ESCUELA TÉCNICA SUPERIOR DE INGENIEROS
INDUSTRIALES Y DE TELECOMUNICACIÓN

UNIVERSIDAD DE CANTABRIA

Proyecto Fin de Carrera

Diseño, simulación y validación de filtros activos de Sallen-Key y de Rauch en el entorno de matlab
(Design, simulation and validation of active filters Sallen-Key and Rauch in matlab environment)

Para acceder al Titulo de

INGENIERO TÉCNICO DE TELECOMUNICACIÓN

Autor: Alberto Portilla Agüero
Octubre-2014

E.T.S. DE INGENIEROS INDUSTRIALES Y DE TELECOMUNICACION

INGENIERÍA TÉCNICA DE TELECOMUNICACIÓN

CALIFICACIÓN DEL PROYECTO FIN DE CARRERA

Realizado por: Alberto Portilla Agüero
Director del PFC: Tomás Fernández Ibáñez

Título: "Diseño, simulación y validación de filtros activos de Sallen-Key y de Rauch en el entorno de matlab"
Title: "Design, simulation and validation of active filters Sallen-Key and Rauch in matlab environment"

Presentado a examen el día: Martes 28 de Octubre de 2014

para acceder al Título de

INGENIERO TÉCNICO DE TELECOMUNICACIÓN, ESPECIALIDAD EN SISTEMAS ELECTRÓNICOS

Composición del Tribunal:
Presidente (Apellidos, Nombre): Tazón Puente, Antonio
Secretario (Apellidos, Nombre): Fernández Ibáñez, Tomás
Vocal (Apellidos, Nombre): Valle López, Luis

Este Tribunal ha resuelto otorgar la calificación de: ....

Fdo.: El Presidente

Fdo.: El Secretario

Fdo.: El Vocal

Fdo.: El Director del PFC
(sólo si es distinto del Secretario)

Vº Bº del Subdirector

Proyecto Fin de Carrera Nº
(a asignar por Secretaría)

Agradecimientos

En este momento quisiera agradecer en primer lugar, a mi director de proyecto Tomás Fernández Ibáñez por su inestimable ayuda y sin el cual este trabajo no hubiera sido posible.

También a mis padres y a mis hermanos su paciencia infinita y todos los sacrificios que les ha supuesto darme toda la formación que ahora tengo.

A mi amigo Moro por sus ánimos y su optimismo en cualquier circunstancia y a todos aquellos que pasaron por mi vida y me aportaron muchas nuevas experiencias.

A los compañeros de la sala de investigación de microondas, a Nael, David, Quique y en especial a Javi por su ayuda desinteresada siempre y todos los buenos momentos vividos.

A la gente que conocí en el Máster, Eusebio, Rubén y cía.

Y por supuesto dedicado con mucho cariño a Carmen por su apoyo, su comprensión y su paciencia a lo largo de todo el proyecto aquí desarrollado.

ÍNDICE

CAPÍTULO I. Introducción. **1**

CAPÍTULO II. Funciones de Red. **3**

2.1. Introducción. 3

2.2. Propiedades generales de las funciones de red. 3

CAPÍTULO III. Introducción a la Teoría General de Filtros .. **7**

3.1. Origen y evolución histórica. 7

3.2. Generalidades. 9

3.2.1 Características. 9

3.2.1.1. Función de transferencia. 9
3.2.1.2. Función de pérdidas. 10
3.2.1.3. Respuesta en frecuencia. 10
3.2.1.4. Orden. 10
3.2.1.5. Frecuencia de corte. 11
3.2.1.6. Frecuencia central. 11
3.2.1.7. Ancho de banda 3 dB. 12

3.2.1.8. Rizado. 12
3.2.1.9. Factor de calidad. 13
3.2.1.10. Retardo de fase y retardo de grupo. ... 15

3.2.2 Clasificación. 18

3.2.2.1. Según las frecuencias atenuadas. 18
3.2.2.2. Según la tecnología empleada. 20
3.2.2.3. Según la función matemática utilizada. . 24

CAPÍTULO IV. Filtros Activos de Sallen – Key y de Rauch. . 34

4.1. Filtros de Sallen – Key. 34

4.1.1. Etapa paso bajo de primer orden. 35
4.1.2. Etapa paso bajo de segundo orden. 37
4.1.3. Etapa paso alto de primer orden. 40
4.1.4. Etapa paso alto de segundo orden. 42

4.2. Filtros de Rauch. 46

4.2.1. Etapa paso bajo de primer orden. 46
4.2.2. Etapa paso bajo de segundo orden. 49
4.2.3. Etapa paso alto de primer orden. 53
4.2.4. Etapa paso alto de segundo orden. 56

4.3. Implementación del filtro diseñado utilizando celdas básicas Sallen – Key o Rauch. 60

CAPÍTULO V. Descripción del software desarrollado. 63

5.1. Introducción a Matlab. 63

5.2. Creación del interfaz gráfico (GUIDE). 63

5.2.1. Funcionamiento del GUI. 64

5.3. Descripción del Programa de Control. 65

5.3.1. Requisitos previos. 65

5.3.2. Especificaciones. 67

5.3.3. Arquitectura. 68

5.3.4. Descripción del entorno gráfico y de las funciones más importantes. 73

CAPÍTULO VI. Ejemplos de filtros activos en Matlab y Spice. .. 97

6.1. Filtro paso bajo Butterworth de 2º orden implementado con celdas de Sallen - Key. 98

6.2. Filtro paso alto Chebyshev de 3º orden con rizado 2 dB implementado con celdas de Sallen - Key. ... 104

6.3. Filtro paso banda Bessel de 5º orden implementado con celdas de Sallen - Key. 111

6.4. Filtro paso bajo Chebyshev de 1º orden con rizado 3 dB implementado con celdas de Rauch. 119

6.5. Filtro paso alto Bessel de 4º orden implementado con celdas de Rauch. 126

6.6. Filtro paso banda Butterworth de 5º orden implementado con celdas de Rauch. 134

CAPÍTULO VII. Conclusiones y Líneas Futuras. 144

Anexo. Tabla de Coeficientes Butterworth, Chebyshev y Bessel. 149

Bibliografía. 156

CAPÍTULO I

INTRODUCCIÓN

Antes de comenzar expondremos a modo de resumen, las líneas maestras que regirán nuestro proyecto.

En el trabajo aquí desarrollado presentaremos una herramienta software creada para el diseño y la simulación de filtros activos, la cual recibe el nombre de FILACT.

El programa utilizado para ello ha sido Matlab debido a la gran versatilidad y a las múltiples posibilidades que ofrece en lo que a interfaces gráficas se refiere. En este sentido, el software fue diseñado a través de la herramienta GUIDE empleada en generar varias GUIS (Interfaces Gráficas de Usuario) en Matlab.

Estas GUIS son una serie de paneles o ventanas gráficas que permiten la interacción del programa con dicho usuario, de tal manera que bajo este punto de vista, la programación en Matlab se hace transparente.

El programa podemos dividirlo en tres partes. Un primer bloque está basado en el diseño y simulación del filtro que se ha especificado por parte del usuario.

En este apartado podrá elegirse el tipo de filtro que se desea (paso bajo, paso alto o paso banda), la función de aproximación con la que se pretende trabajar (Butterworth, Chebyshev con diferentes valores de rizado o Bessel) así como las celdas básicas o circuitos con las que implementaremos nuestro filtro (celdas Sallen-Key o Rauch).

Una vez seleccionado el filtro concreto que se prefiere desarrollar, podrá introducirse por teclado algunas de sus características, tales como el orden (de 1 a 6), las frecuencias de corte, la ganancia si la hubiera (filtros Rauch), el factor de rizado (en el caso de los Chebyshev) etc.

De la misma manera podrán decidirse el valor de algunos de los componentes que formarán el filtro (lease resistencias y condensadores).

Superada la fase de diseño, el software permite simular el filtro seleccionado con anterioridad. En este punto, el usuario tendrá la posibilidad de visualizar el diagrama de bode (representación gráfica de la ganancia y de la fase) del filtro en cuestión y de almacenar en su caso los resultados de dicho diagrama en un fichero.

Así mismo también se ofrecerá la opción de guardar todos los datos de la aplicación (características, valor de los componentes que conforman el diseño, así como de los puntos visualizados tanto de ganancia como de fase a una determinada frecuencia).

Por último y como complemento a la labor del software desarrollado, nos planteamos la posibilidad de verificar los resultados obtenidos utilizando la aplicación PSPICE, herramienta a través de la cual realizaremos la simulación de diversos ejemplos de filtros programados previamente en Matlab, con el objetivo de poder contrastar ambos análisis y validar en su caso el trabajo efectuado.

En este sentido, la comparación anterior nos permitió extraer una serie de conclusiones que quedan reflejadas en el capítulo final de este proyecto.

CAPÍTULO II

FUNCIONES DE RED

2.1. – Introducción

En este primer capítulo se introducirán algunas propiedades básicas de las funciones de transferencia que serán utilizadas más adelante para definir los filtros planteados.

Dichas funciones de transferencia forman parte de un grupo más amplío de funciones, llamadas *funciones de red*.

Las ***funciones de red*** son todas aquellas funciones matemáticas que se obtienen mediante el análisis de una red lineal, invariante y de parámetros localizados, siendo racionales en la variable ***s***.

Esta variable ***s*** puede interpretarse como la variable que aparece en la transformada de Laplace, en cuyo caso *s* es un número complejo y la función de red también lo es. Aquí se deberán tener en cuenta las condiciones iníciales de la red y añadirse al circuito como fuente de tensión o de corriente.

Pero además, **s** puede ser el operador derivada, en este supuesto la función de red es un operador. Ahora las tensiones y corrientes son funciones reales de la variable real tiempo, mientras que en el caso anterior son sus transformadas.

Para corrientes alternas estacionarias se puede escribir ***s = jω***, dependiendo las funciones de red complejas de la variable real *ω*. Así mismo, tanto las tensiones como las corrientes son constantes complejas.

En nuestro caso, consideraremos a *s* como un número complejo, sabiendo que *s = jω*.

2.2. – Propiedades generales de las funciones de red

Una función de red en general, es el cociente de dos polinomios en la variable *s* con coeficientes reales, es decir:

$$\boxed{H(s) = \frac{\sum_{K=0}^{N} A_K * p^K}{\sum_{K=0}^{M} B_K * p^K}} \quad \text{con } A_K, B_k \text{ reales.} \qquad (2.1)$$

Esta función tiene las siguientes propiedades:

* *El diagrama de polos*[1] *y ceros*[2] *es siempre simétrico respecto del eje real del plano complejo*.

Esto se debe a que las raíces de un polinomio son reales o parejas complejas conjugadas.

Un ejemplo práctico de una función de red es:

$$H(s) = \frac{s^3+2*s^2+7*s+2}{s^4+5*s^3+8*s^2+s+1}$$

Resolviendo esta función, los ceros y los polos obtenidos son:

Z_{12} = - 0.8456 ± 2.40057 j; P_{12} = - 2.476988 ± 1.22705 j;

Z_3 = - 0.3087; P_{34} = - 2.30116*10^{-2} ± 0.361027

Los polos están representados mediante aspas y los ceros de la función son los círculos. El eje de abscisas simula la parte real de *s*, siendo representado con la letra sigma (*σ*), mientras que el eje de ordenadas corresponde a la parte imaginaria de *s*, identificada con la letra omega (ω).

De esta manera, la variable *s* se expresa como $s = \sigma + j\,\omega$. Si la parte real $\sigma = 0$, la variable *s* es imaginaria pura, siendo este el caso de la corriente alterna estacionaria.

[1] *Los polos de una función de red son todos aquellos valores para los cuales, se anula el denominador de dicha función.*

[2] *Se llaman ceros a todos los valores que anulan el numerador de la función en cuestión.*

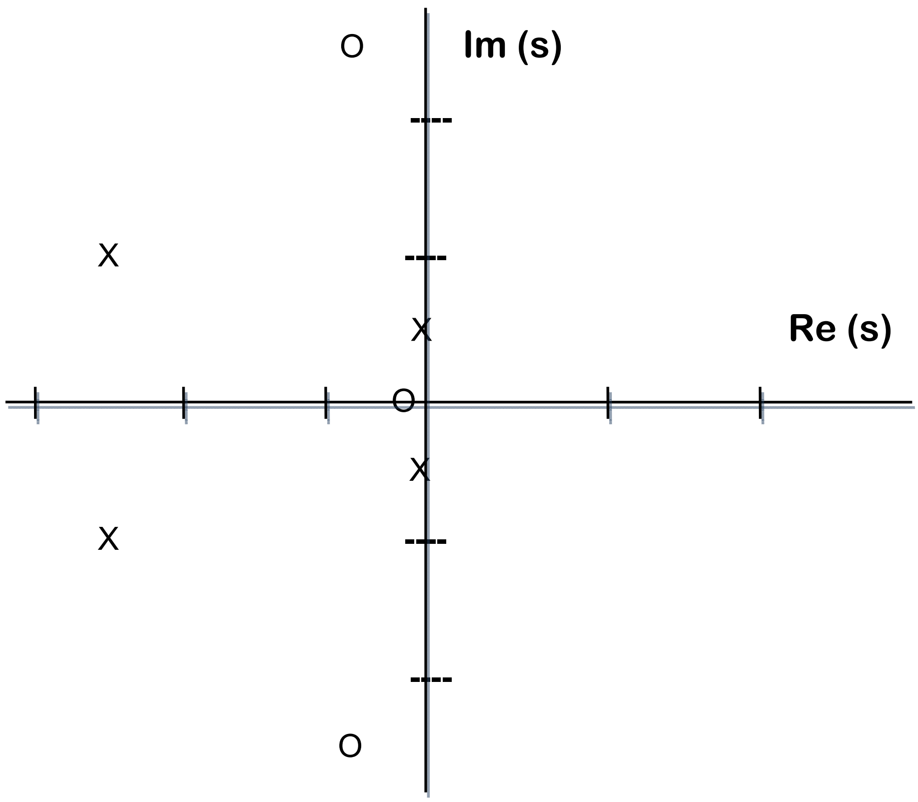


Figura 2.1: Diagrama de polos y ceros simétrico respecto al eje real.

* *Los polos de una función de transferencia están siempre en la parte izquierda del plano complejo, sea cual sea su grado de multiplicidad y si están sobre el eje imaginario, son siempre simples.*

* *Si los polos de la función de transferencia son reales y negativos, podemos asegurar que la red es estable.*

* *Si s = jω, el módulo es función par en ω y el argumento es función impar. Así mismo, la parte real es par y la parte imaginaria es impar.*

Sabemos que las funciones par e impar son:

$$\text{Par } H(s) = \frac{H(s) + H(-s)}{2} \qquad \text{Impar } H(s) = \frac{H(s) - H(-s)}{2} \qquad (2.2)$$

En ellas vemos que Par H(s) es una función par porque no cambia al sustituir *s* por *–s*. De la misma manera, Impar H(s) es una función impar porque cambia de signo cuando convertimos *s* en *–s*.

Por otro lado, si hacemos *s = jω*, por definición de números complejos que la parte real y la parte imaginaria de H(jω) son:

$$\text{Re } H(j\omega) = \frac{H(j\omega) + H(j\omega)^*}{2} \qquad \text{Imag } H(j\omega) = \frac{H(j\omega) - H(j\omega)^*}{2j} \qquad (2.3)$$

siendo $H(j\omega)^* = H(-j\omega)$, el conjugado de $H(j\omega)$.

Comparando ambas ecuaciones con las funciones par e impar anteriormente reseñadas, obtenemos las siguientes relaciones:

$$\text{Par } H(j\omega) = \text{Re } H(j\omega); \qquad \text{Impar } H(j\omega) = j \text{ Imag } H(j\omega); \qquad (2.4)$$

De aquí, podemos deducir que la parte real de H(jω) es una función par y la parte imaginaria es impar.

Y por tanto, como consecuencia, el módulo de H(jω) es par y el argumento impar:

$$|H(j\omega)| = \sqrt{[Re\ H(j\omega)]^2 + [Imag\ H(j\omega)]^2}$$

$$\arg H(j\omega) = \frac{Imag\ H(j\omega)}{Re\ H(j\omega)} \qquad (2.5)$$

CAPÍTULO III

INTRODUCCIÓN A LA TEORÍA GENERAL DE FILTROS

En este capítulo realizaremos un estudio pormenorizado de lo que son los filtros, incluyendo su evolución a lo largo de los años, definiremos cuales son algunas de sus características principales, así como el tipo de filtros que existen en la actualidad y las diversas aplicaciones para las cuales están construidos y que nos podemos encontrar en nuestra vida cotidiana.

3.1. – Origen y evolución histórica

El concepto básico en el que se sustentan los filtros eléctricos fue desarrollado originalmente y de forma independiente en ambos casos, por *G.Campbell* en Estados Unidos y por *K.W.Wagner* en Alemania, en 1915.

Con la aparición de la radio en la década 1910-1920, surgió la necesidad de reducir el efecto de ruido de la estática en el radiorreceptor.

Las investigaciones posteriores han seguido 2 líneas de trabajo diferenciadas: una es el diseño a través de *parámetros imagen* y la otra por *pérdidas de inserción*.

El método de *parámetros imagen* se basa en la idea de conectar en cascada varias secciones fundamentales o prototipos con las impedancias imagen acopladas.

Esta forma de diseño recibió un gran impulso con las investigaciones de *Zobel* en 1923, sobre los filtros m-derivados y los de resistencia constante, siendo el método predominante durante más de 30 años.

Sus trabajos en el diseño de filtros fueron revolucionarios y provocaron importantes avances comerciales de la empresa AT&T (para la cual trabajaba), en el campo de la multiplexación por división de frecuencia (FDM). Aun hoy en día, sus hallazgos constituyen la base de la teoría del filtro.

Al principio, se consideraba que las resistencias no variaban con la frecuencia, la reactancia de una bobina aumentaba en proporción a ella y la de un condensador disminuía con relación a la misma.

La combinación LC en serie forma un circuito *resonante* con una impedancia mínima a la frecuencia $\omega = \frac{1}{\sqrt{LC}}$, mientras que la implementación LC en paralelo es como un circuito "*tapón*".

Años después, se comprobó que una secuencia de bobinas en serie y de capacidades en paralelo daban lugar a un *filtro paso bajo* y que la configuración inversa, es decir (C serie, L paralelo), desembocaba en un *filtro paso alto.*

Así mismo, observaron que con la sucesión de *circuitos resonantes* en serie y de *circuitos "tapón"* en paralelo se obtenía un *filtro paso banda*, mientras que si realizaba la combinación inversa lo que se tenía era un *circuito rechazo de banda*.

Cuando surgieron las transmisiones regulares de radio en la década de 1920, *Campbell* y otros desarrollaron el *filtro RLC* utilizando inductores, capacidades y resistencias. A estos filtros se les llamó *filtros pasivos* debido a que se componen de elementos pasivos.

En la década de 1930, *S.Darlington, S.Butterworth* y *E.A.Guillemin* desarrollaron la teoría necesaria para diseñar *filtros pasivos*. En concreto, el filtro paso bajo de *Butterworth* se dio a conocer en *Wireless Engineering*, en 1930.

Este tipo de circuitos pasivos son adecuados si trabajamos en alta frecuencia. Sin embargo, en el rango de las frecuencias bajas, el valor de los condensadores a emplear se incrementa de forma considerable, pudiendo incluso, salirse del rango de valores comerciales en el que nos manejamos.

La posterior incorporación de *dispositivos activos* con la aparición de los transistores y más tarde, de los amplificadores operacionales en un filtro eléctrico, dio lugar a los llamados *filtros activos*.

Puesto que las bobinas son relativamente grandes y pesadas, los *filtros activos* suelen construirse sin ellas, empleando de forma típica sólo amplificadores operaciones, resistencias y condensadores.

Estos filtros, además de conseguir una amplificación de las señales, tienen un buen comportamiento a bajas frecuencias. Sin embargo su sensibilidad es mayor, es decir, son más susceptibles de variar sus características con la tensión aplicada.

Los primeros *filtros activos RC* prácticos se inventaron durante la Segunda Guerra Mundial y se documentaron en un escrito clásico de *R.P. Sallen* y *E.L.Key* (*Sallen* y *Key*, 1955), los cuales dan nombre a una celda básica de circuito activo empleada en nuestro proyecto.

3.2. - Generalidades

Un filtro es un dispositivo por el que pasan señales eléctricas a una cierta frecuencia o rango de frecuencias, mientras previene el paso de otras. Para todas las señales de entrada que pertenezcan a dicho rango, la señal de salida no se verá disminuida y llegará sin atenuación.

Por tanto, se trata de un circuito que tiene una determinada respuesta en frecuencia, tanto en ganancia como en fase. Al no introducir ninguna frecuencia nueva pueden ser considerados como dispositivos lineales.

En este apartado expondremos algunas de las características principales que definen un filtro.

3.2.1 – Características

3.2.1.1 - Función de Transferencia

La función de transferencia de un filtro es la relación existente entre la salida y la entrada (**V_O / V_i**), en el dominio de Laplace.

Es una magnitud compleja que se puede representar por la amplitud (ganancia) y el ángulo de fase en función de la frecuencia.

$$\boxed{\mathrm{H}\,(s) = \frac{V_O}{V_i} = \frac{N(s)}{D(s)}} \qquad (3.1)$$

Siendo ***N(s)*** y ***D(s)*** polinomios en ***s***, se debe cumplir que el orden del numerador sea igual o menor que el orden del denominador.

A su vez, el orden de ***D(s)*** coincide con el del filtro y de forma general, en un filtro activo dicho orden se corresponde con el número de condensadores del circuito.

3.2.1.2 - Función de Pérdidas

La función de pérdidas se define como la inversa de la función de transferencia ($\frac{1}{H(s)}$).

3.2.1.3 - Respuesta en Frecuencia

La respuesta en frecuencia expresa la función de transferencia del filtro y suele representarse gráficamente mediante dos diagramas, uno de magnitud de la función de transferencia $|H(s)|$ y otro de fase de dicha función, en este caso, $\varphi = \arg H(s)$.

En cualquier caso, existen varios métodos para representar dicha respuesta, entre ellos el diagrama de Bode, el diagrama polar y el relativo al logaritmo del módulo en función de la fase.

3.2.1.4 - Orden

El orden de un filtro es el número ***N*** de secciones que se requieren para su implementación. El aumento de ***N*** mejora su respuesta en frecuencia, aproximándola a la ideal.

Por contra, dicho incremento tiene efectos adversos sobre la respuesta temporal y además, provoca que el número de elementos necesarios para su construcción sea mayor.

El valor ***N*** seleccionado debe satisfacer al menos, las especificaciones particulares de la respuesta en el dominio de la frecuencia del filtro.

3.2.1.5 - Frecuencia de Corte

En los filtros paso bajo y paso alto, la frecuencia de corte $\boldsymbol{f_c}$ se corresponde con el punto donde la ganancia en escala lineal cae $\frac{1}{\sqrt{2}}$ respecto de su valor máximo (es decir, 3 dB en escala logarítmica), tal y como se muestra en la figura 3.1:

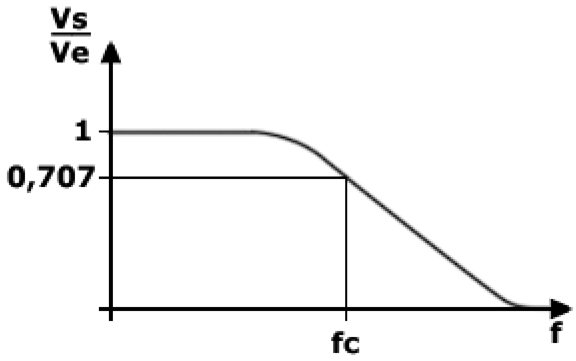


Figura 3.1: Frecuencia de corte de un filtro paso bajo

3.2.1.6 - Frecuencia Central

En los filtros paso banda y rechazo de banda, es la raíz cuadrada del producto entre las frecuencias de corte inferior $\boldsymbol{f_1}$ y superior $\boldsymbol{f_2}$:

$$f_o = \sqrt{f_1 * f_2} \qquad (3.2)$$

Por otra parte en aquellos circuitos con factores de calidad $Q \geq 10$, la frecuencia central $\boldsymbol{f_0}$ se puede aproximar por simetría aritmética como:

$$f_o = \frac{f_1 + f_2}{2} \qquad (3.3)$$

En la figura 3.1, aparece representada la frecuencia central f_0 de un filtro paso banda:

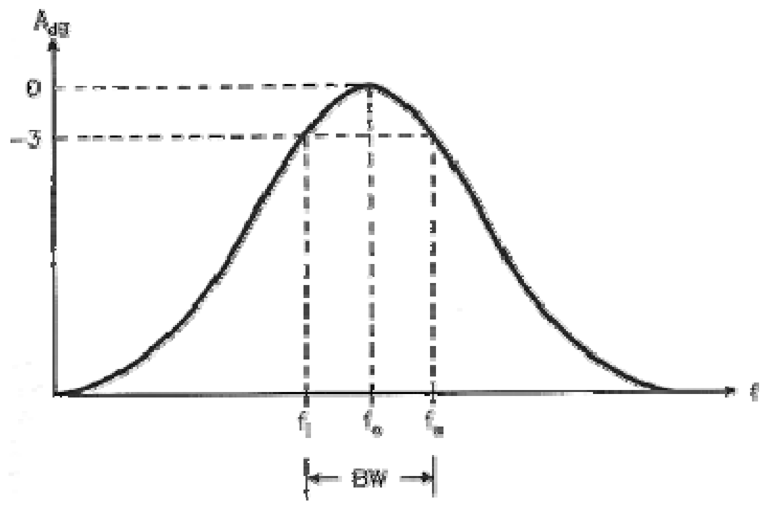


Figura 3.1: Frecuencia central f_0 de un filtro paso banda.

3.2.1.7 - Ancho de Banda 3 dB

Una manera de caracterizar un filtro es conocer el ancho de banda a 3 dB (***BW$_{3dB}$***[1]). Como definición, podemos decir que el ancho de banda de un filtro es la diferencia entre las frecuencias cuya atenuación se mantiene igual o inferior a 3 dB en comparación con la frecuencia central de pico $\boldsymbol{f_0}$, tal y como queda reflejado en la gráfica 3.1 expuesta en el apartado anterior.

3.2.1.8 - Rizado

El Rizado es una medida de lo plana que es la respuesta en amplitud en la banda de paso del filtro y se define como la diferencia entre las atenuaciones máxima y mínima de dicha banda de paso.

Se trata de un parámetro característico de un determinado tipo de filtros, llamados *Chebyshev.*

[1]*El decibelio (dB) es una unidad utilizada para expresar el nivel de potencia de un filtro y que índica la relación entre las dos potencias comparadas.*

$$P\ (dB) = 10 * \log_{10}\ (P_{salida} / P_{entrada}).$$

En la figura siguiente, podemos observar los diferentes rizados de filtros paso bajo *Chebyshev*, en función de su orden:

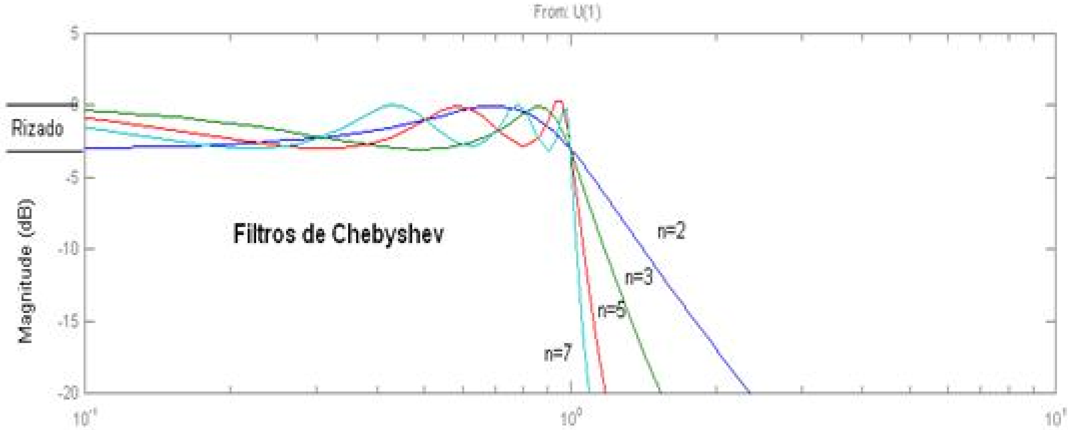


Figura 3.2: Representación gráfica de filtros Chebyshev de diferentes ordenes y niveles de rizado.

3.2.1.9 - Factor de Calidad

El factor de calidad ***Q*** es un parámetro adimensional de diseño equivalente al orden del filtro *N*, es decir, su realización se puede plantear a partir de un cierto factor de calidad ***Q***, en lugar de diseñar un filtro de un orden determinado.

En los filtros paso bajo y paso alto, *Q* representa la calidad del polo y se define como:

$$\boxed{Q = \frac{\sqrt{b_i}}{a_i}} \qquad (3.4)$$

Siendo a_i, b_i coeficientes que dependen del tipo de filtro seleccionado, del orden que tenga y de la etapa del filtro a la que nos estemos refiriendo y cuyos valores tabulados quedan recogidos en el anexo de este proyecto.

En los filtros paso banda, Q representa la relación entre la frecuencia central f_m y el ancho de banda de los puntos de caída 3 dB:

$$Q = \frac{f_m}{BW_{3dB}} = \frac{f_m}{f_2 - f_1} \quad (3.5)$$

Siendo $\boldsymbol{f_1}$, $\boldsymbol{f_2}$ las frecuencias de corte inferior y superior respectivamente.

Si se tienen valores de factor de calidad altos, estos se pueden representar gráficamente como la distancia entre la línea de 0 dB y el punto de pico de ganancia en la respuesta de ganancia de un filtro, tal y como se muestra en la figura 3.3:

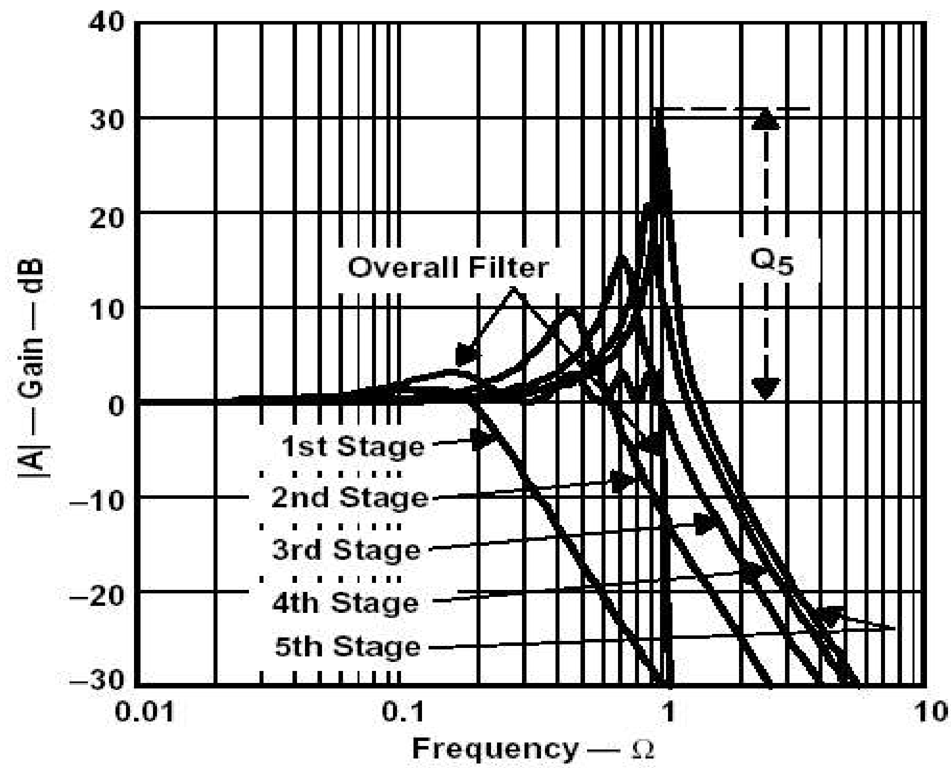


Figura 3.3: Representación gráfica del factor de calidad ***Q*** *en Filtros Paso Bajo Chebyshev de orden 10 y rizado 3 dB.*

Dicha aproximación gráfica se puede utilizar para Q > 3. Factores de calidad más pequeños hacen que los valores teóricos difieran de forma considerable respecto de los resultados gráficos.

En cualquier caso sólo nos preocuparán factores de calidad altos, puesto que cuanto mayor sean estos, más riesgo existirá de que el filtro, en la mayoría de los casos se haga inestable.

3.2.1.10 - Retardo de fase y Retardo de grupo

Cuando se obtiene la función de transferencia de un circuito de dos puertas, está demostrado que la señal alterna de salida tiene un desfase respecto de la señal de entrada.

Dicho desfase es el argumento de la función de transferencia, ya mencionado en el apartado anterior, es decir, **φ** = arg H(s), (expresado en radianes).

Si dividimos este desfase entre la frecuencia angular **ω**, (definida en radianes / segundo), tenemos la expresión del retardo de la salida respecto de la entrada:

$$\boxed{\zeta = \frac{-\varphi}{\omega}} \tag{3.6}$$

El signo negativo se emplea para que el retardo sea positivo, si es que realmente hay un retraso de la señal. Por el contrario si hubiera un adelanto, por convenio dicho retardo se consideraría negativo.

Otra manera de referirse al retardo es con la fase definida en grados y la frecuencia en Hz. En este caso, el retardo puede escribirse como:

$$\boxed{\zeta = \frac{-\varphi\ (grados)}{360 * f\ (Hz)}} \tag{3.7}$$

Este retardo referido a una sola sinusoide pura de frecuencia **ω**, se conoce como ***retardo de fase***.

Si la señal de entrada no es una sinusoide perfecta, el retardo no se puede calcular con la expresión 3.6 puesto que una señal no sinusoidal está formada por un conjunto de señales sinusoidales de frecuencias distintas.

Esto significa que cada una de estas señales de entrada tendrá un retardo diferente y al superponerse para formar la señal de salida, se produce una deformación de la forma de onda que complica el cálculo del retardo global producido.

Por tanto, para poder comparar el retardo existente entre dos señales es necesario que ambas presenten el mismo aspecto aunque sus tamaños no coincidan.

En el caso de una señal periódica $v(t)$, la misma señal retrasada un tiempo ζ es $v(t-\zeta)$. Si esta señal está formada por N componentes, tenemos lo siguiente:

$$\boxed{V_1(t) = \sum_{K=1}^{N} V_{\boldsymbol{K}} * \cos(K*w_0*t)} \qquad (3.8)$$

Si se la retrasa un tiempo ζ, obtenemos:

$$V_1(t-\zeta) = \sum_{K=1}^{N} V_{\boldsymbol{K}} * \cos[K*w_0*(t-\zeta)] \quad \rightarrow$$

$$\rightarrow \quad V_1(t-\zeta) = \sum_{K=1}^{N} V_{\boldsymbol{K}} * \cos(w_0*t+\varphi_K) \quad \text{siendo} \quad \varphi_K = -K*w_0*\zeta$$

Es decir, los desfases serán: $-w_0*\zeta$, $-2*w_0*\zeta$, $-3*w_0*\zeta\ldots$

Desarrollando estas ecuaciones tendremos que:

$$\boldsymbol{\zeta} = -\frac{\varphi_{K+1} - \varphi_K}{w_0} \tag{3.9}$$

Siendo el retardo perfecto para $\zeta = \zeta_1 = \zeta_2 \ldots = \zeta_N$

Por otra parte si la señal no es periódica, esta no se puede descomponer en serie de Fourier. Ahora bien, el espectro de una señal no periódica es continuo y por lo tanto, en su espectro continuo, la distancia entre una componente de frecuencia y otra tiende a cero.

Es decir, w_0 tiende a cero y se puede sustituir por un diferencial de frecuencia dw. Al ser también la diferencia entre los desfases en K+1 y K infinitesimales, la relación 3.9 para señales periódicas, se transforma para las no periódicas tal y como se índica:

$$\boldsymbol{\zeta}\,(\omega) = \frac{-\,d\varphi}{d\omega} \tag{3.10}$$

Esta función es conocida como ***función retardo*** y es la que representa el ***retardo de grupo***.

Si se pretende que una señal sufra un retardo ζ sin que se altere su forma, es necesario que la función retardo permanezca constante para todas las frecuencias del espectro continuo de la señal.

Esto en teoría no es posible puesto que no hay señales de banda totalmente limitada pero sí puede cumplirse con gran aproximación, por lo que la función retardo es muy útil en el diseño de ciertos tipos de filtro.

Un filtro ideal es aquel que presenta un retardo de grupo lo más constante posible en toda la banda de paso.

En la práctica no se ha logrado obtener un filtro con un retardo de grupo constante y que a la vez garantice una buena respuesta de amplitud.

Por tanto, resultará necesario elegir si queremos señales transitorias sin deformación adaptándose a su respuesta de amplitud o bien, considerar únicamente la atenuación en la construcción del filtro (es lo que se suele hacer), adaptándose a que el retardo de grupo no cumpla el objetivo de ser constante y como consecuencia, se tenga una señal deformada.

3.2.2 - Clasificación

Es posible establecer diferentes clasificaciones de filtros eléctricos en función del aspecto concreto al que nos refiramos.

En este sentido, podemos agruparlos según la gama de frecuencias atenuadas, según la tecnología empleada para su construcción o según la función matemática o de aproximación utilizada para proyectar el filtro.

3.2.2.1 - Según el conjunto de frecuencias atenuadas

Filtro Paso Bajo: Idealmente solo permite el paso de las frecuencias inferiores a una determinada, llamada *frecuencia de corte* (**f_c**). Por contra, las frecuencias superiores resultan atenuadas de la manera que se ilustra a continuación:

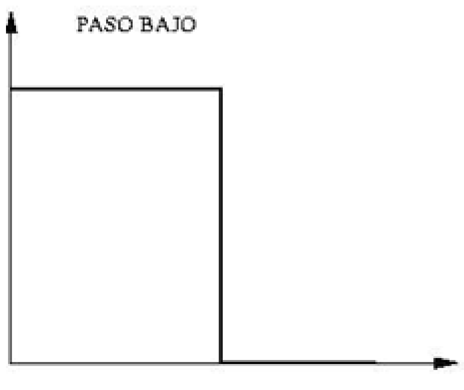


Figura 3.4: Respuesta en frecuencia de un Filtro Paso Bajo Ideal

Filtro Paso Alto: De manera inversa al anterior este filtro permite el paso de todas aquellas frecuencias superiores a una dada $\boldsymbol{f_c}$ (frecuencia de corte), atenuando el resto también de forma ideal, tal y como se índica en el gráfico expuesto:

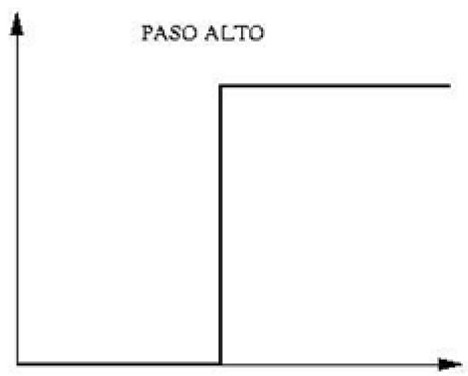


Figura 3.5: Respuesta en frecuencia de un Filtro Paso Alto Ideal

Filtro Paso Banda: Esta configuración permite el paso a frecuencias situadas dentro de una banda delimitada por una frecuencia de corte inferior $\boldsymbol{f_1}$ y otra superior $\boldsymbol{f_2}$, atenuando aquellas otras que se encuentran fuera de este rango.

