



BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA

FACULTAD DE CIENCIAS DE LA ELECTRÓNICA

TÍTULO DE TESIS:

ANÁLISIS DE APROXIMACIONES DE INTEGRADORES
FRACCIONARIOS PARA LA CREACIÓN DE OSCILADORES
CAÓTICOS

T E S I S

QUE PARA OBTENER EL TÍTULO DE:

LIC. EN ING. EN MECATRÓNICA

PRESENTA:

CIRO FABIÁN BERMÚDEZ MÁRQUEZ

TUTORES:

DR. JESÚS MANUEL MUÑOZ PACHECO



FCEE

Facultad de Ciencias
de la Electrónica

PUEBLA, PUEBLA, 2020

Agradecimientos

Agradezco a mi familia ...

Agradezco al CONACYT ...

Agradezco a la Facultad de Ciencias de la Electrónica de la BUAP ...

Índice general

Agradecimientos	I
Índice general	IV
Índice de figuras	V
Índice de tablas	VII
Resumen	XI
1. Introducción	1
1.1. Justificación	2
1.2. Objetivos	2
1.2.1. Objetivo general	2
1.2.2. Objetivos específicos	2
2. Fundamentos teóricos	3
2.1. Definición de Grünwald-Letnikov	3
2.1.1. Definición de derivada de Grünwald-Letnikov	4
2.1.2. Definición de integral de Grünwald-Letnikov	4
2.1.3. Método numérico para la definición de GL	4
2.2. Definición de Riemann-Liouville	5
2.2.1. Definición de integral de Riemann-Liouville	5
2.2.2. Definición de derivada de Riemann-Liouville	6
2.3. Transformada de Laplace de integrales y derivadas fraccionarias	6
2.4. Expansión de fracciones continuas (CFE)	6
2.4.1. Análisis de error de la CFE	10
2.5. Escalamiento en frecuencia	14
2.6. Teoría de filtros	15
2.6.1. Filtros de primer orden	15
2.6.2. Filtros de segundo orden	16

3. Implementación	19
3.1. ¿Qué es una FPAA?	19
3.2. Características de la tarjeta y requerimientos	19
3.2.1. Alimentación de la tarjeta	19
3.2.2. Instalación de drivers	20
3.2.3. DIP Switches	20
3.2.4. Filtros Rauch y buffers de salida	21
3.3. AnadigmDesigner2	21
3.4. Implementación con aproximación de primer orden	23
4. Analisis de no se que ahorita	25
A. Códigos	27
B. Diagramas de flujo	29
C. Gráficas de análisis de integrador fraccionario con CFE	31
D. Esquemático de QuadApex v2.0	41
Bibliografía	43

Índice de figuras

2.1. Diagramas de bode comparativos de integrador fraccionario, funciones de transferencia de primer hasta quinto orden.	10
2.2. Diagrama de bode comparativo de función de transferencia escalada. . .	15
2.3. Definición de los parámetros ω_0 y Q de un par de polos conjugados. . .	17
C.1. Diagramas de magnitud de aproximaciones de integrador fraccionario general.	32
C.2. Diagramas de fase de aproximaciones de integrador fraccionario general.	33
C.3. Diagramas de error de magnitud de aproximaciones de integrador fraccionario general.	34
C.4. Diagramas de error de fase de aproximaciones de integrador fraccionario general.	35
C.5. Diagramas de magnitud normalizada de aproximaciones de integrador fraccionario general.	36
C.6. Diagramas de fase normalizada de aproximaciones de integrador fraccionario general.	37
C.7. Diagramas de error de magnitud normalizada de aproximaciones de integrador fraccionario general.	38
C.8. Diagramas de error de fase normalizada de aproximaciones de integrador fraccionario general.	39
D.1. Diagrama esquemático de QuadApex v2.0	41

Índice de tablas

2.1.	Aproximaciones racionales de $\frac{1}{s^{0.5}}$	9
2.2.	Máximo error absoluto de magnitud en dB variando α y orden de función de transferencia.	11
2.3.	Máximo error absoluto de fase en grados variando α y orden de función de transferencia.	11
2.4.	Máximo error absoluto de magnitud normalizado en % variando α y orden de función de transferencia.	12
2.5.	Máximo error absoluto de fase normalizado en % variando α y orden de función de transferencia.	12
2.6.	Promedio de error absoluto de magnitud normalizado en % variando α y orden de función de transferencia.	13
2.7.	Promedio error absoluto de fase normalizado en % variando α y orden de función de transferencia.	13
3.1.	DIP Switches	21
3.2.	CAMs básicos de AD2	22

Lista de códigos

A.1. Función <code>syms2tf</code>	27
A.2. Función <code>cfetf</code>	27

Resumen

Aquí va un resumen bien chido

Capítulo 1

Introducción

El caos se refiere a un tipo de comportamiento dinámico complejo que posee algunas características muy especiales, tales como extrema sensibilidad a pequeñas variaciones de la condición inicial, trayectorias encerradas en el espacio de fase pero con un exponente de Lyapunov positivo, un espectro de potencia continuo entre muchas otras. En pocas palabras, el caos es simplemente un comportamiento impredecible de un sistema determinista. Es de interés resaltar que los sistemas caóticos ya eran conocidos desde hace mucho tiempo atrás y que no fue hasta hace poco que se logró demostrar que el caos puede ser controlado y debido a esto impactar en muchas áreas, tanto en áreas cercanas a la electrónica como técnicas de modulación, sistemas de comunicación, técnicas de encriptación de datos, como también en áreas relacionadas a los sistemas biológicos, reacciones químicas, toma de decisiones críticas en política, economía, eventos militares, etc [1]. El caos es un fenómeno que ocurre en muchos sistemas no lineales, donde la naturaleza determinista de la estructura se conjuga con la irregularidad del comportamiento, esto significa que, a pesar del hecho de que el sistema se describe mediante un conjunto de ecuaciones diferenciales ordinarias, donde todos los términos son perfectamente conocidos, su comportamiento es irregular y muy sensible a las condiciones iniciales. La primera evidencia de imprevisibilidad en los sistemas deterministas se encuentra en el trabajo del matemático y científico Henri Poincaré sobre el movimiento celestial, mientras que la primera formulación del caos en un modelo matemático expresado por un conjunto de ecuaciones diferenciales ordinarias que exhiben el caos se debe al matemático y meteorólogo Edward Lorenz que estaba estudiando un modelo de movimiento del aire en la atmósfera y descubrió cómo pequeñas variaciones en los valores iniciales de las variables de su modelo dieron como resultado predicciones meteorológicas divergentes [2]. Para el momento de estos estudios faltaba una prueba experimental definitiva del caos y la tecnología y poder de cómputo no eran suficientes para pensar aún en soluciones y ni pensar en aplicaciones. Debido al constante avance de la electrónica, hoy en día somos capaces de sintetizar mediante dispositivos electrónicos sistemas caóticos, utilizando técnicas de modelado e implementación es posible

crear representaciones de estos, no obstante, todas estas se basan en aproximaciones que aún no han sido exploradas en su totalidad. Por otro lado el cálculo fraccionario es un tema que tiene más de 300 años de antigüedad y que se remonta a cartas enviadas a Leibnitz por parte de Bernoulli y de L'Hôpital preguntando acerca de la derivada a la $1/2$ e indagando sobre su significado. Con el paso de los años la teoría de cálculo fraccionario se fue desarrollando pasando por las manos de nombres conocidos como Euler, Lagrange, Laplace, Fourier hasta llegar a Liouville, Riemann, Grünwald, Letnikov, Caputo entre muchos otros [3]. Pero aún con todo ese desarrollo no fue hasta hace poco que la comunidad científica comenzó a interesarse por esta rama del cálculo y la razón principal de este cambio es que los cálculos necesarios para cualquier posible implementación eran demasiado complejos y lentos, un panorama totalmente diferente se vive en la actualidad, el rápido avance de la tecnología ha logrado realizar avances notables en esta área. El rango de aplicaciones para el cálculo fraccionario es inmenso, por mencionar algunas de las recientes en los últimos años, la modelación de derivadas fraccionarias para obtener una mejor representación comportamental de un sistema industrial metalúrgico [4], la incorporación de dinámica de orden fraccional para mejorar la robustez de un control PI/PID para motores DC [5, 6], la modelación de señales biológicas como ECG, EMG y EEG debido a su respuesta de magnitud de $20\alpha\text{dB}$, modelos fisiológicos basados en ecuaciones diferenciales lineales que describen fenómenos complejos en el cuerpo humano como la oxigenación de la sangre entre otros [7]. El cálculo fraccionario y los sistemas caóticos se complementan al añadir un nivel de profundidad en la creación de osciladores caóticos modelados como un conjunto de ecuaciones diferenciales no lineales fraccionarias las cuales son materia prima la creación de nuevas aplicaciones y áreas de desarrollo.

1.1. Justificación

1.2. Objetivos

1.2.1. Objetivo general

1.2.2. Objetivos específicos

Capítulo 2

Fundamentos teóricos

Al igual que cuando se comienza a estudiar cálculo de orden entero, es necesario familiarizarse con la notación de los operadores matemáticos de la derivada y la integral. En la actualidad la notación más utilizada para el cálculo entero es la dada por Leibniz en (1686), donde el operador diferencial de n -ésimo orden está definido como: $\frac{d^n}{dt^n}$, D_t^n o simplemente D^n con $n \in \mathbb{N}$. Utilizando el mismo razonamiento, puede definirse su operador inverso (antiderivada) de manera que el operador inverso de la derivada de n -ésimo orden está dado por: ${}_a D_t^{-n}$, donde $n \in \mathbb{N}$ y $a \in \mathbb{R}$ representa el límite inferior del dominio de la región donde se aplica dicho operador.

Para generalizar el operador diferencial e integral para orden fraccionario se considera que este puede definirse para parámetros de orden real o incluso complejo. Esto implica que los operadores pueden definirse respectivamente como: D^α y ${}_a D_t^\alpha$ con $\alpha \in \mathbb{R}$.

Es importante resaltar que no una hay una única definición de operadores diferencial fraccional ni integral sino varias expresiones definidas por diferentes autores, entre las mas usadas se encuentran la definición de Grünwald-Letnikov (GL), la de Riemann-Liouville (RL) y la de Caputo (Ca), cada una de estas con sus ventajas y desventajas desde el punto de vista del análisis matemático, complejidad computacional e implementación [3].

2.1. Definición de Grünwald-Letnikov

Comenzamos considerando que para el caso de orden entero la n -ésima derivada para una función f con $n \in \mathbb{N}$ y $j > n$ está dada por:

$$f^{(n)}(t) = \frac{d^n f}{dt^n} = \lim_{h \rightarrow 0} \frac{1}{h^n} \sum_{j=0}^n (-1)^j \binom{n}{j} f(t - jh) \quad (2.1)$$

donde $\binom{n}{j}$ representa el coeficiente binomial dado por la expresión:

$$\binom{n}{j} = \frac{n!}{j!(n-j)!} \quad (2.2)$$

Considerando valores negativos de n tenemos:

$$\binom{-n}{j} = \frac{-n(-n-1)(-n-2)\cdots(-n-j+1)}{j!} = (-1)^j \begin{bmatrix} n \\ j \end{bmatrix} \quad (2.3)$$

donde $\begin{bmatrix} n \\ j \end{bmatrix}$ esta definido como:

$$\begin{bmatrix} n \\ j \end{bmatrix} = \frac{2(n+1)\cdots(n+j-1)}{j!} \quad (2.4)$$

2.1.1. Definición de derivada de Grünwald-Letnikov

Generalizando la ecuación (2.1) podemos escribir la definición de derivada de orden fraccionario de orden α , ($\alpha \in \mathbb{R}$) como:

$$D_t^\alpha f(t) = \lim_{h \rightarrow 0} \frac{1}{h^\alpha} \sum_{j=0}^{\infty} (-1)^j \binom{\alpha}{j} f(t-jh) \quad (2.5)$$

