

Grado en Ingeniería Electrónica Industrial y Automática

2017-2018

Trabajo de Fin de Grado

“Herramienta de Modelado y Simulación
de Convertidores Sobremuestrados en
MATLAB/Simulink”

Víctor Manuel Medina Huamán

Tutora

Dra. Susana Patón Álvarez

Leganés, a 25 de septiembre del 2018



[Incluir en el caso del interés en su publicación en el archivo abierto]

Esta obra se encuentra sujeta a la licencia Creative Commons **Reconocimiento - No Comercial - Sin Obra Derivada**

RESUMEN

En el presente documento se presenta una nueva herramienta de modelado basado en MATLAB/SIMULINK para el desarrollo de convertidores sobremuestreados . Este tipo de convertidores suelen estar presentes en aplicaciones de comunicaciones y en sensores. De forma tradicional, están formador por OTAs, sin embargo, con el avance en los procesos CMOS de las últimas décadas, la parte analógica se ha vuelto complicada en las tecnología nanométricas actuales. Por el contrario, la parte digital se beneficia de esta empequeñecimiento y por lo tanto, una nueva alternativa ha surgido, convertidores ADC basados en VCO

Debido a la naturaleza del proceso de manufacturación, una serie de pasos son necesario para llegar a un circuito final. Uno de estos primeros pasos en el diseño ocurre a nivel de sistema, y precisamente es el foco de atención del presente trabajo. Esta parte consiste en el modelado comportamental que busca un continuo balance entre precisión y rendimiento de las simulaciones: la primera necesita ser adecuada para asegurar un modelo fiable y la segunda tiene que ser lo suficientemente rápida dado que se necesitan realizar un gran número de simulaciones. Además, con cada iteración en el modelo, el balance se debe mantener. Así, surge la necesidad de una herramienta que facilite esta tarea y así, se propone en el presente documento una biblioteca en Simulink con una serie de componentes que modelan elementos clave, esto es, OTAs y VCOs, en un convertidor ADC sobremuestreado. Un estudio y descripción de los elementos se presentan, así como la implementación en un modelo comportamental completo de un ADC basado en VCO.

Palabras clave:ADC, Delta Sigma, MATLAB, SIMULINK, ADC-VCO, Modelado comportamental.

ABSTRACT

This work presents a new tool based in MATLAB/SIMULINK for the design of over-sampled Analog-to-Digital data Converters (ADC) in continuous-time at system level. These type of data converters are usually present in communications applications and some sensors. Traditionally, they are constituted by OTAs, albeit with the advances in the last decades in the CMOS processes, the analog part have become troublesome in current nanometer technologies. Conversely, the digital part benefits from this shrink and consequently a new alternative has appeared, that is, VCO-based ADC.

Given the nature of the manufacturing process, a series of steps are needed in order to yield the final circuit. One of the first steps in the design occurs at system level, and precisely that is the one in which this work focuses on. This stage consists on a behavioral model that craves for a constant balance between accuracy and speed simulation: the former needs to be adequate in order to assure a trustworthy model and the latter needs to be fast enough given that a notable numbers of simulation runs are going to be performed. Furthermore, with each iteration on the model the balance must be kept. Thus, the need for a tool that eases this task arises and therefore, here it is proposed a Simulink library with a series of components, modelling some key elements, that is, OTAs and VCOs, in an oversampled ADC. A study and description of the elements are presented, along with an implementation on a full behavioral model of an ADC.

It is noted that as a consequence of a inherent property of the system level, the library is neither limited nor linked to a certain CMOS process, so it can be applied, pretty much, when required. Furthermore, although in the present document the application highlighted is on oversampled data-converters, the library can be of use on another area of application as long as it fulfills the requirements needed by the designer.

Palabras clave:ADC, Delta Sigma, MATLAB, SIMULINK, ADC-VCO, Behavioural modelling.

DEDICATORIA

Quiero dar las gracias a todas aquellas personas que han contribuido en llegar a donde estoy.

Primeramente agradezco a mis padres por su apoyo y cariño durante todos estos años de grado.

Agradezco enormemente a Susana, mi tutora, por su dedicación y esfuerzo en explicarme esto de los convertidores Delta-Sigma y ofrecerme la oportunidad de poder participar en proyectos de Investigación.

Finalmente agradezco a todos aquellos amigos del grupo 21, especialmente a Oriol, por revisar y ayudarme en la revisión del presente documento.

Estas pobres palabras no son capaces de expresar mi enorme agradecimiento.

ÍNDICE GENERAL

1. INTRODUCCIÓN.	1
1.1. Motivación del trabajo	1
1.2. Objetivos	2
1.3. Marco Regulador.	2
1.4. Estructura del documento	3
2. ESTADO DE LA TÉCNICA.	4
2.1. Proceso de diseño de un circuito integrada de señal mixta	4
2.2. Modelado de ADCs sobremuestreados.	7
3. MODELADO COMPORTAMENTAL DE CIRCUITOS ADCS BASADOS EN VCOS	9
3.1. Fundamentos convertidores ADC.	9
3.1.1. Convertidor ADC	9
3.1.2. Convertidor ADC sobremuestreado con conformado espectral de ruido	13
3.1.3. Modulador $\Sigma\Delta$ de primer orden	13
3.2. Convertidor ADC basado en VCO	16
3.2.1. Convertidor $\Sigma\Delta$ ADC de primer orden basado en VCO	16
3.3. Integrador RC	19
3.3.1. Ganancia finita, primer y segundo polo	23
3.3.2. Condensador parásito	24
3.3.3. Integrador con <i>feedforward</i>	25
3.4. Integrador RC con dos, tres y cuatro entradas	26
3.4.1. Ganancia finita, primer y segundo polo	27
3.4.2. Condensador parásito a la entrada	29
3.4.3. Integrador <i>feedforward</i>	31
3.4.4. Oscilador en anillo	33
3.5. Biblioteca desarrollada	35
3.5.1. Integradores.	35
3.5.2. Integradores con múltiples entradas	39
3.5.3. Oscilador en anillo controlado por tensión.	39

3.5.4. Validación de la biblioteca	42
4. USO DE LA BIBLIOTECA PARA LA CARACTERIZACIÓN DE UN VCO- ADC	45
4.1. Arquitectura de convertidor ADC basado en VCO con frecuencia de muestreo aumentada.	45
4.2. Modelo comportamental del VCO-ADC.	47
4.3. Velocidad y Ganancia del OTA	49
4.3.1. Ganancia por Ancho de banda	50
4.3.2. Ganancia	52
4.4. Sensibilidad del ADC a la resolución y no linealidad del VCO y a la velocidad de muestreo.	53
4.4.1. Cambios en la resolución del VCO y en la velocidad de muestreo	53
4.5. VCO con comportamiento no lineal	55
5. CONCLUSIONES Y TRABAJO FUTURO	59
5.1. Conclusiones	59
5.2. Trabajo futuro	60
6. ENTORNO SOCIOECONÓMICO	61
7. PRESUPUESTO Y PLANIFICACIÓN	62
7.1. Presupuesto	62
7.1.1. Coste del personal	62
7.1.2. Coste de licencias	62
7.1.3. Presupuesto total.	63
7.2. Planificación	63

ÍNDICE DE FIGURAS

2.1	Ejemplo de SoC [1].	5
2.2	Flujo de diseño de un circuito de señal mixta [2].	6
2.3	Floor plan de un chip Mitsubishi M30620EFCP 16-BIT CMOS MICRO-COMPUTER. [3]	6
3.1	Diagrama de bloques de un convertidor ADC [4].	9
3.2	Error de cuantificación.	10
3.3	Función de densidad de probabilidad del error de cuantificación	11
3.4	Comparación entre la potencia de ruido para convertidores Nyquist y para convertidores sobremuestreados [15].	12
3.5	Diagrama de bloques de un convertidor ADC $\Sigma\Delta$ en tiempo discreto [15].	13
3.6	Modulador $\Delta\Sigma$ de primer orden en tiempo discreto.	14
3.7	Modelo lineal de un modulador $\Delta\Sigma$ de primer orden en tiempo discreto.	14
3.8	Conformado de ruido para distinto orden de moduladores [15].	15
3.9	Formas de implementar un integrador [5]	16
3.10	Diagrama de bloques de un modulador $\Sigma\Delta$ en bucle abierto.	17
3.11	Diagrama de bloques VCO-ADC de primer orden en bucle abierto y su correspondiente modelo linealizado en el dominio de la frecuencia [6].	17
3.12	Diagrama de bloques VCO-ADC que incluye la no linealidad de la ganancia del VCO. [6].	18
3.13	ADC-VCO con bucle de realimentación para corregir la no linealidad del VCO [6]	18
3.14	(a) Integrador. (b) Integrador con camino feedforward.	20
3.15	(a)Circuito con un operacional. (b)Circuito equivalente	21
3.16	Modelado de un integrador mediante diagrama de bloques.	22
3.17	Circuito equivalente a un integrador con ganancia finita y/o presencia de polos en el operacional.	23
3.18	Circuito equivalente a un integrador con efectos parásitos capacitivos.	24
3.19	Circuito equivalente a un integrador <i>feedforward</i>	25

3.20	Circuito equivalente a un integrador <i>feedforward</i> con efectos parásitos capacitivos.	26
3.21	Diagrama de bloques equivalente para un integrador de dos entradas.	27
3.22	(a) Integrador RC con dos entradas. (b) Integrador <i>feedforward</i> RC	27
3.23	Circuito equivalente al de la figura 3.22 sin considerar efectos capacitivos, considerando que Z_2 es una resistencia.	28
3.24	Circuito equivalente al de la figura 3.22 sin considerar efectos capacitivos, considerando que Z_2 es un condensador.	29
3.25	Circuito equivalente al de la figura 3.22 considerando efectos capacitivos, considerando que Z_2 es un condensador.	30
3.26	Circuito equivalente al de la figura 3.22 considerando efectos capacitivos, considerando que Z_2 es un condensador.	31
3.27	Circuito equivalente al de la figura 3.22 con una resistencia en el camino de realimentación, sin considerar efectos capacitivos parásitos, y siendo Z_2 una resistencia.	32
3.28	Circuito equivalente al de la figura 3.22 con una resistencia en el camino de realimentación, sin considerar efectos capacitivos parásitos, y siendo Z_2 un condensador.	32
3.29	Circuito equivalente al de la figura 3.22 con una resistencia en el camino de realimentación, sin considerar efectos capacitivos parásitos, y siendo Z_2 un condensador.	33
3.30	Oscilador en anillo con 5 inversores.	33
3.31	Modelo oscilador en anillo. Se muestran dos inversores de una cadena.	34
3.32	Ejemplo de creación de subsistema.	36
3.33	Imagen del elemento Integrador 1 entrada en SIMULINK	36
3.34	Captura de la interfaz del elemento Integrador 1 entrada en SIMULINK	37
3.35	Captura de la pestaña <i>Initialization commands</i> . Se muestra parte del código utilizado.	38
3.36	Interfaz para la introducción de parámetros para los integradores de 3 y 4 entradas.	40

3.37	Modelo oscilador en anillo. En la parte superior se muestran dos inversores de un oscilador en anillo; en la parte inferior se muestra un modelo compacto de un oscilador en anillo de 5 inversores.	41
3.38	Subsistema del VCO y su correspondiente máscara	41
3.39	Ejemplo de un VCO con 64 etapas creado con la herramienta.	42
3.40	Captura que muestra la pestaña <i>Initialization commands</i> y parte del código que se encuentra en ella	43
3.41	Diferentes casos y diferencias máximas obtenidas	44
4.1	Oscilador en anillo con matriz de muestreo mediante cadena de retrasos y circuito decodificador. [7]	45
4.2	Convertor ADC basado en VCO con frecuencia de muestreo aumentada.	47
4.3	Modelo de oscilador en anillo con 5 etapas.	48
4.4	Matriz de sobremuestreo y circuito de decodificación.	49
4.5	Espectro de salida obtenido para condiciones de simulación nominales.	50
4.6	(a) Barrido de GBW_{OTA} para un valor alto pero fijo de ganancia. (b) Barrido de A_{vOTA} habiendo seleccionado un valor de $GBW_{OTA} = 10\text{GHz}$	51
4.7	Espectro de salida para $GBW_{OTA} = 10\text{GHz}$, $\text{SNR} = 68\text{dB}$	52
4.8	Espectro de salida para $GBW_{OTA} = 10\text{GHz}$, $A_{vOTA} = 32\text{dB}$	53
4.9	Espectro de salida para $GBW_{OTA} = 10\text{GHz}$, $A_{vOTA} = 32\text{dB}$, segundo polo en 12GHz	54
4.10	Espectro de salida para los diferentes casos de número de etapas y número de relojes.	56
4.11	Comparación entre la relación ideal y un polinomio para un oscilador en anillo en 28nm	56
4.12	Espectro de salida para el convertidor ADC basado en VCO con una OTA con $GBW_{OTA} = 10\text{GHz}$, $A_{vOTA} = 32\text{dB}$, segundo polo en 12GHz	57
4.13	Rango Dinámico para el convertidor ADC basado en VCO con una OTA con $GBW_{OTA} = 10\text{GHz}$, $A_{vOTA} = 32\text{dB}$, segundo polo en 12GHz	57
7.1	Diagrama de Gantt	63

ÍNDICE DE TABLAS

4.1	Tabla con los valores nominales para la caracterización del convertidor ADC.	47
4.2	Parámetros nominales.	49
4.3	Tabla con los valores nominales para la caracterización del convertidor ADC.	54
4.4	Tabla con los valores nominales para la caracterización del convertidor ADC.	55
7.1	Tabla coste asociado al personal.	62
7.2	Tabla coste asociado al uso de licencias.	62
7.3	Tabla coste asociado al personal.	63

1. INTRODUCCIÓN

1.1. Motivación del trabajo

Los convertidores de datos analógico-digitales (ADC) son necesarios allá donde exista una señal del mundo físico que necesite ser tratada en el dominio digital, por lo que están presentes en cualquier tipo de sensor y de radio, sistema de instrumentación. La implementación de estos convertidores se realiza en circuitos microelectrónicos, por lo que el máximo desempeño de éstos está fuertemente condicionado por la tecnología de fabricación.

Como bien supo ver Gordon Moore en 1965 [8] [9], existe una tendencia a doblar el número de transistores que se pueden implementar en un mismo chip cada uno o dos años. Esto sería conocido más tarde como ley de Moore.

Esta integrabilidad provoca que en un mismo chip se pueda implementar un circuito que realice más funciones o que realice la misma función pero de forma más rápida y eficiente. Un ejemplo es el almacenamiento de datos o la velocidad de los microprocesadores. Esta mejora continua de la tecnología de fabricación de transistores, resulta en circuitos microelectrónicos cada vez más pequeños que requieren a su vez menos tensión de alimentación, estando actualmente en procesos de fabricación de 5 nanómetros.

La miniaturización de los circuitos ha permitido dotarles de más funcionalidades y la creación de nuevos productos. No obstante, ha generado algunas dificultades en el proceso de diseño. Entre otras dificultades, la fabricación es enormemente costosa, por lo que resulta difícil tener más de un prototipo de verificación antes de pasar a producción, lo que era habitual hace una década. Además las opciones de test que se pueden incluir en el prototipo son limitadas, en especial para los circuitos de señal mixta, y no siempre permiten detectar la causa por la que no cumple las especificaciones marcadas. Como consecuencia, el desarrollo de modelos precisos de simulación permite optimizar las fases de diseño, y también apoyar las fases de verificación experimental del prototipo. Existen diferentes niveles de abstracción donde se pueden realizar modelos, y este trabajo se centra en modelos comportamentales de circuitos de señal mixta para comunicaciones.

CAPÍTULO 1. 1.2. OBJETIVOS

Con estos precedentes surge este TFG bajo una Beca de investigación con Intel Austria para desarrollar un convertidor ADC para comunicaciones en 16nm. Concretamente, se plantea realizar una biblioteca de bloques en SIMULINK, algunos específicos para los ADCs basados en osciladores controlados por tensión o VCO, y otros comunes a ADCs sobremuestreados. Esta biblioteca se usa para la caracterización comportamental del ADC desarrollado en el proyecto de investigación.

1.2. Objetivos

El presente trabajo tiene como objetivo crear y validar una biblioteca en SIMULINK para modelar bloques que se pueden encontrar habitualmente en convertidores de datos sobremuestreados, concretamente:

- Celda integradora, basada en circuito RC con un amplificador operacional: Desarrollo de elementos que modelen el funcionamiento de este integrador, incluyendo distintas variaciones respecto al modelo básico.
- Celda del VCO u oscilador controlado por tensión, basado en Oscilador en anillo: Se busca el desarrollo de un modelo parametrizable que permite cambiar el número de etapas, y por tanto, la longitud del anillo.

Para demostrar el uso práctico de la biblioteca generada, se propone un objetivo complementario:

- Caracterización de un ADC basado en VCO: Se pretende simular temporalmente un modelo completo en SIMULINK, para su uso en el proyecto de investigación, usando los elementos de la biblioteca.

1.3. Marco Regulator

Este trabajo está desarrollado bajo una beca de colaboración con el grupo de Diseño Microelectrónico y Aplicaciones (DMA) del Departamento de Tecnología Electrónica de la Universidad Carlos III de Madrid (UC3M), bajo un contrato Art. 83 de la Ley Orgánica 6/2001, de 21 de Diciembre, de Universidades (LOU) con Intel Austria GmbH. Este contrato pertenece a un acuerdo marco de colaboración para I+D con duración de 2017 a

HERRAMIENTA DE MODELADO Y SIMULACIÓN DE CONVERTIDORES SOBREMUESTREADOS EN MATLAB/SIMULINK

2020 y que regula los términos de cooperación, la confidencialidad y los derechos sobre los resultados. Para el acceso a las tecnologías de Intel, se han firmado unos documentos de confidencialidad NDA (Non Disclosure Agreement). Los derechos de explotación de los resultados se han cedido a Intel Austria GmbH, y la UC3M mantiene derechos académicos sobre los resultados, pudiendo hacer uso de ellos en Trabajos de Fin de Grado, Trabajos de Fin de Máster, Tesis Doctorales y en docencia.

1.4. Estructura del documento

El documento se divide en siete capítulos distintos.

Se empieza con una introducción indicando la motivación del trabajo, además de indicar los objetivos que se pretende alcanzar con este, así como establecer el marco regulador.

En el segundo capítulo se analiza el estado del arte para contextualizar este trabajo. Con este fin, se describe el proceso de diseño de un prototipo integrado o *testchip* de señal mixta y luego se describen las herramientas de modelado para ADCs sobremuestreados que aparecen en la literatura científica.

En el tercer capítulo se revisan los conceptos teóricos de un ADC sobremuestreado, los llamados $\Sigma\Delta$, así como los asociados a los ADCs basados en VCO. Así mismo, se describen los distintos modelos desarrollados, y su validación con ayuda de herramientas de simulación de circuitos.

En el cuarto capítulo se caracteriza un ADC basado en VCO utilizando los elementos de la biblioteca desarrollada, y se hacen varias pruebas con este sistema.

En el quinto capítulo se describen y comentan los resultados del trabajo lo que lleva a una serie de conclusiones. Así también, se plantean futuras mejoras que se pueden llevar a cabo.

El sexto y séptimo capítulos se dedican al entorno socioeconómico del proyecto, y al presupuesto y planificación del mismo respectivamente.

2. ESTADO DE LA TÉCNICA

Dentro del diseño de circuitos integrados, los de señal mixta son aquellos que presentan circuitos analógicos y digitales en un mismo dado de silicio, lo que generalmente conlleva que el precio del proceso de fabricación disminuya debido al ahorro de componentes. Se pueden encontrar, prácticamente, en cualquier tipo de dispositivo electrónico convencional, como una radio, o moderno, como por ejemplo un teléfono inteligente o un *wearable*. Además, éstos últimos participan de forma directa en la ola del denominado Internet de las Cosas (*Internet of Things*, IoT), que por su propia naturaleza requieren de forma inherente un ADC.

2.1. Proceso de diseño de un circuito integrada de señal mixta

Los circuitos de señal mixta tienen su propio flujo de diseño, más próximo al flujo de diseño de circuitos analógicos, que al de circuitos digitales. Debido al estado actual de la tecnología, es muy habitual encontrarse con sistemas completos integrados en un único *chip* o SoC, como el que se muestra en la figura 2.1, donde deben combinarse distintos flujos de diseño para cada uno de los macrobloques que aparecen, y donde el uso de herramientas EDA (*Electronic Design Automation*) no siempre es posible. Por otro lado, la elevada competencia comercial de las grandes empresas de microelectrónica imponen una presión muy elevada en el tiempo que transcurre desde que se concibe un nuevo producto hasta que llega al mercado. Para garantizar el buen funcionamiento de un SoC determinado es necesario abordar el diseño dividiéndolo jerárquicamente en macrobloques, bloques y subbloques, y fabricar circuitos de prueba o *testchip* de cada uno de estos bloques o macrobloques antes de su integración final, lo que choca con el requisito de acortar los tiempos de desarrollo [10]. Generalmente, este conflicto se resuelve reduciendo el número de *testchip* a desarrollar, trabajando en una mejora de las herramientas EDA y de modelado y simulación, y reduciendo las pruebas funcionales que se realizan sobre el SoC final.

Si se considera el flujo de diseño de uno de los bloques, como es el caso del ADC de comunicaciones objeto del proyecto en el que se enmarca este trabajo, el flujo de diseño

HERRAMIENTA DE MODELADO Y SIMULACIÓN DE CONVERTIDORES SOBREMUESTREADOS EN MATLAB/SIMULINK

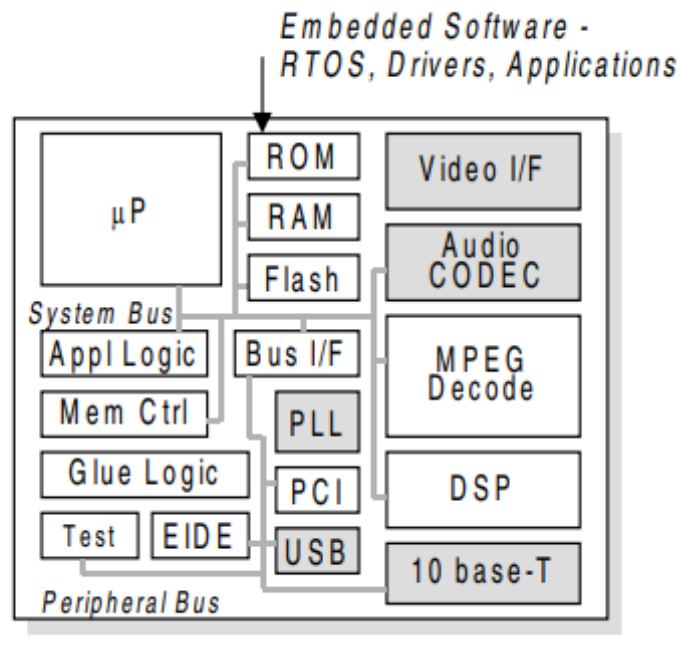


Fig. 2.1. Ejemplo de SoC [1].

puede resumirse en las fases que se muestran en la figura.

