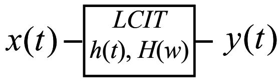
Tarea 8 correspondiente a las actividades no presenciales de la asignatura *Hardware de aplicación específica* (curso 2019-2020).



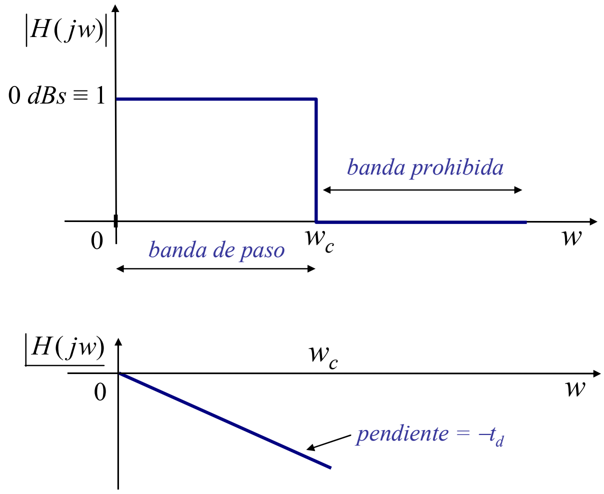
1) Dado un sistema *lineal*, *continuo* e *invariante en el tiempo* (LCIT), cuya respuesta en frecuencia es $H(w)$, si su entrada cumple que $x(t) = A \cdot sen(w_1 \cdot t + \theta_1)$, en el enunciado de la tarea 4 se ha demostrado que en tal caso su salida cumple lo siguiente:

$$y(t) = A \cdot \left|H(w_1)\right| sen\left[w_1 \cdot t + \vartheta_1 + \angle H(w_1)\right] \qquad (1)$$

siendo $\left|H(w_1)\right|$ el valor del módulo de $H(w)$ a la frecuencia $w = w_1$ y siendo $\angle H(w_1)$ la fase de $H(w)$ a la frecuencia $w = w_1$.

Nota: $H(w)$ es una función que depende de una variable ($w \in \Re$) que representa una frecuencia. En general, el valor que toma la función H para cada valor de $w \in \Re$ es un número complejo. De ahí que para representar gráficamente el valor de $H(w)$ se indiquen dos curvas, una correspondiente a su módulo y otra correspondiente a su fase.

Existen sistemas lineales, continuos e invariantes en el tiempo, cuya respuesta en frecuencia $H(w)$ se caracteriza porque su módulo $|H(w)|$ se caracteriza por tener un valor muy próximo a 1 para un determinado conjunto de frecuencias (≡ valores de $w \in \Re$), mientras que tiene un valor muy pequeño (próximo a cero) para otro determinado conjunto de frecuencias (≡ valores de $w \in \Re$). Los sistemas que presentan esta característica se denominan genéricamente como filtros. En la parte derecha se representa el módulo y de la fase de la respuesta en frecuencia de un determinado filtro ideal. Como se puede apreciar en la representación del valor del módulo de $H(w)$, su valor es igual a 1 para $0 \le w \le w_c$ y su valor es igual a 0 para $w > w_c$.

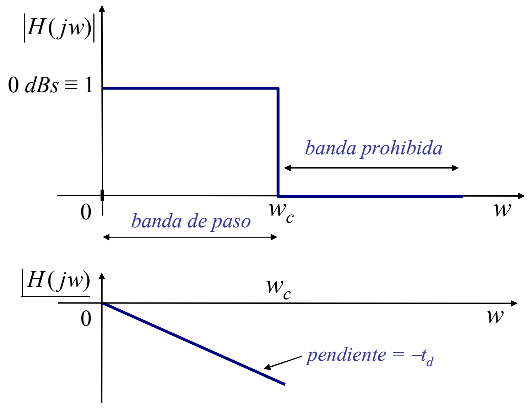


De acuerdo con lo anterior, si en la entrada de un filtro se aplica una señal senoidal cuya frecuencia pertenece al conjunto de frecuencias para las que el módulo de la respuesta en frecuencia del filtro presenta un valor igual a 1 (o muy próximo a 1), de acuerdo con (1) en la salida del filtro aparecerá una señal senoidal igual a la que hay en su entrada (mismo valor de pico y misma frecuencia), aunque con un cierto desfase en el tiempo con respecto a la senoide que hay en la entrada del filtro. Por otra parte, si en la entrada de un filtro se aplica una señal senoidal cuya frecuencia pertenece al conjunto de frecuencias para las que el módulo de la respuesta en frecuencia del filtro presenta un valor igual a 0 (o muy próximo a 0), de acuerdo con (1) en la salida del filtro no aparecerá una señal senoidal o aparecerá una señal senoidal con un valor de pico muy, muy pequeño (tan pequeño que se puede considerar que en la salida no hay una señal senoidal).

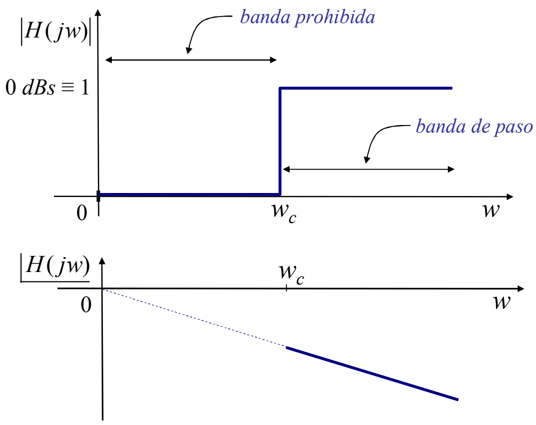
Se define la *banda de paso* de un filtro como el conjunto de frecuencias (≡ valores de $w \in \Re$) para las que el modulo $|H(w)|$ de su respuesta en frecuencia es igual a 1 (o un valor muy próximo a 1 en el caso de un filtro real). Las señales senoidales que se apliquen en la entrada de un filtro cuyas frecuencias pertenezcan a su banda de paso aparecerán en la salida del filtro (idealmente sin sufrir ningún tipo de atenuación, aunque presentarán un cierto desfase con respecto a las senoide presente en la entrada). Es decir, un filtro *deja pasar* las señales senoidales que haya en su entrada cuya frecuencia pertenezca a la banda de paso del filtro.

Se define la *banda prohibida* de un filtro como el conjunto de frecuencias (≡ valores de $w \in \Re$) para las que el modulo $|H(w)|$ de su respuesta en frecuencia es igual a 0 (o un valor muy próximo a 0 en el caso de un filtro real). Las señales senoidales que se apliquen en la entrada de un filtro cuyas frecuencias pertenezcan a su banda prohibida no aparecerán en la salida del filtro. Es decir, un filtro *no deja pasar* (o llegar a su salida) las señales senoidales, que haya en su entrada, cuya frecuencia pertenezca a su banda prohibida.

