

Filtros Digitales

Introducción a los Filtros Digitales
Filtros digitales vs. Filtros analógicos
Clasificación de filtros -Repaso
Características de los filtros prácticos
Filtro Digital Básico
Propiedades de los Filtros
Ecuación General de un Filtro
Filtros IIR
Filtro FIR
FIR vs IIR
Diseño de Filtros -FIR método de la ventana
Ejemplos en Python
Herramientas

Introducción a los Filtros Digitales

Un filtro digital, es un filtro que opera sobre señales digitales. Es una operación matemática que toma una secuencia de números (la señal de entrada) y la modifica produciendo otra secuencia de números (la señal de salida) con el objetivo de resaltar o atenuar ciertas características. Puede existir como una fórmula en un papel, un loop en un programa de computadora, como un circuito integrado en un chip.

Por ejemplo

- Disminuir el nivel de ruido
- Separar dos o más señales que están mezcladas
- Ecualizar la señal
- Restaurar la señal (eliminar desenfoque o artefactos de grabación)

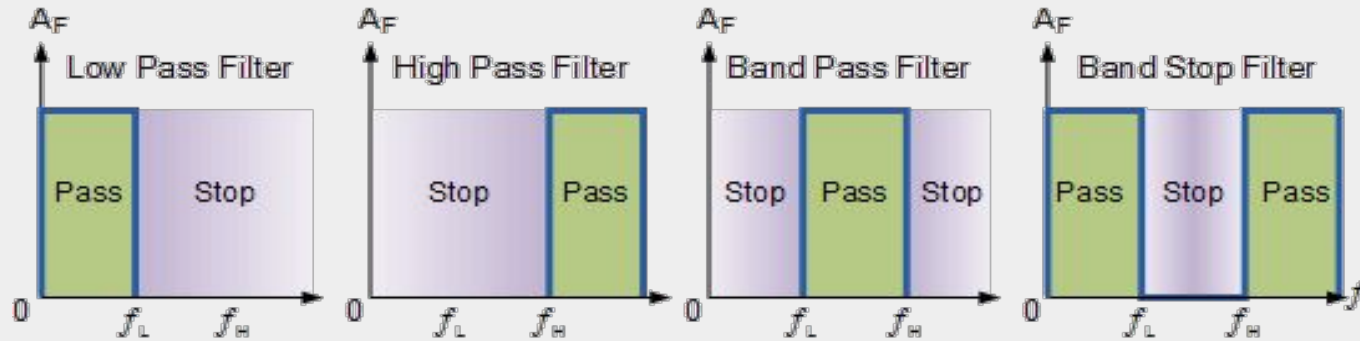
Llamamos **filtro digital** a los filtros aplicados a señales digitales y hablamos de **señal filtrada** para referirnos a la salida del filtro

Filtros digitales vs. Filtros analógicos

- Los filtros digitales pueden tener características que son imposibles de conseguir con filtros analógicos, como por ejemplo una respuesta de fase exactamente lineal.
- El desempeño de los filtros digitales no varía con las condiciones ambientales (temperatura, humedad, etc.) como sí ocurre con los filtros analógicos, lo que elimina la necesidad de calibrarlos periódicamente.
- Si el filtro se implementa utilizando un procesador programable la respuesta en frecuencia de un filtro digital puede ajustarse a voluntad (filtrado adaptivo). El mismo filtro puede procesar varias señales o canales de entrada sin necesidad de replicar el hardware.
- Las señales filtradas y sin filtrar pueden almacenarse para uso o análisis posterior.
- Los avances en las técnicas de integración VLSI hacen que sea posible fabricar filtros digitales pequeños, de bajo consumo, y de muy bajo costo.
- La precisión con que un filtro digital verifica las especificaciones de diseño está limitada solamente por la longitud de palabra (bits) utilizada para representar los coeficientes del filtro y ejecutar las operaciones aritméticas; con los filtros analógicos es difícil lograr atenuaciones que excedan los 60 o 70 dB en la banda de rechazo (utilizando componentes convencionales).
- Los filtros digitales pueden utilizarse a muy bajas frecuencias, como las que se encuentran en aplicaciones biomédicas, donde el empleo de filtros analógicos es poco práctico por los valores muy elevados de los componentes pasivos involucrados (capacitores, inductancias). Además, los filtros digitales pueden trabajar sobre un amplio rango de frecuencias simplemente cambiando la frecuencia de muestreo.

Clasificación de filtros -Repaso

Esta clasificación será de acuerdo a la parte del espectro que dejan pasar o no !!!



De izquierda a derecha tenemos:

- Filtro pasa bajo: Anula las frecuencias altas. Sirve para suavizar
- Filtro pasa alto: Anula las frecuencias bajas. Sirve para detectar cambios
- Filtro pasa banda: Anula todo excepto una banda continua de frecuencias
- Filtro rechaza banda: Anula sólo una banda continua de frecuencias

Las llamamos “ideales” por que en general los cortes de los filtros no pueden ser tan abruptos como se muestra en la figura

Otras Clasificaciones serían:

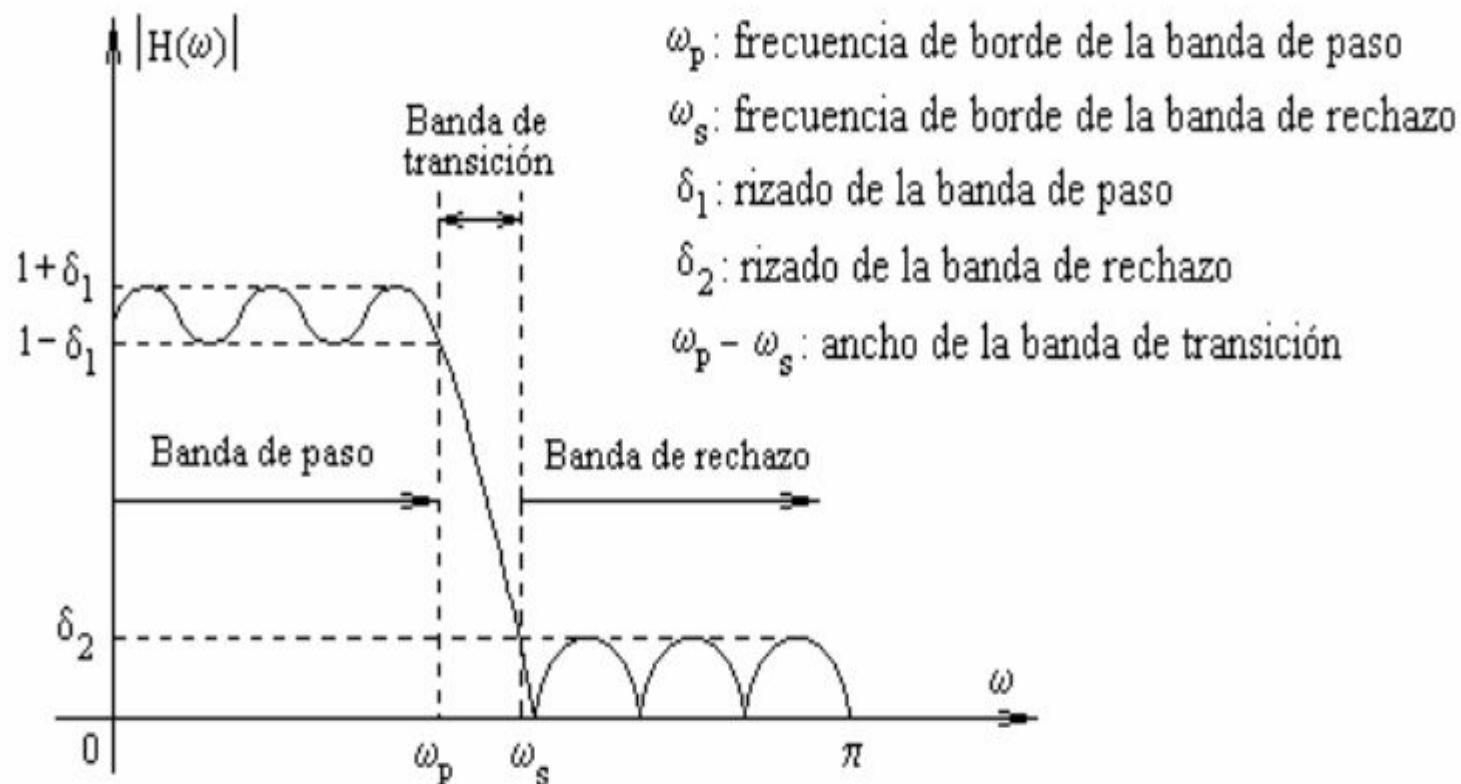
De acuerdo con su orden:

- primer orden
- segundo orden, etc.

De acuerdo con el tipo de respuesta ante entrada unitaria:

- FIR (Finite Impulse Response)
- IIR (Infinite Impulse Response)
- TIIR (Truncated Infinite Impulse Response)

Características de los filtros prácticos



No siempre la amplitud $|H(\omega)|$ es una constante, se puede tolerar un pequeño rizado en la banda de paso.

En un problema de diseño de filtro usualmente se especifican:

- 1) El rizado máximo tolerable en la banda de paso δ_1 .
- 2) El rizado máximo tolerable en la banda de rechazo δ_2 .
- 3) La frecuencia ω_p
- 4) La frecuencia ω_s

Las especificaciones de ganancia y de rizados máximos se suelen dar en dB, y se definen entonces:

función de pérdida: $A(\omega) = -20 \log H(\omega) \text{ [dB]}$

rizado máximo de la banda de paso: $\alpha_p = -20 \log(1 - \delta_1) \text{ [dB]}$

atenuación mínima de la banda de rechazo: $\alpha_r = -20 \log(\delta_2) \text{ [dB]}$

db



EQUATION 14-1

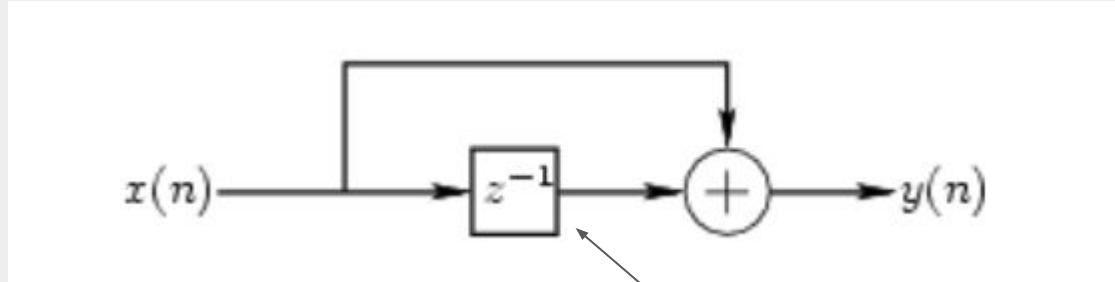
Definition of decibels. Decibels are a way of expressing a *ratio* between two signals. Ratios of power (P_1 & P_2) use a different equation from ratios of amplitude (A_1 & A_2).

$$\text{dB} = 10 \log_{10} \frac{P_2}{P_1}$$

$$\text{dB} = 20 \log_{10} \frac{A_2}{A_1}$$

60dB	=	1000
40dB	=	100
20dB	=	10
0dB	=	1
-20dB	=	0.1
-40dB	=	0.01
-60dB	=	0.001

Filtro Digital Básico



Línea o bloque de retardo

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

Forma General

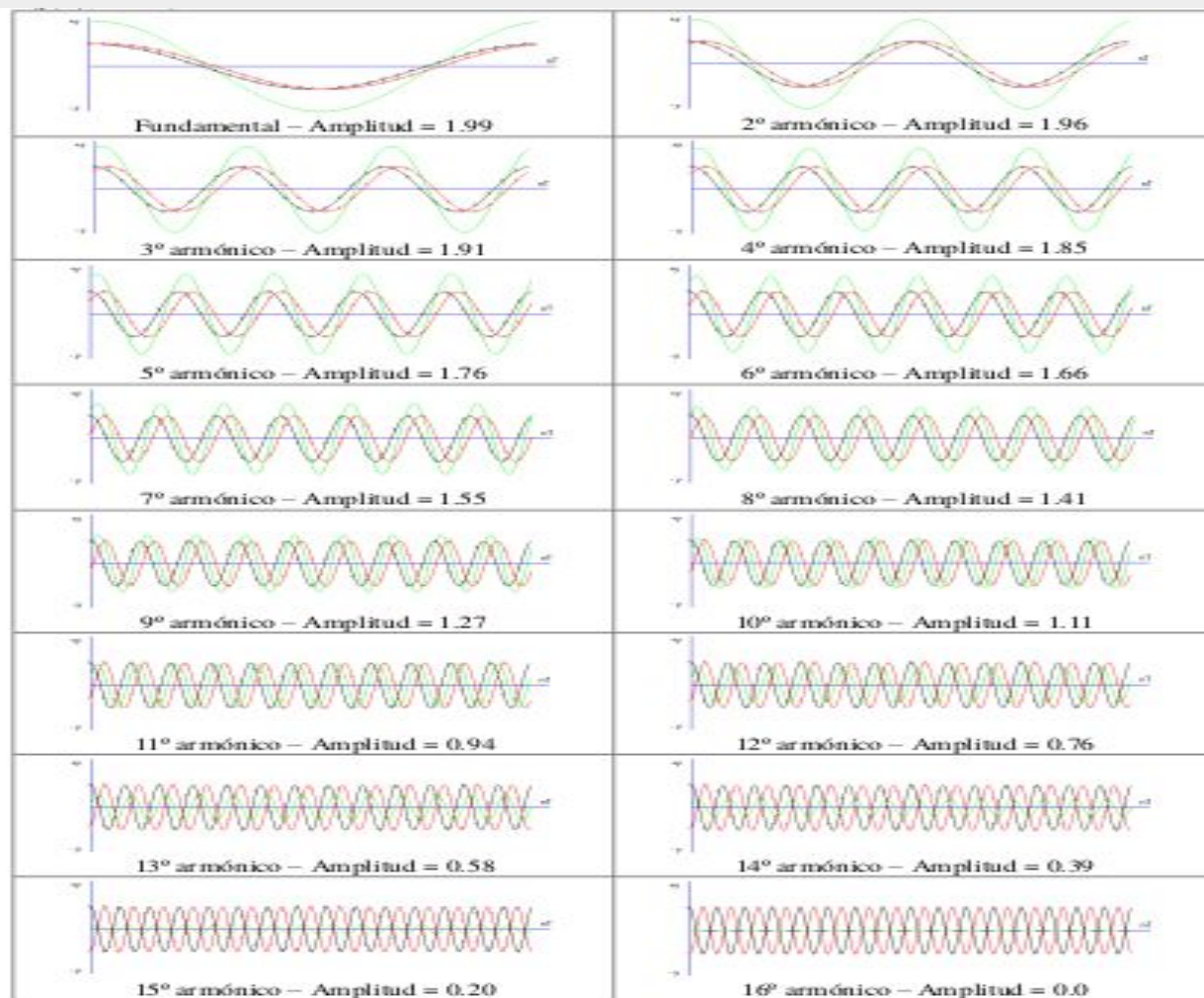


$$y(n)=x(n)+g*x(n-r)$$

r: cantidad de muestras de retardo
g: Constante

Recordar que $n=n*T$, estamos considerando $T=1$, muestras por segundo

Ver código `ejemplobasico.py`



Respuesta en frecuencia del filtro pasa bajo aplicando una señal senoidal discreta

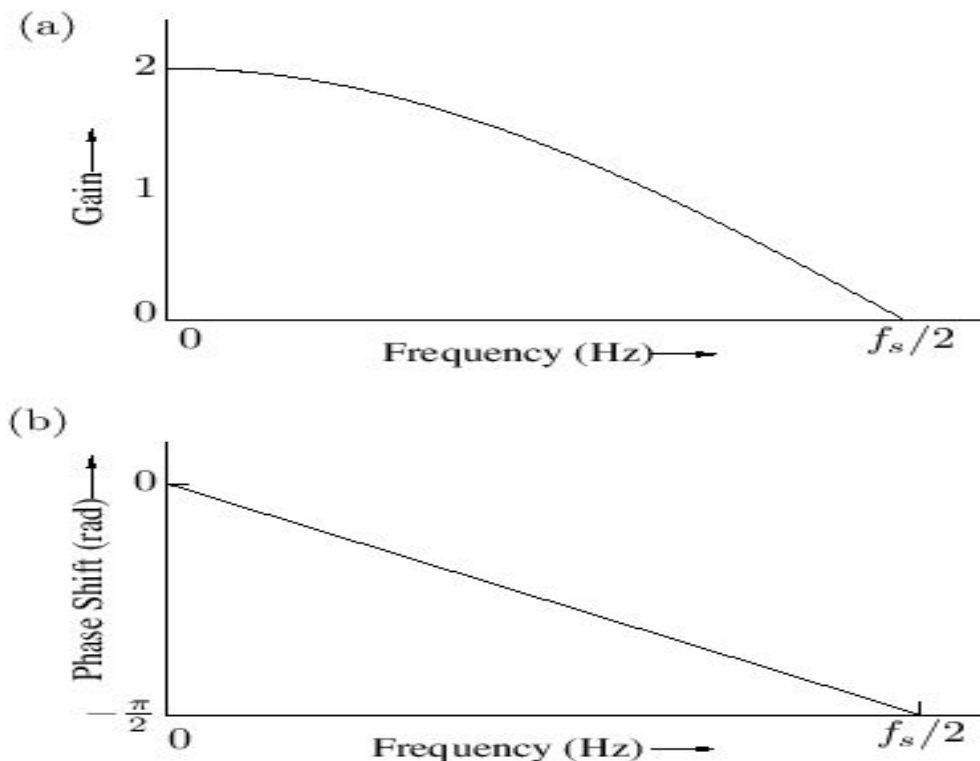


Figura 1.7: Respuesta de frecuencia del filtro . (a) Respuesta de amplitud. (b) Respuesta de fase.

