Energy-Efficient Time-Based Encoders and Digital Signal Processors in Continuous Time

Sharvil Patil

Submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in the Graduate School of Arts and Sciences

COLUMBIA UNIVERSITY
2017

©2017

Sharvil Patil

All Rights Reserved

Abstract

Energy-Efficient Time-Based Encoders and Digital Signal Processors in Continuous Time

Sharvil Patil

Continuous-time (CT) data conversion and continuous-time digital signal processing (DSP) are an interesting alternative to conventional methods of signal conversion and processing. This alternative proposes time-based encoding that may not suffer from aliasing; shows superior spectral properties (e.g. no quantization noise floor); and enables time-based, event-driven, flexible signal processing using digital circuits, thus scaling well with technology. Despite these interesting features, this approach has so far been limited by the CT encoder, due to both its relatively poor energy efficiency and the constraints it imposes on the subsequent CT DSP. In this thesis, we present three principles that address these limitations and help improve the CT ADC/DSP system.

First, an adaptive-resolution encoding scheme that achieves first-order reconstruction with simple circuitry is proposed. It is shown that for certain signals, the scheme can significantly reduce the number of samples generated per unit of time for a given accuracy compared to schemes based on zero-order-hold reconstruction, thus promising to lead to low dynamic power dissipation at the system level.

Presented next is a novel time-based CT ADC architecture, and associated encoding scheme, that allows a compact, energy-efficient circuit implementation, and achieves first-order quantization error spectral shaping. The design of a test chip, implemented in a 0.65-V 28-nm FDSOI process, that includes this CT ADC and a 10-tap programmable FIR CT DSP to process its output is described. The system achieves 32 dB - 42 dB SNDR over a 10 MHz - 50 MHz bandwidth, occupies 0.093 mm², and dissipates 15 μ W-163 μ W as the input amplitude goes from zero to full scale.

Finally, an investigation into the possibility of CT encoding using voltage-controlled oscillators is undertaken, and it leads to a CT ADC/DSP system architecture composed primarily of asynchronous digital delays. The latter makes the system highly digital and technology-scaling-friendly and, hence, is particularly attractive from the point of view of technology migration. The design of a test chip, where this delay-based CT ADC/DSP system architecture is used to implement a 16-tap programmable FIR filter, in a 1.2-V 28-nm FDSOI process, is described. Simulations show that the system will achieve a 33 dB – 40 dB SNDR over a 600 MHz bandwidth, while dissipating 4 mW.

Contents

L	List of Figures		
List of Tables		xv	
1	Intro	duction	1
	1.1	Signal Processors and Processing Domains	1
	1.2	Continuous-Time Data Conversion and DSP	3
		1.2.1 CT ADC	4
		1.2.2 CT DSP	9
		1.2.3 Considerations for Low Power and Area in a CT ADC/DSP System	14
	1.3	Thesis Goals and Organization	20
2	Adap	tive Derivative Level-Crossing Sampling	23
	2.1	Introduction	23
	2.2	Derivative Level-Crossing Sampling	26
	2.3	Companded DLCS	29
	2.4	Adaptive-Resolution (AR) DLCS	29
		2.4.1 System Description	29
		2.4.2 System Design Procedure	31
	2.5	SER Comparison	32

	2.6	Sample Generation Rate and Figure of Merit	33
		2.6.1 FOM Definition	33
		2.6.2 Simulation Results	34
	2.7	Practical Considerations.	36
	2.8	Conclusions	37
3	An Eı	rror-Shaping Alias-Free CT ADC/DSP/DAC System	38
	3.1	Introduction	38
	3.2	Overview of Existing Medium-Resolution CT ADC Architectures	40
	3.3	Proposed CT ADC Architecture	42
		3.3.1 Operation	42
		3.3.2 Model	44
		3.3.3 Design Considerations	47
		3.3.4 CT ADC Integrated Implementation	50
		3.3.5 Measurement Results	57
		3.3.6 Comparison of CT ADC with the State of the Art	64
	3.4	CT DSP	67
		3.4.1 Integrated Implementation	71
		3.4.2 Simulation Results	85
		3.4.3 Comparison with the State of the Art	90
	3.5	Conclusions	91
4	Conti	nuous-Time Data Conversion and DSP Using Voltage-Controlled Oscillators	92
	<i>1</i> 1	Introduction	02

4.2	Pulse Width Modulation Using a VCO	94
	4.2.1 System Architecture	96
	4.2.2 Simulation Results	99
	4.2.3 Non-Idealities and Practical Considerations	104
	4.2.4 Conclusions	107
4.3	Pulse Frequency Modulation Using a VCO	107
	4.3.1 System Architecture	108
	4.3.2 System Model and Spectral Description	112
	4.3.3 Simulation Results	121
	4.3.4 Practical Considerations.	126
	4.3.5 Conclusions	128
4.4	4 Chapter Summary	128
5 A I	Delay-Based CT ADC/DSP/DAC System	129
5.1	I Introduction	129
5.2	2 Top-level Architecture	130
	5.2.1 Choice of Tap Delay, τ	132
	5.2.2 PFM Encoder Architecture	134
5.3	3 Integrated Implementation	141
	5.3.1 Specifications and Targets	141
	5.3.2 Delay Cell Design for Delay Line	144
	5.3.3 Delay Cell Design for the ADC	150
	5.3.4 MDAC Design	160

		5.3.6 System-Level Simulation Results and Comparisons	169
5.	.4	Conclusions	178
6 C	onc	lusions	179
6.	.1	Thesis Contributions	179
6.	.2	Suggestions for Future Work	180
Bibl	liog	raphy	186

List of Figures

1.1. Signal processing alternatives based on signal domains	2
1.2. A typical CT DSP signal processing chain.	3
1.3. Level-crossing sampling: When an input crosses a level, the digital out	put transitions to a code
that corresponds to that level.	4
1.4. Spectral comparison between DT and CT ADCs [1]. For sinusoidal inp	outs, CT ADCs produce
only harmonic distortion in the output spectrum, whereas DT ADC	s additionally alias the
distortion components, thereby creating "quantization noise floor"	6
1.5. An example plot showing the exponential worsening of $NTPS$ and T_{GR}	RAN with LCS quantizer
resolution	8
1.6. The CT DSP that follows the CT ADC uses a transversal structure to in	mplement an FIR filter.
	9
1.7. To preserve timing details, each tap delay, T_{TAP} , is implemented as a	a cascade of unit delay
cells, each with a delay of T_{GRAN}	10
1.8. System-level view of the CT ADC/DSP/DAC system from Ref. [17]	11
1.9. Dynamic power dissipation and delay line chip area of an 8-tap FIR (CT DSP with $T_{TAP} = 25$
$μ$ s and for f_{in} = 2 kHz, estimated using (1.1) and (1.2), using typical $μ$	numbers obtained from
Ref. [16]: $E_{Del} = 50$ fJ, $E_{Arithmetic} = 150$ pJ, $A_{Del} = 20 \mu \text{m}^2$	13