Su representación gráfica queda reflejada en la siguiente figura:

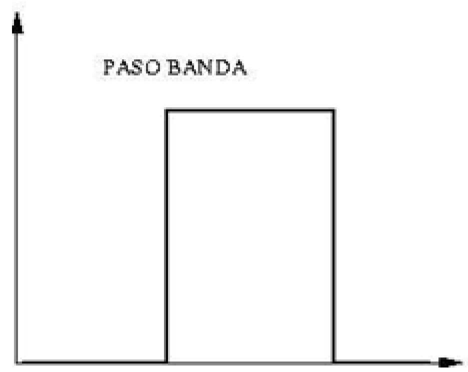


Figura 3.6: Respuesta en frecuencia de un Filtro Paso Banda Ideal

Filtro Rechazo de Banda*:* Al revés que el anterior, deja pasar las frecuencias ubicadas fuera de la banda definida por dos frecuencias de corte (superior e inferior), atenuando las que se encuentran dentro de ella (banda rechazada), del modo que se muestra en la ilustración:

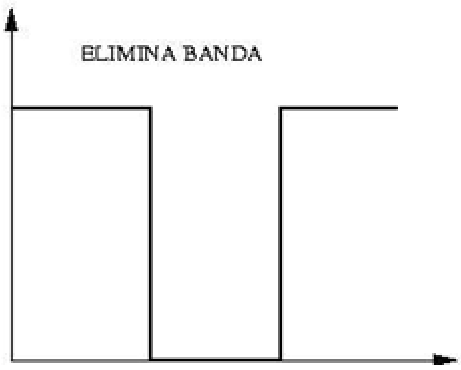


Figura 3.7: Respuesta en frecuencia de un Filtro Banda Rechazada Ideal

Todos estos supuestos son ideales. En los filtros reales, hay que señalar que la ganancia en la banda de paso no será estrictamente plana sino que se apreciará un determinado rizado.

Así mismo, la atenuación en la banda rechazada no será infinita para todas las frecuencias sino que tendrá un valor alto, asumiendo además que la transición de una banda a otra no será abrupta, contará más bien con una cierta inclinación.

3.2.2.2 - Según la tecnología empleada

Otra posible clasificación que se puede hacer de los filtros eléctricos tiene en cuenta la tecnología utilizada en su construcción. Considerando este aspecto, podemos dividirlos en filtros pasivos, activos y digitales.

Filtros Pasivos*:* Reciben este nombre porque para su implementación, requieren dispositivos pasivos tales como resistencias, condensadores e inductancias, formando circuitos resonantes.

El margen de frecuencias en el que trabajan varía desde los pocos MHz[1] hasta 1.5 GHz[2], valor a partir del cual se sitúa la banda de Radiofrecuencia (RF).

La principal desventaja de este tipo de filtros es que, a frecuencias bajas (hasta 100 KHz[3]), las autoinductancias que se necesitan son muy grandes y voluminosas, provocando que las características del circuito se alejen de sus parámetros ideales.

El tamaño de las bobinas condiciona en gran medida el nivel de integración del filtro, el cual sería mayor si no existiesen. Además las bobinas son caras y tienen un comportamiento inexacto.

Otro inconveniente de estos diseños son las pérdidas parásitas al aumentar la frecuencia.

Por último, un aspecto importante que no se debe olvidar es la adaptación de impedancias, ya que el circuito tiene una impedancia de entrada y una de salida, así como la resistencia de generador y de carga que influyen en su comportamiento.

Filtros activos*:* Son los filtros sobre los que realizamos nuestro proyecto. A diferencia de los anteriores, son dispositivos que incorporan un elemento de ganancia, generalmente un amplificador operacional a los que se añaden resistencias y condensadores.

La razón es que si se emplearan circuitos RLC (resistencia, inductancia y condensador), los valores requeridos estarían fuera del rango comercial en muchos casos.

Su rango de funcionamiento es en bajas frecuencias (hasta pocos MHz). En relación a los filtros pasivos, cuentan con una serie de ventajas y algún inconveniente sobre los que vamos a hacer hincapié.

[1]MHz es la abreviatura de megahercios, [2]GHz significa gigahercios y [3]KHz son los kilohercios.

Ventajas

- Como ya se ha comentado no utilizan autoinducciones, con lo que disminuye su coste, mejora su comportamiento y reduce su tamaño (sobre todo si empleamos condensadores de tantalio), gracias a lo cual se aplica tecnología microelectrónica, lo que permite construir filtros activos de proporciones microscópicas.

- En general, tienen una impedancia de entrada muy alta (idealmente ∞), por lo que la impedancia de entrada total del circuito solo dependerá de los componentes adicionales que añadamos.

Sin embargo, su impedancia de salida es muy baja (idealmente 0) y como consecuencia de ello, la resistencia de carga que coloquemos no va a tener influencia en el filtro.

El resultado es que estos circuitos presentan una muy buena capacidad de aislamiento, pudiéndose conectar en cascada varios filtros sin que afecten los unos a los otros, es decir, sus funciones de transferencia se multiplican al no depender de las impedancias de generador y de carga.

- Posibilidad de amplificación, tanto de tensión como de corriente, particularidad esta importante para señales de bajo nivel.

- Poseen un factor de calidad ***Q*** relativamente grande, alcanzando valores hasta de 500.

- Facilidad para realizar ajustes en el filtro, por ejemplo modificando el valor de una resistencia, que puede ser un potenciómetro.

Inconvenientes

- Necesidad de contar con 1 o 2 fuentes de alimentación que pueden introducir ruido.

- Su respuesta en frecuencia está limitada por la capacidad de los amplificadores operacionales utilizados. El ancho de banda del filtro estará limitado por el ancho de banda de ganancia unidad f_t del amplificador.

- El margen dinámico de salida está acotado para valores por encima de

± 10 V de amplitud de la señal de entrada. En esta situación, el amplificador operacional puede saturarse, además la corriente de salida apenas tendrá un valor de algunos miliamperios.

Por contra, para valores bajos de amplitud de la señal de entrada, el ruido intrínseco del amplificador operacional puede enmascarar la señal.

- Son muy sensibles a los cambios de temperatura y al envejecimiento de los componentes, que provocan un desplazamiento de los polos de la función de transferencia que pueden hacer inestable al circuito.

- Es imposible su aplicación en sistemas de medida y alta potencia.

Filtros digitales*:* En este tipo de filtros, se muestrea la señal analógica de entrada, se convierte a señal digital a través de convertidores A/D (Analógico / Digital).

Dicha señal se introduce a la entrada del filtro, situando ahora en su salida y de manera inversa a la anterior, un convertidor D/A (Digital / Analógico) que nos permita obtener la señal analógica resultante.

Son útiles para procesar muchos canales de transmisión de forma simultánea. Así mismo, pueden definirse como programas software.

Filtros de microondas: Son empleados para trabajar a frecuencias muy elevadas (desde varios MHz hasta el rango de GHz). Se sintetizan en configuraciones de guía de onda, siendo las bobinas y los condensadores sintetizados con trozos de líneas de transmisión.

3.2.2.3 - Según la función matemática utilizada

A la hora de realizar un filtro, no solo debemos fijarnos en la frecuencia de corte o en el ancho de banda que tendrá, sino también en que la ganancia caiga determinados decibelios a una frecuencia específica, manteniendo constante o bien el ancho de banda o bien la frecuencia de corte.

Dicho de otro modo, seleccionar un tipo de respuesta en frecuencia u otro variará según el tipo de señal que lo atraviese, según el tipo de aplicación para el cual se va a necesitar o según el comportamiento en fase que se pretenda conseguir.

En este sentido, los tipos de aproximación de respuesta en frecuencia para filtros paso bajo, es decir, las diferentes formas que existen de representar la amplitud y la fase en función de la frecuencia pueden expresarse mediante funciones matemáticas, llamadas *funciones de aproximación*, las cuales nos permitirán obtener la función de transferencia del filtro en cuestión y de esta manera, llegar a conocer la estructura y el valor de sus elementos.

Las *funciones de aproximación* se dividen en dos categorías:

- ***Funciones Polinómicas***:

En este grupo se encuentran las funciones de *Butterworth*, *Lagrange*, *Chebyshev* y *Bessel*. Los filtros polinomiales son fáciles de diseñar pero de características relativamente pobres. Estos filtros no presentan ningún cero de transmisión, no hay ninguna frecuencia para la cual, la atenuación sea infinita.

- ***Filtros Butterworth***

Son filtros que presentan una respuesta en amplitud máximamente plana en la banda de paso, es decir, la gráfica es plana en esa zona y la atenuación es de 0 dB. Por tanto, la salida va a ser igual a la entrada para el rango de frecuencias de paso.

Sin embargo, tienen el inconveniente de que la atenuación se va incrementando de manera muy gradual fuera de dicha banda, con lo cual se requiere de un orden del filtro alto para obtener una atenuación grande.

Si la representación se realiza en escalas logarítmicas utilizando los decibelios (dB) y se considera que la frecuencia **w** es mucho más grande que la frecuencia de corte, la pendiente en esa situación es de 20*N dB / década o su equivalente 6*N dB / octava, siendo **N** el orden del filtro.

Por otra parte, la atenuación de un filtro *Butterworth* es siempre de 3 dB ($\frac{1}{\sqrt{2}}$ = 0.707 en escala lineal) a frecuencia w = 1 radián / segundo.

Así mismo, este tipo de filtros presentan una respuesta poco sensible a las variaciones de sus componentes, lo cual es una ventaja si no es posible ajustar de forma precisa dichos elementos como en el caso de trabajar a frecuencias muy altas.

En cuanto a su campo de aplicación, los *Butterworth* suelen emplearse en sistemas de conversión de datos donde se requieren niveles de señal exactos en toda la banda de paso.

En la figura 3.8 podemos observar las curvas de ganancia de filtros *Butterworth* para diferentes *órdenes* y como la pendiente de la curva en la banda de transición se hace más abrupta a medida que se incrementa el orden del filtro.

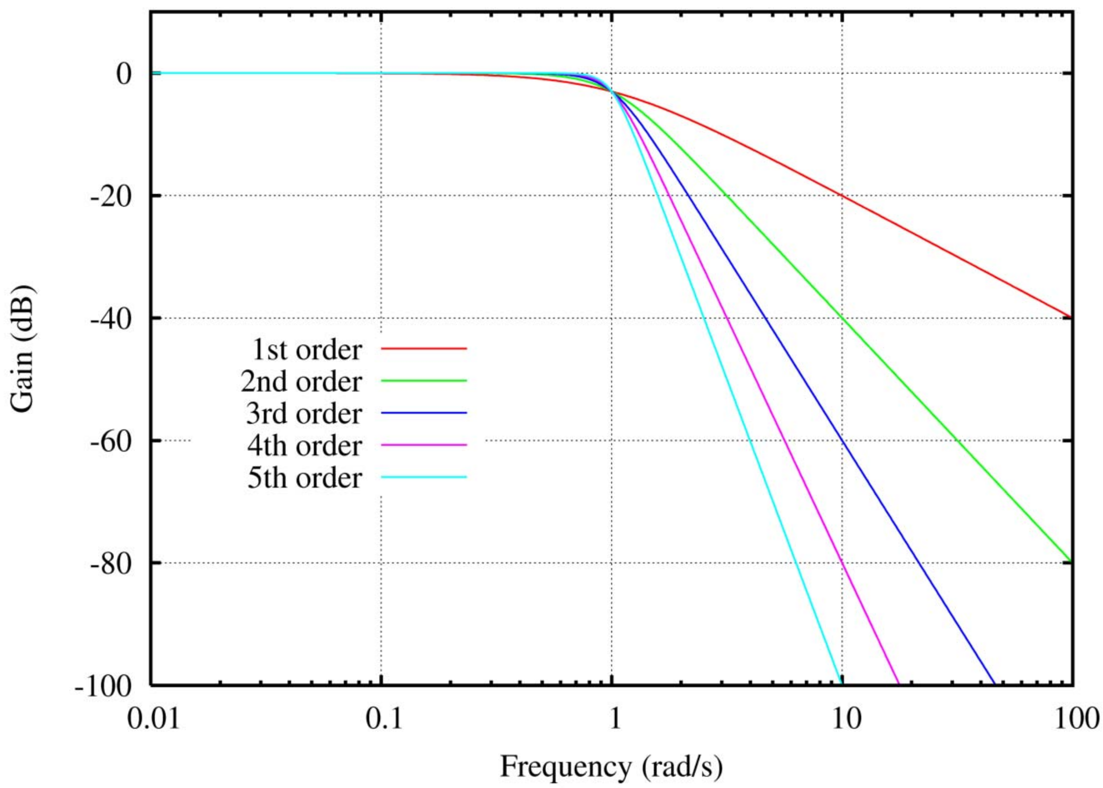


Figura 3.8: Curvas de ganancia de filtros paso bajo Butterworth de diferentes órdenes en escala logarítmica.

- ***Filtros de Lagrange o de Papoulis***

Este tipo de dispositivos posee una curva de atenuación que se incrementa de manera uniforme, al igual de lo que sucedía anteriormente en los filtros *Butterworth*.

Sin embargo, a diferencia de estos, presentan una respuesta en frecuencia óptima a la frecuencia de corte, es decir, su pendiente se hace lo más abrupta posible en este punto.

Las funciones características de *Lagrange* quedan definidas a través de una derivada $\frac{dL}{dw}$, la cual es siempre positiva y adopta en $w = 1$ un valor mayor que en las otras funciones características.

Dichas funciones fueron planteadas por A. Papoulis en 1958 en diversas publicaciones y complementadas por otros investigadores posteriormente.

- ***Filtros de Chebyshev***

En este caso, hablamos de filtros que consiguen tener un rechazo más abrupto en la banda atenuada (por tanto mayor selectividad) para igual *orden* del filtro que el alcanzado por *Butterworth*.

En concreto a frecuencias altas, su atenuación crece a razón de 20*N dB / década + 6*(N-1) dB / década, con lo que su comportamiento es mejor que el *Butterworth* en este rango elevado de frecuencias.

No obstante, el precio a pagar es la aparición de un rizado en la banda de paso y una respuesta en fase peor que el que se tiene para los filtros *Bessel*. En la gráfica 3.9 expuesta más abajo, quedan reflejadas las respuestas de amplitud de un filtro *Chebyshev* de *orden* 4 con diferentes *rizados*.

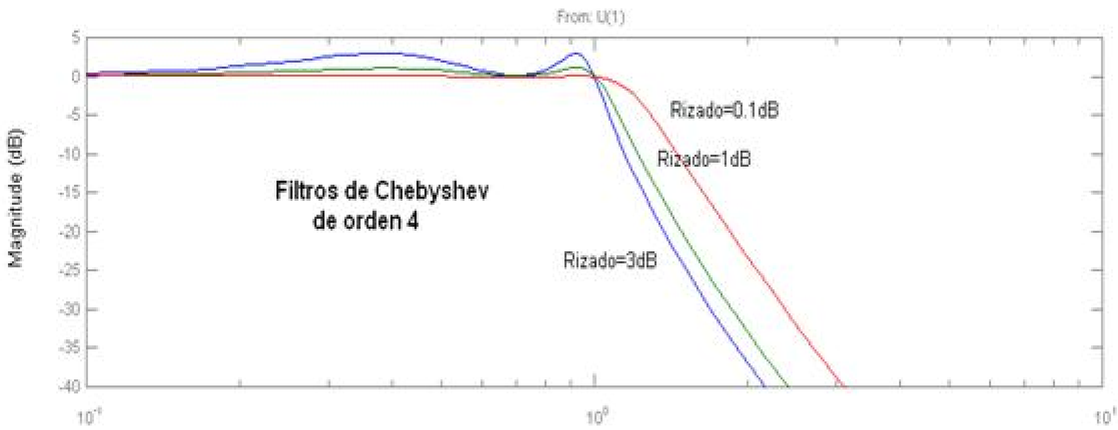


Figura 3.9: Curvas de ganancia de filtros paso bajo Chebyshev de orden 4 con diferentes rizados.

Si comparamos cada una de las gráficas, nos damos cuenta de que cuanto mayor es el *rizado*, más abrupta se hace la pendiente de la curva en la banda de transición, es decir, el inconveniente que supone tener un mayor *rizado* en la banda de paso conlleva un mejor comportamiento del filtro en la banda rechazada, al ser mayor su atenuación.

El número de mínimos y de máximos que contiene el *rizado* depende del *orden* del filtro, a mayor *orden* más mínimos. Además, en función de si el *orden* es par o impar, habrá primero un máximo o un mínimo y generarán *rizado* por encima de la línea de 0 dB (*orden* par) o por debajo (*orden* impar).

Circunstancias todas estas que se ponen de manifiesto en la siguiente ilustración:

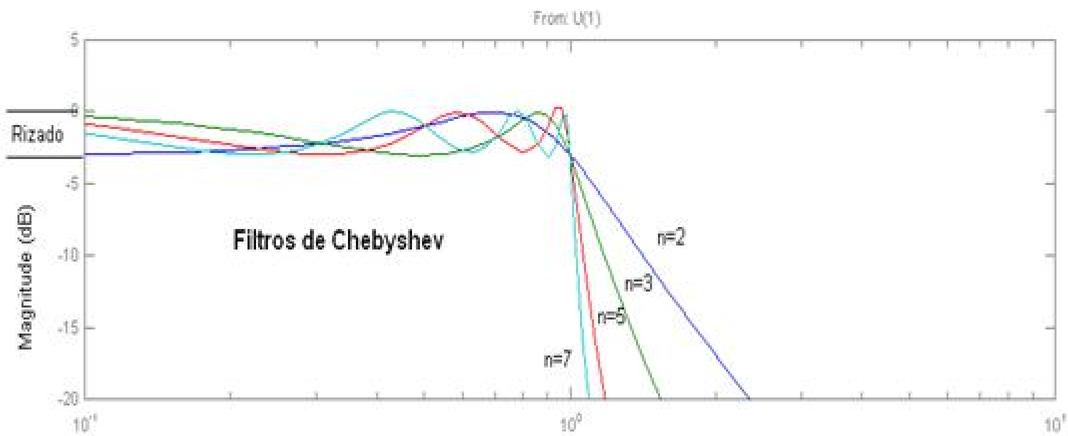


Figura 3.10: Curvas de respuesta de amplitud de filtros paso bajo Chebyshev de diferentes órdenes.

Si nos fijamos en los filtros, las respuestas de amplitud de los *Chebyshev* de *orden* impar (en este caso 3, 5 y 7) comienzan con un máximo, mientras que la respuesta de *orden* par (en este caso orden 2) comienza con mínimo.

Para unas mismas especificaciones dadas, el *orden* del filtro *Chebyshev* es menor que el de *Butterworth*, lo cual implica un menor número de componentes en su implementación y como consecuencia un menor coste.

Por contra, los filtros *Chebyshev* no presentan regularidad en el tiempo de propagación de grupo en la banda de paso ni tampoco poseen un comportamiento frente a los transitorios tan bueno como el de los *Butterworth* o *Lagrange*.

Teniendo en cuenta todo lo anterior, se recurrirá a este tipo de filtros sólo cuando el *rizado* en la banda de paso y la regularidad en el tiempo de propagación de grupo no representen un inconveniente, es decir, se restringirá a todas aquellas aplicaciones en las que el contenido de frecuencia es más importante que la magnitud a manejar.

- ***Filtros de Bessel***

También llamados *filtros de retardo constante*, reciben su nombre por la utilización que se hace de los polinomios de Bessel. Son dispositivos en los que se busca que tengan una respuesta lineal con respecto a la *fase*, lo cual da como resultado un *retardo* constante en todo el *ancho de banda* deseado, situación esta que queda reflejada en la gráfica comparativa :

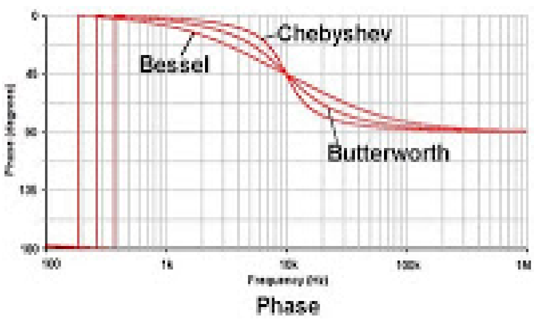


Figura 3.11: Respuesta de fase de los filtros Butterworth, Chebyshev y Bessel.

El principal criterio de optimización es la regularidad del *tiempo de propagación de grupo* en la banda de paso. Por tanto, el mayor interés se centra en el retardo (*diagrama de fase*) y no en la *atenuación*.

$$\text{Retraso} = \frac{d}{d\omega}\left(-\angle H(s)\right) \quad \text{(seg)} \qquad (3.11)$$

A diferencia del resto de filtros estudiados, en los que los retrasos contienen muchos picos de variación, en los *filtros de Bessel* dicho parámetro es lo más plano posible en el rango de frecuencias de interés.

Sin embargo, su respuesta en frecuencia es mucho menos selectiva en la banda rechazada, es decir, su *atenuación* es mucho más pequeña que en los filtros de *Chebyshev*, mientras que en la banda de paso no es tan plana como en el caso de los filtros de *Butterworth* por ejemplo, tal y como se puede observar en la figura 3.12.

Los polinomios de Bessel que aparecen en el denominador de la función de transmisión de estos filtros, se calcularon tomando como unidad el *tiempo de propagación de grupo*.

Esto significa que se obtendrán curvas de respuesta en amplitud en las que la *atenuación* en la frecuencia unidad no será de 3 dB sino que tendrá un valor impredecible.

En función de todas estas características, dichos dispositivos proporcionan un comportamiento de transmisión de onda cuadrada óptimo y su utilización quedará reducida a todas aquellas aplicaciones que no admitan ninguna deformación en régimen transitorio.

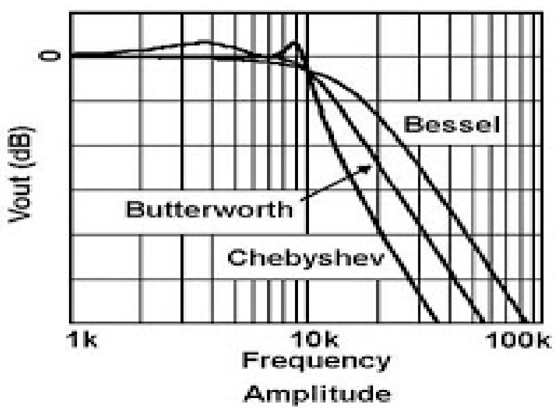


Figura 3.12: Respuesta frecuencial de filtros Butterworth, Chebyshev y Bessel.

- ***Filtros no polinomiales:***

Corresponden a todas aquellos cuyas funciones características son fracciones racionales. Una de ellas son las funciones de Zolotareff que dan origen a los filtros *Cauer*. Al contrario que los anteriores, estos filtros presentan mejores características pero son difíciles de calcular y cuentan siempre con una serie de ceros.

A diferencia de las funciones polinómicas, este tipo de diseños presentan ceros de transmisión a frecuencias finitas.

Los filtros polinomiales tienen una función de transferencia del tipo:

$$H(s) = \left| \frac{V_i}{V_o} \right| = a_n * S^n + a_{n-1} * S^{n-1} + \ldots + a_1 * S + a_0 \qquad (3.12)$$

Y su función en frecuencia resulta ser también un polinomio en ω^2:

$$\left| \frac{V_i}{V_o} \right|^2 = \left| H(s) * H(-s) \right|_{s = j\omega} = h(\omega^2) \qquad (3.13)$$

De aquí que para cualquier valor finito de la frecuencia, tendremos a su vez un valor finito de atenuación por lo que un cero de transmisión sólo se consigue si la frecuencia ω tiende a infinito.

La introducción de ceros de transmisión supone eliminar frecuencias indeseables y hacer por ejemplo, que la pendiente en la zona de transición del filtro sea mucho más abrupta, situando dicho cero después de la frecuencia de corte y obtenerlo de esta manera, sin necesidad de incrementar el *orden* del filtro que es lo que nos sucedía en todos los supuestos estudiados anteriormente.

- ***Filtros de Cauer***

Este tipo de dispositivos, también llamados *filtros elípticos*, se engloban dentro de la familia de los *filtros no polinomiales*.

En los diseños previos, la atenuación en la banda rechazada aumenta indefinidamente a medida que se incrementa la frecuencia pero esto no siempre es necesario, sino que en muchas ocasiones será suficiente con mantener una determinada *atenuación* constante.

Y esta característica es la que se consigue con los filtros de *Cauer*, destacando a su vez, por ser de un *orden* igual o menor que los filtros *Butterworth* o *Chebyshev* para unas mismas especificaciones dadas.

Los filtros *Cauer* tienen un *rizado* tanto en la banda de paso (similar a la de los *Chebyshev*) como en la banda rechazada, aunque ambos están acotados, tal y como queda reflejado en la figura 3.13 representada más abajo.

Sin embargo, pueden ajustarse hasta conseguir *rizado* nulo y presentar una característica plana en cualquiera de las bandas si unimos los mínimos del *rizado* para cada una de ellas.

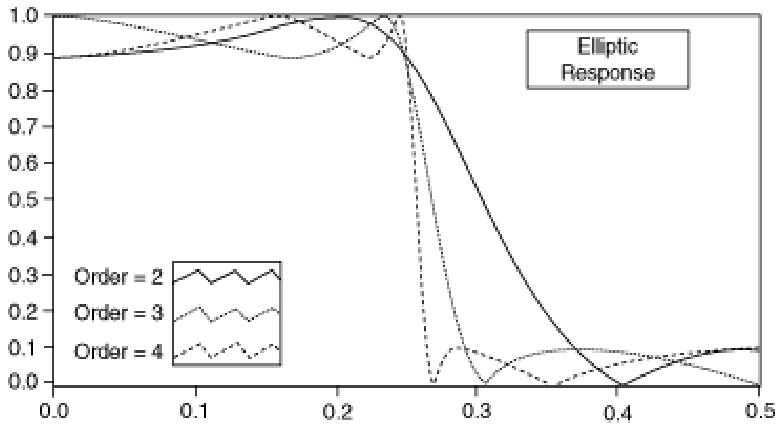


Figura 3.13: Respuesta en frecuencia de filtros Cauer de diferentes órdenes.

En cuanto a su campo de aplicación, los filtros *Cauer* son empleados en sistemas que requieran altas prestaciones.

Por contra, uno de sus inconvenientes radica en la enorme irregularidad del *tiempo de propagación de grupo* alrededor de la *frecuencia de corte*, así como la necesidad de su implementación con circuitos de doble T que suponen incremento considerable en el número de componentes a la hora de realizar el montaje completo.

CAPÍTULO IV

FILTROS ACTIVOS DE SALLEN - KEY Y DE RAUCH

En este capítulo analizaremos de forma más definida los *filtros activos* utilizados en el diseño de nuestro programa, centrándonos en este sentido, en los circuitos de *Sallen - Key* y de *Rauch*.

Como ya sabemos del capítulo anterior, los *filtros activos* emplean un elemento de *ganancia* que suele ser un amplificador operacional, resistencias y condensadores.

Así mismo, presentan una serie de ventajas como su facilidad de ajuste, un coste más reducido en comparación con los *filtros pasivos* y una *impedancia de entrada* elevada junto a una baja *impedancia de salida* que permite que se puedan conectar en cascada varios filtros fácilmente sin que afecten los unos a los otros, es decir, sus funciones de transferencia se multiplican.

Y esta característica de poder conectar en cascada una serie de filtros, será la que manejaremos para la implementación de nuestro diseño, aspecto que será desarrollado en el último apartado del capítulo.

En primer lugar, abordaremos las celdas básicas *paso bajo* y *paso alto* de *Sallen-Key* y de *Rauch*, analizando cada uno de los circuitos y calculando sus funciones de transferencia.

4.1. – Filtros de Sallen-Key

Con el objetivo de reducir el número de amplificadores operacionales en los circuitos de los filtros, *Sallen* y *Key* diseñaron estructuras capaces de obtener una función cuadrática con *polos imaginarios*, utilizando un solo operacional.

Por ello, las células *Sallen* y *Key* son un tipo especifico de filtro activo particularmente valioso por su simplicidad. A continuación analizaremos los circuitos básicos empleados en el diseño de nuestro filtro.

4.1.1. - Etapa paso bajo de primer orden

El circuito asociado a esta etapa que seleccionamos es el mostrado en la imagen 4.1:

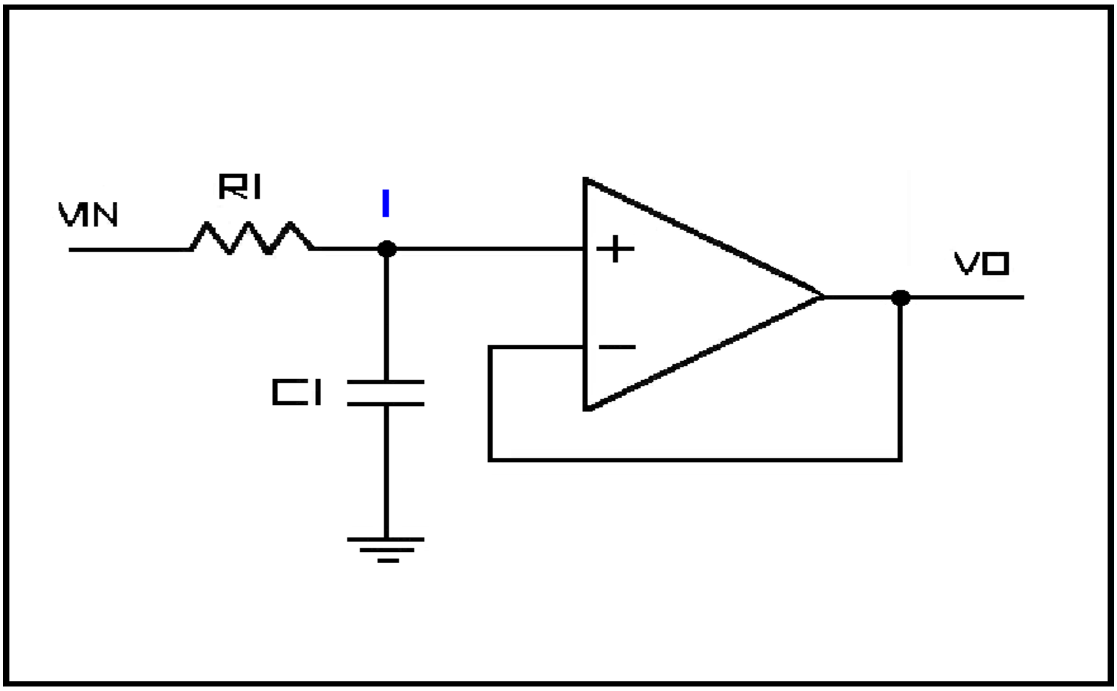


Figura 4.1: Filtro paso bajo Sallen – Key de primer orden en configuración de seguidor de tensión, con ganancia unitaria.

Observando el montaje de la *figura 4.1*, nos damos cuenta que:

(1) $$V(1) = \frac{\frac{1}{C1*s} * V_{IN}}{\frac{1}{C1*s} + R1} = \frac{V_{IN}}{1 + R1*C1*s}$$

(2) $$V_O = V(1)$$

Sustituyendo (1) en (2):

$$V_O = \frac{1}{1 + R1*C1*s} * V_{IN} \longrightarrow \boxed{H(s) = \frac{V_O}{V_{IN}} = \frac{1}{1 + R1*C1*s}} \qquad (4.1)$$

Si pretendemos sintetizar una función de transferencia normalizada del tipo $H(s_n) = \dfrac{1}{1 + a_1 * s_n}$, donde la variable compleja "*s*" es normalizada y sustituida por la expresión correspondiente $s_n = \dfrac{s}{w_{corte}} = \dfrac{j*w}{w_{corte}}$, el coeficiente asociado a_1 deberá tener la siguiente relación $a_1 = w_{corte} * R1 * C1$ para conseguir que la función de transferencia $H(s_n)$ sea adimensional, siendo $w_{corte} = 2 * \Pi * f_{corte}$. Así mismo, la *frecuencia de corte* del circuito vendrá determinada por $f_{corte} = \dfrac{1}{2*\Pi*R1*C1}$.

Por tanto, para el diseño del circuito tenemos como dato la *frecuencia de corte* (la *ganancia* como vemos es unitaria) y elegiremos el valor de la capacidad C1 (en nuestro programa ambos serán introducidos por parte del usuario). Entonces para resolver el filtro, sólo nos queda calcular la resistencia R1, cuya ecuación es la que se muestra:

$$\boxed{R1 = \frac{a_1}{2*\Pi*f_{corte}*C1}} \qquad (4.2)$$

Identificando términos mencionar que a_i y b_i son coeficientes específicos que dependerán del *orden* que tenga el filtro, de la etapa a la que pertenezca la célula básica empleada y de la aproximación matemática establecida por el usuario (*Butterworth, Chebyshev de rizado 0.5, 1, 2 o 3 dB o Bessel*), estando dichos coeficientes tabulados y recogidos en una serie de tablas impresas en el anexo de este proyecto.

4.1.2. – Etapa paso bajo de segundo orden

La célula *Sallen-Key* paso bajo de segundo *orden* está formada por 2 resistencias, 2 condensadores y un amplificador operacional que para todos los supuestos consideraremos ideal, tal y como se observa en el montaje de la *figura 4.2.*

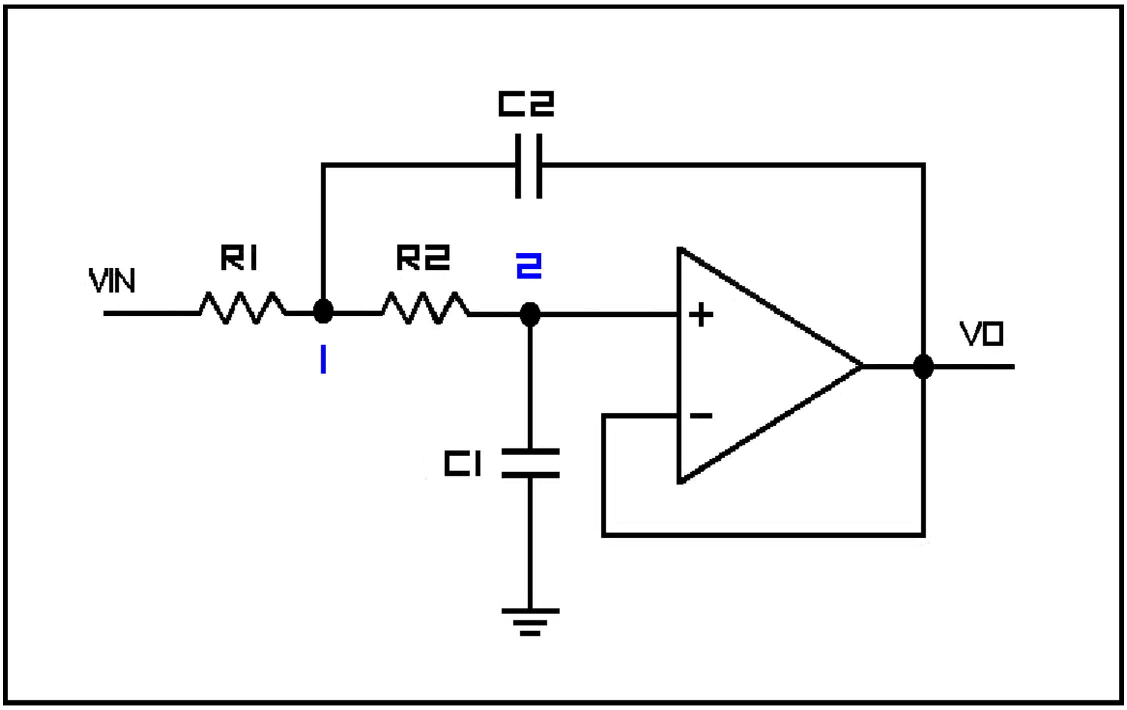


Figura 4.2: Filtro paso bajo Sallen – Key de segundo orden en configuración de seguidor de tensión, con ganancia unitaria.

Inspeccionando el circuito, vemos que:

(1) $$\frac{V(1)-V_{IN}}{R1} + \frac{V(1)-V_O}{\frac{1}{C2*s}} + \frac{V(1)-V(2)}{R2} = 0$$

(2) $V_O = V(2)$

(3) $$\frac{V(1)-V(2)}{R2} + \frac{V(2)}{\frac{1}{C1*s}} = 0$$

Sustituyendo (2) en (3):

$$\frac{V_O-V(1)}{R2} + V_O*C1*s = 0 \quad \longrightarrow \quad V_O - V(1) + V_O*C1*R2*s = 0$$

$$\longrightarrow \quad V(1) = V_O * (1 + C1*R2*s)$$

Ahora despejamos en (1):

$$\frac{V_O*(1 + C1*R2*s) - V_{IN}}{R1} + V_O * (1 + C1*R2*s - 1) * C2*s +$$

$$+ \frac{V_O*(1 + C1*R2*s - 1)}{R2} = 0 \quad \longrightarrow$$

$$\longrightarrow \quad V_O * (1 + C1*R2*s - 1) + V_O * (R1*R2*C1*C2*s^2) + V_O * (C1*R1*s) = 0$$

$$\boxed{H(s) = \frac{V_O}{V_{IN}} = \frac{1}{R1*R2*C1*C2*s^2 + C1*(R1 + R2)*s + 1}} \quad (4.3)$$

De nuevo, si queremos obtener la función de transferencia normalizada H (s_n) donde $s_n = \frac{s}{w_{corte}} = \frac{j*w}{w_{corte}}$, en este caso los coeficientes específicos tomarán los valores $a_i = w_{corte}$* C1* (R1 + R2) y $b_i = w_{corte}^2$*R1 * R2 * C1* C2.

Conocidos dichos coeficientes, obtendremos el valor de las resistencias R1 y R2, cuyas ecuaciones son las siguientes:

$$R1 = \frac{a_i * C2 + \sqrt{(a_i * C2)^2 - 4 * b_i * C1 * C2}}{4 * \Pi * f_{corte} * C1 * C2} \qquad (4.4)$$

$$R2 = \frac{a_i * C2 - \sqrt{(a_i * C2)^2 - 4 * b_i * C1 * C2}}{4 * \Pi * f_{corte} * C1 * C2} \qquad (4.5)$$

Para que las resistencias R1 y R2 tengan valores reales (es decir, que no sean negativas), se debe de cumplir que:

$$C2 > C1 * \frac{4 * b_i}{a_i^2} \qquad (4.6)$$

Esta será una condición impuesta al usuario en el momento de realizar el diseño del filtro. En este sentido, introducirá primero el valor C1 y el programa enviará un mensaje, especificando cuanto debe valer como mínimo la capacidad C2 respecto del dato introducido C1.

