

TESIS DOCTORAL

ESTRUCTURAS AVANZADAS DE CONVERTIDORES
ANALÓGICO-DIGITAL: METODOLOGÍAS DE DISEÑO,
SIMULACIÓN Y ENSEÑANZA

Camilo Quintáns Graña, Ingeniero Industrial

Escuela Técnica Superior de Ingeniería Industrial

Departamento de Ingeniería Eléctrica, Electrónica y de Control

Universidad Nacional de Educación a Distancia

Madrid, 2008

DEPARTAMENTO DE INGENIERÍA ELÉCTRICA,
ELECTRÓNICA Y DE CONTROL

Escuela Técnica Superior de Ingeniería Industrial
Universidad Nacional de Educación a Distancia

Estructuras Avanzadas de Conversión Analógico-Digital:
Metodologías de Diseño, Simulación y Enseñanza

Autor: D. Camilo Quintáns Graña, Ingeniero Industrial

Director: Dr. D. Manuel Alonso Castro Gil

Madrid, 2008

Agradecimientos

Siento la necesidad de agradecer a los demás el haber llegado hasta aquí, porque nadie puede desarrollarse como persona y como profesional en solitario. Siempre se necesita de los demás, especialmente del grupo, porque sólo compartiendo ilusiones encontramos fuerzas para seguir adelante. Es habitual, cuando se pretende agradecer, comenzar por aquellas personas que más te han ayudado y que mejor se han portado contigo, y es cierto que debe ser así. Sin embargo, creo que las personas que en ocasiones más pueden motivarte son, precisamente, aquellas que te han hecho daño o han intentado perjudicarte, porque te han hecho reaccionar, defenderte, enfocar la crisis y hacerte más fuerte. Al fin y al cabo ¿Qué es esta vida sino una constante lucha? Es evidente que estas palabras sólo pueden venir de un luchador. Perdonar la inmodestia, pero me considero un luchador y me siento vivo. Por ello, comienzo agradeciendo a todas aquellas personas que me han hecho mejor, aunque haya sido a base de “palos”. Que sepan que siempre agradezco la oportunidad de competir, aunque en ocasiones con menos oportunidades o siguiendo normas injustas. A esas personas quiero recordarles, además, que sigo aquí, trabajando día a día, paso a paso y honestamente.

Sin duda quiero agradecer también, y ahora con mayúsculas, a aquellas personas que de buena fé me han ayudado y a las que les debo, además, mi amistad y mi constante comprensión, porque los que en ocasiones aparentamos fuerza y seguridad somos también vulnerables y sensibles y, sin el apoyo de estos compañeros de viaje, no es posible avanzar.

Gracias a toda mi familia, en especial a mi madre Mercedes y a mi padre Amando, que me enseñó el oficio de electrónico, y que fue el verdadero culpable de que hoy pueda presentar este trabajo. Gracias a mi esposa Ana y a mis hijas Andrea y Silvia, ¡sobran los porqués! Gracias a todos los compañeros de trabajo que he tenido en los 20 años que llevo de profesión. Siempre será un orgullo haber trabajado con ellos.

Por la cercanía en el tiempo, mi agradecimiento por lo que me han enseñado, a los compañeros de la Universidad de Vigo y de la UNED, a los alumnos, al PAS, a los profesores y tutores, y en especial al Director de esta Tesis.

ÍNDICE GENERAL

LISTA DE SÍMBOLOS, ABREVIATURAS Y SIGLAS	XI
ÍNDICE DE FIGURAS.....	XVII
ÍNDICE DE TABLAS	XXV
RESUMEN	XXVII
ABSTRACT.....	XXIX
CAPÍTULO 1. INTRODUCCIÓN.....	1
1.1. Antecedentes y objetivos de la presente tesis	1
1.2. Organización del contenido de esta tesis	2
CAPÍTULO 2. LOS CIRCUITOS CONVERTIDORES A/D	5
2.1. Introducción	5
2.2. Clasificación de los convertidores A/D.....	8
2.3. Estado actual de los convertidores A/D	10
2.4. Traza de un convertidor ideal A/D.....	12
2.5. Error de cuantificación, relación señal ruido y bits efectivos	14
2.6. Especificación de las características estáticas de los convertidores A/D	17
2.7. Especificación de las características dinámicas de los convertidores A/D	21
2.8. Normativa y métodos de ensayo para los circuitos ADC.....	30
2.8.1. Normativa	30
2.8.2. Ensayo y caracterización de los circuitos ADC	31
2.8.2.1. Introducción.....	31
2.8.2.2. Configuración del escenario para realizar los ensayos	34
2.8.2.3. Métodos de ensayo en el dominio del tiempo.....	35
2.8.2.4. Método de ensayo en el dominio de la frecuencia.....	37
2.9. Metodología para seleccionar los circuitos ADC según las características dinámicas de la señal	38

2.9.1. Introducción	38
2.9.2. Problemática.....	39
2.9.3. Deducción del método de selección	40
2.9.4. Aplicación a la selección del número de bits del ADC	42
2.9.5. Resultados de simulación y experimentales.....	49
CAPÍTULO 3. LOS CIRCUITOS CONVERTIDORES A/D SIGMA-DELTA	53
3.1. Introducción	53
3.2. Origen y principio de funcionamiento del modulador sigma-delta.....	55
3.3. Sobremuestreo y conformado del ruido	57
3.4. Modulador sigma-delta de segundo orden	64
3.5. Deducción de la ganancia de resolución mediante técnicas de sobremuestreo	65
3.6. Etapas de filtrado y diezmado en los convertidores sigma-delta.....	69
3.6.1. Introducción	69
3.6.2. Filtrado y diezmado con filtro de promediado	70
3.6.3. Procedimiento para ecualizar la respuesta del filtro de promediado	74
3.6.4. Ejemplo de obtención de un filtro FIR por muestreo en frecuencia.....	75
3.6.5. Ecualización de la respuesta del convertidor sigma-delta con el filtro FIR	78
3.7. Simulación de convertidores sigma-delta.....	79
3.7.1. Simulación de un circuito modulador sigma-delta con OrCAD PSpice	79
3.7.2. Simulación de un convertidor A/D sigma-delta	80
3.8. Variantes en las estructuras de los convertidores A/D sigma-delta.....	85
3.8.1. El modulador sigma-delta con estructura paralela generalizada	87
3.8.2. El modulador sigma-delta multietapa.....	88
CAPÍTULO 4. LOS CIRCUITOS CONVERTIDORES A/D PIPELINE	91
4.1. Introducción	91
4.2. Antecedentes de los convertidores A/D <i>pipeline</i>	93
4.3. Principio de funcionamiento de los convertidores A/D <i>pipeline</i>	96
4.4. Diseño y simulación de un convertidor A/D <i>pipeline</i> mediante OrCAD PSpice	99
4.4.1. Diseño del convertidor A/D <i>pipeline</i> de 1 bit por etapa.....	99

4.4.2. Implementación del convertidor A/D <i>pipeline</i> de 1 bit por etapa	101
4.4.3. Simulación del convertidor <i>pipeline</i> de 1 bit por etapa.....	103
4.4.4. Caracterización de la dinámica del convertidor A/D <i>pipeline</i>	104
4.5. Variantes en la estructura de un convertidor A/D <i>pipeline</i>	107
CAPÍTULO 5. METODOLOGÍAS ALTERNATIVAS DE DISEÑO DE CIRCUITOS ADC	
PIPELINE Y SIGMA-DELTA.....	111
5.1. Introducción	111
5.2. Nueva metodología de diseño de convertidores multietapa.....	111
5.2.1. Estructura y modo de funcionamiento del ADC multietapa	111
5.2.2. Diseño y simulación de un ADC de 5 etapas.....	113
5.3. Nueva metodología de diseño de convertidores sigma-delta	116
5.3.1. Introducción	116
5.3.2. Diseño alternativo del modulador sigma-delta	117
5.3.3. Análisis del modelo del modulador de primer orden.....	118
5.3.4. Simulación mediante Matlab de los modelos discretos de los moduladores sigma- delta	123
5.3.5. Simulación mediante OrCAD PSpice de los modelos continuos de los moduladores sigma-delta.....	128
5.3.6. Evaluación del principio de funcionamiento de los nuevos modelos de moduladores	131
5.3.7. Construcción de un prototipo y prueba del modulador sigma-delta de segundo orden ...	133
5.3.8. Adquisición y procesado digital de la señal de salida del modulador.....	139
5.3.9. Resultados prácticos del modulador después del filtrado	144
CAPÍTULO 6. METODOLOGÍA DE ENSEÑANZA DE LOS CIRCUITOS ADC PIPELINE Y	
SIGMA-DELTA	147
6.1. Introducción	147
6.2. Objetivos docentes y metodología general.....	148
6.3. Recursos que se utilizan.....	150
6.4. Prerrequisitos docentes	151
6.5. Metodología para la enseñanza de los convertidores A/D <i>pipeline</i>	151

ÍNDICE GENERAL

6.6. Metodología para la enseñanza de los convertidores A/D sigma-delta.....	153
6.7. Metodología de evaluación	156
6.8. Resultados de la aplicación de la metodología docente	156
CAPÍTULO 7. RESULTADOS OBTENIDOS.....	161
7.1. Resultados sobre el estado del arte de los circuitos ADC	161
7.2. Resultados del método de selección de los circuitos ADC	162
7.3. Resultados de la metodología de diseño de convertidores sigma-delta y <i>pipeline</i>	162
7.4. Resultados de la aplicación de la metodología docente para la enseñanza de los convertidores A/D <i>pipeline</i> y sigma-delta.....	165
CAPÍTULO 8. CONCLUSIONES FINALES.....	167
8.1. Aportaciones de esta tesis.....	167
8.2. Sugerencias para futuros estudios y desarrollos.....	169
BIBLIOGRAFÍA.....	171
ANEXO I: CONTENIDO DEL CDROM.....	185

LISTA DE SÍMBOLOS, ABREVIATURAS Y SIGLAS

ac	<i>Alternating Current</i> (Corriente alterna).
A/D	Analógico a Digital.
ADC	<i>Analog to Digital Converter</i> (Convertidor analógico a digital).
ADM	<i>Analog to Digital Module</i> (Módulo analógico a digital).
ASIC	<i>Application Specific Integrated Circuit</i> (Circuito integrado de aplicación específica).
BNC	<i>Bayonet Neil Concelman</i> (Conector de Concelman de tipo bayoneta).
BCD	<i>Binary Coded Decimal</i> (Decimal codificado en binario).
CFCS	<i>Commutated Feedback Capacitor Switching</i> (Circuitos de capacidades conmutadas realimentados).
CMOS	<i>Complementary Metal Oxide Semiconductor</i> (Semiconductores complementarios, tipos N y P, de óxidos de metal).
D	Factor de diezmado
DAC	<i>Digital to Analog Converter</i> (Convertidor de analógico a digital).
DAS	<i>Data Acquisition System</i> (Sistema de adquisición de datos).
dc	<i>Direct Current</i> (Corriente directa).
DDS	<i>Direct Digital Synthesis</i> (Síntesis digital directa).
DFS	<i>Data Format Select</i> (Selección del formato de datos).
DFT	<i>Discrete Fourier Transform</i> (Transformada discreta de Fourier).
DIEEC	Departamento de Ingeniería Eléctrica Electrónica y de Control.
DIP	<i>Dual Inline Package</i> (Encapsulado de doble línea).
DNL	<i>Differential Non-Linearity</i> (No linealidad diferencial).
DT	Director de la Tesis.
DSO	<i>Digital Signal Oscilloscope</i> (Osciloscopio digital de señales).
DSP	<i>Digital Signal Processor</i> (Procesador digital de señales).
DVM	<i>Digital VoltMeter</i> (Voltímetro digital).
EEES	Espacio Europeo de la Educación Superior.
ENOB	<i>Effective Number of Bits</i> (Número efectivo de bits).

LISTA DE SÍMBOLOS, ABREVIATURAS Y SIGLAS

ESR	<i>Effective Series Resistance</i> (Resistencia serie equivalente).
ETSII	Escuela Técnica Superior de Ingenieros Industriales.
fco	Frecuencia más elevada a la que la respuesta en amplitud es de -3 dB.
fd	Frecuencia de muestreo de un registro después del proceso de diezmado.
FET	<i>Field Effect Transistor</i> (Transistor de efecto de campo).
fi	Frecuencia de la señal de entrada.
FIT	<i>Failures In Time</i> (Fallos en tiempo).
FIR	<i>Finite Impulse Response</i> (Respuesta impulsional finita)
FFT	<i>Fast Fourier Transform</i> (Transformada rápida de Fourier).
fopt	Frecuencia de entrada optima para las pruebas.
FPBW	<i>Full Power BandWidth</i> (Ancho de banda total de potencia).
FPGA	<i>Field Programmable Gate Array</i> (Campo de matriz de puertas programables).
fr	<i>Input signal reference frequency</i> o <i>input signal repetition rate</i> (Frecuencia de la señal de referencia de entrada).
fs	<i>Sampling frequency</i> (Frecuencia de muestreo).
FS	<i>Full Scale</i> (Escala o rango total).
GSPS	<i>Giga Sample Per Second</i> (Giga muestra por segundo).
H(f)	Respuesta en frecuencia de un ADC bajo prueba.
H(fk)	Transformada discreta de Fourier de la secuencia h(n).
HMI	<i>Human Machine Interface</i> (Interfaz hombre-máquina).
IC	<i>Integrated Circuit</i> (Circuito integrado).
IIR	<i>Infinite Impulse Response</i> (Respuesta impulsional infinita)
IMD	<i>Intermodulation Distortion</i> (Distorsión de intermodulación).
INL	<i>Integral Non Linearity</i> (No linealidad integral).
IRE	<i>The Institute of Radio Engineers, Inc.</i>
JTAG	<i>Joint Test Action Group</i> (es la denominación que se le suele dar al estándar IEEE 1149.1, que se denomina: <i>Standard Test Access Port and Boundary-Scan Architecture</i>).
LDO	<i>Low Drop Out</i> (Baja caída de salida).

k	En el ámbito de los circuitos convertidores A/D sigma-delta se suele utilizar para indicar el orden del filtro de la etapa de salida.
kb	<i>Kilo bit.</i>
L	Orden del modulador sigma-delta (también se suele denominar por la letra <i>p</i>).
LED	<i>Light Emission Diode</i> (Diodo de emisión de luz).
LSB	<i>Least Significant Bit</i> (Bit menos significativo o de menor peso).
LQFP	<i>48-Lead Quad Flatpack Plastic</i> (Encapsulado plano de plástico de 48 terminales).
M	Número de muestras en un registro.
MASH	<i>Multi-stage noise-Shaping</i> (Conformador de ruido multietapa).
MCS	<i>Merged-Capacitor Switching</i> (Interruptor de capacitor).
Md	Número de muestras en un registro después del proceso de diezmado.
MDAC	<i>Multiplying ADC</i> (ADC multiplicador).
MIF	<i>Memory Initializing File</i> (Fichero para inicialización de la memoria).
MOSFET	<i>Metal Oxide Semiconductor Field Effect Transistor</i> (Transistor de efecto de campo de oxido de metal).
MSB	<i>More Significant Bit</i> (Bit más significativo).
MSPS	<i>Mega Samples Per Second</i> (Millones de muestras por segundo).
N	Número de bits.
NBW	<i>Noise Band Width</i> (Ancho de banda del ruido).
NTF	<i>Noise Transfer Function</i> (Función de transferencia del ruido).
Offset	Decalaje o desviación del valor de continua de la señal.
OSR	<i>Over Sampling Ratio</i> (Relación de sobremuestreo).
OTR	<i>Out of Range signal</i> (Señal fuera de rango).
p	Orden del modulador sigma-delta (también se suele denominar por la letra <i>L</i>).
PCM	<i>Pulse Code Modulator</i> (Modulador código pulso)
PSD	<i>Power Spectral Density</i> (Densidad de potencia espectral).
ppm/°C	partes por millón por grado centígrado.
Q	Es el rango analógico total de un convertidor dividido por el número de códigos posibles.
RAM	<i>Random Access Memory</i> (Memoria de acceso aleatorio).

LISTA DE SÍMBOLOS, ABREVIATURAS Y SIGLAS

ROM	<i>Read Only Memory</i> (Memoria de sólo lectura).
rms	<i>Root Mean Square</i> (Raíz cuadrática media).
SAD	Sistema de Adquisición de Datos.
SAR	<i>Successive Approximation Register</i> (Registro de aproximaciones sucesivas).
$\Sigma\Delta$	Sigma-delta.
SC	<i>Switched-Capacitor</i> (Capacitor conmutado).
SOT	<i>Small outline transistor</i> (Transistor de perfil bajo).
SFDR	<i>Spurious Free Dynamic Range</i> (Rango dinámico libre de espurios).
SINAD	<i>Signal to Noise And Distortion</i> (Relación señal-ruido más distorsión).
SR	<i>Slew Rate</i> (Velocidad de cambio).
SNR	<i>Signal to Noise Ratio</i> (Relación señal ruido).
S/N+D	<i>Signal to Noise Plus Distortion</i> (Relación señal-ruido más distorsión).
SMD	<i>Surface Mounted Device</i> (Dispositivo de montaje superficial).
SQNR	<i>Signal-to-Quantization-Noise Ratio</i> (Relación señal-ruido de cuantificación).
STF	<i>Signal Transfer Function</i> (Función de transferencia de la señal).
S&H	<i>Sample and Hold</i> (Muestreo y retención).
THD	<i>Total Harmonic Distortion</i> (Distorsión armónica total).
TPQF	<i>Thin Plastic Quad Flatpack</i> (Encapsulado plano cuadrado de plástico de perfil bajo).
T&H	<i>Track and Hold</i> (Sigue y retiene).
TTL	<i>Transistor-Transistor Logic</i> (Lógica transistor-transistor).
UNED	Universidad Nacional de Educación a Distancia.
USB	<i>Universal Serial Bus</i> (Bus serie universal).
VHDL	<i>VHSIC (Very High Speed IC) Hardware Description Language</i> (Lenguaje de descripción hardware para circuitos integrados de alta velocidad).
Vpp	Voltios Pico a Pico.
X	Valor de la señal de entrada al circuito ADC.
Y[k]	El k-ésimo valor de la transformada discreta de Fourier de la secuencia de salida y[n].
y[n]	El valor enésimo de la secuencia de salida, es equivalente a yn.
ε	Error, se utiliza para indicar el error total o en una banda determinada.

ϵ_{rms}	Error en valor rms.
$\epsilon[k]$	Secuencia de error con los valores correspondientes a la diferencia entre la señal de referencia y la señal convertida.

ÍNDICE DE FIGURAS

Figura 2.1. Interfaz electrónica entre el mundo analógico y el mundo digital.....	5
Figura 2.2. Resolución y velocidad de conversión de los convertidores más utilizados (actualizado en octubre de 2007).	12
Figura 2.3. Funciones de transferencia de un convertidor: a) modo de entrada unipolar y salida en binario natural, b) modo de entrada bipolar y salida en binario desplazado.....	13
Figura 2.4 Función de transferencia de un convertidor con entrada bipolar y salida en complemento a 2.....	13
Figura 2.5. Cuantificación de la amplitud de una señal analógica, error de cuantificación.	15
Figura 2.6. Modelo de un circuito ADC.....	16
Figura 2.7. Espectro de frecuencia del ruido de cuantificación.	17
Figura 2.8. Función de transferencia de un ADC ideal de 3 bits.....	18
Figura 2.9. Representación del error de no linealidad diferencial de un ADC.	19
Figura 2.10. Curva del error de no linealidad diferencial del ADC AD9288 (cortesía de Analog Devices, Inc.).....	19
Figura 2.11. Representación del error de no linealidad integral en un ADC de 3 bits.....	19
Figura 2.12. Curva del error de no linealidad integral del ADC AD9288.	20
Figura 2.13. Error de offset de un ADC con entrada unipolar.	20
Figura 2.14. Error de ganancia de un convertidor unipolar.....	20
Figura 2.15. Error de offset de un ADC con entrada bipolar.	20
Figura 2.16. Error de ganancia de un ADC con entrada bipolar.	20
Figura 2.17. Distorsión total del ADC AD9280 para diferentes amplitudes de entrada.	22
Figura 2.18. Curva SINAD del convertidor AD9288 en función de la frecuencia de la señal de entrada.	23
Figura 2.19. Rango dinámico libre de espurios del convertidor AD9288.....	24
Figura 2.20. Ejemplo de curva de respuesta en frecuencia del ADC modelo AD9280.	24
Figura 2.21. Productos de intermodulación.....	26
Figura 2.22. Distorsión de intermodulación.....	26

ÍNDICE DE FIGURAS

Figura 2.23. Error debido a la fluctuación en el instante del muestreo.	27
Figura 2.24. Disminución de la SNR debida a la fluctuación del reloj de muestreo.	28
Figura 2.25. Latencia en un ADC.	28
Figura 2.26. Variación de la relación señal ruido del convertidor AD9288 frente al ancho del pulso del reloj de muestreo.	29
Figura 2.27. Variación de la SINAD y de la SFDR frente al ciclo de trabajo del reloj de muestreo.	29
Figura 2.28. Configuración para realizar el test de la dinámica de un ADC.	34
Figura 2.29. Espectro de frecuencia de la salida del convertidor AD9288.	34
Figura 2.30. Configuración para el ensayo de un ADC mediante excitación sinusoidal.	35
Figura 2.31. Ejemplo de obtención del error de conversión a partir de un registro de datos.	36
Figura 2.32. Componentes del espectro de frecuencia de la señal convertida (componentes de la DFS).	37
Figura 2.33. Información de entrada y de salida para seleccionar la frecuencia de muestreo f_s y la resolución N.	41
Figura 2.34. Diagrama de bloques de la fuente de señal (B_1) y de un proceso (B_2).	41
Figura 2.35. Diagrama de los bloques generador y convertidor.	43
Figura 2.36. SNR total frente al número de bits (N) del ADC.	44
Figura 2.37. Mínimo número de bits necesarios frente a la SNR_L que se pierde para una señal de entrada con una cierta SNR_s	46
Figura 2.38. Señal muestreada a una frecuencia demasiado baja ($f_s = 11 \cdot f_i$).	46
Figura 2.39. Señal muestreada a una frecuencia demasiado alta ($f_s = 33 \cdot f_i$).	47
Figura 2.40. Señal muestreada a una frecuencia óptima ($f_s = 22 \cdot f_i$).	47
Figura 2.41. Máxima derivada de tensión de la señal.	48
Figura 2.42. Ejemplo de una señal con 15 dB de SNR muestreada con un ADC de 4 bits.	49
Figura 2.43. Resultado de convertir, con una SNR del ADC fijada en 42 dB, señales con distintas SNR.	50
Figura 2.44. Resultado de muestrear una señal con una SNR de 25 dB para distintos valores de la SNR del convertidor.	50

Figura 2.45. Vista del osciloscopio digital. El canal C1 representa la señal sinusoidal, el canal C2 el ruido que se le añade y el canal C3 la señal resultante de 25 dB de SNR.	51
Figura 3.1. Diagrama de bloques básico de un convertidor de sobremuestreo sigma-delta.....	53
Figura 3.2. Principio de la realimentación negativa del error para aumentar la resolución en los convertidores de sobremuestreo mediante un post-filtrado.	54
Figura 3.3. Modelo del modulador básico.....	55
Figura 3.4. Modelo simplificado del modulador básico.....	55
Figura 3.5. Modulador Delta.....	56
Figura 3.6. Demodulador Delta.....	56
Figura 3.7. Modulador delta más la primera etapa del demodulador.....	57
Figura 3.8. Esquema simplificado del modulador sigma-delta básico.....	57
Figura 3.9. Modelo lineal del modulador sigma-delta.....	57
Figura 3.10. Modelo de ruido aditivo del cuantificador.....	58
Figura 3.11. Espectro del ruido de cuantificación.....	58
Figura 3.12. Ganancia de resolución debida al sobremuestreo.....	60
Figura 3.13. Espectro del ruido del cuantificador a dos frecuencias de conversión.....	60
Figura 3.14. Modelo del cuantificador con filtrado del ruido.....	61
Figura 3.15. Espectro del ruido de cuantificación con sobremuestreo y filtrado.....	62
Figura 3.16. Modelo de tiempo discreto del modulador sigma-delta de primer orden.....	63
Figura 3.17. Modelo detallado de tiempo discreto del modulador sigma-delta de segundo orden.....	64
Figura 3.18. Modelo de tiempo discreto del modulador sigma-delta de segundo orden.....	65
Figura 3.19. Potencia del espectro del ruido en los convertidores de sobremuestreo.....	68
Figura 3.20. Ganancia de resolución en los convertidores de sobremuestreo.....	69
Figura 3.21. Ganancia de la SNR según el factor de sobremuestreo.....	69
Figura 3.22. Diagrama de bloques del filtro de orden $k=1$	72
Figura 3.23. Diagrama de bloques del filtro de orden $k=2$	73
Figura 3.24. Respuesta en frecuencia del filtro sinc con un factor de diezmado $D=32$	74
Figura 3.25. Caída de la respuesta en frecuencia debida al filtro de promediado basado en la función sinc.....	74

ÍNDICE DE FIGURAS

Figura 3.26. Respuesta deseada en frecuencia $H_d(w)$.	76
Figura 3.27. Respuesta impulsional $h(n)$ del filtro ecualizador.	77
Figura 3.28. Respuesta del filtro calculado $H_c(w)$.	77
Figura 3.29. Respuesta en frecuencia del filtro de promediado.	78
Figura 3.30. Diagrama de bloques del modulador sigma-delta de primer orden.	79
Figura 3.31. Esquema del circuito modulador sigma-delta de primer orden.	80
Figura 3.32. Resultado de la simulación del modulador sigma-delta de primer orden.	80
Figura 3.33. Diagrama de bloques del demodulador del ADC sigma-delta.	82
Figura 3.34. Diagrama bloques de la etapa de filtrado y diezmado del convertidor Σ - Δ .	83
Figura 3.35. Simulación del ADC sigma-delta.	84
Figura 3.36. Resultados de simulación del ADC sigma-delta.	84
Figura 3.37. Modulador sigma-delta con estructura paralela generalizada.	87
Figura 3.38. Modulador de segundo orden con estructura paralela generalizada.	87
Figura 3.39. Esquema del modulador sigma-delta multietapa.	88
Figura 3.40. Modelo discreto del modulador sigma-delta multietapa.	88
Figura 4.1. Diagrama de bloques básico de un convertidor semiparalelo.	94
Figura 4.2. Diagrama de tiempos de las señales de sincronismo del convertidor semiparalelo.	94
Figura 4.3. Diagrama de un convertidor de dos etapas pipeline.	95
Figura 4.4. Principio de funcionamiento de un ADC de dos etapas.	95
Figura 4.5. Arquitectura de un ADC de tipo pipeline con cuatro etapas de 3 bits (cada una codifica 2 bits).	96
Figura 4.6. Latencia en los convertidores pipeline.	98
Figura 4.7. Ejemplo de una etapa pipeline de 1 bit.	99
Figura 4.8. Estructura del ADC pipeline simulado.	99
Figura 4.9. Señales de sincronización del ADC pipeline simulado.	100
Figura 4.10. Distribución de las señales de sincronización del ADC pipeline simulado.	101
Figura 4.11. Esquema de una etapa del ADC pipeline simulado.	102
Figura 4.12. Esquema del circuito S&H.	102

Figura 4.13. Esquema electrónico del ADC de tipo pipeline de 5 bits simulado mediante OrCAD PSpice (PP1).....	103
Figura 4.14. Ampliación de una porción del resultado de la simulación del ADC pipeline de 5 bits.	104
Figura 4.15. Componentes de frecuencia de la señal obtenida a la salida del convertidor.	106
Figura 4.16. Señales de referencia, de salida del convertidor y de ruido.....	106
Figura 4.17. Circuito de muestreo y retención de capacidades conmutadas.	109
Figura 5.1. Estructura de una etapa del nuevo convertidor.	112
Figura 5.2. Traza del nuevo convertidor multietapa, a) con comparación con $-Q/2$, b) con comparación con cero.....	113
Figura 5.3. Esquema del nuevo circuito convertidor A/D multietapa (PP2).....	114
Figura 5.4. Circuito de una etapa del nuevo convertidor A/D multietapa.....	114
Figura 5.5. Gráficas del Probe obtenidas de la simulación del nuevo convertidor A/D multietapa....	115
Figura 5.6. Resultado de la conversión de una señal sinusoidal con el nuevo convertidor A/D.....	115
Figura 5.7. Modelo del modulador sigma-delta básico.	117
Figura 5.8. Funciones de transferencia: a) de la señal de entrada, b) del ruido de cuantificación.	117
Figura 5.9. Equivalencias entre las funciones de transferencia de la señal y del ruido.....	118
Figura 5.10. Modulador sigma-delta diseñado a partir de la función de transferencia deseada para el ruido.....	118
Figura 5.11. Circuito integrador.....	119
Figura 5.12. Respuesta en frecuencia de los modelos de primer orden cuyas funciones de transferencia de la señal y del ruido corresponden, respectivamente, con la Tabla 5.3 y la Tabla 5.4.	122
Figura 5.13. Estructura del modulador SD1.....	123
Figura 5.14. Estructura del modulador SD2.....	123
Figura 5.15. Respuesta del modulador de primer orden con la estructura clásica (SD1).....	123
Figura 5.16. Respuesta del modulador de primer orden con la nueva estructura (SD2).....	124
Figura 5.17. Estructura del modulador clásico de segundo orden denominado SD3.....	124
Figura 5.18. Estructura del modulador de segundo orden SD4.....	125
Figura 5.19. Respuesta del modulador de segundo orden con la estructura clásica (SD3).	126

ÍNDICE DE FIGURAS

Figura 5.20. Respuesta del modulador de segundo orden con la nueva estructura (SD4).....	126
Figura 5.21. Circuito del modulador de primer orden.....	128
Figura 5.22. FFT de la salida del modulador sigma-delta de primer orden simulado con PSpice.	129
Figura 5.23. Circuito del modulador sigma-delta de segundo orden.....	129
Figura 5.24. Circuito del modulador sigma-delta de tercer orden.....	130
Figura 5.25. Respuesta en frecuencia del modulador de segundo orden con una señal de entrada de 100 Hz.....	130
Figura 5.26. Respuesta en frecuencia del modulador de tercer orden con una señal de entrada de 100 Hz.....	130
Figura 5.27. Circuito que implementa las funciones NTF y STF del modulador de segundo orden. .	131
Figura 5.28. Respuesta en frecuencia de las funciones NTF y STF del modulador de segundo orden.....	132
Figura 5.29. Circuito que implementa las funciones NTF y STF del modulador de cuarto orden.....	132
Figura 5.30. Respuesta en frecuencia de las funciones NTF y STF del modulador de cuarto orden. .	132
Figura 5.31. Fotografía del prototipo del modulador sigma-delta.....	133
Figura 5.32. Oscilograma del test de la frecuencia de corte del circuito que implementa la función NTF.....	135
Figura 5.33. Oscilograma del test de la frecuencia de corte del circuito que implementa la función STF.....	135
Figura 5.34. Respuesta en módulo de los circuitos de la NTF y de la STF.....	136
Figura 5.35. Respuesta en fase de los circuitos de la NTF y de la STF.....	136
Figura 5.36. Respuesta transitoria del circuito comparador LT1016 para una entrada de 2 MHz.	137
Figura 5.37. Respuesta del circuito multiplexor AD8044 con una carga de 2 k Ω	138
Figura 5.38. Sinusoidal de entrada de 10 kHz y salida del modulador.....	138
Figura 5.39. Salida del modulador de segundo orden con una señal de entrada sinusoidal de 70 kHz y 4 500 mV muestreada a 3 MHz.....	139
Figura 5.40. Fotografía del prototipo del convertidor sigma-delta con el modulador de segundo orden.....	140
Figura 5.42. Esquema del diseño de la FPGA del prototipo de convertidor sigma-delta.....	140

Figura 5.41. Panel de control del programa Signal Tap para adquirir señales de la FPGA.	141
Figura 5.43. Respuesta del modulador ($f_i = 700$ Hz).	143
Figura 5.44. Respuesta del modulador ($f_i = 7$ kHz).	143
Figura 5.45. Respuesta del modulador ($f_i = 70$ kHz).	143
Figura 5.46. Respuesta en frecuencia del convertidor Sigma Delta ($f_s = 6$ MHz, $f_c = 30$ kHz, $f_i = 10$ kHz).	144
Figura 5.47. Respuesta en frecuencia del convertidor sigma-delta ($f_s = 6$ MHz, $f_c = 300$ kHz, $f_i = 200$ kHz).	145
Figura 6.1. Evolución de los convertidores paralelos a los semiparalelos, a los pipeline de dos etapas y, finalmente, a los multietapa.	153
Figura 6.2. Esquema simplificado de los pasos para obtener la ficha técnica del convertidor a partir del análisis de los resultados de la simulación en el dominio del tiempo (transitorio) y de la frecuencia.	153
Figura 6.3. Simulación del modulador de primer orden mediante un algoritmo programado en lenguaje C y mediante un circuito electrónico simulado con OrCAD PSpice, en este caso se muestran los resultados de la simulación transitoria mediante el programa Probe.	155
Figura 6.4. Esquema de la etapa de filtrado y diezmado del convertidor sigma-delta. La simulación se realiza con el software Quartus II y los datos de entrada se obtienen de la ROM, cuyo contenido se inicializa con los datos obtenidos de la simulación del modulador que se describió con el algoritmo programado en C.	155
Figura 6.5. Los resultados de simular el convertidor se pasan a la hoja de cálculo y se obtienen las características más importantes comparando la señal de salida con la sinusoidal que mejor se ajusta.	156
Figura 7.1. Resolución y frecuencia de conversión de los convertidores más utilizados.	161
Figura 7.2. SNR total frente al número de bits (N) del ADC.	162
Figura 7.3. Estructura del modulador sigma-delta indicando las distintas funciones de transferencia que lo caracterizan.	163

ÍNDICE DE FIGURAS

Figura 7.4. Respuesta en frecuencia de los modelos de primer orden de los moduladores sigma-delta, que corresponden a la estructura clásica (STF1 y NTF1) y a la nueva estructura (STF2 y NTF2).	163
Figura 7.5. Respuesta en frecuencia del convertidor sigma-delta ($f_s= 6$ MHz, $f_c= 300$ kHz, $f_i= 200$ kHz).	163
Figura 7.6. Alternativas presentadas para las etapas de los convertidores pipeline, a) estructura con amplificación del residuo (PP1), b) nueva estructura con atenuación de la referencia (PP2).	164

ÍNDICE DE TABLAS

Tabla 2.1. Clasificación de los ADC según la velocidad de conversión.....	8
Tabla 2.2. Cronología de los principales avances en los circuitos convertidores A/D.....	9
Tabla 2.3. Comparativa entre los distintos tipos de ADC.....	10
Tabla 2.4. Clasificación de los convertidores según el formato de los datos digitales.	14
Tabla 2.5. Resolución y rango dinámico de los circuitos ADC.	16
Tabla 2.6. Términos utilizados para el análisis de la relación señal ruido.	43
Tabla 3.1. Comparación entre los dos métodos para obtener la ganancia de resolución.	68
Tabla 3.2. Código de la función FIRFS.	76
Tabla 3.3. Procedimiento de Matlab que calcula la respuesta en frecuencia.	77
Tabla 3.4. Simulación del modulador sigma-delta mediante un algoritmo programado en C.....	81
Tabla 3.5. Porción del contenido de la ROM con los datos de salida del modulador.....	82
Tabla 3.6. Diseño en VHDL del filtro de diezmado para implementar en la FPGA.....	83
Tabla 4.1. Funcionamiento de un ADC pipeline de 6 etapas.	98
Tabla 4.2. Recursos utilizados por los ADC pipeline de N bits de 1-Bit por etapa.	103
Tabla 4.3. Listado del programa de Matlab para el test del ADC pipeline de 5 bits.....	105
Tabla 4.4. Función para el cálculo de la serie de Fourier de una secuencia.....	106
Tabla 4.5. Resumen de las características del ADC pipeline simulado (fs= 200 kHz).....	107
Tabla 5.1. Resultados del test del nuevo convertidor multietapa.	116
Tabla 5.2. Valores de la ganancia g en función de la frecuencia de corte del modulador sigma- delta.	121
Tabla 5.3. Funciones de transferencia de las señales correspondientes a los modelos de los moduladores de primer orden con valor del coeficiente g igual a la unidad.	122
Tabla 5.4. Funciones de transferencia de los ruidos correspondientes a los modelos de los moduladores de primer orden con valor del coeficiente g igual a la unidad.	122
Tabla 5.5. Código de Matlab que simula los moduladores de primero y de segundo orden.....	127
Tabla 5.6. Código VHDL del divisor de frecuencia.....	141
Tabla 5.7. Código de Matlab que procesa la señal de salida del modulador.....	142

ÍNDICE DE TABLAS

Tabla 5.8. Función que calcula el módulo en dB de la respuesta en frecuencia de una secuencia.....	142
Tabla 5.9. Resultados de la caracterización de la dinámica del modulador.	144
Tabla 6.1. Recursos que se utilizan en función de las tareas y objetivos.	151
Tabla 6.2. Cuestiones sobre el uso de los simuladores de circuitos electrónicos.	158
Tabla 6.3. Cuestiones sobre la opinión de los alumnos a cerca de la importancia del tema.	158
Tabla 6.4. Cuestiones sobre el uso de la normativa.....	158
Tabla 6.5. Resultados del cuestionario, sobre conocimientos generales de convertidores A/D, realizados antes y después del curso. (-: respuesta en blanco, x: incorrecta, v: correcta). Se presenta la reducción en % de las respuestas totales en blanco e incorrectas. $\% = \{1 - [(-+X)_{\text{antes}} / (-+X)_{\text{después}}]\} \times 100$	159
Tabla 6.6. Resultados del cuestionario, sobre conocimientos de la etapas de filtrado y diezmado, realizados antes y después del curso. (-: respuesta en blanco, x: incorrecta, v: correcta). Se presenta la reducción en % de las respuestas totales en blanco e incorrectas.....	159
Tabla 6.7. Resultados del cuestionario, sobre conocimientos específicos de los convertidores sigma-delta y pipeline, realizados antes y después del curso. (-: respuesta en blanco, x: incorrecta, v: correcta). Se presenta la reducción en % de las respuestas totales en blanco e incorrectas.....	160
Tabla 7.1. Resumen de los resultados obtenidos de los distintos convertidores A/D.....	164

RESUMEN

En esta tesis se presenta un estudio de las estructuras de convertidores A/D (analógico-digital) o ADCs (*Analog-to-Digital Converters*) de tipo *pipeline* y sigma-delta, que son las que han experimentado un mayor desarrollo en los últimos años y las que presentan mejores prestaciones. Primero se hace una introducción a los circuitos ADC en la que se analiza su estado actual de desarrollo, se presentan los fundamentos de la teoría de cuantificación y las características más importantes, estáticas y dinámicas, con que se especifican estos circuitos. Después, se resumen las distintas normas que se aplican a los convertidores resaltando los temas clave para su caracterización. Finalmente, se termina la introducción estudiando la problemática que supone la selección de la resolución y de la frecuencia de conversión de los ADC y se presenta un método basado en la relación señal-ruido de la señal.

Una vez se han abordado los temas genéricos de los convertidores se estudian en detalle cada uno de los dos tipos, los sigma-delta y los *pipeline*. En este estudio se tratan tanto el principio de funcionamiento como las metodologías de diseño y de simulación, poniendo de relieve los aspectos más importantes propios de cada estructura de convertidor. Una vez se ha comprobado la complejidad de los circuitos convertidores sigma-delta y *pipeline*, que proporcionan una buena resolución y una alta velocidad de conversión, se presentan dos estructuras originales de convertidores A/D. Una para los convertidores *pipeline* que evita tener que amplificar el residuo en las sucesivas etapas, y otra para los convertidores sigma-delta que facilita el diseño porque se parte de la especificación de la función de transferencia del ruido. Ambas propuestas se estudian mediante simulación y en el caso del convertidor sigma-delta se presentan los resultados prácticos obtenidos a partir de un prototipo real.

Como aplicación del trabajo realizado en desgranar los principios de funcionamiento de las estructuras avanzadas de convertidores, de cómo se diseñan y de cómo se simulan, se presenta una metodología docente para la enseñanza de los convertidores *pipeline* y sigma-delta, que incluye los resultados de su aplicación en un programa de enseñanza reglada.

ABSTRACT

This thesis deals with the advanced structures of the ADC (Analog-to-Digital Converters) circuits of the type of pipelined and sigma-delta, which have undergone a great development in recent years and also provide a better performance in comparison with traditional ones.

Firstly, an introduction to the ADC circuits is presented, including: the state of the art, the present development level, the foundations of the quantization theory and their main static and dynamic features, which characterize this kind of circuits.

After this, the standards to be applied to the ADCs are summarized and the key topics related to their testing are highlighted. Finally, the introduction ends by studying the problematic of how the resolution and the conversion rate of an ADC should be selected and presenting a method based on the signal-to-noise ratio of the signal to face up this problem.

Once the basics topics have been discussed, the sigma-delta and pipelined converters are studied in detail. In this study the working principles and the design and simulation methodologies are presented by highlighting the most important aspects of each type of converter.

After the complexity of the sigma-delta and the pipelined converters, which provide a good resolution and conversion rate, has been shown, two original structures of A/D converters are presented; one for the pipelined converters avoiding the need to amplify the residue in each stage and another one for the sigma-delta converters, which facilitates their design because it starts from the noise transfer function. Both proposals are studied by simulation and, in the case of the sigma-delta converter, practical results obtained from a real prototype are presented.

As an application of the work in outlining the principles of the ADC advanced structures and their design and simulation methods, a methodology to aid the ADC circuits learning process is presented, including the feedback results of their application setup in two consecutive academic courses.

CAPÍTULO 1. INTRODUCCIÓN

1.1. Antecedentes y objetivos de la presente tesis

En las últimas cuatro décadas, los circuitos convertidores A/D (Analógico a Digital) o ADC (*Analog-to-Digital Converter*) han evolucionado drásticamente [1]-[3]. Se ha conseguido que un convertidor típico (por ejemplo, el ADC-12 U de Analog Device, Inc.), que consumía 2.3 W, costaba 800 dólares, ocupaba una tarjeta de circuito impreso de aproximadamente 100 cm², tenía una velocidad de conversión de 100 kSPS (cien mil muestras por segundo) y una resolución de 12 bits, de paso a otros que disminuyen del orden de 100 veces los costes, de dinero y de espacio y, al mismo tiempo, multiplican por mil la velocidad de conversión y dividen por 20 el consumo de potencia. Un buen ejemplo es el convertidor modelo AD9215, que tiene una velocidad de conversión de 105 MSPS, 10 bits de resolución, ocupa 25 mm², sólo consume 145 mW y cuesta 8 dólares [4].

Este vertiginoso avance se debió fundamentalmente a tres cuestiones. La primera el avance de la tecnología electrónica, en especial de la microelectrónica (anteriormente a la aparición de los circuitos integrados ya existían convertidores A/D realizados con tubos de vacío [5]). La segunda, el desarrollo de nuevas estructuras de circuitos convertidores. Y la tercera, el desarrollo de la teoría de procesado de señal, en especial de sistemas discretos. Por eso, aquellos convertidores de aproximaciones sucesivas, paralelos, e integrales dieron paso a los nuevos convertidores *pipeline* y sigma-delta (Σ - Δ).

Estudiar a fondo estos convertidores, sus estructuras básicas, y proponer nuevas ideas de diseño es el primer objetivo de esta tesis. Para ello se deben cumplir los objetivos de revisar los conceptos generales de los convertidores A/D, demostrar la importancia que tienen estos circuitos y, además, desarrollar una metodología para entender cómo funcionan las estructuras avanzadas de los circuitos ADC [6], cómo se modelan [7] y cómo se simulan. Después de estudiar en profundidad los circuitos ADC sigma-delta y *pipeline*, el objetivo es evaluar sus puntos débiles y proponer algunas ideas que aporten soluciones originales para, por ejemplo, obtener:

- Un método sencillo para sincronizar las etapas de un convertidor *pipeline*.
- Un diseño para una etapa de convertidor *pipeline* que sea más inmune a las no idealidades de los componentes.
- Una metodología de enseñanza práctica de los circuitos ADC modernos.

- Un nuevo modelo que simplifique el diseño de los convertidores sigma-delta [8], [9].
- Un nuevo método para seleccionar la resolución y frecuencia de muestreo de los circuitos ADC basado en las características dinámicas de la señal a convertir.

1.2. Organización del contenido de esta tesis

El contenido de esta memoria se ha distribuido de la siguiente forma:

En el Capítulo 2 se hace una introducción a los circuitos convertidores A/D. Primero se clasifican y luego se analiza el estado actual de desarrollo y de su utilización, llegando a la conclusión de que los convertidores sigma-delta y *pipeline* son los más utilizados, junto con los de aproximaciones sucesivas. A continuación se introducen los fundamentos de la teoría de cuantificación y las características más importantes, estáticas y dinámicas, con que se especifican estos circuitos. Después, se resumen las distintas normas que se aplican a los convertidores resaltando los temas clave para la caracterización de los convertidores. Finalmente, se termina el capítulo presentando un método de selección de los circuitos convertidores A/D basado en la relación señal-ruido de la señal a convertir.

Una vez introducidos los conceptos generales sobre los convertidores A/D en el Capítulo 2, el Capítulo 3 se centra en presentar los convertidores de tipo sigma-delta. Después de hacer una breve introducción, se presenta cuál fue su origen y cómo evolucionó, explicando, bajo esta perspectiva histórica, su principio de funcionamiento. A continuación, se profundiza, primero en la parte del modulador y luego en la de filtrado y diezmado. Una vez analizadas la estructura y el modo de funcionamiento, se hacen simulaciones del modulador y de un convertidor completo. Finalmente, se realiza el test del convertidor simulado y se presentan algunas de las variantes de las estructuras básicas que han tenido más éxito.

De forma similar a como se presentaron los convertidores sigma-delta, en el Capítulo 4 se hace lo mismo con los de tipo *pipeline*. Se hace una introducción de los conceptos específicos de los convertidores *pipeline* y se exponen sus antecedentes. A continuación se explica cuál es su principio de funcionamiento y se diseña y simula un convertidor de este tipo de 5 bits. Después, se caracteriza y, finalmente, se presentan algunas de las variantes de la estructura básica que aparecen en la bibliografía.

Una vez introducidos todos los elementos necesarios para abordar la problemática de los convertidores, o sea, cómo funcionan, cómo se caracterizan y cómo se simulan, en el Capítulo 5 se estudian distintas alternativas de diseño de convertidores multietapa y con modulación sigma-delta. Con esto se consigue ejercitar los principios teóricos sobre los que se profundizó, presentando nuevas ideas alternativas, que ponen a prueba las estructuras clásicas. Estos estudios alternativos se concretan en la presentación de un nuevo convertidor de 5 bits multietapa y de una nueva forma de diseñar los moduladores sigma-delta. De ambos diseños alternativos se realizan numerosas simulaciones, teniendo especial interés por los resultados obtenidos el convertidor sigma-delta. De este último se realiza el montaje de un prototipo cuyos resultados validan los estudios realizados.

Como aplicación del trabajo que se realizó en desgranar los principios de funcionamiento de las estructuras avanzadas de convertidores, de cómo se diseñan y de cómo se simulan, en el Capítulo 6 se presenta una metodología docente para la enseñanza de los convertidores *pipeline* y sigma-delta. Esta metodología incluye los resultados de su aplicación en un programa de enseñanzas regladas en los cursos académicos 2006-2007 y 2007-2008.

Por último, se resumen los resultados obtenidos en el Capítulo 7 y se presentan las conclusiones finales en el Capítulo 8, en el que también se hace una propuesta de los trabajos futuros siguiendo la línea de investigación de esta tesis.

Al final de la memoria se incluye un anexo en el que se describe el contenido del CDROM que se adjunta y, que, básicamente contiene los proyectos de las simulaciones que se han realizado y las hojas de cálculo con las que se han hecho las gráficas más importantes. También se incluye una presentación con las diapositivas utilizadas en el curso de convertidores en el que se aplicó la metodología docente presentada en el Capítulo 6.

CAPÍTULO 2. LOS CIRCUITOS CONVERTIDORES A/D

2.1. Introducción

La tecnología electrónica ha experimentado un vertiginoso cambio que la ha hecho omnipresente en todos los sistemas tecnológicos que nos facilitan la vida diaria. Esto significa que el mundo físico en el que nos desenvolvemos está conectado de alguna manera a otro virtual soportado por estructuras matemáticas discretas que hacen constantemente medidas sobre variables físicas, toman decisiones en base a dichas medidas y, finalmente, actúan de nuevo sobre el mundo físico.

La Figura 2.1 muestra cómo la información de las variables del mundo físico se maneja con lo que se denominan señales, que son esencialmente analógicas y, en cambio, la tecnología electrónica que interacciona con ese mundo físico es digital. Esto implica que es necesaria una interfaz bidireccional entre estos dos mundos, el analógico (sistemas continuos) y el digital (sistemas discretos) [10], [11]. Por un lado las señales del mundo físico (tiempo y amplitud continuos) se convierten a digital (tiempo y amplitud discretos) mediante un convertidor A/D (Analógico a Digital) o ADC (*Analog to Digital Converter*) y, por otro lado, las señales digitales se transmiten al mundo analógico mediante un convertidor D/A (Digital a Analógico) o DAC (*Digital-to-Analog Converter*). De este análisis se puede deducir que la conversión de las señales analógicas a un formato digital es una de las bases de todos los sistemas electrónicos modernos que funcionan con aplicaciones reales.

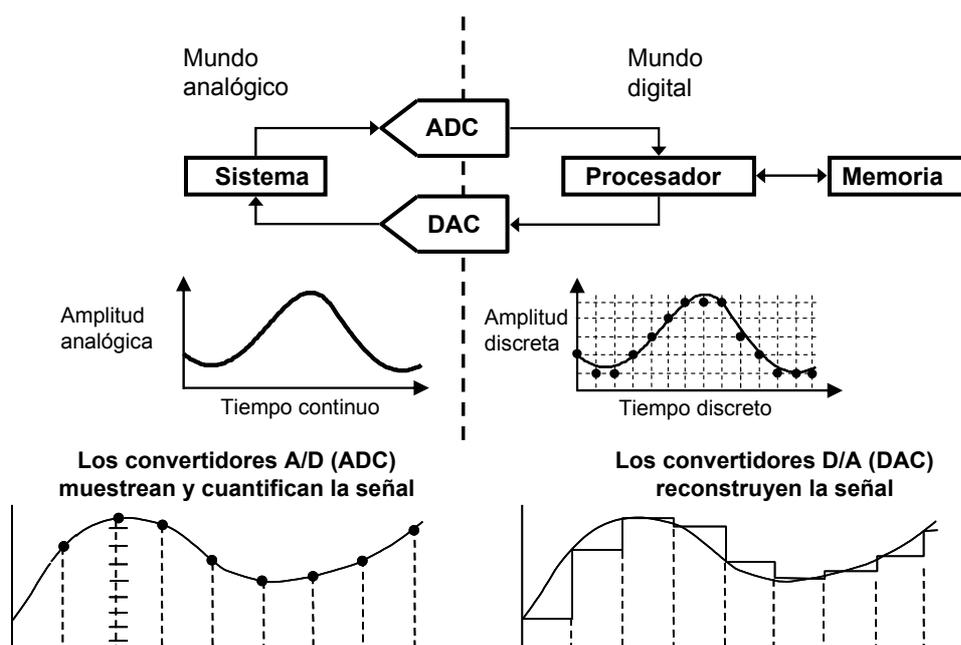


Figura 2.1. Interfaz electrónica entre el mundo analógico y el mundo digital.

Los sistemas electrónicos que, por un lado, adaptan las señales del mundo físico, las convierten a digital, las procesan y las almacenan en una memoria, y por otro lado convierten en analógicas las señales digitales y luego las amplifican para poder actuar sobre los sistemas físicos, son los sistemas electrónicos de instrumentación de medida y generación de señales que se denominan SAD (Sistemas de Adquisición de Datos) o DAQ (*Data Acquisition Systems*) [12], [13].

Un SAD, que básicamente es un sistema electrónico especializado en la obtención de información proveniente de un proceso del mundo real, está formado por los siguientes elementos:

- Un captador que se encarga de transformar la variable física a medir en una señal eléctrica.
- Un acondicionador que adapta la señal eléctrica obtenida por el captador.
- Una interfaz que transforma la señal analógica en una señal digital para acoplarla a un DSP (*Digital Signal Processor*).
- Un procesador digital cuyas tareas más comunes son: el procesamiento de los datos, la visualización, el almacenamiento y la transferencia de la información. Para realizar estas tareas el procesador debe disponer de capacidad de cálculo, de alguna interfaz hombre-máquina o HMI (*Human Machine Interface*), de recursos de almacenamiento y de comunicación con otros sistemas.
- Una interfaz de salida que transforma las señales digitales obtenidas por el procesador en analógicas.
- Un acondicionamiento para acoplar la señal analógica de salida a los actuadores del medio físico.
- Además, suelen tener entradas digitales de conteo.

Los SAD tradicionales suelen multiplexar la entrada al ADC para poder muestrear varias señales, esto disminuye la velocidad de muestreo por canal y hace necesario el uso de circuitos de muestreo y retención o S&H (*Sample & Hold*) para cada entrada, si se quieren muestrear todas ellas en el mismo instante de tiempo.

El acceso al procesador digital, un microcontrolador, un DSP [14] o un PC [15], [16], se realiza habitualmente a través de un bus de datos compartido con otros periféricos, esto hace que la velocidad de transacción máxima para leer el dato por el procesador sea reducida. La utilización de FPGAs (*Field Programmable Gate Array*) [17] en los sistemas de instrumentación, frente a los procesadores de arquitectura fija, evita todas estas desventajas [18]-[23]. En primer lugar, la disponibilidad de un gran número de terminales de entrada y salida hace posible acoplar cada entrada directamente con un

convertidor dedicado, siendo adquiridos de forma simultánea los datos analógicos por el procesador. En segundo lugar, el procesador puede ser construido en base a una arquitectura basada en un hardware, funcionando todo el conjunto de forma paralela. Además, disponen de recursos de memoria y de procesamiento que pueden simplificar los acondicionamientos y convertidores externos si se realizan algunas de sus tareas de forma digital, por ejemplo: las de filtrado, diezmado, interpolación, análisis frecuencial, etc. De esta forma se consigue mejorar la exactitud, la precisión, la fiabilidad y los costes de los SAD. Debido a estas ventajas los ADC se han desarrollado en la línea de complicar su parte digital y simplificar la analógica haciéndose menos dependientes de las características de los componentes analógicos, que, además, varían debido a las tolerancias de fabricación, cambios en las variables medioambientales, etc. Sea cual sea el tipo de procesador digital todos los SAD tienen, al menos, un ADC por lo que estos circuitos constituyen su parte esencial dando por hecho que se incluye en él el circuito S&H.

Las características más importantes de los ADC [24]-[26], son:

- El formato analógico de la señal de entrada, que puede ser unipolar o bipolar, en modo común o en modo diferencial.
- El formato digital de los datos de salida que puede ser binario natural o cualquier otro que entienda el procesador.
- El tiempo de conversión o tiempo que tarda el convertidor en realizar la conversión, desde que se da la orden de inicio desde el procesador hasta que el dato está preparado para ser leído por éste, es decir, hasta que las señales digitales están estables en el bus de datos.
- La resolución, cuya medida más directa es el número de bits del dato digital, cuanto mayor sea menos diferencia hay entre un código y el consecutivo.
- La estructura o topología del circuito, que define la forma en que se realiza la conversión y que predetermina en gran medida las demás características.
- El consumo de energía, que es una característica cada vez más importante, sobre todo en los dispositivos portátiles.
- La interfaz de acoplamiento al procesador digital que, principalmente puede ser serie o paralela.

2.2. Clasificación de los convertidores A/D

En este apartado se realiza una clasificación a los diversos tipos de circuitos convertidores A/D, de los que, dependiendo de la aplicación de que se trate, se requieren unas u otras características.

Los convertidores A/D se pueden clasificar, atendiendo a sus características principales, por ejemplo: según el formato digital, el formato analógico, la velocidad de conversión, la resolución, la estructura interna y el modo de operación. En la Tabla 2.1 se clasifican cualitativamente los ADC según la velocidad de conversión, en la Tabla 2.2 se presentan las fechas en las que se dieron los principales avances en los circuitos ADC [1] y en la Tabla 2.3 se comparan sus características más importantes clasificándolos según su estructura.

Tabla 2.1. Clasificación de los ADC según la velocidad de conversión.

Tipo de convertidor	Muy lento	Lento	De media velocidad	Rápido	Muy rápido
Velocidad de conversión	≤ 10 Hz	1 kHz	20 kHz	1 MHz	≥ 200 MHz

Las arquitecturas más populares de circuitos ADC [27], ordenados de menor a mayor velocidad de conversión, son:

- 1) *Convertidores de integración o integrales*: Estos convertidores se caracterizan por tener una baja velocidad de conversión frente a una alta resolución y un alto rechazo al ruido. Debido a estas características han sido utilizados extensivamente en aplicaciones de instrumentación de medida y sistemas de adquisición de baja velocidad. Actualmente están siendo sustituidos en la mayoría de las aplicaciones por los convertidores sigma-delta.
- 2) *Convertidores sigma-delta (Σ - Δ)* [28]: Se utilizan principalmente en aplicaciones de baja y media velocidad que requieren una elevada resolución. Son comunes en aplicaciones de audio, instrumentación, sonar, etc. Los anchos de banda son típicamente inferiores a una MSPS con resoluciones entre 12 y 24 bits efectivos.
- 3) *Convertidores SAR o de aproximaciones sucesivas*: La arquitectura frecuentemente elegida para aplicaciones de media y alta resolución con velocidades de hasta 8 MSPS son los convertidores SAR (*Successive Approximation Register*). Los rangos de resolución suelen estar entre 8 y 16 bits, tienen un bajo consumo de potencia, una baja figura de ruido y no tienen latencia en los

datos de salida. Esta combinación los hace ideales para una gran variedad de aplicaciones, tales como: adquisición de datos, control industrial, instrumentos portátiles, etc.

- 4) *Convertidores multietapa de tipo pipeline* [29], [30]: Este tipo de convertidores son los más utilizados en aplicaciones que requieren una frecuencia de muestreo entre unas pocas MSPS y unos cientos, con resoluciones entre 6 y 16 bits. Estas dos características hacen que sean los únicos que mantienen una buena relación velocidad-resolución-consumo, por ello son tan populares y cubren aplicaciones muy variadas: sensores de imagen, comunicaciones, video digital, etc.
- 5) *Convertidores flash o paralelos* [31]: Es la arquitectura que convierte señales analógicas a mayor velocidad y son muy adecuados para aplicaciones que requieren un gran ancho de banda. Sin embargo, este tipo de convertidores consumen una gran cantidad de energía, están limitados en resolución y tienen un elevado coste. Por todo ello se limitan a aplicaciones de muy alta velocidad, tales como: radar, satélite, osciloscopio de muestreo [32], etc. Existe una variante de los convertidores paralelos muy utilizada, que son los convertidores de dos pasos, que se denominan *half-flash* o semiparalelos. Estos convertidores están a medio camino entre un convertidor paralelo y uno *pipeline* (cuando utilizan dos circuitos S&H, uno en la entrada de cada etapa), llegando a obtener un compromiso entre el coste, la velocidad y la resolución que los hace muy competitivos en aplicaciones de procesamiento de señales a alta velocidad.

Tabla 2.2. Cronología de los principales avances en los circuitos convertidores A/D.

Año de aparición	Tipo de convertidor que aparece
1946	Aproximaciones sucesivas
1948	Paralelo con codificador de tubo de vacío
1950	Modulación Delta
1954	Sobremuestreo con conformado del ruido
1957	Doble rampa
1956	Semiparalelo
1962	Primer ADC denominado sigma-delta
1966	<i>Pipeline</i>
1988	Sigma-delta paso-banda

Tabla 2.3. Comparativa entre los distintos tipos de ADC.

Característica	Paralelo	Aproximaciones sucesivas	Integrales	Pipeline	Sigma-delta
Se elige esta arquitectura cuando se requiere:	Ultra alta velocidad sin que el consumo y el coste sean una restricción	Resoluciones elevadas y velocidades de hasta 5 MSPS, consumo y coste reducido y pequeño tamaño	Monitorización de señales de dc, alta resolución, bajo consumo, buena relación señal-ruido	Alta velocidad, velocidad de hasta 100 o más MSPS, de 8 a 16 bits de resolución, menos consumo que los paralelo	Alta resolución, baja y media velocidad
Método de conversión	Utiliza 2^N-1 comparadores, con cada bit de resolución se multiplica por 2 el nº de comparadores necesarios	Utiliza un algoritmo de búsqueda binaria, los circuitos internos trabajan a alta velocidad	El voltaje desconocido se integra y se compara con una referencia. Doble o simple rampa.	Pequeñas estructuras paralelas, cada etapa trabaja con uno o unos pocos bits	Sobremuestreo, rechazo programable de señales de 5 Hz a 60 Hz
Método de codificación	Se compara la señal con niveles de un divisor, la salida se decodifica digitalmente	Mediante registro de aproximaciones sucesivas (búsqueda binaria)	Integración analógica simple o doble	Segmentación y corrección digital del error	Modulador de sobremuestreo, filtro de diezmando digital
Desventajas	Códigos transitorios erróneos y metaestabilidad, alto consumo de potencia, gran tamaño, alto coste	La velocidad se limita a unas 5 MSPS. Puede necesitar un filtro anti-aliasing	Grado de conversión lento	El paralelismo aumenta las prestaciones a costa de aumentar el consumo y la latencia	Ancho de banda de entrada limitado. Baja velocidad de conversión
Tiempo de conversión frente a la resolución	No cambia al aumentar la resolución	Aumenta linealmente con la resolución	El tiempo de conversión se duplica con cada bit que aumenta la resolución	Aumenta con la resolución y el número de etapas	Compromiso entre la velocidad de conversión y el ruido
Resolución	La cantidad de componentes limita la resolución a unos 8 bits	Los componentes utilizados se duplican con cada bit que aumenta la resolución	Los componentes utilizados no aumentan significativamente con la resolución	Se duplican los componentes cada vez que se aumenta un bit la resolución	Un aumento en la resolución requiere aumentar la combinación de elementos
Tamaño	El tamaño y el consumo aumentan exponencialmente con la resolución	Aumenta linealmente con la resolución	No varía significativamente al aumentar la resolución	Aumenta linealmente con la resolución	No cambia significativamente al aumentar la resolución

2.3. Estado actual de los convertidores A/D

En las dos últimas décadas los circuitos ADC han experimentado una espectacular mejora en cuanto a sus características de velocidad de conversión, resolución, potencia consumida y coste económico. Además, se ha extendido su utilización hasta el punto de resultar imprescindible en cualquier sistema electrónico conectado con su entorno físico-químico. Esto ha sido debido a que en estos años se ha extendido el uso de los procesadores electrónicos digitales a todas las aplicaciones en

las que ha sido posible, y éstas han sido la mayoría. Para conseguir abarcar el gran abanico de aplicaciones en las que los ADCs se han ido incluyendo, éstos han tenido que evolucionar desde velocidades de conversión de unos pocos kHz a los GHz, con resoluciones que llegan a los 24 bits.

Estas prestaciones se han conseguido en parte gracias a la mejora de los métodos de fabricación de circuitos integrados y en parte a la aparición de nuevas estructuras que, combinando la teoría de procesado de señal con la tecnología electrónica, han dado lugar a los convertidores sigma-delta y *pipeline*. Hay que señalar que una de las áreas de aplicación que más ha exigido convertidores con mejores prestaciones, además de los sistemas de adquisición de datos y la instrumentación electrónica, ha sido la de los sistemas digitales de comunicación [33].

La gran difusión a la que han llegado estos convertidores ha provocado que se desbanquen los convertidores de integración y paralelos. Hoy en día los únicos convertidores que han sobrevivido con éxito han sido los de aproximaciones sucesivas o SAR. Esto se ha debido a que han prevalecido dos características importantes, que son: la fiabilidad y el tiempo real. Esto quiere decir que estos convertidores no necesitan de ningún tipo de post-procesado y que a cada ciclo de reloj proporcionan una nueva conversión correspondiente a la última muestra de la señal de entrada, sin latencia y sin retardos. En la Figura 2.2 se presenta el estado actual del lugar que cada uno de los tres grandes tipos de convertidores ocupa en cuanto a velocidad y resolución. Se advierte la gran velocidad de conversión a la que han llegado los convertidores SAR y Σ - Δ , que pasan del MHz de velocidad con más de 12 bits. De esta gráfica también se deduce que:

- Los convertidores Σ - Δ se han consolidado como los convertidores de mayor resolución sin que por ello se dejen de alcanzar altas velocidades de conversión, por ejemplo 24 bits a 1.5 MHz.
- Los convertidores de tipo *pipeline* son los más rápidos consiguiendo al mismo tiempo resoluciones relativamente elevadas, por ejemplo 16 bits a 120 MHz.
- Los convertidores de tipo SAR siguen manteniendo buenas prestaciones de velocidad y resolución, por lo que siguen siendo los idóneos para aplicaciones de control en las que prima el tiempo real.

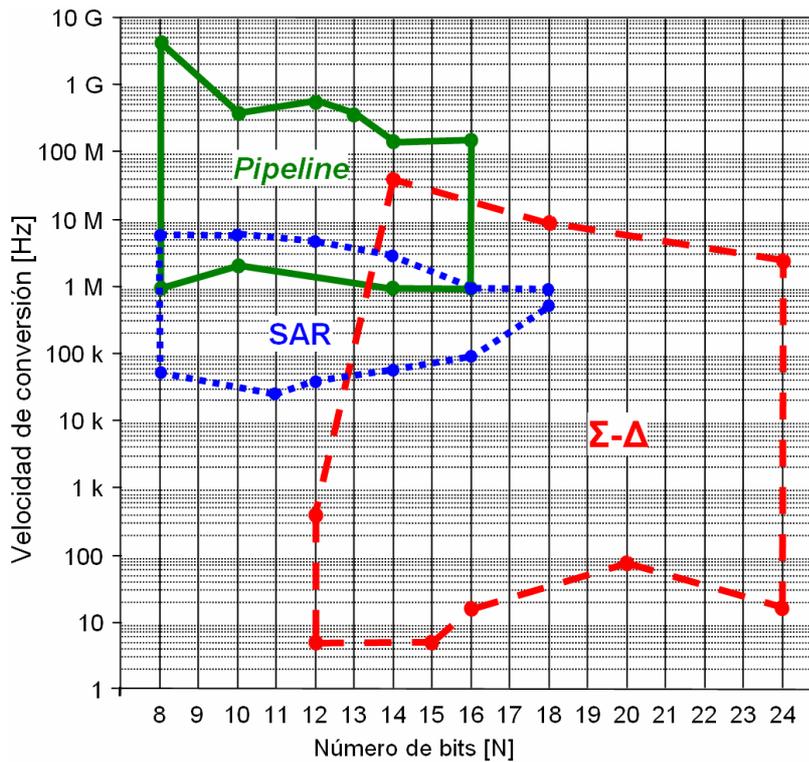


Figura 2.2. Resolución y velocidad de conversión de los convertidores más utilizados (actualizado en febrero de 2008).

En los siguientes apartados se realiza una introducción genérica a los convertidores A/D y se repasan sus conceptos básicos. En el Capítulo 3 se presenta un estudio del principio de funcionamiento de los convertidores sigma-delta y en el Capítulo 4 de los *pipeline*.

2.4. Traza de un convertidor ideal A/D

Una de las características más importantes de los convertidores A/D es el rango de tensión de entrada y el formato de los datos digitales de salida [32], [34]. La relación que hay entre la entrada y la salida se denomina traza o función de transferencia. En la Figura 2.3 y la Figura 2.4 muestran las funciones de transferencia ideales de tres convertidores con diferentes tipos de entradas analógicas y de salidas digitales.

La función de transferencia unipolar de la Figura 2.3.a. representa la traza de un convertidor que codifica una señal de entrada con un rango FS (*Full Scale*) y cuyo código digital de salida va desde uno con todos los bits “0”, que corresponde con 0 V, hasta otro con todos los bits a “1”, que corresponde con una tensión de FS-1LSB V, o sea, la tensión correspondiente al rango total menos la tensión equivalente al bit menos significativo o LSB (*Least Significant Bit*). Este tipo de codificación

también se suele llamar directa o natural. El número de valores discretos a lo largo de la recta depende de la resolución del convertidor.

La función de transferencia de la Figura 2.3.b. corresponde con un convertidor que tiene entrada bipolar con un rango de $-FS$ a $+FS-1LSB$. En este caso, el código digital se denomina binario desplazado (*offset binary*) donde la salida digital va desde un código con todos "0" correspondiente a la entrada $-FS$ a un código con todos "1" correspondiente a una señal analógica de $FS-1LSB$ V, pasando por el código un "1" y los demás "0" (el MSB a 1 y los demás bits a "0") que corresponde con una señal analógica de 0 V. Finalmente, la Figura 2.4 corresponde a un convertidor con un rango de entrada bipolar como el que se acaba de describir pero con una codificación digital en complemento a 2.

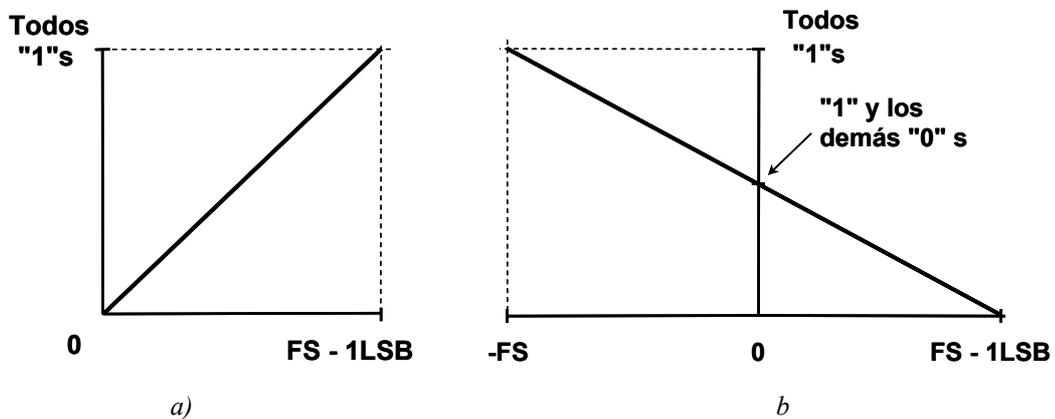


Figura 2.3. Funciones de transferencia de un convertidor: a) modo de entrada unipolar y salida en binario natural, b) modo de entrada bipolar y salida en binario desplazado.

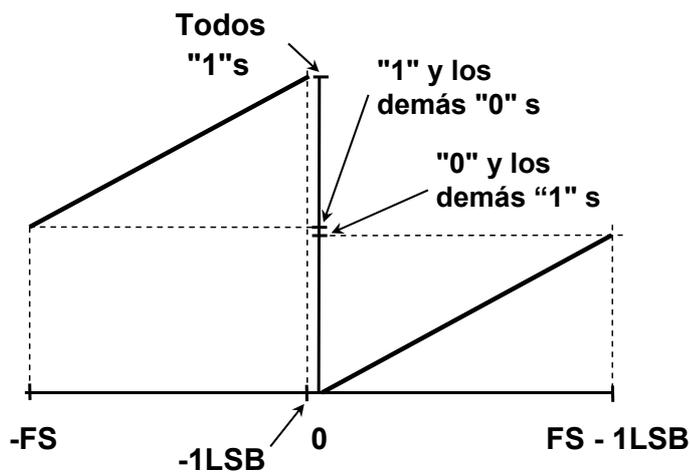


Figura 2.4 Función de transferencia de un convertidor con entrada bipolar y salida en complemento a 2.