Para calcular el coeficiente binomial podemos utilizar la relación entre la función Gamma de Euler y el factorial definido como:

$$\binom{\alpha}{j} = \frac{\alpha!}{j!(\alpha-j)!} = \frac{\Gamma(\alpha+1)}{\Gamma(j+1)\Gamma(\alpha-j+1)} \quad (2.6)$$

donde la función Gamma de Euler con $r > 0$ esta definida como:

$$\Gamma(r) = \int_0^{\infty} t^{r-1} e^{-t} dt \quad (2.7)$$

2.1.2. Definición de integral de Grünwald-Letnikov

Utilizando la ecuación (2.5) se puede definir un operador de tipo integral para la función f sobre el dominio temporal (a, t) considerando $n = \frac{t-a}{h}$ donde $a \in \mathbb{R}$ como:

$${}_a D_f^\alpha = \lim_{h \rightarrow 0} \frac{1}{h^\alpha} \sum_{j=0}^{\left[\frac{t-a}{h}\right]} (-1)^j \binom{n}{j} f(t-jh) \quad (2.8)$$

2.1.3. Método numérico para la definición de GL

Utilizando como base la ecuación (2.5) esta se puede discretizar para los puntos kh , ($k = 1, 2, \dots$) de la siguiente manera:

$$({\frac{L_m}{h}})D_{t_k}^\alpha f(t) \approx \frac{1}{h^\alpha} \sum_{j=0}^k (-1)^j \binom{\alpha}{j} f(t_{k-j}) \quad (2.9)$$

donde L_m es el tamaño de memoria (memory length), $t_k = kh$, h es el paso de tiempo del cálculo y $(-1)^j \binom{\alpha}{j}$ son coeficientes binomiales $C_j^{(\alpha)}$ ($j = 0, 1, \dots$). Para su calculo utilizamos la siguiente expresión:

$$C_0^{(\alpha)} = 1, \quad C_j^{(\alpha)} = \left(1 - \frac{1+\alpha}{j}\right) C_{j-1}^{(\alpha)} \quad (2.10)$$

Entonces, la solución numérica general de la ecuación diferencial fraccional:

$${}_a D_t^\alpha y(t) = f(y(t), t) \quad (2.11)$$

puede expresarse como:

$$y(t_k) = f(y(t_{k-1}), t_{k-1})h^\alpha - \sum_{j=1}^k C_j^{(\alpha)} y(t_{k-j}) \quad (2.12)$$

Para el termino de la memoria expresada por la sumatoria, el principio de memoria corta puede utilizarse. Entonces el indice superior de la sumatoria en la ecuación (2.12) se cambiará por ν con las siguientes consideraciones: se usa $\nu = k$ para $k < (\frac{L_m}{h})$ y $\nu = (\frac{L_m}{h})$ para $k \geq (\frac{L_m}{h})$, o sin usar el principio de memoria corta se utiliza $\nu = k$ para toda k .

2.2. Definición de Riemann-Liouville

Para esta definición consideramos la fórmula de Cauchy para la integral repetida que esta dada por:

$$f^{(-n)}(t) = \int_a^t \int_a^{\sigma_1} \dots \int_a^{\sigma_{n-1}} f(\sigma_n) d\sigma_n \dots d\sigma_2 d\sigma_1 = \frac{1}{(n-1)!} \int_a^t \frac{f(\tau)}{(t-\tau)^{1-n}} d\tau \quad (2.13)$$

2.2.1. Definición de integral de Riemann-Liouville

Utilizando las propiedades de la función Gamma de Euler con el factorial y la ecuación (2.13) se puede escribir la definición de integral fraccionaria como:

$${}_a D_t^{-\alpha} f(t) = \frac{1}{\Gamma(\alpha)} \int_a^t \frac{f(\tau)}{(t-\tau)^{1-\alpha}} d\tau \quad (2.14)$$

para $\alpha < 0$ y $a \in \mathbb{R}$. No obstante para el caso de $0 < \alpha < 1$ y $f(t)$ siendo una función casual, esto es, $f(t) = 0$ para $t < 0$, la integral fraccionaria esta definida como:

$${}_0D_t^{-\alpha}f(t) = \frac{1}{\Gamma(\alpha)} \int_0^t \frac{f(\tau)}{(t-\tau)^{1-\alpha}} d\tau, \quad \text{para } 0 < \alpha < 1, \quad t > 0 \quad (2.15)$$

2.2.2. Definición de derivada de Riemann-Liouville

De la ecuación (2.14) se puede escribir la definición de derivada fraccionaria de orden α de la siguiente manera:

$${}_aD_t^\alpha f(t) = \frac{1}{\Gamma(n-\alpha)} \frac{d^n}{dt^n} \int_a^t \frac{f(\tau)}{(t-\tau)^{\alpha-n+1}} d\tau \quad (2.16)$$

donde $(n-1 < \alpha < n)$. Pero igual que con la integral si consideramos $0 < \alpha < 1$ y $f(t)$ una función casual, la derivada de orden fraccionaria se puede reescribir como:

$${}_0D_t^\alpha f(t) = \frac{1}{\Gamma(n-\alpha)} \frac{d^n}{dt^n} \int_0^t \frac{f(\tau)}{(t-\tau)^{\alpha-n+1}} d\tau \quad (2.17)$$

2.3. Transformada de Laplace de integrales y derivadas fraccionarias

La transformada de Laplace de la integral fraccionaria ya sea para Riemann-Liouville o para Grünwald-Letnikov esta definida como:

$$\mathcal{L}\{{}_0D_t^{-p}f(t)\} = s^{-p}F(s) \quad (2.18)$$

y dadas condiciones iniciales cero la transformada de Laplace de la derivada fraccionaria de orden r para Grünwald-Letnikov, Riemann-Liouville y Caputo se reduce a:

$$\mathcal{L}\{{}_0D_t^r f(t)\} = s^r F(s) \quad (2.19)$$

2.4. Expansión de fracciones continuas (CFE)

A una expresión de la forma:

$$a_1 + \frac{b_1}{a_2 + \frac{b_2}{a_3 + \frac{b_3}{a_4 + \ddots}}} \quad (2.20)$$

se le conoce como una fracción continua. En general $a_1, a_2, a_3, \dots, b_1, b_2, b_3$ pueden ser cualquier número real o complejo, y el número de términos pueden ser finito o infinito.

Una manera más conveniente de escribir la ecuación (2.20) es:

$$a_1 + \frac{b_1}{a_2 + \frac{b_2}{a_3 + \frac{b_3}{a_4 + \dots}}} \quad (2.21)$$

y es la que se encontrará normalmente en libros y artículos. Ambas notaciones son muy similar y se puede pasar de una a otra sin mayor complicación.

De la ecuación (2.21) se pueden formar las siguientes fracciones:

$$c_1 = \frac{a_1}{1}, \quad c_2 = a_1 + \frac{b_1}{a_2}, \quad c_3 = a_1 + \frac{b_1}{a_2 + \frac{b_2}{a_3}}, \quad \dots \quad (2.22)$$

las cuales se obtienen, en sucesión, de cortar el proceso de expansión después del primer, segundo, tercer, \dots término. Estas fracciones son llamadas primer, segundo, tercer, \dots convergente, respectivamente, de la fracción continua. El n -ésimo convergente es:

$$c_n = a_1 + \frac{b_1}{a_2 + \frac{b_2}{a_3 + \dots + \frac{b_{n-1}}{a_n}}} \quad (2.23)$$

En 1776 Lagrange obtuvo la expansión de fracciones continuas (CFE) para la ecuación $(1+x)^\alpha$ como se muestra a continuación [8]:

$$(1+x)^\alpha = \frac{1}{1 - \frac{\alpha x}{1 + \frac{\frac{1(1+\alpha)}{1 \cdot 2} x}{1 + \frac{\frac{1(1-\alpha)}{2 \cdot 3} x}{1 + \frac{\frac{2(2+\alpha)}{3 \cdot 4} x}{1 + \frac{\frac{2(2-k)}{4 \cdot 5} x}{1 + \frac{\frac{3(3+\alpha)}{5 \cdot 6} x}{1 + \dots}}}}}}}} \quad (2.24)$$

y escrita de una manera más compacta:

$$(1+x)^\alpha = \frac{1}{1 - \frac{\alpha x}{1 + \frac{\frac{1(1+\alpha)}{1 \cdot 2} x}{1 + \frac{\frac{1(1-\alpha)}{2 \cdot 3} x}{1 + \frac{\frac{2(2+\alpha)}{3 \cdot 4} x}{1 + \frac{\frac{2(2-\alpha)}{4 \cdot 5} x}{1 + \dots}}}}}} \quad (2.25)$$

la ecuación (2.25) puede reescribirse convenientemente multiplicando un m en el numerador y en el denominador como se muestra a continuación:

$$(1+x)^\alpha = \frac{1}{1-} \frac{\alpha x}{1} + \frac{\textcolor{red}{2} \cdot \frac{1(1+\alpha)}{1 \cdot 2} x}{\textcolor{red}{2} \cdot 1} + \frac{\textcolor{blue}{3} \cdot \textcolor{red}{2} \cdot \frac{1(1-\alpha)}{2 \cdot 3} x}{\textcolor{blue}{3} \cdot 1} + \frac{\textcolor{blue}{3} \cdot \frac{2(2+\alpha)}{3 \cdot 4} x}{1} + \dots \quad (2.26)$$

hay que notar que cada denominador esta compuesto por 2 términos, esto se puede ver claramente en la ecuación (2.24), y que contando el término del numerador, m se tiene que agregar en 3 lugares distintos. Si se eligen $m_1 = 2$, $m_2 = 3$, $m_3 = 2$, \dots de manera que se simplifique la ecuación obtenemos:

$$(1+x)^\alpha = \frac{1}{1-} \frac{\alpha x}{1} + \frac{(1+\alpha)x}{2} + \frac{(1-\alpha)x}{3} + \frac{(2+\alpha)x}{2} + \frac{(2-\alpha)x}{5} + \dots \quad (2.27)$$

La ecuación (2.27) se puede encontrar en distintos artículos [9, 10], no obstante para programar un algoritmo que obtenga la aproximación de $(1+x)^\alpha$ hasta el n -ésimo convergente resulta poco intuitiva. Para este fin la ecuación (2.25) resulta más sencilla y contiene un patrón que puede explotarse.

El n -ésimo término de la expansión de fracciones continuas para la ecuación (2.25) se puede calcular utilizando la siguiente ecuación:

$$\frac{\psi(n) [\psi(n) + (-1)^n \alpha]}{(n-1)n} \quad (2.28)$$

donde la función $\psi(x)$ para $x \geq 2$, $x \in \mathbb{Z}^+$ esta definida como¹:

$$\psi(x) = \left\lfloor \frac{x}{2} \right\rfloor \quad (2.29)$$

La ecuación (2.28) se puede utilizar de manera recursiva desde el n -ésimo término hasta el segundo sin olvidar que cada uno de estos siempre debe ir acompañado de la suma de un uno. También vale la pena resaltar que el primer término de la expansión

es $\frac{1}{1-} \frac{\alpha x}{1}$ en conjunto.

Sustituyendo $x = s - 1$ y limitando el número de términos de la ecuación (2.25) obtenemos la aproximación racional para s^α y para obtener la aproximación racional de $\frac{1}{s^\alpha}$ la expresión tiene que ser simplemente invertida. En el apéndice A.2 se muestra un programa en MATLAB que calcula la aproximación para un integrador fraccionario de orden α eligiendo el número de términos n , utilizando el método de CFE descrito previamente.

En general la aproximación utilizando la CFE para un integrador fraccionario $\frac{1}{s^\alpha}$ utilizando los primeros dos términos resulta en una función de transferencia de primer orden como se muestra a continuación:

¹ $\lfloor x \rfloor$ es la función redondeo hacia el entero inferior anterior.

$$\underset{(c_2)}{\frac{1}{s^\alpha}} \approx \frac{(1-\alpha)s + (1+\alpha)}{(1+\alpha)s + (1-\alpha)} \quad (2.30)$$

Al utilizar un número impar de términos el grado del numerador de la función de transferencia siempre será mayor en uno al del denominador, además de que el coeficiente de mayor grado del numerador siempre tendrá signo negativo, esto resulta problemático en la implementación y debido a estas observaciones es recomendable solo trabajar con un número par de términos.