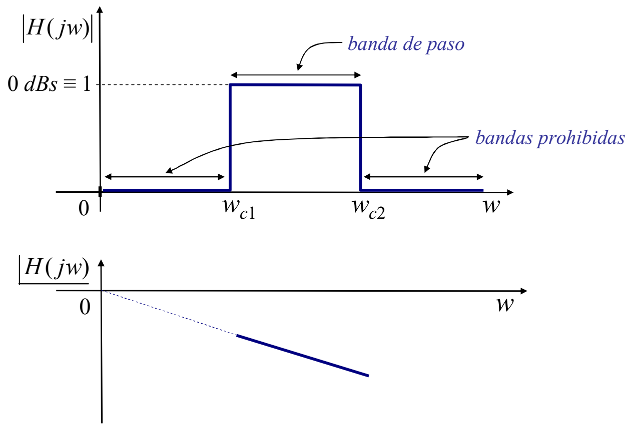
En la práctica, los filtros *ideales* se clasifican según su respuesta en frecuencia en 4 tipos básicos:



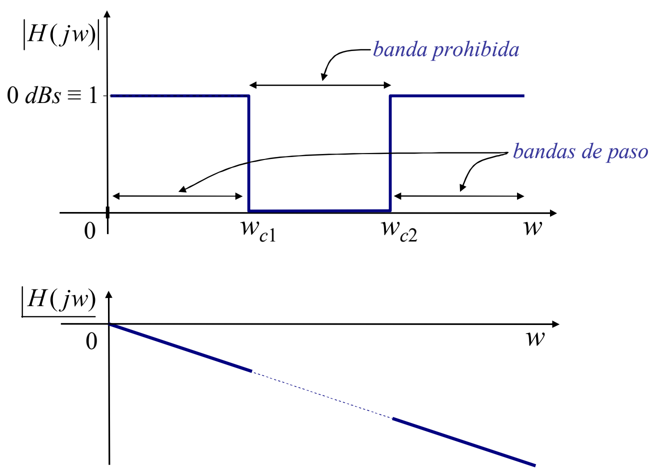
_ Filtros *paso bajo*: se caracterizan porque su *banda de paso* es un conjunto de frecuencias comprendido entre 0 y una frecuencia w_c que se denomina *frecuencia de corte* del filtro. Es decir, la banda de paso de estos filtros corresponde al intervalo de frecuencias [0, w_c]. Mientras que la *banda prohibida* corresponde al intervalo de frecuencias [w_c, ∞]



_ Filtros *paso alto*: se caracterizan porque su *banda de paso* está formada por todas las frecuencias mayores que una frecuencia w_c dada (frecuencia de corte). Es decir, la banda de paso de un filtro paso alto corresponde al intervalo de frecuencias [w_c, ∞], mientras que su banda prohibida corresponde al intervalo de frecuencias [0, w_c].



_ Filtros *paso banda*: se caracterizan porque su *banda de paso* es el conjunto de frecuencias comprendido entre dos frecuencias dadas w_{c1} y w_{c2} (frecuencias de corte). Estos filtros tienen dos bandas prohibidas, una corresponde al conjunto de frecuencias [0, w_{c1}] y la otra corresponde al conjunto de frecuencias [w_{c2}, ∞].

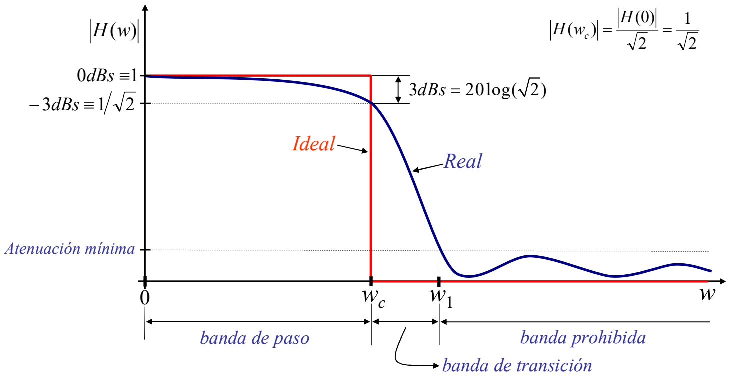


_ Filtros *banda prohibida*: se caracterizan porque su *banda prohibida* es el conjunto de frecuencias comprendido entre dos frecuencias dadas w_{c1} y w_{c2}. Tienen dos bandas de paso, una corresponde al intervalo de frecuencias [0, w_{c1}] y la otra corresponde al intervalo de frecuencias [w_{c2}, ∞].

Las respuestas en frecuencia de los cuatro tipos de filtros básicos representados en la página anterior corresponden a filtros con un comportamiento *ideal* (o si se prefiere, con una respuesta en frecuencia *ideal*). Dichos filtros ideales corresponden a sistemas *no causales*, lo que implica que no se pueden implementar físicamente (al menos si deben de filtrar señales en *tiempo real*).

Los *filtros reales* (que sí se pueden implementar físicamente) son aproximaciones matemáticas de los *filtros ideales* indicados en la página anterior. Su diseño se basa en el uso de diferentes aproximaciones polinómicas a los comportamientos ideales indicados en la página anterior (≡ aproximaciones a las respuestas en frecuencia ideales indicadas en la página anterior). En la práctica, más del 90% de los filtros se diseñan utilizando 4 aproximaciones que se conocen como aproximaciones: *Butterworth*, *Chebyshev* (y *Chebyshev inverso*) *Bessel* y *Cauer* (o *elíptico*). Con cada una de estas aproximaciones polinómicas se consigue diseñar filtros reales que se aproximan más en algunas características y menos en otras al correspondiente filtro ideal. A la hora de diseñar un filtro real hay que tener en cuenta lo siguiente:

_ El valor del módulo de su respuesta en frecuencia en su *banda de paso* no es constante (a diferencia de lo que ocurre en un filtro ideal). Por otra parte, el valor del módulo de su respuesta en frecuencia en su *banda prohibida* no es igual a cero (tal como ocurre en un filtro ideal). En la figura de la parte derecha, la curva de color azul representa el valor del módulo de la respuesta en frecuencia de un filtro paso bajo *real*, mientras que la línea de color rojo corresponde a la respuesta que tendría el filtro si fuese *ideal*.



_ La *banda de paso* y la *banda prohibida* están separadas por un intervalo de frecuencias que se denomina *banda de transición*. Así, por ejemplo, en el módulo de la respuesta en frecuencia indicado en la parte derecha se considera que la *banda de paso* abarca el conjunto de frecuencias comprendido entre 0 y w_c, mientras que la *banda prohibida* no empieza en la frecuencia w_c sino que comienza en la frecuencia w_1. Lo que hace que la banda de transición corresponda al intervalo de frecuencias $w_c \leq w \leq w_1$.

_ En un filtro real, el valor de la frecuencia o frecuencias (según el tipo de filtro que se considere) que delimitan su *banda de paso* son arbitrarias. Así, por ejemplo, en la representación anterior del módulo de la respuesta en frecuencia de un *filtro paso bajo* se acostumbra a considerar que la *banda de paso* finaliza en la frecuencia (w_c) para la que se cumple que el valor del módulo (≡ ganancia del filtro) tiene un valor igual a $1/\sqrt{2}$. Por otra parte, la frecuencia w_1 en la que empieza la *banda prohibida* corresponde a la *atenuación mínima* que el diseñador considere que debe de introducir en filtro a las señales senoidales que estando presentes en la entrada del filtro no deban aparecer en su salida (en un filtro real dichas señales aparecerán en la salida, pero con un valor de pico muy pequeño… despreciable).

_ Cuanto mayor es el orden N de un filtro *real* (≡ mayor es el orden del polinomio utilizado) más se parece la respuesta de dicho filtro a la respuesta de un filtro *ideal*. El inconveniente de esto es que cuanto mayor es el orden de un filtro, mayor es el número de componentes electrónicos que es necesario utilizar para implementarlo o bien mayor es el número de operaciones matemáticas que tiene que realizar el procesador que ejecute el algoritmo que lo implementa digitalmente.

▪ Características de los filtros diseñados utilizando polinomios de *Butterworth*

_ La *magnitud* de su respuesta en frecuencia no presenta oscilaciones en la banda de paso y es monótona decreciente tanto en la banda de transición como en la banda prohibida.

