

Universidad Central “Marta Abreu” de Las Villas

Facultad de Ingeniería Eléctrica

Departamento de Automática y Sistemas Computacionales



TRABAJO DE DIPLOMA

**Evaluación mediante análisis de Monte Carlo de
filtros activos basado en redes doble T**

Autor: Roberto Daniel Céspedes Sierra

rcespedes@uclv.edu.cu

Tutor: Dr. Rubén Orozco Morales

rorozco@uclv.edu.cu

**Tesis presentada en opción al Título Académico de Ingeniero
en Telecomunicaciones y Electrónica**

Santa Clara

2015

“Año 57 de la Revolución”

Universidad Central “Marta Abreu” de Las Villas

Facultad de Ingeniería Eléctrica

Departamento de Automática y Sistemas Computacionales



TRABAJO DE DIPLOMA

**Evaluación mediante análisis de Monte Carlo de
filtros activos basado en redes doble T**

Autor: Roberto Daniel Céspedes Sierra

rcespedes@uclv.edu.cu

Tutor: Dr. Rubén Orozco Morales

rorozco@uclv.edu.cu

**Tesis presentada en opción al Título Académico de Ingeniero
en Telecomunicaciones y Electrónica**

Santa Clara

2015

“Año 57 de la Revolución”



Hago constar que el presente trabajo de diploma fue realizado en la Universidad Central “Marta Abreu” de Las Villas como parte de la culminación de estudios de la especialidad de Ingeniería en Automática, autorizando a que el mismo sea utilizado por la Institución, para los fines que estime conveniente, tanto de forma parcial como total y que además no podrá ser presentado en eventos, ni publicados sin autorización de la Universidad.

Firma del Autor

Los abajo firmantes certificamos que el presente trabajo ha sido realizado según acuerdo de la dirección de nuestro centro y el mismo cumple con los requisitos que debe tener un trabajo de esta envergadura referido a la temática señalada.

Firma del Autor

Firma del Jefe de Departamento
donde se defiende el trabajo

Firma del Responsable de
Información Científico-Técnica

PENSAMIENTO

"¿Por qué esta magnífica tecnología científica, que ahorra trabajo y nos hace la vida más fácil nos aporta tan poca felicidad? La respuesta es esta, simplemente: porque aún no hemos aprendido a usarla con tino."

Albert Einstein

DEDICATORIA

A mis padres, a mi familia y amigos y a todas las personas que de una forma u otra influyeron en parte de este sueño.

AGRADECIMIENTOS

A mi tutor Rubén Orozco Morales y a la profesora Niana.

TAREA TÉCNICA

- Realización de un estudio de los métodos actuales de análisis de Filtros activos.
- Familiarización con las herramientas computacionales que permiten el diseño y análisis de filtros activos.
- Utilización de MATLAB simbólico para la obtención de las funciones de red de Filtros activos diseñados previamente.
- Simulación de los diseños realizados en las diferentes herramientas computacionales y comparación de los resultados alcanzados.
- Obtención en las simulaciones y análisis en cuanto a la variación de la tolerancia y sensibilidad en los componentes de los filtros diseñados.

Firma del Autor

Firma del Tutor

RESUMEN

La tolerancia y las variaciones de los valores nominales de los componentes de un circuito afectan sus respuestas y comprometen su estabilidad. En el mundo competitivo actual la confiabilidad de un dispositivo electrónico es un aspecto a no descuidar, dado que la tolerancia afecta tanto la calidad, como el costo de los circuitos electrónicos. Con el advenimiento de los algoritmos simbólicos puede simplificarse y sistematizarse el análisis de la sensibilidad para el caso de filtros activos basados en redes doble T lo cual se expone en este trabajo. Se da a conocer la forma de obtener, por métodos simbólicos la función de red de una sección de segundo orden de un filtro activo, cómo determinar los parámetros de esta en función de los componentes del circuito y las expresiones de diseño. Se expone la manera de calcular la sensibilidad de los parámetros de la función de red tanto por métodos simbólicos como por el método de Monte Carlo y peor caso empleando el software Multisim. Se demuestra la gran potencialidad de los métodos de computación simbólica de MATLAB para el análisis y diseño de filtros, lo cual es validado mediante Multisim

Palabras claves: Monte Carlo, *Matlab*, doble T, sensibilidad, función de red, filtros activos, TINA, Multisim.

TABLA DE CONTENIDOS

PENSAMIENTO	i
DEDICATORIA	ii
AGRADECIMIENTOS	iii
TAREA TÉCNICA	iv
RESUMEN	v
INTRODUCCIÓN	1
Organización del informe	3
CAPÍTULO 1. REFERENTES TEÓRICOS ACERCA DEL DISEÑO DE REDES BASADAS EN DOBLE T	5
1.1 Utilizaciones prácticas de los filtros activos que emplean redes doble T	5
1.2 Configuración de las redes doble T	6
1.3 Funciones de red de la configuración de circuito	8
1.4 Pasos para el diseño de filtros activos basados en redes doble T	12
1.5 Herramientas computacionales para el diseño y análisis de filtros	13
1.6 Sensibilidad	17
1.7 Análisis de Monte Carlo	19
Conclusiones del capítulo	21
CAPÍTULO 2. PROCEDIMIENTO PARA EL DISEÑO DE REDES BASADAS EN DOBLE T	23
2.1 Diseño de filtros	23
2.1.1 Aproximaciones	24
2.1.2 Escoger una configuración circuital.....	26

2.2	Diseño del experimento.....	27
2.3	Especificaciones	30
2.3.1	MATLAB.....	30
2.3.2	Tina	31
2.3.3	Multisim.....	32
2.3.4	Tipo de PC	33
	Conclusiones del capítulo	33
CAPÍTULO 3. EVALUACIÓN MEDIANTE LOS MÉTODOS DE DISEÑO DE HERRAMIENTAS CAD		34
3.1	Primera configuración circuital.....	34
3.2	Segunda configuración circuital	40
3.3	Tercera configuración circuital	45
	Conclusiones parciales del capítulo.....	49
CONCLUSIONES Y RECOMENDACIONES		51
	Conclusiones	51
	Recomendaciones	52
REFERENCIAS BIBLIOGRÁFICAS		53
ANEXOS		56
Anexo I	Programas realizados en MATLAB.....	56
Anexo II	Circuitos montados y simulados	66

INTRODUCCIÓN

Debido a la tolerancia, el envejecimiento, la temperatura, y otros factores, el valor de los componentes de un filtro activo dista del valor exacto necesario para brindar determinada respuesta; como resultado, la verdadera respuesta de un filtro se desvía de la que tendría si empleara componentes exactos todo el tiempo. Como una forma de predecir tal desviación los ingenieros utilizan el concepto de sensibilidad, siendo una figura de mérito de cada configuración de circuito utilizada como elemento de comparación entre distintas redes a la hora de seleccionar la más adecuada para un fin determinado, o para saber qué componentes deben seleccionarse con mejor calidad pues influyen más en el desempeño esperado.

Los estudios de la sensibilidad están muy vinculados a los análisis de tolerancia en los filtros eléctricos. Estos estudios tomaron auge durante los años setenta y ochenta del siglo XX, pero en ese entonces no se disponía de las modernas herramientas computacionales actuales, las cuales han hecho retomar el tema con nuevos enfoques y posibilidades, principalmente los que ofrecen los métodos simbólicos de programación que comienzan a ganar popularidad en la primera década del siglo XXI.

Consumar el diseño de un filtro incluye garantizar, con mínimo costo de producción, todas sus respuestas, tanto de frecuencias como transitorias. Debe tenerse en cuenta que después de escoger la configuración apropiada para el circuito del filtro, lo antes mencionado está muy estrechamente relacionado con el precio de los componentes en el mercado, pues se fabrican con un valor de venta medio para su principal parámetro (ej. resistencia o capacitancia) y cierta tolerancia. Menor tolerancia requiere mayor seguridad tecnológica durante su producción, lo cual involucra incremento en el costo del componente y viceversa.

Dado que el diseño ingenieril debe procurar la utilización de componentes tan económicos como se pueda sin comprometer el desempeño del circuito, ello obliga a decidir cuales componentes son

críticos y cuan baja es la tolerancia que estos requieren. Tal decisión es posible solo a través de un riguroso análisis de sensibilidad acompañado de análisis de tolerancia. En consecuencia, el problema relacionado con sensibilidades y tolerancias se convierte necesario en el desarrollo de filtros activos, principalmente si estos serán producidos masivamente en serie. Así las cosas, solo los componentes críticos serán de alta calidad y costosos, mientras que para los no críticos pueden destinarse los menos costosos y de mayor tolerancia. De esta manera se minimiza el costo al tiempo que se evita un desempeño inesperado del filtro.

Aunque el análisis de Montecarlo es un método ampliamente aceptado para determinar el comportamiento estadístico de alguna respuesta de un circuito ante variaciones de sus componentes, actualmente no todas las herramientas de software lo implementan. Al mismo tiempo, dado el carácter estocástico con que se hacen variar los valores de los componentes dentro de un conjunto finito de alternativas, no existe garantía de que cuando varíen simultáneamente todos los componentes estos conduzcan a situaciones extremas en la respuesta estudiada, por lo que tampoco todas las herramientas computacionales realizan el análisis de casos extremos. Entre las herramientas que son capaces de realizar estos análisis se encuentra el *Multisim* de *National Instruments*.

Dentro de las configuraciones de filtros activos las redes T-punteadas (*Twin-T*) se caracterizan por ser capaces de realizar funciones con muy alto factor de calidad, lo que permite alcanzar elevada selectividad; no obstante, su sensibilidad depende de las relaciones entre sus componentes, por lo que los estudios de su sensibilidad se convierten en imprescindibles durante el proceso de diseño. De aquí que el problema científico de esta investigación sea: ¿Cómo influye en la respuesta de frecuencia de un filtro activo basado en doble T, la variación del valor de sus componentes?

Para dar respuesta a este problema se declara como objetivo general: Evaluar la sensibilidad de filtros activos basados en redes doble T mediante análisis de Monte Carlo y métodos simbólicos

Para alcanzar este propósito se plantean los siguientes objetivos específicos:

- Establecer los referentes teóricos acerca del diseño de redes basadas en doble T.
- Desarrollar el procedimiento para el diseño de filtros y la evaluación de su sensibilidad utilizando herramientas de computación simbólicas.
- Describir las técnicas de obtención de las ecuaciones de análisis de la sensibilidad de los filtros activos basados doble T utilizando herramientas de computación simbólicas.

- Determinar la sensibilidad de los parámetros a la variación de los valores de los componentes dentro de un rango de tolerancia.
- Utilizar herramientas CAD para verificar los métodos de diseño, así como para realizar análisis de Monte Carlo y caso extremo de los diseños realizados que validen los análisis de sensibilidad teórica.

Con este proyecto se pretende contribuir al desarrollo de métodos de análisis y síntesis de filtros activos basados en doble T utilizando modernas herramientas computacionales que hacen uso de métodos simbólicos y el diseño asistido mediante computadoras.

Con la ejecución del proyecto se dan soluciones a problemáticas modernas vinculadas con la adquisición de circuitos integrados de gran valor cuya adquisición no es viable actualmente en nuestro país. Además de contribuir al ahorro de la economía mediante una compra efectiva de los componentes del circuito; así como determinar cuáles son los componentes que más se ajustan al circuito en concreto.

Los resultados de la investigación poseen una aplicación práctica y teórica de gran trascendencia para todos los especialistas, investigadores y diseñadores puesto que en menor tiempo se puede determinar cuáles son los componentes que más se ajustan al circuito y cuáles son los que menos variación producen del resultado óptimo teniendo cierto margen de tolerancia, además se logra recopilar los resultados de una serie de publicaciones permitiendo así un fácil acceso a la información respecto al tema. Además los procedimientos utilizados son extensibles a otros tipos de circuitos.

Organización del informe

El informe está conformado por un capítulo 1 donde se hace una revisión bibliográfica sobre las redes de tipo doble T, los métodos de aproximación y realización a base de estas redes, las propiedades de las funciones de red, así como los fundamentos de los análisis de la sensibilidad aplicados a estas redes. El capítulo 2 expone qué redes y variantes serán analizadas, con qué herramientas computacionales se realizan los análisis, establece el porqué de las realizaciones y pruebas que se efectúan, exponiendo el cómo realizar los diseños y su verificación por medio de las herramientas computacionales que se utilizan. El capítulo 3 recoge los resultados de los análisis realizados, los que incluyen la verificación de la respuesta utilizando componentes exactos y

componentes con un margen de tolerancia para evaluar el comportamiento de caso extremo, acompañado de análisis de Monte Carlo, con lo cual se verifica la sensibilidad de cada circuito. El incluye además las conclusiones del trabajo realizado, recomendaciones para trabajos futuros, la bibliografía utilizada, así como varios anexos con variantes de circuitos basados en doble T y códigos en MATLAB.

.

CAPÍTULO 1. REFERENTES TEÓRICOS ACERCA DEL DISEÑO DE REDES BASADAS EN DOBLE T

Las redes de realimentación *Twin-T* están compuestas por tres resistores y tres capacitores conectados entre la entrada y la salida de uno o dos amplificadores operacionales, siendo populares para la realización de filtros supresores de banda muy estrecha o de hendidura (*notch*) y con el atractivo de que toda la red puede ser implementada con un solo valor de resistor y de capacitor. En este capítulo se aborda el procedimiento de análisis y realización de diversas configuraciones de redes doble T utilizando herramientas computacionales de mucha actualidad

1.1 Utilizaciones prácticas de los filtros activos que emplean redes doble T

Dada la elevada selectividad de las redes doble T que les permite realizar redes supresoras de banda muy estrecha (*notch*), una de sus principales aplicaciones es la supresión de tonos o interferencias, como lo puede ser la señal de alimentación (60 Hz) inducida en la electrónica de diversos dispositivos, por ejemplo equipos de registro de señales biomédicas. Dentro de estas señales se encuentran las electrocardiográficas, las electroencefalográficas, electromiográficas y otras cuyos anchos de banda incluyen los 60 Hz. En estos casos se reporta que la relación señal a ruido puede alcanzar hasta los +3 dB. A pesar de que mediante procesamiento digital pueden implementarse muchas estrategias de supresión de ruido, parte del acondicionamiento de la señal pre-muestreo debe incluir mejorar al máximo la relación señal a ruido, por lo que la literatura sigue reportando el uso de redes activas para este fin. (Bai, Cheng et al. 2003)

Otro ejemplo que verifica la utilización de filtros doble T es cuando se emplea al eliminar la banda de frecuencias que queda entre un filtro paso alto y un filtro paso bajo; pues al ser el punto de ruptura de un filtro de paso alto considerablemente superior a la del filtro de paso bajo, queda una banda de frecuencias entre los dos que debe ser rechazada, por lo que se debe usar tres

amplificadores operacionales como mínimo ,pero en realidad es bastante estrecha la banda de frecuencias que debe ser rechazada ,y la topología de tres amplificadores operacionales, no funcionará para estrechar frecuencias espaciadas , por lo que deja que se empleen filtros de corte para hacer este trabajo, ya que con las muescas ligeramente desplazados en frecuencia forman una respuesta que es más ancha; pero cada muesca toma dos amplificadores operacionales y significaría cuatro amplificadores operacionales como mínimo. A pesar de sus limitaciones, esto hace que la topología de doble T sea el único camino viable para hacer un solo amplificador que amplíe la banda de filtro de rechazo. Una banda muy común de rechazo de la aplicación del filtro es deshacerse de 50 y 60 Hz, línea de zumbido de los productos que serán comercializados en Europa y Estados Unidos.(Bai, Cheng et al. 2003)

1.2 Configuración de las redes doble T

Las redes de realimentación doble T son fácilmente sintonizable y proporcionan propiedades de sensibilidad relativamente favorables a los filtros pasivos. La topología de doble T utiliza uno o dos amplificadores operacionales. Se basa en una red RC pasiva que utiliza tres resistencias y tres capacitores, como muestra la figura 1.1. Para aumentar Q el filtro pasivo se implementa en el bucle de realimentación de un amplificador convirtiéndose así en un filtro de rechazo de banda activa figura 1.2. La coincidencia de estos seis componentes pasivos es fundamental; afortunadamente, también es fácil. Toda la red se puede construir a partir de un único valor de resistencia y un único valor de la capacitancia, pues el resistor de valor $R/2$ y el capacitor de valor $2C$ se logran poniendo en paralelo dos resistores de valor R y C respectivamente ya que los componentes de un mismo lote es más probable que tengan características muy similares.(Thomas, Haddad et al. 1998)

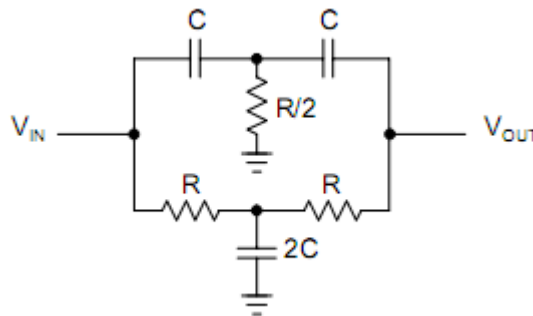
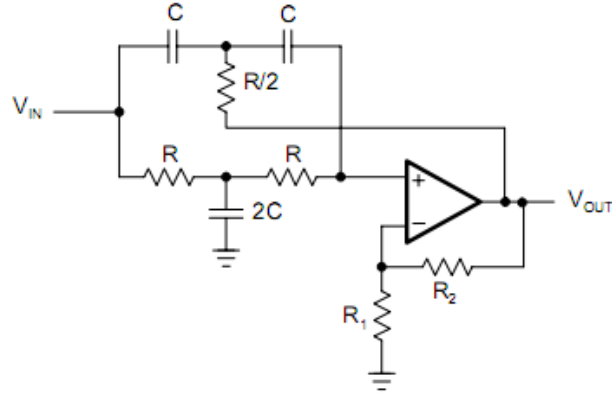


Figura 1.1 Filtro Pasivo *Twin-T*

Figura1.2 Filtro Activo *Twin-T*

La función de transferencia del filtro de doble T activo es:

$$A(s) = \frac{k(1 + s^2)}{1 + 2(2 - k)s + s^2} \quad (1.1)$$

$$A(s) = \frac{A(1+s^2)}{1+\frac{1}{Q}s^2+s^2} \quad (1.2)$$

La comparación de las variables de la ecuación con la ecuación 1.1 y 1.2 proporciona la ecuación que determinan los parámetros de filtro:

$$\text{Frecuencia media: } f_m = \frac{1}{2\pi RC} \quad (1.3)$$

$$\text{Ganancia interna: } G = 1 + \frac{R_2}{R_1} \quad (1.4)$$

$$\text{Ganancia de banda de paso : } A_0 = G \quad (1.5)$$

$$\text{Factor de calidad: } Q = \frac{1}{2(2-G)} \quad (1.6)$$

El circuito de doble T tiene la ventaja de que el factor de calidad (Q) se puede variar a través de la ganancia interna (G) sin modificar la frecuencia media (f_m). Sin embargo, Q y A_m no se pueden ajustar de forma independiente. Para ajustar la frecuencia media de la banda de paso, fijamos f_m y C , y luego resolver para R . (Carter 2000)

$$R = \frac{1}{2\pi f m C} \quad (1.7)$$

Debido a la dependencia entre Q y A_m , hay dos opciones para resolver para R_2 :