Primeramente, se encuentra la concepción teórica *testchip* que nace a partir de, entre otros, objetivos a alcanzar, límites de la tecnología, estado de la técnica, presupuesto, etc. Esto lleva hacia que surjan, posiblemente, distintas arquitecturas para alcanzar los objetivos marcados. A partir de las arquitecturas definidas, se especifican los subbloques, realizando un modelado comportamental de la parte analógica. Como resultado de este modelado comportamental, se definen unas especificaciones para los circuitos a desarrollar. Una vez diseñados y validados los circuitos, se procede al *layout* por subbloques, y a una simulación posterior con todas las capacidades parásitas detectadas, lo cual es solo posible para circuitos de tamaño y complejidad media. Una vez se ha superado esta etapa, hay que interconectar los diferentes subbloques, ya que van a ir fabricados en el mismo dado de silicio. Esto es el denominado *floorplan*. Aquí se deciden donde colocar los distintos bloques, establecer las conexiones entre ellos, las conexiones a alimentación y masa, así como las conexiones a los pines del encapsulado. Finalmente, se procede a generar las máscaras de fabricación, se manda a fabricar el prototipo, se encapsula y se testea en un laboratorio para validar su funcionamiento. Esta caracterización se hace con una PCB que sirve para aplicar una batería de pruebas al chip y así comprobar que cumplan los objetivos que se marcaron en un inicio. Una vez, que el *testchip* se ha validado, es

CAPÍTULO 2. 2.1. PROCESO DE DISEÑO DE UN CIRCUITO INTEGRADA DE SEÑAL MIXTA

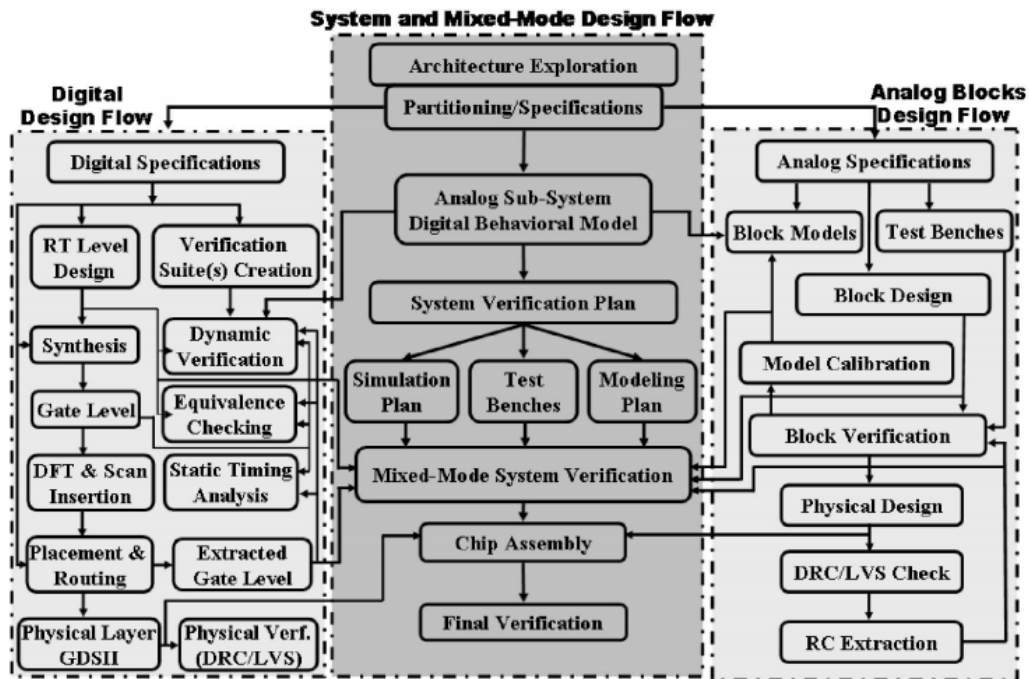


Fig. 2.2. Flujo de diseño de un circuito de señal mixta [2].

necesario hacer algunas modificaciones para pasarlo a producción.

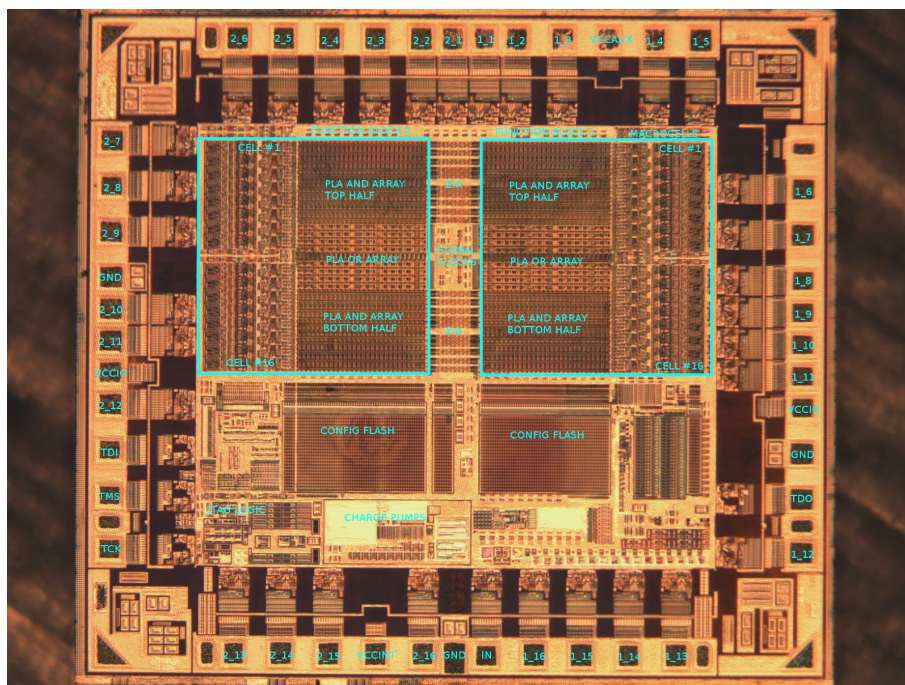


Fig. 2.3. Floor plan de un chip Mitsubishi M30620ECFP 16-BIT CMOS MICROCOMPUTER. [3]

Para el modelado comportamental de circuitos analógicos y de señal mixta se usa MATLAB/SIMULINK, programas en C/C++ o Descripción comportamental en AHDL o Verilog-A [11]. El modelado en MATLAB/SIMULINK o en C/C++ solo se suele ha-

cer a nivel de señal, mientras que si se quiere modelar tensiones y corrientes, aunque sea de forma comportamental, se suele usar AHDL, Verilog-A o circuitos descritos con esquemáticos.

Una forma de trabajo habitual en el grupo DMA es realizar el modelado comportamental y la definición de arquitectura en MATLAB/SIMULINK; posteriormente se realiza un modelo más detallado en Cadence (Herramienta profesional para el desarrollo de circuitos integrados), usando VerilogA o esquemáticos comportamentales, para después hacer el diseño detallado a nivel de transistor de cada uno de los bloques. Esta forma de trabajo permite un flujo iterativo y la optimización de cada bloque, tanto a nivel de sistema como a nivel de circuito, y ha sido empleada con éxito en numerosos proyectos de investigación.

2.2. Modelado de ADCs sobremuestreados

Este trabajo se centra en el modelado comportamental de bloques fundamentales de un ADC $\Sigma\Delta$ o sobremuestreado.

Tanto en la literatura como en trabajos de otras universidades, se pueden encontrar diversas propuestas al modelado de ADCs sobremuestreados, se destacan:

- **The Delta Sigma Toolbox:** Es una toolbox desarrollada por R. Schreier [12] [5]. Esta toolbox está enfocada a nivel de sistema y sirve como un posible primer paso para el diseño de un convertidor ADC. Sin embargo, no trata de forma directa lo que se plantea en los objetivos, ya que no implementa el caso de usar integradores con no idealidades ni tampoco un modelo para el VCO. Sin embargo, cabe destacar que es una toolbox de gran importancia y presenta varias aplicaciones que son de mucha utilidad a nivel de sistema.
- **SIMSIDES:** Esta es una toolbox desarrollada por el Instituto de Microelectrónica de Sevilla (IMSE-CNM) [13]. Esta herramienta presenta diversos modelos para diseñar un ADC de forma completa, tanto en tiempo continuo como en discreto, además de presentar una gran cantidad de herramientas para analizar los resultados. Sin embargo, presenta una serie de características que hacen que no se adecue a lo que se busca con este trabajo, como que esté diseñado para trabajar en su totalidad en ese entorno, por lo que el intentar utilizar elementos de otras bibliotecas pue-

CAPÍTULO 2. 2.2. MODELADO DE ADCS SOBREMUESTREADOS

de llegar a causar incompatibilidades; además, no presenta un modelo para VCOs. Esto provoca que no se ajuste a los objetivos buscados en este trabajo.

- Uni Ulm Sigma-Delta Synthesis Tool: Es una página web desarrollada por la Universidad de Ulm [14]. Es una herramienta con el fin de diseñar ADCs sobremuestreados basados en la búsqueda heurística. Concretamente, lo que se diseña son los coeficientes del modulador, tanto en tiempo continuo como en tiempo discreto, pudiendo llegar a obtener valores para el posible integrador; sin embargo, de querer aplicar estos valores en MATLAB y SIMULINK hay que tener previamente el modelo diseñado, y esto la herramienta web no lo proporciona. Esta herramienta no ofrece soporte para VCOs.

Después de haber buscado y estudiado distintas opciones, se puede afirmar que no existe una alternativa que satisfaga los objetivos que se plantean en el presente trabajo. Además, en particular, no existe una biblioteca o toolbox que utilice los VCOs en el ámbito de ADCs sobremuestreados.

3. MODELADO COMPORTAMENTAL DE CIRCUITOS ADCS BASADOS EN VCOS

3.1. Fundamentos convertidores ADC

El objetivo de un ADC es convertir una señal física, que puede tomar infinitos valores dentro de un intervalo, a una señal que pertenezca al dominio digital, con unos valores discretos. A continuación, se sientan unos fundamentos básicos de los convertidores ADC.

3.1.1. Convertidor ADC

El diagrama de bloques de un convertidor ADC se representa en la figura 3.1 donde un *Sample and Hold* muestrea la señal de entrada, V_{in} , con un período T_{clock} . Luego esta señal muestreada se convierte al dominio digital.

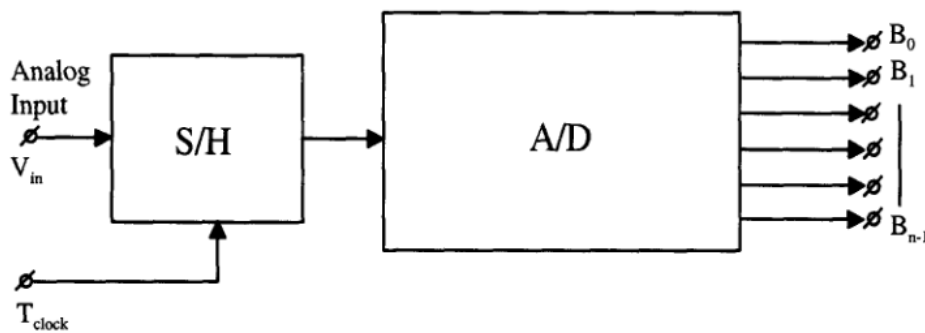


Fig. 3.1. Diagrama de bloques de un convertidor ADC [4].

La ecuación que relaciona salida y entrada es la siguiente:

$$V_{in} = D_{out} + V_x = V_{ref} \cdot \sum_{m=0}^{n-1} B_m 2^m + V_x, \quad (3.1)$$

donde V_{ref} indica una tensión de referencia, D_{out} representa la señal digitalizada de V_{in} , B_m representa al bit número m ¹ y V_x representa el error de cuantificación. Este error

¹ B_{n-1} representa el bit más significativo (MSB) y B_0 representa el bit menos significativo (LSB). El factor 2^m indica el peso binario que corresponde a cada bit.

CAPÍTULO 3. 3.1. FUNDAMENTOS CONVERTIDORES ADC

de cuantificación indica la diferencia entre el valor cuantificado digital obtenido multiplicado por la tensión V_{ref} y la señal analógica de entrada cuando se usan n niveles de cuantificación [4].

Suponiendo que los distintos niveles de cuantificación estén separados de igual forma, se tiene que la magnitud del mínimo escalón viene definido por:

$$V_{LSB} = \frac{V_{ref}}{2^n}. \quad (3.2)$$

Entonces, V_x tendría un valor tal que:

$$-\frac{1}{2}V_{LSB} < V_x < \frac{1}{2}V_{LSB} \quad (3.3)$$

En la figura 3.2 se ve un ejemplo de cómo V_x , en efecto, respeta la ecuación (3.3), se observa que si la señal V_{in} no estuviera dentro de $V_p \pm V_{LSB}$ pasaría a otro nivel de cuantificación distinto.

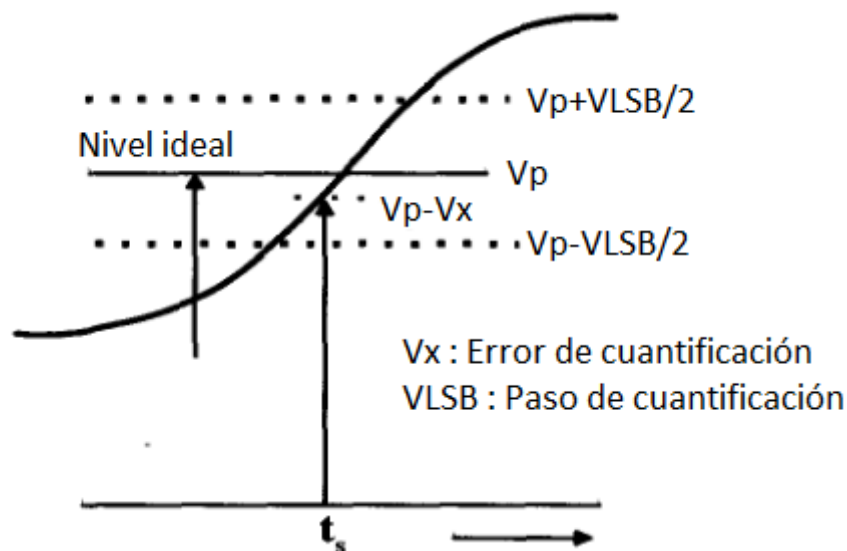


Fig. 3.2. Error de cuantificación.

En la figura figura 3.3 se muestra la función de densidad de probabilidad del error. Se observa que no hay relación entre la frecuencia de la señal y la frecuencia de muestreo. Se asume que para un elevado período de tiempo, se suceden los diferentes niveles de incertidumbre dentro de un mismo intervalo $V_p \pm V_{LSB}$. Entonces, se denomina $V_{Q(rms)}$ al

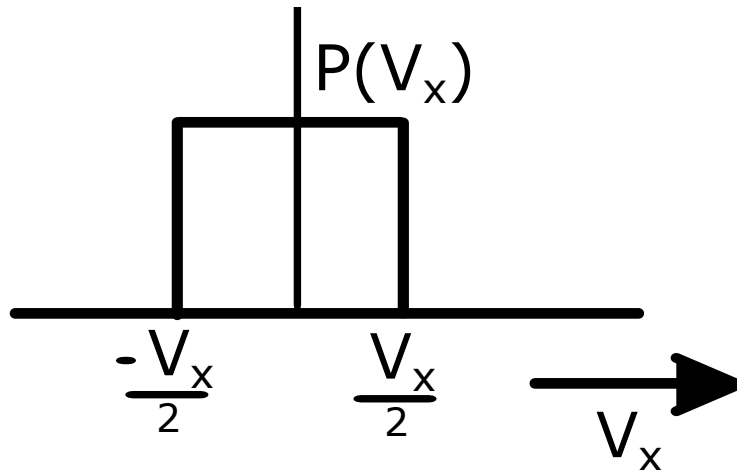


Fig. 3.3. Función de densidad de probabilidad del error de cuantificación

valor eficaz de este error y se define como:

$$V_{Q(rms)} = \frac{1}{\sqrt{12}} V_{LSB} \quad (3.4)$$

Así, para una señal, V_{in} sinusoidal de amplitud $V_{pp} = 2^n V_{LSB}$, se tiene que:

$$V_{in(rms)} = \frac{2^n V_{LSB}}{2\sqrt{12}} \quad (3.5)$$

Finalmente, relacionando el valor *rms* de la señal de entrada y el error se obtiene lo que se denomina *Signal to Quantization Noise Ratio* (SQNR, relación señal a ruido de cuantificación):

$$SQNR = 20 \log \frac{V_{in}}{V_Q} = 20 \log \frac{V_{LSB} 2^n / (2\sqrt{2})}{V_{LSB} / \sqrt{12}} = 6,02n + 1,76dB. \quad (3.6)$$

Tipos de convertidores ADC Típicamente se distinguen dos tipos de convertidores [15]:

- **Convertidores de Nyquist:** Estos tipo de convertidores se caracterizan por tener una frecuencia de muestreo, f_s , igual al doble del ancho de banda de la señal (Teorema de Nyquist) y que cada para un instante k , la señal de salida se corresponde con un rango de valores de la señal de entrada, este rango viene dado por el paso de cuantificación. Estos tipos de convertidores siguen la ecuación 3.6.

CAPÍTULO 3. 3.1. FUNDAMENTOS CONVERTIDORES ADC

- **Convertidores sobremuestreados:** Se caracterizan por operar con frecuencias de muestreo superiores, de 6 a 10 veces mayor como mínimo, a la frecuencia de Nyquist. Suelen ir acompañados de un bloque que aplica al espectro del error de cuantificación lo que se denomina conformado espectral de ruido.

Convertidores sobremuestreados El aumento de la frecuencia de muestreo, provoca que la densidad espectral del ruido pase a estar distribuida entre $-f_s/2$ y $f_s/2$ a estar, por ejemplo si se aumenta la frecuencia de muestreo 10 veces ($OSR = 20$), $f'_s = 10f_s$, entre $-5f_s$ y $5f_s$ siendo en ambos casos el área de potencia total el mismo. Así, filtrando el ruido que se encuentre fuera de la banda de interés, la nueva expresión para la SQNR es:

$$SQNR = 6,02n + 1,76dB + 10\log(OSR), \quad (3.7)$$

siendo OSR (Oversampling Ratio) el ratio de sobremuestreo que viene definido por

$$OSR = \frac{f_s}{2f_B}, \quad (3.8)$$

donde f_B representa el máximo ancho de banda de la señal.

Se observa que cada vez que se dobla el OSR, la SQNR aumenta en 3dB. En la figura 3.4 se observa que el ruido que se tiene en cuenta para un convertidores sobremuestreado es, en efecto, menor.

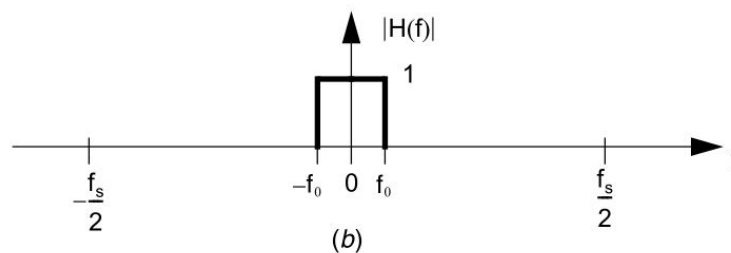


Fig. 3.4. Comparación entre la potencia de ruido para convertidores Nyquist y para convertidores sobremuestreados [15].

3.1.2. Convertidor ADC sobremuestreado con conformado espectral de ruido

Aquí se busca no sólo aumentar la frecuencia de sobremuestreo, si no que además se busca aplicar cierto conformado espectral al ruido. Un típico convertidor ADC con conformado espectral de ruido se presenta en la figura 3.5. Los convertidores $\Sigma\Delta$ en tiempo continuo no necesitan de un filtro anti-aliasing a diferencia de los de tiempo discreto [16]. Del mismo modo el bloque de *Sample and hold* no sería necesario.

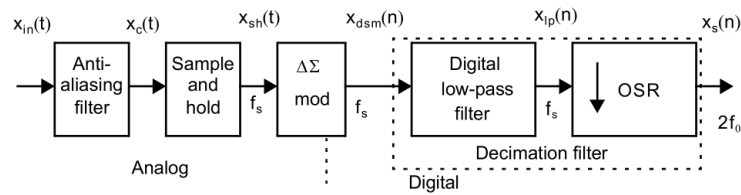


Fig. 3.5. Diagrama de bloques de un convertidor ADC $\Sigma\Delta$ en tiempo discreto [15].

Es el bloque ' $\Delta\Sigma$ mod' el que lleva a cabo el conformado espectral de ruido. A continuación se estudian el caso de primer orden.

3.1.3. Modulador $\Sigma\Delta$ de primer orden

Un modulador $\Delta\Sigma$ paso bajo de primer orden en tiempo discreto [5], se compone de un filtro que actúa como integrador, un cuantificador y un camino de realimentación negativa (Fig. 3.6) . Así, el integrador sería la parte Σ del sistema y la realimentación la parte Δ . Linealizando el circuito se obtiene la Fig. 3.7, se ha modelado el cuantificador como un error aditivo con una determinada secuencia que presenta un comportamiento espectral de tipo blanco dentro de un intervalo de cuantificación [15]. Se considera que el DAC es de tipo Non Return to Zero (NRZ).

Siendo $X(z)$ la señal de entrada, $E(z)$ el error del cuantificador y $Y(z)$ la salida del convertidor, se obtiene la siguiente relación:

$$Y(z) = z^{-1}X(z) + (1 - z^{-1})E(z). \quad (3.9)$$

Se denomina STF (Signal Transfer Function) a la función de transferencia de la salida respecto a la entrada, y NTF (Noise Transfer Function) a la función de transferencia de la salida respecto al error de cuantificación. De acuerdo a la ecuación ((3.9)) la STF no es

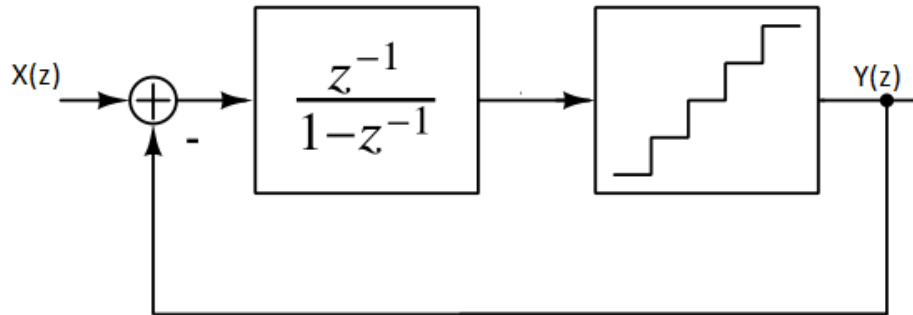


Fig. 3.6. Modulador $\Delta\Sigma$ de primer orden en tiempo discreto.

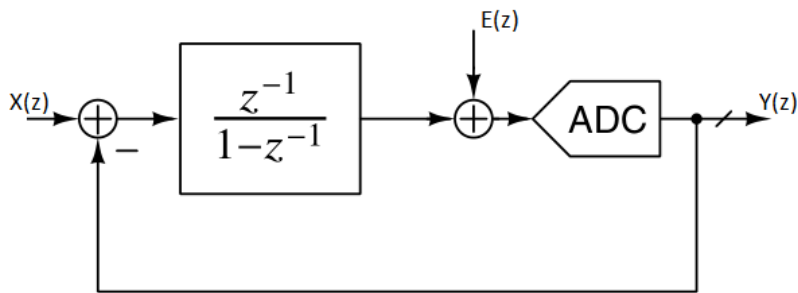


Fig. 3.7. Modelo lineal de un modulador $\Delta\Sigma$ de primer orden en tiempo discreto.

más que un retraso, esto es, que la señal de salida es la señal de entrada con un retraso de una muestra. Por otra parte, la NTF es un diferenciador en tiempo discreto, presentando así el error de cuantificación un comportamiento espectral de tipo filtro paso alto de primer orden.

Dado que se trata de un convertidor ADC sobremuestreado, la frecuencia de muestreo, f_s , va a ser teóricamente como mínimo 2 veces superior a la frecuencia máxima de la señal, f_B .

El valor de la SQNR para un modulador de primer orden viene entonces definido por:

$$SQNR = 6,02N + 1,76 - 5,17 + 30\log(OSR) \quad (3.10)$$

Se observa que el conformado espectral de primer orden consigue mejorar en 9dB cada

vez que se dobla el valor de OSR, comparado con los 3dB que se conseguían solamente aplicando sobremuestreo (Eq. 3.7).

Moduladores $\Delta\Sigma$ de orden superior Si añadimos en cascada otro integrador a aquel de la figura 3.6 se puede aumentar el orden del conformado espectral de ruido, y por lo tanto mejorar el rendimiento del sistema. Por lo tanto, es de esperar que añadiendo más integradores en cascada el orden sea mayor (Fig. 3.8). Sin embargo, cuanto mayor sea el orden del modulador, mayor es la complejidad de este y es más propenso a presentar inestabilidad.