4.1.3. – Etapa paso alto de primer orden

El circuito *Sallen-Key* correspondiente a esta configuración es el que se describe a continuación:

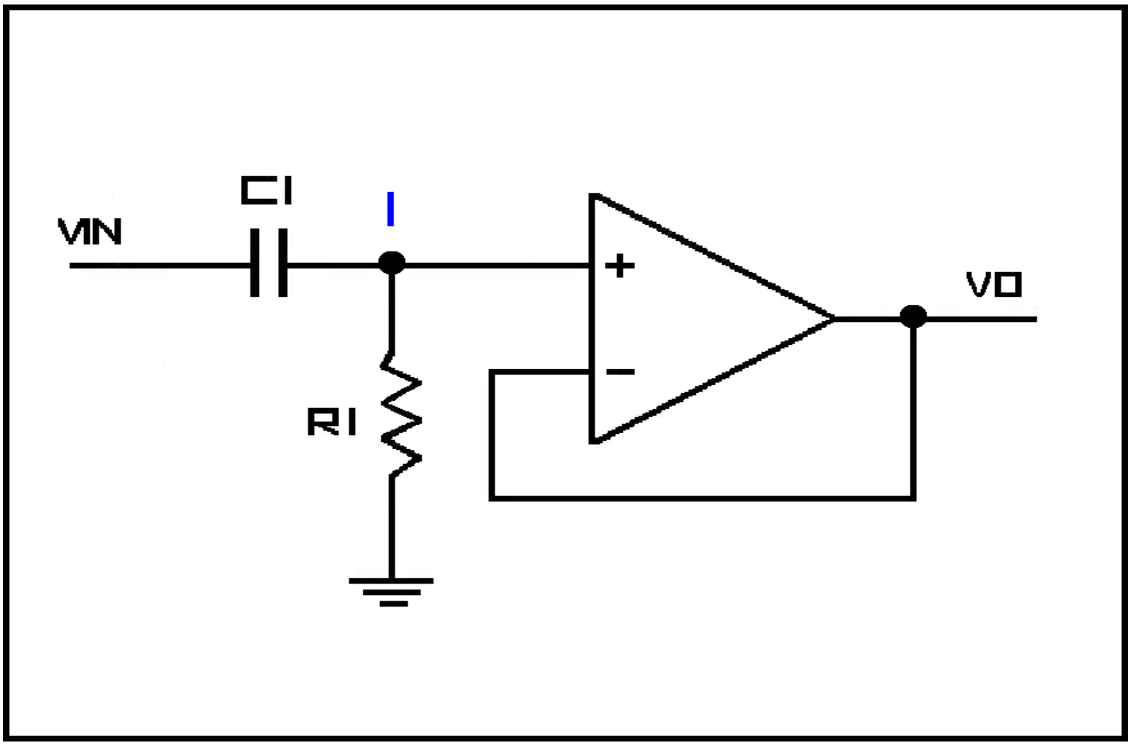


Figura 4.3: Filtro paso alto Sallen – Key de primer orden en configuración de seguidor de tensión, con ganancia unitaria.

Para este supuesto del circuito de la figura 4.3, tenemos lo siguiente:

(1) $$\frac{V(1)-V_{IN}}{\frac{1}{C1*s}} + \frac{V(1)}{R1} = 0$$

(2) $V_O = V(1)$

Sustituyendo (2) en (1):

$$(V_O - V_{IN}) * C1 * s + \frac{V_O}{R1} = 0 \quad \longrightarrow \quad (V_O - V_{IN}) * C1 * R1 * s + V_O = 0$$

$$\boxed{H(s) = \frac{V_O}{V_{IN}} = \frac{C1*R1*s}{C1*R1*s + 1}} \tag{4.7}$$

Si queremos tener la función de transferencia normalizada $H(s_n)$ cambiando la variable "s" por $s_n = s / w_{corte} = j*w / w_{corte}$ en la expresión 4.7 obtendremos:

$$H(s_n) = \frac{V_O}{V_{IN}} = \frac{\mathrm{C1} * \mathrm{R1} * \mathrm{s_n} * \mathrm{w_{corte}}}{\mathrm{C1} * \mathrm{R1} * \mathrm{s_n} * \mathrm{w_{corte}} + 1}$$

Dividiendo arriba y abajo esta ecuación por $C1*R1*s_n*w_{corte}$ tenemos que

$$H(s_n) = \frac{1}{1 + \frac{1}{w_{corte}*R1*C1} * \frac{1}{s_n}} \quad \text{siendo } a_1 = \frac{1}{w_{corte}*R1*C1}$$

Al igual que para el caso del filtro *paso bajo*, tanto la *frecuencia de corte* como el valor del condensador C1 serán datos introducidos por el usuario. Una vez identificados estos valores y para completar todos los componentes, calcularemos la resistencia R1 del filtro:

$$R1 = \frac{1}{2*\Pi*f_{corte}*a_1*C1} \tag{4.8}$$

4.1.4 – Etapa paso alto de segundo orden

El esquema circuital asociado a esta etapa es el expuesto:

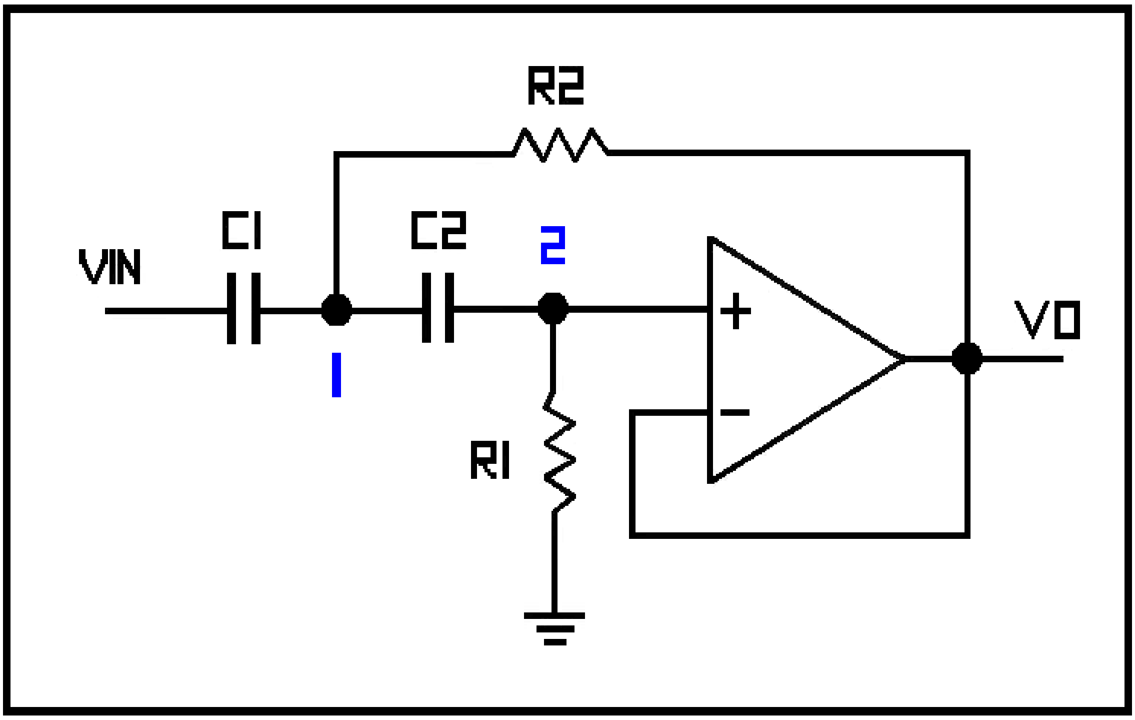


Figura 4.4: Filtro paso alto Sallen – Key de segundo orden en configuración de seguidor de tensión, con ganancia unitaria

Si analizamos la celda *paso alto Sallen-Key* de la figura 4.4, deducimos lo siguiente:

(1) $$\frac{V(1)-V_{IN}}{\frac{1}{C1*s}} + \frac{V(1)-V_O}{R2} + \frac{V(1)-V(2)}{\frac{1}{C2*s}} = 0$$

(2) $$\frac{V(2)-V(1)}{\frac{1}{C2*s}} + \frac{V(2)}{R1} = 0$$

(3) $V_O = V(2)$

Sustituyendo (3) en (2):

$$[(V_O - V(1)] * C2 * s + \frac{V_O}{R1} = 0 \longrightarrow [(V_O - V(1)] * R1*C2*s + V_O = 0$$

$$\longrightarrow V_O*(R1*C2*s + 1) = V(1)*R1*C2*s \longrightarrow V(1) = \frac{1+R1*C2*s}{R1*C2*s} * V_o$$

Una vez conocidas las relaciones entre las tensiones V(1) y V(2) respecto a la tensión de salida V_O, lo trasladamos a la expresión (1):

$$\left[\frac{1+R1*C2*s}{R1*C2*s} * V_O - V_{IN}\right] * C1*s + \frac{\frac{1+R1*C2*s}{R1*C2*s} * V_O - V_O}{R2} +$$

$$+ \left[\frac{1+R1*C2*s}{R1*C2*s} * V_O - V_O\right] * C2*s = 0 \quad \longrightarrow$$

$$\longrightarrow \quad (1 + R1*C2*s) * C1*R2*s * V_O - R1*C2*S*R2*s*C1* V_{IN} + V_O +$$

$$+ V_O * R1*C2*s = 0 \quad \longrightarrow$$

$$\longrightarrow V_O*(C1*R2*s + R1*R2*C1*C2*s^2 + R2*C2*s + 1) = R1*R2*C1*C2*s^2*V_{IN}$$

$$H(s) = \frac{V_O}{V_{IN}} = \frac{R1*R2*C1*C2*s^2}{R1*R2*C1*C2*s^2 + R2*(C1+C2)*s + 1} \tag{4.9}$$

Manipulando esta expresión, si sustituimos "s" por $s_n = s / w_{corte}$ y dividimos el numerador y el denominador por R1 * R2 * C1 * C2 * s_n^2 * w_{corte}^2 llegamos a que:

$$H(s_n) = \frac{1}{\frac{1}{w_{corte}^2*R1*R2*C1*C2}*\frac{1}{s_n^2} + \frac{R2*(C1+C2)}{w_{corte}*R1*R2*C1*C2}*\frac{1}{s_n} + 1}$$

En esta igualdad, teniendo en cuenta que fijaremos C1= C2, tenemos:

$$a_i = \frac{2}{w_{corte}*R1*C1} \quad y \quad b_i = \frac{1}{w_{corte}^2*R1*R2*C1^2}$$

Deducidos ambos coeficientes, determinada la *frecuencia de corte* impuesta por el usuario y establecida la condición de que las capacidades C1 y C2 sean iguales (C1=C2, valor definido previamente) para facilitar la resolución del circuito, solo nos queda conocer el dato de las resistencias:

$$R1 = \frac{1}{\Pi * f_{corte} * a_i * C1} \tag{4.10}$$

$$R2 = \frac{a_i}{4 * \Pi * f_{corte} * b_i * C1} \tag{4.11}$$

4.2. – Filtros de Rauch

La estructura de *Rauch* incorpora una característica adicional respecto de los filtros de *Sallen - Key* empleados en el proyecto que es la *ganancia G*, parámetro este que será introducido por el usuario en la fase de diseño.

Las células básicas utilizadas en la implementación del filtro completo serán objeto de estudio en los siguientes apartados.

4.2.1 – Etapa paso bajo de primer orden

La estructura de Rauch relacionada con este filtro responde a la siguiente figura:

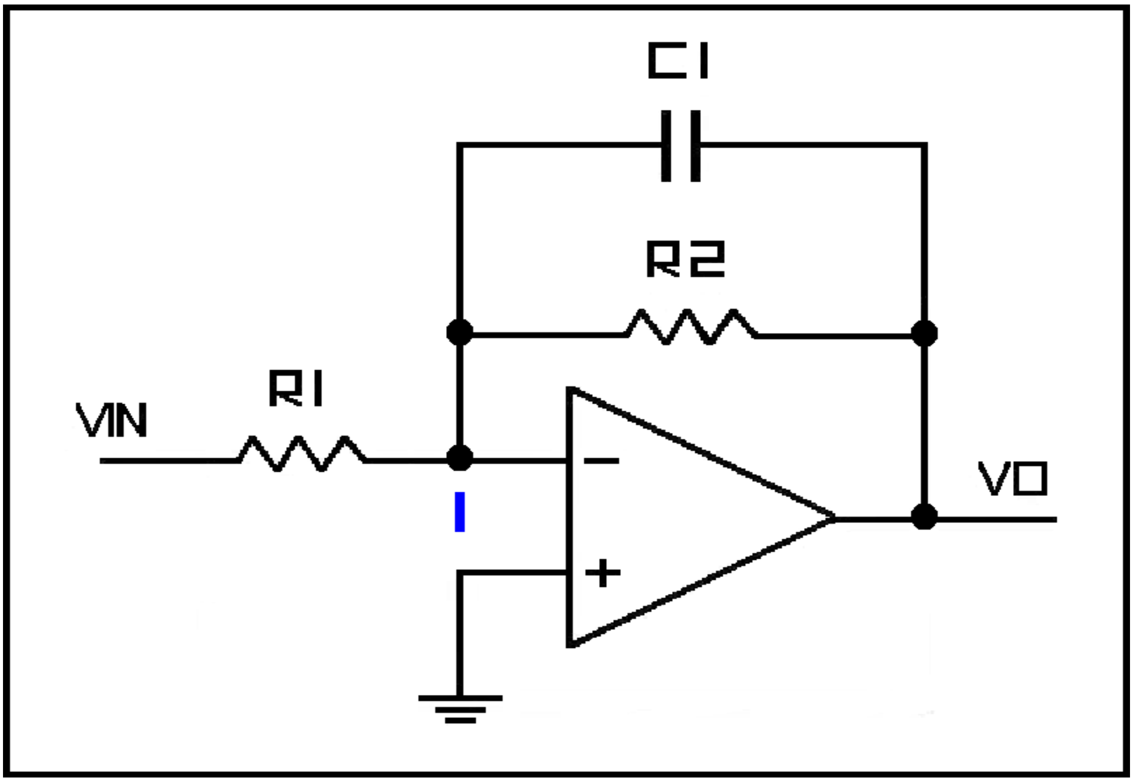


Figura 4.5: Filtro paso bajo de Rauch de primer orden en configuración de inversor.

Como se puede observar cuenta con una resistencia más que el filtro *Sallen - Key* de primer *orden.* Atendiendo al circuito en sí mismo y teniendo en cuenta que V(1) = 0, nos damos cuenta de que:

$$- \frac{V_{IN}}{R1} - \frac{V_O}{\frac{1}{C1*s}} - \frac{V_O}{R2} = 0 \quad \longrightarrow \quad - \frac{V_{IN}}{R1} - V_O*C1*s - \frac{V_O}{R2} = 0$$

$$\longrightarrow \quad - R2 * V_{IN} - R1 * R2 * C1 * s * V_O - R1 * V_O = 0$$

$$\longrightarrow \quad \frac{V_O}{V_{IN}} = \frac{-R2}{R1+R1*R2*C1*s}$$

Dividiendo arriba y abajo esta última expresión por R1, nos queda:

$$\boxed{H(s) = \frac{V_O}{V_{IN}} = \frac{-\mathrm{R2/R1}}{R2*C1*s + 1}} \qquad (4.12)$$

La presencia del signo negativo en la función de transferencia significa que el amplificador operacional en su configuración como inversor, provoca un desfase de 180° de la entrada con respecto a la salida.

Al igual que para los filtros anteriores, normalizaremos la variable compleja "s" por la expresión de $s_n = s / w_{corte} = j*w / w_{corte}$ en este caso:

$$H(s_n) = \frac{V_O}{V_{IN}} = \frac{-\mathrm{R2/R1}}{\mathrm{R2*C1}*s_n*w_{corte} + 1}$$

En esta igualdad, vemos que la *ganancia* $G = \frac{R2}{R1}$ (en valor absoluto) y también que el coeficiente a_i = R2 * C1 * s_n * w_{corte}. En este caso, el usuario establecerá el valor de la *frecuencia de corte f_{corte}*, la *ganancia G* y el dato del condensador C1.

La *ganancia* de cada etapa del filtro completo variará en función del número de etapas que este tenga puesto que la *ganancia* total *G* se repartirá por igual entre cada una de ellas.

Así por ejemplo, las células de primer *orden* que nos ocupan se utilizarán como primera etapa en todos aquellos filtros de *orden* impar (1, 3 y 5), los cuales constan de 1, 2 y 3 etapas respectivamente.

Si queremos que la función de transferencia del filtro absoluto sea el producto de las funciones de transferencia de cada una de las células individuales, la *ganancia* en cada etapa tomará el valor *G* (filtros de oden 1), $\sqrt{G}$ (filtros de orden 3) y $\sqrt[3]{G}$ (filtros de orden 5).

Asumiendo esta ciscunstancia, calcularemos las resistencias R1 y R2, sabiendo que el valor de R1 depende de la *ganancia* y que por tanto, su valor será diferente según las etapas que tenga el filtro diseñado:

$$\boxed{R2 = \frac{a_i}{2*\Pi*f_{corte}*C1}} \tag{4.13}$$

- Filtro de primer *orden* (1 etapa) $$R1 = \frac{R2}{G}$$ (en valor absoluto) (4.14)

- Filtro de tercer *orden* (2 etapas) $$R1 = \frac{R2}{\sqrt{G}}$$ (en valor absoluto) (4.15)

- Filtro de quinto *orden* (3 etapas) $$R1 = \frac{R2}{\sqrt[3]{G}}$$ (en valor absoluto) (4.16)

4.2.2 – Etapa paso bajo de segundo orden

La estructura básica que se ajusta a este tipo de filtros es la que se plantea a continuación:

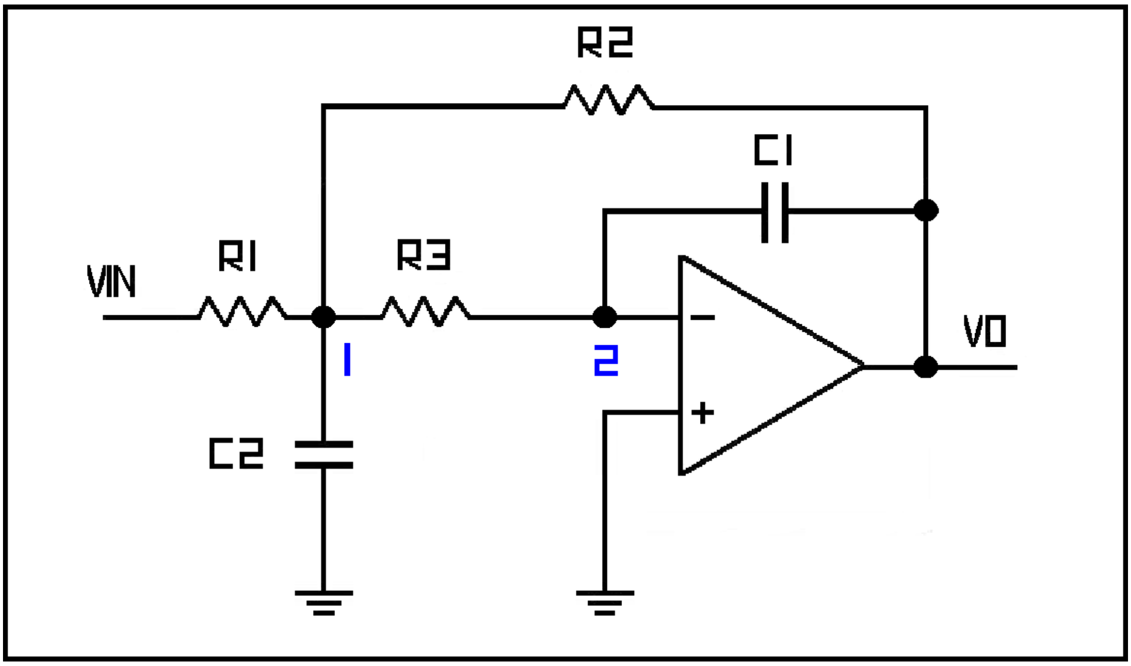


Figura 4.6: Filtro paso bajo de Rauch de segundo orden en configuración de inversor.

A diferencia de las etapas *Sallen - Key* anteriores, en este caso se incluye una resistencia adicional. Inspeccionando el circuito, vemos que la tensión en el punto 2, V(2) = 0. A partir de ahí, nos encontramos con las siguientes relaciones:

(1) $$\frac{V(1)-V_{IN}}{R1} + \frac{V(1)}{\frac{1}{C2*s}} + \frac{V(1)-V_O}{R2} + \frac{V(1)}{R3} = 0$$

(2) $$-\frac{V(1)}{R3} - \frac{V_O}{\frac{1}{C1*s}} = 0 \longrightarrow -\frac{V(1)}{R3} - V_O*C1*s = 0 \longrightarrow$$

$$\longrightarrow -V(1) - R3*C1*s = 0 \longrightarrow V(1) = -R3*C1*s*V_O$$

Si encajamos este valor de V(1) en el desarrollo (1):

$$-\frac{R3*C1*s*V_O - V_{IN}}{R1} + R3*C1*s*V_O*C2*s - \left(\frac{R3*C1*s*V_O + V_O}{R2}\right)$$

$$-\frac{R3*C1*s*V_O}{R3} = 0 \longrightarrow$$

$$\longrightarrow - R3*R2*C1*s*V_O - R2*V_{IN} - R1*R2*R3*C1*C2*s^2*V_O -$$

$$- R3*R1*C1*s*V_O - R1*V_O - R1*R2*C1*s*V_O = 0 \longrightarrow$$

$$\frac{V_O}{V_{IN}} = \frac{-R2}{R1*R2*R3*C1*C2*s^2+(R3*R2+R3*R1+R1*R2)*C1*s+R1}$$

Al igual que en el filtro anterior, dividimos numerador y denominador por R1, obteniendo la función de transferencia que sigue:

(4.17)

$$H(s) = \frac{V_O}{V_{IN}} = \frac{-\,\mathrm{R2/R1}}{R2{*}R3{*}C1{*}C2{*}s^2 + C1{*}\left(R2 + R3 + \frac{R3{*}R2}{R1}\right){*}s{+}1}$$

Haciendo el cambio de variable "s" por la variable compleja normalizada $s_n = s / w_{corte}$ nos queda:

$$H(s_n) = \frac{-\,\mathrm{R2/R1}}{R2{*}R3{*}C1{*}C2{*}s_n^2{*}w_{corte}^2 + C1{*}\left(R2 + R3 + \frac{R3{*}R2}{R1}\right){*}s_n{*}w_{corte} + 1}$$

De la ecuación anterior, la *ganancia* $G = \frac{R2}{R1}$ (en valor absoluto), los coeficientes $a_i = C1 * (R2 + R3 + \frac{R2{*}R3}{R1}) * w_{corte}$, mientras que el otro

$b_i = R2{*}R3{*}C1{*}C2{*}w_{corte}^2$.

Una vez más, la *frecuencia de corte* f_{corte}, la *ganancia G* y el valor de las capacidades C1 y C2 serán elegidos por el usuario y junto a los coeficientes específicos a_i b_i son datos conocidos.

La *ganancia* de cada etapa de segundo *orden* del filtro será diferente según el número de etapas que conformen el diseño final.

Utilizando el mismo razonamiento, para el cálculo de las resistencias R1 y R2 se emplearán ecuaciones en las que cambiará el valor de la *ganancia*.

- Filtros de 2° *orden* (1 etapa):

$$R2 = \frac{a_i * C2 - \sqrt{(a_1 * C2)^2 - 4 * b_i * C1 * C2(1+G)}}{4 * \Pi * f_{corte} * C1 * C2} \qquad (4.18)$$

$$R1 = \frac{R2}{G} \qquad (4.19)$$

$$R3 = \frac{b_i}{4 * \Pi^2 * f_{corte}^2 * C1 * C2 * R2} \qquad (4.20)$$

- Filtros de 3° y 4° *orden*: Las expresiones son las mismas, sustituyendo la *ganancia G* por $\sqrt{G}$. Al ser un filtro de 2 etapas, la *ganancia* se reparte a la mitad entre cada una de ellas.

- Filtros de 5° y 6° *orden*: En este caso, se intercambia el dato de la *ganancia G* por $\sqrt[3]{G}$ (tenemos 3 etapas y por tanto, se distribuye la tercera parte de la *ganancia*).

Para que la resistencia R2 tenga valores reales (es decir, que no sea negativa), se debe de cumplir una cierta relación entre los condensadores C1 y C2, en la que, al estar también involucrados los datos de *ganancia G* y de los coeficientes a_i b_i, dicha relación variará en función del número de etapas totales y de la célula de segundo *orden* que estemos diseñando.

- Filtros de 2° *orden* (1 etapa): $$C2 > \frac{C1*4*b_i*(1+G)}{a_i^2}$$ (4.21)

- Filtros de 3° y 4° *orden* (2 etapas): $$C2 > \frac{C1*4*b_i*(1+\sqrt{G})}{a_i^2}$$ (4.22)

- Filtros de 5° y 6° *orden* (3 etapas): $$C2 > \frac{C1*4*b_i*(1+\sqrt[3]{G})}{a_i^2}$$ (4.23)

4.2.3 – Etapa paso alto de primer orden

El filtro de *Rauch* asociado a esta estructura es el siguiente:

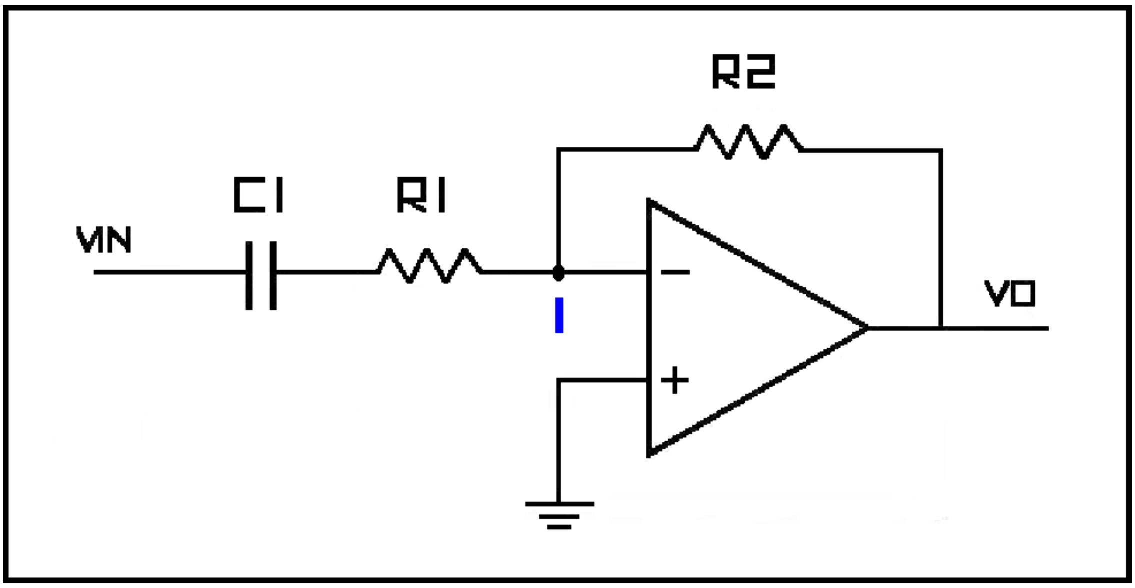


Figura 4.7: Filtro paso alto de Rauch de primer orden en configuración de inversor.

En lo que se refiere al filtro paso alto Rauch de la figura 4.7 y deduciendo que la tensión en el punto 1, V(1) = 0, observamos lo siguiente:

$$- \frac{V_{IN}}{\frac{1}{C1*s} + R1} - \frac{V_O}{R2} = 0 \quad \longrightarrow \quad - \frac{C1*s*V_{IN}}{1 + R1*C1*s} - \frac{V_O}{R2} = 0$$

$$\longrightarrow \quad - C1*R2*s*V_{IN} - (1 + R1*C1*s)*V_O = 0 \quad \longrightarrow \quad \frac{V_O}{V_{IN}} = \frac{-C1*R2*s}{1 + R1*C1*s}$$

Manipulando el resultado anterior, lo que se hace es dividir ambos términos de la expresión por el factor R1*C1*s con lo que la función resultante es la mostrada a continuación:

$$\boxed{H(s) = \frac{V_O}{V_{IN}} = \frac{-\ \mathrm{R2/R1}}{1 + \frac{1}{R1*C1} * \frac{1}{s}}} \qquad (4.24)$$

Si normalizamos "s" por la expresión $s_n = s / w_{corte}$, la función de transferencia normalizada es:

$$H(s_n) = \frac{-\ \mathrm{R2/R1}}{1 + \frac{1}{R1*C1} * \frac{1}{s_n * w_{corte}}}$$

Definiendo el usuario la *frecuencia de corte* f_{corte} , la *ganancia G* y el dato de la capacidad C1 y asumiendo conocido el coeficiente $a_i = \frac{1}{R1 * C1 * w_{corte}}$,

obtendremos el valor de los componentes resistivos R1 y R2, considerando que esta última depende de la *ganancia* y que, por tanto la expresión que lo representa cambiará en función del *orden* del filtro.

$$\boxed{R1 = \frac{1}{2 * \Pi * f_{corte} * a_1 * C1}} \qquad (4.25)$$

$$\boxed{R2 = R1 * G} \qquad \text{(filtro de orden 1)} \qquad (4.26)$$

$$\boxed{R2 = R1 * \sqrt{G}} \qquad \text{(filtro de orden 3)} \qquad (4.27)$$

$$\boxed{R2 = R1 * \sqrt[3]{G}} \qquad \text{(filtro de orden 5)} \qquad (4.28)$$

4.2.4 – Etapa paso alto de segundo orden

En este punto se abordará el análisis de la celda Rauch paso alto de segundo orden cuya representación gráfica corresponde a la figura 4.

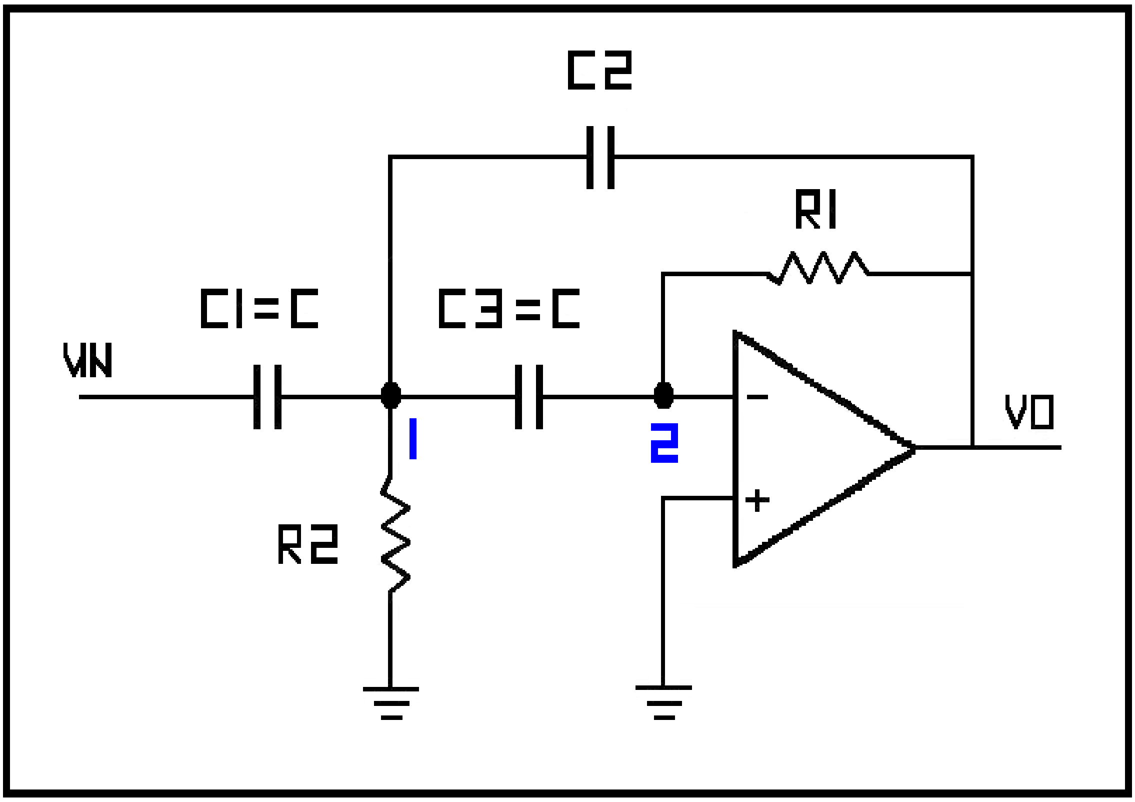


Figura 4.8: Filtro paso alto de Rauch de segundo orden en configuración de inversor.

Continuando con el mismo proceso utilizado para los filtros anteriores, plantearemos las ecuaciones de nudos, considerando que la tensión en el punto 2, V(2) = 0:

(1) $$\frac{V(1)-V_{IN}}{\frac{1}{C1*s}} + \frac{V(1)}{R2} + \frac{V(1)-V_O}{\frac{1}{C2*s}} + \frac{V(1)}{\frac{1}{C3*s}} = 0$$

(2) $- \dfrac{V(1)}{\frac{1}{C3*s}} - \dfrac{V_O}{R1} = 0 \longrightarrow - V(1)*C3*s - \dfrac{V_O}{R1} = 0$

$\longrightarrow$ V(1) = $- \dfrac{V_O}{R1*C3*s}$

Aplicando este cociente V(1) en la ecuación (1), tenemos lo siguiente:

$$- \left[\frac{V_O}{R1*C3*s} - V_{IN}\right] *C1*s - \frac{V_O}{R1*R2*C3*s} - \left[\frac{V_O}{R1*C3*s} - V_O\right] *C2*s$$

$$- \frac{V_O*C3*s}{R1*C3*s} = 0 \longrightarrow$$

$$\longrightarrow V_O*C1*R2 * s + R1*R2*C1*C3*s^2 * V_{IN} + V_O + V_O * C2*R2*s$$

$$+ R1*R2*C2*C3*s^2*V_O + R2*C3*s*V_O = 0 \longrightarrow$$

$$\longrightarrow V_O * [R1*R2*C2*C3*s^2 + R2*(C1+C2+C3)*s + 1] = - R1*R2*C1*C3*s^2 * V_{IN}$$

$$\longrightarrow \frac{V_O}{V_{IN}} = \frac{- R1*R2*C1*C3*s^2}{R1*R2*C2*C3*s^2 + R2*(C1+C2+C3)*s + 1}$$

Para simplificar los cálculos, fijaremos la condición previa de que los condensadores C1 y C3 sean iguales (C1 = C3 = C). Teniendo en cuenta esta premisa, el desarrollo queda como sigue:

$$H(s) = \frac{V_O}{V_{IN}} = \frac{-\ R1*R2*C^2*s^2}{R1*R2*C*C2*s^2 + R2*(2*C + C2)*s + 1} \tag{4.29}$$

Dividiendo ambos términos del cociente por el factor R1*R2*C*C2*S^2, el resultado es el siguiente:

$$H(s) = \frac{\frac{-C}{C2}}{\frac{1}{R1*R2*C*C2} * \frac{1}{s^2} + \frac{2*C + C2}{R1*C*C2} * \frac{1}{s} + 1}$$

El signo negativo al igual que en el filtro anterior, denota que existe un desfase de 180º de la entrada respecto de la salida al estar el amplificador configurado como inversor.

Realizando el cambio de la variable compleja "s" por el de la normalizada $s_n = s / w_{corte}$ la expresión modificada es la que se indica:

$$H(s_n) = \frac{\frac{-C}{C2}}{\frac{1}{R1*R2*C*C2} * \frac{1}{s_n^2 * w_{corte}^2} + \frac{2*C + C2}{R1*C*C2} * \frac{1}{s_n * w_{corte}} + 1}$$

De la ecuación resultante, se desprende que la *ganancia* $G = \frac{C}{C2}$ (en valor absoluto), los coeficientes $a_i = \frac{2*C+C2}{R1*C*C2*w_{corte}}$ mientras que el otro $b_i = \frac{1}{R1*R2*C*C2*w_{corte}^2}$.

El usuario elegirá el valor C de los condensadores C1 y C3. Obtenido este dato, calcularemos la capacidad C2 que, como vemos depende de la *ganancia G* y por lo tanto, la expresión que lo representa será diferente según las etapas que contenga el filtro completo.

- Filtros de 2° *orden* (1 etapa): $$C2 = \frac{C}{G} \tag{4.30}$$

- Filtros de 3° y 4° *orden* (2 etapas): $$C2 = \frac{C}{\sqrt{G}} \tag{4.31}$$

- Filtros de 5° y 6° *orden* (3 etapas): $$C2 = \frac{C}{\sqrt[3]{G}} \tag{4.32}$$

El razonamiento en cuanto al reparto equitativo de la *ganancia* entre las diferentes etapas, es el mismo al planteado para todos los filtros de *Rauch* analizados.

Considerando conocida la *frecuencia de corte* f_{corte} , la *ganancia G*, los coeficientes a_i b_i y el valor de los condensadores C y C2, deduciremos las resistencias R1 y R2.

$$R1 = \frac{1 + 2 * (C/C2)}{2 * \Pi * f_{corte} * a_i * C} \tag{4.33}$$

$$R2 = \frac{a_i}{2 * \Pi * f_{corte} * b_i * C * (1 + 2 * (C/C2))} \tag{4.34}$$

4.3. – Implementación del filtro diseñado utilizando celdas básicas Sallen – Key y Rauch

Las etapas de 1º y 2º *orden* estudiadas, tanto de *Sallen - Key* como de *Rauch* constituyen los bloques elementales para la construcción de filtros de un *orden* mayor.

Esta estrategia de conectar en cascada estructuras de 1º y 2º *orden* para conseguir filtros superiores (hasta de 6º *orden* en nuestro proyecto) es la que desarrollamos a lo largo del trabajo.

Como ya se comentó en alguna ocasión, la función de transferencia *H(s)* del filtro total es el producto de las funciones de transferencia de cada una de las etapas que lo componen.

La figura 4.9 muestra de qué manera se obtienen los filtros de *órdenes* más elevados empleando básicamente células de 1º y 2º *orden*.