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

Se puede demostrar que la ganancia en función de la frecuencia de este filtro cuando se le aplica una señal Senoidal es:

$$\boxed{G(\omega) = 2 \cos(\pi f T)} \quad |f| \leq \frac{f_s}{2}$$

Y la ecuación para la respuesta en fase sería:

$$\boxed{\Theta(\omega) = -\omega T/2.}$$

Con estas funciones, podemos predecir la salida del filtro para cualquier entrada sinusoidal. Dado que, según la teoría de Fourier, cada señal se puede representar como una suma de sinusoides, también hemos resuelto el problema más general de predecir la salida dada *cualquier* señal de entrada.

Propiedades de los Filtros

Los filtros que trataremos en este curso se denominan lineales e invariantes en el tiempo

Es Lineal cuando cumple que:

- La amplitud de la salida es siempre proporcional a la amplitud de la entrada.
- Si sumamos dos señales y las ingresamos al filtro, la salida es la misma que la suma de las señales de entrada filtradas por separado.

Un filtro es invariante en el tiempo cuando ejecuta siempre la misma operación, independientemente del tiempo transcurrido.

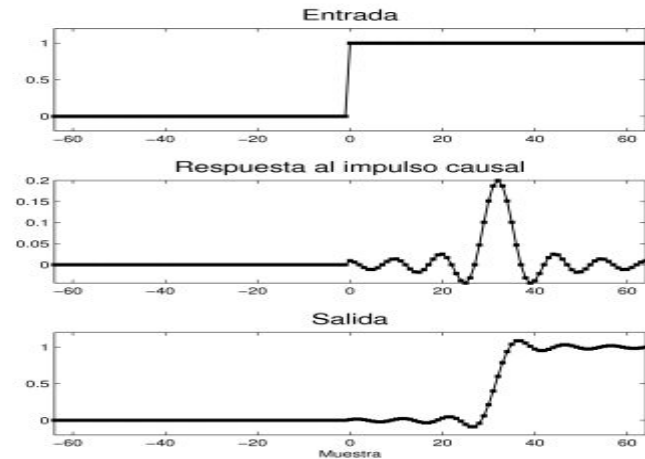
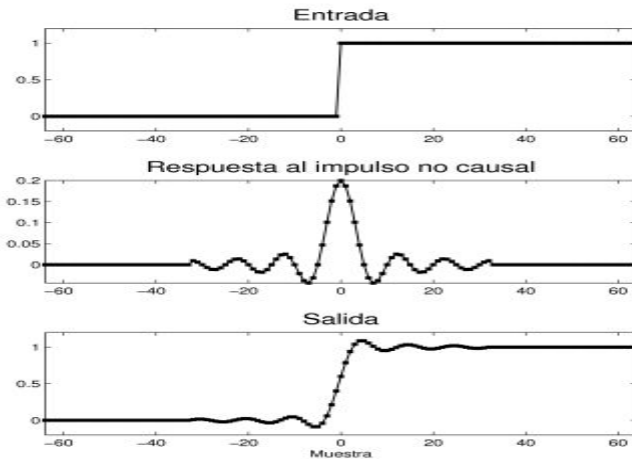
Filtro Causal y No Causal

Decimos que un filtro es causal cuando su salida no depende de muestras de entrada “futuras”. Los filtros vistos anteriormente son causales.

Un filtro no causal es el siguiente:

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

en el cual $(n + 1)$ alude a una muestra futura. Los filtros no causales poseen ciertas limitaciones cuando se emplean en tiempo real.



Un filtro es causal si cada efecto en la salida ocurre luego de la causa correspondiente en la entrada.

Condición para causalidad: $h[n] = 0$ si $n < 0$

Si esta condición no se cumple, la salida depende de muestras futuras de la entrada:

$$\begin{aligned}y[n] &= (h * x)[n] = \sum_{k=-\infty}^{\infty} h[k]x[n - k] \\&= \dots + h[-2]x[n + 2] + h[-1]x[n + 1] + h[0]x[n] + h[1]x[n - 1] + \dots\end{aligned}$$

Por ejemplo, la décima muestra de la salida se calcula como:

$$y[10] = \dots + h[-2]x[12] + h[-1]x[11] + h[0]x[10] + h[1]x[9] + \dots$$

Observaciones

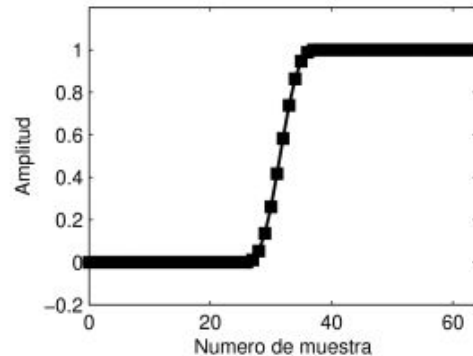
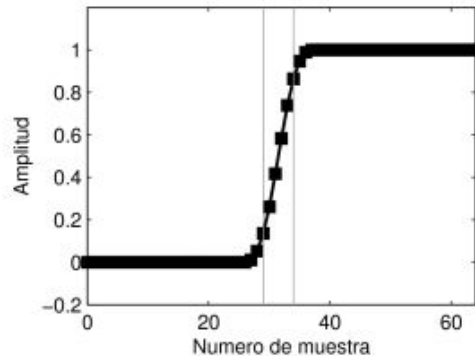
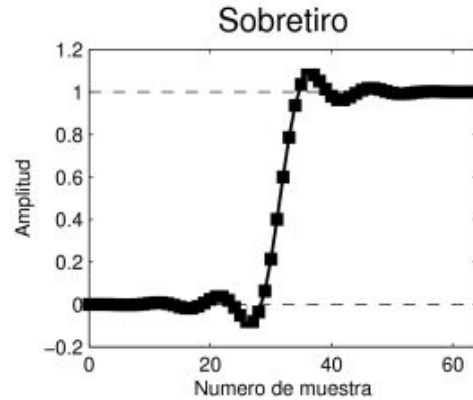
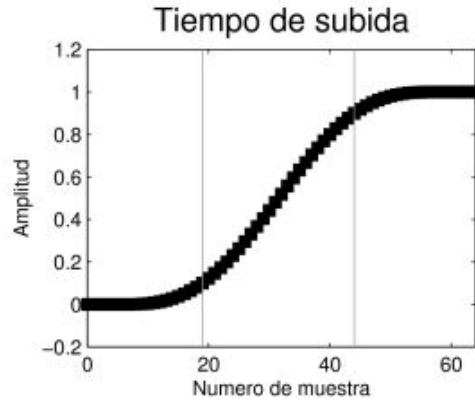
- Los filtros no causales son *irrealizables* en la práctica. No es posible construir un filtro no causal que opere en tiempo real.
- Cuando se trabaja en una computadora, la señal de entrada y de salida del filtro son secuencias de números almacenadas en memoria. En este caso, la salida puede depender de cualquier muestra de la entrada.
- **Retardo de los filtros causales.** Los filtros causales producen un retardo de la salida respecto a la entrada. Si la respuesta al impulso del filtro es simétrica, el retardo es la muestra del centro de simetría.

Filtros Recursivos y NO Recursivos

Un filtro **no recursivo** sólo emplea muestras de la señal de entrada ($x(n - 1)$, por ejemplo) para calcular la salida: **$y(n) = x(n) + x(n - 1)$**

Un filtro **recursivo** es aquel que, además de usar las muestras de entrada, utiliza muestras ya calculadas de su salida ($n - 1$), por ejemplo) para generar la muestra actual. Estos últimos poseen realimentación (feedback), vale decir, reutilizan la salida: **$y(n) = x(n) + y(n - 1)$**

Parámetros en el dominio del Tiempo

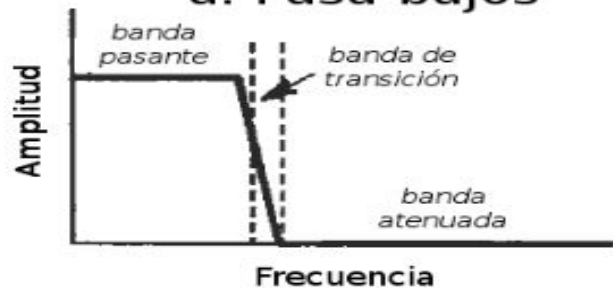


Tiempo de subida (step response): Es la cantidad de muestras en pasar del 10 % al 90% de amplitud de la respuesta al escalón. El tiempo de subida debe ser rápido para identificar eventos cercanos en el tiempo.

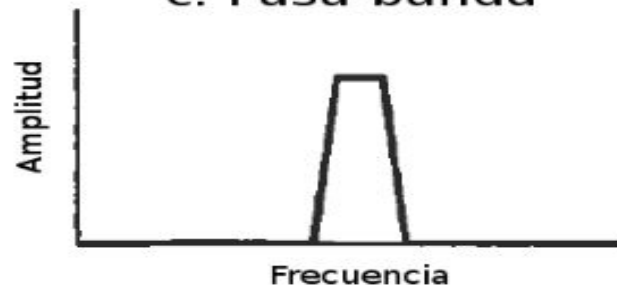
Sobretiro (Overshoot): Amplitud del pico que sobrepasa la amplitud del escalón. Sobretiros grandes producen distorsión en la forma de onda.

Respuesta en frecuencia

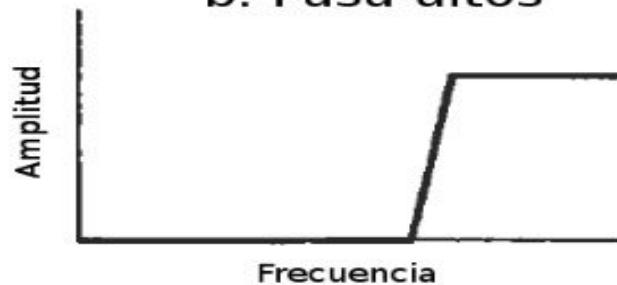
a. Pasa-bajos



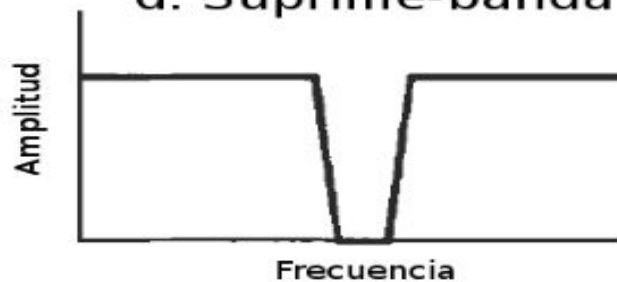
c. Pasa-banda



b. Pasa-altos

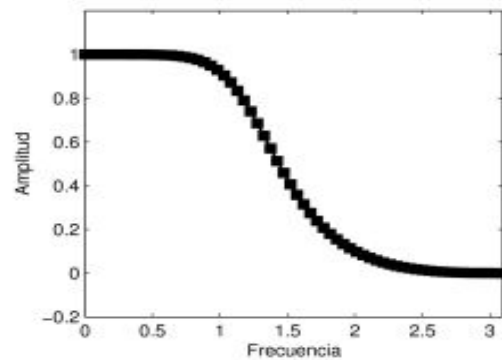


d. Suprime-banda

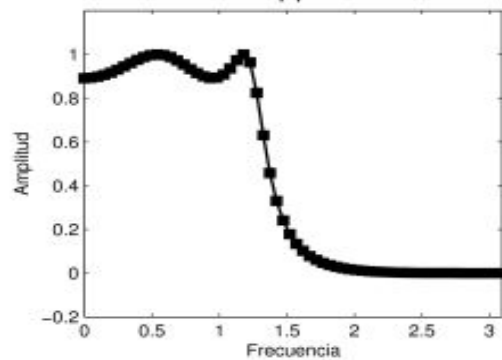


Parámetros en el dominio de la frecuencia

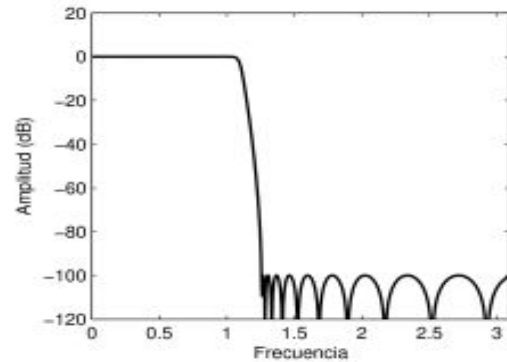
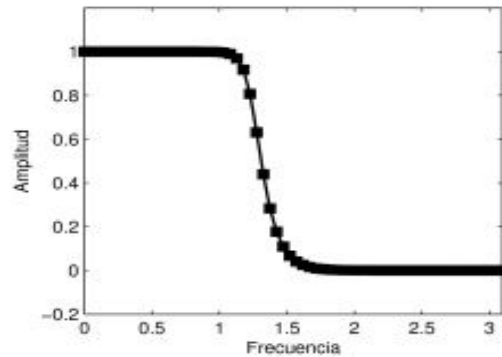
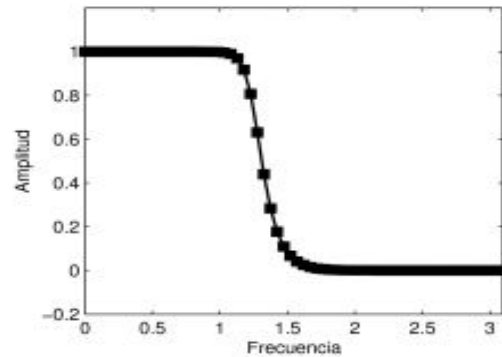
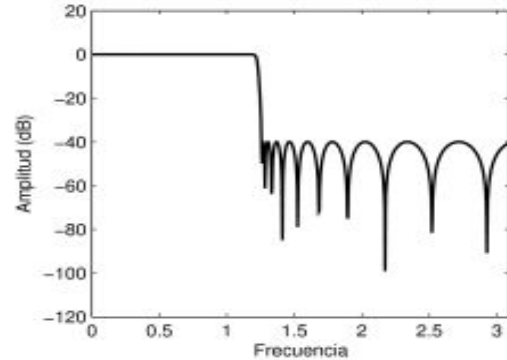
Roll-off



Ripple



Atenuacion



- **Roll-off:** Es el ancho de la banda de transición. Un filtro de roll-off rápido significa que la banda de transición es angosta. Para separar componentes de frecuencia cercanos, el roll-off debe ser rápido.
- **Ripple en la banda pasante:** oscilaciones en la banda pasante de la respuesta en magnitud. Para no alterar la magnitud de los componentes espectrales de la banda pasante, el filtro no debe tener ripple.
- **Atenuación en la banda atenuada:** Es deseable buena atenuación en la banda atenuada para eliminar los componentes espectrales en esa región.

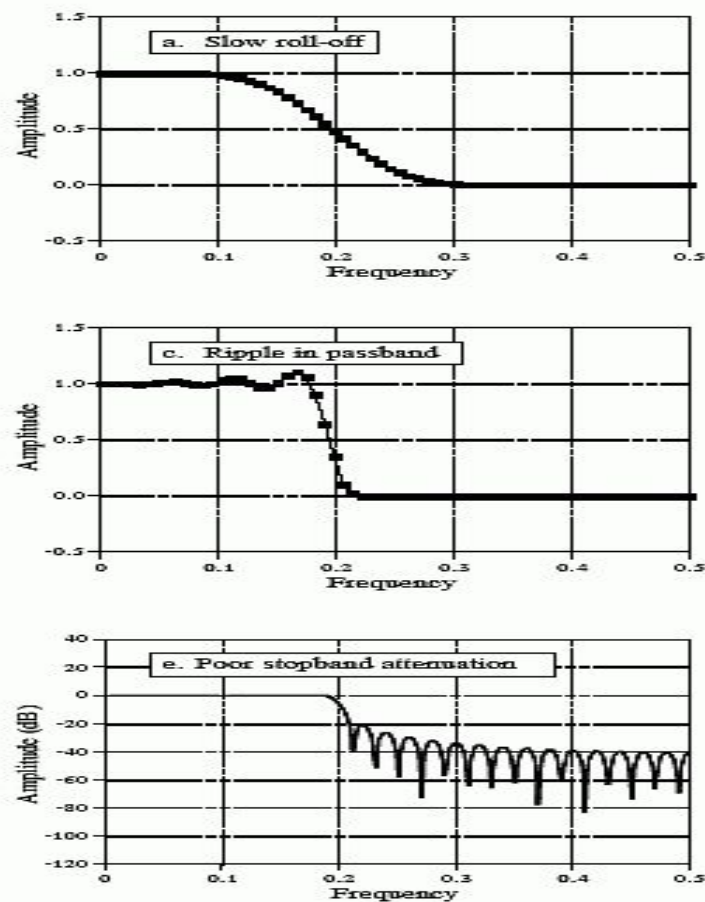
Recordar siempre que estamos hablando de señales que tienen tanto respuesta en frecuencia tanto reales como complejas.

Porque no aparece nada relacionado con la fase? .

