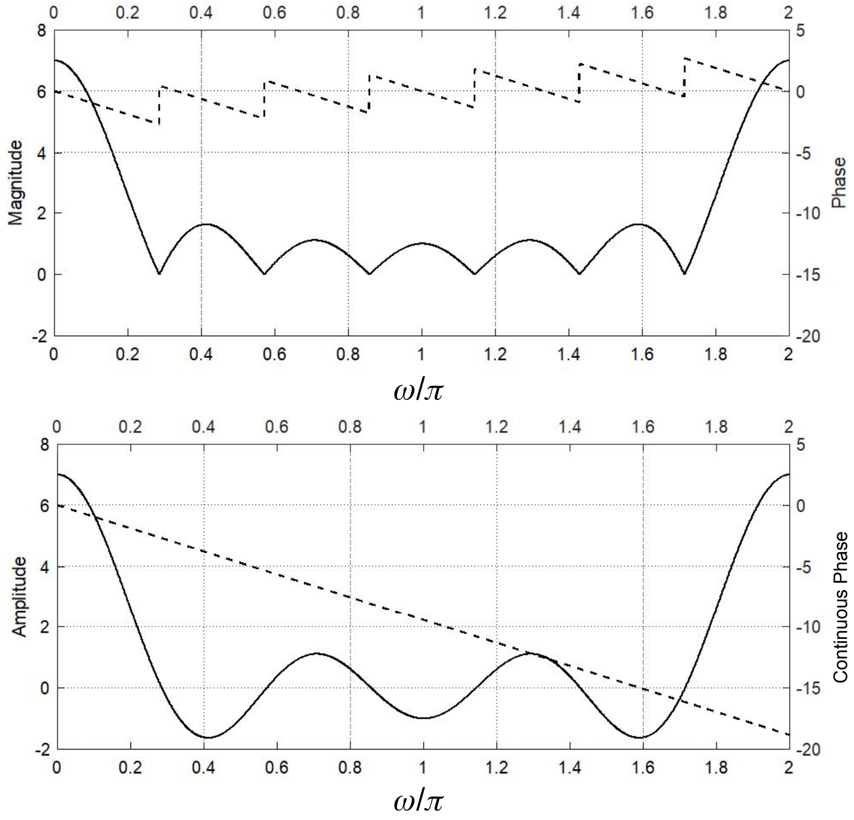
Representación de la respuesta en frecuencia
Función Amplitud y Fase Continua

Módulo y fase (al tomar el módulo, se pueden generar saltos de π en la fase en cada cambio de signo de amplitud)

$$H(\omega) = |H(\omega)|e^{j\theta(\omega)}$$

Función Amplitud (o *Función Real*) y *Fase* ***Continua*** (evita discontinuidades de π)

$$H(\omega) = A(\omega)e^{j\phi(\omega)}$$



Filtros de fase lineal generalizada (FLG)

Expresión general

$$H(\omega) = A(\omega)e^{j\phi(\omega)} = A(\omega)e^{j(-\omega\tau+\phi_0)}$$ Expresión general

$$\phi(\omega) = -\omega\tau + \phi_0$$

$$\downarrow$$

$$\tau_g(\omega) = -\frac{d\phi(\omega)}{d\omega}$$

$$\downarrow$$

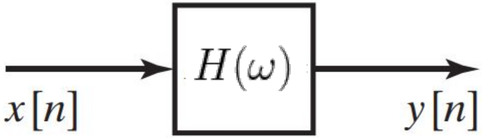
$$\boxed{\tau_g(\omega) = \tau = cte}$$

Para ser FLG debe cumplir ciertas restricciones:
(N : orden)

$$\begin{cases} \tau = \dfrac{N}{2} & \begin{cases} K \in \mathbb{Z} & \text{(N par)} \\ K + \frac{1}{2} & \text{(N impar)} \end{cases} \\ \phi_0 & \begin{cases} 0 & \text{(h(n) sim)} \\ \dfrac{\pi}{2} & \text{(h(n) antisim)} \end{cases} \end{cases}$$