A continuación, en la Tabla 2.4 [35], se clasifican los circuitos ADC según el formato de los datos de salida. Para ello, se define **Dato** como el valor de la variable codificada con **N** bits, donde **b_i** es el bit *i*-ésimo.

Tabla 2.4. Clasificación de los convertidores según el formato de los datos digitales.

Denominación del código	Representación del Dato de N bits	Rango
Binario natural	$Dato = \sum_{i=1}^N b_i \cdot 2^{-i}$	$0 \leq Dato \leq 1 - 2^{-N}$
Decimal codificado en binario o BCD	$Dato = 10^{-1} \cdot D_1 + 10^{-2} \cdot D_2 + \dots + 10^{-k} \cdot D_k$ con $D_k = \sum_{i=1}^4 b_i(k) \cdot 2^{-i+1}$	$0 \leq Dato \leq 0.999\dots$
Binario desplazado	$Dato = -1 + \sum_{i=1}^N b_i \cdot 2^{-i+1}$	$-1 \leq Dato \leq 1 - 2^{-N+1}$
Complemento a 2	$Dato = -MSB + \sum_{i=1}^{N-1} b_i \cdot 2^{-i+1}$ donde <i>MSB</i> es el bit de mayor peso	$-1 \leq Dato \leq 1 - 2^{-N+1}$
Complemento a 1	$Dato = (2^{-N+1} - 1) \cdot MSB + \sum_{i=1}^{N-1} b_i \cdot 2^{-i+1}$	$2^{-N+1} - 1 \leq Dato \leq 1 - 2^{-N+1}$
Signo-magnitud	$Dato = (-1)^{BS} + \sum_{i=1}^{N-1} b_i \cdot 2^{-i+1}$ donde <i>BS</i> es el bit de signo	$2^{-N+1} - 1 \leq Dato \leq 1 - 2^{-N+1}$

2.5. Error de cuantificación, relación señal ruido y bits efectivos

Cuando se convierte una señal continua con un circuito ADC, la salida toma una serie de valores discretos, esto es equivalente a decir que posee una resolución finita. La Figura 2.5 muestra la salida digital de una ADC de tres bits correspondiente a una señal analógica de entrada que varía desde cero hasta el rango total de entrada FS (*Full Scale*). En la parte inferior se presenta el error de conversión (la diferencia entre la entrada y la salida), que tiene forma de diente de sierra con una amplitud $Q/2$ (Q es la diferencia de nivel entre dos codificaciones consecutivas). Según la definición de valor eficaz de una onda triangular, el valor *rms* del error de conversión viene dado por la ecuación (2.1).

$$Ruido_{rms} = \frac{V_p}{\sqrt{3}} = \frac{Q/2}{\sqrt{3}} = \frac{Q}{\sqrt{12}} \quad (2.1)$$

El valor de Q se puede calcular dividiendo el rango de entrada entre el número de combinaciones posibles 2^N , según la ecuación (2.2). En un convertidor ideal, sin fuentes de error, el valor teórico del ruido de cuantificación rms es independiente de la amplitud y de la frecuencia de la señal.

$$Q = \frac{FS}{2^N} \tag{2.2}$$

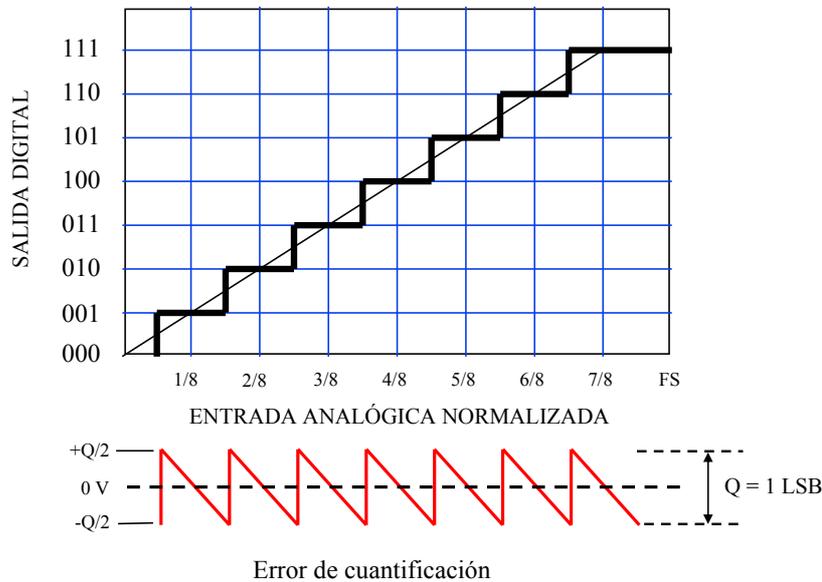


Figura 2.5. Cuantificación de la amplitud de una señal analógica, error de cuantificación.

La relación señal ruido o SNR de un convertidor es la relación entre el valor rms de una señal sinusoidal de amplitud igual al rango de entrada, que se corresponde con la ecuación (2.3), y el valor rms del ruido de cuantificación, dado por la ecuación (2.1), en la banda de Nyquist ($fs/2$). Estas dos ecuaciones se sustituyen en la ecuación (2.4) la cual, a su vez, se puede transformar en la expresión (2.5). Finalmente, se obtiene la ecuación (2.6) de la relación señal ruido del convertidor A/D en función del número de bits. La forma más usual de utilizar esta ecuación es con dos decimales, como se muestra en la ecuación (2.7).

$$Señal_{rms} = \frac{Q \cdot 2^{N-1}}{\sqrt{2}} \tag{2.3}$$

$$SNR = 20 \cdot \log \frac{Señal_{rms}}{Ruido_{rms}} = 20 \cdot \log \frac{\frac{Q \cdot 2^{N-1}}{\sqrt{2}}}{\frac{Q}{\sqrt{12}}} = 20 \cdot \log \frac{\sqrt{12} \cdot 2^{N-1}}{\sqrt{2}} = 20 \cdot \log \sqrt{6} \cdot 2^{N-1} \tag{2.4}$$

$$SNR = 20 \cdot \log \sqrt{6} \cdot 2^{N-1} = 20 \cdot \log \frac{\sqrt{6}}{2} \cdot 2^N = 20 \cdot \log \frac{\sqrt{6}}{2} + 20 \cdot N \cdot \log 2 \tag{2.5}$$

$$SNR = 1.7609 + N \cdot 6.0259 \text{ dB} \tag{2.6}$$

$$SNR = 6.02 \cdot N + 1.76 \text{ dB} \tag{2.7}$$

A partir de la ecuación anterior se deduce que, en un convertidor ideal, la única forma de reducir el ruido es aumentando el número de bits. En la Tabla 2.5 se representan, frente al número de bits, la proporción entre el peso del bit menos significativo y el rango de entrada, y la relación señal ruido.

Tabla 2.5. Resolución y rango dinámico de los circuitos ADC.

Número de bits	Número de niveles	Peso del bit menos significativo (%)	Relación señal ruido ideal (dB)
N	2^N	%, $100/2^N$	$SNR = 6.02 \cdot N + 1.76 \text{ dB}$
8	256	0.390625	48.2
10	1024	0.097656	60.2
12	4096	0.024414	72.2
14	16384	0.006104	84.3
16	65536	0.001526	96.3
18	262144	0.000381	108.4
20	1048576	0.000095	120.4
22	4194304	0.000024	132.5
24	16777216	0.000006	144.5

El diagrama de bloques equivalente del circuito ADC se representa en la Figura 2.6, éste es un modelo simplificado, actualmente se trabaja con modelos mas complejos que incluyen los distintos tipos de errores característicos de los subcircuitos que forman el convertidor [7]. En dicho modelo de la Figura 2.6 el convertidor se sustituye por un bloque sumador en el que se añade a la señal un ruido. Como de forma ideal la señal de error es triangular, su distribución de probabilidad uniforme, esto hace que el espectro de frecuencias sea una constante en el intervalo $[-f_s/2, +f_s/2]$, tal y como se muestra en la Figura 2.7. La varianza del ruido viene dada por el valor eficaz de la señal según (2.8).

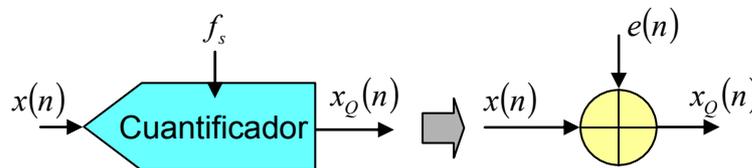


Figura 2.6. Modelo de un circuito ADC.

$$\sigma_e^2 = E[e^2(n)] = \frac{Q^2}{12} \ ; \ Q = \frac{FS}{2^N} \tag{2.8}$$

En el dominio de la frecuencia, se asume que el error $e(n)$ es un ruido blanco y, por tanto, su espectro es plano. Más precisamente, la potencia total promedio del ruido $e(n)$ es σ_e^2 y está uniformemente distribuido en el intervalo de frecuencias de Nyquist $[-f_s/2, +f_s/2]$. Por tanto, la potencia por unidad de frecuencia, o la densidad del espectro de frecuencia del error $e(n)$, viene dado por (2.9).

$$S_{ee}(f) = \frac{\sigma_e^2}{f_s} = \frac{Q^2}{12 \cdot f_s}, \text{ para } -\frac{f_s}{2} \leq f \leq \frac{f_s}{2} \quad (2.9)$$

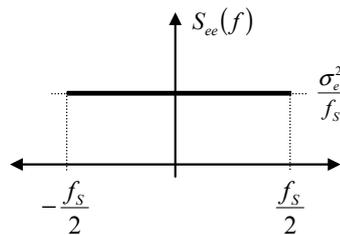


Figura 2.7. Espectro de frecuencia del ruido de cuantificación.

Se debe advertir que, aunque generalmente el ruido *rms* de cuantificación en toda la banda se aproxima por el ruido en la banda de Nyquist, hay ciertas condiciones para las cuales esto no es cierto. Por ejemplo, si existe correlación entre el error de cuantificación y la señal que se está digitalizando, entonces el ruido de cuantificación puede estar más concentrado en armónicos de la señal de entrada que repartidos por la banda total. Esto es más fácil que ocurra cuando la sinusoidal de entrada es un subarmónico de la frecuencia de muestreo.

2.6. Especificación de las características estáticas de los convertidores A/D

Para hacer una elección correcta del tipo de convertidor, el que mejor se adapte a la aplicación en cuestión, hay que evaluar, en primer lugar, las características estáticas, que derivan en errores que llevan a la pérdida de códigos, las más importantes son:

- No linealidad diferencial, DNL (*Differential Non-Linearity*).
- No linealidad integral, INL (*Integral Non-Linearity*).
- Pérdida de códigos (*Missing Codes*).
- Error de ganancia (*Gain Error*).
- Error de offset (*Offset Error*).

La Figura 2.8 representa la función de transferencia ideal de un convertidor de 3 bits. En el eje horizontal se representa la entrada analógica y en el vertical el código digital de salida. La variación en la señal de entrada que provoca una transición en el código de salida se denomina ancho del código, cuyo valor ideal es exactamente 1 LSB. En la práctica, cada ancho de código es diferente al de al lado.

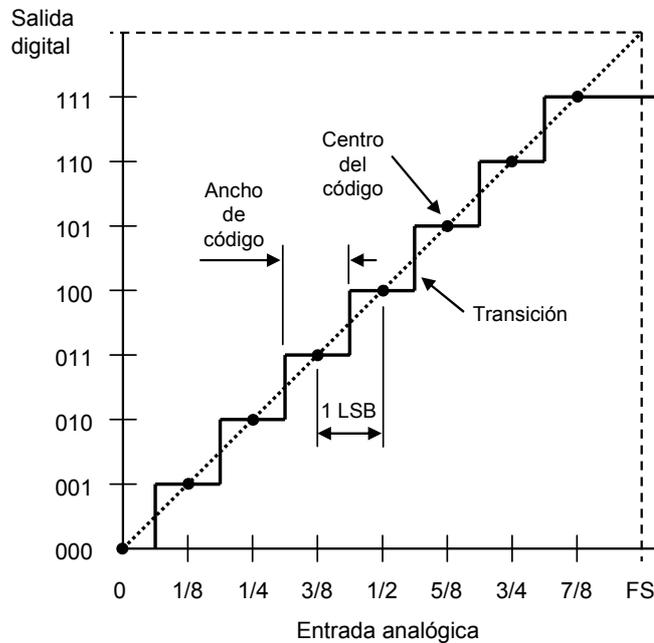


Figura 2.8. Función de transferencia de un ADC ideal de 3 bits.

La DNL es la desviación en el ancho de los códigos del valor ideal LSB. En la Figura 2.9 se muestra la función de transferencia real de un convertidor de 3 bits con varios errores, en ella se aprecia que se ha perdido el código “100”, esto es debido a una DNL demasiado grande en los códigos adyacentes. La pérdida de códigos puede producir oscilaciones indeseadas en sistemas en bucle cerrado, por tanto, la DNL es un parámetro importante a considerar cuando se selecciona un ADC. En la Figura 2.10 se muestra la gráfica de la DNL del convertidor AD9288, cuya DNL oscila entre -0.5 y +0.5 LSB.

La no linealidad integral o INL se suele medir con respecto al centro de los códigos. En la Figura 2.11 se muestra un ejemplo de este error. El error INL es la distancia que hay entre la línea ideal y el centro del código. Por tanto, la mayor desviación entre cualquier centro de código y la línea recta ideal es el error de no linealidad integral. La Figura 2.12 se corresponde con una curva de la INL del convertidor comercial AD9288. Los errores de ganancia y de *offset*, que se muestran en la Figura 2.14 y la Figura 2.16, respectivamente, son constantes en todo el rango, por lo que habitualmente se pueden corregir fuera del convertidor.

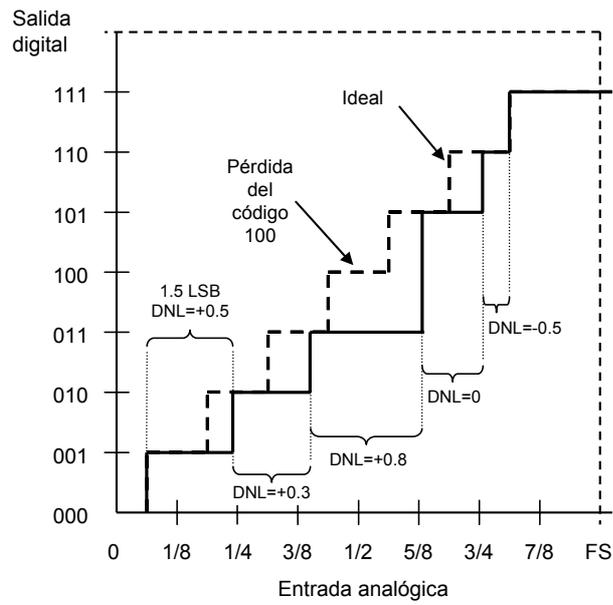


Figura 2.9. Representación del error de no linealidad diferencial de un ADC.

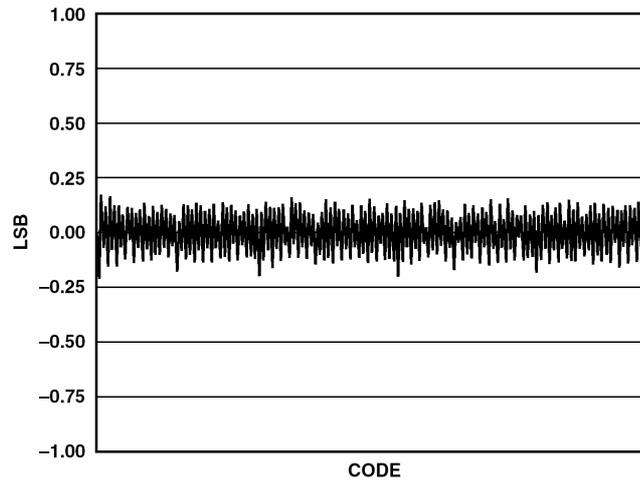


Figura 2.10. Curva del error de no linealidad diferencial del ADC AD9288 (cortesía de Analog Devices, Inc.).

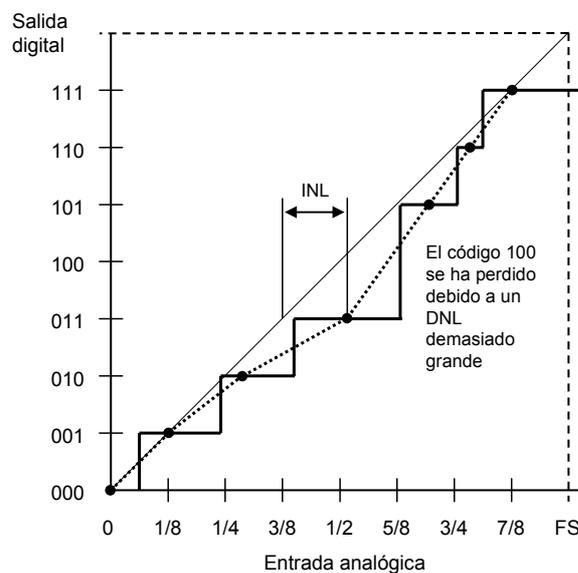


Figura 2.11. Representación del error de no linealidad integral en un ADC de 3 bits.

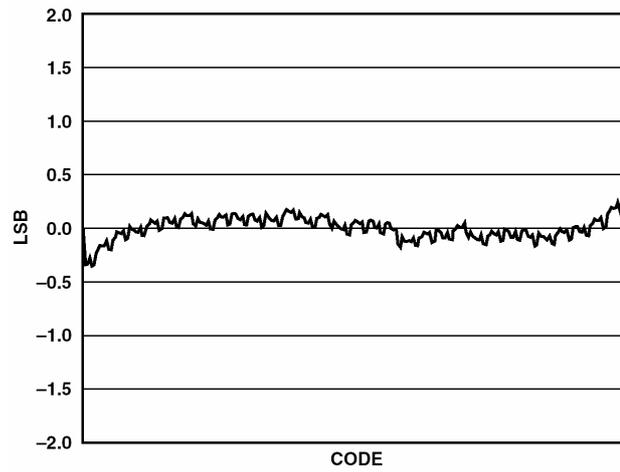


Figura 2.12. Curva del error de no linealidad integral del ADC AD9288.

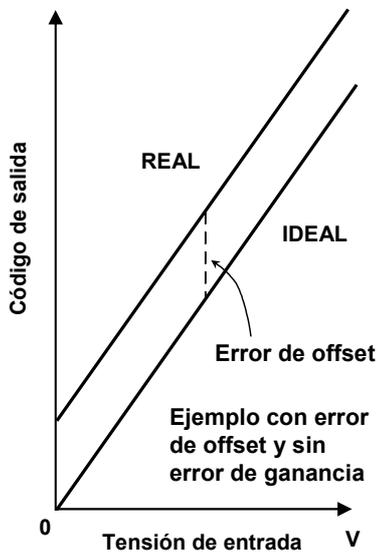


Figura 2.13. Error de offset de un ADC con entrada unipolar.



Figura 2.14. Error de ganancia de un convertidor unipolar.

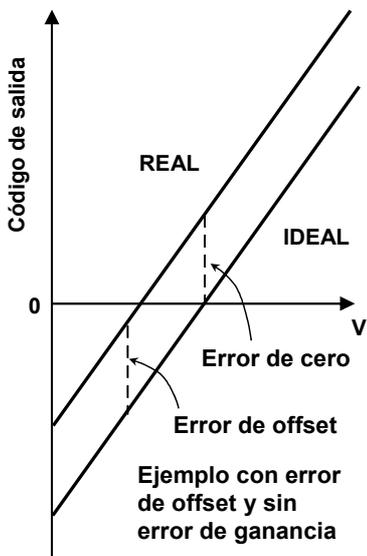


Figura 2.15. Error de offset de un ADC con entrada bipolar.

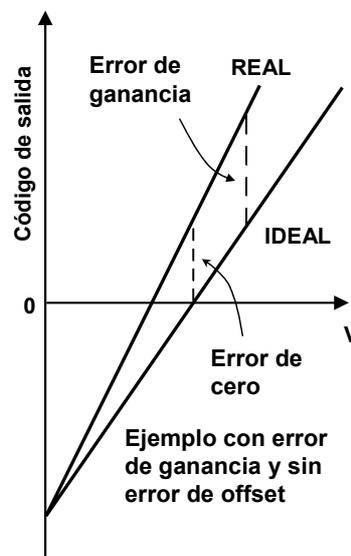


Figura 2.16. Error de ganancia de un ADC con entrada bipolar.

2.7. Especificación de las características dinámicas de los convertidores A/D

En la mayoría de las aplicaciones se requiere de los circuitos ADC, además de unas buenas características estáticas, que posean unas buenas características dinámicas, sobre todo en las aplicaciones de procesado de señal. A continuación se presentan las características dinámicas más importantes a tener en cuenta en los ADC [11]:

- Distorsión armónica total (THD).
- Relación señal ruido más distorsión (S/N+D) y número efectivo de bits (ENOB).
- Espurios de pico, contenido de armónicos de pico y rango dinámico libre de espurios (SFDR).
- Ancho de banda total de potencia.
- Ancho de banda lineal total.
- Distorsión de intermodulación (IMD).
- Retardo del instante de muestreo y su variación (*Aperture Delay Time* y *Aperture Jitter*).
- Respuesta transitoria.
- Latencia o retardo de un ADC.
- Recuperación de sobrevoltaje.
- Sensibilidad del reloj de muestreo al ciclo de trabajo.

Hay un conjunto de arquitecturas que son adecuadas para diseños de circuitos ADC para DSP y la mayoría de ellas disponen a su entrada de un circuito de muestreo y retención o S&H (*Sample and Hold*). Alguna excepción destacable es la de los convertidores de tipo paralelo (*flash*) y, sobre todo, los de tipo sigma-delta, que se verán en el Capítulo 3. Las características del conjunto S&H+ADC están bien determinadas debido a que ambos suelen estar contenidos en el mismo encapsulado por lo que su interfaz esta bien definida.

La distorsión armónica total (THD)

La distorsión armónica total es la relación entre la suma de todos los valores *rms* de las componentes armónicas y el valor *rms* de la entrada a fondo de escala. Habitualmente los fabricantes de convertidores dan este dato teniendo en cuenta las seis primeras componentes armónicas, como muestra la ecuación (2.10), y se suele expresar en tanto por cien o en dB.

$$THD = \sqrt{\frac{V_{f2}^2 + V_{f3}^2 + V_{f4}^2 + V_{f5}^2 + V_{f6}^2 + V_{f7}^2}{V_{f1}^2}} \quad (2.10)$$

En la Figura 2.17 se muestra un ejemplo de una curva de distorsión en dB, teniendo en cuenta las seis primeras componentes armónicas, frente a la frecuencia de la señal de entrada. En este caso se proporcionan tres curvas para diferentes valores de amplitud de entrada, la que correspondería con la definición sería la de -0.5 dB.

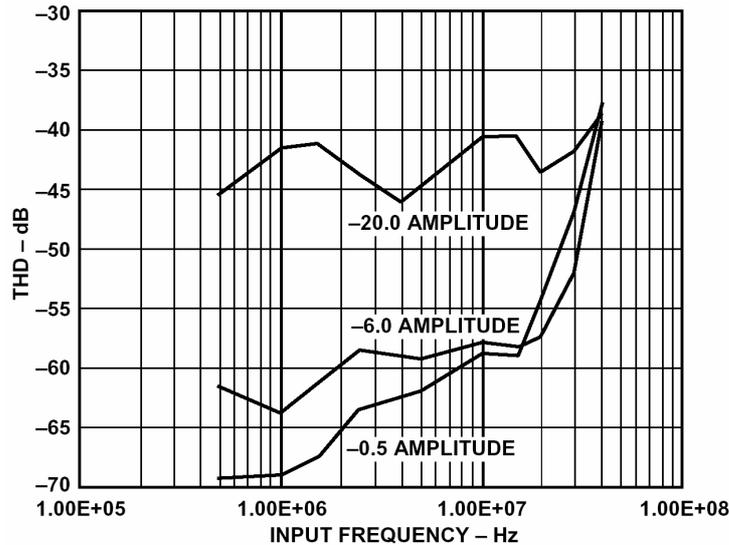


Figura 2.17. Distorsión total del ADC AD9280 para diferentes amplitudes de entrada.

Relación señal ruido más distorsión (S/N+D) y número efectivo de bits

En el cálculo del valor *rms* del ruido se acostumbra a incluir los armónicos de la señal fundamental. Esto se denomina relación señal ruido más distorsión, cuyas siglas en inglés son SINAD (*Signal to Noise And Distortion*), ésta es la relación entre la amplitud *rms* de la señal (a 1 dB por debajo del rango total de entrada) y el valor *rms* de la suma de todas las componentes del espectro de frecuencia en la banda de Nyquist, incluyendo los armónicos y excluyendo la componente de *dc*, tal como se muestra en la ecuación (2.11). En la Figura 2.18 se representa la curva SINAD del convertidor AD9288, que es del fabricante Analog Devices, Inc.

$$SINAD = \frac{V_{f1}}{\sqrt{V_{f2}^2 + V_{f3}^2 + V_{f4}^2 + V_{f5}^2 + V_{f6}^2 + V_{f7}^2 + V_{ruido\ rms}^2}} \quad (2.11)$$

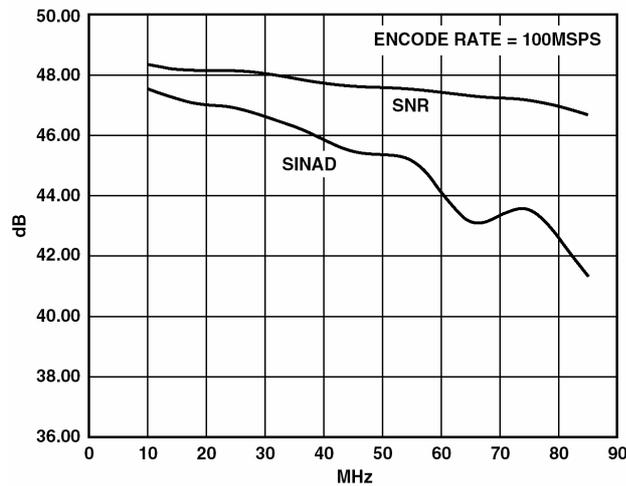


Figura 2.18. Curva SINAD del convertidor AD9288 en función de la frecuencia de la señal de entrada.

Otra forma de interpretar el valor de la SNR es en términos del número efectivo de bits o ENOB (*Effective Number of Bits*). El ENOB se calcula sustituyendo el valor medido de la SNR en la ecuación (2.7) y despejando N, tal y como queda en la ecuación (2.12).

Para el convertidor de 8 bits AD9288 correspondiente a la Figura 2.18, a 70 MHz la SNR es de 47.5 dB, por lo que según la ecuación (2.12) el valor del ENOB es de 7.6 bits.

$$ENOB = \frac{SNR_{REAL} - 1.76 \text{ dB}}{6.02} \tag{2.12}$$

El valor del ENOB se puede medir utilizando el método de ajuste a una curva sinusoidal. En este método se aplica una señal sinusoidal a la entrada del convertidor y se reúnen un conjunto de muestras. A continuación, en vez de calcular la FFT, se calcula la senoide que mejor se ajusta a las muestras que se han obtenido. Los valores de la amplitud, la fase y la frecuencia de la senoide se calculan minimizando el valor *rms* del error entre la senoide real de entrada y la de la onda teórica. Como el valor *rms* teórico del error de cuantificación para un ADC ideal viene dado por la ecuación (2.1), se calcula el valor *rms* del error teórico y el medido de la senoide reconstruida, y se calcula el ENOB utilizando la ecuación (2.13), y para el cálculo de la SNR se aplica la ecuación (2.14).

Q_A : Valor real del error *rms* medido de la senoide.

Q_B : Valor teórico del error *rms* correspondiente a los N bits de la senoide que mejor se ajusta.

$$ENOB = N - \log_2 \left[\frac{Q_A}{Q_B} \right] \tag{2.13}$$

$$SNR = \frac{SNR_{REAL} - 1.76 \text{ dB} + \text{Nivel de la senoide por debajo del rango total}}{6.02} \tag{2.14}$$

Rango dinámico libre de espurios o SFDR

El pico espurio o armónico de pico es la mayor componente del espectro excluyendo la componente de *dc* y la fundamental. Este valor se expresa en dB relativos al valor *rms* de la señal de entrada de fondo de escala. Esta especificación también se suele referir al rango dinámico libre de espurios o SFDR, que es una figura de merito del ADC muy utilizada. Una gráfica típica mostrando este rango se presenta en la Figura 2.19.

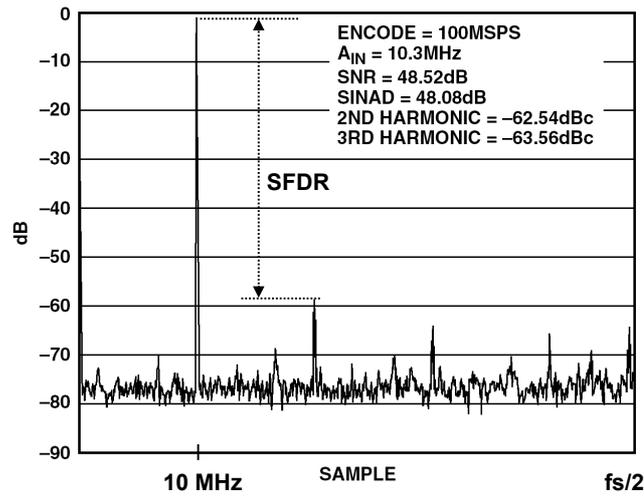


Figura 2.19. Rango dinámico libre de espurios del convertidor AD9288.

Ancho de banda total de potencia

El ancho de banda total de potencia o FPBW (*Full-Power BandWidth*) es la frecuencia a la que la amplitud de la señal fundamental reconstruida es de -3 dB relativa a una señal de entrada que tiene una amplitud igual al rango total. En la Figura 2.20 se muestra un ejemplo de este parámetro.

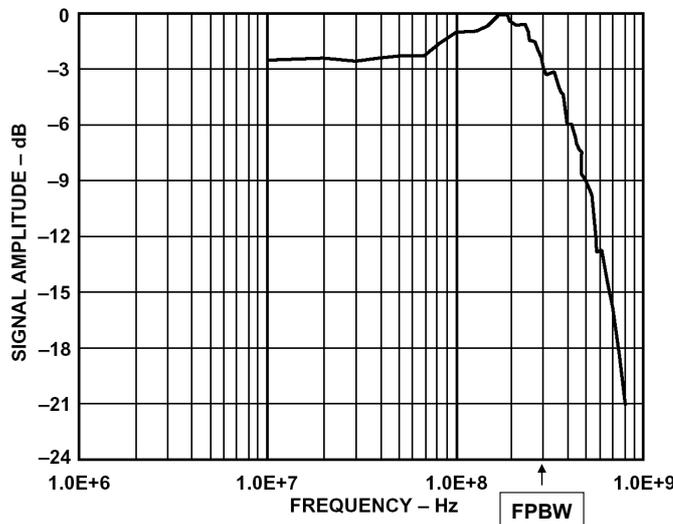


Figura 2.20. Ejemplo de curva de respuesta en frecuencia del ADC modelo AD9280.

Ancho de banda lineal total

El ancho de banda lineal total de un ADC es la frecuencia de entrada a la cual se alcanza la velocidad máxima (*Slew Rate*) del circuito de entrada de muestreo y retención. En este punto, la amplitud de una sinusoidal reconstruida se degrada menos de 0.1 dB. Por encima de esta frecuencia la distorsión de la señal de entrada muestreada aumenta significativamente. Se suelen diseñar los circuitos para que esta frecuencia esté por encima de la frecuencia de Nyquist para poder submuestrear señales de mayor frecuencia pudiendo reconstruir señales periódicas. Esta técnica se puede utilizar con éxito, por ejemplo, en el muestreo multiperiodico.

Distorsión de intermodulación (IMD)

La distorsión de intermodulación tiene lugar cuando la entrada es una señal compuesta por dos o más componentes de distintas frecuencias, típicamente se suele evaluar este parámetro con una señal formada por dos componentes F_1 y F_2 . Cualquier dispositivo con no linealidades creará productos de distorsión, del orden $m+n$, a las frecuencias suma y diferencia de $m \cdot F_1 \pm n \cdot F_2$, donde $m, n = 1, 2, 3, \dots$. En este caso, los términos de segundo orden son $(F_1 + F_2)$ y $(F_1 - F_2)$, y los de tercer orden son $(2F_1 + F_2)$ y $(2F_1 - F_2)$, $(F_1 + 2F_2)$ y $(F_1 - 2F_2)$ (Figura 2.21). Los productos IMD se expresan como la relación en dB entre el valor *rms* de la suma de la señales de entrada y el valor *rms* de los productos de distorsión. Las dos señales aplicadas a la entrada del ADC son de la misma amplitud y el valor de pico de la suma debe estar a -0.5 dB del rango máximo de entrada. Los productos IMD están normalizados a 0 dB de la señal de entrada. Como se puede comprobar en la Figura 2.21 los terceros productos de intermodulación están muy cerca de la señal, por esto se debe poner especial atención en ellos, especialmente en aplicaciones de radiofrecuencia, ya que son difíciles de filtrar. Para caracterizar este parámetro se utilizan dos tonos lo más cercanos posibles (frecuencia de separación entre 1/100 y 1/400 de la frecuencia de muestreo) y a una frecuencia tal que los primeros productos estén dentro de la banda de Nyquist. La Figura 2.22 es la gráfica del análisis frecuencial del convertidor AD9288, el eje horizontal representa la banda de Nyquist que, dado que la frecuencia de muestreo es de 100 MHz, resulta ser de 50 MHz. En esta gráfica se señala la IMD dada por el fabricante del convertidor, al que se le ha inyectado dos tonos separados 0.6 MHz.

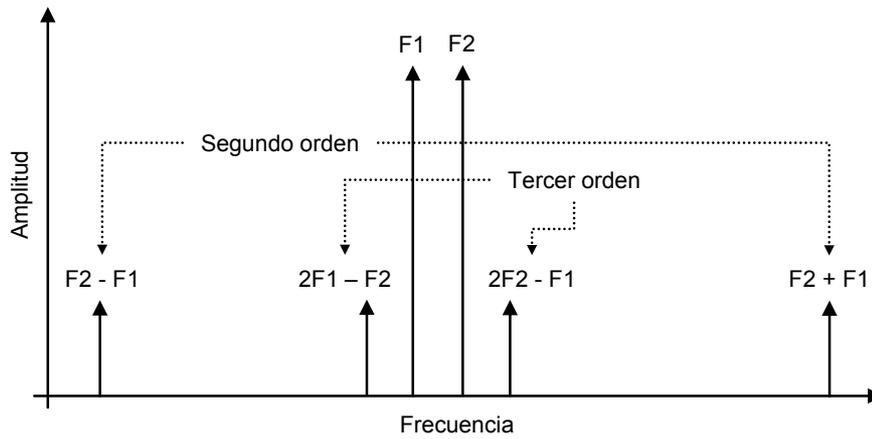


Figura 2.21. Productos de intermodulación.

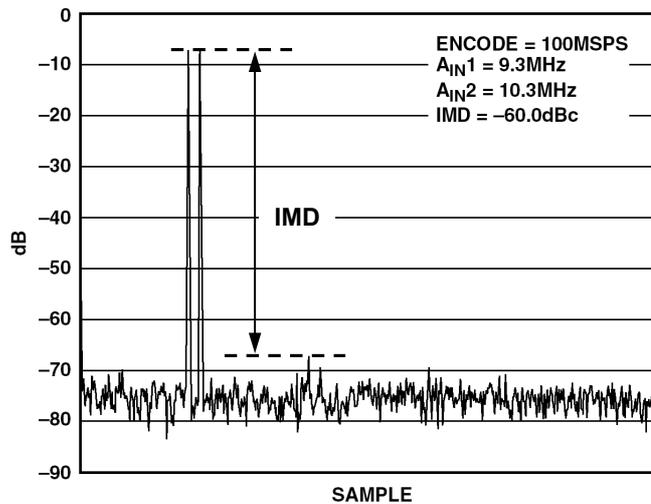


Figura 2.22. Distorsión de intermodulación.

Retardo del instante de muestreo (Aperture Delay Time)

El retardo del instante de muestreo es el tiempo que transcurre desde que tiene lugar el flanco de subida del reloj de muestreo hasta que el convertidor captura efectivamente la muestra. Esta especificación es importante porque ayuda al usuario a conocer cuándo aplicar la señal de reloj con respecto a la señal de entrada. Las diferencias que pueden existir en este parámetro entre diferentes unidades del dispositivo se deben tener especialmente en cuenta cuando las aplicaciones requieren muestreo simultáneo.

Margen de fluctuación del instante de muestreo (Aperture clock Jitter)

En un circuito ADC el retardo del instante de muestreo no es constante, es decir, varía muestra a muestra, esta fluctuación del instante de muestreo se conoce como *clock jitter* [36]-[39]. Y el margen

de fluctuación de este instante, que se muestra en la Figura 2.23, se conoce como *aperture jitter*. Este error suele ser causado por diferentes motivos. En un ADC real, a menudo la fase del reloj de muestreo se modula de forma no deseada debido a diversas fuentes, por ejemplo: ruido aleatorio en el origen de la señal de reloj, ruido en la alimentación o ruido inducido debido al trazado (*layout*) del circuito. El error final se puede expresar en términos de un tiempo *rms* de *jitter*, que se denomina t_j . El error de voltaje debido a este error de tiempo hace que disminuya la relación señal ruido, tal y como muestra la ecuación (2.15), que representa la SNR que se pierde debida exclusivamente al *jitter* (la variable f es la frecuencia de la señal muestreada).

$$SNR_{Jitter} = 20 \cdot \log_{10} \left[\frac{1}{2 \cdot \pi \cdot f \cdot t_j} \right] \quad (2.15)$$

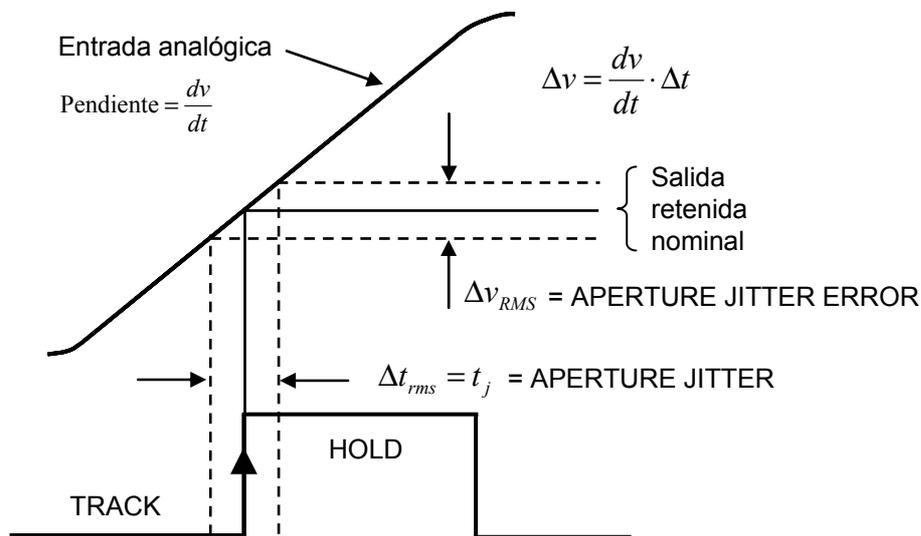


Figura 2.23. Error debido a la fluctuación en el instante del muestreo.

La Figura 2.24 muestra la SNR que se pierde debido al *jitter* del reloj de muestreo como función de la frecuencia de la señal de entrada, que es una sinusoidal con una amplitud igual al rango total de entrada. Según esta gráfica, si se tuviese un ADC ideal (sin ningún tipo de error y con un número muy elevado de bits) pero el reloj de muestreo tuviese un *jitter* de 10 ps, la mejor relación señal ruido conseguida por el sistema, para una señal de entrada sinusoidal de 100 MHz, sería de unos 44 dB y, en consecuencia, el número efectivo de bits se vería limitado a sólo 7. Este ejemplo resalta la importancia que tiene el reloj de muestreo en el proceso de conversión, ya que por mucha resolución que tuviese el convertidor no valdría de nada si no se pusiese especial atención en la calidad de la señal de reloj.

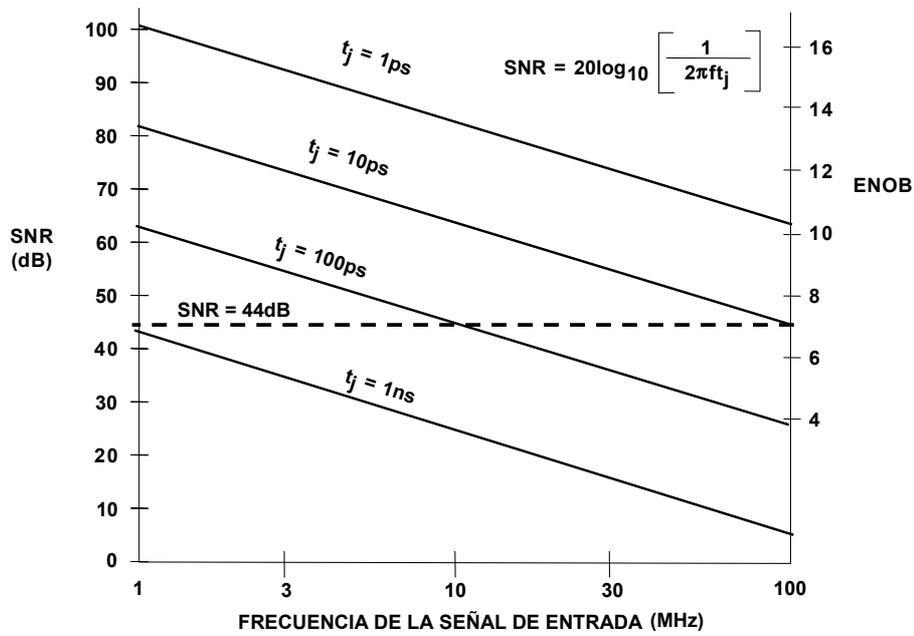


Figura 2.24. Disminución de la SNR debida a la fluctuación del reloj de muestreo.

Tiempo de establecimiento o respuesta transitoria

Es el tiempo empleado por el convertidor en alcanzar la salida con la exactitud deseada, cuando se aplica a la entrada un escalón con una amplitud correspondiente al rango de entrada total.

Latencia de un ADC o retardo pipeline

La latencia de un convertidor A/D es el tiempo, en ciclos de reloj, que tarda en obtener a su salida el código que representa el nivel de la señal de entrada. En la Figura 2.25 se presenta un ejemplo en el que la latencia es de tres ciclos de reloj. La latencia es una característica muy importante a tener en cuenta en aplicaciones de tiempo real, tales como el control en lazo cerrado.

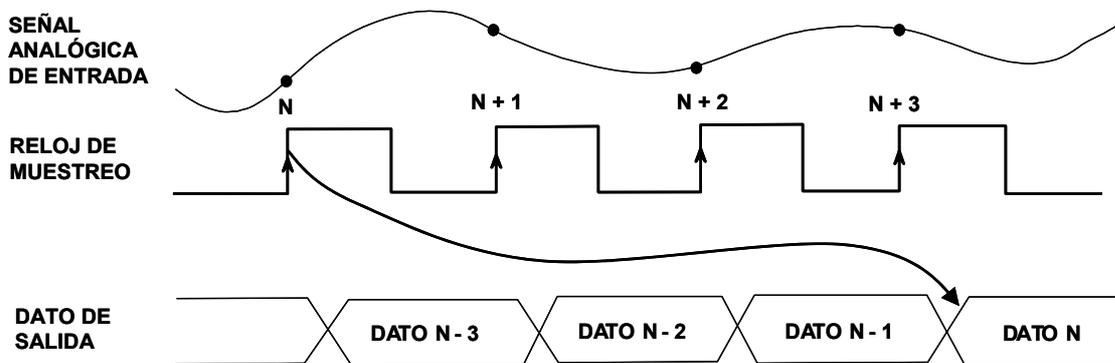


Figura 2.25. Latencia en un ADC.

Sensibilidad al ciclo de trabajo del reloj de muestreo

Los circuitos ADC suelen ser sensibles al ciclo de trabajo del reloj de muestreo, tanto por necesitar una relación t_{ON}/t_{OFF} lo más parecida a 0.5, como por necesitar un t_{ON} determinado. Un ejemplo de un convertidor que necesita mantener el t_{ON} en un rango óptimo es el convertidor AD9288, el cual, tal y como muestra la Figura 2.26, necesita un t_{ON} de entre 4 y 6 ns para mantener una buena relación señal ruido. Los convertidores que utilizan los dos flancos del reloj son más sensibles a su ciclo de trabajo, la Figura 2.27 representa una curva típica de un convertidor de este tipo.

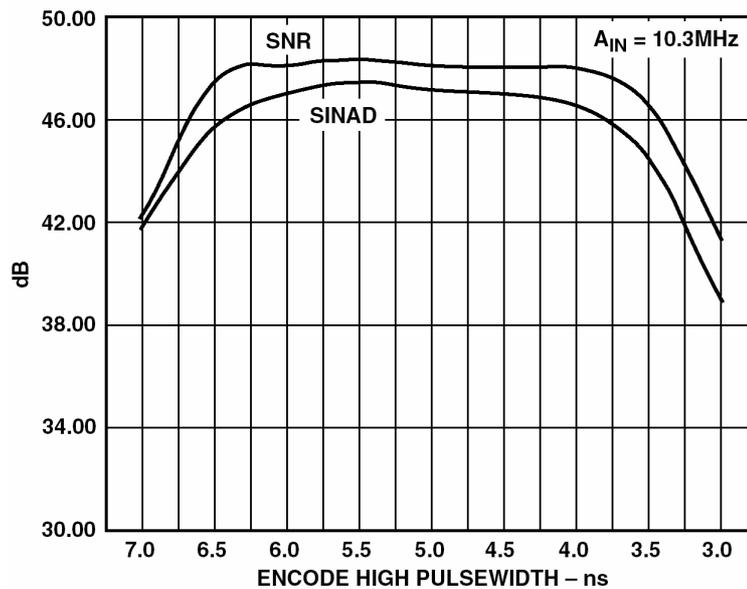


Figura 2.26. Variación de la relación señal ruido del convertidor AD9288 frente al ancho del pulso del reloj de muestreo.

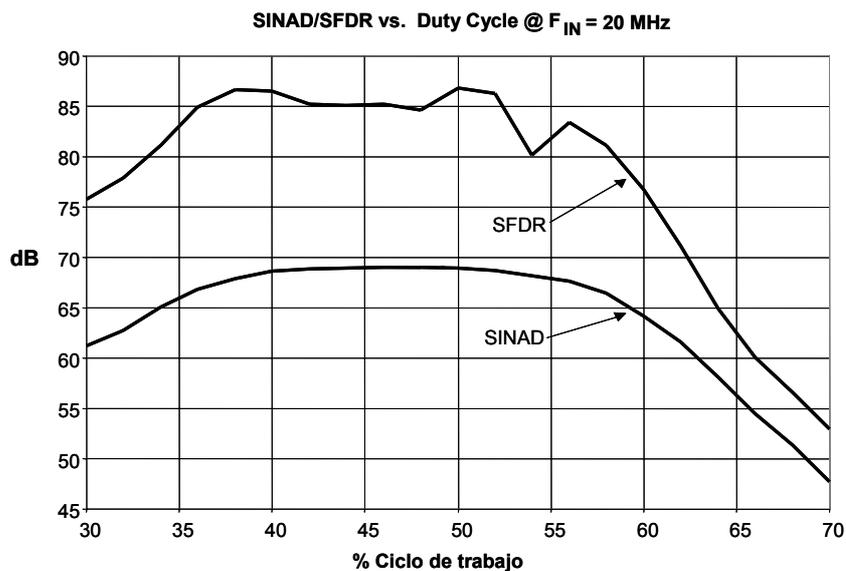


Figura 2.27. Variación de la SINAD y de la SFDR frente al ciclo de trabajo del reloj de muestreo.

2.8. Normativa y métodos de ensayo para los circuitos ADC

2.8.1. Normativa

En los últimos años, la Comisión Electrotécnica Internacional (IEC) y el grupo TC-10 de la Sociedad de Instrumentación y Medida del IEEE han realizado una gran labor de normalización de los aspectos que afectan a los circuitos convertidores, y que se han plasmado en las normas internacionales que a continuación se resumen brevemente [40]-[43]:

IEC 62008-2006. Característica de funcionamiento y métodos de calibración para los sistemas de adquisición de datos digitales y su software asociado

Esta norma internacional especifica las características y métodos de calibración destinados a los sistemas de adquisición de datos digitales y a su software asociado, de forma que se asegure que todos los sistemas de medida basados en estos dispositivos de adquisición de datos multifunción (DAQ) respondan a una norma común. Esta norma trata de la conversión de señal de baja frecuencia, por ejemplo, en las aplicaciones de control de fábrica, en la medida de las vibraciones, los vibro-diagnósticos, las medidas acústicas o ultrasónicas, de temperatura, de presión, la medida en electrónica de potencia, etc.

Esta norma abarca los siguientes aspectos:

1. Las especificaciones mínimas del dispositivo DAQ que el fabricante debe suministrar para describir las características de su módulo analógico-digital (ADM).
2. Las estrategias de ensayo normalizadas para verificar un conjunto mínimo de especificaciones.
3. La información mínima relativa a la calibración, exigida por el ADM, que será almacenada en el dispositivo DAQ.
4. Los requisitos mínimos del software necesario para la calibración externa y la autocalibración del ADM del dispositivo DAQ.

IEC 60748-4-3-2006. Semiconductor devices – Integrated circuits – Dynamic criteria for analogue-digital converters (ADC)

Esta parte de la norma IEC 60748 especifica un conjunto de métodos de medida y de requisitos para calibrar convertidores A/D bajo condiciones dinámicas, junto con la terminología y características asociadas.

IEEE Std. 1241-2000. IEEE Standard for Terminology and Test Methods for Analog-to-Digital Converters

Esta norma IEEE se refiere a la terminología y a los métodos de ensayo de los convertidores A/D, identifica las fuentes de error y proporciona métodos de ensayo con los que conocer dichos errores de medida. La información contenida en este estándar está orientada tanto a los fabricantes como a los usuarios de los circuitos ADC y facilita la base para poder comparar distintos dispositivos. Además, proporciona plantillas para poder escribir las especificaciones nuevas.

IEEE Std. 1057-R2001. IEEE Standard for Digitizing Waveform Recorders

Contiene la terminología y los métodos de ensayo para describir el comportamiento y características de los registradores de forma de onda de digitalización.

2.8.2. Ensayo y caracterización de los circuitos ADC

2.8.2.1. Introducción

Los ensayos que se realizan sobre los circuitos ADC se clasifican, según qué características se desean obtener, en estáticos y dinámicos [44], existiendo diversos métodos para realizar cada uno de ellos [45]-[47]. Básicamente consisten en inyectar una señal conocida al convertidor y compararla con los resultados obtenidos a su salida. A continuación se describen someramente los métodos que se utilizan para realizar los test de ensayo de las características estáticas de los ADC, que son los más utilizados en los convertidores de media y baja velocidad y alta resolución [48].

Inicialmente, para realizar las medidas de las características estáticas de un convertidor A/D, además de utilizar un generador y un DVM (voltímetro digital), se solía disponer de un convertidor D/A de mejores prestaciones para reconstruir la señal muestreada y compararla con la señal de

referencia que se muestreaba [49]. De esta forma, el error obtenido se podía visualizar y analizar directamente en un osciloscopio. Posteriormente, con el estándar IEEE-1057 [43] se comenzó la normalización de los métodos de test estáticos. Según éste, los niveles de transición se determinaban de uno en uno aplicando a la entrada del ADC una señal constante que se iba incrementando hasta que se pasaba al siguiente nivel de transición ($T[i]$). Cada vez que se cambiaba de nivel se adquirían un conjunto de muestras de la salida cuyo número dependía de la desviación típica del ruido y del nivel de confianza del resultado que se desease conseguir. Posteriormente se comparaban estadísticamente los resultados obtenidos con los valores de la señal que se inyectaba. Este método necesitaba un generador de alta precisión (con una resolución mejor que $Q/4$) y un tiempo de establecimiento del orden de 1 s para cada nivel de transición, por lo que este procedimiento de test resultaba de un coste elevado, no sólo económico, si no también de tiempo. Estos inconvenientes fueron tratados en el desarrollo del nuevo estándar IEEE-1241 [40], en el que se estableció un procedimiento de test totalmente nuevo. Este método está basado en el uso de un bucle de realimentación en el que un DAC genera la señal de realimentación que se aplica al ADC bajo test. El valor del dato digital de entrada al DAC se incrementa o decrementa dependiendo del valor de la última conversión del ADC. Primero se genera un valor ligeramente inferior al esperado para el nivel de transición que se está probando. Después, el código digital de salida del ADC se compara con el generado, que se incrementará hasta que la diferencia sea nula o de una unidad. Un valor práctico para establecer el número de muestras que se toman por cada nivel que se prueba se establece según la expresión: $2 \cdot (\sigma_1 / \sigma_2)^2$, donde σ_1 es la desviación típica del ruido y σ_2 representa la incertidumbre en el código que se está generando, a estas variables se les suele dar los valores: $\sigma_1 = 2 \text{ LSB}$ y $\sigma_2 = 1/4 \text{ LSB}$. Por tanto, un valor típico para el número de muestras a promediar para cada nivel es de 128. Este método, al utilizar realimentación, facilita el establecimiento de la señal de referencia de forma precisa y con un menor tiempo de establecimiento y, por tanto, disminuye el coste total. Sin embargo, el tiempo necesario para realizar el test continua siendo elevado.

Estos métodos de test estáticos, en los que se realiza un barrido de la señal de entrada con incrementos del orden de $1/4$ de Q y se adquieren, para cada valor, un conjunto de muestras M que luego se tratan estadísticamente, se denominan, según la norma IEC-62008:2006 [41], como métodos

tipo A. En la misma norma IEEE-1241 se expone el método tipo B, que utiliza como señal de entrada al ADC ondas triangulares de pequeña amplitud con un nivel de dc que se va incrementando progresivamente para explorar todo el rango de entrada del ADC. Este método reduce el número de muestras necesarias para un determinado nivel de incertidumbre en la medida y disminuye, también, el número de escalones para explorar todo el rango de entrada. Por último, añadir que en la norma IEC-62008:2006 el análisis sobre la caracterización de los ADC se extiende al sistema completo de adquisición [50], [51], e introduce una guía para evaluar la incertidumbre de medida total, que se basa en la Guía para la Expresión de la Incertidumbre de Medida [52].

Los métodos de ensayo de las características dinámicas de los convertidores más populares son los que utilizan el análisis en el dominio de la frecuencia mediante la descomposición en series de Fourier y mediante la Transformada Discreta de Fourier o DFT [53], [54]. También se han desarrollado metodologías probabilísticas que se basan en elegir como señal de referencia una cuya distribución de probabilidad de las muestras sea conocida, por ejemplo una señal triangular. Entonces, se adquiere una gran cantidad de muestras con el ADC bajo test y se compara su distribución de probabilidad con la de la señal muestreada, en estos métodos estadísticos se suelen utilizar como presentación de resultados los histogramas [55].

Los métodos que utilizan la transformada de Fourier no resuelven directamente toda la problemática que conlleva caracterizar dinámicamente un convertidor. Los resultados varían con el tipo de ventana de datos utilizada y, además, los resultados obtenidos de la transformada necesitan ser procesados para obtener la información buscada. Por ejemplo, para obtener la SNR por métodos frecuenciales aparece el problema de diferenciar en el espectro las componentes armónicas de las del ruido. Además, la amplitud de las componentes armónicas se refiere a la energía del espectro y no al valor en voltios rms , lo cual es otro problema, en [56] y [57] se analizan estas problemáticas. Existe otra solución para analizar los errores a nivel de bit que utiliza la transformación de Walsh que da una sola componente de salida para cada bit erróneo lo cual hace más fácil realizar diagnósticos sobre el funcionamiento a nivel de bit [58].

Un ejemplo de aplicación de los métodos estadísticos está en [59], en el que se hace una comparación entre los convertidores paralelos y los SAR, utilizando como factor de mérito la desviación cuadrática media en presencia de error de *offset* o de ruido en los comparadores.

Para realizar el ensayo de un convertidor se suele seguir el esquema simplificado de la Figura 2.28. Primero, un generador sinusoidal introduce una señal de una frecuencia determinada f_i en el convertidor de forma que su amplitud cubra el rango de entrada (FSR). Después, se registra una cantidad M de muestras que, finalmente, se analizan mediante técnicas DSP. Por ejemplo, el procesador calcula la FFT (*Fast Fourier Transform*) a un número finito de muestras y las convierte en un espectro de frecuencias. En la Figura 2.29 se muestra un ejemplo de análisis en frecuencia del convertidor AD9288 de 100 MSPS [60].

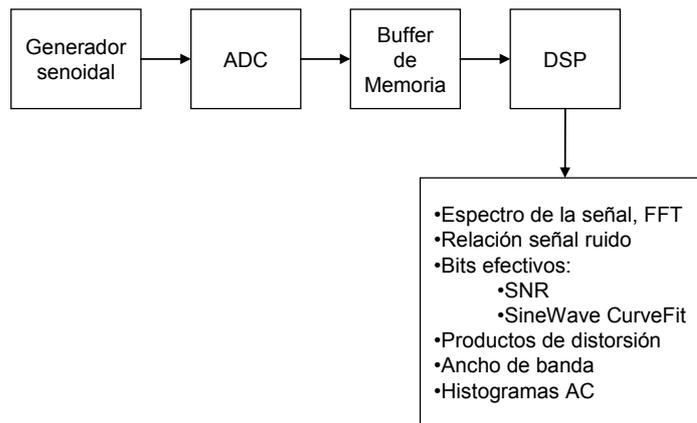


Figura 2.28. Configuración para realizar el test de la dinámica de un ADC.

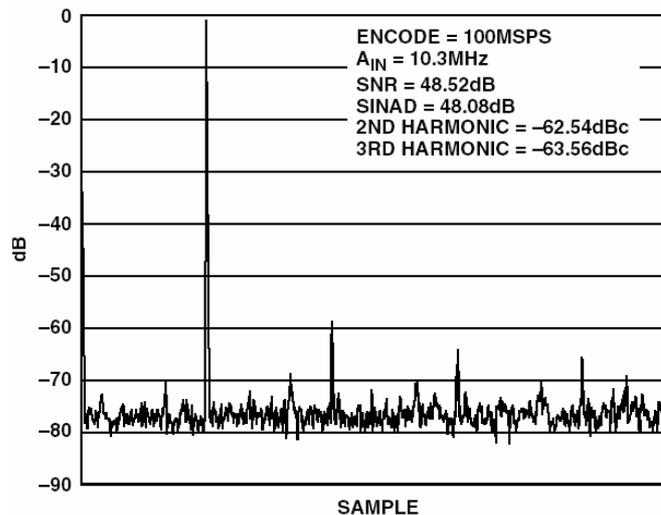


Figura 2.29. Espectro de frecuencia de la salida del convertidor AD9288.

2.8.2.2. Configuración del escenario para realizar los ensayos

La Figura 2.30 muestra la configuración para realizar el ensayo del ADC mediante excitación sinusoidal. Las señales sinusoidales se utilizan frecuentemente por la facilidad para obtenerlas y porque también es fácil establecer su calidad. Para realizar este ensayo es necesario, además del generador sinusoidal, un generador de reloj. También es posible combinar más de un generador

sinusoidal para proporcionar dos o tres tonos, si el ensayo lo requiere, por ejemplo, para obtener la distorsión de intermodulación. Finalmente, de forma excepcional se puede disponer de un tercer generador de ruido en el caso de hacer un ensayo con señales que tengan un pequeño nivel de ruido (*dithering*). Si es posible, el generador de la señal a convertir y el del reloj de conversión deben estar adecuadamente sincronizados para facilitar el procesamiento de los datos. También puede ser necesario algún filtro para acondicionar las señales de salida de los generadores.

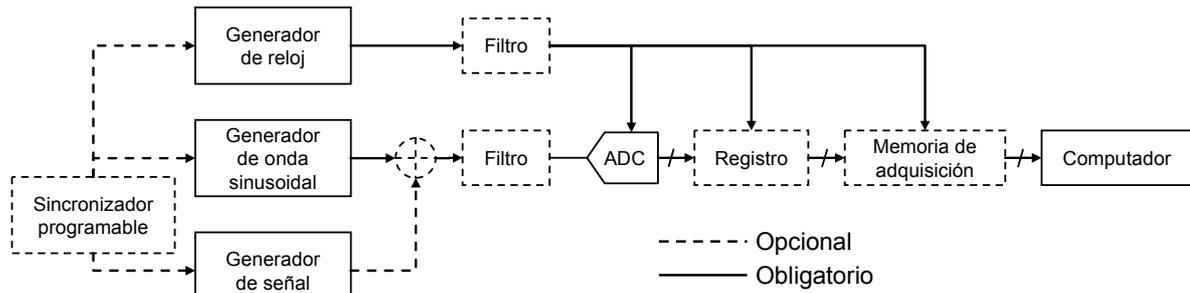


Figura 2.30. Configuración para el ensayo de un ADC mediante excitación sinusoidal.

A continuación se presentan, de forma resumida, los métodos de análisis basados en el dominio del tiempo y de la frecuencia. En ambos métodos se parte de un registro de tamaño M de las muestras de la señal de entrada, cuya frecuencia y amplitud se han establecido adecuadamente.

2.8.2.3. Métodos de ensayo en el dominio del tiempo

El objetivo de este ensayo es obtener la SINAD y el ENOB del convertidor a partir del análisis en el dominio del tiempo.

En primer lugar se establece el tamaño M del registro de muestras teniendo en cuenta que, para que al menos haya una muestra para cada código posible, se debe tomar para el valor de M el expresado por (2.16).

$$M \geq \pi \cdot 2^N \tag{2.16}$$

Después, si también se va a realizar un análisis en el dominio de la frecuencia utilizando la FFT, se debe tomar el múltiplo de potencias de 2 inmediatamente superior al valor obtenido de M. Por ejemplo, si se tratase de un convertidor de 5 bits, M sería de 101, pero se tomarían 128 muestras, para así poder realizar la FFT, que es más rápida de calcular que la DFT.

En segundo lugar se determina la frecuencia óptima f_{opt} de la señal de entrada para que, a la frecuencia de muestreo deseada, se obtenga uno o varios ciclos completos de la señal convertida en el registro de tamaño M. Para ello se aplica la ecuación (2.17).

$$f_{opt} = \frac{f_s}{M} \quad (2.17)$$

Una vez configurada la señal sinusoidal de entrada, la frecuencia de muestreo y el tamaño del registro de datos, se procede a realizar las conversiones y a registrarlas en la memoria de adquisición.

Después, utilizando el método de los tres o el de los cuatro parámetros [40], se busca la sinusoidal que mejor se ajusta a los datos del registro minimizando el error entre ambas señales (la registrada y la de referencia). Una vez obtenida la senoide que mejor se ajusta, se obtiene el vector de error mediante la ecuación (2.18). En la Figura 2.31 se muestra, a modo de ejemplo, una gráfica con los datos y_n de salida de un convertidor, los de la sinusoidal que mejor se ajusta y'_n y el vector de error \mathcal{E}_n .

$$\mathcal{E}_n = y_n - y'_n \quad (2.18)$$

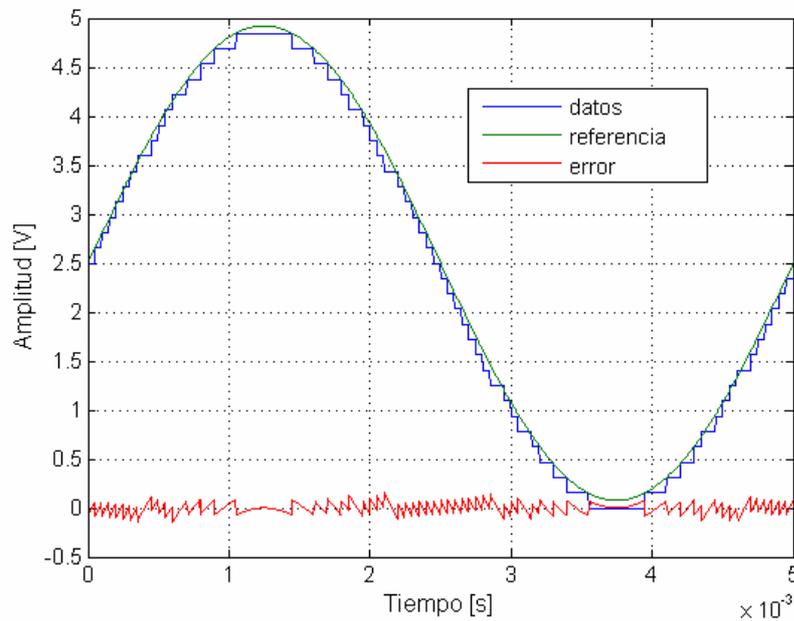


Figura 2.31. Ejemplo de obtención del error de conversión a partir de un registro de datos.

Luego, se calcula el valor eficaz del vector de error utilizando la fórmula (2.19) [40]. Finalmente sólo queda calcular la SINAD con la ecuación (2.20) y, a partir de ella, obtener el ENOB mediante (2.21).

$$\text{ruido rms} = \sqrt{\frac{1}{M} \cdot \sum_{n=1}^M \varepsilon_n^2} \quad (2.19)$$

$$\text{SINAD} = 20 \cdot \log\left(\frac{\text{señal rms}}{\text{ruido rms}}\right) \quad (2.20)$$

$$\text{ENOB} = \frac{\text{SINAD} - 1.76}{6.02} \quad (2.21)$$

2.8.2.4. Método de ensayo en el dominio de la frecuencia

A partir de la serie de Fourier del registro de los datos del convertidor, de tamaño M , se obtiene una secuencia de tamaño $(M/2)+1$ (desde 0 hasta $M/2$) correspondiente a sus componentes de frecuencia, la primera es la componente de continua, la segunda es la fundamental (de la frecuencia de la señal), y las demás son las componentes armónicas, de las que se obtiene la SNR, la SINAD y la THD. En la Figura 2.32 se muestra una gráfica con las componentes de frecuencia de un registro obtenido a partir del mismo convertidor que el del análisis temporal de la Figura 2.31.

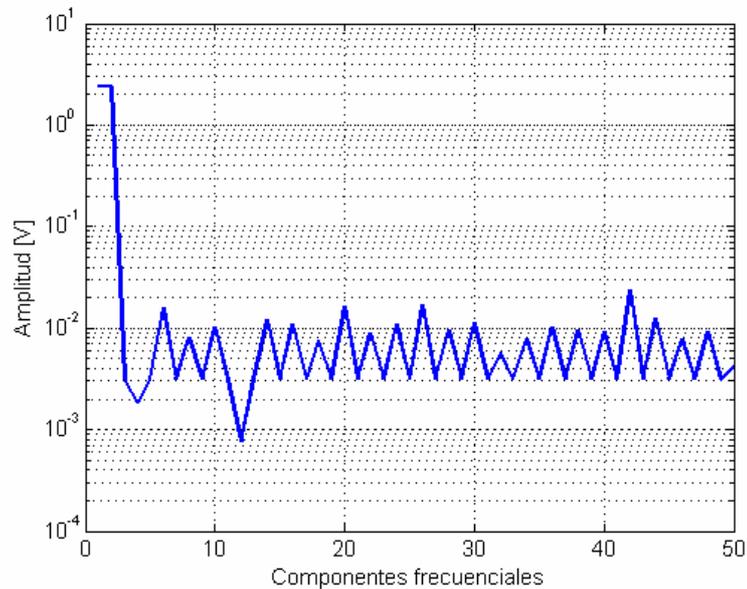


Figura 2.32. Componentes del espectro de frecuencia de la señal convertida (componentes de la DFS).

El valor *rms* del ruido se obtiene a partir de la ecuación (2.22), sumando todos los valores *rms* de las componentes armónicas, y de la relación entre los valores *rms* de la señal y del ruido se obtiene, aplicando la ecuación (2.20), la relación señal ruido más distorsión (SINAD).

$$ruido\ rms = \sqrt{\frac{1}{2 \cdot \left(\frac{M}{2} - 1\right)} \cdot \sum_{h=2}^{\frac{M}{2}} (X[h])^2} \quad (2.22)$$

Para calcular la distorsión armónica se aplica la ecuación (2.23) [42], que relaciona las diez primeras componentes armónicas (salvo que se indique lo contrario) con la componente fundamental.

$$\%THD = \sqrt{\frac{\sum_{h=2}^{11} (X[h])^2}{X[1]}} \cdot 100 \quad (2.23)$$

Ahora, si para calcular el ruido, se prescinde de las diez primeras componentes en la ecuación (2.22), entonces, en vez de obtener la SINAD se obtiene la SNR, como se indica en (2.24).

$$SNR = 20 \cdot \log \left(\frac{\left(\frac{M}{2} - 1\right) \cdot X[1]}{\sqrt{\sum_{h=12}^{\frac{M}{2}} (X[h])^2}} \right) \quad (2.24)$$

Resumiendo, si la serie de Fourier es: $X_0, X_1, X_2, \dots, X_{11}, X_{12}, \dots, X_{M/2}$; entonces, para calcular la:

- SINAD: se computan todas las componentes armónicas desde la X_2 hasta la $X_{M/2}$.
- SNR: se computan las componentes armónicas desde la X_{12} hasta la $X_{M/2}$.
- THD: se computan sólo las diez primeras componentes armónicas desde la X_2 hasta la X_{11} .

2.9. Metodología para seleccionar los circuitos ADC según las características dinámicas de la señal

2.9.1. Introducción

Cuando una señal analógica se convierte en digital mediante un ADC aparecen los efectos de la discretización del tiempo (muestreo) y de la amplitud (cuantificación) [61]. Estos efectos provocan una disminución en la calidad de la señal, que se suele indicar por su relación señal ruido o SNR. Esta SNR que se pierde disminuye cuando se aumenta la resolución y la frecuencia de conversión, por lo que una solución al problema consistiría en elegir un circuito con la mayor resolución y frecuencia de conversión posible. Sin embargo, si no se toma el mínimo valor de dichos parámetros se aumentará innecesariamente el coste y la complejidad del circuito. En este apartado se presenta un método para

seleccionar la resolución y la frecuencia de conversión de un ADC de forma óptima. Para ello se utiliza un criterio de selección basado en la característica dinámica, tanto del convertidor como de la señal a convertir, y se pone como restricción la calidad de la señal que se desea obtener. Por tanto, se parte de la SNR de la señal y de la SNR que se está dispuesto a perder y se obtiene el número de bits N del convertidor. Luego, se deduce la frecuencia de conversión f_s necesaria. Se realiza un análisis matemático que se prueba, primero, mediante un ejemplo de simulación y, después, mediante dos experimentos prácticos. Los resultados se pueden aplicar de forma directa como criterio para elegir las dos características dinámicas más importantes de un circuito ADC y, en general, de un sistema de adquisición de datos.

2.9.2. Problemática

Cuando hay que elegir un convertidor A/D, y en general un sistema de adquisición de datos, para una aplicación determinada, aparecen las cuestiones de cuántos bits de resolución debe tener y cuál debe ser la frecuencia a la que debe convertir las señales [62].