1.1(D. Diagram depicting design considerations for achieving low power in a CT DSP system 13
2.1.	Comparison of (a) zero-order and (b) first-order reconstruction.
2.2.	Principle of derivative level-crossing sampling scheme: (a) actual scheme in a communication
	system; (b) conceptually equivalent system for analysis purposes
2.3.	Blow-up of DLCS (first-order) and LCS (zero-order) reconstruction for a full-scale sinusoida
	input signal at 2 kHz. 26
2.4.	Signal-to-error ratio (SER) for full-scale sinusoids using DLCS reconstruction; dashed lines
	correspond to the LCS SER: 6.02 <i>N</i> +1.76 dB for <i>N</i> bits of resolution.
2.5.	Adaptive-resolution derivative level-crossing sampling and reconstruction principle
	$\Delta(xdd(t))$ denotes the variable quantization step size, which depends on the value of $xdd(t)$
	30
2.6.	Quantizer resolution versus the magnitude of the second derivative of the input for the system
	depicted in Fig. 2.5. We consider full-scale sinusoids from 0 to 4 kHz
2.7.	Signal-to-error ratio (SER) for DLCS, companded DLCS, AR DLCS—where quantize
	resolution varies from 4.5 to 11 bits with an average resolution of around 5.7 bits—, and LCS
	for full-scale sinusoidal inputs.
3.1.	Processing alternatives based on signal domain type in the ULP RX system context 39
3.2.	The asynchronous delta modulator architecture used in prior work
3.3.	Proposed CT ADC architecture. 42
3.4.	Example waveforms of the differential input, output and some key internal signals 43

3.5.	Illustrating the development of a model for the ADC of Fig. 3.3, with a sinusoidal input (not
	shown). The two upper waveforms are fictitious ones (see text). The ADC generates pulses
	at instants where the LCS-quantized version of the unfolded integrated signal makes step-
	transitions. 45
3.6.	The proposed ADC is modeled as a cascade of an integrator, a level-crossing sampling
	quantizer, and a Δ block (which behaves like a differentiator). The input signal and input noise
	components pass through an integrator-differentiator cascade and come out without
	frequency shaping. The quantizer adds quantization error and thermal noise, which are first-
	order shaped by the differentiator transfer function of the Δ block. Time waveforms are shown
	below the corresponding spectra
3.7.	Example showing overshoots and an overflow situation when the integrator input changes
	sign while its output has exceeded the comparison window set by V_C . A crossing of V_{SAT} is
	detected and integrator outputs are reset. 48
3.8.	The output token rate of the ADC increases in proportion to the input signal amplitude. A
	higher amplitude results in faster integration, and hence, a higher rate of threshold crossings.
	The polarity of the output (INC/DEC) depends on that of the input
3.9.	Transconductor (G_m) - C integrator circuit. 51
3.10	. (a) Threshold setting scheme with the comparator architecture. (b) Waveforms illustrating
	the threshold setting mechanism. 52
3.11	. The comparator delay drops as the slope of its input increases
3 12	CT ADC chip micrograph 57

3.13. Output extraction and reconstruction. The output spectrum is obtained by performing an FF
on the difference between <i>INC_out</i> and <i>DEC_out</i> signals.
3.14. Measured output spectra for -3 dBFS single-tone inputs at (a) 10 MHz and (b) 50 MHz, wit
$V_C = 80$ mV and $I_{GM} = 4$ μ A. It contains the signal component and its first-order-shape
harmonics5
3.15. An out-of-band test tone at 60 MHz does not result in any degradation due to aliasing of
increased noise ($V_C = 80 \text{ mV}$ and $I_{GM} = 4 \mu\text{A}$).
3.16. Plot of single-tone SNR/SNDR versus input frequency. The input amplitude is -3 dBFS, an
$V_C = 80 \text{ mV} \text{ and } I_{GM} = 4 \mu\text{A}.$
3.17. Two-tone output spectrum; the input tones are at 48 MHz and 50 MHz; $V_C = 80$ mV and I_{GL}
= $4 \mu A$. The output consists of signal components and IM products. The low-frequency nois
floor does not show first-order shaping, and is attributed to the input noise from the two-ton
signal generator, which was different from the one used for single-tone tests
3.18. SFDR measurements for a two-tone input with two tones at 48 MHz and 50 MHz. V_C and
I_{GM} are changed from their nominal values to demonstrate programmability. (a) SFDR and
power dissipation vs. the input amplitude for different V_C values with $I_{GM} = 10 \mu A$; (b) SFDI
and power dissipation for different values of I_{GM} with $V_C = 44$ mV and a 130 mV _{p-p} input. 6.
3.19. Effect of back-bias (V_{BB}) used for digital circuits on ADC performance. A higher back-bia
lowers delay and offers better linearity at the expense of power dissipation. Test set up is th
same as that used to generate Fig. 3.18(a), but with different values of V_{BB} (V_{BB} is the absolute
value of the back-gate bias, the latter being positive for NMOS and negative for PMOS) 6.

3.20	. Comparison of the presented CT ADC with state-of-the-art ADCs in the Murmann survey
	w.r.t. (a) energy per conversion; and (b) Walden figure of merit
3.21	. (a) Specifications for the desired filter frequency response with an example center frequency
	of 50 MHz and an $f_{s,FILT}$ of 200 MHz; (b) frequency response of (a) for an $f_{s,FILT}$ of 10 MHz.
3.22	. For a minimum intersample time, T_{GRAN} , of 2 ns, each tap delay, T_{TAP} , of 100 ns is
	implemented using a cascade of 50 2-ns digital delay cells
3.23	. A tap delay in a parallelized delay line with N_P parallel paths. Each path of the tap delay is implemented as a cascade of delay cells, each with a delay of $N_P T_{GRAN}$. The input to the tap
	is connected to one path at a time and circulated in a round-robin fashion
3.24	. Principle of the delay cell presented in Ref. [25] along with example time waveforms 75
3.25	. Architecture of the delay cell used in the proposed CT DSP
3.26	. Time waveforms for some key signals in the delay cell in Fig. 3.25
3.27	. Plot of the delay and energy/operation versus the charging current, I_D , for the delay cell in
	Fig. 3.25
3.28	One path (of the 5) in a delay tap along with the additional calibration circuitry: 2 10-ns (coarse) delay cells, 5 1-ns (fine) delay cells, and multiplexers for programmability 80
3.29	. Schematic of the arithmetic unit (multipliers and adder) [50]
3.30	. Chip micrograph of the CT ADC/DSP/DAC system
3.31	. The CT DSP is configured in simulations to implement different frequency responses (a) by
	changing tap coefficients, $c_{0.9}$; and (b) by tuning the tap delay, T_{TAP}

3.32	. Interferer rejection using the CT/ADC/DSP/DAC system: (a) Input spectrum, with a weak
	signal and strong interfering components; (b) spectrum at the CT ADC output; and (c) that at
	the CT DSP output. 87
3.33	. Power dissipation of the entire CT ADC/DSP/DAC system, configured as a bandpass filter
	with a passband center frequency of 50 MHz, versus input amplitude for a single-tone input
	at 50 MHz
4.1.	(a) A general VCO; and (b) its terminal waveforms
4.2.	(a) Pseudo-differential voltage-controlled oscillators implementing an analog integrator; (b)
	example waveforms. 94
4.3.	CT ADC/DSP/DAC systems based on pseudo-differential VCOs: (a) general system; (b)
	general system with details of CT DSP block; (c) practical CT DSP implementation for
	avoiding very narrow pulses. 97
4.4.	Differentiator transfer function
4.5.	Spectrum of the output of the VCO-based PWM encoder (post differentiation) for a full-scale
	single-tone input at 200 Hz. 100
4.6.	System implementation for multiphase operation—each DSP slice has the CT DSP
	architecture shown in Fig. 4.3(c).
4.7.	Example spectrum demonstrating aliasing in the proposed system