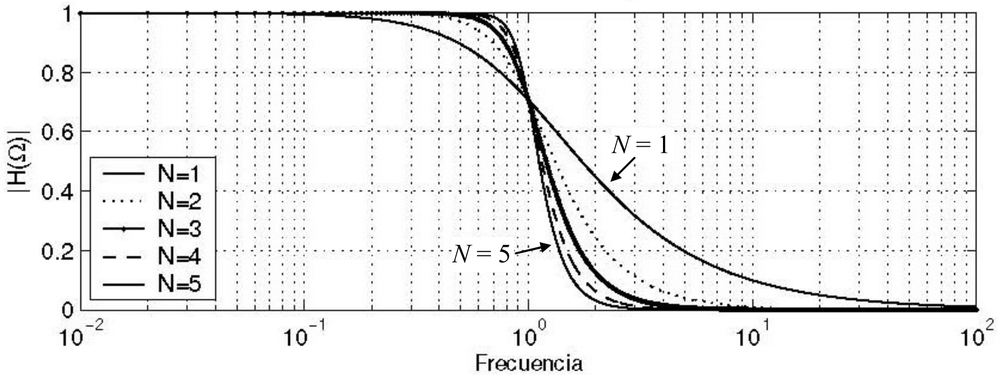
_ La *fase* de su respuesta en frecuencia no es lineal con la frecuencia.

_ Son adecuados para aplicaciones de audio.

_ Cuanto mayor sea el orden (*N*) del filtro, más estrecha será su banda de transición.

_ En las aplicaciones reales, el orden *N* del filtro se elige de modo que se consiga la atenuación mínima deseada a partir de la frecuencia en la que comienza la banda prohibida.

A continuación se muestra, a modo de ejemplo, la representación del módulo de 5 respuestas en frecuencia de filtros paso bajo (en el eje de abscisas se utiliza escala logarítmica). Como se puede apreciar en las figuras representadas, cuanto mayor es el orden (*N*) del filtro, más se aproxima su respuesta en frecuencia a la de un filtro ideal.



▪ **Características de los filtros diseñados utilizando polinomios de *Chebyshev*** (o de *Chebyshev* de tipo I)

_ La magnitud de la respuesta en frecuencia presenta un rizado de amplitud constante en la *banda de paso*, el cual es independiente del orden del filtro. En la *banda de transición* y en la *banda prohibida* la magnitud de la respuesta en frecuencia presenta una amplitud monótona decreciente (ver ejemplo más abajo).

_ La fase de la respuesta en frecuencia no es lineal (con la frecuencia)

_ La banda de transición es muy estrecha.

_ Este tipo de aproximación matemática no resulta adecuada para aplicaciones de procesado de audio, debido al rizado que presenta la curva de módulos en la banda de paso. Sin embargo resulta adecuada para:

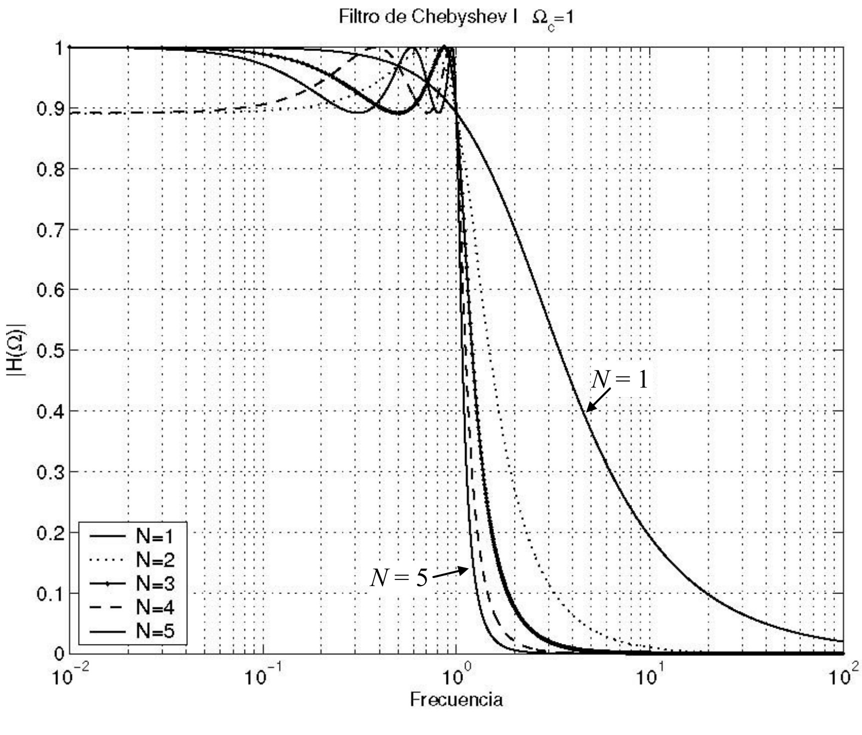
· Separar/discriminar 2 armónicos muy próximos, debido a lo estrecha que es la banda de transición.

· Generar una señal senoidal a partir de una señal rectangular.

· Filtrado de armónicos.

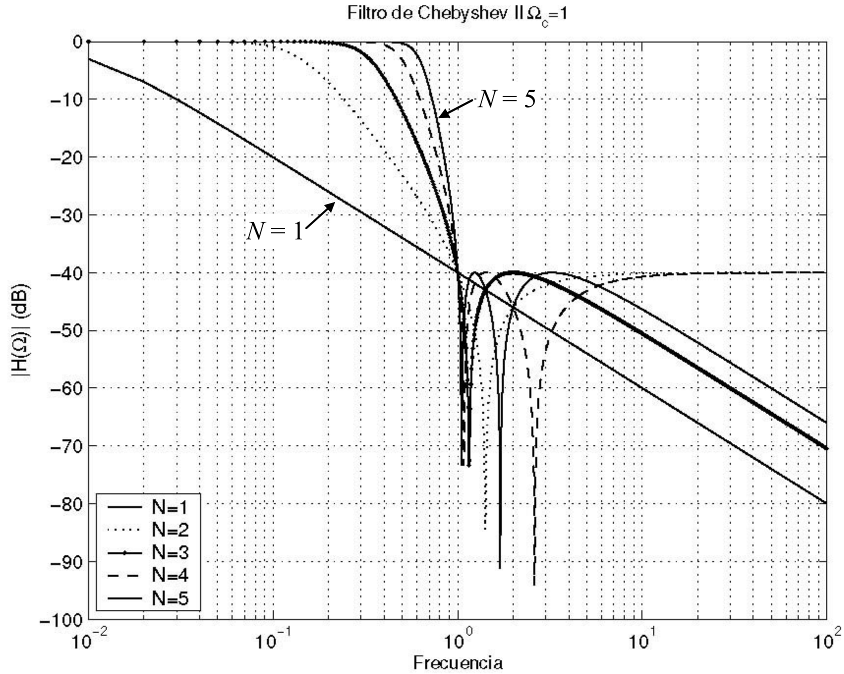
_ Su función de transferencia sólo tiene polos (no tiene ceros)

En la parte derecha se muestra, a modo de ejemplo, la representación del módulo de 5 respuestas en frecuencia de filtros paso bajo (en el eje de abscisas se utiliza escala logarítmica). Como se puede apreciar en las figuras representadas, cuanto mayor es el orden (*N*) del filtro, más se aproxima su respuesta en frecuencia a la de un filtro ideal.



▪ **Características de los filtros diseñados utilizando polinomios de *Inverse Chebyshev*** (o de *Chebyshev* de tipo II)

_ La magnitud de su respuesta en frecuencia es constante o monótona decreciente en la banda de paso y en la banda de transición. Mientras que en la banda prohibida presenta un rizado de amplitud constante.
_ Su función de transferencia tiene polos y ceros
A continuación se muestra, a modo de ejemplo, la representación del módulo de 5 respuestas en frecuencia de filtros paso bajo (en el eje de abscisas se utiliza escala logarítmica, en el eje de ordenadas los valores se representan en dBs). Como se puede apreciar en las figuras representadas, cuanto mayor es el orden (N) del filtro, más se aproxima su respuesta en frecuencia a la de un filtro ideal.