La aproximación de segundo orden tiene la forma:

$$\underset{(c_4)}{\frac{1}{s^\alpha}} \approx \frac{(\alpha^2 - 3\alpha + 2)s^2 + (8 - 2\alpha^2)s + (\alpha^2 + 3\alpha + 2)}{(\alpha^2 + 3\alpha + 2)s^2 + (8 - 2\alpha^2)s + (\alpha^2 - 3\alpha + 2)} \quad (2.31)$$

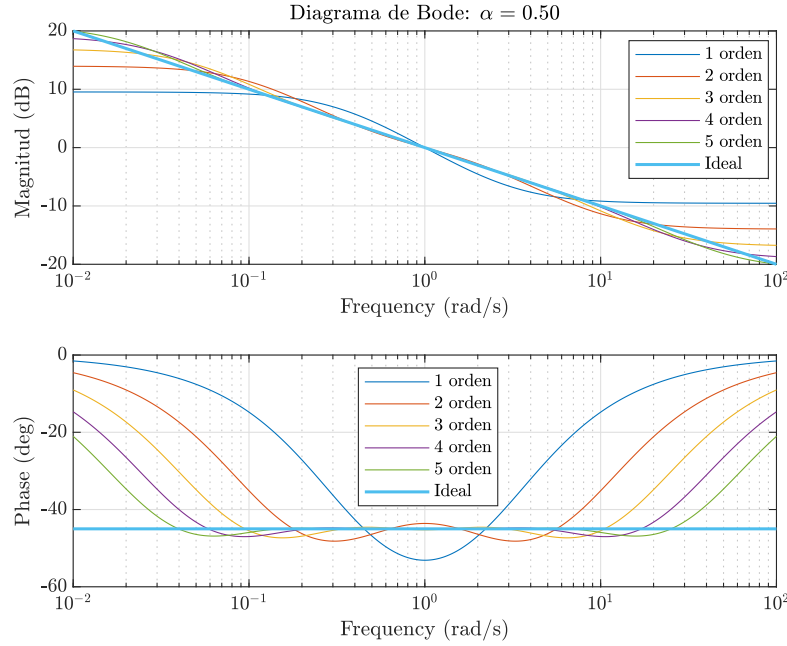
la ventaja de utilizar la aproximación de CFE es que convertimos el problema de orden fraccionario a uno de orden entero de manera sistemática. Por ejemplo para un integrador de orden fraccionario con $\alpha = 0.5$ sus aproximaciones son las mostradas en la Tabla 2.1 y sus correspondientes diagramas de bode se muestran en la Figura 2.1.

Tabla 2.1: Aproximaciones racionales de $\frac{1}{s^{0.5}}$

Orden	No. de términos	Aproximación racional
1	2	$\frac{s+3}{3s+1}$
2	4	$\frac{s^2+10s+5}{5s^2+10s+1}$
3	6	$\frac{s^3+21s^2+35s+7}{7s^3+35s^2+21s+1}$
4	8	$\frac{s^4+36s^3+126s^2+84s+9}{9s^4+84s^3+126s^2+36s+1}$
5	10	$\frac{s^5+55s^4+330s^3+462s^2+165s+11}{11s^5+165s^4+462s^3+330s^2+55s+1}$

Analizando la Figura 2.1 se puede resaltar que en general la aproximación aumenta su ancho de banda conforme el orden de la función de transferencia aumenta y que de mismo modo el error con respecto al integrador ideal disminuye, lo cual era de esperarse, el ancho de banda útil es de 10^{-1} rad/s hasta 10^1 rad/s, es decir aproximadamente dos décadas, no obstante un análisis más profundo es necesario para notar otras peculiaridades que ocurren en este tipo de aproximaciones.

Figura 2.1: Diagramas de bode comparativos de integrador fraccionario, funciones de transferencia de primer hasta quinto orden.



2.4.1. Análisis de error de la CFE

Si variamos el orden α del integrador y el orden de la función de transferencia aproximada podemos notar que entre más pequeño sea α más se alejada la aproximación con respecto a un integrador fraccionario ideal, esto considerando el ancho de banda útil (ver Figuras C.1, C.2). Para cuantificar lo anterior existen dos tipos de errores de interés, el error sin normalizar y el error normalizado.

El error en dB de la magnitud sin normalizar se puede calcular utilizando la siguiente ecuación:

$$\text{error}_{\text{dB}} = 20 \log_{10} \left| \frac{H(j\omega)}{G(j\omega)} \right| \quad (2.32)$$

donde $H(j\omega)$ es el integrador ideal $\frac{1}{s^\alpha}$ y $G(j\omega)$ es la función de transferencia aproximada del integrador utilizando CFE $(c_n) \frac{1}{s^\alpha}$.

El error en grados de la fase sin normalizar se obtiene utilizando la siguiente ecuación:

$$\text{error}_{\text{deg}} = \angle H(j\omega) - \angle G(j\omega) \quad (2.33)$$

Debido a que un integrador fraccionario ideal es en esencia un polo en el origen elevado a una potencia, este tiene un comportamiento de -20α dB/década en magnitud y de -90α grados en fase [11], para normalizar el error este hecho resulta importante ya que la magnitud del integrador ideal en 0.1 rad/seg para cualquier α siempre será 20α dB y para 10 rad/seg de -20α dB, de manera similar la fase permanece constante

en -90α grados para cualquier α .

Entonces la ecuación para el error normalizado de magnitud es:

$$\text{error}_{\text{norm de mag}} = \frac{20 \log_{10} \left| \frac{H(j\omega)}{G(j\omega)} \right|}{20\alpha} = \frac{\log_{10} \left| \frac{H(j\omega)}{G(j\omega)} \right|}{\alpha} \quad (2.34)$$

y la ecuación para el error normalizado de fase es:

$$\text{error}_{\text{norm de fase}} = \frac{\angle H(j\omega) - \angle G(j\omega)}{90\alpha} \quad (2.35)$$

Para poder comparar los errores de manera precisa se debe calcular el valor absoluto de los errores para tener una medida de cuanto se aleja del integrador fraccionario ideal. En las Tablas 2.2 y 2.3 se muestran los errores absolutos máximos sin normalizar para la magnitud y la fase dependiente de α y el orden de la función de transferencia, estos datos se puede corroborar dirigiéndose a las gráficas de las Figuras C.3 y C.4.

Tabla 2.2: Máximo error absoluto de magnitud en dB variando α y orden de función de transferencia.

α/Orden	1 ^{er}	2 ^{do}	3 ^{er}	4 ^{to}	5 ^{to}
0.1	0.4587	0.3333	0.2427	0.0620	0.0220
0.2	0.8931	0.6524	0.4665	0.1172	0.0423
0.3	1.2792	0.9426	0.6527	0.1596	0.0591
0.4	1.5942	1.1889	0.7838	0.1848	0.0710
0.5	1.8155	1.3748	0.8449	0.1904	0.0767
0.6	1.9193	1.4807	0.8259	0.1767	0.0755
0.7	1.8769	1.4676	0.7232	0.1458	0.0669
0.8	1.6464	1.2410	0.5415	0.1021	0.0510
0.9	1.1460	0.7507	0.2939	0.0513	0.0284

Tabla 2.3: Máximo error absoluto de fase en grados variando α y orden de función de transferencia.

α/Orden	1 ^{er}	2 ^{do}	3 ^{er}	4 ^{to}	5 ^{to}
0.1	6.7092	2.8727	0.6548	0.5632	0.2796
0.2	13.2833	5.5250	1.2598	1.0838	0.5338
0.3	19.5614	7.7297	1.7677	1.5208	0.7390
0.4	25.3200	9.2514	2.1362	1.8358	0.8760
0.5	30.2099	9.8632	2.3294	1.9960	0.9311
0.6	33.6307	9.3936	2.3198	1.9760	0.8975
0.7	34.4722	7.8155	2.0880	1.7611	0.7756
0.8	30.6494	5.3543	1.6237	1.3492	0.5739
0.9	19.0601	2.5243	0.9255	0.7528	0.3080

A simple vista se podría creer que el error máximo aumenta conforme aumenta α , no obstante esto es falso debido a que no se esta teniendo en cuenta la escala de la

gráfica y por lo tanto las Tablas 2.2 y 2.3 únicamente nos dan información del total de error y no del porcentaje. Para tomar en cuenta la escala es necesario normalizar el error y de mismo modo calcular el valor absoluto de este. Las gráficas de la respuesta de magnitud y fase normalizadas se pueden encontrar en las Figuras C.5 y C.6, y las gráficas de los errores normalizados en las Figuras C.7 y C.8.

En las Tablas 2.4 y 2.5 se muestran los porcentajes del máximo error absoluto normalizados para la magnitud y la fase, haciendo un análisis de estas se puede llegar a la conclusión de que en efecto **el porcentaje de error es mayor cuanto más pequeño sea α** y esto es de gran interés para poder generar reglas de diseño para la implementación de circuitos integradores fraccionarios.

Tabla 2.4: Máximo error absoluto de magnitud normalizado en % variando α y orden de función de transferencia.

α /Orden	1 ^{er}	2 ^{do}	3 ^{er}	4 ^{to}	5 ^{to}
0.1	22.9364	16.6661	12.1365	3.1006	1.1022
0.2	22.3267	16.3092	11.6624	2.9299	1.0576
0.3	21.3204	15.7107	10.8780	2.6595	0.9850
0.4	19.9281	14.8618	9.7974	2.3094	0.8869
0.5	18.1555	13.7481	8.4493	1.9044	0.7669
0.6	15.9941	12.3390	6.8823	1.4724	0.6290
0.7	13.4063	10.4826	5.1656	1.0415	0.4779
0.8	10.2900	7.7565	3.3847	0.6380	0.3190
0.9	6.3665	4.1706	1.6328	0.2848	0.1578

Tabla 2.5: Máximo error absoluto de fase normalizado en % variando α y orden de función de transferencia.

α /Orden	1 ^{er}	2 ^{do}	3 ^{er}	4 ^{to}	5 ^{to}
0.1	74.5462	31.9184	7.2756	6.2574	3.1070
0.2	73.7962	30.6945	6.9988	6.0212	2.9653
0.3	72.4496	28.6284	6.5472	5.6324	2.7370
0.4	70.3334	25.6983	5.9338	5.0995	2.4334
0.5	67.1331	21.9182	5.1765	4.4355	2.0692
0.6	62.2790	17.3955	4.2958	3.6593	1.6620
0.7	54.7178	12.4056	3.3143	2.7954	1.2312
0.8	42.5686	7.4365	2.2552	1.8739	0.7971
0.9	23.5310	3.1165	1.1426	0.9293	0.3802

Dado que el máximo error ocurre en un rango de frecuencias muy corto es recomendable conocer de igual manera el error promedio, en las Tablas 2.6 y 2.7 se muestra el error promedio normalizado, este tipo de error presenta el mismo efecto de porcentaje de error con respecto a α que el máximo error normalizado. Analizando los datos de las tablas podemos resaltar que en cuanto a magnitud, utilizar una aproximación de segundo orden en lugar de una de primer orden disminuye el error un 5.86 % para $\alpha = 0.1$

y un 2.96 % para $\alpha = 0.9$, en fase el cambio es aún mas notorio, el error disminuye un 26.71 % para $\alpha = 0.1$ y un 5.61 % para $\alpha = 0.9$. No obstante si nos detenemos a pensar detenidamente, con un $\alpha = 0.9$ y una aproximación de primer orden el error promedio es de 4.22 % en magnitud y de 6.55 % en fase, el cual para algunas aplicaciones puede resultar aceptable y cambiar por una aproximación de segundo orden no agregaría gran mejora, lo contrario ocurre con un $\alpha = 0.1$, al cambiar a una aproximación de segundo el error se reduce significativamente, aproximadamente a la mitad en magnitud y a un cuarto en fase, en este caso en definitiva vale la pena cambiar.

Tabla 2.6: Promedio de error absoluto de magnitud normalizado en % variando α y orden de función de transferencia.

α /Orden	1 ^{er}	2 ^{do}	3 ^{er}	4 ^{to}	5 ^{to}
0.1	13.4397	7.5755	2.4058	0.5407	0.2754
0.2	13.1089	7.3398	2.2960	0.5155	0.2639
0.3	12.5614	6.9432	2.1189	0.4751	0.2454
0.4	11.8034	6.3812	1.8827	0.4217	0.2203
0.5	10.8454	5.6502	1.5991	0.3580	0.1897
0.6	9.7075	4.7507	1.2822	0.2875	0.1548
0.7	8.4275	3.6945	0.9478	0.2133	0.1168
0.8	6.8606	2.5119	0.6123	0.1387	0.0772
0.9	4.2249	1.2563	0.2916	0.0667	0.0377

Tabla 2.7: Promedio error absoluto de fase normalizado en % variando α y orden de función de transferencia.