Como primera aproximación, el número de bits necesarios se puede especificar teniendo en cuenta la mínima variación de la señal de entrada que se debe poder detectar, dicho valor se suele denominar Q . Este coeficiente se calcula dividiendo el rango total de entrada por el número de códigos de salida que se pueden obtener y se suele elegir el convertidor de forma que Q sea menor que la mitad de esa mínima variación que se quiere detectar. Este criterio para elegir el nivel de cuantificación es, según [61] similar al del teorema del muestreo. Este es el método utilizado más común y da buenos resultados desde un punto de vista de las características estáticas del ADC, pero no es suficiente cuando se quiere tener en cuenta la calidad de la señal, éste es el punto de vista dinámico.

Por un lado, la característica que define la calidad de la señal desde un punto de vista dinámico es la SNR. Por otro lado, el proceso de conversión, que implica el muestreo y la cuantificación, tiene dos efectos negativos sobre la SNR de la señal. El efecto del muestreo depende de la frecuencia de conversión f_s y el de la cuantificación de la resolución o número de bits N .

Por tanto, la SNR de la señal a la salida del ADC disminuye en función del f_s y N . Para ayudar a elegir estos parámetros es necesario relacionarlos por medio de reglas o ecuaciones que sean fáciles de aplicar. La cuestión es: ¿Qué porción de la SNR de la señal original se está dispuesto a perder en el

proceso de conversión? Para responder a esta pregunta, primero se relaciona la cantidad de SNR que se está dispuesto a perder con el número de bits N del convertidor. Luego, con el valor obtenido de N , se calcula la mínima frecuencia de conversión. De esta forma se optimizan la dimensión y la frecuencia de muestreo del convertidor.

Al estudiar estas relaciones se ha encontrado que, cuando se va aumentando la resolución del ADC, la SNR de la señal resultante a la salida también mejora, pero solo hasta un cierto valor de N , por encima del cual no se consigue mejorar el resultado. Por tanto, no se gana nada seleccionando un ADC con mejor resolución.

En cuanto a la frecuencia de conversión, se deduce que para optimizar el número de bits para conseguir una cierta calidad, es necesario utilizar algún criterio que de la frecuencia mínima de muestreo. Entonces, ese criterio, que debe ser más fuerte que el de Nyquist, debe garantizar que con una señal periódica se alcance la calidad deseada. De ese análisis, considerando que la señal siempre pierde algo de calidad, aunque sea pequeña (por ejemplo 0.1 dB), se obtendrán dos ecuaciones.

Para validar el análisis presentado, utilizando un ruido aleatorio uniformemente distribuido, para la señal de entrada y para el convertidor, se hacen diversas simulaciones y experimentos prácticos, tanto variando el número de bits del convertidor como la SNR de la señal.

Los resultados obtenidos se pueden aplicar directamente como criterio dinámico para elegir las características del convertidor y en general del DAS. Además, puede ser útil para resolver la siguiente cuestión: si la SNR de una señal analógica es desconocida y se convierte a un formato digital, ¿se podría deducir su SNR? Bien, si se conocen las características de la señal original, la SNR del ADC y se calcula la SNR de la señal convertida, entonces se puede deducir la SNR de la señal original.

En la siguiente sección se define en detalle el problema que se analiza y se presenta una solución. En la sección 2.9.5 se prueba con un ejemplo simulado y de forma experimental utilizando un DAS real que incluye los circuitos de acondicionamiento, los convertidores y la interfaz con el computador [23].

2.9.3. Deducción del método de selección

El problema planteado sigue el esquema de la Figura 2.33. Los datos de partida son la relación señal ruido (SNR_S) de la señal y el valor máximo de ella que se está dispuesto a perder (SNR_L). Con

estos datos se obtiene el número mínimo de bits N . Finalmente, con N y la frecuencia máxima de la señal de entrada se deduce la frecuencia mínima de conversión.

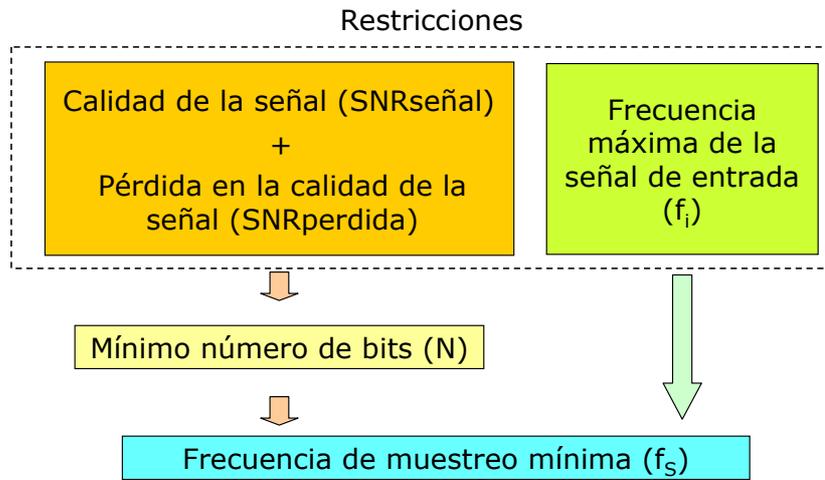


Figura 2.33. Información de entrada y de salida para seleccionar la frecuencia de muestreo f_s y la resolución N .

Para analizar el problema se sigue el diagrama indicado en la Figura 2.34, que consiste en dos bloques B_1 y B_2 conectados en serie y que se caracterizan por su relación señal ruido.

El bloque B_1 constituye la fuente de la señal original, que se denomina S_1 y que se compone de una información x y de un ruido r_1 , por tanto, la salida del generador B_1 es $S_1 = x + r_1$. Esta señal pasa por el bloque B_2 , el cual empeora la SNR de la señal ya que introduce el ruido r_2 . Para hacer el análisis se hacen dos suposiciones. La primera, es que la información x se mantiene inalterada después de pasar por los bloques. La segunda, que los ruidos r_1 y r_2 son considerados variables aleatorias independientes. Entonces, el ruido total a la salida del bloque B_2 es igual a la suma en cuadratura de los dos ruidos, o sea $(r_1^2 + r_2^2)^{1/2}$ [10] y la información de salida sigue siendo x . Con estas suposiciones la señal final se puede escribir como $S = x + (r_1^2 + r_2^2)^{1/2}$.

Para distinguir cómo se expresa la relación señal ruido, se utilizan las letras minúsculas (snr) cuando se expresa en una escala lineal, y en mayúsculas (SNR) cuando se expresa en escala logarítmica. Por tanto, la snr de B_1 es $snr_1 = x/r_1$ y la de B_2 es $snr_2 = x/r_2$. Cuando estas relaciones se expresan en escala logarítmica se tienen $SNR_1 = 20 \log(x/r_1)$ y $SNR_2 = 20 \log(x/r_2)$, ambas expresadas en dB.

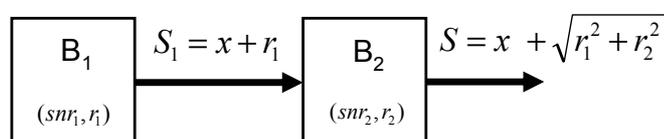


Figura 2.34. Diagrama de bloques de la fuente de señal (B_1) y de un proceso (B_2).

La *snr* total viene dada por la ecuación (2.25) y, teniendo en cuenta el ruido de cada bloque expresados por (2.26), la relación señal ruido total dependiente de la de cada bloque se obtiene a partir de (2.27). Esta ecuación se puede generalizar a más bloques por medio de (2.28). Habitualmente la relación señal ruido se expresa en escala logarítmica de acuerdo con la expresión (2.29). De esta forma, la *SNR* total de los dos bloques se expresa por (2.30). Esta ecuación se concreta, para el ejemplo mostrado en la Figura 2.34, por (2.31).

$$snr = \frac{x}{\sqrt{r_1^2 + r_2^2}} \quad (2.25)$$

$$r_1 = \frac{x}{snr_1} ; r_2 = \frac{x}{snr_2} \quad (2.26)$$

$$snr = \frac{x}{\sqrt{\frac{x^2}{snr_1^2} + \frac{x^2}{snr_2^2}}} = \frac{snr_1 \cdot snr_2}{\sqrt{snr_1^2 + snr_2^2}} \quad (2.27)$$

$$snr = \frac{\prod_{m=1}^2 snr_m}{\sqrt{\sum_{m=1}^2 snr_m^2}} \quad (2.28)$$

$$SNR = 20 \cdot \log snr ; snr = 10^{\frac{SNR}{20}} \quad (2.29)$$

$$SNR = 20 \cdot \log \frac{\prod_{m=1}^2 10^{\frac{SNR_m}{20}}}{\sqrt{\sum_{m=1}^2 10^{\frac{SNR_m}{20} \cdot 2}}} \quad (2.30)$$

$$SNR = 20 \cdot \log \frac{10^{\sum_{m=1}^2 \frac{SNR_m}{20}}}{\sqrt{\sum_{m=1}^2 10^{\frac{SNR_m}{10}}}} = 20 \cdot \log \frac{10^{\frac{SNR_1 + SNR_2}{20}}}{\sqrt{10^{\frac{SNR_1}{10}} + 10^{\frac{SNR_2}{10}}}} \quad (2.31)$$

2.9.4. Aplicación a la selección del número de bits del ADC

El análisis anterior se puede aplicar directamente al problema de calcular el mínimo número de bits necesarios para limitar la relación señal ruido que se pierde en el proceso de conversión. Para ello, de acuerdo al esquema de la Figura 2.35, el bloque B₁ de la Figura 2.34 se sustituye por un generador

G, que representa el proceso que proporciona la señal a convertir, y el bloque B₂ se cambia por el circuito convertidor ADC. Para simplificar, se sigue la terminología de la Tabla 2.6.

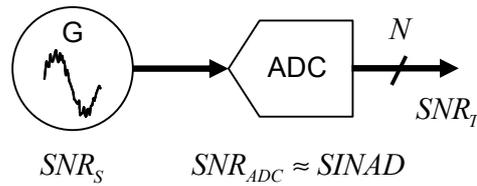


Figura 2.35. Diagrama de los bloques generador y convertidor.

Tabla 2.6. Términos utilizados para el análisis de la relación señal ruido.

Termino	Definición
N	Número de bits
snr_S	SNR de la señal de entrada
snr_{ADC}	SNR del ADC
snr_L	SNR que se pierde de la señal
snr_T	SNR total a la salida
SNR_S	SNR de la señal de entrada en dB
SNR_{ADC}	SNR del ADC en dB
SNR_L	SNR que se pierde de la señal en dB
SNR_T	SNR total a la salida en dB

Sustituyendo en (2.31) los términos de la Tabla 2.6 se obtiene la expresión de la SNR total dada por (2.32). En esta ecuación el valor de SNR_{ADC} se puede sustituir por (2.33). Así, se obtiene la SNR_T en función de la SNR_S y del número de bits N , tal y como se expresa en (2.34).

$$SNR_T = 20 \cdot \log \frac{10^{\frac{SNR_S + SNR_{ADC}}{20}}}{\sqrt{10^{\frac{SNR_S}{10}} + 10^{\frac{SNR_{ADC}}{10}}}} \quad (2.32)$$

$$SNR_{ADC} = 6.02 \cdot N + 1.76 \quad (2.33)$$

$$SNR_T = 20 \cdot \log \frac{10^{\frac{SNR_S + 6.02 \cdot N + 1.76}{20}}}{\sqrt{10^{\frac{SNR_S}{10}} + 10^{\frac{6.02 \cdot N + 1.76}{10}}}} \quad (2.34)$$

De acuerdo con (2.34) la Figura 2.36 representa cómo la calidad de la señal convertida (indicada por SNR_S) mejora cuando el número de bits (N) del convertidor aumenta. En este ejemplo, la señal de entrada tiene una SNR (SNR_S) igual a 50 dB (gráfica azul). La SNR total (SNR_T) aumenta significativamente hasta cierto N (alrededor de los 9-10 bits). Después de este punto, incluso si se aumenta N , no se consigue mejorar la SNR_T .

$$\log snr_T = \log \frac{snr_S}{snr_L} \quad snr_T = \frac{snr_S}{snr_L} \quad (2.37)$$

$$\frac{snr_S}{snr_L} = \frac{snr_S \cdot snr_{ADC}}{\sqrt{snr_S^2 + snr_{ADC}^2}} \quad ; \quad snr_{ADC}^2 = \frac{snr_S^2 + snr_{ADC}^2}{snr_L^2} \quad ; \quad snr_{ADC}^2 \cdot \left(1 - \frac{1}{snr_L^2}\right) = \frac{snr_S^2}{snr_L^2}$$

$$snr_{ADC} = \sqrt{\frac{snr_S^2}{snr_L^2 - 1}} \quad (2.38)$$

$$6.02 \cdot N + 1.76 = 20 \cdot \log \sqrt{\frac{snr_S^2}{snr_L^2 - 1}}$$

$$20 \cdot N \cdot \log 2 + 20 \cdot \log \frac{\sqrt{6}}{2} = 20 \cdot \log \sqrt{\frac{snr_S^2}{snr_L^2 - 1}} \quad (2.39)$$

$$\log \left(2^N \cdot \frac{\sqrt{6}}{2} \right) = \log \sqrt{\frac{snr_S^2}{snr_L^2 - 1}} \quad ; \quad 2^N = \frac{2}{\sqrt{6}} \cdot \sqrt{\frac{snr_S^2}{snr_L^2 - 1}}$$

$$N = \log_2 \frac{2}{\sqrt{6}} \cdot \sqrt{\frac{snr_S^2}{snr_L^2 - 1}} = \frac{1}{2} \cdot \log_2 \left(\frac{2}{3} \cdot \frac{snr_S^2}{snr_L^2 - 1} \right) = \frac{1}{2} \cdot \left[\log_2 \frac{2}{3} + \log_2 snr_S^2 - \log_2 (snr_L^2 - 1) \right] =$$

$$N = \frac{1}{2} \cdot \left[\log_2 \frac{2}{3} + \log_2 \left(10^{\frac{SNR_S}{10}} \right) - \log_2 \left(10^{\frac{SNR_L}{10}} - 1 \right) \right] =$$

$$N = \frac{1}{2} \cdot \left[\log_2 \frac{2}{3} + \frac{SNR_S}{10} \cdot \log_2 10 - \log_2 \left(10^{\frac{SNR_L}{10}} - 1 \right) \right] \quad (2.40)$$

Entonces, si se conocen la SNR de la señal a convertir y la SNR del convertidor, sólo hay que aplicar (2.40) para deducir N. Sin embargo, es importante destacar que en los sistemas reales se utiliza el ENOB (*Effective Number of Bits*) en vez de N. Por tanto, la ecuación (2.40) se debe cambiar por la (2.41).

$$ENOB = \frac{1}{2} \cdot \left[\log_2 \frac{2}{3} + \frac{SNR_S}{10} \cdot \log_2 10 - \log_2 \left(10^{\frac{SNR_L}{10}} - 1 \right) \right] \quad (2.41)$$

En la Figura 2.37 se representa el valor del número de bits N que se debe tomar cuando se conocen la SNR de la señal y la máxima SNR que se puede perder. Dicha figura tiene tres curvas que corresponden con tres valores diferentes de la SNR que se pierde (0.1, 1.0 y 3.0 dB). Por ejemplo, si se

parte de una señal con una SNR de unos 70 dB, y se desea que no se degrade en más de 0.1 dB, entonces, se debe elegir un ADC con un ENOB mínimo de 14 bits.

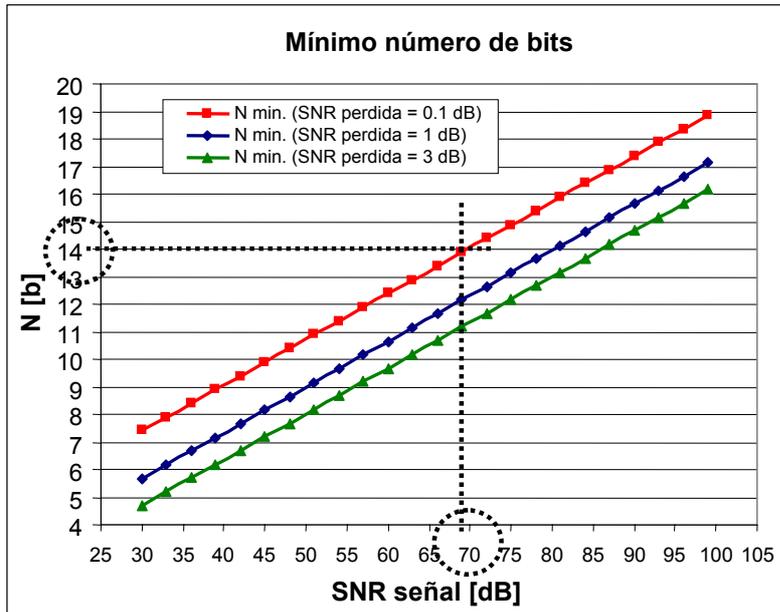


Figura 2.37. Mínimo número de bits necesarios frente a la SNR_L que se pierde para una señal de entrada con una cierta SNR_S .

Una vez obtenido el mínimo número de bits o la mínima resolución del convertidor, se debe obtener la frecuencia óptima de muestreo, es decir, la frecuencia a la que todos los códigos tengan, al menos, una muestra. De esta forma se evita la pérdida de algún código, debido a que el cambio en la amplitud de la señal entre dos instantes de muestreo es mayor que el incremento correspondiente a dos códigos consecutivos (Figura 2.38), y se evita también el obtener varias muestras consecutivas con el mismo código, como ocurre en el ejemplo de la Figura 2.39.

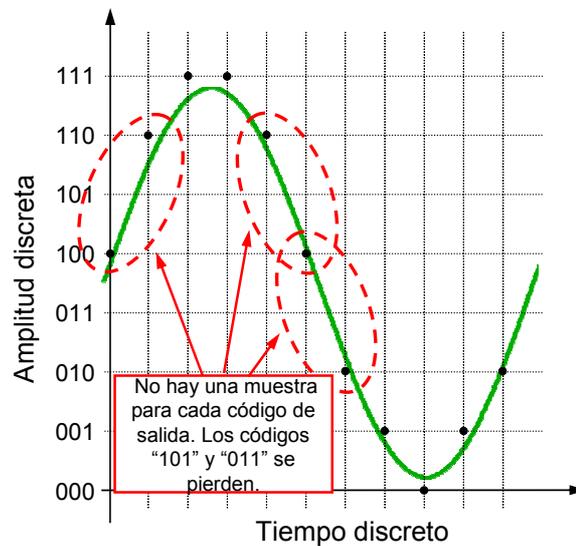


Figura 2.38. Señal muestreada a una frecuencia demasiado baja ($f_s = 11 \cdot f_i$).

La frecuencia óptima corresponde con la Figura 2.40, en ésta se aprecia cómo en la parte en la que la pendiente es máxima hay el mínimo número de muestras, o sea, una, lo que garantiza que se obtiene, al menos, un código por cada instante de muestreo.

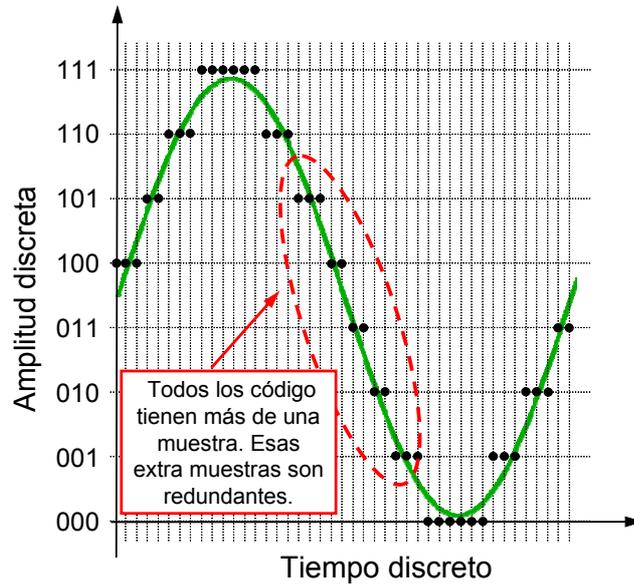


Figura 2.39. Señal muestreada a una frecuencia demasiado alta ($f_s = 33 \cdot f_i$).

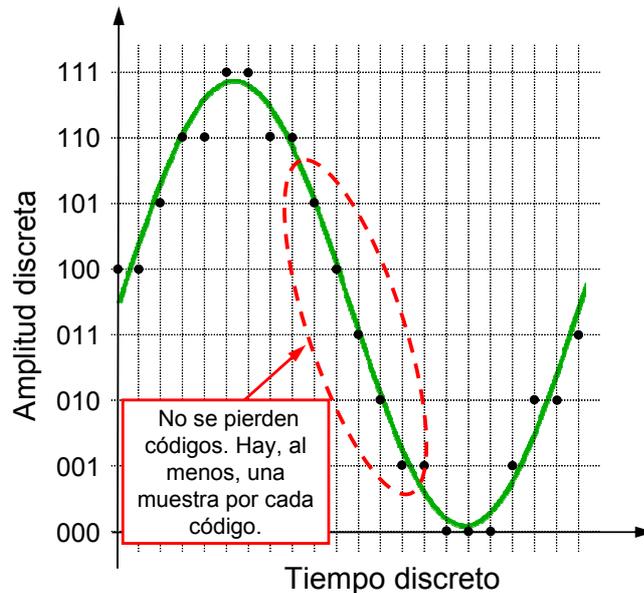


Figura 2.40. Señal muestreada a una frecuencia óptima ($f_s = 22 \cdot f_i$).

Para obtener este valor de frecuencia óptima se debe partir de la máxima derivada de tensión que se desea tener en cuenta cuando se discretiza la señal. Si se parte de una señal compuesta por diferentes armónicos, se debe elegir el de mayor frecuencia que queramos considerar. En cualquier caso se debe tener en cuenta la máxima derivada de tensión. Para realizar el análisis se supone que la señal representada por (2.42) es la de mayor frecuencia f_i y que su amplitud es de $A_0 = FS/2$. Derivando (2.42) dos veces e igualando el resultado a cero, se obtiene el punto de máxima pendiente (2.44), que

corresponde con el cruce por cero de la señal. Substituyendo este punto en (2.43) se obtiene la expresión (2.45) de la derivada máxima.

$$v(t) = A_0 \cdot \sin(\omega \cdot t) \quad (2.42)$$

$$\frac{dv(t)}{dt} = A_0 \cdot \omega \cdot \cos(\omega \cdot t) \quad (2.43)$$

$$\frac{d^2v(t)}{dt} = A_0 \cdot \omega^2 \cdot \sin(\omega \cdot t) = 0 \quad ; \quad \omega \cdot t = 0, \pi, 2\pi \dots \quad (2.44)$$

$$\left. \frac{dv(t)}{dt} \right)_{\max} = A_0 \cdot \omega = A_0 \cdot 2 \cdot \pi \cdot f_i \quad (2.45)$$

$$f_s \geq \frac{2^{N-1}}{FS} \cdot \frac{FS}{2} \cdot 2 \cdot \pi \cdot f_i \quad (2.46)$$

Ahora, la pendiente de la señal se compara con la máxima pendiente que el convertidor puede detectar (Figura 2.41). Este valor debe ser mayor o igual que el de la expresión (2.45), entonces, la ecuación general será (2.46), la cual se puede simplificar a (2.47).

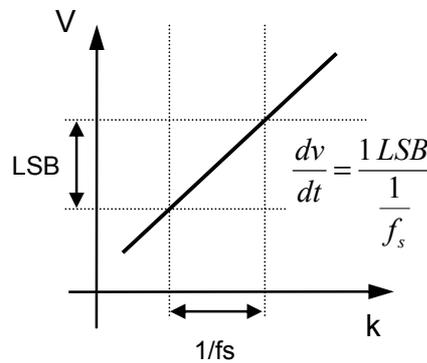


Figura 2.41. Máxima derivada de tensión de la señal.

En el ejemplo mostrado en la Figura 2.38, la Figura 2.39 y la Figura 2.40, el número de bits es 3, y el número de muestras por periodo de la señal es de 11, 33 y 22 respectivamente. Aplicando (2.47) se obtiene la relación (2.48), que demuestra, como era de esperar, que el valor de 22 muestras por periodo es el valor optimo.

$$f_s \geq (2^N - 1) \cdot \pi \cdot f_i \quad (2.47)$$

$$f_s = (2^3 - 1) \cdot \pi \cdot f_i = 21.98 \cdot f_i \quad (2.48)$$

2.9.5. Resultados de simulación y experimentales

La Figura 2.42 muestra una señal que tiene una SNR de 15 dB y que se muestrea a una frecuencia $f_s=5f_i$ mediante un ADC de 4 bits. El ADC tiene una SNR igual a 25 dB, por tanto, está 10 dB por encima de la SNR de la señal. Con los datos de simulación la pérdida en la SNR al muestrear la señal es de 0.137 dB que, de acuerdo con la regla presentada, se puede considerar un resultado correcto porque esta cerca del valor teórico de 0.1 dB.

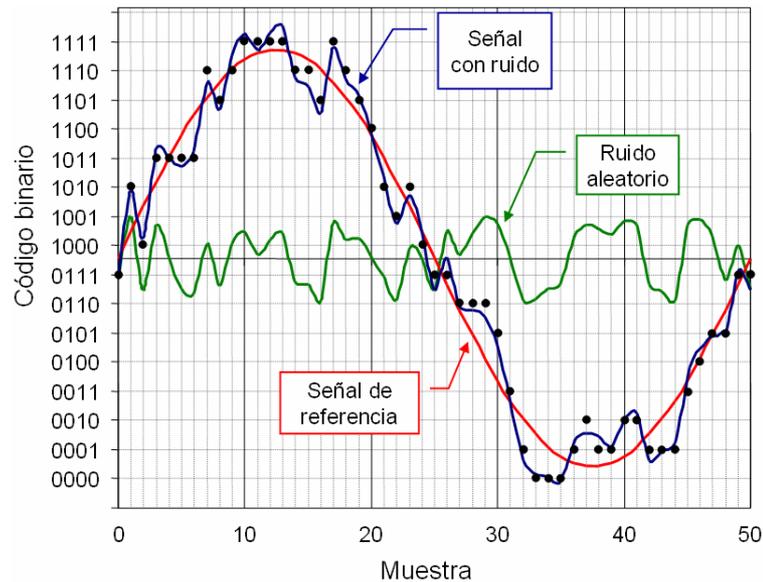


Figura 2.42. Ejemplo de una señal con 15 dB de SNR muestreada con un ADC de 4 bits.

Para los resultados, a continuación se presentan dos experimentos prácticos. Para ello, se ha utilizado un sistema de adquisición reconfigurable que incluye los circuitos de acondicionamiento, los convertidores y una FPGA conectada a un computador a través del USB [18], [23].

En el primer experimento se han adquirido y analizado un conjunto de señales compuestas de un tono de 24 kHz y por un ruido uniformemente distribuido (Figura 2.43).

En el segundo experimento se ha fijado la SNR de la señal a adquirir en 25 dB (Figura 2.45) y la SNR del ADC se ha variado entre 20 y 42 dB conectando los bits necesarios (entre 3 y 8) (Figura 2.44). Cada vez que se cambia el número de bits del ADC, se mide su SNR adquiriendo una sinusoidal pura (desconectando el ruido). Después, el ruido se añade a dicha señal y el resultado se adquiere y analiza para deducir la SNR final (Figura 2.44).

En ambos experimentos la frecuencia de muestreo se ha mantenido en 24 MHz y, de acuerdo con (2.47) y teniendo en cuenta que el máximo número de bits utilizados en los experimentos es de 8, el tamaño M del registro de datos se estableció en 1 000 muestras, ya que la señal es de 24 kHz.

En la Figura 2.43 se muestra cómo la SNR total no mejora cuando la SNR de la señal está 10 dB por encima de la del ADC. Es decir, según la curva rosa la SNR del ADC es de unos 42 dB y cuando la señal tiene una SNR mayor de unos 51 dB la salida ya no mejora más. Por tanto, el ADC está limitando la SNR resultante. Recíprocamente, la Figura 2.44 muestra que cuando la SNR del ADC (eje horizontal) está 10 dB por encima de la de la señal de entrada (trazo azul), en este caso igual a 25 dB (una sinusoidal de 8 Vpp con un ruido blanco de 162 mV rms), la SNR final (trazo rojo) no mejora y, por tanto, muestrear con una resolución mayor no tiene sentido.

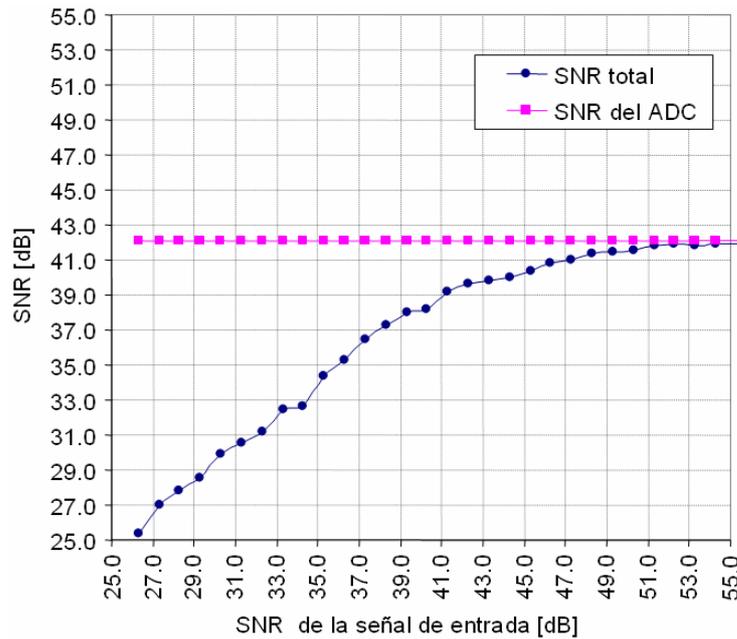


Figura 2.43. Resultado de convertir, con una SNR del ADC fijada en 42 dB, señales con distintas SNR.

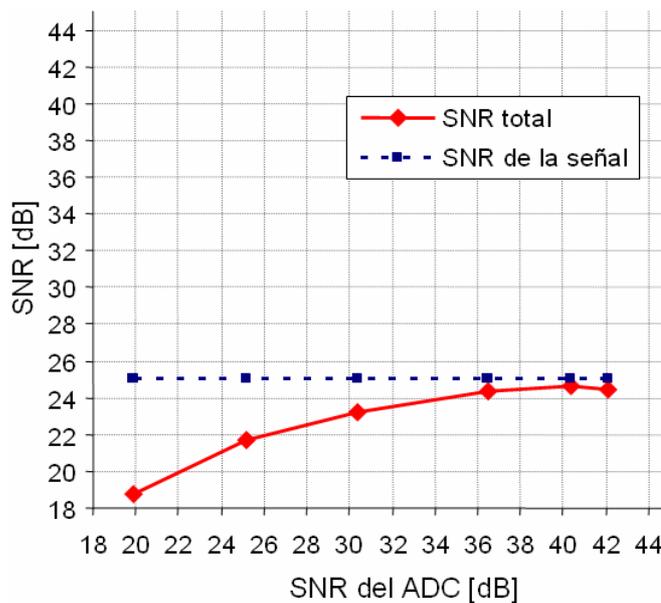


Figura 2.44. Resultado de muestrear una señal con una SNR de 25 dB para distintos valores de la SNR del convertidor.

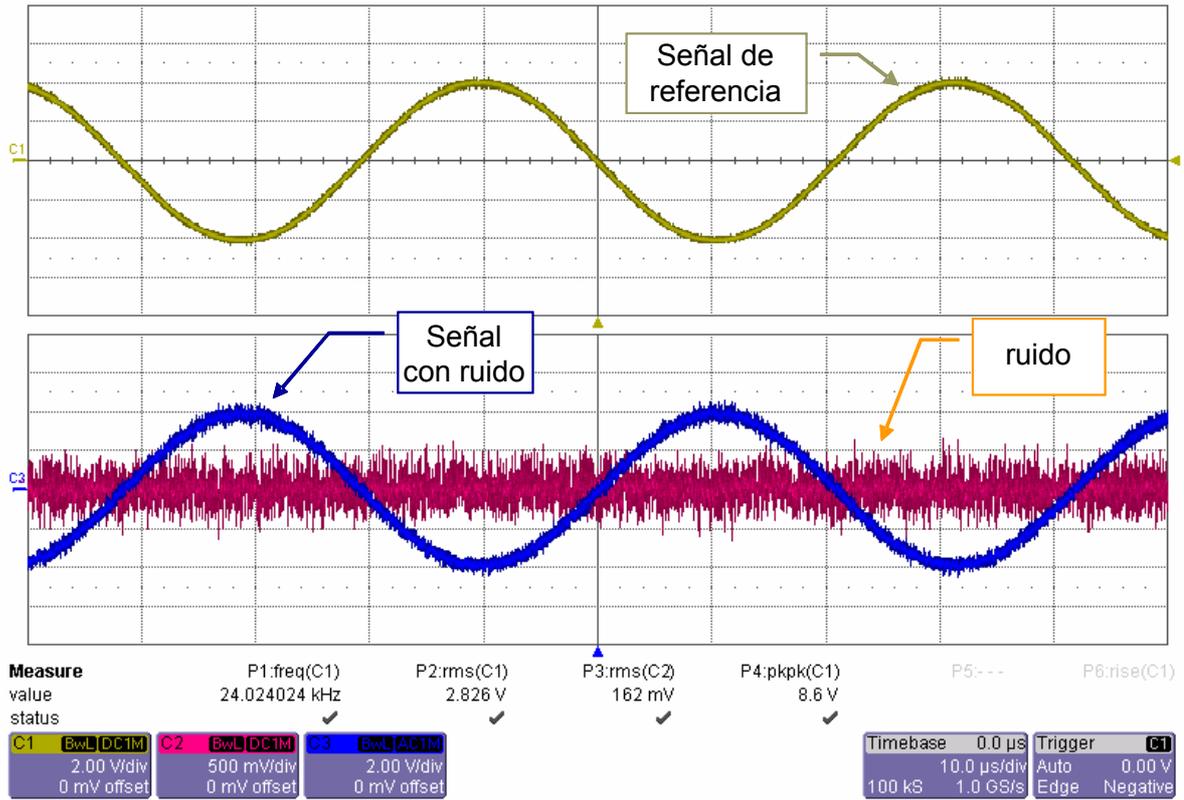


Figura 2.45. Vista del osciloscopio digital. El canal C1 representa la señal sinusoidal, el canal C2 el ruido que se le añade y el canal C3 la señal resultante de 25 dB de SNR.

CAPÍTULO 3. LOS CIRCUITOS CONVERTIDORES A/D SIGMA-DELTA

3.1. Introducción

El objetivo de este capítulo es definir cualitativamente y cuantitativamente, de forma analítica, cómo funciona un convertidor sigma-delta [28], [63]-[65], cuyo esquema básico se muestra en la Figura 3.1, comenzando por su núcleo, que es el modulador, y terminando por las etapas de filtrado y diezmado [66].

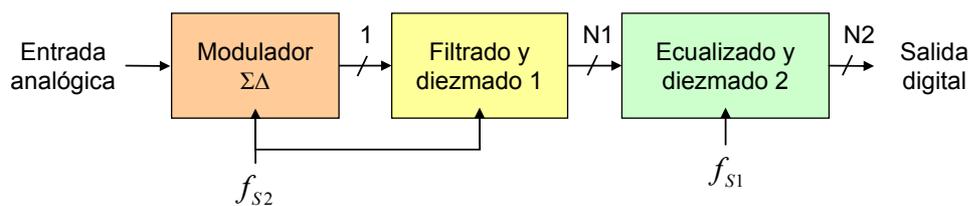


Figura 3.1. Diagrama de bloques básico de un convertidor de sobremuestreo sigma-delta.

En los sistemas electrónicos digitales que procesan señales analógicas son necesarios los circuitos ADC, cuya resolución y velocidad limita las prestaciones de dichos sistemas. Para conseguir altas resoluciones se utilizan técnicas de procesamiento de las muestras, que se toman a una frecuencia mucho mayor. Esta técnica, que se denomina sobremuestreo, se basa en muestrear a una frecuencia mucho mayor que la de la señal de entrada y, a continuación, filtrarla y diezmarla digitalmente. En este proceso se aumenta la resolución y se disminuye la frecuencia final de conversión, es decir, se intercambia resolución por velocidad. Sin embargo, esto no es la panacea, pues para conseguir un pequeño aumento en la resolución es necesario disminuir mucho la frecuencia de conversión, por lo que en la práctica esta técnica no se utiliza. El principio del sobremuestreo se ha aplicado cuando se ha combinado, como se muestra en la Figura 3.2, con una realimentación del error de conversión (o modulación) en el mismo circuito [67], [68]. El objetivo de dicha realimentación es cambiar la forma del espectro del ruido de cuantificación de forma que se disminuya todo lo posible en la banda de la señal [69].

De esta forma al combinar el sobremuestreo con la realimentación se consigue disminuir mucho el ruido en la banda de interés y, por tanto, aumentar la resolución. El circuito más popular que implementa estas técnicas es el modulador sigma-delta.

Los convertidores de sobremuestreo de tipo sigma-delta tienen las siguientes ventajas:

1. No es imprescindible el uso de un circuito S&H a su entrada.
2. Necesitan un filtro *antialiasing* mucho menos selectivo que en los demás ADC, por lo que es mas sencillo de implementar. Normalmente una simple red RC es suficiente.
3. Es posible alcanzar resoluciones elevadas, mayores que 24 bits.
4. El circuito analógico es mucho más sencillo.
5. La calidad de la señal convertida depende menos de las características no lineales de los circuitos analógicos utilizados.
6. Facilita el acoplamiento al procesador cuando se utiliza un modulador independiente, ya que sólo tiene un bit de salida, sobre todo cuando se necesita aislar eléctricamente la parte analógica de la parte digital [70].

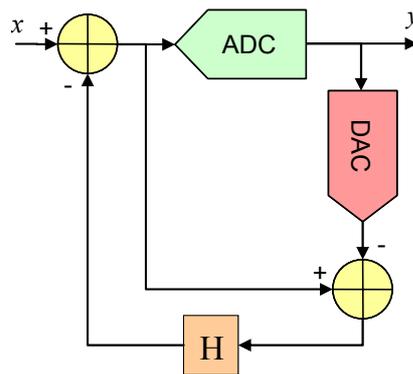


Figura 3.2. Principio de la realimentación negativa del error para aumentar la resolución en los convertidores de sobremuestreo mediante un post-filtrado.

El esquema de bloques de la Figura 3.2 se puede sustituir por el modelo discreto de la Figura 3.3, y éste, a su vez, por el de la Figura 3.4, más simplificado. Este cambio se deduce a partir de las ecuaciones (3.1) y (3.2), de las que se obtiene (3.3), según la cual, si se desea que el ruido se filtre con un filtro paso-alto de primer orden, cuya ecuación se corresponde con (3.4), entonces, la función H que debemos implementar se obtiene restando de la unidad la ecuación del filtro, tal y como representa (3.5). Para simplificar todavía más la solución se fija la frecuencia de corte del filtro en un cuarto de la de muestreo, de esta forma el coeficiente *a* del filtro se anula y queda la ecuación final (3.6).

$$\xi = X - H \cdot (Y - \xi) \quad (3.1)$$

$$Y = \xi + E \quad (3.2)$$

$$Y(z) = X(z) + E(z) \cdot (1 - H) \tag{3.3}$$

$$(1 - H) = \frac{1 - a}{2} \cdot \frac{1 - z^{-1}}{1 - a \cdot z^{-1}} \tag{3.4}$$

$$H = 1 - \frac{1 - a}{2} \cdot \frac{1 - z^{-1}}{1 - a \cdot z^{-1}} = \frac{(1 - a) \cdot z^{-1} + (1 - a)}{2 \cdot (1 - a \cdot z^{-1})} \tag{3.5}$$

$$H(z) = \frac{z^{-1} + 1}{2} = \frac{1 + z}{2 \cdot z} \tag{3.6}$$

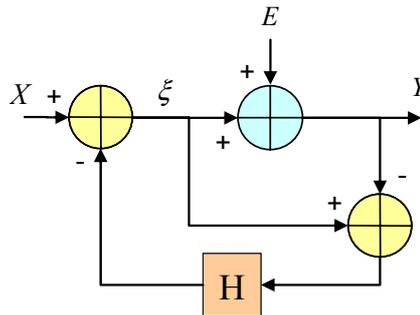


Figura 3.3. Modelo del modulador básico.

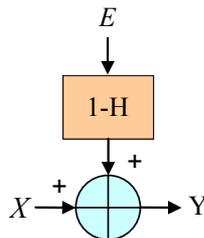


Figura 3.4. Modelo simplificado del modulador básico.

En los siguientes apartados se explica cómo ha evolucionado el modulador, se profundiza en la teoría de sobremuestreo, y se presentan los moduladores básicos y las etapas de post-filtrado y diezmado que, junto con el modulador, forman el convertidor A/D. En el Apartado 5.3 del Capítulo 5 se vuelve a la base de estos moduladores para presentar una alternativa de diseño, y se presentan los resultados de las pruebas de laboratorio realizadas con un circuito real. Finalmente, en el Apartado 6.6 del Capítulo 6 se presenta una metodología que aplica los conceptos teóricos y prácticos estudiados a la enseñanza de los convertidores sigma-delta.

3.2. Origen y principio de funcionamiento del modulador sigma-delta

El modulador sigma-delta no se inventó para convertir señales analógicas a digital, sino que ésta fue una aplicación que se le dio, y que con el tiempo ha cobrado más importancia. La idea de

cuantificar una señal analógica mediante una señal digital, o sea, discreta, de un solo bit surgió por la necesidad de mejorar las comunicaciones, tanto para alcanzar mayores distancias como para mejorar la relación señal-ruido. Los primeros avances en este terreno quedaron reflejados en forma de patentes de las principales empresas americanas de telecomunicaciones [71]. Con esta perspectiva, el modulador sigma-delta tiene su antecesor en el modulador delta [63] (Figura 3.5), que se basa en cuantificar los cambios en la señal de entrada muestra a muestra, en vez de cuantificar el valor absoluto de cada una. A continuación se verá cómo a partir de este modulador se llega al circuito del convertidor [9].

El término del error $x(t) - \hat{x}(t)$ de la predicción actual se cuantifica y se usa para hacer la siguiente predicción. Los moduladores Delta, además, muestran sobrecarga de pendiente para señales de entrada que varían rápidamente (distorsión por sobrecarga de pendiente, el otro error introducido es el ruido granular).

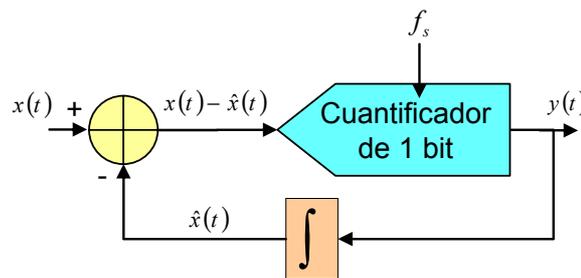


Figura 3.5. Modulador Delta.

Para recuperar la señal $x(t)$, a partir de la salida modulada $y(t)$, se sigue el esquema de la Figura 3.6, que incluye un integrador (primer paso del demodulador) y luego un filtro paso-bajo.

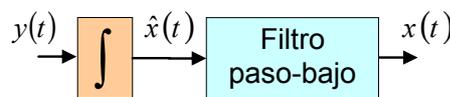


Figura 3.6. Demodulador Delta.

Si se incluye el integrador en la salida del modulador, se obtiene el sistema de la Figura 3.7 y, como es lineal, el integrador se puede trasladar a la entrada y, entonces, los dos integradores que están antes del bloque sumador se pueden sustituir por un único integrador situado a su salida, como se muestra en la Figura 3.8. Del hecho de poner el integrador, cuyo símbolo es la letra Σ , a la salida del modulador delta, cuyo símbolo es la letra Δ , se le dio a este circuito el nombre de modulador Σ - Δ (sigma-delta).

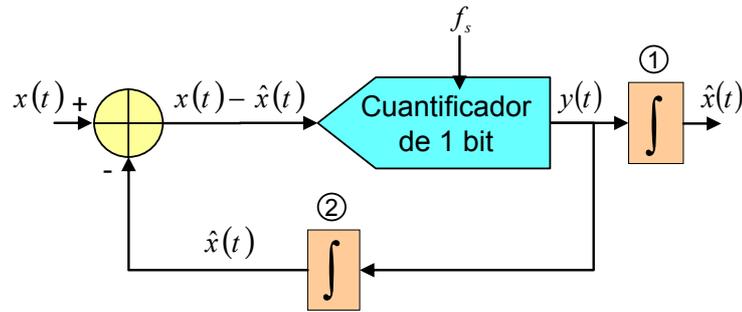


Figura 3.7. Modulador delta más la primera etapa del demodulador.

La modulación sigma-delta es una técnica que traslada el ruido de cuantificación fuera de la banda base por medio del filtrado del error, que se realiza de forma sencilla gracias a la realimentación. Además, es tolerante a la variación en los parámetros de los circuitos y una sencilla teoría lineal proporciona una buena descripción de esta modulación. En la Figura 3.9 se muestra el modelo lineal de modulador de primer orden, que es el más sencillo, la entrada $E(s)$ representa el ruido de cuantificación.

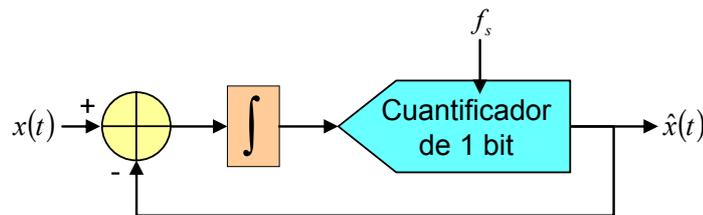


Figura 3.8. Esquema simplificado del modulador sigma-delta básico.

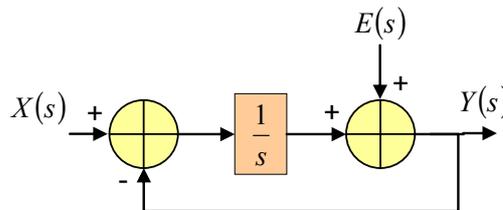


Figura 3.9. Modelo lineal del modulador sigma-delta.

3.3. Sobremuestreo y conformado del ruido

Los cuantificadores de sobremuestreo cambian el espectro del ruido desplazándolo fuera de la banda de interés, lo cual es equivalente a aumentar la resolución efectiva de salida que se puede alcanzar cuando se lleva a cabo un post filtrado con diezmado. Por tanto, los convertidores que utilizan técnicas de sobremuestreo proporcionan una resolución, cambiándola por velocidad, mayor que los convertidores de muestreo de Nyquist. Además, con el sobremuestreo, el filtro *antialiasing* es mucho más fácil de realizar, ya que puede ser mucho menos selectivo [72]-[73].

Un cuantificador se puede representar por un modelo como el de la Figura 3.10, que consiste en un bloque sumador que añade a la señal a convertir $x(n)$ un ruido blanco $e(n)$ proporcionando la señal de salida $x_Q(n)$ [74].

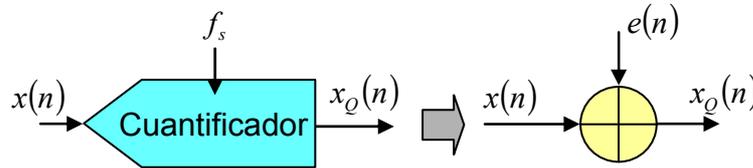


Figura 3.10. Modelo de ruido aditivo del cuantificador.

La suposición de que $e(n)$ es un ruido blanco significa que tiene un espectro de frecuencia plano. Concretamente, la potencia total media σ_e^2 de $e(n)$, definida por (3.7), esta uniformemente distribuida sobre la banda de Nyquist $[-f_s/2, +f_s/2]$, como se muestra en al Figura 3.11. Entonces, la potencia por unidad del intervalo de frecuencia o densidad de potencia espectral de $e(n)$, viene dada por (3.8).

$$\sigma_e^2 = E[e^2(n)] = \frac{Q^2}{12} \quad ; \quad Q = \frac{FSR}{2^N} \tag{3.7}$$

$$S_{ee}(f) = \frac{\sigma_e^2}{f_s} = \frac{Q^2}{12 \cdot f_s}, \text{ para } -\frac{f_s}{2} \leq f \leq \frac{f_s}{2} \tag{3.8}$$

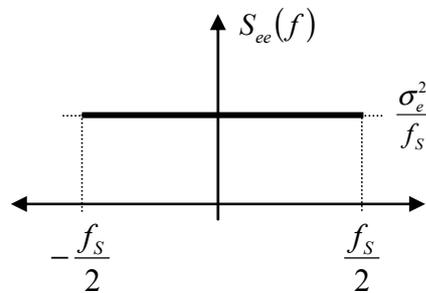


Figura 3.11. Espectro del ruido de cuantificación.

En un circuito ADC el ruido *rms* de cuantificación se mide sobre toda la banda de Nyquist, es decir, desde *dc* hasta $f_s/2$. En muchas aplicaciones, la señal de interés ocupa una pequeña banda B, y si se utiliza un filtrado digital para rechazar las componentes de ruido fuera de dicha banda, entonces se debe incluir un factor de corrección en la ecuación general de la SNR denominado ganancia de resolución. Este proceso en el que se muestra una señal a una frecuencia superior a dos veces su ancho de banda se denomina sobremuestreo [75].

La relación de sobremuestreo, o sea, la relación entre la frecuencia de muestreo y la de la banda de la señal se denomina mediante las siglas OSR (*Over Sampling Ratio*).

En un principio, el sobremuestreo fue tratado como una técnica que disminuía la necesidad de utilizar filtros de elevada calidad y fue utilizado para intercambiar velocidad por bits. En otras palabras, si se muestrea a una frecuencia elevada, se puede utilizar un cuantificador de poca resolución, de forma que cada muestra es menos exacta pero hay muchas de ellas y, procesándolas, se recupera la exactitud perdida.

La idea es similar a la que lleva a cabo múltiples medidas de una cierta cantidad, por ejemplo x , tomando σ_x^2 como el error medio cuadrático de dicha medida, si se hacen un número m de medidas independientes de la señal x , siguiendo la ley de los número grandes, el error de medida se reducirá a σ_x^2 / m , mejorando la exactitud de la medida.

De forma similar, si se aumenta σ_x^2 , haciendo cada medida individual menos precisa, es posible mantener el mismo nivel de calidad si se aumenta el número de medidas m de forma que se mantenga la relación σ_x^2 / m constante.

Considerando dos casos, uno con una frecuencia de muestreo f_{S1} y con N_1 bits por muestra, y otro con una frecuencia f_{S2} mayor y con N_2 bits por muestra, la cantidad $OSR = f_{S2} / f_{S1}$ se denomina relación de sobremuestreo y suele ser un valor entero. Asumiendo el mismo rango de entrada FS para los dos cuantificadores, se tienen los dos siguientes valores del ancho de cuantificación: $Q_1 = FS / 2^{N_1}$, $Q_2 = FS / 2^{N_2}$, y los siguientes potencias del ruido: $\sigma_{e1}^2 = Q_1^2 / 12$, $\sigma_{e2}^2 = Q_2^2 / 12$. Para mantener la misma calidad en los dos casos, es necesario que las dos densidades espectrales sean iguales, es decir: $\sigma_{e1}^2 / f_{S1} = \sigma_{e2}^2 / f_{S2}$, entonces: $\sigma_{e1}^2 = f_{S1} \cdot (\sigma_{e2}^2 / f_{S2}) = \sigma_{e2}^2 / OSR$, de esta ecuación se deduce la expresión (3.9).

Finalmente, resolviendo (3.9) se llega a la expresión (3.10) de la ganancia de resolución, la cual indica que se consigue aumentar la resolución en medio bit cada vez que se doble la relación de sobremuestreo OSR. Mediante la ecuación (3.11) se obtiene ΔN en función de un logaritmo decimal.

Además, esta mejora en la resolución, que se muestra en la Figura 3.12, también supone un aumento de la relación señal ruido, que está dada por (3.12) [8].

$$OSR = \frac{\sigma_{e1}^2}{\sigma_{e2}^2} = \frac{Q_2^2/12}{Q_1^2/12} = \frac{Q_2^2}{Q_1^2} = \left(\frac{FS \cdot 2^{-N2}}{FS \cdot 2^{-N1}} \right)^2 = 2^{2(N1-N2)} = 2^{2\Delta N} \quad (3.9)$$

$$\Delta N = 0.5 \cdot \log_2 OSR \quad (3.10)$$

$$\Delta N = 0.5 \cdot \frac{\log_{10} OSR}{\log_{10} 2} = 0.5 \cdot \frac{\log_{10} OSR}{0.30103} = 1.66096 \cdot \log_{10} OSR \quad (3.11)$$

$$SNR = 6.02 \cdot (N + \Delta N) + 1.76 = 6.0259 \cdot N + 1.76 + 10 \cdot \log_{10} OSR \quad (3.12)$$

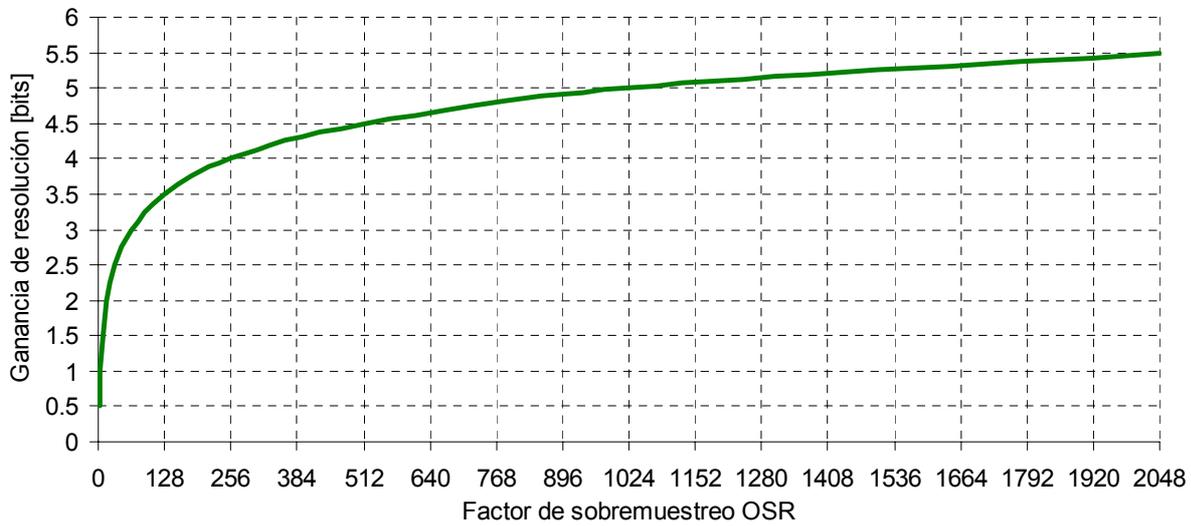


Figura 3.12. Ganancia de resolución debida al sobremuestreo.

Si se limita la frecuencia de las señales a convertir a f_{S1} (ver la zona sombreada de la Figura 3.13), y se muestra a f_{S2} , para hacer efectiva la ganancia de resolución, debida al sobremuestreo $OSR = f_{S2} / f_{S1}$, y que está dada por (3.12), se necesita un filtro paso-bajo con una banda de paso f_{S1} .

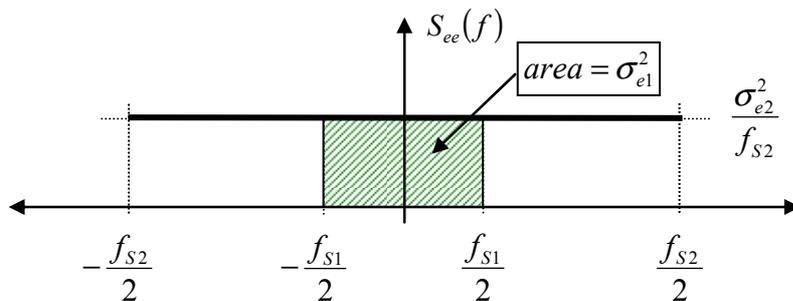


Figura 3.13. Espectro del ruido del cuantificador a dos frecuencias de conversión.

A pesar de que la técnica de sobremuestreo consigue mejorar la resolución, resulta poco eficiente porque necesita un factor de sobremuestreo muy alto. Por ejemplo, para conseguir una resolución de 16 bits a partir de un convertidor de 1 bit, o sea, un ΔN de 15, es necesario un valor de $OSR=2^{30}$, lo cual resulta impracticable.

Si en vez de la forma de la Figura 3.13 el cuantificador tuviese otra forma, por ejemplo, una en la que el nivel de ruido a bajas frecuencias fuese menor, entonces, mejoraría el rendimiento del sobremuestreo. Para llegar a esta solución, se recurre a un cuantificador que cambia la forma del espectro del ruido. Esto se consigue, como se muestra en la Figura 3.14, filtrando el ruido blanco $e(n)$ mediante el filtro $H_{NS}(f)$ que proporciona otro a su salida, denominado $\varepsilon(n)$, con el espectro deseado.

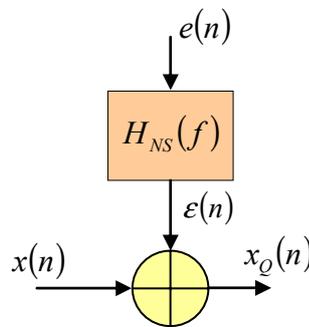


Figura 3.14. Modelo del cuantificador con filtrado del ruido.

Como el ruido tiene un espectro plano, el espectro de la secuencia de salida $\varepsilon(n)$ del filtro toma la misma forma que dicho filtro $H_{NS}(f)$ y, por tanto, su densidad de potencia espectral dada por (3.13) ya no es plana.

$$S_{\varepsilon\varepsilon}(f) = |H_{NS}(f)|^2 \cdot S_{ee}(f) = |H_{NS}(f)|^2 \cdot \frac{\sigma^2}{f_S} \quad (3.13)$$

La potencia del ruido en un sub-intervalo $[f_a, f_b]$ se obtiene integrando (3.13) sobre dicho sub-intervalo, tal y como se muestra en (3.14).

$$\text{Potencia en la banda } [f_a, f_b] = \int_{f_a}^{f_b} S_{\varepsilon\varepsilon}(f) \cdot df = \frac{\sigma^2}{f_S} \cdot \int_{f_a}^{f_b} |H_{NS}(f)|^2 \cdot df \quad (3.14)$$

En la Figura 3.15 se muestra el proceso descrito. Inicialmente el ruido $e(n)$ tiene una respuesta plana pero, como $H_{NS}(f)$ funciona como un filtro paso-alto, se consigue que la mayor parte del ruido quede fuera de la banda de interés. El área sombreada de la Figura 3.15 es la potencia total del ruido

de cuantificación dentro de la banda de interés f_{S1} [75], o sea, en el intervalo original de Nyquist, esta área se calcula integrando la ecuación (3.15). Luego, teniendo en cuenta la ganancia de resolución (3.9), se obtiene (3.16) y despejando el incremento de bits, esta ecuación se transforma en la (3.17).

$$\sigma_{e1}^2 = \frac{\sigma_{e2}^2}{f_{S2}} \cdot \int_{-\frac{f_{S1}}{2}}^{\frac{f_{S1}}{2}} |H_{NS}(f)|^2 \cdot df \quad (3.15)$$

$$2^{-2 \cdot \Delta N} = \frac{1}{f_{S2}} \cdot \int_{-\frac{f_{S1}}{2}}^{\frac{f_{S1}}{2}} |H_{NS}(f)|^2 \cdot df \quad (3.16)$$

$$\Delta N = -0.5 \cdot \log_2 \left(\frac{1}{f_{S2}} \cdot \int_{-\frac{f_{S1}}{2}}^{\frac{f_{S1}}{2}} |H_{NS}(f)|^2 \cdot df \right) \quad (3.17)$$

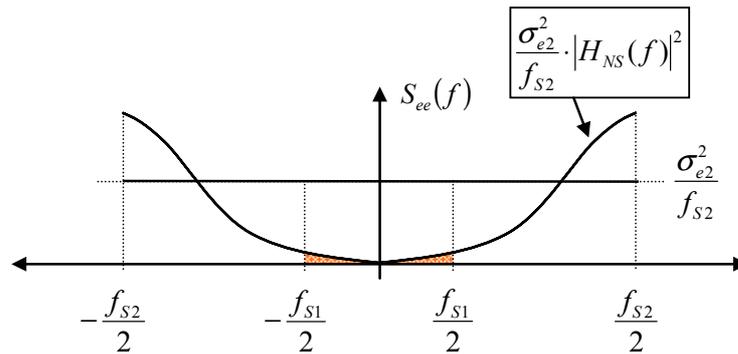


Figura 3.15. Espectro del ruido de cuantificación con sobremuestreo y filtrado.

Se puede resumir que:

- 1) Los cuantificadores que sobremuestran utilizan la modulación y la realimentación para cambiar la forma del espectro del ruido, de forma que en la banda de interés sea mínimo. En otras palabras, la acción del filtro $H_{NS}(f)$ es llevada a cabo por la realimentación de la salida del cuantificador.
- 2) Los cuantificadores más populares que modifican la forma del espectro del ruido de cuantificación son los denominados sigma-delta.
- 3) Los conceptos de muestreo y cuantificación son independientes el uno del otro. El primero se corresponde con la discretización del tiempo y el segundo de la amplitud. Sin embargo, es posible relacionarlos e intercambiar uno por el otro.

La Figura 3.16 muestra el modelo de tiempo discreto del modulador sigma-delta, en el que las muestras, que se realizan a la frecuencia f_{s2} , se denominan $X_{s2}(z)$.

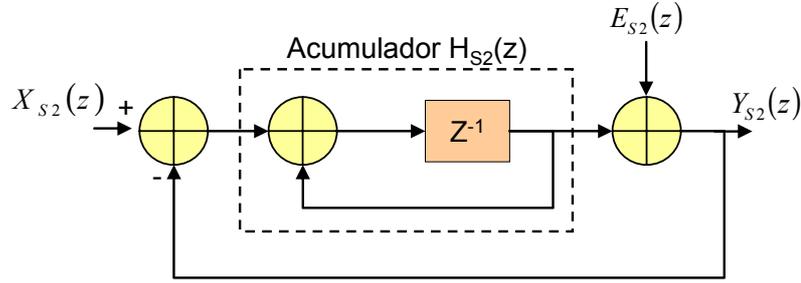


Figura 3.16. Modelo de tiempo discreto del modulador sigma-delta de primer orden.

La función de transferencia $H_{s2}(z)$ del acumulador viene dada por la expresión (3.18) y la expresión general del modulador completo por (3.19), la cual, después de algunos cambios se puede reescribir como (3.20).

$$H_{s2}(z) = \frac{V_{s2}(z)}{W_{s2}(z)} = \frac{z^{-1}}{1 - z^{-1}} \quad (3.18)$$

$$H_{s2}(z) \cdot [X_{s2}(z) - Y_{s2}(z)] + E_{s2}(z) = Y_{s2}(z) \quad (3.19)$$

$$H_{s2}(z) \cdot X_{s2}(z) + E_{s2}(z) = Y_{s2}(z) \cdot [1 + H_{s2}(z)] \quad (3.20)$$

$$\frac{H_{s2}(z)}{1 + H_{s2}(z)} \cdot X_{s2}(z) + \frac{1}{1 + H_{s2}(z)} \cdot E_{s2}(z) = Y_{s2}(z) \quad (3.21)$$

$$H_X(z) \cdot X_{s2}(z) + H_{NS}(z) \cdot E_{s2}(z) = Y_{s2}(z) \quad (3.22)$$

En (3.22) las funciones de transferencia $H_{NS}(z)$, que filtra el ruido, y $H_X(z)$, que filtra la entrada, están definidas por las siguientes ecuaciones: $H_{NS}(z) = \frac{H_{s2}(z)}{1 + H_{s2}(z)}$; $H_X(z) = \frac{1}{1 + H_{s2}(z)}$

Sustituyendo (3.18) en éstas dos expresiones, se obtienen:

$$H_X(z) = \frac{\frac{z^{-1}}{1 - z^{-1}}}{1 + \frac{z^{-1}}{1 - z^{-1}}} = \frac{z^{-1}}{1 - z^{-1} + z^{-1}} = z^{-1} \quad (3.23)$$

$$H_{NS}(z) = \frac{1}{1 + \frac{z^{-1}}{1 - z^{-1}}} = \frac{1 - z^{-1}}{1 - z^{-1} + z^{-1}} = 1 - z^{-1} \quad (3.24)$$

De (3.23) y (3.24) se deduce que $H_{NS}(z)$ es simplemente un filtro paso-alto y que $H_X(z)$ es un retardo puro. Entonces, la ecuación general (3.22) queda de la siguiente forma:

$$Y_{S2}(z) = z^{-1} \cdot X_{S2}(z) + (1 - z^{-1}) \cdot E_{S2}(z) \tag{3.25}$$

Por tanto, la salida esta formada por la entrada retrasada un periodo del reloj f_{S2} y por el ruido de cuantificación filtrado. El modulador actúa para la señal de entrada como un filtro paso-banda y como un filtro paso-alto para el ruido [8]. Debido a que el ruido se traslada a frecuencias elevadas con un filtro paso-bajo diezmadador se tiende a anular dicho ruido y, al mismo tiempo, se promedia y diezma la señal de entrada. Como se mostró antes, el error cuadrático medio se obtiene integrando la densidad espectral de potencia de la señal más el ruido sobre el intervalo de frecuencia de la señal, dando la condición (3.15). En el Apartado 3.5 se deducirá la expresión general de la ganancia de resolución.

3.4. Modulador sigma-delta de segundo orden

Para conseguir desplazar una mayor cantidad de ruido hacia frecuencias altas, más de lo que hace el modulador de primer orden, se utilizan moduladores de mayor orden [76]-[78]. Pero, como el efecto del modulador sobre el ruido es el de un derivador, si se aumenta su orden, aparecen problemas de estabilidad [79]. Por ello, el modulador más utilizado, por el compromiso entre calidad y estabilidad, es el de segundo orden (Figura 3.17 y Figura 3.18). A partir de las ecuaciones (2.26) a la (3.30) se obtiene la función de transferencia y se deduce que, en general, para un orden p del modulador, la función $H_{NS}(z)$ es la misma que en la de primer orden pero elevada a p (la letra p indica el orden del modulador y también se suele utilizar la letra L), como se muestra en la ecuación (3.31).

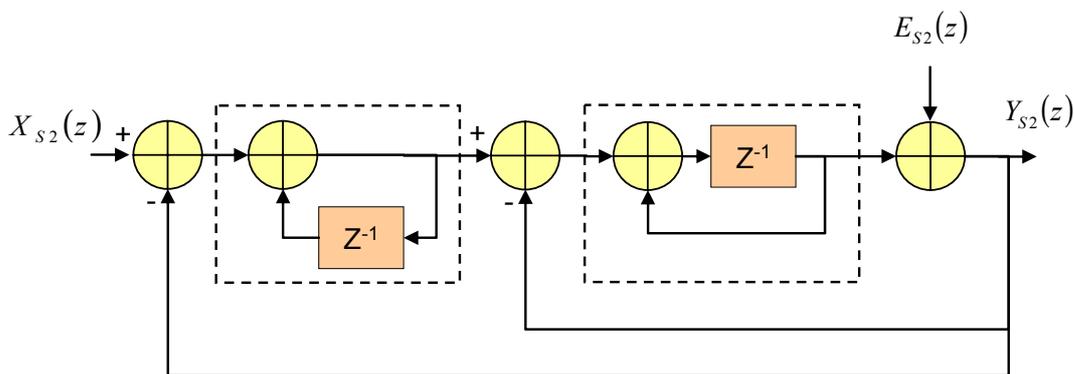


Figura 3.17. Modelo detallado de tiempo discreto del modulador sigma-delta de segundo orden.