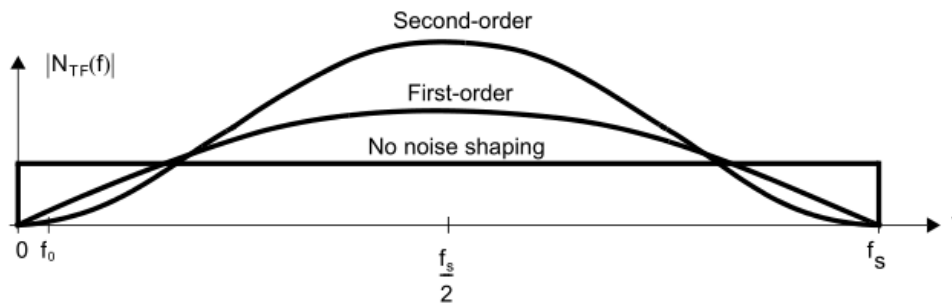


Fig. 3.8. Conformado de ruido para distinto orden de moduladores [15].

Implementación de Moduladores $\Delta\Sigma$ Los principios descritos anteriormente son válidos aunque se usen distintas implementaciones de los circuitos integradores y de los convertidores internos. Normalmente se distingue entre moduladores de tiempo discreto, que usan circuitos de capacidades conmutadas y que operan de forma síncrona a un reloj, y los moduladores de tiempo continuo, que realizan el muestreo de la señal internamente dentro del bucle de realimentación, y que contienen celdas integradoras que operan en tiempo continuo.

Este trabajo se centra en los moduladores de tiempo continuo. De forma típica se ha implementado el integrador del modulador $\Sigma\Delta$ usando bien un Opamp-RC, un OTA-RC o un Gm-C [5] (Fig. 3.9). La diferencia entre ellos es que los primeros funcionan en bucle cerrado y suelen ser más lineales, mientras que el último suele ser más rápido.

CAPÍTULO 3. 3.2. CONVERTIDOR ADC BASADO EN VCO

Un OTA no es más que un operacional sin etapa de buffer a la salida, y está pensado para manejar cargas capacitivas, que suele ser el caso de las celdas integradoras.

En el modelado que se desarrolla en este proyecto se consideran igual los operacionales tipo opamp u OTA, ya que no se tienen en cuenta los efectos de carga en las etapas primeras de diseño y definición de la arquitectura, como ya se ha comentado en el capítulo 2.

3.2. Convertidor ADC basado en VCO

En los últimos años, nuevos diseños han aparecido que utilizan un oscilador en anillo controlado por tensión (VCO) o por corriente (CCO), como sustitutos a la clásica implementación de los moduladores [17] [6] [18] [19], eliminando la necesidad de usar OpAmps. El VCO es una opción interesante ya que presenta una estructura digital, es decir, escala muy bien con la tecnología al contrario que la parte analógica y presenta un consumo menor. Consecuentemente, es conveniente investigar nuevas topologías de convertidores que basen su funcionamiento en VCO ya que son bastante prometedoras, como más adelante se pasará a describir una.

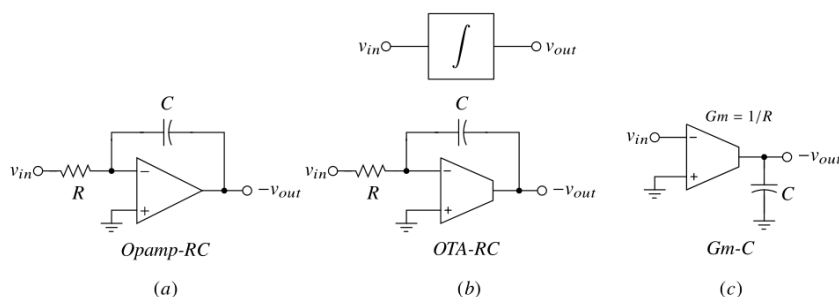
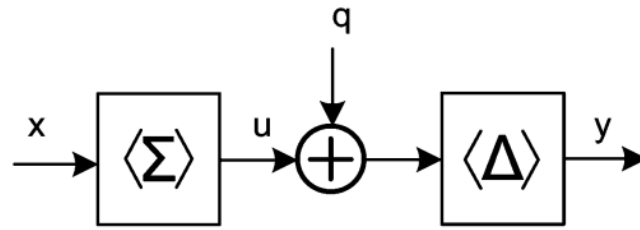


Fig. 3.9. Formas de implementar un integrador [5]

3.2.1. Convertidor $\Sigma\Delta$ ADC de primer orden basado en VCO

Para realizar un modulador $\Sigma\Delta$ no es necesario realimentar la salida [5] [20]. La figura 3.10 muestra el diagrama de bloques de un modulador sigma-delta en bucle abierto.

La figura 3.10 puede ser implementada mediante un VCO (Fig. 3.11). El VCO produce una señal cuadrada a su salida, cuya frecuencia depende de la amplitud de la entrada. Es



[20].

Fig. 3.10. Diagrama de bloques de un modulador $\Sigma\Delta$ en bucle abierto.

decir, se mapea del dominio de la tensión al dominio de la frecuencia. El VCO integra, y después mediante un conjunto de registros se muestrea su salida. Finalmente se tiene un bloque que realiza la primera diferencia, que se correspondería con el módulo diferenciador de la figura 3.10, obteniéndose así un modulador sigma-delta de primer orden. Cabe destacar que el VCO presenta un ruido de fase que también es sujeto al conformado espectral de primer orden, lo que resulta en un espectro con forma plana a la salida para bajas frecuencias.

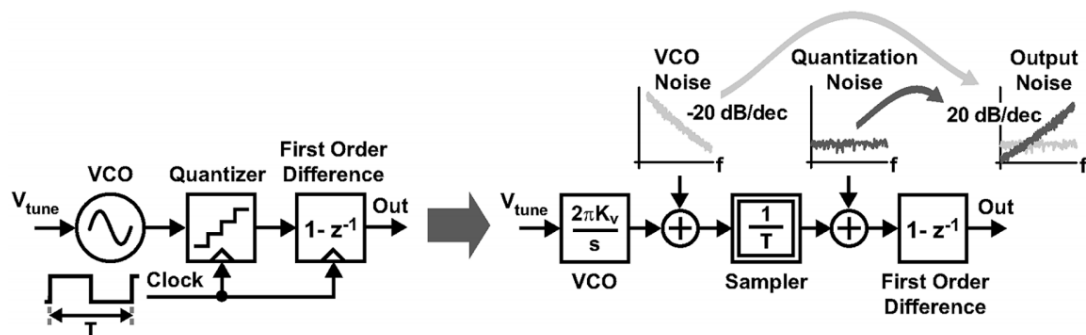


Fig. 3.11. Diagrama de bloques VCO-ADC de primer orden en bucle abierto y su correspondiente modelo linealizado en el dominio de la frecuencia [6].

Además, la curva voltaje-frecuencia del VCO en la realidad no es lineal (Fig. 3.12). Esto implica la aparición de armónicos que degrada la SNDR del convertidor. [6]

El SNR equivalente para un VCO-ADC con N inversores [21] [7], se aproxima a uno de primer orden con N niveles de cuantificación:

$$SNR = 6 \cdot \log_2(N) - 5,17 + 30\log(OSR). \quad (3.11)$$

CAPÍTULO 3. 3.2. CONVERTIDOR ADC BASADO EN VCO

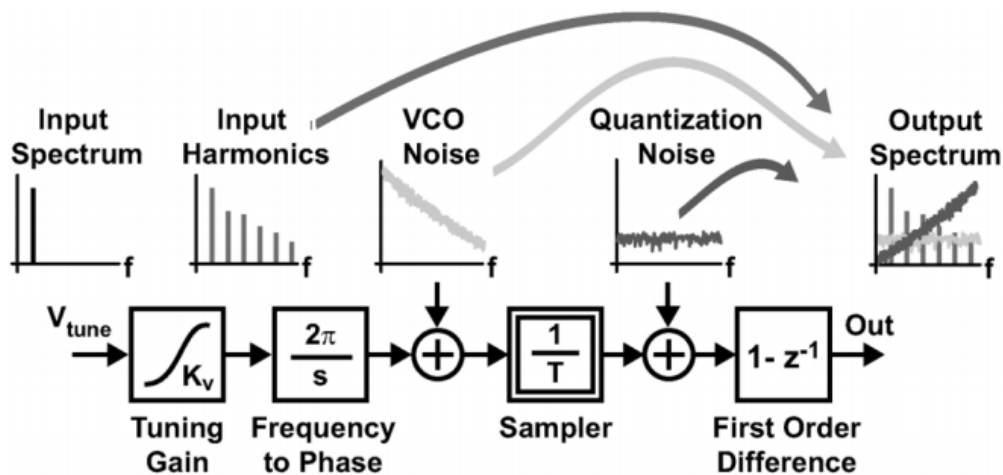


Fig. 3.12. Diagrama de bloques VCO-ADC que incluye la no linealidad de de la ganancia del VCO. [6].

Para contrarrestar la no linealidad del VCO se puede cerrar el bucle [22] [23] de tal forma que la realimentación reduce el rango de entrada del VCO, con la consiguiente limitación del uso en la curva voltaje-frecuencia lo que conlleva una mejora en la linealidad. Esto se consigue mediante una estructura como el de la figura 3.13.

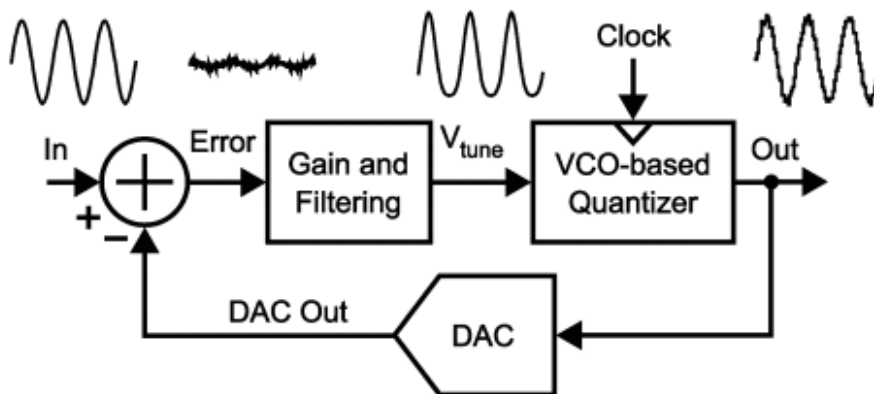


Fig. 3.13. ADC-VCO con bucle de realimentación para corregir la no linealidad del VCO [6]

Así entonces, se puede conseguir no solo mejorar la linealidad si no que además, aumentar el orden del conformado espectral de ruido del convertidor ADC. Se pueden identificar los siguientes elementos en la estructura de la figura 3.13:

- Filtro de tiempo continuo: Se trata de un integrador; puede implementarse de distintas formas, como un OpAmp RC, un OTA-RC o un Gm-C (Fig. 3.14). En el presente trabajo se centra el foco de atención en integradores en cascada, cuya cel-

da constructiva es un integrador RC con un OTA.

- VCO: Actúa como integrador y cuantificador; su frecuencia de oscilación se ve modulada por la amplitud que entra en este bloque. Está formado por un oscilador en anillo constituido por un número impar de inversores cuya corriente se controla mediante un transconductor. Puede presentar configuración diferencial.
- Registro con circuito de decodificación: Este circuito se encarga de muestrear la salida de las distintas fases del VCO para su posterior decodificación, esto es, aplicar por ejemplo, $1 - z^{-1}$. Tras aplicar la primera diferencia, los distintos resultados se pueden sumar consiguiéndose así la señal equivalente de salida.
- DAC: Se utiliza para convertir la señal de salida cuantificada, ya en el dominio digital, al dominio analógico para realimentarlo en el sistema. Existen distintas formas de pulso, Non Return to Zero, Return to Zero, exponencial decreciente, etc... Cada una con sus ventajas e inconvenientes que se aplican dependiendo del tipo de sistema. En este caso, el DAC presentará un elemento por cada inversor que haya en el VCO, si además se trata de una configuración diferencial a la entrada del bucle, el número será el doble.

Cabe destacar que el primer filtro puede tratarse también de un VCO [24] [25]; sin embargo, este tipo de estructuras se encuentra en una temprana fase de investigación por parte de la comunidad científica. Posteriormente, en este documento, se va a describir un convertidor ADC de segundo orden basado en VCO cuyo primer filtro es un integrador RC con un OTA.

A continuación se describe el modelado del integrador RC y oscilador en anillo dado que se implementarán como elementos en la biblioteca a desarrollar, que es el objetivo de este documento.

3.3. Integrador RC

Puede ser usado como filtro activo de un convertidor ADC en tiempo continuo. Se busca que realice una integración, ya que en el modulador de tiempo continuo, la función de transferencia del bloque es $1/s$.

Puede presentar dos configuraciones, integrador sin *feedforward* e integrador con *feed-*

CAPÍTULO 3. 3.3. INTEGRADOR RC

forward. La diferencia radica en que el modelo con *feedforward* presenta un cero adicional. Este modelo se utiliza para poder colapsar el segundo DAC a través del primero [5] y así ahorrar un DAC y todo lo que ello conlleva, consumo, área, etc.

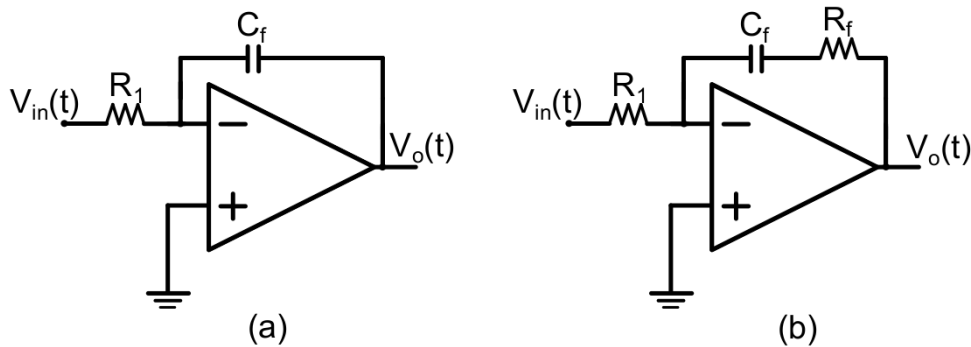


Fig. 3.14. (a) Integrador. (b) Integrador con camino feedforward.

En cualquier caso, de cara a su modelado y posterior implementación en el presente documento, se ha analizado de la misma forma.

Para los distintos niveles de modelado aquí considerados siempre se ha supuesto que, la resistencia de entrada del integrador, R_{in} , es infinita y que la resistencia de salida, R_{out} , es cero. Esto se ha considerado así ya que en la realidad, tanto R_{in} como R_{out} presentan valores muy altos y bajos, respectivamente, por lo que su modelado no aporta información útil.

Primeramente, consideramos un integrador RC real como el de la figura 3.14(a). Dado que es totalmente ideal y a que está realimentado negativamente, se puede aplicar masa virtual y por lo tanto, el terminal + y el terminal - presentan el mismo potencial, 0V. Así, obtenemos la siguiente relación:

$$\frac{V_o(s)}{V_i(s)} = -\frac{1}{sR_1C_f}. \quad (3.12)$$

Dado que se ha considerado que es ideal, la ganancia es infinita y el operacional no presenta polos por sí mismo. Se observa que se obtiene el equivalente al filtro del modulador continuo de primer orden salvo por el signo, esto es que nuestro OpAmp es un integrador

HERRAMIENTA DE MODELADO Y SIMULACIÓN DE CONVERTIDORES SOBREMUESTREADOS EN MATLAB/SIMULINK

que también invierte el signo.

De igual forma, se procede a calcular la función de transferencia que relaciona entrada y salida para el caso *feedforward* (Fig. 3.14/(b)). Se obtiene:

$$\frac{V_o(s)}{V_i(s)} = -\frac{sR_f C_f + 1}{sC_f R_1}. \quad (3.13)$$

En este caso, la función de transferencia representa el primer filtro continuo de un modulador de segundo orden, cuyo segundo DAC ha sido colapsado con el primero [26].

Si suponemos que el operacional no es ideal, una serie de modelos surgen; sin embargo, todos ellos se pueden clasificar y estudiar de la misma forma; para ello se pone como ejemplo el operacional de la figura 3.15(a) y se procede a su análisis. Se compone de dos elementos, Z_1 y Z_f , que pueden presentar comportamiento resistivo, inductivo o capacitivo, ahora mismo son cajas negras. Considerando que el operacional tiene una ganancia finita y pueda además presentar polos, esto conlleva ya tener un modelo más fiel, entonces se deja de suponer que el circuito equivalente presenta masa virtual.

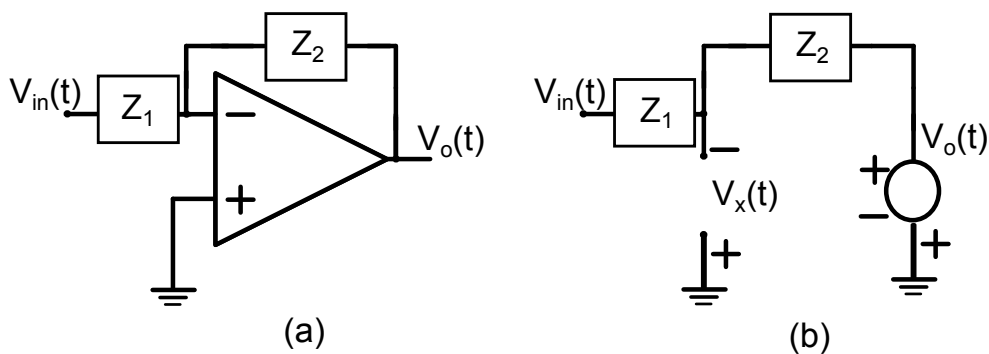


Fig. 3.15. (a)Circuito con un operacional. (b)Circuito equivalente

Su circuito equivalente se plantea en la figura 3.15(b); $V_x(s)$ representa la tensión de error en los terminales del OpAmp, y $A_v(s)$ es la relación del OpAmp entre $V_x(s)$ y $V_o(s)$. Resolviendo la sencilla del circuito [27] de forma sencilla, se procede a calcular la expresión que envuelve a todos los términos:

$$\frac{V_{in}(s) + V_x(s)}{Z_1(s)} = -\frac{V_x(s) - V_o(s)}{Z_2(s)}, \quad (3.14)$$

CAPÍTULO 3. 3.3. INTEGRADOR RC

sabiendo además que la relación entre $V_o(s)$ y $V_x(s)$ es

$$V_o(s) = A_v(s) \cdot V_x(s) A_v(s) = A_{OL} \cdot \prod_i^n \frac{1}{s + \omega_{pi}}, A_v = A_{OL}, \text{ solamente si el OpAmp no presenta polos.} \quad (3.15)$$

donde n es el número de polos que se quiera considerar y A_{OL} es la ganancia en bucle abierto del OpAmp. Despejando $V_x(s)$ de la Eq 3.14 se obtiene:

$$V_x(s) = -\frac{Z_2(s) \cdot V_{in}(s) + Z_1(s) \cdot V_o(s)}{Z_1(s) + Z_2(s)}. \quad (3.16)$$

A partir de las ecuaciones (3.15) y (3.16) se puede obtener el diagrama de bloques de la figura 3.16, que puede representar todas las combinaciones posibles que plantea la figura 3.15(a) y como se verá más adelante, también sirve en el caso de considerar condensadores parásitos a la entrada (el circuito equivalente sería distinto pero mantendría el mismo diagrama de bloques).

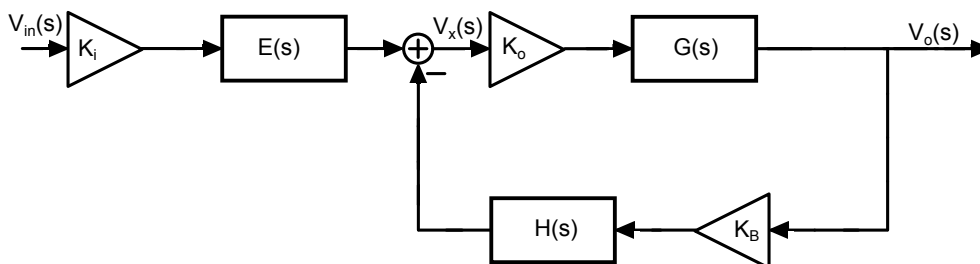


Fig. 3.16. Modelado de un integrador mediante diagrama de bloques.

A continuación se procede a definir cada bloque de la Fig. 3.16

- K_i : Ganancia que se aplica a la entrada $V_{in}(s)$
- $E(s)$: Función de transferencia que se aplica a la entrada $V_{in}(s)$ antes de entrar al bucle.
- K_o : Ganancia de $V_o(s)$ respecto a $V_x(s)$.
- $G(s)$: Función de transferencia que relaciona $V_o(s)$ con $V_x(s)$, es decir, indica los polos del operacional.
- K_b : Ganancia de la realimentación; aplicada a $V_o(s)$
- $H(s)$: Función de transferencia de la realimentación; aplicada a $V_o(s)$

Una vez definido el diagrama de bloques genérico se procede a enumerar y describir los modelos que se incluyen en la biblioteca:

3.3.1. Ganancia finita, primer y segundo polo

Partiendo de la figura 3.14 se considera que el circuito equivalente es el de la figura 3.17.

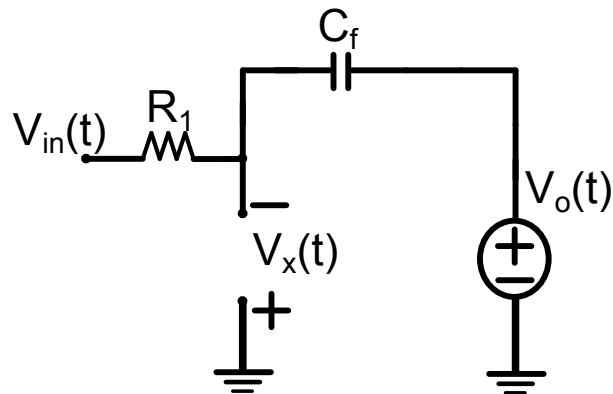


Fig. 3.17. Circuito equivalente a un integrador con ganancia finita y/o presencia de polos en el operacional.

Se busca obtener una ecuación que permita describir las distintas relaciones de la figura 3.16, pero no necesariamente tiene que ser igual a la ecuación (3.16) ya que esa es una expresión genérica que de no simplificar presentaría una apariencia muy aparatosa y más complicada de lo que realmente es; haciendo el análisis del circuito y simplificando la ecuación resultante, se obtiene:

$$V_x(s) = -\frac{V_{in}(s) + sR_1C_f \cdot V_o(s)}{1 + sR_fC_f} \quad (3.17)$$

con

$$V_o(s) = A_v(s) \cdot V_x(s) A_v(s) = A_{OL} \cdot \prod_i^n \frac{1}{s + \omega_{pi}}, n = 1, 2; A_v = A_{OL}, \text{ solamente si el OpAmp no presenta polos.} \quad (3.18)$$

Así, queda definido el modelo de un OpAmp RC con ganancia finita, con la opción de presentar uno o dos polos. A partir de la ecuación (3.18) se pueden obtener los distintos valores de las funciones de transferencia que aparecen en la figura 3.16.