α /Orden	1 ^{er}	2 ^{do}	3 ^{er}	4 ^{to}	5 ^{to}
0.1	34.9335	8.2174	2.6873	1.3313	0.3667
0.2	34.0430	7.8450	2.5975	1.2737	0.3498
0.3	32.5319	7.2389	2.4512	1.1805	0.3226
0.4	30.3547	6.4219	2.2473	1.0556	0.2864
0.5	27.4424	5.4308	1.9840	0.9042	0.2431
0.6	23.6947	4.3175	1.6640	0.7329	0.1948
0.7	18.9857	3.1492	1.2933	0.5488	0.1439
0.8	13.2152	2.0009	0.8821	0.3598	0.0929
0.9	6.5510	0.9388	0.4450	0.1742	0.0441

De ser necesario para convertir el error de magnitud normalizado de % a dB se utiliza la siguiente ecuación:

$$\text{error}_{\text{dB}} = 20\alpha \cdot \left(\frac{\%_{\text{error mag}}}{100} \right) \quad (2.36)$$

y para el error de fase normalizado de % a grados:

$$\text{error}_{\text{deg}} = 90\alpha \cdot \left(\frac{\%_{\text{error fase}}}{100} \right) \quad (2.37)$$

2.5. Escalamiento en frecuencia

El escalamiento en frecuencia es el proceso de correr la respuesta en frecuencia de una red por arriba o por abajo del eje de frecuencia mientras se mantiene igual la impedancia [11].

El escalamiento de frecuencia se consigue multiplicando ésta por un factor de escalamiento k_f mientras se mantiene la impedancia igual. Si consideramos p la variable de frecuencia compleja actual y s la escalada, el proceso de escalamiento se define por la siguiente relación [12]:

$$s = k_f p \quad (2.38)$$

la ecuación 2.12 también se puede ver de la siguiente manera:

$$k_f = \frac{s}{p} = \frac{j\omega'}{j\omega} = \frac{\omega'}{\omega} \quad (2.39)$$

donde ω' es la frecuencia escalada y ω es la frecuencia actual de referencia.

Para entender de mejor manera el escalamiento en frecuencia analicemos el siguiente ejemplo, consideremos la siguiente función de transferencia:

$$N(p) = \frac{20p}{p^2 + 12p + 20} \quad (2.40)$$

la cual corta el eje de la frecuencia en 0.1 rad/s y 200 rad/s, si deseamos que la respuesta en magnitud permanezca idéntica pero desplazada a la derecha en un factor de cien, es decir, que las frecuencias escaladas corten al eje de frecuencia en 10 rad/s y 20k rad/s, entonces $k_f = \frac{10}{0.1} = 100$, sustituyendo la ecuación (2.38) en (2.40):

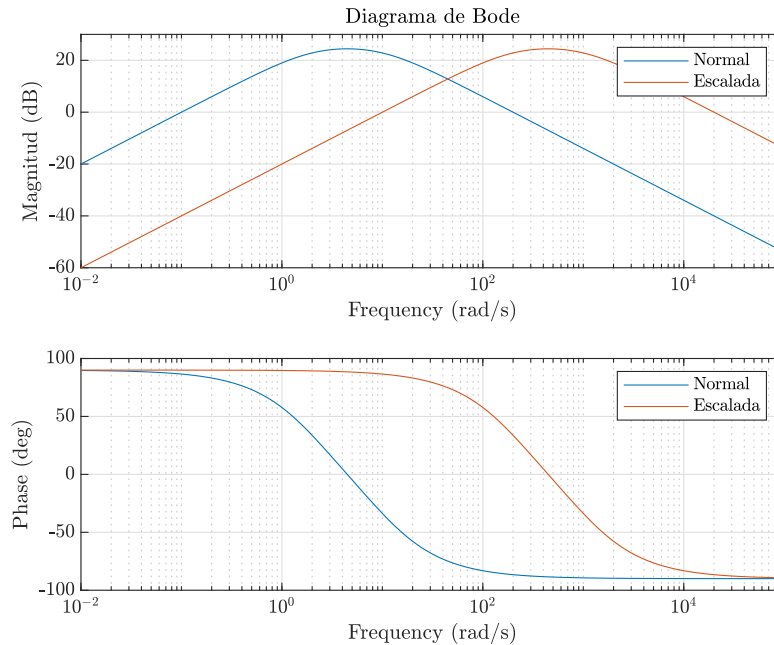
$$N(s) = \frac{200sk_f^{-1}}{s^2k_f^{-2} + 12sk_f^{-1} + 20} = \frac{200k_f s}{s^2 + 12k_f s + 20k_f^2} \quad (2.41)$$

entonces la función de transferencia escalada es:

$$N(s) = \frac{200(100)s}{s^2 + 12(100)s + 20(100)^2} \quad (2.42)$$

En la Figura 2.2 se muestra el resultado de aplicar el escalamiento en frecuencia a la ecuación 2.40. Si se desea aplicar el escalamiento en Hz, simplemente se tiene que tener en consideración el factor de conversión, por ejemplo, si se desea que las frecuencias que corten el eje de la frecuencia sean 0.1 Hz y 200 Hz entonces:

$$k_f = \frac{0.1 \text{ Hz} \cdot \frac{2\pi \text{ rad/s}}{1 \text{ Hz}}}{0.1 \text{ rad/s}} = 2\pi \quad (2.43)$$

Figura 2.2: Diagrama de bode comparativo de función de transferencia escalada.

2.6. Teoría de filtros

2.6.1. Filtros de primer orden

La función de transferencia general de primer orden esta dada por la siguiente ecuación:

$$T(s) = \frac{a_1 s + a_0}{s + \omega_0} \quad (2.44)$$

esta **función de transferencia bilineal** se caracteriza por tener un polo en $s = -\omega_0$, un cero en $s = -\frac{a_0}{a_1}$ y una ganancia de alta frecuencia que tiende a a_1 . Los coeficientes del numerador, a_0 y a_1 , determinan el tipo de filtro, ya sea pasabajas (LP), pasaaltas (HP), pasatodas (AP) o general. La realización activa provee considerablemente más versatilidad que su contra parte pasiva, en muchos casos la ganancia puede ser ajustada a un valor deseado, y algunos parámetros de las funciones de transferencia también sin afectar otros. La impedancia de salida de los circuitos activos es muy baja, haciendo que la colocación en cascada sea muy fácil. Sin embargo, los amplificadores operacionales limitan la operación a altas frecuencias de los circuitos activos [13].

Diferentes tipos de filtros nacen de ubicar el cero de la función de transferencia bilineal general (2.44) en distintos lugares, un cero en el infinito genera un filtro pasabajas de primer orden, y un cero en el origen un filtro pasaaltas. Las ecuaciones básicas de los filtros de primer orden son las siguientes:

Pasabajas (LP)

$$T(s) = \frac{a_0}{s + \omega_0} \quad (2.45)$$

Pasaaltas (HP)

$$T(s) = \frac{a_1 s}{s + \omega_0} \quad (2.46)$$

Pasatodas (AP)

$$T(s) = -a_1 \frac{s - \omega_0}{s + \omega_0} \quad (2.47)$$

2.6.2. Filtros de segundo orden

La función de transferencia general de un filtro de segundo orden (o **bicuadrática**) se expresa en lo general en la forma estándar:

$$T(s) = \frac{a_2 s^2 + a_1 s + a_0}{s^2 + \left(\frac{\omega_0}{Q}\right) s + \omega_0^2} \quad (2.48)$$

donde ω_0 y Q determinan los polos de acuerdo a la siguiente ecuación:

$$p_{1,2} = -\frac{\omega_0}{2Q} \pm j\omega_0 \sqrt{1 - \left(\frac{1}{4Q^2}\right)} \quad (2.49)$$

al parámetro ω_0 se le conoce como la **frecuencia de polo**, y representa la distancia radial entre el origen y el polo en el plano complejo s , a Q se le conoce como el **factor de calidad de polo** y determina la distancia de los polos desde el eje $j\omega$, una representación gráfica de estos parametros se muestra en la Figura 2.3, entre más grande sea el valor de Q , más cerca estarán los polos el eje $j\omega$, y la respuesta del filtro se vuelve más selectiva. Un valor infinito para Q localiza a los polos sobre el eje $j\omega$ y puede producir oscilaciones sostenidas en la realización del circuito. Un valor negativo de Q implica que los polos se encuentran en la mitad derecha del plano s , lo cual ciertamente produce oscilaciones. Los ceros del filtro de segundo orden están determinados por los coeficientes del numerador, a_0 , a_1 y a_2 . Los coeficientes del numerador determinan el tipo de filtro de segundo orden [13].

Para un filtro pasabajas (LP) de segundo orden, los dos ceros están en $s = \infty$ y su respuesta de magnitud muestra un pico para $Q > \frac{1}{\sqrt{2}}$. La respuesta obtenida con $Q = \frac{1}{\sqrt{2}}$ es la respuesta Butterworth o máxima plana. Para un pasaaltas (HP) los ceros están en $s = 0$ y su respuesta de magnitud muestra un pico para $Q > \frac{1}{\sqrt{2}}$. Para un pasabanda (BP) un cero esta en $s = \infty$ y el otro en $s = 0$, la respuesta de magnitud tiene un pico en $\omega = \omega_0$, por lo tanto, la **frecuencia central** del filtro pasabandas es igual a la frecuencia de polo ω_0 . La selectividad de del filtro pasabanda de segundo orden es usualmente medida por su ancho de banda de 3 dB. Este es la diferencia entre

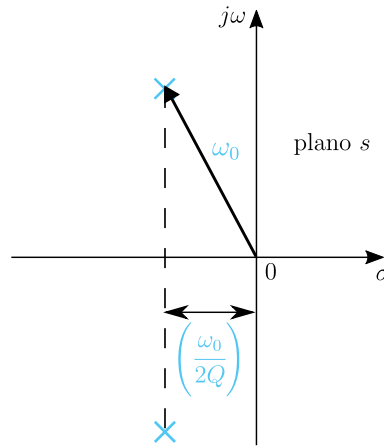
las dos frecuencias ω_1 y ω_2 en las cuales la respuesta de magnitud es 3 dB por debajo de su valor máximo el cual ocurre en ω_0 . La ecuación que describe a las frecuencias ω_1, ω_2 es la siguiente:

$$\omega_1, \omega_2 = \omega_0 \sqrt{1 + \left(\frac{1}{4Q^2}\right)} \pm \frac{\omega_0}{2Q} \quad (2.50)$$

Entonces el ancho de banda BW se define como:

$$BW = \omega_2 - \omega_1 = \frac{\omega_0}{Q} \quad (2.51)$$

Figura 2.3: Definición de los parámetros ω_0 y Q de un par de polos conjugados.



Las ecuaciones básicas de filtros de segundo orden son las siguientes:

Pasabajas (LP)

$$T(s) = \frac{a_0}{s^2 + \frac{\omega_0}{Q}s + \omega_0^2} \quad (2.52)$$

Pasaaltas (HP)

$$T(s) = \frac{a_2 s^2}{s^2 + \frac{\omega_0}{Q}s + \omega_0^2} \quad (2.53)$$

Pasabandas (BP)

$$T(s) = \frac{a_1 s}{s^2 + \frac{\omega_0}{Q}s + \omega_0^2} \quad (2.54)$$

Pasatodas (AP)

$$T(s) = a_2 \frac{s^2 - \frac{\omega_o}{Q}s + \omega_0^2}{s^2 + \frac{\omega_0}{Q}s + \omega_0^2} \quad (2.55)$$

Capítulo 3

Implementación

3.1. ¿Qué es una FPAA?