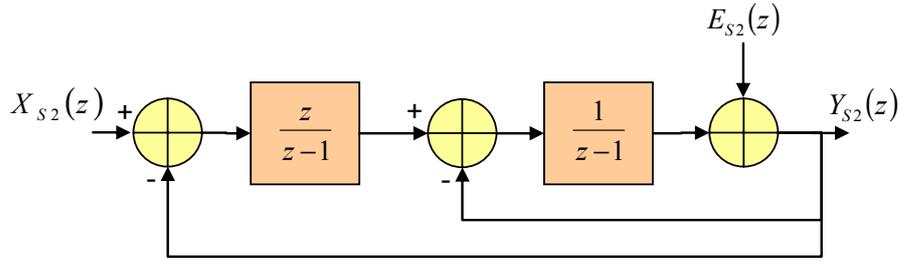


Figura 3.18. Modelo de tiempo discreto del modulador sigma-delta de segundo orden.

$$\left\{ \left[X_{s2}(z) - Y_{s2}(z) \right] \cdot \frac{z}{z-1} \right\} - Y_{s2}(z) \cdot \frac{1}{z-1} + E_{s2}(z) = Y_{s2}(z) \quad (3.26)$$

$$\frac{z}{(z-1)^2} \cdot X_{s2}(z) - Y_{s2}(z) \cdot \frac{z}{(z-1)^2} - Y_{s2}(z) \cdot \frac{1}{(z-1)^2} + E_{s2}(z) = Y_{s2}(z) \quad (3.27)$$

$$\frac{z}{(z-1)^2} \cdot X_{s2}(z) + E_{s2}(z) = Y_{s2}(z) \cdot \left[1 + \frac{1}{(z-1)^2} + \frac{z}{(z-1)^2} \right] = Y_{s2}(z) \cdot \frac{z^2}{(z-1)^2} \quad (3.28)$$

$$\frac{z}{(z-1)^2} \cdot \frac{(z-1)^2}{z^2} X_{s2}(z) + \frac{(z-1)^2}{z^2} \cdot E_{s2}(z) = Y_{s2}(z) \quad (3.29)$$

$$z^{-1} \cdot X_{s2}(z) + (1 - z^{-1})^2 \cdot E_{s2}(z) = Y_{s2}(z) \quad (3.30)$$

$$z^{-1} \cdot X_{s2}(z) + (1 - z^{-1})^p \cdot E_{s2}(z) = Y_{s2}(z) \quad (3.31)$$

3.5. Deducción de la ganancia de resolución mediante técnicas de sobremuestreo

Para encontrar la relación entre la ganancia de resolución ΔN y el factor de sobremuestreo OSR , se debe resolver la expresión (3.32), que se obtuvo a partir de la ecuación (3.16).

$$\frac{1}{f_{S2}} \cdot \int_{-\frac{f_{S1}}{2}}^{\frac{f_{S1}}{2}} |H_{NS}(f)|^2 \cdot df \quad (3.32)$$

Para el modulador de primer orden la función de transferencia que filtra el ruido es $H_{NS}(z) = 1 - z^{-1}$, entonces su función de transferencia en el dominio de la frecuencia vendrá dada por (3.33), cuyo módulo al cuadrado es (3.34) que, sustituido en (3.32), da la integral (3.35).

$$H_{NS}(f) = 1 - e^{-2 \cdot \pi \cdot j \cdot \frac{f}{f_{S2}}} \quad (3.33)$$

$$\begin{aligned}
 [H_{NS}(f)]^2 &= \left[1 - \cos\left(2 \cdot \pi \cdot \frac{f}{f_{S2}}\right) \right]^2 + \left[\text{sen}\left(2 \cdot \pi \cdot \frac{f}{f_{S2}}\right) \right]^2 = \\
 &= 1 - 2 \cdot \cos\left(2 \cdot \pi \cdot \frac{f}{f_{S2}}\right) + \cos^2\left(2 \cdot \pi \cdot \frac{f}{f_{S2}}\right) + \text{sen}^2\left(2 \cdot \pi \cdot \frac{f}{f_{S2}}\right) = \\
 &= 2 - 2 \cdot \cos\left(2 \cdot \pi \cdot \frac{f}{f_{S2}}\right) = 4 \cdot \text{sen}^2\left(\pi \cdot \frac{f}{f_{S2}}\right)
 \end{aligned} \tag{3.34}$$

$$\frac{1}{f_{S2}} \cdot \int_{-\frac{f_{S1}}{2}}^{\frac{f_{S1}}{2}} \left[2 - 2 \cdot \cos\left(2 \cdot \pi \cdot \frac{f}{f_{S2}}\right) \right] \cdot df \tag{3.35}$$

Ahora, resolviendo la integral (3.35), se obtiene (3.36), la cual combinada con la ecuación (3.16), da como resultado la expresión (3.37), de la que despejando la ganancia de resolución se obtiene la ecuación buscada (3.38).

$$\begin{aligned}
 \frac{1}{f_{S2}} \cdot \int_{-\frac{f_{S1}}{2}}^{\frac{f_{S1}}{2}} \left[2 - 2 \cdot \cos\left(2 \cdot \pi \cdot \frac{f}{f_{S2}}\right) \right] \cdot df &= \frac{1}{f_{S2}} \cdot 2 \cdot \int_0^{\frac{f_{S1}}{2}} \left[2 - 2 \cdot \cos\left(2 \cdot \pi \cdot \frac{f}{f_{S2}}\right) \right] \cdot df = \\
 &= \frac{4}{f_{S2}} \cdot \left[f - \frac{\text{sen}\left(2 \cdot \pi \cdot \frac{f}{f_{S2}}\right)}{\frac{2 \cdot \pi}{f_{S2}}} \right]_0^{\frac{f_{S1}}{2}} = \frac{4}{f_{S2}} \cdot \left[\frac{f_{S1}}{2} - \frac{\text{sen}\left(\pi \cdot \frac{f_{S1}}{f_{S2}}\right)}{\frac{2 \cdot \pi}{f_{S2}}} \right] = \\
 &= \frac{2}{\frac{f_{S2}}{f_{S1}}} - \frac{2 \cdot \text{sen}\left(\frac{\pi}{\frac{f_{S2}}{f_{S1}}}\right)}{\pi} = \frac{2}{D} - \frac{2 \cdot \text{sen}\left(\frac{\pi}{D}\right)}{\pi}
 \end{aligned} \tag{3.36}$$

$$2^{-2 \cdot \Delta N} = \frac{2}{D} - \frac{2 \cdot \text{sen}\left(\frac{\pi}{D}\right)}{\pi} \tag{3.37}$$

$$\Delta N_1 = -0.5 \cdot \log_2 \left[\frac{2}{D} - \frac{2 \cdot \text{sen}\left(\frac{\pi}{D}\right)}{\pi} \right] \tag{3.38}$$

La ecuación (3.38), aunque ofrece una buena solución, es difícil de resolver y, en su lugar, se utiliza otra más sencilla, que se obtiene a partir de la ecuación (3.34). Efectivamente, si se considera que la frecuencia f_{S2} es mucho mayor que la f_{S1} , o lo que es lo mismo, que el factor de sobremuestreo OSR es grande, entonces, el seno de la ecuación (3.34) tiende a cero y, por tanto, se puede sustituir por la variable, quedando simplificada según (3.39).

$$[H_{NS}(f)]^2 = 4 \cdot \left(\pi \cdot \frac{f}{f_{S2}} \right)^2 \quad (3.39)$$

Esta expresión es más sencilla para calcular la ganancia en moduladores de mayor orden. En general, si el orden del modulador es p se aplica la ecuación (3.40), volviendo a hacer la integral (3.36) y aplicando el logaritmo en base dos, se obtiene la expresión simplificada (3.42).

$$[H_{NS}(f)]^2 = \left(2 \cdot \pi \cdot \frac{f}{f_{S2}} \right)^{2 \cdot p} \quad (3.40)$$

$$\begin{aligned} 2^{-2 \cdot \Delta N} &= \frac{1}{f_{S2}} \cdot \int_{-\frac{f_{S1}}{2}}^{\frac{f_{S1}}{2}} \left(2 \cdot \pi \cdot \frac{f}{f_{S2}} \right)^{2 \cdot p} \cdot df = \frac{2^{2 \cdot p} \cdot \pi^{2 \cdot p}}{f_{S2}^{2 \cdot p+1}} \cdot \int_{-\frac{f_{S1}}{2}}^{\frac{f_{S1}}{2}} f^{2 \cdot p} \cdot df = \\ &= \frac{2^{2 \cdot p} \cdot \pi^{2 \cdot p}}{f_{S2}^{2 \cdot p+1}} \cdot 2 \cdot \int_0^{\frac{f_{S1}}{2}} f^{2 \cdot p} \cdot df = \frac{2^{2 \cdot p} \cdot \pi^{2 \cdot p}}{f_{S2}^{2 \cdot p+1}} \cdot 2 \cdot \left[\frac{f^{2 \cdot p+1}}{2 \cdot p+1} \right]_0^{\frac{f_{S1}}{2}} = \\ &= \frac{2^{2 \cdot p} \cdot \pi^{2 \cdot p}}{f_{S2}^{2 \cdot p+1}} \cdot 2 \cdot \frac{\left(\frac{f_{S1}}{2} \right)^{2 \cdot p+1}}{2 \cdot p+1} = \frac{f_{S1}^{2 \cdot p+1}}{f_{S2}^{2 \cdot p+1}} \cdot \frac{\pi^{2 \cdot p}}{2 \cdot p+1} \cdot \frac{2^{2 \cdot p} \cdot 2}{2^{2 \cdot p+1}} = \\ &= \frac{f_{S1}^{2 \cdot p+1}}{f_{S2}^{2 \cdot p+1}} \cdot \frac{\pi^{2 \cdot p}}{2 \cdot p+1} \cdot \frac{2^{2 \cdot p} \cdot 2}{2^{2 \cdot p+1}} = \frac{1}{OSR^{2 \cdot p+1}} \cdot \frac{\pi^{2 \cdot p}}{2 \cdot p+1} \end{aligned} \quad (3.41)$$

$$\Delta N_2 = -0.5 \cdot \log_2 \left[\frac{\pi^{2 \cdot p}}{(1 + 2 \cdot p) \cdot OSR^{2 \cdot p+1}} \right] \quad (3.42)$$

A continuación se presenta la Tabla 3.1, que muestra la ganancia de resolución que se obtiene según los dos métodos deducidos, el exacto (ΔN_1) y el aproximado (ΔN_2). Se comprueba que los dos valores convergen rápidamente al aumentar OSR . En adelante se utilizará la ecuación (3.42) dada su simplicidad y buena exactitud. Sin embargo, si se desea hacer correctamente la gráfica de la potencia espectral se debe utilizar la ecuación exacta (3.43).

Tabla 3.1. Comparación entre los dos métodos para obtener la ganancia de resolución.

OSR	$\Delta N1$	$\Delta N2$
2	0.4607	0.6410
4	2.1632	2.1410
8	3.6465	3.6410
16	5.1424	5.1410
32	6.6413	6.6410
64	8.1411	8.1410
128	9.6410	9.6410
256	11.1410	11.1410
512	12.6410	12.6410
1024	14.1410	14.1410
2048	15.6410	15.6410

$$[H_{NS}(f)]^2 = \left[2 \cdot \text{sen} \left(\pi \cdot \frac{f}{f_{S2}} \right) \right]^{2-p} \tag{3.43}$$

Utilizando esta ecuación, en la Figura 3.19 se muestra la potencia espectral del ruido de cuantificadores de distintos órdenes. En la Figura 3.20 se compara la ganancia de resolución sin modulación y con moduladores de primero y segundo orden.

Si el orden del modulador es $p=1$, y el factor de sobremuestreo es $OSR=32$, entonces, de acuerdo con (3.42), la ganancia teórica de resolución ΔN que se puede obtener es de 6.6 bits. Por tanto, la resolución total sería de $N+\Delta N= 1+6.6= 7.7$ b. En la Figura 3.21 se presenta la ganancia en la SNR del convertidor frente al factor de sobremuestreo en escala logarítmica.

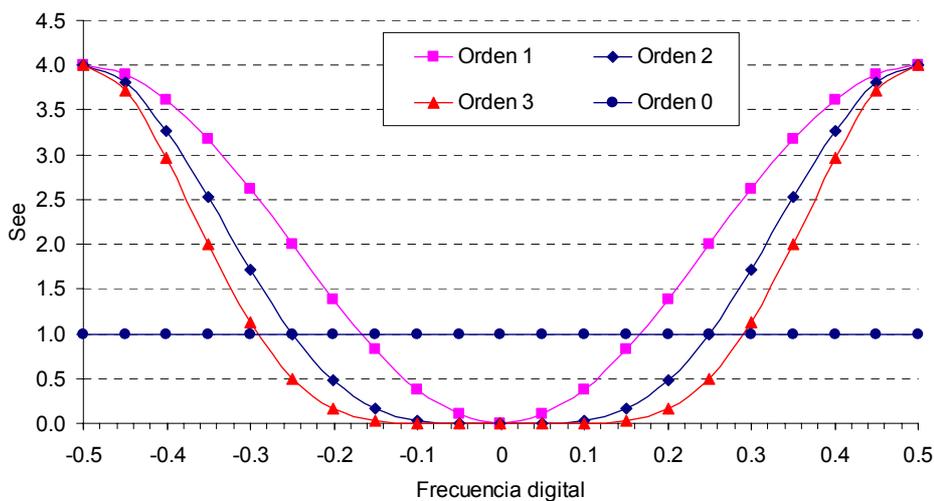


Figura 3.19. Potencia del espectro del ruido en los convertidores de sobremuestreo.

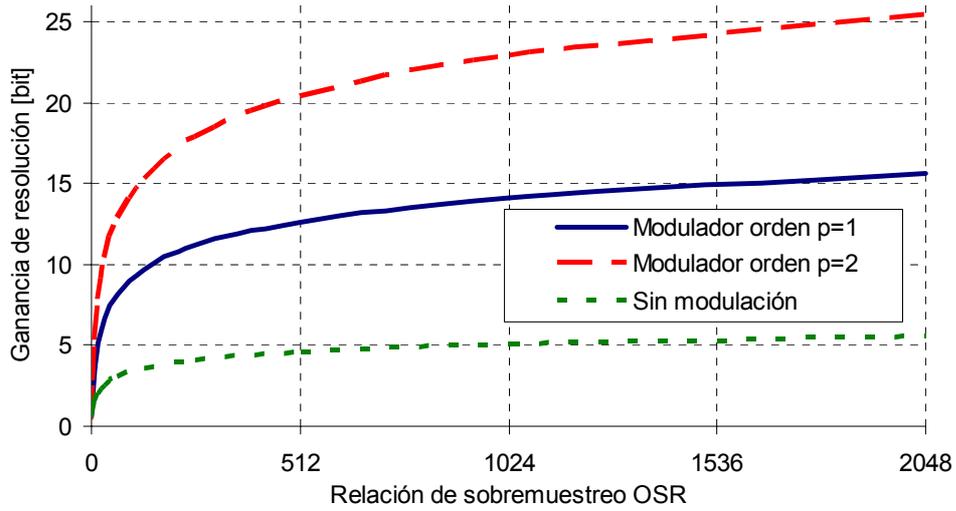


Figura 3.20. Ganancia de resolución en los convertidores de sobremuestreo.

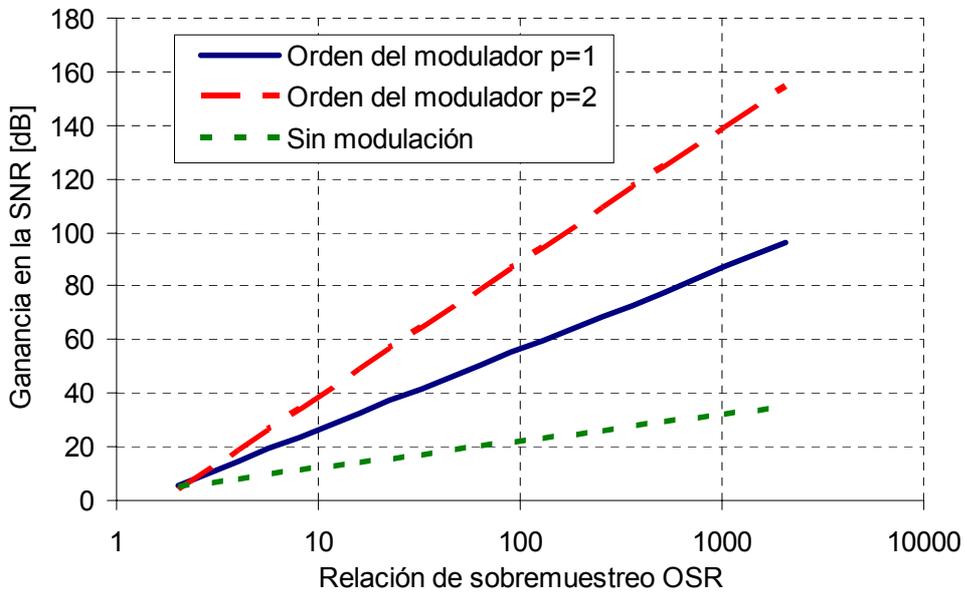


Figura 3.21. Ganancia de la SNR según el factor de sobremuestreo.

3.6. Etapas de filtrado y diezmado en los convertidores sigma-delta

3.6.1. Introducción

Como se mostró en la Figura 3.1, después de la etapa moduladora, el convertidor $\Sigma\text{-}\Delta$ tiene las etapas de filtrado, que llevan a cabo las siguientes tareas [63], [80]:

1. Rechazar el ruido que se trasladó fuera de la banda de la señal. Además, tiene dos efectos no deseados. El primero que deja una pequeña cantidad de ruido dentro de dicha banda. El segundo, que la banda de entrada queda limitada por la frecuencia de corte de dicho filtro.

2. Reducir el número de muestras de salida después de filtrar el ruido de alta frecuencia. Este proceso se conoce como diezmado [81], y se caracteriza por la relación entre el número de muestras de entrada y de salida, que se representa por la letra D .
3. Hacer de filtro *antialiasing*. En la práctica, las señales de entrada no suelen estar limitadas a un cierto ancho de banda pero, como el modulador esta muestreando a una frecuencia mucho mayor que la de Nyquist, el filtro analógico de entrada no tiene que ser muy selectivo. Cuando el procesador digital reduce la velocidad de muestreo a un valor cercano a la frecuencia de Nyquist, es necesario un filtrado adicional *antialiasing*.

Para conseguir que la respuesta en frecuencia en la banda de interés sea lo más plana posible, sin que el coste aumente innecesariamente, se tienen dos opciones:

1. Utilizar un filtro de promediado combinado con un post filtrado que ecualice su respuesta mediante un filtro de tipo FIR o IIR [82], [83]. De esta forma también se puede conseguir otro diezmado adicional, si fuese necesario.
2. Prescindir del filtro inicial de promediado y realizar un filtro FIR o IIR optimizado [84], [85] para consumir el mínimo de recursos lógicos, de tiempo y de potencia. También se han desarrollado otros tipos de filtros avanzados, como los filtros recursivos de *Laguerre* y el filtrado de *Kalman* [86].

3.6.2. Filtrado y diezmado con filtro de promediado

En general los filtros de diezmado son difíciles de implementar porque la frecuencia de las muestras es muy alta y los algoritmos de procesado se deben implementar en tiempo real. Además, el proceso de conformado del ruido produce una gran cantidad de ruido que se debe filtrar. Otro inconveniente aparece cuando no se puede tolerar un error de fase o de amplitud. Finalmente, el filtro digital se debe implementar con la mínima cantidad de recursos lógicos para facilitar su implementación en las distintas tecnologías, por ejemplo: en un ASIC que implemente una unidad aritmética con una ROM para los coeficientes y una RAM para los valores intermedios de las muestras que se procesan [87], en un DSP que implemente el algoritmo del filtro [80], [88], o en una FPGA que configure el hardware que implemente el filtro [83].

La forma más sencilla y económica de realizar un filtro que, además, reduzca la frecuencia de las muestras es el filtro de promediado (también se denominan: *sinc filter* o *comb-filter*), que tiene una buena respuesta temporal y es fácil de implementar porque sus coeficientes son unitarios y, por tanto, no necesita ningún multiplicador [82]. Sin embargo, este tipo de filtro tiene el inconveniente de tener una respuesta en frecuencia pobre, ya que a partir de dc presenta una atenuación considerable que aumenta rápidamente con la frecuencia. Por tanto, si no se ecualiza dicha respuesta en la banda de interés, aparecerá una distorsión de frecuencia en la señal de salida del convertidor.

El funcionamiento de un filtro basado en la función *sinc* es equivalente a un filtro FIR (filtro de respuesta finita) con una ventana rectangular. Pese a su sencillez, este filtro es muy efectivo a la hora de rechazar el ruido generado en los moduladores sigma-delta, aunque su respuesta en bajas frecuencias no es plana por lo que introduce cierta distorsión. Por ello, rara vez se utiliza este tipo de filtro sin otra etapa de post-filtrado que actúe como ecualizador, además de reducir más el ruido y diezmar más la señal. Por tanto, se suelen realizar las tareas de filtrar y diezmar en varias etapas. Por ejemplo, en las dos etapas siguientes:

1. Inicialmente se reduce la frecuencia de trabajo mediante un filtro de promediado o *comb filter*.
2. La segunda etapa se puede implementar mediante un filtro FIR paso-bajo con valores de los coeficientes simétricos para mantener una respuesta de fase lineal. Además, esta etapa añade un diezmo adicional y una compensación de amplitud debida a la etapa previa. Por ello, los coeficientes del filtro se deben calcular par realizar dicha ecualización en la banda de la señal y para atenuar lo máximo en las demás frecuencias.

También se suele decir que el filtro de promediado es un filtro de media móvil, en el que cada muestra de la salida es el promedio de las muestras de entrada adyacentes [10]. Éste se puede llevar a cabo por convolución o de forma recursiva. A pesar de su sencillez, el filtro de media móvil es adecuado cuando es necesario reducir un ruido blanco al mismo tiempo que se desea mantener una buena respuesta transitoria. Esto hace que sea el filtro estrella para filtrar señales temporales. Sin embargo, este filtro tiene una respuesta en frecuencia mala con poca selectividad a la hora de separar bandas de frecuencias.

Un filtro de promediado de longitud D es un filtro FIR con todos sus coeficientes iguales a '1' y cuya función de transferencia es

$$H(z) = \frac{1}{D} \cdot \sum_{n=0}^{D-1} z^{-n} = \frac{Y(z)}{X(z)} \quad (3.44)$$

y para $D=4$ la ecuación (3.44) esta dada por: $y(n) = \frac{x(n) + x(n-1) + x(n-2) + x(n-3)}{4}$

Esta claro que este tipo de filtro es un simple acumulador que lleva a cabo la media móvil. Utilizando la ecuación de la suma geométrica, la ecuación (3.44) se puede expresar mediante la ecuación (3.45), que corresponde (para $D=4$) con la ecuación en diferencias (3.46).

$$H(z) = \frac{1}{D} \cdot \frac{1 - z^{-D}}{1 - z^{-1}} \quad (3.45)$$

$$y(n) = \frac{x(n) - x(n-4) + y(n-1)}{4} \quad (3.46)$$

De esta forma, utilizando la ecuación (3.46), el número de sumas se ha reducido de forma que es independiente del factor D . Esta solución se puede factorizar en dos procesos independientes, una integración seguida de una diferenciación, tal y como se muestra a continuación en (3.47):

$$Y(z) = \left(\frac{1}{1 - z^{-1}} \right) \cdot \frac{1}{D} \cdot (1 - z^{-D}) \cdot X(z) \quad (3.47)$$

Ahora, ya que el filtro va seguido de una relación de diezmado D , el proceso diferenciador se puede hacer a una frecuencia menor si se realiza después del diezmado [82]. Por tanto, el proceso (3.45) se ha dividido en tres subprocesos, como se muestra en la Figura 3.22.

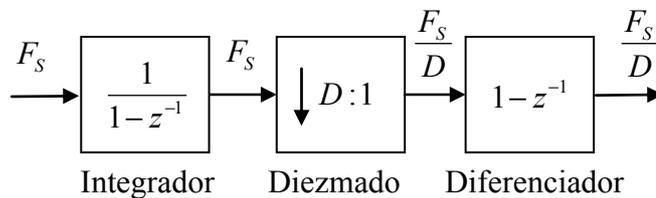


Figura 3.22. Diagrama de bloques del filtro de orden $k=1$.

Por otro lado, para aumentar la atenuación en la banda de rechazo y para prevenir un posible efecto de *aliasing* después del diezmado se pueden utilizar varios filtros en cascada. La Figura 3.23 muestra dos filtros en cascada que configuran un filtro de orden $k=2$. En general, según [89], las

mejores prestaciones del filtro, para un modulador Σ - Δ de orden p , se consiguen con un filtro de orden $p+1$ (3.48).

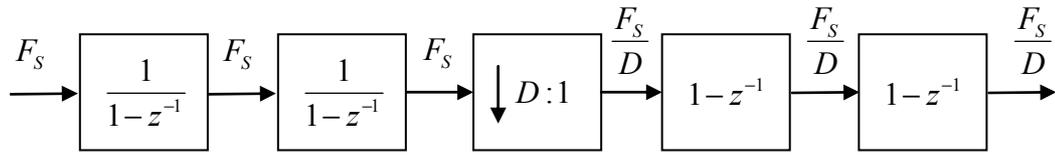


Figura 3.23. Diagrama de bloques del filtro de orden $k=2$.

$$k = 1 + p \tag{3.48}$$

Si D es un número impar la media se puede hacer simétricamente. Esto equivale a cambiar los extremos de la suma (3.44) de $n = \frac{-D+1}{2}$ a $n = \frac{D-1}{2}$. Se debe tener en cuenta que la media móvil es un filtro que utiliza un núcleo muy sencillo. Por ejemplo, si es un filtro que promedia 9 valores, los coeficientes serán: ...0, 0, 1/9, 1/9, 1/9, 1/9, 1/9, 1/9, 1/9, 1/9, 0, 0... Esto quiere decir que el filtro de media móvil es la convolución de la entrada con una ventana rectangular de área unidad [10]. Ahora, si la DFT de una ventana rectangular corresponde con la respuesta en frecuencia del filtro [53], [54], ésta sigue la ecuación (3.49), y en el caso de que se trate de k filtros en cascada, la expresión es la misma pero elevada a k , o sea (3.50).

$$H(f) = \frac{\sin(\pi \cdot f \cdot D)}{D \cdot \sin(\pi \cdot f)} \tag{3.49}$$

$$H(f) = \left[\frac{\sin(\pi \cdot f \cdot D)}{D \cdot \sin(\pi \cdot f)} \right]^k \tag{3.50}$$

La Figura 3.24 muestra la respuesta de tres filtros de promediado, de primer, segundo y tercer orden, con un factor D igual a 32. Estos filtros presentan una respuesta en frecuencia pobre, no tienen una respuesta plana en la banda de paso (Figura 3.25), tienen una pendiente de caída lenta en la banda de transición (en realidad se confunden ambas bandas) y poca atenuación en la banda de atenuación. Queda claro que este tipo de filtrado no se puede utilizar para separar unas frecuencias de otras. Hay que recordar la regla que dice: “una buena respuesta en frecuencia implica una mala respuesta temporal y viceversa”. En otras palabras, un filtro de media móvil es un buen filtro de suavizado (la acción en el dominio del tiempo), pero un mal filtro paso-bajo (la acción en el dominio de la frecuencia).

Los filtros de media móvil multietapa implican pasar la señal varias veces por uno o más filtros. El filtro de primer orden equivale a una ventana rectangular y el de segundo orden equivale a utilizar una ventana triangular (una ventana rectangular convolucionada consigo misma). Si se sigue aumentando el número de etapas, después de cuatro o más, el núcleo del filtro se asemeja a uno de Gauss. La respuesta en frecuencia esta dada por (3.49) multiplicada por si misma tantas veces como sea el orden del filtro deseado, esto es (3.50).

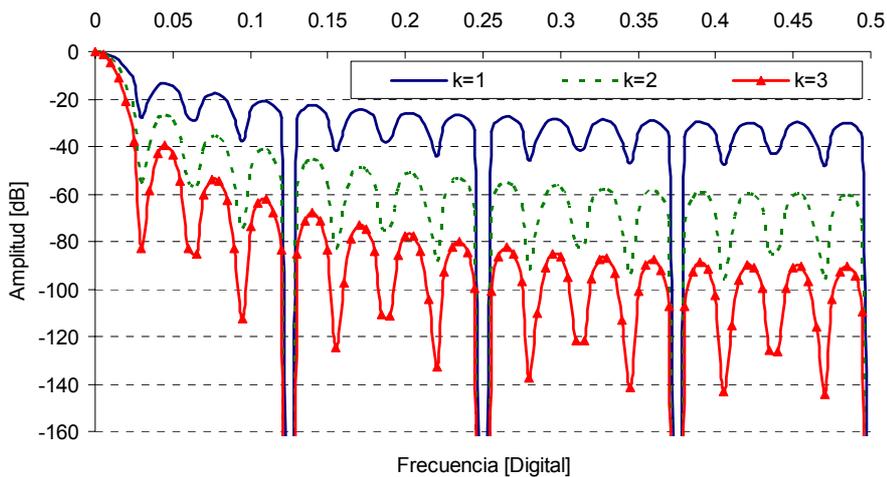


Figura 3.24. Respuesta en frecuencia del filtro sinc con un factor de diezmado $D= 32$.

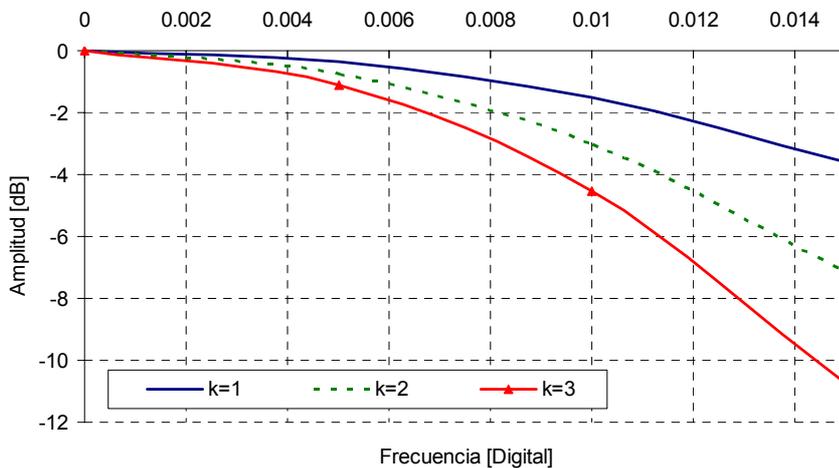


Figura 3.25. Caída de la respuesta en frecuencia debida al filtro de promediado basado en la función sinc.

3.6.3. Procedimiento para equalizar la respuesta del filtro de promediado

Un método para conseguir un filtro FIR de fase lineal que contrarreste la caída de respuesta que se produce en el filtro *sinc* (Figura 3.25) consiste en especificar la respuesta necesaria del filtro e

implementarlo por el método de muestreo en frecuencia [90], [91]. Los pasos para conseguir los coeficientes del filtro son los siguientes:

1. Se especifica la respuesta en frecuencia $H_d(\omega)$ que se desea.
2. Se muestrea dicha respuesta en un conjunto de frecuencias ω_k equiespaciadas:

$$\omega_k = \frac{2 \cdot \pi}{M} \cdot k \quad ; \quad k = 0, 1, \dots, \frac{M-1}{2} \text{ si } M \text{ es impar,}$$

$$k = 0, 1, \dots, \frac{M}{2} - 1 \text{ si } M \text{ es par,}$$

donde M es la longitud del filtro.

3. Se obtiene la respuesta impulsional simétrica con la siguiente ecuación:

$$h(n) = G(1) + 2 \cdot \sum_{k=1}^n G(k) \cdot \cos \left\{ \frac{2 \cdot \pi \cdot k}{M} \cdot \left[(n-1) + \frac{1}{2} \right] \right\} ; n = 1, 2, \dots, M$$

3.6.4. Ejemplo de obtención de un filtro FIR por muestreo en frecuencia

Se desea implementar un filtro que tenga la respuesta en frecuencia de la Figura 3.26, que tiene una respuesta unitaria a frecuencia cero y un máximo de 1.15 en la banda de paso. El coeficiente 0.5 se denomina coeficiente de transición, los demás son los de la banda de rechazo, que se desea sean cero. Hay que señalar que el coeficiente de la banda de transición se debe obtener para optimizar el filtro de forma que el lóbulo lateral sea lo menor posible, en este caso se ha seguido el criterio de simplicidad y se ha establecido como coeficiente de transición un valor de 0.5. La secuencia $Hd(w)$, tomando las muestras equiespaciadas, es la siguiente: $Hd(w) = \{1, 1.007, 1.03, 1.07, 1.15, 0.5, 0, 0, 0, 0\}$. A partir de la secuencia $Hd(w)$, con el programa de Matlab de la Tabla 3.2, se obtiene la secuencia $h(n)$, correspondiente al núcleo del filtro buscado.

Para comprobar que la secuencia del núcleo del filtro obtenido, que se muestra en la Figura 3.27, es la correcta, se calcula su respuesta en frecuencia, mediante el algoritmo de Matlab de la Tabla 3.3. La respuesta obtenida, que se presenta en la Figura 3.28, es prácticamente idéntica a la deseada, lo cual valida el método utilizado.

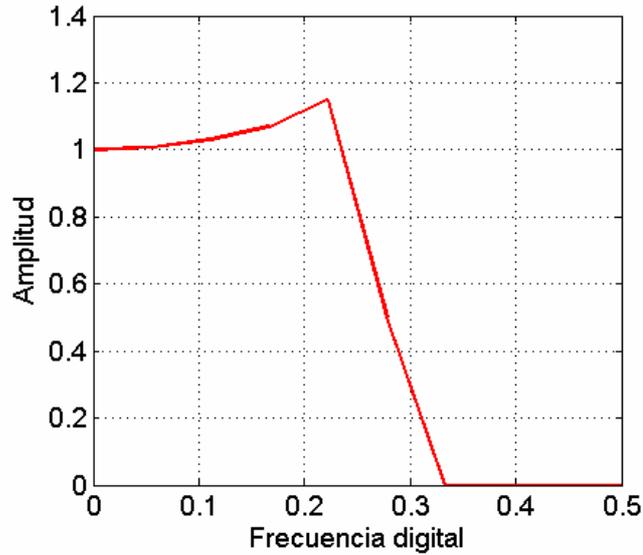


Figura 3.26. Respuesta deseada en frecuencia $H_d(w)$.

Tabla 3.2. Código de la función FIRFS.

```
%Función FIRFS (FIR Frequency Sampled)
%Camilo Quintáns Graña 25-03-2007
%Calculo de un filtro FIR de fase lineal mediante el
%método de muestreo en frecuencia
%Proakis and Manolakis, "Tratamiento digital de señales, principios,
%algoritmos y aplicaciones," Prentice Hall, 2001, pp. 639-643.
%-----parámetros de la función-----
%fr = Frequency response
%nfr= Frequency response number of points
%devuelve una secuencia con la respuesta impulsional del
%filtro FIR. El número de puntos es nfr*2 +1

function[h]=FIRFS(fr)
%longitud del filtro
u=length(fr);
M=2*u+1;

%Generación de la secuencia de coeficientes
h=zeros(M,1);
%Generación de la secuencia G a partir de fr
G=zeros(u,1);

for i=1 : u;
    G(i)=power(-1,i-1)*fr(i);
end;

for i=1 : M;
    suma=0;
    for k=2 : u;
        suma=suma+G(k)*cos((2*pi*(k-1)/M)*(i-1+0.5));
    end;
    h(i)=(G(1)+2*suma)/M;
end;
end
```

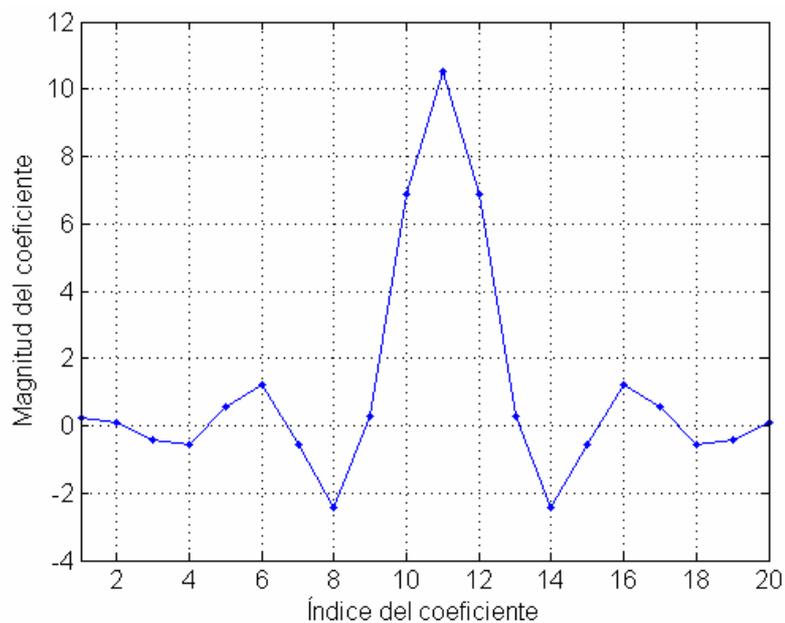


Figura 3.27. Respuesta impulsional $h(n)$ del filtro ecualizador.

Tabla 3.3. Procedimiento de Matlab que calcula la respuesta en frecuencia.

```

%respuesta en frecuencia de una secuencia
function [shr]=respuesta (sh)
shifft=fft (sh);
nshifft=length (sh);
shr=zeros ((nshifft/2),1);
for i=1 : (nshifft/2);
    shr (i)=sqrt (real (shifft (i))^2+imag (shifft (i))^2);
end;
end
    
```

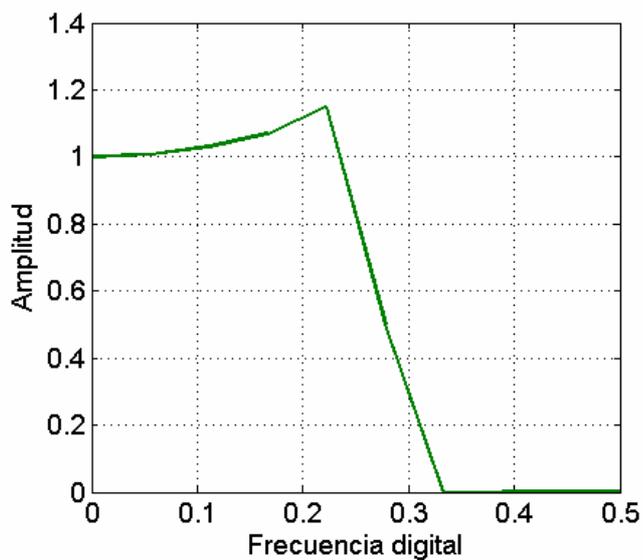


Figura 3.28. Respuesta del filtro calculado $H_c(w)$.

3.6.5. Ecualización de la respuesta del convertidor sigma-delta con el filtro FIR

En el apartado anterior se estudió el procedimiento por el que se obtiene un filtro FIR a partir de la especificación en frecuencia. En este apartado se aplica ese método para ecualizar la respuesta en frecuencia del filtro de promediado de un convertidor sigma-delta.

La Figura 3.24 mostraba la respuesta de un filtro de razón de diezmado $D=32$, por tanto la banda de frecuencias pasa a ser de $0.5/32=0.015$. La curva de la respuesta en esta banda se muestra en escala lineal en la Figura 3.29.

El filtro FIR de ecualización se diseña para que compense la caída de respuesta y, además, para aumentar la selectividad total. Para ello, se compensará hasta la frecuencia 0.012 y luego se atenuará lo máximo posible. De este modo, hasta 0.012 la respuesta deseada será la inversa de la de la Figura 3.29, y a partir de ella la respuesta será nula. Por tanto, la secuencia de valores de la respuesta en frecuencia deseada para el filtro ecualizador es la siguiente:

$$H_d = \{1, 1.003, 1.031, 1.089, 1.183, 1.324, 1.530, 0.000, 0.000\}$$

Los coeficientes del filtro son, según la función $FIRFS(H_d)$ que se presentó en la Tabla 3.2, los siguientes: $h = \{-0.839, 1.741, -0.5677, -1.479, 2.229, -0.207, -3.947, 4.942, 13.258, 4.942, -3.947, -0.207, 2.229, -1.479, -0.5677, 1.741, -0.839\}$

Por último, el filtro ecualizador, cuya respuesta H_{real} se muestra a continuación, compensa perfectamente la caída en la respuesta del filtro de promediado, tal y como era de esperar.

$$H_{real} = \{1, 1.003, 1.089, 1.183, 1.324, 1.530, 1.037e-015, 1.6894e-016, 6.7819e-016\}$$

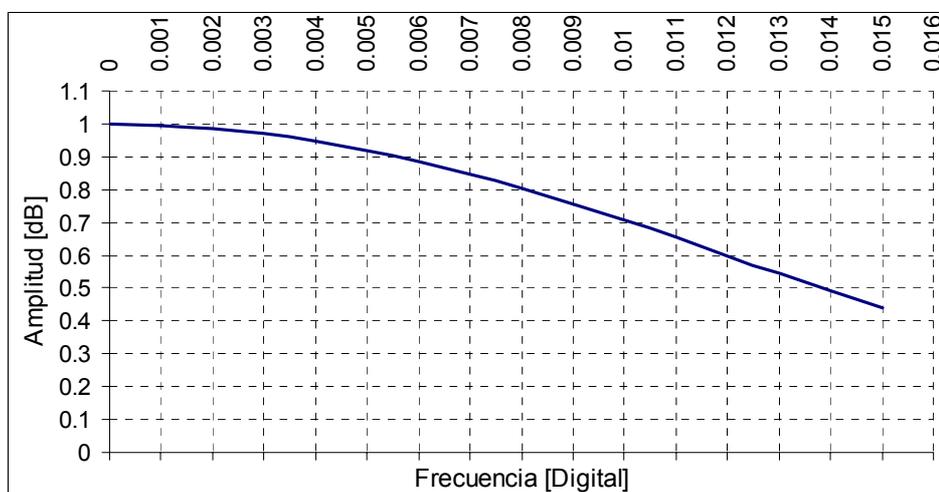


Figura 3.29. Respuesta en frecuencia del filtro de promediado.

3.7. Simulación de convertidores sigma-delta

En este apartado se exponen dos métodos de simulación de un convertidor sigma-delta. En el Apartado 3.7.1 se presenta un modulador de primer orden que se simula mediante OrCAD-PSpice [92] y en el Apartado 3.7.2 se simula un convertidor completo, primero se simula el modulador mediante una rutina en lenguaje C y, a continuación, se simula el filtro diezmador mediante el software QuartusII de Altera. Por último, se analizan los resultados de la simulación con el método de test que se presentó en el Apartado 2.8.2.

3.7.1. Simulación de un circuito modulador sigma-delta con OrCAD PSpice

En la Figura 3.30 se muestra el diagrama de bloques de un modulador de primer orden cuyo circuito electrónico corresponde al esquema de la Figura 3.31. El integrador se ha realizado mediante un amplificador operacional TL082, que tiene una entrada FET. El convertidor A/D de un bit se ha implementado con un comparador LM111 cuya salida en colector abierto facilita la adaptación al nivel de 5 V TTL de la entrada del biestable 7474, que memoriza el bit convertido durante un ciclo del reloj de muestreo CLK. La salida de este biestable es la salida SDM del modulador, la cual se vuelve a convertir a analógica mediante el subcircuito señalado como “1 BIT DAC”. Este subcircuito proporciona una realimentación de 5 V cuando la salida SDM es ‘1’ y de -5 V cuando es ‘0’.

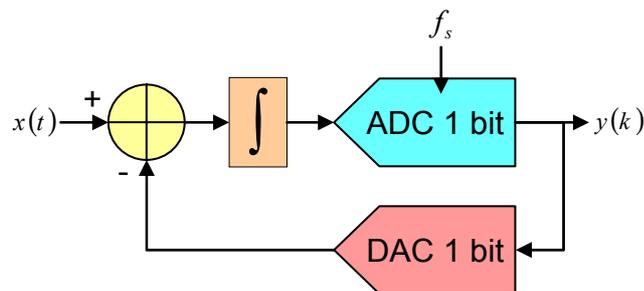


Figura 3.30. Diagrama de bloques del modulador sigma-delta de primer orden.

En la Figura 3.32 se muestra el resultado de simulación del circuito modulador cuando se conecta a su entrada una señal rampa entre -4.95 V y +4.95 V, o sea, entre los márgenes máximos de entrada del modulador. Se observa que el funcionamiento es correcto porque no hay saturación a la salida del integrador. Además, se observa, como era de esperar en el modulador de primer orden, cierta correlación entre la salida del integrador y la entrada.

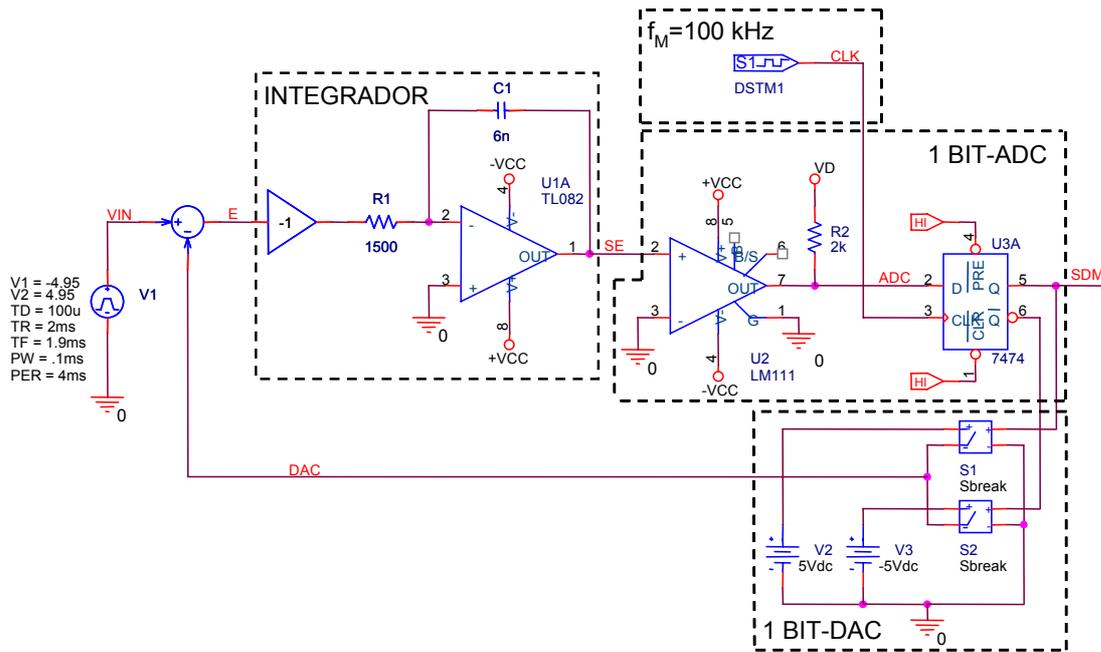


Figura 3.31. Esquema del circuito modulador sigma-delta de primer orden.

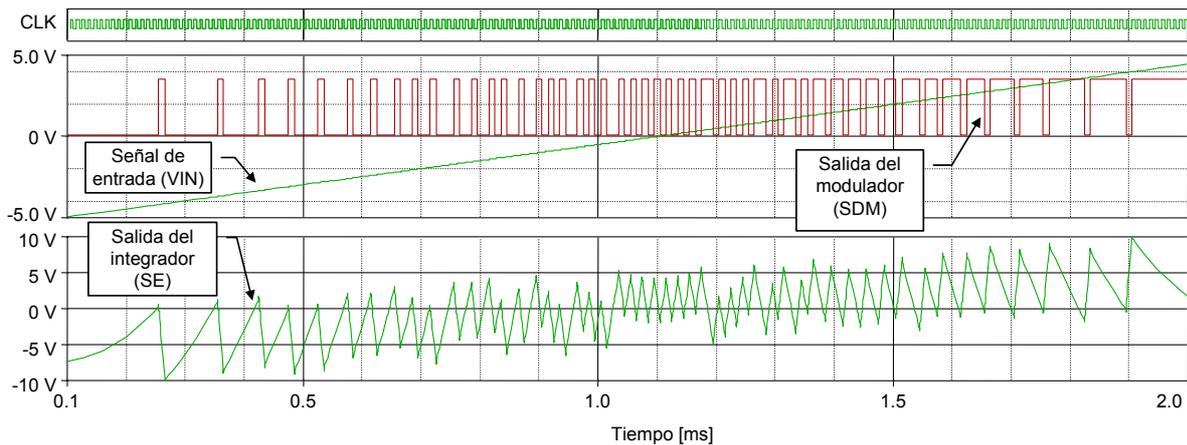


Figura 3.32. Resultado de la simulación del modulador sigma-delta de primer orden.

3.7.2. Simulación de un convertidor A/D sigma-delta

En este apartado se simula un convertidor sigma-delta completo. El modulador se diseña mediante un algoritmo codificado en C y el filtro mediante una descripción VHDL [93] para sintetizar en una FPGA.

El algoritmo del modulador genera un fichero de tipo texto con los datos de su salida que sirve para inicializar una memoria de la FPGA que será la entrada del filtro de diezmado. Finalmente, se realiza el test de los resultados para obtener el ENOB del ADC.

Se parte de que la frecuencia del modulador f_M es de 1 MHz y, después del diezmado, la frecuencia de conversión del convertidor será de $f_S = 31\,250$ kHz ya que el factor de diezmado es de $D=32$. Por otro lado, el tamaño mínimo óptimo M del registro de los datos para poder realizar correctamente el test del ADC, según (3.51), es de 128 porque, aunque el mínimo número es de 101, 128 es mejor porque facilita el análisis en el dominio de la frecuencia mediante la FFT [42]. Como se necesitarán 128 muestras y el factor de diezmado es de 32, el modulador calculará 4 096 puntos, tal y como muestra la Tabla 3.4, que incluye el programa en C que simula el modulador.

$$M \geq \pi \cdot 2^N = \pi \cdot 2^5 = 100.5 \quad (\text{muestras}) \quad (3.51)$$

Ahora, con los valores de M y de f_S y la ecuación (3.52) se obtiene la frecuencia óptima de la señal a convertir f_{opt} , que es de 244.14 Hz.

$$f_{opt} = \frac{f_S}{M} = \frac{31\,250}{128} = 244.14 \text{ Hz} \quad (3.52)$$

Tabla 3.4. Simulación del modulador sigma-delta mediante un algoritmo programado en C.

```

/** Routine example for initializing a ROM for being implemented in a FPGA
a First order sigma-delta modulator, 28-01-2007 Camilo Quintáns Graña
***/

#include "stdafx.h"
#include "math.h"
#define M 4096

int main(int argc, char* argv[])
{
    double PI=3.141592654;
    unsigned long i=0;
    unsigned char b=0x00;
    FILE *f;
    /* x => input ; e => error ; v,w => intermediate signals
    y => output signal */
    double x=0,e=0,v=0,w=0,y=0;

    f=fopen("ROMdata.MIF","w");

    fprintf(f,"--Example of a generated Memory Initialization File (.mif)\n\n");
    fprintf(f,"WIDTH=1;\nDEPTH=%d;\n\n",M);
    fprintf(f,"ADDRESS_RADIX=UNS;\n");
    fprintf(f,"DATA_RADIX=UNS;\n\n");
    fprintf(f,"CONTENT BEGIN\n");

    for(i=0;i<M;i++)
    {
        x=0.94*sin(2*PI*(i/(double)M));
        e=x-y;
        v=e+w;
        w=v;
        if(w>0)
            { y=1; b=1;}
        else
            { y=-1; b=0;}

        fprintf(f," %d : %d;\n",i,b);
    }
    fprintf(f,"END;");
    fclose(f);
    return 0;
}

```

Con el algoritmo de la Tabla 3.4, que simula el modulador, se obtiene el fichero de texto que se presenta resumido en la Tabla 3.5, y que sirve para inicializar una memoria de tipo ROM para ser implementada en la FPGA de Altera en la que se probará el filtro del convertidor. En la Figura 3.33 se muestra el esquema del diseño de la FPGA, que contiene: la memoria ROM inicializada con el contenido de la Tabla 3.5, que se obtuvo de la salida del modulador; un contador para leer la memoria secuencialmente y el bloque que implementa el filtro. El diseño de este bloque, que sigue el esquema del diagrama de bloques de la Figura 3.34, se describe mediante el código VHDL que se presenta en la Tabla 3.6.

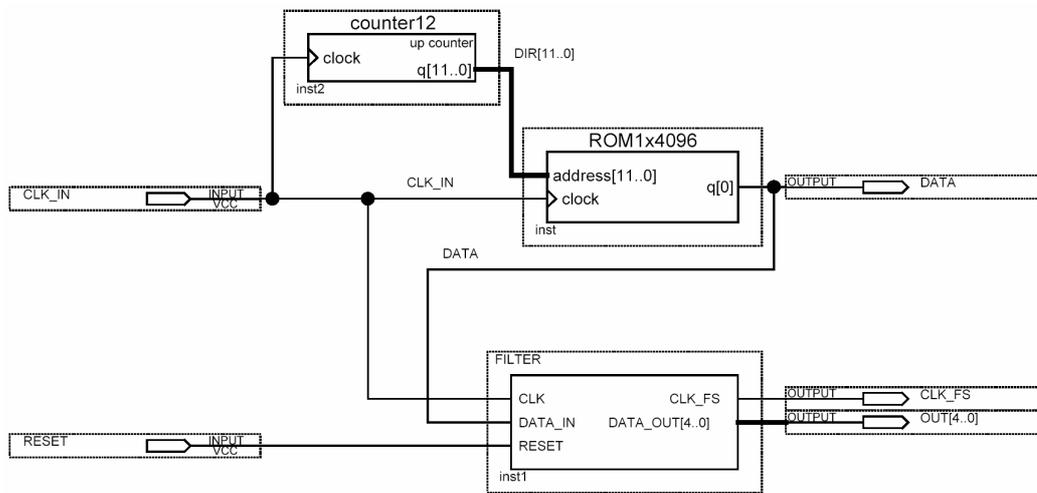


Figura 3.33. Diagrama de bloques del demodulador del ADC sigma-delta.

Tabla 3.5. Porción del contenido de la ROM con los datos de salida del modulador.

```

--Example of a generated Memory Initialization File (.mif)
WIDTH=1;
DEPTH=4096;
ADDRESS_RADIX=UNS;
DATA_RADIX=UNS;
CONTENT BEGIN
0 : 0;
1 : 1;
2 : 1;
3 : 0;
4 : 1;
5 : 0;
6 : 1;
7 : 0;
8 : 1;
9 : 0;
10 : 1;
11 : 0;
.....
4092 : 1;
4093 : 0;
4094 : 1;
4095 : 0;
END;
    
```

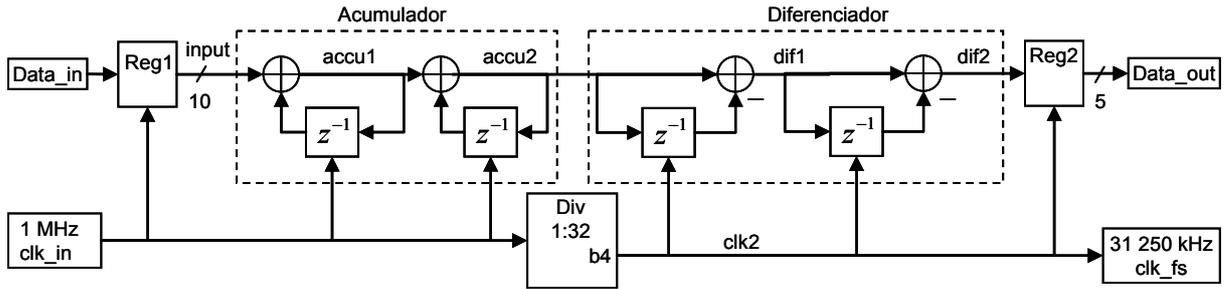


Figura 3.34. Diagrama bloques de la etapa de filtrado y diezrado del convertidor $\Sigma\text{-}\Delta$.

Tabla 3.6. Diseño en VHDL del filtro de diezrado para implementar en la FPGA.

```

-----
--filtering and decimation implementation by moving average.
--29-01-2007, Camilo Quintáns Graña
-----
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;
use ieee.std_logic_unsigned.all;

entity filter is
    port ( clk      : in std_logic;
          data_in   : in std_logic;
          reset     : in std_logic;
          clk_fs    : out std_logic;
          data_out  : out std_logic_vector(4 downto 0) );
end filter;

architecture filter_architecture of filter is
    shared variable input: std_logic_vector(9 downto 0);
    shared variable accu1: std_logic_vector(9 downto 0);
    shared variable accu2: std_logic_vector(9 downto 0);
    shared variable dif1 : std_logic_vector(9 downto 0);
    shared variable dif2 : std_logic_vector(9 downto 0);
    shared variable accu2_d: std_logic_vector(9 downto 0);
    shared variable dif1_d: std_logic_vector(9 downto 0);
    signal counter: std_logic_vector(9 downto 0);
    signal clk2: std_logic;

begin
    --data input of the filter actualization
    --matching the one bit input to the data word length
    process1: process(data_in)
    begin
        if (data_in='1') then
            input:="0000000001";
        else
            input:="0000000000";
        end if;
    end process process1;

    --implementing accumulators at the modulator rate
    process2: process(clk,reset)
    begin
        if reset='1' then
            accu1:="0000000000";
            accu2:="0000000000";
        else
            if(clk='0' and clk'event) then
                accu1:=accu1+input;
                accu2:=accu2+accu1;
            end if;
        end if;
    end process process2;

    --obtaining the lower clock
    process3: process(clk,reset)
    begin
        if reset='1' then
            counter<="0000000000";
        else
            if(clk='1' and clk'event) then
                counter<=counter+"0000000001";
            end if;
        end process process3;

    --updating the lower clock and its output
    process4: process(counter)
    begin
        clk_fs<=counter(4);
        clk2<=counter(4);
    end process process4;

    --implementing differentiators at the lower rate
    process5: process(clk2,reset)
    begin
        if reset='1' then
            dif1:="0000000000";
            dif2:="0000000000";
            accu2_d:="0000000000";
            dif1_d:="0000000000";
        else
            if(clk2='0' and clk2'event) then
                dif1:=accu2-accu2_d;
                dif2:=dif1-dif1_d;
                accu2_d:=accu2;
                dif1_d:=dif1;
            end if;
        end if;
    end process process5;

    --data output update and dividing
    process6: process(clk2)
    begin
        data_out(4)<=dif2(9);
        data_out(3)<=dif2(8);
        data_out(2)<=dif2(7);
        data_out(1)<=dif2(6);
        data_out(0)<=dif2(5);
    end process process6;
end filter_architecture;

```

Una vez simulado el filtro diezmador se introducen los datos de salida correspondientes a un periodo de la señal en la hoja de cálculo y se comparan con la sinusoidal que mejor se ajusta para obtener el error eficaz dado por (3.53). Una vez obtenido el error se obtiene la relación señal ruido mediante la ecuación (3.54) y el número equivalente de bits con (3.55). Para este convertidor se ha obtenido un ENOB de 4.8 bits.

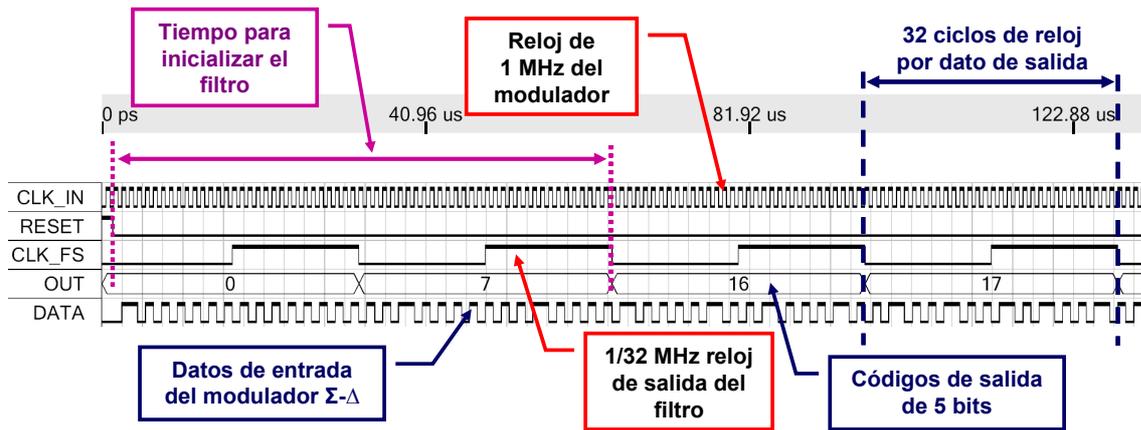


Figura 3.35. Simulación del ADC sigma-delta.

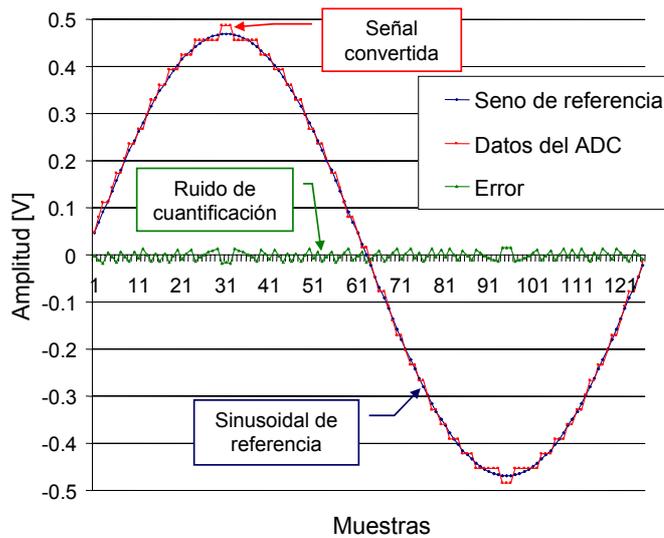


Figura 3.36. Resultados de simulación del ADC sigma-delta.

$$ruido\ rms = \sqrt{\frac{\sum_{n=1}^M \epsilon_n^2}{n}} = 0.0192\ V \tag{3.53}$$

$$SINAD = 20 \cdot \log\left(\frac{rms\ signal}{rms\ noise}\right) = 20 \cdot \log\left(\frac{0.94/\sqrt{2}}{0.0192}\right) = 30.8\ dB \tag{3.54}$$

$$ENOB = \frac{SINAD - 1.76}{6.02} = 4.84\ bit \tag{3.55}$$

3.8. Variantes en las estructuras de los convertidores A/D sigma-delta

Los moduladores de primer y segundo orden, que se analizaron en los apartados anteriores, se denominan como básicos, y no se suelen implementar para órdenes superiores a tres debido a los problemas de estabilidad. Por otro lado, en los moduladores básicos los coeficientes son unitarios y, aunque son sencillos, no dan los mejores resultados. Además, debido a la creciente aplicación de los moduladores sigma-delta en sistemas de comunicaciones, ha surgido la necesidad de desarrollar moduladores paso-banda. Por todo ello, existen diversas variantes de las estructuras básicas que se presentan a continuación:

- 1) Los moduladores multietapa. También se denominan MASH (*Multi-stage noise-Shaping*) [94] y su principal característica es que consiguen los resultados de moduladores de orden superior pero evitando los problemas de pérdida de estabilidad. En estos moduladores, en cada etapa se procesa la señal y el error de cuantificación resultante se convierte, por el modulador de la siguiente etapa, en datos digitales. Las salidas digitales de las etapas se combinan en una única salida digital que cancela el error de la etapa anterior.
- 2) Los moduladores con estructura paralela generalizada o de realimentación múltiple [28], [77]. Estos moduladores permiten optimizar los resultados fijando los coeficientes de las funciones de transferencia de la señal y del ruido. Su estructura es similar a la de los filtros IIR.
- 3) Los moduladores paso-banda [28], [95]. En estos moduladores, la banda de la señal, en vez de comenzar a frecuencia cero, lo hace a una más alta, por lo que se caracteriza por la frecuencia central y por el ancho de banda. Por tanto, la función NTF se diseña como una función rechazo-banda y la STF como una paso-banda.
- 4) Moduladores sigma-delta multibit [77]. Para conseguir aumentar la resolución del convertidor sigma-delta hay que aumentar el orden del modulador, el factor de sobremuestreo o el número de bits del cuantificador. Esta última solución se denomina multibit y mejora la estabilidad. Aunque tiene la desventaja de que la linealidad del DAC en el lazo de realimentación limita la linealidad total del convertidor, por ello esta técnica se utiliza sólo combinada con otras que corrigen estos errores.

- 5) Moduladores que combinan varias técnicas. El grado de desarrollo de las técnicas de modulación y de conversión *pipeline* (se verá en el siguiente capítulo), ha hecho que para alcanzar mayores prestaciones sea necesario combinar diversas técnicas [96]. Un buen ejemplo de ello es el convertidor sigma-delta/*pipeline* multietapa paso-banda de 80 MHz de frecuencia de conversión, 14 bits de resolución y 5 MHz de ancho de banda presentado en [97]. Otro ejemplo que combina varias técnicas es la estructura que se presenta en [98], que combina la técnica de realimentación múltiple con la de multibit. Con ello, para reducir los efectos de la no linealidad en los DAC, reduce su número de bits, siendo el ADC de N bits y los DAC de M bits ($M < N$), de esta forma intercambia no linealidad por ruido de truncado, el cual se filtra incluyendo en cada realimentación, previamente al DAC, un modulador sigma-delta digital. Otro método de diseño común, que permite alcanzar altas resoluciones con relaciones de sobremuestreo bajas, consiste en combinar la técnica MASH con la de multibit [99].
- 6) Moduladores sigma-delta para sensores. El modulador es un circuito sencillo que convierten una señal de entrada analógica a una de salida de un bit, si en vez de modificar la señal de entrada, ésta se mantiene constante y se utiliza un componente, por ejemplo, un resistor o un capacitor, del modulador como elemento sensible (sensor) a alguna magnitud física o química, entonces se consigue acondicionar y convertir la señal de dicho sensor en una sola etapa. Utilizando este principio se han desarrollado, por ejemplo: un sensor de presión capacitivo que utiliza un modulador paso-banda [100], un modulador que funciona como sensor de temperatura [101] y un acondicionador para sensores resistivos diferenciales [102]. Actualmente, se utilizan cada vez más las FPGAs para implementar los sistemas de control de los convertidores de potencia, en estos casos los moduladores Σ - Δ son sencillos y fáciles de acoplar a la FPGA (el biestable y la señal de reloj se implementan dentro de la propia FPGA) y, además, cuando se requiere obtener valores medios de las magnitudes eléctricas el post-procesado resulta sencillo y apropiado, pues se consiguen buenas resoluciones [103].

Por su importancia y difusión, a continuación se presentan las variantes del modulador básico que utilizan las técnicas de realimentación múltiple y de multietapa.

3.8.1. El modulador sigma-delta con estructura paralela generalizada

Con las estructuras paralelas (también denominados de realimentación múltiple hacia delante y hacia atrás: *Feedback-Feed-Forward* [104]) se pretende diseñar la función de transferencia del ruido de forma que se optimicen los resultados a expensas de aumentar la complejidad del modulador, tanto por su estructura como por utilizar coeficientes distintos de la unidad [105]. En la Figura 3.37 se presenta la estructura paralela generalizada del modulador sigma-delta, cuya versión de segundo orden se muestra en la Figura 3.38, y cuyas funciones de transferencia de la señal y del ruido son, respectivamente, las ecuaciones (3.56) y (3.57).

$$STF = \frac{B(z)}{A(z)} = \frac{b_1 + b_2 \cdot (1 - z^{-1}) + b_3 \cdot (1 - z^{-1})^2}{1 + (a_1 + a_2 + a_3 - 2) \cdot z^{-1} + (1 - a_2 - 2 \cdot a_3) \cdot z^{-2} + a_3 \cdot z^{-3}} \quad (3.56)$$

$$NTF = \frac{(1 - z^{-1})^2}{A(z)} = \frac{(1 - z^{-1})^2}{1 + (a_1 + a_2 + a_3 - 2) \cdot z^{-1} + (1 - a_2 - 2 \cdot a_3) \cdot z^{-2} + a_3 \cdot z^{-3}} \quad (3.57)$$

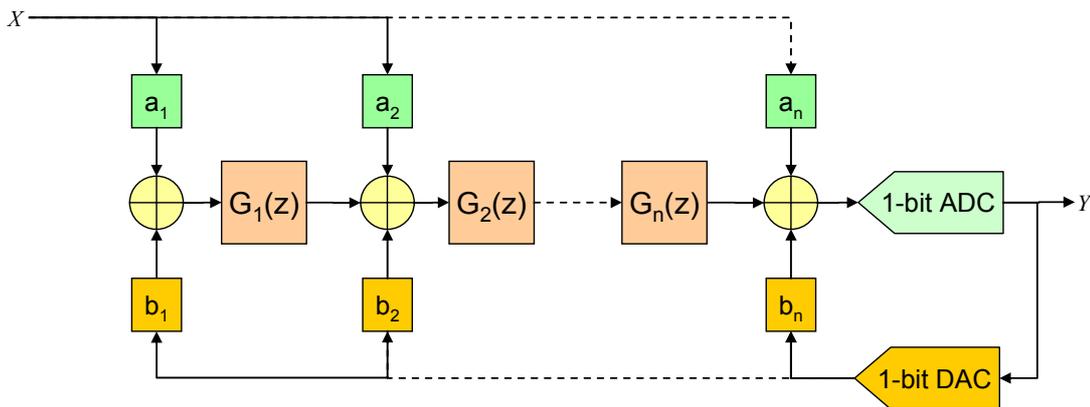


Figura 3.37. Modulador sigma-delta con estructura paralela generalizada.

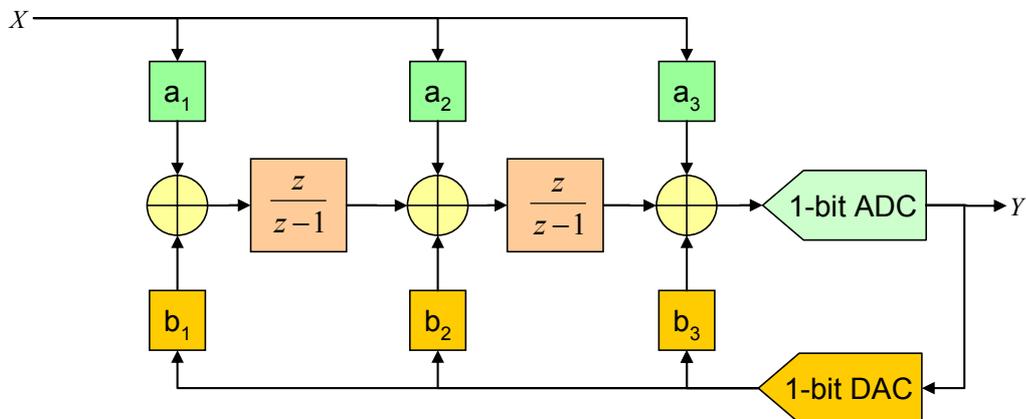


Figura 3.38. Modulador de segundo orden con estructura paralela generalizada.

3.8.2. El modulador sigma-delta multietapa

En la Figura 3.39 se muestra el diagrama de bloques correspondiente a la estructura del modulador sigma-delta multietapa, que consiste en dos moduladores básicos, uno es el que tiene la entrada x y la salida y_1 , y el otro la entrada x_2 y la salida y_2 . Se trata de probar, primero que la señal x_2 se corresponde con el error del primer modulador, de forma que sea la entrada al segundo, después, se trata de buscar las funciones G_1 y G_2 adecuadas para que ese error se cancele, pero manteniendo la función STF unitaria y la función NTF lo más selectiva posible en la banda de la señal.

En definitiva se trata de mostrar cómo se logra aumentar el orden de la función NTF sin hacer lo mismo con el del modulador. Con la estructura que se presenta con dos moduladores de segundo orden, que son estables, se consigue un modulador de cuarto orden, con la consecuente ganancia de resolución.

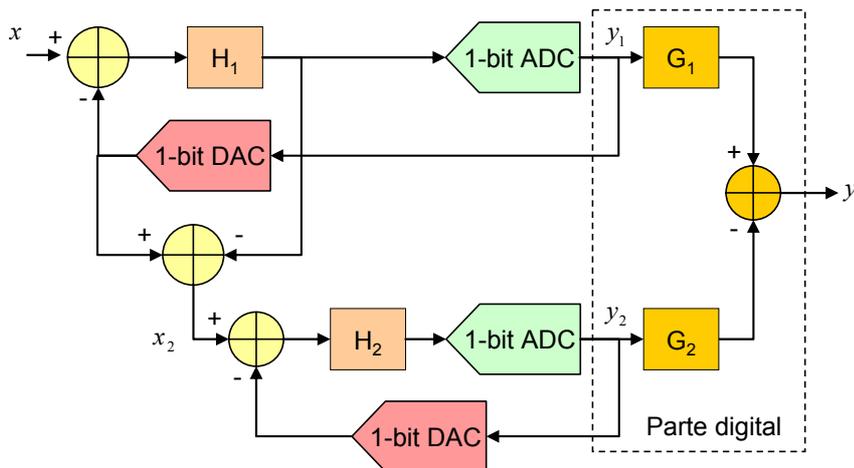


Figura 3.39. Esquema del modulador sigma-delta multietapa.

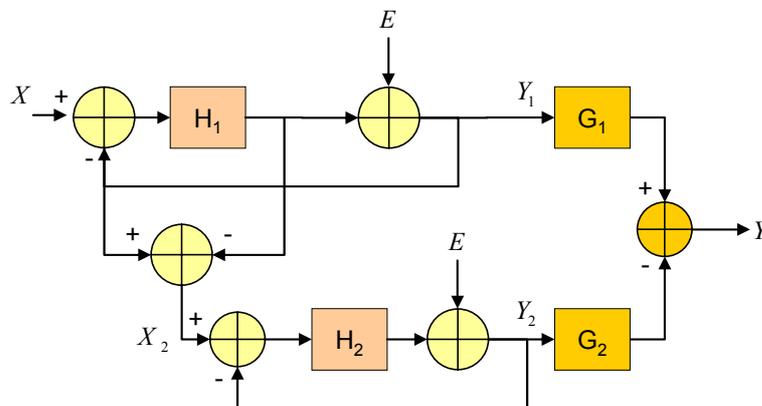


Figura 3.40. Modelo discreto del modulador sigma-delta multietapa.