Ajustar la ganancia a frecuencias medias, según:

$$R_2 = (A_0 - 1)R_1 \quad (1.8)$$

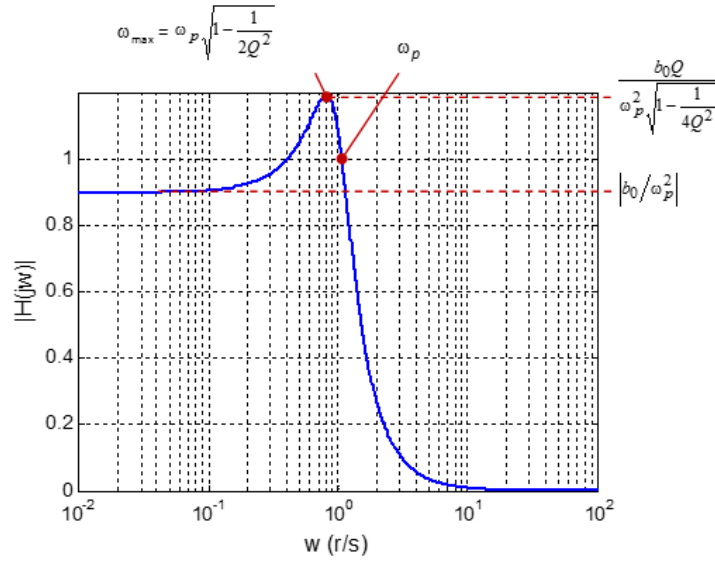
O diseñar una Q

$$R_2 = R_1 \left(1 - \frac{1}{2Q}\right) \quad (1.9)$$

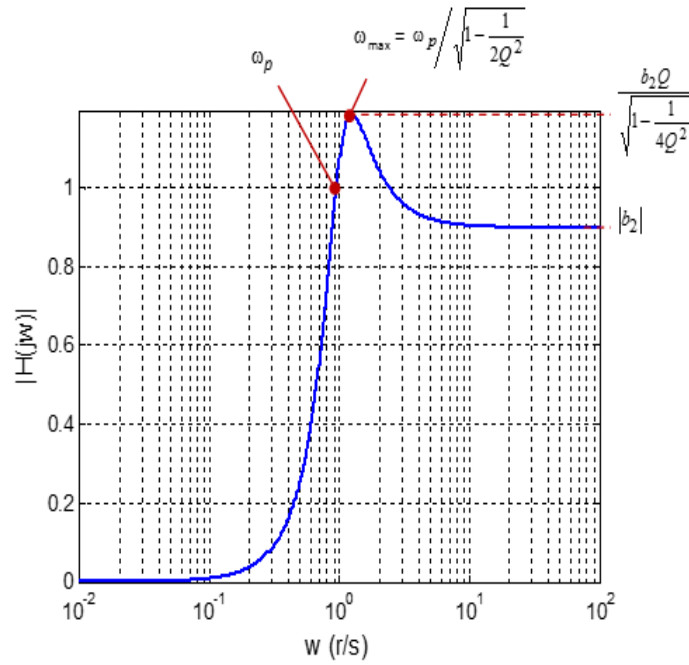
1.3 Funciones de red de la configuración de circuito

Las redes doble T permiten realizar funciones de red de tipo pasobajo (LP), pasoalto (HP), pasobanda (BP) y supresora de banda (BR), obtenidas por medio de diversos procesos de aproximación. Estos procesos de aproximación devuelven funciones de red cuya forma depende del tipo de respuesta de frecuencias. Para el caso de secciones de segundo orden la forma de la función de red adopta alguna de las expresiones que se relacionan a continuación, dependiendo las propiedades de la respuesta de frecuencias de los parámetros de la función como se indica en cada caso. (Moschytz 1999)

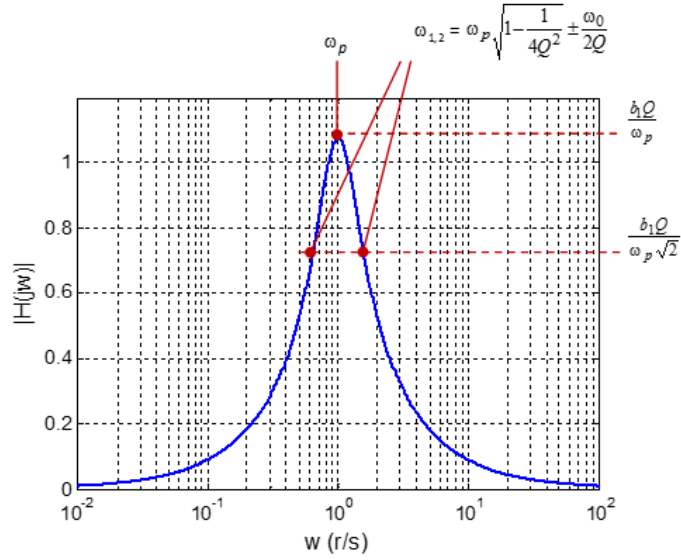
Función de red Pasobajo (LP). $H(s) = \frac{b_0}{s^2 + \frac{\omega_p}{Q}s + \omega_p^2}$.



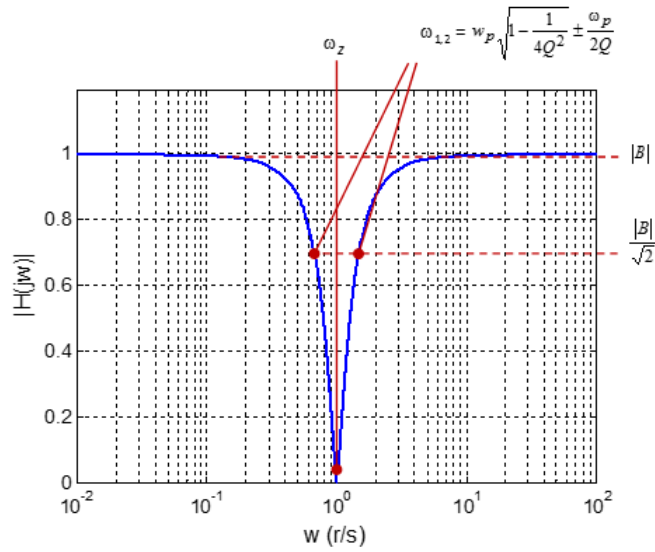
Función de red Pasoalto (HP). $H(s) = \frac{b_2 s^2}{s^2 + \frac{\omega_p}{Q}s + \omega_p^2}$.



Función de red Pasobanda (BP). $H(s) = \frac{b_1 s}{s^2 + \frac{\omega_p}{Q} s + \omega_p^2}$.

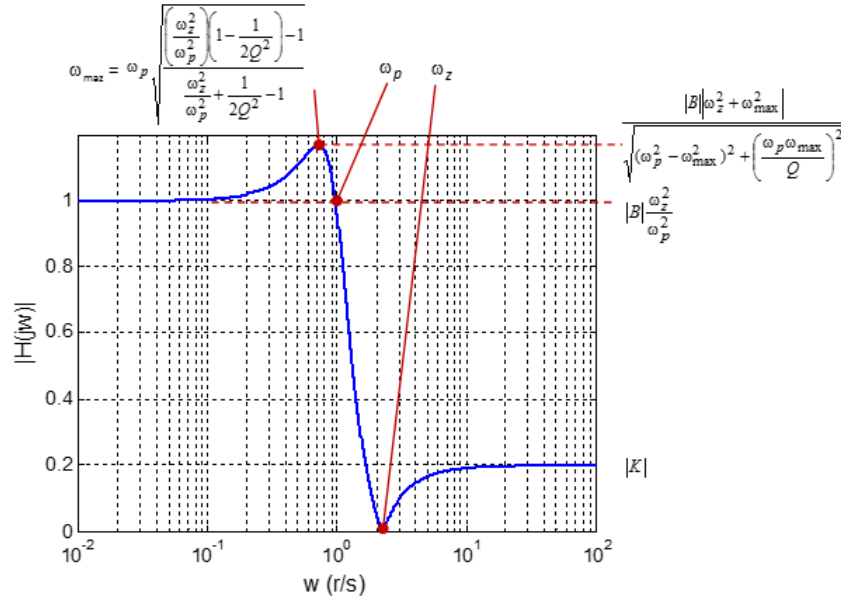


Función de red Notch. $H(s) = \frac{B(s^2 + \omega_z^2)}{s^2 + \frac{\omega_p}{Q} s + \omega_p^2}$.



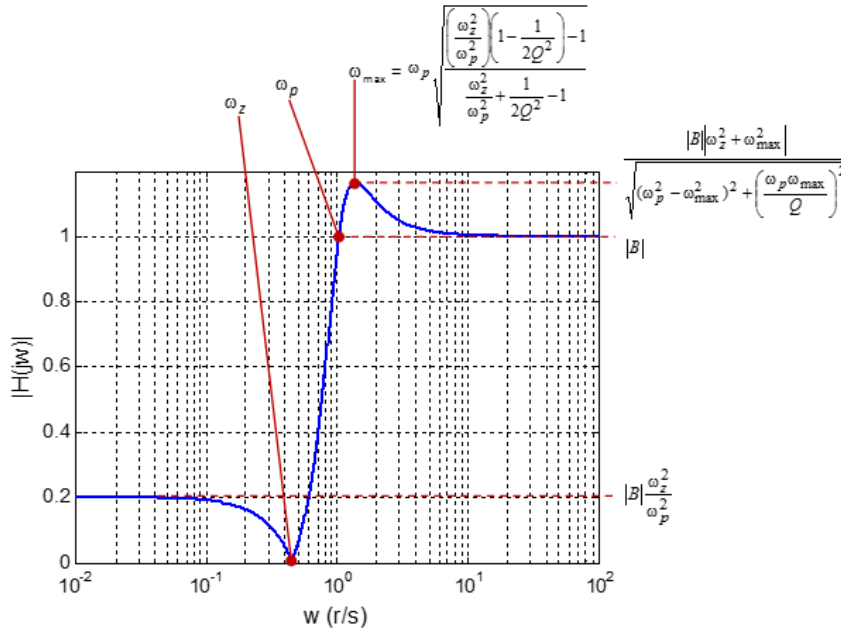
Función de red Pasobajo Notch (LPN)

$$H(s) = \frac{B(s^2 + \omega_z^2)}{s^2 + \frac{\omega_p}{Q}s + \omega_p^2}.$$



Función de red Pasoalto Notch (HPN).

$$H(s) = \frac{B(s^2 + \omega_z^2)}{s^2 + \frac{\omega_p}{Q}s + \omega_p^2}.$$



1.4 Pasos para el diseño de filtros activos basados en redes doble T

Todos los dispositivos digitales modernos (ej. celulares, MP3, etc.) utilizan filtros de señal analógica previo a la conversión A/D y/o posterior a la conversión D/A.

Las funciones de aproximación para filtros analógicos permiten, a partir de unas especificaciones de diseño deseadas, obtener la función de red, $H(s)$, a realizar posteriormente mediante un circuito eléctrico.

Existen varios software que tienen programados diversos métodos de aproximación, por lo que el problema actual de los ingenieros es dominar la esencia del proceso de aproximación y saber las cualidades de cada uno de los métodos disponibles.

Con la normalización del eje de frecuencias, el problema de la aproximación se sistematiza en los pasos siguientes:

I. Definir las especificaciones (plantilla) de diseño.

- Tipo de respuesta de magnitud – frecuencias, valores de las frecuencias esquinas y magnitud deseada en ellas.

II. Normalizar el eje de frecuencias.

- Obtener la plantilla prototipo pasobajo.

III. Cálculo de la función de red prototipo.

- Obtener una $H(s_x)$ que satisfaga las especificaciones del filtro prototipo.

IV. Desnormalización de frecuencias.

- Obtener la $H(s)$ que responda al tipo de respuesta de frecuencias deseada.

Cualquier configuración de 2do orden puede ser utilizada realizando alguno o todos los pasos siguientes:

I. Plantear el sistema de ecuaciones de nodo del circuito.

II. Aplicar métodos simbólicos para obtener la función de red del circuito.

III. A partir de la función de red del circuito obtener las expresiones de su ω_p , Q , ω_z y K en función de los componentes.

- IV. Decidir cuales componentes del circuito son variables independientes (asumiendo sus valores) y cuales dependientes.
- V. Determinar por métodos simbólicos las expresiones de realización de los componentes que son variables dependientes en función de los valores de ω_p , Q , ω_z y K , así como de los componentes independientes. (Gielen, Walscharts et al. 1989, Kolev 2002, Raut and Swamy 2010)

1.5 Herramientas computacionales para el diseño y análisis de filtros

Se han desarrollado diversas herramientas computacionales para el análisis y diseño de filtros activos y aunque son confiables y excelentes para llevar a cabo los diseños utilizados en este trabajo, no todas son de acceso gratuito, por lo que usamos los software a nuestro alcance.

MATLAB

Dentro de las herramientas computacionales que más aceptación logran por sus posibilidades en computación simbólica se encuentran MATLAB y Mathematic.

MATLAB es un entorno de cálculo técnico de altas prestaciones para cálculo numérico y visualización. Integra:

- Análisis numérico.
- Cálculo matricial.
- Procesamiento de señales.
- Gráficos.

En un entorno fácil de usar, donde los problemas y las soluciones pueden expresarse como se escriben matemáticamente, sin la programación tradicional. El nombre MATLAB proviene de "*MATrix LABoratory*" (Laboratorio de Matrices). MATLAB fue escrito originalmente para proporcionar un acceso sencillo al software matricial desarrollado por los proyectos *LINPACK* y *EISPACK*, que juntos representan lo más avanzado en programas de cálculo matricial. MATLAB es un sistema interactivo cuyo elemento básico de datos es una matriz que no requiere dimensionamiento. Esto permite resolver muchos problemas numéricos en una fracción del tiempo que llevaría hacerlo en lenguajes como *C*, *BASIC* o *FORTRAN*. MATLAB ha evolucionado en los

últimos años a partir de la colaboración de muchos usuarios. En entornos universitarios se ha convertido en la herramienta de enseñanza estándar para cursos de introducción en álgebra lineal aplicada, así como cursos avanzados en otras áreas. En la industria, MATLAB se utiliza para investigación y para resolver problemas prácticos de ingeniería y matemáticas, con un gran énfasis en aplicaciones de control y procesamiento de señales. MATLAB también proporciona una serie de soluciones específicas en diversas cajas de herramientas (*TOOLBOXES*). Estas son muy importantes para la mayoría de los usuarios de MATLAB ya que extienden el entorno para resolver clases particulares de problemas como:

- Procesamiento de señales.
- Diseño de sistemas de control.
- Simulación de sistemas dinámicos.
- Identificación de sistemas.
- Redes neuronales y otros.

Entre estas funciones se encuentran:

SIGNAL PROCESSING TOOLBOX

MATLAB tiene una gran colección de funciones para el procesamiento de señal en el Signal Processing Toolbox. Este incluye funciones para:

Análisis de filtros digitales incluyendo respuesta en frecuencia, retardo de grupo, retardo de fase.

Implementación de filtros, tanto directo como usando técnicas en el dominio de la frecuencia basadas en la (transformada rápida de Fourier) FFT.

Diseño de filtros IIR (*Infinite Impulse Response* o *Respuesta infinita al impulso*), incluyendo Butterworth, Chebyshev tipo I, Chebyshev tipo II y elíptico.

Diseño de filtros FIR (*Finite Impulse Response* o *Respuesta finita al impulso*) mediante el algoritmo óptimo de Parks-McClellan.

Procesamiento de la FFT, incluyendo la transformación para potencias de dos y su inversa, y transformada para no potencias de dos. (Toolbox 1999)

SYMBOLIC MATH TOOLBOX

El *Symbolic Math Toolbox* sale al mercado desde la versión 6 de MATLAB al introducir las capacidades de programación sintáctica de *Maple*, pero ampliando las posibilidades de este con las numéricas y gráficas que distinguen a MATLAB. En 2008 *Mathworks* compró *MuPAD*, siendo a partir de entonces la caja de herramientas simbólicas de MATLAB. Actualmente el *Symbolic Math Toolbox* se encuentra en su versión 3, el cual además de poder resolver sistemas de ecuaciones, puede también realizar otros cálculos simbólicamente, tales como integración, diferenciación, simplificación de funciones, cálculo de transformadas de funciones (ej. Laplace y Fourier). Muchas de estas posibilidades son las que han permitido retomar el análisis de la sensibilidad y la tolerancia de configuraciones de filtros activos desde la poderosa y retadora perspectiva simbólica. El reto no radica en saber utilizar las herramientas simbólicas, sino en dominar los principios ingenieriles subyacentes en el problema donde esta se aplica. (Erdei and Horgoş 2007, Jurišić 2011)

Probablemente la característica más importante de MATLAB es su capacidad de crecimiento. Esto permite convertir al usuario en un autor contribuyente, creando sus propias aplicaciones. En resumen, las prestaciones más importantes de MATLAB son:

- Escritura del programa en lenguaje matemático.
- Implementación de las matrices como elemento básico del lenguaje, lo que permite una gran reducción del código, al no necesitar implementar el cálculo matricial.
- Implementación de aritmética compleja.
- Un gran contenido de órdenes específicas, agrupadas en *TOOLBOXES*.

Posibilidad de ampliar y adaptar el lenguaje, mediante ficheros de script y funciones .m. (Kolev 2002)

Multisim

Multisim es una aplicación de captura esquemática y simulación del conjunto de diseño de circuito de National Instruments, un conjunto de herramientas de EDA (Electronics Design Automation) que asisten en los principales pasos para el diseño del circuito, aunque limitado a muy pocas configuraciones de filtros, por lo que no incluye las redes doble T. Incorpora SPICE3F5 y XSPICE al núcleo de su motor de simulaciones, con mejoras diseñadas por Electronics Workbench específicamente para optimizar el desempeño de los análisis con simulaciones digitales y mixtas.

SPICE3F5 y XSPICE son estándares aceptados en la industria de domino público. SPICE3F5 es una edición de SPICE (Simulation Program with Integrated Circuit Emphasis) diseñada por la Universidad de California en Berkeley. XSPICE es un set de mejoras únicas hechas a SPICE, bajo contrato de las Fuerzas Armadas de Estados Unidos, que incluye un modo mixto de simulación evento-consecuencia y un subsistema extensible de modelaje de usuario final. Electronic Workbench ha mejorado los núcleos con algunas características de compatibilidad no SPICE – estándar PSPICE, para permitir el uso de un amplio rango de modelos atípicos de SPICE.

Multisim también ofrece una amplia gama de análisis, entre los que se encuentran Monte Carlo, Fourier, de ruido, de sensibilidad, del peor caso, etc. Para la comprobación y análisis del circuito, nos provee de varios instrumentos de medición virtuales. Estos se conectan, usan y leen igual que sus equivalentes reales. Usando instrumentos virtuales es la vía más fácil de examinar el comportamiento del circuito.(Luther and Rodriguez 2006, Báez-López, Guerrero-Castro et al. 2012)

Tina

Tina TI es un software realizado por Texas Instruments y Designsoft y una herramienta más ligera que Tina Design Suite. Es una potente herramienta de simulación de circuitos que es ideal para la simulación de circuitos analógicos y circuitos de fuente conmutadas. Este software basado en simulador SPICE tiene una poderosa capacidad de análisis, una intuitiva y simple interfaz gráfica.

TINA TI es un poderoso y costeable paquete de software para analizar, diseñar y simulación a tiempo real de circuitos analógicos, digitales, VHDL y mixtos. Se puede también analizar radiofrecuencias, comunicaciones y circuitos optoelectrónicos, y probar y depurar aplicaciones de microprocesadores y microcontroladores. Para los ingenieros Tina es un software de uso fácil y una herramienta de alto rendimiento, mientras que los educadores lo optan por sus características únicas para el ambiente favorable a la enseñanza. Empezando v7, Tina está distribuida en dos versiones mayores – Tina v7 y Tina Design Suite v7. Tina v7 incluye solo simulación de circuitos, mientras que Tina Design Suite v7. Este módulo de trazado integrado tiene todas las características que se necesitan para el diseño avanzado de PCB, incluyendo un Bloque de Control de Procesos multicapas. Tina TI puede ser también utilizada en su ambiente de entrenamiento. Este incluye una herramienta única para hacer probar el conocimiento de los estudiantes, monitoreando el progreso e introduciendo técnicas de localización de fallas.(Quick, Pipes et al. 1995)

1.6 Sensibilidad

Debido a la tolerancia, el envejecimiento, la temperatura y otros factores, el valor de los componentes de un filtro activo dista del valor exacto necesario para brindar determinada respuesta; como resultado, la verdadera respuesta de un filtro se desvía de la que tendría si empleara componentes exactos todo el tiempo. Como una forma de predecir tal desviación los ingenieros utilizan el concepto de sensibilidad, siendo una figura de mérito de cada configuración de circuito utilizada como elemento de comparación entre distintas redes a la hora de seleccionar la más adecuada para un fin determinado, o para saber qué componentes deben seleccionarse con mejor calidad pues influyen más en el desempeño esperado.