1- La fase no es tan importante en la mayoría de las aplicaciones en el dominio de la frecuencia. Por ejemplo, la fase de una señal de audio es casi completamente aleatoria y contiene poca información útil.

2-Si la fase es importante es muy fácil hacer filtros digitales con una respuesta de fase perfecta. En comparación a los filtros analógicos que son malos en ese sentido

POOR



GOOD

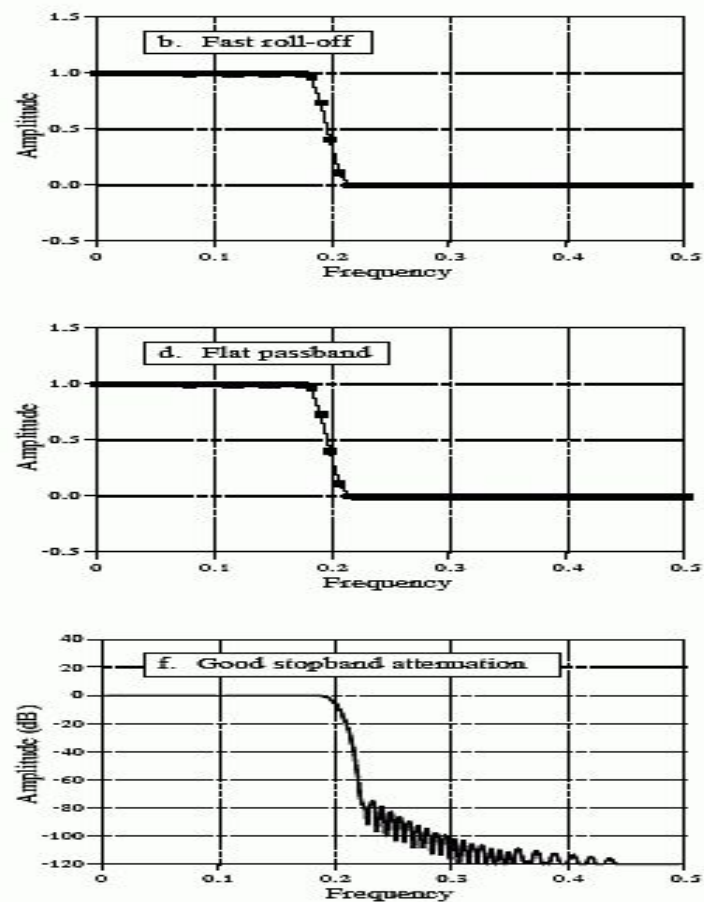


FIGURE 14.4

Parameters for evaluating *frequency domain* performance. The frequency responses shown are for low-pass filters. Three parameters are important: (1) roll-off sharpness, shown in (a) and (b), (2) passband ripple, shown in (c) and (d), and (3) stopband attenuation, shown in (e) and (f).

Ecuación General de un Filtro

Los filtros causales, lineales e invariantes pueden representarse en el dominio del tiempo por su ecuación en diferencias. Esta ecuación es una fórmula que permite calcular la salida de un filtro de acuerdo a las muestras presentes y pasadas de la señal de entrada (parte no recursiva) y las muestras pasadas de la señal de salida (parte recursiva).

$$y(n) = b_0x(n) + b_1x(n-1) + \dots + b_mx(n-m) - a_1y(n-1) - \dots - a_Ny(n-N)$$

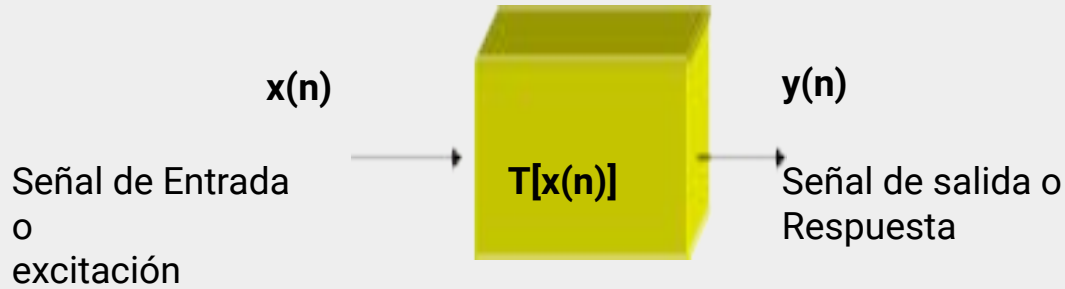
Los valores de las constantes a y las b se denominan coeficientes del filtro. y pueden ser valores reales o complejos.

$$y(n) = \sum_{i=0}^M b_i x(n-i) - \sum_{i=1}^N a_i y(n-i)$$

Denominamos orden del filtro al número mayor de muestras de retardo que posee su parte recursiva o su parte no recursiva (valor máximo entre M y N)

Observemos que a esta ecuación se podría “colocar” en cualquier programa y diseñar cualquier filtro.

Recordando de la parte de sistemas



Dijimos a que si al sistema T , aplicamos el impulso unitario el sistema se podría caracterizar mediante la fórmula:

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

Es la convolución de la señal de entrada con la respuesta al impulso unitario

$$y[n] = x[n] * h[n] = h[n] * x[n]$$

y una de las propiedades de Fourier decía que la convolución de dos señales, en el dominio del tiempo, esa respuesta en el dominio de la frecuencia era :

$$Y[w] = X[w] \times H[w]$$

Se puede ver entonces que la salida de cualquier filtro LTI se puede calcular convolucionando la entrada con la respuesta de impulso.

Podemos comparar este método de implementación de filtros con el uso de ecuaciones en diferencias:

$$y(n) = \sum_{i=0}^M b_i x(n-i) - \sum_{i=1}^N a_i y(n-i)$$

Si no hay retroalimentación ($a_i=0$), entonces la ecuación en diferencia y la fórmula de convolución son idénticas

Para filtros recursivos, podemos convertir la ecuación de diferencia en una convolución calculando la respuesta de impulso del filtro. Sin embargo, esto puede ser bastante tedioso, ya que con coeficientes de retroalimentación distintos de cero, la respuesta de impulso generalmente dura para siempre.

Por supuesto, para filtros estables la respuesta es infinita solo en teoría; en la práctica, se puede truncar la respuesta después de un período de tiempo apropiado, como después de que cae por debajo del nivel de ruido de cuantificación debido a un error de redondeo.

Comparemos

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

Ecuación de salida en función de la entrada de la señal y la respuesta al impulso al sistema

$$y(n) = \sum_{i=0}^M b_i x(n-i) - \sum_{i=1}^N a_i y(n-i)$$

Ecuación general de un filtro con parte recursiva y no recursiva

Se puede deducir que si el filtro es un FIR

$$h(n) \triangleq \begin{cases} 0, & n < 0 \\ b_n, & 0 \leq n \leq M \\ 0, & n > M \end{cases}$$

La respuesta al impulso de un sistema me permite recuperar los coeficientes del sistema en caso de que no los conociera

Filtros FIR

Un filtro FIR (por finite impulse response, respuesta finita al impulso) sólo consta de la parte no recursiva, vale decir que sólo utiliza muestras actuales y anteriores de la señal de entrada. También se denominan filtro transversal o tapped delay line.

La respuesta en amplitud de un filtro FIR está formada por depresiones o anti-resonancias, que constituyen los ceros del filtro. Cada cero corresponde a un coeficiente b_i distinto de 0, para i mayor que 0 ($b_1, b_2, b_3 \dots$). Un filtro FIR siempre es estable, dado que no existe recursividad.

También tienen la ventaja que pueden diseñarse para ser de fase lineal, es decir, no introducen desfases en la señal, a diferencia de los IIR o los filtros analógicos.

Filtros FIR

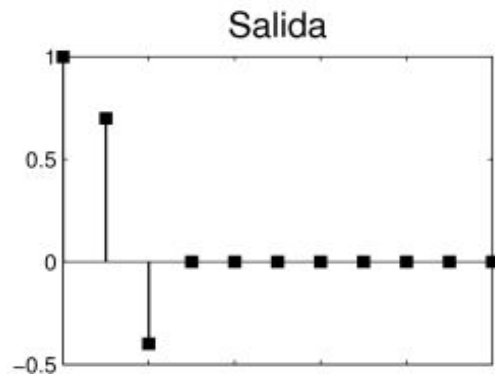
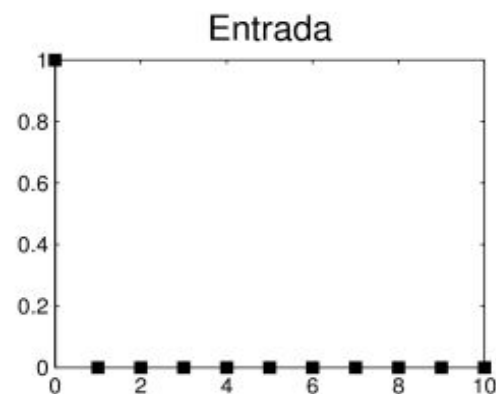
Coeficientes de filtro FIR

$$\begin{aligned}y[n] &= (x * h)[n] \\&= \sum_{i=0}^M h[k]x[n - k] \\&= h[0]x[n] + h[1]x[n - 1] + h[2]x[n - 2] + \cdots + h[M]x[n - M] \\&= b_0x[n] + b_1x[n - 1] + b_2x[n - 2] + \cdots + b_Mx[n - M]\end{aligned}$$

En un filtro FIR, los coeficientes de prealimentación de la ecuación de recurrencia son los coeficientes de la respuesta al impulso y los coeficientes de realimentación son nulos.

Filtros FIR

Ejemplo: Filtro FIR de segundo orden

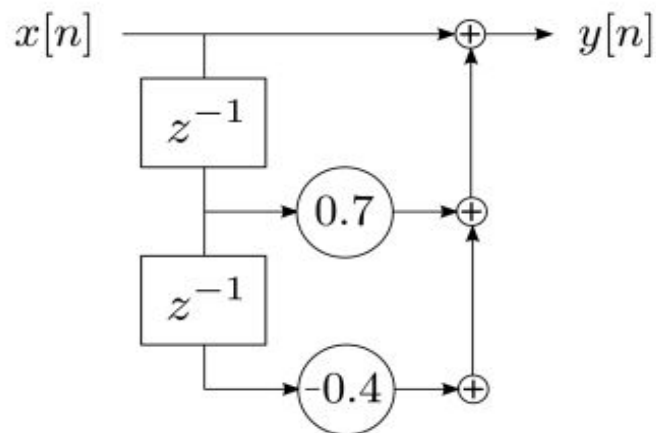


Ecuación de recursión

$$y[n] = x[n] + 0,7x[n - 1] - 0,4x[n - 2]$$

Respuesta al impulso

$$h[n] = \delta[n] + 0,7\delta[n - 1] - 0,4\delta[n - 2]$$



Filtros IIR

A diferencia del filtro FIR, el IIR (por Infinite Impulse Response, respuesta infinita al impulso) consta sólo de una parte recursiva, o bien de ambas, recursiva y no recursiva. Cada término de la ecuación en diferencia que manifiesta una realimentación (cada coeficiente a_i distinto de 0) conforma un polo del filtro, y produce un pico de resonancia en la respuesta en amplitud.

Ecuación en recursión

$$y[n] = b_0x[n] + b_1x[n-1] + b_2x[n-2] + \cdots + b_Mx[n-M] \\ - a_1y[n-1] - a_2y[n-2] - \cdots - a_Ny[n-N]$$

- Las constantes b_i , $i=1,\dots,M$ y a_j , $j=1,\dots,N$ se llaman *coeficientes del filtro*. El filtro queda completamente especificado con los valores de todos los coeficientes.
- Los valores b_i se llaman *coeficientes de prealimentación (feedforward)* y los valores a_j se llaman *coeficientes de realimentación (backward)*.
- El filtro es recursivo si tiene algún coeficiente de realimentación no nulo. En ese caso, es un filtro IIR. En caso contrario, no hay realimentación y el filtro es FIR, o equivalentemente, no recursivo.
- El retardo máximo usado por la ecuación en recurrencia se llama *orden* del filtro. El orden es el máximo entre N y M .

Ejemplo: Cálculo de la respuesta al impulso de filtro IIR de primer orden

Se quiere calcular la respuesta al impulso del filtro dado por la siguiente ecuación en recursión:

$$y[n] = 0,7y[n - 1] + x[n]$$

La entrada al filtro es entonces,

$$x[n] = \delta[n].$$

Imponemos la siguiente condición inicial,

$$y[-1] = 0.$$

Resolviendo a mano la ecuación en recursión,

$$y[0] = 0,7y[-1] + x[0] = 1$$

$$y[1] = 0,7y[0] + x[1] = 0,7$$

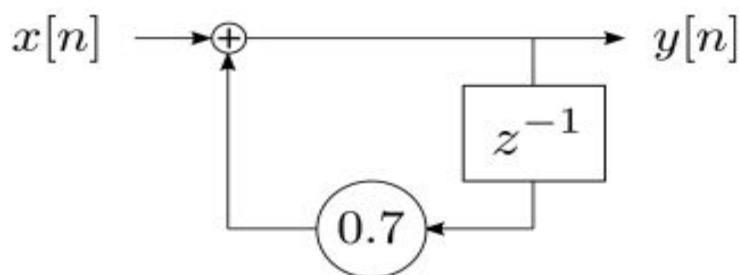
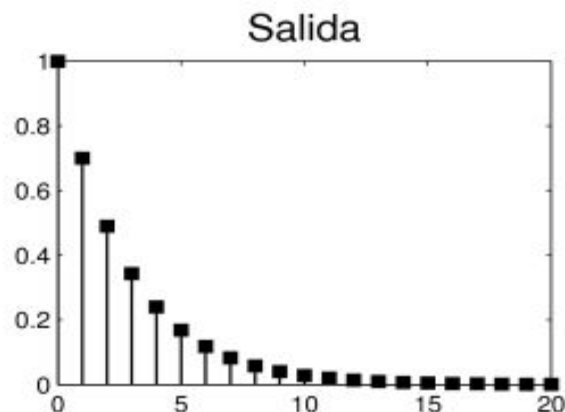
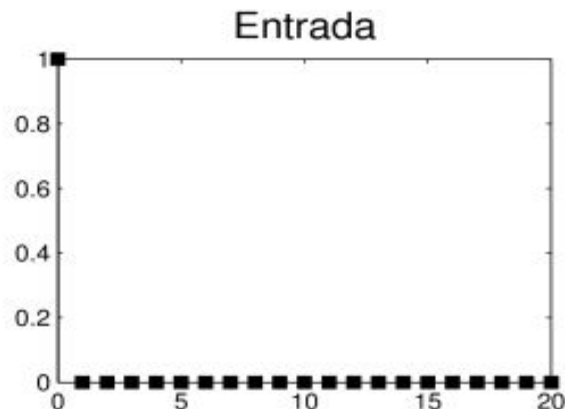
$$y[2] = 0,7y[1] + x[2] = 0,7^2$$

$$y[3] = 0,7y[2] + x[3] = 0,7^3$$

$$\vdots$$

$$y[n] = 0,7^n$$

Ejemplo: Cálculo de la respuesta al impulso de filtro IIR de primer orden



- El filtro tiene realimentación entre la entrada y la salida.
- En general, la respuesta al impulso de un filtro recursivo es una combinación de exponenciales y sinusoides decrecientes.

Observaciones

- Todo filtro, sea FIR o IIR, tiene una respuesta al impulso. En el caso en que el filtro está dado por la ecuación en recurrencia, la expresión analítica de respuesta al impulso puede ser difícil de calcular.
- Si un filtro está definido por la ecuación en recurrencia (y tiene coeficientes de realimentación no nulos), la respuesta al impulso es IIR.
- Si el filtro está definido por la respuesta al impulso, se implementa mediante el producto convolución.

FIR vs IIR

La elección entre una implementación FIR e IIR depende de las ventajas relativas de cada uno de estos dos tipos de filtros.

Los filtros FIR se pueden diseñar para tener una respuesta de fase estrictamente lineal (distorsión de fase nula), lo que es importante en muchas aplicaciones, como transmisión de datos, audio digital y procesamiento de imágenes. La respuesta de fase de filtros IIR no es lineal, en especial en cercanías de la zona de transición.

Los filtros FIR implementados de forma no recursiva, son inherentemente estables. En cambio, la estabilidad de los filtros IIR siempre debe comprobarse, ya que son sistemas realimentados.

Los efectos causados por la implementación con aritmética de punto fijo, tales como los errores de cuantización de los coeficientes y los errores por redondeo en las operaciones aritméticas, son mucho más severos en los filtros IIR que en los FIR.

Para satisfacer unas especificaciones dadas los filtros FIR necesitan un mayor número de coeficientes que los filtros IIR, sobre todo si las bandas de transición son estrechas. En consecuencia, los requerimientos de memoria, el número de operaciones y los tiempos de procesamiento son mayores para los FIR que para los IIR. Sin embargo, la posibilidad de implementar los FIR mediante la técnica de convolución rápida usando FFT y también el empleo de técnicas “multirate” permite aumentar significativamente la eficiencia de las implementaciones

Un filtro analógico convencional puede convertirse en un filtro digital IIR equivalente que satisfaga las especificaciones de diseño de manera sencilla. Esto no es posible con filtros FIR pues no tienen una contraparte analógica. Sin embargo es más sencillo sintetizar filtros con respuestas en frecuencia arbitrarias utilizando filtros FIR.