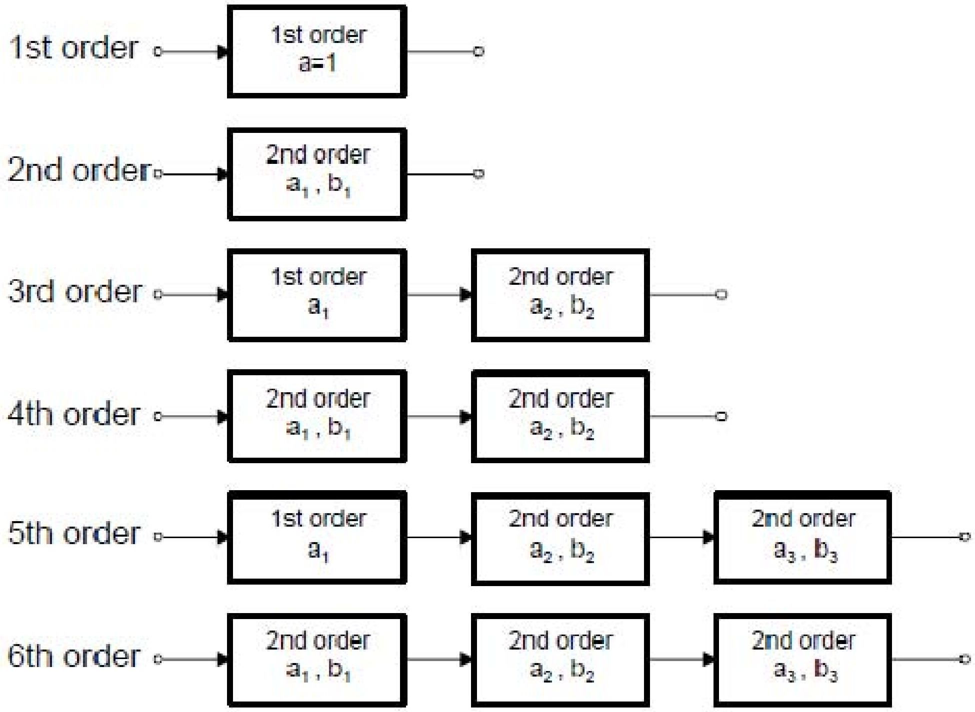


Figura 4.9: Conexión en cascada de filtros de 1° y 2° orden para la obtención de filtros de orden superior.

Observando el esquema, nos damos cuenta de que los filtros de *orden* impar (1, 3 y 5) cuentan con una etapa de 1º *orden* al inicio, mientras que el resto están integrados únicamente por estructuras de 2º *orden*.

Por otra parte, los coeficientes de cada etapa son diferentes según la situación que ocupen dentro del filtro general (1º, 2º o 3º etapa). Este hecho queda reflejado en las tablas de coeficientes incluidas en el anexo que acompaña al proyecto.

A su vez, cabe mencionar que la implementación de los filtros *paso banda* se realizó a partir de las células *paso bajo* y *paso alto* de 1º y 2º *orden* estudiadas a lo largo del capítulo, tal y como aparece representado en el diagrama posterior 4.10.

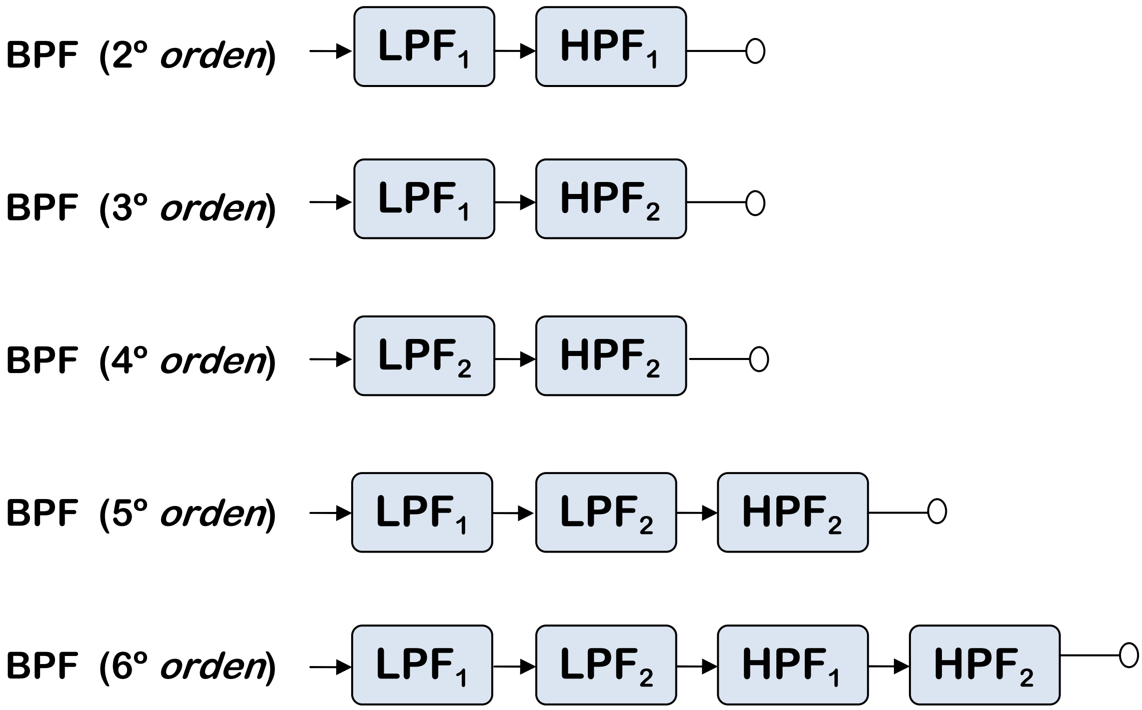


Figura 4.10: Conexión en cascada de filtros paso bajo y paso alto de 1° y 2°orden para la obtención de filtros paso banda.

Identificando términos, **LPF** son los filtros *paso bajo*, **HPF** los filtros *paso alto*, **BPF** los filtros *paso banda* y los subíndices 1 y 2 corresponden al *orden* de cada etapa (1º y 2º).

CAPÍTULO V

DESCRIPCIÓN DEL SOFTWARE DESARROLLADO

En este capítulo abordaremos el análisis detallado del software desarrollado en la realización del presente proyecto incluyendo, no obstante, una breve introducción al entorno Matlab y a la utilidad seleccionada para crear la interfaz gráfica (GUIDE), así como la justificación de su elección para el desarrollo de todo el trabajo.

5.1. – Introducción a Matlab

Matlab es al mismo tiempo, un entorno y un lenguaje de programación de alto nivel. Se trata de un programa interactivo para computación numérica y visualización de datos basado en matrices y vectores. Así mismo es una plataforma fácil de usar, flexible y potente.

Sus siglas provienen originalmente de las iníciales "***MAT****rix* ***LAB****oratory*". Además del módulo principal, existe la posibilidad de añadir a su entorno múltiples módulos de aplicación especializados en diferentes ramas de la ciencia e ingeniería (*toolboxes*) como puede ser el procesado de señal, los sistemas de control, el estudio de las ondas, etc.

En este sentido ha sido capaz de extender sus capacidades de cálculo a numerosos campos profesionales, siendo muy requerido en el desarrollo de algoritmos, adquisición de datos, modelado, simulación y análisis entre otras aplicaciones.

5.2. – Creación del interfaz gráfico (GUIDE)

Como ya hemos comentado, la herramienta GUIDE es la utilidad que nos proporciona Matlab para la realización de la parte gráfica de nuestro programa y es la que emplearemos en la creación de las GUIs (Interfaces Gráficas de Usuario).

Dicha herramienta GUIDE se compone de dos partes específicas. Por un lado, construiremos el *dibujo del GUI*. En este apartado incluiremos ventanas, botones, paneles, menús desplegables, etc.

En definitiva, formaremos el entorno gráfico a través del cual, el usuario podrá interactuar. Cada una de esta interfaces diseñadas, se guardarán por defecto como ficheros de extensión *.fig*.

Una vez decidido cómo será nuestra GUI, nos queda realizar la parte de programación. La herramienta GUIDE nos permite programar cada uno de los controles que forman nuestra interfaz gráfica, de tal manera que se ejecuten los procesos y las sentencias que nosotros deseemos.

Así por ejemplo, si lo que se pretende es conservar el resultado obtenido de una aplicación, apretando el botón correspondiente para ello, se iniciará el proceso que pueda permitir guardar dichos datos de la forma que nosotros hayamos programado con anterioridad.

Toda esta tarea se almacenará en ficheros de extensión .m, los cuales recibirán el mismo nombre que sus archivos *.fig* asociados y previamente originados por el usuario.

5.2.1 - Funcionamiento del GUI

Cada elemento introducido en nuestro GUI tiene asociado un código de programa (*callback*). La ejecución de cada *callback* se inicia en el momento en el que el usuario realiza una determinada acción, dependiendo del tipo de control.

En este sentido, un menú desplegable procederá a la activación de su *callback* cuando hagamos click sobre la opción seleccionada. Nosotros como programadores y en función de nuestras necesidades, decidiremos los eventos que se van a ejecutar.

El modelo de programación empleado, más conocido como *programación conducida por eventos* es asíncrono, puesto que depende únicamente de la interacción del usuario con el GUI.

Dicha característica tan singular ha representado en la práctica, una dificultad añadida en nuestra tarea por el hecho de tener que predecir y corregir situaciones imprevisibles y no deseadas que pudieran ser generadas por el usuario.

Una muestra de todo ello es la introducción incorrecta de datos requeridos por pantalla (resistencias, condensadores o frecuencias de valor nulo o negativo por ejemplo), así como una ilógica secuencia de ejecución del software (intentar simular el circuito sin conocer el valor de sus componentes es una posibilidad), cuestiones estas, que pudieran derivar en un deficiente funcionamiento y que ya fueron previstas junto con algunas otras.

5.3. – Descripción del Programa de Control

En este apartado describiremos el proceso realizado para la elaboración del programa así como su funcionamiento, incluyendo las distintas utilidades y opciones que presenta.

5.3.1 - Requisitos previos

El proyecto que se plantea tiene como objetivo el diseño primero y la simulación después mediante la obtención del *diagrama de Bode* de filtros *paso bajo, paso alto* o *paso banda* hasta orden 6, utilizando las aproximaciones matemáticas de *Butterworth*, de *Chebyshev* o de *Bessel* e implementados con los circuitos de *Sallen-Key* o de *Rauch* descritos en el capítulo anterior.

Teniendo claro este objetivo inicial y definiendo como queremos que sea la resolución posterior, nos planteamos una serie de requisitos que debe cumplir nuestro software.

- Los resultados del programa (valor de los componentes del circuito, *frecuencia de corte, ganancia, ancho de banda, factor de calidad*, etc, así como las gráficas de amplitud y de fase del *diagrama de Bode* correspondiente) deben poder ser utilizados de manera sencilla y eficaz.

- Así mismo, el software tiene que ser construido en un entorno visual, de tal manera que sea muy directa la interacción con el usuario y como consecuencia, resulte fácil la introducción e interpretación posterior de los datos.

- Siempre que nos sea posible, es recomendable realizar el programa de forma modular para que cualquier persona pueda ampliarlo o modificarlo en un futuro sin aparente dificultad.

- Otro aspecto que pudiera ser interesante es la posibilidad de diseñar un programa capaz de ser ejecutado en otros sistemas operativos y hardware del mercado.

Planteando inicialmente estos requisitos, vemos que Matlab está capacitado para cumplir todos y cada uno de ellos.

Por una parte, el usuario podrá interactuar con los datos, tanto de entrada como de salida de un modo muy simple. Para ello, la herramienta GUIDE de Matlab nos permite trabajar en un entorno gráfico que proporcionará al interesado una mayor comodidad e interrelación con el programa desarrollado.

En cuanto a la realización de un software lo más modular posible, comentar que se crearon funciones de entrada y de salida de datos, funciones de representación de pantalla, rutinas de cálculo independientes, etc.

Todo ello enfocado a la posibilidad de realizar cambios en un futuro, para los cuales no haya que modificar todo el código sino solamente la función concreta correspondiente.

Por último, mencionar que Matlab posee herramientas de compilación para crear un ejecutable independiente. Por tanto, existiría la opción de ejecutar el software en cualquier ordenador, incluso sin haber sido instalado y sea cual sea su sistema operativo. Sin embargo, esta alternativa teórica no fue comprobada en la práctica.

5.3.2 - Especificaciones

Una vez justificada la utilización y la óptima adecuación del entorno Matlab en la realización de nuestro proyecto, definiremos como y de qué manera queremos que funcione el programa.

En primer lugar, se pretende que el software sea capaz de proporcionar al usuario las herramientas necesarias para el diseño del filtro. Para ello, tendremos que construir por un lado la parte gráfica y por el otro las rutinas del programa que lo hagan operativo.

En la elaboración de la ventana para decidir el diseño, contaremos con los botones, las casillas de introducción de datos y los menús correspondientes en cada caso, a partir de los cuales se puedan seleccionar tanto el tipo de filtro (*paso bajo*, *paso alto* o *paso banda*) como la aproximación matemática empleada (*Butterworth*, *Chebyshev* o *Bessel*).

Todo ello junto al circuito o estructura básica seleccionada (*Sallen-Key* o *Rauch*) que suponga el punto de partida para su implementación.

A su vez, deberá poder introducirse por pantalla el *orden*, la *frecuencia de corte* (que será *inferior* o *superior* en filtros *paso banda*) y la *ganancia* (sólo en el caso de filtros *Rauch*, puesto que la configuración *Sallen-Key* tiene ganancia unidad, tal y como vimos en el capítulo anterior).

Superada la fase de diseño, el programa deberá incorporar las opciones que precise para la simulación del filtro proyectado (en concreto, para escenificar el volcado por pantalla del *diagrama de Bode* correspondiente).

Con este objetivo y como paso previo, se definirá el valor de todos los componentes, ya sean fijados de forma interactiva por el usuario o calculados internamente.

Conocidas todas las características del filtro, obtendremos su *diagrama de Bode*, teniendo la posibilidad de guardar tanto los resultados numéricos (valores de *ganancia* y de *fase* junto a sus frecuencias asociadas) como los datos de diseño que lo definen.

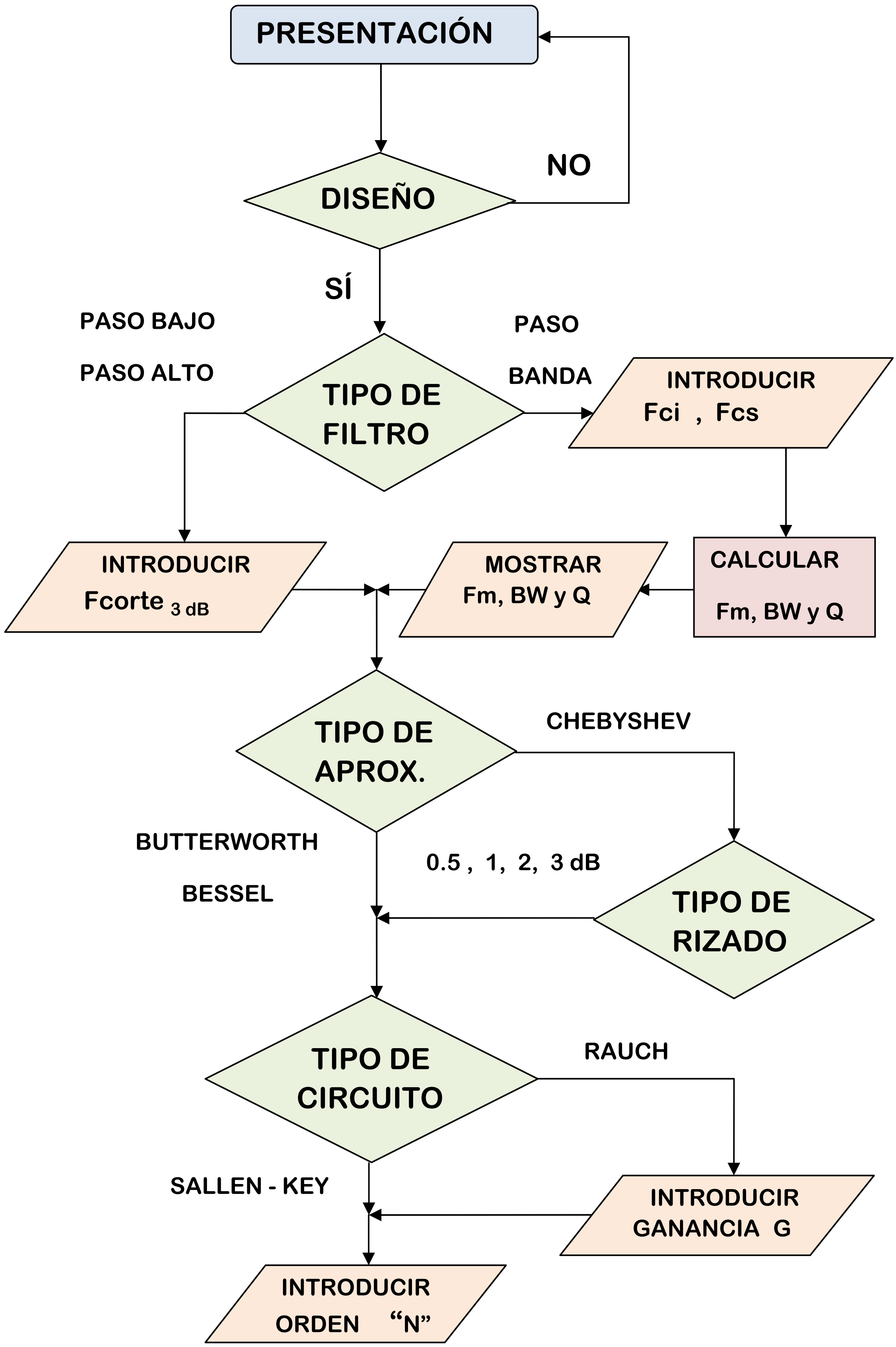
5.3.3 – Arquitectura

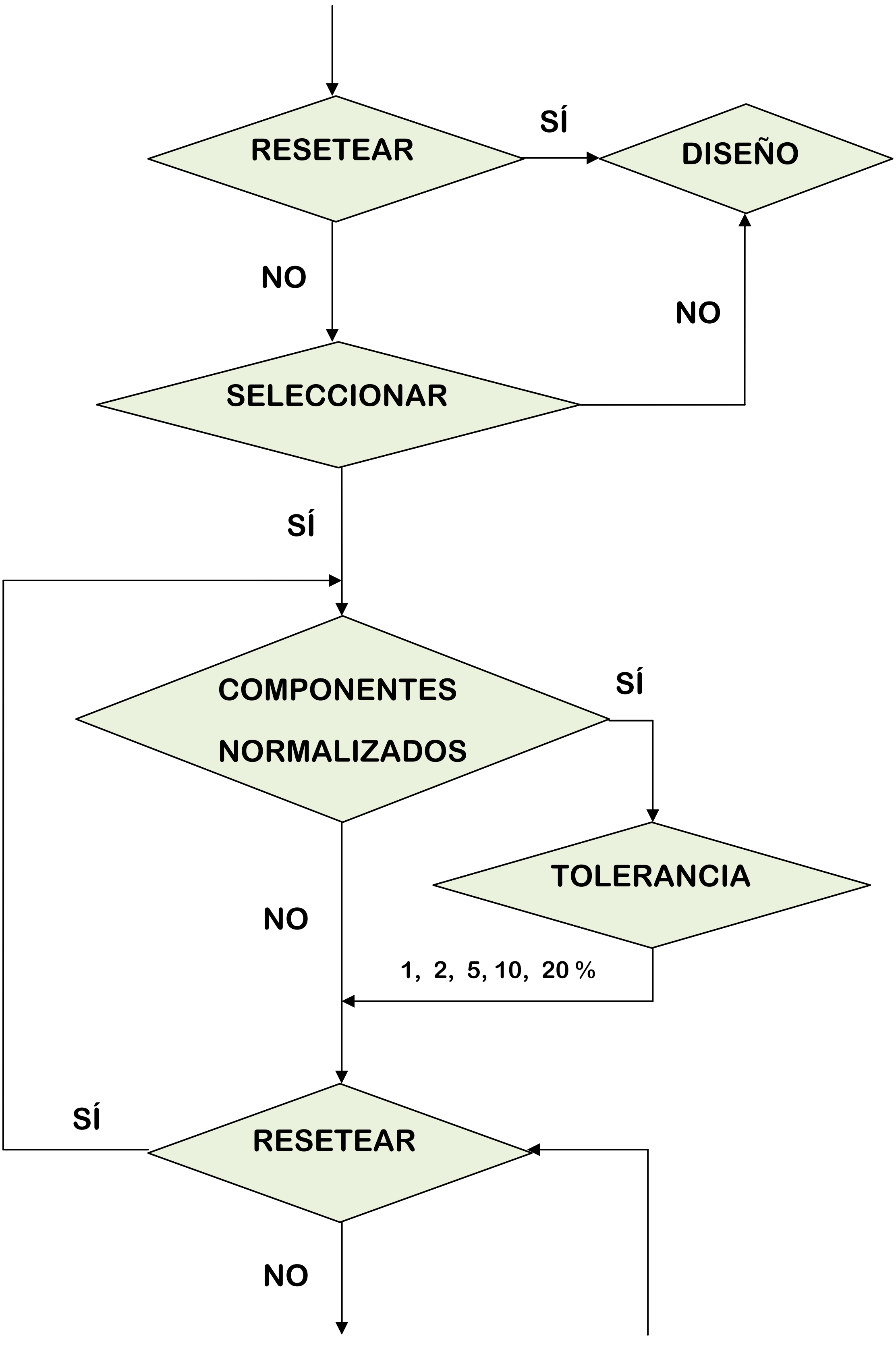
A modo de resumen o esquema general del funcionamiento de nuestro programa, adjuntamos su *diagrama de flujo*, tal y como aparece en la figura 5.1 descrita en las páginas sucesivas.

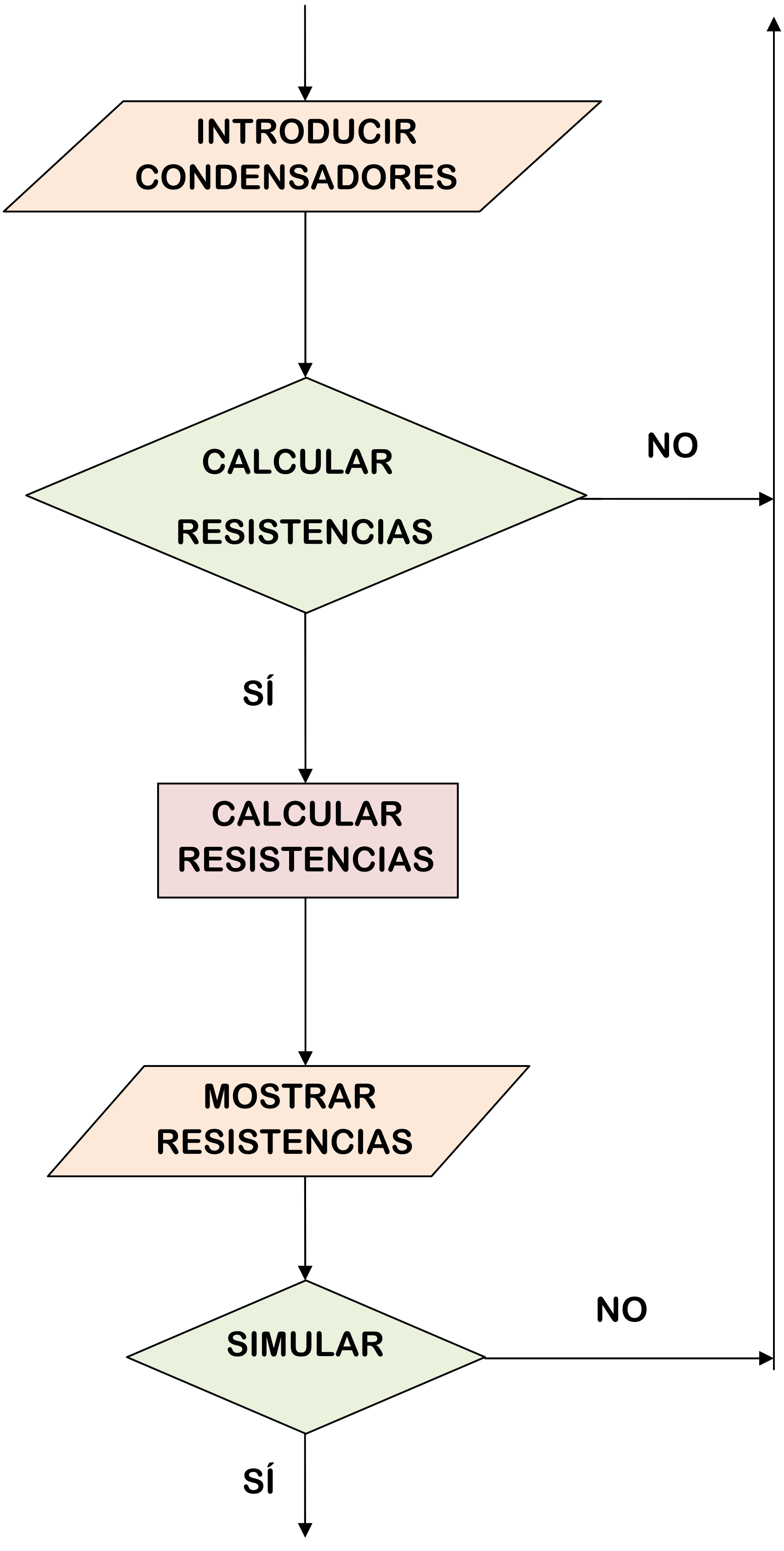
De esta manera, se representa tanto su secuencia lógica de ejecución como las rutinas fundamentales que lo componen. Más adelante, detallaremos los procesos que tienen lugar, las funciones involucradas y los eventos responsables de su puesta en práctica.

Como puede observarse, el programa se inicia con una pantalla de presentación que da lugar a la ventana correspondiente al diseño del filtro.

Una vez situado en dicha ventana y en función de cada una de las opciones seleccionadas, se activarán unas casillas de introducción de datos u otras y se irán abordando los diferentes eventos presentados.


PRESENTACIÓN
DISEÑO
NO
SÍ
PASO BAJO
PASO ALTO
TIPO DE FILTRO
PASO
BANDA
INTRODUCIR
Fci , Fcs
INTRODUCIR
Fcorte 3 dB
MOSTRAR
Fm, BW y Q
CALCULAR
Fm, BW y Q
TIPO DE APROX.
CHEBYSHEV
BUTTERWORTH
BESSEL
0.5 , 1, 2, 3 dB
TIPO DE RIZADO
TIPO DE CIRCUITO
RAUCH
SALLEN - KEY
INTRODUCIR
GANANCIA G
INTRODUCIR
ORDEN "N"


RESETEAR
SÍ
DISEÑO
NO
NO
SELECCIONAR
SÍ
COMPONENTES
NORMALIZADOS
SÍ
TOLERANCIA
NO
1, 2, 5, 10, 20 %
SÍ
RESETEAR
NO


INTRODUCIR CONDENSADORES
CALCULAR RESISTENCIAS
NO
SÍ
CALCULAR RESISTENCIAS
MOSTRAR RESISTENCIAS
SIMULAR
NO
SÍ

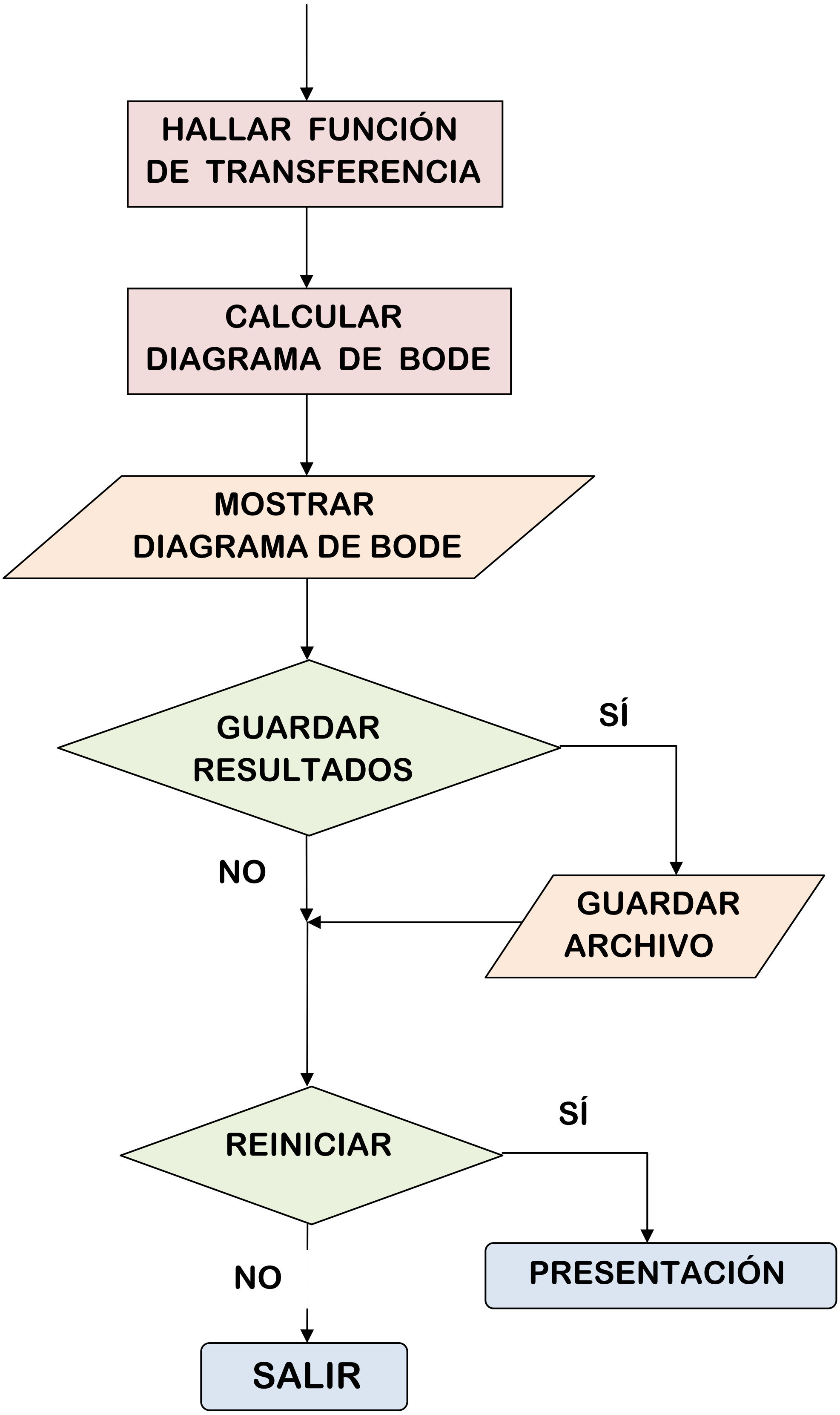


Figura 5.1: Diagrama de Flujo del software desarrollado.

5.3.4 - Descripción del entorno gráfico y de las funciones más importantes

Inicialmente la aplicación se ejecuta escribiendo en la línea de comandos de Matlab, la palabra *FILACT* (Filtros Activos), circunstancia esta que da origen a la ventana de presentación de nuestro programa, tal y como se puede observar más abajo.

Existe la alternativa de compilar el código en otro lenguaje y crear un ejecutable, opción que no ha sido probada.

Figura 5.2: Pantalla de presentación del software.

La función creada ***FILACT.m*** junto a la asociada de extensión *.fig* que genera la interfaz gráfica, es la que se encarga de lanzar la aplicación.

Dicha ventana contiene además del título, el botón "**Continuar**" que al ser apretado da lugar a la pantalla de diseño del filtro, a la cual corresponde la siguiente imagen:

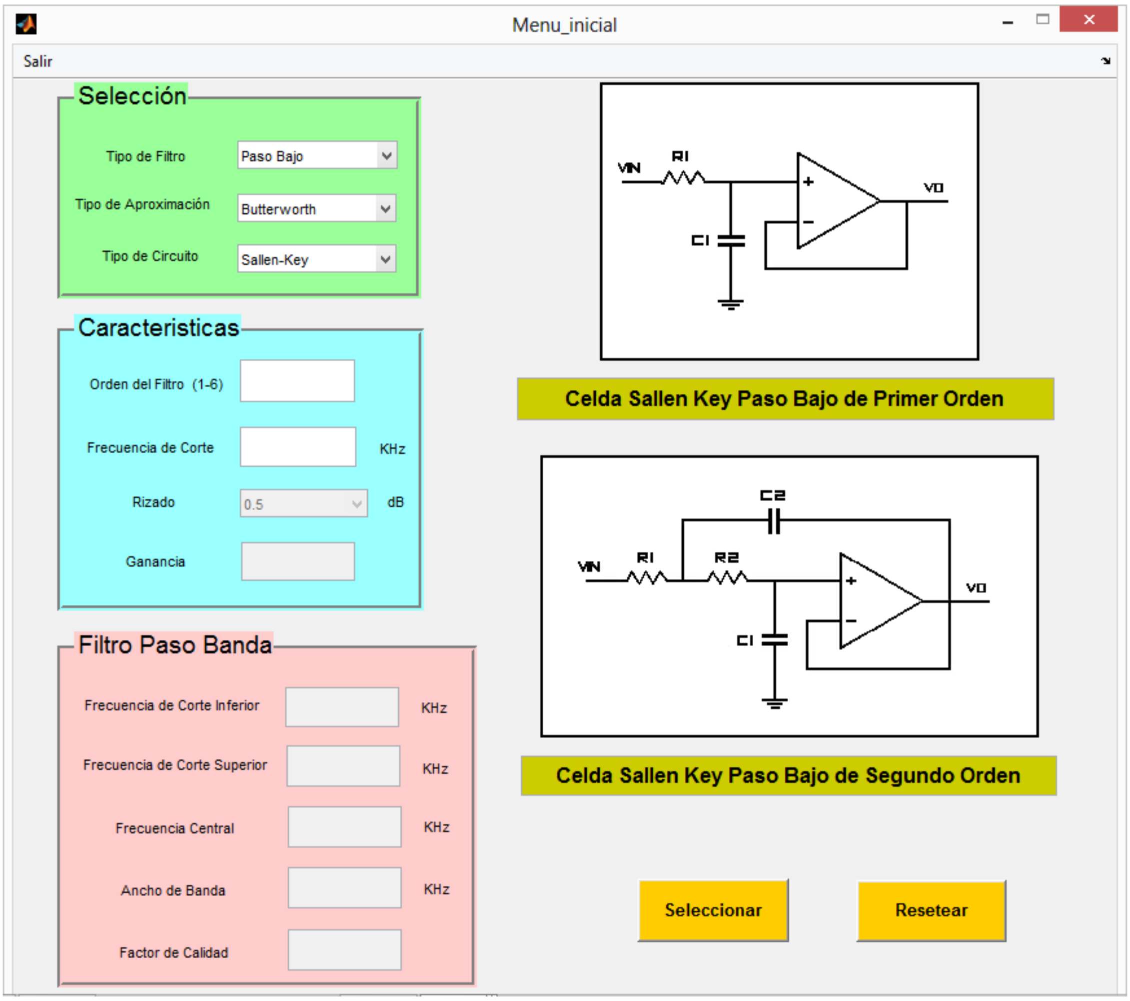


Figura 5.3: Menú inicial de diseño del filtro.

Esta ventana fue programada dentro de la función ***Menu_inicial.m***. Si nos fijamos de forma más detallada, vemos que se puede seleccionar por un lado, el tipo de filtro deseado (*paso bajo, paso alto* o *paso banda*) y por el otro, la aproximación matemática que aplicaremos (*Butterworth, Chebyshev* o *Bessel*).

Así mismo el usuario también tiene la posibilidad de elegir los circuitos, a través de los cuales implementar el filtro diseñado (*Sallen-Key* o *Rauch*).

Para ello, desplegará cada uno de los menús que aparecen, pinchando en la flecha de menú correspondiente, de la manera que se muestra:

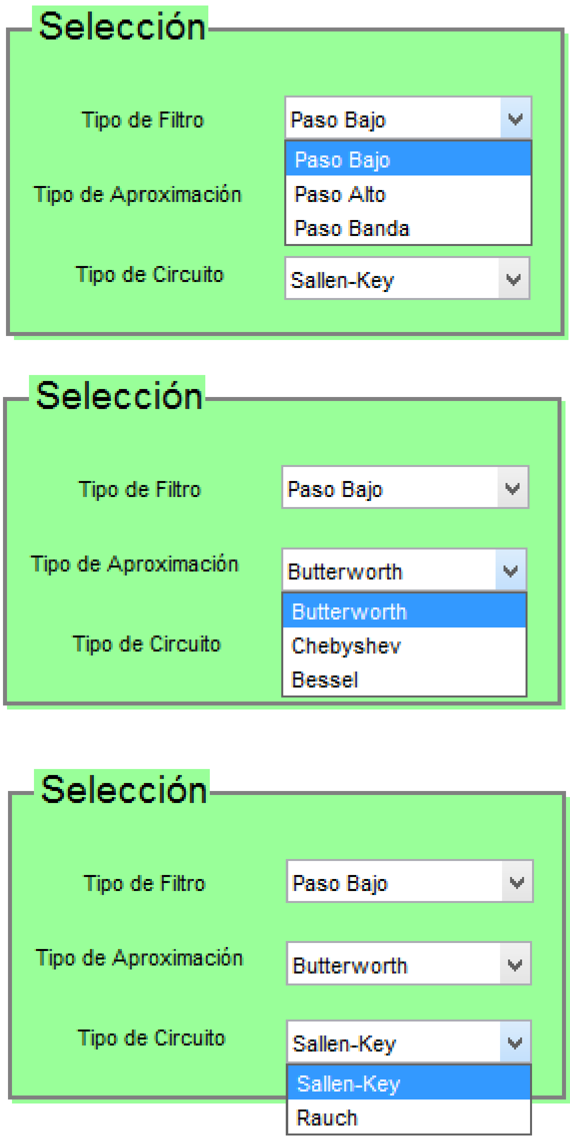


Figura 5.4: Menús desplegables de clasificación del filtro.

En función de la selección realizada, se habilitarán unas u otras opciones de configuración.

Así por ejemplo, si elegimos un tipo de filtro *paso banda* se activarán las casillas asignadas a las características propias del mismo, como son las *frecuencias de corte inferior* y *superior*, la *frecuencia central*, etc (ver recuadro adjunto 5.5).

Figura 5.5: Parámetros de diseño de filtros paso banda.

Como vemos en la imagen, el usuario podrá introducir las *frecuencias de corte inferior* y *superior*.

A modo de recomendación, el software ejecutará una ventana con un mensaje, aconsejando que el valor de la frecuencia más alta sea como mínimo diez veces el de la más baja (un orden de magnitud mayor) para que los *filtros paso bajo* y *paso alto* con los que se construyen se solapen y garanticen una ganancia uniforme en toda la banda de paso del filtro.

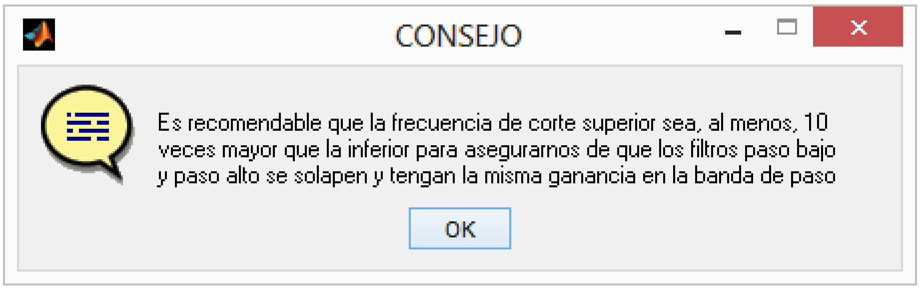


Figura 5.6: Recomendación acerca de la relación entre las frecuencias de corte inferior y superior en filtros paso banda.

Una vez insertadas ambas frecuencias, el programa diseñado calculará la *frecuencia central* (f_m = $\sqrt{f_{ci} * f_{cs}}$), el *ancho de banda* de los puntos de caída 3 *dB* (BW_{3dB} = f_{cs} - f_{ci}) y el *factor de calidad* del filtro (Q =$\frac{f_m}{BW_{3dB}}$), características todas ellas que fueron analizadas el capítulo 3.