Para hacer el análisis se comienza por la expresión general (3.58) del primer modulador, la cual se pone en función STF, obteniéndose la ecuación (3.59). Luego, mediante (3.60), se prueba que la entrada del segundo modulador coincide con la señal de ruido del primero, y se sustituye en la expresión (3.61), que es la función de transferencia del segundo modulador. Una vez obtenidas las funciones de ambos moduladores, se sustituyen en la ecuación general (3.62). Ahora, como interesa que se anule el efecto del ruido del primer modulador, se aplica la condición (3.63) y, aplicando el criterio de simplicidad, se llega a las expresiones (3.64) y (3.65), que se sustituyen en (3.62), dando como resultado la ecuación (3.66). Ahora, como la respuesta de la función STF debe ser unitaria, se aplica la igualdad (3.67).

$$Y_1 = STF_1 \cdot X + NTF_1 \cdot E_1 = STF_1 \cdot X + (1 - STF_1) \cdot E_1 \quad (3.58)$$

$$\begin{aligned} X_2 &= Y_1 - (X - Y_1) \cdot H_1 = Y_1 \cdot (1 + H_1) - X \cdot H_1 = \\ &= Y_1 \cdot \left(1 + \frac{STF_1}{1 - STF_1}\right) - X \cdot \frac{STF_1}{1 - STF_1} = \\ &= Y_1 \cdot \frac{1}{1 - STF_1} - X \cdot \frac{STF_1}{1 - STF_1} \end{aligned} \quad (3.59)$$

$$\begin{aligned} X_2 &= [STF_1 \cdot X + (1 - STF_1) \cdot E_1] \cdot \frac{1}{1 - STF_1} - X \cdot \frac{STF_1}{1 - STF_1} = \\ &= \frac{1 - STF_1}{1 - STF_1} \cdot E_1 + X \cdot \frac{STF_1}{1 - STF_1} - X \cdot \frac{STF_1}{1 - STF_1} = E_1 \end{aligned} \quad (3.60)$$

$$Y_2 = STF_2 \cdot X_2 + NTF_2 \cdot E_2 = STF_2 \cdot E_1 + NTF_2 \cdot E_2 \quad (3.61)$$

$$\begin{aligned} Y &= G_1 \cdot Y_1 - G_2 \cdot Y_2 = \\ &= G_1 \cdot (STF_1 \cdot X + NTF_1 \cdot E_1) - G_2 \cdot (STF_2 \cdot E_1 + NTF_2 \cdot E_2) = \\ &= G_1 \cdot STF_1 \cdot X + E_1 \cdot (G_1 \cdot NTF_1 - G_2 \cdot STF_2) - G_2 \cdot NTF_2 \cdot E_2 \end{aligned} \quad (3.62)$$

$$G_1 \cdot NTF_1 - G_2 \cdot STF_2 = 0 \quad (3.63)$$

$$G_1 = k \cdot STF_2 \quad (3.64)$$

$$G_2 = k \cdot NTF_1 \quad (3.65)$$

$$Y = G_1 \cdot STF_1 \cdot X - G_2 \cdot NTF_2 \cdot E_2 = k \cdot STF_2 \cdot STF_1 \cdot X - k \cdot NTF_1 \cdot NTF_2 \cdot E_2 \quad (3.66)$$

$$STF_2 = \frac{1}{k} \cdot STF_1 \quad (3.67)$$

Si los moduladores son de segundo orden con $STF_1 = z^{-1}$, $NTF_1 = NTF_2 = (1 - z^{-1})^2$ y $k = 2$, entonces, la función de transferencia total, suponiendo que los dos ruidos E_1 y E_2 son del mismo tipo, es la siguiente:

$$Y = STF_1^2 \cdot X - 2 \cdot NTF_1^2 \cdot E_2 = STF \cdot X + NTF \cdot E = z^{-2} \cdot X - 2 \cdot (1 - z^{-1})^4 \cdot E \quad (3.68)$$

Por tanto, las funciones de transferencia totales para el ruido y para la señal son, respectivamente:

$NTF = -2 \cdot NTF_1^2$ y $STF = STF_1^2$. Las funciones de cada modulador son:

$Y_1 = z^{-1} \cdot X + (1 - z^{-1})^2 \cdot E_1$ e $Y_2 = z^{-1} \cdot X_2 + (1 - z^{-1})^2 \cdot E_2$, y las de las funciones de salida:

$G_1 = STF_1 = z^{-1}$ y $G_2 = 2 \cdot NTF_1 = 2 \cdot (1 - z^{-1})^2$.

Queda probado que el modulador completo es de cuarto orden, aunque en la práctica es difícil cumplir la condición (3.63) y, por tanto, en la ecuación general aparecerá un término, dependiente del ruido del primer modulador, que tenderá a disminuir la ganancia de resolución.

CAPÍTULO 4. LOS CIRCUITOS CONVERTIDORES A/D PIPELINE

4.1. Introducción

La arquitectura *pipeline* [29] [30] es la más utilizada en convertidores A/D para aplicaciones que requieren una alta velocidad de conversión (desde 1 MSPS hasta 1 GSPS) y una buena resolución (ente 8 y 16 bits) a un bajo coste. Esta arquitectura combina las ventajas de los convertidores de tipo paralelo con técnicas avanzadas de tratamiento de señales, esto es, circuitos de muestreo y retención rápidos, amplificadores de gran ancho de banda y baja figura de ruido, circuitos digitales de control y de corrección del error, etc. Con las características de estos convertidores se cubre una amplia gama de aplicaciones: osciloscopios digitales, tratamiento de imagen, electromedicina, comunicaciones digitales, etc.

Las ventajas de un ADC de tipo *pipeline* son su elevada resolución y velocidad (por ejemplo, 14 bits a 100 MSPS) y su gran ancho de banda. Como desventajas destacan su elevada complejidad y la aparición de retardos.

Un convertidor *pipeline* utiliza una estructura segmentada en varias etapas que trabajan concurrentemente sobre muestras sucesivas, para ello necesita un circuito S&H en cada etapa, que lleva a cabo una conversión de uno o varios bits. Esta forma de trabajar provee de rapidez a expensas de la potencia consumida y de la latencia. Los convertidores *pipeline* necesitan de una amplificación precisa en los convertidores D/A y de amplificadores en cada etapa con buena linealidad y exactitud.

Los convertidores más rápidos se construyen utilizando tecnología bipolar, los convertidores flash de este tipo sobrepasan el GHz de frecuencia de muestreo [27], [104]. Por otro lado, los convertidores de tecnología CMOS no superan unos pocos cientos de MHz. Como compromiso intermedio se encuentran los convertidores que utilizan tecnología mixta BiCMOS.

Aunque el motivo principal por el que se consigue una velocidad de conversión elevada es la propia tecnología utilizada, sin embargo, es un tópico pensar que las altas velocidades de conversión se consiguen haciendo circuitos cada vez más rápidos, mas bien se trata de lo contrario, cada vez se consiguen velocidades más elevadas con circuitos más lentos. Esto último no es una contradicción, sino que es la consecuencia de aplicar paralelismo y concurrencia al proceso de conversión. Estos dos

principios se aplican sistemáticamente en las estructuras modernas de los convertidores. Por eso, hoy en día, lo que marca la diferencia es la arquitectura utilizada. La conversión paralela se basa en una arquitectura básica y simple que se sigue utilizando pero cuya aplicación se suele limitar a pocos bits de resolución debido a que resulta demasiado costosa y a que el consumo de potencia es elevado. Por ello ha surgido una arquitectura denominada semi-paralela (*half flash*), que divide en dos los bits que codifica el convertidor, utilizando dos convertidores paralelos de la mitad de bits que el convertidor completo. Esta filosofía de dividir se puede seguir aplicando a un número inferior de bits hasta el límite de convertir bit a bit, que es la base de los convertidores de estructura segmentada. A los convertidores con esta estructura que poseen un circuito S&H en la entrada de cada etapa, de forma que todas están realizando simultáneamente una conversión parcial de muestras sucesivas, se les denomina de tipo *pipeline*, en [107] hay un ejemplo que introduce este tipo de convertidores y en [6] hay un completo ejemplo simulado mediante PSpice en el que se muestra cómo se sincronizan las distintas etapas del convertidor.

Esta filosofía, que aplica paralelismo al menor número posible de bits (segmentación), no tendría sentido si no se realizasen estas conversiones parciales simultáneamente (conurrencia). Añadiendo circuitos de muestreo y retención a los circuitos que realizan estas conversiones parciales se pueden estar convirtiendo fracciones de muestras diferentes al mismo tiempo, con esta concurrencia se consigue que se obtenga una conversión completa en cada ciclo de reloj, el único peaje que se paga es la aparición de latencia en los datos de salida.

Estas nuevas estructuras han posibilitado la construcción de convertidores rápidos de alta resolución y bajo consumo de potencia, en poco espacio de silicio y en una tecnología económica como es la CMOS. En [35] hay una completa exposición de los circuitos y sistemas basados en tecnología CMOS para instrumentación y procesado, continuo y discreto, especialmente los basados en tecnología de capacidades conmutadas o SC (*Switched-Capacitor*).

4.2. Antecedentes de los convertidores A/D pipeline

Tradicionalmente se han utilizado los convertidores A/D de tipo paralelo para las aplicaciones en las que es necesario adquirir señales analógicas de frecuencias elevadas. Estos convertidores se caracterizan por tener una baja resolución y un elevado coste en precio y en consumo de energía. Por ello se han desarrollado nuevos circuitos A/D que combinan el paralelismo con la segmentación y que tienen estructuras más complejas. En un principio, se pasó de los convertidores A/D paralelos o *flash* a los semiparalelos o *half-flash*, que tienen dos etapas, y posteriormente a los multietapa. Luego, cuando se le añadió un circuito S&H a la entrada de cada etapa, los convertidores multietapa pasaron a denominarse *pipeline*.

Los convertidores semiparalelos siguen el principio de funcionamiento basado en el viejo refrán “divide y vencerás”. Con este principio la señal analógica se convierte utilizando dos convertidores paralelos en cascada de la mitad de bits ($N/2$) que el convertidor completo (N). Esto reduce el número de comparadores de $2^N - 1$ a $2 \cdot (2^{N/2} - 1)$ y, en la práctica, se suelen hacer de entre 15 y 63 comparadores (de 4 a 6 bits). Como contrapartida hay que señalar que el hecho de utilizar etapas en cascada provoca un aumento del tiempo de conversión total, que es, aproximadamente, del doble.

En la Figura 4.1 se muestra el diagrama de bloques de un convertidor de tipo semiparalelo de 6 bits (N). Las señales de control Φ_1 y Φ_2 sincronizan los eventos en las distintas etapas del convertidor según se representa en la Figura 4.2. En el instante t_1 se inicia una conversión, que se lleva a cabo en los siguientes pasos: primero la señal Φ_1 da la orden de muestrear la señal de entrada y de retenerla (S&H), luego se realiza una conversión gruesa (*Coarse Flash ADC*) de tres bits de la señal retenida que serán los tres bits más significativos (3 MSB) del dato de salida, a continuación se convierten a analógico (DAC) estos tres bits y la señal que se obtiene se sustrae de la señal de entrada que fue retenida en el primer paso, el resultado de esta resta se denomina residuo. Transcurrido el tiempo necesario para realizar estas operaciones, la señal Φ_2 genera la orden de realizar la conversión fina (*Fine Flash ADC*) del residuo para obtener los tres bits menos significativos (3 LSB).

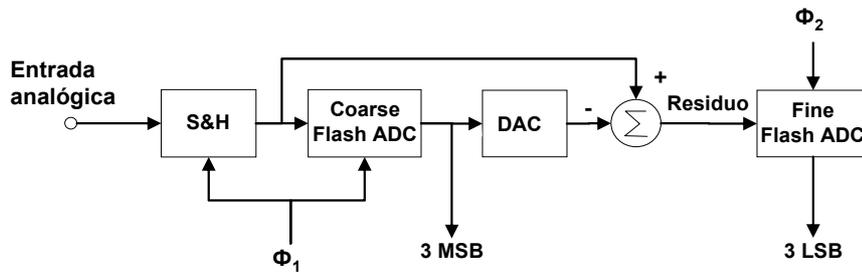


Figura 4.1. Diagrama de bloques básico de un convertidor semiparalelo.

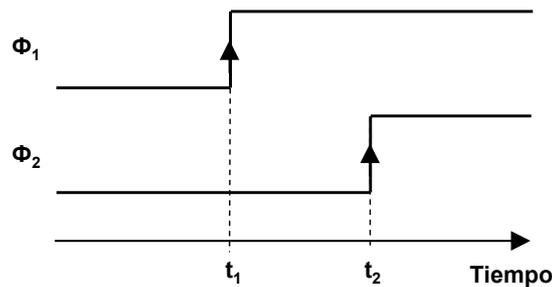


Figura 4.2. Diagrama de tiempos de las señales de sincronismo del convertidor semiparalelo.

Hay que señalar que solamente hay un instante en el que se realiza el muestreo, que corresponde a t_1 . Por tanto hay que contar con los retardos de la cadena de circuitos que hay desde el circuito de muestreo de entrada hasta la última etapa para poder disponer de un código de salida válido.

Sin embargo, si se muestrease y retuviese el residuo, mientras se realiza la segunda conversión se podría realizar un nuevo muestreo en la etapa de entrada, de esta forma se conseguiría duplicar la velocidad del convertidor a costa de introducir una latencia de un ciclo de la señal de sincronismo. A los convertidores que utilizan esta técnica en la que se divide la conversión y se realiza de forma concurrente en dos etapas se les denominan *pipeline* de dos etapas o *sub-ranging* [108].

La estructura *sub-ranging* se utiliza ampliamente para conseguir altas resoluciones a alta velocidad sin las limitaciones de coste y potencia de los convertidores paralelos. Por ejemplo, un convertidor de 12 bits de tipo paralelo utilizaría $2^{12} - 1 = 4095$ comparadores y, en cambio, uno de tipo *sub-ranging* sólo utilizaría $2 \cdot (2^6 - 1) = 126$. Esta diferencia tan grande justifica el aumento de la complejidad del convertidor *sub-ranging* que necesita de circuitos auxiliares que no hay en un convertidor paralelo.

La Figura 4.3 representa la estructura básica de un ADC de 6 bits en el que la conversión se divide en dos etapas de tres bits cada una. Las dos etapas se basan en un convertidor A/D paralelo y un

DAC, ambos de 3 bits. El S&H₁ toma una muestra de la entrada analógica en el instante t_1 para que el ADC₁ realice la conversión gruesa en la que se obtienen los tres bits más significativos. Como se observa en la Figura 4.4, el dato estará entre las combinaciones 101000_b y 110000_b. Al mismo tiempo estos bits se convierten a analógico mediante el DAC, cuya salida se sustrae de la señal total en el bloque restador, obteniéndose el residuo de la señal, que todavía queda por convertir. El residuo se amplifica en un factor G para adaptarlo al rango de entrada del ADC₂. El factor G depende del número de bits de las etapas, si cada etapa es de k bits, entonces el factor G vale 2^k . En el instante t_2 la parte alta se memoriza en el registro de la primera etapa y el residuo se muestrea en el S&H₂. Ahora, mientras se convierte el residuo de la muestra actual de la que se han memorizado los bits más significativos, el S&H₁ toma otra muestra para realizar una nueva conversión.

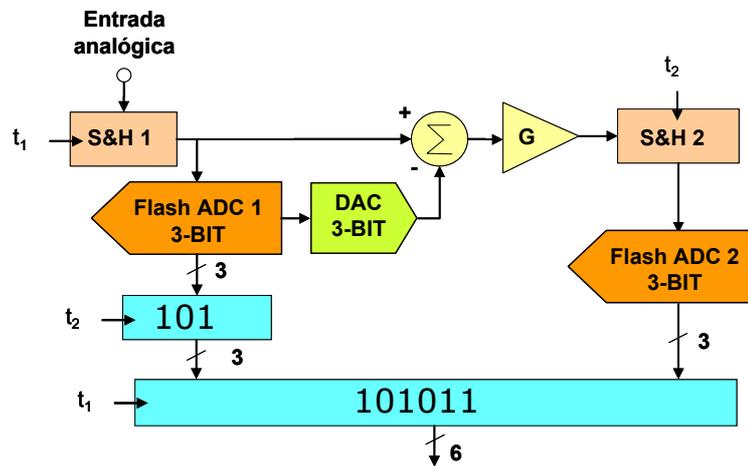


Figura 4.3. Diagrama de un convertidor de dos etapas pipeline.

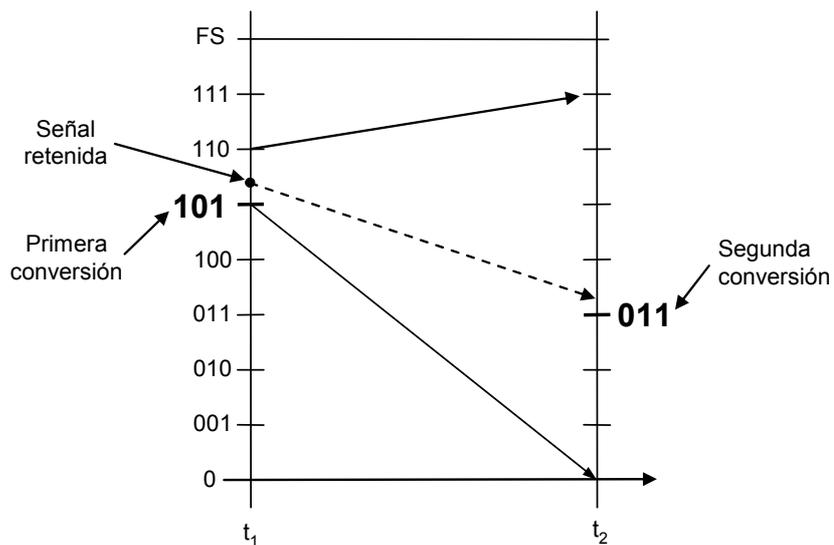


Figura 4.4. Principio de funcionamiento de un ADC de dos etapas.

4.3. Principio de funcionamiento de los convertidores A/D pipeline

Un convertidor *pipeline* utiliza una estructura segmentada en varias etapas que trabajan concurrentemente sobre muestras sucesivas, para ello, cada etapa necesitan un circuito S&H a su entrada. Cada etapa realiza una conversión parcial de uno o varios bits. Esta forma de trabajar proporciona una gran rapidez de conversión a expensas de la latencia.

La Figura 4.5 muestra el diagrama de bloques de una posible arquitectura para un ADC *pipeline* de 12 bits de resolución.

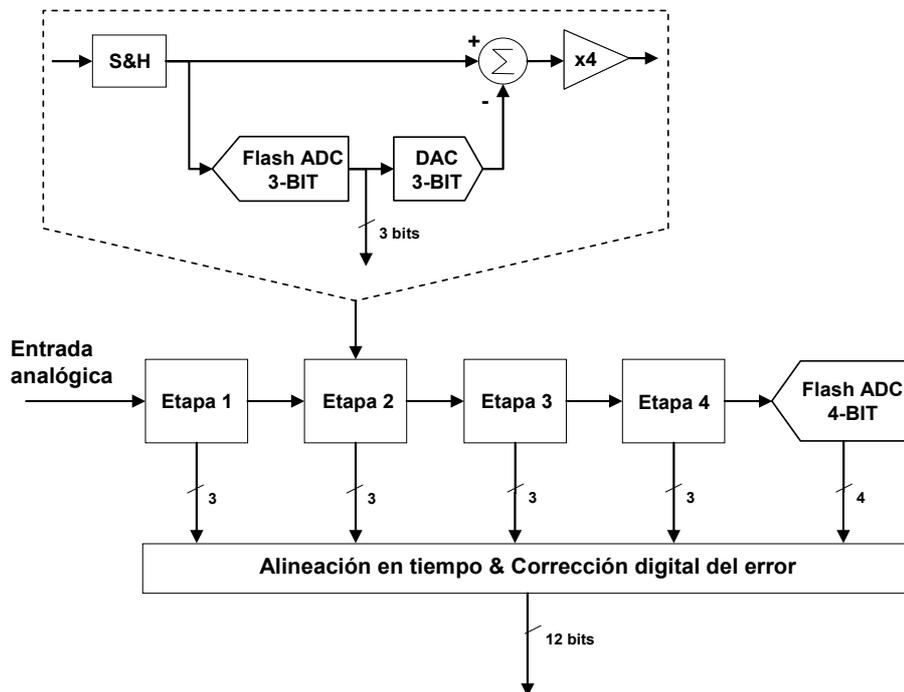


Figura 4.5. Arquitectura de un ADC de tipo pipeline con cuatro etapas de 3 bits (cada una codifica 2 bits).

En esta estructura, mediante un amplificador S&H, la entrada analógica se muestrea y, mientras la retiene, la etapa 1 la cuantifica a una combinación binaria de 3 bits. Entonces, estos 3 bits son, de nuevo, convertidos a analógico y sustraídos de la señal de entrada. A continuación el residuo se multiplica por un factor 4 de ganancia, esta señal obtenida es la entrada a la etapa 2. Estas operaciones sobre la señal se repiten en las etapas 2, 3 y 4 hasta llegar al convertidor ADC de 4 bits final, el cual cuantifica los 4 LSB. Como los bits de cada etapa se determinan en instantes diferentes de tiempo, todos los bits correspondientes a la misma muestra se ordenan (alinean en el tiempo) mediante registros de desplazamiento antes de ser pasados al circuito de corrección. Tan pronto como una etapa termina de procesar una muestra, de codificarla y de pasarle el residuo a la siguiente etapa, gracias a

que hay un S&H entre ambas, puede comenzar una nueva conversión. Esta forma de trabajar en cadena es la forma en que se consigue una alta velocidad de conversión.

El circuito de corrección disminuye considerablemente la exactitud necesaria en los convertidores A/D paralelos y, por tanto, en cada comparador individualmente. Como se comprueba en la Figura 4.5 la salida del restador tiene un margen dinámico de un octavo respecto a la etapa de entrada y, sin embargo, sólo se multiplica por un factor de ganancia 4. Por tanto, la entrada de la etapa 2 solo utiliza la mitad del rango del convertidor (esto es así cuando no se produce ningún error).

Si en un ADC paralelo de una etapa n del convertidor de la Figura 4.5, hay un comparador con un error de *offset* significativo, cuando una señal de entrada atraviesa este punto de la excursión del comparador, se produce un código de 3 bits incorrecto y, por tanto, también una salida incorrecta en el DAC que generará un residuo diferente. Sin embargo, gracias a que sólo se aprovecha la mitad del rango de entrada de la etapa $n+1$, si el residuo es mayor que el que debiera, dicha etapa no se saturará. Esta cantidad extra, generará un código mayor que el que debiera en la etapa n , pero será restado en la misma cantidad de la señal de entrada, lo cual generará un código final de salida correcto.

La consecuencia es que ninguno de los ADC de la Figura 4.5 tiene que ser tan exacto como el ADC completo. De hecho, los ADCs de 3 bits de las etapas de la uno a la cuatro necesitan solamente 4 bits de exactitud. La corrección digital del error no funcionará para el convertidor de 4 bits del final. Sin embargo, cualquier error generado aquí se minimiza debido a la ganancia acumulada en las etapas precedentes, siendo suficiente una resolución de 4 bits.

En el ejemplo de la Figura 4.5, aunque cada etapa genera 3 bits brutos, debido a que la ganancia entre etapas es sólo de 4, cada etapa (de la uno a la cuatro) codifica sólo 2 bits efectivos. El bit extra simplemente es para reducir el tamaño del residuo a la mitad, permitiendo un rango extra en el siguiente ADC de 3 bits para la corrección digital del error, como se explicó anteriormente. Esto se denomina solapamiento de 1 bit entre etapas adyacentes. El número efectivo de bits del convertidor completo es de $2+2+2+2+4=12$ bits.

Debido a que cada muestra tiene que propagarse a través de la cadena completa de etapas antes de que estén disponibles todos los bits para el circuito de corrección del error, aparece una latencia de los

datos asociada a los ADC con arquitectura *pipeline*. En el ejemplo de la Figura 4.5 la latencia es de cuatro ciclos, tal y como se muestra en la Figura 4.6.

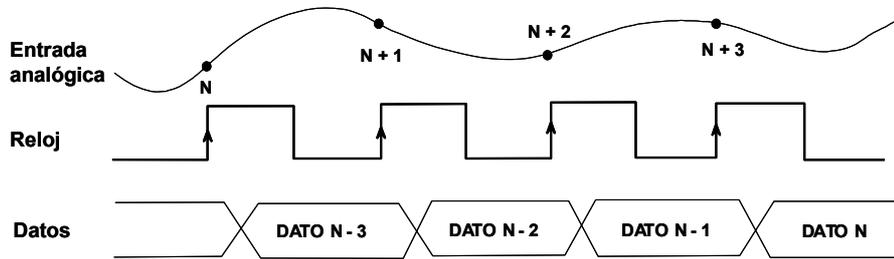


Figura 4.6. Latencia en los convertidores pipeline.

En la Figura 4.7 se muestra una etapa pipeline de 1 bit, la referencia de tensión es de 2.5 V y el rango de entrada es de 5 V unipolar. El funcionamiento de un ADC, por ejemplo con $k=6$ etapas, suponiendo que en la entrada (etapa 1) hay una tensión de 2.000 V sería como sigue (Tabla 4.1). Cada fila de la tabla corresponde con una etapa del convertidor, la tensión de entrada es la de la primera etapa, o sea 2 V, como es menor que la tensión de referencia $V_{ref}=2.5$ V, entonces la salida de la primera etapa es un '0'. Como la salida ha sido un '0' esto produce 0 V a la salida del DAC, por tanto el residuo será de $(V_{in}-V_{DAC}) \times 2 = 4$ V. Ahora el residuo de la etapa 1 es la entrada de la etapa 2, la cual genera a su vez un residuo de 3 V y un bit de salida de '1'. El proceso continua hasta llegar a la última etapa. Finalmente, la combinación de salida correspondiente a los 2 V de entrada sería **011001_b**. Hay que resaltar que, como cada etapa tiene un S&H, una vez calculado el residuo y el bit de salida, se puede realizar la siguiente conversión porque la etapa siguiente memoriza el residuo. De esta forma todas las etapas están realizando simultáneamente la conversión de 6 muestras consecutivas.

Tabla 4.1. Funcionamiento de un ADC pipeline de 6 etapas.

Etapa	Vin	ADC	Res
1	2	$2 < 2.5 \Rightarrow '0'$	$(2-0.0) \times 2 = 4$
2	4	$4 > 2.5 \Rightarrow '1'$	$(4-2.5) \times 2 = 3$
3	3	$3 > 2.5 \Rightarrow '1'$	$(3-2.5) \times 2 = 1$
4	1	$1 < 2.5 \Rightarrow '0'$	$(1-0.0) \times 2 = 2$
5	2	$2 < 2.5 \Rightarrow '0'$	$(2-0.0) \times 2 = 4$
6	4	$4 > 2.5 \Rightarrow '1'$	$(4-2.5) \times 2 = 3$

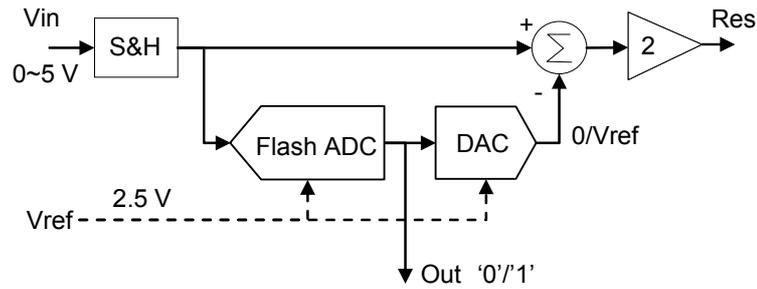


Figura 4.7. Ejemplo de una etapa pipeline de 1 bit.

4.4. Diseño y simulación de un convertidor A/D pipeline mediante OrCAD PSpice

En este apartado se realiza un convertidor *pipeline* de 5 bits. Se analiza su estructura, se hace un diseño del hardware utilizando componentes discretos, a continuación se simula utilizando el software OrCAD PSpice y, finalmente, se analizan los resultados para caracterizar el convertidor utilizando el software Matlab. Además, se expone las metodologías seguidas para sincronizar las etapas del convertidor y para caracterizarlo.

4.4.1. Diseño del convertidor A/D pipeline de 1 bit por etapa

La Figura 4.8 muestra el diagrama de bloques del convertidor A/D *pipeline*, que se compone de cinco etapas de un bit cada una. Las cinco etapas son iguales a excepción de la última, la cual no tiene los componentes de la salida: el multiplexor, el restador y el amplificador.

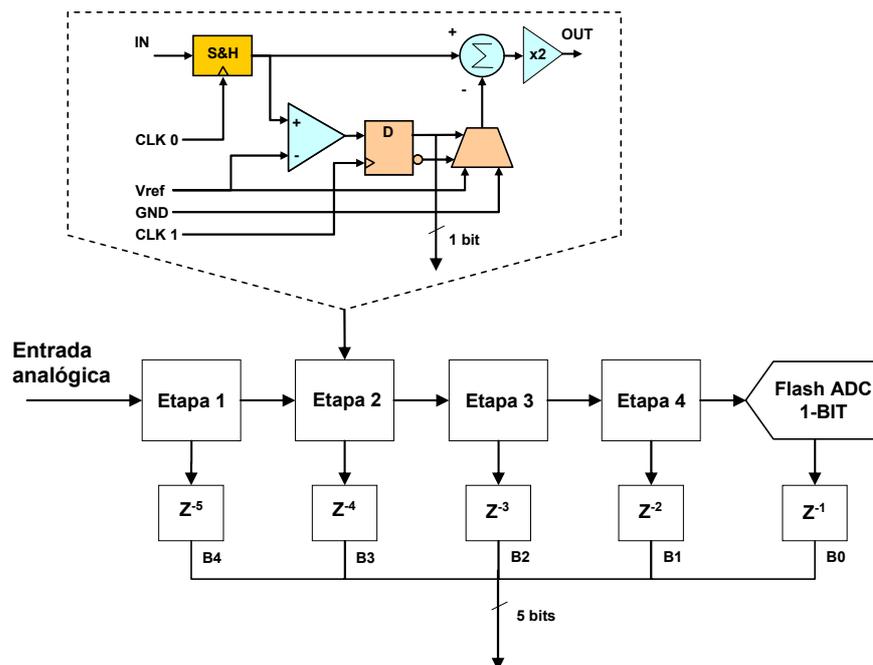


Figura 4.8. Estructura del ADC pipeline simulado.

Cada etapa comienza con un circuito S&H que memoriza la salida de la etapa anterior, a excepción de la primera etapa que memoriza la señal de entrada. Su salida se compara con la señal de referencia y la salida del comparador se registra en un biestable cuya salida es el bit obtenido en dicha etapa. Si la señal retenida en el S&H es mayor que la de referencia, ésta última se sustrae de la primera, y si es menor no se sustrae nada. Finalmente, se amplifica el resultado de la diferencia obteniéndose el valor de salida OUT, también llamado residuo. Cada etapa proporciona un bit, la primera el MSB y la última el LSB. El primer bit aparece en la salida con un retardo de 5 ciclos de reloj respecto al último, que es la latencia que genera el convertidor. Uno de los aspectos más importantes a la hora de obtener un resultado satisfactorio en el funcionamiento de un convertidor *pipeline* es la buena secuenciación de las señales de reloj. En la Figura 4.9 aparece el diagrama de tiempos correspondiente a la secuencia de las señales de reloj. En el instante t_1 se toma una muestra en el S&H de entrada, tras un tiempo aparece un nuevo valor analógico a su salida que se compara con la señal de referencia. Después del tiempo que lleva la operación de comparación, el bit de salida del comparador está estable en la entrada del biestable, en el instante t_2 se memoriza su valor. A continuación el valor del nuevo bit hace aparecer la señal a la entrada negativa del restador, se realiza la operación de resta y se amplifica en un factor de 2. Después del tiempo necesario para que el resultado se estabilice se puede realizar el muestreo de éste en la siguiente etapa, que corresponde al instante t_3 . Como se puede comprobar hay tres instantes que se deben secuenciar, por lo que son necesarias tres señales de reloj, CLK0, CLK1 y CLK2.

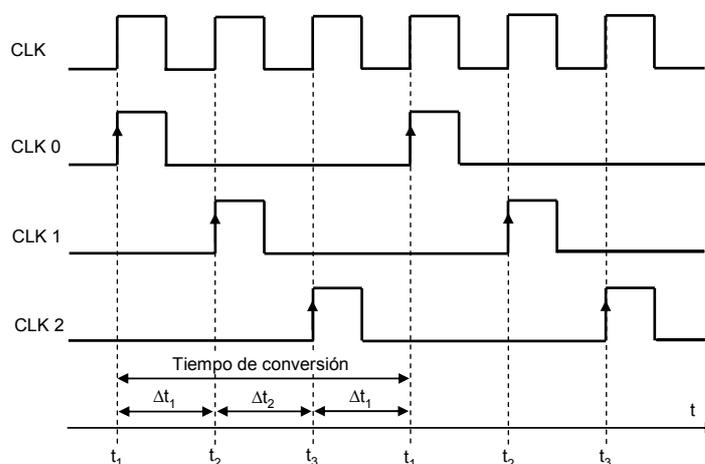


Figura 4.9. Señales de sincronización del ADC pipeline simulado.

El convertidor esta diseñado para trabajar en modo unipolar entre 0 y 5 V, ya que la señal de referencia VREF se ha fijado a 2.5 V. En la Figura 4.13 se presenta el esquema electrónico completo del convertidor *pipeline*.

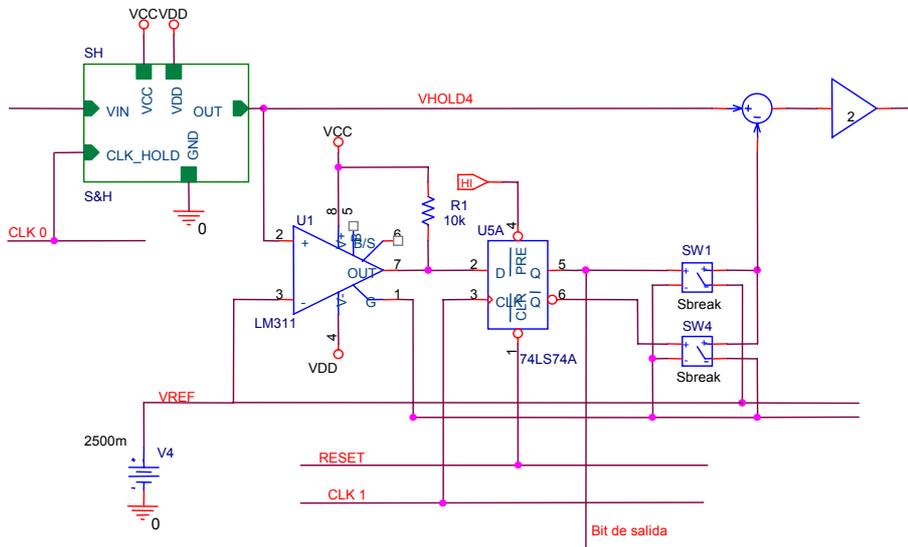


Figura 4.11. Esquema de una etapa del ADC pipeline simulado.

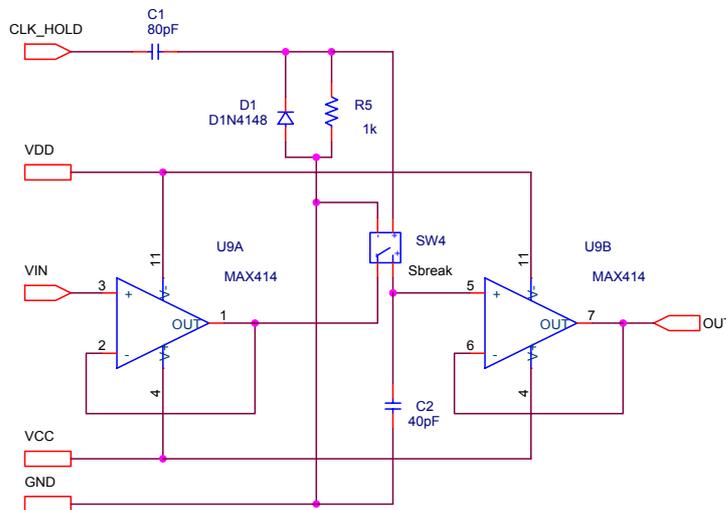


Figura 4.12. Esquema del circuito S&H.

Para obtener las tres señales de reloj se parte de un reloj base CLOCK que activa un contador binario de módulo 3 cuya salida de dos bits se conecta a un decodificador binario-decimal, el cual irá activando sus tres salidas de menor peso secuencialmente, obteniéndose las tres señales de reloj necesarias (Figura 4.13).

Los recursos utilizados por un convertidor de estas características varían de acuerdo con el número de bits N según la Tabla 4.2.

Tabla 4.2. Recursos utilizados por los ADC pipeline de N bits de 1-Bit por etapa.

Recurso	Cantidad
S&H	N
Restadores	N-1
Amplificadores	N-1
Conmutadores	N-1
Comparadores	N
Memorias 1 Bit	$(N^2+N)/2$

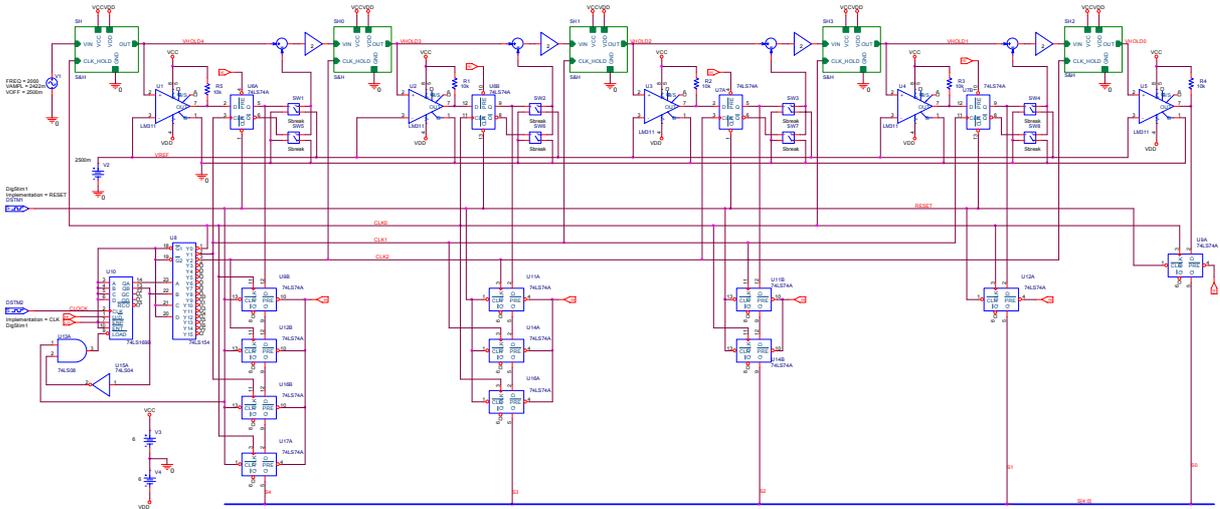


Figura 4.13. Esquema electrónico del ADC de tipo pipeline de 5 bits simulado mediante OrCAD PSpice (PP1).

4.4.3. Simulación del convertidor pipeline de 1 bit por etapa

Para simular el convertidor se ha utilizado un estímulo para el reloj base (CLOCK) de 600 kHz, por lo que la frecuencia de muestreo resultante es de 200 kHz. Los 5 bits de salida se han agrupado en un bus denominado S[4:0]. La señal analógica a convertir es una onda sinusoidal de 2 kHz de frecuencia, de 2 422 mV de amplitud y de 2.5 V de *offset*.

El tiempo de simulación total ha sido de 550 μ s. En la Figura 4.14 se presenta una fracción del tiempo de simulación donde se pueden ver los valores de salida del convertidor y las señales de reloj, así como la señal analógica de entrada que se está convirtiendo (SH.VIN) y la salida del S&H de la primera etapa (VHOLD4).

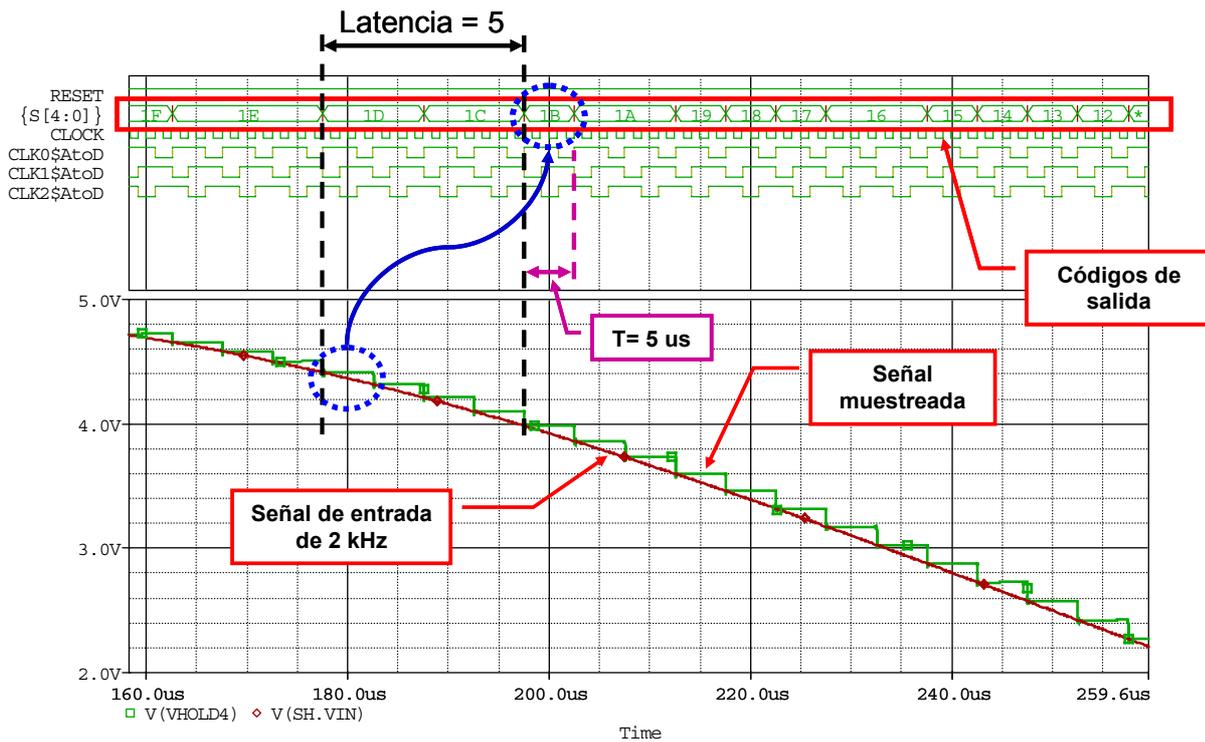


Figura 4.14. Ampliación de una porción del resultado de la simulación del ADC pipeline de 5 bits.

4.4.4. Caracterización de la dinámica del convertidor A/D pipeline

Para caracterizar la dinámica del convertidor simulado se sigue el procedimiento que se presentó en el Apartado 2.8.2. Para ello se utiliza el programa de Matlab de la Tabla 4.3, que implementa los dos tipos de análisis, en el dominio de la frecuencia y del tiempo.

Para poder representar gráficamente los vectores de muestras y de referencia en el dominio del tiempo se genera un vector de tiempos (t), y para representar en el espacio de la frecuencia las componentes del desarrollo se genera un vector de frecuencias (frec). Este vector va desde *dc* hasta la mitad de la frecuencia de muestreo, o sea, cubre la banda de Nyquist.

Para hacer el análisis temporal se genera un vector de valores correspondientes al periodo de una señal sinusoidal (signal) como la que se inyectó al convertidor en la simulación. Esta señal servirá de referencia en los cálculos.

A continuación se realiza el desarrollo en series de Fourier del vector de muestras, para lo cual se utiliza la función de la Tabla 4.4, el resultado del desarrollo se muestra en la Figura 4.14. De la componente de *dc* se puede calcular directamente el error de *offset* del convertidor. Después, con la componentes fundamental y armónicas de la señal se calcula la THD. Luego, se procede al cálculo de

la SINAD, en escala logarítmica, en la que se compara los valores eficaces de la componente fundamental y del ruido de cuantificación. El ruido se obtiene calculando el valor eficaz del vector de ruido, que es la diferencia entre los vectores de la sinusoidal de referencia y de salida del convertidor, como se muestra en la Figura 4.16. Finalmente, con la relación señal ruido calculada a partir de los resultados prácticos (la real) se deduce el número equivalente de bits o ENOB. En la Tabla 4.5 se hace un resumen de las características del convertidor simulado.

Tabla 4.3. Listado del programa de Matlab para el test del ADC pipeline de 5 bits.

<pre> %% %UNED-UVIGO. CAMILO QUINTÁNS GRAÑA, 14-08-2005 % %TEST DE UN CONVERTIDOR PIPELINE DE 5 BITS % %DE 1 BIT POR ETAPA % %% %Se ha simulado el convertidor a una frecuencia %de muestreo de 200 kHz %Se le ha inyectado una señal a convertir %senoidal de amplitud 2422 mV de 2 kHz y %VDC=2500 mV %data es la secuencia de los datos obtenidos del %convertidor %Se obtiene una senoidal correspondiente a la %señal que se ha muestreado for i=1:100, signal(i)=sin(((i-1)*2*pi)/100), end signal=signal' signal=signal*2.422+2.500 %Se calculan las componentes de la serie de %fourier para el análisis sf=dfs(data,100,51) sf=sf' %Se obtiene un vector de frecuencias que %corresponda con la banda de Nyquist for i=1:51, frec(i)=(i-1)*(2000), end %La primera componente es la de continua dc=sf(1) %La siguiente es la fundamental fundamental=sf(2) %Se obtiene un vector de tiempos t=[0:99]/200000 t=t' %Se calcula el valor eficaz de la fundamental %y de todas las componentes de la serie fundamental_rms=fundamental/sqrt(2) for i=1:51, sf_rms=sf/sqrt(2), end </pre>	<pre> %Cálculo de la distorsión armónica total teniendo %en cuenta todas las componentes en la banda de %Nyquist suma_armonicos=0 for i=3:51, suma_armonicos=suma_armonicos+sf_rms(i)^2, end THD=sqrt(suma_armonicos/fundamental_rms^2) THD=THD*100 %THD=2.3483% %Se calcula el error de continua error_DC=2.500-dc %error_DC=0.0766 %Se obtienen los valores de AC de signal % y de data signal_AC=signal-2.5 signal_AC=signal_AC' data_AC=data-dc %cálculo del ruido de cuantificación ruido=signal_AC-data_AC %Cálculo del ruido en valor eficaz sumaruido=0 for i=1:100, sumaruido=sumaruido+ruido(i)^2, end ruido_rms=sqrt(sumaruido/100) %ruido_rms=0.0717 %Cálculo de la relación señal ruido SNR=20*log10(fundamental_rms/ruido_rms) %SNR=27.5719 dB %Cálculo del número efectivo de bits ENOB=(SNR-1.76)/6.02 %ENOB=4.2877 bits %FIN </pre>
--	---

Tabla 4.4. Función para el cálculo de la serie de Fourier de una secuencia.

```

%Calcula la amplitud de las nfs componentes
%de la serie de fourier de una secuencia s
%de tamaño ts
function[mag]=dfs(s,ts,nfs)
for k=1:nfs
    ak=0
    bk=0
    for i=1:ts
        ak=ak+s(i)*cos((2*pi*(i-1)*(k-1))/ts)
        bk=bk+s(i)*sin((2*pi*(i-1)*(k-1))/ts)
    end
    if k==1
        ak=ak/ts
    else
        ak=ak*2/ts
    end
    bk=bk*2/ts
    mag(k)=sqrt(ak^2+bk^2)
end
    
```

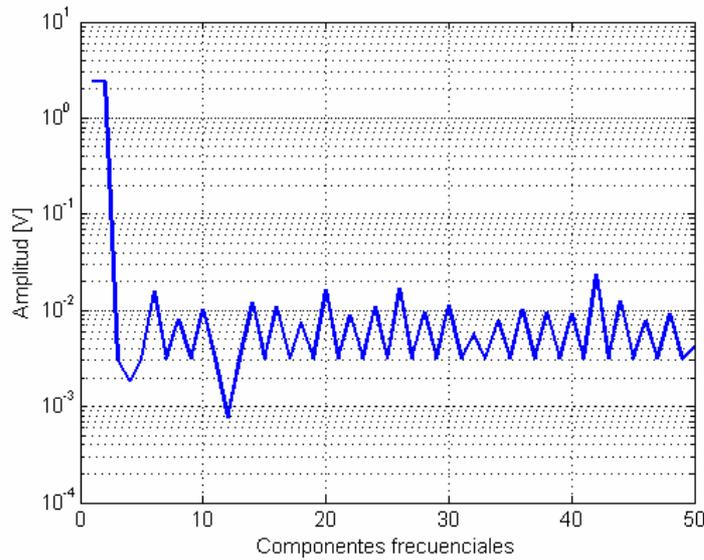


Figura 4.15. Componentes de frecuencia de la señal obtenida a la salida del convertidor.

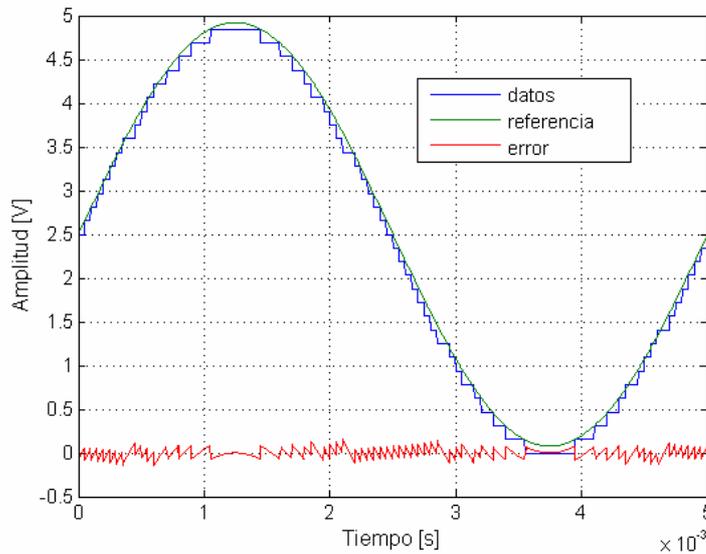


Figura 4.16. Señales de referencia, de salida del convertidor y de ruido.

Tabla 4.5. Resumen de las características del ADC pipeline simulado ($f_s = 200$ kHz).

Parámetro	Valor del parámetro
Tipo de convertidor	<i>Pipeline</i>
Número de bits	5
Tipo de entrada	Unipolar
Rango de entrada	De 0 V a FS-1LSB: 0 V- 4.84 V
THD	2.35%
SINAD	27.6 dB
ENOB	4.28 b
Error de <i>offset</i>	76.6 mV

4.5. Variantes en la estructura de un convertidor A/D *pipeline*

Los convertidores AD de tipo *pipeline* son relativamente recientes y en los últimos años han ido apareciendo modificaciones en las estructuras convencionales de los circuitos que los forman. Los objetivos que se han perseguido han sido: por un lado, el ahorro en espacio, complejidad y coste, y por otro, el aumento de las prestaciones, sobre todo, disminución de los parámetros INL y DNL, y el aumento de la relación señal ruido y de la velocidad. Además, en todas las mejoras que han supuesto un avance en las prestaciones se ha tenido en cuenta el consumo de potencia, ya que el uso de este tipo de convertidores se ha visto frenado en algunas aplicaciones debido al relativamente excesivo consumo. A continuación se resumen algunos de los cambios introducidos en la estructura de este tipo de convertidores.

Estructuras paralelas con varios ADC pipeline pseudo diferenciales

Para aumentar la velocidad de los ADCs *pipeline* se puede recurrir a una estructura que implemente en paralelo varios convertidores, en la referencia [109] se representa una de estas soluciones. Además, presenta otra característica importante, implementa las etapas con amplificadores diferenciales para reducir los efectos del ruido.

Estructuras de DACs modificados en vez de los totalmente paralelos o “Full Flash”

En las sucesivas etapas de los convertidores ADC *pipeline* se utilizan convertidores DAC de tipo *flash*, estas arquitecturas tienen un elevado consumo de potencia, utilizan un elevado número de comparadores y tienen una elevada capacidad de entrada. Una arquitectura para los DAC que combina comparadores con multiplexores analógicos reduce la complejidad del circuito y la disipación de potencia. Esta estructura *flash* modificada de convertidor DAC utiliza comparadores optimizados con

memorización para conseguir que se comporte como un ADC totalmente paralelo. El resultado es un ahorro elevado de comparadores, consumo de potencia y espacio de integración [110].

Estructuras MDAC para los convertidores D/A de las etapas de un ADC pipeline

En los ADCs *pipeline* de múltiples bits por etapa se suelen utilizar convertidores D/A del tipo MDAC (*Multiplying ADC*) que se basan en una técnica denominada MCS (*Merged-Capacitor Switching*) que utiliza una estructura de capacidades conmutadas. Esta estructura presenta la ventaja de obtener buenas prestaciones a alta velocidad. Otra técnica derivada de ésta es la denominada CFCS (*Commutated Feedback-Capacitor Switching*), que utiliza una estructura modificada que reduce el número de capacidades y conmutadores necesarios así como los recursos de conexión debido a que incorpora la realimentación en el circuito de conversión [111].

Estructuras que reducen el número de amplificadores

En los convertidores que utilizan capacidades conmutadas es posible reducir el número de amplificadores [112]. Una técnica utilizada consiste en compartir el amplificador de salida de cada etapa con el amplificador de muestreo y retención de la siguiente. Esto es posible si se modifica la topología utilizada en este amplificador introduciendo la capacidad de retención de una etapa en la realimentación del amplificador de salida de la etapa anterior, constituyendo un amplificador telemétrico [113].

Estructuras optimizadas para sistemas de adquisición

Los fabricantes de instrumentación de alta velocidad han desarrollado sus propios circuitos convertidores de alta velocidad que incorporan generadores de reloj, circuitos T&H (Track & Hold), bloques de memoria de adquisición y circuitos de multiplexado. En [114] se realiza una fabulosa presentación de un convertidor de estas características que alcanza las 20 GSPS.

Estructuras que incorporan la corrección del error íntegramente en la parte digital

En [115] se presenta una arquitectura de ADC pipeline que pasa la parte de conversión de código termométrico del DAC paralelo a la parte digital, y se modifica el convertidor ADC siguiente para que convierta directamente la combinación en código termométrico a analógico. El autor denomina esta técnica como “Cancelación del Error en el DAC”.

Etapa de entrada en los convertidores A/D de alta velocidad

Los convertidores A/D necesitan un circuito de entrada que tome una muestra de la señal a convertir y la mantenga mientras se realiza la conversión, estos circuitos se denominan S&H o T&H, en [116] hay una exposición más amplia de esta terminología.

En la Figura 4.17 se presenta, a modo de ejemplo, un circuito T&H diferencial de capacidades conmutadas con tecnología CMOS. Las posiciones de los interruptores corresponden al estado *Track*. Los interruptores se abren y cierran a la frecuencia de muestreo. Los condensadores de 16 pF representan la suma de la capacidad equivalente de los interruptores S1 y S2 y de la capacidad de pérdidas de entrada. Los condensadores C_S de 4 pF son los condensadores de muestreo, y los condensadores C_H los de *Hold*. Aunque el circuito de entrada es diferencial, el ADC puede manejar señales en modo común fijando una de las entradas a una señal de referencia adecuada. Sin embargo, para un valor óptimo de la SFDR, se debe utilizar un acoplamiento en *ac* mediante un transformador de tensión para acoplar la señal de entrada.

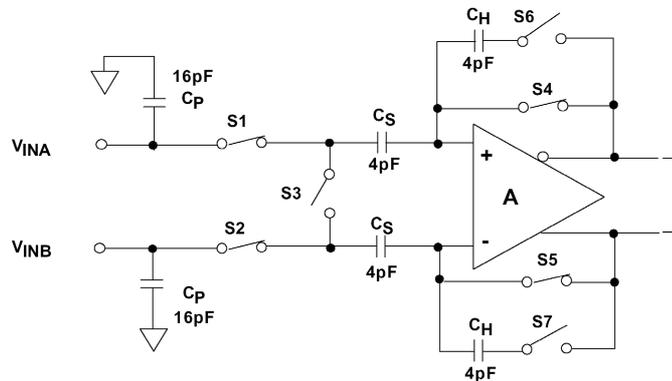


Figura 4.17. Circuito de muestreo y retención de capacidades conmutadas.

En el modo *Track*, la entrada de tensión diferencial se aplica a los condensadores C_S . Cuando el circuito pasa a modo *Hold*, la tensión en los condensadores de muestreo se transfiere a los condensadores de retención C_H por medio del amplificador A que funciona como *buffer*. Los interruptores se controlan con el desfase apropiado respecto al reloj de muestreo. Cuando el T&H vuelve al modo *Track*, la entrada de señal debe cargar o descargar la capacidad de muestreo C_S hasta conseguir el nuevo valor de tensión. Esta acción de carga y descarga de la capacidad de muestreo, promediada en un periodo y para una frecuencia de muestreo determinada, hace que aparezca una

LOS CIRCUITOS CONVERTIDORES A/D PIPELINE

impedancia de entrada con una componente resistiva que resulta beneficiosa, aunque considerando un sólo periodo hay que tener en cuenta las corrientes pulsantes que generan las capacidades.

CAPÍTULO 5. METODOLOGÍAS ALTERNATIVAS DE DISEÑO DE CIRCUITOS ADC PIPELINE Y SIGMA-DELTA

5.1. Introducción

En los capítulos anteriores se ha comprobado la complejidad de los circuitos convertidores de alta resolución y/o alta velocidad de conversión. Es evidente que el aumento en las prestaciones supone un aumento en la complejidad de los circuitos. En este capítulo se presentan dos nuevas estructuras de convertidores A/D. Una para los convertidores multietapa que evita tener que amplificar el residuo en las sucesivas etapas, y otra para los convertidores sigma-delta que facilita el diseño porque se parte de la especificación de la función de transferencia del ruido. Ambas propuestas se estudian mediante simulación y en el caso del convertidor sigma-delta se presentan los resultados prácticos obtenidos a partir de un prototipo.

5.2. Nueva metodología de diseño de convertidores multietapa

5.2.1. Estructura y modo de funcionamiento del ADC multietapa

Se presenta una nueva estructura para un convertidor multietapa que, si se le añade un circuito S&H en la entrada de cada etapa y el correspondiente circuito de alineación del dato de salida, se convierte en un convertidor *pipeline*. Como estos dos circuitos son sencillos y no son el objetivo de este estudio, en este apartado solamente se tratará el convertidor como multietapa.

Con la nueva metodología, un ADC de N bits tiene una estructura multietapa que consiste en una etapa previa, en N-1 etapas iguales y en una etapa final con un circuito comparador (para el LSB). La Figura 5.1 muestra la estructura de una de las N-1 etapas, que está compuesta por un divisor por dos, por un comparador y por un bloque sumador/restador de 1 bit.

En cada etapa, si la tensión de entrada v_i es mayor que $-Q/2$, se le resta la mitad de la tensión de referencia v_{Ri} y el bit de salida b_n valdrá '1', si no, se le suma, y el bit de salida valdrá '0'. La tensión de referencia de salida v_{Ro} es la mitad de la de entrada. Este funcionamiento tiene la ventaja de no amplificar la señal en cada etapa de la cadena directa, como alternativa, en cada etapa se divide la tensión de referencia. A continuación se ejercitan cuatro casos de prueba de un hipotético convertidor de 3 bits, basado en la nueva estructura, con un rango de 5 V y una referencia de 2.5 V.

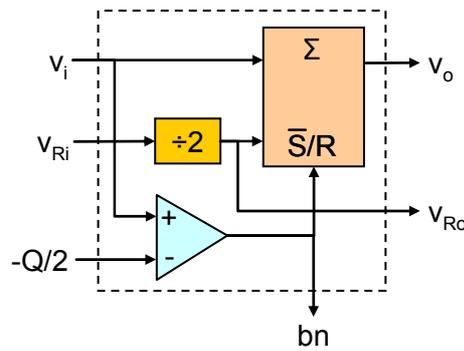


Figura 5.1. Estructura de una etapa del nuevo convertidor.

Caso 1: El dato de salida para una tensión de entrada de 0 V es 000_b

Etapa previa: $v_{i1} = v_i - v_R = 0 - 2.5 = -2.5 \text{ V}$

Primera etapa: $v_{i1} < -\frac{Q}{2} \Rightarrow b_2 = 0$; $v_{o1} = v_{i1} + \frac{v_{Ri1}}{2} = -2.5 + 1.25 = -1.25 \text{ V}$

Segunda etapa: $v_{i2} = v_{o1} < -\frac{Q}{2} \Rightarrow b_1 = 0$; $v_{o2} = v_{i2} + \frac{v_{Ri2}}{2} = -1.25 + 0.625 = -0.625 \text{ V}$

Última etapa comparadora: $v_{o2} < -\frac{Q}{2} \Rightarrow b_0 = 0$

Caso 2: El dato de salida para una tensión de entrada de 0.625 V es 001_b

Etapa previa: $v_{i1} = v_i - v_R = 0.625 - 2.5 = -1.875 \text{ V}$

Primera etapa: $v_{i1} < -\frac{Q}{2} \Rightarrow b_2 = 0$; $v_{o1} = v_{i1} + \frac{v_{Ri1}}{2} = -1.875 + 1.25 = -0.625 \text{ V}$

Segunda etapa: $v_{i2} = v_{o1} < -\frac{Q}{2} \Rightarrow b_1 = 0$; $v_{o2} = v_{i2} + \frac{v_{Ri2}}{2} = -0.625 + 0.625 = 0 \text{ V}$

Última etapa comparadora: $v_{o2} > -\frac{Q}{2} \Rightarrow b_0 = 1$

Caso 3: El dato de salida para una tensión de entrada de 1.5625 V es 011_b

Etapa previa: $v_{i1} = v_i - v_R = 0.625 - 2.5 = -1.875 \text{ V}$

Primera etapa: $v_{i1} < -\frac{Q}{2} \Rightarrow b_2 = 0$; $v_{o1} = v_{i1} + \frac{v_{Ri1}}{2} = -1.875 + 1.25 = -0.625 \text{ V}$

Segunda etapa: $v_{i2} = v_{o1} < -\frac{Q}{2} \Rightarrow b_1 = 0$; $v_{o2} = v_{i2} + \frac{v_{Ri2}}{2} = -0.625 + 0.625 = 0 \text{ V}$

Última etapa comparadora: $v_{o2} > -\frac{Q}{2} \Rightarrow b_0 = 1$

Caso 4: Después de que la entrada llegue a 4.375 (FSR-q), el dato de salida es 111_b

Etapa previa: $v_{i1} = v_i - v_R = 4.375 - 2.5 = +1.875 \text{ V}$

Primera etapa: $v_{i1} > -\frac{Q}{2} \Rightarrow b_2 = 1$; $v_{o1} = v_{i1} + \frac{v_{Ri1}}{2} = 1.875 - 1.25 = 0.625 \text{ V}$

Segunda etapa: $v_{i2} = v_{o1} > -\frac{Q}{2} \Rightarrow b_1 = 1$; $v_{o2} = v_{i2} + \frac{v_{Ri2}}{2} = 0.625 - 0.625 = 0 \text{ V}$

Última etapa comparadora: $v_{o2} > -\frac{Q}{2} \Rightarrow b_0 = 1$

El caso 3 está justo en la mitad de dos códigos, por tanto, podrá ser cualquiera de los dos, el residuo coincide con $-Q/2$, por eso se compara con éste valor, así el error es menor. Si el valor de la entrada supera el código en más de la mitad de Q, entonces, se elige el código siguiente.

Cuando se utiliza el valor de $-Q/2$ como constante de comparación se obtiene la traza de la Figura 5.2.a, en este caso el error es simétrico y varía entre $+Q/2$ y $-Q/2$. Si se compara con 0 V, se obtiene la traza de la Figura 5.2.b, en este caso el error tiene un *offset* de $-Q/2$, aunque puede resultar interesante porque es más sencillo compara con 0 V y el error de *offset* es fácil de corregir.

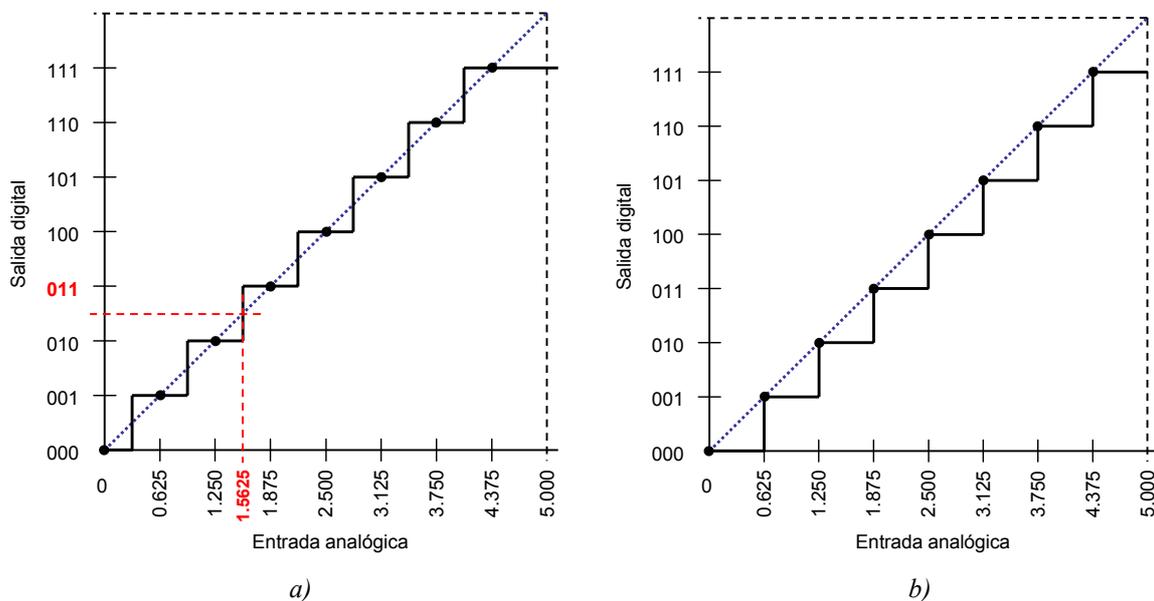


Figura 5.2. Trazas del nuevo convertidor multietapa, a) con comparación con $-Q/2$, b) con comparación con cero.

5.2.2. Diseño y simulación de un ADC de 5 etapas

Para validar la nueva estructura se ha simulado un convertidor multietapa de 5 bits cuyo esquema completo se muestra en la Figura 5.3. El bloque SH, que se corresponde con el circuito

S&H, se implementa con el circuito que se utilizó en el Capítulo 4 y que corresponde con el esquema de la Figura 4.12. Cada etapa, que se basa en la estructura de la Figura 5.1, se implementa mediante el circuito de la Figura 5.4. La entrada VIN es unipolar de 0 a 5 V y la referencia VR es de 2.5 V. La etapa previa consiste en el bloque restador que le resta a la señal de salida del S&H la tensión de referencia.

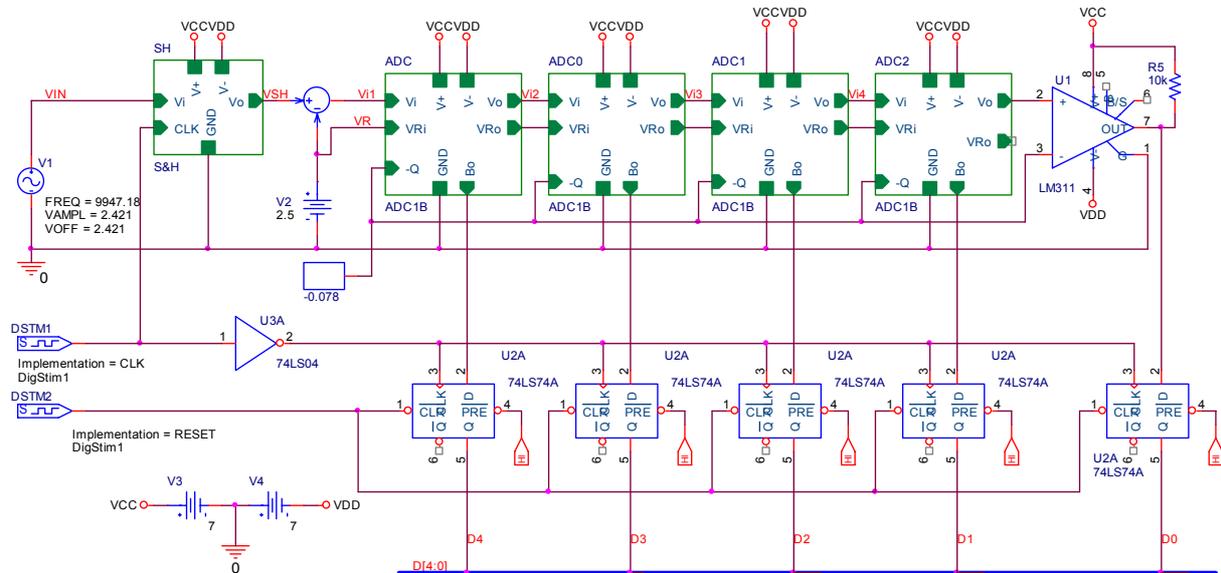


Figura 5.3. Esquema del nuevo convertidor A/D multietapa (PP2).

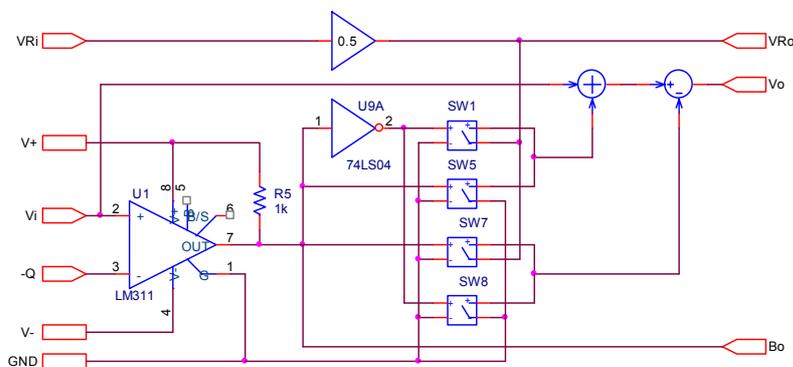


Figura 5.4. Circuito de una etapa del nuevo convertidor A/D multietapa.

Para simular el circuito, que funciona con un reloj de 1 MHz de frecuencia, primero se calcula la cantidad óptima de muestras por periodo que viene dada por la expresión (5.1), luego se sustituye este valor en la ecuación (5.2) y se obtiene la frecuencia de la señal sinusoidal de entrada, que es de 9 947 Hz.

$$M = \pi \cdot 2^N \cong 100.53 \text{ muestras} \quad (5.1)$$

$$f_i = f_s \cdot \frac{J}{M} = 1000\,000 \cdot \frac{1}{100.53} = 9\,947 \text{ Hz} \quad (5.2)$$

En la gráfica de la Figura 5.5, que es el resultado de la simulación con OrCAD PSpice del circuito convertidor multietapa, se muestra una porción de la señal de entrada y de la salida del S&H, así como la señal de reloj y los datos de salida. Los datos de salida correspondientes a un ciclo de la señal, que se han pasado a la hoja de cálculo y que se muestran en la Figura 5.6, se procesan para obtener las características más importantes del convertidor, que se muestran en la Tabla 5.1.