Los filtros FIR ofrecen en general una respuesta de fase más lineal y no entran jamás en oscilación (es decir, no se vuelven inestables), ya que no poseen realimentación. Por otro lado, requieren un gran número de términos en sus ecuaciones y eso les hace más costosos en cuanto a cálculo o carga computacional.

Un filtro FIR con un corte muy abrupto (es decir, que tenga una banda de transición muy corta) puede requerir hasta centenas de retardos.

En cuanto a los filtros IIR, son muy eficaces y pueden proporcionar pendientes de corte muy pronunciadas. Por otro lado, al poseer características de realimentación (o feedback), tienen tendencia a entrar en oscilación y en resonancia.

<https://www.juansaudio.com/post/iir-vs-fir-entendiendo-realmente-sus-diferencias>

Diseño de Filtros Digitales

El diseño de un filtro digital involucra los siguientes pasos:

- Especificación de los requerimientos del filtro.
- Elección de una aproximación conveniente y cálculo de los coeficientes.
- Representación del filtro utilizando una estructura adecuada (realización).
- Análisis de los efectos de la longitud finita de palabra en el desempeño.
- Implementación en hardware o software.

Estos cinco pasos no son necesariamente independientes, ni necesitan seguirse en el orden descrito; actualmente, las técnicas de diseño disponibles combinan el segundo y parte del tercero y cuarto. Para lograr un filtro eficiente es necesario iterar entre las distintas etapas, especialmente si, como es habitual, las especificaciones de diseño dejan cierto grado de libertad al diseñador, o si se desean explorar otras alternativas de diseño.

Especificaciones de Diseño

Los requerimientos incluyen la especificación de:

1. Las características de las señales: tipo de fuente de señal, interfaz de entrada-salida, velocidad de procesamiento, la mayor frecuencia de interés.
2. Las características del filtro: la respuesta en módulo y/o fase deseados y sus tolerancias, la velocidad de operación, el modo de filtrado (en línea o fuera de línea).
3. La forma de implementación: como una rutina de alto nivel en una computadora, o un programa específico para un DSP.

<https://www.digikey.com/es/articles/why-and-how-to-use-digital-filters-for-analog-to-digital-conversions>

Elección de la aproximación y cálculo de los coeficientes

En esta etapa se elige alguno de los métodos de aproximación:

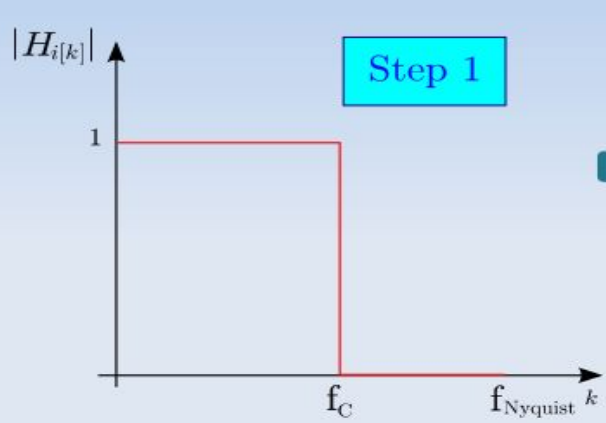
- Butterworth, Chebyshev, elíptico, etc. si el filtro es **IIR**
- Equirriple, óptimo, con ventanas, etc. si es **FIR**

Se calculan los valores de los coeficientes $h[n]$ del filtro FIR, o a_k y b_k del filtro IIR, de modo de satisfacer las especificaciones. El método empleado para determinar los coeficientes es distinto según el filtro sea IIR o FIR.

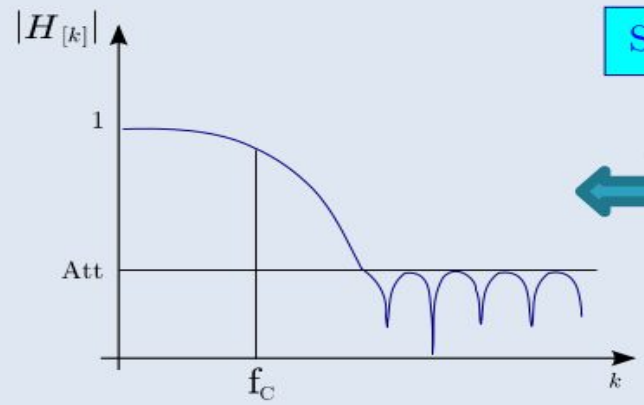
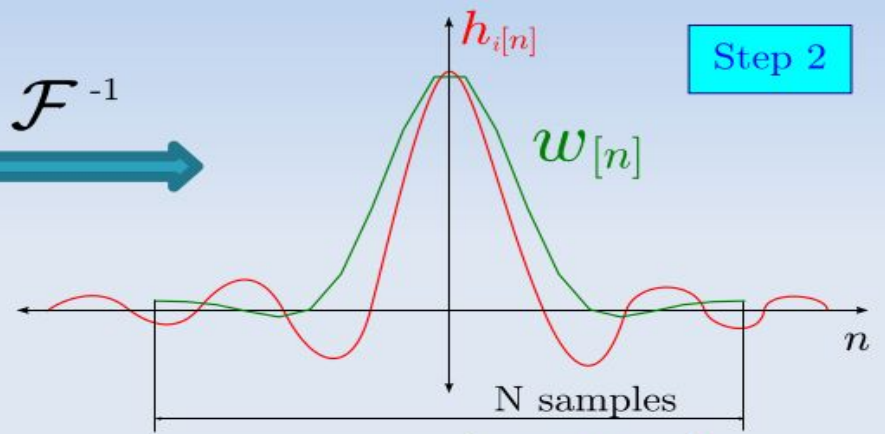
IIR	FIR
Invariación al impulso	Ventanas
Transformada bilineal	Bandas de Transición suaves
Ubicación directa de polos y ceros	Muestreo de Frecuencia
Óptimos	Óptimos

Diseño de Filtros FIR: Método de la Ventana

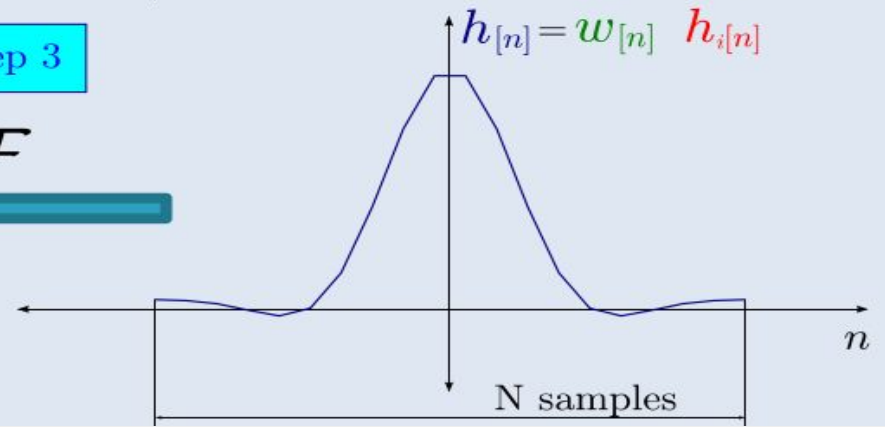
- 1-Establecer la Plantilla de Diseño (Respuesta en Frecuencia deseada).
- 2- Obtener la Respuesta al Impulso discreta en el tiempo del filtro ideal correspondiente, a partir de la TFDT inversa.
- 3- Truncar la Respuesta al Impulso a $M+1$ muestras (M es el orden del filtro) con la ventana adecuada.
- 4- Verificar que las especificaciones del filtro se cumplan (en dB).



\mathcal{F}^{-1}

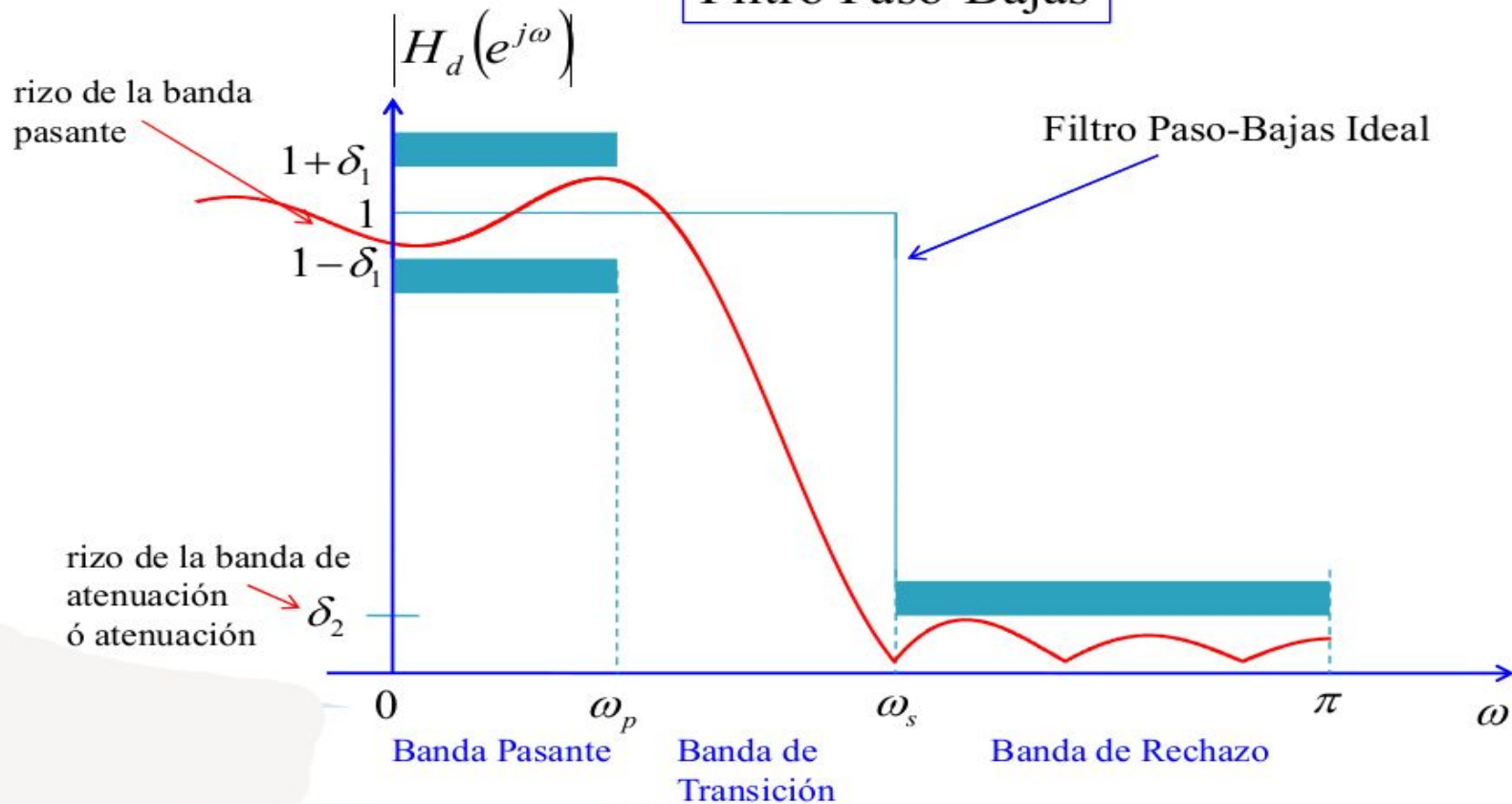


\mathcal{F}



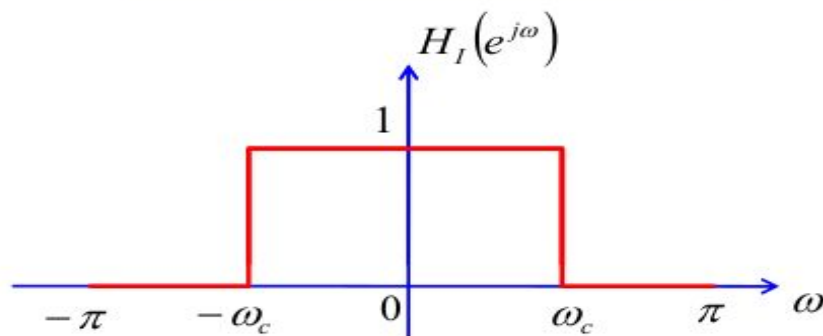
Establecer la **Plantilla de Diseño**: Respuesta en Frecuencia deseada $H_d(e^{j\omega})$.

Filtro Paso-Bajas



Obtener la **Respuesta al Impulso** discreta en el tiempo del **filtro ideal** correspondiente, a partir de la **TFDT inversa**.

Filtro Paso-Bajas IDEAL



$$h_I(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} H_I(e^{j\omega}) e^{j\omega n} d\omega = \frac{1}{2\pi} \int_{-\omega_c}^{\omega_c} e^{j\omega n} d\omega$$

$$h_I(n) = \frac{1}{j2\pi n} [e^{j\omega_c n} - e^{-j\omega_c n}] = \frac{\text{sen}(\omega_c n)}{\pi n} = \frac{\omega_c}{\pi} \frac{\text{sen}\left(\pi \frac{\omega_c n}{\pi}\right)}{\pi \frac{\omega_c n}{\pi}} = \frac{\omega_c}{\pi} \text{sinc}\left(\frac{\omega_c n}{\pi}\right)$$

Truncar la Respuesta al Impulso a **$M+1$ muestras** (M es el orden del filtro) con la ventana adecuada.

$$h_I(n) = \frac{\omega_c}{\pi} \operatorname{sinc}\left(\frac{\omega_c n}{\pi}\right) \quad -\infty < n < \infty$$

Al truncar esta respuesta al impulso a $M+1$ puntos, con la ventana natural o rectangular, tenemos:

$$h_{IT}(n) = h_I(n)w(n) = \frac{\omega_c}{\pi} \operatorname{sinc}\left(\frac{\omega_c n}{\pi}\right) \quad -(M-1)/2 \leq n \leq (M-1)/2$$

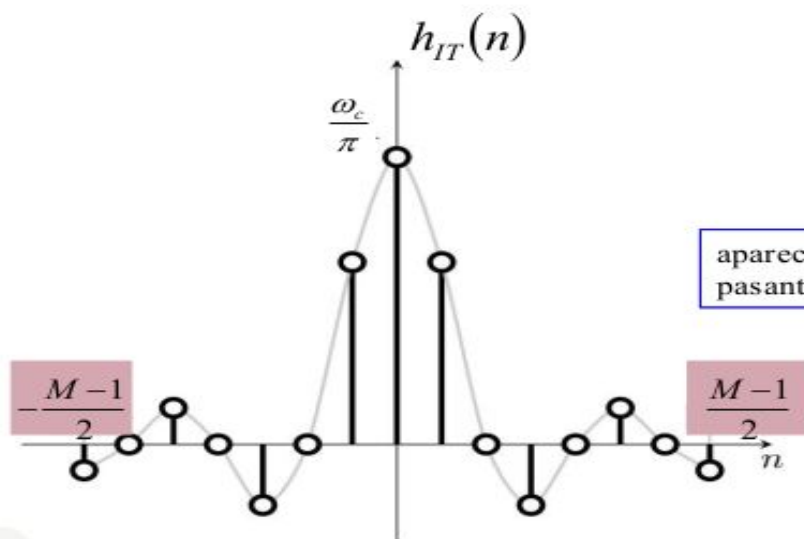
En la frecuencia:

$$H_{IT}(e^{j\omega}) = H_I(e^{j\omega}) * W(e^{j\omega}) = H_I(e^{j\omega}) * \frac{\operatorname{sen}(\omega(M+1)/2)}{\operatorname{sen}(\omega/2)}$$

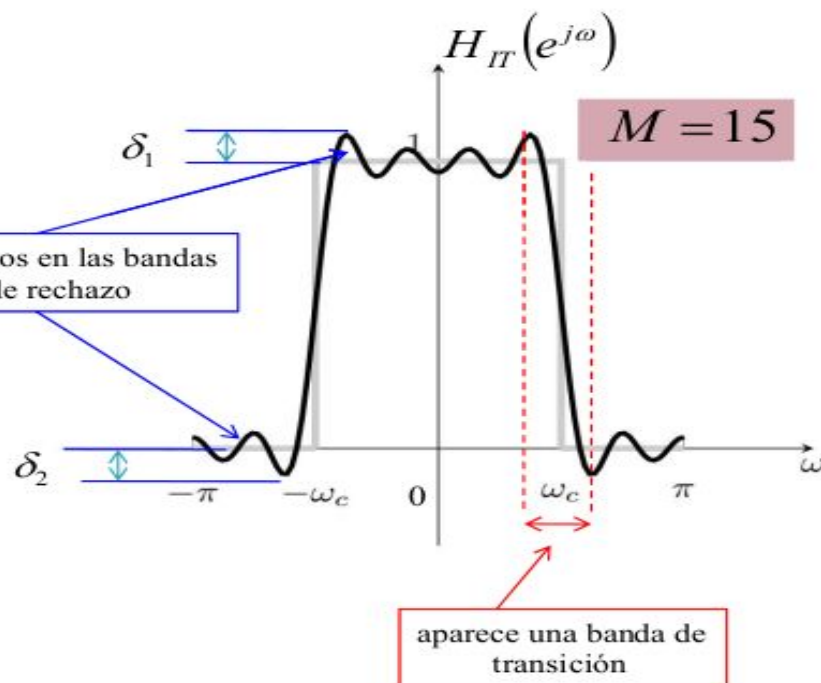
función de Dirichlet

Truncar la Respuesta al Impulso a **$M+1$ muestras** (M es el orden del filtro) con la ventana adecuada.