Consumar el diseño de un filtro incluye garantizar, con mínimo costo de producción, todas sus respuestas, tanto de frecuencias como transitorias. Debe tenerse en cuenta que después de escoger la configuración apropiada para el circuito del filtro, lo antes mencionado está muy estrechamente relacionado con el precio de los componentes en el mercado, pues se fabrican con un valor de venta medio para su principal parámetro (ej. resistencia o capacitancia) y cierta tolerancia. Menor tolerancia requiere mayor seguridad tecnológica durante su producción, lo cual involucra incremento en el costo del componente y viceversa.

Adoptar una posición filosóficamente opuesta, en la que se acepte que los componentes puedan tener cualquier tolerancia y entonces traten de encontrar configuraciones de circuitos que sean lo más insensibles posible a las variación de los componentes no ha llevado hasta ahora a resultados aceptables.(Erdei and Horgoş 2007, Mandache, IORDACHE et al. 2010)

Definición

Supongamos una característica P de un circuito que depende de un cierto parámetro x . Dicha característica dependerá también, normalmente, de la frecuencia s por lo que se puede poner $P = P(s, x)$.

La desviación en P causada por un error $dx = x - x_0$ de x se puede obtener mediante el desarrollo de Taylor alrededor del valor nominal x_0 :

$$P(s, x) = P(s, x_0) + \left. \frac{\partial P(s, x)}{\partial x} \right|_{x_0} dx + \frac{1}{2} \left. \frac{\partial^2 P(s, x)}{\partial x^2} \right|_{x_0} (dx)^2 + \dots \quad (1.10)$$

Si asumimos que $\frac{dx}{x_0} \ll 1$ y que la curvatura de $P(s, x)$ cerca de x_0 no es demasiado grande, podemos despreciar los términos del desarrollo a partir de la segunda derivada, con lo que:

$$\Delta P(s, x_0) = P(s, x_0 + dx) - P(s, x_0) \approx \left. \frac{\partial P(s, x)}{\partial x} \right|_{x_0} dx \quad (1.11)$$

La mayor parte de las veces no se está tan interesado en cambios absolutos (ΔP) originados por cambios de x ($dx = x - x_0$) sino en los cambios relativos, es decir:

$$\frac{\Delta P(s, x_0)}{P(s, x_0)} \approx \frac{x_0}{P(s, x_0)} \left. \frac{\partial P(s, x)}{\partial x} \right|_{x_0} \frac{dx}{x_0} \quad (1.12)$$

donde:

$$S_x^P = \frac{x_0}{P(s, x_0)} \left. \frac{\partial P(s, x)}{\partial x} \right|_{x_0} \frac{\partial P/P}{\partial x/x} \bigg|_{x_0} = \left. \frac{d(\ln P)}{d(\ln x)} \right|_{x_0} \quad (1.13)$$

es la sensibilidad de P a un pequeño cambio en x .

Por tanto la sensibilidad se define:

$$S_x^P = \frac{x}{P} \frac{\partial P(s, x)}{\partial x} = \left. \frac{d(\ln P)}{d(\ln x)} \right|_{x_0} \quad (1.14)$$

Para pequeñas variaciones de x :

$$\frac{\Delta P}{P} = S_x^P \frac{\Delta x}{x} \quad (1.15)$$

que nos indica la variabilidad (cambio porcentual o relativo de P debido a un determinado cambio porcentual o relativo de x).

Se observa por tanto, que los mejores circuitos serán aquellos que presenten menores sensibilidades a sus componentes. Si S es pequeña podrán utilizarse componentes con tolerancias mayores sin que ello produzca grandes variaciones de la respuesta deseada ($\frac{\Delta P}{P}$) y dichos componentes serán más baratos. (Erdei and Horgoş 2007, Yang 2007, Palma-Rodriguez, Tlelo-Cuautle et al. 2012)

1.7 Análisis de Monte Carlo

El análisis de Monte Carlo hace variar aleatoriamente dentro, de un intervalo, a los valores de los componentes de interés en el circuito y determina el conjunto de respuestas que estas variaciones producen. Con ello puede realizarse un análisis estadístico de las respuestas logradas, pudiendo determinarse el comportamiento medio, las respuestas extremas.

Con el advenimiento de herramientas computacionales como el MATLAB simbólico puede realizarse este análisis evitando posibles errores y facilitando el trabajo, lo cual permite reservar el mayor esfuerzo a la interpretación de los resultados.

Análisis de Monte Carlo de la respuesta de magnitud-frecuencia para variaciones de componentes de un circuito.

La sensibilidad de un parámetro de la función de red de un circuito (ej. ω_z , ω_p , Q y G) a la tolerancia de alguna componente, en la forma vista anteriormente, devuelve un valor numérico que no expresa cómo varían las respuestas de interés del filtro (ej. respuestas de frecuencia o transitorias) ante la tolerancia de alguno o varios componentes en caso de que puedan modificar sus valores cualquiera de estos simultáneamente. Es por ello que ha sido una práctica el análisis de Monte Carlo de las respuestas del filtro ante las variaciones de los valores sus componentes a causa de la tolerancia.(Moschytz 1999)

El análisis de Monte Carlo, también denominado en la literatura en lengua inglesa como *Statistic Analogical Experiment Analysis* o también *Random Analogy Analysis* hace variar aleatoriamente dentro, de un intervalo, a los valores de los componentes de interés en el circuito y determina el conjunto de respuestas que estas variaciones producen. Por tanto, es un método estadístico de análisis que calcula la respuesta de un circuito cuando los parámetros del mismo varían aleatoriamente. Para ello utiliza múltiples cálculos numéricos de la respuesta a analizar basándose en la teoría de las probabilidades para asignar valores a los componentes del circuito de acuerdo a alguna distribución estadística que se especifique y dentro de su margen de tolerancia. Con el análisis de Monte Carlo puede realizarse un análisis estadístico de las respuestas logradas, pudiendo determinarse el comportamiento medio, las respuestas extremas, confeccionar histogramas de los resultados, etc. No obstante, su seguridad está limitada por el tiempo de cálculo o por el monto de variantes a calcular, mientras más variantes más confianza en los resultados, por

lo que generalmente se efectúan cientos o miles de corridas para poder arribar a resultados con mayor nivel de confianza.(Erdei and Horgoş 2007, Fitzpatrick 2011)

El análisis de Monte Carlo forma parte de las opciones de muchas herramientas para simulación y análisis de circuitos como *Tina-TI* y *PSPice (MultiSim)*.

Muy importante es saber relacionar el resultado de un cálculo teórico de la sensibilidad con el resultado práctico que debe producir un circuito. El análisis de Monte Carlo generalmente se realiza variando aleatoriamente uno o varios componentes pasivos del circuito; no obstante, puede aplicarse cuando se desea conocer las variaciones que se producen en un parámetro de la respuesta del circuito ante las variaciones de alguno de sus componentes.

El análisis de Monte Carlo es un método costoso pues se basa en múltiples simulaciones que se realizan repetidamente. Ello se debe a que es un método basado en la estadística, lo cual hace que necesite de varios cientos, o tal vez miles, de muestras para arribar a resultados estadísticamente aceptables. Mientras más simulaciones mayor precisión en los resultados, principalmente cuando se desean determinar parámetros estadísticos de las respuestas obtenidas, tales como valor medio, varianza, etc. Teóricamente, si N es el número de simulaciones o respuestas del filtro ante N combinaciones de variaciones aleatorias de sus componentes, las N respuestas obtenidas definen el área esperada de la verdadera respuesta.(Hayes, Tadeo et al.)

Es por ello que el análisis de Monte Carlo demanda un alto esfuerzo computacional y tiempo de cálculo, lo cual ha conducido a que recientemente se hayan encontrado alternativas más eficientes basadas en el análisis de valores extremos de tolerancia (*WCTA: worst-case tolerance analysis*), los diagramas de decisión determinante (*DDD: Determinant Decision Diagrams*)(Palma-Rodriguez, Tlelo-Cuautle et al. 2012) o los algoritmos simbólicos (Mandache, IORDACHE et al. 2010) destinándose fundamentalmente el análisis de Monte Carlo para condiciones en los que no se dispone de un modelo matemático del circuito (Hongliang) No obstante, actualmente muchas herramientas para simulación y análisis de circuitos (ej. *Multisim*, *Tina-TI*, etc.). Ofrecen la posibilidad del análisis de Monte Carlo. Por ejemplo, para realizar el análisis de la sensibilidad de un parámetro (ej. f_0) a una componente individual debe asignarse diversos valores a esta (como parámetro variable) y determinar para cada valor de la componente el valor medio del parámetro en el análisis de Monte Carlo para luego calcular. (Raut and Swamy 2010)

En particular el método *WCTA*, que busca determinar los márgenes de tolerancia en los componentes que aseguran que los límites máximo y mínimo de una respuesta del filtro no sobrepase determinado margen de desempeño, adolece de la incertidumbre debida a que es difícil predecir cuál es la combinación de valores extremos de los componentes que conduce al peor caso en el desempeño de un circuito, principalmente cuando el *WCTA* se acomete mediante análisis de Monte Carlo (Kolev 2002) . La razón de ello radica en que los análisis de peor caso se sustentan en la idea de que si todos los componentes del circuito asumen simultáneamente sus valores máximos y mínimos (extremos) dentro de sus respectivas tolerancias es posible que las respuestas logradas sean el peor caso. Como quiera que sea, encontrar las condiciones para la peor variante es todo un reto. Al efecto se han publicado algunos trabajos como los de (Hayes, Tadeo et al.) basados en el uso de valores singulares estructurados a partir de la teoría del control robusto que tiene la ventaja de determinar el peor caso basado en repeticiones pero no de forma estadísticas, sino en un procedimiento equivalente a los análisis de estabilidad de sistemas. Si bien el método tiene su potencialidad requiere de sólidos conocimientos de teoría del control y de la aplicación de espacios de estado para su aplicación

Conclusiones del capítulo

En este capítulo se llevó a cabo una recopilación de información importante de los principales aspectos relacionados con las redes doble T dándose a conocer algunas de sus aplicaciones, las funciones de red de secciones de segundo orden que estas pueden realizar, se expusieron los pasos para el diseño empleando métodos simbólicos, así como las características y posibilidades que ofrecen algunas herramientas computacionales para el análisis y simulación de estos filtros ,donde presentan aplicaciones prácticas en el diseño y fabricación de circuitos integrados ,también que los análisis de sensibilidad y Monte Carlo son vitales a la hora de diseño o fabricación de estos circuitos, pues a través de ellos podemos tener en cuenta las afectaciones que pueden llegar a sufrir las respuestas debido a las variaciones de sus componentes, ya sea por su tolerancia o envejecimiento y deterioro.

CAPÍTULO 2. PROCEDIMIENTO PARA EL DISEÑO DE REDES BASADAS EN DOBLE T

Para el diseño de los filtros de redes doble T pueden encontrarse ecuaciones simplificadas en algunos manuales de diseño; no obstante, la utilización de herramientas computacionales como MATLAB permite utilizar métodos rigurosos de diseño a partir del planteamiento del sistema de ecuaciones del circuito. En este capítulo se desarrolla el procedimiento completo de diseño de filtros de redes doble T, se especifican las herramientas computacionales que se utilizan tanto para la realización como para la simulación de los filtros, así como el procedimiento que debe seguirse para la verificación del diseño y la evaluación de la sensibilidad de los filtros

2.1 Diseño de filtros

El procedimiento para el diseño de los filtros de redes doble T seguido en este trabajo se basa en la utilización de ecuaciones exactas a partir del sistema de las ecuaciones de nodo de cada circuito. Ello permite, utilizando las herramientas simbólicas de MATLAB, seguir un procedimiento sistemático para la realización a través de la obtención de la función de red, el cálculo de las expresiones de los parámetros de esta en función de los componentes del circuito, y por último la confección de las ecuaciones de realización. Una vez obtenidas las expresiones de realización de las componentes del circuito, se procede al diseño utilizando los parámetros de la respuesta de frecuencias deseada por medio de un proceso de aproximación. (Thomas, Haddad et al. 1998)

2.1.1 Aproximaciones

Se utilizaron tres configuraciones de circuito *Twin-T*, una *TAF-HAF*¹; otra *TAN*² y por último una *TAF-LPF*³.

A la hora de diseñar los diferentes filtros basados en doble T utilizamos el *toolbox* simbólico de *Matlab*. Basándonos para el diseño en el orden (que siempre será 2 ya que órdenes superiores serían estos mismos filtros en cascada), la frecuencia de corte y otros. Para diseñar el filtro pasoalto (*TAF-HAF*) se utilizó aproximadamente de Chebychev I con frecuencia esquina de la banda de paso en 1 kHz y error en esa banda de 1 dB. Para ello se utilizó *Filter Design Toolbox* de MATLAB a través de la línea de comando siguiente:

```
>> [b, a] = cheby1(2, 2, 2*pi*1000, 'high', 's')
b =
    7.943282347242815e-01         0         0
a =
    1.000000000000000e+00    6.136277993429839e+03    4.796539394318957e+07
```

Con estos coeficientes se construye la función de red

$$H_{HP_{Cheby1}}(s) = \frac{0.7943282347242815}{s^2 + 6.136277993429839 \times 10^3 s + 4.796539394318957 \times 10^7}$$

A partir de esta expresión se obtienen los valores de sus parámetros (ver anexo 1), siendo:

$$w_p = 6.925705302941324e+03, \text{ y}$$

$$Q = 1.128649208910798,$$

Siendo estos los que se utilizarán en el tercer capítulo para realizar el filtro.

Para trazar la respuesta de magnitud frecuencias de la función de aproximación calculada se utilizan las líneas de comando siguientes:

```
>> f = logspace(0, 5, 1e5);
>> [H, w] = freqs(b, a, 2*pi*f);
>> semilogx(w/(2*pi), 20*log10(abs(H))+2); grid
```

¹ TAF-HPF: *Two-amplifiers filter – High-pass filter*.

² TAN: *Two-amplifiers notch*.

³ TAF-LPN: *Two-amplifiers filter – Low-pass filter*.

```
>> axis([10 1e4 -60 10])
>> xlabel('Frequency (Hz)'); ylabel('Magnitude')
>> title('Twin-T TAF HPF Cheby1')
```

La respuesta se traza en la figura 2.1.

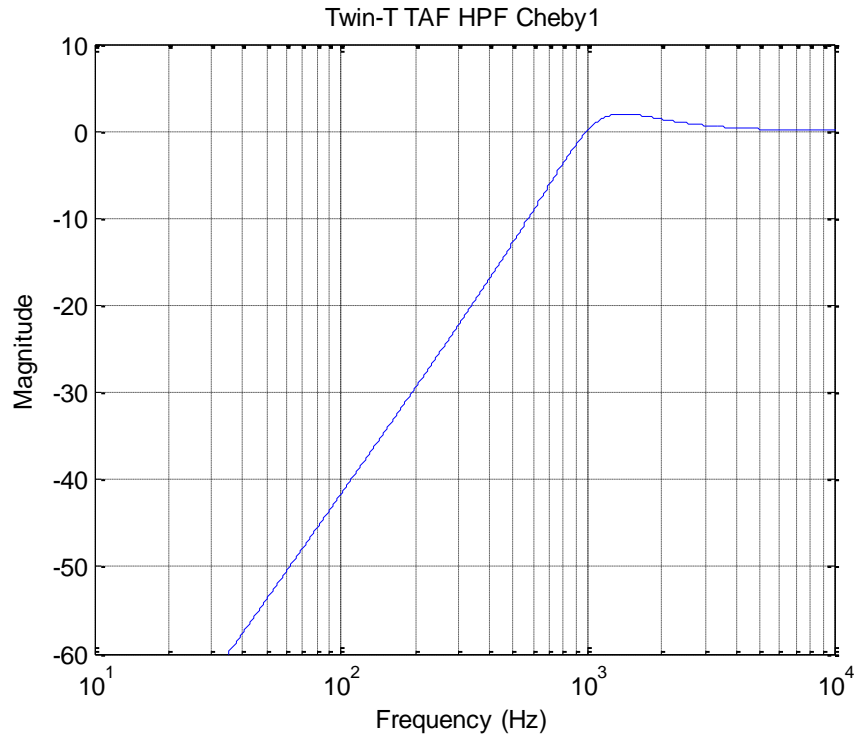


Figura 2.1. Respuesta de magnitud-frecuencias de la función de red pasoalto de Chebyshev I con segundo orden, frecuencia esquina de la banda de paso en 1 kHz y rizado en la banda de paso igual a 1 dB.

El diseño *notch* (NF) se obtiene utilizando también aproximante de Chebyshev I para una frecuencia de la hendidura en 4 kHz, mediante las líneas de comando siguiente

```
>> wp1 = 3200; wp2 = 4000^2/wp1;
>> [b, a] = cheby1(1, 1, 2*pi*[wp1 wp2], 'stop', 's');
```

En este caso se obtienen $w_p = 251333$ y $Q = 4.3672$.

Finalmente, para el filtro pasobajo (*TAF-LPF*) se utilizó también la aproximación de Chebyshev I, indicando frecuencia esquina de la banda de paso en 100 Hz y rizado en esa banda de 1 dB, para obtener $w_p = 659.74$ y $Q = 0.95652$. (Thomas, Haddad et al. 1998)

2.1.2 Escoger una configuración circuital

A la hora de elegir el circuito utilizamos las configuraciones: TAF-HPF: *Two-amplifiers filter – High-pass filter*, TAN: *Two-amplifiers notch*, TAF-LPN: *Two-amplifiers filter – Low-pass notch* a las cuales le hayamos su respuesta de frecuencia que depende ahora de sus componentes, los que se obtuvieron siguiendo las mismas especificaciones que para hallar las respuestas de frecuencia de las aproximaciones, se llegó a las ecuaciones necesarias de los componentes de los circuitos a partir de las ecuaciones de diseño de las configuraciones y el MATLAB (ver anexo 1). Entonces, cómo saber si los circuitos diseñados tienen las mismas características que las respuestas de frecuencia a partir de las que fueron realizados? Para esto se usa el software TINA TI o Multisim, con los cuales se probó el circuito real con los valores nominales de los componentes pasivos (exactos) y utilizando un modelo del OPA363 en TINA TI como el que se muestra en la figura 2.2 y el OPA353 en Multisim. Estos software de simulación trazan la gráfica de respuesta de magnitud-frecuencia, por lo que un análisis de la misma permite verificar que se satisfacen las especificaciones de la respuesta deseada. Su uso es sencillo, luego de tener todos los valores de los componentes del circuito, solamente lo dibujamos en su plataforma de edición esquemática:

$$R_1 = 14.44e3 \, \Omega$$

$$R_2 = 14.44e3 \, \Omega$$

$$R_3 = 7.22e3 \, \Omega$$

$$C_1 = 10e-9 \, F$$

$$C_2 = 10e-9 \, F$$

$$C_3 = 20e-9 \, F$$

$$R_4 = 1e3 \, \Omega$$

$$R_0 = 3.51e3 \, \Omega$$

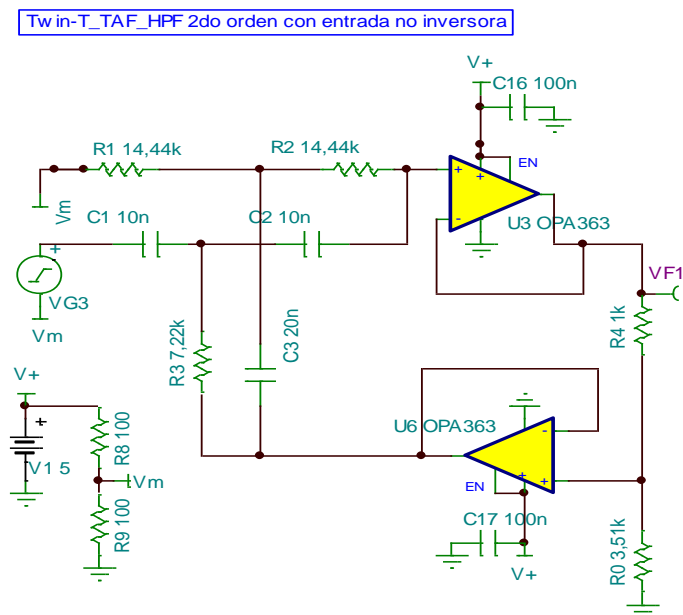


Figura 2.2 Filtro *Twin-T TAF HPF* simulado en TINA TI

Para verificar la respuesta de magnitud-frecuencias se le realiza un análisis de corriente alterna, donde se especifican características como la frecuencia de inicio y final del barrido, de tipo logarítmico y magnitud en dB. Este análisis arroja una gráfica que nos permite comprobar cuan correcto está nuestro diseño, análisis que se realizará en el tercer capítulo.(Fitzpatrick 2011)

2.2 Diseño del experimento

Con este trabajo se pretende llevar a cabo un análisis detallado sobre el nivel de variación de la respuesta de magnitud-frecuencia de un circuito ante las variaciones de sus diversos componentes y por tanto de la sensibilidad del mismo, para filtros de segundo orden basados en doble T, que son circuitos que utilizan dos amplificadores operacionales. Utilizando como red nominal la que emplea valores exactos de componentes obtenidos a partir de los métodos de aproximación, se determinará la sensibilidad de estas redes fijando una tolerancia para los componentes del 10%. Como se explicó en el primer capítulo, el objetivo es facilitar que se adquieran los componentes del circuito que menos distorsionen las respuestas de magnitud-frecuencia de los filtros con una mayor tolerancia lo que equivale a un menor precio, lo que es el objetivo principal de todo ingeniero, realizar más con menos. Para llevar a cabo este análisis vamos a comparar los resultados de aplicar el método de Monte Carlo con el de peor caso (*Worst case*) como indicadores de la sensibilidad de los parámetros de la función de red del circuito (ω_p , Q , ω_z y G) ante las variaciones de sus componente.