4.8.	Spectrum of the output of the proposed ADC and that of the following DSP for a two ton
	input with two tones at 200 Hz and 2 kHz. The DSP is a 67 th -order low-pass FIR filter wit
	an <i>f</i> _{-3dB} of 500 Hz
4.9.	Output spectrum from Fig. 4.5 repeated with 4-kHz-bandlimited (voltage) white noise adde
	at the input of the VCOs. 10
4.10	An example of pulse frequency modulation. 10
4.11	(a) A VCO-based PFM encoder with (b) example waveforms
4.12	A typical output spectrum of a pulse-frequency modulated signal. The componer
	magnitudes are from the expression in (4.6)
4.13	Plot of $J_n(\frac{\Delta fp}{f_{in}})$ (in dB) versus n for two different values of $(\frac{\Delta fp}{f_{in}})$; the higher the latter's value
	the slower the fall in the value of the Bessel function w.r.t. n
4.14	Example waveforms to demonstrate the phase-domain model of the PFM encoder. The
	encoder output is amplified by 10× for better visibility
4.15	The PFM encoder is modeled as one that integrates the input signal with an offset to generate
	the phase signal of (4.13), then quantizes this signal, and produces a pulse at every step of the
	quantized signal. The latter operation is achieved through the Δ block
4.16	A DT VCO ADC can be modeled as a cascade of a PFM encoder and a clocked sample
4.17	Spectra of the PFM-encoded output for a single-tone input at 200 Hz with the output (a) i
	impulse-form and (b) with $T_{PW} = 40 \mu s$.

5.1.	Top-level architecture of the PFM-encoder-based CT ADC/DSP/DAC system. The PFM
	encoder produces a 1-bit pulse train at its output, and the CT DSP delays it along a tapped
	delay line composed of asynchronous delays (labelled τ). The multiplying DAC (MDAC
	multiplies the pulses at each tap output with a coefficient c_i and outputs a proportional current
	The output currents of all MDACs are summed by shorting their outputs together and
	connecting to a low-impedance node
5.2.	The out-of-band modulation products are rejected using a combination of the CT DSP transfer
	function and the sinc transfer function created by using a non-zero pulse width for the PFM
	output
5.3.	The asynchronous digital delay cell; example time waveforms; and its $T_D(V_{IN})$ characteristic
	When input equals V_B the delay of the cell is τ .
5.4.	(a) A PFM encoder made out of two asynchronous digital delays identical to the ones in the
	following CT DSP (also shown); and (b) example waveforms at the PFM encoder terminals
	following a <i>START</i> signal trigger
5.5.	Parallelized version of the PFM-encoder-based CT ADC/DSP/DAC system from Fig. 5.4(a)
5.6.	(a) Architecture of the delay cell used in the delay line; and (b) time waveforms depicting
	operation. 145
5.7.	Architecture of the delay cell used in the ADC150
5.8.	Plot of HD2, HD3, and THD versus δ
5.9.	Plot of delay of the asynchronous delay cell in the ADC versus input voltage, $v_{in} + V_{B}$ 159

5.10	. (a) The switch driver for $MDAC_i$ is made of an SR latch, which is set by TAP_i and reset by
	TAP_{i+1} ; its resulting output, Q_i , then controls the switches of the 7-bit current DAC. (b) For
	the two-path delay line structure, each $MDAC_i$ has two SR-latches, whose outputs are
	combined using an OR gate to generate the final control output Q_i
5.11	. Current-mode (a) NMOS DAC (NDAC) and (b) PMOS DAC (PDAC), 6-bit each, together
	implement a 7-bit DAC. Both DACs can never be on at the same time
5.12	. Calibration loop for calibrating the delay of the cells in the delay line
5.13	. Calibration loop for calibrating V_{Rs}
5.14	. (a) ADC output spectrum obtained from a noiseless transient simulation for a full-scale
	single-tone input at 100 MHz. (b) SDR vs. iteration number for a Monte-Carlo simulation.
5.15	. Plot of in-band SDR and SNDR of the PFM encoder output versus single-tone input
	frequency
5.16	. Plot of in-band SDR and SNDR of the PFM encoder output versus input amplitude for a
	single-tone input at 100 MHz.
5.17	. ADC output spectrum for a two-tone input with two equal-amplitude tones at 450 MHz and
	500 MHz obtained from a transient noise simulation. 173
5.18	. Placement of the proposed ADC in the (a) energy plot and (b) the Walden FOM plot of the
	Murmann survey [49]

5.19. System output spectrum, obtained from a transient noise simulation, for a full-scale single-
tone input at 100 MHz when the DSP is configured to implement a 16-tap decimation filter;
filter frequency response is also shown (dashed).
5.20. Spectra obtained from a transient simulation for a two-tone input with two tones at 50 MHz
and 500 MHz, and with the DSP configured to implement a 16-tap high-pass transfer
function. (a) ADC output spectrum; and (b) system output spectrum, showing a 15 dB
attenuation of the component at 50 MHz relative to the one at 500 MHz. Filter frequency
response is also shown (dashed).
6.1. Connecting the different principles proposed in this thesis through models of (a) an LCS
quantizer; (b) the DLCS quantizer of Chap. 2; (c) the error-shaping modulator of Chap. 3;
and (d) the PFM encoder of Chaps. 4-5.
6.2. A general CT encoder can be developed by preceding an LCS quantizer with a general transfer
function $H(s)$ and by following it with the inverse of the transfer function

List of Tables

1.1	signal domains and corresponding processors for continuous/discrete combinations of time			
	and amplitude1			
1.2	Performance summary of prior CT ADC work			
1.3	Comparison of prior CT DSP work with state-of-the-art DT DSPs			
2.1	Performance comparison of LCS (6b), DLCS (6b), and AR DLCS (signal-dependent average			
	resolution) for different input signals. All inputs are full scale			
3.1	Sizing of transistors and values of other components in the transconductor circuit in Fig. 3.9.			
	51			
3.2	Comparison of the proposed ADC with sampled ADCs with bandwidths ≤100 MHz and			
	modest SNDR values. 64			
3.3	Comparison of the proposed ADC with other CT ADCs. 65			
3.4	Table summarizing the specifications of the FIR filter and those of its individual blocks 70			
3.5	Sizing/values of different components in the delay cell shown in Fig. 3.25			
3.6	Summary of the delay cell performance and comparison with other similar delay cells in prior			
	art			

3.7	Comparison of proposed CT ADC/DSP/DAC system with state-of-the-art CT/DT DSPs	P/DAC system with state-of-the-art CT/DT DSPs and		
	analog FIR filters.	. 90		
4.1	Comparison of the VCO-based PWM encoder system with an LCS CT ADC/DSP system	for		
	identical CT DSP specifications.	103		
4.2	2 Comparison of the VCO-based PFM encoder system with an 8-bit LCS CT ADC/DSP sys	tem		
	and the VCO-based PWM encoder system for identical CT DSP specifications	125		
5.1	Sizes of transistors in the delay cell used in the delay line, shown in Fig. 5.6(a)	146		
5.2	Performance summary of the asynchronous digital delay cell in the delay line	149		
5.3	Sizes of transistors in the delay cell used in the ADC, shown in Fig. 5.7.	151		
5.4	Performance summary of the asynchronous digital delay cell in the ADC.	160		
5.5	Values for width and length parameters for the current DACs shown in Fig. 5.11	164		
5.6	Performance summary of a unit NDAC/PDAC.	165		
5.7	Comparison of the PFM CT ADC with prior-published CT ADCs.	175		
5.8	3 Comparison of the proposed CT ADC/DSP/DAC system with relevant state-of-the-art	СТ		
	DSP, DT DSP, and analog processors.	178		

Acknowledgements

I would like to thank my advisor, Prof. Yannis Tsividis, for taking me under his wing and giving me an exciting research topic to work on. He allowed me absolute freedom to pursue and lead my own research ideas, intervening only to steer me in the right direction. His penchant for pursuing bold research ideas and broad interests is inspiring and contagious. It is thus no surprise that working under him has been a truly transformative experience.