Filtros de fase lineal

Retardo de grupo

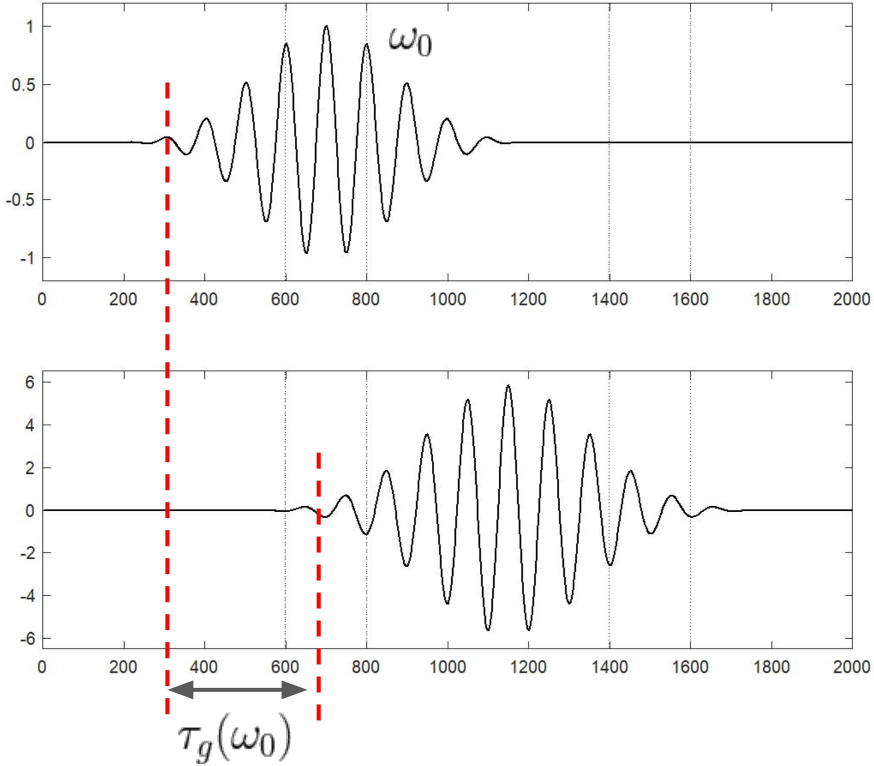


$$x[n] = s[n]\cos(\omega_0 n).$$

Pendiente de la curva de fase

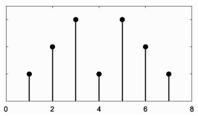
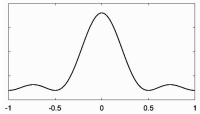
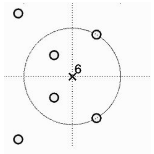
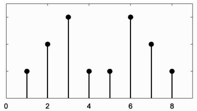
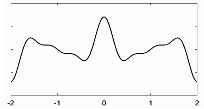
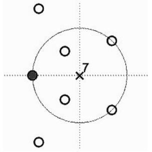
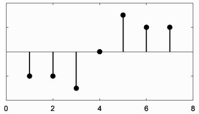
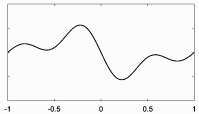
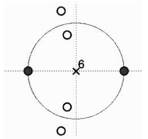
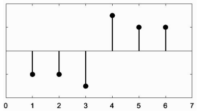
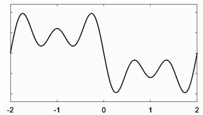
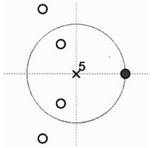
$$\tau_g(\omega) = -\frac{d\phi(\omega)}{d\omega}$$

Número de muestras que se desplaza la salida del filtro para una dada frecuencia



Filtros de fase lineal generalizada (FLG)