▪ **Características de los filtros de *Cauer o elípticos***

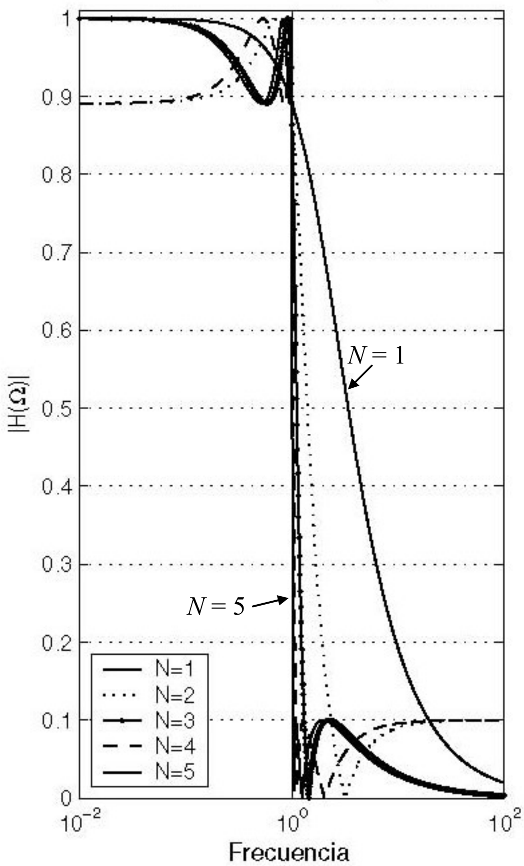
_ La curva de módulos presenta un cierto rizado tanto en la banda de paso como en la banda prohibida.

_ Es el tipo de filtro que presenta una *banda de transición* más estrecha para un orden N dado del filtro.

_ Son el tipo de filtro que presenta una fase menos lineal con la frecuencia, en particular, en el extremo de la banda de paso. Lo que hace que sea el tipo de filtro que introduce una mayor distorsión armónica de fase.

_ No resulta adecuado para aplicaciones de audio

En la parte derecha se muestra, a modo de ejemplo, la representación del módulo de 5 respuestas en frecuencia de filtros paso bajo.

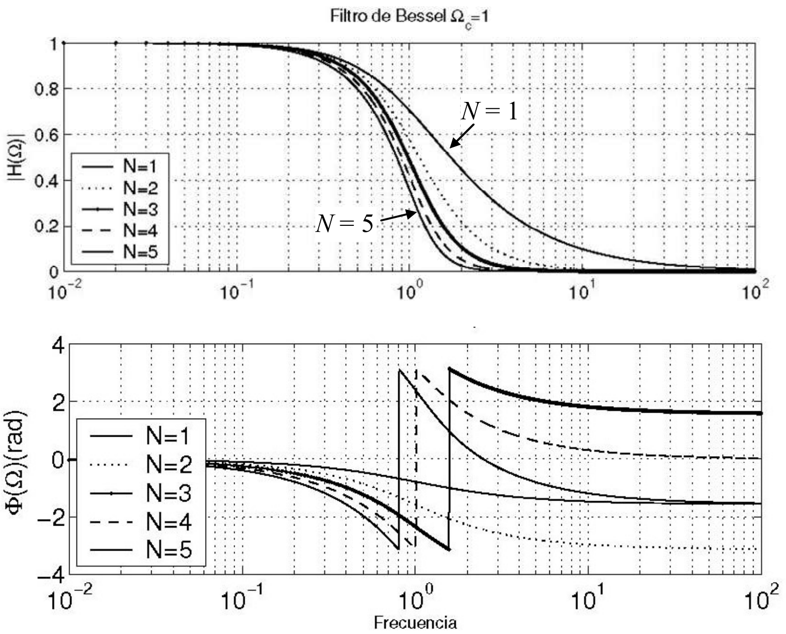


▪ **Características de los filtros de *Bessel***

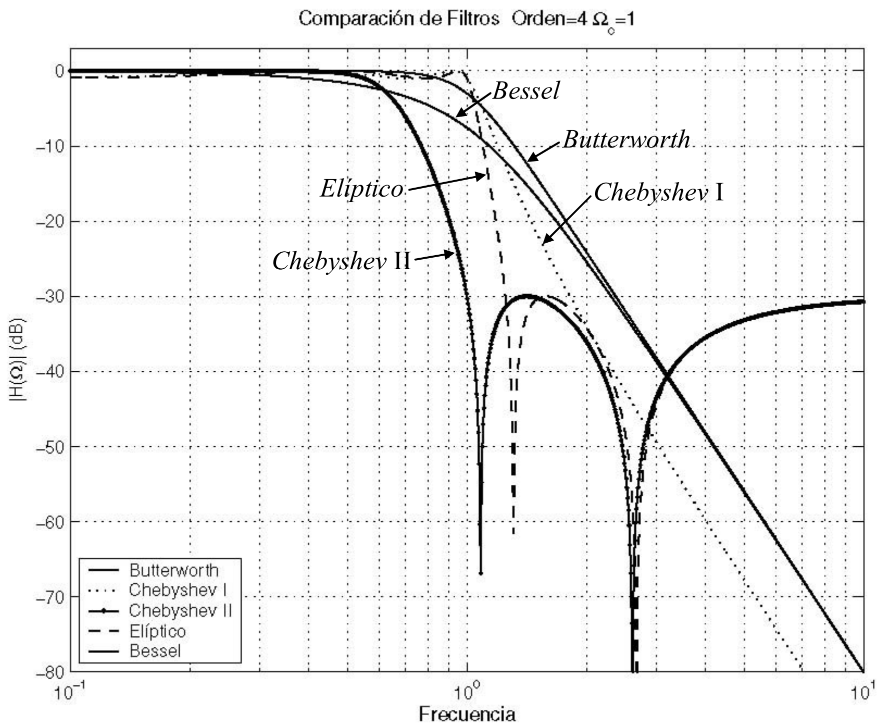
_ El objetivo de estos filtros no es la de minimizar el ancho de la banda de transición, sino que su objetivo es el de conseguir que la curva de ángulos de su respuesta en frecuencia sea *lineal* con la frecuencia en la banda de paso (con los demás tipos de filtros la curva de ángulos en la banda de paso se aleja del comportamiento lineal a medida que la frecuencia se aproxima a la frecuencia de corte). El que este tipo de filtro presente una curva de ángulos lineal hace que no introduzcan distorsión armónica de fase y que resulten adecuados para introducir retardos en señales.

_ Este tipo de filtros deja de tener una fase lineal con la frecuencia al discretizarlos. De ahí que se hayan desarrollado los filtros discretos de tipo FIR (≡ *finite impulse response*)

_ La función de transferencia de este tipo de filtros se caracteriza porque sólo tiene polos.



A continuación se representa el módulo de la respuesta en frecuencia de 5 filtros *paso bajo* correspondientes a los tipos indicados en los párrafos anteriores. Todas las respuestas en frecuencia corresponden a filtros de orden $N = 4$



Ejemplo de diseño de un filtro paso bajo utilizando el programa WFilter

El filtro que se va a diseñar a continuación está pensado para eliminar todas las componentes en frecuencia de una señal de audio (que se aplicará a la entrada del filtro) cuya frecuencia sea superior a 6366 hercios. De acuerdo con esto, la *banda prohibida* del filtro a diseñar deberá abarcar el intervalo de frecuencias [6333, ∞].
Nota: realmente las componentes en frecuencia cuya frecuencia sea superior a 6366 hercios aparecerán en la salida del filtro pero con un valor de pico tan pequeño que en la práctica se puede considerar que dichas componentes no forman parte de la señal presente en la salida del filtro.