Thanks are due to my defense committee members—Prof. Timothy Dickson, Dr. Dominique Morche, Prof. Matthias Preindl, and Prof. Mingoo Seok—for their patience in reading the thesis and providing valuable feedback.

A major part of my thesis was done as part of a collaboration with Dominique Morche and Alin Ratiu at CEA-LETI. I am thankful to Dominique for his unique perspective which helped us evaluate many ideas early on. Alin's passion for research was infectious, and it drove us to achieve something significant together. Colleagues at LETI, including Iulia Tunaru, Robert Polster, and Matthieu Verdy, made my time in Grenoble very memorable. Observing the French way of life first hand in Grenoble was an eye-opening experience, one that I will always cherish.

Working with Pablo Martinez-Nuevo early on in my PhD gave me the confidence to pursue unconventional ideas; I thank him for being a great colleague. I also thank Xiaoyang Zhang and Prof. Yong Lian for their wisdom, guidance, and support during our collaboration. I thank Prof. Thao Nguyen for his enlightening lectures and stimulating discussions, some of which changed the way I think for the better. I also thank Prof. Peter Kinget for providing thoughtful advice over the years. Thanks are also due to Bob Schell for many helpful discussions.

I was lucky to have Suhas, Arun, and Sean to assist me in my projects; I thank them for their diligent help. I am grateful to Anuya Patil, Piyush Patil, Ranjit Desai, Vihang Patil, and Arjun Gupta for their help in editing this thesis. I thank the electrical engineering department administration, especially Elsa Sanchez and Laura Castillo, for their assistance throughout.

I would like to thank all my colleagues at CISL. In particular, I thank Kshitij for teaching me the ways of the PhD early on; Baradwaj, Jayanth, Jianxun, and Karthik for their advice and support on myriad issues; and Tugce, Ning, Sarthak, Shravan, and Vivek for their camaraderie. I found a great colleague, collaborator—and friend—in Yu Chen; my many interactions with him helped me grow as a researcher. I thank Mayank Misra for his camaraderie and friendship during our PhD journey. His many jokes made the tough times seem a lot easier.

I thank Prof. Anurup Mitra at BITS, Pilani for making me fall in love with analog circuits, and Dr. S.C. Bose at CEERI, Pilani for helping me reaffirm and pursue it further. I am also thankful to Mr. Santosh Kumar at CEERI for his mentorship during my time at CEERI. I am indebted to colleagues at STMicroelectronics for expanding my technical knowhow.

I owe a debt of gratitude to my entire family for their unwavering support. I thank my parents, Pradeep and Sanyogita, for teaching me the value of honest hard work. I am also grateful to my brothers, Piyush and Mimoh, for their invaluable friendship.

Finally, the value of being married to a (experimental) psychologist became clear to me during the tumult of grad-school life. I would thus like to thank Anuya for being by my side throughout this incredible journey, sometimes as a colleague (even drawing some of the figures in this thesis) and sometimes as reviewer-number-two. And though I will miss my time at Columbia, I look forward to likewise being by her side as she pursues her own research journey.

Chapter 1

Introduction

1.1 Signal Processors and Processing Domains

Advances in CMOS technology have resulted in a very diverse application space today. Developing a one-size-fits-all signal processor without compromising performance or energy efficiency is thus not practical. The correct approach involves an educated processor choice tailored to a specific application. As regards to processor choice, if we split signal domains along the time and amplitude axes, we end up with four possibilities shown in Table 1.1 [1] with distinct signal processor types. When both time and amplitude axes are continuous, the resulting signal is continuous-time (CT) analog; the processor is a classical analog one. When both axes are discrete, the discrete-time (DT) digital domain results; a classical DT digital signal processor (DSP¹) processes signals in this domain. Signals in which the time axis is discrete while the amplitude

Time	Amplitude	Signal domain	Processor
Continuous	Continuous	CT analog	Classical analog
Discrete	Discrete	DT digital	Classical DT DSP
Discrete	Continuous	DT analog	Analog sampled-data
Continuous	Discrete	CT digital	CT DSP

Table 1.1: Signal domains and corresponding processors for continuous/discrete combinations of time and amplitude.

¹ In this thesis, DSP will stand as an abbreviation for both "digital signal processing" and "digital signal processor" depending on the context.

axis is continuous are DT analog and can be processed by sampled-data analog processors (e.g. switched-capacitor filters). Finally, by symmetry, the signal domain with a continuous time axis and a discrete amplitude axis is called "CT digital", and the corresponding processor is called "CT DSP" [1].

Fig. 1.1 depicts the different signal processing alternatives, drawn based on these signal domains. Each processing category has unique features and limitations; the former can be exploited while the latter will act as hindrances in the context of specific applications. For instance, analog signal processing, while power efficient, does not offer the desired programmability, making it inappropriate for applications that demand a high degree of the latter. DT DSP allows a high degree of programmability. However, it requires a DT analog-to-digital converter (ADC)² with sampling at regular clock intervals. Sampling results in aliasing and, in the case of Nyquist-rate sampling, an antialiasing filter with stringent specifications needs to precede the DT ADC so

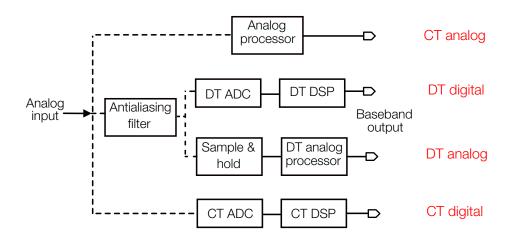


Fig. 1.1. Signal processing alternatives based on signal domains.

_

² In this thesis, an ADC will also be referred to as an "encoder" or, in some cases, a "modulator".

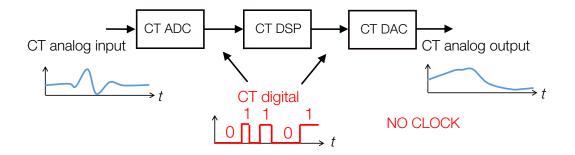


Fig. 1.2. A typical CT DSP signal processing chain.

as to band-limit the input signal. Such an antialiasing filter can be quite power-hungry. Oversampling can simplify the filter specifications, but the high sampling rate results in a major power overhead for the DT ADC and the DT DSP. The DT analog representation, too, suffers from aliasing, and requires an antialiasing filter. Thus, power-efficient handling of DT digital and DT analog signals is a challenge. In contrast, the CT digital domain and CT DSP present an interesting signal processing paradigm that allows A/D conversion with no sampling in time, and thus, with no aliasing. It is discussed in detail next.