3.3.2. Condensador parásito

Se considera también el caso de un condensador parásito entre los terminales de entrada del operacional. A partir del circuito del operacional de la figura 3.18, se halla el circuito equivalente incluyendo un condensador parásitos entre los terminales de entrada. Antes se afirmó que la figura 3.16, obtenido a partir de la ecuación (3.16) que a su vez viene del circuito de la figura 3.15, describía todos los tipos de modelos considerados en este documento. Aunque el circuito equivalente del presente caso, teniendo en cuenta el condensador parásito, no sea parecido al de la figura 3.15, las ecuaciones que lo describen sí que lo son como se demuestra a continuación. Analizando el circuito es obtiene que

$$\frac{V_{in}(s) + V_x(s)}{R_1} = -sC_p \cdot V_x(s) - sC_f \cdot (V_x(s) + V_o(s)), \quad (3.19)$$

y despejando $V_x(s)$ se llega a

$$V_x(s) = -\frac{V_{in}(s) + sR_1C_f \cdot V_o(s)}{1 + sR_f(C_f + Cp)}. \quad (3.20)$$

Se puede observar que la posición del polo de la ecuación (3.20) cuando el valor del condensador parásito a la entrada, C_p , sea de orden de magnitud similar al condensador C_f . Aquí, al igual que en la anterior subsección, $V_x(s)$ viene determinado por la ecuación 3.18. Así que el operacional con condensador parásito a la entrada ha quedado modelado.

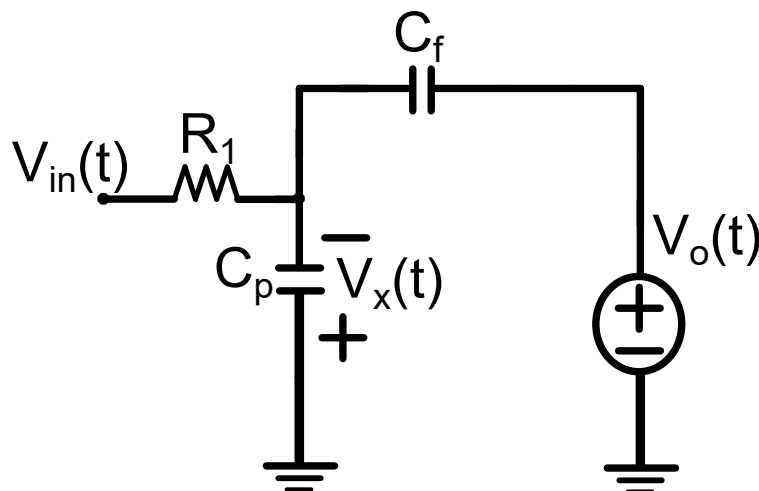


Fig. 3.18. Circuito equivalente a un integrador con efectos parásitos capacitivos.

3.3.3. Integrador con *feedforward*

Este circuito 3.14(b) es una modificación del integrador básico. Se añade una resistencia en serie, R_f con el condensador presente en el camino de realimentación, C_f . Si se integra C_f y R_f en una impedancia $Z_2(s)$ se tiene el caso de la figura 3.15, por lo tanto su estudio no plantea mayor novedad. Analizando el circuito equivalente sin incluir el condensador parásito en la entrada (Fig. 3.19), se obtiene que

$$\frac{V_{in}(s) + V_x(s)}{R_1} = -\frac{sC_f \cdot (V_{in}(s) + V_x(s))}{sR_fC_f}, \quad (3.21)$$

y despejando V_x se llega a

$$V_x(s) = -\frac{(1 + sR_fC_f) \cdot V_{in}(s) + sR_1C_f \cdot V_o(s)}{1 + sC_f(R_1 + R_f)}. \quad (3.22)$$

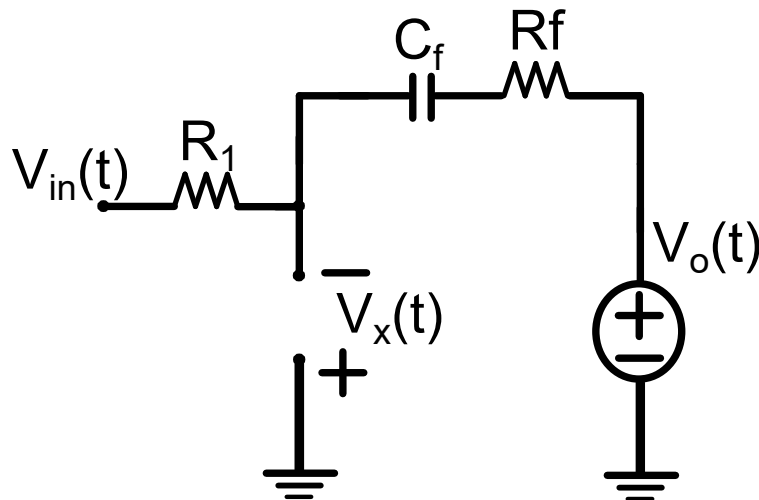


Fig. 3.19. Circuito equivalente a un integrador *feedforward*.

Se demuestra así que las ecuaciones llegan otra vez a la misma estructura que puede permitir extraer datos para luego crear un bloque del tipo de la figura 3.16.

Cabe destacar que sólo se va a implementar este modelo cuando el operacional tenga dos polos y se modela si presenta un condensador parásito a la entrada. El modelo con dos polos es directo y se obtiene con aplicar la ecuación 3.18.

Para modelar también el caso del condensador parásito, se parte del circuito equivalente de la figura 3.20. Analizando las corrientes y considerando que $I(s) = I_{CP}(s) + I_C(s)$, se

CAPÍTULO 3. 3.4. INTEGRADOR RC CON DOS, TRES Y CUATRO ENTRADAS

obtiene la siguiente ecuación que describe una relación general entre los distintos puntos del circuito:

$$\frac{V_{in}(s) + V_x(s)}{R_1} = -sC_p \cdot V_x(s) - \frac{sC_p \cdot (V_x(s) + V_o(s))}{sR_fC_f + 1} \quad (3.23)$$

y despejando V_x se llega a

$$V_x(s) = -\frac{(1 + sR_fC_f) \cdot V_{in}(s) + sR_1C_f \cdot V_o(s)}{1 + sC_fR_1 + sR_fC_f + sR_1C_p}. \quad (3.24)$$

De este modo, se acabaría con el modelado de integradores que presentan una sola entrada. Describiendo todos los modelos acorde al diagrama de bloques (Fig. 3.16) se puede llegar a obtener una única estructura que describa a los diferentes modelos.

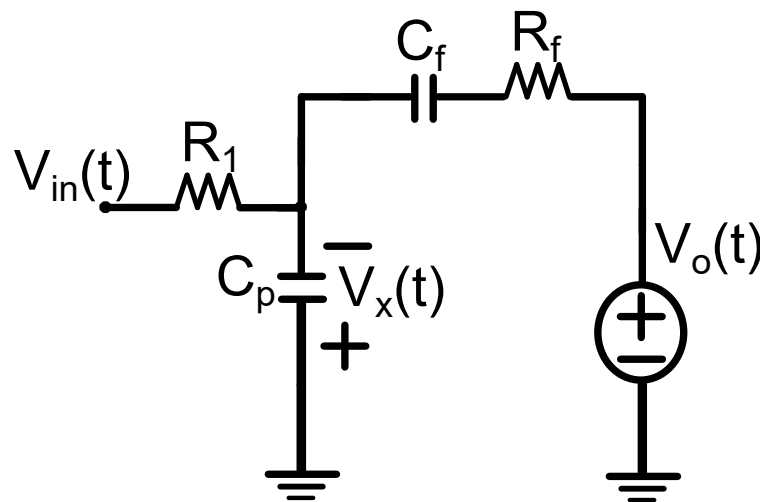


Fig. 3.20. Circuito equivalente a un integrador *feedforward* con efectos parásitos capacitivos.

3.4. Integrador RC con dos, tres y cuatro entradas

Hasta el momento se ha considerado solamente integradores con una sola entrada, se han descrito sus distintos modelos y obtenido sus respectivas ecuaciones. Sin embargo, es también útil modelar un integrador con más de una entrada. Como mínimo se considera que una entrada, presenta un elemento resistivo, para que se pueda producir el efecto integrador y por lo tanto seguir siendo centro de atención de este trabajo.

A pesar de presentar dos entradas, los diferentes modelos también pueden expresarse en diagrama de bloques, tal y como refleja la figura 3.21. Además, se observa que este dia-

grama de bloques actualizado consiste simplemente en añadir una ganancia y una función de transferencia entre la nueva entrada y el sumador.

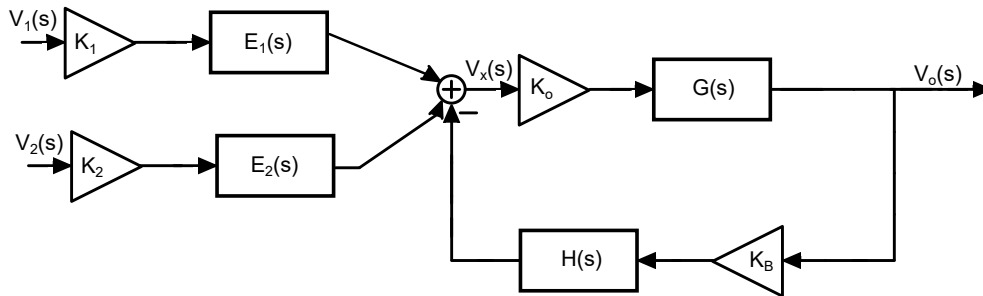


Fig. 3.21. Diagrama de bloques equivalente para un integrador de dos entradas.

Como se va a demostrar más adelante, una vez obtenidas las ecuaciones de un integrador con dos entradas con sus diferentes variantes, obtener las ecuaciones de modelos de integradores con más entradas superiores es sencillo. Así entonces, se procede al desarrollo de los distintos modelos del integrador de segundo orden como los mostrados en la figura 3.22, donde Z_2 puede representar un condensador o una resistencia.

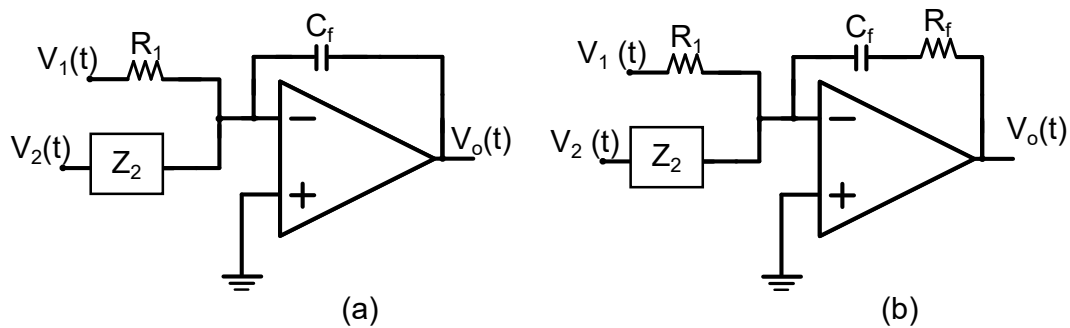


Fig. 3.22. (a) Integrador RC con dos entradas. (b) Integrador *feedforward* RC

3.4.1. Ganancia finita, primer y segundo polo

Se pueden distinguir dos casos principales, que Z_2 sea una resistencia o que sea un condensador.

Z_2 como resistencia: El circuito equivalente es de la figura 3.23; para realizar su análisis dado que se compone de dos entradas, se procede a aplicar el teorema de la superposición [27], por lo que cortocircuitando la primera entrada se obtendría $V_{x1}(s)$ y cortocircuitando

CAPÍTULO 3. 3.4. INTEGRADOR RC CON DOS, TRES Y CUATRO ENTRADAS

la segunda entrada se obtendría $V_{x2}(s)$. Realizando el análisis, se obtiene que $V_{x1}(s)$ es igual a:

$$\frac{V_1(s) + V_{x1}(s)}{R_1} = \frac{V_{x1}(s)}{R_2} - \frac{(V_{x1}(s) + V_{o1}(s))}{sC_f}. \quad (3.25)$$

$$V_{x1}(s) = -\frac{R_2 \cdot V_1(s) + sR_1R_2C_f \cdot V_{o1}(s)}{sR_1R_2C_f + R_1 + R_2}. \quad (3.26)$$

De igual forma para $V_{x2}(s)$ se obtiene que:

$$V_{x2}(s) = -\frac{R_1 \cdot V_2(s) + sR_1R_2C_f \cdot V_{o2}(s)}{sR_1R_2C_f + R_1 + R_2}. \quad (3.27)$$

Cabe destacar que al aplicar superposición y dado que $V_o(s)$ es dependiente de $V_x(s)$ también presentará dos versiones, $V_{o1}(s)$ y $V_{o2}(s)$, que sumadas son iguales a $V_o(s)$. Entonces $V_x(s)$ no es más que la suma de $V_{x1}(s)$ y $V_{x2}(s)$:

$$V_x(s) = V_{x1}(s) + V_{x2}(s) = -\frac{R_2 \cdot V_1(s) + R_2 \cdot V_2(s) + sR_1R_2C_f \cdot V_o(s)}{sR_1R_2C_f + R_1 + R_2} \quad (3.28)$$

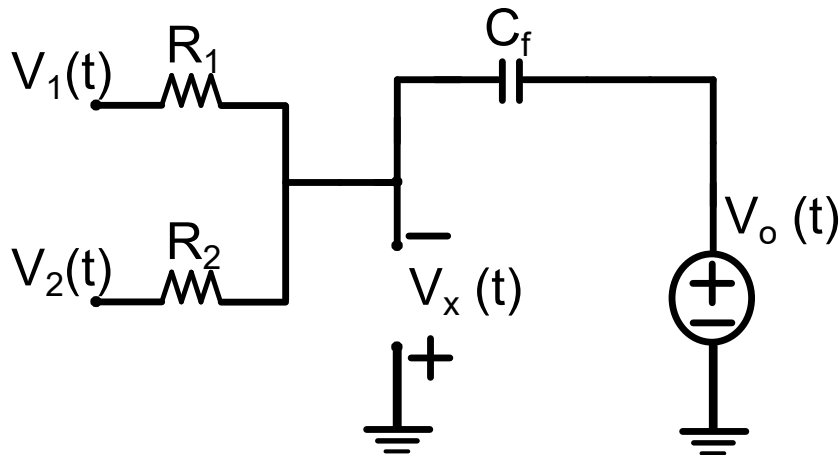


Fig. 3.23. Circuito equivalente al de la figura 3.22 sin considerar efectos capacitivos, considerando que Z_2 es una resistencia.

Z_2 como condensador: El circuito equivalente sería el de la figura 3.24. Realizando el análisis aplicando superposición se obtiene que:

$$V_{x1}(s) = -\frac{V_1(s) + sR_1C_f \cdot V_{o1}(s)}{sR_1(C_f + C_2) + 1}. \quad (3.29)$$

$$V_{x2}(s) = -\frac{R_1 \cdot V_2(s) + sR_1C_f \cdot V_{o1}(s)}{sR_1(C_f + C_2) + 1}. \quad (3.30)$$

Se puede observar que $V_{o1}(s)$ se puede analizar de la misma forma que el caso del condensador parásito a la entrada del operacional. $V_x(s)$ se obtiene como

$$V_x(s) = V_{x1}(s) + V_{x2}(s) = -\frac{V_1(s) + sR_1C_2 \cdot V_2(s) + sR_1C_f \cdot V_o(s)}{sR_1(C_f + C_2)}. \quad (3.31)$$

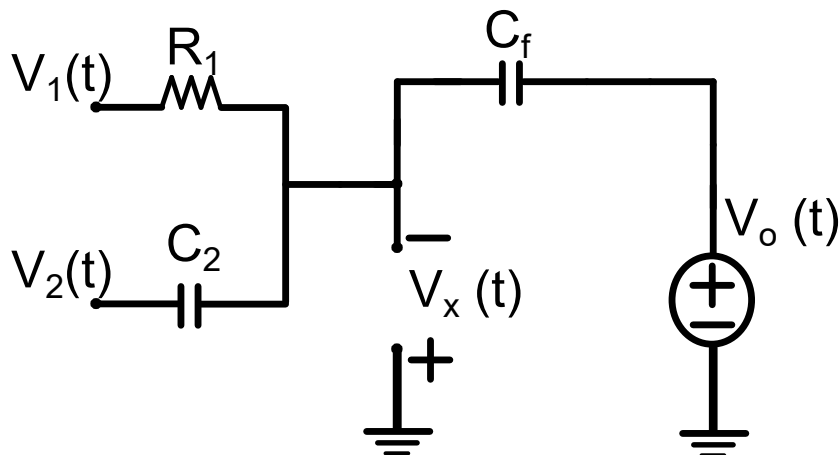


Fig. 3.24. Circuito equivalente al de la figura 3.22 sin considerar efectos capacitivos, considerando que Z_2 es un condensador.

Para representar los modelos de uno y dos polos, basta con utilizar la ecuación 3.18, así por lo tanto quedan descritos los modelos con ganancia finita, un polo y dos polos.

3.4.2. Condensador parásito a la entrada

Al igual que en el caso anterior, se puede considerar que Z_2 sea un condensador o una resistencia.

CAPÍTULO 3. 3.4. INTEGRADOR RC CON DOS, TRES Y CUATRO ENTRADAS

Z₂ como resistencia: El circuito equivalente se representa en la figura 3.25. Siguiendo la misma metodología se procede a aplicar la superposición, obteniéndose $V_{x1}(s)$ y $V_{x2}(s)$:

$$V_{x1}(s) = -\frac{R_2 \cdot V_1(s) + sR_1R_2C_f \cdot V_{o1}(s)}{sR_1R_2(C_f + C_p) + R_1 + R_2}. \quad (3.32)$$

$$V_{x2}(s) = -\frac{R_1 \cdot V_2(s) + sR_1R_2C_f \cdot V_{o2}(s)}{sR_1R_2(C_f + C_p) + R_1 + R_2}. \quad (3.33)$$

Se observa que las expresiones para $V_{x1}(s)$ y $V_{x2}(s)$ son iguales pero cambiando $V_1(s)$ y R_1 por $V_2(s)$ y R_2 donde corresponda, esto se debe a que sus circuitos equivalentes son iguales cuando se aplica el correspondiente cortocircuito. $V_x(s)$ es

$$V_x(s) = -\frac{R_2 \cdot V_1(s) + R_1 \cdot V_2(s) + sR_1R_2C_f \cdot V_o(s)}{sR_1R_2(C_f + C_p) + R_1 + R_2}. \quad (3.34)$$

Aplicando la ecuación (3.18) se obtienen los modelos con ninguno, uno o dos polos.

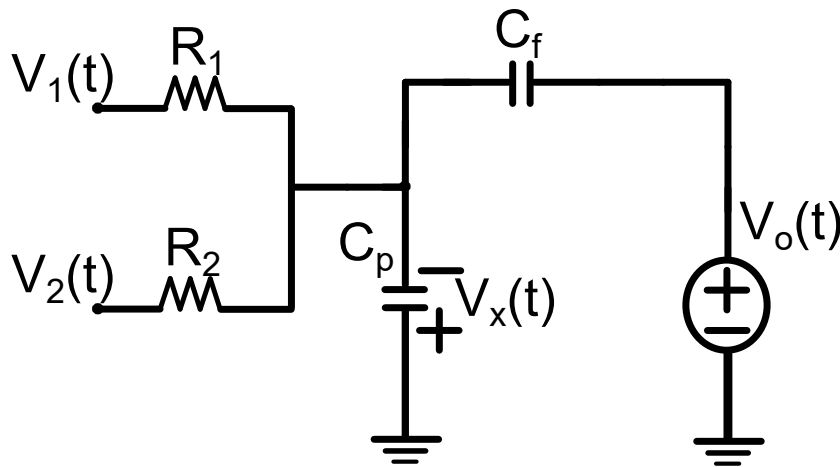


Fig. 3.25. Circuito equivalente al de la figura 3.22 considerando efectos capacitivos, considerando que Z_2 es un condensador.

Z₂ como condensador: El circuito equivalente se representa en la figura 3.26. Se puede observar que aplicando superposición, cuando se cortocircuita la primera entrada, se pone en paralelo C_2 y C_p por lo que su estudio es equivalente a aquel de la sección 3.3.2. Analizando el circuito se obtiene que:

$$V_{x1}(s) = -\frac{V_1(s) + sR_1C_f \cdot V_{o1}(s)}{sR_1(C_f + C_p + C_2) + 1}. \quad (3.35)$$

$$V_{x2}(s) = -\frac{sR_1C_2 \cdot V_2(s) + sR_1C_f \cdot V_{o1}(s)}{sR_1(C_f + C_p + C_2) + 1}. \quad (3.36)$$

$$V_x(s) = -\frac{V_1(s) + sR_1C_2 \cdot V_2(s) + sR_1C_f \cdot V_o(s)}{sR_1(C_f + C_p + C_2) + 1}. \quad (3.37)$$

Utilizando la ecuación (3.18) se obtienen los modelos con ninguno, uno o dos polos.

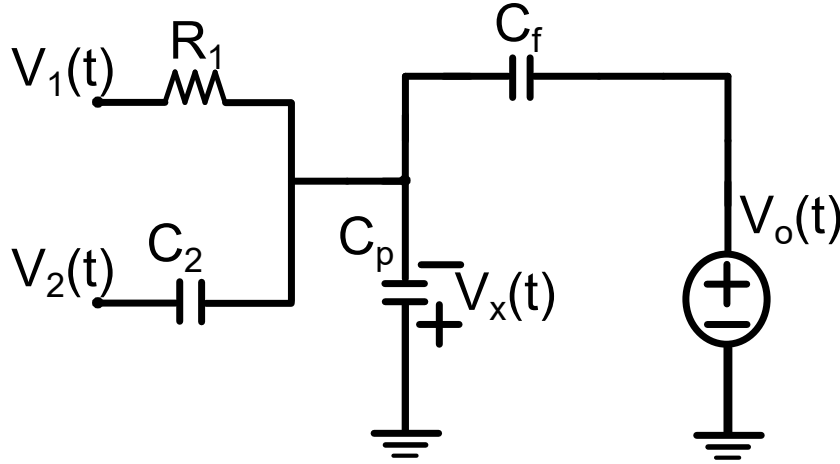


Fig. 3.26. Circuito equivalente al de la figura 3.22 considerando efectos capacitivos, considerando que Z_2 es un condensador.

3.4.3. Integrador *feedforward*

De igual forma que en los casos anteriores, dependiendo de la naturaleza de Z_2 se pueden distinguir varios casos.

Z_2 como resistencia: El circuito equivalente se representa en la figura 3.27. Analizando el circuito equivalente, se obtienen las siguientes ecuaciones:

$$V_{x1}(s) = -\frac{(sR_2R_fC_f + 1) \cdot V_1(s) + sR_1R_2C_f \cdot V_{o1}(s)}{sR_1R_2C_f + sR_1R_fC_f + sR_2R_fC_f + R_1 + R_2}. \quad (3.38)$$

$$V_{x2}(s) = -\frac{(sR_1R_fC_f + 1) \cdot V_2(s) + sR_1R_2C_f \cdot V_{o2}(s)}{sR_1R_2C_f + sR_1R_fC_f + sR_2R_fC_f + R_1 + R_2}. \quad (3.39)$$

$$V_x(s) = -\frac{(sR_2R_fC_f + 1) \cdot V_1(s) + (sR_1R_fC_f + 1) \cdot V_2(s) + sR_1C_f \cdot V_o(s)}{sR_1R_2C_f + sR_1R_fC_f + sR_2R_fC_f + R_1 + R_2}. \quad (3.40)$$

CAPÍTULO 3. 3.4. INTEGRADOR RC CON DOS, TRES Y CUATRO ENTRADAS

Se puede ir observando que las ecuaciones van adquiriendo una longitud considerable.

Utilizando la ecuación (3.18) se obtienen los modelos con ninguno, uno o dos polos.