Como primer paso se utiliza el *Symbolic Math Toolbox* de MATLAB para determinar la sensibilidad teórica de un circuito a una frecuencia determinada con respecto a un componente, dándole una tolerancia común para ese componente, ya sea resistor o capacitor, esto se hace partiendo de la propiedad $s_X^{cY} = s_X^Y$ y que la constante $c = 1/2\pi$, entonces:

$$S_X^Y = \frac{\partial Y/Y}{\partial X/X} = \frac{\partial(\ln Y)}{\partial(\ln X)}, \quad (2.1)$$

siendo X el componente en función e Y el parámetro a analizar, que como se explicó anteriormente puede ser ω_p , Q , ω_z y G . Este procedimiento se realizó a través de MATLAB, lo que resulta más práctico, ya que se toma ventaja de la función $Y=\text{diff}(X)$ que calcula la diferencia entre los elementos adyacentes de X (MathWorks 2002)

El resultado teórico de la sensibilidad se contrasta con el obtenido experimentalmente por medio del análisis de peor caso. Para ello se calcula gráficamente la variación que sufre el parámetro que se analiza, Y , con respecto al componente que varía, X , utilizando igualmente la expresión 2.1. Dado que, $S_{X_i}^Y$ expresa el % de cambio que sufre el parámetro Y de la función de red ante un % de cambio del componente X_i del circuito, si por ejemplo, en una configuración de circuito para un filtro activo de 2do orden la sensibilidad de su ω_p con respecto al resistor R_2 es -0.5 , lo cual se denota mediante $S_{R_2}^{\omega_p} = -0.5$, ello significa que si R_2 se incrementa o disminuye en un 1%, entonces debido al signo negativo el valor de ω_p disminuye o incrementa un 0.5%, o lo que es lo mismo, en un factor -0.005 . (Mandache, IORDACHE et al. 2010)

El análisis del peor caso (*Worts Case*) permite determinar los peores efectos posibles que tiene sobre un parámetro la variación de un componente, lo cual es consecuencia de la sensibilidad del parámetro para cada circuito. De las herramientas computacionales disponibles, la que ofrece esta posibilidad es Multisim. El procedimiento que sigue este software para determinar la respuesta de peor caso es el siguiente:

1. Realiza un análisis de sensibilidad (*AC sensitivity*) para determinar la sensibilidad del voltaje de salida al componente.
2. De acuerdo a los resultados de la sensibilidad se determina el valor mínimo o máximo del componente seleccionado.
3. Se realiza el *AC analysis* utilizando el valor calculado para el componente.

Utilizando tolerancia de 10 % se obtendrán las respuestas de frecuencia extremas alrededor de la nominal, lo cual permitirá realizar un análisis de redes doble T, además de verificar experimentalmente el procedimiento teórico utilizado para determinar la sensibilidad por medio de métodos simbólicos. El análisis del peor caso puede realizarse con cada uno de los componentes que conforman al circuito.

Resulta oportuno recalcar que el análisis de la sensibilidad se realiza basado en el análisis de la tolerancia de los componentes del circuito. Dado que todos los componentes pueden variar alrededor de su valor nominal y dentro de su margen de tolerancia, el análisis de la tolerancia que puede admitirse para cada componente es la base para el análisis de la sensibilidad con el propósito de encontrar el rango de variación de los parámetros de interés de las respuestas del filtro. Esto es,

el problema del análisis de la tolerancia durante el diseño de un filtro consiste en encontrar el posible margen de variación de cada componente que garantice que las variaciones de los parámetros de las respuestas del circuito (ω_p , Q , ω_z y G) tengan una distribución aceptable para que no comprometan su desempeño. (Kolev 2002)

Para determinar la mencionada distribución de la variación de cada uno de los parámetros de la función de red del filtro se han desarrollado varias técnicas, pero las que se reconocen como de mayor potencialidad y eficacia se han basado hasta ahora en el análisis de Monte Carlo. Ello permite, ante todo, determinar el tipo de distribución estadística que tienen las variaciones del parámetro. (Choi, Agarwal et al. 2010)

El análisis de Monte Carlo será realizado mediante el software Multisim el que posee un ambiente de trabajo parecido al Tina, por lo que las simulaciones se harán sobre circuitos que utilizan para los componentes modelos que simulan la realidad. Este análisis permite determinar las variaciones que sufre la respuesta de frecuencia nominal analizada anteriormente con Tina TI o Multisim ya que arroja una serie de curvas de respuesta de magnitud-frecuencia, pasobajo, pasoalto, pasabanda o *notch* que donde se analiza no solo el peor y mejor caso para una tolerancia determinada, sino que se toman un número determinado de casos, pero todos aleatorios, por lo que puede o no estar recogido el peor y mejor caso. Las herramientas utilizadas para la ayuda en el análisis permiten especificar la tolerancia que tendrá el componente a realizarle el análisis, también la frecuencia de inicio y final de la corrida, así como el número de corridas, la variable de salida, el tipo de análisis, el tipo de barrido y la cantidad de puntos de análisis como se muestra en la figura 2.3 donde se realizan quince corridas entre 10 Hz y 10 kHz.

Para la realización del análisis de Monte Carlo se utilizarán variaciones de cada componente de forma aleatoria con media igual a su valor nominal y distribución de densidad probabilística uniforme dentro del intervalo del T % (tolerancia) de su valor nominal

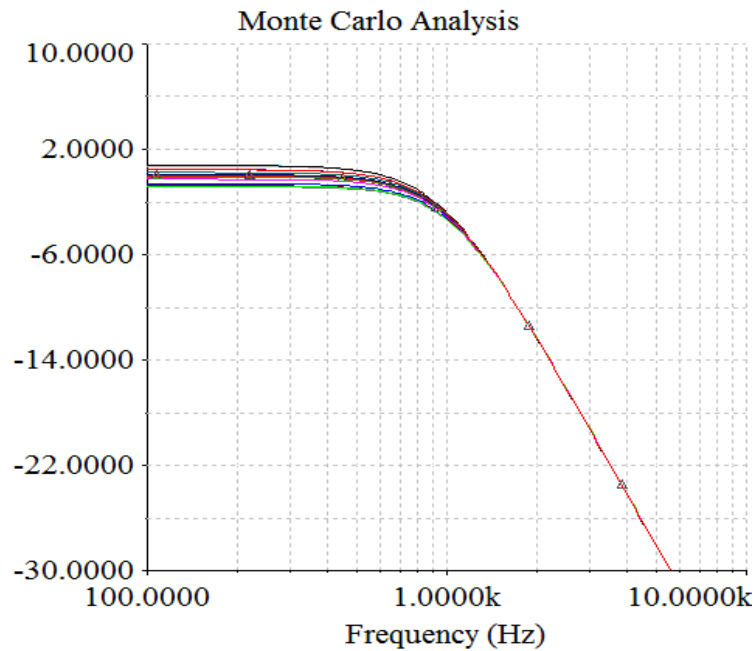


Figura 2.3. Análisis de Monte Carlo con 15 corridas en Multisim

2.3 Especificaciones

2.3.1 MATLAB

MATLAB un lenguaje de computación técnica de alto nivel, un entorno interactivo para el desarrollo de algoritmos y una moderna herramienta de análisis de datos, permite un orden para reducir el tiempo de solución para las tareas típicas y simplifica en gran medida el desarrollo de nuevos algoritmos, es la principal herramienta para resolver una amplia gama de problemas científicos y aplicados.

La manera más fácil de visualizar a MATLAB es pensar en él como en una calculadora totalmente equipada.(Toolbox 1999)

Propiedades de MATLAB

Año: 2010

Versión: 7.10.0.499 (R2010a x64)

Desarrollador: *The MathWorks, Inc.*

Plataforma: *Windows*

Compatibilidad con Vista: Completa

Requisitos del sistema:

OS:

Windows XP Service Pack 3

Windows Server 2003 R2 con el Service Pack 2

Windows Vista Service Pack 1 o 2

Windows Server 2008 Service Pack 2 o R2

Windows 7

Procesadores:

Cualquier procesador Intel o AMD x86 con soporte conjunto de instrucciones SSE2

Espacio:

1 GB para MATLAB solamente,

4.3 GB para una instalación típica

RAM:

1024 MB

(Por lo menos 2048 MB recomendado)

Idioma: Inglés solamente.

2.3.2 Tina

Cada año, los circuitos electrónicos se vuelven más rápidos y más complejos, y por lo tanto requieren más poder computacional para analizar su funcionamiento. TINA 7 es un potente y asequible paquete de software para el análisis y el diseño en tiempo real de pruebas analógicas, digitales,

Tina TI es un *software* de simulación de circuitos eléctricos fácil de usar, pero potente, basado en un motor de *SPICE*. Tina TI se limita a los circuitos con dos circuitos integrados y hasta 20 nodos adicionales.

La mayor funcionalidad que tiene Tina TI es que permite de manera gráfica poder representar circuitos eléctricos de distintos grados de complejidad. Esto se debe al subprograma *Schematics* Editor el cual permite representar visualmente los circuitos eléctricos (lo que es un gran avance ya que antes del *Schematics*, los programas de simulación de circuitos eléctricos se manejaban mediante códigos y algoritmos.

Propiedades de Tina TI

Intel Pentium o procesador equivalente

256 MB de RAM

200 MB de espacio disponible en disco duro

Microsoft Windows 9x / ME / NT / 2000/ XP / Vista / Windows 7

TINA-TI es una versión totalmente funcional de Tina TI, cargada de una biblioteca de macro modelos TI, incluyendo modelos pasivos y activos. Tina TI no tiene límites en el tamaño de circuito, el número de nodos, o el número de circuitos integrados. Se puede utilizar para el diseño de filtros de 1 a 10 polos.

2.3.3 Multisim

Multisim provee herramientas para analizar el comportamiento de los circuitos. Es una plataforma de *software* intuitiva y fácil de usar que combina los esquemas y simulaciones. Multisim abstrae las complejidades y dificultades de la simulación tradicional basada en sintaxis, se puede usar para identificar errores, validar el comportamiento de diseño y generar prototipos más rápido por lo que los esquemas pueden ser transferidos al diseño *NI Ultiboard*

Se utilizó la versión 10.1 de Multisim

Windows 2000 Service Pack 3 or later, Windows XP, Vista, or 64-bit Vista.

- Pentium 4 class microprocessor or equivalent (Pentium III class minimum).
- 512 MB of memory (256 MB minimum).
- 1.5 GB of free hard disk space (1 GB minimum).
- Open GL® capable 3D graphics card recommended (SVGA resolution video adapter with 800 × 600 video resolution minimum, 1024 × 768 or higher preferred).
- To develop custom LabVIEW based instruments for use in Multisim, LabVIEW 8.2.x or 8.5.x is required.

2.3.4 Tipo de PC

Trabajamos en una maquina con *Microsoft* Windows XP Profesional

Vesion 2002

Service Pack 3

Equipo Intel (R)

Pentium (R) D CPU 3.000GHz

504MB de RAM

Conclusiones del capítulo

Como conclusión de este capítulo se destaca que para el diseño de filtros activos basados en doble T que es necesario hacer énfasis en la importancia de los métodos simbólicos de MATLAB, ya que facilitan la solución de sistemas de ecuaciones para determinar la función de red del circuito, obtener las expresiones de diseño, así como las de la sensibilidad sin necesidad de evaluar los componentes del circuito. También podemos concluir que las herramientas computacionales son vitales para hallar y analizar las respuestas de magnitud-frecuencia de las configuraciones de los circuitos, ya que permiten comprobar los valores de sensibilidad de los circuitos a través de los análisis del peor caso y Montecarlo, lo cual es posible a través de Multisim.

CAPÍTULO 3. EVALUACIÓN MEDIANTE LOS MÉTODOS DE DISEÑO DE HERRAMIENTAS CAD

Se realiza una simulación funcional del diseño realizado mediante las herramientas de simulación de MATLAB, Tina-TI y Multisim, evaluándose los resultados obtenidos. Para un entendimiento más completo del análisis llevado a cabo con el propósito de conocer cuales componentes de un filtro activo afectan mayormente la respuesta de magnitud-frecuencia del mismo y a la sensibilidad del circuito en sí, se evaluaron los diseños realizados a partir de las aproximaciones de Chebychev I. Presentamos tres de los diseños de circuitos realizados, a los cuales se les realiza análisis de Monte Carlo y del peor caso para comprobar las variaciones que sufren las respuestas de magnitud-frecuencia respecto a la frecuencia central, la ganancia y el factor de calidad. En cada circuito hallamos los valores de sensibilidad respectivos a los componentes que más influyen en la variación de los parámetros de la respuesta del filtro y además se muestra una forma experimental de comprobación de la sensibilidad a partir de las curvas de peor caso trazadas mediante Multisim.

3.1 Primera configuración de circuito

La primera configuración de circuito para filtro que se analiza es la *Twin-T* con dos amplificadores operacionales pasoalto (*TAF-HPF*) que se muestra en la figura 3.1, la cual ha sido implementada con función aproximante de Chebyshev I de segundo orden para una frecuencia esquina de la banda de paso de 1 kHz y rizado en la banda de paso de 1 dB. Con estas especificaciones se realizó el circuito que se muestra en la figura 3.1 utilizando el procedimiento de realización simbólica expuesto también el capítulo 2 y teniendo en cuenta que $C_1 = C_2 = C$, $C_3 = 2C$, $R_1 = R_2 = R$ y $R_3 = R/2$, el; cual arroja que la función de red es

$$H(s) = \frac{(C^2 R^2 s^2)}{(C^2 R^2 s^2 + (4 C R - 4 C K R) s + 1)}$$

Además, mediante el procedimiento simbólico se obtuvo que

$$\omega_p = \frac{1}{RC},$$

$$Q = \frac{1}{4(1-K)} \quad \text{y}$$

$$K = \frac{R_0}{R_0 + R_4}.$$

De estas últimas expresiones, y con los valores de ω_p y Q obtenidos en el capítulo 2 para esta configuración, sumiendo que $C_1 = C_2 = 10 \text{ nF}$, $C_3 = 2C_1$ y $R_4 = 1\text{k}\Omega$, se obtiene siguiendo el procedimiento simbólico que $R_1 = R_2 = 14.4 \text{ k}\Omega$, $R_3 = R_1/2$ y $R_0 = 3.51 \text{ k}\Omega$.

Mediante la simulación del filtro a través de Multisim con valores nominales (exactos) para sus componentes puede obtenerse la respuesta de magnitud-frecuencias con el propósito de compararla con la ideal obtenida del proceso de aproximación realizado mediante MATLAB.

La simulación realizada de esta red mediante Multisim arroja la respuesta de magnitud-frecuencias que muestra la figura 3.2 (parte derecha), la cual puede compararse con la respuesta teórica que brinda Matlab para la función de aproximación (parte izquierda). Se puede observar que tanto la respuesta ideal como la simulada en Multisim a partir del diseño poseen gran coincidencia en sus principales parámetros, lo que demuestra la efectividad de los métodos de diseño simbólicos realizados y lo acertado del diseño. Debe tenerse presente que aunque se utilizan los valores de los componentes pasivos nominales, los AOp utilizan modelos que simulan dispositivos reales.

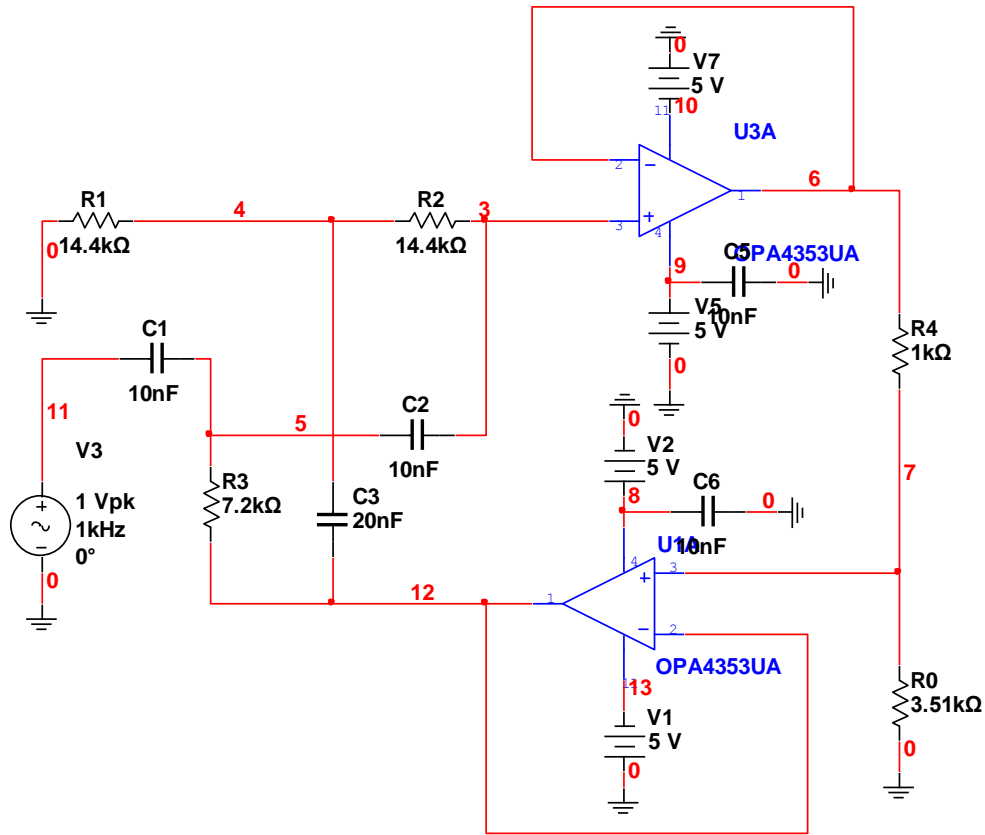


Figura 3.1. Configuración *Twin-T* de segundo orden con dos AOp pasoalto (TAF-HPF), función aproximante de Chebyshev I para frecuencia esquina de la banda de paso 1 kHz y rizado de 1 dB.