1.2 Continuous-Time Data Conversion and DSP

The CT DSP signal processing chain is shown in Fig. 1.2. An input analog signal is converted into CT digital form by a clockless CT ADC (to be described later) [2], [3]; the CT digital output is then processed *directly* by a clockless CT DSP with no sampling in time, producing another (processed) CT digital signal at its output. The DSP output can be converted back to analog form using a CT digital-to-analog converter (DAC). CT DSP presents a principle where digital signals that are binary functions of *continuous* time are processed, while their timing details, as they evolve in continuous time, are preserved in the DSP. Being digital, such a DSP has the amplitude noise immunity and programmability of a conventional DT DSP.

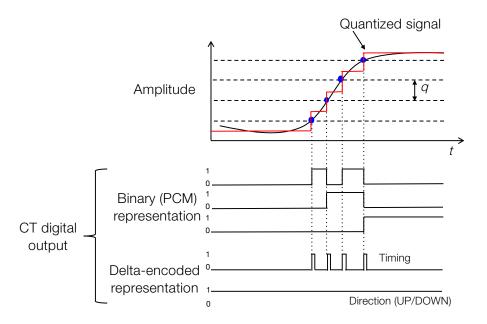


Fig. 1.3. Level-crossing sampling: When an input crosses a level, the digital output transitions to a code that corresponds to that level.

Much of the prior work in CT DSP systems is based on CT A/D conversion using level-crossing sampling (LCS) [3]–[13]. Therefore, even though CT DSP is not restricted to LCS, we will use the latter as vehicle to describe the constraints and low-power design considerations of a general CT DSP system. In the process, we will highlight the achievements and limitations of prior work on CT DSP. Once the latter is understood, methods can be developed to further improve these systems.

1.2.1 CT ADC

System description

A CT ADC converts an analog input into a CT digital form. LCS is one possible method of encoding analog signals in CT digital form. For an *N*-bit LCS ADC, there are 2^N amplitude levels (Fig. 1.3), separated by $q = V_{FS}/2^N$ in amplitude, where V_{FS} is the full-scale amplitude range

of the ADC and q is the amplitude quantization step. Each level has an N-bit binary code associated with it. Every time the input crosses a level, the output of the ADC³ transitions to a code that corresponds to that particular level. The ADC output "token" or "sample" is thus a bundle of a timing signal—which indicates the instant of crossing—and the N-bit digital code that represents the value of the level, giving its pulse-code-modulated (PCM) representation [14]. This output is CT digital as its transitions are not synchronized to any clock, and can, in principle, occur at any point in time (provided a level is crossed). It is important to note that this output cannot be termed "asynchronous" as, unlike such signals, the timing of the level crossing is an integral part of this CT digital signal encoding and needs to be preserved in any subsequent processing [1].

The PCM code at the CT ADC output can be converted back to the analog amplitude level it represents using a CT DAC. Shown in Fig. 1.3 as the "Quantized signal", this represents the zero-order-hold (ZOH) reconstruction of the original analog input to the ADC^4 . In a uniform-resolution LCS system, any two consecutive crossed levels are always spaced in amplitude by one quantization step, q. The digital code thus always changes by \pm 1 between any two crossings. Therefore, the N-bit PCM code can be compressed to 2-bit form using delta encoding (Fig. 1.3), with one bit indicating the timing of the crossing and the other one representing the sign of crossing (UP/DOWN).

-

³ The most straightforward LCS encoder is a clockless flash ADC [69]. For an N-bit ADC, there are 2^N clockless comparators that detect level crossings, and their 2^N outputs together give a thermometric-encoded version of the digital output. The latter, when fed into a thermometric decoder, generates the corresponding N-bit PCM code.

⁴ The reconstructed analog amplitude levels are not exactly equal to the levels used during quantization; they are instead placed at the midpoints between quantization levels. This is done to minimize reconstruction error [3].

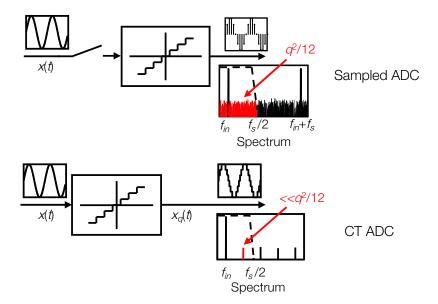


Fig. 1.4. Spectral comparison between DT and CT ADCs [1]. For sinusoidal inputs, CT ADCs produce only harmonic distortion in the output spectrum, whereas DT ADCs additionally alias the distortion components, thereby creating "quantization noise floor".

Output spectrum and SER

The cascade of an LCS CT ADC and a CT DAC represents an amplitude quantizer—an LCS quantizer—without any sampling in time. As there is no sampling in time, no aliasing occurs. No anti-aliasing filter is thus required in an LCS CT ADC/DSP/DAC system [1]. In a Nyquist-rate-clocked DT ADC, the analog input in sampled in time, followed by amplitude quantization of the sampled value. Sampling create aliases of the input in the spectrum, which extend over an infinite bandwidth; the nonlinearity of the amplitude quantizer then creates intermodulation products of these aliases that stretch right into baseband, creating what is often called "quantization noise". Assuming a sufficient numbers of level are crossed (i.e. quantization is not too coarse), the total power of this quantization noise in a bandwidth equal to half the Nyquist sampling rate, f_s , is $q^2/12$, where q is the amplitude quantization step size (see Fig. 1.4). For a full-scale single-tone input, this results in the well-known signal-to-error ratio (SER) of 6*N*+1.76 dB [15], for an *N*-bit

ADC. In contrast, an LCS quantizer only quantizes the analog input, with no sampling in time. The quantizer nonlinearity results in quantization distortion at the output⁵, which, for single-tone inputs, manifests itself as harmonics in the output spectrum⁶. The total integrated power of all these harmonics in an infinite bandwidth is $q^2/12$, for an amplitude quantization step size of q. Therefore, the total power of the quantization error harmonics that fall in the signal band can be much lower than $q^2/12$, and, consequently, the SER can be higher than 6N+1.76 dB, for an N-bit LCS quantizer. Therefore, for a given quantization step, CT LCS ADC can result in a much higher SER than a Nyquist ADC, provided the power of the noise generated by circuit components is much lower than the quantization error power.

System Parameters

There are two important parameters of any CT ADC (LCS or otherwise) that have significant implications towards the subsequent CT DSP design and system-level power budget specifications. We will discuss them now.

1. **Granularity** (T_{GRAN}): Timing is a crucial aspect of the CT digital signal representation at the CT ADC output, and it has to be preserved precisely along the processing chain. The tightest constraints to achieve this arise when the time between two consecutive CT ADC output tokens is at its minimum. This minimum is termed the granularity, T_{GRAN} , and, to a significant extent, defines the CT DSP design as we will describe soon.

⁵ We thus say that LCS quantization results only in "quantization error" and not "quantization noise" floor.

7

_

⁶ A two-tone input will create intermodulation products in the output spectrum.

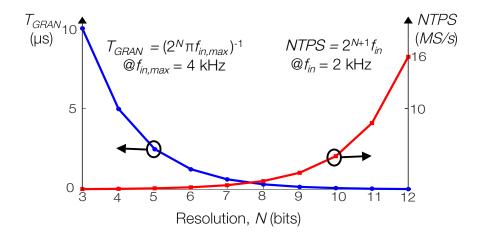


Fig. 1.5. An example plot showing the exponential worsening of NTPS and T_{GRAN} with LCS quantizer resolution.