Por otra parte el filtro está pensado para dejar pasar "a través de él" (de su entrada hacia su salida) todas las componentes en frecuencia de la señal presente en su entrada cuya frecuencia sea inferior a 1008 hercios. De acuerdo con esto, la *banda de paso* del filtro a diseñar deberá abarcar el intervalo de frecuencias [0, 1008] hercios.

De los datos indicados en los párrafos anteriores se deduce que la *banda de transición* del filtro a diseñar abarcará el intervalo de frecuencias [1008, 6366] hercios. Nota: cuanto más ancha sea la banda de transición, menor será el orden del filtro (menor será el número de componentes electrónicos que es necesario utilizar para construir el filtro o menor será el número de operaciones matemáticas que tiene que realizar un procesador, en el caso de que el filtro se implemente digitalmente).

Para diseñar un filtro se puede utilizar cualquiera de los innumerables programas que existen. En este ejemplo (y en esta tarea) se va a utilizar el programa *WFilter* (está en *faitic*). Dicho programa fue creado por el profesor *Les Thede* y lo facilita junto con el libro del que es autor, titulado: *Practical analog and digital filter design* (Edit.: *Artech House*).

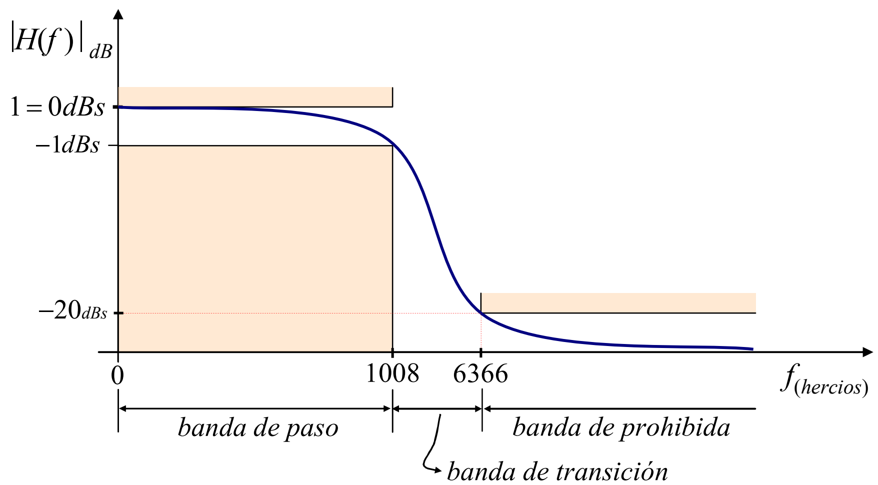
Teniendo en cuenta los datos indicados en los párrafos anteriores, a continuación se va a diseñar un filtro *paso bajo* de tipo *Butterworth* caracterizado por lo siguiente:

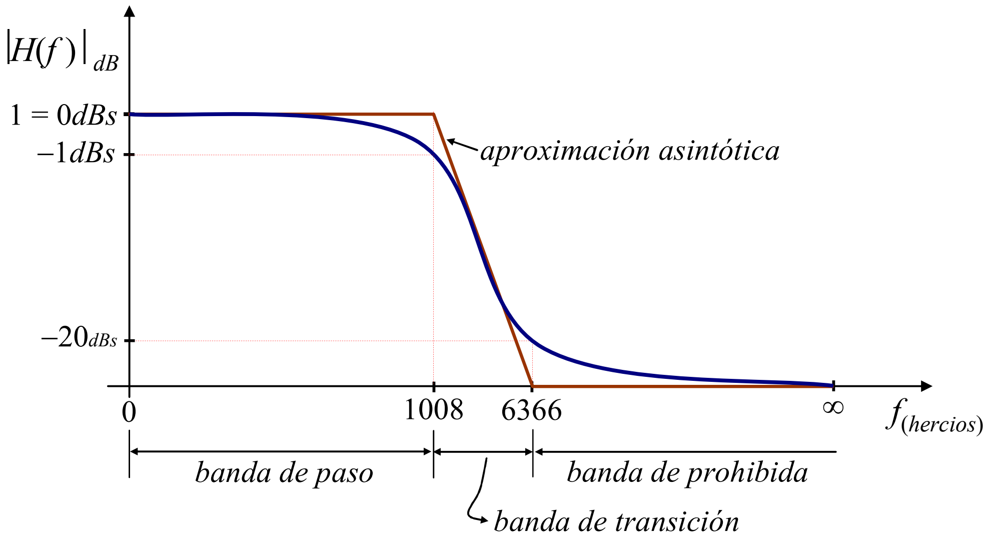
_ El valor mínimo del módulo de la respuesta en frecuencia del filtro en su banda de paso (≡ valor mínimo de la *ganancia del filtro* en la banda de paso ≡ *Gain in Passband Specification*$_{WFilter}$) es de −1dB (≡ $10^{-1/20} \simeq 0{,}89$).
Nota: $A_{\mathrm{dBs}} = 20 \cdot \log_{10}(A)$.

_ La *frecuencia de corte* del filtro es igual a 1008 hercios [≡ *Edge Freq.* (Hz) *in Passband Specification*$_{WFilter}$]. La *banda de paso* del filtro abarcará el intervalo [0, 1008] hercios.

_ El valor mínimo del módulo de la respuesta en frecuencia del filtro en su *banda prohibida* (≡ valor mínimo de la *ganancia del filtro* en la banda prohibida ≡ *Gain in Stopband Specification*$_{WFilter}$) es de –20dB (≡ $10^{-20/20} = 0{,}1$). Al tratarse de un filtro de tipo *Butterworth*, este parámetro establece la atenuación mínima que introducirá el filtro a las señales senoidales (≡ componentes en frecuencia) presentes en la entrada del filtro, cuya frecuencia pertenezca a la banda prohibida del filtro.

_ La frecuencia en la que comienza la banda prohibida del filtro *paso bajo* a diseñar [≡ *Edge Freq.* (*Hz*) *in Stopband Specification*$_{WFilter}$) es igual a 6366 hercios.





Muy importante: los datos relativos al tipo de filtro (*paso bajo*), aproximación utilizada (*Butteworth*), frecuencias que delimitan la banda de paso (1008 Hz) y la banda prohibida (6366 Hz), atenuación máxima en la banda de paso (1dB) y la atenuación mínima en la banda prohibida (−20dBs) se deciden en función de las características de la señal que se aplicará en la entrada del filtro. Una vez determinados dichos datos, el diseño de un filtro es algo mecánico… se puede programar. Lo que no se puede programar es la elección de los datos anteriores.

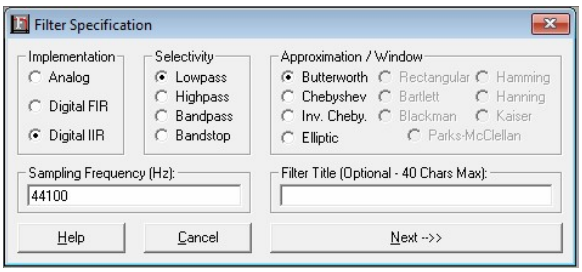
Recordatorio 1: idealmente, el objetivo del filtro es eliminar ciertas componentes en frecuencia de la señal de entrada del filtro. En la práctica el filtro lo que hace es atenuar (disminuir) el valor de pico de dichas componentes, de modo que la contribución de dichas componentes a la señal que aparece en la salida del filtro sea completamente despreciable.