Una FPAA por sus siglas en inglés (Field Programmable Analog Arrays) es un dispositivo analógico equivalente a las FPGA (Field Programmable Gate Arrays). A diferencia de las FPGA que contienen una gran cantidad de módulos y conexiones que permiten configuraciones arbitrarias de lógica combinatorial y secuencial, los FPAA generalmente contienen una pequeña cantidad de CABs (Configurable Analog Blocks). Los FPAA dirigidos al diseño analógico estándar generalmente presentan un CAB que contiene un amplificador operacional, un arreglo de capacitores programables, y ya sea un arreglo de resistencias programables para circuitos en tiempo continuo o switches configurables para circuitos de capacitores conmutados. Se trabajó con la tarjeta Anadigm QuadApex Development Board v2.0 de la empresa Anadigm, la cual contiene 4 FPAAs AN231E04 que pueden conectarse en cadena y es programada mediante el software AnadigmDesigner2 (AD2). El diagrama esquemático de la tarjeta se puede ver en la Figura D.1 del apéndice.

3.2. Características de la tarjeta y requerimientos

3.2.1. Alimentación de la tarjeta

Para el correcto funcionamiento de la placa esta debe ser alimentada con una fuente de voltaje regulada a 5V de al menos 500mA conectada a la clema de dos terminales. Hay un LED de color verde que indica que la placa se ha encendido correctamente, la placa esta protegida contra la conexión de una fuente de voltaje con la polarización incorrecta.

3.2.2. Instalación de drivers

No conecte la placa QuadApex v2.0 a la PC vía cable USB, ni tampoco inicie AD2 ahora, el driver debe instalarse primero. Se asume que AD2 ya ha sido instalado y registrado en su computadora, de lo contrario entre al link que aparece a continuación.

Para instalar AD2 basta con acceder al siguiente link y seguir los pasos que muestra la pagina:

https://www.anadigm.com/sup_downloadcenter.asp?tab=ad2

es recomendable guardar los datos de registro en un lugar seguro, al iniciar el programa por primera vez es necesario ingresar el **License ID** y la **License key** estos estarán en el correo que le proporcionó a Anadigm.

El driver **CP210x_Drivers.exe** está incluido en el AD2 CD o se puede encontrar en la página de Silicon Labs. Siga los siguientes pasos si es la primera vez que instala este driver, de lo contrario desinstale las versiones anteriores antes de continuar:

1. Ejecute como administrador el ejecutable CP210x_Drivers.exe. El destino por defecto del ejecutable es "C:\Silabs\Mcu\CP210x".
2. Para completar la instalación conecte la QuadApex v2.0 y enciéndala.
3. Acceda al administrador de dispositivos y en **Puertos (COM y LPT)** asegúrese que el driver este bien configurado, si no aparece un signo de admiración y se le asigne un puerto COM la instalación fue exitosa, de lo contrario dar clic derecho sobre el dispositivo y seleccionar **Actualizar controlador**, después buscar el driver en la ruta del paso 1.

El driver en este punto ya está instalado. La instalación del driver y la asignación del puerto es necesaria solo una vez. Si se conecta subsecuentemente a otro puerto de USB de la PC puede ser necesario repetir el paso 3.

3.2.3. DIP Switches

El usuario puede hacer sus propias conexiones en la placa con cables, pero hay un conjunto de DIP switches que permiten una fácil conexión de ciertas rutas entre los FPAA vecinos y entre los FPAA y los **input/output buffers**. Estos interruptores están abiertos por defecto, lo que significa que todos están hacia la izquierda. El usuario puede cerrar los interruptores empujándolos hacia la derecha. En la Tabla 3.1 se muestra un resumen de los switches. Los DIP switches son pequeños y del tipo deslizantes, por lo que se recomienda una herramienta afilada como un destornillador delgado para abrir y cerrar los switches.

Función	Tipo	Labels
Conectar filtros Rauch a FPAA i/ps	4way	S8,9,10,11,15,16,17,18
Conectar entre FPAA's	4way	S2,3,4,5,6,7
Conectar FPAA o/ps a buffers	2way	S12,13,14,19

Tabla 3.1: DIP Switches

3.2.4. Filtros Rauch y buffers de salida

La tarjeta cuenta con 2 buffers de entrada llamados filtros Rauch listos para usar, **Rauch_#1_I01** y **Rauch_#2_I01**, estos son filtros multipropósito no obstante su principal función es convertir una señal single-ended a una diferencial en la FPAA, internamente la FPAA trabaja únicamente con señales diferenciales y debido a esto el uso de estos filtros es imprescindible para introducir señales externas, por ejemplo de generadores de funciones u otros circuitos. Si se usa una señal single-ended, es necesario conectar IN- a GND y conectar la señal a IN+. Para activar el filtro es necesario hacerlo desde el software AD2 haciendo doble clic en la IO cell apropiada (IOCell1-4), seleccionar Input y después Amplifier.

Rauch_#1_I01 está conectado a I/O1 de la FPAA #1 y está configurado con una frecuencia de corte muy alta y ganancia unitaria ($F_o = 490\text{KHz}$)


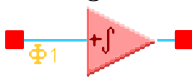

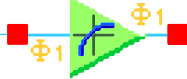
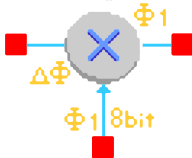
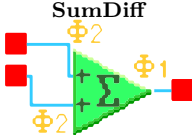

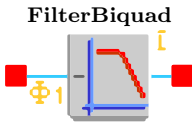
Para habilitar el amplificador en AD2, haga doble clic en la IO cell apropiada (IOCell1-4), seleccione el radio button marcado Input, después seleccione el radio button marcado Amplifier

3.3. AnadigmDesigner2

AD2 trabaja con módulos llamados CAMs (Configurable Analog Modules), estos aportan flexibilidad y sencillez en el proceso de diseño debido a que son bloques que hacen desde funciones sencillas como inversores o comparadores hasta diseños completos como filtros y multiplicadores, los CAMs se pueden interconectar fácilmente unos con otros y únicamente necesitan pequeñas configuraciones para su correcto funcionamiento. Los CAMs más utilizados en el diseño de osciladores caóticos son los mostrados en la Tabla 3.2.

Las configuraciones de los CAMs dependen de las frecuencias de reloj que se seleccionen para cada FPAA. Cada FPAA tiene dos fuentes de frecuencias de reloj principales **Sys1** y **Sys2**, y cinco frecuencias de reloj de chip, desde **Clock 0** hasta **Clock 5** que son subdivisiones de cualquiera de las fuentes de reloj principales. Seleccionar y configurar correctamente los relojes es importante ya que el rango de operación de los CAMs depende de esto.

Tabla 3.2: CAMs básicos de AD2

Nombre	Función de transferencia	Descripción
GainInv 	$\frac{V_{out}(s)}{V_{in}(s)} = -G$	Ganancia inversora. Gain: 0.01 - 100.0 V/V
Integrator 	$\frac{V_{out}(s)}{V_{in}(s)} = \frac{\pm K}{s}$	Integrador con una constante de integración programable. La salida puede ser inversora o no inversora.
Voltage 	$V_{out} = \pm 2$	Referencia de voltaje de ± 2 V.
TransferFunction 		Lookup Table: función de transferencia especificada por el usuario de 256 de pasos de cuantificación.
Multiplier 	$V_{out} = M \cdot V_x \cdot V_y$	V_x es la entrada de voltaje izquierda. V_y es la entrada de voltaje inferior cuantificado de 8 bits. M factor de multiplicación.
SumDiff 	$V_{out} = \pm G_1 V_{in1} \pm G_2 V_{in2} \pm G_3 V_{in3} \pm G_4 V_{in4}$	Las entradas pueden ser inversoras o no inversoras. Cada entrada tiene una ganancia programable. Configurable desde 2 hasta 4 entradas.
FilterBilinear 	Low Pass Bilinear Filter $\frac{V_{out}(s)}{V_{in}(s)} = \pm \frac{2\pi f_0 G}{s^2 + 2\pi f_0 s + 2\pi f_0^2}$ High Pass Bilinear Filter $\frac{V_{out}(s)}{V_{in}(s)} = -\frac{Gs}{s + 2\pi f_0}$ ⋮	Puede ser configurado como pasabajas, pasaaltas, pasatodas o general (Polo y cero).
FilterBiquad 	Low Pass Biquadratic Filter $\frac{V_{out}(s)}{V_{in}(s)} = \frac{\pm 4\pi^2 f_0^2 G}{s^2 + \frac{2\pi f_0}{Q} s + 4\pi^2 f_0^2}$ High Pass Biquadratic Filter $\frac{V_{out}(s)}{V_{in}(s)} = \frac{-Gs^2}{s^2 + \frac{2\pi f_0}{Q} s + 4\pi^2 f_0^2}$ ⋮	Puede ser configurado como pasabajas, pasaaltas, pasabanda, rechazabanda o general (Polos y ceros).

3.4. Implementación con aproximación de primer orden

Para realizar la implementación física de la aproximación de primer orden del integrador fraccionario se utilizó el CAM **FilterBilinear** en su modo **Pole and Zero**, esto debido a su flexibilidad y semejanza con la función de transferencia de la aproximación de la CFE de primer orden. La función de transferencia del CAM en este modo es la siguiente:

$$\frac{V_{\text{out}}(s)}{V_{\text{in}}(s)} = -\frac{G_H(s + 2\pi f_z)}{s + 2\pi f_p} \quad (3.1)$$

donde G_L esta definida como:

$$G_L = \frac{f_z}{f_p} G_H \quad (3.2)$$

y donde G_L es la ganancia en DC, G_H es la ganancia de alta frecuencia, f_p es la frecuencia del polo y f_z es la frecuencia del cero.

La función de transferencia de la aproximación con la CFE de primer orden, como se mencionó anteriormente es la siguiente:

$$\frac{1}{(c_2) s^\alpha} \approx \frac{(1 - \alpha)s + (1 + \alpha)}{(1 + \alpha)s + (1 - \alpha)} \quad (3.3)$$

si consideramos la siguiente sustitución:

$$A = \frac{1 - \alpha}{1 + \alpha} \quad (3.4)$$

podemos reescribir la ecuación (3.3) de la siguiente manera:

$$\frac{1}{(c_2) s^\alpha} \approx \frac{As + 1}{s + A} \quad (3.5)$$

aplicando el escalamiento en frecuencia a la ecuación (3.5) se convierte en:

$$\frac{As + k_f}{s + Ak_f} \quad (3.6)$$

Capítulo 4

Analisis de no se que ahorita

Apéndice A

Códigos

```
1 % Convertir sym a funcion de transferencia
2 function R = syms2tf(G)
3     [symNum,symDen] = numden(G); % Obtener num y den funcion simbolica
4     TFnum = sym2poly(symNum);    % Convertir num sym a polinomio
5     TFden = sym2poly(symDen);    % Convertir den sym a polinomio
6     R = tf(TFnum,TFden);         % Generar funcion de transferencia
7 end
```

Código A.1: Función syms2tf

```
1 function R = cfetf(alfa,n)
2 % Calcula la aproximacion utilizando CFE de un integrador fraccional
3 % 1/s^(alfa)
4 %     alfa: es el orden del integrador
5 %     n    : es el numero de terminos de la aproximacion
6     syms s x;
7     eqns = sym(zeros(n,1));
8     for i=n:-1:2
9         if i == n
10             eqns(i) = 1 + n_term_cfe(i,alfa)*x;
11         else
12             eqns(i) = 1 + (n_term_cfe(i,alfa)*x)/eqns(i+1);
13         end
14     end
15     eqns(1) = 1/(1 - (alfa*x/eqns(2)));
16     derivate = simplify(subs(eqns(1),x,s-1));
17     integrator = collect(1/derivate);
18     sys = syms2tf(integrator);
19 %     pretty(integrator);
20     R = sys;
21 end
22 %% Funciones
23 function R = psi_cfe(x)
24     R = floor(x/2);
25 end
26
27 function R = n_term_cfe(n,k)
28     R = ( psi_cfe(n) * ( psi_cfe(n) + k*(-1)^n ) ) / ( (n-1)*(n) );
29 end
```

Código A.2: Función cfetf

Apéndice B

Diagramas de flujo

Apéndice C

Gráficas de análisis de integrador fraccionario con CFE

Figura C.1: Diagramas de magnitud de aproximaciones de integrador fraccionario general.