respuesta al impulso truncada (FIR)

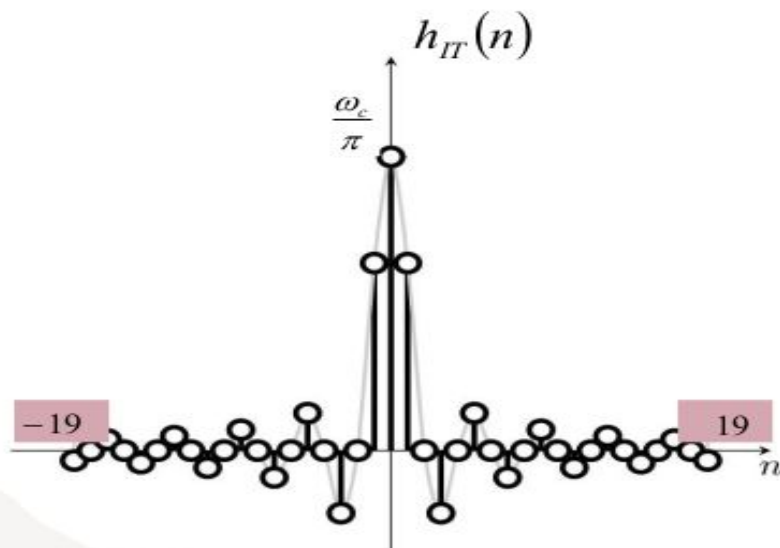


respuesta en frecuencia

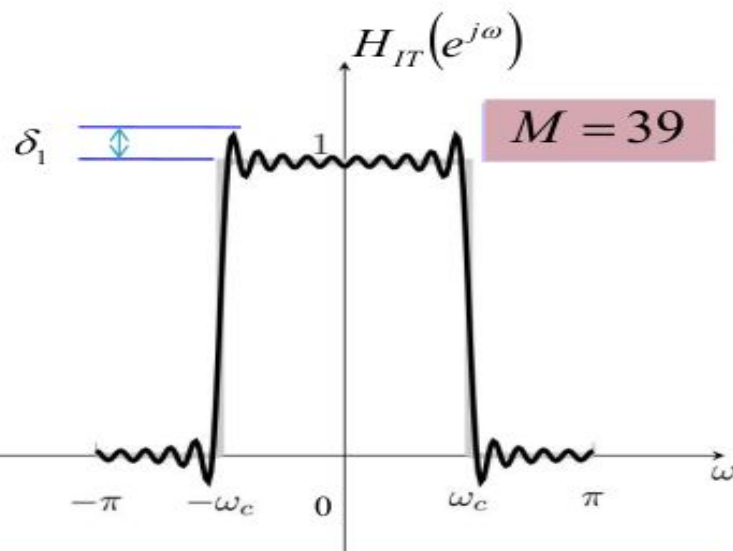


Truncar la Respuesta al Impulso a **$M+1$ muestras** (M es el orden del filtro) con la ventana adecuada.

respuesta al impulso truncada (FIR)



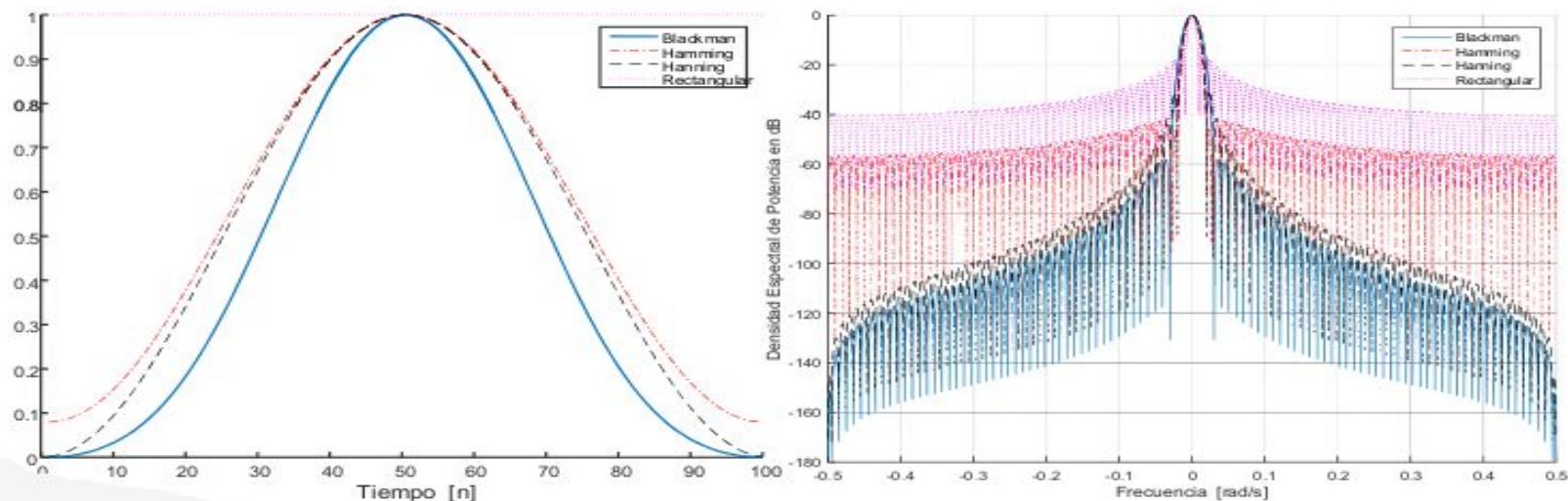
respuesta en frecuencia



δ_1 no disminuye su amplitud al aumentar el orden M en cambio la banda de transición se hace más angosta.

Truncar la Respuesta al Impulso a $M+1$ muestras (M es el orden del filtro) con la **ventana adecuada**.

Dado que el rizo máximo es controlado por la magnitud del primer lóbulo lateral del espectro de la ventana rectangular y la banda de transición por el ancho del lóbulo principal, podemos utilizar las ventanas conocidas para hacer cumplir las especificaciones de la plantilla de diseño lo más cercano posible.



Truncar la Respuesta al Impulso a $N+1$ muestras (N es el orden del filtro) con la **ventana adecuada**.

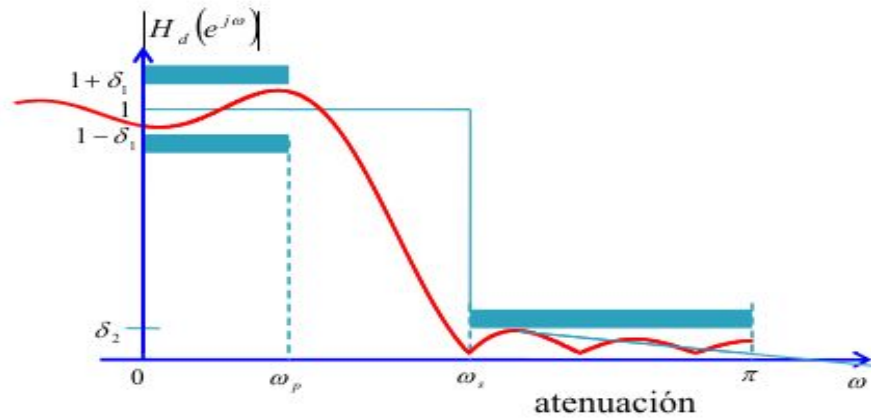
TABLE 7.1 COMPARISON OF COMMONLY USED WINDOWS

Type of Window	Peak Side-Lobe Amplitude (Relative)	Approximate Width of Main Lobe	Peak Approximation Error, $20 \log_{10} \delta$ (dB)	Equivalent Kaiser Window, β	Transition Width of Equivalent Kaiser Window
Rectangular	-13	$4\pi/(M+1)$	-21	0	$1.81\pi/M$
Bartlett	-25	$8\pi/M$	-25	1.33	$2.37\pi/M$
Hanning	-31	$8\pi/M$	-44	3.86	$5.01\pi/M$
Hamming	-41	$8\pi/M$	-53	4.86	$6.27\pi/M$
Blackman	-57	$12\pi/M$	-74	7.04	$9.19\pi/M$

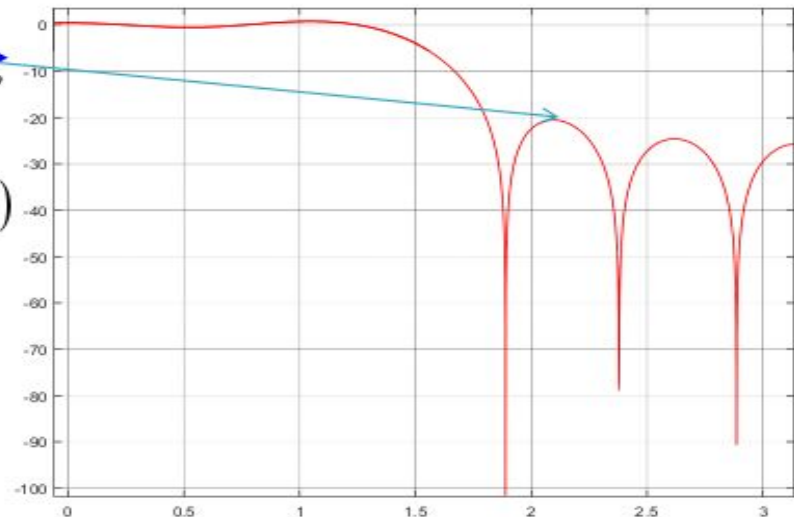
valor considerado
para estimar el orden
 M

valor en dB considerado
para los rizos (atenuación)

Verificar que las especificaciones del filtro se cumplan (en dB).

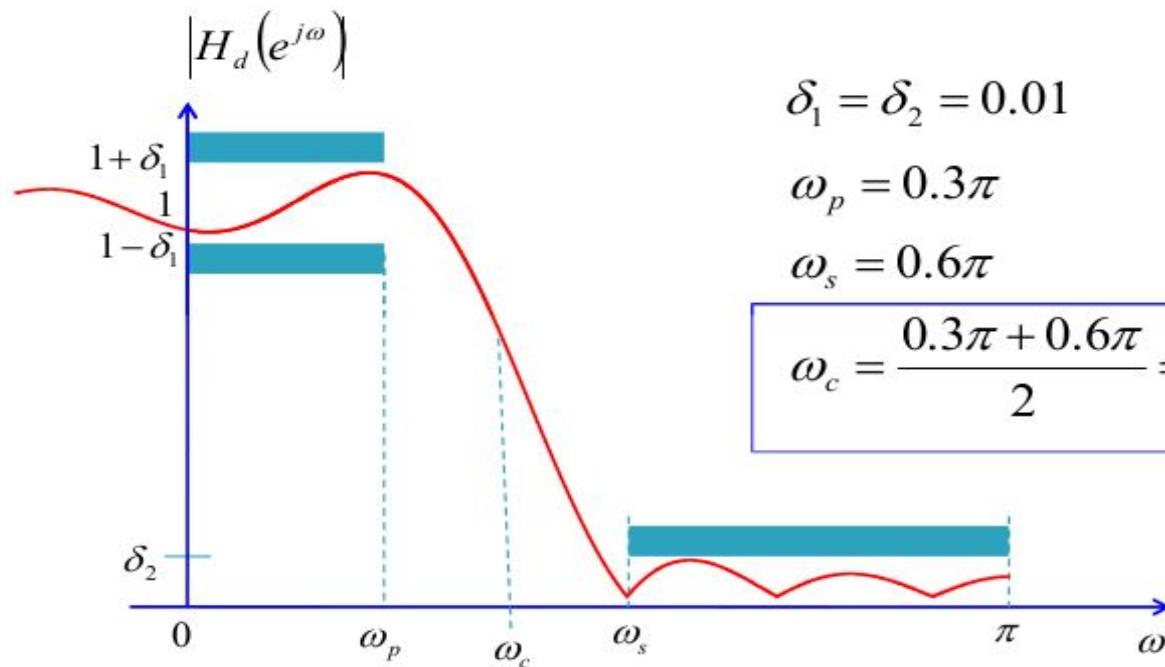


$$20 \log(|H_d(e^{j\omega})|)$$



Ejemplo

Diseña un filtro FIR paso-bajas con las siguientes características:



$$\delta_1 = \delta_2 = 0.01$$

$$\omega_p = 0.3\pi$$

$$\omega_s = 0.6\pi$$

$$\omega_c = \frac{0.3\pi + 0.6\pi}{2} = 0.45\pi$$

Ejemplo

Atenuación

Calculando el orden del filtro: $\delta_2 = 0.01 \Rightarrow \delta_2 = 20\log(0.01) = -40 \text{ dB}$

TABLE 7.1 COMPARISON OF COMMONLY USED WINDOWS

Type of Window	Peak Side-Lobe Amplitude (Relative)	Approximate Width of Main Lobe	Peak Approximation Error, $20\log_{10} \delta$ (dB)	Equivalent Kaiser Window, β	Transition Width of Equivalent Kaiser Window
Rectangular	-13	$4\pi/(M+1)$	-21	0	$1.81\pi/M$
Bartlett	-25	$8\pi/M$	-25	1.33	$2.37\pi/M$
Hanning	-31	$8\pi/M$	-44	3.86	$5.01\pi/M$
Hamming	-41	$8\pi/M$	-53	4.86	$6.27\pi/M$
Blackman	-57	$12\pi/M$	-74	7.04	$9.19\pi/M$

Esto corresponde al ancho de la banda de transición

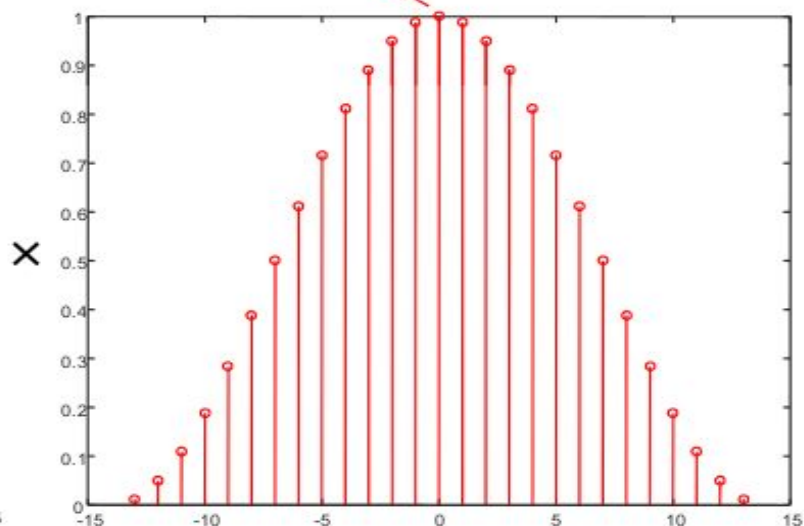
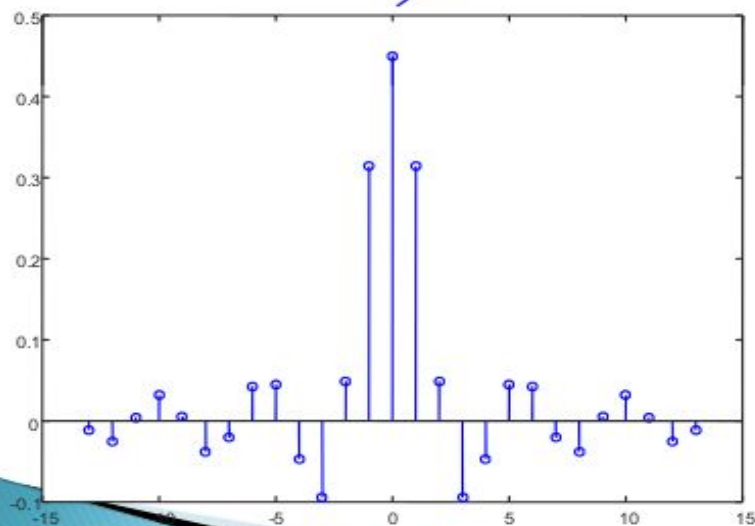
$$\frac{8\pi}{M} = \omega_s - \omega_p = 0.3\pi$$