2. **Number of tokens per second** (*NTPS*): As we will discuss next, the CT DSP is event-driven with power dissipation that varies with input activity [1]. Therefore, every token produced by the CT ADC has a certain DSP energy cost, and the total DSP power varies directly with the number of tokens produced per second (*NTPS*) by the ADC.

For an *N*-bit LCS ADC handling a full-scale sinusoidal input, $T_{GRAN} = (2^N \pi f_{in,max})^{-1}$ and $NTPS = 2^{N+1} f_{in}$ [3], where f_{in} is the input frequency and $f_{in,max}$ is the maximum input frequency. As can be seen, NTPS and T_{GRAN} worsen exponentially as the ADC resolution increases, and respectively worsen linearly and hyperbolically with a rising $f_{in,max}$ [3]. As an example, T_{GRAN} and NTPS are plotted against N in Fig. 1.5. The implications of this on the CT DSP will be discussed soon.

1.2.2 CT DSP

The output of the CT ADC is processed by a clockless CT DSP that preserves the timing details of the digital output as it evolves in CT. While a number of CT DSP implementations are possible, we restrict ourselves to linear DSP. CT linear finite-impulse-response (FIR) DSP, implemented using a transversal structure shown in Fig. 1.6, has already been demonstrated for kHz-GHz range applications [3], [10], [16] and has been shown to handle signals in many CT/DT digital formats [16]. So far, only transversal structures have been demonstrated, while recursive ones, used to implement infinite impulse response (IIR) filters, remain a work in progress. This thesis will consider only FIR DSP, and all proposed improvements in the encoders and the processors will be focused towards optimizing a system that contains an FIR DSP.

An FIR (Fig. 1.6) DSP processes an input signal by delaying it along a tapped delay-line, multiplying the tap outputs with appropriate coefficients, and then summing the multiplier outputs to generate a single final output. An N^{th} -order FIR filter has N delays, each with value T_{TAP} , and N+1 taps/coefficients. The frequency response of such a filter repeats every $f_s = 1/T_{TAP}$, and its nature can be modified by changing the filter coefficients. This transversal structure can handle

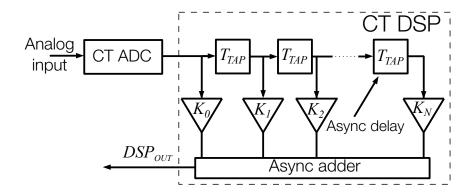


Fig. 1.6. The CT DSP that follows the CT ADC uses a transversal structure to implement an FIR filter.

CT/DT analog/digital signals with appropriate modifications. For instance, if the input to the FIR is CT analog, the delays, multipliers and adder are analog circuits; if the input is DT digital, these blocks are clocked digital circuits. In our case, the input to the CT DSP is the output of the CT ADC, which is CT digital in nature. Therefore, the tap delays are implemented using (clockless) asynchronous digital delays, and the multipliers and adder are asynchronous digital circuits [3]. All signals inside the block diagram shown in Fig. 1.6 are binary functions of continuous time. The highly digital nature of the structure allows a good degree of programmability in terms of response type (e.g. lowpass/bandpass), performance (e.g. number of taps), and specifications (e.g. passband width).

As the CT DSP delays the ADC output tokens along its delay-line, it needs to precisely preserve their timing details. The time-spacing between these tokens can be as small as the granularity, T_{GRAN} . The tap delay in the CT DSP, T_{TAP} , is usually much larger than T_{GRAN} .

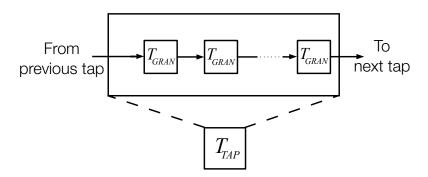


Fig. 1.7. To preserve timing details, each tap delay, T_{TAP} , is implemented as a cascade of unit delay cells, each with a delay of T_{GRAN} .

⁷ In some CT DSPs, the adder is implemented in the analog domain, by converting the multiplier output from digital to analog domain using a CT DAC [10]. We will study the impact of signal domains on efficiency of operations in Sec. 1.2.3.

Therefore, in order to preserve all the tokens and their time-spacing, each tap delay is implemented using a cascade of $N_D = [T_{TAP}/T_{GRAN}]$ number of clockless digital delay cells, each implementing a delay of T_{GRAN} , as shown in Fig. 1.7. For example, for an 8-bit LCS case with a full-scale 4 kHz $f_{in,max}$, $T_{GRAN} = 300$ ns. A tap delay of 25 μ s will then require each tap delay to be implemented as a cascade of 25μ s/300 ns ≈ 83 cells. Every ADC output token goes through all the 83 delay cells to undergo a delay defined by a single tap.

To connect the CT DSP described thus far with the complete system, we refer to Fig. 1.8, which gives a system-level view of the CT ADC/DSP/DAC system described in Ref. [17]. A CT ADC decomposes the analog input into several binary CT digital signals, $b_{1-N}(t)$. Each of the latter is processed in parallel by a slice of a CT digital FIR filter, composed of CT digital delays and digital multipliers as described above. The outputs of all slices are weighted and summed in a CT digital adder, whose output is converted to analog form using a CT DAC. Note, however, that this

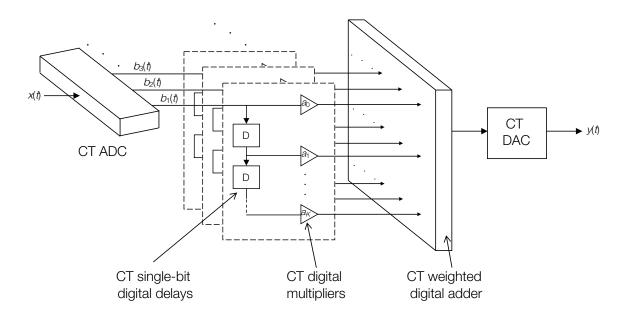


Fig. 1.8. System-level view of the CT ADC/DSP/DAC system from Ref. [17].

system shows only the principle. There are now more efficient implementations, some of which are described in this thesis.

Let us now consider the power dissipation and chip area of the CT DSP. The digital blocks that implement each of the three operations in the FIR filter—delay, multiplication, and addition—are event-driven. There cannot be a delay/multiply/add operation unless there is an input token; if there is no input token, no operation takes place, and no energy is spent (besides that due to leakage in circuits, which we assume negligible for now). Every operation takes a certain amount of energy per token. Therefore, as the number of tokens per second, *NTPS*, increases, the FIR filter power dissipation, which is the product of energy per token and the *NTPS*, increases. This is why a CT FIR DSP is said to be event-driven with activity-dependent power dissipation [1], [3], [10]. The power dissipation of the DSP (ignoring leakage) is given by [3]:

$$P_{DSP} = P_{Delay-Line} + P_{Arithmetic}$$

$$= \left[\left| \frac{T_{TAP}}{T_{GPAN}} \right| \times (N_{taps} - 1) \times E_{Del} + E_{Arithmetic} \right] \times NTPS$$
(1.1)

where E_{Del} and $E_{Arithmetic}$ are respectively the energy taken by a single delay cell to delay a token and the energy dissipated by the arithmetic blocks—the mutiplier-adder combination—per token; N_{taps} is the number of filter taps (equal to number of tap delays plus 1). The chip area of the CT DSP is dominated by that of the delay line (as compared to the arithmetic blocks in it, especially when dealing with low-frequency signals) [3] and is given by:

$$A_{Delay-line} = \left| \frac{T_{TAP}}{T_{GRAN}} \right| \times (N_{taps} - 1) \times A_{Del}$$
(1.2)

where A_{Del} is the chip area occupied by a single delay cell unit.