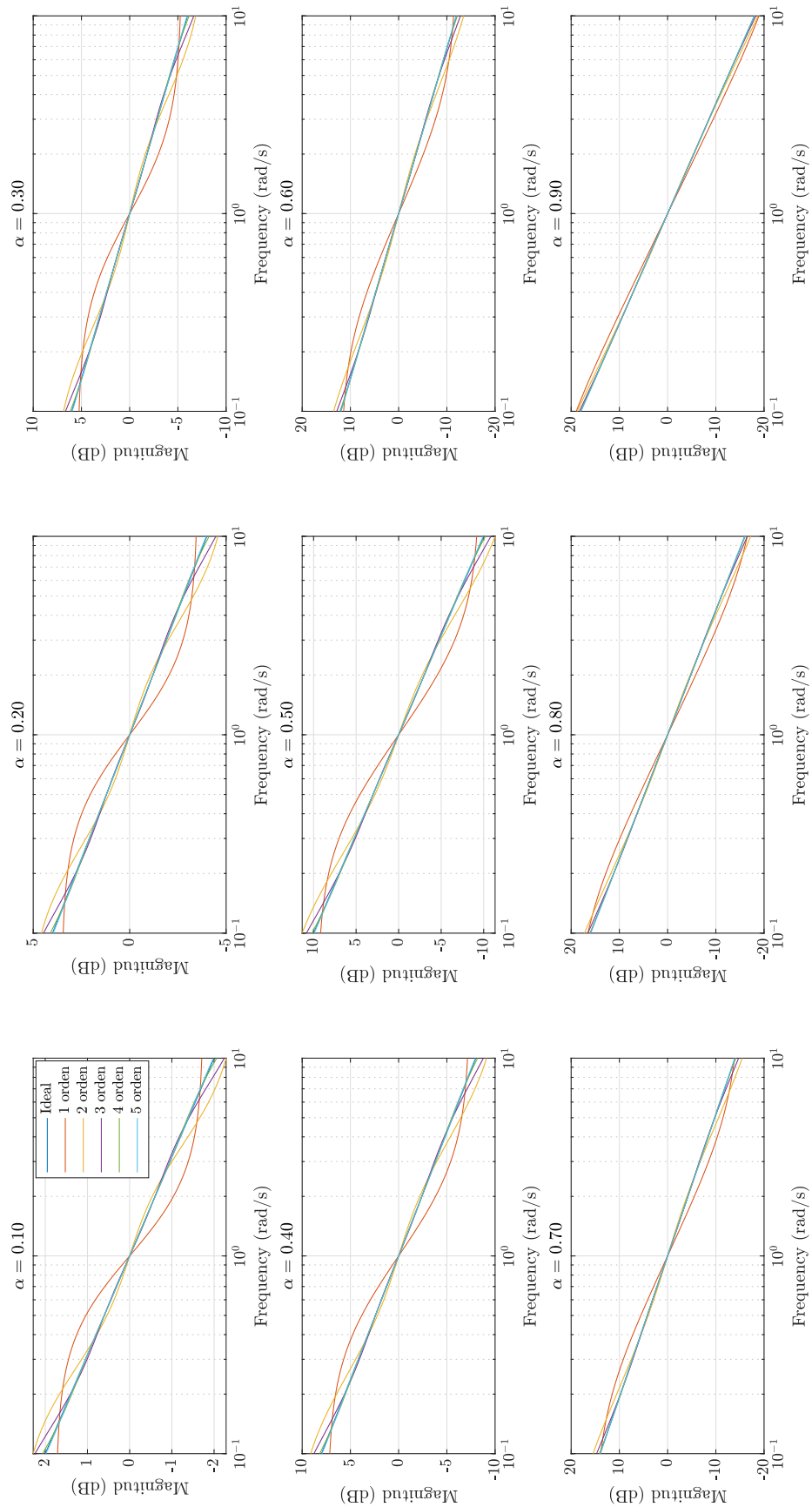


Figura C.2: Diagramas de fase de aproximaciones de integrador fraccionario general.

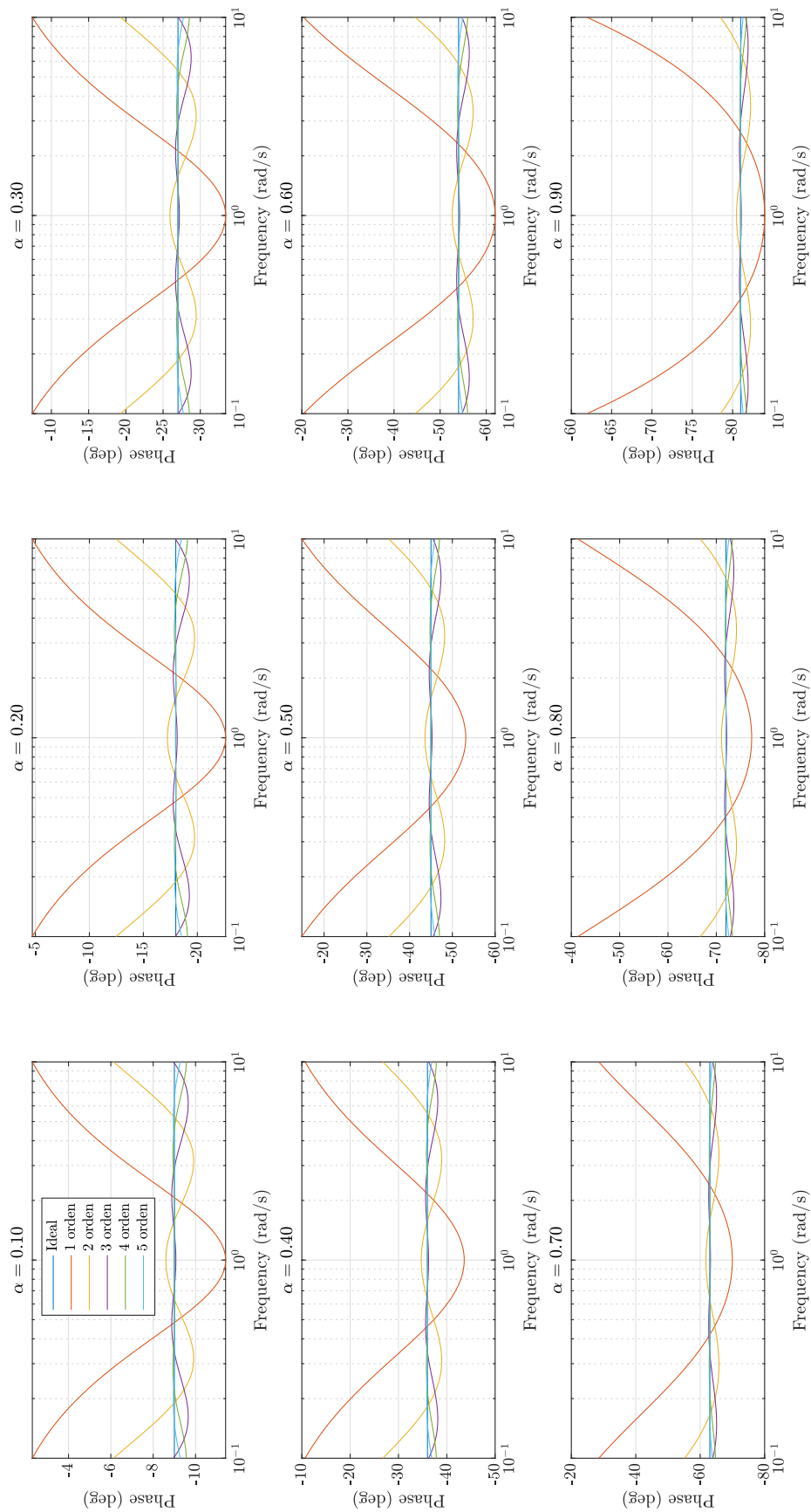


Figura C.3: Diagramas de error de magnitud de aproximaciones de integrador fraccionario general.

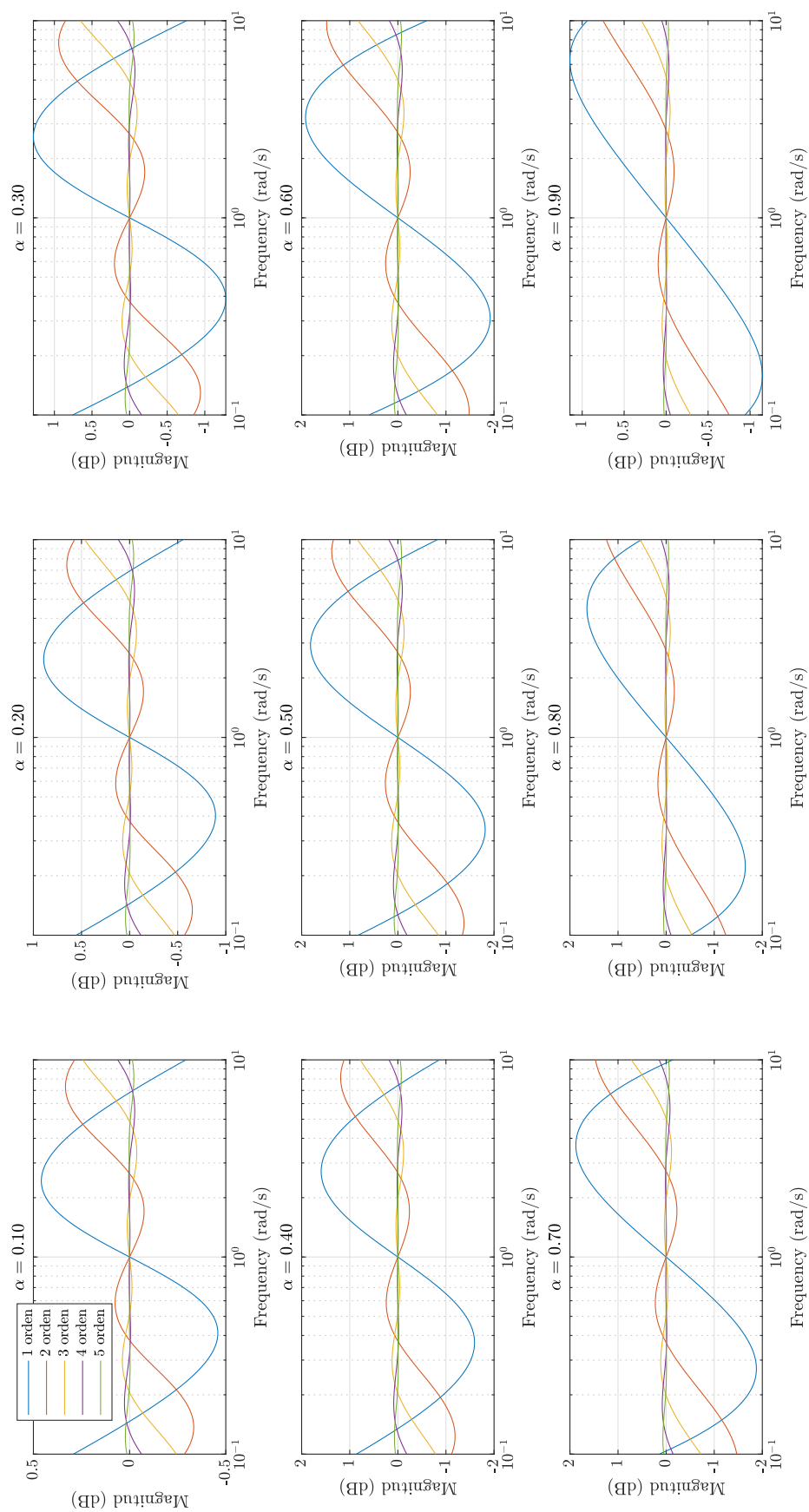


Figura C.4: Diagramas de error de fase de aproximaciones de integrador fraccionario general.