De igual manera, la elección de un filtro *Chebyshev* ofrecerá al usuario la posibilidad de escoger el nivel de *rizado* que será de 0.5, 1, 2 o 3 dB, utilizando el menú creado a tal efecto.

Figura 5.7: Menú desplegable del nivel de rizado del filtro Chebyshev.

Así mismo, al ser el circuito *Sallen-Key* de ganancia unidad, sólo se podrá introducir el valor de *ganancia* si se decantó previamente por la estructura *Rauch* para la implementación.

Por otra parte especificar que consideramos innecesario, la simulación de filtros de *órdenes* superiores a 6 en nuestro proyecto, de ahí la limitación en cuanto al *orden* entre 1 y 6 para filtros *paso bajo* y *paso alto* y entre 2 y 6 para filtros *paso banda*.

La estrategia utilizada para la construcción de filtros de *órdenes* superiores a 2 fue la conexión en cascada de celdas básicas de 1º y 2º *orden* de filtros *paso bajo* y/o *paso alto* como ya explicamos en el capítulo 4 anterior.

En el caso de la configuración *paso banda* y aprovechando el trabajo realizado con antelación, los filtros fueron obtenidos a partir de la unión en cascada de células *paso bajo* y *paso alto*, circunstancia esta analizada en el capítulo previo.

No obstante y según el tipo de filtro que el usuario vaya seleccionando a través de los menús de la parte superior, aparecerán los circuitos de *Sallen-Key* y de *Rauch* (celdas básicas *paso bajo* y/o *paso alto* de primer y segundo *orden*) en los recuadros de la derecha.

Dichos esquemas permitirán conocer con mayor precisión cuales son las distribuciones más elementales con las que estamos trabajando y de cuyas combinaciones múltiples en serie resultará el montaje final de nuestro filtro.

Por último, indicar que esta ventana inicial de diseño se completa con dos botones en la parte inferior derecha. Uno de ellos es el botón "**Resetear"** que como su propio nombre índica, ofrece al usuario la posibilidad de borrar en cualquier momento todos los datos introducidos con anterioridad en las casillas habilitadas a tal efecto.

Esta opción de "*Reset*" se planteará también en otras ocasiones a lo largo del programa, cumpliendo en todos los casos la misma finalidad.

En lo que respecta al botón "**Seleccionar**", una vez presionado, genera una nueva pantalla en la que se determinarán los valores de los componentes del filtro completo, así como su *función de transferencia* **H(s)**, gracias a la cual obtendremos la representación gráfica del *diagrama de Bode* resultante, objetivo final de nuestra aplicación.

Por tanto, al pinchar sobre el botón anterior, afrontaremos una nueva fase de nuestro programa. Dependiendo del tipo de filtro elegido, se abrirán ventanas diferentes ya que el número de dispositivos varía de unos montajes a otros.

➢ *Filtros paso bajo de Sallen-Key*

Si optamos por un filtro de este tipo, se ejecutará la función **Menu_etapas_SK_Basico.m** que volcará la ventana de la figura 5.8 sobre la pantalla:

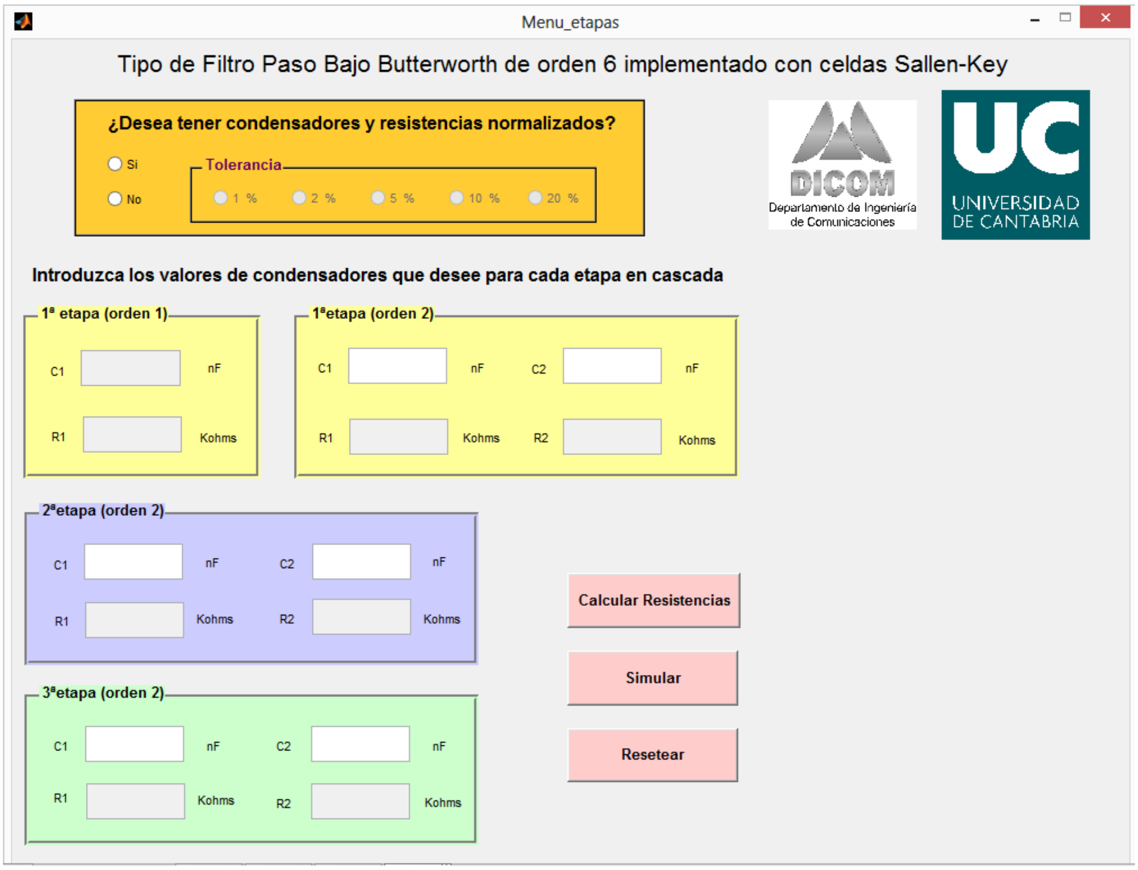


Figura 5.8: Pantalla de simulación y cálculo de los componentes del filtro paso bajo y paso alto de Sallen-Key.

Según observamos en la parte superior, el programa ofrece al usuario la posibilidad de trabajar con valores normalizados de resistencias y condensadores.

En caso afirmativo, podrá escoger además la *tolerancia* deseada que será del 1, 2, 5, 10 o 20%, apareciendo así mismo una nueva ventana con una tabla normalizada de capacidades, que será ejecutada a partir de la función **Tabla_Normaliz_Condensadores.m** y que servirá como referencia para el usuario a la hora de introducir los valores que considere más oportunos.

Tabla Normalizada de Condensadores

CONDENSADORES			
0.82 pf	47 pf	2.7 nf	180 nf
1 pf	56 pf	3.3 nf	220 nf
1.2 pf	68 pf	3.9 nf	330 nf
1.5 pf	82 pf	4.7 nf	470 nf
1.8 pf	100 pf	5.6 nf	560 nf
2.2 pf	120 pf	6.8 nf	680 nf
2.7 pf	150 pf	8.2 nf	820 nf
3.3 pf	180 pf	10 nf	1 µf
3.9 pf	220 pf	15 nf	1.2 µf
4.7 pf	270 pf	18 nf	1.5 µf
5.6 pf	330 pf	22 nf	1.8 µf
6.8 pf	390 pf	27 nf	2.2 µf
8.2 pf	470 pf	33 nf	2.7 µf
10 pf	560 pf	39 nf	3.3 µf
12 pf	680 pf	47 nf	3.9 µf
15 pf	820 pf	56 nf	4.7 µf
18 pf	1000 pf	68 nf	5.6 µf
22 pf	1.2 nf	82 nf	6.8 µf
27 pf	1.5 nf	100 nf	8.2 µf
33 pf	1.8 nf	120 nf	
39 pf	2.2 nf	150 nf	

Figura 5.9: Tabla normalizada de condensadores.

Por otro lado, en función del *orden* del filtro se habilitarán o no cada una de las etapas (1º, 2º y 3º) que conectadas en cascada, sean necesarias para su implementación.

Así por ejemplo, en filtros de *orden impar* se activará la célula de 1º *orden* que marcará el inicio de toda la estructura posterior, mientras que si el *orden* es *par*, la que se accionará será la primera de las etapas de 2º *orden* y así sucesivamente.

Los filtros de 3º y 4º *orden* requerirán de 2 etapas en serie, precisando los más grandes de orden 5 y 6 la conexión de 3 celdas básicas diferentes.

En el caso de las etapas de 2º *orden* de filtros *paso bajo Sallen-Key*, los condensadores deben cumplir como ya sabemos, una determinada relación (ecuación 4.6 del tema precedente).

Cuando el usuario escribe el dato del condensador C1, el programa ejecuta una ventana emergente informando acerca del valor mínimo que debe tener el segundo de ellos C2.

Este mensaje aparecerá para cada etapa de 2º *orden*, puesto que la relación entre las capacidades varía entre una y otra célula.

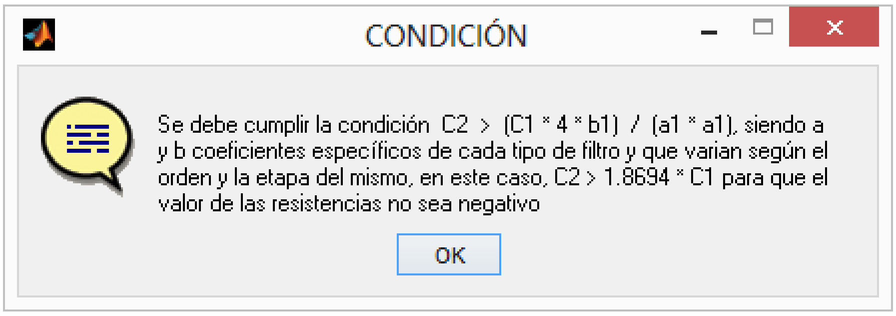


Figura 5.10: Condición impuesta a los condensadores de una determinada etapa de segundo orden en filtros paso bajo Sallen-Key.

Los coeficientes están registrados en unas tablas que se incluyen en el *anexo* de este proyecto como ya se comentó. Las funciones **Coef_Butt.m, Coef_Cheby_05.m, Coef_Cheby_1.m, Coef_Cheby_2.m, Coef_Cheby_3.m, Coef_Bessel.m** y **Coef_a1_b1** son las encargadas de recoger todos estos valores y devolverlos en cada llamada del programa principal.

Una vez introducido el valor de todas las capacidades correspondientes a las diferentes etapas que intervienen en la implementación del filtro y cumplida la relación que debe existir entre ellos, obtendremos las resistencias oportunas.

Para ello, haremos click en el botón **"Calcular Resistencias"** habilitado a tal efecto. Esta acción iniciará el proceso de cálculo interno de las mismas por parte del programa y el posterior volcado de los resultados en las casillas respectivas.

Si el filtro comienza con una distribución de 1º *orden* (filtro de *orden* impar), el valor de la resistencia R1 responde a la expresión 4.2 planteada en el capítulo previo.

Así mismo, el cálculo de las resistencias R1 y R2 en las celdas básicas de 2º *orden* se realizó en base a las ecuaciones 4.4 y 4.5 definidas igualmente en el tema anterior.

Por otra parte, si los resultados obtenidos para las resistencias son muy grandes (del orden de MΩ), el software emitirá un mensaje de error y reseteará el valor de los condensadores, puesto que no tiene sentido trabajar con registros tan altos en nuestro proyecto.

Teniendo en cuenta esta excepción, los datos extraídos de las resistencias serán normalizados o no (dependerá de la decisión del usuario) en función de la tolerancia seleccionada, siendo en cualquier caso presentados por pantalla en las casillas asignadas al respecto.

La función **Normaliz_Resist.m** recibe el valor resultante sin normalizar, y junto a las rutinas internas que utiliza **Resist_Normaliz_serie_1K_T1.m**, **Resist_Normaliz_serie_1K_T2.m**, incluso **Resist_Normaliz_serie_1K_T5.m**, **Resist_Normaliz_serie_1K_T10.m** y **Resist_Normaliz_serie_1K_T20.m** lo que hace es devolver la resistencia ya normalizada con la tolerancia exigida.

Finalmente, una vez definidos todos los valores de los componentes que forman el filtro, obtendremos el *diagrama de Bode* del mismo, guardando si se desea, los resultados numéricos alcanzados.

Para ello, el usuario pinchará sobre el botón **"Simular"**, produciéndose en ese momento, una llamada a la función **Grafica_salida.m**, la cual recibe como dato la función de transferencia *H(s)* del filtro completo, resultante del producto de las funciones de transferencia de todas las etapas de 1º y/o 2º *orden* que intervienen.

En este sentido, se emplean las rutinas **TF_SK_1.m** y **TF_SK_2.m** que devuelven las funciones de transferencia deducidas en el capítulo 4 (expresiones 4.1 y 4.3) de los filtros *paso bajo Sallen-Key* de 1º y 2º *orden* respectivamente.

El archivo **Grafica_salida.m** generará una última ventana gráfica como la mostrada en la imagen:

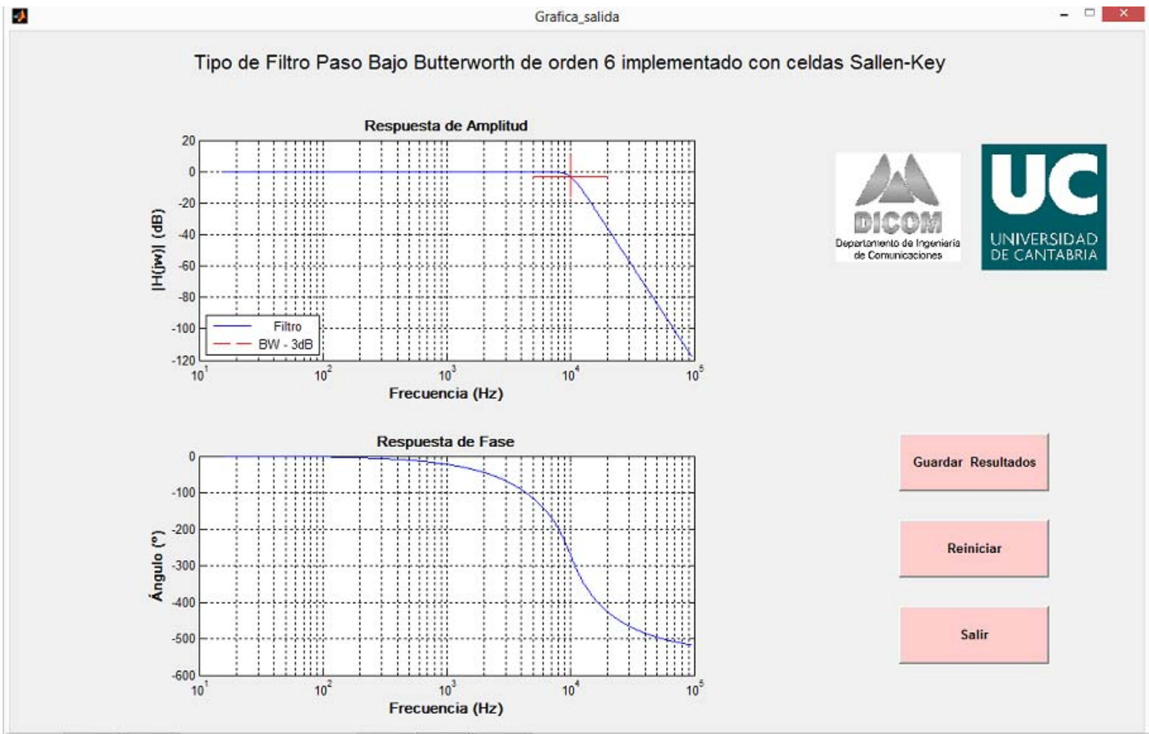
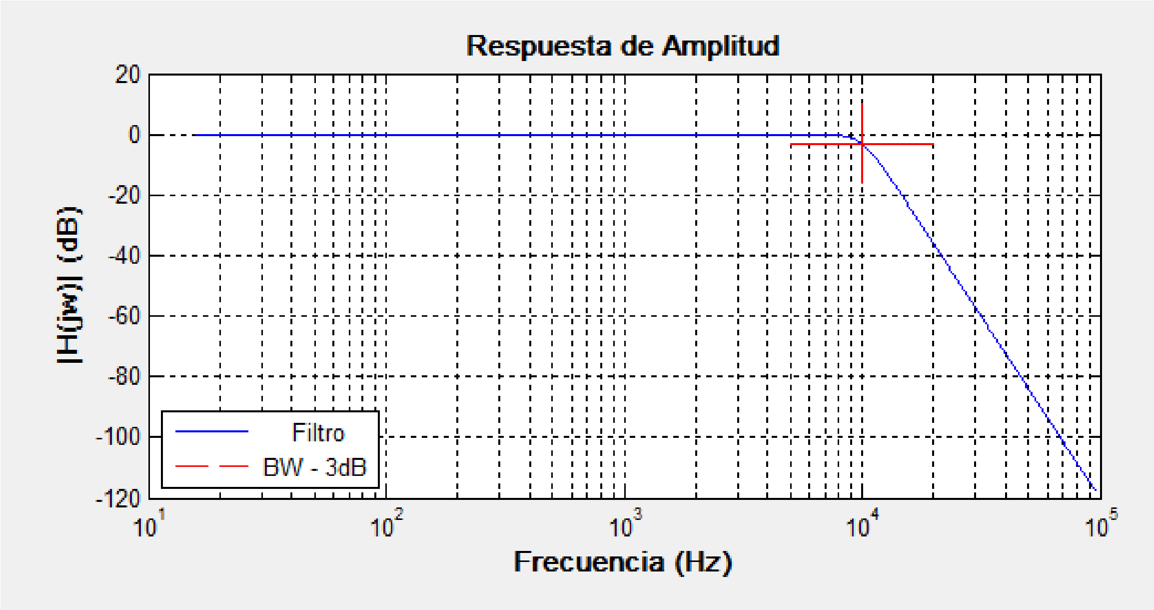


Figura 5.11: Representación gráfica del diagrama de Bode de filtros paso bajo Sallen-Key.

Ampliando un poco más las gráficas:



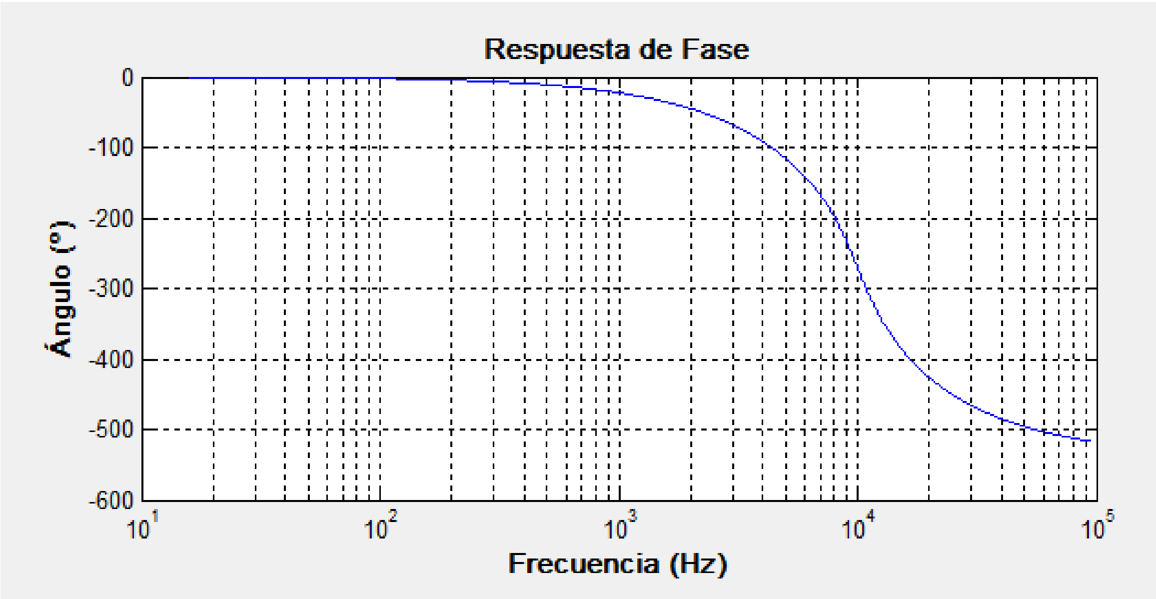


Figura 5.12: Vista ampliada del diagrama de Bode anterior.

En dichas figuras se reflejan la respuesta de amplitud y de fase del filtro correspondiente. La curva de *ganancia* se expresa en *decibelios (dB)* en función de la *frecuencia (Hz)*, mientras que la de *fase* se muestra en *grados (º).*

A modo orientativo y para facilitar una rápida identificación, el punto característico de caída de *ganancia* 3 *dB* respecto de su valor en la *banda de paso* (0 *dB* en este caso) que se producirá a la *frecuencia de corte* del filtro, aparece marcado con una cruz de color rojo.

Así mismo, el usuario podrá ampliar la visualización haciendo *zoom* en una zona determinada de la gráfica, si pincha en el botón izquierdo del ratón y lo arrastra sobre el área de interés.

Por último, la aplicación ofrece al usuario la posibilidad de guardar en un fichero, los resultados que se extraen de la representación gráfica.

Para ello, deberá pinchar en el botón **"Guardar"** habilitado al efecto. Esta acción derivará en una llamada a la función **Guardar_Filtro.m**, la cual provocará la aparición en primer plano de una ventana como la siguiente:

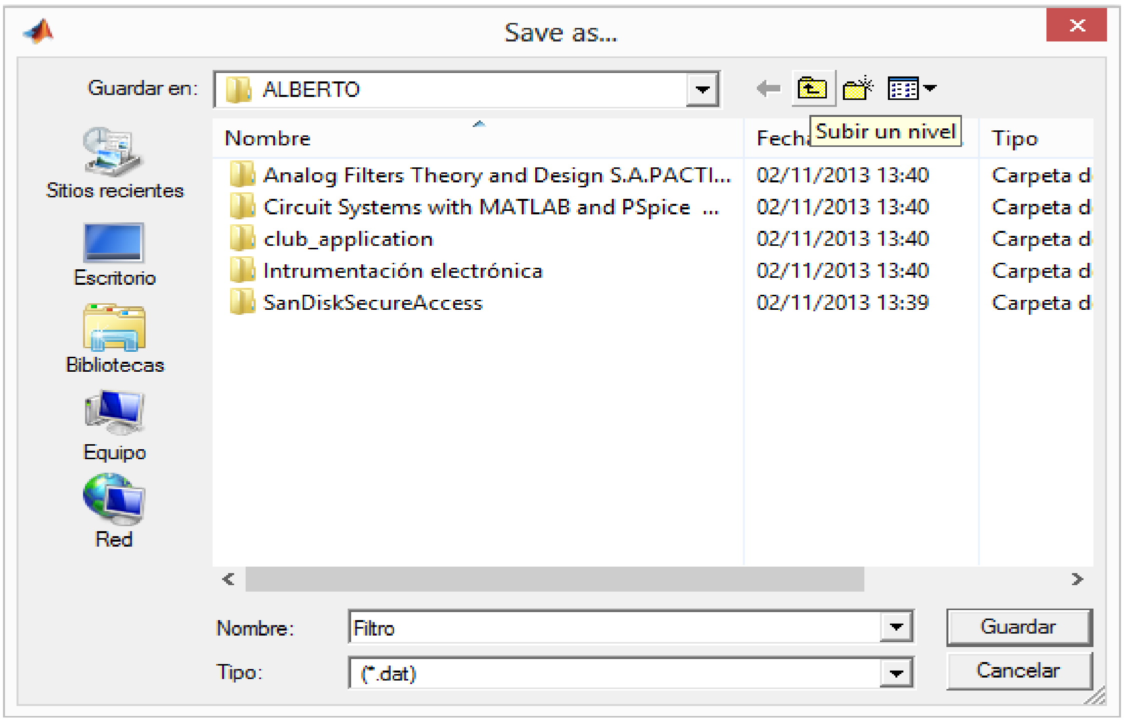


Figura 5.13: Selección del nombre del fichero que contiene los resultados del diagrama de Bode y de la ruta especificada para su ubicación.

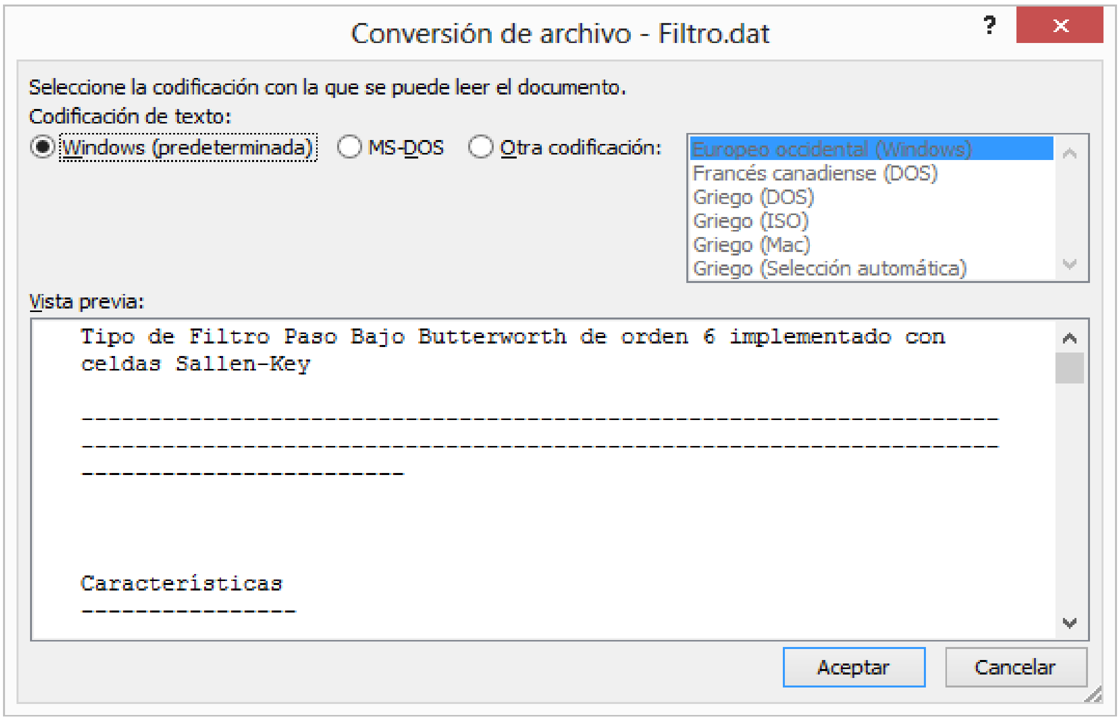


Figura 5.14: Opciones de codificación del fichero creado.

Establecido el nombre y la carpeta en la que será guardado por parte del usuario, emergerá una nueva pantalla (imagen 5.14) que permitirá elegir la codificación con la que poder leer el documento.

En este sentido, elegimos la opción de Windows predeterminada para nuestro proyecto y no profundizamos en otras posibles alternativas.

Además de los resultados (tabla numérica de *ganancia* y de *fase* a la *frecuencia* correspondiente), el archivo de extensión *.dat* en cuestión incluye el tipo de filtro diseñado con sus características más relevantes (aproximación empleada, *orden, frecuencia de corte*, etc) y también el valor de todos los componentes que lo integran (resistencias y capacidades) repartidos por etapas.

Terminado todo el proceso de diseño y simulación del filtro, el botón **"Reiniciar"** cierra todas las ventanas y nos devuelve a la pantalla del Menú inicial con el fin de poder modificar el tipo de filtro a desarrollar, mientras que el botón **"Salir"** deja abierto únicamente el editor de comandos de Matlab.

- *Filtros paso alto de Sallen-Key*

Si elegimos este tipo de filtros para nuestro diseño, aparecerá la ventana de simulación de la figura 5.8 anterior, igual que la generada para las estructuras *paso bajo Sallen-Key*, teniendo en cuenta obviamente que las celdas básicas con las que se trabaja son diferentes.

La única variación en cuanto al manejo de la pantalla, reside en la imposición de que los condensadores de una misma etapa deben ser iguales, informando al usuario de tal circunstancia a través del siguiente mensaje:

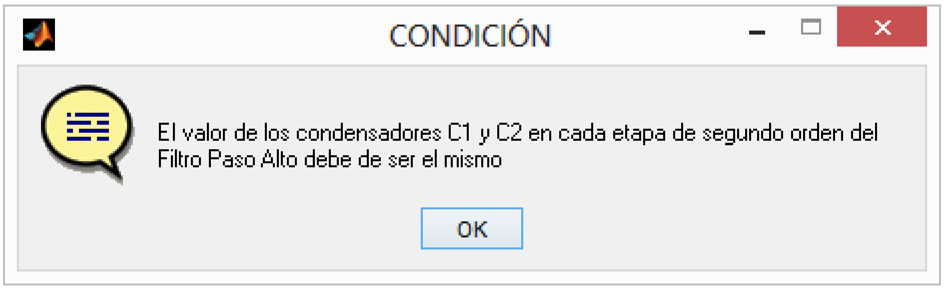


Figura 5.15: Condición requerida a los condensadores de una misma etapa de segundo orden en filtros paso alto Sallen-Key.

Establecido el valor de las capacidades al igual que para el filtro anterior, realizaremos el cálculo de las resistencias pinchando en el botón indicado.

De nuevo, en el supuesto de que el filtro sea de *orden* impar, este contará con una etapa inicial de 1º *orden*, cuyo valor de la resistencia R1 se corresponderá con la expresión 4.8 detallada en el capítulo que antecede.

Por lo que se refiere a las celdas básicas de 2º *orden* que van a componer el filtro completo, las expresiones de las resistencias R1 y R2 de cada etapa quedan recogidas en las ecuaciones 4.10 y 4.11 ya analizadas.

No obstante, damos por hecho que se cumple la condición de que los condensadores de una misma etapa en cada caso son iguales (C1=C2), de lo contrario el programa generaría un mensaje de error, advirtiendo de tal circunstancia.

De la misma manera que para los filtros *paso bajo Sallen-Key*, obtendremos valores normalizados de resistencias si el usuario así lo requiere, asegurándonos también de que los resultados alcanzados no representan tener que manejar componentes resistivos excesivamente grandes (del orden de los MΩ como ya se comentó).

Y el proceso de simulación y de obtención del *diagrama de Bode* del filtro junto a la posibilidad posterior de guardar los resultados, es idéntico en su ejecución también al descrito para el *paso bajo*.

➢ *<u>Filtros paso banda de Sallen-Key</u>*

En este supuesto, el programa principal realiza una llamada a la función **Menu_etapas_SK_BPF.m**, generando la ventana que se muestra:

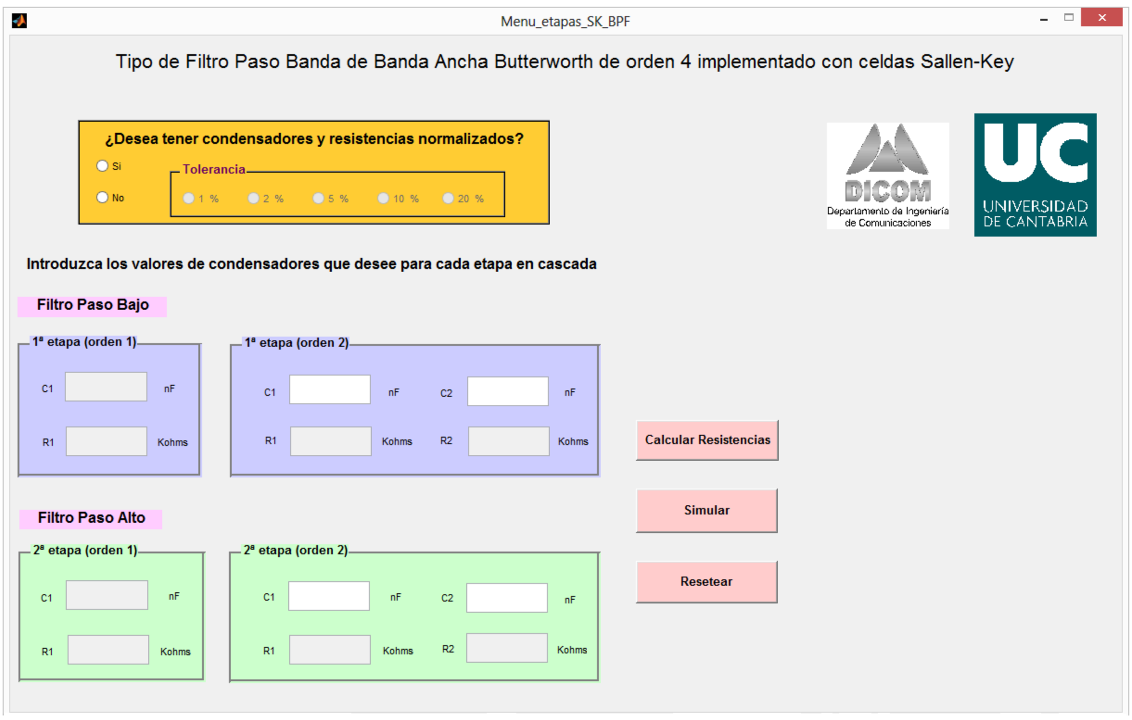


Figura 5.16: Pantalla de simulación y cálculo de los componentes del filtro paso banda de Sallen-Key.

Como puede observarse, el filtro *paso banda* es el resultado de la conexión en cascada de un *paso bajo* y un *paso alto*. En función del tamaño del filtro seleccionado, se habilitarán las estructuras básicas de 1º y/o 2º *orden* correspondientes.

Las combinaciones de ambos que dan lugar a los filtros *paso banda* de diferentes *órdenes* ya fueron descritas en el episodio precedente.

Las restricciones aplicadas a las capacidades que se deben introducir en etapas de 2º *orden*, tanto en configuraciones de *paso bajo* como de *paso alto* son las mismas que se plantearon en los supuestos anteriores, es decir, su relación dependerá de una serie de coeficientes en el primer caso mientras que en el segundo, deberán tener un valor idéntico.

En este sentido, el proceso de cálculo de las diferentes resistencias que conforman el filtro y las expresiones utilizadas para su obtención, es análogo a lo realizado en las configuraciones tanto *paso bajo* como *paso alto* de *Sallen - Key* (ecuaciones 4.2, 4.4, 4.5, 4.8, 4.10 y 4.11), sustituyendo la *frecuencia de corte* por la *frecuencia de corte superior* o *inferior*, en función de que las células sean *paso bajo* o *paso alto* respectivamente.

En cuanto a la simulación y representación posterior del *diagrama de Bode*, nada que añadir excepto que en la respuesta de amplitud aparecerán señalizadas no una, sino dos marcas rojas que coinciden con los puntos de caída de *ganancia* 3 *dB* a las *frecuencias de corte inferior y superior* propias del filtro *paso banda*.

- *Filtros paso bajo de Rauch*

Analizando esta configuración, el programa ejecuta la función **Menu_etapas_Rauch_LPF.m**, la cual da origen a la pantalla posterior:

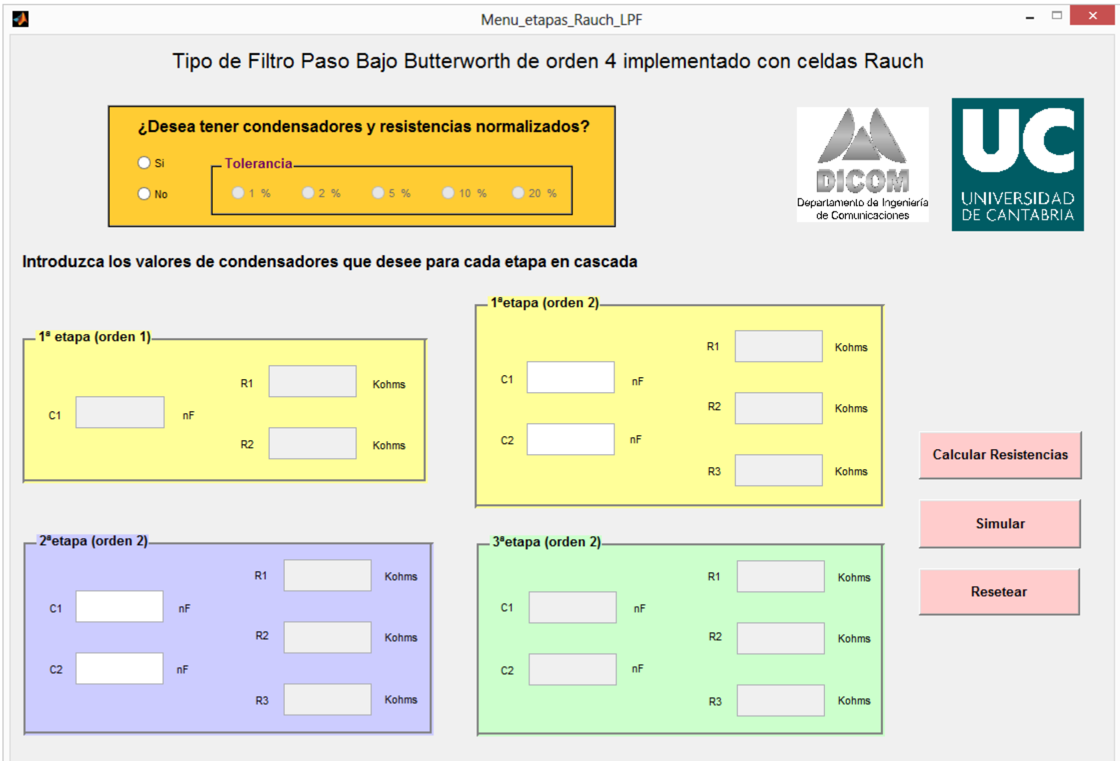


Figura 5.17: Pantalla de simulación y cálculo de los componentes del filtro paso bajo de Rauch.

Inspeccionando el aspecto de esta nueva estructura, vemos que las celdas básicas tanto de 1º como de 2º *orden* tienen una resistencia más en comparación con los supuestos analizados hasta el momento.

Otro factor diferencial reside en la posibilidad que existe por parte del usuario de introducir un valor de *ganancia G* determinado para el filtro que se pretende implementar.

Dicho parámetro modifica el aspecto final de las *funciones de transferencia* de las etapas empleadas y por extensión del filtro completo, tal y como vimos en el capítulo 4.

Como consecuencia de ello y con el objetivo de simplificar y de calcular los valores de los componentes del filtro en cuestión, también varían ligeramente las condiciones impuestas a los condensadores así como la expresión de las resistencias en celdas básicas de 2º *orden*.

En este sentido, habrá que tener en cuenta el número de etapas (1, 2 o 3) que componen el filtro completo como ya vimos en el tema anterior, puesto que repartiremos por igual la *ganancia* entre ellas.