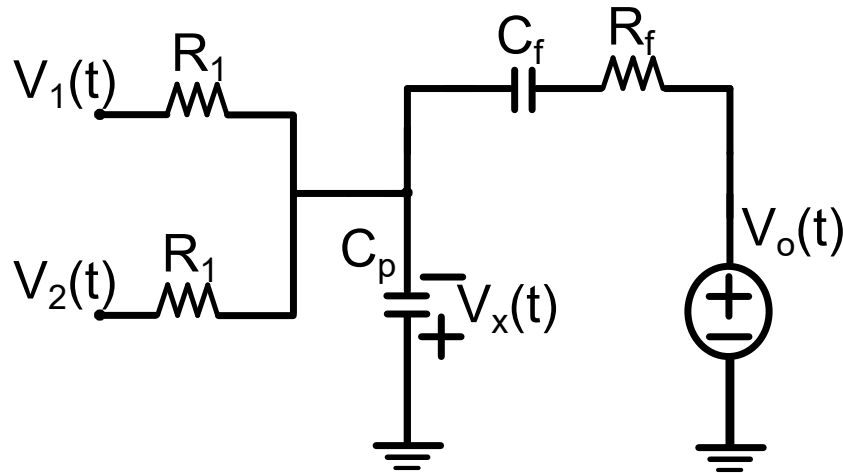


Fig. 3.27. Circuito equivalente al de la figura 3.22 con una resistencia en el camino de realimentación, sin considerar efectos capacitivos parásitos, y siendo Z_2 una resistencia.

Z_2 como condensador: Se puede observar que las ecuaciones se pueden obtener a partir de los modelos anteriores y por lo tanto, se decide para aquí con la descripción repetitiva de estas.

Al igual, el modelo con condensador parásítico a la entrada, podrá ser descrito con ecuaciones vistas previamente, ya que los circuitos equivalentes son parecidos.

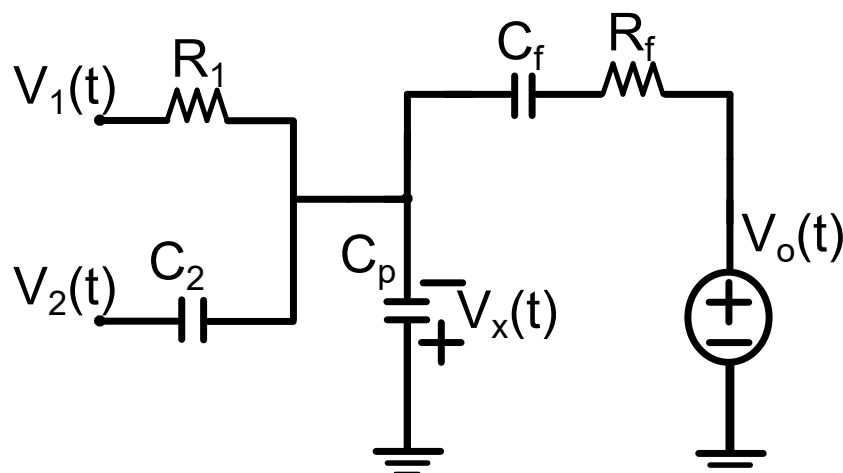


Fig. 3.28. Circuito equivalente al de la figura 3.22 con una resistencia en el camino de realimentación, sin considerar efectos capacitivos parásitos, y siendo Z_2 un condensador.

Integradores con más entradas: Como se presenta en la figura 3.29 es posible describir de igual forma que se hizo con el modelo de dos entradas, el comportamiento de aquellos integradores que presenten más. Así entonces, sus ecuaciones podrán ser obtenidas aplicando superposición e ir analizando los diferentes circuitos equivalentes. Sin embargo, se pueden hallar fácilmente a través de aquellas calculadas anteriormente, por lo que para evitar un exceso de ecuaciones, no se describen.

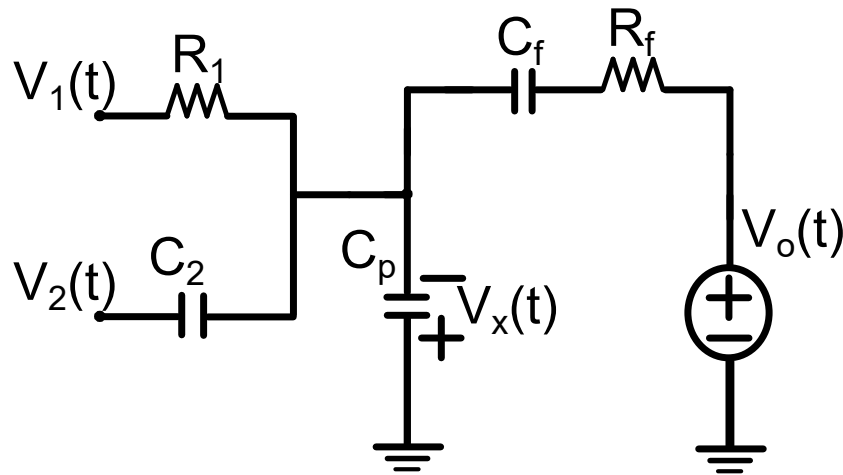


Fig. 3.29. Circuito equivalente al de la figura 3.22 con una resistencia en el camino de realimentación, sin considerar efectos capacitivos parásitos, y siendo Z_2 un condensador.

3.4.4. Oscilador en anillo

Un oscilador en anillo (Fig. 3.30) es, en esencia, un circuito electrónica digital formado por una serie de N inversores conectados en cadena; donde N es un número impar. Un transconductor es el encargado de proporcionar corriente a los inversores. Un esquemático de un VCO se muestra en la figura 3.30.

El VCO integra y posteriormente con un circuito de decodificación se consigue mues-

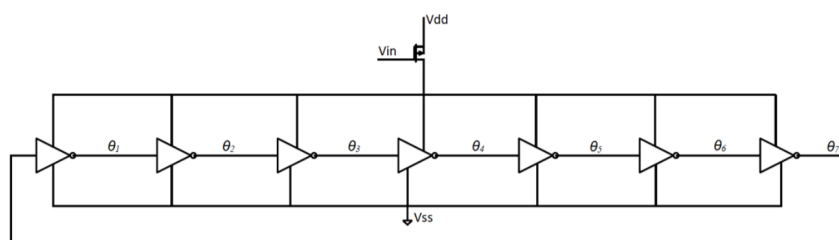


Fig. 3.30. Oscilador en anillo con 5 inversores.

CAPÍTULO 3. 3.4. INTEGRADOR RC CON DOS, TRES Y CUATRO ENTRADAS

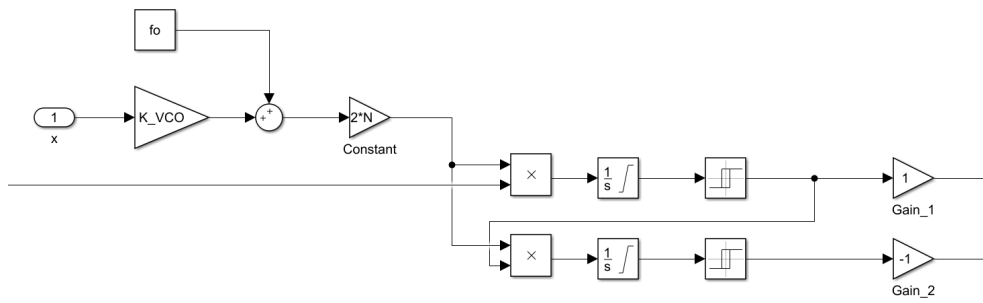


Fig. 3.31. Modelo oscilador en anillo. Se muestran dos inversores de una cadena.

tratar el sistema. La fase de salida del VCO [21] viene dada por

$$\theta[n] = \int_{(n-1)T_s}^{nT_s} 2\pi(K_{VCO}x(t) + f_o)dt, \quad (3.41)$$

donde T_s es el período de muestreo, K_{VCO} es la ganancia del VCO, f_o es la frecuencia central del VCO (cuando su entrada es cero), y $x(t)$ es la tensión de control del VCO normalizado en el intervalo $[-1, +1]$. Dado que la fase es la integral de la frecuencia, se puede expresar la frecuencia de oscilación del VCO [28] como

$$f_{osc} = f_o + K_{VCO} \cdot x(t). \quad (3.42)$$

También se puede expresar la frecuencia de oscilación en función del retraso que aporta cada inversor. En un oscilador con N inversores, donde cada uno tiene un retraso τ_d , el primer cambio de fase completo se produce en $N\Delta\tau_d$, ya que cada inversor aporta un cambio de fase π/N . Entonces, la frecuencia de oscilación se puede expresar también como

$$f_{osc} = \frac{1}{2N\tau_d}. \quad (3.43)$$

Partiendo de las ecuaciones previamente descritas, se puede plantear un modelo que representa el oscilador en anillo. El diagrama de bloques de la figura reffig:modelosimulinkvco representa el conjunto de las ecuaciones (3.41), (3.42) y (3.43). Se puede observar que el inversor recibe como entrada una señal $x(t)$ que posteriormente es modulada en el dominio de la frecuencia, que utilizando después un circuito de decodificación se obtiene la señal de salida.

Lo interesante de este modelo, es que el número de inversores es variable, ya que algunas veces interesará trabajar con unos pocos o bien con cientos de ellos. Entonces, la implementación del VCO en la biblioteca tiene que tener en cuenta este detalle.

3.5. Biblioteca desarrollada

Para la implementación de los modelos anteriormente descritos, con la salvedad de los integradores totalmente ideales, se ha buscado que la biblioteca sea lo más compacta posible ya que el número de casos implementables, al menos en el caso de los integradores, es bastante elevado como para tener una biblioteca de tantos elementos. Además, los diferentes modelos de integradores se han descrito siguiendo un mismo diagrama de bloques 3.16, por lo que es posible agruparlos según el número de entradas. Así pues, se reduce el número de elementos de la biblioteca a tan solamente 5: 3 integradores con una, dos, tres y cuatro entradas y el oscilador en anillo.

3.5.1. Integradores

Se agrupan según el número de entradas que tenga el integrador, lo que resulta en 4 modelos distintos.

Integrador con una entrada: Aquí se emplea un diagrama de bloques en SIMULINK como el mostrado en la siguiente figura 3.16.

Dado que se busca que este diagrama de bloques represente todos los casos de una sola entrada, es necesario crear un subsistema y añadir a este una máscara. Un subsistema consiste en agrupar, en este caso, la figura 3.16 en una sola 'caja' que tan solo muestre entrada y salida (Fig. 3.32). Una máscara en SIMULINK tiene como objetivo añadir dinámica al subsistema, esto es, desde cambiar ciertos parámetros hasta crear o eliminar nuevos bloques dentro de dicho subsistema. Esto es justamente lo que se pretende hacer, ya que habilita la sintetización de muchos modelos en un solo subsistema.

Primeramente, se procede crear el subsistema para inmediatamente después añadirle la máscara. La máscara permite también añadir una imagen al subsistema; esta imagen va embebida en la propia máscara (Fig. 3.33) y una vez se carga el archivo la imagen se

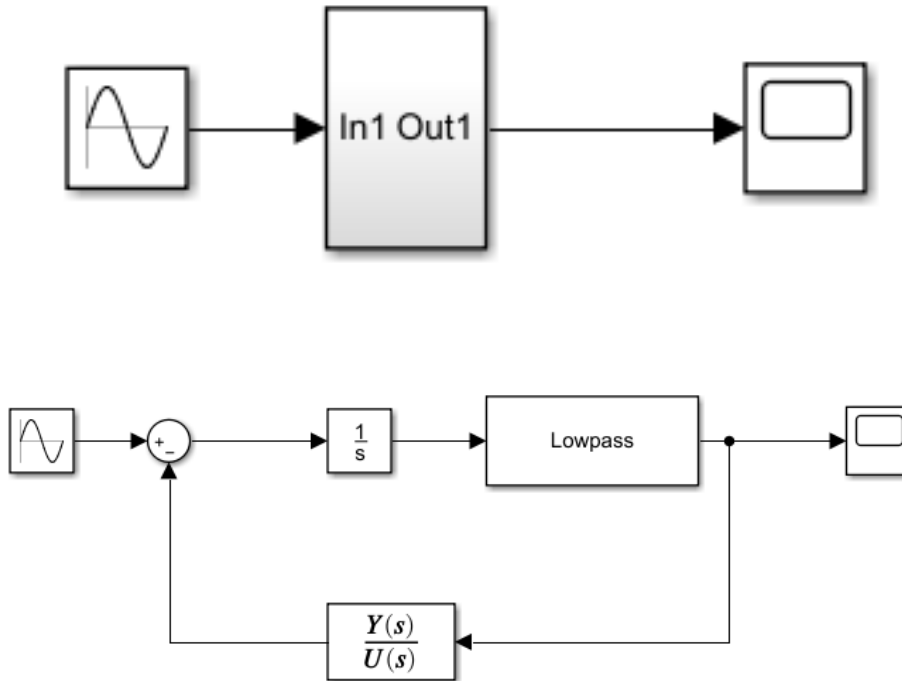


Fig. 3.32. Ejemplo de creación de subsistema.

queda guardada.

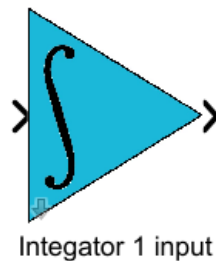


Fig. 3.33. Imagen del elemento Integrador 1 entrada en SIMULINK

Entonces, se procede a crear la interfaz de usuario; esto tiene lugar en la pestaña *Parameters Dialog* de la máscara. Aquí se añaden los parámetros que son objeto de modificación en el integrador, así como la programación de cierta dinámica dentro de la propia interfaz, como por ejemplo, *checkboxes* que va marcando el usuario para así determinar qué modelo desea, como la presencia de polos en el OpAmp, existencia o no de Slew Rate y saturación, etc. Una vez definido todos los parámetros y completada la programación de

HERRAMIENTA DE MODELADO Y SIMULACIÓN DE CONVERTIDORES SOBREMUESTREADOS EN MATLAB/SIMULINK

la interfaz, esta pestaña queda como en la figura 3.34

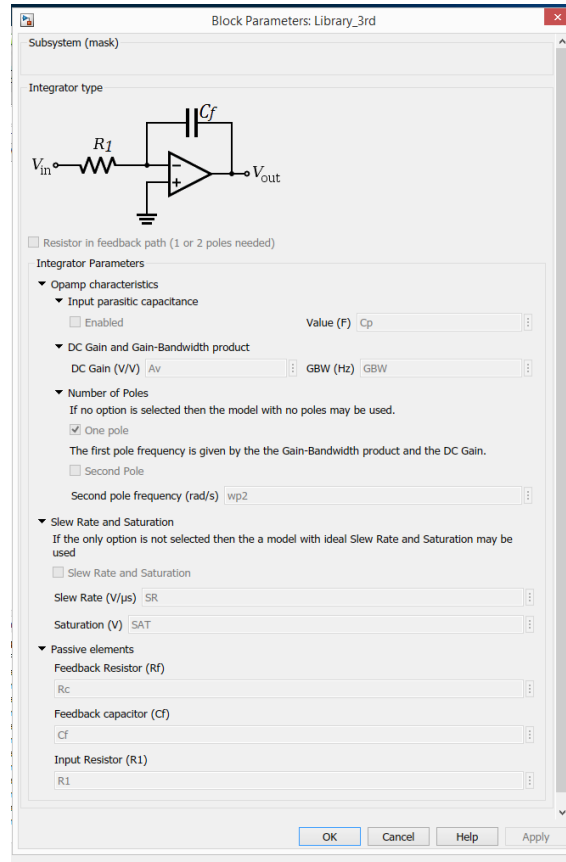


Fig. 3.34. Captura de la interfaz del elemento Integrador 1 entrada en SIMULINK

Ahora bien, hasta el momento solo se han definido la interfaz y dentro de ésta, qué parámetros del subsistema son modificables por el usuario. Ahora bien, no basta con esto, para que estos parámetros sean modificados, hay que volver a la máscara y modificar la pestaña que se denomina: 'Initialization Commands'. Aquí se programa cómo se cambian los datos introducidos por el usuario.

De esta pestaña hay que destacar lo siguiente:

- Es una ventana en la que solamente se escribe código. Presenta similitud a un archivo .m programable en MATLAB; sin embargo, está mucho más limitado.
- Presenta su propio espacio de trabajo, por lo que no entra en conflicto con aquellas variables declaradas en el espacio de trabajo principal de MATLAB, pero sí que incluye aquellas variables definidas en la pestaña 'Parameters Dialog', que es precisamente lo que interesa.
- El código presente en esta pestaña solo se ejecuta cuando se compila el modelo, se

CAPÍTULO 3. 3.5. BIBLIOTECA DESARROLLADA

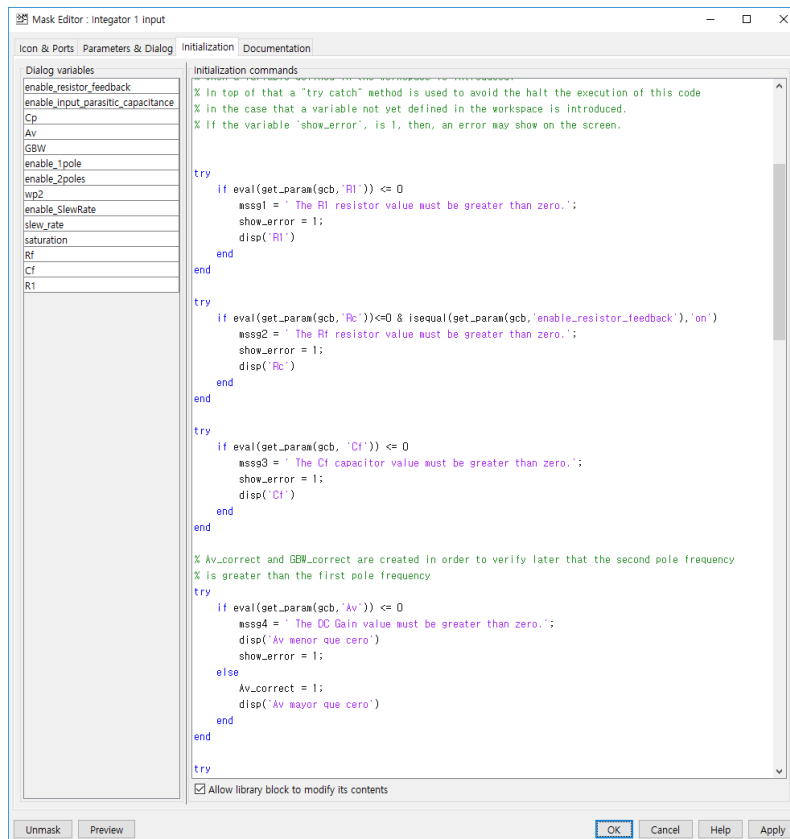


Fig. 3.35. Captura de la pestaña *Initialization commands*. Se muestra parte del código utilizado.

aplican cambios en este, y antes de ejecutar una simulación en el subsistema.

- Solamente modifica lo que hay debajo del subsistema, en ningún caso se encarga de la interfaz del subsistema.
- Permite la detección de errores a la hora de introducir parámetros, evitando así simular un subsistema con datos que no sean consistentes, como por ejemplo, resistencias y condensadores de valor negativo, que la posición del segundo polo sea menor que aquella del primer polo, etc.

En la siguiente figura se muestra una captura de la pestaña *Initialization Commands*, mostrando parte del código.

Es importante marcar la casilla de 'ALLOW SELF MODIFICATIONS' ya que esto permite que el sistema mediante la pestaña de Initialization Commands pueda modificarse a sí misma.

En la figura 3.34 muestra cómo queda la interfaz del elemento de la biblioteca; se aprecia que permite diferenciar entre integrador e integrador con camino feedforward.

Mediante el uso de bloques se ha conseguido sintetizar en un solo elemento todos los modelos relativos al integrador con una sola señal de entrada, sea o no el integrador con camino feedforward,

3.5.2. Integradores con múltiples entradas

Aquí se crean tres elementos en total, para el integrador de dos, tres y cuatro entradas. Independientemente del número de entradas, los pasos a seguir para implementar el subsistema y definir la máscara es exactamente igual al modelo de 1 entrada. Cada integrador presenta su propio diagrama de bloques equivalentes al de la figura 3.29.

Como se comentó al presentar el modelado de los integradores con más de una entrada, se considera siempre que el componente pasivo entre el terminal de entrada negativo del OpAmp y lo que se considera $V_1(s)$ es siempre una resistencia. Esto es importante de cara a la creación de la máscara, ya que aunque se pueda cambiar la naturaleza de los demás elementos, el primero siempre es una resistencia.

Una vez aclarada la diferencia, se presentan las distintas interfaces en la figura 3.36. Debido a la existencia de más entradas, y que el componente pasivo puede ser un condensador o una resistencia, el número de modelos es considerable. Sin embargo, como se ha demostrado anteriormente, para un mismo número de entradas, se pueden sintetizar todos los modelos en un solo diagrama de bloques, y por lo tanto en un solo elemento.

En total, el número de modelos de integradores es de 70, pero el número de elementos en la biblioteca de SIMULINK, respecto a integradores, es tan solamente de 4.

3.5.3. Oscilador en anillo controlado por tensión

Con el fin de implementar un modelo en SIMULINK del oscilador en anillo, se usa el modelo descrito en la sección correspondiente. El paso de este modelo a SIMULINK es inmediato, con tan solo un par de ajustes. Un ejemplo de su implementación se presenta en la figura 3.37

A diferencia del integrador, con el oscilador en anillo el número de modelos viene definido, además de por sus no idealidades, por el número de inversores que presente. Esto plantea un problema dado que, en principio, sería necesario crear un elemento en la

CAPÍTULO 3. 3.5. BIBLIOTECA DESARROLLADA

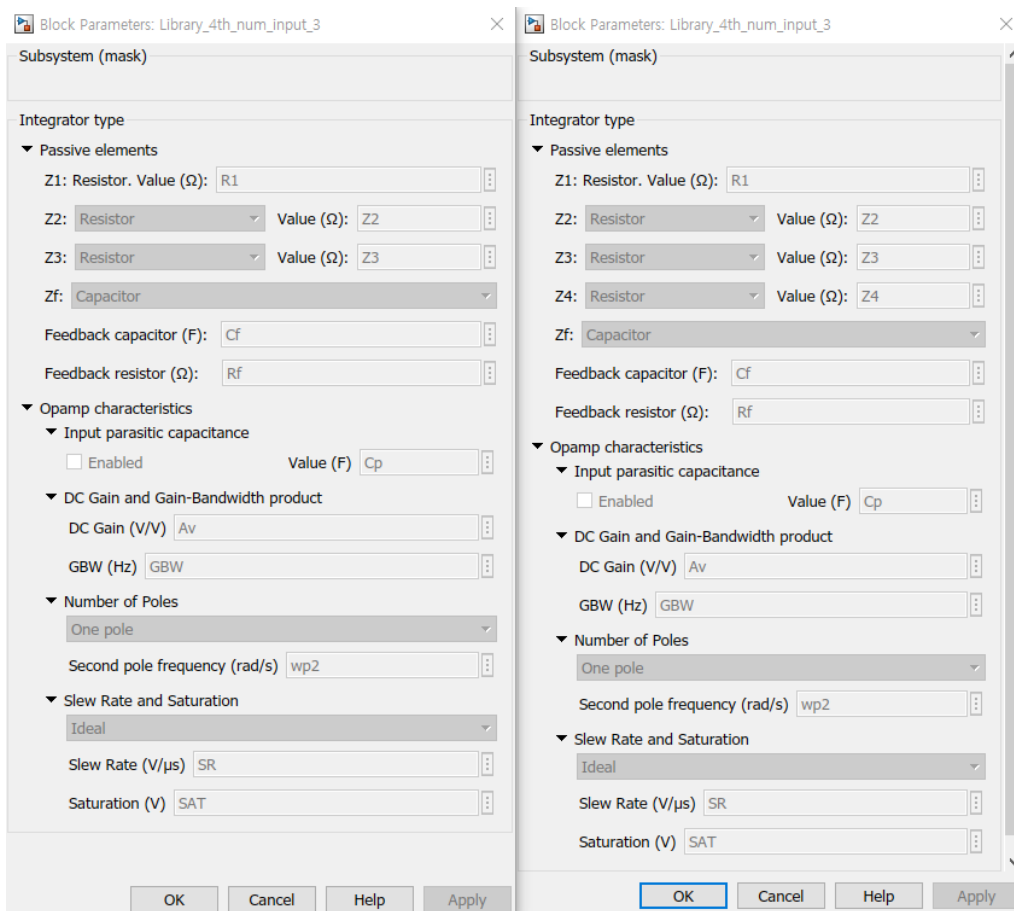


Fig. 3.36. Interfaz para la introducción de parámetros para los integradores de 3 y 4 entradas.

biblioteca de SIMULINK para cada caso diferente. Es decir, un elemento, por ejemplo, para 11 inversores, otro para 13 inversores, etc. Obviamente realizar esto es muy ineficiente, además tener una biblioteca con demasiados elementos no es práctico ni mucho menos *user friendly*. Por esto mismo, se decide hacer uso otra vez de los subsistemas y sus máscaras.