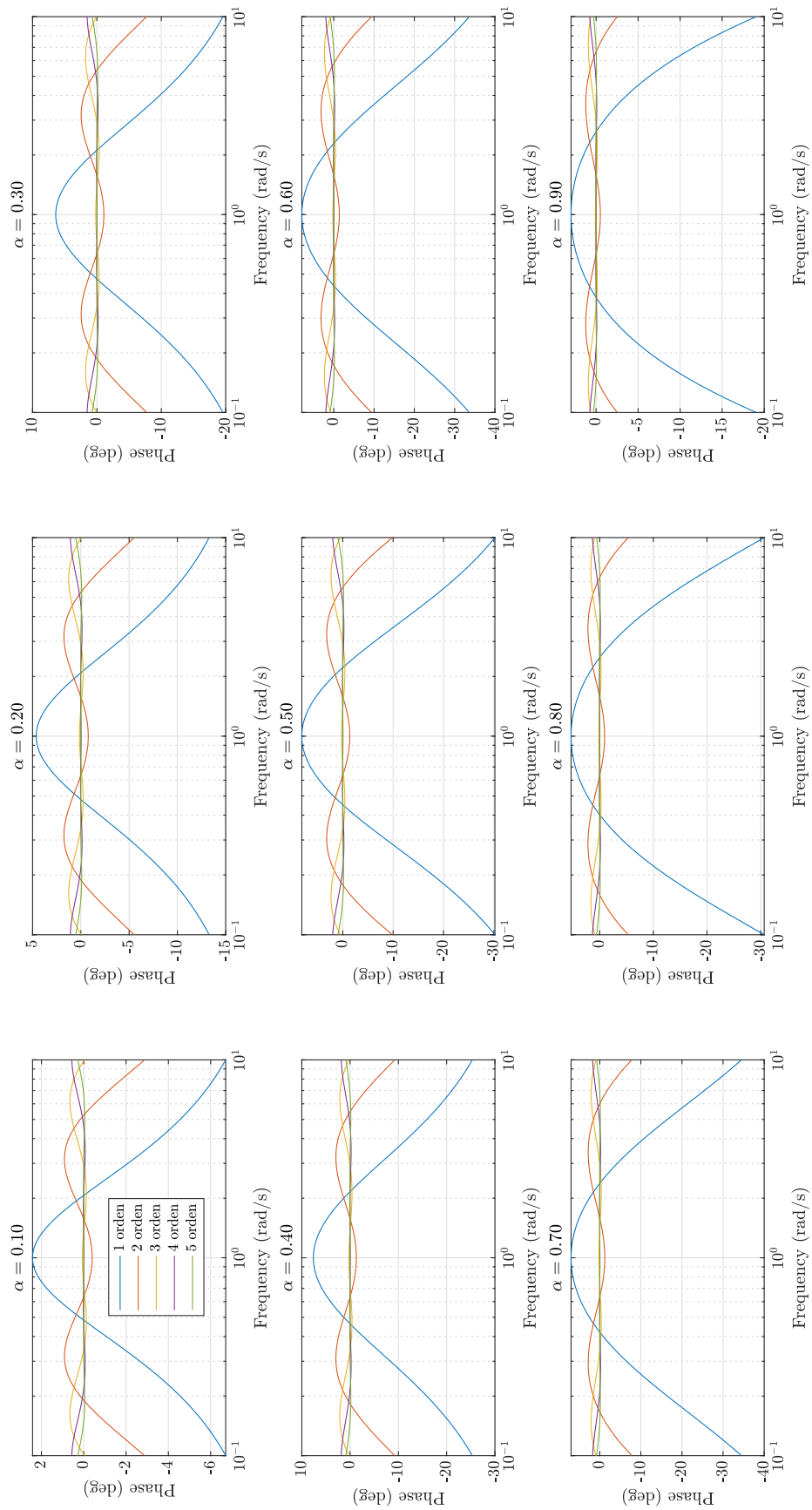


Figura C.5: Diagramas de magnitud normalizada de aproximaciones de integrador fraccionario general.

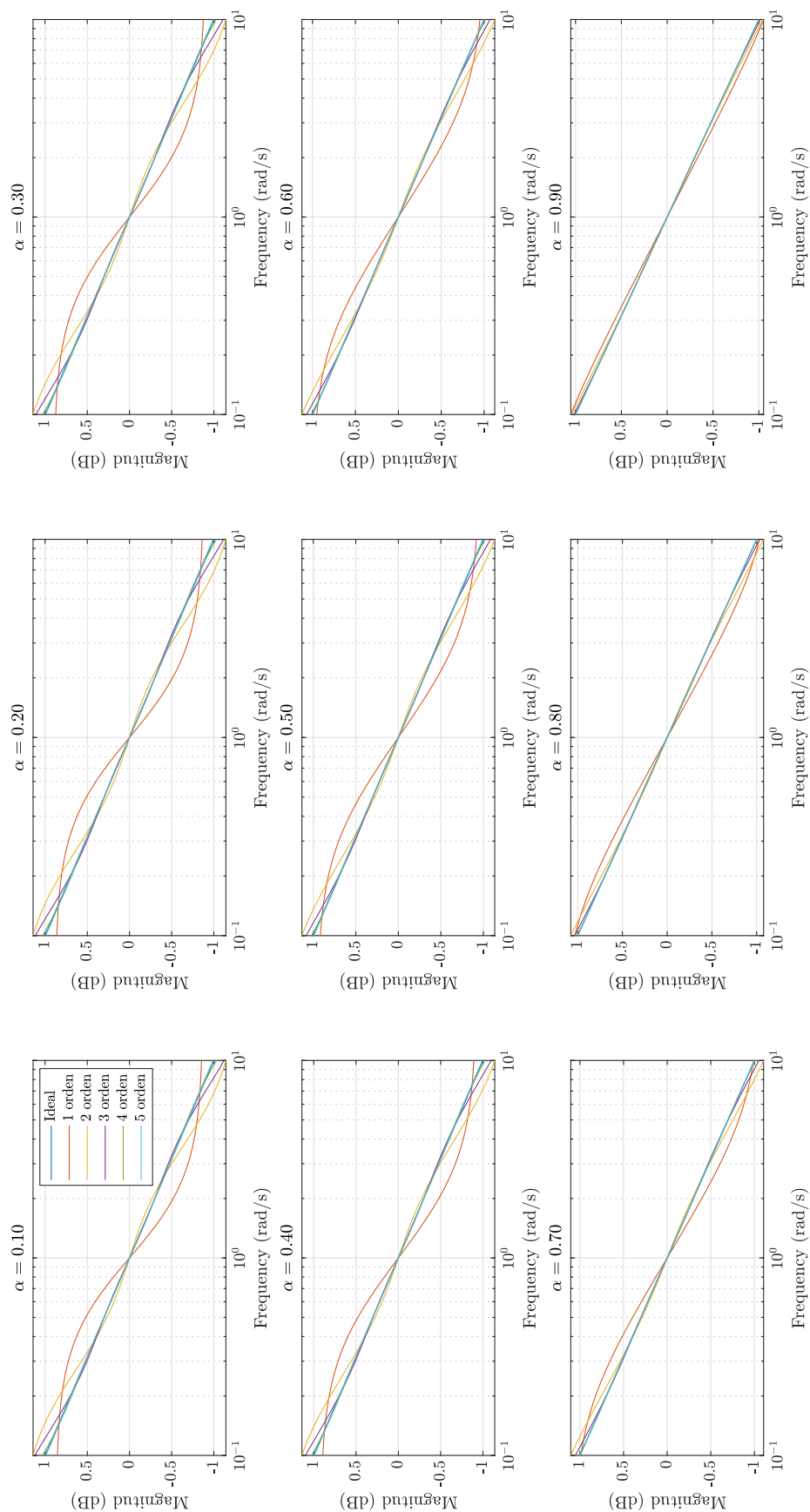


Figura C.6: Diagramas de fase normalizada de aproximaciones de integrador fraccional general.

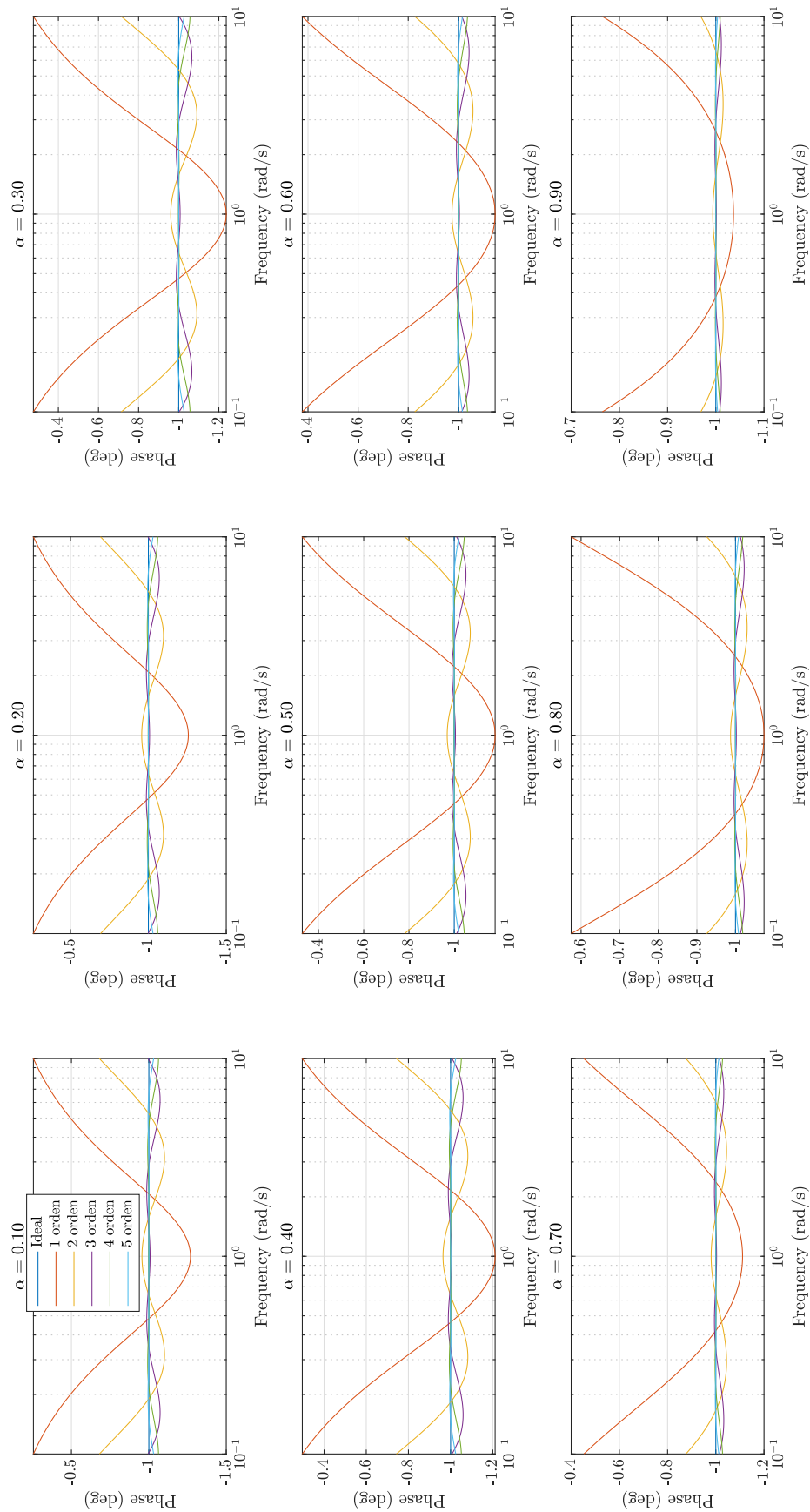


Figura C.7: Diagramas de error de magnitud normalizada de aproximaciones de integrador fraccionario general.