Recordatorio 2: en una clase de teoría se explicó que una señal *no periódica* $x(t)$ se puede interpretar como la suma de un número infinito de señales senoidales, cumpliéndose que cada señal senoidal tiene una frecuencia distinta y, en general, un valor de pico y una fase distintos. Cada una de dichas señales senoidales es una de las *componentes en frecuencia* de la señal $x(t)$.

La principal aplicación de los filtros (o, mejor dicho, una de sus principales aplicaciones) es eliminar "algunas" de las *componentes en frecuencia* de una señal (de audio, de video, etc.). Un filtro se diseña de modo que su banda prohibida abarque a todas las senoides (componentes en frecuencia) que se deseen eliminar (la banda de paso debe abarcar todas o la mayor parte de las senoides que corresponden a "información"… que no son ruido y que se quieren conservar y que, por lo tanto, deben aparecer en la salida del filtro). A veces hay componentes en frecuencia de una señal que está formadas por la suma de una componente que forma parte de la información + una componente que es ruido. En tales casos la eliminación de la componente que es ruido conlleva la eliminación de la componente que es ruido.

Una vez instalado el programa *WFilter*, los pasos a dar para diseñar el filtro (digital IIR) indicado en los párrafos anteriores son:

1º: *File* → *New*: Implementation: *Digital IIR*
Selectivity: *Lowpass*
Sampling frequency (Hz): 44100
Approximation/window: *Butterworth*
clic en *Next*

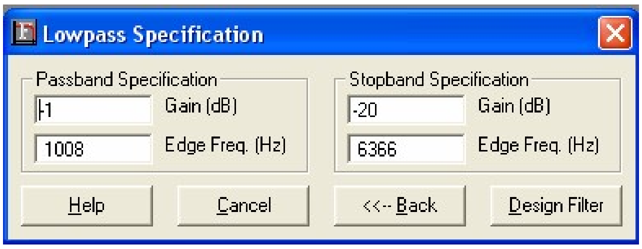


Nota: se supone que la señal a filtrar se muestrea con una frecuencia de muestreo (*sampling frequency*) de 44100 hercios.

2º: En la siguiente ventana hay que indicar los datos (ganancias/atenuaciones y frecuencias) que caracterizan al filtro IIR a diseñar (ver figura de la derecha)

Passband specification: Gain (dB): –1dB
Edge Freq (Hz): 1008 Hz

Stopband specification: Gain (dB): –20 dB
Edge Freq (Hz): 6366 Hz



clic en *Design Filter*

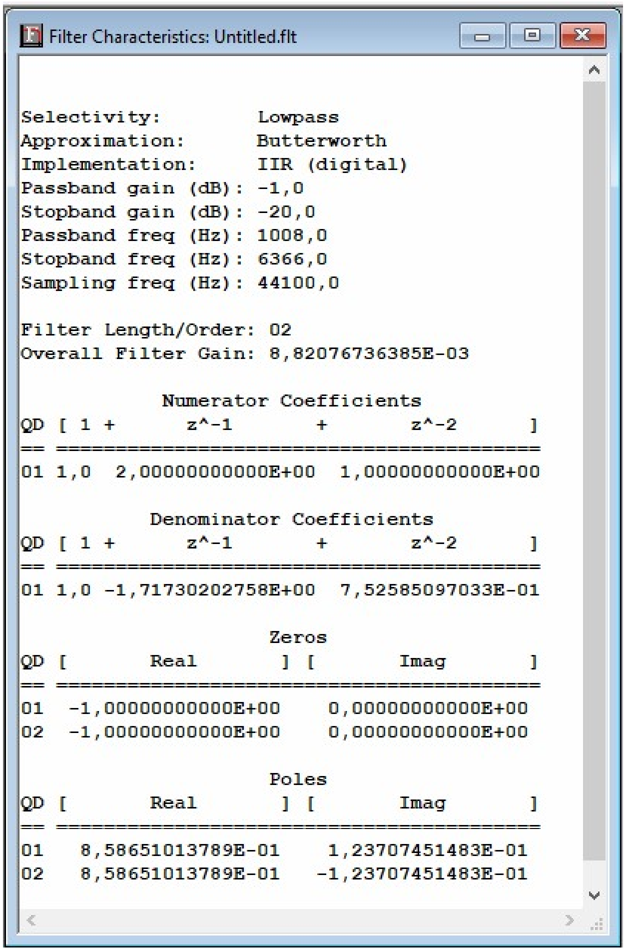
En la pantalla se abre una ventana como la indicada en la parte derecha. En la parte superior de dicha ventana se muestran los datos del filtro diseñado (≡ los datos que se le han proporcionado al programa). A continuación se muestran los datos relativos al filtro diseñado por el programa *WFilter* a partir de dichos datos:

_ *Filter length/Order* = 2 indica lo que se conoce como orden del filtro (≡ es el orden de la función de transferencia que modela el comportamiento del filtro)

_ Los siguientes datos corresponden a la *función de transferencia* del filtro en el plano Z, cumpliéndose que:

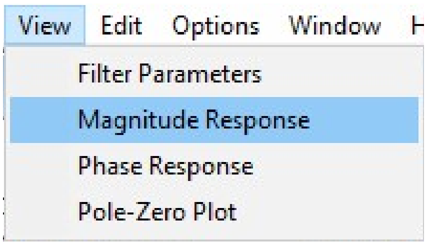
$$H(z)=\frac{8{,}82{\cdot}10^{-3}{\cdot}(1+2{\cdot}z^{-1}+z^{-2})}{1-1{,}71{\cdot}z^{-1}+7{,}52{\cdot}10^{-1}{\cdot}z^{-2}}$$

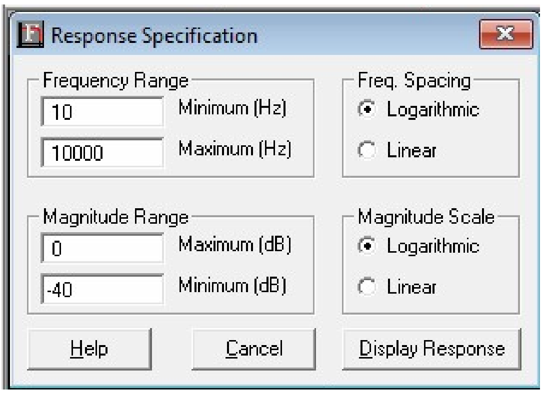
Nota: la función de transferencia es una función que modela el comportamiento del filtro, en este caso en el plano Z.