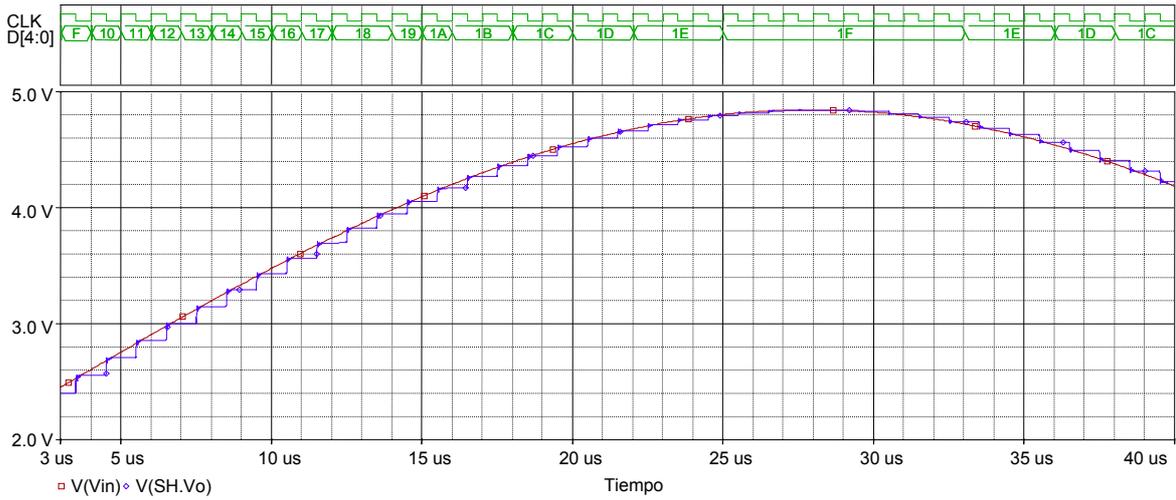


Figura 5.5. Gráficas del Probe obtenidas de la simulación del nuevo convertidor A/D multietapa.

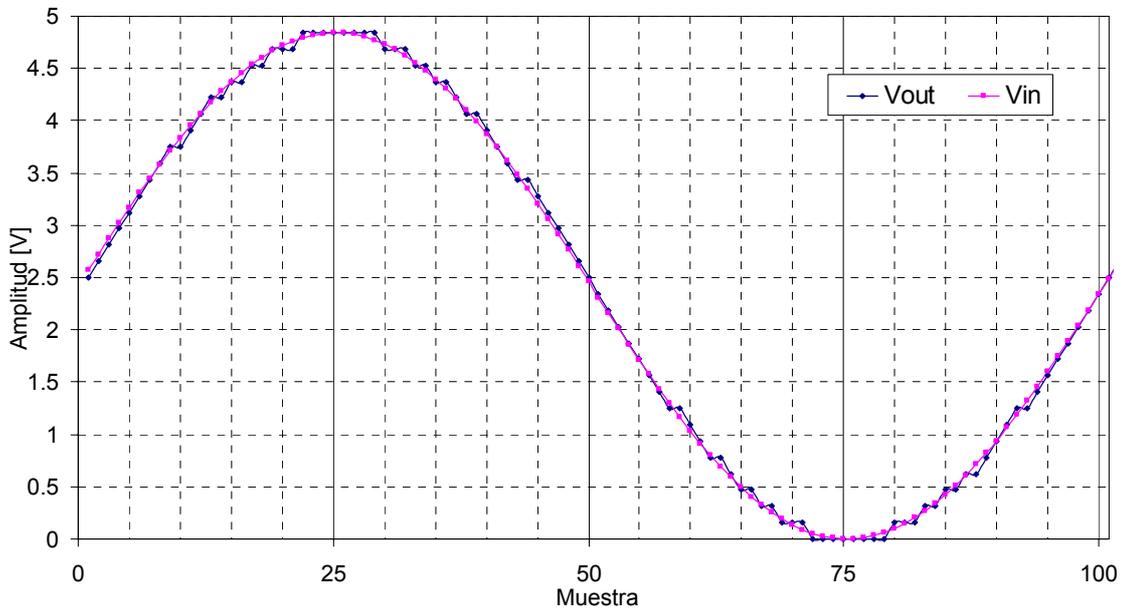


Figura 5.6. Resultado de la conversión de una señal sinusoidal con el nuevo convertidor A/D.

Tabla 5.1. Resultados del test del nuevo convertidor multietapa.

Parámetro		Test
Denominación	Abreviatura	
Frecuencia de muestreo	fs	1 MHz
Frecuencia de la señal de entrada	fi	9 947 Hz
Relación señal ruido más distorsión	SINAD	31.18 dB
Número equivalente de bits	ENOB	4.89 bits

Esta alternativa parece tener, con los resultados de simulación, la ventaja de ser más sencillas las etapas y de introducir menos ruido en la señal puesto que al pasar por las sucesivas etapas no se amplifica el residuo y, por tanto, tampoco el ruido y las no linealidades. Por ejemplo, en un convertidor tradicional de 10 bits y un bit por etapa se amplificaría la señal x2 en cada una, con lo que habría una ganancia total al final de 20. Esto, además de adaptar la señal de cada etapa a la siguiente, también amplifica el ruido y las distorsiones que se añaden a la señal en cada etapa.

5.3. Nueva metodología de diseño de convertidores sigma-delta

5.3.1. Introducción

Como se explicó en el Capítulo 3, el objetivo del modulador sigma-delta es modificar el espectro del ruido de cuantificación de forma que la mayor cantidad de éste quede fuera de la banda de la señal. Aunque con el modulador más simple, que incorpora un integrador, se consiguen buenos resultados, para conseguir altas resoluciones es necesario aumentar considerablemente el factor de sobremuestreo, por ello se han diseñado distintos tipos de moduladores con más etapas, de mayor orden y de más bits. Los resultados que se obtienen son muy satisfactorios y, actualmente, se ha llegado a un estado de desarrollo muy avanzado, en el que se consiguen resoluciones de 24 bits a más de 1 MHz de velocidad de conversión. En este apartado se analizan las estructuras básicas de los moduladores de primero y segundo orden, y se propone una alternativa de diseño en la que se parte de la función de transferencia que se desea para el ruido y, a partir de ella, se diseña el modulador. Se simulan varios ejemplos con Matlab y PSpice, y se presentan los resultados prácticos obtenidos a partir del montaje de un modulador realizado en base a la nueva estructura.

5.3.2. Diseño alternativo del modulador sigma-delta

El modulador básico sigue la estructura de la Figura 5.7 que se corresponde con la función de transferencia total (5.3). Ésta función, que tiene una entrada de señal X y otra del ruido de cuantificación E , se puede dividir en dos funciones como se muestra en la Figura 5.8. La primera, que se muestra en la Figura 5.8.a, se denomina *STF* (*Signal Transfer Function*) y afecta a la señal de entrada. La segunda, que se muestra en la Figura 5.8.b, se denomina *NTF* (*Noise Transfer Function*) y afecta al ruido de cuantificación.

$$Y(z) = STF(z) \cdot X(z) + NTF(z) \cdot E(z) \quad (5.3)$$

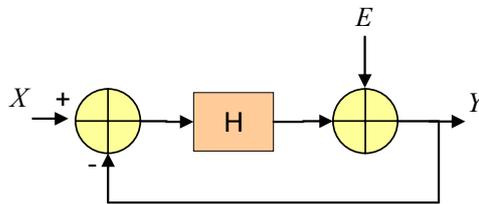


Figura 5.7. Modelo del modulador sigma-delta básico.

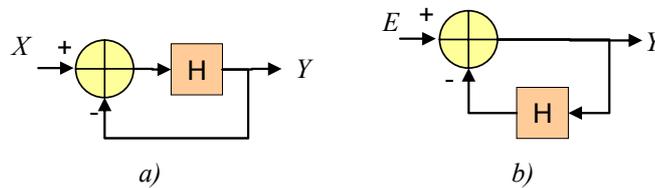


Figura 5.8. Funciones de transferencia: a) de la señal de entrada, b) del ruido de cuantificación.

La función *NTF* es la que tiene mayor interés, pues debe modelar el ruido desplazándolo fuera de la banda de interés para la señal, por tanto, ésta función *NTF* es la que se debe definir. Sin embargo, a partir del esquema del modulador de la Figura 5.7, la función que se diseña es la H , por lo que se debe buscar su relación con la *NTF*. Si se despeja H de las ecuaciones (5.4) y (5.5), se obtiene la ecuación (5.6), que relaciona la función H con la *STF* y ésta última con la *NTF*. Con estas relaciones, que se indican en la Figura 5.9, se puede modificar la estructura básica del modulador de la forma que se indica en la Figura 5.10.

$$NTF = \frac{N}{D} = \frac{1}{1+H} \quad (5.4)$$

$$STF = \frac{H}{1+H} = 1 - NTF = \frac{D-N}{D} \quad (5.5)$$

$$H = \frac{STF}{1 - STF} = \frac{1 - NTF}{NTF} = \frac{D - N}{N} \quad (5.6)$$

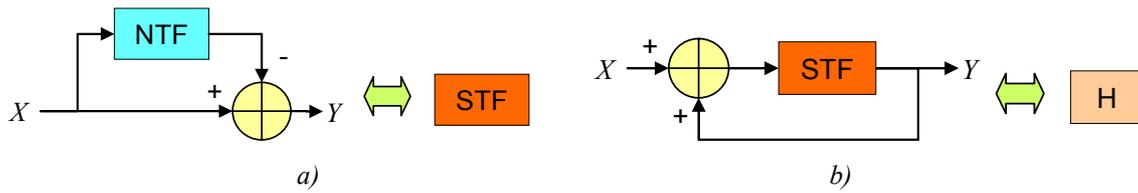


Figura 5.9. Equivalencias entre las funciones de transferencia de la señal y del ruido.

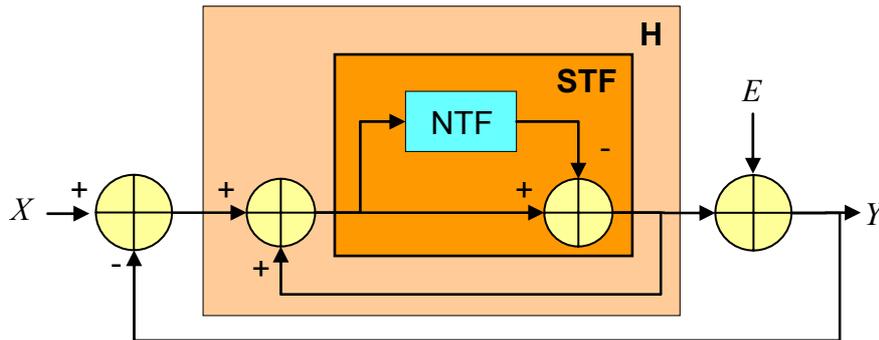


Figura 5.10. Modulador sigma-delta diseñado a partir de la función de transferencia deseada para el ruido.

5.3.3. Análisis del modelo del modulador de primer orden

Como se expuso en el Apartado 3.3, según la ecuación (3.3), si se considera la expresión (5.7) como la función H más simple, la función de transferencia de la señal es $STF = z^{-1}$ y la del ruido es $NTF = 1 - z^{-1} = \frac{z-1}{z}$.

$$H = \frac{1}{z-1} \quad (5.7)$$

Por otro lado, si se transforma s en z utilizando la aproximación rectangular inferior, dada por (5.8), se observa que, si $T=1$, la ecuación (5.8) coincide con la (5.7), quiere decir que para implementar el modulador con un circuito analógico se debe partir de la expresión (5.9), en la que T es el periodo de muestreo.

$$\frac{1}{s} \approx T \cdot \frac{z^{-1}}{1 - z^{-1}} = T \cdot \frac{1}{z-1} \quad (5.8)$$

$$H_1 = \frac{1}{z-1} \approx \frac{1}{T \cdot s} \quad (5.9)$$

Por ejemplo, si $T = 10 \mu s$ (correspondiente a una frecuencia de muestreo de 100 kHz), entonces el integrador, que tendría la función de transferencia (5.10), se implementaría mediante el circuito de la Figura 5.11.

$$v_o = \frac{1}{R \cdot C \cdot s} \cdot v_i \quad (5.10)$$

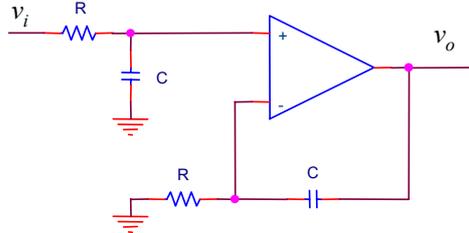


Figura 5.11. Circuito integrador.

Si ahora se transforma s en z utilizando la aproximación trapezoidal, dada por (5.11), la función H viene dada por la expresión (5.12), la función de transferencia de la señal por (5.13) y la de transferencia del ruido por (5.14). Esto quiere decir que la función de transferencia continua de la señal (5.15) es un filtro paso-bajo de primer orden, y que la función de transferencia del ruido

(5.16) es un filtro paso-alto. Ambos filtros tienen una frecuencia de corte de $f_c = \frac{1}{2 \cdot \pi \cdot \frac{T}{2}} = \frac{f_s}{\pi}$.

Por ejemplo, si la frecuencia de muestreo es de 100 kHz, entonces la frecuencia de corte f_c es de 31.83 kHz.

$$\frac{1}{s} \approx \frac{T}{2} \cdot \frac{1+z^{-1}}{1-z^{-1}} \quad (5.11)$$

$$H_2 = \frac{z+1}{z-1} \approx \frac{2}{T \cdot s} \quad (5.12)$$

$$STF_2 = \frac{H_2}{H_2 + 1} = \frac{z+1}{2 \cdot z} \quad (5.13)$$

$$NTF_2 = \frac{1}{H_2 + 1} = \frac{z-1}{2 \cdot z} \quad (5.14)$$

$$STF_2(s) = \frac{2}{2 + T \cdot s} = \frac{1}{\frac{T}{2} \cdot s + 1} \quad (5.15)$$

$$NTF_2(s) = \frac{1}{\frac{T \cdot s}{2} + 1} = \frac{\frac{T}{2} \cdot s}{\frac{T}{2} \cdot s + 1} \quad (5.16)$$

Con estos dos planteamientos se ha partido de que el modelo continuo del modulador tiene un integrador ideal, y que se puede implementar con el integrador real de la Figura 4.10 o con un filtro paso-bajo según la ecuación (5.17).

También se puede deducir que se puede implementar el modulador, de forma discreta, utilizando la aproximación rectangular o la trapezoidal. A continuación se verá qué forma es la más adecuada.

Supongamos que se pretende filtrar el ruido con el filtro paso-alto genérico de la ecuación (5.17) [74]. Por tanto, la función de transferencia del ruido es (5.18), y la de la señal viene dada por (5.19). Finalmente, si se sustituyen las funciones de los dos filtros, en la ecuación general del modulador, se obtiene la expresión (5.20).

$$HPF(z) = b \cdot \frac{1 - z^{-1}}{1 - a \cdot z^{-1}} \quad ; \quad a = \frac{1 - \alpha}{1 + \alpha} \quad ; \quad b = \frac{1 + a}{2} \quad (5.17)$$

$$NTF = HPF(z) = \frac{1 + a}{2} \cdot \frac{1 - z^{-1}}{1 - a \cdot z^{-1}} \quad (5.18)$$

$$\begin{aligned} STF = 1 - NTF &= 1 - \frac{1 + a}{2} \cdot \frac{1 - z^{-1}}{1 - a \cdot z^{-1}} = \frac{2 \cdot (1 - a \cdot z^{-1}) - (1 + a) \cdot (1 - z^{-1})}{2 \cdot (1 - a \cdot z^{-1})} = \\ &= \frac{1 - a + z^{-1} \cdot (-2 \cdot a \cdot + 1 + a)}{2 \cdot (1 - a \cdot z^{-1})} = \frac{1 - a}{2} \cdot \frac{1 + z^{-1}}{1 - a \cdot z^{-1}} \end{aligned} \quad (5.19)$$

$$Y(z) = STF \cdot X(z) + NTF \cdot E(z) = LPF(z) \cdot X(z) + HPF(z) \cdot E(z) =$$

$$Y(z) = \frac{1 - a}{2} \cdot \frac{1 + z^{-1}}{1 - a \cdot z^{-1}} \cdot X(z) + \frac{1 + a}{2} \cdot \frac{1 - z^{-1}}{1 - a \cdot z^{-1}} \cdot E(z) \quad (5.20)$$

Ahora, si en la expresión general (5.20) se anula la variable a , se minimiza el factor que afecta al ruido. Pero, si a es cero, teniendo en cuenta (5.17), α debe ser 1, y como $\alpha = \tan \frac{\omega_c}{2}$,

entonces, la frecuencia de corte es $\omega_c = \frac{\pi}{2}$ y, en la escala de frecuencias digital es $f_c = 0.25$, ya que

$$\omega_c = \frac{2 \cdot \pi \cdot f_c}{f_s} = \frac{\pi}{2} \text{ y, por tanto, } f_c = \frac{f_s}{4}.$$

Si se sustituye el valor de la función *STF* en la expresión de *H* dada por (5.6), se obtiene la ecuación (5.21), que es la función discreta del integrador ideal continuo $H(s)=1/s$, en su aproximación trapezoidal.

$$H = \frac{STF}{1 - STF} = \frac{\frac{1 + z^{-1}}{2}}{1 - \frac{1 + z^{-1}}{2}} = \frac{1 + z^{-1}}{1 - z^{-1}} \quad (5.21)$$

Si ahora se obtiene la expresión general de *H*, por ejemplo, a partir de (5.19), ésta queda como sigue:

$$H = \frac{STF}{1 - STF} = \frac{1 - a}{1 + a} \cdot \frac{1 + z^{-1}}{1 - z^{-1}} = g \cdot \frac{1 + z^{-1}}{1 - z^{-1}} \quad (5.22)$$

donde *g* es un coeficiente que varía entre 1 (para $f_c = 0.25$) e infinito (para $f_c = 0.5$) y que está en serie con la función *H* básica, ésta también es una conclusión importante porque visto de este modo, *g* equivale a la ganancia del convertidor A/D que está en la cadena directa del modulador (véase la Figura 3.8). Esta ganancia, que en [28] se presenta como un factor para mejorar la eficacia del modulador y que se obtiene de forma experimental, se demuestra aquí, que lo que fija es la frecuencia de corte del filtro paso-alto para el ruido y del paso-bajo para la señal. En la Tabla 5.2 se presentan algunos de los valores de *g* para distintas frecuencias de corte.

Tabla 5.2. Valores de la ganancia *g* en función de la frecuencia de corte del modulador sigma-delta.

fc	g
0.06	0.19
0.15	0.51
0.16	0.75
0.25	1.00
0.30	1.38
0.35	1.96
0.45	6.31

Para validar estos resultados, a continuación se compara la estructura básica con la nueva que se ha deducido, mediante Matlab, simulando las funciones de transferencia de la señal (Tabla 5.3) y del ruido (Tabla 5.4) de los modelos de primer orden correspondientes a un valor del coeficiente

$g=1$. Con esta simulación, que se muestra en las gráficas de la Figura 5.12, se comprueba que efectivamente la frecuencia de corte es de 0.25 y que con la nueva estructura se consigue disminuir el ruido en más de 5 dB. Para comprobar mejor estos resultados, en el siguiente apartado se extienden estas simulaciones a los modelos discretos de los moduladores de primero y segundo orden.

Tabla 5.3. Funciones de transferencia de las señales correspondientes a los modelos de los moduladores de primer orden con valor del coeficiente g igual a la unidad.

Modulador	H	STF	Ecuaciones de Matlab
Básico	$H_1 = \frac{1}{1-z^{-1}}$	$STF_1 = \frac{1}{z} = \frac{0+1 \cdot z^{-1}}{1+0 \cdot z^{-1}}$	$STF_1 = \frac{B}{A} = \frac{\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}}$
Con aproximación trapezoidal	$H_2 = \frac{1+z^{-1}}{1-z^{-1}}$	$STF_2 = \frac{z+1}{2 \cdot z} = \frac{0.5 \cdot z^0 + 0.5 \cdot z^{-1}}{1 \cdot z^0 + 0 \cdot z^{-1}}$	$STF_2 = \frac{B}{A} = \frac{\begin{pmatrix} 0.5 & 0.5 \\ 1 & 0 \end{pmatrix}}$

Tabla 5.4. Funciones de transferencia de los ruidos correspondientes a los modelos de los moduladores de primer orden con valor del coeficiente g igual a la unidad.

Modulador	H	NTF	Ecuaciones de Matlab
Básico	$H_1 = \frac{1}{1-z^{-1}}$	$NTF_1 = \frac{z-1}{z} = \frac{1 \cdot z^0 - 1 \cdot z^{-1}}{1 \cdot z^0 + 0 \cdot z^{-1}}$	$NTF_1 = \frac{B}{A} = \frac{\begin{pmatrix} 1 & -1 \\ 1 & 0 \end{pmatrix}}$
Con aproximación trapezoidal	$H_2 = \frac{1+z^{-1}}{1-z^{-1}}$	$NTF_2 = \frac{z-1}{2 \cdot z} = \frac{0.5 \cdot z^0 - 0.5 \cdot z^{-1}}{1 \cdot z^0 + 0 \cdot z^{-1}}$	$NTF_2 = \frac{B}{A} = \frac{\begin{pmatrix} 0.5 & -0.5 \\ 1 & 0 \end{pmatrix}}$

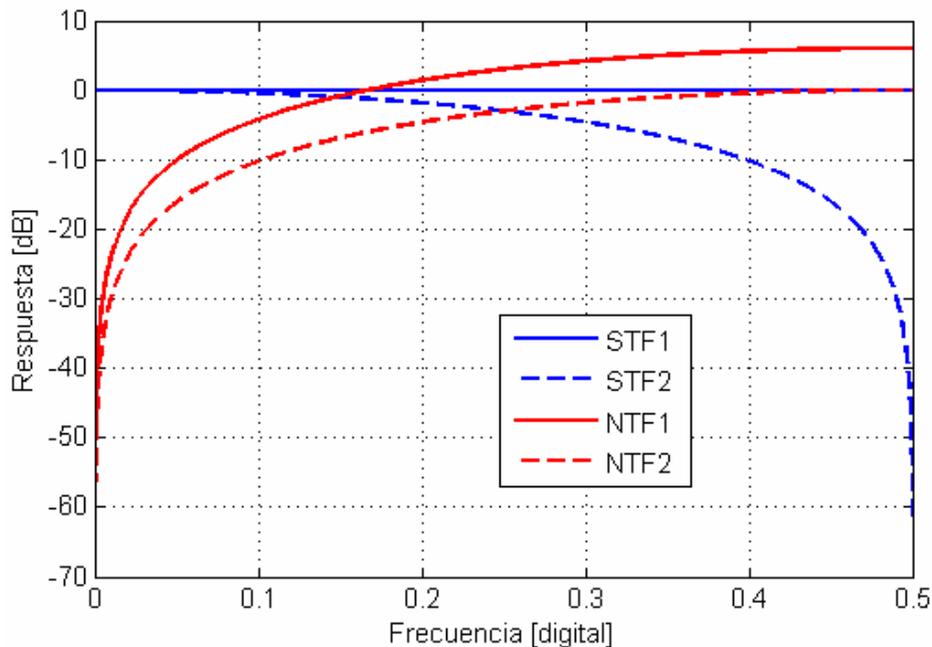


Figura 5.12. Respuesta en frecuencia de los modelos de primer orden cuyas funciones de transferencia de la señal y del ruido corresponden, respectivamente, con la Tabla 5.3 y la Tabla 5.4.

5.3.4. Simulación mediante Matlab de los modelos discretos de los moduladores sigma-delta

Se comienza simulando dos versiones del modulador de primer orden, que se denominan respectivamente SD1 y SD2. El primero tiene la estructura básica de la Figura 5.13 y el segundo la estructura modificada, obtenida del análisis del apartado anterior, de la Figura 5.14. La Tabla 5.5 contiene el código de los moduladores simulados. En la Figura 5.15 se muestra la respuesta en frecuencia del modulador SD1 y en la Figura 5.16 la del SD2. El primer y segundo armónico en el SD2 están 10 dB por debajo de los del SD1. Lo cual indica una clara mejora de la SFDR en el modulador que utiliza la nueva estructura.

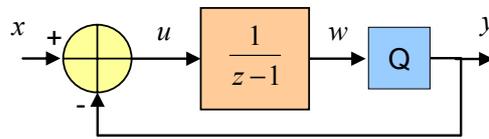


Figura 5.13. Estructura del modulador SD1.

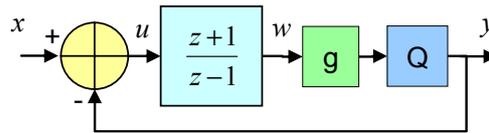


Figura 5.14. Estructura del modulador SD2.

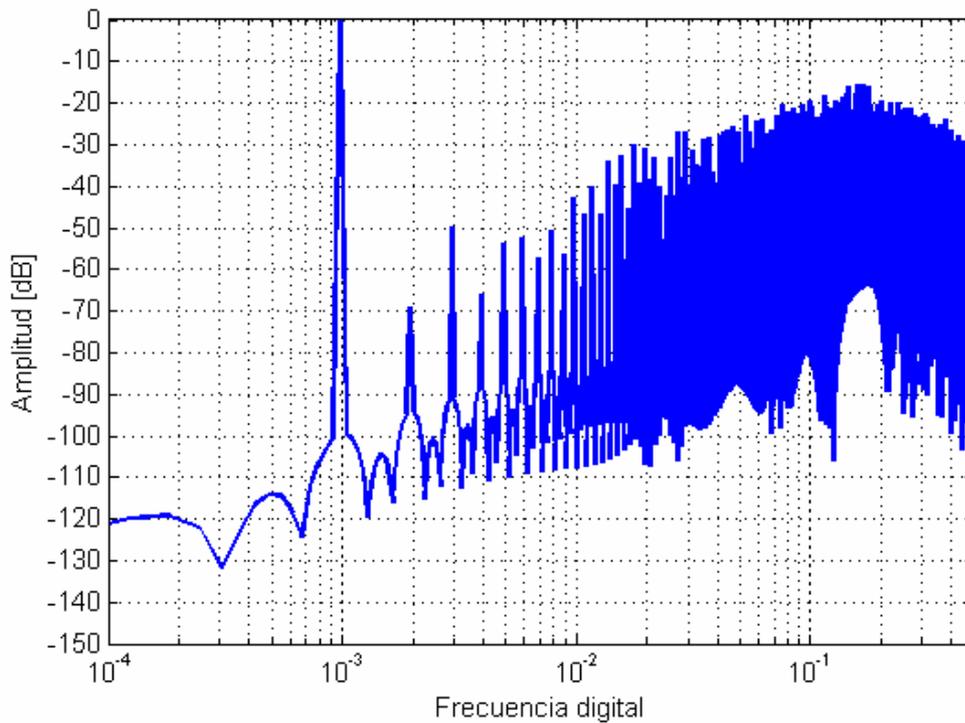


Figura 5.15. Respuesta del modulador de primer orden con la estructura clásica (SD1).

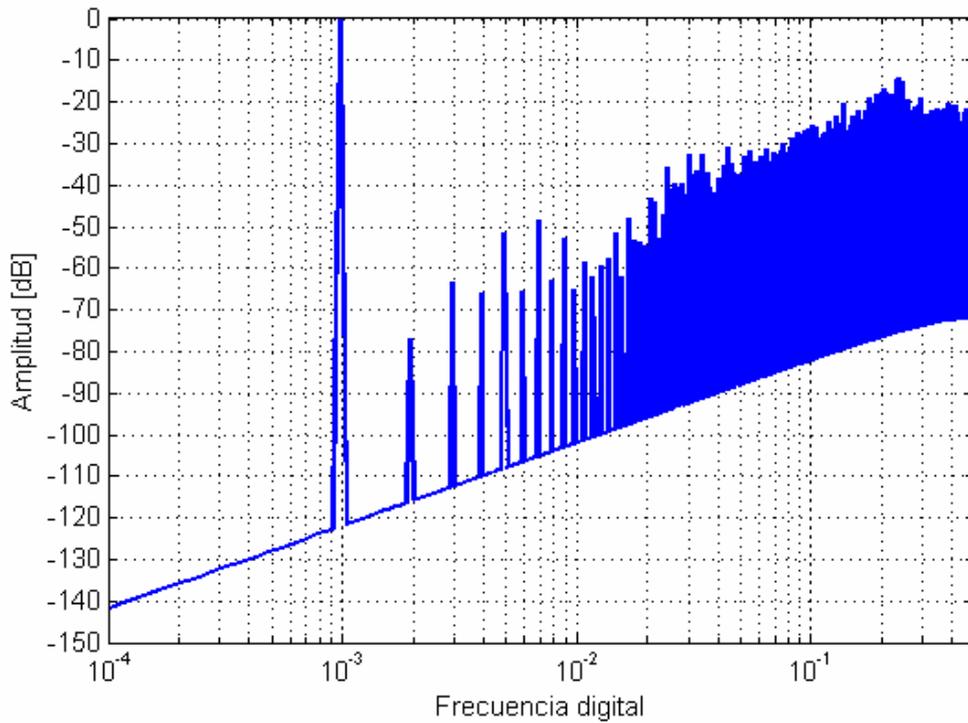


Figura 5.16. Respuesta del modulador de primer orden con la nueva estructura (SD2).

Para simular el modulador clásico de segundo orden se sigue la estructura de la Figura 5.17, cuya ecuación en diferencias de la señal a la entrada del cuantificador está dada por (5.26), que se obtiene sustituyendo (5.23) en (5.24), y ésta, a su vez, en (5.25).

$$w_1(n) = u_1(n-1) + u_1(n) \tag{5.23}$$

$$u_2(n) = w_1(n) - y(n) \tag{5.24}$$

$$w_2(n) = w_2(n-1) + u_2(n-1) \tag{5.25}$$

$$w_2(n) = w_2(n-1) + u_1(n-1) + u_1(n-2) - y(n-1) \tag{5.26}$$

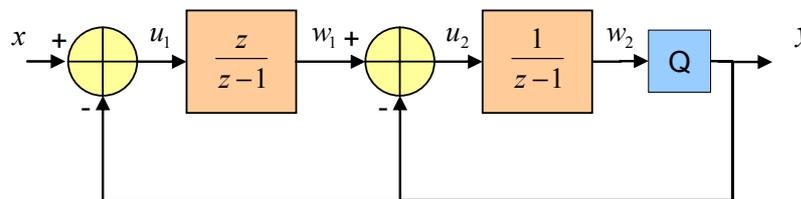


Figura 5.17. Estructura del modulador clásico de segundo orden denominado SD3.

Para implementar el modulador de segundo orden con la nueva estructura, que corresponde con la Figura 5.18, se comienza especificando la función *NTF* de transferencia del ruido, en este caso se ha diseñado como un filtro paso-alto de segundo orden de Butterworth. Luego, se deduce la función de transferencia *H* teniendo en cuenta la expresión (5.6). Finalmente, se aplica el algoritmo que implementa la ecuación en diferencias del modulador que calcula la señal de salida.

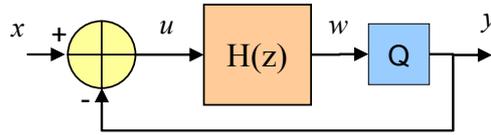


Figura 5.18. Estructura del modulador de segundo orden SD4.

El filtro paso-alto de la función *NTF* se implementa con la ecuación (5.27) [74]. En esta ecuación el coeficiente G_1 se calcula a partir de la ecuación (5.28), que tiene otros dos coeficiente, Ω_0 y θ_1 . El coeficiente Ω_0 se calcula con (5.29) y depende de la frecuencia de corte f_c , que se ha fijado en 0.05. Ahora θ_1 se obtiene a partir del orden del filtro, que en este caso es 2, resultando un valor de $\theta_1 = 3\pi/4$. Después de obtener el coeficiente G_1 , se calculan a_{11} y a_{12} con las ecuaciones (5.30) y (5.31), respectivamente. Una vez se ha determinado la función *NTF* deseada se obtiene H aplicando (5.32) y con ésta se deduce la ecuación en diferencias para programar el algoritmo, como se muestra en la Tabla 5.5.

$$NTF(z) = \frac{N}{D} = \frac{G_1 \cdot (1 - z^{-1})^2}{1 + a_{11} \cdot z^{-1} + a_{12} \cdot z^{-2}} \quad (5.27)$$

$$G_1 = \frac{\Omega_0^2}{1 - 2 \cdot \Omega_0 \cdot \cos \vartheta_1 + \Omega_0^2} \quad (5.28)$$

$$\Omega_0 = \cot\left(\frac{\omega_c}{2}\right) = \cot\left(\frac{\pi \cdot f_c}{f_s}\right) \quad (5.29)$$

$$a_{11} = -\frac{2 \cdot (\Omega_0^2 - 1)}{1 - 2 \cdot \Omega_0 \cdot \cos \vartheta_1 + \Omega_0^2} \quad (5.30)$$

$$a_{12} = \frac{1 + 2 \cdot \Omega_0 \cdot \cos \vartheta_1 + \Omega_0^2}{1 - 2 \cdot \Omega_0 \cdot \cos \vartheta_1 + \Omega_0^2} \quad (5.31)$$

$$H(z) = \frac{NH}{DH} = \frac{D - N}{N} \quad (5.32)$$

En la Figura 5.19 se muestra la respuesta en frecuencia del modulador SD3, realizado con la estructura clásica de segundo orden presentada anteriormente en la Figura 5.17, y en la Figura 5.20 se muestra la respuesta del modulador SD4, realizado con la nueva estructura de la Figura 5.18. Se observa claramente la ventaja del nuevo modulador SD4, pues tiene menos ruido alrededor del lóbulo de la señal de entrada y, sobre todo, porque no tiene los armónicos pares, y los impares son

de menor amplitud. Con estos resultados se mejora la SFDR, la THD y, por tanto, se mejora la S/N+D.

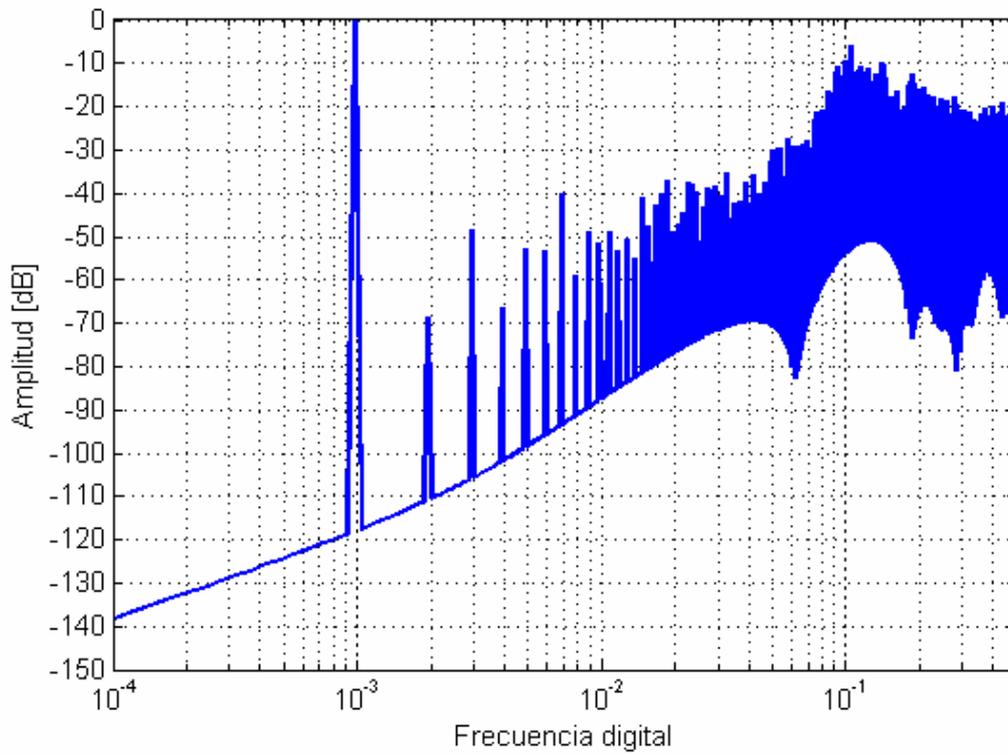


Figura 5.19. Respuesta del modulador de segundo orden con la estructura clásica (SD3).

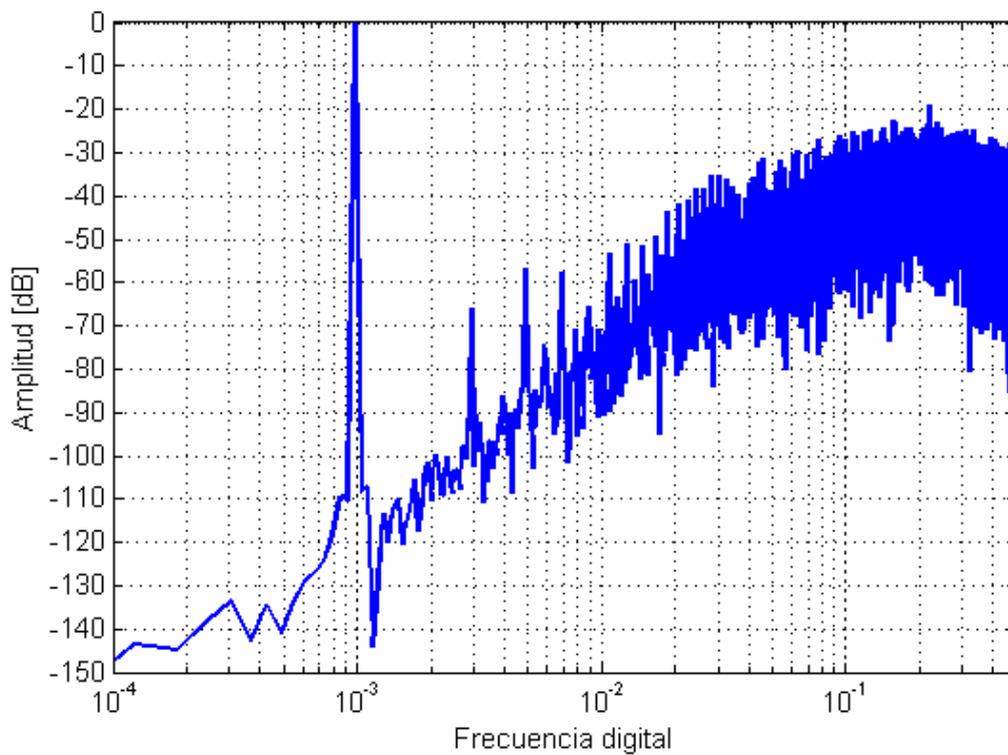


Figura 5.20. Respuesta del modulador de segundo orden con la nueva estructura (SD4).

Tabla 5.5. Código de Matlab que simula los moduladores de primero y de segundo orden.

<pre> %Modulador primer orden con estructuras 1 y 2 M=2^10; %puntos por ciclo del seno N=2^4; %número de ciclos del seno A0=0.95; %amplitud del seno d=M*N; k=zeros(1,d); %señales del modulador x=zeros(1,d); %Vector de entrada y=zeros(1,d); %Salida w=zeros(1,d); %Salida de H u=zeros(1,d); %entrada de H %Ganancia del cuantificador g=1; %Generación del vector de entrada for i=1 : d; x(i)=A0*sin((2*pi*(i-1))/M); k(i)=i; end; %% %Estructura clasica del modulador de primer orden for i=2 : d-1; u(i)=x(i)-y(i); w(i)=w(i-1)+u(i-1); if w(i)*g > 0 y(i+1)=1; else y(i+1)=-1; end; end; %respuesta del modulador 1 SD1=respuestalog(y); m=max(SD1); for i=1 : length(SD1); SD1(i)=SD1(i)-m; end; f=zeros(1,length(SD1)); %Vector de frecuencias for i=1 : length(SD1); f(i)=0.5*((i-1)/length(SD1)); end; %% %Nueva estructura del modulador de primer orden for i=2 : d-1; u(i)=x(i)-y(i); w(i)=w(i-1)+u(i)+u(i-1); if w(i)*g > 0 y(i+1)=1; else y(i+1)=-1; end; end; %respuesta del modulador 2 SD2=respuestalog(y); m=max(SD2); for i=1 : length(SD2); SD2(i)=SD2(i)-m; end; </pre>	<pre> %% %Estructura clasica del modulador de segundo orden for i=3 : d-1; u(i)=x(i)-y(i); w(i)=w(i-1)+u(i-2)+u(i-1)-y(i-1); if w(i)*g > 0 y(i+1)=1; else y(i+1)=-1; end; end; %respuesta del modulador 1 de segundo orden SD3=respuestalog(y); m=max(SD3); for i=1 : length(SD3); SD3(i)=SD3(i)-m; end; %% %Nueva estructura del modulador de segundo orden %Filtro 2do orden paso alto para la NTF fc=0.05 wc=2*pi*fc O=cot(wc/2) tita=3*pi/4 %Calculo del numerador de la NTF G=(O^2) / (1 -2*O*cos(tita)+O^2) N=[1 -2 1] N=N*G %Calculo del denominador de la NTF a1=-2*(O^2-1) / (1 -2*O*cos(tita)+O^2) a2=(1 +2*O*cos(tita)+O^2) / (1 -2*O*cos(tita)+O^2) D=[1 a1 a2] %Calculo de H a partir de la NTF NH=D-N; DH=N; %código del modulador for i=3 : d-1; u(i)=x(i)-y(i); w(i)=(-DH(2)*w(i-1)-DH(3)*w(i-2)+NH(1)*u(i) +NH(2)*u(i-1)+NH(3)*u(i-2))/DH(1); if w(i)*g > 0 y(i+1)=1; else y(i+1)=-1; end; end; %respuesta del modulador 2 de segundo orden SD4=respuestalog(y); m=max(SD4); for i=1 : length(SD4); SD4(i)=SD4(i)-m; end; semilogx(f,SD1,f,SD2,f,SD3,f,SD4); grid on; axis([0.0001 0.5 -150 0]); </pre>
--	---

5.3.5. Simulación mediante OrCAD PSpice de los modelos continuos de los moduladores sigma-delta

Después de simular las distintas estructuras que se analizaron en los apartados anteriores mediante algoritmos de Matlab, el siguiente paso ha sido realizar otra serie de simulaciones, más realistas, de tres circuitos que implementen los circuitos de los moduladores con las nuevas estructuras, por lo que se ha utilizado el programa OrCAD PSpice. En la Figura 5.21 se muestra el esquema electrónico del modulador sigma-delta de primer orden, que se ha implementado con el circuito integrador diferencial con una frecuencia de corte de 25 kHz, que corresponde con 1/8 de la frecuencia de Nyquist, ya que la de muestreo es de 400 kHz. La FFT de la salida de la simulación, que se muestra en la Figura 5.22, indica un nivel de 15 mV para el ruido y de 3.97 V para la señal, lo cual, tratándose de un modulador de primer orden, es un buen margen (48 dB).

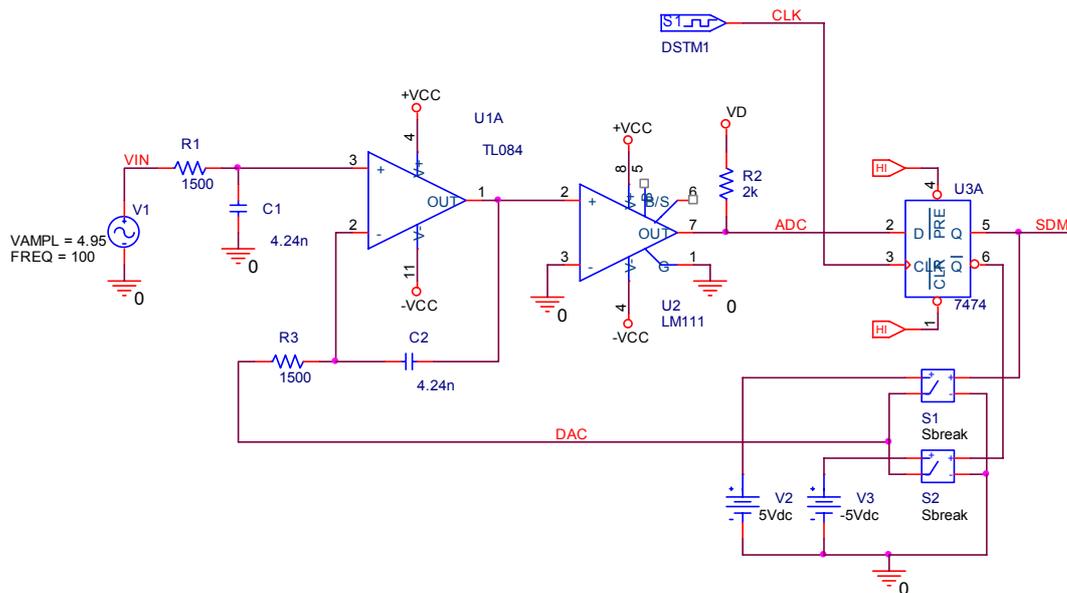


Figura 5.21. Circuito del modulador de primer orden.

En la Figura 5.23 se muestra el modulador de segundo orden y en la Figura 5.24 el de tercer orden. En ambos se identifica con líneas de trazos qué parte del circuito corresponde con las funciones de transferencia del ruido (NTF) y de la señal (STF). En la Figura 5.25 se muestra el resultado de la FFT de la salida del modulador de segundo orden y en la Figura 5.26 la del de tercer orden. Entre el modulador de primer y segundo orden se aprecia una clara mejora, ya que se pasa de un nivel de ruido de 15 mV a menos de 10 mV, en cambio, en el de tercer orden ya no se aprecia

tanto. Esto es debido a que con niveles de 3 mV los errores de los circuitos ya no permiten mejorar las prestaciones del modulador. También se aprecia que en los moduladores de segundo y tercer orden aparece una distorsión que se muestra con un armónico a 300 Hz.

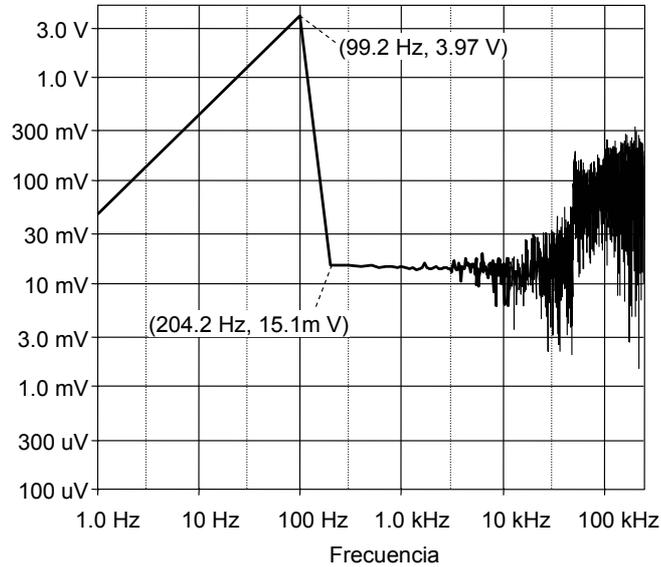


Figura 5.22. FFT de la salida del modulador sigma-delta de primer orden simulado con PSpice.

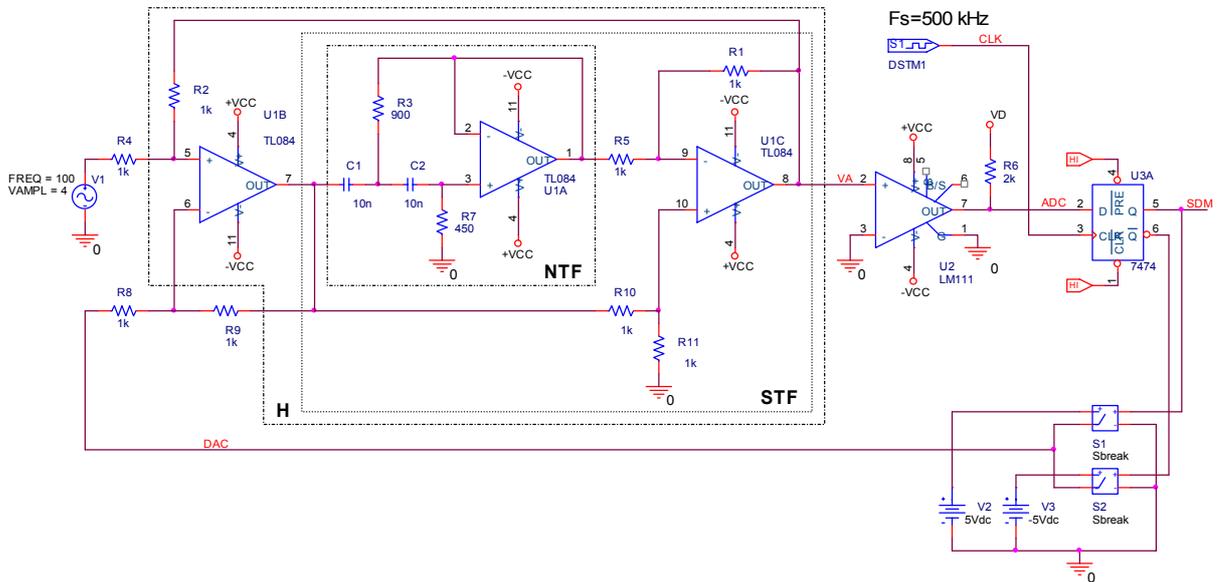


Figura 5.23. Circuito del modulador sigma-delta de segundo orden.

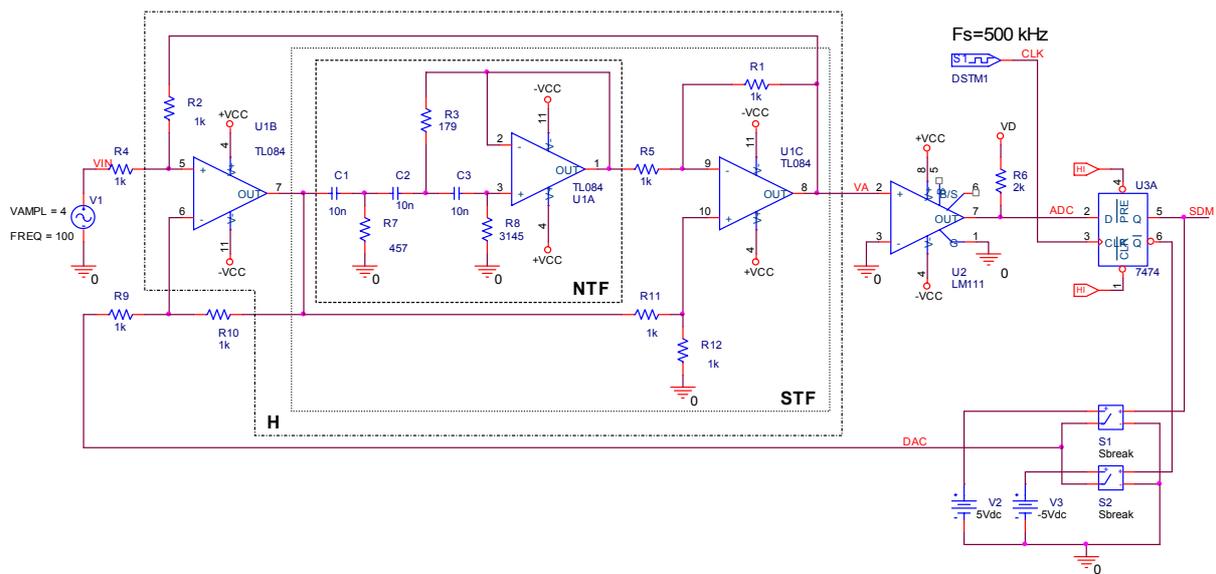


Figura 5.24. Circuito del modulador sigma-delta de tercer orden.

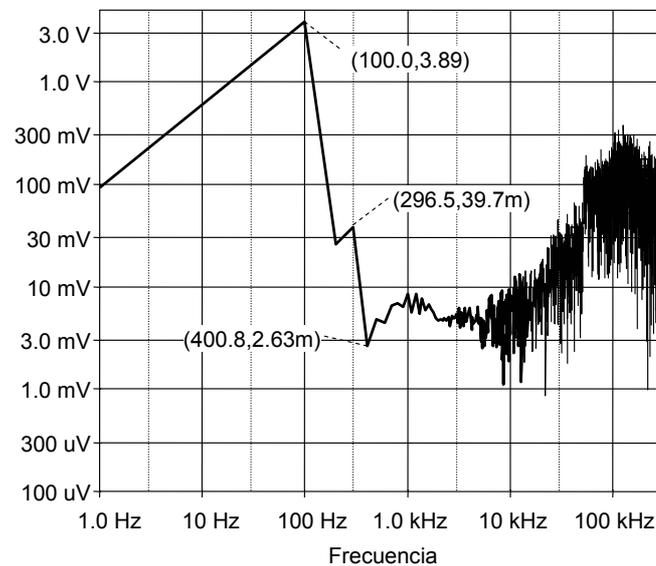


Figura 5.25. Respuesta en frecuencia del modulador de segundo orden con una señal de entrada de 100 Hz.

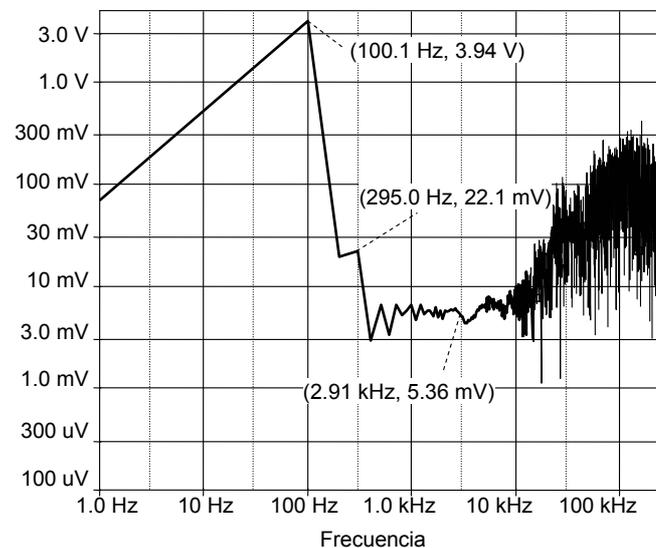


Figura 5.26. Respuesta en frecuencia del modulador de tercer orden con una señal de entrada de 100 Hz.

5.3.6. Evaluación del principio de funcionamiento de los nuevos modelos de moduladores

En el apartado anterior se evaluaron los moduladores mediante un análisis transitorio y la posterior realización de la FFT de la señal de salida. Para probar que efectivamente las estructuras que se simularon siguen el modelo de la Figura 5.10, en este apartado se realiza un barrido en frecuencia de las funciones de transferencia del ruido de la señal. Por ejemplo, para el modulador de segundo orden de la Figura 5.23, la función de transferencia del ruido es el filtro paso-alto de segundo orden de Butterword definido por la ecuación (5.33), y la función de transferencia de la señal, según la ecuación (5.5), se corresponde con la ecuación (5.34). Ambas ecuaciones se han implementado con el circuito de la Figura 5.27 y se han simulado haciendo un barrido en frecuencia con la frecuencia de corte como parámetro que demuestra, según la Figura 5.28, que hay una banda en la que la respuesta es plana para la señal y, al mismo tiempo, proporciona una gran atenuación para el ruido. De forma similar, en la Figura 5.29 se muestra el circuito que implementa la función STF del modulador de cuarto orden y en la Figura 5.30 las respuestas en frecuencia de sus funciones NTF y STF.

$$NTF_{2P}(s) = \frac{N_{2P}(s)}{D_{2P}(s)} = \frac{s^2}{s^2 + 1.414 \cdot \omega_c \cdot s + \omega_c^2} = \frac{s^2}{s^2 + 1.414 \cdot 2 \cdot \pi \cdot f_c \cdot s + 4 \cdot \pi^2 \cdot f_c^2} \quad (5.33)$$

$$\begin{aligned} STF_{2P}(s) &= \frac{D_{2P}(s) - N_{2P}(s)}{D_{2P}(s)} = \frac{1.414 \cdot \omega_c \cdot s + \omega_c^2}{s^2 + 1.414 \cdot \omega_c \cdot s + \omega_c^2} = \\ &= \frac{1.414 \cdot 2 \cdot \pi \cdot f_c \cdot s + 4 \cdot \pi^2 \cdot f_c^2}{s^2 + 1.414 \cdot 2 \cdot \pi \cdot f_c \cdot s + 4 \cdot \pi^2 \cdot f_c^2} \end{aligned} \quad (5.34)$$

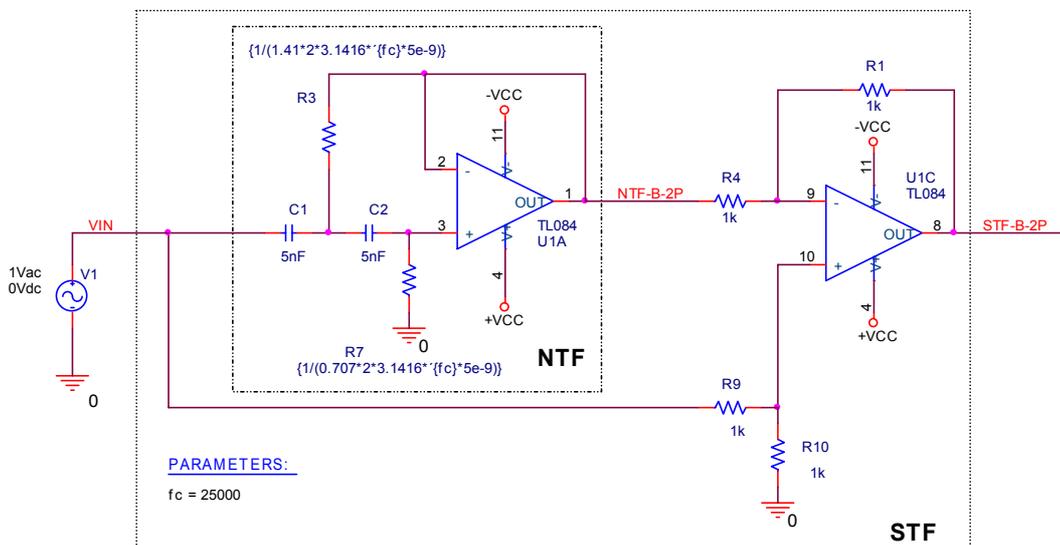


Figura 5.27. Circuito que implementa las funciones NTF y STF del modulador de segundo orden.

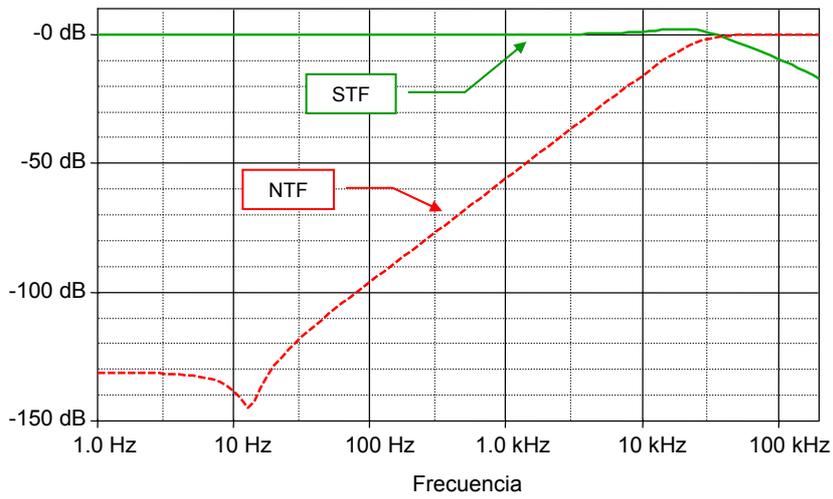


Figura 5.28. Respuesta en frecuencia de las funciones NTF y STF del modulador de segundo orden.

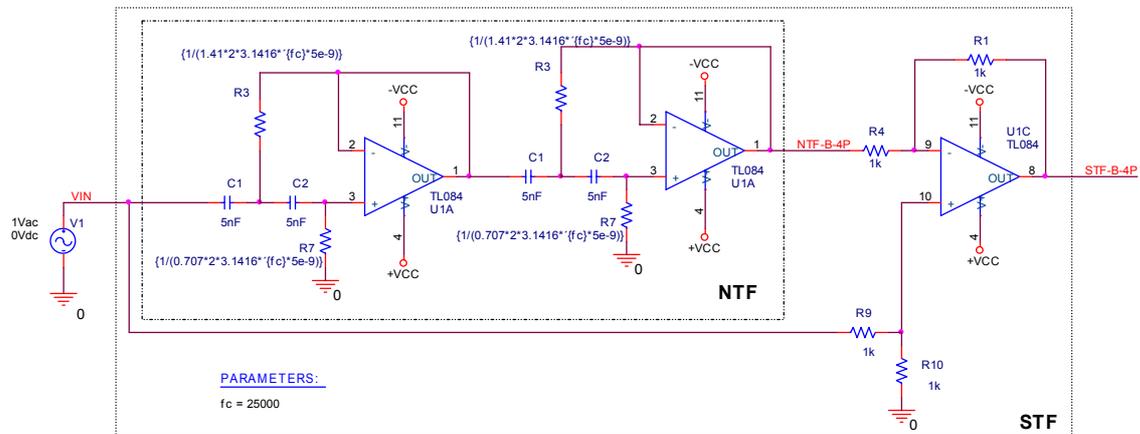


Figura 5.29. Circuito que implementa las funciones NTF y STF del modulador de cuarto orden.

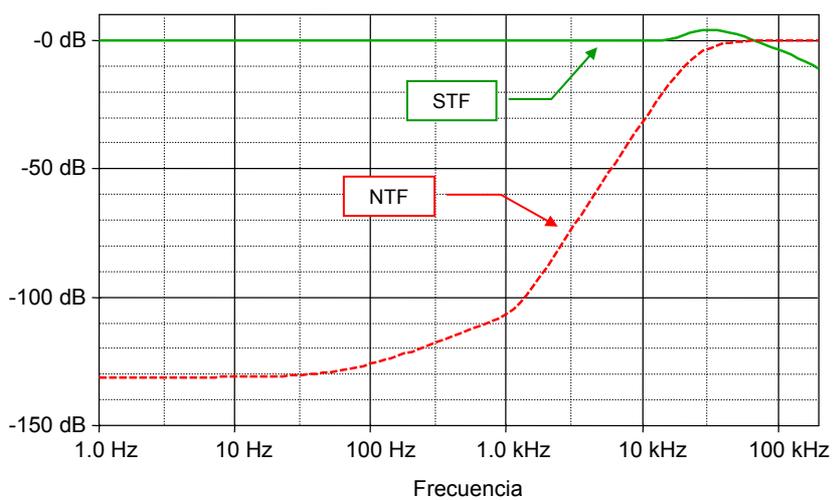


Figura 5.30. Respuesta en frecuencia de las funciones NTF y STF del modulador de cuarto orden.

5.3.7. Construcción de un prototipo y prueba del modulador sigma-delta de segundo orden

Para evaluar el funcionamiento real de la metodología propuesta que se simuló en los apartados anteriores, se ha construido un prototipo del modulador de segundo orden (Figura 5.31), que corresponde con el esquema de la Figura 5.23 salvo con los cambios, en los dispositivos, que se mencionan a continuación:

1. Se alimenta con una tensión simétrica de 5 V.
2. Incorpora unas referencias de tensión simétricas de ± 2.5 V, que se basan en el circuito LM4120.
3. Se utilizan los amplificadores operacionales de alta velocidad AD8044 (150 MHz de ancho de banda y 170 V/ μ s de *slew rate*).
4. Se utiliza el comparador de alta velocidad LT1016, cuya principal característica es su bajo tiempo de respuesta, que es de 10 ns.
5. Los interruptores ideales S1 y S2 se han sustituido por el multiplexor analógico de alta velocidad AD8184 (700 MHz de ancho de banda y 750 V/ μ s de *slew rate*).
6. La señal de reloj y el biestable D se han implementado en una FPGA Cyclone [117] (Figura 5.40), con la ventaja de, además de la gran velocidad de trabajo, la configurabilidad de la frecuencia del reloj y la disponibilidad de memoria de adquisición y una conexión con el computador para enviar los datos de salida del modulador.

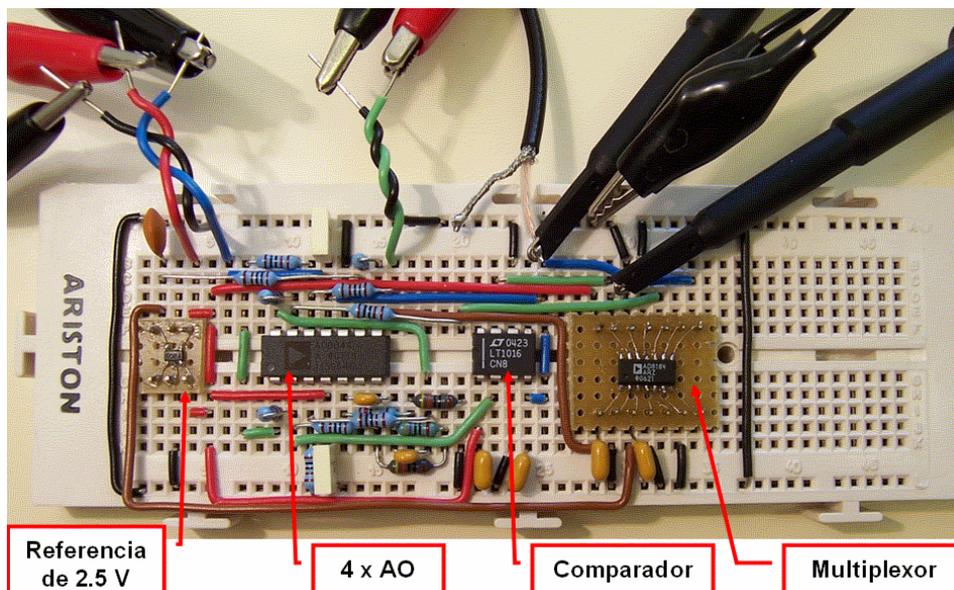


Figura 5.31. Fotografía del prototipo del modulador sigma-delta.

Con estos cambios se ha conseguido aumentar notablemente la velocidad de trabajo del modulador, del orden de 10 veces, y se han podido hacer pruebas del orden de los Megahercios. La limitación en cuanto a velocidad que se ha encontrado se ha debido a las características propias del prototipo, es decir: los acoplamientos entre señales de alta frecuencia, la falta de un plano de masa, la falta de aislamiento entre las tensiones de la parte digital y analógica, las distancias de los conductores, etc. Pese a estas limitaciones, los resultados obtenidos han sido mejores de lo esperado, teniendo en cuenta que se ha trabajado a una frecuencia de muestreo de 6 MHz, con lo que la parte analógica ha estado funcionando en intervalos de 166 ns.

Siguiendo la metodología de diseño presentada, se ha comenzado por montar y probar el filtro paso alto que se encarga de implementar la función *NTF*. Se ha construido un filtro de 2° orden basado en la estructura Sallen-Key utilizada en la simulación (Figura 5.23) pero modificando los valores de las redes *RC* para aumentar la frecuencia de corte. Los nuevos valores son: $R3= 900 \Omega$, $R7= 452 \Omega$ y $C1=C2= 1 \text{ nF}$. Con estos valores se ha conseguido una frecuencia de corte aproximada de 700 kHz. En la Figura 3.18 se presenta la respuesta del filtro con excitación sinusoidal a la frecuencia de corte (el canal 1 es la señal de entrada y el canal 2 la de salida), se deduce que el filtro proporciona un adelanto de fase de unos 45° y una atenuación de 3 dB.

Además de la caracterización del filtro a la frecuencia de corte, se han tomado medidas en un rango de frecuencias desde los 30 Hz hasta los 30 MHz.

Después de probar el funcionamiento del filtro paso alto, se probó el correspondiente filtro paso bajo, que implementa la *STF*, añadiendo la etapa restadora basada en el circuito UC1C (Figura 5.23), este filtro también se caracterizó para el mismo rango de frecuencias que el paso alto, la respuesta a la frecuencia de corte se muestra en la Figura 5.33. Ahora se aprecia cómo lo que introduce el filtro es un retraso de fase.

Finalmente, se construyó una gráfica con las respuestas en frecuencia de la magnitud (Figura 5.34) y la fase (Figura 5.35) de ambos filtros. Se comprueba que la respuesta de ambos filtros coinciden a unos 700 kHz y que, lo más importante, por debajo de los 100 kHz la respuesta del paso alto atenuará el ruido en más de 20 dB y, en cambio, la respuesta del paso-bajo distorsionará mínimamente la señal.

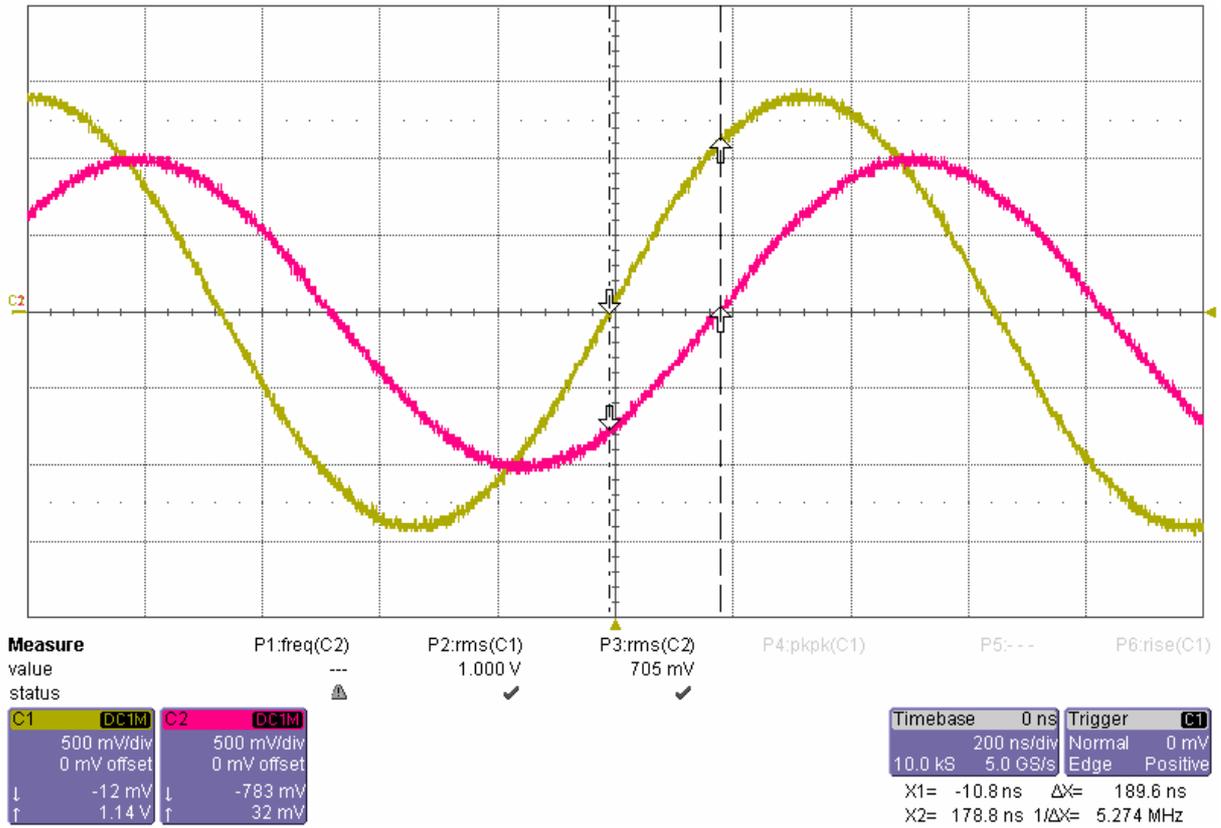


Figura 5.32. Oscilograma del test de la frecuencia de corte del circuito que implementa la función NTF.