Teniendo todo esto en cuenta, las condiciones exigidas a los condensadores en estructuras de 2º *orden* y diseñadas para no obtener posteriormente resistencias negativas, están asociadas a las ecuaciones 4.21, 4.22 y 4.23 descritas así mismo en el tema 4.

De nuevo, el programa emitirá un mensaje emergente, informando de las relaciones entre las capacidades en cada celda básica de segundo *orden*.

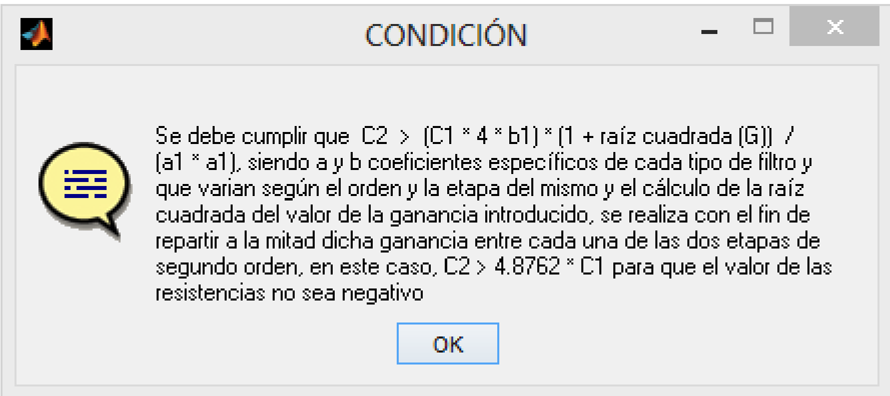


Figura 5.18: Condición impuesta a los condensadores de una etapa en filtros paso bajo Rauch de tercer o cuarto orden.

Definidas todas las capacidades del filtro completo, nos ocuparemos del cálculo de las resistencias haciendo uso del botón habilitado para ello.

Al igual que en el caso de los condensadores, será fundamental conocer el número de etapas que forman el filtro ya que las expresiones serán diferentes dependiendo del reparto de la *ganancia* que tengamos que hacer entre dichas etapas.

Así, las ecuaciones que manejamos para obtener los valores de R1 y R2 de esa primera etapa de primer *orden* se corresponden con las expresiones 4.13, 4.14, 4.15 y 4.16 definidas igualmente en el capítulo anterior.

De la misma manera, el cálculo de las resistencias R1, R2 y R3 de las etapas de segundo *orden* de las que consta el filtro se relaciona con las igualdades 4.18, 4.19 y 4.20 precedentes.

Mientras que para filtros de 3° y 4° *orden*, las ecuaciones son las mismas sustituyendo la *ganancia* *G* por $\sqrt{G}$ y en filtros de 5° y 6° *orden*, se intercambia el dato de la *ganancia* *G* por $\sqrt[3]{G}$ por una cuestión de reparto equitativo de la *ganancia* entre todas las etapas.

La presunta eventualidad de obtener valores resistivos demasiado elevados fue junto al proceso de normalización, previsto y solucionado de igual manera que para los filtros *Sallen-Key*.

Y el resto del manejo de la aplicación ya es conocido. Esta vez eso sí, tanto en el *diagrama de Bode* resultante como en el momento si procede de guardar los resultados (se agregará un nuevo parámetro), habrá que tener en cuenta que la *ganancia* ya no es la unidad sino la que el usuario introduce en la fase de diseño.

➢ *Filtros paso alto de Rauch*

Si esta fue nuestra selección, el software activa la llamada a la función **Menu_etapas_Rauch_HPF.m**, creando dicha rutina la ventana consiguiente:

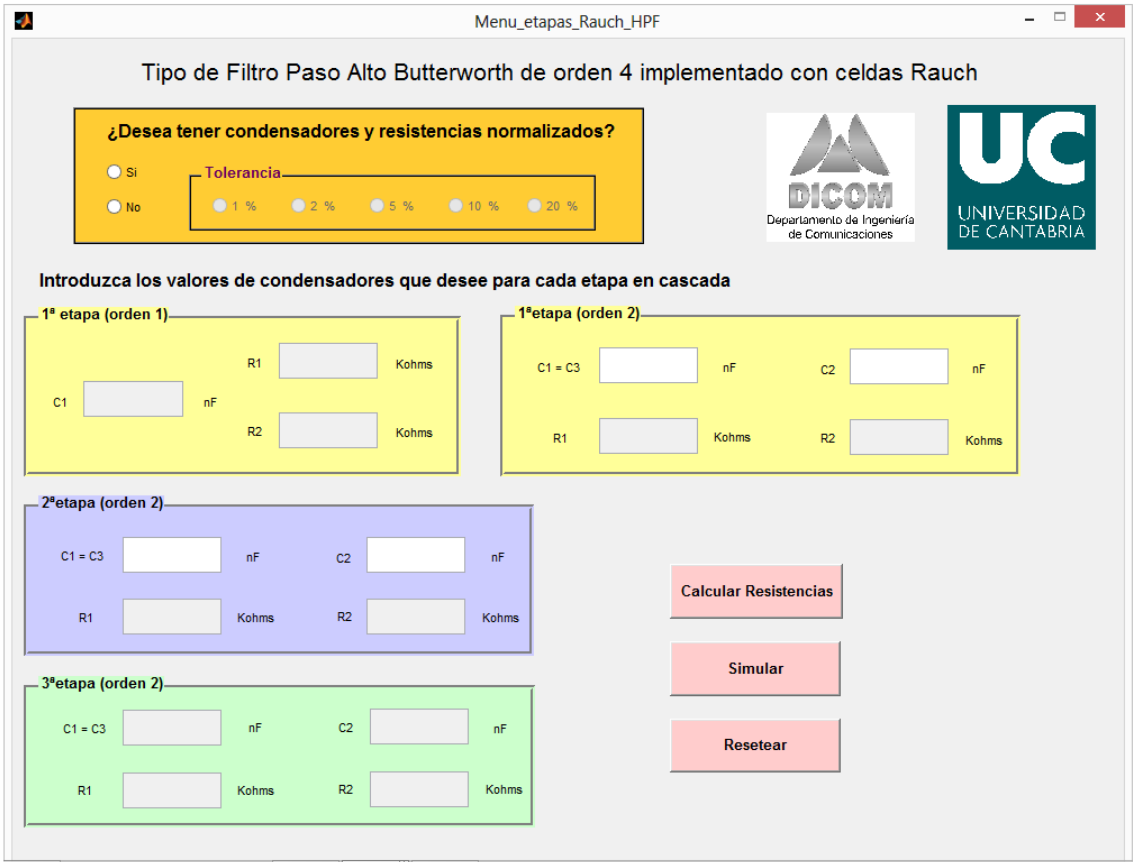


Figura 5.19: Pantalla de simulación y cálculo de los componentes del filtro paso alto de Rauch.

Al contrario que en los filtros *paso bajo* de *Rauch*, vemos que las etapas de 2º *orden* están formadas por 3 capacidades (dos de las cuales tienen el mismo valor) y 2 resistencias.

Para simplificar los cálculos, como ya hemos comentado establecimos la condición de que 2 de los condensadores fueran iguales (C1=C3=C), siendo incorporado a la aplicación este valor C por parte del usuario.

Conocido este dato C, calculamos el valor de la capacidad C2 a partir de las expresiones 4.30, 4.31 y 4.32 detalladas previamente en el capítulo 4. El razonamiento en cuanto a la distribución igualitaria de la *ganancia* entre las diferentes etapas, es el mismo al planteado para el caso *paso bajo* de *Rauch*.

En lo que respecta a la forma de obtener el valor de las resistencias, el procedimiento es de nuevo, análogo al filtro anterior.

Para etapas de primer *orden* (filtros de *orden* impar 1, 3 y 5), las expresiones asociadas a las resistencias R1 y R2 corresponden a las ecuaciones 4.25, 4.26, 4.27 y 4.28 analizadas con anterioridad.

Mientras que los valores resistivos R1 y R2 en las etapas de segundo *orden*, responden a los criterios marcados en las igualdades 4.33 y 4.34 estudiadas también en el tema 4.

Obtenidas las resistencias e introducidas las capacidades, se realiza la simulación del filtro y se obtiene la representación gráfica del *diagrama de Bode* de la misma manera que para el supuesto anterior *paso bajo* de *Rauch*.

- *Filtros paso banda de Rauch*

Si por el contrario el usuario se decantó por esta clase de filtros, la aplicación efectúa una llamada a la función **Menu_etapas_Rauch_BPF.m**, que a su vez produce la ventana aquí representada:

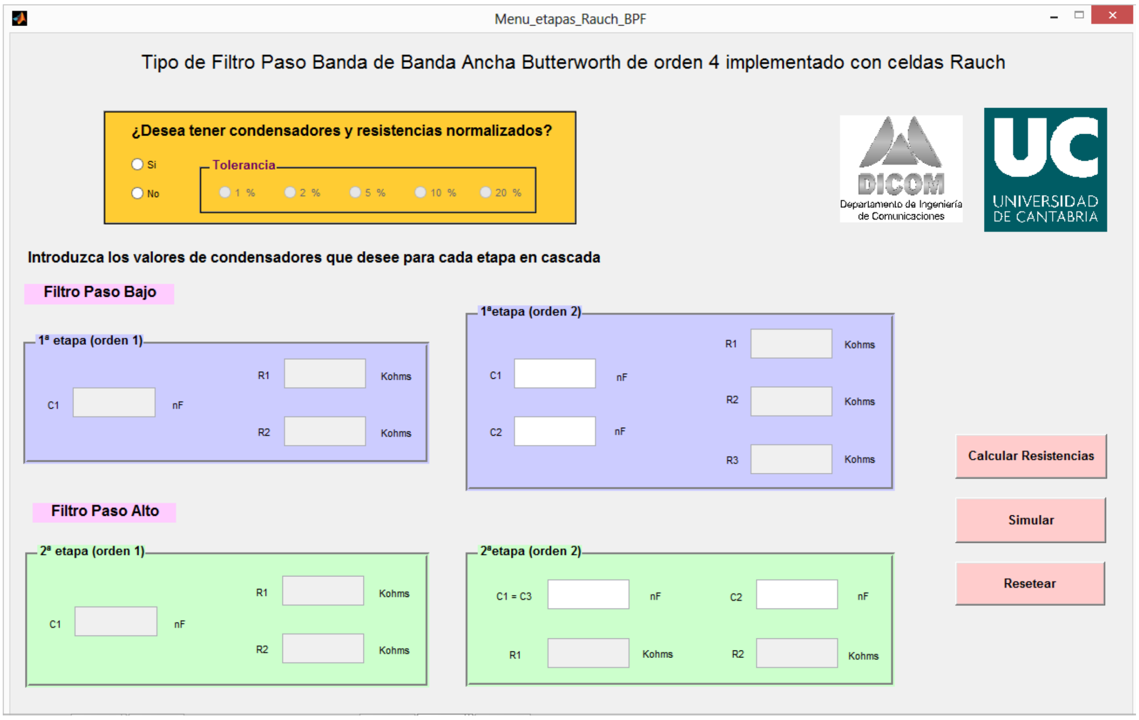


Figura 5.20: Pantalla de simulación y cálculo de los componentes del filtro paso banda de Rauch.

De la misma manera que para el caso de *Sallen-Key*, el filtro *paso banda* se obtiene a partir de la conexión en cascada de un *paso bajo* y un *paso alto* de *Rauch*.

Las condiciones respecto a las capacidades que se deben introducir en etapas de 2º *orden*, tanto en estructuras de *paso bajo* como de *paso alto* son las mismas que las utilizadas en los supuestos de *Rauch* previos.

Es decir, su relación quedará establecida en función de una serie de coeficientes y del porcentaje de la *ganancia* que se repartirá en cada una de las etapas en el primer caso (expresiones 4.21, 4.22 y 4.23), mientras que en el segundo, C1=C3=C y las ecuaciones empleadas 4.30, 4.31 y 4.32.

Por idéntica razón, el proceso de cálculo de las resistencias y las expresiones utilizadas para su obtención, es análogo a lo realizado en las configuraciones anteriores *paso bajo* (ecuaciones 4.13, 4.14, 4.15, 4.16, 4.18, 4.19 y 4.20) y *paso alto* de *Rauch* (igualdades 4.25, 4.26, 4.27, 4.28, 4.33 y 4.34), intercambiando la *frecuencia de corte* por la *frecuencia de corte superior* o *inferior*, en función de que las celdas sean *paso bajo* o *paso alto* respectivamente.

El único matiz diferente tanto para la condición impuesta a los condensadores como para el cálculo de las resistencias reside en el caso del filtro *paso banda* de sexto *orden*. Dicho filtro está formado por la conexión en cascada de 4 etapas básicas (dos de *paso bajo* y dos de *paso alto* de primer y segundo *orden* cada una de ellas).

Por tanto, si repartimos la *ganancia G* por igual entre las 4 etapas, el valor que tendrá en cada celda será de $\sqrt[4]{G}$, modificación que aplicaremos en las expresiones respectivas anteriormente indicadas para etapas de primer y segundo *orden* de *paso bajo* y *paso alto*.

En este punto, tiene lugar la simulación del filtro completo con el volcado del *diagrama de Bode* subsiguiente, existiendo esta vez dos puntos de caída 3 *dB* respecto de la *ganancia* en la banda de paso que coinciden con las *frecuencias de corte inferior y superior* ya mencionadas.

CAPÍTULO VI

EJEMPLOS DE FILTROS ACTIVOS EN MATLAB Y SPICE

En este capítulo realizaremos la simulación de diferentes ejemplos de filtros de *Sallen-Key* y de *Rauch*, obteniendo su *diagrama de Bode* correspondiente. Para ello utilizaremos el software desarrollado en MATLAB y expuesto ampliamente en la sección anterior.

Así mismo, con el objetivo de validar los resultados aportados a través de nuestra aplicación y de esta manera poder comprobar su correcto funcionamiento, consideramos conveniente efectuar la simulación del mismo filtro en el programa PSPICE (*Personal Simulation Program with Integrated Circuits Emphasis*).

PSPICE es un estándar internacional cuya finalidad consiste en simular circuitos electrónicos analógicos formados por resistencias, condensadores, diodos, transistores, amplificadores operacionales, etc.

En primer lugar hay que describir el circuito con el valor de cada componente que lo integra y luego elegir el tipo de simulación deseada (temporal, en frecuencia, en continua, paramétrico, Montecarlo, etc).

Los ficheros *.cir* generados por PSPICE en la realización de los filtros serán expuestos y comentados para cada uno de los ejemplos que se plantean, pudiendo contrastar los *diagramas de Bode* obtenidos a partir de dichos ficheros con los alcanzados a través de nuestro software en MATLAB.

6.1. – Filtro paso bajo Butterworth de 2º orden implementado con celdas de Sallen-Key

El circuito asociado a este filtro es el siguiente:

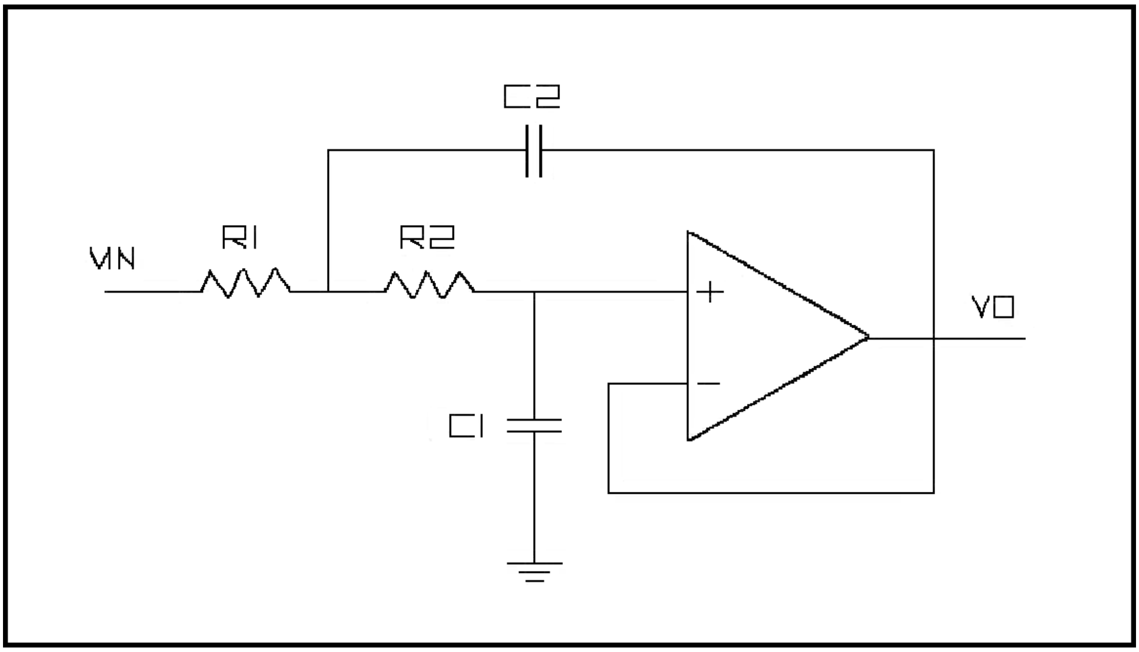


Figura 6.1: Filtro paso bajo Butterworth de 2º orden implementado con celdas de Sallen-Key de ganancia unitaria.

Características

Orden = 2 Frecuencia de Corte = 1.0000 KHz

Componentes

C1 = 10.0000 nF C2 = 22.0000 nF

R1 = 8.2000 KΩ R2 = 15.0000 KΩ

Aplicando la expresión 4.3 y sustituyendo el valor de sus componentes, la función de transferencia obtenida en Matlab responde a:

$$H(s) = \frac{1}{2.706 \cdot 10^{-8}\, s^2 + 2.32 \cdot 10^{-4}\, s + 1} \qquad (6.1)$$

Mientras que el diagrama de Bode resultante es el que se muestra a continuación:

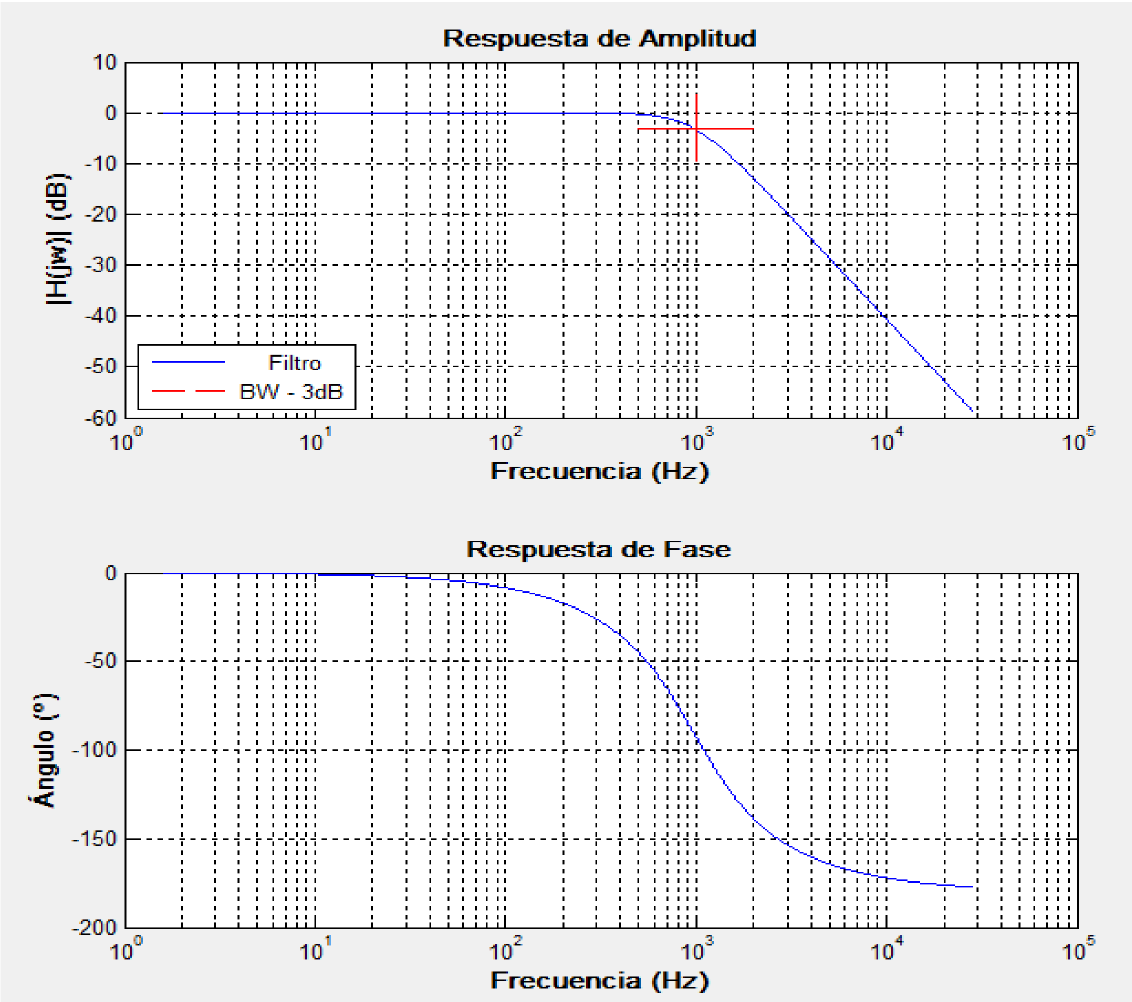


Figura 6.2: Diagrama de Bode del filtro paso bajo Butterworth de 2º orden implementado con celdas Sallen-Key obtenido en Matlab.

Por otra parte, para la simulación del filtro en Pspice realizamos el fichero *.cir* descrito más abajo:

```
VAC  1  0  AC 1
R1   1  2  8200
R2   2  3  15000
C1   3  0  10E-9
C2   2  4  22E-9
xop1  3  4  5  6  4  TL082
VDCp  5  0  dc  12
VDCn  6  0  dc  -12
raux  4  0  1e10

.ac  lin  10001  1  1e5
.probe

* Entrada nudo  1
* Salida nudo  4
* Dibujar Gráfica de ganancia  20*log10(V(4)/V(1))  en  dB  y  de fase  VP(4)

* SUBCIRCUITO OPERACIONAL TL082
*-------------------------------------------------------------------------
.subckt TL082    1 2 3 4 5
*
  c1   11  12    2.412E-12
```

```
c2   6  7  18.00E-12
css  10  99  5.400E-12
dc   5  53  dx
de  54  5  dx
dlp  90  91  dx
dln  92  90  dx
dp   4  3  dx
egnd  99  0  poly(2) (3,0) (4,0) 0 .5 .5
fb   7  99  poly(5)  vb  vc  ve  vlp  vln  0  3.467E6  -3E6  3E6  3E6  -3E6
ga   6  0  11  12  339.3E-6
gcm  0  6  10  99  17.01E-9
iss  10  4  dc  234.0E-6
hlim  90  0  vlim  1K
j1  11  2 10  jx
j2  12  1 10  jx
r2   6  9  100.0E3
rd1  3  11  2.947E3
rd2  3  12  2.947E3
ro1  8  5  50
ro2  7  99  170
rp   3  4  20.00E3
rss  10  99  854.7E3
vb   9  0  dc  0
vc   3  53  dc  1.500
```

```
ve   54  4  dc  1.500
vlim  7  8  dc  0
vlp  91  0  dc  50
vln   0  92  dc  50
.model  dx  D(Is=800.0E-18 Rs=1)
.model  jx  NJF(Is=2.500E-12 Beta=984.2E-6 Vto=-1)
.ends


.end
```

Figura 6.3: Fichero .cir en Pspice que genera la simulación del filtro paso bajo de Butterworth de 2º orden implementado con celdas Sallen-Key.

De este fichero comentar que se efectúa el análisis en pequeña señal *.AC* con 10001 puntos de simulación hasta una frecuencia de 100 KHz.

Esta limitación frecuencial queda condicionada en gran medida por el rango de validez del amplificador operacional utilizado TL082 (definido a través del comando .subckt), el cual presenta un comportamiento no deseado si trabajamos a frecuencias altas, a diferencia de la simulación en Matlab donde consideramos un OPAMP ideal.

Para poder comparar con mayor precisión ambas simulaciones, el rango de frecuencias representadas en el *diagrama de Bode* será siempre el mismo en los dos casos.

Por otra parte, si queremos obtener la gráfica de ganancia en Pspice en términos de decibelios (dB) y de esta manera contrastar los resultados con los de Matlab, aplicaremos la ecuación correspondiente que se define como $20*\log_{10}$ (V (nudo de salida) / V (nudo de entrada)) mientras que si hablamos de la fase, la expresión introducida será VP (nudo de salida).

Por último, en cuanto al fichero *.cir* analizado, mencionar que fue añadida una resistencia auxiliar de valor muy alto (10^{10} Ω) conectada a tierra a la salida del circuito, lo que a efectos prácticos se puede considerar como un circuito abierto pero que evita en su caso, dejar el montaje al aire, circunstancia esta que fue tenida en cuenta en todos los ejemplos realizados.

El resultado de la simulación en Pspice incluidos los condicionantes anteriores es el que corresponde a la gráfica 6.4:

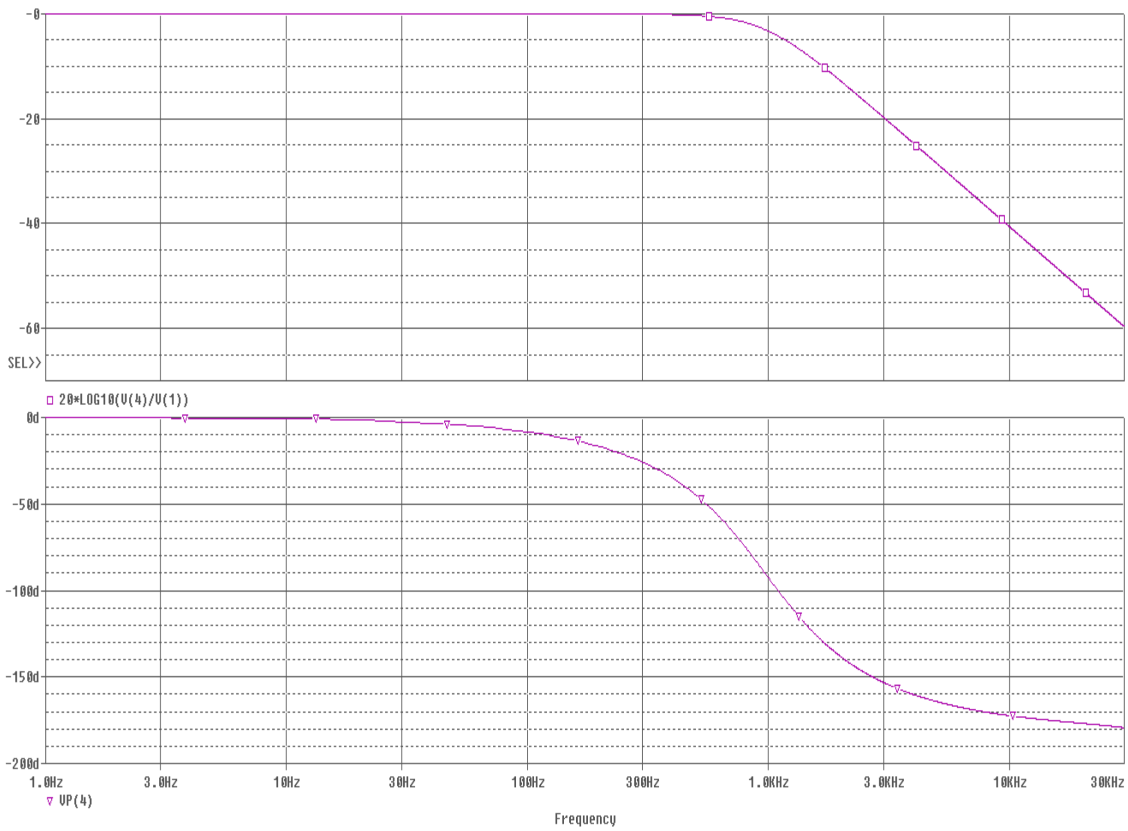


Figura 6.4: Diagrama de Bode del filtro paso bajo Butterworth de 2º orden implementado con celdas Sallen-Key obtenido en Pspice.

6.2. – Filtro paso alto Chebyshev de 3º orden con rizado 2 dB implementado con celdas de Sallen-Key

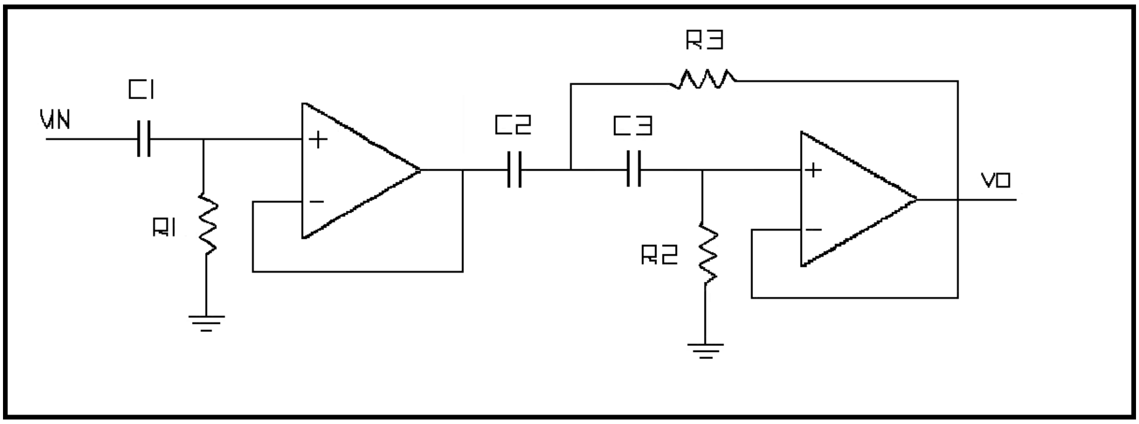


Figura 6.5: Filtro paso alto Chebyshev de 3º orden con rizado 2 dB implementado con celdas de Sallen-Key de ganancia unitaria.

Características

Orden = 3   Frecuencia de Corte = 10.0000 KHz

Componentes

Primera etapa (Primer Orden)

C1 = 10.0000 nF   R1 = 0.5620 KΩ

Segunda etapa (Segundo Orden)

C2 = 22.0000 nF   C3 = 22.0000 nF   R2 = 3.4000 KΩ   R3 = 0.1300 KΩ

En este caso, la función de transferencia $H(s)$ del filtro completo obtenida en Matlab es el producto de las funciones de transferencia $H_1(s)$ y $H_2(s)$ de las dos etapas de 1º y 2º *orden* que lo conforman:

$$H_1(s) = \frac{5.62 \cdot 10^{-6}\, s}{5.62 \cdot 10^{-6}\, s + 1} \tag{6.2}$$

$$H_2(s) = \frac{2.139 \cdot 10^{-10}\, s^2}{2.139 \cdot 10^{-10}\, s^2 + 5.72 \cdot 10^{-6}\, s + 1} \tag{6.3}$$

Dividiendo arriba y abajo la primera expresión por $5.62 \cdot 10^{-6}$ y la segunda por $2.139 \cdot 10^{-10}$ y multiplicando ambas ecuaciones, la función de transferencia requerida $H(s) = H_1(s) * H_2(s)$ queda como sigue:

$$H(s) = \frac{s}{s + 177.936 \cdot 10^{3}} * \frac{s^2}{s^2 + 26.741 \cdot 10^{3}\, s + 4.68 \cdot 10^{10}} \tag{6.4}$$

El *diagrama de Bode* representado en Matlab a partir de la función de transferencia anterior, tiene el siguiente aspecto:

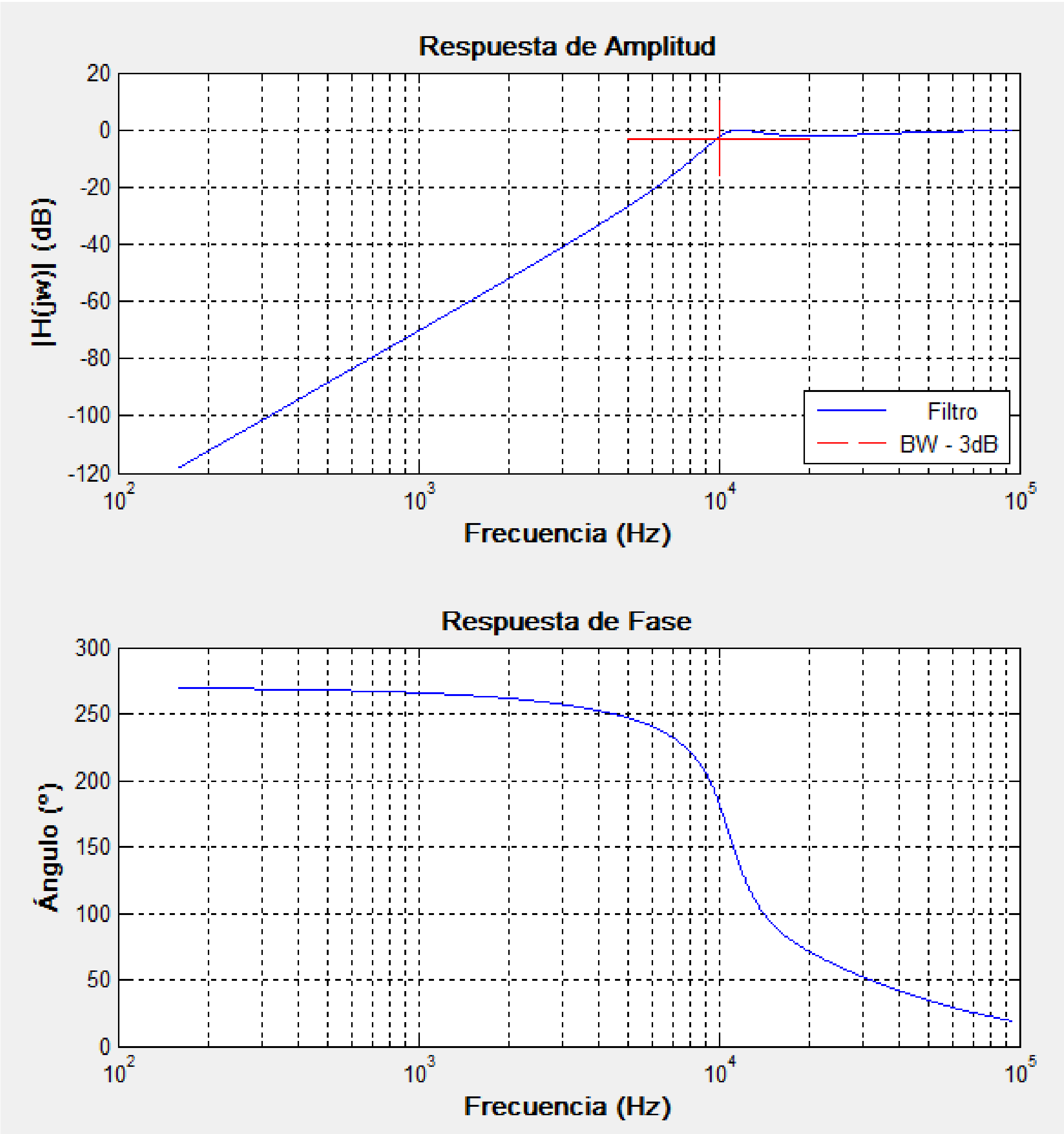


Figura 6.6: Diagrama de Bode del Filtro paso alto Chebyshev de 3º orden con rizado 2 dB implementado con celdas de Sallen-Key.

El fichero *.cir* elaborado en Pspice para la simulación de este filtro corresponde al código planteado en las siguientes líneas:

```
VAC  1  0  AC 1

C1   1  2  10E-9

R1   2  0  562

C2   3  6  22E-9

C3   6  7  22E-9

R2   7  0  3400

R3   6  8  130

xop1  2  3  4  5  3  TL082

VDCp   4  0  dc  12

VDCn   5  0  dc  -12

xop2  7  8  4  5  8  TL082

raux   8  0  1e10


.ac  lin  10001  1  1e5

.probe


* Entrada nudo 1

* Salida nudo 8

* Dibujar Gráfica de ganancia  20*log10(V(8)/V(1))  en dB  y  de  fase  VP(8)


* SUBCIRCUITO OPERACIONAL TL082
*--------------------------------------------------------------------------------
```

```
.subckt TL082    1 2 3 4 5
*

c1   11  12   2.412E-12

c2    6   7   18.00E-12

css  10  99   5.400E-12

dc    5  53   dx

de   54  5   dx

dlp  90  91  dx

dln  92  90  dx

dp    4  3  dx

egnd  99  0  poly(2) (3,0) (4,0) 0 .5 .5

fb    7  99  poly(5)  vb  vc  ve  vlp  vln  0 3.467E6  -3E6  3E6  3E6  -3E6

ga    6  0 11 12  339.3E-6

gcm   0  6 10  99  17.01E-9

iss  10  4  dc  234.0E-6

hlim  90  0  vlim  1K

j1   11  2  10  jx

j2   12  1  10  jx

r2    6   9  100.0E3

rd1   3  11   2.947E3

rd2   3  12   2.947E3

ro1   8  5  50

ro2   7  99  170
```

```
rp   3   4   20.00E3
rss   10  99   854.7E3
vb    9   0  dc  0
vc    3   53  dc  1.500
ve   54   4  dc   1.500
vlim   7   8  dc   0
vlp   91  0  dc  50
vln   0  92  dc  50
.model  dx  D(Is=800.0E-18 Rs=1)
.model  jx  NJF(Is=2.500E-12 Beta=984.2E-6 Vto=-1)
.ends
.end
```

Figura 6.7: Fichero .cir en Pspice que genera la simulación del filtro paso alto Chebyshev de 3º orden con rizado 2 dB implementado con celdas Sallen-Key.

De nuevo, de este fichero comentar la utilización del operacional TL082 que limitará un tanto el rango de validez de la simulación a frecuencias altas y por lo demás, los aspectos a considerar (análisis *.AC* para pequeña señal, entrada en el nudo 1 y salida en el 8, definición del subcircuito del OPAMP, resistencia auxiliar a la salida, etc) son los mismos que se tuvieron en cuenta para la realización del filtro anterior.