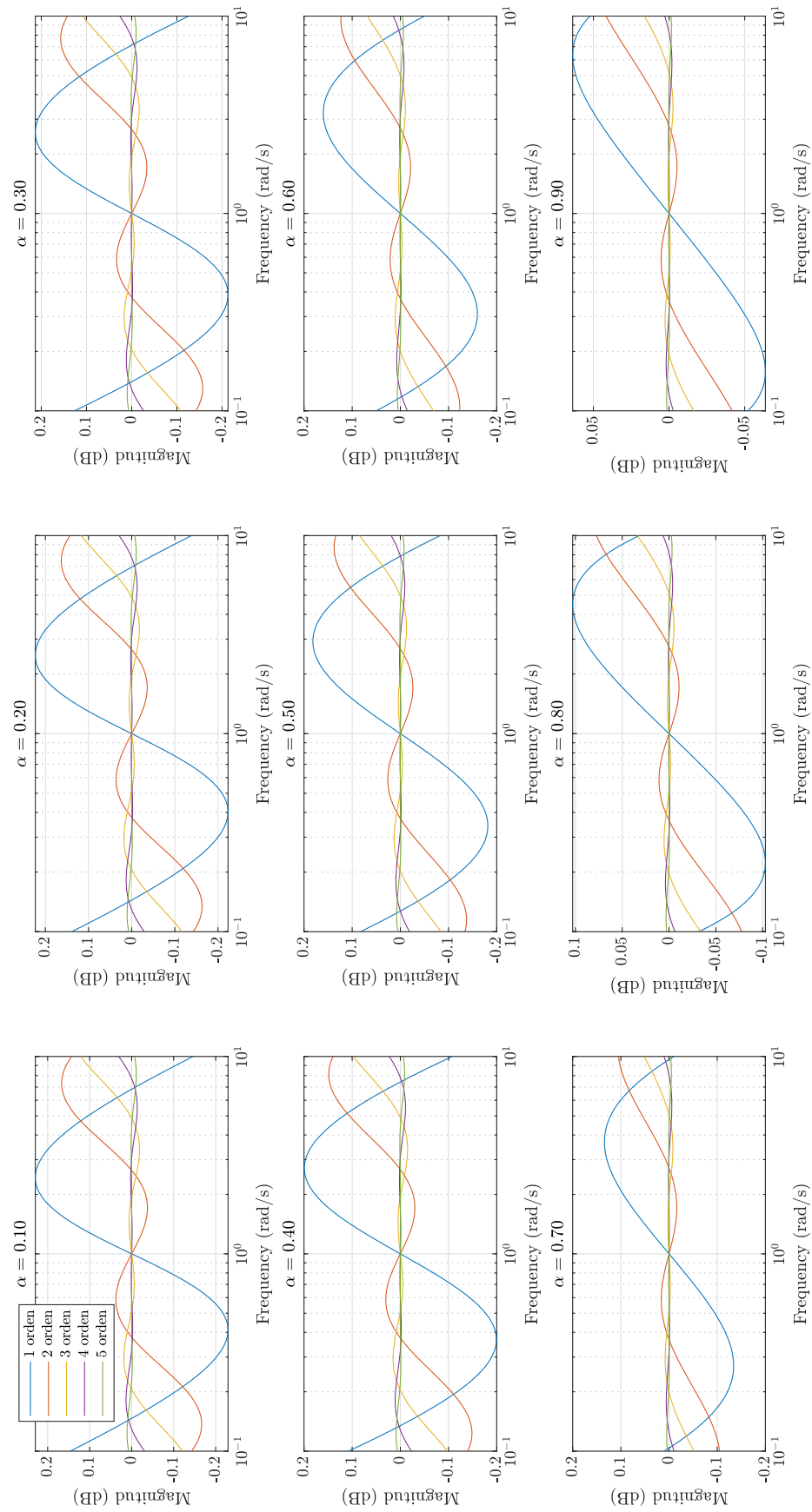
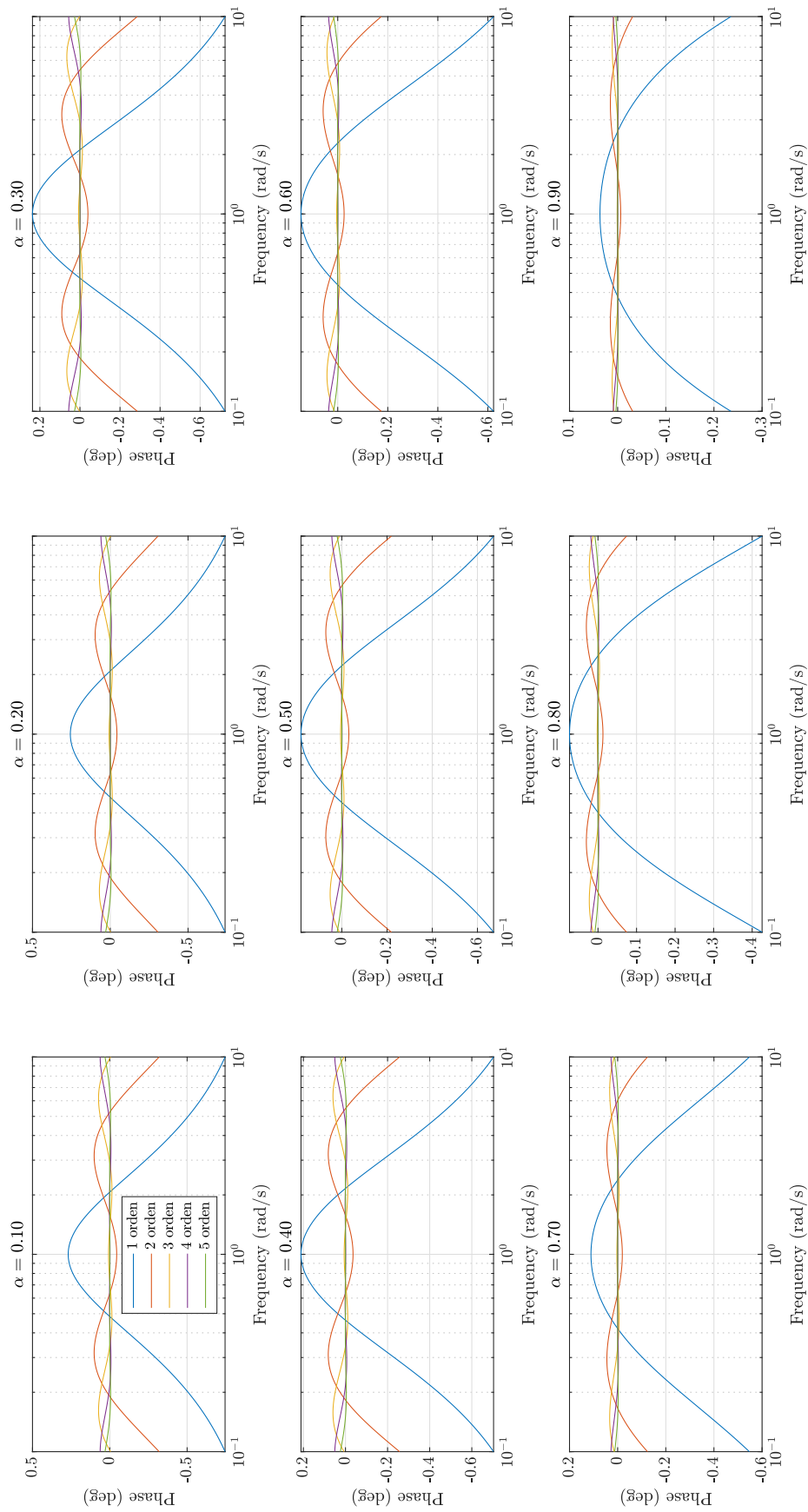
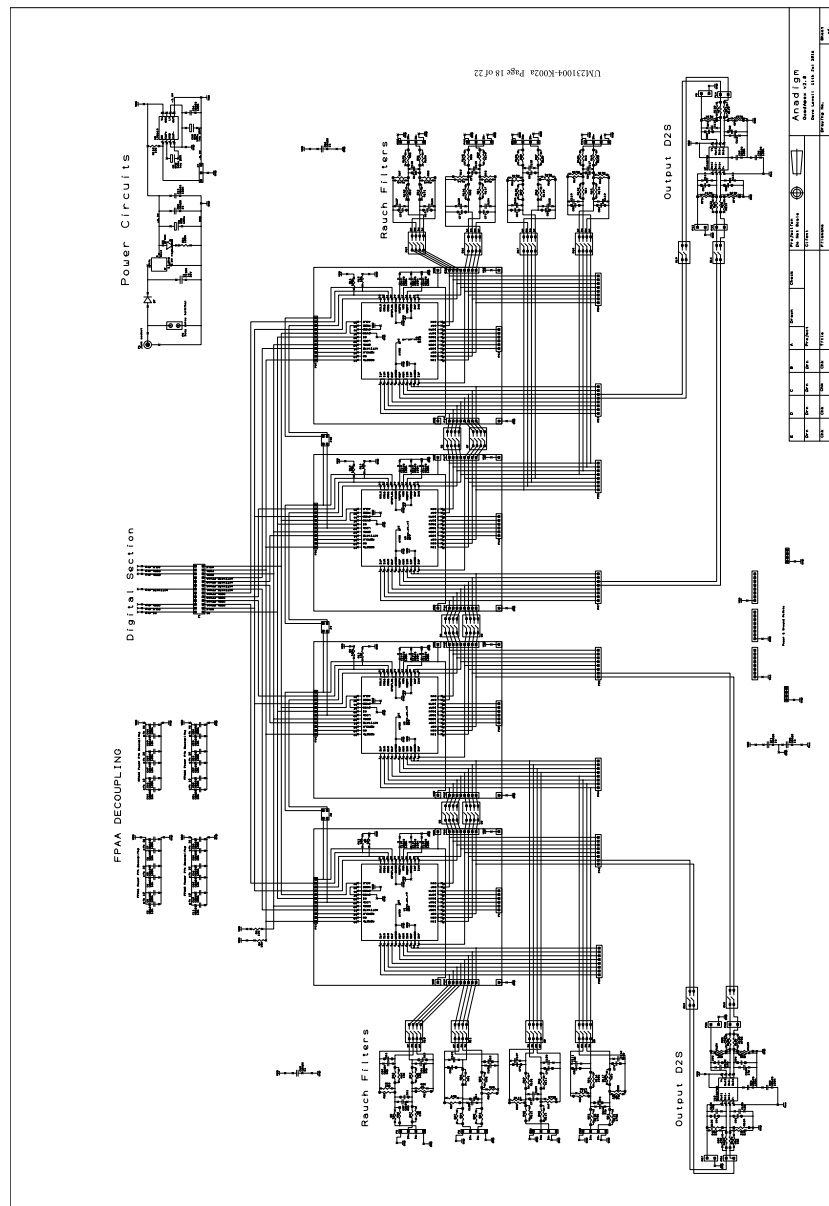


Figura C.8: Diagramas de error de fase normalizada de aproximaciones de integrador fraccionario general.



Esquemático de QuadApex v2.0

Figura D.1: Diagrama esquemático de QuadApex v2.0



Bibliografía

- [1] J. M. M. Pacheco and E. T. Cuautle, *Electronic Design Automation of Multi-Scroll Chaos Generators*. BENTHAM SCIENCE PUB, 2010.
- [2] A. Buscarino, L. Fortuna, M. Frasca, and G. Sciuto, *A Concise Guide to Chaotic Electronic Circuits*. Springer-Verlag GmbH, 2014.
- [3] I. Petráš, *Fractional-Order Nonlinear Systems*. Springer Berlin Heidelberg, 2011.
- [4] I. Petráš and J. Terpak, “Fractional calculus as a simple tool for modeling and analysis of long memory process in industry,” *Mathematics*, vol. 7, p. 511, jun 2019.
- [5] A. Tepljakov, E. A. Gonzalez, E. Petlenkov, J. Belikov, C. A. Monje, and I. Petráš, “Incorporation of fractional-order dynamics into an existing PI/PID DC motor control loop,” *ISA Transactions*, vol. 60, pp. 262–273, jan 2016.
- [6] S. W. Khubalkar, A. S. Junghare, M. V. Aware, A. S. Chopade, and S. Das, “Demonstrative fractional order – PID controller based DC motor drive on digital platform,” *ISA Transactions*, vol. 82, pp. 79–93, nov 2018.
- [7] M. D. Ortigueira, *Fractional Calculus for Scientists and Engineers*. Springer-Verlag GmbH, 2011.
- [8] C. D. Olds, *Continued Fractions*. The Mathematical Association of America, 2009.
- [9] B. T. Krishna and K. V. V. S. Reddy, “Active and passive realization of fractance device of order $1/2$,” *Active and Passive Electronic Components*, vol. 2008, pp. 1–5, 2008.
- [10] B. Krishna, “Studies on fractional order differentiators and integrators: A survey,” *Signal Processing*, vol. 91, pp. 386–426, Mar. 2011.
- [11] M. S. Charles Alexander, *Fundamentals of Electric Circuits*. McGraw-Hill Education, 2016.

- [12] L. P. Huelsman and P. E. Allen, *Introduction to the Theory and Design of Active Filters (Electrical Engineering Series)*. McGraw-Hill Book Company, 1980.
- [13] K. S. Adel S. Sedra, *Microelectronic Circuits*. Oxford University Press Inc, 2015.