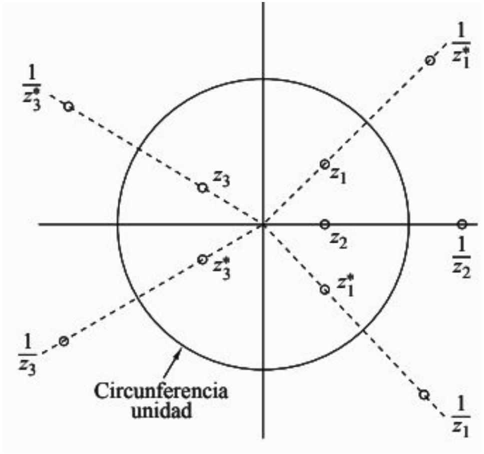
Resumen (M=N+1)

Tipo FLG	Parámetros	Simetría *respuesta impulsiva*	Simetría *función amplitud*	Ceros en z = 1 o z = -1
I	N par (*largo impar*) $\phi_0 = 0$	Simétrica: $h(n) = h(N-n)$ 	Simétrica: $A(\omega) = A(-\omega)$ periodo 2π 	Si no contiene, implica tipo I 
II	N impar (*largo par*) $\phi_0 = 0$	Simétrica: $h(n) = h(N-n)$ 	Simétrica: $A(\omega) = A(-\omega)$ periodo 4π 	Si es tipo II Cero en z=-1 
III	N par (*largo impar*) $\phi_0 = \pi/2$	Antisimétrica: $h(n) = -h(N-n)$ 	Antisimétrica: $A(\omega) = -A(-\omega)$ periodo 2π 	Si es tipo III Cero en z=1 y z=-1 
IV	N impar (*largo par*) $\phi_0 = \pi/2$	Antisimétrica: $h(n) = -h(N-n)$ 	Antisimétrica: $A(\omega) = -A(-\omega)$ periodo 4π 	Si es tipo IV Cero en z=1 

Filtros de fase lineal generalizada (FLG)

Resumen (M = N+1)

Continuación...

Tipo FLG	Ubicación de los ceros en general	Función amplitud
I	Si, $z=z_i$ es un cero de H(z), entonces también son ceros: $z=z_i^*$ $z=1/z_i$ $z=1/z_i^*$	$A(\omega) = h(\frac{M-1}{2}) + 2\sum_{n=0}^{\frac{M-3}{2}} h(n)\cos\omega\left(\frac{M-1}{2} - n\right)$
II		$A(\omega) = 2\sum_{n=0}^{\frac{M}{2}-1} h(n)\cos\omega\left(\frac{M-1}{2} - n\right)$
III		$A(\omega) = 2\sum_{n=0}^{\frac{M-3}{2}} h(n)\sin\omega\left(\frac{M-1}{2} - n\right)$
IV		$A(\omega) = 2\sum_{n=0}^{\frac{M}{2}-1} h(n)\sin\omega\left(\frac{M-1}{2} - n\right)$

Herramientas de diseño mediante Toolbox (Matlab)