A partir de este diseño, se hallarán las sensibilidades teóricas respectivas de los parámetros de la respuesta de frecuencias (f_p , y Q) a sus diversos componentes. En este caso, dado que $\omega_p = 2\pi f_p$, y 2π es una constante, la $S_X^{\omega_p} = S_X^{f_p}$. Para ello se procede mediante MATLAB simbólico de la forma siguiente:

```
>> syms K fp Q R0 R1 R4 C1
>> fp = (1/(R1*C1))/(2*pi);
>> Q = 1/(4*(1-(R0/(R4+R0))));
>> dfpdR1 = diff(fp, R1); disp 'dfp/dR1 = '; pretty(dfpdR1)
dfp/dR1 =
  1
  -----
      2
  2 C1 pi R1

>> dfpdC1 = diff(fp, C1);
>> disp 'dfp/dC1 = '; pretty(dfpdC1)
```

$$\frac{df_p}{dC_1} = \frac{1}{2 C_1 \pi R_1}$$

```
>> dQdR0 = diff(Q, R0);
>> disp 'dQ/dR0 = '; pretty(dQdR0)
```

$$\frac{dQ}{dR_0} = \frac{4 R_0^4}{(R_0^2 + R_4^2) \sqrt{R_0^2 + R_4^2} - 4 R_0^2}$$

```
>> dQdR4 = diff(Q, R4);
>> disp 'dQ/dR4 = '; pretty(dQdR4)
```

$$\frac{dQ}{dR_4} = \frac{4 R_0^4}{(R_0^2 + R_4^2) \sqrt{R_0^2 + R_4^2} - 4 R_0^2}$$

De las expresiones de anteriores se observa que f_p no depende de R_0 ni R_4 , mientras que Q no depende de R_1 ni C_1 . Es por ello que se describirá seguidamente el procedimiento para determinar $S_{R_0}^Q$, por lo que si se desea determinar otra sensibilidad puede procederse de forma semejante.

Para determinar $S_{R_0}^Q$ se necesitan los valores numéricos de los componentes del circuito, por lo que utilizando MATLAB, a partir de la expresión obtenida se procede de la forma siguiente

```
>> R1 = 14.4e3;
>> R2 = 14.4e3;
>> R3 = 7.2e3;
>> R4 = 1e3;
>> R0 = 3.51e3;
>> C1 = 10e-9;
>> C2 = 10e-9;
```

```
>> C3 = 20e-9;
>> SQdR0 = vpa(subs(dQdR0)*(R0/subs(Q)), 4)
SQdR0 =
0.7783
```

Este resultado equivale a que un aumento de R_0 del 10 % produce un aumento de Q del 7.783 %.

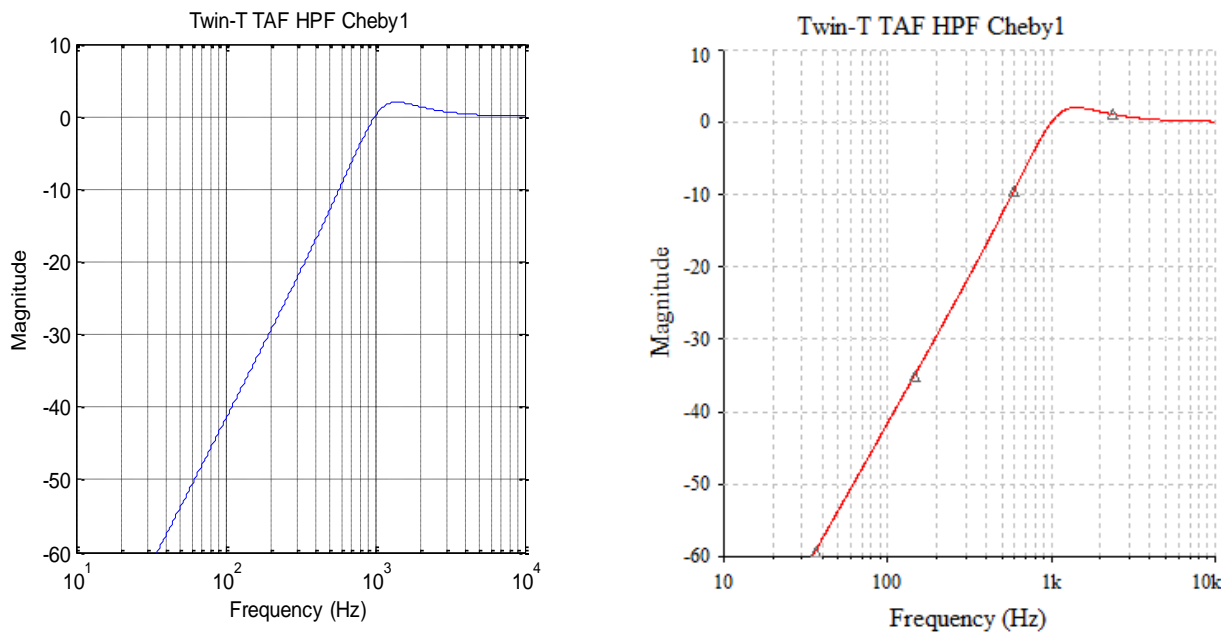


Fig. 3.2. Respuestas de magnitud-frecuencias de la red *Twin-T* TAF HPF de segundo orden realizada mediante aproximante de Chebyshev I mostrada en la figura 3.1, la cual se implementa con valores de $f_p = 925.37\text{Hz}$, $Q = 0.956$ y $G_{lp} = 0.891$. En la parte izquierda la respuesta de magnitud teórica obtenida mediante MATLAB y a su derecha el resultado de la simulación mediante Multisim.

A partir de análisis anterior se procederá a verificar experimentalmente la sensibilidad de Q con respecto al resistor R_0 . Para ello se realiza en Miltisim el análisis de peor caso, con el cual se obtienen las gráficas que muestra la figura 3.3, en las cuales se muestra solo el segmento de frecuencias desde 1 kHz hasta 10 kHz y eje de magnitud con escala lineal entre 1 y 1.35. En esta figura el trazo inferior corresponde a la respuesta de magnitud-frecuencias nominal, mientras que el superior al peor caso variando R_0 hasta 3.861 k Ω que es el 10 % de su valor nominal (3.51 k Ω).

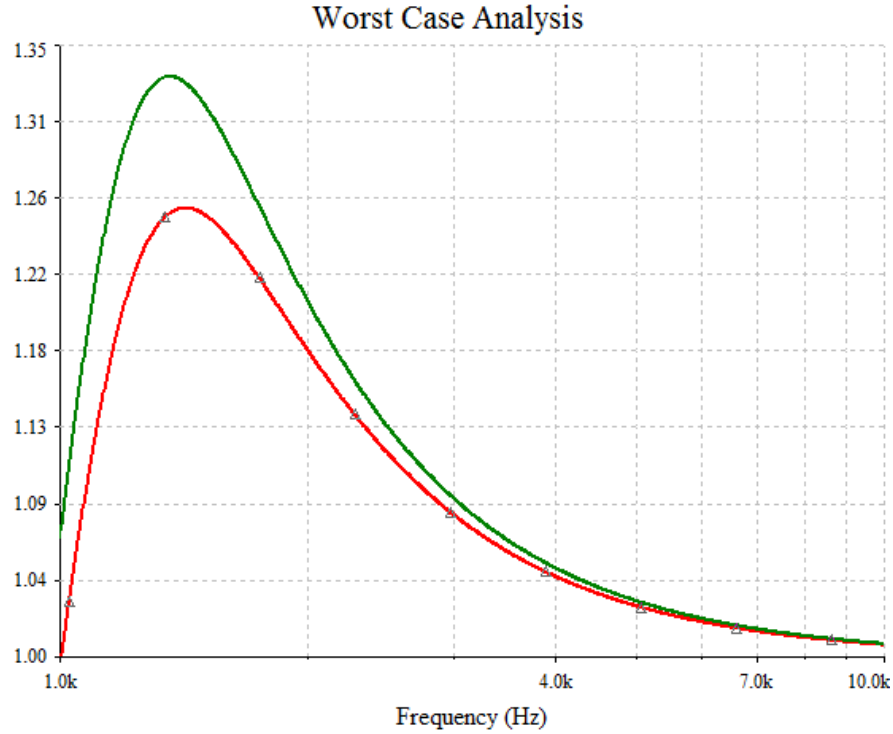


Figura 3.3. Análisis de peor caso realizado mediante Multisim para el filtro *Twin-T TAF-HPF* variando R_0 utilizando un eje de ordenadas lineal.

Las variaciones que presenta el factor de calidad en la figura 3.3 pueden determinarse a partir de las relaciones que existen entre el valor del máximo y su frecuencia en la respuesta de magnitud, las cuales fueron expuestas en el epígrafe 1.3. A partir de estas relaciones se confeccionó el programa 6 de anexo 1 que, mediante métodos simbólicos, se determina las variaciones de Q . Como datos para este programa se indican el coeficiente b_2 del numerador de la $H(s)$ obtenida en el proceso de aproximación (epígrafe 2.1.1), que es 0.7943282347242815, el máximo de la respuesta nominal que ocurre en 1416.4 Hz con magnitud 1.2565, así como el máximo de la respuesta de peor caso que ocurre en 1355.5 con magnitud 1.3325. Con estos datos utilizando el citado programa, se obtiene que la $S_{R_0}^Q = 0.6883$. Como puede apreciarse, este resultado es ligeramente inferior al determinado teóricamente, lo cual se debe a que Multisim realiza varias corridas asignando valores aleatorios a R_0 para quedarse con el peor de los casos.

En el análisis de Monte Carlo (figura 3.4) nos deja más claro las variaciones que sufre la respuesta de magnitud en el entorno de f_p a partir de la variación de un solo componente, R_0 . En este caso la respuesta para R_0 nominal es la que se desplaza al centro del grupo de trazados, siendo las que

están por encima y por debajo de esta las correspondientes a valores de R_0 por encima y por debajo de su valor nominal respectivamente ya que $S_{R_0}^Q$ tiene signo positivo.

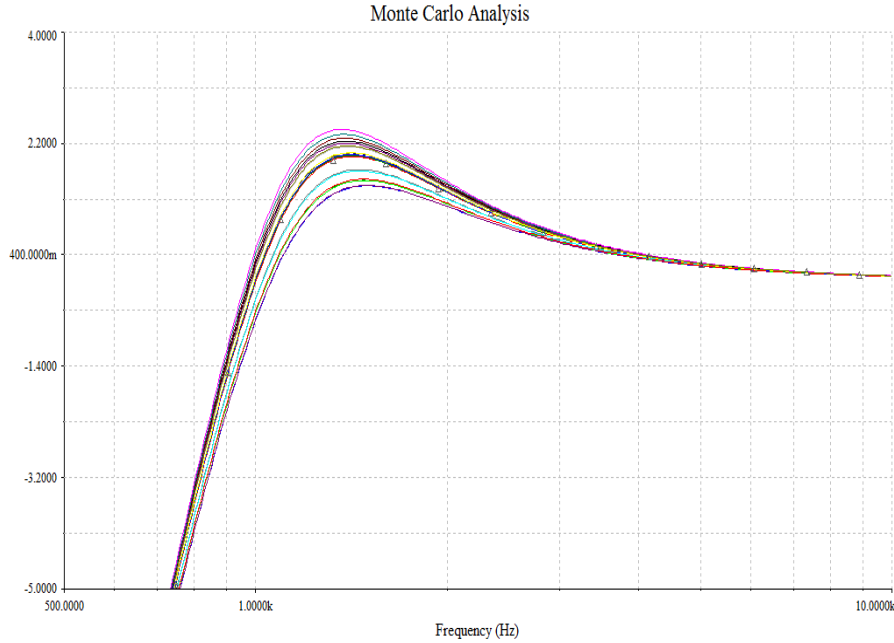


Figura 3.4. Análisis de Monte Carlo para las variaciones de la respuesta de magnitud del filtro *Twin-T TAF-HPF* debidas a las variaciones de R_0 dentro del 10 % de su valor nominal

3.2 Segunda configuración de circuito

La segunda configuración de circuito para filtro que se analiza es la red (*Twin-T*) de hendidura (*notch*) con dos amplificadores operacionales (TAN) que se muestra en la figura 3.5 la cual ha sido implementada con función aproximante de Chebyshev I de segundo orden para frecuencias esquina de la banda de paso en 3.2 y 5.0 kHz, con lo cual la frecuencia central de la hendidura se localice en 4.0 kHz. Para determinar los parámetros de la respuesta de frecuencias de la función de red se utiliza la función de MATLAB *cheby1*, para la cual, además de las dos frecuencias esquina de la banda suprimida antes mencionadas se especifica un rizado en la banda de paso de 1 dB. Dado que se desea que la función de red de hendidura sea de segundo orden debe especificarse en la función *cheby1* que el orden sea uno, por lo cual no será apreciado el error en la banda de paso. Con las especificaciones anteriores se obtiene que la función de red debe tener $\omega_p = 25133$ y $Q = 4.3672$.

Utilizando la función nombrada “Red doble T (*Twin-T*) Notch (TAN) General” del anexo 1 se lleva a cabo el proceso de realización que conduce a la red de la figura 3.5.

La simulación del filtro de hendidura se realiza a través de Multisim con valores nominales (exactos) para sus componentes, obteniéndose la respuesta de magnitud-frecuencias con el propósito de compararla con la ideal obtenida del proceso de aproximación realizado mediante MATLAB.

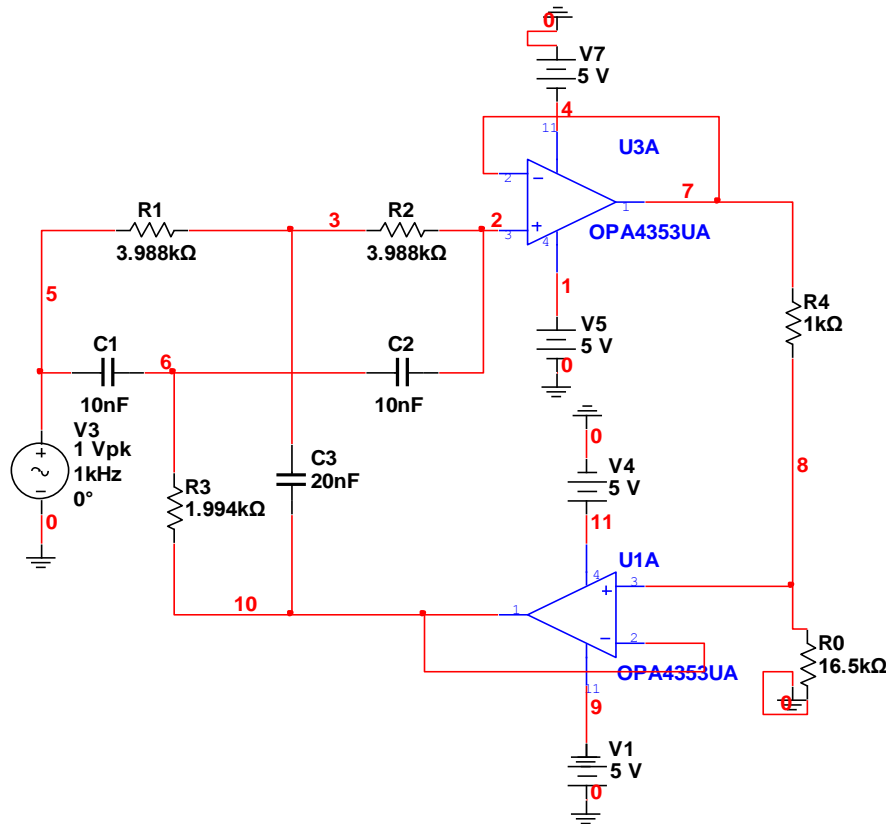


Figura 3.5. Configuración *Twin-T* de segundo orden con dos AOp *Notch* (TAN), función aproximante de Chebyshev I para frecuencia central de la banda de paso en 4 kHz y factor de calidad 4.3672.

La simulación realizada de esta red mediante Multisim arroja la respuesta de magnitud-frecuencias que muestra la figura 3.6 (parte derecha), la cual puede compararse con la respuesta teórica que brinda MATLAB para la función de aproximación (parte izquierda). Se puede observar que tanto la respuesta ideal como la simulada en Multisim a partir del diseño poseen gran coincidencia en sus principales parámetros, lo que demuestra la efectividad de los métodos de diseño simbólicos realizados y lo acertado del diseño.

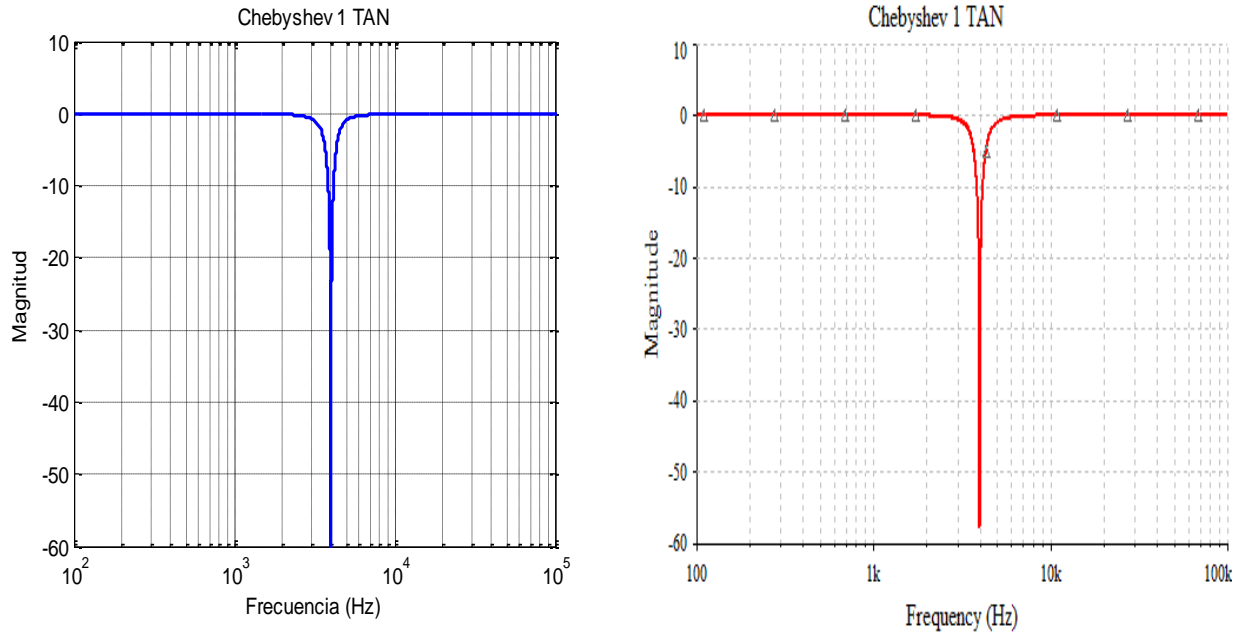


Fig. 3.6. Respuestas de magnitud-frecuencias de la red *Twin-T Notch* (TAN), de segundo orden realizada mediante aproximante de Chebyshev I mostrada en la figura 3.5, la cual se implementa con valores de $f_p = 4.0$ kHz y $Q = 4.3672$. En la parte izquierda la respuesta teórica obtenida mediante MATLAB y a su derecha el resultado de la simulación del circuito mediante Multisim.

A partir de este diseño, se realizó un estudio de su sensibilidad de los parámetros de su respuesta de frecuencias a los diversos componentes, del cual se presenta a continuación el análisis para el caso de la sensibilidad del factor de calidad con respecto al resistor R_0 , esto es $S_{R_0}^Q$.

La expresión del factor de calidad en función de los componentes de este circuito es la misma que la de la red *Twin-T* TAF-HPF utilizada en el epígrafe anterior, por lo que

$$dQ/dR_0 = \frac{\frac{4 R_0}{(R_0 + R_4)^2} - \frac{4}{R_0 + R_4}}{\frac{\sqrt{4 R_0}}{\sqrt{R_0 + R_4}} - \frac{\sqrt{2}}{4}}$$

No obstante, dado que los valores de R_0 y R_4 de la red de hendidura son diferentes a los del filtro pasoalto anterior, el nuevo valor teórico de $S_{R_0}^Q = 0.943$.

Haciendo análisis de peor caso en Multisim se verifica experimentalmente la sensibilidad calculada anteriormente de forma teórica. La figura 3.7 muestra en su parte superior el análisis de peor caso de la respuesta de magnitud-frecuencias variando el resistor R_0 dentro del 10 % de su valor nominal confirmando que la frecuencia de la hendidura no depende de R_0 , la cual se mantiene en 4.0 kHz, mientras que en su parte inferior la figura muestra un detalle de esta respuesta para frecuencias desde 3.0 hasta 5.0 kHz de donde puede obtenerse que el ancho de banda (-3 dB) de la respuesta nominal es de 962 Hz (trazado exterior), mientras que el de peor caso es de 811.4 Hz (trazado interior). Con estos valores se determina que los factores de calidad son $Q_n = 4.385$ para R_0 nominal y $Q_{wc} = 4.8199$ para el peor caso. De esta forma, el valor experimental obtenido para $S_{R_0}^Q = 1.026$.