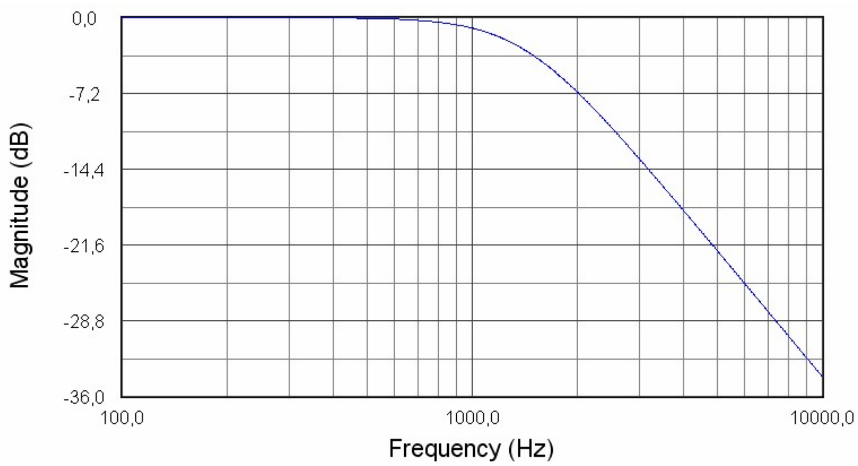


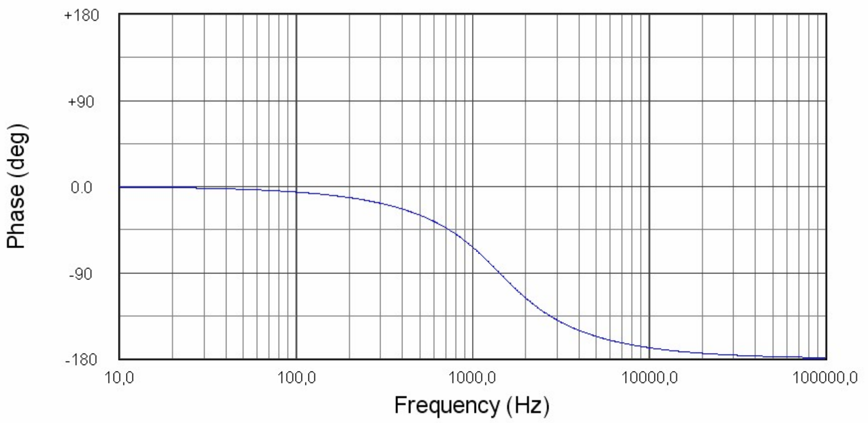
```
Filter Characteristics: Untitled.flt

Selectivity:          Lowpass
Approximation:        Butterworth
Implementation:       IIR (digital)
Passband gain (dB):   -1,0
Stopband gain (dB):   -20,0
Passband freq (Hz):   1008,0
Stopband freq (Hz):   6366,0
Sampling freq (Hz):   44100,0

Filter Length/Order:  02
Overall Filter Gain:  8,82076736385E-03

             Numerator Coefficients
QD [ 1 +      z^-1         +        z^-2       ]
== ========================================
01 1,0  2,00000000000E+00  1,00000000000E+00

             Denominator Coefficients
QD [ 1 +      z^-1         +        z^-2       ]
== ========================================
01 1,0 -1,71730202758E+00  7,52585097033E-01

                     Zeros
QD [         Real        ] [         Imag      ]
== ========================================
01  -1,00000000000E+00     0,00000000000E+00
02  -1,00000000000E+00     0,00000000000E+00

                     Poles
QD [         Real        ] [         Imag      ]
== ========================================
01   8,58651013789E-01     1,23707451483E-01
02   8,58651013789E-01    -1,23707451483E-01
```

Para que el programa *WFilter* represente la curva de módulos y la curva de ángulos de la respuesta en frecuencia del filtro diseñado hay que ir a *View* y hacer clic con el ratón en *Magnitude Response*. A continuación se abrirá una ventana denominada *Response Specification* en la que se puede indicar el rango de frecuencias a representar en el eje de abscisas así como el rango de valores a representar en el eje de ordenadas (en dBs si la *Magnitude Scale* es logarítmica). A continuación sólo hay que hacer clic en *Display Response*. En la ventana aparecerán representadas la curva de módulos (*Magnitude Response*) y la curva de ángulos (*Phase Response*) de la respuesta en frecuencia del filtro diseñado.

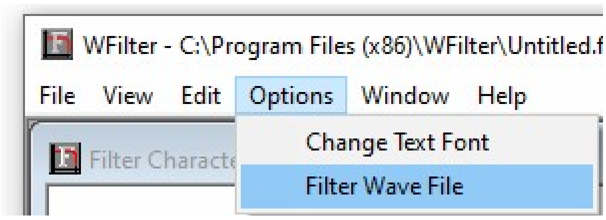








Si la señal a filtrar fuese una señal de audio, guardada en un archivo con formato *wav*, el programa *WFilter* permite ver (escuchar) el efecto del filtro diseñado sobre dicha señal. Para ello sólo hay que ir a la pestaña *Options* y hacer clic con el ratón en la opción *Filter Wave File*.

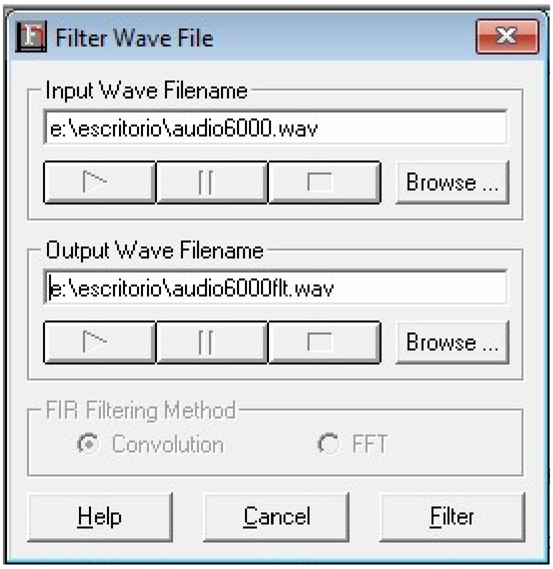


A continuación se abre una ventana denominada *Filter Wave File* en la que:

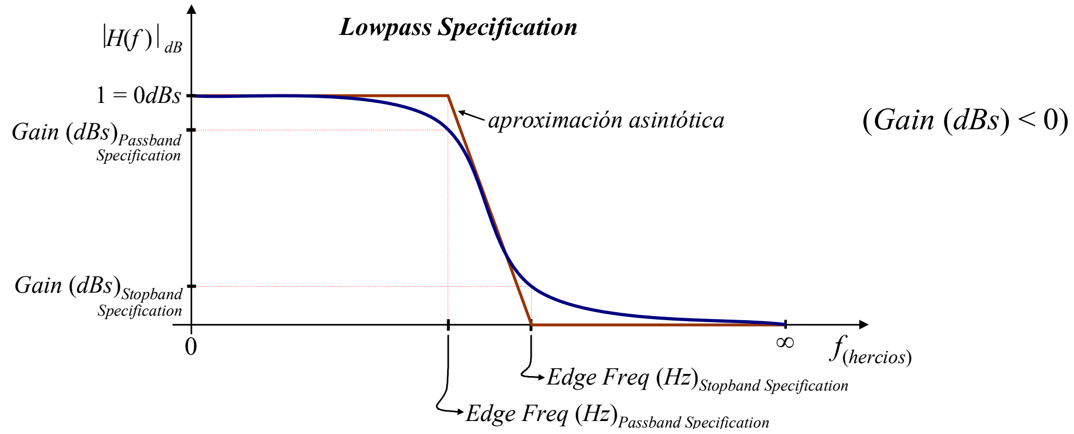
_ En *Input Wave Filename* hay que indicar el archivo *wav* que contiene la señal a filtrar y en dónde está guardado dicho archivo.

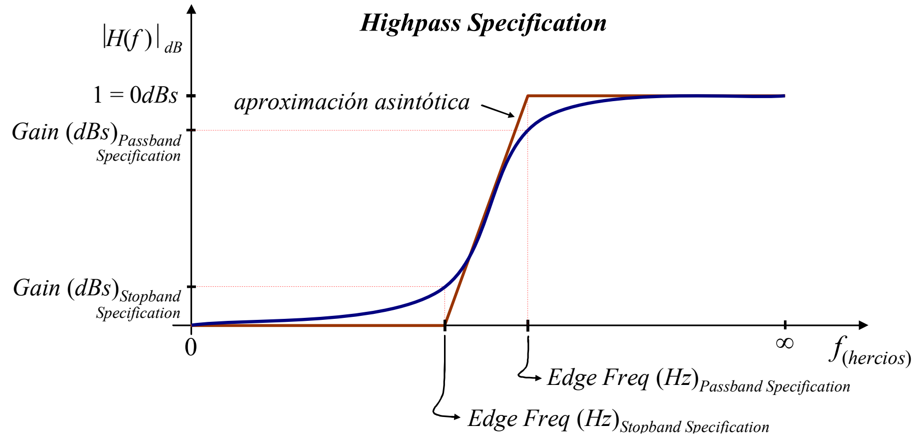
_ En *Output Wave Filename* hay que indicar en dónde se quiere que el programa guarde el archivo *wav* con las muestras de la señal de audio que genera el filtro diseñado.