From (1.1) and (1.2), it is clear that, in order to keep the DSP power dissipation and chip area low for a given set of specifications (T_{TAP} , N_{taps}), the NTPS needs to be low and T_{GRAN} needs to be high. Interestingly, both of these parameters are set by the CT ADC alone, as was discussed in Sec. 1.2.1. For the case of LCS, an exponential worsening of T_{GRAN} and NTPS with ADC resolution results in an exponential rise in the DSP power dissipation and chip area. Using typical values of E_{Del} and $E_{Arithmetic}$, obtained from the integrated implementation in Ref. [16], P_{DSP} and $A_{Delay-line}$ were calculated using (1.1) and (1.2) for an 8-tap CT FIR filter, and plotted against ADC resolution, N, in Fig. 1.9. The exponential rise in P_{DSP} and $A_{Delay-line}$ is clear.

We end this section by noting that the existence of the CT digital signal domain predates its categorization in Ref. [1]. For instance, signals that are asynchronous pulse-width modulated [18], pulse-frequency modulated [19], asynchronous delta [20] or sigma-delta modulated [21] are all CT digital. Most, if not all, of these benefit from alias-free generation and have unique spectral properties. However, the CT digital domain is not fully exploited to enable greater programmability and scalability in systems that involve such signals. Considering this, the

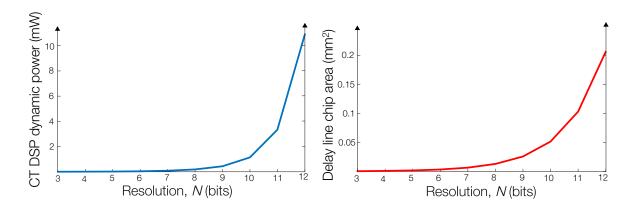


Fig. 1.9. Dynamic power dissipation and delay line chip area of an 8-tap FIR CT DSP with $T_{TAP} = 25 \, \mu s$ and for $f_{in} = 2 \, kHz$, estimated using (1.1) and (1.2), using typical numbers obtained from Ref. [16]: $E_{Del} = 50 \, \text{fJ}$, $E_{Arithmetic} = 150 \, \text{pJ}$, $A_{Del} = 20 \, \mu \text{m}^2$.

fundamental contribution of Ref. [2] was the demonstration that such CT digital signals can be processed directly by a DSP in continuous time using digital blocks like clockless delays, multipliers and adders, allowing flexible filtering capabilities with good programmability. This thesis attempts to take the state of research in this field one step closer to what it promises to be.

1.2.3 Considerations for Low Power and Area in a CT ADC/DSP System

Implementing a CT ADC/DSP system with low power dissipation and chip area requires optimization of both the ADC and DSP for it. The design considerations to achieve low power are summarized in the diagram shown in Fig. 1.10. The techniques presented in this thesis exploit one or more of these considerations to achieve improved energy efficiency in a complete CT ADC/DSP system. We will discuss each of them in detail now.

1. *NTPS and T_{GRAN}*: The total power dissipation (ignoring leakage) and chip area of a CT ADC/DSP system are given by:

$$P_{SYS} = P_{ADC} + P_{DSP} = P_{ADC} + \left[\left| \frac{T_{TAP}}{T_{GRAN}} \right| \times (N_{taps} - 1) \times E_{Del} + E_{Arithmetic} \right] \times NTPS$$
 (1.3)

 $A_{SYS} = A_{ADC} + A_{Delay-line} + A_{Arithmetic}$

$$= A_{ADC} + \left| \frac{T_{TAP}}{T_{GRAN}} \right| \times (N_{taps} - 1) \times A_{Del} + A_{Arithmetic}$$
 (1.4)

Where P_{ADC} and A_{ADC} are respectively the power dissipation and chip area of the ADC, $A_{Arithmetic}$ is the area occupied by the arithmetic blocks (multipliers and adder), and the expressions for P_{DSP} and $A_{Delay-line}$ are obtained from (1.1) and (1.2).

Clearly, the NTPS and T_{GRAN} directly impact the system power and area. Both the NTPS and T_{GRAN} are defined based on the encoding scheme in the CT ADC. As was discussed in the

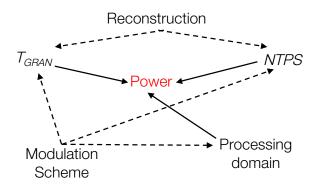


Fig. 1.10. Diagram depicting design considerations for achieving low power in a CT DSP system.

previous section and clearly seen in Fig. 1.9, in the case of LCS, they worsen exponentially with resolution and impose a severe penalty on the CT DSP power dissipation and area. This has so far limited the integrated CT DSP implementations to low orders [3], [10], [16]. It is thus clear that any CT DSP system that wishes to achieve high energy efficiency needs to adopt an encoder that can significantly relax the tight exponential trade-off between the *NTPS*, T_{GRAN} , and the ADC resolution. In this thesis, we consider a number of approaches to achieve this.

2. *Modulation scheme*: A CT ADC encodes analog information by modulating one or more aspects of a CT digital signal at its output. For instance, LCS modulates the binary code of the CT digital output in proportion to the analog input, resulting in the classical pulse-code modulation [14], but in continuous time. On the other hand, delta modulation encodes the analog input by modulating the pulse density of the CT digital output in proportion to the input slope [20].

The choice of modulation scheme influences the CT ADC/DSP system in many important ways. First, the modulation scheme defines the power dissipation of the CT ADC—the modulator/encoder. Next, it directly affects the NTPS and T_{GRAN} , and, as discussed above, the system power dissipation. Finally, the nature of the encoded output—single/multi-bit—influences

the power dissipation of the CT FIR filter. The latter involves delay, multiply, and add operations. Delaying a 1-bit digital signal in continuous time is more energy efficient [3] than delaying a multibit one, as the latter requires memory to store and access digital information [16]; this is in addition to needing asynchronous delay units to delay the timing information. Furthermore, CT/DT digital multiplication becomes extremely simple and energy-efficient when one of the operands (say, the A/D encoder output) is 1-bit, as the multiplier can be implemented using pass-gates [10]. In contrast, multiplying two multi-bit digital operands—CT or DT—is significantly more power hungry [16].

The choice of modulation scheme can thus significantly affect the energy efficiency of the CT ADC/DSP system. In this thesis, modulation schemes other than the hitherto-common LCS are considered and shown to give drastic improvements over existing CT DSP systems in terms of energy efficiency.

3. *Reconstruction*: The output of the CT ADC (or CT DSP) can be converted back to analog form using a CT DAC. In Sec. 1.2.1, ZOH—or piecewise constant—reconstruction was discussed. Using higher-order reconstruction, the quantization error can be significantly reduced. Conversely, for a given accuracy requirement, a CT system with higher-order reconstruction can use a CT ADC with a lower resolution than that in one that uses ZOH reconstruction. For example, it was shown in Ref. [22] that an LCS ADC with only 4-bit resolution can achieve single-tone SER of above 100 dB using higher-order reconstruction schemes. Given the exponential dependence of *NTPS* and T_{GRAN} on resolution and their implications on system power, the gains obtained from going for higher-order over ZOH reconstruction can be significant. The catch in this, however, is that higher-order reconstruction schemes transfer the onus from the ADC to the DAC. Unless the power dissipation constraints on the DAC side are significantly relaxed, higher-order reconstruction may

not be feasible. Besides, higher-order reconstruction schemes are generally slow and non-real-time in nature [22].