El funcionamiento de la máscara en la pestaña *Parameters Dialog* es similar a aquella de los integradores, aunque los parámetros a definir sean distintos, esto carece de importancia. Sin embargo, la pestaña *Initialization Commands* es muy diferente, ya que ahora no se limita sólo a cambiar parámetros, si no que además ahora crea bloques. Esta creación de bloques permite que en un solo elemento de la biblioteca se pueda reflejar un oscilador en anillo con un número de inversores que el usuario requiera; es decir, se consigue una síntesis máxima.

Como muestra la figura 3.38, se puede elegir el número de inversores, definir la frecuencia central de oscilación y la ganancia o bien, introducir un polinomio que defina la

HERRAMIENTA DE MODELADO Y SIMULACIÓN DE CONVERTIDORES SOBREMUESTREADOS EN MATLAB/SIMULINK

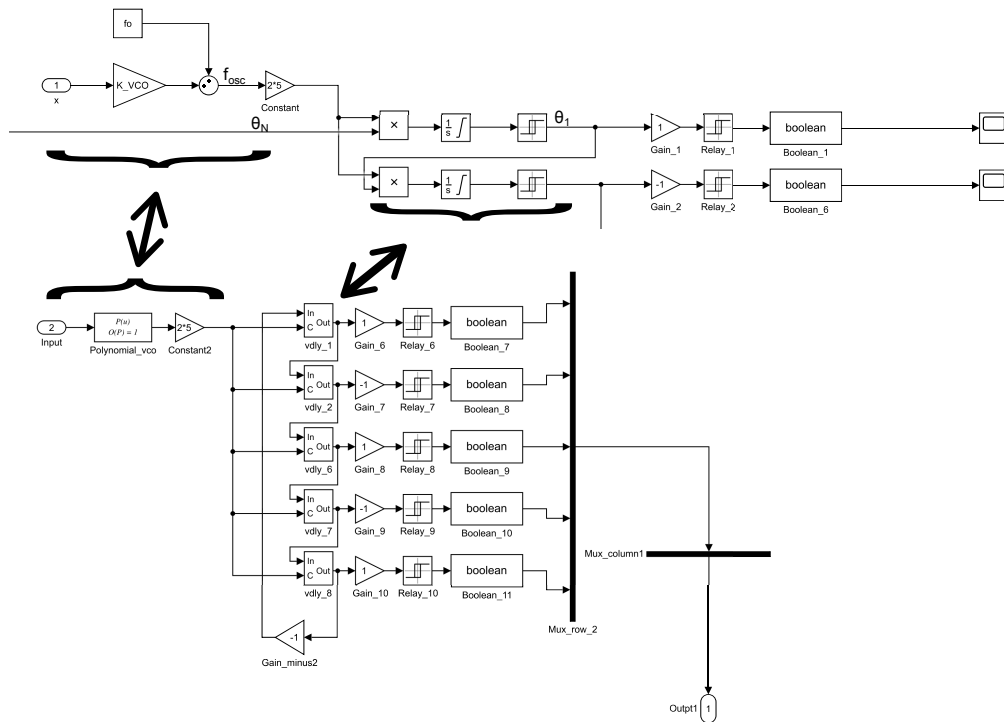


Fig. 3.37. Modelo oscilador en anillo. En la parte superior se muestran dos inversores de un oscilador en anillo; en la parte inferior se muestra un modelo compacto de un oscilador en anillo de 5 inversores.

no linealidad del VCO.

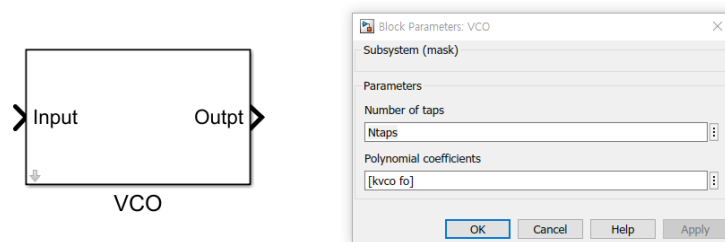


Fig. 3.38. Subsistema del VCO y su correspondiente máscara

Cabe destacar que debido a la creación de nuevos bloques, la colocación y conexión entre estos, hace que cuanto mayor sea el cambio entre el nuevo y anterior modelo, el tiempo que tome la máscara en aplicar los cambios es mayor. Se muestra como es el bloque por dentro en la figura 3.39

Dentro de la pestaña 'Initialization Commands' se encuentra el código responsable de

CAPÍTULO 3. 3.5. BIBLIOTECA DESARROLLADA

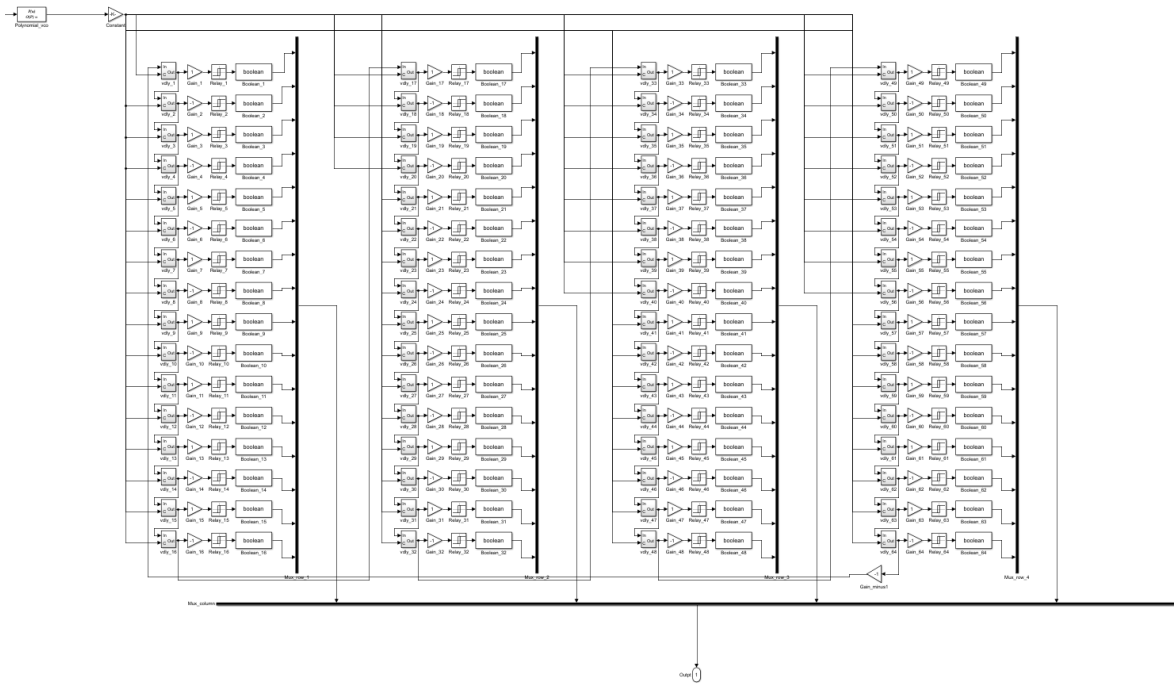


Fig. 3.39. Ejemplo de un VCO con 64 etapas creado con la herramienta.

realizar los cambios dentro del subsistema y tomando como datos aquellos introducidos en la pestaña de parámetros.

3.5.4. Validación de la biblioteca

Para validar y comprobar el correcto funcionamiento de los elementos que componen la biblioteca se ha procedido a realizar varias pruebas. Diferenciamos las pruebas realizadas al integrador y aquellas realizadas al oscilador en anillo. Dado que el VCO viene a nacer directamente de unas ecuaciones, no es necesaria la verificación del modelo.

Integrador: Para el integrador se ha optado por verificar los modelos desarrollados con el software LTSpice. Dado que el número de modelos es muy alto, se presentan solamente unas cuantas comparaciones, presentando primeramente el modelo utilizado en LTSpice y luego comparando respuestas transitorias.

La metodología de comparación es la siguiente: Se comparan simulaciones temporales y la respuesta en frecuencia. Aunque se vea que superponiendo las figuras de LTSpice y de los elementos de la biblioteca, tomando en ambos casos el mismo paso máximo de simulación. Las simulaciones se han realizado sobre el integrador de una entrada, dos y

HERRAMIENTA DE MODELADO Y SIMULACIÓN DE CONVERTIDORES SOBREMUESTREADOS EN MATLAB/SIMULINK

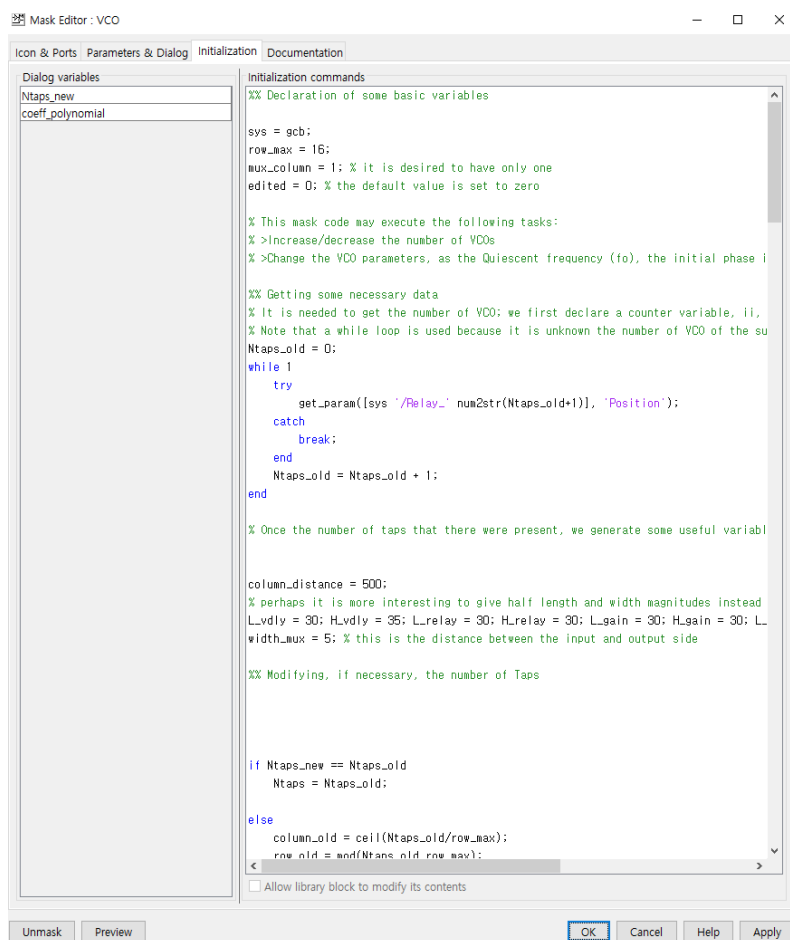


Fig. 3.40. Captura que muestra la pestaña *Initialization commands* y parte del código que se encuentra en ella

tres entradas, se presentan diversos casos:

- Caso 1: Integrador de una entrada y dos polos.
- Caso 2: Integrador *feedforward* de una entrada, condensador parásito a la entrada y dos polos.
- Caso 3: Integrador con dos entradas con elementos resistivos y condensador parásito, con un polo.
- Caso 3: Integrador con dos entradas con elementos resistivos y condensador parásito, con un polo.
- Caso 4: Integrador *feedforward* con tres entradas, siendo el elemento de entrada en todas del tipo resistivo, con condensador parásito y sin polos.
- Caso 5: Integrador *feedforward* con tres entradas, siendo solamente el primer ele-

CAPÍTULO 3. 3.5. BIBLIOTECA DESARROLLADA

mento de entrada de tipo resistivo, se considera que solo tiene un polo.

Debido a que LTspice no ofrece la oportunidad de tomar datos cada cierto tiempo fijo, los resultados no se pueden comparar número a número, así que se ha procedido a coger los máximos valores de tensión y también la diferencia en la tensión media para un mismo tiempo de simulación, que se producen a la salida, tanto en LTspice y en SIMULINK y se resumen en la siguiente figura estas:

	Caso 1:	Caso 2:	Caso 3	Caso 4	Caso 5
dif_max (V)	0.362 μ	518 μ	216 μ	4.82 μ	0.482 μ
dif_min (V)	0.675 μ	33.7 μ	122 μ	0.337 μ	33.7n
dif_media (V)	13.4 μ	671 μ	309 μ	6.71 μ	0.671 μ

Fig. 3.41. Diferentes casos y diferencias máximas obtenidas

4. USO DE LA BIBLIOTECA PARA LA CARACTERIZACIÓN DE UN VCO-ADC

Sentadas las bases teóricas necesarias para entender el funcionamiento de los Convertidores ADC basados en VCO, se procede a describir una arquitectura en concreto: Convertidor ADC basado en VCO con frecuencia de muestreo incrementada (*Time Interleaved*). Varias formas de implementar este proceso han sido propuestas en la literatura [29] [30]; sin embargo, en el proyecto de colaboración se basará en la que se propone en [31].

4.1. Arquitectura de convertidor ADC basado en VCO con frecuencia de muestreo aumentada.

La arquitectura propuesta en [31] define una forma de aumentar la frecuencia de muestreo de forma virtual añadiendo líneas de retraso de N elementos en cada fase de salida del VCO (Fig. 4.1).

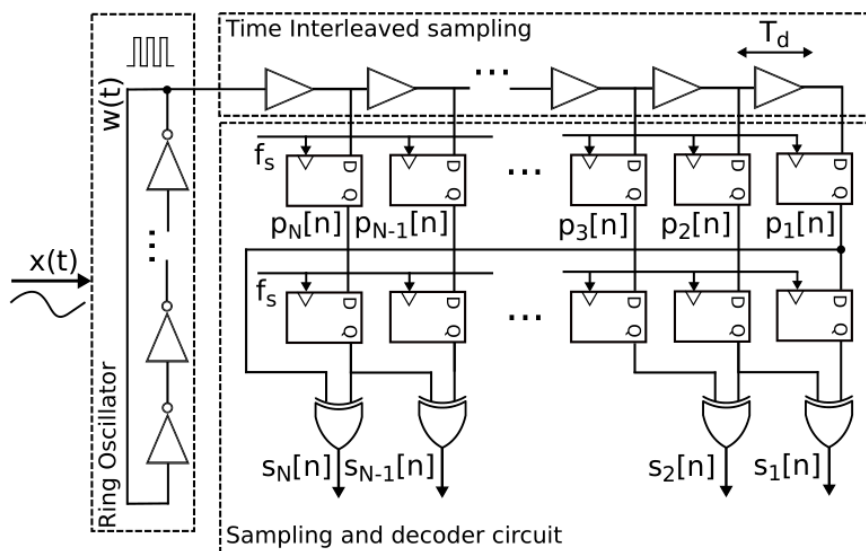


Fig. 4.1. Oscilador en anillo con matriz de muestreo mediante cadena de retrasos y circuito decodificador. [7]

El tiempo de retardo de cada elemento vendría dado por $T_d = T_s/N$, donde T_s es el período de muestreo, y N el número de elementos de delay. La salida $y[n]$ se toma como una

CAPÍTULO 4. 4.1. ARQUITECTURA DE CONVERTIDOR ADC BASADO EN VCO CON FRECUENCIA DE MUESTREO AUMENTADA.

cadena ordenada tal que:

$$y_j[n] = \sum_{i=1}^N y_{ji}[n], \quad (4.1)$$

donde j representa el número de la fase del VCO.

Se puede observar que cada elemento de $y_j[n]$ estará demodulada en el tiempo y por lo tanto presentará también conformado espectral de ruido de primer orden. Así entonces, la frecuencia efectiva de muestreo sería ahora f_d , equivalente a $N \cdot T_s$. Estos retrasos, a nivel de circuito son implementados con buffers; sin embargo, hacerlo con N relojes con una frecuencia f_s de muestreo y retrasados uno respecto al anterior T_s/N es también equivalente. Además, los relojes son más estables que los buffers ante variaciones PVT.

Así entonces, quedaría definido un convertidor ADC basado en VCO con conformado espectral de ruido de primer orden con frecuencia aumentada de muestreo. Esta misma técnica se aplica a un convertidor ADC basado en VCO con conformado espectral de ruido de segundo orden, donde el primer integrador sería un OTA y el segundo el VCO.

Así pues, en los siguientes apartados se procede a estudiar el funcionamiento del sistema utilizando los elementos de la biblioteca propuestos en el presente documento. Para analizar de forma eficiente el funcionamiento del sistema es recomendable empezar considerando que es cuasiideal. Esto, que en un primer momento puede llegar a parecer inadecuado ya que el circuito que será finalmente implementado es, al fin y al cabo, real con todos los problemas que eso conlleva, es una forma útil de poder ir resolviendo los distintos problemas relacionados que surjan con cada parte modelada del circuito.

Las simulaciones se realizan con los datos de la tabla 4.1, donde el ancho de banda y el SNR son las especificaciones del sistema, los demás datos son los que se han tomado como nominales para la caracterización, donde $DAC_{(p+n)}$ es el número total de elementos del DAC, rama positiva y negativa incluidas, f_c es la frecuencia de los relojes, f_s es la frecuencia equivalente de muestreo, f_{VCOmax} es la frecuencia máxima de oscilación del VCO, $T_{inv-min}$ es el tiempo mínimo de inversión y $T_{inv-mid}$ es el tiempo de inversión medio.

HERRAMIENTA DE MODELADO Y SIMULACIÓN DE CONVERTIDORES SOBREMUESTREADOS EN MATLAB/SIMULINK

Caso	Nº etapas VCO	Nº Relojes	DAC _(p+n)	f _c	f _s	f _{VCOmax}	T _{inv-min}	T _{inv-mid}
1	5	4	40	4GHz	16GHz	8GHz	12.5ps	25ps

TABLA 4.1. TABLA CON LOS VALORES NOMINALES PARA LA CARACTERIZACIÓN DEL CONVERTIDOR ADC.

4.2. Modelo comportamental del VCO-ADC

El modelo comportamental del convertidor ADC se describe en SIMULINK. En la Fig. 4.2 se señalan los distintos bloques que modelan el sistema. Concretamente, se va a tener como foco de atención los elementos de la biblioteca usados en el sistema. Brevemente se describen los distintos elementos:

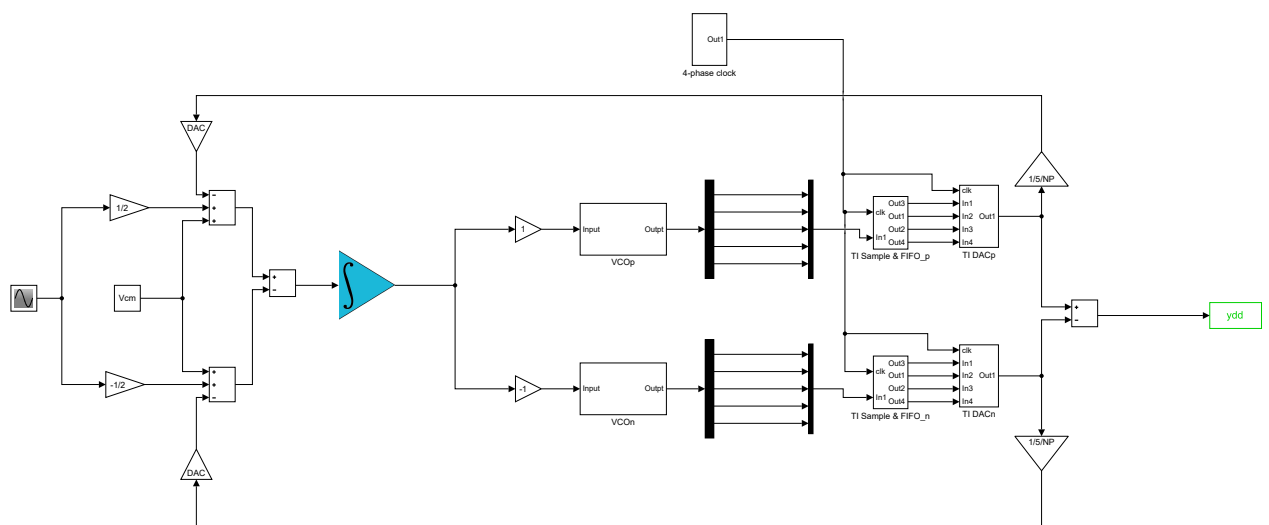


Fig. 4.2. Conversor ADC basado en VCO con frecuencia de muestreo aumentada.

- Entrada: Señal senoidal de entrada, tiene una frecuencia de 25MHz y una amplitud de -10dB. Entra al modulador en forma diferencial.
- Integrador RC: Primer filtro del convertidor basado en un OTA.
- VCO: Segundo filtro del convertidor, además de cuantificador y modulador de la señal que tiene a su entrada. Presenta 5 etapas (Fig. 4.3).
- Relojes: Bloque que saca en un bus 4 señales pertenecientes a 4 relojes con la misma

CAPÍTULO 4. 4.2. MODELO COMPORTAMENTAL DEL VCO-ADC

frecuencia retrasados uno respecto a otro $T_s/4$.

- Matriz de muestreo y bloque decodificador: Aquí, un mismo bloque aúna ambas partes que es la encargada del aumento de la frecuencia del muestreo y la decodificación de la salida de los datos del VCO. La matriz de muestreo funciona con 4 relojes (Fig. 4.4).
- DAC: Se encarga de realimentar de forma analógica los datos que salen del bloque inmediatamente anterior, se compone de 20 elementos, ya que hay 5 etapas en el VCO y 4 relojes.
- Las distintas ganancias simplemente son para ajustar las señales al fondo de escala que corresponda en cada momento.

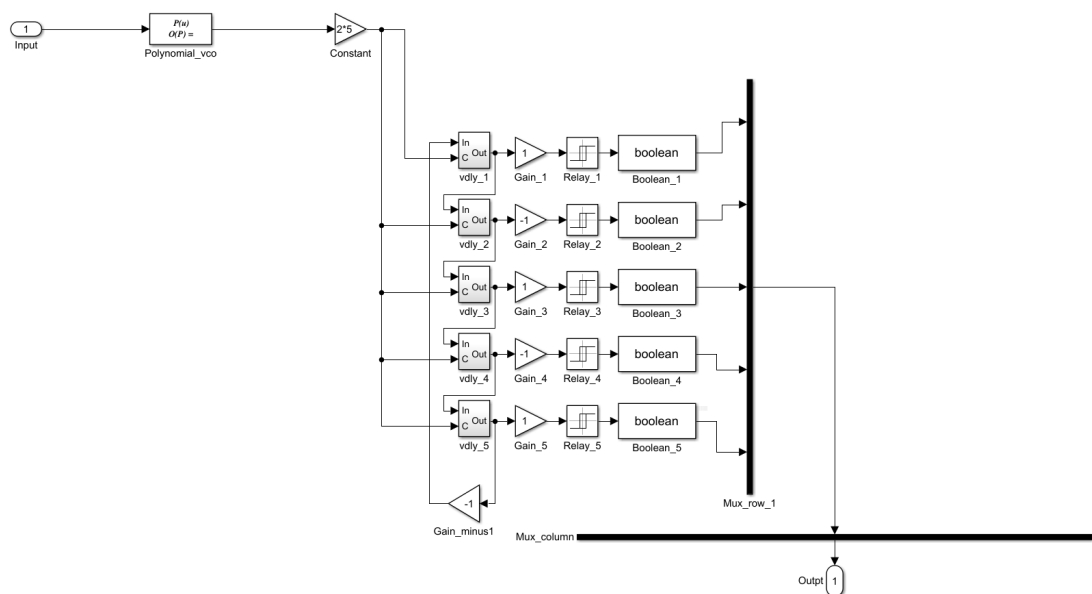


Fig. 4.3. Modelo de oscilador en anillo con 5 etapas.