Se puede observar que los valores de sensibilidad hallada a partir de los valores de las curvas en las simulaciones realizadas mediante Multisim, son muy similares a los calculados a partir de las expresiones teóricas de los parámetros por medio de MATLAB, tanto en magnitud como en signo. Debe tenerse en cuenta que el resultado experimental depende de determinar un punto exacto sobre una curva, lo cual, además de ser un procedimiento generalmente inexacto, está sujeto a la subjetividad de la persona que lo determina, por lo cual debe esperarse una ligera diferencia entre ambos métodos.

Como en el epígrafe anterior, la verificación final de la sensibilidad del factor de calidad con respecto a las variaciones de R_0 culmina con el análisis de Monte Carlo con el cual puede conocerse la forma en que se distribuyen las diferentes respuestas de magnitud-frecuencia ante las variaciones aleatorias de R_0 dentro de un $\pm 10\%$ de su valor nominal. Este análisis se muestra en la figura 3.8 con 20 corridas, dejándonos más claro las variaciones que sufre el ancho de banda alrededor de la frecuencia de la hendidura. Como se ha explicado anteriormente, por un problema de estadística se necesita un alto número de corridas para llegar a resultados más precisos. En este caso la respuesta para R_0 nominal es la que se desplaza al centro del grupo de trazados, siendo las que están por encima y por debajo de esta las correspondientes a valores de R_0 por encima y por debajo de su valor nominal respectivamente ya que $S_{R_0}^Q$ tiene signo positivo.

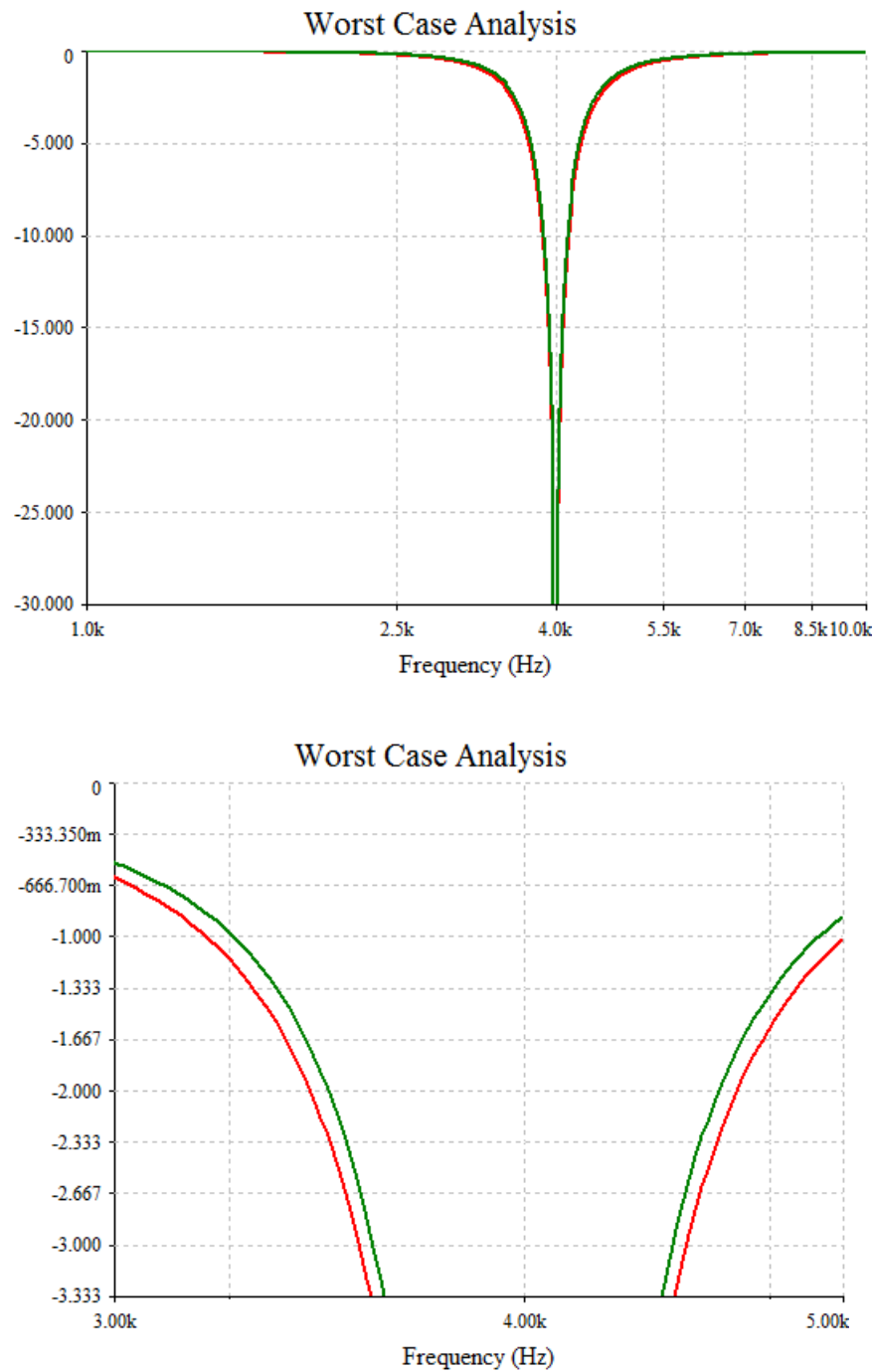


Figura 3.7. Análisis de peor caso realizado mediante Multisim para el filtro *Twin-T TAN* variando R_0 un 10 % por encima de su valor nominal.

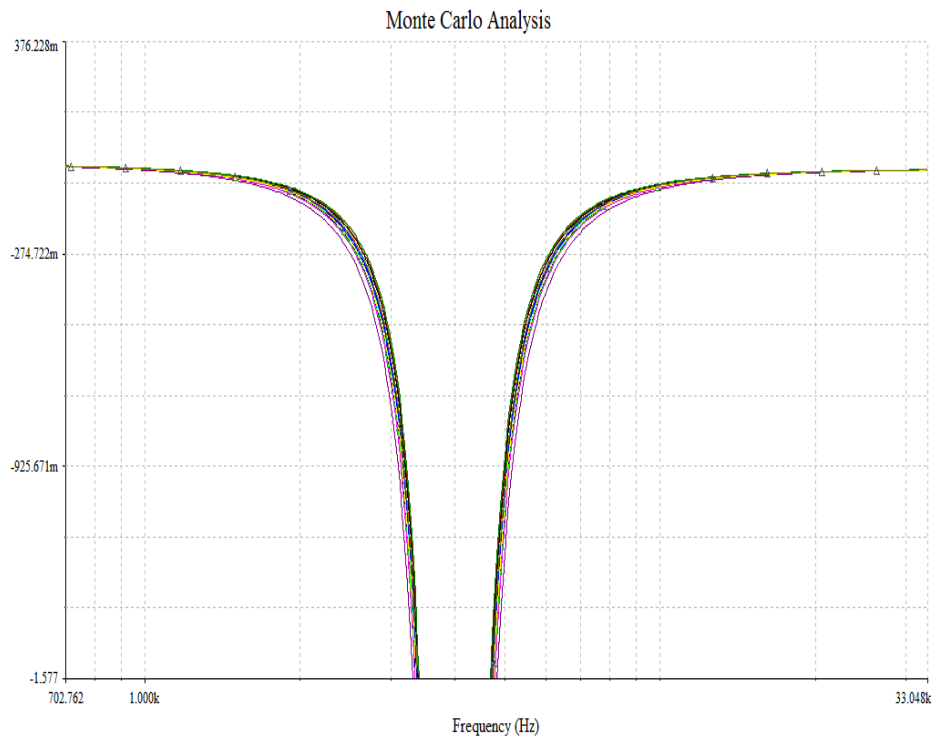


Figura 3.8. Análisis de Monte Carlo para las variaciones de la respuesta de magnitud del filtro *Twin-TAN* debidas a las variaciones de R_0 dentro del 10 % de su valor nominal.

3.3 Tercera configuración de circuito

La tercera configuración de circuito para filtro que se analiza es la *Twin-T* con dos amplificadores operacionales pasabajo (*TAF-LPF*) que se muestra en la figura 3.9, la cual se ha implementado con función aproximante de Chebyshev I de segundo orden para una frecuencia esquina de la banda de paso de 100 Hz y rizado en esa banda de 1 dB. Con estas especificaciones se realizó el circuito que se muestra en la figura 3.9

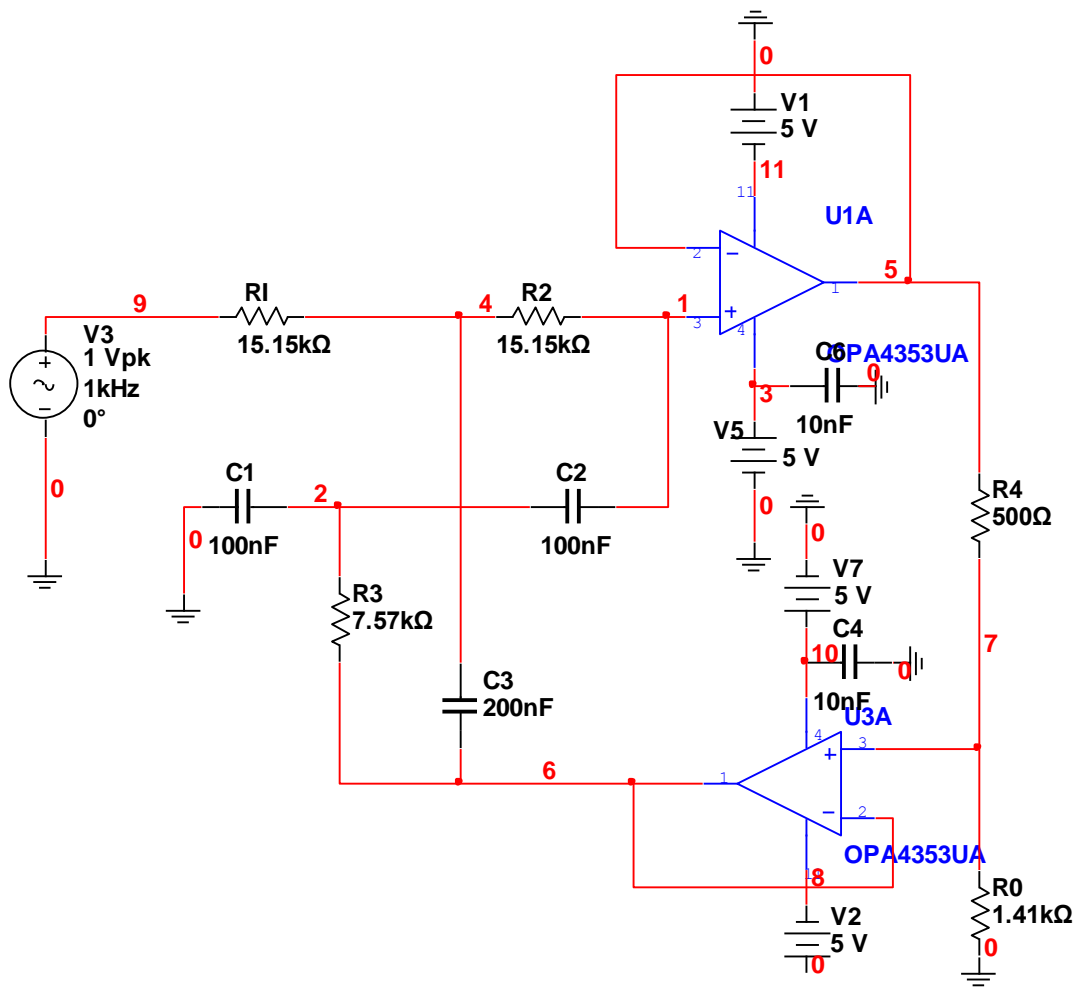


Figura 3.9. Configuración *Twin-T* de segundo orden con dos AOp pasobajo (TAF-LPF), función aproximante de Chebyshev I para frecuencia esquina de la banda de paso 100 Hz y rizado en esa banda de 1 dB.

Con simulación realizada de esta red mediante Multisim se arroja la respuesta de magnitud-frecuencias que muestra la figura 3.10 (parte derecha), la cual puede compararse con la respuesta teórica que brinda MATLAB para la función de aproximación (parte izquierda). Se puede observar que tanto la respuesta ideal como la simulada en Multisim a partir del diseño poseen gran coincidencia en sus principales parámetros, lo que demuestra la efectividad de los métodos de diseño simbólicos realizados y lo acertado del diseño.

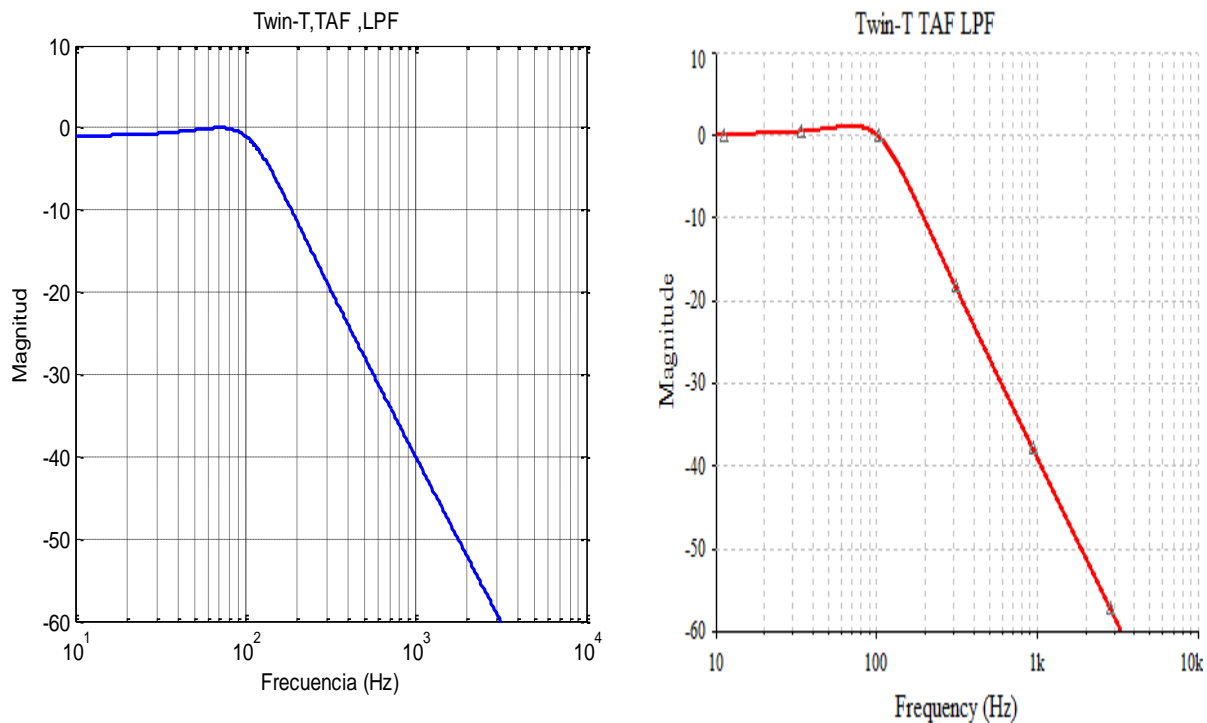


Fig. 3.10. Respuestas de magnitud-frecuencias de la red *Twin-T* TAF-LPF de segundo orden realizada mediante aproximante de Chebyshev I mostrada en la figura 3.9, la cual se implementa con valores exactos de sus componentes. En la parte izquierda la respuesta de magnitud teórica obtenida mediante MATLAB y a su derecha el resultado de la simulación mediante Multisim.

Se puede observar al igual que en la primera configuración de filtro, que en la figura 3.10 tanto la respuesta ideal como la simulada en Multisim a partir del diseño poseen gran coincidencia en sus principales parámetros, aunque como bien se dice en la configuración anterior siempre se debe tener presente que aunque se utilizan los valores de los componentes pasivos nominales, los AOp utilizan modelos que simulan dispositivos reales.

La tabla 3.1 resume los resultados de los valores de la sensibilidad del parámetro Q con respecto al componente R_0 , tanto determinada teóricamente por métodos de computación simbólica como a través del método directo de las gráficas de peor caso obtenidas mediante Multisim. Puede apreciarse la coincidencia en estos resultados tanto en magnitud como en signo.

Tabla 3.1. Valor de la sensibilidad de Q al componente R_0 de la red de (TAN).

	$S_{R_0}^Q$	
MATLAB	0.3333	Teórico
Multisim	0.2369	Práctico

Las respuestas de peor caso ante las variaciones del capacitor R_0 se muestran en la figura 3.11.

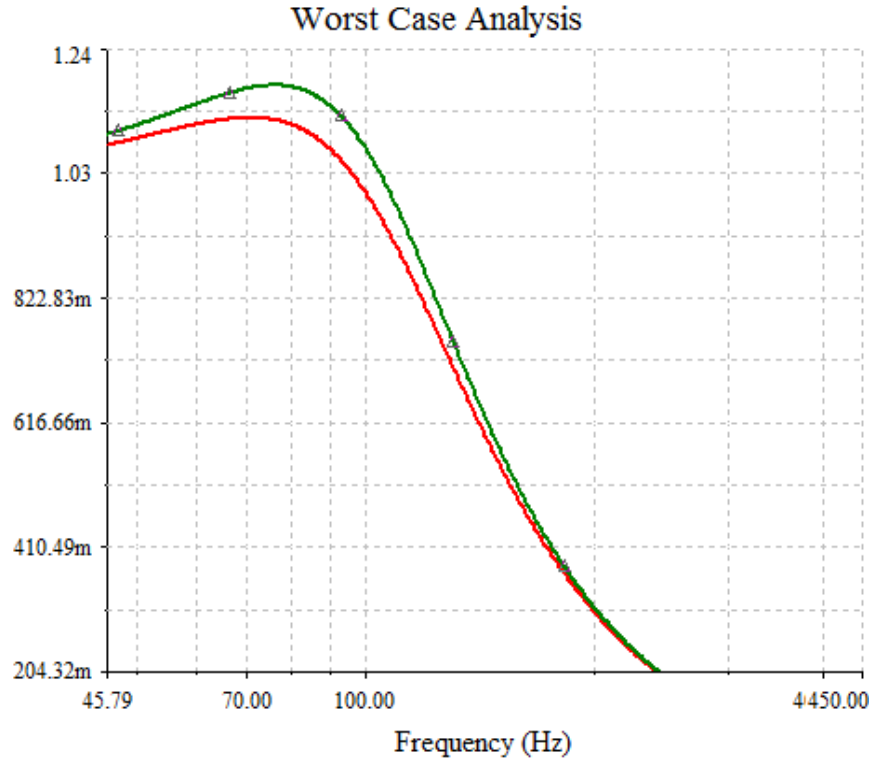


Figura 3.11. Análisis de peor caso realizado mediante Multisim para el filtro *Twin-T TAF-LPF* variando R_0 utilizando un eje de ordenadas lineal

La respuesta de magnitud-frecuencias correspondiente al valor nominal de R_0 , que es de 1.41 k Ω en la parte inferior (curva roja), se muestra el valor de f_0 en 70.2368 Hz, mientras que a su derecha, la obtenida para un incremento de R_0 del 10 %, que es 1.551 k Ω , aumentando el valor de f_0 hasta 75.4589 Hz.

Luego realizamos las curvas del análisis de Montecarlo para ver las posibles variaciones de la respuesta de magnitud-frecuencia que puede sufrir ante variaciones de un $\pm 10\%$ de R_0 , ya que en el circuito real el componente puede tener un valor cualquiera dentro de su rango de tolerancia, como se muestra en la figura 3.12.