Herramienta de diseño
Diseño de Filtros Digitales mediante un *TOOLBOX*

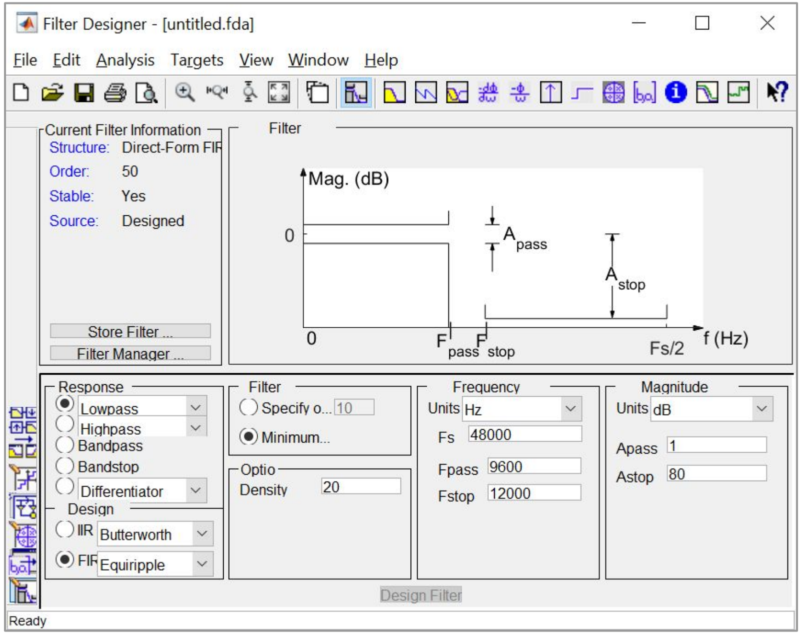
Ejemplo: Vamos a usar ***filterDesigner*** para diseñar un filtro pasabajos utilizando los mismos parámetros del ejemplo anterior:

- Frecuencia de corte fc = 2400 Hz,
- Frecuencia de muestreo fs = 7200 Hz
- Orden del filtro N=200
- Ventana rectangular
- Explorar otras opciones

** Destildar la opción "Scale Passband"*

Diseño de Filtros digitales

FDATOOL / *FILTER DESIGNER* (toolbox de Matlab)



Actividad 2

Actividad 2

Explorando los métodos de diseño y sus propiedades mediante el *toolbox*

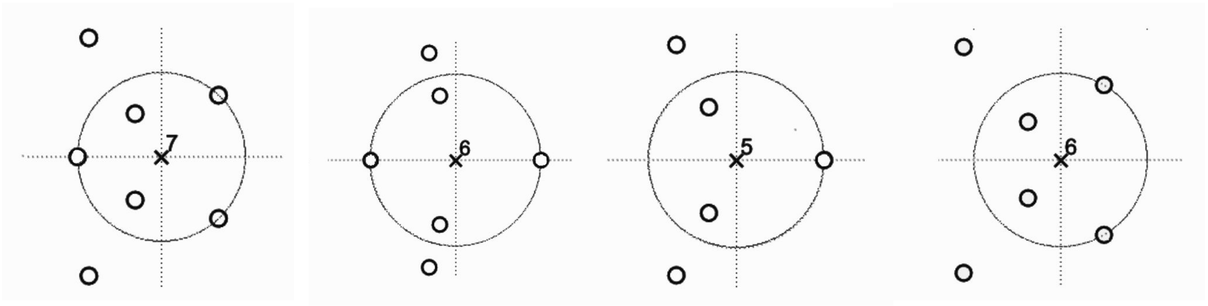
En el fdatool seleccione la opción de editor de polos y ceros

1) Colocando 2 polos complejos conjugados cercanos a:
   a) z=1
   b) z=-1

   Observe la respuestas de $h(n)$ y $A(\omega)$, tanto en su estabilidad como el comportamiento (LP, HP y BP)

(tenga en cuenta que se grafica la respuesta causal)

2) Genere los diagramas similares a los siguientes de manera que se garantice filtros FLG. Observe $h(n)$ y $A(\omega)$ e indique tipo de filtro.



Trabajo Práctico 1

Presentación de Trabajo Práctico 1

“Diseño de filtros digitales”

- Se requiere implementar un filtro FIR (FLG) mediante el método ***equirriple*** (que veremos en las siguientes clases).
- El problema consiste en definir las especificaciones del filtro, de acuerdo a ciertos requerimientos, para eliminar interferencias (de banda angosta) presentes en una señal de audio, implementar el filtro y verificar su desempeño.
- Para obtener la respuesta impulsiva se puede utilizar el algoritmo de Parks-McClellan usando la función de Matlab, Octave, etc.
- El trabajo es en grupos de dos personas (les daremos acceso a una planilla para que se anoten).
- El trabajo deberá entregarse el 8/09 y se resolverá en clase el mismo día de entrega.