$$M = \frac{8\pi}{0.3\pi} = 26.67$$

$$M = 27$$

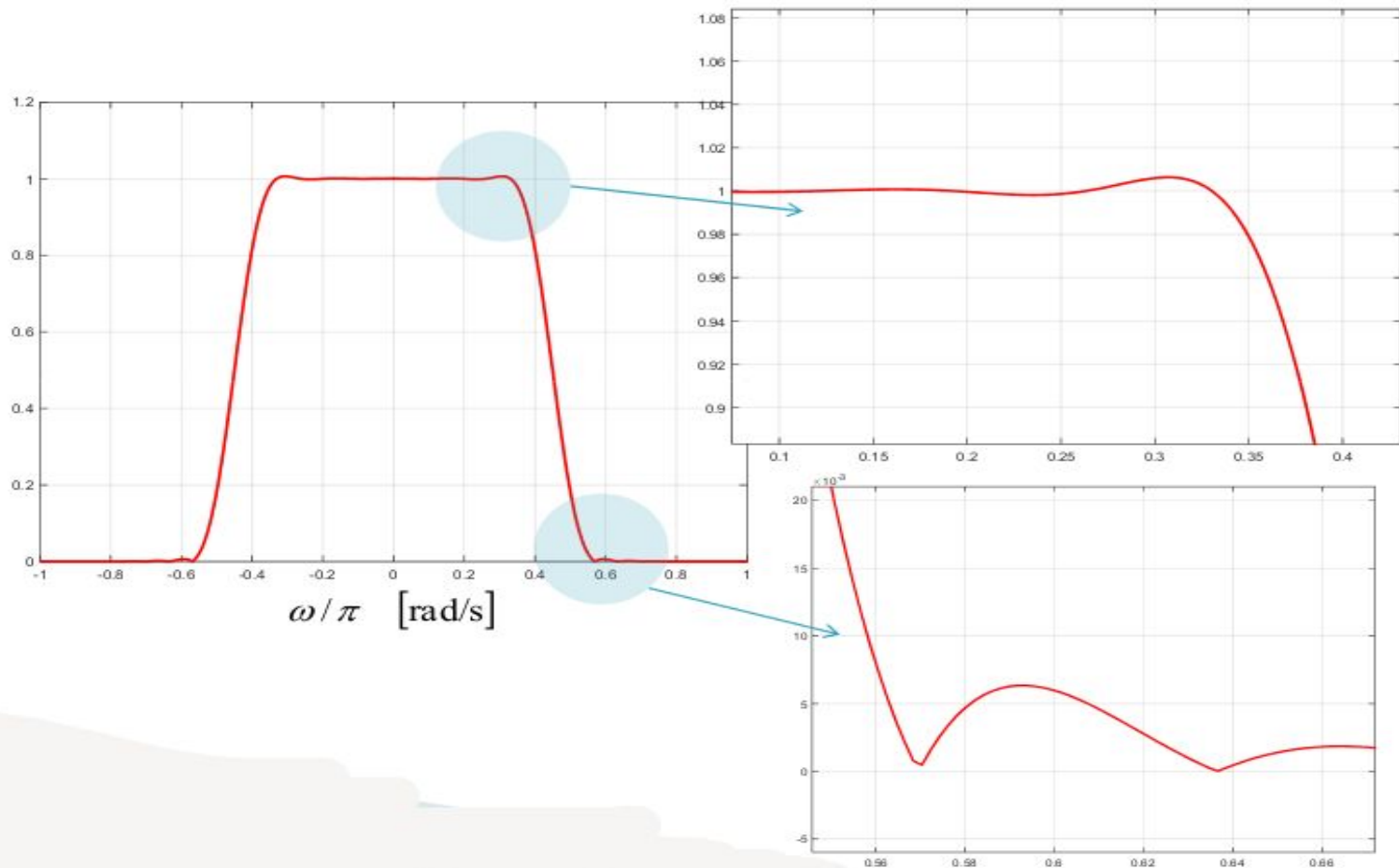
Con el orden M calculado evaluamos la respuesta al impulso truncada ponderada por la ventana seleccionada, en este caso **Hanning**:

$$h_{IT}(n) = h_I(n)w_{Hann}(n) = \frac{0.45\pi}{\pi} \operatorname{sinc}\left(\frac{0.45\pi n}{\pi}\right) [w_{Hann}(n)] \quad -(M-1)/2 \leq n \leq (M-1)/2$$

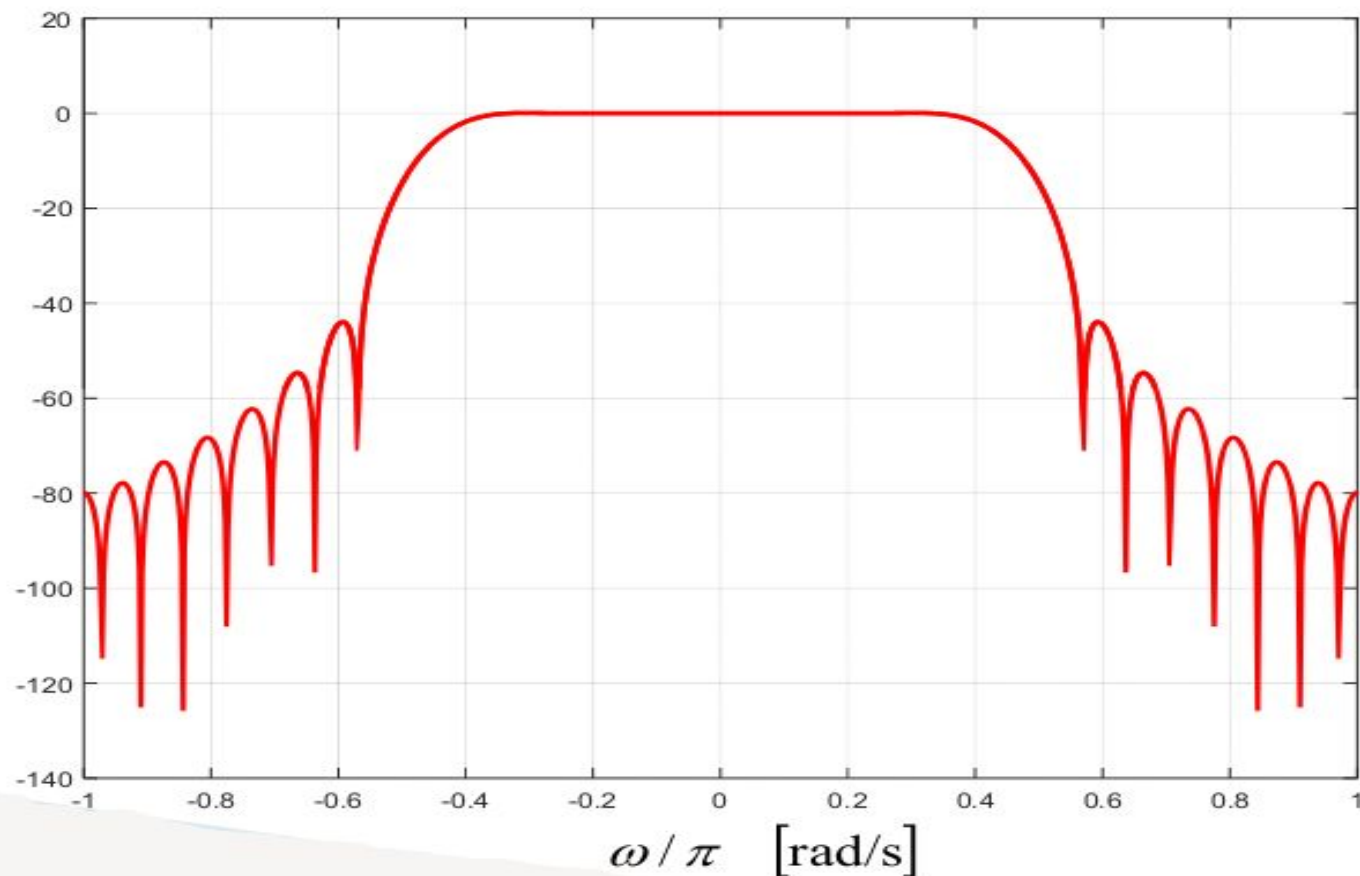


×

Obteniendo la FFT y verificando las especificaciones en escala lineal:



Obteniendo la FFT y verificando las especificaciones en dB:



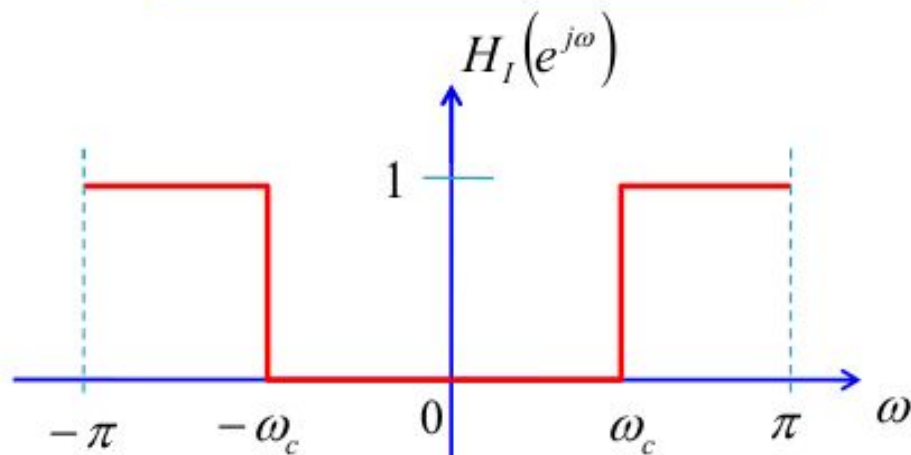
Y los pasa Altos , pasa banda , rechaza banda ?

Se podrían hacer con un pasa bajos a todos?

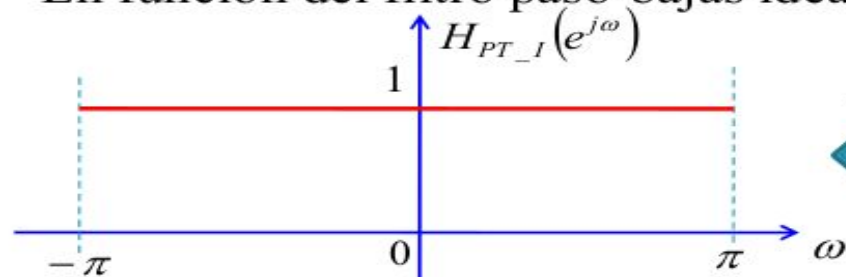
Diseño de Filtros FIR paso-altas, paso-banda y rechaza-banda

El diseño de estos filtros comienza con la plantilla IDEAL de cada uno de ellos, expresado en función de un filtro paso-bajas ideal:

Filtro Paso-Altas IDEAL

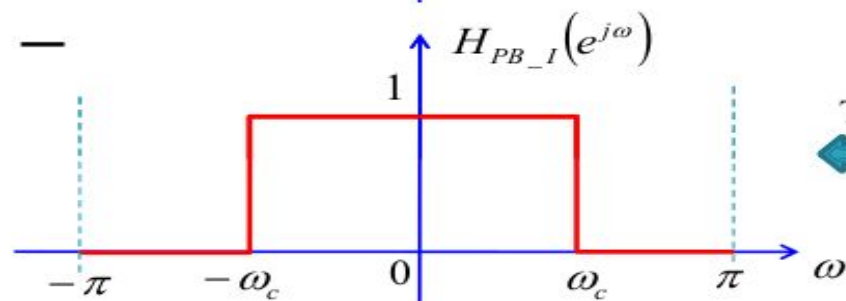


En función del filtro paso-bajas ideal:



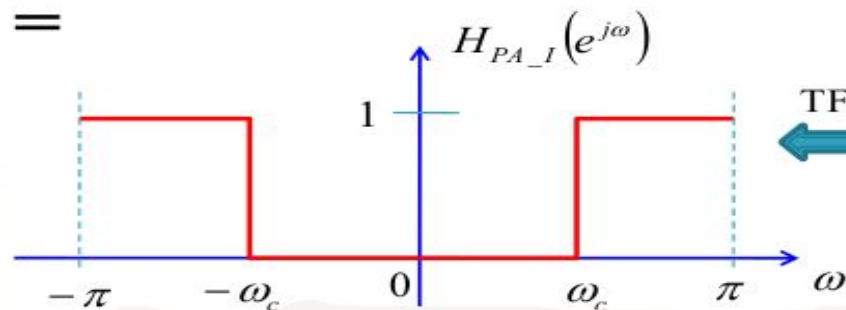
TFDT
 \longleftrightarrow

$$h_{PT_I}(n) = \delta(n)$$



TFDT
 \longleftrightarrow

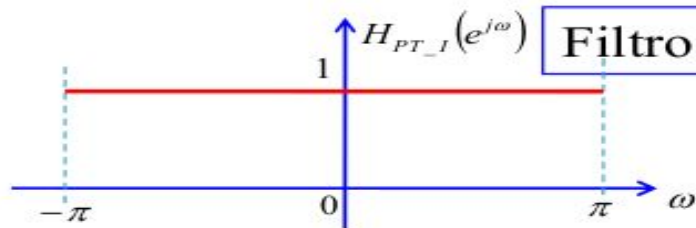
$$h_{PB_I}(n) = \frac{\omega_c}{\pi} \text{sinc}\left(\frac{\omega_c n}{\pi}\right)$$



TFDT
 \longleftrightarrow

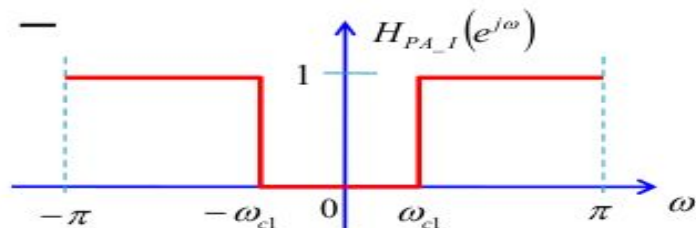
$$h_{PA_I}(n) = \delta(n) - \frac{\omega_c}{\pi} \text{sinc}\left(\frac{\omega_c n}{\pi}\right)$$

Filtro Paso-Banda IDEAL



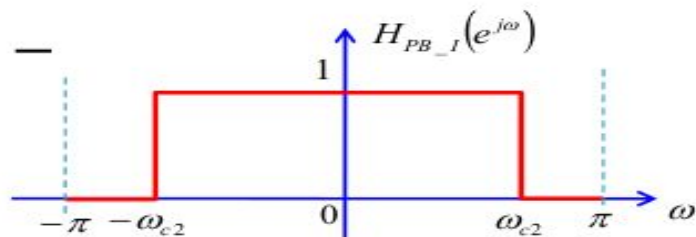
TFDT
↔

$$h_{PT_I}(n) = \delta(n)$$



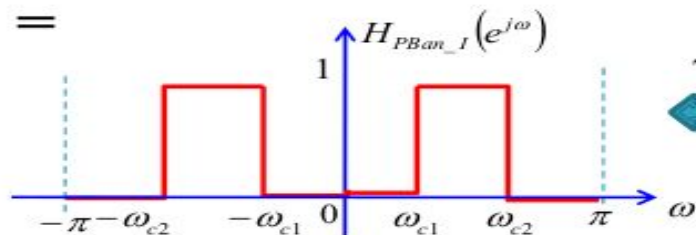
TFDT
↔

$$h_{PA_I}(n) = \delta(n) - \frac{\omega_{c1}}{\pi} \text{sinc}\left(\frac{\omega_{c1}n}{\pi}\right)$$



TFDT
↔

$$h_{PB_I}(n) = \frac{\omega_{c2}}{\pi} \text{sinc}\left(\frac{\omega_{c2}n}{\pi}\right)$$



TFDT
↔

$$h_{PBan_I}(n) = \frac{\omega_{c1}}{\pi} \text{sinc}\left(\frac{\omega_{c1}n}{\pi}\right) - \frac{\omega_{c2}}{\pi} \text{sinc}\left(\frac{\omega_{c2}n}{\pi}\right)$$

ojo falta un delta

Ejemplo

Diseña un filtro FIR paso-altas con las siguientes características:

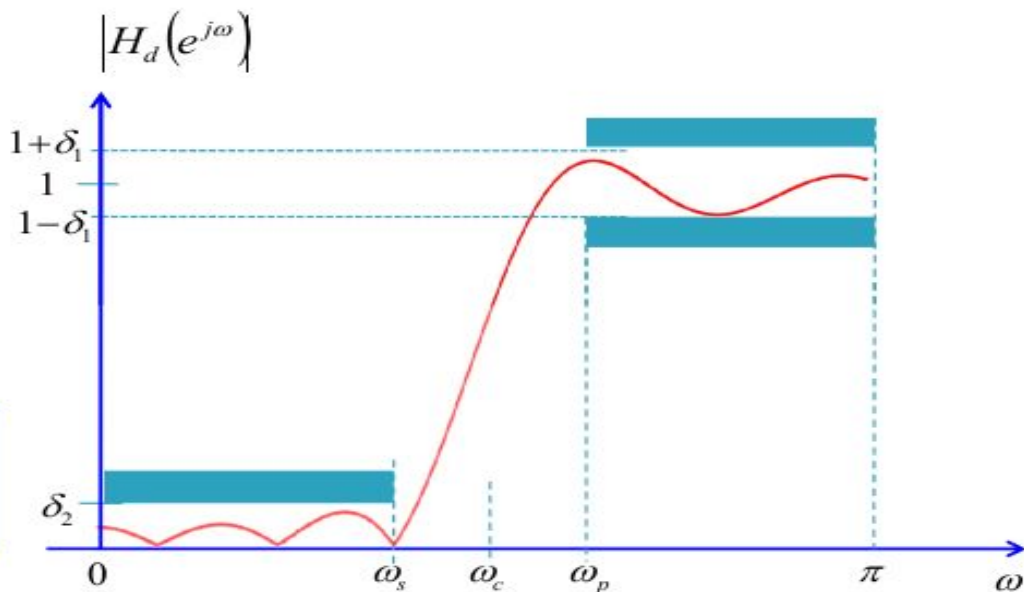
$$\delta_1 = \delta_2 = 0.001$$

$$\omega_p = 0.7\pi$$

$$\omega_s = 0.5\pi$$

$$\omega_c = \frac{0.7\pi + 0.5\pi}{2} = 0.6\pi$$

frecuencia de corte



Calculando el orden del filtro: $\delta_2 = 0.001 \Rightarrow \delta_2 = 20\log(0.001) = -60 \text{ dB}$