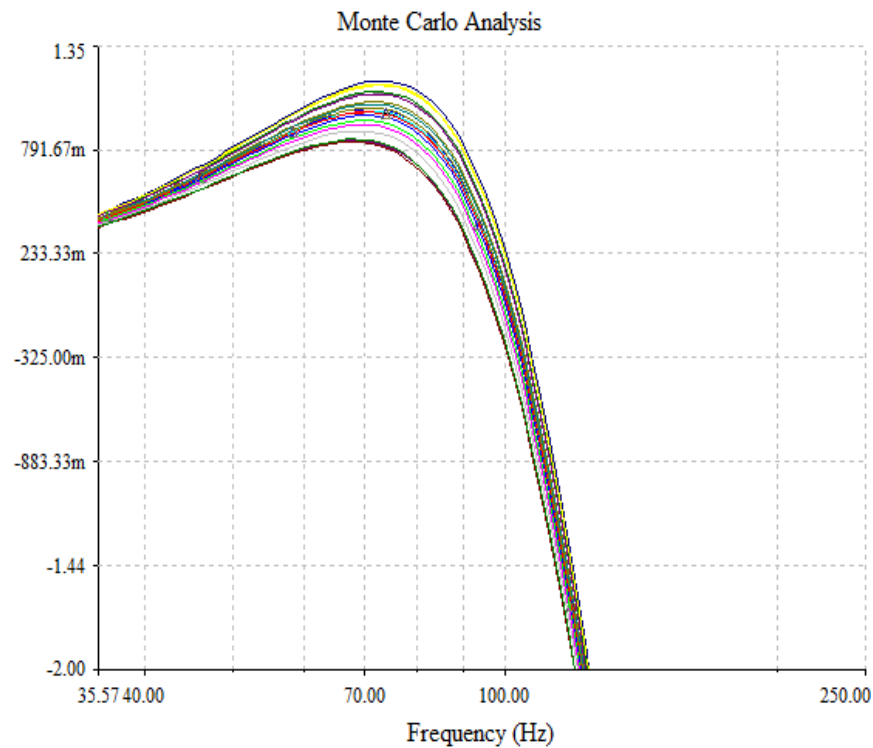


Figura 3.12. Análisis de Monte Carlo para las variaciones de la respuesta de magnitud del filtro *Twin-T TAF-LPF* debidas a las variaciones de R_0 dentro del 10 % de su valor nominal

Observandose que para 20 corridas de la respuesta pasabajo del filtro LPF se obtienen una serie de curvas que tienen valores aleatorios, variando dentro de un 10% de tolerancia de R_0 , por lo que por un problema de estadística se necesitan un alto número de corridas para llegar a resultados más precisos.

Conclusiones parciales del capítulo

En este capítulo se llegó a la conclusión de que existe coincidencia entre las respuestas de magnitud-frecuencias nominales obtenidas a partir de la función de aproximación teórica realizada en MATLAB y la simulada en Multisim con los componentes calculados a través de los procedimientos desarrollados y utilizando amplificadores operacionales reales. Se pudo comprobar que el análisis de la sensibilidad basada en el método de Monte Carlo muestra cómo varía una respuesta del filtro ante las modificaciones de los componentes del circuito, pero requiere de gran número de corridas por lo que es costoso computacionalmente; no obstante, dado que es la más aceptada y difundida desde el punto de vista estadístico, sus resultados nos mostraron su eficacia

como método de análisis más general que el de peor caso. Esta conclusión es fruto de que el método de peor caso conduce a resultados extremos de las variaciones de las respuestas del filtro ante las modificaciones de un componente pero no permite conocer la distribución de los resultados intermedios, siendo por esta razón de vital importancia en este trabajo la utilización de ambos análisis para una comprensión mayor del impacto de la sensibilidad en un circuito.

CONCLUSIONES Y RECOMENDACIONES

Conclusiones

Como resultado de la investigación realizada sobre los filtros activos basados en redes doble T se llegaron a las siguientes conclusiones:

- Los referentes teóricos establecidos acerca del diseño de filtros basados en doble T son el resultado de recopilar información dispersa en diversas fuentes y permiten sistematizar los procedimientos de diseño y análisis de estas redes, con particular aplicación al caso de la sensibilidad.
- De las diversas estructuras de redes doble T que pueden encontrarse en la literatura, las configuraciones de circuito conocidas como redes de *Two-amplifiers filter*, *High-pass filter*, *Two-amplifiers* y *notch*. *Two-amplifiers filter* – *Low-pass filter* se reportan como las que tienen mayor aplicación en la vida práctica, sobre todo en la fabricación de circuitos integrados.
- Los procedimientos desarrollados en este trabajo para evaluar la sensibilidad de parámetros de la respuesta de frecuencias de redes doble T utilizando las herramientas de computación simbólica que brinda MATLAB resultan de gran utilidad debido a que permiten arribar a expresiones teóricas en función de los componentes de la red.
- Las herramientas CAD como Multisim y Tina permitieron verificar de forma eficiente los métodos de diseño a través de la comparación entre las respuestas de frecuencias teórica y la simulada, así como comprobar que los valores de sensibilidad obtenidos mediante las expresiones teóricas determinadas coinciden con los valores que arrojan las simulaciones.

Recomendaciones

A pesar del extenso trabajo realizado la riqueza del mismo no permite abarcar todos los posibles análisis para todas las configuraciones de circuitos, por lo que para futuros proyectos es recomendable abordar los aspectos siguientes:

- Desarrollar versiones de los métodos de realización y análisis simbólicos realizados en MATLAB que utilicen software libre.
- Ampliar el análisis de la sensibilidad a otras configuraciones de circuitos a variables de estado.
- Utilizar los procedimientos desarrollados en este trabajo para el diseño y análisis de filtros en la docencia.

REFERENCIAS BIBLIOGRÁFICAS

- Báez-López, D., et al. (2012). "Advanced Circuit Simulation Using Multisim Workbench." Synthesis Lectures on Digital Circuits and Systems **7**(1): 1-144.
- Bai, Y.-W., et al. (2003). "Adjustable 60Hz noise reduction and ECG signal amplification of a remote electrocardiogram system." system **18**: 141.
- Besterfield, D. H. (2009). Control de calidad, Pearson Educación.
- Carlson, A. B., et al. (1986). "Communication systems: an introduction to signals and noise in electrical communication." Guía Académica: 129.
- Carter, B. (2000). "A single-supply op-amp circuit collection." Texas Instruments Literature, Number SLOA058.
- Choi, S., et al. (2010). Symmetric current balancing circuit for multiple DC loads. Applied Power Electronics Conference and Exposition (APEC), 2010 Twenty-Fifth Annual IEEE, IEEE.
- Erdei, A. Z. and M. Horgoş (2007). "THE COMPARISON BETWEEN ORCAD AND MATLAB ON TOLERANCE ANALYSIS FOR ANALOG CIRCUIT." Scientific Bulletin, serie C, fascicle Mechanics, Tribology, Machine Manufacturing Technology **21**: 229-236.
- Fitzpatrick, D. (2011). Analog design and simulation using OrCAD Capture and PSpice, Elsevier.
- Gielen, G. G., et al. (1989). "ISAAC: A symbolic simulator for analog integrated circuits." Solid-State Circuits, IEEE Journal of **24**(6): 1587-1597.
- Hamann, J. C., et al. (1999). "Using Monte Carlo simulations to introduce tolerance design to undergraduates." Education, IEEE Transactions on **42**(1): 1-7.
- Hayes, M., et al. (2007) "Evaluation of the Worst-Case Performance of Active Filters, using Robust Control Ideas." The 2007 IEEE International Symposium on, IEEE.
- Hongliang, W. S. W. Z. G. (2003) "The research on tolerance design and second-design for optimizing parameter in electronic circuit."
- Huang, W., et al. (2007). "System accuracy analysis of the multiphase voltage regulator module." Power Electronics, IEEE Transactions on **22**(3): 1019-1026.

- Juriaić, D., et al. (2001). Low-sensitivity SAB band-pass active-RC filter using impedance tapering. Circuits and Systems, 2001. ISCAS 2001. The 2001 IEEE International Symposium on, IEEE.
- Juriscic, D., et al. (2002). Low-sensitivity active-RC high-and band-pass second-order sallen & key allpole filters. Circuits and Systems, 2002. ISCAS 2002. IEEE International Symposium on, IEEE.
- Jurišić, D. and N. Mijat (2005). Low-Sensitivity, Low-Noise, Band-Rejection and All-Pass Active-RC Filters. The IEEE International Symposium on Industrial Electronics 2005.
- Jurišić, D. (2011). "Low-Noise, Low-Sensitivity Active-RC Allpole Filters Using MATLAB Optimization." The IEEE International Symposium on Industrial Electronics 2011 51(4), 361-373
- Kolev, L. (2002). "Worst-case tolerance analysis of linear DC and AC electric circuits." Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on **49**(12): 1693-1701.
- Luther, E. and J. Rodriguez (2006). Introduction to Multisim Schematic Capture and SPICE Simulation, Connexions, Rice University.
- Mandache, L., et al. (2010). "Sensitivity and Tolerance Analysis in Analog Circuits Using Symbolic Methods." Development and Application Systems: 66.
- MathWorks (2002). Symbolic Math Toolbox, User's Guide.
- Moschytz, G. S. (1999). "Low-sensitivity, low-power active-RC allpole filters using impedance tapering." Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on **46**(8): 1009-1026.
- Moschytz, G. (2002). Trade-Offs in Sensitivity, Component Spread and Component Tolerance in Active Filter Design. Trade-Offs in Analog Circuit Design, Springer: 315-339.
- Palma-Rodriguez, A. A., et al. (2012). DDD-based symbolic sensitivity analysis of active filters. Devices, Circuits and Systems (ICDCS), 2012 8th International Caribbean Conference on, IEEE.
- Porter, L., et al. (2010). "Fault Detection for Smart Grid Reliability Enhancement." Journal of Management & Engineering Integration 3(2): 106.
- Prajna, S., et al. (2002). Introducing SOSTOOLS: A general purpose sum of squares programming solver. Decision and Control, 2002, Proceedings of the 41st IEEE Conference on, IEEE.
- Proakis, J. G. (2001). Intersymbol Interference in Digital Communication Systems, Wiley Online Library.
- Quick, M., et al. (1995). "UBC Watershed Model Manual-Version 4.0." Department of Civil Engineering, University of British Columbia, Vancouver, BC, Canada.
- Raut, R. and M. N. Swamy (2010). Modern Analog Filter Analysis And Design: A Practical Approach, John Wiley & Sons.
- Thomas, T., et al. (1998). "Design and performance of active power filters." Industry Applications Magazine, IEEE **4**(5): 38-46.

Toolbox, S. M. (1999). "For use with MATLAB, The Math Works." Inc., Natick, MA.

Yang, H. (2007). Symbolic sensitivity analysis techniques and applications in analog circuit synthesis, University of Cincinnati.

ANEXOS

Anexo I Programas realizados en MATLAB

Programa # 1 Realización de Filtros Activos Red Twin-T - TAF - HPF

```
.
%% Realización de Filtros Activos.
% Red Twin-T - TAF - HPF.
%

clc
clear all

% Sistema de Ecuaciones.
syms R R1 R0 C K;
syms s V1 V2 Vo Vi;
SistEcuac = solve(...
    ' V1*(2/R + s*2*C) - Vo*(1/R + s*2*K*C) = 0' ,...
    ' V2*(2/R + s*2*C) - Vo*(2*K/R + s*C) = Vi*s*C' ,...
    '-V1/R - V2*s*C + Vo*(1/R + s*C) = 0' ,...
    'V1', 'V2', 'Vo');

Hs = SistEcuac.Vo / Vi;
HsCol = collect(Hs, s);
disp 'H(s) = '; pretty(HsCol); disp ' '

[num, den] = numden(HsCol);
numCoefic = coeffs(num, s)
denCoefic = coeffs(den, s)

%% Parámetros de la función de red.
syms b2 a1 a0 wp Qp
b2 = simplify(numCoefic / denCoefic(3));
disp ' '; disp 'b2 = '; pretty(b2)
a1 = simplify(denCoefic(2) / denCoefic(3));
disp ' '; disp 'a1 = '; pretty(a1)
a0 = simplify(denCoefic(1) / denCoefic(3));
disp ' '; disp 'a0 = ' ; pretty(a0)
%%
wp = simplify(sqrt(a0))
% disp ' '; disp 'wp = '; pretty(wp)
Q = simplify(wp / a1)
% disp ' '; disp 'Q = '; pretty(Q)

%% Sistemas de ecuaciones (2)
syms Kt wpt Qt R0t R1t Ct Rt;
```

```

Realiza = solve(...
    'wpt = 1/(Rt*Ct)' ,...
    'Qt = 1/(4*(1 - (R0t/(R0t + R1t))))' ,...
    'Rt', 'R0t');

R = Realiza.Rt;
disp ' '; disp 'R = '; pretty(R)
R0 = simplify(Realiza.R0t);
disp ' '; disp 'R0 = '; pretty(R0)

%% Método de aproximación
format long e
[b, a] = cheby1(2, 2, 2*pi*1000, 'high', 's')
f = logspace(0, 5, 1e5);
[H, w] = freqs(b, a, 2*pi*f);
semilogx(w/(2*pi), 20*log10(abs(H)))
grid
axis([10 1e5 -60 0])
xlabel('f (Hz)')
ylabel('|H(e^{jw})|_{dB}')
title('Chebyshev 1 TAF , HPF ,')
Cheb1LPN = tf(b, a)

%% Sustitución de las ecuaciones (2) para obtener valores numéricos
wpt = sqrt(a(3))
Qt = wpt / a(2)
Ct = 10e-9
R1t = 1e3
R = vpa(subs(R), 6)
R0 = vpa(subs(R0), 6)

```

Programa # 2 Realización de Filtros Activos. Red (Twin-T) Notch (TAN)

```

%% Realización de Filtros Activos.
% Red doble T (Twin-T) Notch (TAN) General.
%
% Ref. Kugelstadt (2008) "Active Filters Design Techniques",
% Texas Instruments, pp. 38.

clc
clear all

% Sistema de Ecuaciones.
syms R C K s V2 V3 Vi Vo;

SistEcuac = solve(...
    '-V2*s*C - V3/R + Vo*(1/R + s*C) = 0',...
    ' V2*(s*2*C + 2/R) - Vo*(s*C + 2*K/R) = Vi*s*C',...
    ' V3*(2/R + s*2*C) - Vo*(s*2*K*C + 1/R) = Vi/R',...
    ' V2', 'V3', 'Vo');

Hs = SistEcuac.Vo / Vi;
HsCol = collect(Hs, s);
pretty(HsCol); disp ' '

```

```

%% Simplificación.
[num, den] = numden(HsCol);
numCoefic = coeffs(num, s);
denCoefic = coeffs(den, s);

%% Parámetros de la función de red.
syms b0 a2 a1 a0 wp Qp
b0 = simple(numCoefic(1) / denCoefic(3));
disp ' '; disp 'b0 = '; pretty(b0)
b2 = simplify(numCoefic(2) / denCoefic(3));
disp ' '; disp 'b2 = '; pretty(b2)
a1 = simple(denCoefic(2) / denCoefic(3));
disp ' '; disp 'a1 = '; pretty(a1)
a0 = denCoefic(1) / denCoefic(3);
disp ' '; disp 'a0 = '; pretty(a0)

wz = simple(sqrt(b0));
disp ' '; disp 'wz = '; pretty(wz)
wp = simple(sqrt(a0));
disp ' '; disp 'wp = '; pretty(wp)
Q = simple(wp / a1);
disp ' '; disp 'Q = '; pretty(Q)

%% Sistemas de ecuaciones(2)
syms Kt wpt Qt wzt R0t R1t C1t Rt ;
Realiza = solve(...
    'wzt = 1/(C1t*Rt)',...
    'Qt = 1/(4*(1 - Kt))',...
    'Kt = R0t/(R1t+R0t)',...
    'Rt', 'R0t', 'Kt');

R = Realiza.Rt;
disp ' '; disp 'R = '; pretty(R)
K = simplify(Realiza.Kt);
disp ' '; disp 'K = '; pretty(K)
R0 = simplify(Realiza.R0t);
disp ' '; disp 'R0 = '; pretty(R0)

%% TAN
w3 = logspace(2, 6, 1e5);

%% Chebyshev 2
format short e
wp1 = 3200; wp2 = 4000^2/wp1;
[b, a] = cheby1(1, 1, 2*pi*[wp1 wp2], 'stop', 's');
b = round(b*1e8)*1e-8;
[Hch, w] = freqs(b, a, w3);
figure(2); semilogx(w/(2*pi), 20*log10(abs(Hch)),...
    'LineWidth', 2); grid
axis([1e2 1e5 -60 0.2])
xlabel('f (Hz)')
ylabel('|H(e^{jw})|_{dB}')
title('Chebyshev 1 TAN')
Cheb2LPN = tf(b, a)

```

```
G = b(1)
wzt = sqrt(b(3) / b(1))
wpt = sqrt(a(3))
Qt   = wpt / a(2)
```

```
%% Suponiendo.
R1t = 1e3
C1t = 10e-9;
R = vpa(subs(R), 3)
R0 = vpa(subs(R0), 3)
K = vpa(subs(K), 3)
```

Programa# 3 Realización de Filtros Activos Red Twin-T - TAF - LPF.

```
%% Realización de Filtros Activos.
% Red Twin-T - TAF - LPF.
%

clc
clear all

% Sistema de Ecuaciones (1).
syms R R1 R0 C K;
syms s V1 V2 Vo Vi;
SistEcuac = solve(...
    ' V1*(2/R + s*2*C) - Vo*(1/R + s*2*K*C) = Vi/R',...
    ' V2*(2/R + s*2*C) - Vo*(2*K/R + s*C) = 0',...
    '-V1/R - V2*s*C + Vo*(1/R + s*C) = 0',...
    'V1', 'V2', 'Vo');

Hs = SistEcuac.Vo / Vi;
HsCol = collect(Hs, s);
disp 'H(s) = '; pretty(HsCol); disp ' '

[num, den] = numden(HsCol);
numCoefic = coeffs(num, s)
denCoefic = coeffs(den, s)

%% Parámetros de la función de red.
syms b0 a1 a0 wp Qp
b0 = numCoefic(1) / denCoefic(3);
disp ' '; disp 'b0 = '; pretty(b0)
a1 = simplify(denCoefic(2) / denCoefic(3));
disp ' '; disp 'a1 = '; pretty(a1)
a0 = simplify(denCoefic(1) / denCoefic(3));
disp ' '; disp 'a0 = ' ; pretty(a0)
wp = simplify(sqrt(a0));
disp ' '; disp 'wp = ' ; pretty(wp)
Q = simplify(wp / a1);
disp ' '; disp 'Q = ' ; pretty(Q)

%% Sistema de ecuaciones (2)
syms Kt wpt Qt R0t R1t Ct Rt;
Realiza = solve(...
```

```

    'wpt = 1/(Rt*Ct) '
    'Qt = 1/(4*(1 - (R0t/(R0t + R1t)))) ' ,...
    'Rt', 'R0t');

R = Realiza.Rt;
disp ' '; disp 'R = '; pretty(R)
R0 = simplify(Realiza.R0t);
disp ' '; disp 'R0 = '; pretty(R0)

%% LPF
[b, a] = cheby1(2, 1, 2*pi*1e2, 's')
w3 = logspace(0, 6, 1e3);
[Hel, w] = freqs(b, a, w3);
figure(10); semilogx(w/(2*pi), 20*log10(abs(Hel)), ...
'LineWidth', 2); grid
axis([1e1 1e5 -60 0.2])
xlabel('f (Hz)')
ylabel('|H(e^{jw})|_{dB}')
title('TAF ,BPF, Q = 10')
BPF10 = tf(b, a)

%% Suponiendo
wpt = sqrt(a(3))
Qt = wpt/a(2)
Ct = 100e-9
R1t = 500
R = vpa(subs(R), 6)
R0 = vpa(subs(R0), 6)

```

Programa # 4 Método de Aproximación

```

%%Método de Aproximación
%
close all; clc; clear all
format long e

%% LPF
w1 = logspace(0, 6, 1e5);

%% Butterworth LPF
[b, a] = butter(2, 2*pi*1000, 's')
[Hbu, w] = freqs(b, a, w1);
figure(1); semilogx(w/(2*pi), 20*log10(abs(Hbu)), ...
'LineWidth', 2); grid
axis([1 1e5 -60 0.2])
xlabel('f (Hz)')
ylabel('|H(e^{jw})|_{dB}')
title('Butterworth LPF')
ButtLPF = tf(b, a)
ButtLPF1 = vpa(ButtLPF, 6)

%% Bessel LPF
[b, a] = besself(2, 2*pi*1273)
[Hbe, w] = freqs(b, a, w1);