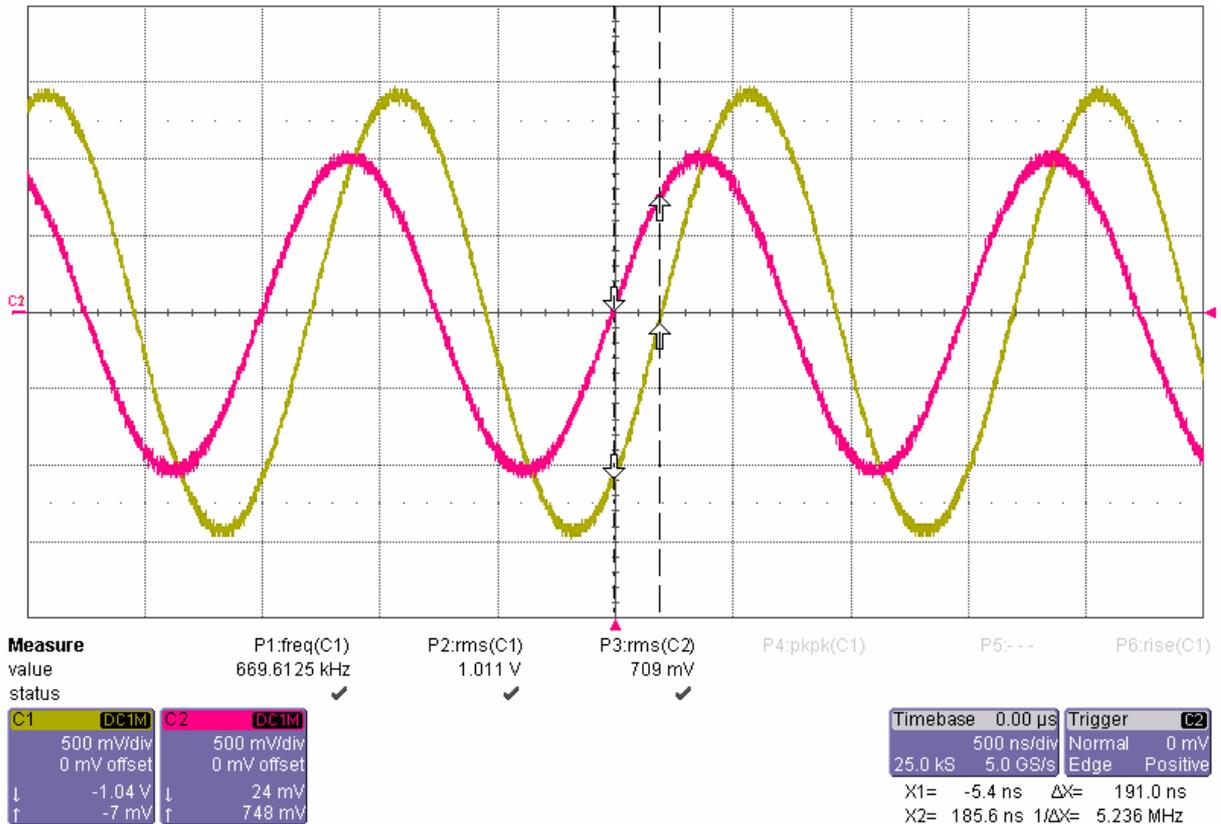


Figura 5.33. Oscilograma del test de la frecuencia de corte del circuito que implementa la función STF.

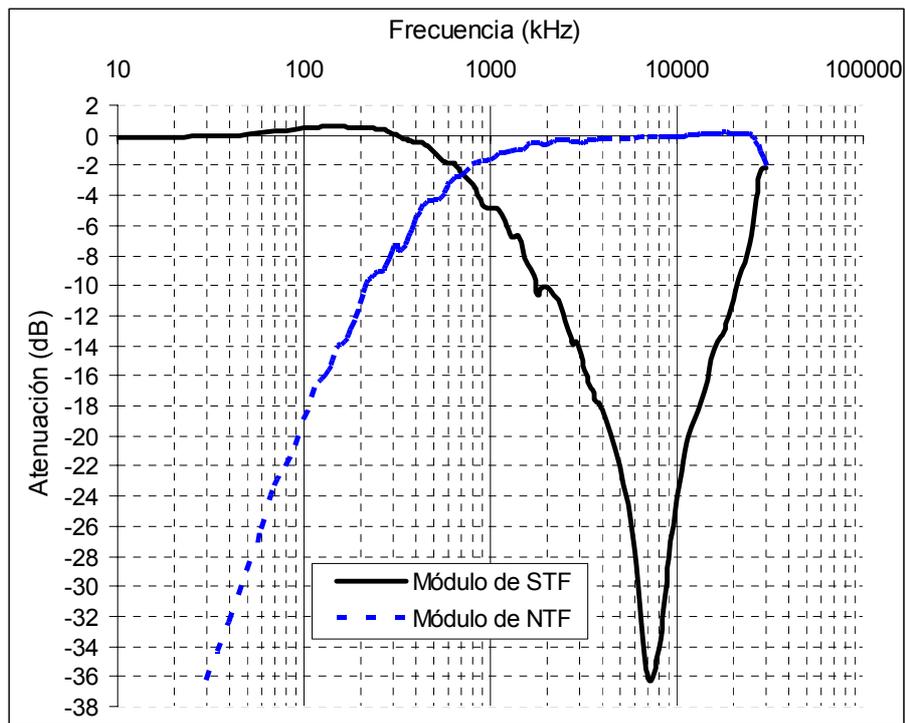


Figura 5.34. Respuesta en módulo de los circuitos de la NTF y de la STF.

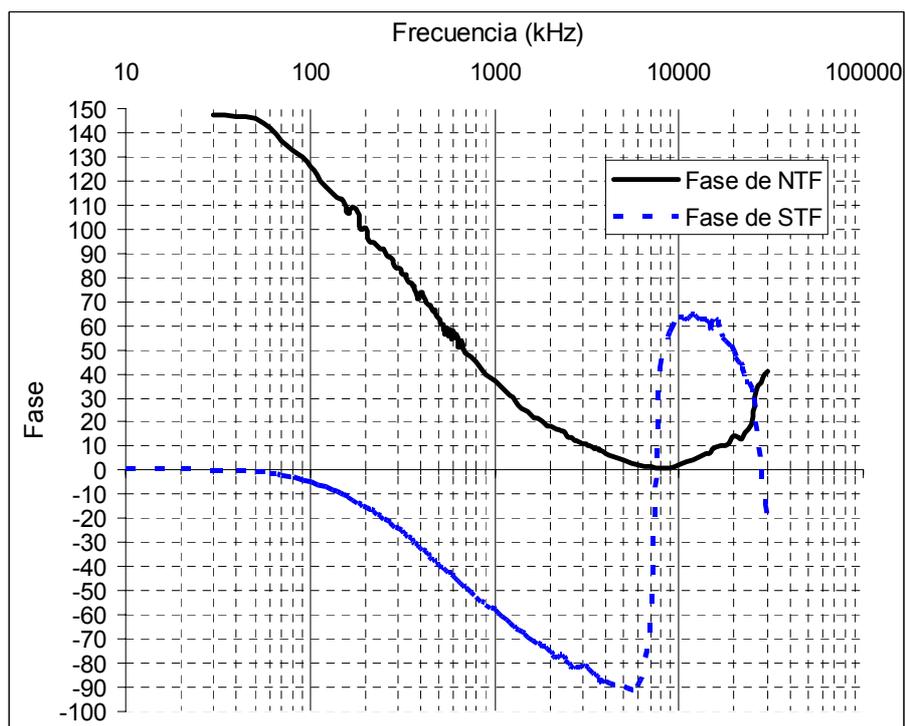


Figura 5.35. Respuesta en fase de los circuitos de la NTF y de la STF.

Después de evaluar el correcto funcionamiento de las etapas de filtrado, se procedió a probar el comparador. Se probó comparando una señal sinusoidal de 7 Vpp con la señal de referencia (0 V) y se constató su rapidez que, como se aprecia en la Figura 5.36, supone un tiempo de respuesta

menor de una sub-división horizontal del osciloscopio, esto es, menos de 20 ns, lo cual es suficiente para esta aplicación.

Una vez probado el comparador, se hizo lo mismo con el multiplexor, el cual mostró un retardo de unos 20 ns (Figura 5.37) desde $-V_{ref}$ hasta $+V_{ref}$. Como se indica en la medida P2 del oscilograma de la Figura 5.37, la amplitud de la salida del multiplexor, con una entrada de tensión bipolar de referencia de 2.5 V, es de 4.489 Vpp. Lo cual es menor que los 5 V ideales, esto es debido a la impedancia de salida del multiplexor. Por tanto, se contará con que las tensiones de referencia reales del modulador son de 2.24 V, en vez de los 2.5 V.

Cuando ya se probaron todos las etapas del modulador en bucle abierto, se montó el modulador completo cerrando el lazo a través del biestable D dentro de la FPGA, y se comprobó cualitativamente su buen funcionamiento, tal y como se muestra en la Figura 5.38, con una señal de entrada de 10 kHz y una frecuencia de reloj de 2.4 MHz.

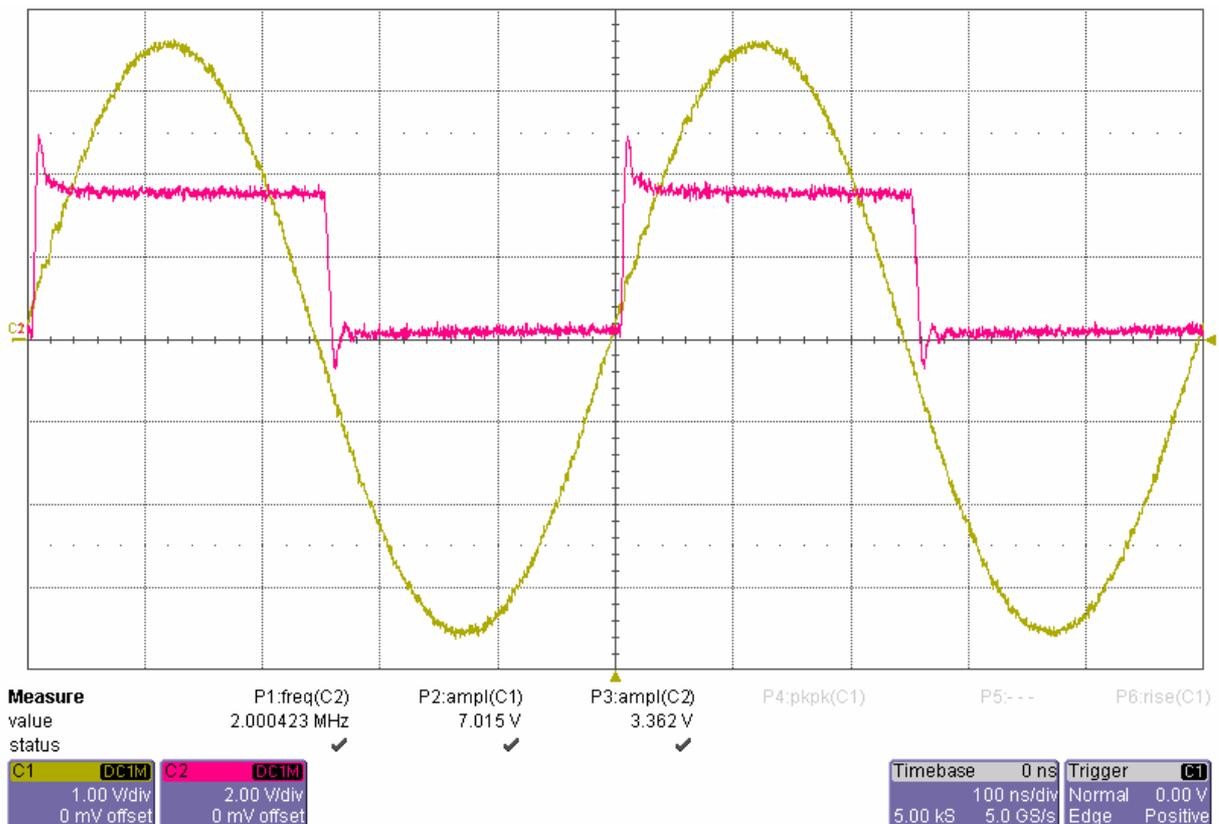


Figura 5.36. Respuesta transitoria del circuito comparador LT1016 para una entrada de 2 MHz.

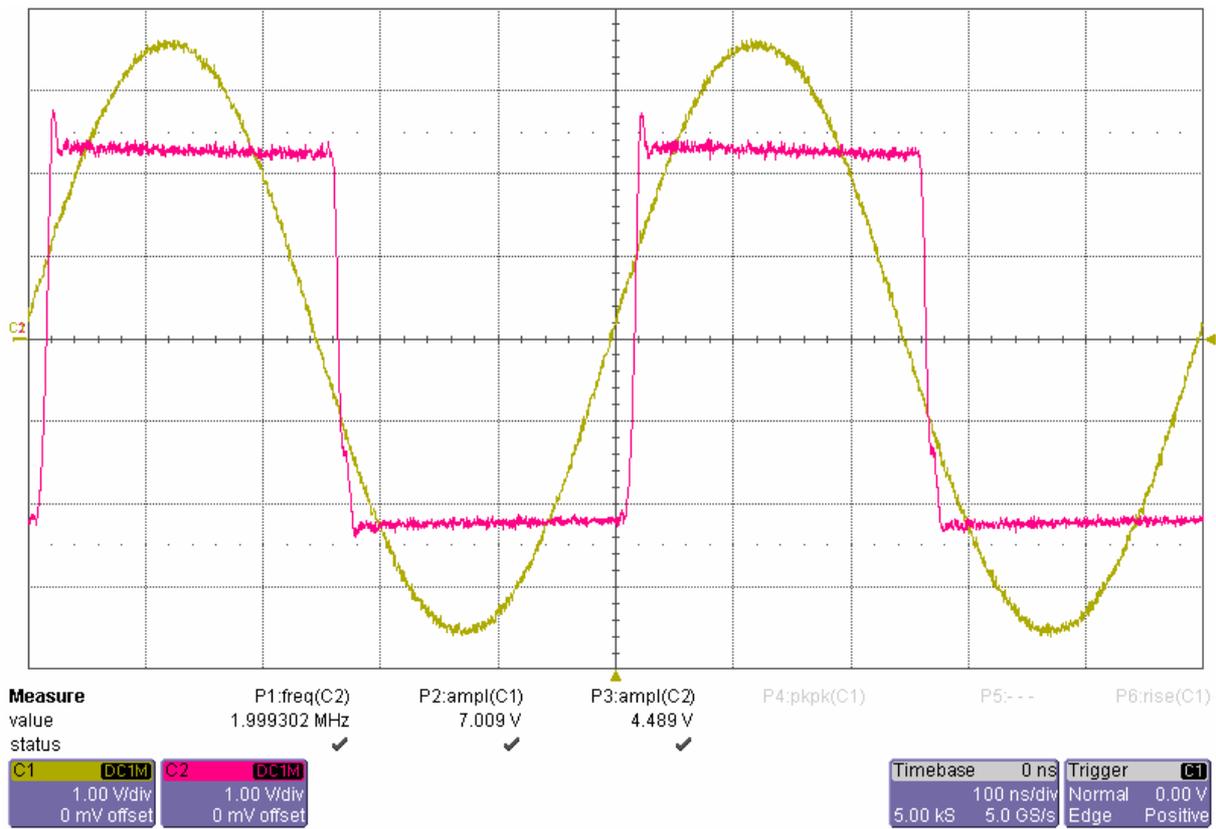


Figura 5.37. Respuesta del circuito multiplexor AD8044 con una carga de 2 kΩ.

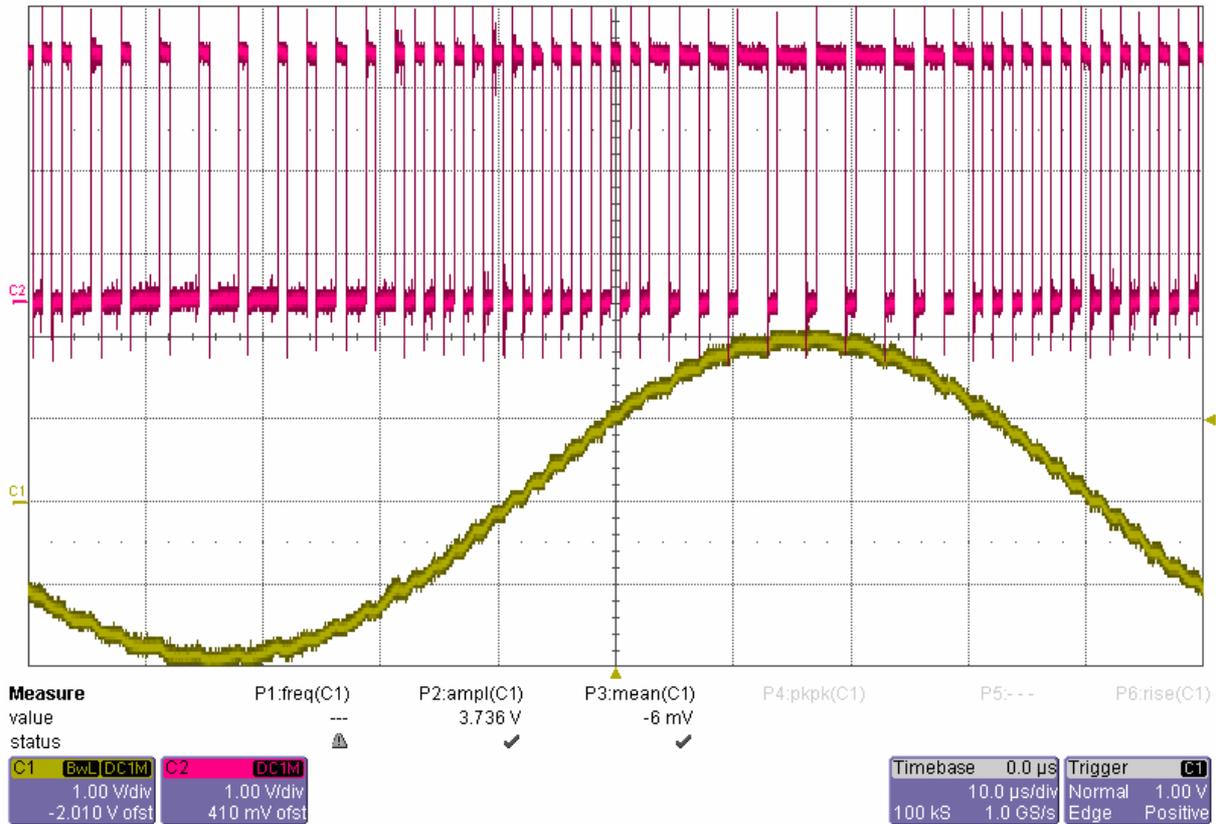


Figura 5.38. Sinusoidal de entrada de 10 kHz y salida del modulador.

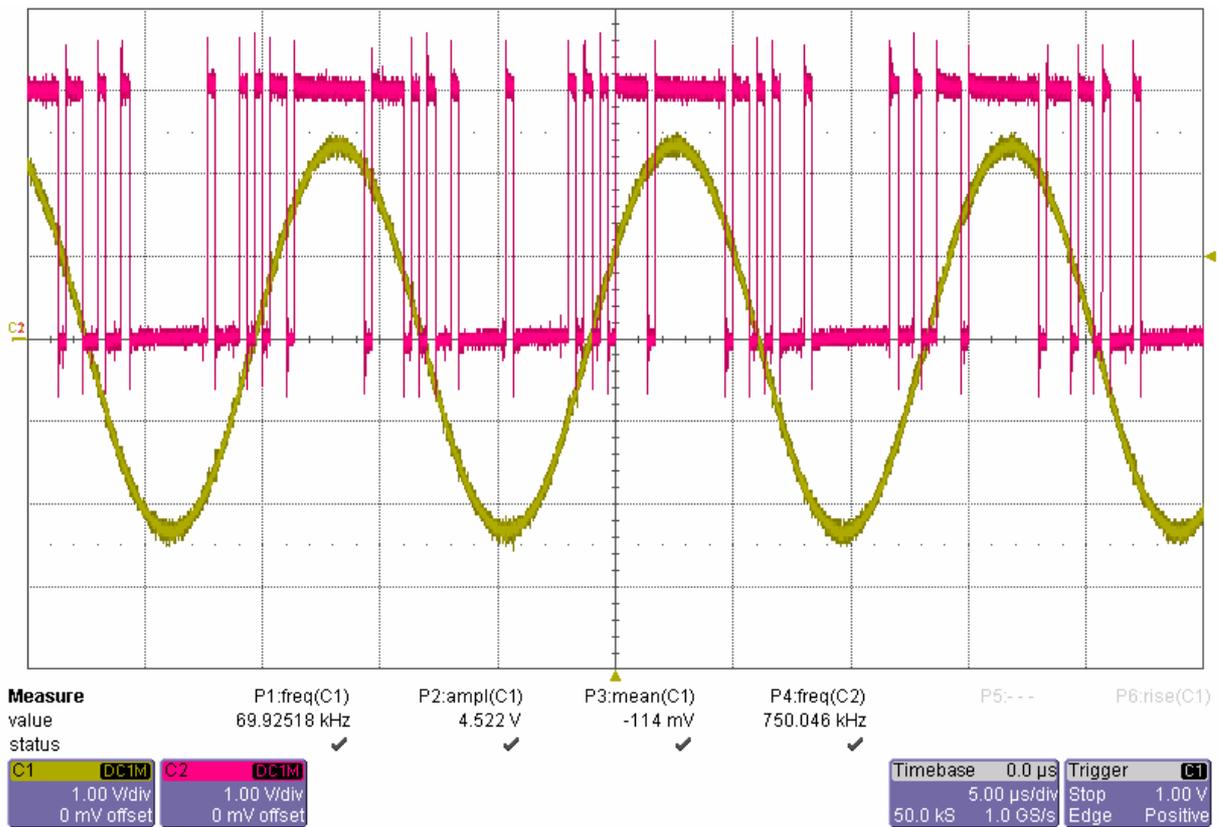


Figura 5.39. Salida del modulador de segundo orden con una señal de entrada sinusoidal de 70 kHz y 4 500 mV muestreada a 3 MHz.

5.3.8. Adquisición y procesado digital de la señal de salida del modulador

Una vez montados y probados (etapa a etapa) los circuitos del prototipo del convertidor A/D sigma-delta que se muestra en la Figura 5.40, se han conectado a la FPGA, cuyo diseño se muestra en la Figura 5.41, que tiene, además del biestable D del convertidor, un divisor de frecuencia, descrito mediante el código VHDL que se muestra en la Tabla 5.6, que reduce la frecuencia del reloj de 48 MHz que tiene la placa de la FPGA a la frecuencia deseada para el modulador. Además, a este diseño se ha añadido el analizador lógico *Signal Tap* de Altera [118], que utiliza la interfaz JTAG [119] de la FPGA para comunicarse con el computador a través de su puerto paralelo. Este analizador lógico se ha configurado para adquirir 64 kb, es decir, 65 536 muestras de un bit, utilizando como reloj de adquisición el mismo que el del modulador, pero utilizando el flanco de bajada, y como señal de disparo la salida de sincronismo digital del generador de señales. De esta forma, cuando comienza un periodo de la señal de entrada se genera la señal de disparo para comenzar la adquisición. Una vez adquirida la señal digital, codificada en binario, se ha convertido a una señal bipolar, correspondiendo el '0' con el '-1' y el '1' con él mismo. Después de esta conversión, se ha importado y procesado utilizando el procedimiento programado en Matlab de la

Tabla 5.7. Con los datos disponibles en el computador se calculó la respuesta del modulador a distintas frecuencias, de muestreo y de la señal de entrada. Para una frecuencia de muestreo de 3 MHz se ha probado con señales de entrada de 700 Hz (Figura 5.43), de 7 kHz (Figura 5.44) y de 70 kHz (Figura 5.45). Las respuestas se han normalizado de forma que el lóbulo correspondiente al tono de la señal se corresponda con 0 dB. Se comprueba que por debajo de unos 150 kHz el nivel del ruido del modulador está por debajo de los 50 dB, por lo que con un factor de sobremuestreo bajo, por ejemplo de 15, se obtienen buenos resultados. En el siguiente apartado se completan las pruebas a una frecuencia de muestreo de 6 MHz e incluyendo la etapa de filtrado del convertidor.

$$OSR = \frac{f_s/2}{f_c} = \frac{1.5 \text{ MHz}}{100 \text{ kHz}} = 15 \quad (5.33)$$

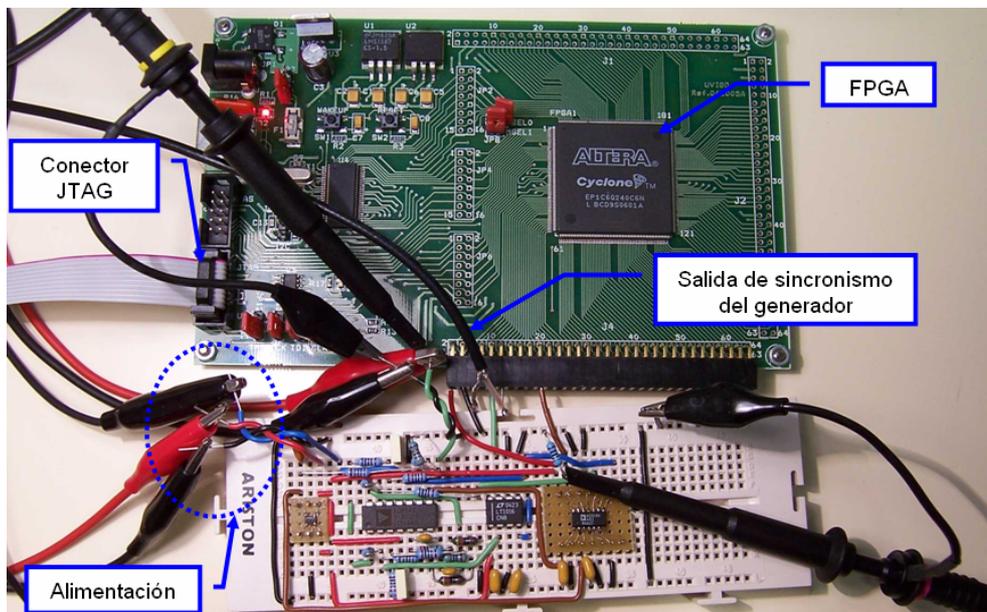


Figura 5.40. Fotografía del prototipo del convertidor sigma-delta con el modulador de segundo orden.

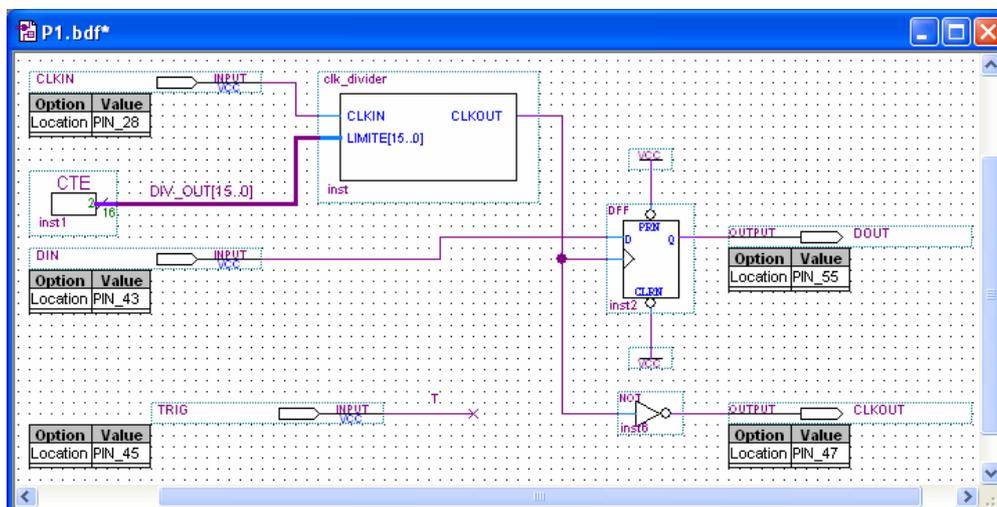


Figura 5.41. Esquema del diseño de la FPGA del prototipo de convertidor sigma-delta.

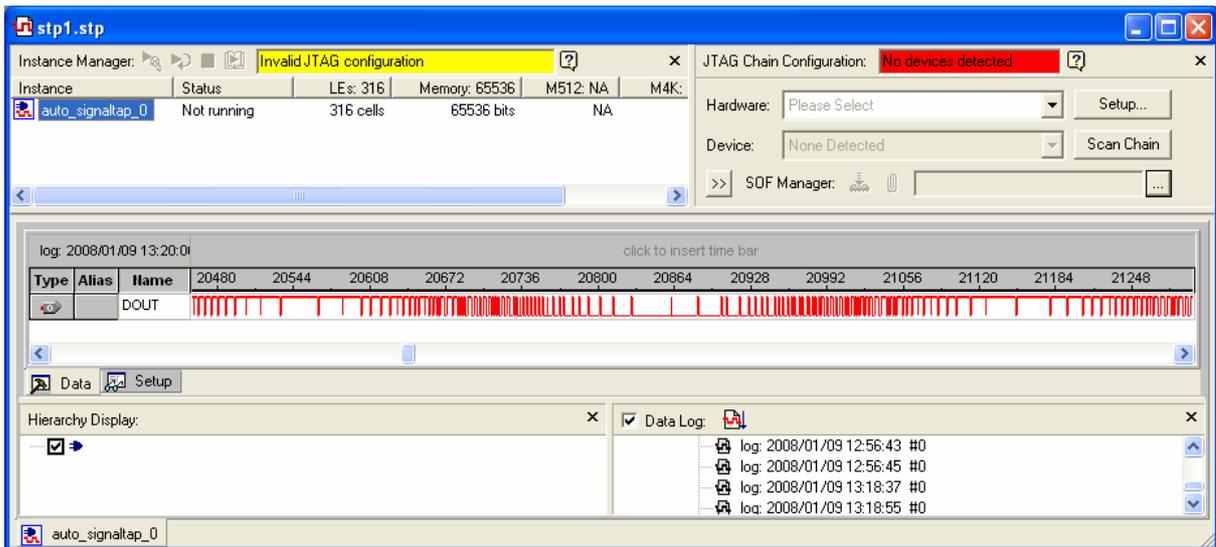


Figura 5.42. Panel de control del programa Signal Tap para adquirir señales de la FPGA.

Tabla 5.6. Código VHDL del divisor de frecuencia.

```

library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;

-----
-- Genera a partir del reloj CLKIN una frecuencia=CLKIN Mhz/(2*contador)
-----

entity clk_divider is
  port( CLKIN: in std_logic; CLKOUT: out std_logic; LIMITE: in std_logic_vector(15 downto 0));
end clk_divider;
architecture clk_divider_comp of clk_divider is
  signal contador: integer range 0 to 65536;
  type estados is (E0,E1);
  signal a_state: estados;
begin
  clk: process(CLKIN)
  begin
    if (CLKIN='1' and CLKIN'event) then
      case a_state is
        when E0=>
          CLKOUT<='1';
          if (limite=conv_std_logic_vector(contador,16)) then
            contador<=1; a_state<= E1;
          else
            contador<=contador+1; a_state<= E0;
          end if;
        when E1=>
          CLKOUT<='0';
          if (limite=conv_std_logic_vector(contador,16)) then
            contador<=1; a_state<= E0;
          else
            contador<=contador+1; a_state<= E1;
          end if;
        when others=> a_state<=E0;
      end case;
    end if;
  end process;
end clk_divider_comp;
    
```

Tabla 5.7. Código de Matlab que procesa la señal de salida del modulador.

<pre> %Nyquist Frequency fs2=3000000; %Cut off frequency of Low Pass FIR Filter Fc=300000; %Filter order m=1024; %Quantity of sampled data to be processed M=2^16; %Data imported from the Modulator through the %FPGA memory d=DATOS100KHZ_4500mV_FS6MHZ; %limiting the data length t=length(d); if M<=t t=M; end; %initializing the variable to stored the data datos=zeros(t,1); for i=1 : t; datos(i)=d(i); end; %FIR Filter coeficients calculus fc=Fc/(2*fs2); h=zeros(m,1); for i=1 : m; k=(i-1)-m/2; if k==0 sinc=2*fc; else sinc=(sin(2*pi*fc*k))/(pi*k); end; window=0.42-0.5*cos(2*pi*(i- 1)/m)+0.08*cos(4*pi*(i-1)/m); h(i)=sinc*window; end; %Normalizing the gain of the filter to the unit s=0; for i=1 : m; s=s+h(i); end; </pre>	<pre> g=1/s; for i=1 : m; h(i)=g*h(i); end; %Fullfilling the filter kernel with zeros h=[h zeros((t-length(h)),1)]; %Convolving the data anf the Filter kernel y=conv(h,datos); ry=respuestalog(y); l=length(ry) m=max(ry); for i=1 : l; ry(i)=ry(i)-m; end; %Calculating the frequency responses h=[h zeros(length(h)-1,1)]; datos=[datos zeros(length(datos)-1,1)]; %function calculating the mag response in log scale rh=respuestalog(h); rdatos=respuestalog(datos); ldatos=length(rdatos) %Establishing the main tone to 0 dB m=max(rdatos); for i=1 : ldatos; rdatos(i)=rdatos(i)-m; end; %Calculating the frequency vector for the x- axis f=zeros(1,1); for i=1 : l; f(i)=-(fs2/l)+(i*fs2/l); end; %Representing the results semilogx(f,rh,f,rdatos,f,ry); grid on; axis([1000 fs2 -90 5]); </pre>
---	--

Tabla 5.8. Función que calcula el módulo en dB de la respuesta en frecuencia de una secuencia.

<pre> %respuesta en frecuencia de una secuencia sh function[shr]=respuestalog(sh) shfft=fft(sh); nshfft=length(sh); shr=zeros(nshfft/2,1); for i=1 : nshfft/2; shr(i)=20*log10(sqrt(real(shfft(i))^2+imag(shfft(i))^2)); end; end; </pre>

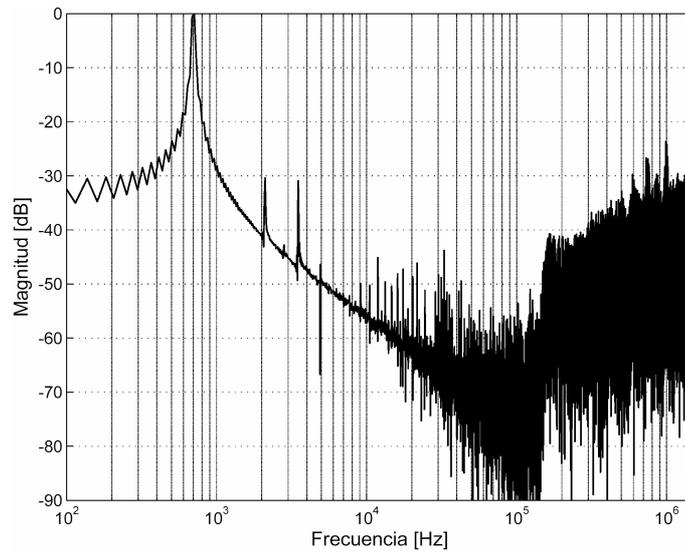


Figura 5.43. Respuesta del modulador ($f_i = 700$ Hz).

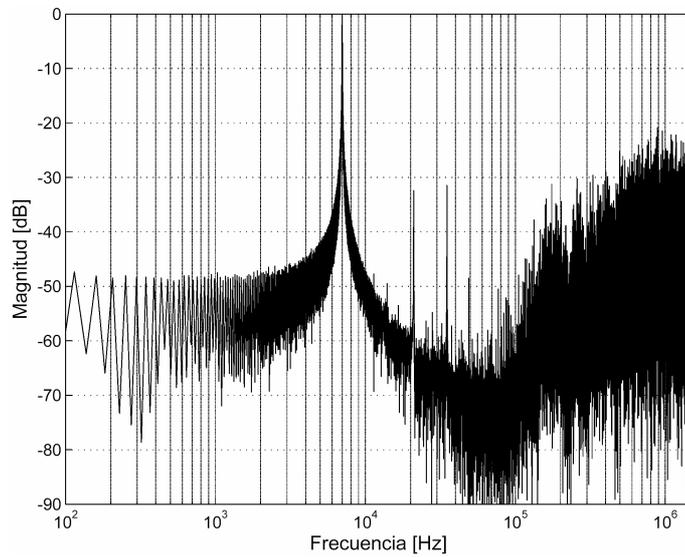


Figura 5.44. Respuesta del modulador ($f_i = 7$ kHz).

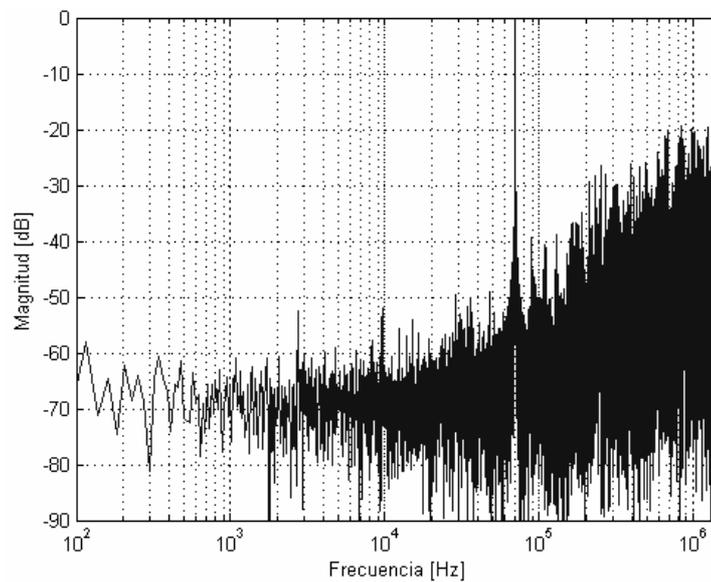


Figura 5.45. Respuesta del modulador ($f_i = 70$ kHz).

5.3.9. Resultados prácticos del modulador después del filtrado

Después de evaluar la respuesta del modulador a 3 MHz se ha aumentado la frecuencia hasta los 6 MHz y se han repetido las pruebas con una señal de entrada de 10 kHz (Figura 5.46) y 200 kHz (Figura 5.47). Como se muestra en dichas figuras, ahora se ha filtrado la señal del modulador con un filtro FIR, incluido en el programa de la Tabla 5.7, con esto se consigue disponer de la secuencia de la señal reconstruida (la señal que se obtiene de la convolución de la secuencia de salida del modulador con el núcleo del filtro) para hacer el análisis de su respuesta dinámica. Este análisis se ha hecho mediante el test de la sinusoidal que mejor se ajusta [40] y se han obtenido los resultados indicados en la Tabla 5.9.

Tabla 5.9. Resultados de la caracterización de la dinámica del modulador.

Parámetro		Test 1	Test 2
Denominación	Abreviatura		
Frecuencia de muestreo	f_s	6 MHz	6 MHz
Frecuencia de la banda	f_B	30 kHz	300 kHz
Factor de sobremuestreo	OSR	100	10
Frecuencia de la señal de entrada	f_i	10 kHz	200 kHz
Amplitud de la señal de entrada	A	4 500 mVpp	4 500 mVpp
Error de ganancia	Gerror	6 %	1 %
Relación señal ruido más distorsión	SINAD	58 dB	71 dB
Número equivalente de bits	ENOB	9.35 bits	11.5 bits

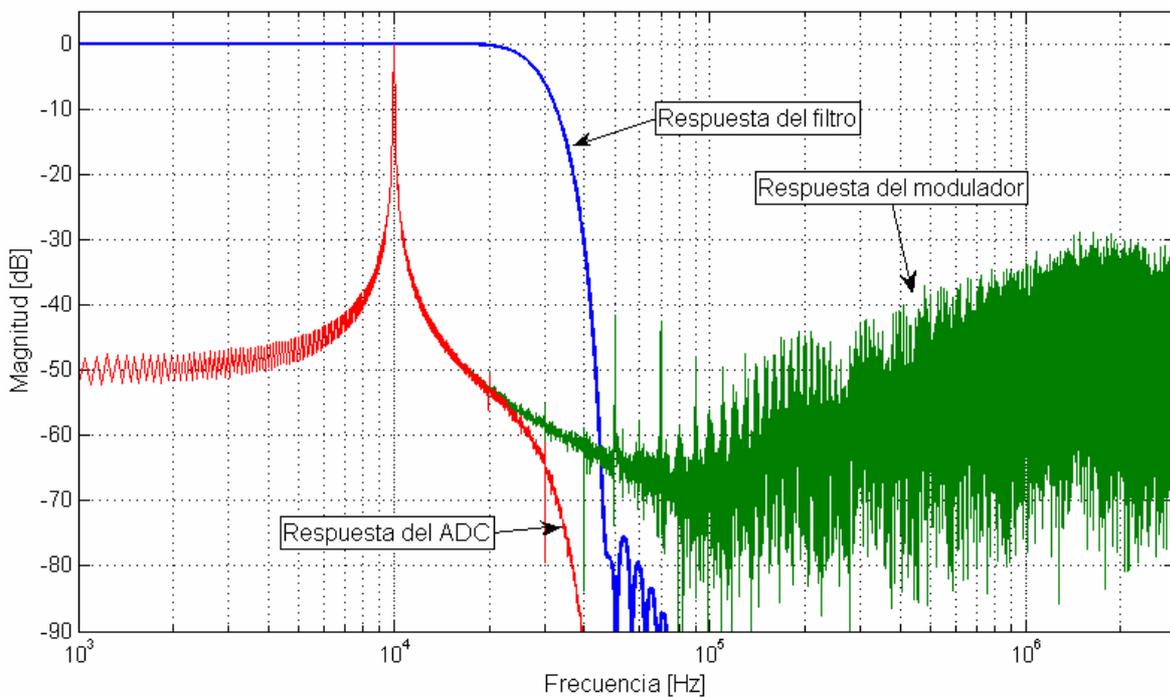


Figura 5.46. Respuesta en frecuencia del convertidor Sigma Delta ($f_s = 6$ MHz, $f_c = 30$ kHz, $f_i = 10$ kHz).

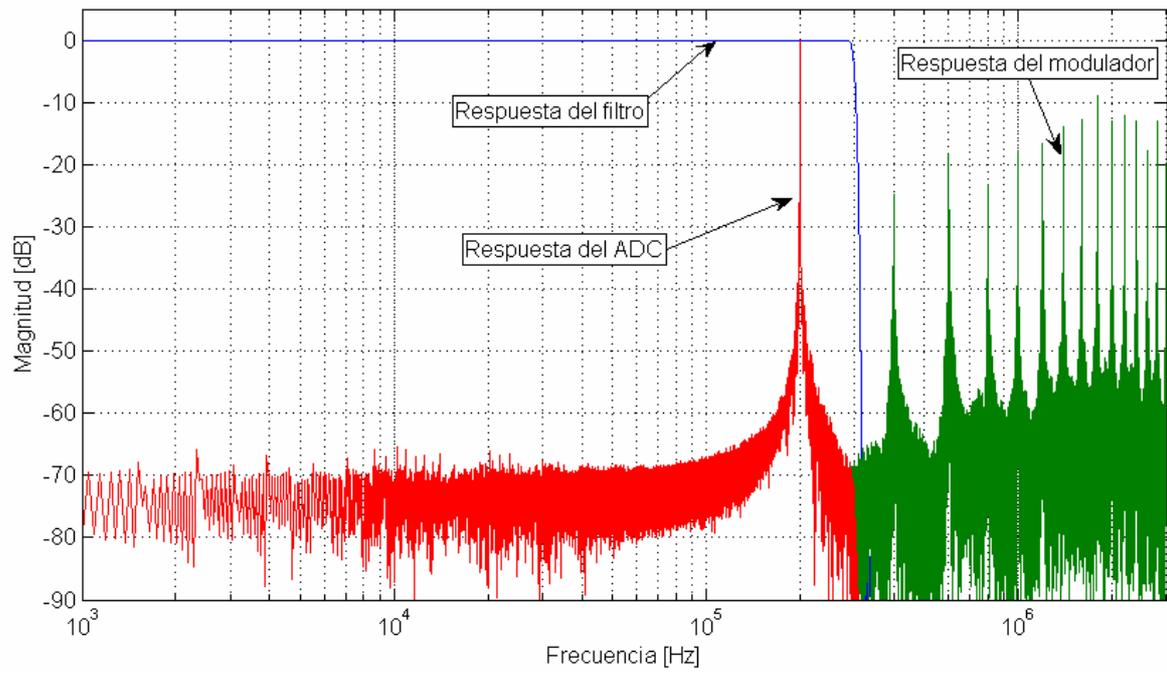


Figura 5.47. Respuesta en frecuencia del convertidor sigma-delta ($f_s = 6$ MHz, $f_c = 300$ kHz, $f_i = 200$ kHz).

CAPÍTULO 6. METODOLOGÍA DE ENSEÑANZA DE LOS CIRCUITOS ADC PIPELINE Y SIGMA-DELTA

6.1. Introducción

En los últimos años el número de estudiantes en las escuelas de ingeniería se ha visto reducido, quizás debido a la percepción de la dificultad y sentido práctico del esfuerzo que requieren estos estudios [120]. Esto no es una novedad en los países avanzados, los cuales ya han sufrido este cambio con anterioridad. Por ello es necesario mejorar las metodologías docentes de forma que se garantice la calidad de la enseñanza, sin disminuir el nivel y, al mismo tiempo, haciendo más asequible el estudio de forma que dicha percepción se invierta. Por otro lado, la enseñanza de la tecnología, que está permanentemente cambiando y revolucionando los hábitos de la vida diaria, también necesita de profesionales con capacidades de adaptación y constante espíritu de superación. Por tanto, conseguir profesionales que sean capaces de enfocar el cambio constante y que puedan mantenerse al límite de sus conocimientos, de forma que mantengan una actitud de progreso, requiere que se desarrollen capacidades de adaptación a los distintos retos y colaboración entre colegas [121].

Por todo ello, en éste capítulo se presenta una metodología docente para la enseñanza de los circuitos convertidores A/D que aplica los conceptos estudiados y las metodologías desarrolladas en los capítulos anteriores.

Dicha metodología docente esta orientada especialmente al estudio de las arquitecturas avanzadas que poseen alta velocidad de conversión y alta resolución. Los convertidores que mejor se ajustan a estas dos características son: en cuanto a la velocidad los *pipeline* y en cuanto a la resolución los sigma-delta. Estos convertidores se basan en importantes conceptos como por ejemplo: funcionamiento multietapa, sobremuestreo, sincronización, latencia, modulación, filtrado y diezmado, etc. La metodología docente que se presenta combina ejemplos prácticos de simulación, para estudiar mejor los principios de funcionamiento de estos convertidores, con el análisis de su comportamiento dinámico para caracterizarlos a partir de dichos resultados de simulación. Se pone especial atención en el uso de la normativa sobre los circuitos convertidores, principalmente en lo relativo a la terminología y definiciones, y a los métodos de caracterización.

6.2. Objetivos docentes y metodología general

La práctica es esencial en cualquier proceso de aprendizaje y, especialmente, en la enseñanza de la ingeniería electrónica, donde se hace imprescindible un proceso de aprendizaje basado en la aplicación de los conceptos teóricos a la práctica, previa simulación y análisis de los resultados [122], es decir, el proceso de aprendizaje se lleva a cabo bajo la perspectiva del proceso de diseño, que es esencial en cualquier proceso de ingeniería [123]. Además, en la enseñanza de la ingeniería es necesario alcanzar dos objetivos básicos [124]. El primero es proporcionar a los estudiantes métodos que les ayuden a resolver problemas y el segundo trabajar con ellos cada curso en un proceso de mejora continua, en el que los resultados de las evaluaciones no sirvan sólo para evaluar a los alumnos, sino también la propia metodología docente. En la parte de la electrónica que estudia los circuitos ADC sigma-delta y *pipeline*, sus arquitecturas, principio de funcionamiento y test, la práctica es especialmente compleja porque es necesario unir conceptos de electrónica analógica-digital con conceptos de procesado de señal, para lo que es necesario aplicar conocimientos matemáticos sobre sistemas continuos y discretos. Además de estos conocimientos teóricos, también es necesario poseer destreza en el manejo del hardware y del software específicos para simular y montar los circuitos que se estudian. Estos aspectos conllevan una serie de inconvenientes metodológicos, operativos, y económicos.

Debido a estas problemáticas se propone utilizar simuladores mixtos de libre distribución cuando la complejidad de las prácticas no requiera versiones profesionales. En el caso de sistemas digitales complejos [125], se propone utilizar software de libre distribución para simulación de diseños hardware descritos en VHDL. En el caso de que el procesado digital sea muy complejo se propone utilizar parcialmente el simulador mixto OrCAD PSpice [126], [127] (o un compilador de C en el caso de que se simule un modelo discreto) y realizar luego el post procesado en el simulador de VHDL utilizando alguna de sus funciones de almacenamiento de datos como interfaz entre el simulador mixto y el de VHDL. Por ejemplo, si se simula un modulador de un convertidor de sobremuestreo en C o Matlab (ver el ejemplo de simulación del Apartado 3.7.2), su salida se puede almacenar en un fichero de disco con el formato adecuado para que, a continuación, sirva para inicializar un componente de memoria en el diseño de VHDL.

Además, habitualmente los libros de texto enseñan este tipo de circuitos bien desde un punto de vista de procesado de señal [74] o bien desde un punto de vista de tecnología electrónica [1], [128], [129], pero no desde las dos perspectivas simultáneamente; aunque, por supuesto, hay excepciones, como en [34] y [28], éstas no están al nivel de un libro de texto para estudiantes de grado. Debido a la dificultad en combinar las dos áreas de conocimiento y a la falta de actualización de los programas docentes, no se suele incluir el estudio práctico de estos convertidores en las enseñanzas de ingeniería electrónica de las universidades o se hace sólo de forma introductoria.

Los objetivos principales de incluir el estudio de los circuitos convertidores avanzados en los programas docentes son:

1. Promocionar el estudio de las técnicas modernas de conversión A/D.
2. Ligar los conocimientos que tienen los estudiantes en los campos de la electrónica y de procesado de señal, de forma que estos conocimientos se consoliden mejor.
3. Utilizar en la medida de lo posible herramientas de simulación profesionales, que tengan versiones de libre distribución, de demostración o de estudiante, de forma que en su futuro profesional su uso les resulte familiar.
4. Introducir el uso de la normativa aplicable habituando a los estudiantes de ingeniería a su utilización.
5. Realizar experiencias de simulación lo más realistas posibles de forma que sea más comprensible su funcionamiento si se construyen, dejando esta posibilidad para los trabajos libres. Estos trabajos se hacen en grupos de dos alumnos y son proyectos cortos en los que los alumnos tienen que desarrollar, bajo la tutela del profesor, una solución que ellos mismos han diseñado [130], [131]. Trabajar con proyectos hace que el alumno se enfrente a las dificultades de forma más realista y que tenga que utilizar un proceso iterativo para llegar a los objetivos. De esta forma, los alumnos fijan los conocimientos que previamente adquirieron de forma receptiva y adquieren otros conceptos nuevos por descubrimiento [132]. También obliga al intercambio de experiencias entre los distintos grupos de alumnos [133], haciendo de la cooperación y del intercambio de conocimiento una necesidad. En este escenario, cuando un grupo necesita transmitir a otro sus

conocimientos, se enfrenta a una dificultad cuya superación le ayuda a alcanzar una mejor comprensión de ese conocimiento. Por tanto, esta metodología también promueve el desarrollo de capacidades de coordinación entre los estudiantes [134].

Con la metodología aplicada se pretende optimizar el tiempo utilizado en el laboratorio y favorecer al autoaprendizaje utilizando herramientas que el alumno ya ha utilizado en otras asignaturas y que, además, poseen versiones de libre distribución de las que el alumno puede disponer para practicar en las horas de estudio. Además, no son necesarios materiales e instrumentación específicos de laboratorio para simular los ejercicios, con la ventaja de que este hecho no dificulta el posterior montaje práctico porque tanto el simulador de circuitos PSpice como el de sistemas digitales VHDL son simuladores para síntesis, es decir, tienen en cuenta las restricciones reales de los componentes y los circuitos se montarían partiendo del mismo diseño que se simuló.

6.3. Recursos que se utilizan

Los recursos necesarios, que se muestran en la Tabla 6.1, son, además de los apuntes y transparencias en PowerPoint, el software de programación/simulación y las normas de aplicación. Concretamente, para diseñar, simular y caracterizar los convertidores A/D (uno *pipeline* y otro Σ - Δ) [135], [136], se utiliza en la medida de lo posible versiones *demo* o de estudiante de los siguientes programas, que los alumnos ya han utilizado en otras asignaturas y, por tanto, con los que están familiarizados:

- OrCAD PSpice para simular los circuitos descritos mediante esquemas o lista de redes.
- Un compilador de C para modelación algorítmica cuando los modelos son discretos.
- Hoja de cálculo Excel para el tratamiento de los datos.
- El compilador de C y la hoja de cálculo se pueden sustituir por el software Matlab para realizar de forma modular un proceso completo (diseño, simulación y test del convertidor) definido mediante funciones, que tenga salida gráfica.
- QuartusII para las simulaciones en VHDL sintetizable.

En cuanto a las normas que se utilizan [40]-[43], éstas son: la IEC 62008, la IEC 60748-4 y el IEEE Std. 1241-2000.

Tabla 6.1. Recursos que se utilizan en función de las tareas y objetivos.

Tareas	Resultados/objetivos de aprendizaje	Medios
Establecimiento de especificaciones a partir de los requisitos	Adquirir los conocimientos y destrezas necesarias	Presentación de PowerPoint, clase magistral, enunciados de los ejercicios
Diseño de los convertidores	Comprensión de los principios de funcionamiento	OrCAD PSpice para la realización de los esquemas de los modelos continuos y un compilador de C para los modelos discretos
Simulación	Análisis cualitativo de los resultados, comparación con las especificaciones	Probe PSpice para las simulaciones: análisis transitorio y FFT. Simulador de VHDL para los filtros digitales
Test y caracterización	Obtención de la ficha técnica adaptada a la normativa	Matlab Hoja de cálculo
	Análisis cuantitativo de los resultados, comparación con las especificaciones	Normativa
Verificación y validación de los resultados	Grado de cumplimiento de las especificaciones	Toda la documentación que se ha manejado

6.4. Prerrequisitos docentes

El estudio de estos convertidores requiere de unos conocimientos previos mínimos en electrónica analógica y digital, instrumentación y procesado de señal. Por tanto, se propone su aplicación en los cursos de las titulaciones de segundo ciclo. Se parte de que los alumnos ya conocen las estructuras tradicionales de convertidores A/D y sus características estáticas (INL, DNL, etc.) [44]. Además, se deben tener conocimientos de manejo del simulador PSpice, de Matlab y de un simulador de VHDL, esto no supone un problema ya que en la mayoría de las escuelas de ingeniería se imparte docencia en la que se utiliza este software [137].

6.5. Metodología para la enseñanza de los convertidores A/D pipeline

Partiendo de los conocimientos de base, se propone: hacer un repaso de los convertidores A/D paralelos y simular una estructura de 3 bits que incluya el decodificador de salida. A continuación diseñar y simular un circuito S&H. Luego, realizar una simulación conjunta. Después, se pasa a simular un convertidor D/A de 3 bits. Con estos circuitos se logran repasar temas ya estudiados y

afianzar su conocimiento. En este punto se está preparado para abordar la construcción de un convertidor semiparalelo o *half-flash*. Esto supone la antesala de la segmentación en los convertidores A/D. Una vez estudiados estos temas se está en condiciones de introducirse en el estudio de los convertidores multietapa de tipo *pipeline*. Para ello, se estudia una estructura lo suficientemente sencilla como para poder realizarla mediante PSpice y, al mismo tiempo, con todos los elementos importantes en estos convertidores. La programación de estos temas, cuyos apartados más importantes se esquematizan en la Figura 6.1 y la Figura 6.2, es la siguiente:

- 1) Introducción del principio de funcionamiento (0.5 h)
 - a. Presentación de la evolución de este tipo de convertidores:
Paralelo -> semi-paralelo -> semi-paralelo + S&H -> pipeline.
 - b. Estudio de los métodos de sincronización en los ADC multietapa.
- 2) Simulación de un convertidor pipeline de 5 etapas de 1-bit por etapa. (2 h)
 - a. Es la estructura más sencilla con muy buenos resultados.
 - b. Es fácil de entender.
- 3) Caracterización del convertidor simulado (2 h)
 - a. Análisis en el dominio de la frecuencia.
 - b. Análisis en el dominio del tiempo.
 - c. Obtención de la hoja de características del convertidor.
- 4) Propuesta de trabajos libres para los alumnos que quieren mejorar su evaluación.
 - a. Simulación de un ADC con más bits por etapa o con más etapas.
 - b. Análisis de los resultados por diferentes métodos de test.
 - c. Simulación de un convertidor que incluya lógica correctora del error.

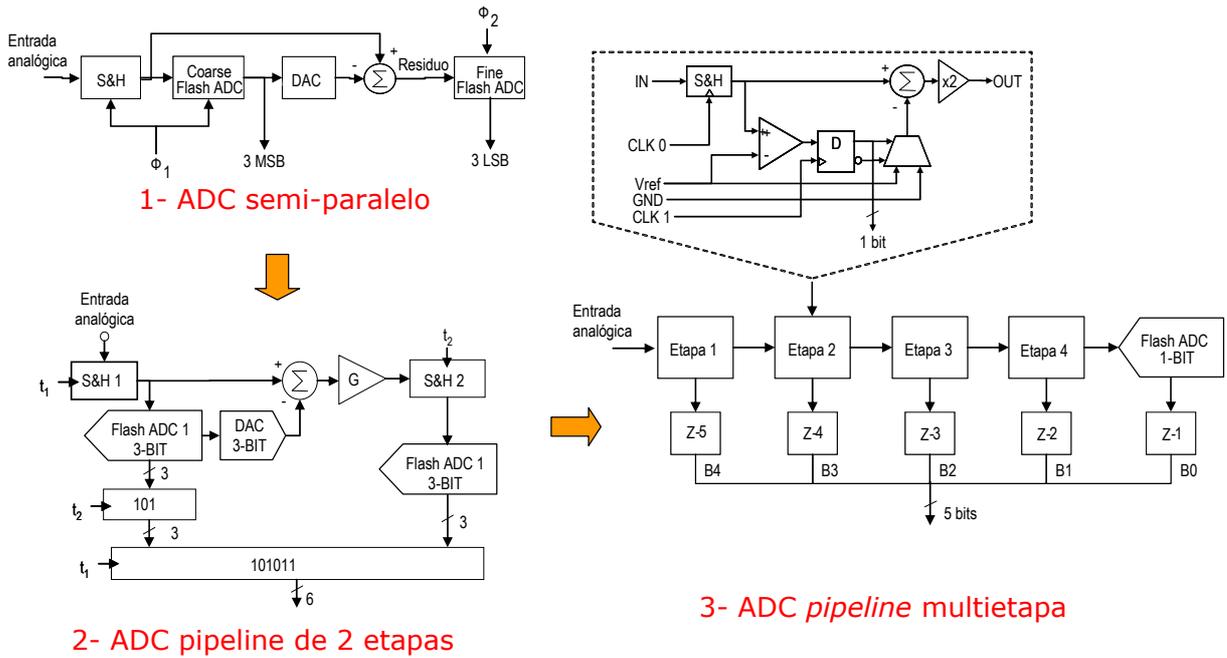


Figura 6.1. Evolución de los convertidores paralelos a los semiparalelos, a los pipeline de dos etapas y, finalmente, a los multietapa.

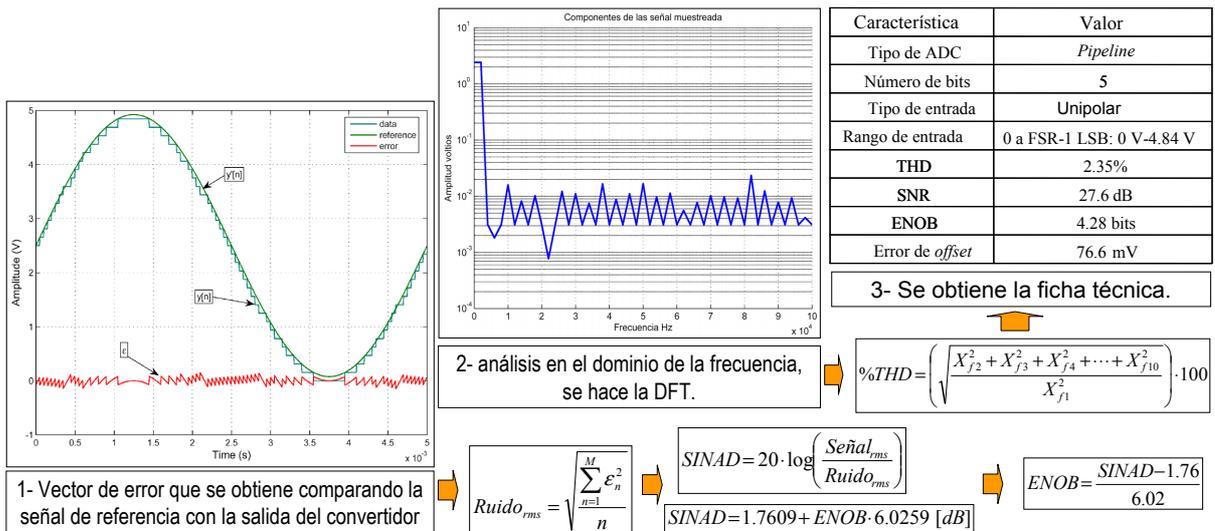


Figura 6.2. Esquema simplificado de los pasos para obtener la ficha técnica del convertidor a partir del análisis de los resultados de la simulación en el dominio del tiempo (transitorio) y de la frecuencia.

6.6. Metodología para la enseñanza de los convertidores A/D sigma-delta

En la metodología a seguir para estudiar los convertidores sigma-delta se comienza por explicar la base teórica del ruido de cuantificación, en qué consiste el sobremuestreo, qué ventajas tiene y cómo se realiza. De esta forma se introduce el tema de la modulación y de la ganancia de resolución. Después, se estudia el origen del modulador, pero en vez de seguir una deducción lógica matemática, se expone cómo se ha desarrollado a partir del modulador PCM (*Pulse Code Modulator*) utilizado en

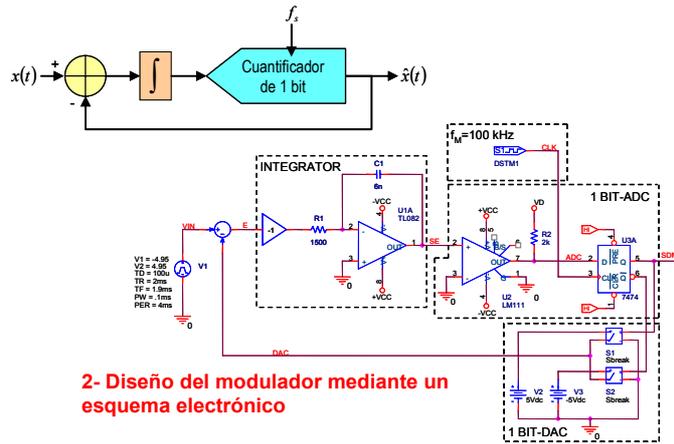
comunicaciones. Para terminar la exposición de los fundamentos, se presenta el diagrama de bloques básico del convertidor y, finalmente, se presenta la etapa de filtrado y diezmado. Una vez hecha la introducción teórica se hacen los ejemplos prácticos de simulación. Los pasos a seguir para simular el modulador se esquematizan en la Figura 6.3 y la forma de simular la etapa de filtrado y diezmado se muestra en la Figura 6.4. Finalmente, los resultados de la simulación y del test se muestran en la Figura 6.5. La programación de estos temas es la siguiente:

- 1) Introducción (1 h).
 - a. Presentación del ruido en los convertidores y el concepto de sobremuestreo.
 - b. Modificación del espectro del ruido de cuantificación.
 - c. Estudio de las posibilidades para obtener una ganancia de resolución.
 - d. Explicación de los orígenes y fundamentos modulador Σ - Δ .
- 2) Simulación de un modulador Σ - Δ por medio de OrCAD PSpice (0.5 h).
- 3) Simulación de un convertidor Σ - Δ completo de 5 bits (2 h).
 - a. Simulación de un modulador de primer orden mediante un algoritmo codificado en lenguaje C.
 - b. Generación de un fichero de datos con la salida del modulador correspondiente a la conversión de una señal sinusoidal. El fichero se genera con el formato de inicialización de memorias RAM de Altera.
 - c. Diseño y simulación de un filtro de diezmado mediante el software QuartusII de Altera.
- 4) Análisis de los resultados de simulación y obtención del ENOB (0.5 h).
- 5) Propuesta de extensión de las simulaciones mediante trabajos libres en los que los alumnos realizan variaciones sobre los ejemplos estudiados. Con estos trabajos voluntarios los alumnos pueden mejorar su evaluación. Ejemplos:
 - a. Implementación hardware del modulador del apartado 2.
 - b. Implementación del filtro de diezmado en un dispositivo FPGA modelo Cyclone.
 - c. Integración de los dos ejercicios anteriores y análisis de su funcionamiento.
 - d. Desarrollo e implementación de un filtro ecualizador de tipo FIR para corregir la respuesta en frecuencia del convertidor del apartado anterior.

1- Diseño del modulador mediante un algoritmo programado en C

```
#include "stdafx.h"
#include "math.h"
#define M 4096
int main(int argc, char* argv[])
{
    double PI=3.141592654;
    unsigned long i=0;
    unsigned char b=0x00;
    FILE *f;

    double x=0,e=0,v=0,w=0,y=0;
    f=fopen("ROMdata.MIF","w");
    fprintf(f,"--Example of a generated Memory Initialization File\n(mif)\n");
    fprintf(f,"WIDTH=1,\nDEPTH=%d,\n\n",M);
    fprintf(f,"ADDRESS_RADIX=UNS;\n");
    fprintf(f,"DATA_RADIX=UNS;\n\n");
    fprintf(f,"CONTENT BEGIN\n");
    for(i=0;i<M;i++)
    {
        x=0.94*sin(2*PI*(i/(double)M));
        e=x-y;
        w=e+w;
        v=e+v;
        if(w>0)
        { y=1; b=1;}
        else
        { y=-1; b=0;}
        fprintf(f,"%d : %d;\n",i,b);
    }
    fprintf(f,"END;");
    fclose(f);
    return 0;
}
```



2- Diseño del modulador mediante un esquema electrónico

3- Simulación temporal del modulador mediante PSpice

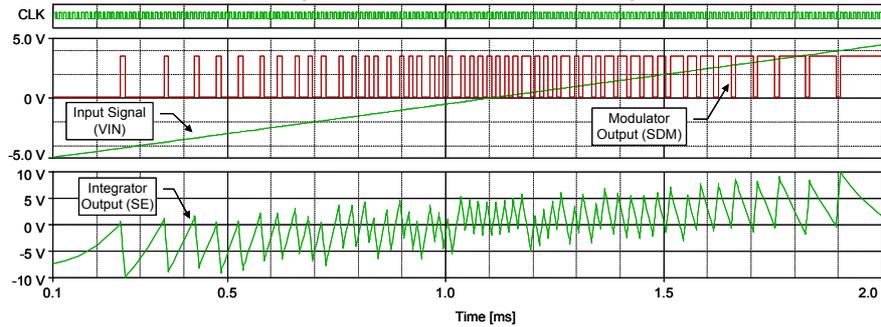


Figura 6.3. Simulación del modulador de primer orden mediante un algoritmo programado en lenguaje C y mediante un circuito electrónico simulado con OrCAD PSpice, en este caso se muestran los resultados de la simulación transitoria mediante el programa Probe.

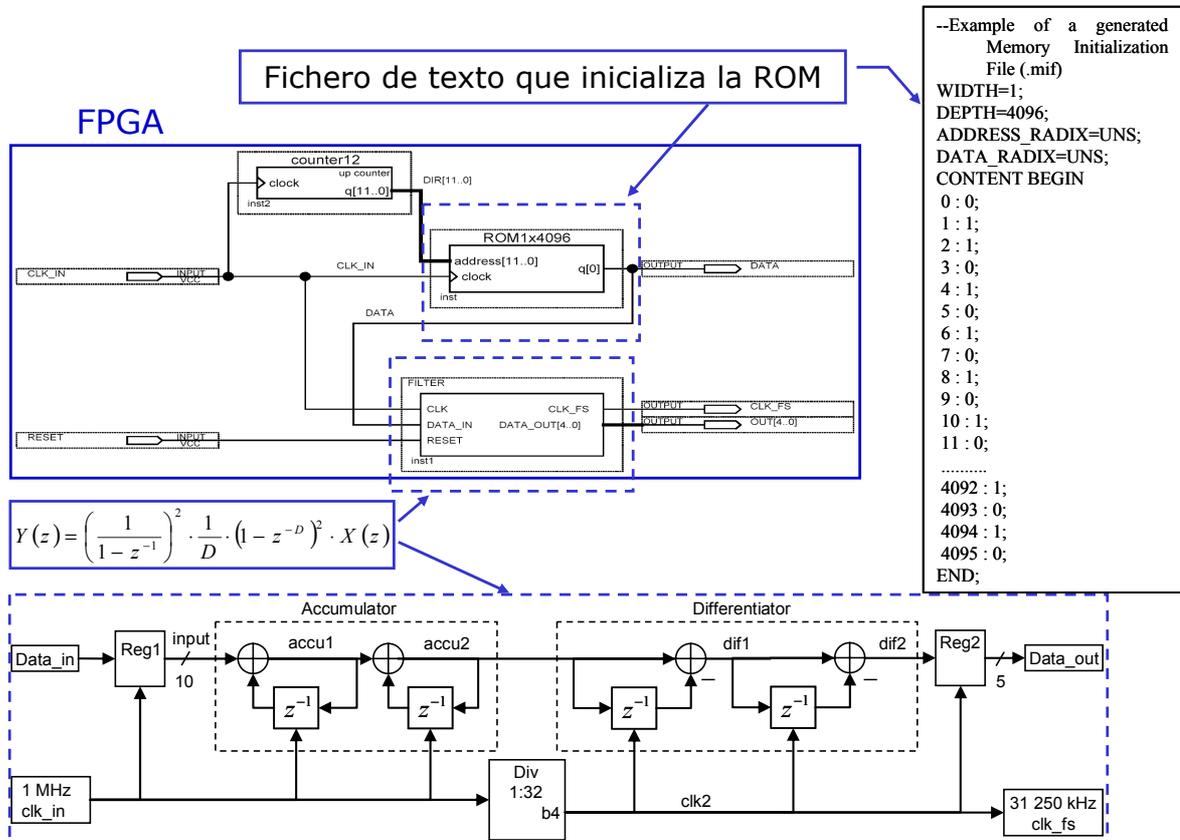


Figura 6.4. Esquema de la etapa de filtrado y diezmo del convertidor sigma-delta. La simulación se realiza con el software Quartus II y los datos de entrada se obtienen de la ROM, cuyo contenido se inicializa con los datos obtenidos de la simulación del modulador que se describió con el algoritmo programado en C.