Compilando el archivo mostrado en Pspice, el *diagrama de Bode* resultante es el asociado a la siguiente imagen:

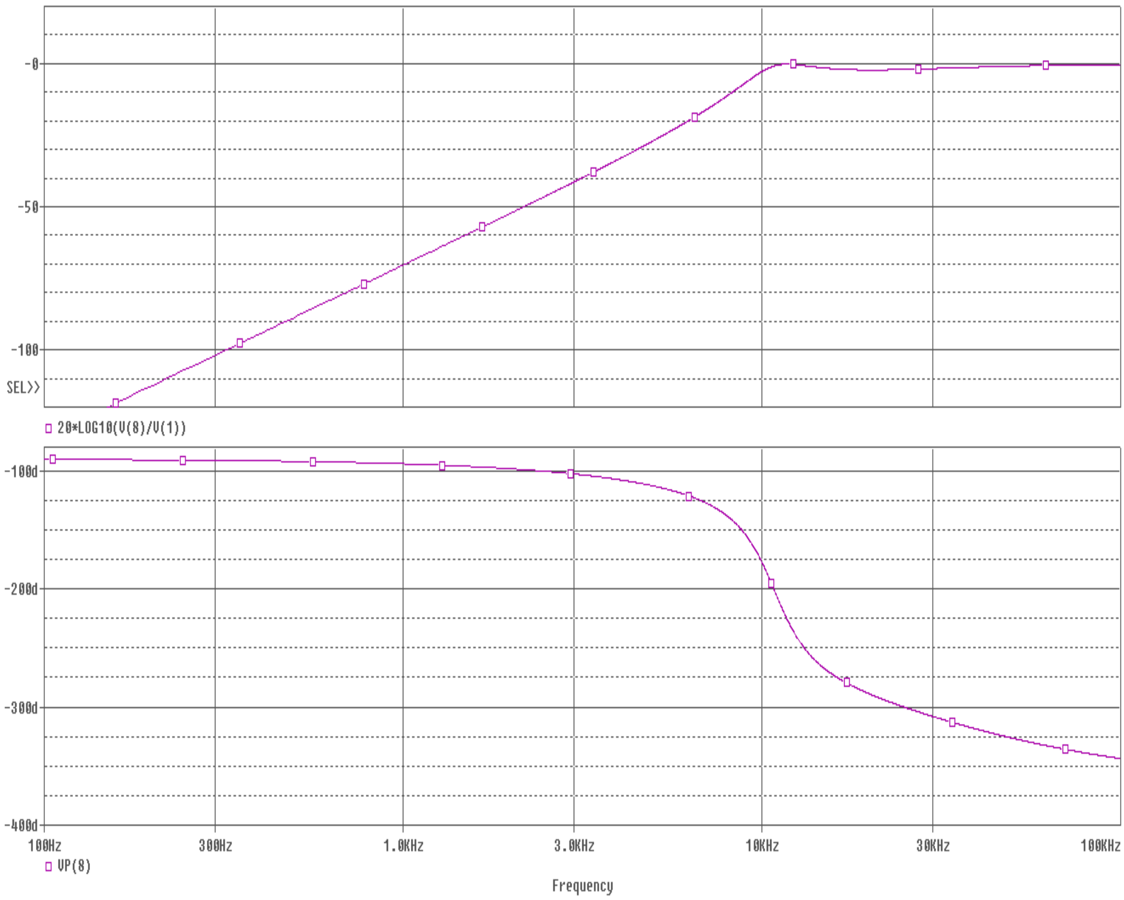


Figura 6.8: Diagrama de Bode del filtro paso alto Chebyshev de 3º orden con rizado 2 dB implementado con celdas Sallen-Key obtenido en Pspice.

6.3 – Filtro paso banda Bessel de 5º orden implementado con celdas de Sallen-Key

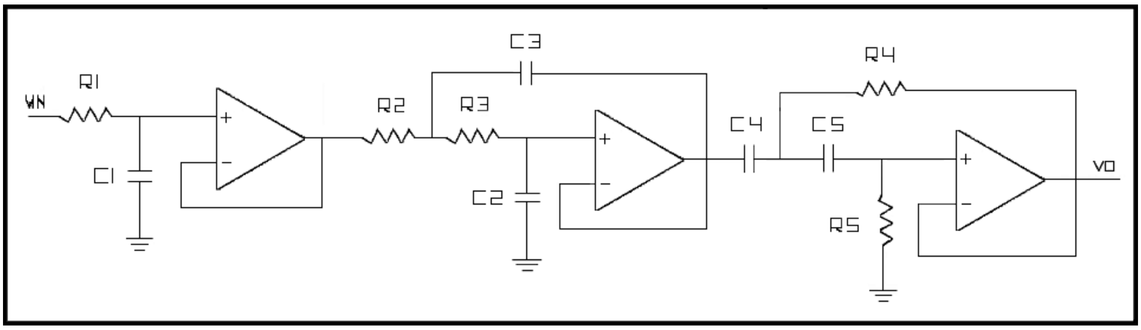


Figura 6.9: Filtro paso banda Bessel de 5º orden implementado con celdas de Sallen-Key de ganancia unitaria.

Características

Orden = 5 Frec. Corte Inferior = 1.0000 KHz

Frec. Corte Superior = 10.0000 KHz Frecuencia Central = 3.1623 KHz

Ancho de Banda = 9.0000 KHz Factor de Calidad: 0.3514

Componentes

Etapa Paso Bajo (Primer Orden)

C1 = 56.0000 nF R1 = 0.2150 KΩ

Etapa Paso Bajo (Segundo Orden)

C2 = 10.0000 nF   C3 = 22.0000 nF   R2 = 0.5110 KΩ   R3 = 1.0700 KΩ

Etapa Paso Alto (Segundo Orden)

C4 = 33.0000 nF   C5 = 33.0000 nF   R4 = 7.1500 KΩ   R5 = 5.3600 KΩ

Inspeccionando el circuito, vemos que está formado por la conexión en cascada de dos etapas paso bajo de 1º y 2º *orden* respectivamente y una tercera etapa paso alto de 2º *orden*.

Por lo tanto, la función de transferencia $H(s)$ del filtro paso banda será el producto de las funciones de transferencia $H_1(s)$, $H_2(s)$ y $H_3(s)$ de cada una de las etapas individuales:

$$H_1(s) = \frac{1}{1.204 \cdot 10^{-5}\ s + 1} \tag{6.5}$$

$$H_2(s) = \frac{1}{1.203 \cdot 10^{-10}\ s^2 + 1.581 \cdot 10^{-5}\ s + 1} \tag{6.6}$$

$$H_3(s) = \frac{4.173 \cdot 10^{-8}\, s^2}{4.173 \cdot 10^{-8}\, s^2 + 3.538 \cdot 10^{-4}\, s + 1} \qquad (6.7)$$

Una vez más, si manipulamos la expresión anterior $H_3(s)$ dividiendo arriba y abajo por $4.173 \cdot 10^{-8}$ y multiplicando la ecuación resultante, llegaremos a la función de la trasferencia $H(s) = H_1(s) * H_2(s) * H_3(s)$ del filtro completo:

$$H(s) = \frac{1}{1.204 \cdot 10^{-5}\, s + 1} * \frac{1}{1.203 \cdot 10^{-10}\, s^2 + 1.581 \cdot 10^{-5}\, s + 1} * \longrightarrow$$

$$\longrightarrow * \frac{s^2}{s^2 + 8.478 \cdot 10^{3}\, s + 2.396 \cdot 10^{7}} \qquad (6.8)$$

Con el cálculo de la función de transferencia, Matlab simula el *diagrama de Bode* perteneciente al filtro paso banda en cuestión:

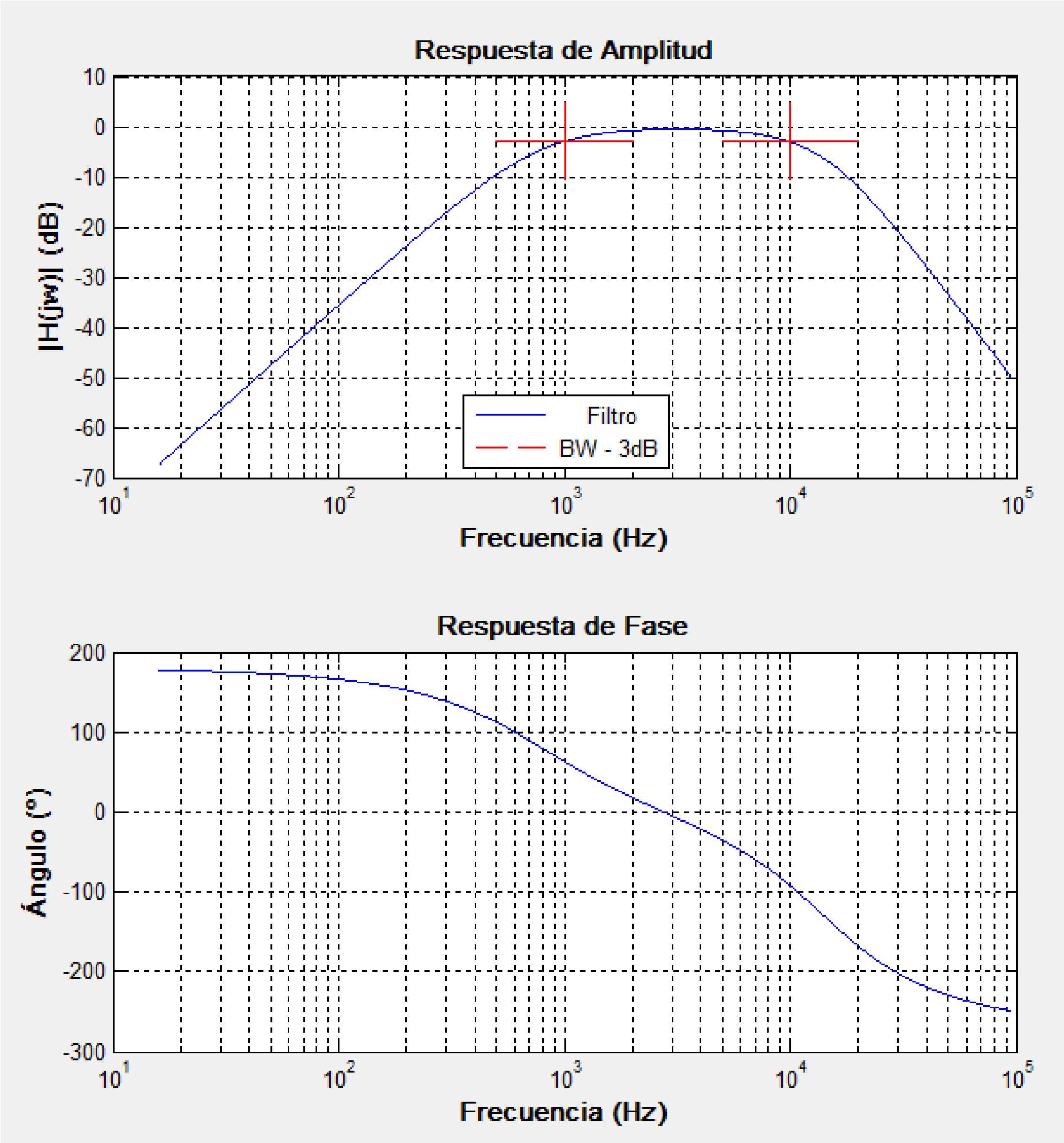


Figura 6.10: Diagrama de Bode del Filtro paso banda Bessel de 5º orden implementado con celdas de Sallen-Key de ganancia unitaria.

Para poder comparar estos resultados, ejecutamos un archivo *.cir* en Pspice con la siguiente estructura:

```
VAC   1  0  AC 1
R1    1   2   215
C1    2   0   56E-9
xop1  2  3  4  5  3  TL082
VDCp   4   0   dc  12
VDCn   5   0   dc  -12
R2    3   6   511
R3    6   7   1070
C2    7   0   10E-9
xop2  7  8  4  5  8  TL082
C3    6   8   22E-9
C4    8   9   33E-9
C5    9   10  33E-9
R5    10   0   7150
xop3  10 11  4  5  11  TL082
R4    9  11  5360
raux   11  0  1e10

.ac  lin  10001  1  1e5
.probe

* Entrada nudo 1
```

```
* Salida nudo 11
* Dibujar Gráfica de ganancia  20*log10(V(11)/V(1)) en dB y de fase VP(11)

* SUBCIRCUITO OPERACIONAL TL082
*-----------------------------------------------------------------------
.subckt TL082    1 2 3 4 5
*
c1   11  12   2.412E-12
c2    6   7   18.00E-12
css  10  99   5.400E-12
dc    5  53  dx
de   54  5  dx
dlp  90  91  dx
dln  92  90  dx
dp    4  3  dx
egnd  99  0  poly(2) (3,0) (4,0) 0 .5 .5
fb    7  99  poly(5)  vb  vc  ve  vlp  vln  0 3.467E6  -3E6  3E6  3E6  -3E6
ga    6  0 11  12 339.3E-6
gcm   0  6 10  99 17.01E-9
iss  10  4  dc  234.0E-6
hlim  90  0  vlim  1K
j1   11  2  10  jx
j2   12  1  10  jx
r2    6   9   100.0E3
```

```
rd1   3  11   2.947E3
rd2   3  12   2.947E3
ro1   8  5  50
ro2   7  99  170
rp    3  4  20.00E3
rss  10  99  854.7E3
vb    9  0  dc  0
vc    3  53  dc  1.500
ve   54  4  dc  1.500
vlim  7  8  dc  0
vlp  91  0  dc  50
vln   0  92  dc  50
.model  dx  D(Is=800.0E-18 Rs=1)
.model  jx  NJF(Is=2.500E-12 Beta=984.2E-6 Vto=-1)
.ends


.end
```

Figura 6.11: Fichero .cir en Pspice que genera la simulación del filtro paso banda Bessel de 5º orden implementado con celdas Sallen-Key.

Como puede observarse al igual que en los otros filtros, la simulación se realiza hasta una frecuencia de 100 KHz, valor a partir del cual, el amplificador operacional deja de tener el comportamiento esperado.

Exceptuando la descripción del circuito, el resto de los aspectos más relevantes se asemejan a los supuestos estudiados previamente.

Dicha codificación da origen al *diagrama de Bode* expuesto en la figura 6.12 que se muestra:

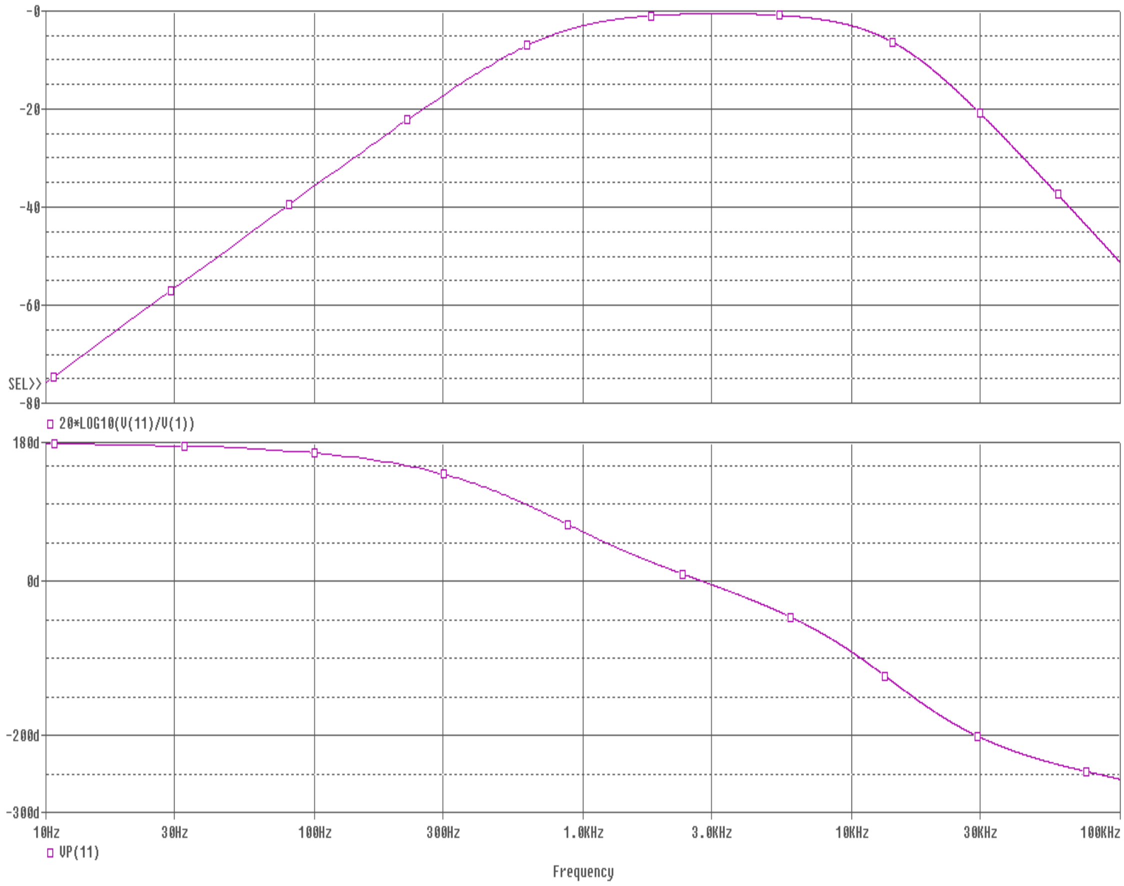


Figura 6.12: Diagrama de Bode del filtro paso banda Bessel de 5º orden implementado con celdas Sallen-Key obtenido en Pspice.

6.4 – Filtro paso bajo Chebyshev de 1º orden con rizado 3 dB implementado con celdas de Rauch

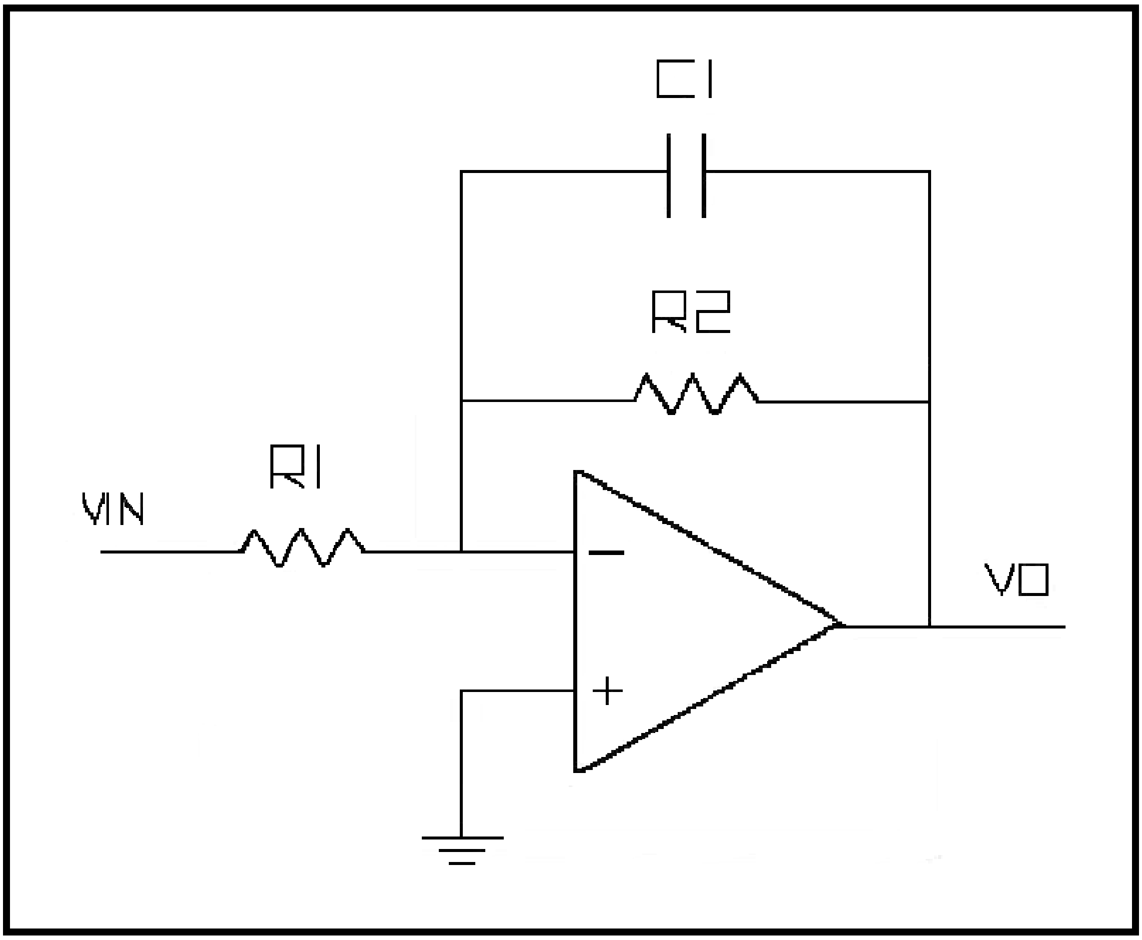


Figura 6.13: Filtro paso bajo Chebyshev de 1º orden con rizado 3 dB implementado con celdas de Rauch en configuración inversora.

Características

Orden = 1 Frecuencia de Corte = 1.0000 KHz Ganancia = 10.0000

Componentes

C1 = 10.0000 nF   R1 = 1.5800 KΩ   R2 = 15.8000 KΩ

En este caso, la función de transferencia del filtro obtenida en Matlab corresponde a la expresión 6.9:

$$H(s) = \frac{-10}{1.58 \cdot 10^{-4}\, s + 1} \qquad (6.9)$$

El signo negativo como ya se comentó, es consecuencia de que el amplificador operacional al estar configurado como inversor, provoca un desfase de 180º de la entrada con respecto a la salida del filtro.

Por otra parte, una diferencia fundamental de los filtros de *Rauch* en relación a los de *Sallen-Key* es la aparición de un determinado valor de ganancia, cuya cuantía es introducida por el usuario en el menú inicial de nuestro software.

En el ejemplo que nos ocupa la ganancia seleccionada es de 10 en escala lineal, si queremos transformarlo en escala logarítmica (dB) y de esta manera comprobar dicho valor en el *diagrama de Bode* resultante, aplicaremos la ecuación G (dB) = $20 \cdot \log_{10}(10) = 20$ dB.

Teniendo esto en cuenta, la representación gráfica del filtro volcada a través de nuestro programa en Matlab tiene el siguiente aspecto:

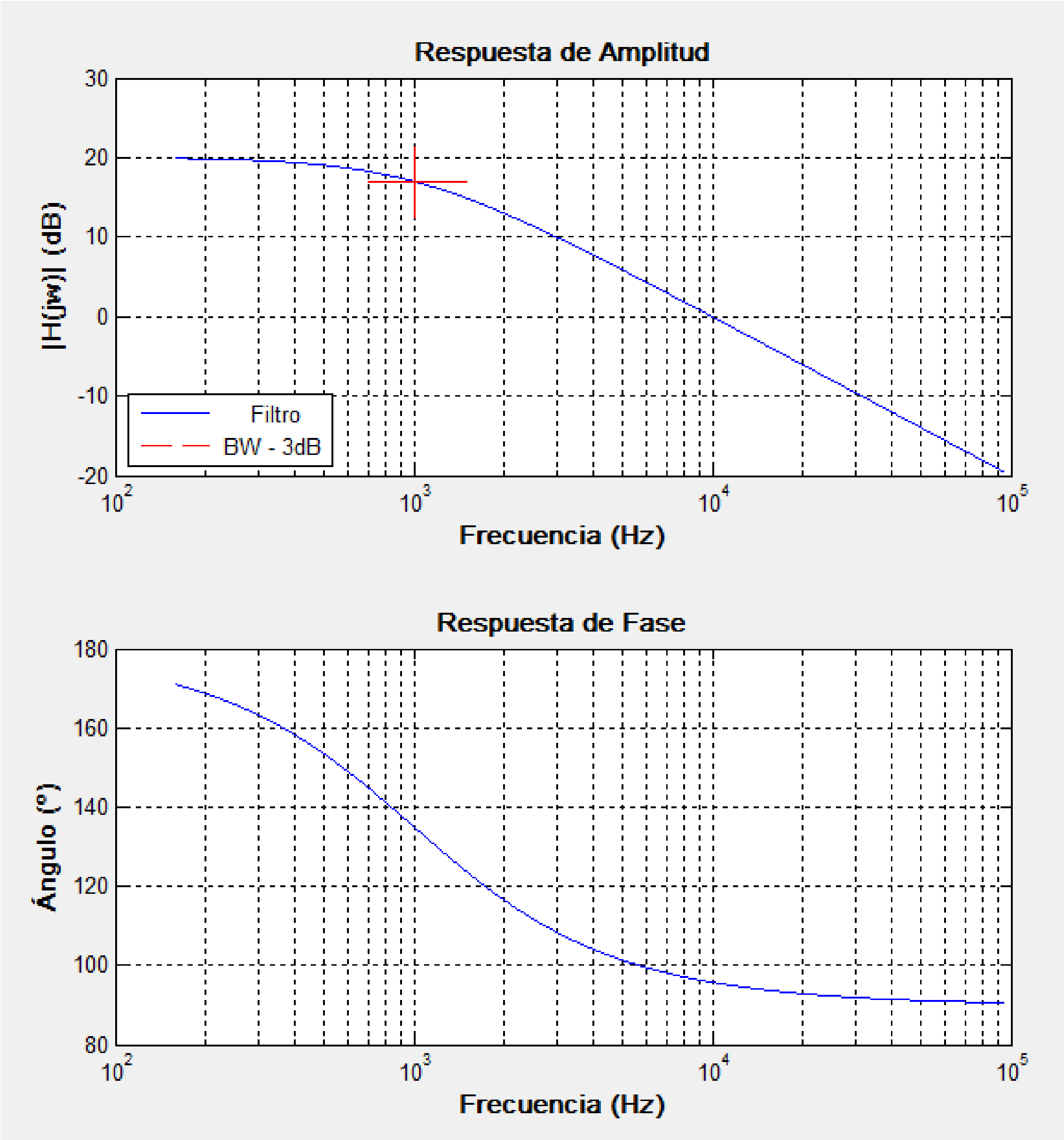


Figura 6.14: Diagrama de Bode del filtro paso bajo Chebyshev de 1º orden con rizado 3 dB implementado con celdas de Rauch obtenido en Matlab.

Así mismo, el fichero *.cir* realizado en Pspice para la simulación del filtro en cuestión queda como sigue:

```
VAC  1  0  AC 1
R1   1  2  1580
C1   2  5  10E-9
R2   2  5  15800
xop1  0  2  3  4  5  TL082
VDCp  3  0  dc  12
VDCn  4  0  dc  -12
raux  5  0  1e10

.ac  lin  10001  1  1e5
.probe

* Entrada nudo 1
* Salida nudo 5
* Dibujar Gráfica de ganancia  20*log10(V(5)/V(1))  en dB  y  de  fase  VP(5)

* SUBCIRCUITO OPERACIONAL TL082
*---------------------------------------------------------------------------
.subckt   TL082    1 2 3 4 5
*
c1   11  12    2.412E-12
c2    6    7     18.00E-12
```

```
css  10  99   5.400E-12
dc    5  53  dx
de    54  5  dx
dlp   90  91  dx
dln   92  90  dx
dp    4  3  dx
egnd  99  0  poly(2) (3,0) (4,0) 0 .5 .5
fb    7  99  poly(5)  vb  vc  ve  vlp  vln  0 3.467E6  -3E6  3E6  3E6  -3E6
ga    6  0 11 12   339.3E-6
gcm   0  6 10  99   17.01E-9
iss  10  4  dc   234.0E-6
hlim  90  0  vlim  1K
j1   11  2  10  jx
j2   12  1  10  jx
r2    6  9  100.0E3
rd1   3  11   2.947E3
rd2   3  12   2.947E3
ro1   8  5  50
ro2   7  99  170
rp    3  4   20.00E3
rss  10  99   854.7E3
vb    9  0  dc  0
vc    3  53  dc  1.500
ve   54  4  dc  1.500
```

```
vlim   7  8  dc  0

vlp   91  0  dc  50

vln   0  92  dc  50

.model  dx  D(Is=800.0E-18 Rs=1)

.model  jx  NJF(Is=2.500E-12 Beta=984.2E-6 Vto=-1)

.ends


.end
```

Figura 6.15: Fichero .cir en Pspice que genera la simulación del filtro paso bajo Chebyshev de 1º orden con rizado 3 dB implementado con celdas de Rauch.

Sabiendo que la salida se encuentra en el nudo 5, simulamos dicho circuito con el objetivo de representar el *diagrama de Bode* asociado.

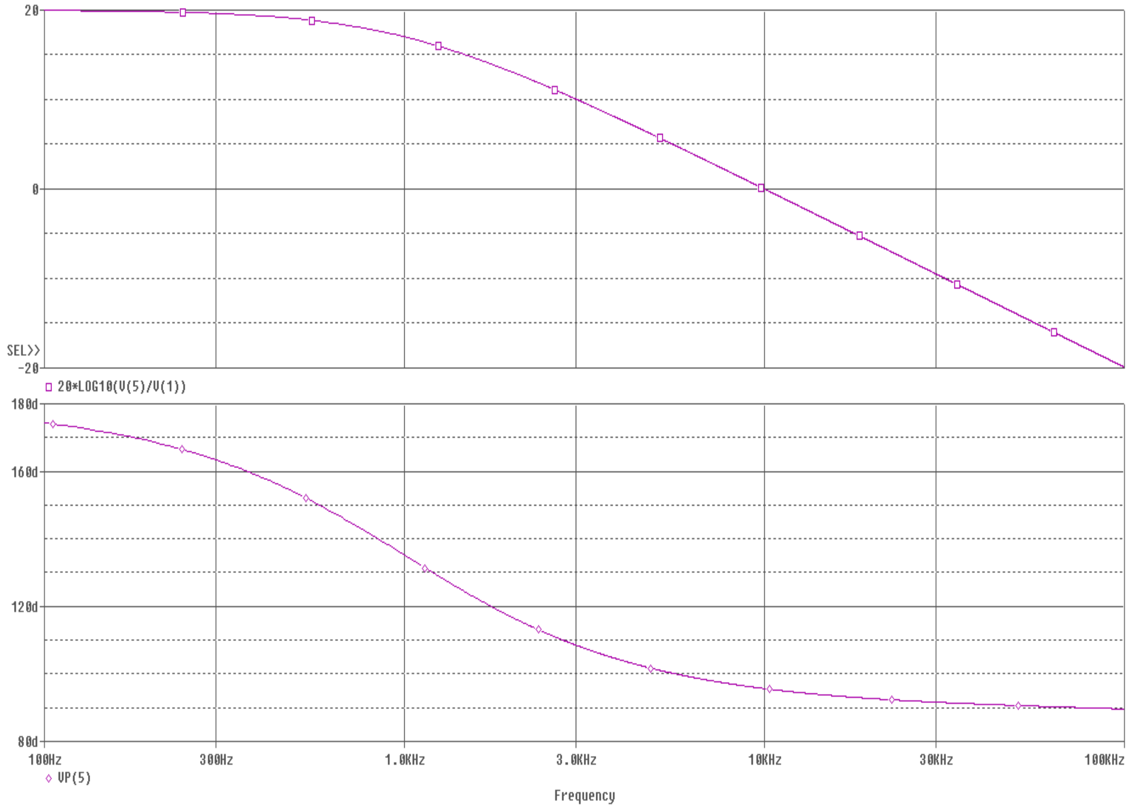


Figura 6.16: Diagrama de Bode del filtro paso bajo Chebyshev de 1º orden con rizado 3 dB implementado con celdas de Rauch obtenido en Pspice.

6.5 – Filtro paso alto Bessel de 4º orden implementado con celdas de Rauch

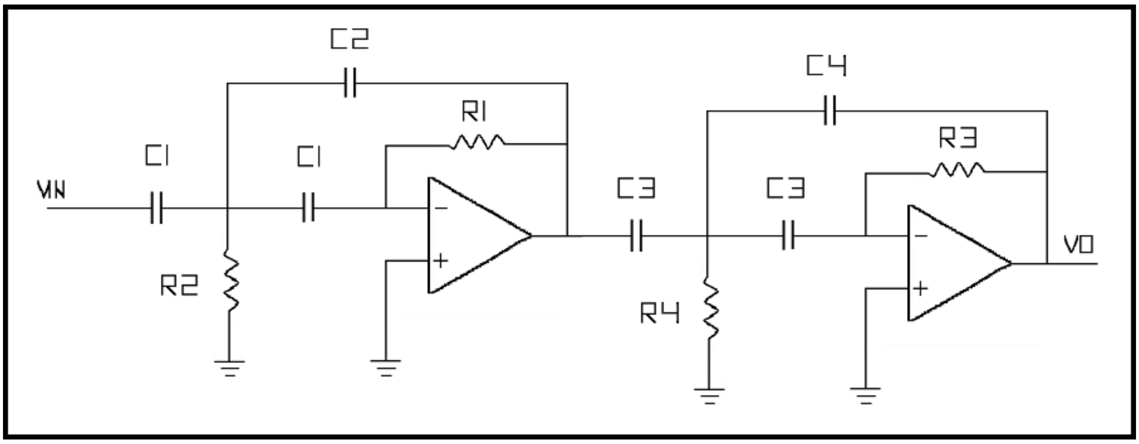


Figura 6.17: Filtro paso alto Bessel de 4º orden implementado con celdas de Rauch en configuración inversora.

Características

Orden = 4 Frecuencia de Corte = 10.0000 KHz Ganancia = 10.0000

Componentes

Primera etapa

C1 = 10.0000 nF C2 = 3.3000 nF

R1 = 8.4500 KΩ R2 = 1.8700 KΩ

Segunda etapa

C3 = 22.0000 nF C4 = 6.8000 nF

R3 = 6.9800 KΩ R4 = 0.6190 KΩ

Observando el circuito, vemos que el filtro está formado por la conexión en cascada de dos etapas de 2º *orden*. Por tanto, la función de transferencia $H(s)$ resultará de multiplicar las funciones de transferencia $H_1(s)$ y $H_2(s)$ de las dos células básicas que lo componen:

$$H_1(s) = \frac{-1.58*10^{-7}\, s^2}{5.214*10^{-8}\, s^2 + 4.357*10^{-4}\, s + 1} \qquad (6.10)$$

$$H_2(s) = \frac{-2.091*10^{-7}\, s^2}{6.464*10^{-8}\, s^2 + 3.145*10^{-4}\, s + 1} \qquad (6.11)$$

Dividiendo arriba y abajo la primera ecuación por $5.214*10^{-8}$ y la segunda por $6.464*10^{-8}$ y multiplicando las expresiones resultantes, la función de transferencia del filtro completo $H(s) = H_1(s) * H_2(s)$ es la descrita a continuación:

$$H(s) = \frac{-3.03\, s^2}{s^2 + 8.356 \cdot 10^3\, s + 1.92 \cdot 10^7} * \frac{-3.235\, s^2}{s^2 + 4.865 \cdot 10^3\, s + 1.548 \cdot 10^7} \tag{6.12}$$

Teóricamente la ganancia del filtro total (10 en este caso), la repartimos por igual entre cada una de las etapas, es decir, que para que el producto de las funciones de transferencia individuales tenga ganancia 10, esta deberá ser de $\sqrt{10}$ (3.162) en cada estructura de 2º *orden*.

Como vemos en la expresión 6.12, las ganancias en las etapas no tienen este valor exacto, debido en gran medida a la elección de valores normalizados para los componentes del filtro (resistencias y condensadores), que son los estándares comerciales que más se aproximan a los resultados obtenidos por las ecuaciones del circuito.

Por otra parte, el *diagrama de Bode* generado en Matlab a partir de la función de transferencia H(s) anterior responde a la gráfica que se adjunta:

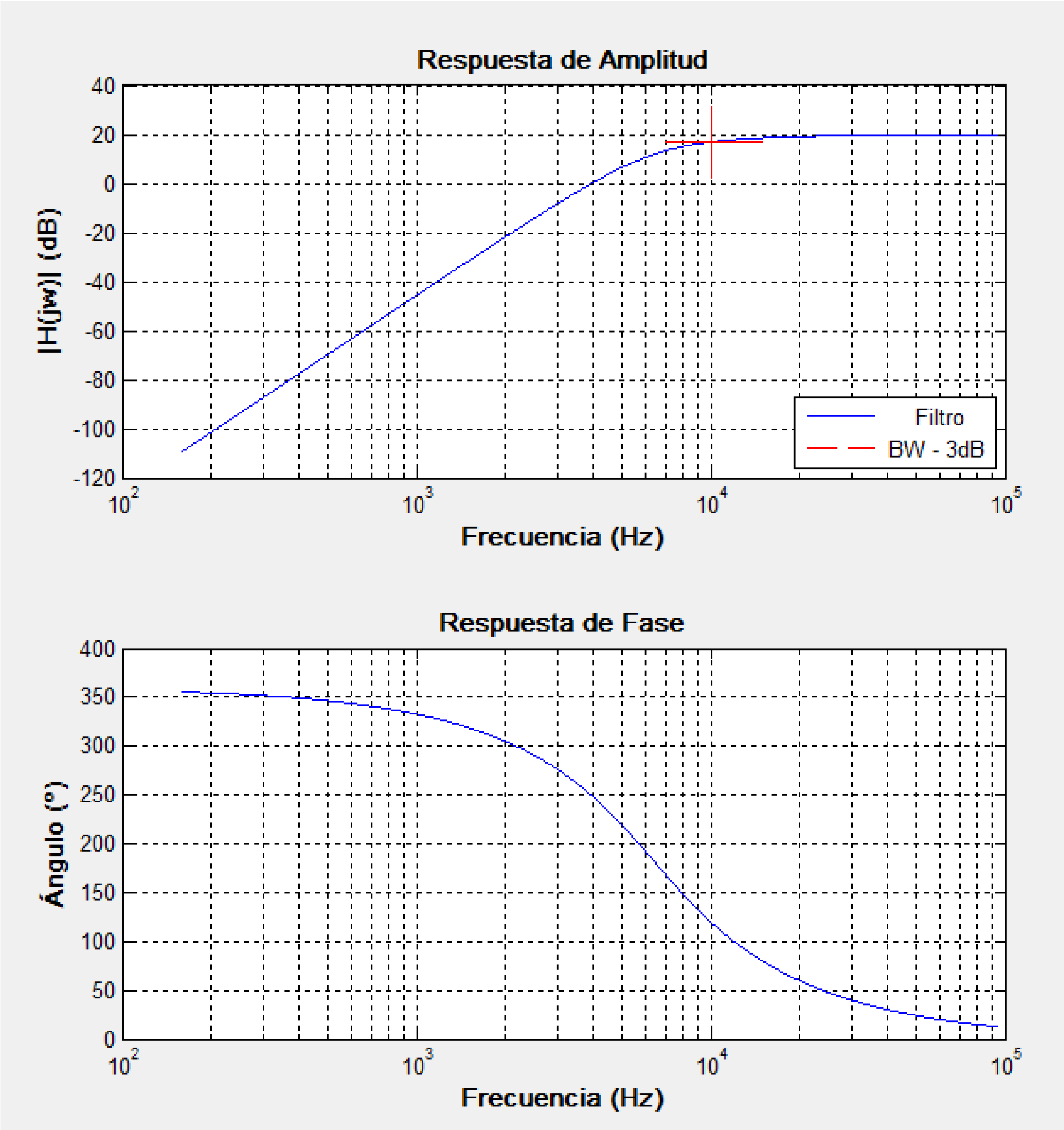


Figura 6.18: Diagrama de Bode del filtro paso alto Bessel de 4º orden implementado con celdas de Rauch obtenido en Matlab.