Se procede a realizar entonces una simulación que será la de ‘referencia’, es decir, aquella que es totalmente ideal, salvo por el integrador RC, que presenta un valor de GBW bastante alto. Los datos de simulación denominados como nominales son los mostrados en la tabla 4.2.

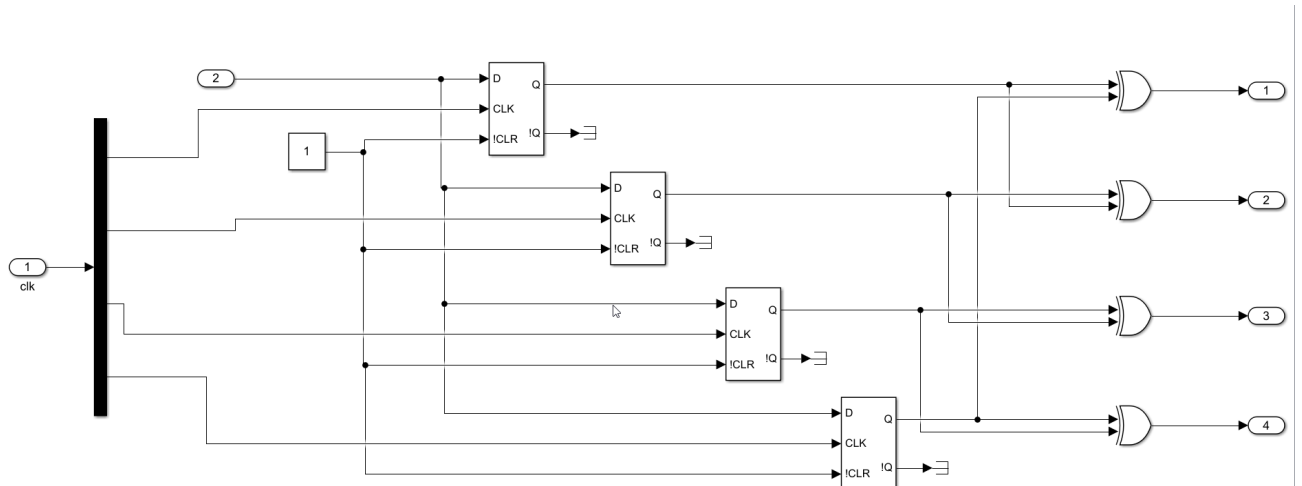


Fig. 4.4. Matriz de sobremuestreo y circuito de decodificación.

Parámetro	Valor
R_1	1000 Ω
R_2	650 Ω
C_1	31.25pF
A_v	80dB
GBW_{OTA}	50GHz

TABLA 4.2. PARÁMETROS NOMINALES.

4.3. Velocidad y Ganancia del OTA

En este apartado lo que se pretende es obtener unas especificaciones respecto a qué valores de ganancia y ancho de banda tiene que cumplir el OTA para que así el sistema funcione correctamente. Es importante ver y analizar cómo se degrada el rendimiento del sistema cuando el OTA no llega a las especificaciones, o cuando se produce una variación en sus valores respecto al considerado nominal.

Todos los resultados que se obtienen al analizar la salida se muestran en el dominio de la frecuencia, ya que se puede observar el conformado espectral de ruido y su orden, si la

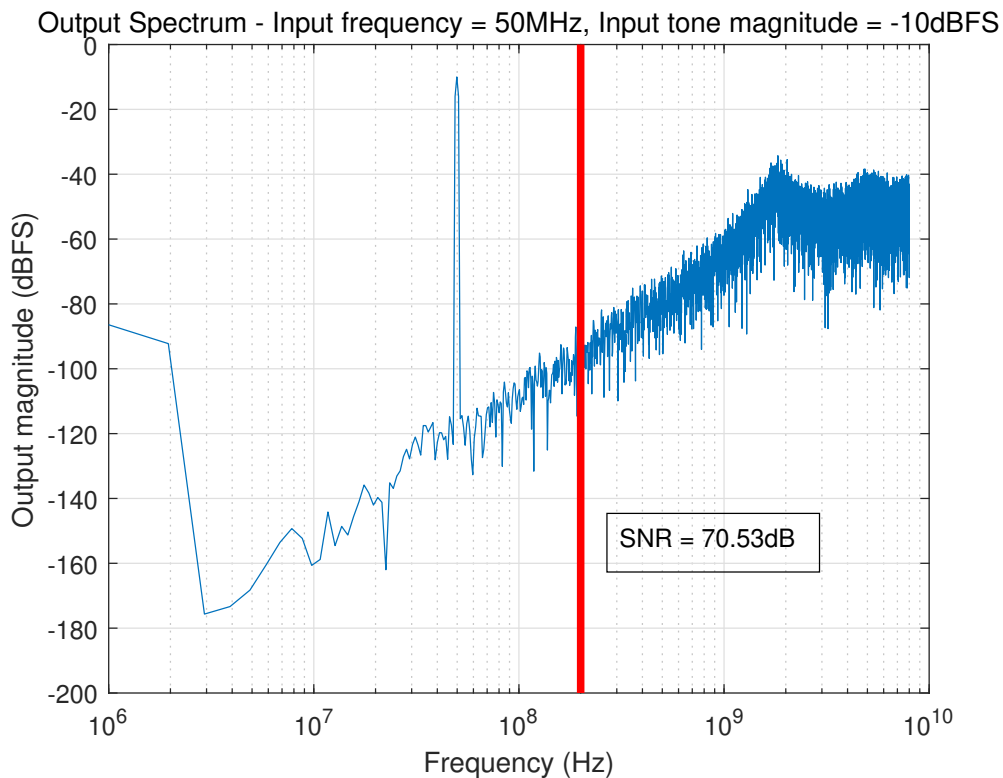


Fig. 4.5. Espectro de salida obtenido para condiciones de simulación nominales.

frecuencia y amplitud del tono de salida corresponde con aquel del tono de entrada.

En la fig. 4.5 se simula el sistema con una OTA con un GBW exageradamente alto. Como primera aproximación se considerará el modelo *feedforward* con ganancia por ancho de banda finito. Luego, se proceden a hacer dos barridos, uno de ancho de banda y otro de ganancia.

4.3.1. Ganancia por Ancho de banda

Aquí la ganancia utilizada no es infinita, aunque tiene un valor muy grande (miles de dB) que permite centrar el foco de atención solamente en la GBW . Entonces, se realiza un barrido para calcular la SNR solamente en función de la GBW del OTA, BW_{OTA} , que no el ancho de banda objetivo del sistema, GBW ; se mantienen los demás parámetros constantes.

Como se ve en la figura Fig. 4.6, la SNDR obtenida en función de la GBW presenta un comportamiento creciente asintótico. Este resultado es de esperar ya que con un integrador real la SNR obtenida tiene un valor finito. Se puede apreciar también que el valor de

HERRAMIENTA DE MODELADO Y SIMULACIÓN DE CONVERTIDORES SOBREMUESTREADOS EN MATLAB/SIMULINK

la GBW a partir del cual el rendimiento del sistema se empieza a deteriorar de forma considerable es 5 GHz. Así entonces, seleccionando un valor de GBW superior al crítico, se calcula el espectro de salida (Fig. 4.7) para observar que, efectivamente, éste presenta un comportamiento característico y no hayan aparecido armónicos o distorsión a causa de un bajo ancho de banda.

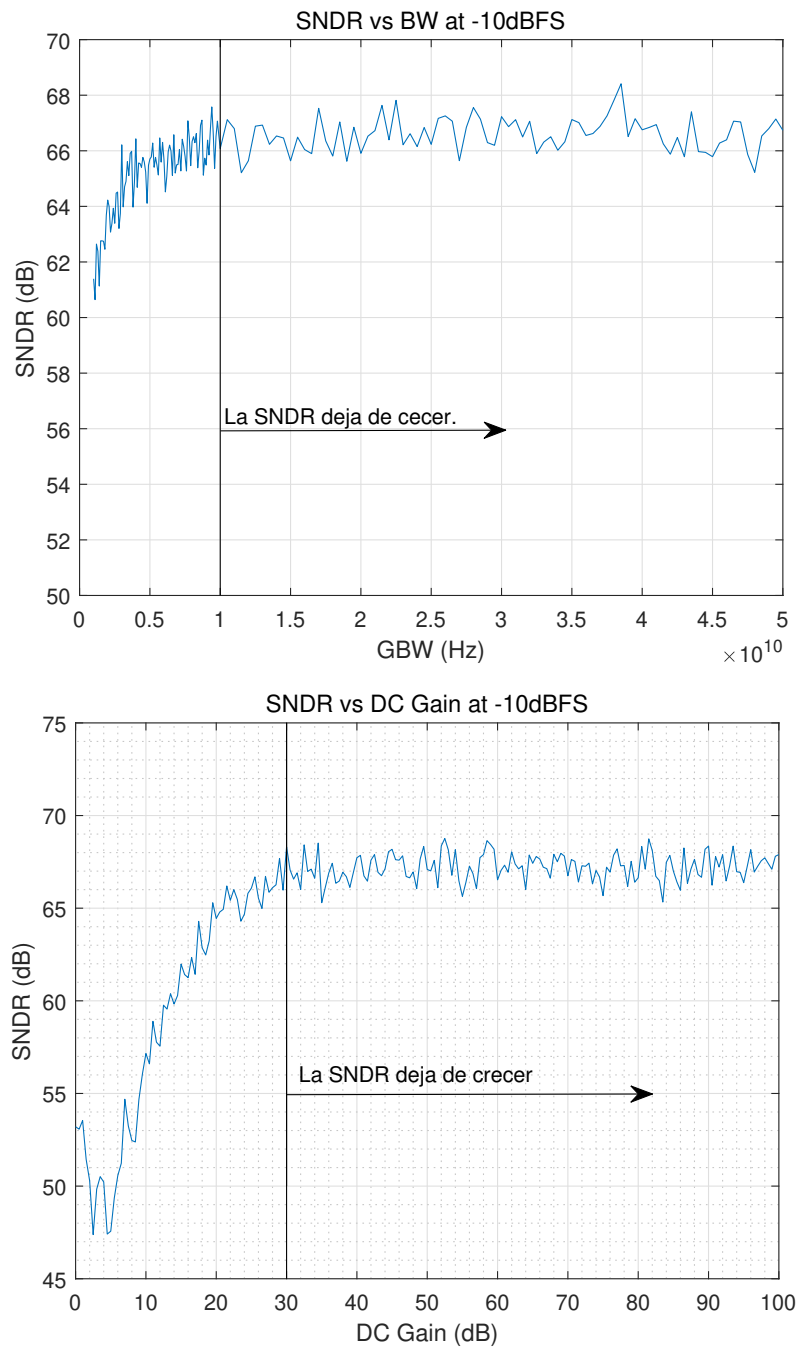


Fig. 4.6. (a) Barrido de GBW_{OTA} para un valor alto pero fijo de ganancia. (b) Barrido de Av_{OTA} habiendo seleccionado un valor de $GBW_{OTA} = 10\text{GHz}$.

4.3.2. Ganancia

Aquí con el valor fijado anteriormente de GBW_{OTA} , se procede a realizar un barrido de la SNR en función de la ganancia del OTA, Av_{OTA} ; al igual que antes, el resto de parámetros permanecen constantes. Se observa otra vez, Fig. 4.6, que el valor de la SNR presente un crecimiento asintótico en función de la ganancia. Como se aprecia el valor crítico de deterioro de SNR es de $Av_{OTA} = 30\text{dB}$. Seleccionando un valor algo superior al crítico para dejar cierto margen a un cambio en la ganancia, se calcula el espectro de salida (Fig. 4.8)

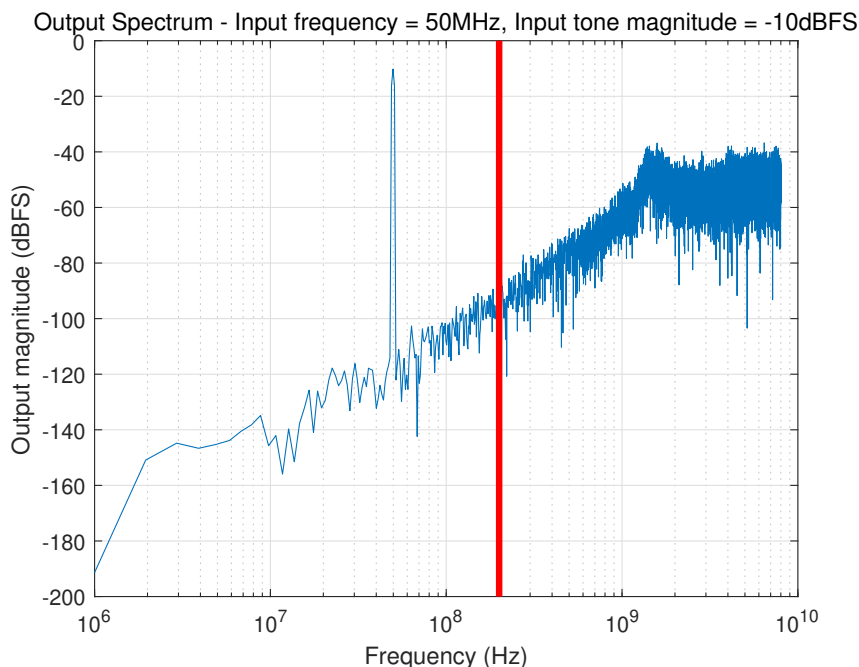


Fig. 4.7. Espectro de salida para $GBW_{OTA} = 10\text{GHz}$, SNR = 68dB.

De este modo se ha logrado caracterizar el OTA en términos de GBW considerando únicamente la presencia de un sólo polo en el OTA. Sin embargo, también se puede analizar y simular el comportamiento del sistema con un OTA de dos polos. Típicamente, usando tecnologías nanométricas se puede esperar que el segundo polo esté a frecuencias de decenas de GHz. Puesto que el dato del segundo polo no es concreto, las simulaciones no son más que una mera estimación para prever cómo sería la dinámica del sistema. Así entonces, se coloca un segundo polo en, por ejemplo, 12GHz y se obtiene el siguiente espectro de salida Fig. 4.9. Se observa que este cambio en el OTA apenas refleja cambios en el espectro y la disminución/aumento de SNR es mínima.

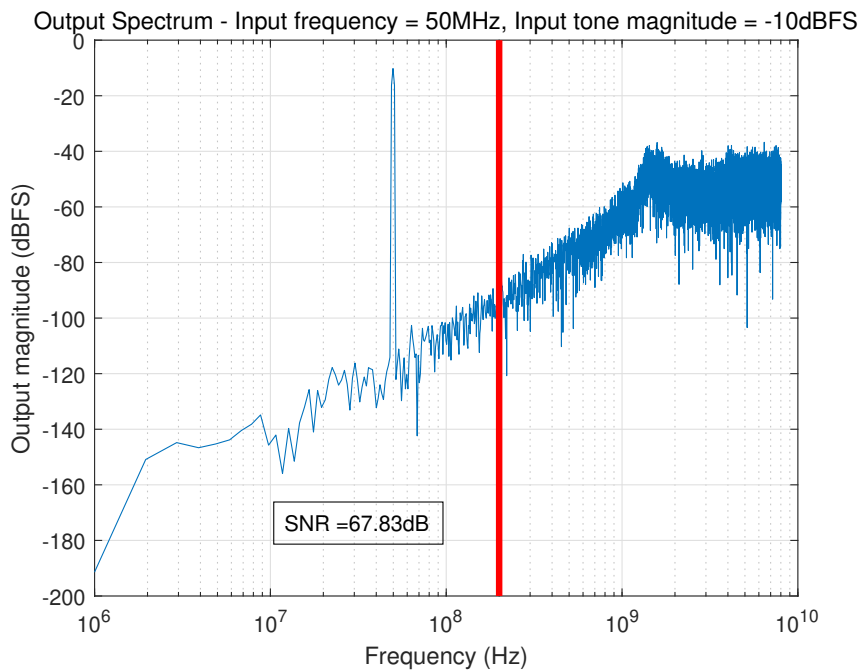


Fig. 4.8. Espectro de salida para $GBW_{OTA} = 10\text{GHz}$, $A_{vOTA} = 32\text{dB}$

Los resultados obtenidos indican que las características obtenidas para el OTA son posibles de alcanzar para una tecnología de 28nm, lo que indica que el OTA, en principio, no es una parte crítica en el desarrollo del convertidor. Además, en caso de haber un polo en 12GHz se ha comprobado que apenas afecta a la dinámica del sistema.

4.4. Sensibilidad del ADC a la resolución y no linealidad del VCO y a la velocidad de muestreo

Este apartado pretende estudiar cómo evoluciona el rendimiento del ADC ante cambios tanto en parámetros del VCO como en la frecuencia de muestreo.

4.4.1. Cambios en la resolución del VCO y en la velocidad de muestreo

Como se ha indicado anteriormente, el VCO también realiza la cuantificación de la señal que tiene a su entrada, es decir, a mayor números de inversores en el VCO, mayor será la resolución del VCO y por tanto mayor será la resolución del ADC tal y como refleja la ecuación (3.11), donde N es el número de etapas del VCO. Esto, sumado a la configuración de *Time Interleaved* que también permite aumentar la frecuencia de muestreo, lo que aumenta el OSR, lo que a su vez aumenta también la resolución del ADC.

CAPÍTULO 4. 4.4. SENSIBILIDAD DEL ADC A LA RESOLUCIÓN Y NO LINEALIDAD DEL VCO Y A LA VELOCIDAD DE MUESTREO

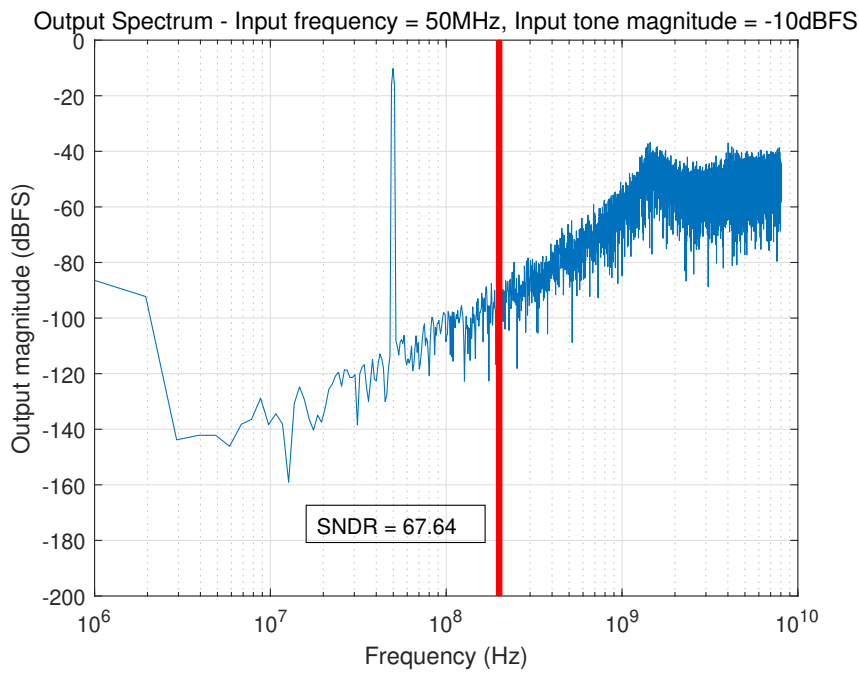


Fig. 4.9. Espectro de salida para $GBW_{OTA} = 10\text{GHz}$, $A_{VOTA} = 32\text{dB}$, segundo polo en 12GHz.

Se cambia el número de etapas en el VCO y el número de relojes para combinar distintas opciones. Esto principalmente conlleva cambios a nivel de cómo se realizaría esta parte del sistema, también provoca cambios en el número de registros, el número de elementos presentes el DAC, así como el propio tiempo de retraso asociado a cada inversor del VCO, etc. En la siguiente tabla se resumen una serie de distintos parámetros cuyos valores cambian directamente con el número de etapas y el número de relojes. Las distintas opciones tienen el mismo SNR objetivo.

Caso	Nº etapas VCO	Nº Relojes	$DAC_{(p+n)}$	f_c	f_s	f_{VCOmax}	$T_{inv-min}$	$T_{inv-max}$
1	5	4	40	4GHz	16GHz	8GHz	12.5ps	25ps
2	27	2	108	4GHz	8GHz	4GHz	4.6ps	9.3ps
3	11	2	44	5.7GHz	11.4GHz	5.7GHz	8ps	15.9ps
4	3	8	48	5GHz	40GHz	20GHz	8.3ps	16.7ps

TABLA 4.3. TABLA CON LOS VALORES NOMINALES PARA LA CARACTERIZACIÓN DEL CONVERTIDOR ADC.

Así pues, obtenemos una serie de distintas opciones, cada una con sus propias ventajas e inconvenientes. Por ejemplo, un tiempo de inversión tan pequeño puede llegar a ser

HERRAMIENTA DE MODELADO Y SIMULACIÓN DE CONVERTIDORES SOBREMUESTREADOS EN MATLAB/SIMULINK

complicado de alcanzar, e implementar un DAC que funcione a una velocidad de decenas de GHz plantea, sin duda alguna, un reto complejo y exigente.

Para simular los distintos casos de la tabla 4.3 es necesario modificar los parámetros del OTA, no sólo en su GBW si no que también en cuanto a los valores de sus componentes pasivos. En la tabla siguiente se resumen los cambios con los que se han obtenido los distintos espectros. La posición del segundo polo se ha mantenido fija.

<i>Caso</i>	R_1	R_2	C_1	GBW_{OTA}	Av_{OTA}
1	1000 Ω	650 Ω	31.25pF	10GHz	32dB
2	990 Ω	300 Ω	31.25pF	20GHz	60dB
3	1000 Ω	650 Ω	31.25pF	10GHz	32dB
4	500 Ω	405 Ω	31.25pF	20GHz	58dB

TABLA 4.4. TABLA CON LOS VALORES NOMINALES PARA LA CARACTERIZACIÓN DEL CONVERTIDOR ADC.

Se obtienen los siguientes espectros de salida:

Los mejores espectros son los del caso 1 y 3. El caso 2 presenta una oscilación en 1GHz que en caso de llevarse a circuito acabaría por volver inestable al sistema. El caso 4 en altas frecuencias presenta cierta distorsión. Se resalta una vez más la importancia de no sólo quedarse con el valor número de la SNR o SNDR. Posiblemente el mal funcionamiento del caso 2 se deba a que su tiempo de inversión es demasiado pequeño. Además, los casos 1 y 3 son los que presentan menores exigencias al OTA, así que son casos bastantes prometedores.

4.5. VCO con comportamiento no lineal

A partir del programa Cadence Virtuoso se obtiene el polinomio del VCO, esto es, una función que aproxima la relación Frecuencia de Salida-Tensión de Entrada del VCO. En la figura 4.11 se comparan la curva ideal con la obtenida en CADENCE para un oscilador en anillo en 28nm.