TABLE 7.1 COMPARISON OF COMMONLY USED WINDOWS

Type of Window	Peak Side-Lobe Amplitude (Relative)	Approximate Width of Main Lobe	Peak Approximation Error, $20\log_{10} \delta$ (dB)	Equivalent Kaiser Window, β	Transition Width of Equivalent Kaiser Window
Rectangular	-13	$4\pi/(M+1)$	-21	0	$1.81\pi/M$
Bartlett	-25	$8\pi/M$	-25	1.33	$2.37\pi/M$
Hanning	-31	$8\pi/M$	-44	3.86	$5.01\pi/M$
Hamming	-41	$8\pi/M$	-53	4.86	$6.27\pi/M$
Blackman	-57	$12\pi/M$	-74	7.04	$9.19\pi/M$

Esto corresponde al ancho de la banda de transición

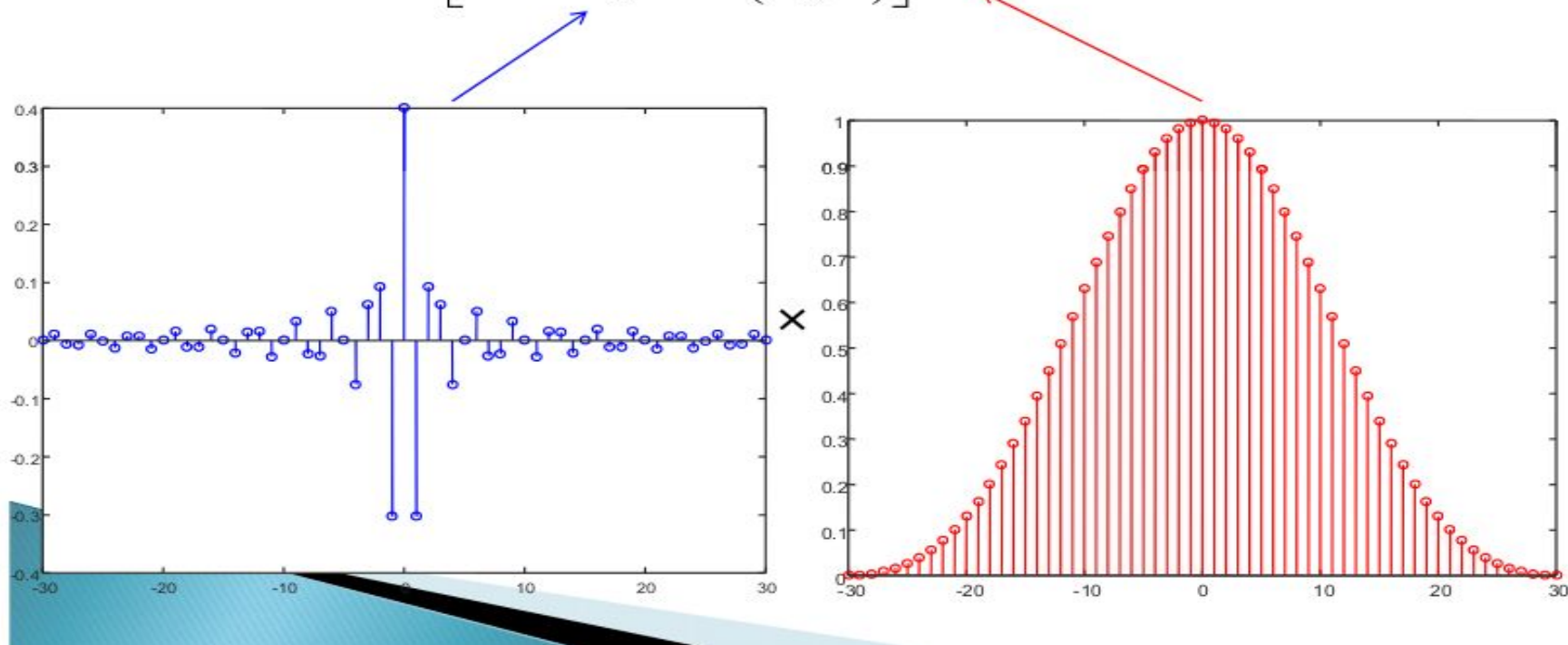
$$\frac{12\pi}{M} = \omega_p - \omega_s = 0.2\pi$$

$$M = \frac{12\pi}{0.2\pi} = 60$$

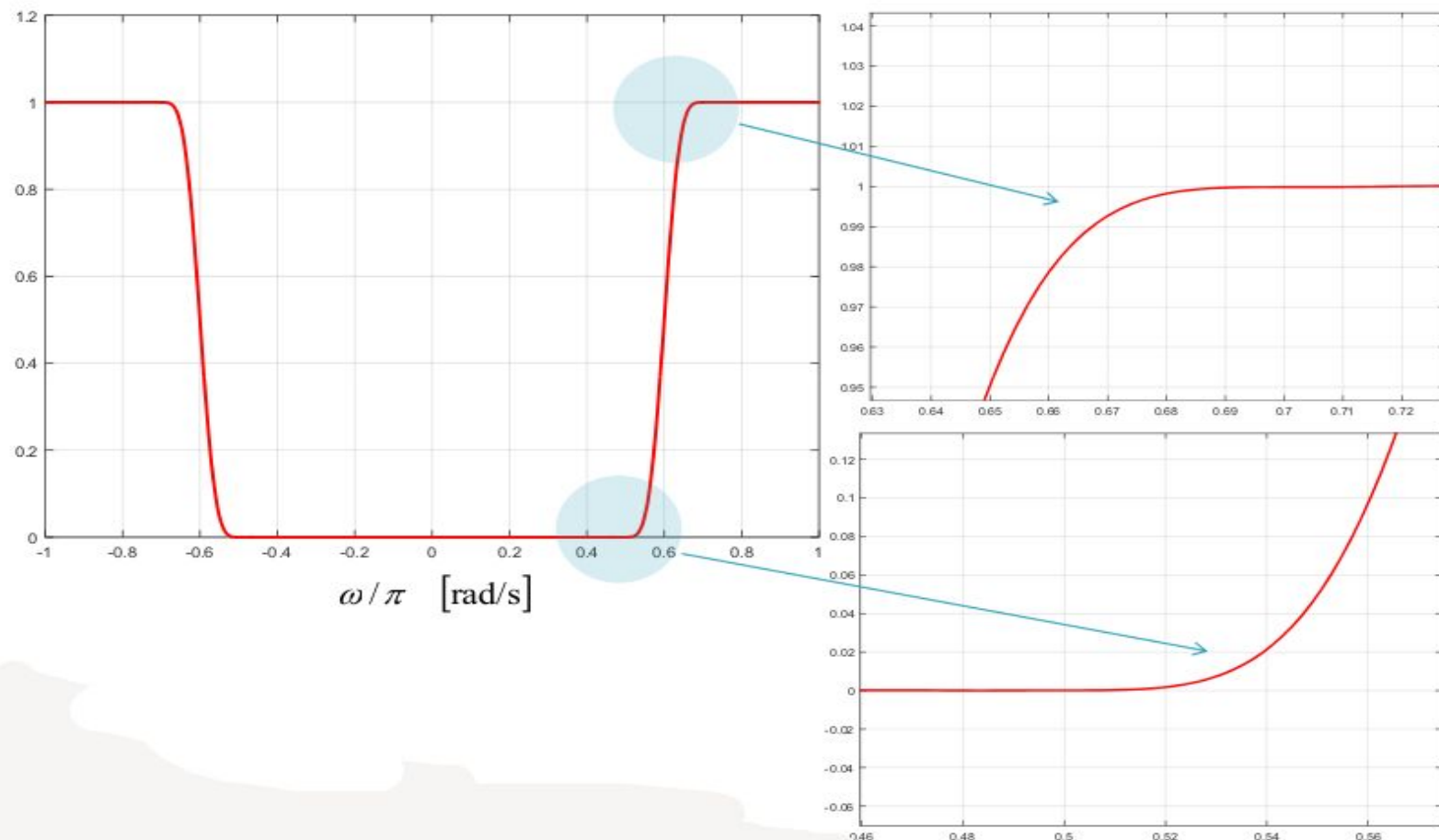
$M = 61$ impar para centrar $h(n)$ en cero

Con el orden M calculado evaluamos la respuesta al impulso truncada ponderada por la ventana seleccionada, en este caso **Blackman**:

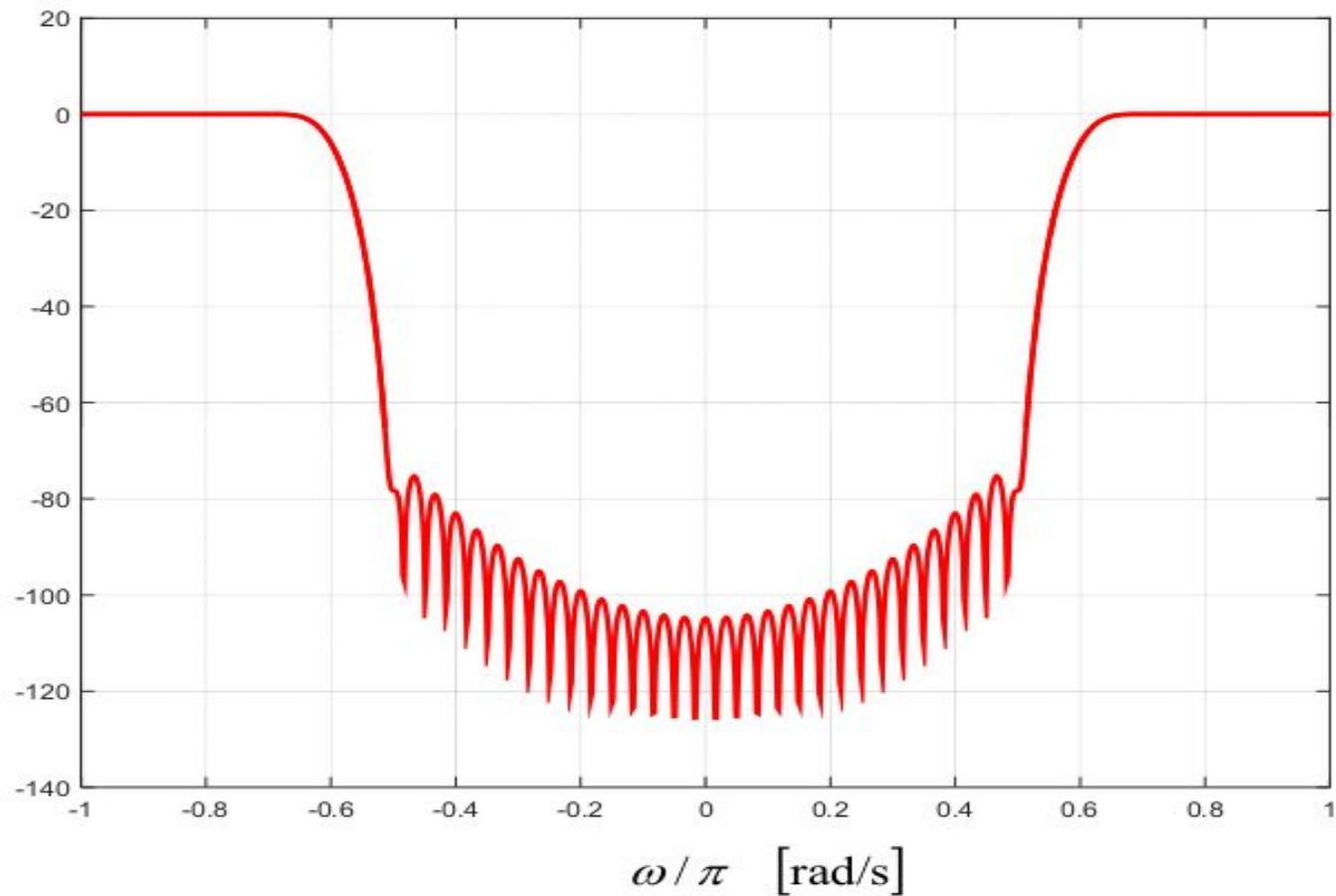
$$h_{IT}(n) = h_I(n)w_{Black}(n) = \left[\delta(n) - \frac{0.6\pi}{\pi} \text{sinc}\left(\frac{0.6\pi n}{\pi}\right) \right] w_{Black}(n) \quad -(M-1)/2 \leq n \leq (M-1)/2$$



Obteniendo la FFT y verificando las especificaciones en escala lineal:



Obteniendo la FFT y verificando las especificaciones en dB:



Ejemplo

Diseña un filtro FIR paso-banda con las siguientes características:

$$\delta_1 = 0.1$$

$$\delta_2 = 0.01$$

banda de transición 1:

$$\Delta f_1 = 300 \text{ Hz}$$

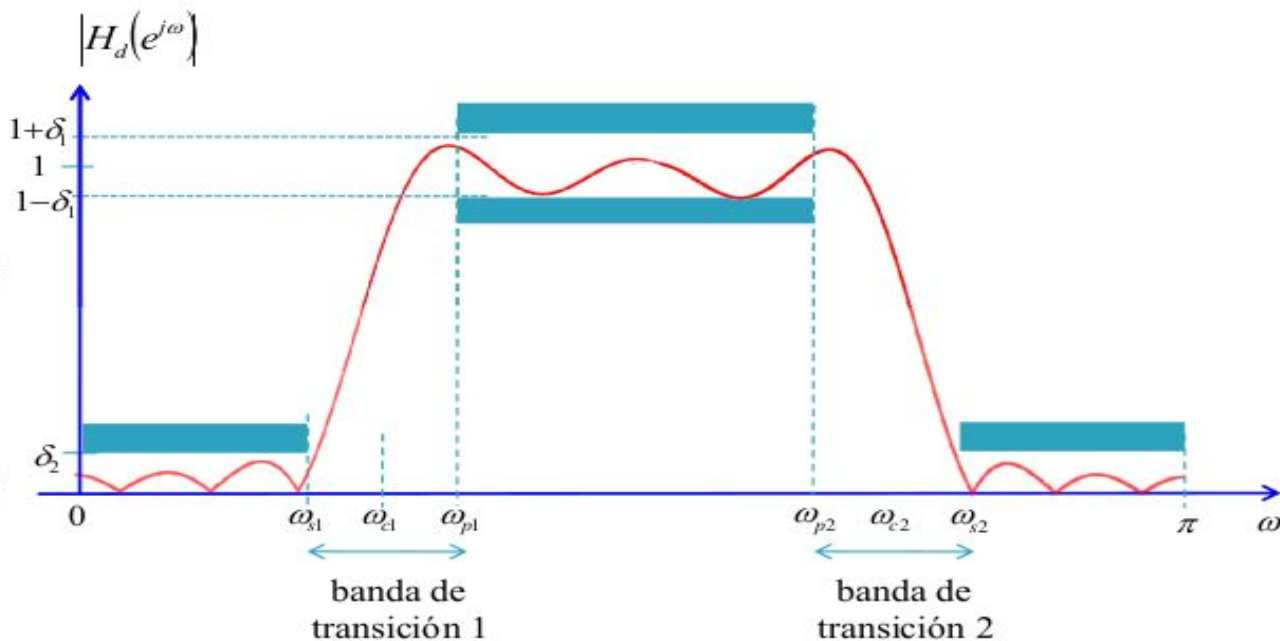
$$f_{c1} = 600 \text{ Hz}$$

banda de transición 2:

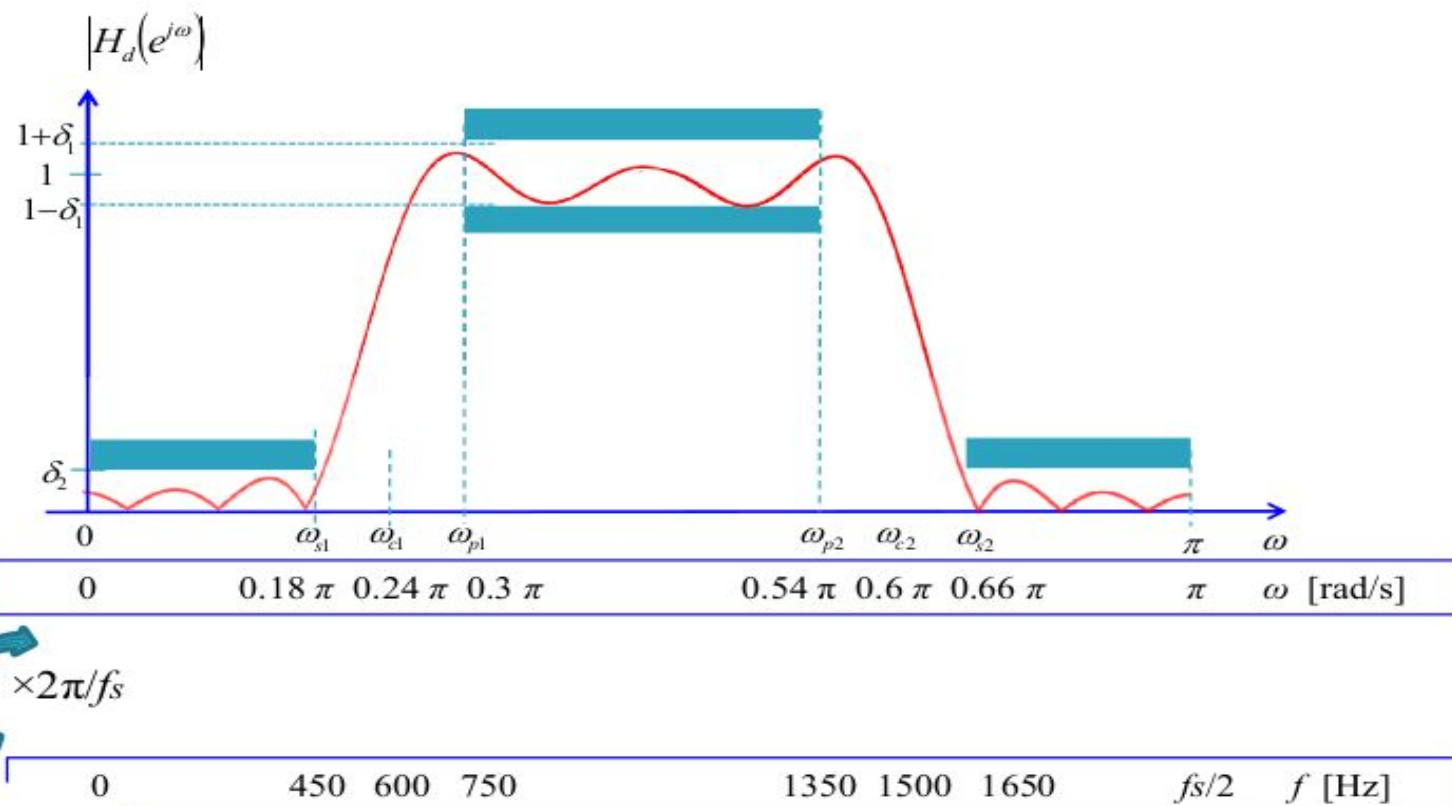
$$\Delta f_2 = 300 \text{ Hz}$$

$$f_{c2} = 1500 \text{ Hz}$$

$$f_s = 5000 \text{ Hz} \quad \text{frecuencia de muestreo}$$



Mapeando las frecuencias en Hz a rad/s:



Atenuación

Calculando el orden del filtro: $\delta_2 = 0.01 < \delta_1 \Rightarrow \delta_2 = 20 \log(0.01) = -40 \text{ dB}$

TABLE 7.1 COMPARISON OF COMMONLY USED WINDOWS

Type of Window	Peak Side-Lobe Amplitude (Relative)	Approximate Width of Main Lobe	Peak Approximation Error, $20 \log_{10} \delta$ (dB)	Equivalent Kaiser Window, β	Transition Width of Equivalent Kaiser Window
Rectangular	-13	$4\pi/(M+1)$	-21	0	$1.81\pi/M$
Bartlett	-25	$8\pi/M$	-25	1.33	$2.37\pi/M$
Hanning	-31	$8\pi/M$	-44	3.86	$5.01\pi/M$
Hamming	-41	$8\pi/M$	-53	4.86	$6.27\pi/M$
Blackman	-57	$12\pi/M$	-74	7.04	$9.19\pi/M$

Esto corresponde al ancho de las bandas de transición

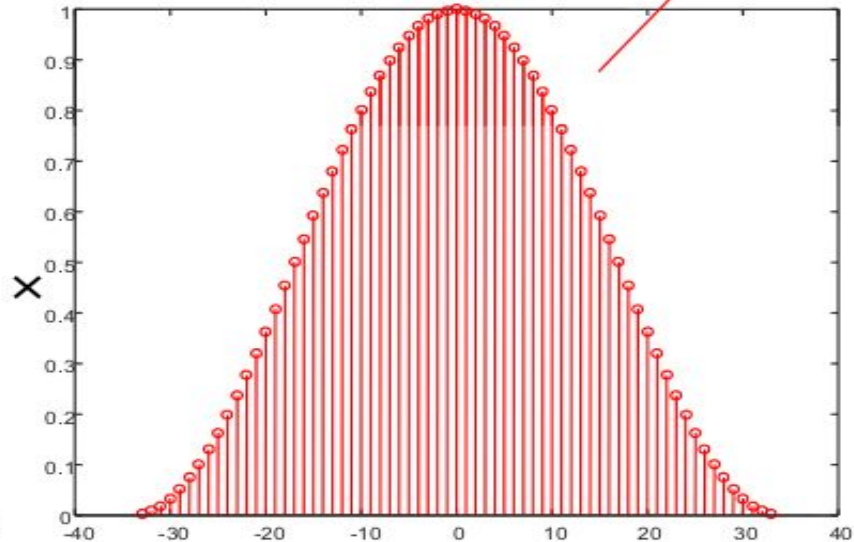
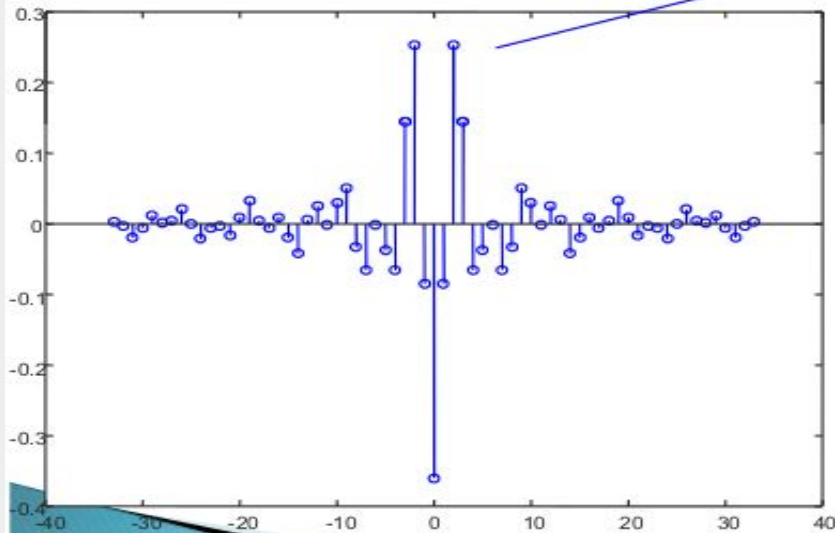
$$\frac{8\pi}{M} = \omega_p - \omega_s = 0.12\pi$$

$$M = \frac{8\pi}{0.12\pi} = 66.7$$

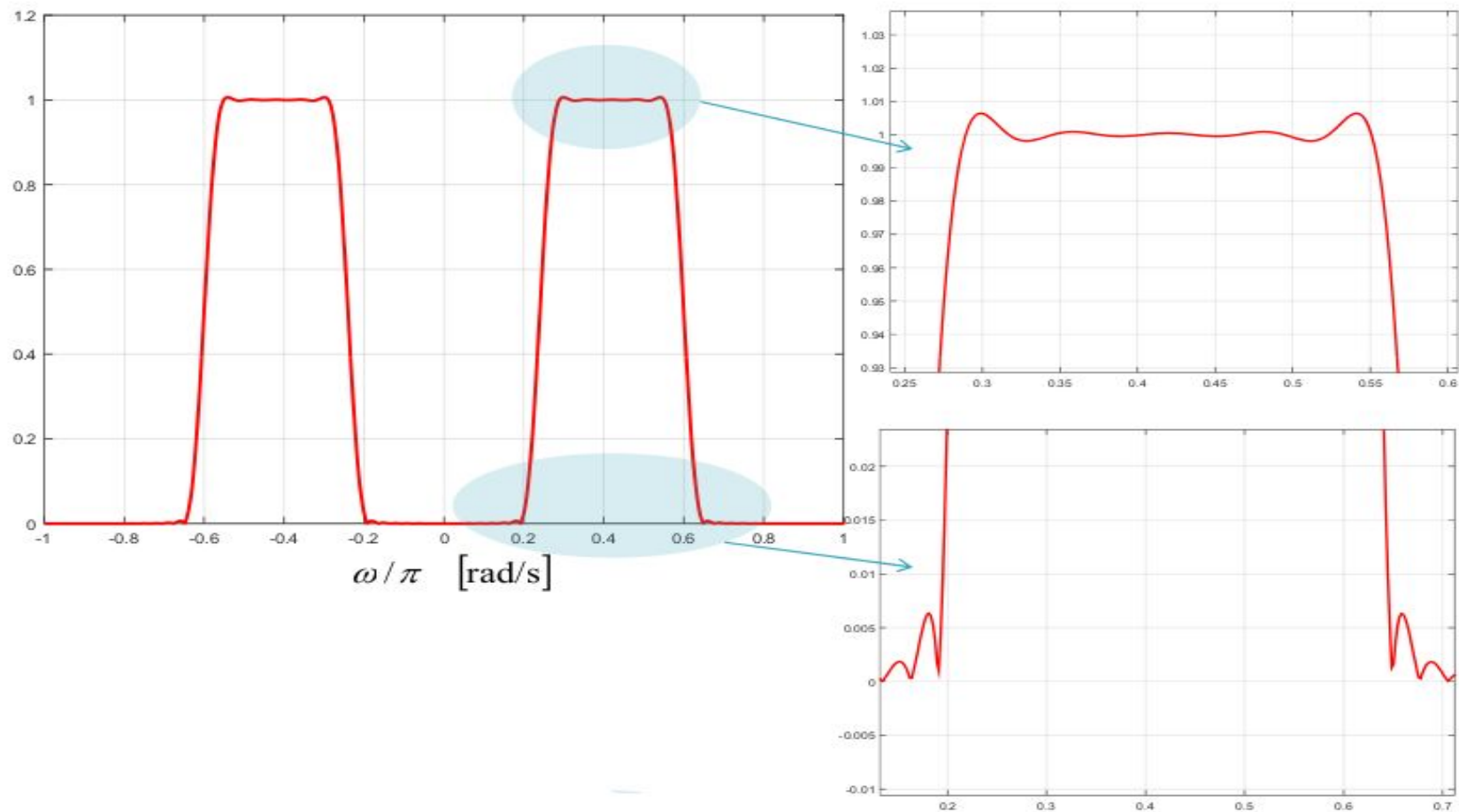
$M = 67$ impar para centrar $h(n)$ en cero

Con el orden M calculado evaluamos la respuesta al impulso truncada ponderada por la ventana seleccionada, en este caso **Hanning**:

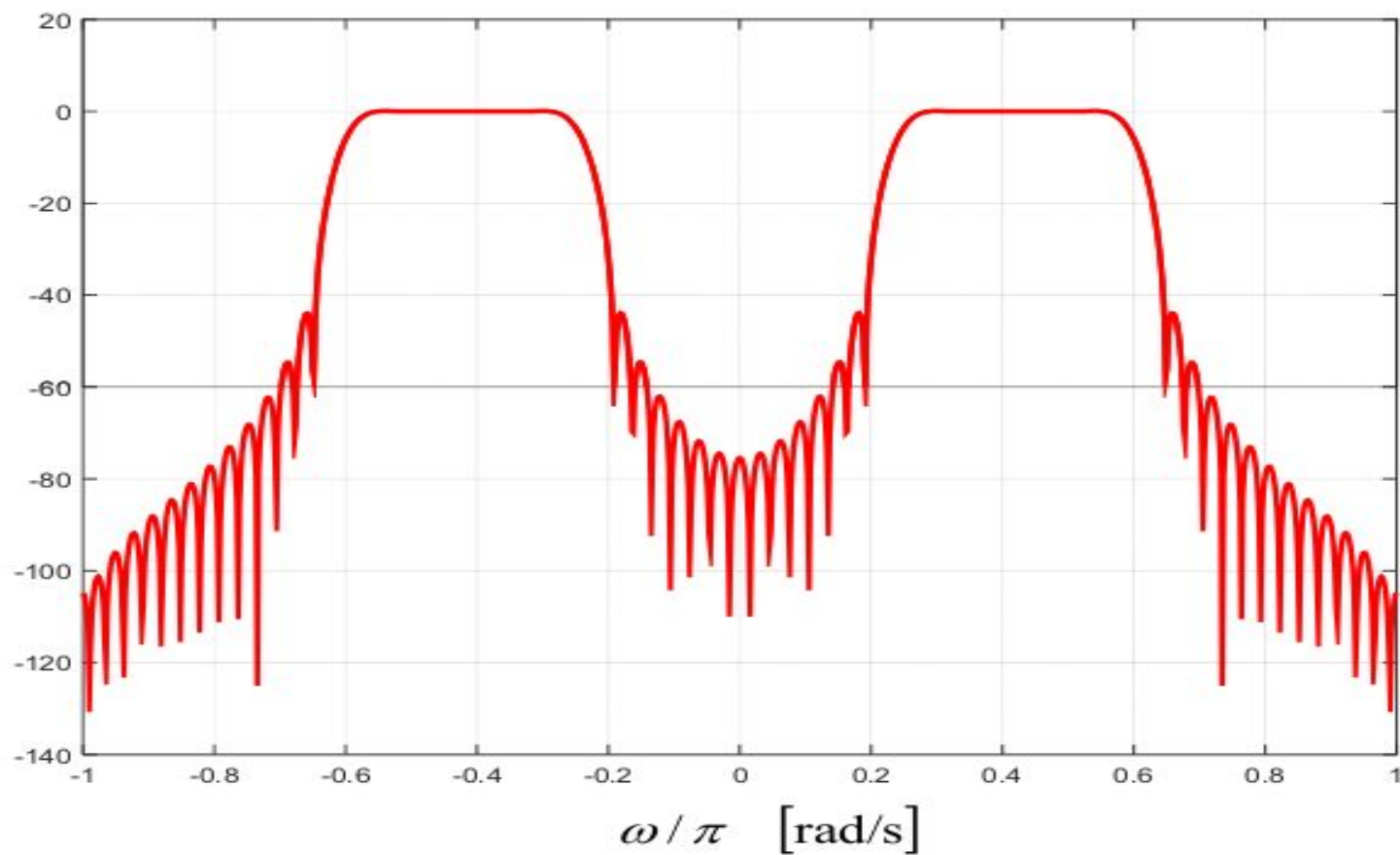
$$h_{IT}(n) = h_I(n)w_{Han}(n) = \left[\frac{0.24\pi}{\pi} \operatorname{sinc}\left(\frac{0.24\pi n}{\pi}\right) - \frac{0.6\pi}{\pi} \operatorname{sinc}\left(\frac{0.6\pi n}{\pi}\right) \right] w_{Han}(n)$$



Obteniendo la FFT y verificando las especificaciones en escala lineal:



Obteniendo la FFT y verificando las especificaciones en dB:



Diseño del filtro FIR-Mediante Python

Para diseñar un filtro FIR a partir de ciertas especificaciones dadas de frecuencias de corte (f_c), frecuencia de muestreo (f_s), ancho de banda en la transición (BW) y ganancia (AdB), se sigue el siguiente procedimiento.

- Se normalizan las frecuencias de corte, y se pasan a rad/s de acuerdo con la expresión:

$$\omega_c = 2\pi f_c / f_s$$

- Se realiza el mismo procedimiento para el ancho de banda de transición:

$$BW_n = (2\pi BW) / f_s$$

- Una regla práctica para determinar el orden del filtro es dividir 4 por el ancho de banda normalizado y tomar la parte entera del resultado:

$$M = \text{int}(4/BW_n)$$

- Se calcula respuesta al impulso del filtro ideal de acuerdo con las siguientes ecuaciones:

Respuesta filtro ideal Tipo de Filtro

$$h_1 = \frac{\omega_c}{\pi} S_a\left(\frac{\omega_c n}{\pi}\right) \text{ Pasa--bajas}$$

$$h_1 = -\frac{\omega_c}{\pi} S_a\left(\frac{\omega_c n}{\pi}\right) \text{ Pasa--altas}$$

$$h_1 = \frac{\omega_{c2}}{\pi} S_a\left(\frac{\omega_{c2} n}{\pi}\right) - \frac{\omega_{c1}}{\pi} S_a\left(\frac{\omega_{c1} n}{\pi}\right) \text{ Pasa--banda}$$

$$h_1 = \frac{\omega_{c1}}{\pi} S_a\left(\frac{\omega_{c1} n}{\pi}\right) - \frac{\omega_{c2}}{\pi} S_a\left(\frac{\omega_{c2} n}{\pi}\right) \text{ Rechaza--banda}$$

En donde S_a es la función SINC

Funciones Importantes que se usan:

```
scipy.signal.freqz(b,a=1, worN=512, whole=False, plot=None, fs=6.283185307179586, include_nyquist=False)
```

<https://docs.scipy.org/doc/scipy/reference/generated/scipy.signal.freqz.html>

Me da la respuesta en frecuencia del filtro cuando le aplicó un pulso unit.

```
scipy.signal.windows.hamming ( M, sym = Verdadero ) ⇒ Genera una ventana Hamming
```

<https://docs.scipy.org/doc/scipy/reference/generated/scipy.signal.windows.hamming.html#scipy.signal.windows.hamming>

Ver filtro de ventana PB.py

Diseño de filtros usando la librería Scipy.signal

<https://docs.scipy.org/doc/scipy/tutorial/signal.html>

Ver ejemplos :

fdPBFirwin.py pasa bajo

fdPAfirwin.py pasa Alto

fdbandpassfirwin.py pasa banda

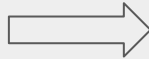
fdbanstopfirwin.py rechaza banda

-Probar con Distintas “Ventanas”

-Otras Señales

-Cambiar Valor de L

<https://www.falstad.com/dfilter/>



Muy didactico

Herramientas

Producto	Compañía	Comentarios
MatLab	The MathWorks	Software muy potente y extensible.
GNU Octave	GNU	Versión MatLab de fuente abierta
SciPy	SciPy	Librería de Python de fuente abierta
DewesoftX	Dewesoft	Software DAQ con amplias capacidades de filtrado, FFT, modales y matemáticas