En lo que respecta al fichero *.cir* implementado en Pspice para la simulación del filtro, es el siguiente:

```
VAC  1  0  AC 1
C1   1  2  10E-9
R2   2  0  1870
C2   2  3  10E-9
xop1  0  3  4  5  6  TL082
VDCp   4  0   dc  12
VDCn   5  0   dc  -12
C3    2  6   3.3E-9
R1    3  6   8450
C4    6  7   22E-9
R3    8  9   6980
C5    7  8   22E-9
xop2  0  8  4  5  9  TL082
C6    7  9   6.8E-9
R4    7  0   619
raux   9  0  1e10

.ac  lin  10001  1  1e5
.probe

* Entrada nudo 1
* Salida nudo 11
```

```
* Dibujar Gráfica de ganancia  20*log10(V(9)/V(1)) en dB y de fase  VP(9)


* SUBCIRCUITO OPERACIONAL TL082
*--------------------------------------------------------------------------
.subckt TL082    1 2 3 4 5
*
c1   11  12   2.412E-12
c2    6   7   18.00E-12
css  10  99   5.400E-12
dc    5  53  dx
de   54  5  dx
dlp  90  91 dx
dln  92  90  dx
dp    4  3  dx
egnd  99  0  poly(2) (3,0) (4,0) 0 .5 .5
fb    7  99  poly(5)  vb  vc  ve  vlp  vln  0  3.467E6  -3E6  3E6  3E6  -3E6
ga    6  0 11 12   339.3E-6
gcm   0  6 10 99   17.01E-9
iss  10  4  dc   234.0E-6
hlim  90  0  vlim  1K
j1   11  2  10  jx
j2   12  1  10  jx
r2    6   9   100.0E3
rd1   3  11   2.947E3
```

```
rd2   3  12   2.947E3

ro1   8  5  50

ro2   7  99  170

rp    3  4   20.00E3

rss  10  99   854.7E3

vb    9  0 dc 0

vc    3  53  dc  1.500

ve   54  4  dc  1.500

vlim  7  8  dc  0

vlp  91  0  dc  50

vln   0  92  dc  50

.model  dx  D(Is=800.0E-18 Rs=1)

.model  jx  NJF(Is=2.500E-12 Beta=984.2E-6 Vto=-1)

.ends


.end
```

Figura 6.19: Fichero .cir en Pspice que genera la simulación del filtro paso alto Bessel de 4º orden implementado con celdas de Rauch.

Considerando que la salida corresponde al nudo 11 y asumiendo la limitación frecuencial de funcionamiento del amplificador operacional (100 KHz), el *diagrama de Bode* generado al ejecutar la aplicación tiene el aspecto que se muestra:

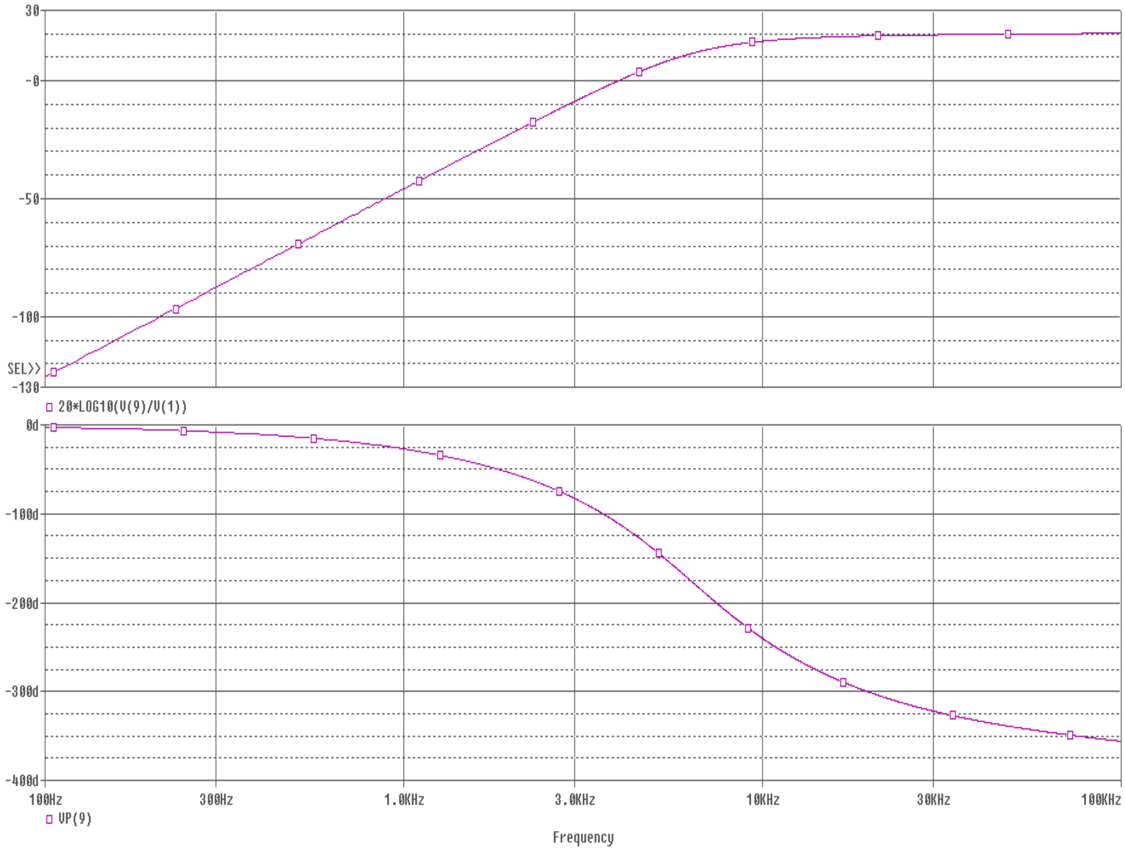


Figura 6.20: Diagrama de Bode del filtro paso alto Bessel de 4º orden implementado con celdas de Rauch obtenido en Pspice.

6.6 – Filtro paso banda Butterworth de 5º orden implementado con celdas de Rauch

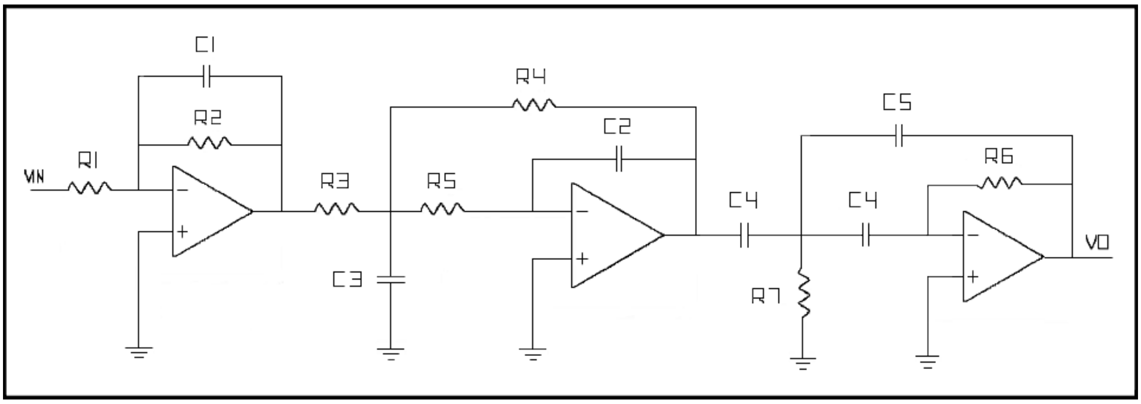


Figura 6.21: Filtro paso banda Butterworth de 5º orden implementado con celdas de Rauch en configuración inversora.

Características

Orden = 5 Ganancia = 10.0000 Frec. Corte Inferior = 1.0000 KHz

Frec. Corte Superior = 10.0000 KHz Frecuencia Central = 3.1623 KHz

Ancho de Banda = 9.0000 KHz Factor de Calidad: 0.3514

Componentes

Etapa Paso Bajo (Primer Orden)

C1 = 22.0000 nF R1 = 0.3320 KΩ R2 = 0.7150 KΩ

Etapa Paso Bajo (Segundo Orden)

C2 = 1.2000 nF   C3 = 18.0000 nF

R3 = 1.8700 KΩ   R4 = 4.0200 KΩ   R5 = 2.9400 KΩ

Etapa Paso Alto (Segundo Orden)

C4 = 22.0000 nF   C5 = 10.0000 nF

R6 = 27.4000 KΩ   R7 = 4.1200 KΩ

De manera similar al ejemplo 6.3, el filtro paso banda está compuesto por dos etapas de 1º y 2º *orden* y como consecuencia de ello, la función de transferencia resultante $H(s)$ surgirá del producto de la funciones de transferencia $H_1(s)$, $H_2(s)$ y $H_3(s)$ de las etapas específicas:

$$H_1(s) = \frac{-2.154}{1.573*10^{-5}\ s + 1} \tag{6.13}$$

$$H_2(s) = \frac{-2.15}{2.553*10^{-10}\ s^2 + 1.594*10^{-5}\ s + 1} \tag{6.14}$$

$$H_3(s) = \frac{-5.464 \cdot 10^{-8}\ s^2}{2.484 \cdot 10^{-8}\ s^2 + 2.225 \cdot 10^{-4}\ s + 1} \qquad (6.15)$$

En este supuesto, transformaremos la ecuación $H_3(s)$ dividiendo numerador y denominador por $2.484 \cdot 10^{-8}$. Realizando esta modificación y multiplicando las funciones de transferencia respectivas, tendremos la ecuación asociada al filtro completo $H(s) = H_1(s) * H_2(s) * H_3(s)$:

$$H(s) = \frac{-2.154}{1.573 \cdot 10^{-5}\ s + 1} * \frac{-2.15}{2.553 \cdot 10^{-10}\ s^2 + 1.594 \cdot 10^{-5}\ s + 1} * \longrightarrow$$

$$\longrightarrow * \frac{-2.2}{s^2 + 8.957 \cdot 10^{3}\ s + 4.026 \cdot 10^{7}} \qquad (6.16)$$

Utilizando el mismo razonamiento expuesto para el caso anterior, la ganancia del filtro completo (de valor 10 en el ejemplo) se reparte de manera equitativa entre cada una de las 3 etapas individuales que lo integran, de tal forma que el producto de las funciones de transferencia de todas ellas proporcione como resultado una ganancia de 10 en la función H(s) del filtro total.

Y esto se consigue estableciendo una ganancia $\sqrt[3]{10}$ (2.154) en las células básicas de 1º y 2º *orden*. Inspeccionando la ecuación 6.16, nos damos cuenta de que este valor de ganancia se aproxima bastante al teórico en las 2 etapas paso bajo, mientras que difiere algo más en la estructura paso alto final.

El motivo de este pequeño desajuste como ya comentamos está en la elección de valores normalizados de resistencias y condensadores que no son los valores exactos obtenidos a través de las ecuaciones asociadas al circuito sino los modelos comerciales más aproximados a dichos resultados.

Definida la función de transferencia del filtro, el *diagrama de Bode* representado a través de la simulación del software en Matlab está ligado a la imagen posterior:

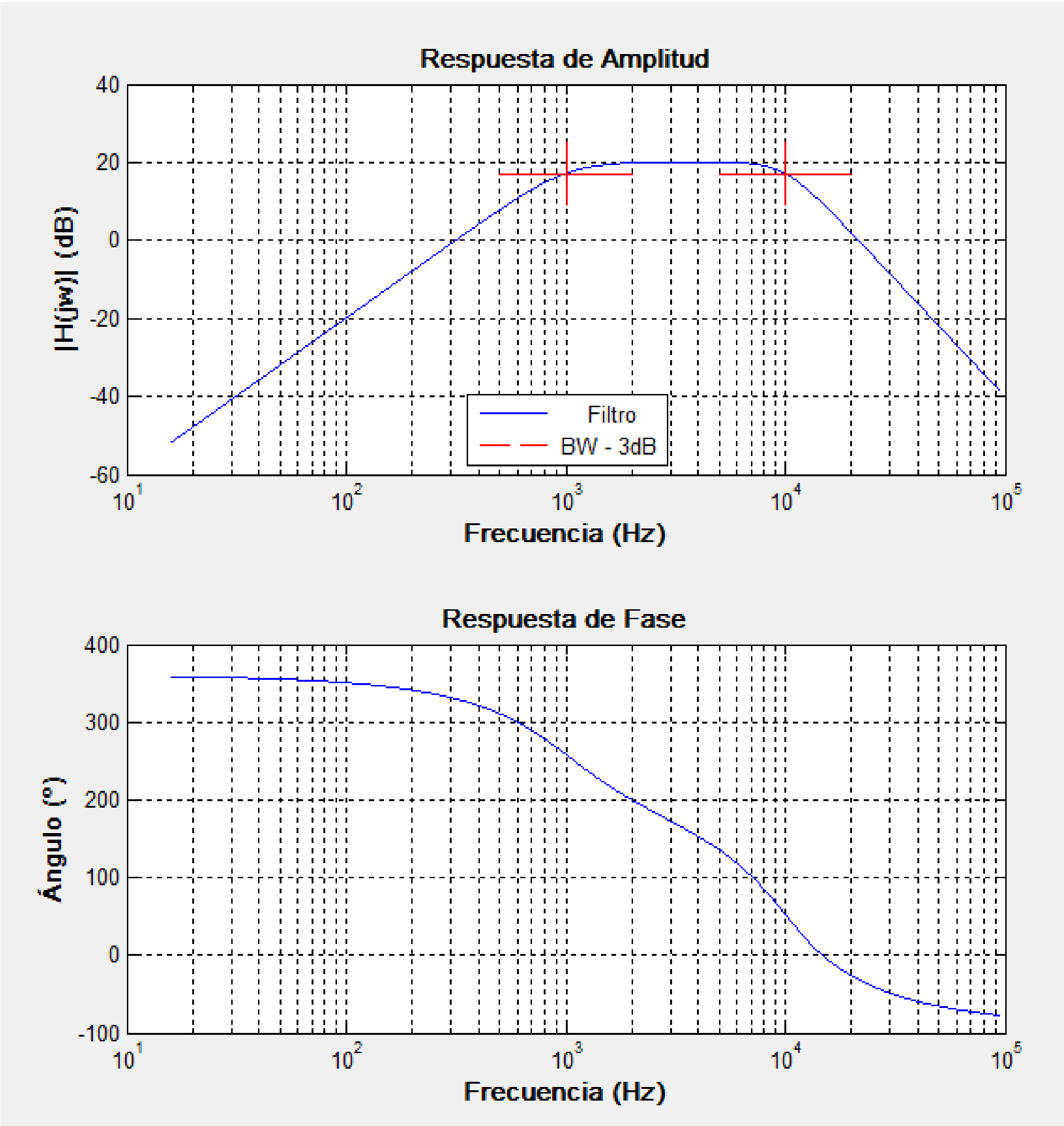


Figura 6.22: Diagrama de Bode del filtro paso banda Butterworth de 5º orden implementado con celdas de Rauch obtenido en Matlab.

Mientras que el fichero *.cir* desarrollado en Pspice para comparar estos resultados, presenta el código que se expone a continuación:

```
VAC   1   0   AC  1

R1    1   2   332

xop1   0  2  3  4  5  TL082

VDCp   3   0   dc   12

VDCn   4   0   dc  -12

C1    2   5    22E-9

R2    2   5    715

R3    5   6    1870

C3    6   0    18E-9

R5    6   7    2940

R4    6   8    4020

C2    7   8    1.2E-9

xop2   0  7  3  4  8  TL082

C4    8   9    22E-9

C5    9  10   22E-9

R7    9  0   4120

xop3   0  10  3  4  11  TL082

C6    9    11   10E-9

R6    10   11   27400

Raux   11   0   1E10


.ac  lin  10001   1  1e5
```

```
.probe


* Entrada nudo 1
* Salida nudo 11
* Dibujar Gráfica de ganancia  20*log10(V(11)/V(1)) en dB y de fase  VP(11)


* SUBCIRCUITO OPERACIONAL TL082
*--------------------------------------------------------------------------
.subckt  TL082    1 2 3 4 5
*
 c1   11  12   2.412E-12
 c2    6   7  18.00E-12
 css  10  99   5.400E-12
 dc    5  53  dx
 de   54  5  dx
 dlp  90  91 dx
 dln  92  90  dx
 dp    4  3  dx
 egnd   99   0  poly(2) (3,0) (4,0) 0 .5 .5
 fb    7  99  poly(5)  vb  vc  ve  vlp  vln  0 3.467E6  -3E6  3E6  3E6  -3E6
 ga    6  0 11 12  339.3E-6
 gcm   0  6 10  99  17.01E-9
 iss  10  4  dc   234.0E-6
 hlim  90  0  vlim  1K
```

```
 j1   11  2  10  jx
 j2   12  1  10  jx
 r2    6  9  100.0E3
 rd1   3  11   2.947E3
 rd2   3  12   2.947E3
 ro1   8  5  50
 ro2   7  99  170
 rp    3  4  20.00E3
 rss  10  99  854.7E3
 vb    9  0  dc  0
 vc    3  53  dc  1.500
 ve   54   4  dc  1.500
 vlim  7  8  dc  0
 vlp   91  0  dc  50
 vln   0  92  dc  50
.model  dx  D(Is=800.0E-18 Rs=1)
.model  jx  NJF(Is=2.500E-12 Beta=984.2E-6 Vto=-1)
.ends


.end
```

Figura 6.23: Fichero .cir en Pspice que genera la simulación del filtro paso banda Butterworth de 5º orden implementado con celdas de Rauch.

Compilando este fichero, sabiendo que la ganancia es $20*\log_{10}(10) = 20$ en dB, que la salida corresponde al nudo 11 y que realizaremos la simulación hasta una frecuencia (100 KHz en este caso) porque es el límite en frecuencia del amplificador operacional TL082, obtendremos su *diagrama de Bode*.

A partir de la frecuencia anterior comprobamos en todas las simulaciones de todos los filtros en Pspice que tanto la ganancia como la fase no presentaban la evolución teórica que deberían presentar y que por ejemplo, si se obtenía con el software desarrollado en Matlab.

Por tanto para poder comparar ambas simulaciones en uno y otro programa, la representación gráfica en los supuestos que se incluyen quedó establecida hasta el valor mencionado de 100 KHz.

No obstante de manera general, el *diagrama de Bode* del filtro realizado en Matlab (objeto central del proyecto que nos ocupa) fue programado para simularse hasta una frecuencia aproximada de 1 MHz, adecuándose sólo para estos ejemplos al límite de 100 KHz.

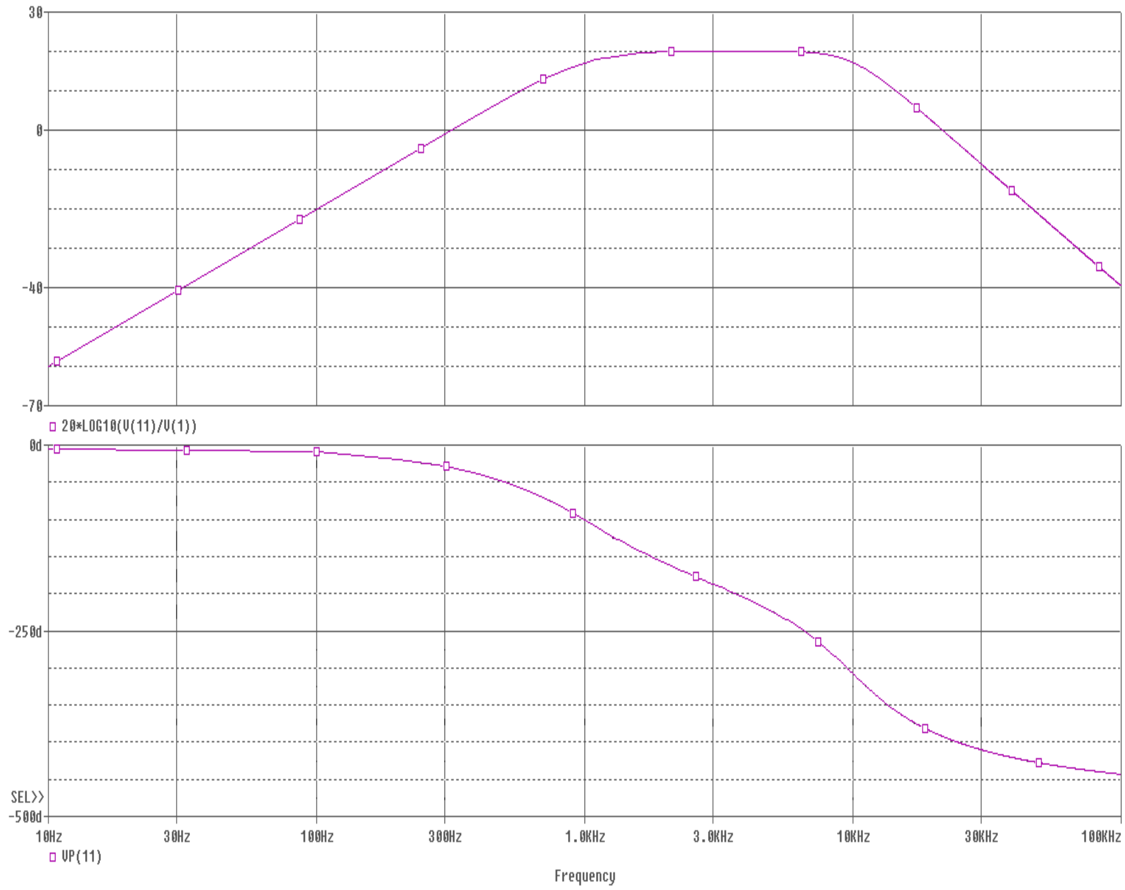


Figura 6.24: Diagrama de Bode del filtro paso banda Butterworth de 5º orden implementado con celdas de Rauch obtenido en Pspice.

CAPÍTULO VII

CONCLUSIONES
Y
LÍNEAS
FUTURAS

En este capítulo realizaremos un breve resumen de todo el trabajo desarrollado a lo largo del proyecto, así como de las conclusiones que pueden extraerse a partir de los resultados obtenidos.

Así mismo plantearemos posibles alternativas y líneas futuras de trabajo, tomando como referencia la aplicación aquí expuesta.

En la primera parte se abordaron los fundamentos teóricos relativos a los filtros que nos permitan asentar los diferentes conceptos y de esta manera, recurrir a ellos para la elaboración de nuestro software.

En este sentido, describimos las propiedades básicas de las *funciones de red*, las cuales engloban a su vez a las *funciones de transferencia*, tan importantes estas en la representación gráfica del *diagrama de Bode* de los circuitos diseñados.

A continuación, se ha profundizado en las características generales de los filtros, tales como el *orden*, la *frecuencia de corte*, el *ancho de banda*, el nivel de *rizado*, el *factor de calidad*, el *retardo de fase y de grupo*, etc.

Relacionado con lo anterior, se llevó a cabo una clasificación de los dispositivos en función de las frecuencias atenuadas (*paso bajo, paso alto, paso banda y rechazo de banda*), de la tecnología empleada (*pasivos* o *activos*) o de la función matemática seleccionada (*Butterworth, Chebyshev, Bessel y Cauer*).

A continuación, nos centramos en el análisis de los *filtros activos* de *Sallen-Key* y de *Rauch*, obteniendo las *funciones de transferencia* de sus respectivas etapas *paso bajo* y *paso alto* de 1º y 2º *orden* y detallando a su vez, las condiciones impuestas y las ecuaciones aplicadas en cada una de ellas para el cálculo de todos los componentes del circuito.

Finalmente, se expuso la manera de implementar los filtros diseñados en nuestro software, a partir de la conexión en cascada de las etapas anteriormente descritas, incluyendo las combinaciones de las estructuras *paso bajo* y *paso alto* existentes en los circuitos *paso banda* de diferentes *órdenes*.

Sentadas las bases para la realización de los diversos tipos de filtros, focalizamos todo el interés en el software desarrollado en Matlab.

Después de una breve introducción referente al entorno, hicimos hincapié en la herramienta GUIDE que contiene, utilidad que nos permitirá la creación de las GUIs (interfaces gráficas de usuario) fundamentales en la elaboración de nuestra aplicación.

A lo largo de este capítulo, detallamos cada uno de los aspectos del programa de control, describiendo el entorno gráfico realizado y las funciones más importantes e incluyendo un diagrama de flujo a modo orientativo, que facilite la comprensión del software de una forma más visual.

En el último capítulo, se realizó la simulación de varios ejemplos de filtros, combinando en la medida de lo posible, todas las tipologías (*paso bajo*, *paso alto* o *paso banda*) con las diferentes funciones matemáticas de aproximación que se pueden seleccionar (*Butterworth*, *Chebyshev* o *Bessel*) e implementándolo con etapas de 1º y 2º *orden*, tanto de *Sallen – Key* como de *Rauch*.

En general, a la hora de construir un filtro con unas especificaciones determinadas, lo primero que debemos decidir es el tipo de diseño que más nos conviene.

Si lo que nos interesa es que tenga unas características transitorias muy buenas, es decir que el *retardo de grupo* sea constante en la banda de paso, optaremos por un filtro *Bessel*.

En cambio si consideramos prioritaria la respuesta en *ganancia*, elegiremos entre un filtro *Butterworth* (máximamente plano en amplitud en la banda de paso pero con una transición menos abrupta en la banda rechazada), o bien nos decantaremos por un filtro *Chebyshev* (con una mayor pendiente de atenuación a partir de la *frecuencia de corte*, a costa de contar con un cierto nivel de *rizado* en la banda de paso), extremos todos estos confirmados en la simulaciones realizadas.

Con el objetivo de poder validar los resultados obtenidos en Matlab, juzgamos oportuno la recreación del mismo diseño en Pspice.

En este sentido, señalar que en Matlab es posible estimar la respuesta real que tendría el filtro, alejada de la ideal, en función de la *tolerancia* de los componentes que se utilizan, lo cual constituye una ventaja a la hora del diseño.

Por el contrario el amplificador operacional se considera ideal en Matlab, a diferencia de lo que sucede en Pspice, simulación en la que empleamos el OPAMP TL082.

Esto supone que el rango de validez de dicho operacional provoque un comportamiento no deseado del mismo a frecuencias altas, (a partir de 100 KHz en los ejemplos simulados) y como consecuencia, exista una limitación frecuencial en nuestro análisis de Pspice.

Sin embargo, como se ha podido comprobar, el *diagrama de Bode* resultante (representado repito hasta 100 KHz) tanto en Matlab como en Pspice, presenta valores de *ganancia* y de *fase* muy similares para las mismas frecuencias.

En cuanto al software desarrollado, mencionar el entorno intuitivo y de fácil manejo que posee, el cual permite la interacción con el usuario en cualquier instante.

Desde el punto de vista personal, la realización de este proyecto fin de carrera ha supuesto la posibilidad de trabajar y adquirir una mayor familiaridad con la herramienta GUIDE de Matlab y su interfaz gráfica (GUI), creando diferentes eventos a partir de ella.

En lo que respecta a líneas futuras de trabajo, el hecho de que el software esté programado en entorno Matlab nos facilita seguir desarrollando nuevas funciones y utilidades que mejoren la caracterización de filtros activos, incluyendo por ejemplo la simulación de filtros de rechazo de banda.

De la misma manera, conseguir ejecutar el programa en cualquier sistema operativo mejoraría en gran medida su portabilidad y aumentaría el número de usuarios potenciales.

Otra alternativa paralela a la realizada en el presente proyecto con los filtros activos, podría pasar por el desarrollo de una herramienta de software para el análisis y diseño de filtros pasivos que permitiera igualmente extraer conclusiones interesantes.

ANEXO

TABLAS DE COEFICIENTES BUTTERWORTH CHEBYSHEV Y BESSEL

Las tablas siguientes contienen los coeficientes asociados a los tipos de filtro *Butterworth*, *Chebyshev* de *rizado* 0.5, 1, 2 o 3 *dB* y *Bessel*, (tablas 16-6, 16-7, 16-8 y 16-9).

Los parámetros que se detallan en cada una de ellas son los expuestos a continuación:

N es el *orden* del filtro correspondiente.

i es el número del filtro parcial, es decir, de la etapa concreta del filtro a la que nos refiramos.

$\mathbf{a_i}$ $\mathbf{b_i}$ son los coeficientes del filtro.

$\mathbf{K_i}$ es la relación entre la *frecuencia de corte* de un filtro parcial ($\mathbf{f_{Ci}}$) y la del filtro total ($\mathbf{f_C}$). Esta relación es utilizada para calcular el *ancho de banda* de *ganancia* unidad de los amplificadores operacionales y para simplificar el análisis en el diseño de un filtro, midiendo f_{Ci} y comparándolo con f_C.

$\mathbf{Q_i}$ es el *factor de calidad* del filtro parcial.

Table 16–4. Bessel Coefficients

N	i	a_i	b_i	$k_i = f_{ci} / f_c$	Q_i
1	1	1.0000	0.0000	1.000	--
2	1	1.3617	0.6180	1.000	0.58
3	1	0.7560	0.0000	1.323	--
	2	0.9996	0.4772	1.414	0.69
4	1	1.3397	0.4889	0.978	0.52
	2	0.7743	0.3890	1.797	0.81
5	1	0.6656	0.0000	1.502	--
	2	1.1402	0.4128	1.184	0.56
	3	0.6216	0.3245	2.138	0.92
6	1	1.2217	0.3887	1.063	0.51
	2	0.9686	0.3505	1.431	0.61
	3	0.5131	0.2756	2.447	1.02
7	1	0.5937	0.0000	1.648	--
	2	1.0944	0.3395	1.207	0.53
	3	0.8304	0.3011	1.695	0.66
	4	0.4332	0.2381	2.731	1.13
8	1	1.1112	0.3162	1.164	0.51
	2	0.9754	0.2979	1.381	0.56
	3	0.7202	0.2621	1.963	0.71
	4	0.3728	0.2087	2.992	1.23
9	1	0.5386	0.0000	1.857	--
	2	1.0244	0.2834	1.277	0.52
	3	0.8710	0.2636	1.574	0.59
	4	0.6320	0.2311	2.226	0.76
	5	0.3257	0.1854	3.237	1.32
10	1	1.0215	0.2650	1.264	0.50
	2	0.9393	0.2549	1.412	0.54
	3	0.7815	0.2351	1.780	0.62
	4	0.5604	0.2059	2.479	0.81
	5	0.2883	0.1665	3.466	1.42

Table 16–5. Butterworth Coefficients

N	i	a_i	b_i	$k_i = f_{ci} / f_c$	Q_i
1	1	1.0000	0.0000	1.000	--
2	1	1.4142	1.0000	1.000	0.71
3	1	1.0000	0.0000	1.000	--
	2	1.0000	1.0000	1.272	1.00
4	1	1.8478	1.0000	0.719	0.54
	2	0.7654	1.0000	1.390	1.31
5	1	1.0000	0.0000	1.000	--
	2	1.6180	1.0000	0.859	0.62
	3	0.6180	1.0000	1.448	1.62
6	1	1.9319	1.0000	0.676	0.52
	2	1.4142	1.0000	1.000	0.71
	3	0.5176	1.0000	1.479	1.93
7	1	1.0000	0.0000	1.000	--
	2	1.8019	1.0000	0.745	0.55
	3	1.2470	1.0000	1.117	0.80
	4	0.4450	1.0000	1.499	2.25
8	1	1.9616	1.0000	0.661	0.51
	2	1.6629	1.0000	0.829	0.60
	3	1.1111	1.0000	1.206	0.90
	4	0.3902	1.0000	1.512	2.56
9	1	1.0000	0.0000	1.000	--
	2	1.8794	1.0000	0.703	0.53
	3	1.5321	1.0000	0.917	0.65
	4	1.0000	1.0000	1.272	1.00
	5	0.3473	1.0000	1.521	2.88
10	1	1.9754	1.0000	0.655	0.51
	2	1.7820	1.0000	0.756	0.56
	3	1.4142	1.0000	1.000	0.71
	4	0.9080	1.0000	1.322	1.10
	5	0.3129	1.0000	1.527	3.20

Table 16–6. Tschebyscheff Coefficients for 0.5 dB Rippel

N	i	a_i	b_i	$k_i = f_{ci} / f_c$	Q_i
1	1	1.0000	0.0000	1.000	--
2	1	1.3614	1.3827	1.000	0.86
3	1	1.8636	0.0000	0.537	--
	2	0.0640	1.1931	1.335	1.71
4	1	2.6282	3.4341	0.538	0.71
	2	0.3648	1.1509	1.419	2.94
5	1	2.9235	0.0000	0.342	--
	2	1.3025	2.3534	0.881	1.18
	3	0.2290	1.0833	1.480	4.54
6	1	3.8645	6.9797	0.366	0.68
	2	0.7528	1.8573	1.078	1.81
	3	0.1589	1.0711	1.495	6.51
7	1	4.0211	0.0000	0.249	--
	2	1.8729	4.1795	0.645	1.09
	3	0.4861	1.5676	1.208	2.58
	4	0.1156	1.0443	1.517	8.84
8	1	5.1117	11.9607	0.276	0.68
	2	1.0639	2.9365	0.844	1.61
	3	0.3439	1.4206	1.284	3.47
	4	0.0885	1.0407	1.521	11.53
9	1	5.1318	0.0000	0.195	--
	2	2.4283	6.6307	0.506	1.06
	3	0.6839	2.2908	0.989	2.21
	4	0.2559	1.3133	1.344	4.48
	5	0.0695	1.0272	1.532	14.58
10	1	6.3648	18.3695	0.222	0.67
	2	1.3582	4.3453	0.689	1.53
	3	0.4822	1.9440	1.091	2.89
	4	0.1994	1.2520	1.381	5.61
	5	0.0563	1.0263	1.533	17.99

Table 16–7. Tschebyscheff Coefficients for 1 dB Rippel

N	i	a_i	b_i	$k_i = f_{ci} / f_c$	Q_i
1	1	1.0000	0.0000	1.000	--
2	1	1.3022	1.5515	1.000	0.96
3	1	2.2156	0.0000	0.451	--
	2	0.5442	1.2057	1.353	2.02
4	1	2.5904	4.1301	0.540	0.78
	2	0.3039	1.1697	1.417	3.56
5	1	3.5711	0.0000	0.280	--
	2	1.1280	2.4896	0.894	1.40
	3	0.1872	1.0814	1.486	5.56
6	1	3.8437	8.5529	0.366	0.76
	2	0.6292	1.9124	1.082	2.20
	3	0.1296	1.0766	1.493	8.00
7	1	4.9520	0.0000	0.202	--
	2	1.6338	4.4899	0.655	1.30
	3	0.3987	1.5834	1.213	3.16
	4	0.0937	1.0432	1.520	10.90
8	1	5.1019	14.7608	0.276	0.75
	2	0.8916	3.0426	0.849	1.96
	3	0.2806	1.4334	1.285	4.27
	4	0.0717	1.0432	1.520	14.24
9	1	6.3415	0.0000	0.158	--
	2	2.1252	7.1711	0.514	1.26
	3	0.5624	2.3278	0.994	2.71
	4	0.2076	1.3166	1.346	5.53
	5	0.0562	1.0258	1.533	18.03
10	1	6.3634	22.7468	0.221	0.75
	2	1.1399	4.5167	0.694	1.86
	3	0.3939	1.9665	1.093	3.56
	4	0.1616	1.2569	1.381	6.94
	5	0.0455	1.0277	1.532	22.26

Table 16–8. Tschebyscheff Coefficients for 2 dB Rippel

N	i	a_i	b_i	$k_i = f_{ci} / f_c$	Q_i
1	1	1.0000	0.0000	1.000	--
2	1	1.1813	1.7775	1.000	1.13
3	1	2.7994	0.0000	0.357	--
	2	0.4300	1.2036	1.378	2.55
4	1	2.4025	4.9862	0.550	0.93
	2	0.2374	1.1896	1.413	4.59
5	1	4.6345	0.0000	0.216	--
	2	0.9090	2.6036	0.908	1.78
	3	0.1434	1.0750	1.493	7.23
6	1	3.5880	10.4648	0.373	0.90
	2	0.4925	1.9622	1.085	2.84
	3	0.0995	1.0826	1.491	10.46
7	1	6.4760	0.0000	0.154	--
	2	1.3258	4.7649	0.665	1.65
	3	0.3067	1.5927	1.218	4.12
	4	0.0714	1.0384	1.523	14.28
8	1	4.7743	18.1510	0.282	0.89
	2	0.6991	3.1353	0.853	2.53
	3	0.2153	1.4449	1.285	5.58
	4	0.0547	1.0461	1.518	18.39
9	1	8.3198	0.0000	0.120	--
	2	1.7299	7.6580	0.522	1.60
	3	0.4337	2.3549	0.998	3.54
	4	0.1583	1.3174	1.349	7.25
	5	0.0427	1.0232	1.536	23.68
10	1	5.9618	28.0376	0.226	0.89
	2	0.8947	4.6644	0.697	2.41
	3	0.3023	1.9858	1.094	4.66
	4	0.1233	1.2614	1.380	9.11
	5	0.0347	1.0294	1.531	29.27

Table 16–9. Tschebyscheff Coefficients for 3 dB Rippel

N	i	a_i	b_i	$k_i = f_{ci} / f_c$	Q_i
1	1	1.0000	0.0000	1.000	--
2	1	1.0650	1.9305	1.000	1.30
3	1	3.3496	0.0000	0.299	--
	2	0.3559	1.1923	1.396	3.07
4	1	2.1853	5.5339	0.557	1.08
	2	0.1964	1.2009	1.410	5.58
5	1	5.6334	0.0000	0.178	--
	2	0.7620	2.6530	0.917	2.14
	3	0.1172	1.0686	1.500	8.82
6	1	3.2721	11.6773	0.379	1.04
	2	0.4077	1.9873	1.086	3.46
	3	0.0815	1.0861	1.489	12.78
7	1	7.9064	0.0000	0.126	--
	2	1.1159	4.8963	0.670	1.98
	3	0.2515	1.5944	1.222	5.02
	4	0.0582	1.0348	1.527	17.46
8	1	4.3583	20.2948	0.286	1.03
	2	0.5791	3.1808	0.855	3.08
	3	0.1765	1.4507	1.285	6.83
	4	0.0448	1.0478	1.517	22.87
9	1	10.1759	0.0000	0.098	--
	2	1.4585	7.8971	0.526	1.93
	3	0.3561	2.3651	1.001	4.32
	4	0.1294	1.3165	1.351	8.87
	5	0.0348	1.0210	1.537	29.00
10	1	5.4449	31.3788	0.230	1.03
	2	0.7414	4.7363	0.699	2.94
	3	0.2479	1.9952	1.094	5.70
	4	0.1008	1.2638	1.380	11.15
	5	0.0283	1.0304	1.530	35.85

BIBLIOGRAFÍA

[1] José Espí López, Gustavo Camps - Valls, Rafael Magdalena Benedito. *Síntesis de redes: Impedancias y Filtros.* Ed. Delta Publicaciones, 2008.

[2] *Active Filters: Theory and Design*. Pactitis S.A. Ed. Boca Ratón, Fla. : CRC Press, cop. 2008.

[3] Won Y. Yang and Seung C. Lee. *Circuit systems with MATLAB and PSpice*. Ed. Singapore : John Wiley & Sons (Asia), cop. 2007.

[4] Mercedes Granda Miguel, Elena Mediavilla Bolado. *Instrumentación Electrónica: Transductores y Acondicionadores de Señal*. PUbliCan Ediciones de la Universidad de Cantabria, 2010.

[5] https://www.ulpgc.es/hege/almacen/download/29/.../**filtros**.pdf

[6] www.eet460rafaela.edu.ar/descargar/apunte/392

[7] www.electro-tech-online.com/custompdfs/2010/11/sloa088.pdf

[8] www.dspace.espol.edu.ec/.../%255B**matlab**%255D_**MATLAB**_GUIDE.pdf