```

```

figure(2); semilogx(w/(2*pi), 20*log10(abs(Hbe)),...
    'LineWidth', 2); grid
axis([1 1e5 -60 0.2])

%% Cheby LPF
close all; clc; clear all
format long e
w1 = logspace(0, 6, 1e5);
[b, a] = cheby1(2, 1, 2*pi*1000, 's')
[Hch, w] = freqs(b, a, w1);
figure(3); semilogx(w/(2*pi), 20*log10(abs(Hch)),...
    'LineWidth', 2); grid
axis([1 1e5 -60 0.2])
ChebyLPF = tf(b, a)
wp=sqrt(a(3))
fp=wp/(2*pi)
Qp=wp/a(2)
Glp=b(3)/a(3)

%% HPF
w2 = logspace(2, 7, 1e5);

%% Butterworth HPF
[b, a] = butter(2, 2*pi*1000, 'high', 's')
[Hbu, w] = freqs(b, a, w2);
figure(4); semilogx(w/(2*pi), 20*log10(abs(Hbu))); grid
axis([1e2 1e6 -40 0.2])

%% Bessel HPF
[b, a] = besself(2, 2*pi*785, 'high')
[Hbe, w] = freqs(b, a, w2);
figure(5); semilogx(w/(2*pi), 20*log10(abs(Hbe))); grid
axis([1e2 1e6 -40 0.2])

%% Cheby HPF
clear all
clc
w2 = logspace(2, 7, 1e5);
[b, a] = cheby1(2, 1, 2*pi*1000, 'high', 's')
[Hch, w] = freqs(b, a, w2);
figure(6); semilogx(w/(2*pi), 20*log10(abs(Hch)),...
    'LineWidth', 2); grid
axis([1e2 1e6 -40 0.2])
ChebyHPF = tf(b, a)

wp=sqrt(a(3))
fp=wp/(2*pi)
Qp=wp/a(2)
Ghp=b(1)

%% LPN
w3 = logspace(1, 6, 1e5);

%% Chebyshev 2
close all; clc; clear all

```

```

format long e
w3 = logspace(1, 6, 1e5);
[b, a] = cheby2(2, 40, 2*pi*4000, 's')
[Hch, w] = freqs(b, a, w3);
figure(7); semilogx(w/(2*pi), 20*log10(abs(Hch)), ...
'LineWidth', 2); grid
axis([1e1 1e5 -60 0.5])

ChebyLPNF = tf(b, a)
wz=sqrt(b(3)/b(1))
fz=wz/(2*pi)
wp=sqrt(a(3))
fp=wp/(2*pi)
Qp=wp/a(2)

%% Elliptic
[b, a] = ellip(2, 1, 30, 2*pi*1000, 's')
[Hel, w] = freqs(b, a, w3);
figure(8); semilogx(w/(2*pi), 20*log10(abs(Hel))); grid
axis([1e1 1e5 -60 0.2])

%% BP Q = 5
clear all; clc
w3 = logspace(1, 6, 1e5);
b = [0 2*pi*1000/5 0]
a = [1 2*pi*1000/5 (2*pi*1000)^2]
[Hel, w] = freqs(b, a, w3);
figure(9); semilogx(w/(2*pi), 20*log10(abs(Hel)), ...
'LineWidth', 2); grid
axis([1e1 1e5 -60 0.2])

BP = tf(b, a)
wp=sqrt(a(3))
fp=wp/(2*pi)
Qp=wp/a(2)
Gbp = b(2)/a(2)

%% BP Q = 10
b = [0 2*pi*100 0]
a = [1 2*pi*100 (2*pi*1000)^2]
[Hel, w] = freqs(b, a, w3);
figure(10); semilogx(w/(2*pi), 20*log10(abs(Hel))); grid
axis([1e1 1e5 -60 0.2])

```

Programa # 5 Cálculo de la sensibilidad teórica

```

% Cálculo de Sensibilidad Teóricas
% TAF-HPF
% Sistamas de ecuaciones
clear all ; clc
syms K fp Q R0 R1 R4 C1
fp = (1/(R1*C1))/(2*pi);
Q = 1/(4*(1-(R0/(R4+R0))));
disp 'fp = '; pretty(fp)

```

```

disp 'Q = '; pretty (Q)

% Derivada de los parámetros en función de los componentes
dfpdR1 = diff(fp, R1);
disp 'dfp/dR1 = '; pretty(dfpdR1)

dfpdC1 = diff(fp, C1);
disp 'dfp/dC1 = '; pretty(dfpdC1)

dQdR0 = diff(Q, R0);
disp 'dQ/dR0 = '; pretty(dQdR0)

dQdR4 = diff(Q, R4);
disp 'dQ/dR4 = '; pretty(dQdR4)

% Componentes reales
R1 = 14.4e3;
R2 = 14.4e3;
R3 = 7.2e3;
R4 = 1e3;
R0 = 3.51e3;
C1 = 10e-9;
C2 = 10e-9;
C3 = 20e-9;

% Fórmula de Sensibilidad
SfpdR1 = vpa(subs(dfpdR1)*(R1/subs(fp)), 4)

SfpdC1 = vpa(subs(dfpdC1)*(C1/subs(fp)), 4)

SQdR0 = vpa(subs(dQdR0)*(R0/subs(Q)), 4)

SQdR4 = vpa(subs(dQdR4)*(R4/subs(Q)), 4)

%TAN (idem al anterior)
clear all ; clc
syms K fp Q R0 R1 R4 C1
fp = (1/(R1*C1))/(2*pi);
Q = 1/(4*(1-(R0/(R4+R0))));
disp 'fp = '; pretty(fp)
disp 'Q = '; pretty (Q)

dQdR0 = diff(Q, R0);
disp 'dQ/dR0 = '; pretty(dQdR0)

R1 = 3.988e3;
R2 = 3.988e3;
R3 = 1.994e3;
R4 = 1e3;
R0 = 16.55e3;
C1 = 10e-9;
C2 = 10e-9;
C3 = 20e-9;
SQdR0 = vpa(subs(dQdR0)*(R0/subs(Q)), 4)

```

```
%TAF LPF (idem a anterior)
clear all ; clc
syms K fp Q R0 R1 R4 C1
fp = (1/(R1*C1))/(2*pi);
Q = 1/(4*(1-(R0/(R4+R0))));
disp 'fp = '; pretty(fp)
disp 'Q = '; pretty(Q)

dQdR0 = diff(Q, R0);
disp 'dQ/dR0 = '; pretty(dQdR0)

R1 = 15.15e3;
R2 = 15.15e3;
R3 = 7.57e3;
R4 = 1e3;
R0 = 500;
C1 = 100e-9;
SQdR0 = vpa(subs(dQdR0)*(R0/subs(Q)), 4)
```

Programa # 6 Cálculo de sensibilidad práctica con valores reales

```
%% Programa
% Twin-T TAF HPF No inversor.
% Determina la sensibilidad experimental a partir de la
% corrida en Multisim del peor caso para R0.
%
% Parámetros de entrada:
% b2: Coeficiente del numerador de H(s) que multiplica a s^2.
% mav: Vector 2x1 con valores de los máximos
%      [ma_peor caso ma_nomin ]. Ej. [1.3319 1.2566].
% wmv: Vector 2x1 con las frecuencias de los máximos
%      [vm_peor caso vm_nomin ]. Ej. [1355.5 1416.4].
% R0v: Vector 2x1 con los valores del resistor R0
%      [R0_peor caso R0_nomin ]. Ej. [3861 3510].
%
% Parámetros intermedios que se calculan :
% wp: Frecuencia del par de polos.
% Q: Factor de calida.
%
% Parámetros de salida:
% SQ_R0: Sensibilidad de Q con respecto a R0.
% Swp_R0: Sensibilidad de wp con respecto a R0.
%
clear all ; clc

%% Determina expresiones de Q y wp en función de b2, wm y ma.
syms b2 wp Q wm ma;
SistEcuac = solve(...
    'ma = b2*Q / sqrt(1 - 1/(4*Q^2))',...
    'wm = wp / sqrt(1 - 1/(2*Q^2))',...
    'Q', 'wp');
Q = SistEcuac.Q; Q = Q(3);
disp 'Q = ' ; pretty(Q)
```

```
wp = SistEcuac.wp; wp = wp(3);
disp ' ' ; disp 'wp = ' ; pretty(wp)

%% Datos.
b2 = 7.943282347242815e-01;
mav = [1.3325 1.2565];
wmv = [1355.5 1416.4];
R0v = [3861 3510];

%% Peor caso de Multisim.
wm = wmv(1); % Frecuencia del máximo.
ma = mav(1); % Valor del máximo.

Qm1 = vpa(subs(Q), 4)
wp1 = vpa(subs(wp), 4)

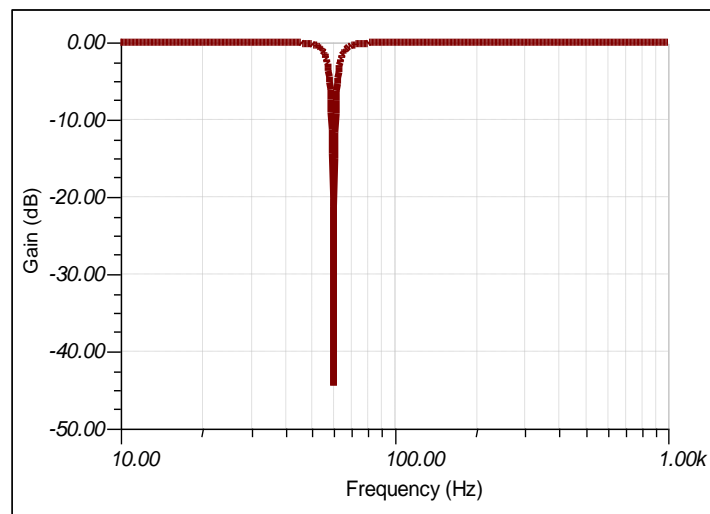
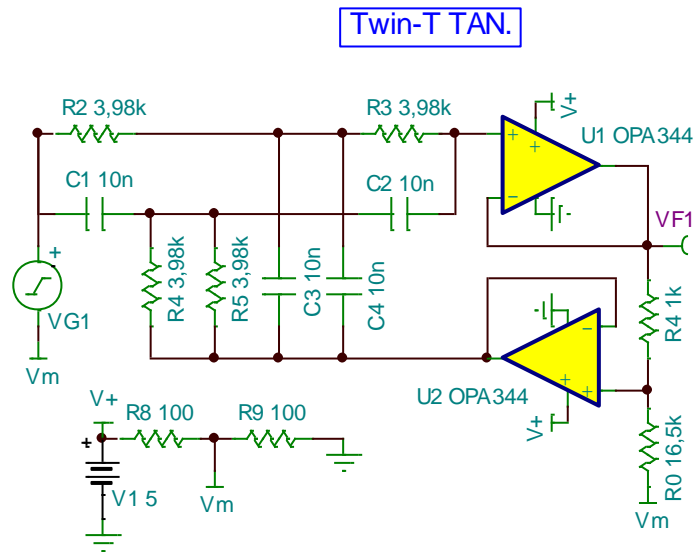
%% Nominal de Multisim.
wm = wmv(2); % Frecuencia del máximo.
ma = mav(2); % Valor del máximo.

Qm2 = vpa(subs(Q), 4)
wp2 = vpa(subs(wp), 4)
wx = wp1 - wp2;
Rx = R0v(1) - R0v(2);
Qx = Qm1 - Qm2;
SQ_R0 = (Qx / Rx) * (R0v(2) / Qm2)
Swp_R0 = (wx / Rx) * (R0v(2) / wp2)
```

Anexo II Circuitos montados y simulados

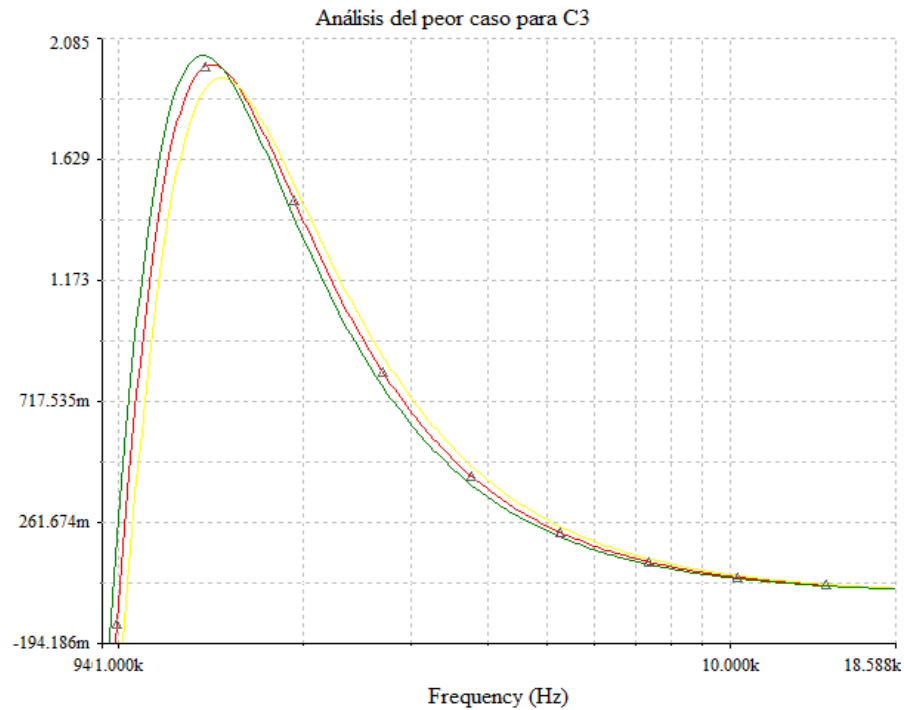
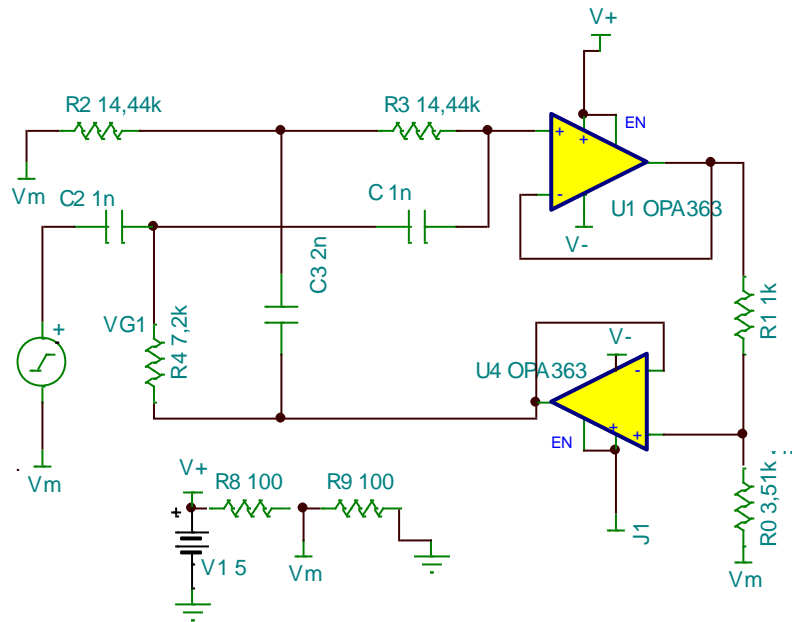
Circuito # 1: TAN

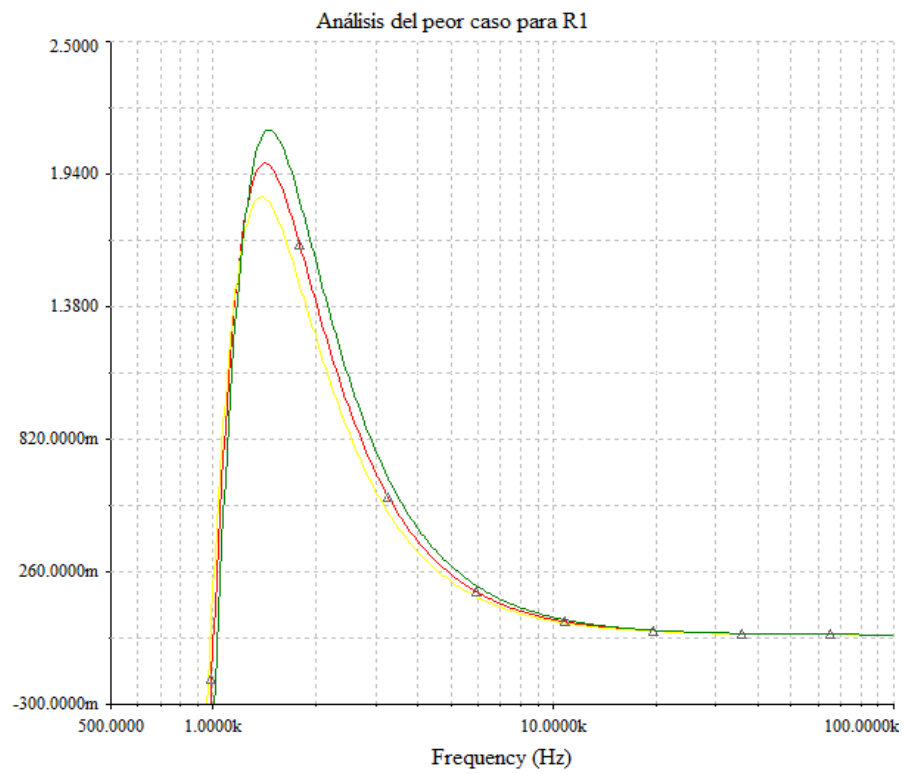
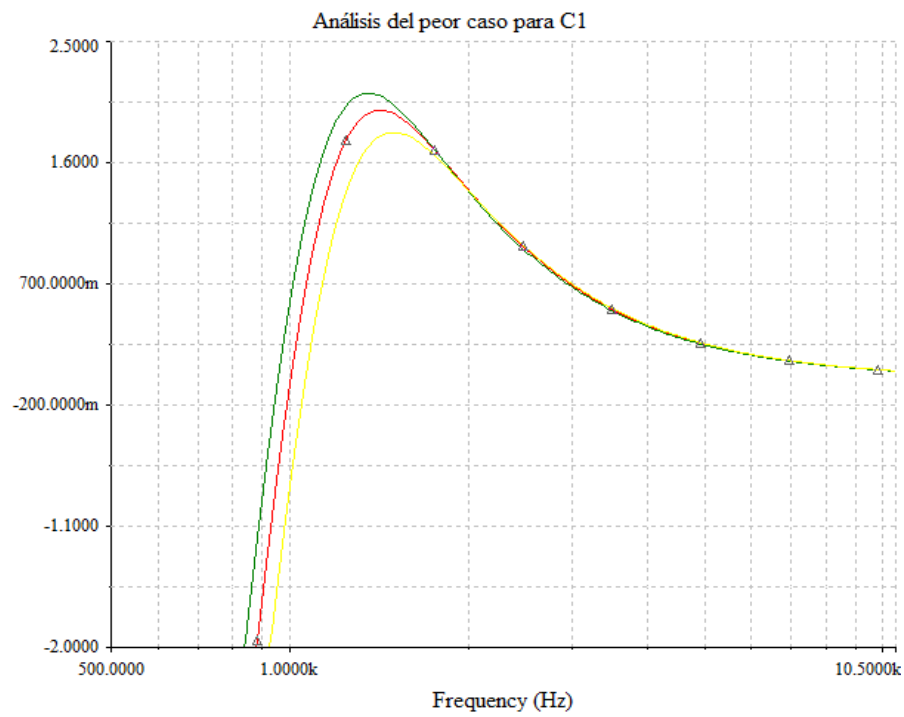
$$H(s) = \frac{(C^2 R^2 s^2 + 1)}{(C^2 R^2 s^2 + (4 C R - 4 C K R) s + 1)}$$



Circuito # 2: TAF-HPF

$$H(s) = \frac{(C^2 R^2) s^2}{(C^2 R^2) s^2 + (4 C R - 4 C K R) s + 1}$$





Circuito # 3: TAF-LPF

$$H(s) = \frac{1}{(C^2 R^2) s^2 + (4 C R - 4 C K R) s + 1}$$