Entonces, utilizando el caso 1, esto es, 4 relojes y 5 VCOs, obteniendo así una frecuencia de muestreo equivalente a 20GHz, se estudia la no linealidad del VCO. Se introduce

CAPÍTULO 4. 4.5. VCO CON COMPORTAMIENTO NO LINEAL

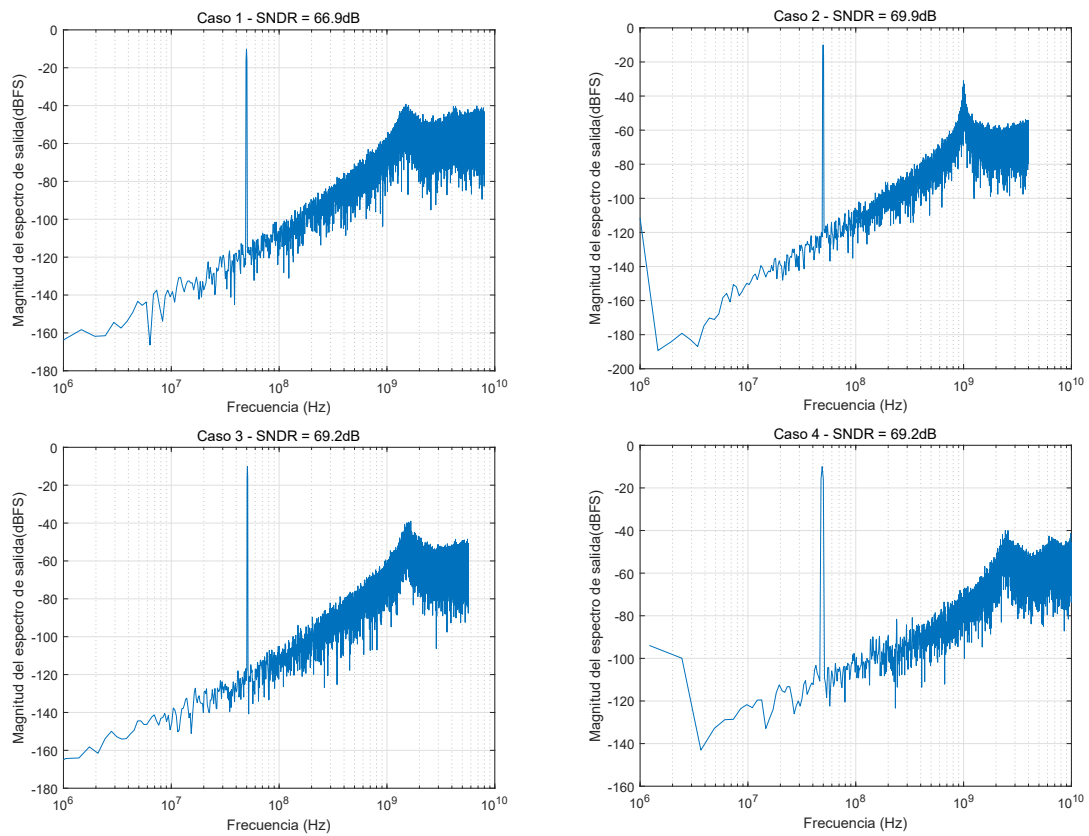


Fig. 4.10. Espectro de salida para los diferentes casos de número de etapas y número de relojes.

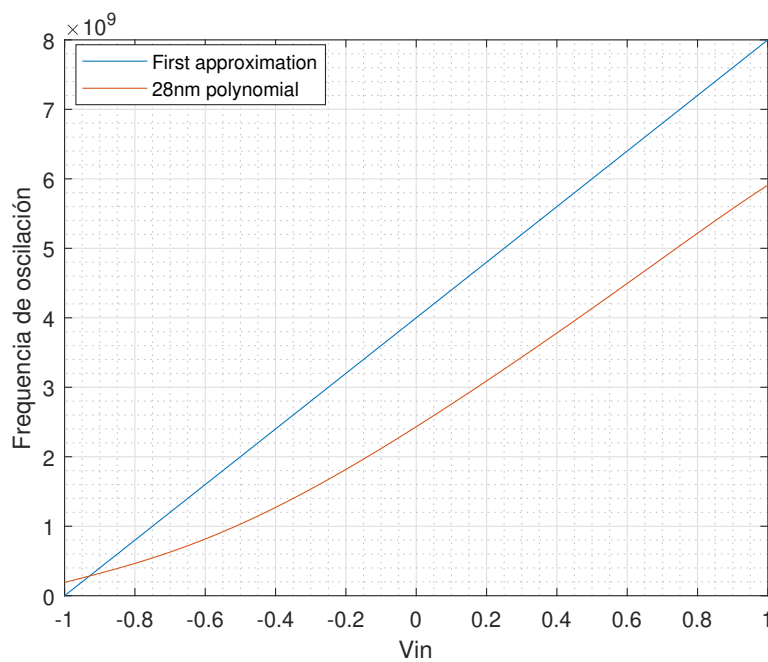


Fig. 4.11. Comparación entre la relación ideal y un polinomio para un oscilador en anillo en 28nm.

HERRAMIENTA DE MODELADO Y SIMULACIÓN DE CONVERTIDORES SOBREMUESTREADOS EN MATLAB/SIMULINK

en el modelo del VCO el polinomio y se procede a comparar tanto el espectro de salida obtenido como el rango dinámico.

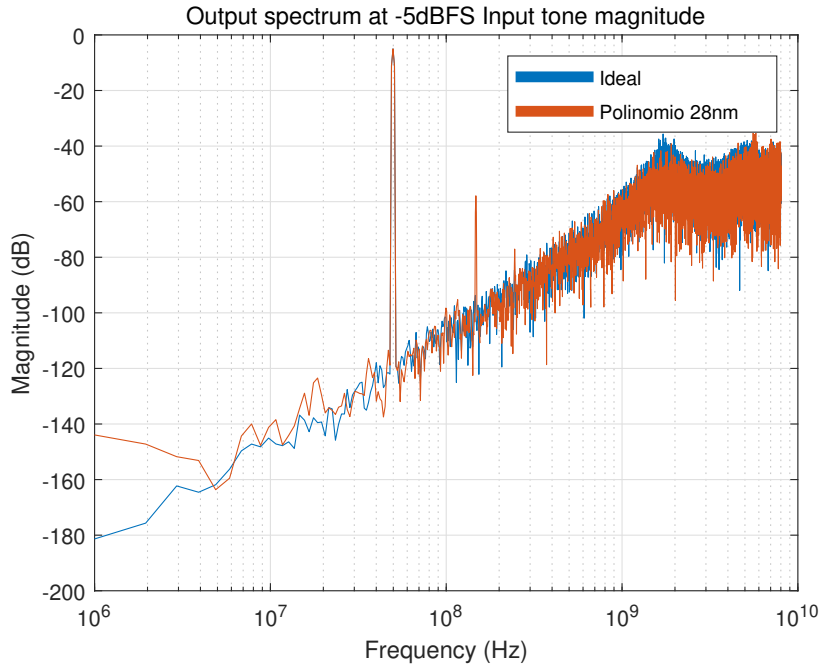


Fig. 4.12. Espectro de salida para el convertidor ADC basado en VCO con una OTA con $GBW_{OTA} = 10\text{GHz}$, $Av_{OTA} = 32\text{dB}$, segundo polo en 12GHz .

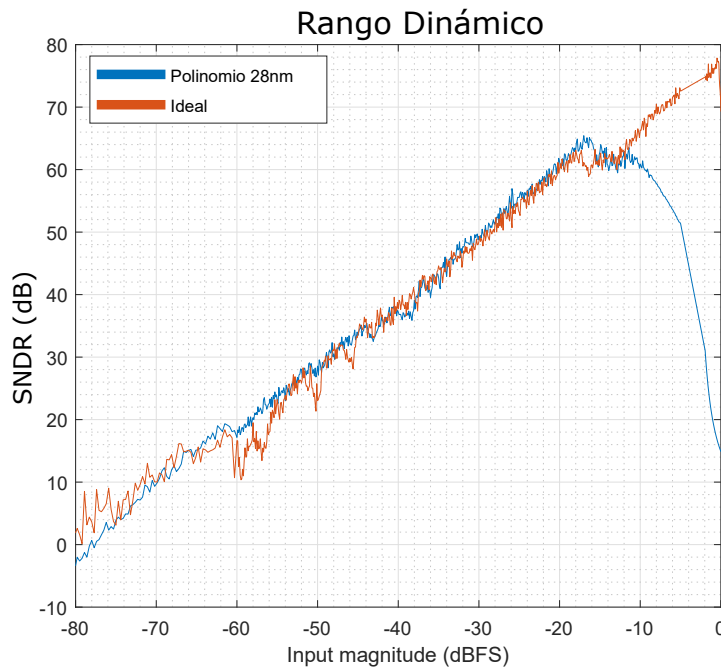


Fig. 4.13. Rango Dinámico para el convertidor ADC basado en VCO con una OTA con $GBW_{OTA} = 10\text{GHz}$, $Av_{OTA} = 32\text{dB}$, segundo polo en 12GHz .

CAPÍTULO 4. 4.5. VCO CON COMPORTAMIENTO NO LINEAL

Se observa en la Fig. 4.13 que como era de esperar el rango dinámico es peor y presenta un comportamiento más realista. Así pues, para entradas de mayor amplitud el sistema sufre una degradación en el rendimiento e incluso a partir de cierto punto se vuelve inestable. Sin embargo, cabe destacar que el valor de amplitud a partir del cual se empieza a sufrir degradación, -15dBFS , es particularmente bajo y que sería óptimo y recomendable que fuera mayor, alrededor de -5dBFS , por ejemplo.

Comparando los dos espectros de salida, del modelo ideal y de aquel con el polinomio, se observa que, aunque otra vez el valor de la SNR pueda dar a parecer que el funcionamiento esté bien ya que incluso este valor es mejor con el polinomio, cuando se calcula el valor de la SNDR resulta que, en efecto, el funcionamiento con el polinomio es peor. Además, esto se contrasta observando el espectro de salida de ambos casos (Fig. 4.12).

Se observa que el espectro de salida del VCO con polinomio presenta armónicos que es justamente lo que degrada la SNDR, además presenta cierta distorsión alrededor de f_0 . Siendo la SNDR del modelo con VCO ideal de 72.35dB y el del polinomio de 55.3dB .

Las pruebas que se han realizado aquí no son más que una parte de un total de simulaciones de no idealidades que se pueden realizar sobre el sistema; sin embargo, respecto a lo que atañe a los elementos de la biblioteca aquí realizados es más que suficiente para demostrar el uso práctico de la biblioteca. Cabe destacar que las pruebas aquí realizadas también han sido simuladas a nivel de SPICE con el software CADENCE por otros miembros del grupo que participan en el proyecto, obteniéndose resultados similares, lo que indica que, efectivamente, los elementos de la biblioteca funcionan correctamente.

5. CONCLUSIONES Y TRABAJO FUTURO

5.1. Conclusiones

El presente trabajo responde a una necesidad para el desarrollo de convertidores ADC sobremuestreados a nivel de sistema. Se ha empezado por poner en contexto el trabajo, así como estudiar el estado actual de la técnica y se ha verificado que en la literatura ni el mercado actual existe lo que se pretende en los objetivos del proyecto. Posteriormente, en el capítulo 3 se han establecido fundamentos teóricos claves para entender el marco técnico del trabajo, así como presentar los distintos modelos matemáticos de los elementos que conforman la biblioteca finalmente desarrollada, integrador RC y oscilador en anillo. Así mismo, en el mismo capítulo se ha presentado cómo ha sido el desarrollo e implementación de dichos elementos en MATLAB-SIMULINK y se ha comparado con simulaciones en LTspice. La biblioteca desarrollada ha demostrado ser de fácil uso ya que presenta una interfaz bastante amigable y transparente para el usuario. En el capítulo 4 se ha descrito un convertidor ADC sobremuestreado basado en VCO con 'Time Interleaved' utilizando elementos de la biblioteca, y se han realizado una batería de pruebas al sistema poniendo el foco en los elementos de la biblioteca con el fin de realizar una caracterización del ADC para así comparar los distintos resultados obtenidos.

Se han conseguido los objetivos marcados para este trabajo, y a modo de resumen se listan a continuación, una serie de conclusiones:

- Se dispone de unos modelos verificados con herramientas de simulación de circuitos que son útiles para encontrar las especificaciones a tener en cuenta en el diseño detallado de distintos bloques, como son, ganancia en continua y ancho de banda requeridos en un amplificador que forma parte de la arquitectura ADC. Estas especificaciones pueden encontrarse en una fase temprana de diseño al encontrarse modeladas de forma comportamental.
- El desarrollo parametrizado del modelo de VCO basado en oscilador en anillo ha permitido la selección de arquitecturas candidatas a implementar un determinado ADC, permitiendo confrontar distintos aspectos del diseño, sin necesidad de desa-

CAPÍTULO 5. 5.2. TRABAJO FUTURO

rollar modelos específicos para cada arquitectura.

- La biblioteca desarrollada puede ser útil en cualquier diseño futuro de ADCs sobremuestreados.

5.2. Trabajo futuro

Un convertidor ADC sobremuestreado presenta varios bloques a nivel de sistema que son objeto de ser modelados ya que se pueden estudiar distintos niveles de profundidad (no idealidades) en éstos. Por este motivo no sólo se puede ampliar el número de elementos de la biblioteca, si no que además se puede añadir más complejidad a los elementos ya presentes, integradores RC y osciladores en anillo. A continuación, se listan una serie de posibles mejoras:

- Modificar el elemento ‘osciladores en anillo’ para que incluya el estudio de mismatch entre los distintos inversores.
- Incluir un nuevo elemento que representara un DAC, que además de plantear distintas formas de pulso, describiera el estudio de la no linealidad y mismatch presentes en este.
- Incluir un nuevo elemento que representara un reloj con jitter. Un reloj real siempre presenta jitter, esto es, que el período no es constante y presenta un tipo concreto de error[referencia]. Por lo que tener un elemento que represente este fenómeno es conveniente.
- Incluir un nuevo elemento que represente registros con metastabilidad. En el caso de un transistor, la metastabilidad representa un estado no definido, que no es ni ‘0’ ni ‘1’, lo que conlleva a que el transistor tarde en tomar una decisión. Esta posible metastabilidad produce retrasos en la salida de los datos por lo que es muy importante tenerlo en cuenta a la hora de diseñar el convertidor.

6. ENTORNO SOCIOECONÓMICO

Los convertidores ADC están presentes, como mínimo, en cualquier entorno donde haya un sensor, o se necesite tratar digitalmente señales del mundo físico. Además, últimamente con el continuo crecimiento de las denominadas redes de sensores, el número de convertidores ADC crece también. Así como su uso en sistemas de comunicación. [32]

Concretamente, este proyecto, se engloba dentro del ámbito I+D en el campo de la microelectrónica, clave para empresas del sector. Básicamente, si no se invierte en I+D se quedan desactualizados y por lo tanto fuera del mercado. Intel es la empresa microelectrónica que más ingresos tiene, y esto se debe en gran parte a que invierte bastante en I+D ya que es consciente de la importancia de la inversión en investigación. Aunque su negocio principal son los microprocesadores, también está presente en dispositivos portátiles que necesitan integrar circuitos adicionales de comunicación.

Los ADC en comunicaciones son clave para poder obtener velocidades mayores de WiFi, que presenten un mayor ancho de banda para los nuevos retos que plantea el mundo del *Internet of Things*, como puede ser el vídeo en *streaming*.

La biblioteca desarrollada puede ser usada en el desarrollo a nivel de sistema de los casos previamente mencionados ya que no está ligada a ninguna tecnología de fabricación en concreto.

7. PRESUPUESTO Y PLANIFICACIÓN

7.1. Presupuesto

A continuación, se muestra un breve resumen económico del proyecto, que se ha dividido en dos partes: coste del personal que está asociado al número de horas para el desarrollo del trabajo; coste de licencias, coste del software que se ha utilizado para el desarrollo.

7.1.1. Coste del personal

Código	Ud.	Resumen	Medición	Precio unitario	Precio total
1		CAPÍTULO I: COSTE DEL PERSONAL			
01	h	Modelado e implementación	200	30.00 €	6000 €
02	h	Redacción del documento	100	30.00 €	3000 €
SUBTOTAL					9000 €

TABLA 7.1. TABLA COSTE ASOCIADO AL PERSONAL.

7.1.2. Coste de licencias

Código	Ud.	Resumen	Medición	Precio unitario	Precio total
1		CAPÍTULO II: COSTE DE LAS LICENCIAS			
01	ud.	MATLAB Licencia anual	1	800.00 €	800 €
02	ud.	SIMULINK Licencia anual	1	1200.00 €	1200 €
SUBTOTAL					2000 €

TABLA 7.2. TABLA COSTE ASOCIADO AL USO DE LICENCIAS.

HERRAMIENTA DE MODELADO Y SIMULACIÓN DE CONVERTIDORES SOBREMUESTREADOS EN MATLAB/SIMULINK

7.1.3. Presupuesto total

Código	Ud.	Resumen	Medición	Precio unitario	Precio total
01	Capítulo	CAPÍTULO I: Coste del personal	1	9000.00 €	9000 €
02	Capítulo	CAPÍTULO II: Coste de licencias	1	2000.00 €	2000 €
TOTAL					11000 €

TABLA 7.3. TABLA COSTE ASOCIADO AL PERSONAL.

7.2. Planificación



Fig. 7.1. Diagrama de Gantt

BIBLIOGRAFÍA

- [1] K. S. Kundert, “Design of mixed-signal systems on chip,” *The Designer’s Guide Community*, 2006. Disponible: <http://designers-guide.org/Design/MS-SOC.pdf>, último acceso 1-septiembre-2018.
- [2] H. G. Bakeer, O. Shaheen, H. M. Eissa, and M. Dessouky, “Analog, digital and mixed-signal design flows,” in *2007 2nd International Design and Test Workshop*, pp. 247–252, Dec 2007.
- [3] IC-cracker, “Ic-cracker,” *IC-cracker*, 2018. Disponible: <http://www.ic-cracker.com/decode-chip-mitsubishi-m30620ecfp/>, último acceso 1-septiembre-2018.
- [4] R. van de Plassche, *CMOS Integrated Analog-to-Digital and Digital-to-Analog Converters*. Kluwer international series in engineering and computer science, 2003.
- [5] S. Pavan, R. Schreier, and G. C. Temes, *Understanding Delta-Sigma Data Converters*, ch. Appendix B The Delta Sigma Toolbox, pp. 499–537. Piscataway: Wiley, 2017.
- [6] M. Z. Straayer and M. H. Perrott, “A 12-bit, 10-mhz bandwidth, continuous-time $\sigma\delta$ adc with a 5-bit, 950-ms/s vco-based quantizer,” *IEEE Journal of Solid-State Circuits*, vol. 43, pp. 805–814, April 2008.
- [7] E. Gutierrez, L. Hernandez, F. Cardes, and P. Rombouts, “A pulse frequency modulation interpretation of vcoss enabling vco-adc architectures with extended noise shaping,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 65, pp. 444–457, Feb 2018.
- [8] E. Mollick, “Establishing moore’s law,” *IEEE Annals of the History of Computing*, vol. 28, pp. 62–75, July 2006.
- [9] C. A. Mack, “Fifty years of moore’s law,” *IEEE Transactions on Semiconductor Manufacturing*, vol. 24, pp. 202–207, May 2011.
- [10] K. Kundert, H. Chang, D. Jefferies, G. Lamant, E. Malavasi, and F. Sendig, “De-

- sign of Mixed-Signal Systems-on-a-Chip,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 19, pp. 1561–1571, Dec 2000.
- [11] F. Maloberti, *Understanding Microelectronics: A Top-Down Approach*. Wiley, 2012.
- [12] R. Schreier, “The delta sigma toolbox,” *MATLAB Central*, 2016. Disponible: <https://es.mathworks.com/matlabcentral/fileexchange/19-delta-sigma-toolbox>.
- [13] IMSE, “Simsides,” *IMSE-CNM*, 2018. Disponible: <http://www2.imse-cnm.csic.es/simsides/download.php>, último acceso 1-septiembre-2018.
- [14] ULM, “Uni ulm sigma-delta synthesis tool,” *Ulm*, 2018. Disponible: <https://www.sigma-delta.de/index.html>, último acceso 1-septiembre-2018.
- [15] T. C. Carusone, D. Johns, and K. Martin, *Analog Integrated Circuit Design (2nd Edition)*. Wiley, 2013.
- [16] R. Schreier and B. Zhang, “Delta-sigma modulators employing continuous-time circuitry,” *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 43, pp. 324–332, April 1996.
- [17] K. Lee, Y. Yoon, and N. Sun, “A scaling-friendly low-power small-area $\delta\sigma$ adc with vco-based integrator and intrinsic mismatch shaping capability,” *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 5, pp. 561–573, Dec 2015.
- [18] M. Z. Straayer and M. H. Perrott, “A multi-path gated ring oscillator tdc with first-order noise shaping,” *IEEE Journal of Solid-State Circuits*, vol. 44, pp. 1089–1098, April 2009.
- [19] M. Park and M. H. Perrott, “A 78 db snr 87 mw 20 mhz bandwidth continuous-time $\delta\sigma$ adc with vco-based integrator and quantizer implemented in 0.13 μ m cmos,” *IEEE Journal of Solid-State Circuits*, vol. 44, pp. 3344–3358, Dec 2009.
- [20] C. Wulff and T. Ytterdal, “Resonators in open-loop sigma–delta modulators,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 56, pp. 2159–2172, Oct 2009.
- [21] J. Kim, T. Jang, Y. Yoon, and S. Cho, “Analysis and design of voltage-controlled oscillator based analog-to-digital converter,” *IEEE Transactions on Circuits and Sys-*

CAPÍTULO 7. BIBLIOGRAFÍA

- tems I: Regular Papers*, vol. 57, pp. 18–30, Jan 2010.
- [22] R. Naiknaware and T. Fiez, “Time-referenced single-path multi-bit $\Delta\Sigma$ adc using a vco based quantizer,” in *ISCAS’99. Proceedings of the 1999 IEEE International Symposium on Circuits and Systems VLSI (Cat. No.99CH36349)*, vol. 2, pp. 33–36 vol.2, May 1999.
- [23] A. Iwata, N. Sakimura, M. Nagata, and T. Morie, “The architecture of delta sigma analog-to-digital converters using a voltage-controlled oscillator as a multibit quantizer,” *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 46, pp. 941–945, July 1999.
- [24] F. Cardes, E. Gutierrez, A. Quintero, C. Buffa, A. Wiesbauer, and L. Hernandez, “0.04-mm² 103-dB-a dynamic range second-order vco-based audio $\Delta\Sigma$ adc in 0.13- μ m cmos,” *IEEE Journal of Solid-State Circuits*, vol. 53, pp. 1731–1742, June 2018.
- [25] E. Gutierrez, L. Hernandez, U. Gaier, S. Walter, and L. Zou, “A low power and low distortion vco based adc using a pulse frequency modulator,” in *2014 IEEE 57th International Midwest Symposium on Circuits and Systems (MWSCAS)*, pp. 487–490, Aug 2014.
- [26] J. A. Cherry and W. M. Snelgrove, *Continuous-time delta-sigma modulators for high-speed A-D conversion : Theory, practice and fundamental performance limits*. Kluwer international series in engineering and computer science, 2000.
- [27] C. P. Basso, *Linear Circuit Transfer Function: An Introduction to Fast Analytical Techniques*. Wiley-IEEE Press, 2016.
- [28] K. Reddy, S. Rao, R. Inti, B. Young, A. Elshazly, M. Talegaonkar, and P. K. Hanumolu, “A 16-mw 78-dB SNDR 10-MHz BW CT $\Delta\Sigma$ adc using residue-cancelling vco-based quantizer,” *IEEE Journal of Solid-State Circuits*, vol. 47, pp. 2916–2927, Dec 2012.
- [29] Y. Yoon, J. Kim, T. Jang, and S. Cho, “A time-based bandpass adc using time-interleaved voltage-controlled oscillators,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 55, pp. 3571–3581, Dec 2008.
- [30] R. Si, F. Li, and C. Zhang, “A 100MHz/s, 7 bit vco-based adc which is used in time interleaved adc architectures,” in *2012 2nd International Conference on Consumer Electronics, Communications and Networks (CECNet)*, pp. 4–7, April 2012.

- [31] A. Quintero, C. Perez, E. Gutierrez, L. Hernandez, and S. Paton, “Noise-shaping time-interleaved adc based on a single ring oscillator and a sampling array,” in *2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)*, pp. 683–686, Aug 2017.
- [32] J. Hauptmann, D. Giotta, and U. Gaier, “Analog design trends in communication systems,” in *Proceedings of the 2013 9th Conference on Ph.D. Research in Micro-electronics and Electronics (PRIME)*, pp. 3–6, June 2013.