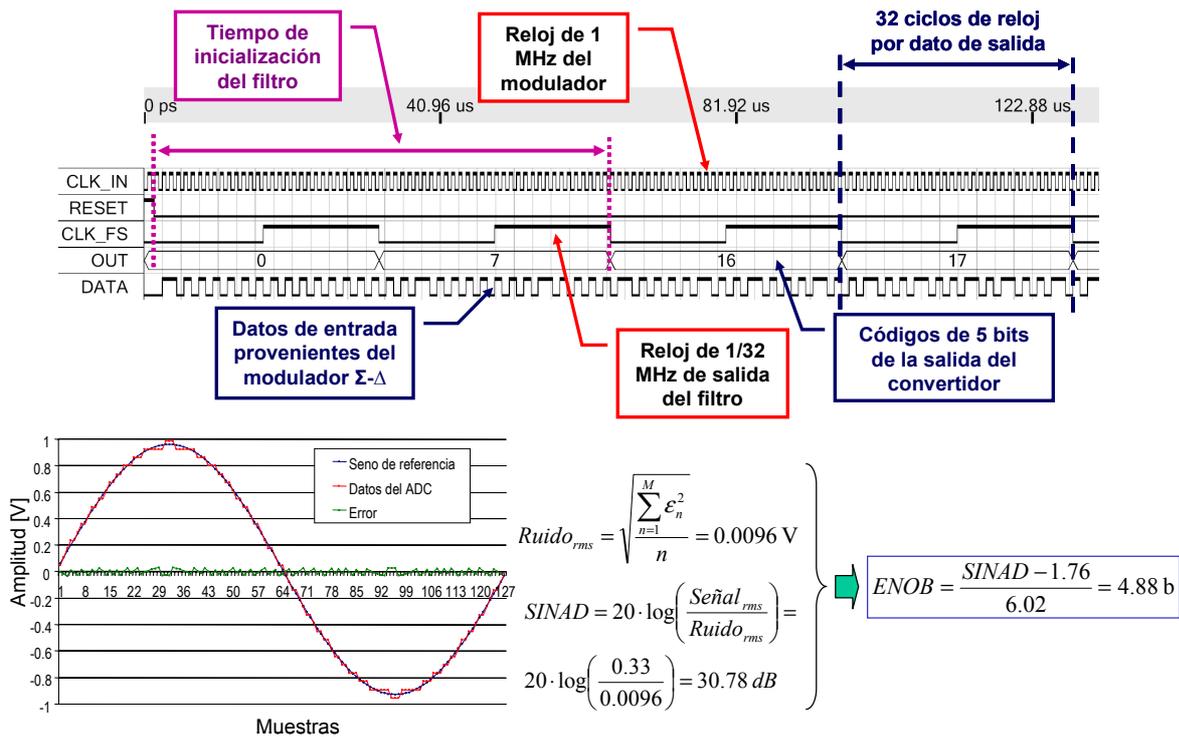


Figura 6.5. Los resultados de simular el convertidor se pasan a la hoja de cálculo y se obtienen las características más importantes comparando la señal de salida con la sinusoidal que mejor se ajusta.

6.7. Metodología de evaluación

Se utiliza una metodología de evaluación continua presencial basada en una evaluación diagnóstica, una evaluación formativa y una prueba final individual [138]. La evaluación diagnóstica consiste en un test previo de conocimientos, la evaluación formativa consiste en un conjunto de casos prácticos con ejercicios de simulación y de análisis de los resultados. Y por último, la prueba personal, que consiste en realizar un test posterior y la realización de un caso práctico. Esta metodología no supone un aumento considerable en el tiempo dedicado por el equipo docente porque los grupos de alumnos son reducidos.

6.8. Resultados de la aplicación de la metodología docente

Esta metodología de enseñanza de los convertidores A/D se ha aplicado en el segundo cuatrimestre de los cursos 2006/2007 y 2007/2008 en la asignatura de 5º curso Laboratorio de Sensores y Acondicionadores de la E.T.S. de Ingenieros de Telecomunicación de la Universidad de Vigo.

Para la evaluación se han realizado una serie de cuestiones agrupadas de la siguiente manera:

- a. Un test para conocer el perfil del alumno en cuanto al uso que ha hecho en el pasado de los programas de simulación de circuitos
- b. Un test de conocimientos previos sobre los temas que se tratan.
- c. Un test sobre el uso que hacen habitualmente de la normativa.
- d. Un test de opinión en el que se mide la importancia relativa que los alumnos le dan a los temas tratados.

En total se ha realizado el test, de forma individual, a un total de 23 alumnos, y los trabajos de simulación y libres los han realizado en grupos de dos, que es como se organiza el Laboratorio de Sensores y Acondicionadores en el que se ha implantado la metodología.

Según los resultados de la Tabla 6.2, todos los alumnos han utilizado alguna vez el software de simulación de circuitos electrónicos y la mayoría lo han hecho asiduamente. Sin embargo, la mayoría nunca ha simulado un circuito convertidor A/D ni con el programa PSpice ni con un lenguaje de programación de alto nivel como C. Estos resultados han sido similares los dos cursos en los que se ha puesto en práctica la metodología presentada. En cuanto a la última cuestión de la Tabla 6.2, el primer año, un cuarto de los alumnos encuestados respondieron que habían utilizado VHDL para simular algún tipo de filtro o demodulador y el resto nunca, y el segundo año ninguno. Se puede concluir que, en general, han utilizado un simulador para comprobar los diseños antes de realizar los montajes de los circuitos.

A través de las cuestiones de la Tabla 6.3 se ha preguntado a los alumnos sobre lo importante que consideran tanto el uso de las herramientas de simulación como el estudio de los convertidores A/D. De las respuestas se deduce que la mayoría consideran importantes o imprescindibles la utilización de los simuladores y el estudio de los convertidores. De la Tabla 6.4 se deduce que la mayoría de los alumnos no saben que existen o desconocen las diferentes normas y métodos de caracterización de convertidores.

En la Tabla 6.5 se presentan los resultados del cuestionario de conocimientos generales sobre los convertidores, en la Tabla 6.6 se muestran las cuestiones relativas a la etapa de filtrado y diezmado y en la Tabla 6.7 se muestran las cuestiones específicas de los convertidores sigma-delta y *pipeline*.

Estas cuestiones se han realizado dos veces, una antes de la realización del curso y otra después. Como resultado, se resalta que las respuestas no correctas se han reducido un promedio de un 39% en el curso 2006/2007 y en un 51% en el curso 2007/2008. Este incremento puede ser debido a que, como en el primer curso la disminución en respuestas incorrectas y en blanco se consideró insuficiente, en el curso 2007/2008, antes de su finalización, se realizó un repaso al tema de 30 min y esto ayudó a aclarar dudas y fijar conceptos. Este resultado indica que la propia evaluación ha servido para mejorar la metodología.

Tabla 6.2. Cuestiones sobre el uso de los simuladores de circuitos electrónicos.

Nº	Pregunta	Curso	Nunca	Alguna vez	Muchas veces	Siempre
1	¿Has utilizado alguna vez el software OrCAD/PSpice para simular circuitos electrónicos?	06/07	0%	75%	17%	8%
		07/08	0%	100%	0%	0%
2	¿Has utilizado alguna vez el software OrCAD/PSpice para simular convertidores Analógico-Digitales?	06/07	92%	8%	0%	0%
		07/08	100%	0%	0%	0%
3	¿Has utilizado alguna vez el lenguaje de programación C o similar para simular convertidores Analógico-Digitales?	06/07	100%	0%	0%	0%
		07/08	100%	0%	0%	0%
4	¿Has utilizado alguna vez el lenguaje VHDL y los circuitos FPGA para implementar algún tipo de demodulador que utilice filtros paso-bajo?	06/07	75%	25%	0%	0%
		07/08	100%	0%	0%	0%
5	¿Has simulado alguna vez los circuitos antes de realizarlos físicamente?	06/07	17%	75%	8%	0%
		07/08	14%	72%	14%	0%

Tabla 6.3. Cuestiones sobre la opinión de los alumnos a cerca de la importancia del tema.

Nº	¿ Crees que ...	Curso	Prescindible	Importante	Imprescindible
6	...las herramientas de simulación en la ingeniería electrónica son?	06/07	0%	25%	75%
		07/08	7%	49%	44%
7	...el estudio de los métodos de conversión A/D es?	06/07	0%	50%	50%
		07/08	14%	58%	28%

Tabla 6.4. Cuestiones sobre el uso de la normativa.

Nº	¿Sabías que existen diferentes...	Curso	No	Sí, pero no las conozco	Sí, pero nunca las he utilizado	Sí, las conozco y las he utilizado
8	...normas para convertidores A/D?	06/07	75%	25%	0%	0%
		07/08	57%	28%	15%	0%
9	...métodos para caracterizar convertidores D/A?	06/07	50%	50%	0%	0%
		07/08	42%	42%	16%	0%

Tabla 6.5. Resultados del cuestionario, sobre conocimientos generales de convertidores A/D, realizados antes y después del curso. (-: respuesta en blanco, x: incorrecta, v: correcta). Se presenta la reducción en % de las respuestas totales en blanco e incorrectas. $\% = \{1 - [(-+X)_{antes} / (-+X)_{después}]\} \times 100$.

Nº	Pregunta	Curso	Antes			Después			Reducción en respuestas incorrectas
			-	X	V	-	X	V	
10	Ordena según la velocidad de conversión los siguientes tipos de convertidores: integración, aproximaciones sucesivas, paralelo, sigma-delta, pipeline.	06/07	4	6	0	0	6	4	40%
		07/08	1	4	2	0	2	5	60%
11	¿Cómo se puede conseguir aumentar la resolución en un ADC?	06/07	5	2	3	0	2	8	71%
		07/08	2	5	0	0	1	6	85%
12	¿Cuánto vale el valor eficaz del ruido de cuantificación de un ADC?	06/07	8	2	0	4	2	4	40%
		07/08	4	1	2	0	1	6	80%
13	¿Cómo se relaciona el ruido de cuantificación y el número efectivo de bits del convertidor?	06/07	6	1	3	1	6	3	0%
		07/08	3	0	4	0	1	6	33%
14	¿Cómo es la distribución del ruido de cuantificación en los ADC?	06/07	7	2	1	1	7	2	11%
		07/08	2	2	3	1	1	5	75%
15	¿Por qué es necesario el diezmado en los convertidores de sobremuestreo?	06/07	8	1	1	2	5	3	22%
		07/08	4	1	2	1	0	6	20%
16	¿Cuál es la mejora que se puede conseguir con el sobre muestreo? ¿Y el coste que supone, cuál es?	06/07	6	2	2	2	4	4	25%
		07/08	4	0	3	2	0	5	75%
17	Cita 5 características importantes de los convertidores A/D.	06/07	4	1	5	0	2	8	60%
		07/08	6	1	0	1	2	4	57%

Tabla 6.6. Resultados del cuestionario, sobre conocimientos de la etapas de filtrado y diezmado, realizados antes y después del curso. (-: respuesta en blanco, x: incorrecta, v: correcta). Se presenta la reducción en % de las respuestas totales en blanco e incorrectas.

Nº	Pregunta	Curso	Antes			Después			Reducción en respuestas incorrectas
			-	X	V	-	X	V	
18	¿Por qué se utiliza mucho un filtro de promediado basado en la función sinc?	06/07	7	2	1	4	5	1	0%
		07/08	3	4	0	1	4	2	28%
19	¿Qué relación hay entre el factor de diezmado y la longitud del filtro de promediado (número de etapas)?	06/07	9	0	1	3	6	1	0%
		07/08	6	1	0	2	4	1	14%
20	¿Por qué es necesario realizar un post-filtrado cuando se utiliza como filtro diezmador el promediado?	06/07	9	1	0	6	4	0	0%
		07/08	6	1	0	3	1	3	42%

Tabla 6.7. Resultados del cuestionario, sobre conocimientos específicos de los convertidores sigma-delta y pipeline, realizados antes y después del curso. (-: respuesta en blanco, x: incorrecta, v: correcta). Se presenta la reducción en % de las respuestas totales en blanco e incorrectas.

Nº	Pregunta	Curso	Antes			Después			Reducción en respuestas incorrectas
			-	X	V	-	X	V	
21	¿Cómo se consigue cambiar la forma de la distribución de probabilidad o del espectro del ruido de cuantificación?	06/07	9	0	1	4	3	3	22%
		07/08	5	0	2	0	3	4	40%
22	¿Cómo están relacionados el número de etapas necesarias para el filtro y el orden del modulador sigma-delta?	06/07	10	0	0	4	0	6	60%
		07/08	5	2	0	2	2	3	42%
23	¿Cuál es el efecto del modulador sigma-delta sobre el ruido de cuantificación?	06/07	10	0	0	2	3	5	40%
		07/08	5	2	0	1	1	4	71%
24	¿En un convertidor sigma-delta, por qué no es imprescindible un filtro antialiasing?	06/07	9	1	0	0	3	7	70%
		07/08	5	2	0	1	1	5	71%
25	¿Cuáles son las ventajas de los convertidores sigma-delta?	06/07	6	3	1	0	0	10	100%
		07/08	6	1	0	1	0	6	85%
26	¿Cómo se consigue aumentar la resolución en un convertidor sigma-delta?	06/07	9	0	1	2	1	7	66%
		07/08	5	1	1	1	0	6	83%
27	¿Por qué no es necesario un circuito S&H en los convertidores sigma-delta?	06/07	7	1	2	1	2	7	62%
		07/08	5	1	1	7	0	0	0%
28	¿Cuál fue el origen del modulador sigma-delta? ¿Cuál fue su predecesor y a qué se aplico?	06/07	10	0	0	0	5	5	50%
		07/08	7	0	0	3	0	4	57%
29	¿Cuál es la característica más importante que diferencia a los convertidores <i>pipeline</i> de los demás?	06/07	9	0	1	0	5	5	44%
		07/08	7	0	0	2	0	5	71%
30	¿Qué se consigue mejorar con un convertidor multietapa?	06/07	9	0	1	0	3	7	66%
		07/08	6	0	1	0	1	6	83%
31	¿Cuál es la diferencia entre un convertidor semi-paralelo y uno <i>pipeline</i> de dos etapas?	06/07	10	0	0	6	2	2	20%
		07/08	7	0	0	0	0	7	100%

CAPÍTULO 7. RESULTADOS OBTENIDOS

En este capítulo se resumen los principales resultados que se han obtenido de los estudios y ensayos que se han presentado en esta tesis doctoral.

7.1. Resultados sobre el estado del arte de los circuitos ADC

Los circuitos ADC utilizan cada vez más las técnicas de procesamiento digitales, que hacen disminuir el ruido, aumentar la velocidad de conversión y disminuir la dependencia de las características de los componentes analógicos. Como se desprende de la gráfica de la Figura 7.1, que se obtuvo recopilando datos de más de 500 convertidores A/D comerciales, por un lado, los convertidores Σ - Δ se han consolidado como los convertidores de mayor resolución sin que por ello se dejen de alcanzar altas velocidades de conversión, por ejemplo 24 bits a 1.5 MHz. En cuanto a los convertidores de tipo *pipeline*, éstos son los más rápidos y consiguen, al mismo tiempo, resoluciones relativamente elevadas, por ejemplo 16 bits a 120 MHz. Por último, los convertidores de tipo SAR siguen manteniendo buenas prestaciones de velocidad y resolución por lo que siguen siendo los idóneos para aplicaciones de control en las que prima el tiempo real. Finalmente, de la bibliografía, se deduce que las estructuras más novedosas de convertidores A/D combinan las técnicas de modulación sigma-delta multibit con las técnicas de conversión multietapa *pipeline* y, en la medida de lo posible, con circuitos de procesamiento digital. Esto es especialmente interesante en circuitos mixtos analógico-digitales que combinan pequeños módulos analógicos con circuitos digitales de tipo FPGA.

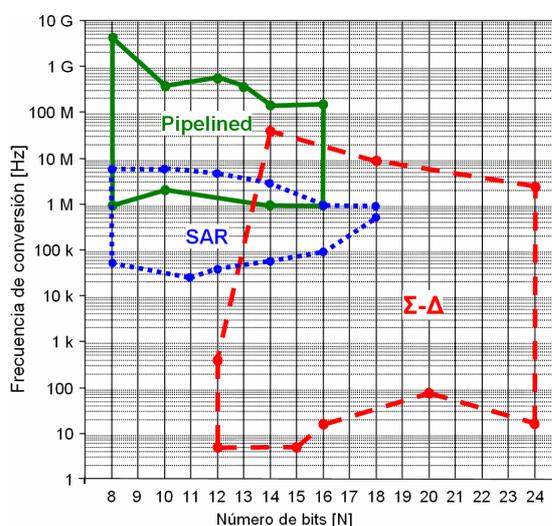


Figura 7.1. Resolución y frecuencia de conversión de los convertidores más utilizados.

7.2. Resultados del método de selección de los circuitos ADC

Como resultado final, según se dedujo en el Apartado 2.9, cuando se selecciona un convertidor A/D para convertir una señal con una determinada SNR, se debe elegir un convertidor que tenga una SINAD de 10 dB más que la SNR de la señal, esta regla se muestra de forma gráfica en la Figura 7.2. También se puede aplicar esta regla, a partir del la SNR máxima que se admite perder, para obtener el ENOB necesario del convertidor, según la siguiente ecuación:

$$ENOB = \frac{1}{2} \cdot \left[\log_2 \frac{2}{3} + \frac{SNR_s}{10} \cdot \log_2 10 - \log_2 \left(10^{\frac{SNR_t}{10}} - 1 \right) \right]$$

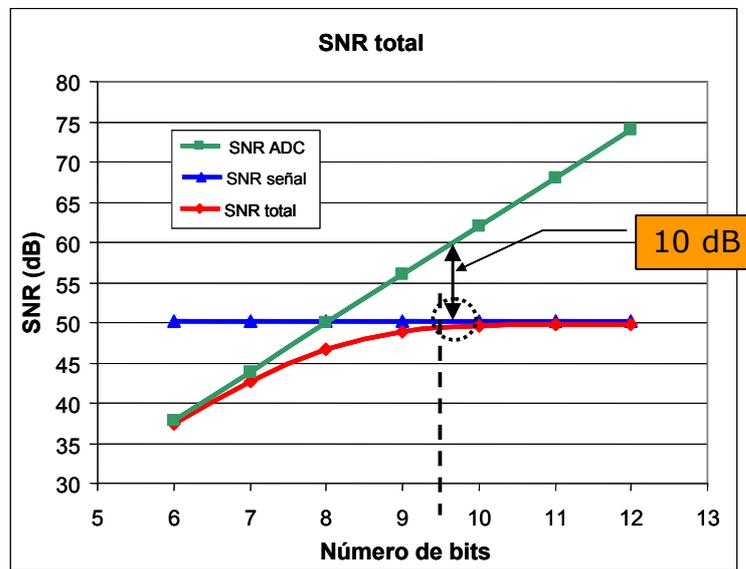


Figura 7.2. SNR total frente al número de bits (N) del ADC.

7.3. Resultados de la metodología de diseño de convertidores sigma-delta y pipeline

Los convertidores sigma-delta tienen una resolución elevada gracias a su complejidad, a pesar de ello son una solución muy adecuada para muchas aplicaciones. Los métodos de diseño más habituales se basan en modelos discretos de estructuras predefinidas que se implementan con circuitos de capacidades conmutadas. Estas estructuras persiguen alcanzar la máxima resolución con la sencillez y la estabilidad como dobles objetivos. En la Figura 7.3 se muestra la nueva estructura del modulador sigma-delta que se ha desarrollado y con la que es posible realizar un diseño partiendo de la función de transferencia que predetermina la resolución, es decir, la función NTF. En este sentido se han diseñado y simulado una serie de ejemplos demostrativos de las ventajas de la nueva metodología y, en

particular, se ha construido un prototipo de un modulador de segundo orden. Con la nueva metodología se consigue mejorar la ganancia de resolución con factores de sobremuestreo bajos. Después de validar los resultados de los estudios hechos mediante simulación se ha construido un prototipo cuyos resultados se presentan en la Figura 7.5.

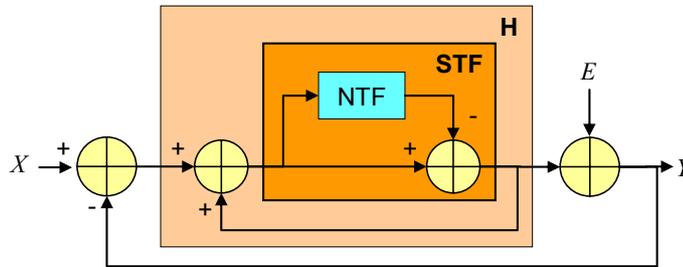


Figura 7.3. Estructura del modulador sigma-delta indicando las distintas funciones de transferencia que lo caracterizan.

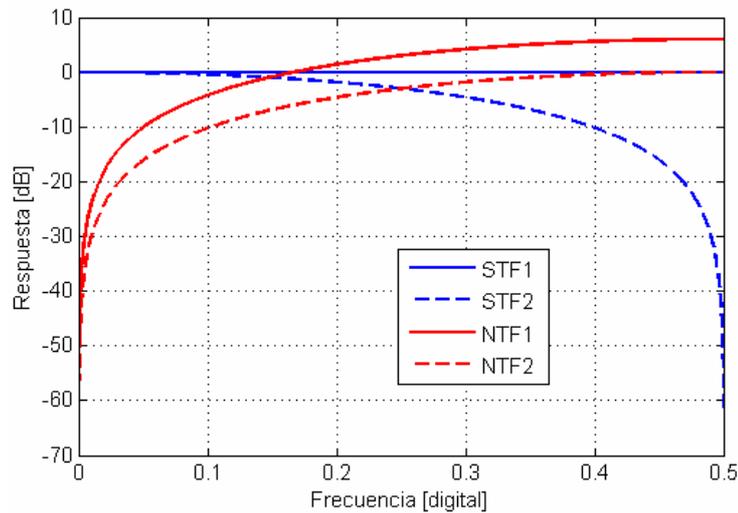


Figura 7.4. Respuesta en frecuencia de los modelos de primer orden de los moduladores sigma-delta, que corresponden a la estructura clásica (STF1 y NTF1) y a la nueva estructura (STF2 y NTF2).

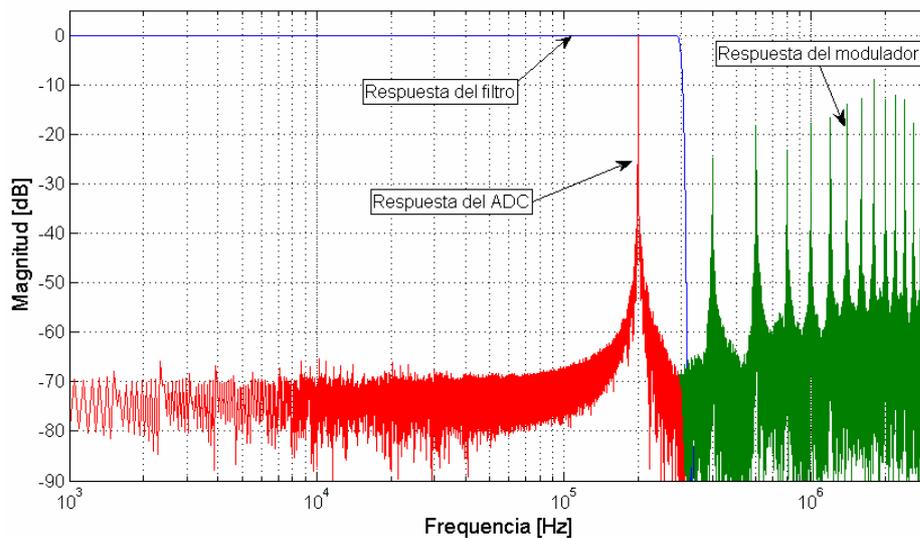


Figura 7.5. Respuesta en frecuencia del convertidor sigma-delta ($f_s = 6 \text{ MHz}$, $f_c = 300 \text{ kHz}$, $f_i = 200 \text{ kHz}$).

RESULTADOS OBTENIDOS

En cuanto a los convertidores A/D pipeline se ha analizado su estructura más sencilla desarrollando una metodología de diseño y simulación que incluye el método de sincronización de las etapas. Luego, se ha diseñado una variante de la estructura básica que se caracteriza por ser más sencillas las etapas e introducir menos ruido en la señal, puesto que al pasar por las sucesivas etapas no se amplifica el residuo y, por tanto, tampoco el ruido ni los efectos de las no linealidades. Los dos tipos de etapas correspondientes a los convertidores que se han simulado se muestran en la Figura 7.6 y los resultados, que se presentan en la Tabla 7.1, indican que con la nueva estructura (Figura 7.6.b) se consigue mejorar la resolución en medio bit.

Finalmente, en la Tabla 7.1 se presenta un resumen de los resultados obtenidos de los principales circuitos que se han probado.

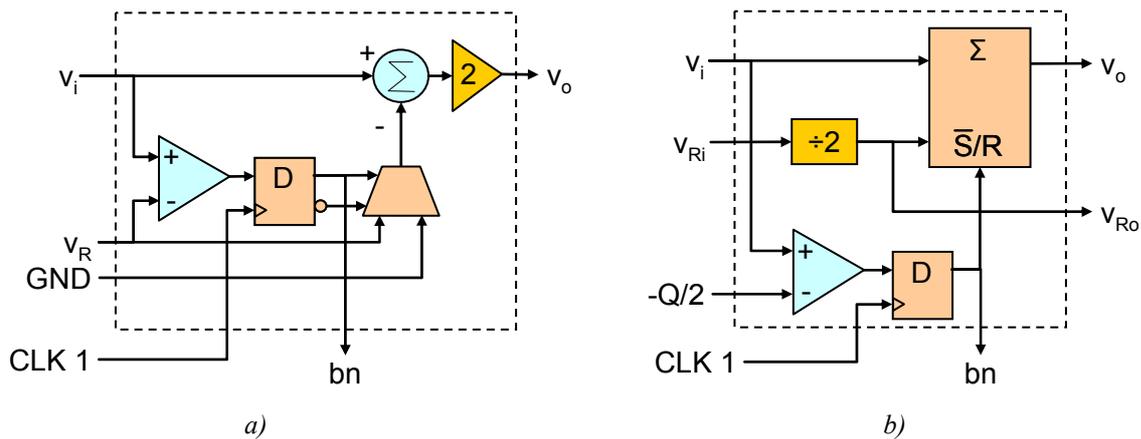


Figura 7.6. Alternativas presentadas para las etapas de los convertidores pipeline, a) estructura con amplificación del residuo (PP1), b) nueva estructura con atenuación de la referencia (PP2).

Tabla 7.1. Resumen de los resultados obtenidos de los distintos convertidores A/D.

Denominación	Tipo	Forma de implementación	Referencia cruzada	Característica de comparación	Resultado
Sigma-delta (SD1)	1er orden	Simulado	Figura 5.21	SFDR	48.4 dB
Sigma-delta (SD2)	2do orden	Simulado	Figura 5.23		54 dB
Sigma-delta (SD3)	3er orden	Simulado	Figura 5.24		57.4 dB
Sigma-delta (SD4)	2do orden	Construido	Figura 5.31		71 dB
Pipeline (PP1)	Básico 5 b	Simulado	Figura 4.13	ENOB	4.28 bits
Pipeline (PP2)	Nuevo 5 b	Simulado	Figura 5.3		4.89 bits

7.4. Resultados de la aplicación de la metodología docente para la enseñanza de los convertidores A/D *pipeline* y sigma-delta

Se ha desarrollado una metodología de enseñanza de los circuitos ADC que se ha aplicado a alumnos con el perfil de estudiante del segundo cuatrimestre de 5° curso de Ingeniero de Telecomunicaciones. La parte práctica de la metodología consiste en realizar una serie de simulaciones utilizando OrCAD PSpice y un simulador de VHDL para síntesis. Estos ejemplos de simulación prácticos fueron bien entendidos por los alumnos debido a sus características realistas.

Dada la cantidad y profundidad de los conceptos manejados, la relación entre el tiempo empleado en el curso, tanto para la presentación de la teoría como para realizar los ejercicios prácticos, y los resultados obtenidos ha sido satisfactoria.

La actitud mostrada por los estudiantes ha sido muy positiva porque han relacionado una gran cantidad de conocimientos y principios que habían estudiado en cursos anteriores y que son difíciles de entender.

Se han propuesto a los alumnos varios trabajos libres consistentes en hacer las simulaciones de los circuitos pero con ligeras modificaciones, de este modo se han asentado mejor los conocimientos. En particular han realizado las simulaciones del convertidor *pipeline* en versión de 4 bits y del sigma-delta en versión de 6 bits.

CAPÍTULO 8. CONCLUSIONES FINALES

8.1. Aportaciones de esta tesis

A través de esta tesis se han ordenado y estudiado las referencias bibliográficas más importantes sobre los circuitos convertidores *pipeline* y Σ - Δ (sigma-delta), y con este trabajo se ha realizado una introducción, desde un punto de vista del procesado de señal y de la tecnología electrónica, al proceso de conversión A/D con los circuitos que más se utilizan en la actualidad (y que presentan un aumento sustancial en su uso día a día). Se ha presentado: su modo de funcionamiento, su clasificación y sus características más importantes desde la perspectiva de las normativas más actuales, haciendo especial hincapié en los métodos de test de estos circuitos. Después de introducir los convertidores A/D se ha analizado cómo afecta el proceso de conversión a la calidad de la señal de entrada, representada por su SNR. En este tema se ha presentado un nuevo método para seleccionar la resolución y la frecuencia de muestreo de estos circuitos ADC basado en las características dinámicas de la señal a convertir.

Se ha estudiado y simulado una estructura básica de un convertidor *pipeline* desarrollando como aportación un método sencillo para sincronizar sus etapas de forma sistemática. Esta estructura, que se basa en amplificar el residuo de una etapa para adaptarla al rango de la siguiente, se ha comparado con una nueva estructura que evita esta amplificación y que presenta mejores prestaciones, pues consigue aumentar la resolución efectiva. Esta nueva estructura alternativa demuestra que, aunque la tecnología y base teórica de los convertidores modernos ha alcanzado una gran madurez, es necesario seguir investigando para mejorar los circuitos haciéndolos más sencillos y más eficientes. Para evaluar las dos estructuras de convertidores *pipeline* se ha utilizado, principalmente, el programa de simulación OrCAD PSpice porque permite la simulación de circuitos mixtos analógico-digitales y porque utiliza modelos de dispositivos reales, lo cual valida los resultados prácticos de simulación de una forma más realista.

En cuanto a los convertidores Σ - Δ , primero se ha presentado un estudio detallado del principio de funcionamiento de los moduladores teniendo en cuenta su evolución histórica, es decir, sus antecedentes, y se han presentado justificadamente sus principios teóricos, los tipos de estructuras más importantes, y se han simulado los modelos continuos y discretos utilizando el simulador mixto

CONCLUSIONES

OrCAD PSpice para su descripción mediante circuitos electrónicos y el programa Matlab y/o un compilador de lenguaje C para su descripción algorítmica. Después de realizar este estudio del modulador Σ - Δ , se ha completado con el estudio de las etapas de filtrado y diezmado, evaluando las ventajas e inconvenientes de los filtros de promediado y presentando un método para ecualizarlos mediante filtros FIR diseñados por el método de muestreo en frecuencia. En esta parte se han implementado mediante simulación dos filtros, uno de promediado descrito mediante el lenguaje de descripción hardware VHDL y otro de respuesta finita descrito mediante un algoritmo programado con Matlab.

Una vez se han analizado, simulado y evaluado las características de los convertidores Σ - Δ , se ha presentado un nuevo modelo que simplifica su diseño y mejora su respuesta en frecuencia. La definición matemática de este modelo, que parte del diseño de la función de transferencia deseada para el ruido y, por tanto, de la resolución que se desea, se ha deducido y comparado con las estructuras clásicas. Para ello, se han diseñado moduladores de distintos órdenes y se han evaluado sus resultados comparando sus respuestas en frecuencia obtenidas de las simulaciones, que se han realizado tanto para los modelos continuos como discretos. Finalmente, se ha construido un prototipo hardware del nuevo modelo del modulador de segundo orden. La parte analógica del circuito se ha implementando mediante componentes discretos y la parte digital mediante una FPGA. Luego, se han realizado diversos ensayos enviando los datos de conversión al computador, donde se analizaron mediante los algoritmos de procesado que se diseñaron para realizar el filtrado de dichos datos y el test del convertidor.

Estas aportaciones, que se han ido consiguiendo a lo largo de los últimos tres años, se han aplicado a la enseñanza, para lo cual se ha ido desarrollando en paralelo una metodología docente para incluir la enseñanza de los dos tipos de convertidores en los programas docentes de las especialidades de electrónica de las titulaciones de Ingeniería. Concretamente esta metodología se ha implantado como plan piloto en un tema práctico de la asignatura de 5º curso Laboratorio de Sensores y Acondicionadores de la E.T.S. de Ingenieros de Telecomunicación de la Universidad de Vigo. De este trabajo se han presentado los resultados de evaluación obtenidos en los cursos 2006/2007 y 2007/2008.

8.2. Sugerencias para futuros estudios y desarrollos

En cuanto a los convertidores multietapa, y en concreto los *pipeline*, los trabajos futuros pueden ir en la línea de modelar los componentes básicos de las etapas de los convertidores *pipeline* caracterizando las no linealidades y el ruido de forma que se puedan comparar las ventajas e inconvenientes de las distintas estructuras. Desarrollar una sola etapa acoplada a una FPGA implementando con ella un convertidor cíclico [139] de forma que se pueda evaluar su rendimiento, y así, evaluando el comportamiento de una sola etapa, extrapolar los resultados a un convertidor multietapa y probar distintos algoritmos de corrección del error en la FPGA de forma flexible.

Las propuestas para futuros trabajos en relación con los convertidores sigma-delta pueden seguir las dos líneas siguientes:

1. Aplicar los estudios realizados sobre los modelos continuos de los convertidores sigma-delta a los discretos implementando la metodología presentada mediante circuitos de capacidades conmutadas.
2. Diseñar en VHDL los filtros digitales utilizados, que se programaron mediante Matlab, de forma que sean parametrizables y que se puedan implementar en una FPGA.

Para seguir las líneas indicadas serían necesarias las siguientes tareas:

1. Añadir a los diseños de la FPGA el módulo de comunicaciones con el computador para, por un lado enviar los parámetros a la FPGA y, por otro, recibir los datos resultados de los procesos bajo prueba.
2. Desarrollar un modulador de segundo orden con la estructura básica y comparar los resultados con los del modulador desarrollado con la nueva metodología.
3. Desarrollar un modulador paso-banda con la nueva metodología y evaluar los resultados frente a otras alternativas.
4. Automatizar el proceso de diseñar un circuito que implemente la función NTF, la STF o la H mediante síntesis directa. Este circuito, que debe ser reconfigurable, será conveniente realizarlo mediante capacidades conmutadas, para poder variar las frecuencias de los polos y ceros, y su estructura. Se puede comenzar por evaluar la utilización de un filtro programable

CONCLUSIONES

comercial. Previamente se realizarán simulaciones para evaluar la topología de filtro más idónea.

Los trabajos futuros propuestos están en la línea de continuar estudiando distintas alternativas implementando circuitos sencillos que permitan evaluarlas. Después, con las soluciones que parezcan más ventajosas se debiera tender a implementarlas mediante un circuito integrado ASIC con tecnología CMOS siguiendo algún programa de los que los fabricantes tienen para universidades.

En cuanto al test de los convertidores se propone automatizarlo mediante un sistema basado en un PC, en una FPGA y en la instrumentación de medida y generación necesaria. El PC tiene la misión de ejecutar los algoritmos de test y de controlar los instrumentos y la placa de la FPGA. De este modo, se puede modificar el tipo y características del filtro que se implemente en la FPGA, los parámetros de las señales de referencia, etc. Con este sistema se propone generar unos resultados gráficos, en el PC, tridimensionales, en los que se presenten los parámetros que indican la calidad del convertidor (SFDR, THD, SINAD) frente al rango de entrada (FSR), el orden (L), la relación de sobremuestreo (OSR), el número de etapas, etc.

Para facilitar la implementación de dicho sistema se propone reutilizar la plataforma de instrumentación reconfigurable presentada en [22] con el hardware actualizado que se presentó en [23].

BIBLIOGRAFÍA

- [1] W. Kester, *The Data Conversion Handbook*. Analog Devices, Inc., Elsevier, 2005.
- [2] F. Maloberti, *Data Converters*. Springer, 2008.
- [3] D.A. Rauth y V.T. Randal, "Analog-to-Digital Conversion," *IEEE Instrumentation and Measurement Magazine*, vol. 8, no. 4, pp. 44-55, Octubre de 2005.
- [4] Analog Device, Inc., *10-bit, 65/80/105 MSPS, 3 V A/D Converter* [en línea]. Hoja de características del convertidor AD9215, 2004 [fecha de consulta: 20 de mayo de 2008]. Disponible en: http://www.analog.com/UploadedFiles/Data_Sheets/AD9215.pdf
- [5] R.P. Stone, C.W. Mueller y W.M. Webster, "A Time-Sampling and Amplitude-Quantizing Tube," *Proceedings of the IRE*, vol. 43, no. 8, pp. 960-966, Agosto de 1955.
- [6] C. Quintáns, M. J. Moure, E. Mandado y M.A. Castro, "A Practical Example to Understand Pipeline A/D Converters Performance Using Software Tools," *Proceedings of the Industrial Electronics Conference, IECON*, pp. 2958-2962, Octubre de 2006.
- [7] P. Arpaia, P. Daponte y S. Rapuano, "A State of the Art on ADC Modeling," *Computer Standards & Interfaces*, Elsevier, vol. 26, 2003, pp. 31-42.
- [8] D. Jarman, *A Brief Introduction to Sigma Delta Conversion* [en línea]. Intersil, Inc., Nota de aplicación 9504, 1995 [fecha de consulta: 27 de marzo de 2008]. Disponible en: <http://www.intersil.com/data/an/an9504.pdf>
- [9] B.M. Nasir, "Sigma Delta and Oversampling Strategies," *IEE Proceedings of the Colloquium on Oversampling and Sigma-Delta Strategies for DSP*, pp. 5/1-5/7, 1995.
- [10] S.W. Smith, *The Scientist and Engineer's Guide to Digital Signal Processing*. California Technical Publishing, San Diego, California, pp. 285-296, 1997.
- [11] Analog Device, Inc., *Fundamentals of Sampled Data Systems* [en línea]. Nota de aplicación AN-282, 2002 [fecha de consulta: 8 de marzo de 2008]. Disponible en: <http://www.analog.com/en/DCcList/0,3090,760%255F%255F43,00.html>
- [12] Keithley Instruments, Inc., *Data Acquisition and Control Handbook*, 2001.

BIBLIOGRAFÍA

- [13] D.A. Bell, *Electronic Instrumentation and Measurements*. Second edition, Prentice Hall & Technology, 1994.
- [14] C. Quintáns y J. Doval (director), *Sistema de Adquisición para el DSP 320C31*. Proyecto final de carrera de la E.U.I.T. Industrial de la Universidad de Vigo, 1997.
- [15] M. Tooley, *PC-based Instrumentation and Control*. Newnes, an imprint of Butterworth-Heinemann, Second Edition, 1995.
- [16] G. Silverman y H Silver, *Modern Instrumentation: A Computer Approach*. Institute of Physics Publishing Bristol and Philadelphia, 1995.
- [17] J.J. Rodríguez-Andina, M.J. Moure, and M.D. Valdés, “Features, Design Tools, and Application Domains of FPGAs,” *IEEE Transactions on Industrial Electronics*, vol. 54, no. 4, pp. 1810-1823, Agosto de 2007.
- [18] C. Quintáns, M.A. Castro (director) y M.J. Moure (codirectora), *Sistema de Adquisición de Señales Analógicas de Alta Frecuencia para Procesadores Digitales Basados en FPGA*. Proyecto final de carrera de la E.T.S.I. Industriales de la Universidad Nacional de Educación a Distancia, 2005.
- [19] C. Quintáns, M.D. Valdés, M.J. Moure y E. Mandado, “Sistema de Desarrollo de Aplicaciones Electrónicas Basado en Dispositivos Lógicos Configurables y en el Bus Serie Universal,” Patente P200301238, España, 2003.
- [20] M.D. Valdés, M.J. Moure, C. Quintáns y E. Mandado, “A Data Acquisition Reconfigurable Coprocessor for Virtual Instrumentation Applications,” *Field-Programmable Logic and Applications. Lectures Notes in Computer Science*, Springer, vol. 2778, pp. 1107-1110, 2003.
- [21] J. Yáñez, D. Quintana, C. Quintáns, J. Fariña y J.J.R. Andina, “FPGA-based System for the Education in Data Acquisition and Signal Generation,” *Proceedings of the Industrial Electronics Conference, IECON*, pp. 2168-2173, 2005.
- [22] C. Quintáns, M.J. Moure, M.D. Valdés y E. Mandado, “A Virtual Instrumentation Laboratory Based on a Reconfigurable Coprocessor,” *IEEE Transactions on Instrumentation and Measurement*, vol. 55, no. 2, pp. 635-645, Abril de 2006.

- [23] C. Quintáns y E. Mandado, “Reconfigurable Platform to Implement Electronic Instrumentation,” *Proceedings of the IEEE Instrumentation and Measurement Technology Conference*, IMTC, Mayo de 2007.
- [24] B.M. Gordon, “Linear Electronic Analog/Digital Conversion Architectures, Their Origins, Parameters, Limitations, and Applications,” *IEEE Transactions on Circuits and Systems*, vol. 25, no. 7, pp. 391-418, Julio de 1978.
- [25] A. Gersho, “Principles of Quantization,” *IEEE Transactions on Circuits and Systems*, vol. 25, no. 7, pp. 427-436, Julio de 1978.
- [26] S. Rapuano, P. Daponte, E. Balestrieri, L. De Vito, S.J. Tilden, S. Max y J.J. Blair, “ADC Parameters and Characteristics,” *IEEE Instrumentation and Measurement Magazine*, vol. 8, no. 5, pp. 44-54, Diciembre de 2005.
- [27] Dallas Semiconductor, Inc., MAXIM, *A Simple ADC Comparison Matrix* [en línea]. Nota de aplicación 2094, 2003 [fecha de consulta: 25 de marzo de 2008]. Disponible en: <http://www.maxim-ic.com/an2094>
- [28] R. Schreier y G.C. Temes, *Understanding Delta-Sigma Data Converters*. IEEE Press, John Wiley & Sons, Inc., 2005.
- [29] Dallas Semiconductor, Inc., MAXIM, *Understanding Pipelined ADCs* [en línea]. Nota de aplicación 1023, 2001 [fecha de consulta: 25 de marzo de 2008]. Disponible en: <http://pdfserv.maxim-ic.com/en/an/AN1023.pdf>
- [30] Dallas Semiconductor, Inc., MAXIM, *Pipeline ADCs Come of Age* [en línea]. Nota de aplicación 634, 2000 [fecha de consulta: 25 de marzo de 2008]. Disponible en: <http://pdfserv.maxim-ic.com/en/an/AN634.pdf>
- [31] Dallas Semiconductor, Inc., MAXIM, *Understanding Flash ADCs* [en línea]. Nota de Aplicación 810, 2001 [fecha de consulta: 25 de marzo de 2008]. Disponible en: <http://pdfserv.maxim-ic.com/en/an/AN810.pdf>
- [32] Tektronix, Inc., *XYZ Oscilloscopes* [en línea]. Nota de Aplicación, 2001 [fecha de consulta: 25 de marzo de 2008]. Disponible en: http://www.tek.com/Measurement/App_Notes/XYZs/03W_8605_2.pdf

BIBLIOGRAFÍA

- [33] F. Maloberti, "High-Speed Data Converters for Communication Systems," *IEEE Circuits and Systems Magazine*, vol. 1, no. 1, pp. 26-36, 2001.
- [34] N.C. Gray, *Evaluating and Using ADCs* [en línea]. National Semiconductors, Inc., Seminario, [fecha de consulta: 25 de marzo de 2008]. Disponible en:
<http://www.national.com/appinfo/adc/files/Web-pres2001.pdf>
- [35] R. Unbehauen y A. Cichocki, *MOS Switched-Capacitors and Continuous-Time Integrated Circuits and Systems*. A. Fetweis, Berlin: Springer-Verlag, pp. 555-627, 1989.
- [36] Dallas Semiconductor, Inc., MAXIM, *An introduction to Jitter in Communications Systems* [en línea]. Nota de Aplicación 1916, 2003 [fecha de consulta: 25 de marzo de 2008]. Disponible en: <http://pdfserv.maxim-ic.com/en/an/AN1916.pdf>
- [37] Y. Langard, J.L. Balat y J. Durand, "An Improved Method of ADC Jitter Measurement," *Proceedings of the IEEE International Test Conference*, pp. 763-770, 1994.
- [38] M.F. Wagdy y S.S. Awad, "Effect of Sampling Jitter on Some Sine Wave Measurements," *IEEE Transactions on Instrumentation and Measurement*, vol. 39, no. 1, pp. 86-89, Febrero de 1990.
- [39] S.S. Awad y M.F. Wagdy, "More on Jitter Effects on Sinewave Measurements," *IEEE Transactions on Instrumentation and Measurement*, vol. 40, no. 3, pp. 549-552, Junio de 1991.
- [40] IEEE Std. 1241-2000, *IEEE Standard for Terminology and Test Methods for Analog-to-Digital Converters*, 2000.
- [41] IEC 62008-2006, *Característica de Funcionamiento y Métodos de Calibración para los Sistemas de Adquisición de Datos Digitales y su Software Asociado*. International Electrotechnical Commission, 3 rue de Varembe, Geneva, Switzerland, 2006.
- [42] IEC 60748-4-3-2006, *Interface Integrated Circuits, Semiconductor Devices-Integrated Circuits*. International Electrotechnical Commission, 3 rue de Varembe, Geneva, Switzerland, 2006.
- [43] IEEE Std. 1057- R2001, *IEEE Standard for Digitizing Waveform Recorders*, 2001.

- [44] E. Thomas, J.J. Blair, S. Rapuano, P. Daponte, E. Balestrieri, L. De Vito, S. Max y S.J. Tilden, “ADC Testing,” *IEEE Instrumentation and Measurement Magazine*, vol. 9, no. 2, pp. 39-49, Abril de 2006.
- [45] J. Blair, “Histogram Measurement of ADC Nonlinearities Using Sine Waves,” *IEEE Transactions on Instrumentation and Measurement*, vol. 43, no. 3, pp. 373–383, Junio de 1994.
- [46] V. Haasz, “Testing of high-resolution/middle-speed A/D converters and modules—problems and ways of solving them,” *Computer Standards & Interfaces*. Elsevier, vol. 28, pp. 194-203, Febrero de 2005.
- [47] K-S. Lee, S. kwon y F. Maloberti, “A Power-Efficient Two-Channel Time-Interleaved $\Sigma\Delta$ Modulator for Broadband Applications,” *IEEE Journal of Solid-State Circuits*, vol. 42, no. 6, pp. 1206-1215, Junio de 2007.
- [48] A.C. Serra, F. Alegria, R. Martins y M. F da Silva, “Analog-to-Digital Converter Testing – New Proposals,” *Computer Standard & Interfaces*, Elsevier, vol. 26, pp. 3-13, 2003.
- [49] J.R. Naylor, “Testing Digital/Analog and Analog/Digital Converters,” *IEEE Transactions on Circuits and Systems*, vol. 25, no. 7, pp. 526-538, Julio de 1978.
- [50] D.W. Braudaway, “Uncertainty Specification for Data Acquisition (DAQ) Devices,” *IEEE Transactions on Instrumentation and Measurement*, vol. 55, no. 1, pp. 74-78, Febrero de 2006.
- [51] F. Attivissimo, N. Giaquinto y M. Savino, “Worst-Case Uncertainty Measurement in ADC-Based Instruments,” *Computer Standard & Interfaces*, Elsevier, vol. 29, no. 1, pp. 5-10, Enero de 2006.
- [52] Centro Español de Metrología, *Guía para la Expresión de la Incertidumbre de Medida* (versión española). Ministerio de Fomento, 1998.
- [53] W. Jenkins y M. Desai, “The Discrete Frequency Fourier Transform,” *IEEE Transactions on Circuits and Systems*, vol. 33, no. 7, pp. 732-734, Julio de 1986.
- [54] J.O. Smith, *Mathematics of the Discrete Fourier Transform (DFT)*. W3K Publishing. Second Edition, 2003.

BIBLIOGRAFÍA

- [55] M. Bossche, J. Schoukens y J. Renneboog, “Dynamic Testing and Diagnostics of A/D Converters,” *IEEE Transactions on Circuits and Systems*, vol. 33, pp. 775-785, Agosto de 1986.
- [56] Y.C. Jenq, “Measuring Harmonic Distortion and Noise Floor of an A/D Converter Using Spectral Averaging,” *Proceedings of the 5th IEEE Instrumentation and Measurement Technology Conference, IMTC*, pp. 212-215, 1988.
- [57] O.M. Solomon “The Use of DFT Windows in Signal-to-Noise Ratio and Harmonic Distortion Computations,” *Proceedings of the IEEE Instrumentation and Measurement Technology Conference*, pp. 103–108, 1993.
- [58] M.F. Wagdy, “Diagnosing ADC Nonlinearity at the Bit Level,” *IEEE Transactions on Instrumentation and Measurement*, vol. 38, no. 6, pp. 1139-1141, Diciembre de 1989.
- [59] M.F. Wagdy y X. Qiong, “Comparative ADC Performance Evaluation Using a New Emulation Model for Flash ADC Architectures,” *Proceedings of the 37th Midwest Symposium on Circuits and Systems*, vol. 2, pp. 1159-1163, 1994.
- [60] Analog Device, Inc., *8-bit, 40/80/100 MSPS Dual A/D Converter* [en línea]. Hoja de características del convertidor AD9288, 2002 [fecha de consulta: 25 de marzo de 2008]. Disponible en: http://www.analog.com/UploadedFiles/Data_Sheets/AD9288.pdf
- [61] B. Widrow, I. Kollár y M.-Ch. Liu, “Statistical Theory of Quantization,” *IEEE Transactions on Instrumentation and Measurement*, vol. 45, no. 2, pp. 353-361, Abril de 1996.
- [62] C. Quintáns, “Selecting the Optimal Resolution and Conversion Frequency for A/D and D/A Converters,” *Proceedings of the IEEE Instrumentation and Measurement Technology Conference, IMTC*, Mayo de 2007.
- [63] S. Park, “Principles of Sigma-Delta Modulation for Analog-to Digital Converters,” *Motorola DSP Operations Application Report, APDR8/D* [en línea]. Motorola, Inc., 1990 [fecha de consulta: 25 de marzo de 2008]. Disponible en: <http://www.numerix-dsp.com/appsnotes/APR8-sigma-delta.pdf>
- [64] R.M. Gray, “Oversampled Sigma-Delta Modulation,” *IEEE Transactions on Communications*, vol. 35, no. 5, pp. 481-489, Mayo de 1987.

- [65] G.I. Bourdopoulos, A. Pnevmatikakis, V. Anastassopoulos y T.L. Deliyannis, *Delta-Sigma Modulators: Modeling, Design and Applications*. Imperial Collage Press, 2006.
- [66] M.B. Yeary, W. Zhang, J.Q. Trelewicz, Y. Zhai y B. McGuire, "Theory and Implementation of a Computationally Efficient Decimation Filter for Power-Aware Embedded Systems," *IEEE Transactions on Instrumentation and Measurement*, vol. 55, no. 5, pp. 1839-1849, Octubre de 2006.
- [67] H.A. Spang y P.M. Schultheiss, "Reduction of Quantizing Noise by Use of Feedback," *IRE Transactions On Communications Systems*, pp. 373-380, 1962.
- [68] L.S. Golding y P.M. Schultheiss, "Study of an Adaptive Quantizer," *Proceedings of the IEEE*, vol. 55, no. 3, pp. 293-297, 1967.
- [69] S.K. Tewksbury y R.W. Hallock, "Oversampled, Linear Predictive and Noise-Shaping coders of Order $N > 1$," *IEEE Transactions on Circuits and Systems*, vol. 25, no. 7, pp. 436-447, 1978.
- [70] Analog Devices, Inc., *Isolated Sigma-Delta Modulator* [en línea]. Hoja de características del AD7400, 2006 [fecha de consulta: 25 de marzo de 2008]. Disponible en: http://www.analog.com/UploadedFiles/Data_Sheets/AD7400.pdf
- [71] C.C. Cutler, "Transmission Systems Employing Quantization," U.S. Patent no. 2,927,962. 1960.
- [72] R.J. Van De Plassche, "A Sigma-Delta Modulator as an A/D Converter," *IEEE Transactions on Circuits and Systems*, vol. 25, pp. 510-514, Julio de 1978.
- [73] Analog Devices, Inc., *Sigma-Delta ADCs and DACs* [en línea]. Nota de aplicación 283 [fecha de consulta: 25 de marzo de 2008]. Disponible en: http://www.analog.com/UploadedFiles/Application_Notes/292524291525717245054923680458171AN283.pdf
- [74] S.J. Orfanidis, *Introduction to Signal Processing*. Prentice Hall International, Inc., 1997.
- [75] B.E. Boser y B.A. Wooley, "Quantization Error Spectrum of Sigma-Delta Modulators," *Proceedings of the IEEE International Symposium on Circuits and Systems, ISCAS*, pp. 2331-2334, 1998.

BIBLIOGRAFÍA

- [76] J.C. Candy, "A Use of Double Integration in Sigma Delta Modulation," *IEEE Transactions on Communications*, vol. 33, pp. 249-258, Marzo de 1985.
- [77] J.C. Candy y G.C. Temes, "Oversampling Methods for Data Conversion," *IEEE Proceedings of the Pacific Rim Conference on Communications, Computers and Signal Processing*, pp. 498-502, 1991.
- [78] R. Koch, B. Heise, F. Eckbauer, E. Engelhardt, J.A. Fisher y F. Parzefall, "A 12-bit Sigma-Delta Analog-to-Digital Converter with a 15-MHz Clock Rate," *IEEE Journal of Solid-State Circuits*, vol. 21, no. 6, pp. 1003-1010, Diciembre de 1986.
- [79] J. Lota, M. Al-jabani e I. Kale, "Stability Analysis of Higher-Order Delta-Sigma Modulators for Sinusoidal Inputs," *Proceedings of the IEEE Instrumentation and Measurement Technology Conference, IMTC, 2007*.
- [80] S. Park, Motorola, Inc., "A Real-Time Implementation of Half-Band Filters to Obtain 18-20 bit Resolution from the DSP56ADC16," *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, vol. 2, pp. 989-992, Abril de 1990.
- [81] E.B. Hogenauer, "An Economical Class of Digital Filters for Decimation and Interpolation," *IEEE Transactions on Acoustics, Speech, and Processing*, vol. 29, no. 2, pp. 155-162, Abril de 1981.
- [82] S. Chu y C.S. Burrus, "Multirate Filters Designs Using Comb Filters," *IEEE Transactions on Circuits and Systems*, vol. 31, pp. 913-924, Noviembre de 1984.
- [83] S. Toscher, T. Reinemann, R. Kasper y M. Hartmann, "A Reconfigurable Delta-Sigma ADC," *Proceedings of the International Symposium of Industrial Electronics, ISIE*, pp. 495-499, Junio de 2006.
- [84] C. Dick y F. Harris, "FPGA Signal Processing Using Sigma-Delta Modulation," *IEEE Signal Processing Magazine*, pp. 20-35, Enero de 2000.
- [85] L.L. Presti, "Efficient Modified-Sinc Filters for Sigma-Delta A/D Converters," *IEEE Transactions on Circuits and Systems*, vol. 47, pp. 1204-1213, Noviembre de 2000.
- [86] S.S. Abeysekera, "Recursive Laguerre and Kalman Filters as Efficient Full-Rate Sigma-Delta (Σ - Δ) Demodulators," *Signal Processing*, Elsevier, vol. 87, pp. 417-431, Julio de 2006.

- [87] E. Dijkstra, O. Nys, C. Piquet y M. Degrauwe, "On the Use of Modulo Arithmetic Comb Filters in Sigma-Delta Modulators," *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, vol. 4, pp. 2001-2004, Abril de 1988.
- [88] S. Park y W. Chen, Motorola, Inc., "Multi-Stage IIR Decimation Filter Design Technique for High Resolution Sigma-Delta A/D Converters," *Proceedings of the IEEE Instrumentation and Measurement Technology Conference, IMTC*, pp. 561-566, Mayo de 1992.
- [89] J.C. Candy, "Decimation for Sigma Delta Modulation," *IEEE Transactions on Communications*, vol. 34, pp. 72-76, Enero de 1986.
- [90] J.G. Proakis y D.G. Manolakis, *Tratamiento Digital de Señales*. Prentice Hall International, Inc., pp. 639-643, 1998.
- [91] E.C. Ifeachor y B.W. Jervis, *Digital Signal Processing*. Addison-Wesley, pp. 317-344, 1993.
- [92] C. Quintáns, *Simulación de Circuitos Electrónicos con OrCAD 16 DEMO*. Marcombo, 2008.
- [93] IEEE Std. 1076-2002, *VHDL Language Reference Manual*.
- [94] J. Silva, U. Moon y G.C. Temes, "Low-Distortion Delta-Sigma Topologies for MASH Architectures," *Proceedings of the IEEE International Symposium on Circuits and Systems, ISCAS*, pp. 1144-1147, 2004.
- [95] D.H. Horrocks, "A Second-Order Oversampled Sigma-Delta Modulator for BandPass Signals," *Proceedings of the IEEE International Symposium on circuits and systems, ISCAS*, pp. 1653-1656, 1990.
- [96] A. Bosi, A. Panigada, G. Cesura y R. Castellano, "An 80 MHz 4x Oversampled Cascade $\Sigma\Delta$ -Pipelined ADC with 75 dB DR and 87 dB SFDR," *IEEE Proceedings of the International Solid-State Circuits Conference*, pp. 174-175, 2005.
- [97] S. Marabelli, A. Fornasari, P. Malcovati y F. Maloberti, "An 80 MHz, 14-bit Bandpass Mash Sigma-Delta/Pipeline A/D Converter with 5 MHz Bandwidth for Third Generation Mobile Communication Systems," *Measurement*, Elsevier, vol. 37, pp. 320-327, 2005.
- [98] J. Yu, F. Maloberti, "A Low-Power Multi-Bit $\Sigma\Delta$ Modulator in 90-nm Digital CMOS Without DEM," *IEEE Journal of Solid-State Circuits*, vol. 40, no. 12, pp. 2428-2436, Diciembre de 2005.

BIBLIOGRAFÍA

- [99] V. Majidzadeh y O. Shoaie, "A Reduce-Sample-Rate Sigma-Delta-Pipeline ADC Architecture for High-Speed High-Resolution Applications," *The Institute of Electronics, Information and Communication Engineers Transactions on Electronics*, IEICE, vol. E89-C, no. 6, Junio de 2006.
- [100] J. Haze, R. Vrba, L. Fucek, J. Forejtek, P. Zavoral, M. Paulik y L. Michaeli, "BandPass Sigma-Delta Modulator for Capacitive Pressure Sensor," *Proceedings of the Instrumentation and Measurement Technology Conference*, IMTC, 2007.
- [101] L.S. Palma, A. Oliveira, R.C.S. Freire y A.B. Fontes, "Thermal Σ - Δ Modulator: Performance Analysis," *Proceedings of the XVIII IMEKO World Congress*, Septiembre de 2006.
- [102] N.M. Mohan, B. George y V.J. Kumar, "A Sigma-Delta Resistance to Digital Converter Suitable for Differential Resistive Sensors," *Proceedings of the IEEE International Instrumentation and Measurement Technology Conference*, Mayo de 2008.
- [103] J. Acero, D. Navarro, L.A. Barragán, I. Garde, J.I. Artigas y J.M. Burdío, "FPGA-Based Power Measuring for Induction Heating Appliances Using Sigma-Delta A/D Conversion," *IEEE Transactions on Industrial Electronics*, vol. 54, no. 4, pp. 1843-1852, Agosto de 2007.
- [104] M. Ortmanns y F. Gerfers, *Continuous-Time Sigma-Delta A/D Conversion: Fundamentals, Performance Limits and Robust Implementation*. Springer Series in Advanced Microelectronics, Springer, 2006.
- [105] M.A. Al-Alaoui y R. Ferzli, "An Enhanced First-Order Sigma-Delta Modulator with a Controllable Signal-to-Noise Ratio," *IEEE Transactions on Circuits and Systems*, vol. 53, no. 3, pp. 634-643, Marzo de 2006.
- [106] K. Poulton, K.L. Knudsen *et. al*, "A 6-b, 4 GSa/s GaAs HBT ADC," *IEEE Journal of Solid-State Circuits*, vol 30, no. 10, pp. 1109-1118, Octubre de 1995.
- [107] S.H. Lewis y P.R. Gray, "A Pipelined 5-Msample/s 9-bit Analog-to-Digital Converter," in *IEEE Journal of Solid-State Circuits*, vol. 22, no. 6, pp. 954-961, Diciembre de 1987.
- [108] K. Sone, N. Nakadal, Y. Nishida, M. Ishida, Y. Sekine y M. Yotsuyanagi, "A 10-bits 100-MSamples/s Pipelined Subranging BICMOS ADC," *IEEE Proceedings of the ISSCC*, pp. 66-67, 1993.

- [109] D. Miyazaki, M. Furuta y S. Kawahito, "A 75 mW 10 bits 120 MSample/s Parallel Pipeline ADC," *Proceedings of the 29th European Solid-State Circuits Conference, ESSCIRC*, pp. 719-722, Septiembre de 2003.
- [110] H.P. Le, A. Zayegh y J. Singh, "A 12-Bit High Performance Low Cost Pipeline ADC," *IEEE Proceedings of the ICECS*, pp. 471-474, 2003.
- [111] S.M. Yoo, J.B. Park, S.H. Lee y U.K. Moon, "A 2.5-V 10b 120-MSample/s CMOS Pipeline ADC Based on Merged-Capacitor switching," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 51, no. 5, pp. 269-275, Mayo de 2004.
- [112] K. Nagaraj, H. Fetterman, J. Anidjar, S. Lewis, y R. Renninger, "A 250-mW, 8-b, 52-Msamples/s Parallel-Pipelined A/D Converter with Reduced Number of Amplifiers", *IEEE Journal of Solid State Circuits*, vol. 32, pp. 312-320, Marzo de 1997.
- [113] B.M. Min, P. Kim, F.W. Bowman, D.M. Boisvert y A.J. Aude, "A 69-mW 10-bits 80-MSamples/s Pipelined CMOS ADC," *IEEE Journal of Solid States Circuits*, vol. 38, no. 12, pp. 2031-2039, Diciembre de 2003.
- [114] K. Poulton, *Building the World's Fastest ADC – in CMOS* [en línea]. Agilent Technologies, Inc., Junio de 2005 [fecha de consulta: 27 de marzo de 2008]. Disponible en: http://poulton.net/papers.public/2005_ims_adc.pdf
- [115] I. Galton, "Digital Cancellation of D/A Converter Noise in Pipelined A/D Converters," *IEEE Transactions on Circuits and Systems*, vol. 47, no. 3, pp. 185-196, Marzo de 2000.
- [116] S.K. Tewksbury, F.C. Meyer, D.C. Rollenhagen, H. Schoenwetter y T.M. Souders, "Terminology Related to the Performance of S/H, A/D, and D/A Circuits," *IEEE Transactions on Circuits and Systems*, vol. 25, no. 7, pp. 419-426, Julio de 1978.
- [117] C. Quintáns, J.M. Lago, L.M. Menéndez y E. Mandado, "Plataforma Reconfigurable para Facilitar el Desarrollo de Sistemas Electrónicos Complejos," *Actas del Seminario Anual de Automática y Electrónica Industrial*, Gijón, España, pp. 649-652, Septiembre de 2006.
- [118] Altera Corporation. *Design Debugging Using the SignalTap II Embedded Logic Analyzer* [en línea]. Octubre de 2007 [fecha de consulta: 27 de marzo de 2008]. Disponible en: http://www.altera.com/literature/hb/qts/qts_qii53009.pdf
- [119] IEEE Std. 1149.1-2001, *IEEE Standard Test Access Port and Boundary-Scan Architecture*.

BIBLIOGRAFÍA

- [120] Q. Li, D. Betsymccoach, H. Swaminathan y J. Tang, "Development of an Instrument to Measure Perspectives of Engineering Education Among College Students," *Journal of Engineering Education*, vol. 97, no. 1, pp. 47-56, Enero de 2008.
- [121] J. Bransford, "Preparing People for Rapidly Changing Environments," *Journal of Engineering Education*, vol. 96, no. 1, pp. 1-3, Enero de 2007.
- [122] J. Margalef, L. Vicent, X. Senmartí, G. Bou y J. Anguera, "The Teaching of Electronics Engineering: Guidelines for the Design of Training In and Out of the Classroom Setting," *Proceedings of the ASEE/IEEE Frontiers in Education Conference, FIE*, 2007.
- [123] C.J. Atman, R.S. Adams, M.E. Cardella, J. Turns, S. Mosborg, J. Saleem, "Engineering Design Processes: A Comparison of Students and Expert Practitioners," *Journal of Engineering Education*, vol. 96, no. 4, pp. 359-379, Octubre de 2007.
- [124] I. Plaza y C.T. Medrano, "Continuous Improvement in Electronic Engineering Education," *IEEE Transactions on Education*, vol. 50, no. 3, pp. 259-269, Agosto de 2007.
- [125] C. Quintáns, M.D. Valdes, M.J. Moure, L.F. Ferreira y E. Mandado, "Digital Electronics Learning System Based on FPGA Applications," *Proceedings of the ASEE/IEEE Frontiers in Education Conference, FIE*, 2005.
- [126] I. Chamas y M.A. El Nokali, "Automated PSpice Simulation as an Effective Design Tool in Teaching Power Electronics," *IEEE Transactions on Education*, vol. 47, no. 3, pp. 415-421, Agosto de 2007.
- [127] A. Moscovici, *High Speed A/D Converters, Understanding Data Converters Through SPICE*. Kluwer Academic Publishers, 2003.
- [128] S. Acha, M. Castro y otros, *Electrónica Digital: Introducción a la Lógica Digital. Teoría, Problemas y Simulación*. Ed. RA-MA, 2002.
- [129] L. Yao, M. Steyaert y W. Sansen, *Low-Power Low-Voltage Sigma-Delta Modulators in Nanometer CMOS*. Springer, 2006.
- [130] J. Macías-Guarasa, J.M. Montero, R. San-Segundo, A. Araujo y O. Nieto-Taladriz, "A Project-Based Learning Approach to Design Electronic Systems Curricula," *IEEE Transactions on Education*, vol. 49, no. 3, pp. 389-397, Agosto de 2006.

- [131] L.R.J. Costa, M. Honkala y A. Lehtovuori, "Applying the Problem-Based Learning Approach to Teach Elementary Circuit Analysis," *IEEE Transactions on Education*, vol. 50, no. 1, pp. 41-48, Febrero de 2007.
- [132] Wikimedia Foundation, Inc., *Mapa Conceptual* [en línea]. 22 de Mayo de 2008 [fecha de consulta: 25 de Mayo de 2008]. Disponible en:http://es.wikipedia.org/wiki/Mapa_conceptual
- [133] T. Williams, "How Do Organizations Learn Lessons From Projects—And Do They?," *IEEE Transactions on Engineering Management*, vol. 55, no. 2, pp. 248-266, Mayo de 2008.
- [134] J. Trevelyan, "Technical Coordination in Engineering Practice," *Journal of Engineering Education*, vol. 96, no. 3, pp. 191-204, Julio de 2007.
- [135] C. Quintáns, M.J. Moure, E. Mandado y M. A. Castro, "Metodología para la Enseñanza de los Convertidores A/D Pipeline Mediante Herramientas Software," *Actas del Seminario Anual de Automática y Electrónica Industrial*, Gijón, España, pp. 120-125, Septiembre de 2006.
- [136] C. Quintáns y M.A. Castro, "A/D Converters Learning Using Software Tools," *Proceedings of the IEEE Instrumentation and Measurement Conference*, IMTC, Mayo de 2007.
- [137] J.C. Boluda, M.A. Martínez, M.A. Larrea, R. Gadea y R.J. Colom, "An Active Methodology for Teaching Electronic Systems Design," *IEEE Transactions on Education*, vol. 49, no. 3, pp. 355-359, Agosto de 2006.
- [138] D. López, J.R. Herrero, A. Pajuelo y A. Duran, "A Proposal for Continuous Assessment at Low Cost," *Proceedings of the ASEE/IEEE Frontiers in Education Conference*, FIE, 2007.
- [139] A. Platonov y L. Malkiewicz, "Particularities of the Cyclic A/D Converters ENOB Definition and Measurement," *Proceedings of the IEEE Instrumentation and Measurement Conference*, IMTC, Abril de 2006.

ANEXO I: CONTENIDO DEL CDROM

Gráficas Sigma-Delta.xls Contiene las simulaciones para obtener las gráficas de: la ganancia de resolución frente al factor de sobremuestreo; del espectro de frecuencia de los moduladores de orden cero a orden tres; y de las respuestas en frecuencia de los filtros de diezmado.

SD-ADC-5-Bits.xls Contiene la caracterización del convertidor Sigma-Delta de 5 bits que se simuló en el Apartado 3.7 y cuyo modulador se simuló mediante PSpice y cuya etapa de filtrado se simuló con el Quartus II.

ADC Pipeline simulado con excel.xls Contiene la simulación mediante la hoja de cálculo de dos convertidores pipeline, uno de 4 bits y otro de 5.

Simulaciones-Selección-Convertidores.xls Contiene las simulaciones de la hoja de cálculo que muestran la pérdida de calidad de la señal en función de la resolución del convertidor. Indica el mínimo número de bits necesario para que no se pierda más que una cierta cantidad de la relación señal ruido de la señal.

P6-Enseñanza ADC.pps Esta presentación incluye las diapositivas utilizadas en las clases en las que se ha puesto en práctica la metodología docente para la enseñanza de los circuitos ADC.

Nuevo-convertidor-multietapa Esta carpeta incluye el proyecto de OrCAD PSpice V.9.0 del convertidor multietapa presentado en el Apartado 5.2.

PIPELINE-5-BITS Esta carpeta contiene el proyecto de OrCAD PSpice que simula el convertidor pipeline de 5-bits que se presentó en el Apartado 4.4.

SDM-FO Esta carpeta contiene el proyecto de OrCAD PSpice que simula la versión del modulador sigma-delta de primer orden que tiene el integrador inversor.

SDM-SO Esta carpeta contiene el proyecto de OrCAD PSpice que simula el modulador sigma-delta de segundo orden.

SDMIP Esta carpeta contiene el proyecto de OrCAD PSpice que simula la versión del modulador sigma-delta de primer orden que tiene el integrador diferencial y está basado en la nueva estructura presentada en el Apartado 5.3.

SDM2P Esta carpeta contiene el proyecto de OrCAD PSpice que simula la versión del modulador sigma-delta de segundo orden que tiene el integrador diferencial y está basado en la nueva estructura presentada en el Apartado 5.3.

SDM3P Esta carpeta contiene el proyecto de OrCAD PSpice que simula la versión del modulador sigma-delta de tercer orden que tiene el integrador diferencial y está basado en la nueva estructura presentada en el Apartado 5.3.

SDMC1 Esta carpeta contiene el proyecto de VisualC++ 6.0 que simula el modulador sigma-delta de primer orden que se presentó en el Apartado 3.7.

Filtro-FPGA Esta carpeta contiene el proyecto del filtro de diezmado para el convertidor sigma-delta simulado en el Apartado 3.7.

NTF2P Esta carpeta contiene el proyecto de OrCAD PSpice que simula el circuito equivalente de la función de transferencia del ruido del modulador de segundo orden presentado en el Apartado 5.3.6.

NTF3P Esta carpeta contiene el proyecto de OrCAD PSpice que simula el circuito equivalente de la función de transferencia del ruido del modulador de tercer orden presentado en el Apartado 5.3.6.

PI Esta carpeta contiene el proyecto de QuartusII, que es la parte del prototipo de convertidor sigma-delta presentado en el Apartado 5.3.8 que se ha implementado en la FPGA.

ANÁLISIS Esta carpeta contiene los scripts de Matlab que se han utilizado en esta tesis y, además, los datos que se adquirieron en la pruebas de conversión con el prototipo del convertidor sigma-delta presentado en el Apartado 5.3.

TESIS-Camilo Quintáns.pdf Archivo de impresión con la memoria de esta tesis.