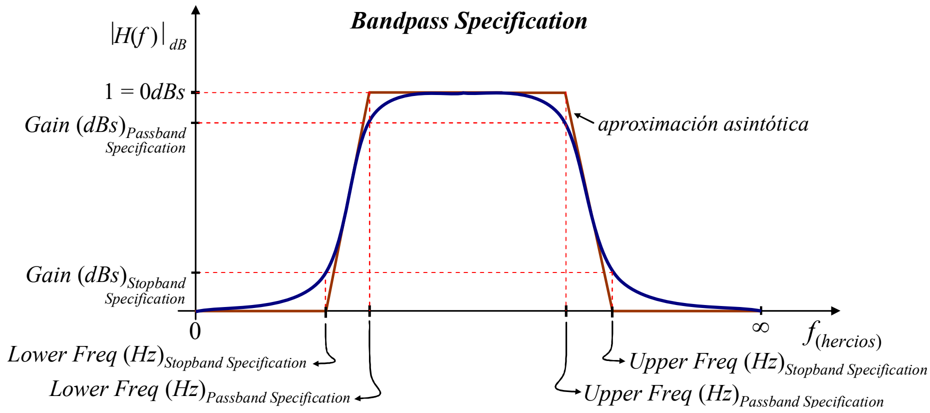
A continuación sólo hay que hacer clic con el ratón en *Filter* y el programa permite reproducir los archivos de entrada y de salida sin más que hacer clic en el correspondiente botón de *play* (▶)

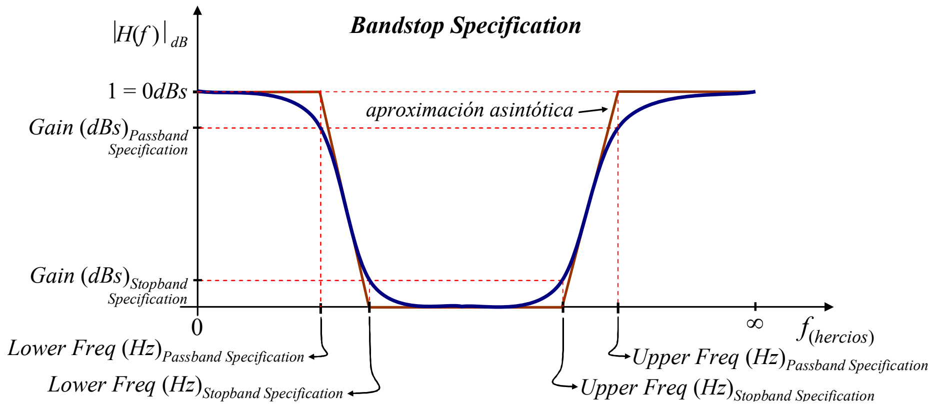


▪ A continuación se explica la nomenclatura que utiliza el programa WFilter para referirse a las *frecuencias* y *ganancias características* de los distintos tipos de *respuestas en frecuencia* que puede calcular.









▪ A la hora de filtrar una señal de audio hay que tener en cuenta lo siguiente:

_ Teóricamente, un oído humano sano puede oír señales senoidales (tonos) de frecuencias comprendidas entre 20Hz y 20kHz.

_ Los tonos entre ≈ 20 y ≈ 64Hz son tonos graves, pero no los oímos todas las personas.

_ Los tonos entre ≈ 65 y ≈ 250Hz son tonos graves medios.

_ Los tonos entre ≈ 250 y ≈ 2000Hz contienen el tono fundamental y los primeros armónicos de la mayoría de las fuentes sonoras.

_ Los tonos entre ≈ 2000 y ≈ 4096Hz corresponden al rango de tonos al que el oído humano tiene una mayor sensibilidad.

_ Los tonos entre ≈ 4097 y ≈ 16000Hz corresponden a sonidos desagradables.

_ Los tonos superiores a ≈ 16000Hz no todas las personas los oímos.

En resumen: el rango de frecuencias de la voz humana que contiene la mayor parte de la información relevante está comprendido entre los 200Hz y los 4000Hz.

Teniendo en cuenta lo indicado en los párrafos (y páginas) anteriores, se pide:

a) Diseña un filtro IIR que elimine el ruido en una señal de audio cuyas muestras están guardadas en un archivo denominado *Audio*6000.*wav* y que está disponible en *faitic*. Los pasos a dar en este apartado se resumen en lo siguiente:

1º: Reproduce el archivo de audio con el fin de comprobar que la señal tiene ruido (pitido).

2º: Sabiendo que el contenido en frecuencia del ruido (pitido) que se escucha de fondo está concentrado a 6000 hercios, diseña un filtro para eliminar dicho ruido utilizando el programa WFilter. La frecuencia con la que se han tomado las muestras guardadas en el archivo *Audio*6000.*wav* es de 44100 hercios (*Sampling frequency*)

3º: Comprueba que la respuesta en frecuencia del filtro diseñado presenta las características deseadas.

4º: Utilizando el programa *WFilter*, reproduce el archivo de audio que guarda las muestras de la señal de salida del filtro diseñado con el fin de comprobar su efectividad.

Como resultado de este ejercicio debes de proporcionar los siguientes datos del filtro diseñado:

_ Selectivity:

_ Approximation:

_ Implementation:

_ Passband gain (dB):

_ Stopband gain (dB):

_ Passband freq (Hz):

_ Stopband freq (Hz):

_ Sampling freq (Hz):

_ Filter Length/order:

_ Overall Filter Gain:

b) En este apartado hay que diseñar un filtro IIR de tipo *paso alto* cuya banda de transición esté comprendida entre 4000Hz y 10000Hz. Usando el programa WFilter escucha el efecto del filtro diseñado sobre la señal de audio cuyas muestras se guardan en el archivo *Audio*6000.*wav*. Explica el efecto del filtro sobre la señal de audio de acuerdo con las características del filtro diseñado. Indica los parámetros del filtro que has diseñado (Selectivity, Approximation, Implementation, Passband gain (dB), Stopband gain (dB), Passband freq (Hz), Stopband freq (Hz), Sampling freq (Hz), Filter Length/order, Overall Filter Gain)   Nota: *Hz ≡ Hertz ≡ Hercios*

c) En este apartado hay que diseñar un filtro IIR de tipo *paso bajo* cuya banda de transición esté comprendida entre 100Hz y 5000Hz. Usando el programa WFilter escucha el efecto del filtro diseñado sobre la señal de audio cuyas muestras se guardan en el archivo *Audio*6000.*wav*. Explica el efecto del filtro sobre la señal de audio de acuerdo con las características del filtro diseñado. Indica los parámetros del filtro que has diseñado (Selectivity, Approximation, Implementation, Passband gain (dB), Stopband gain (dB), Passband freq (Hz), Stopband freq (Hz), Sampling freq (Hz), Filter Length/order, Overall Filter Gain)   Nota: *Hz ≡ Hertz ≡ Hercios*

d) En este apartado hay que diseñar un filtro que sólo deje pasar el ruido (pitido). Indica los parámetros del filtro que diseñes para tal fin (Selectivity, Approximation, Implementation, Passband gain (dB), Stopband gain (dB), Passband freq (Hz), Stopband freq (Hz), Sampling freq (Hz), Filter Length/order, Overall Filter Gain)

Las respuestas a los apartados anteriores deben ser enviadas al correo cmiguens@uvigo.es antes del 16 de mayo.