In this thesis, we assume that the application context forces a tight power budget on every block—the ADC, DSP, and the DAC. This is true in case of applications like wireless sensor nodes, where every node is power constrained, thereby not allowing the DAC, which is expected to be at the receiver sensor node, a very high power budget. In any such application, any encoder that allows a higher-order reconstruction scheme needs to be fast, able to operate in real time, and such that it will not overwhelm the power budget at the DAC end of the chain. This thesis proposes such a scheme.

4. Adaptive-resolution quantization: The NTPS- T_{GRAN} -resolution trade-off is valid in the case of uniform-resolution LCS quantization, in which the encoder uses a fixed quantization resolution independent of the input. Adaptive-resolution quantization attempts to exploit certain signal characteristics and accordingly intervene in the quantization process with an adaptive quantization step, with the aim to relax the NTPS- T_{GRAN} -resolution trade-off while not compromising accuracy. For instance, Ref. [23] proposed an adaptive-resolution (AR) LCS quantization scheme that varies the quantization step in proportion to the input slope—the higher the slope, the higher the quantization step and vice versa. It was shown that the resulting degradation in in-band accuracy is negligible; an order of magnitude relaxation in T_{GRAN} and drastic reduction in $NTPS^8$ was demonstrated [9] for a given quantization accuracy requirement. As this example shows, AR

_

⁸ The exact amount of reduction is input signal dependent.

quantization schemes can be used to improve a given modulation scheme. This aspect is explored in this thesis to improve an existing modulation scheme that also achieves superior reconstruction. 5. Hybrid processing domains: To lower the system power dissipation, one needs to optimize the FIR filter to lower its power dissipation while retaining flexibility for given specifications. This can be achieved by choosing an encoding format and signal domain tailored to lower the energy required for a particular operation—delay, multiply, or add—in the FIR filter. Delay operation can be conducted with high energy efficiency in the CT/DT digital or DT analog domain, unlike the case with the CT analog domain [10]. Multiplication in the CT/DT digital/analog domain can be quite power hungry. However, as discussed above, if the ADC encodes the analog input in a digital signal with few bits (e.g. 2), multiplication becomes extremely simple and energy-efficient, as it can be achieved with pass gates. Finally, addition is far more energy efficient when done in the CT/DT analog domain as compared to the CT/DT digital domain. For instance, the multi-bit CT digital multiply-add operations in the FIR filter in Ref. [16] consume 150 pJ/token, whereas the single-bit CT digital multiplication followed by analog addition in the FIR filter in Ref. [10] dissipates 30 fJ/token/tap or 180 fJ/token. While it is not entirely fair to compare the two due to their different resolutions (8-bit in Ref. [16] versus 3-bit in Ref. [10]), a four-orders-of-magnitude improvement based on appropriate choice of processing domains is significant and noteworthy.

We thus conclude that the energy efficiency of an operation depends on the signal domain and the encoding format (few bits versus too many bits) involved, and so does the energy efficiency of the resulting FIR filter. It is clear that an energy-efficient FIR filter would be one that has a preceding CT/DT ADC that encodes the analog input in a CT/DT digital form with few bits; asynchronous/DT digital delays in the delay-line; simple pass-gates as multipliers; and addition of the digital outputs of the multiplier performed in the analog domain by first converting them using

DACs. This approach of choosing hybrid processing domains was adopted in the CT DSP system in Ref. [10] and demonstrated to process GHz-range signals with an energy efficiency that was almost two order of magnitude better than previous CT DSPs and within a factor of 2 of state-of-the-art DT DSPs. The system, however, had a limited resolution of 3 bits in the flash LCS encoder, and a flash architecture would cause an exponential power and area penalty with any increase of resolution. In this thesis, we propose modulation/encoding schemes that can exploit this concept of hybrid processing domains in the FIR to achieve good energy efficiency, while extending resolution beyond 3 bits.

We conclude this section by noting one topic that we have not considered so far: that of circuit-level improvements in the CT ADC, asynchronous digital delay and the multiplier-adder circuits. These can respectively help keep parameters P_{ADC} , E_{Del} , and $E_{Arithmetic}$ in (1.3)-(1.4) low, thereby lowering power dissipation at the system level. For instance, adaptive biasing of zero-crossing detectors in the CT ADC in Ref. [9] along with adaptive-resolution quantization brought about an order of magnitude improvement in the energy efficiency over prior CT ADC work. The asynchronous delay architecture proposed in Ref. [24] brought about a 2× improvement in the energy efficiency over that in Ref. [25]. Parallelization in the CT DSP can also help relax the granularity constraint, at the expense of area and a higher sensitivity to mismatch [10]. These techniques are universal and can be applied to any of the schemes proposed in this thesis; in fact, in some cases, they have been. However, they cannot be considered to be the central contributions of this thesis. The latter, instead, would be the underlying principles—including the modulation/reconstruction scheme—that allow a drastic system-level relaxation of constraints, thereby facilitating a low-power and low-area implementation.

	Schell, JSSC'08 [3]	Kurchuk, JSSC'12 [10]	Weltin-Wu, JSSC'13 [9]
Technology	90 nm CMOS	65 nm CMOS	130 nm CMOS
Supply (V)	1	1.2	1
Input bandwidth (f_{BW})	10 kHz	2.4 GHz (0.8 GHz-3.2 GHz)	20 kHz
Core area (mm ²)	0.06	0.0036	0.36
SNDR (dB)	58	20.3	47-54
Total power, P (μW)	50	2700	2-8
Walden figure of merit ^a (fJ/conv-step)	3769	66	200-850
$P/(2f_{BW})$ (pJ)	2500	0.56	200
Antialiasing filter required?	No	No	No

^aWalden figure of merit is defined as: $FOM = \frac{P}{2f_{BW}2^{ENOB}}$; ENOB = (SNDR-1.76)/6.

Table 1.2. Performance summary of prior CT ADC work.

1.3 Thesis Goals and Organization

Tables 1.2 and 1.3 respectively summarize the prior CT ADC and CT DSP work, all of which is based on LCS encoding. Table 1.3 also includes relevant DT DSP and analog processors for comparison. Prior CT ADC/DSP systems achieve a degree of programmability that is comparable to DT DSPs. However, despite significant improvements over time, either their energy efficiency [3] or resolution [10] remains relatively poor.

The CT ADC, particularly, is worse by over an order of magnitude compared to state-of-the-art DT ADCs. The CT ADC is crucial towards achieving high system-level energy efficiency as its contribution to the system power dissipation is two-fold. First, it adds to the total system power consumption. Further, it also defines the power dissipation of the CT DSP by setting the NTPS and T_{GRAN} . Once the encoder is fixed, there are very few techniques other than brute-force

	Schell, JSSC'08 [3]	Kurchuk, JSSC'12 [10]	Agrawal, ISSCC'10 [67]	Toifl, ESSCIRC'14 [68]
Technology	90 nm CMOS	65 nm CMOS	32 nm CMOS	32 nm CMOS
Supply (V)	1	1.2	1	0.6
Туре	CT DSP	CT mixed-signal DSP	DT DSP	DT DSP
Input bandwidth, f_{BW}	10 kHz	2.4 GHz (0.8 GHz-3.2 GHz)	1.05 GHz	8 GHz
Average sample rate	0-8 MS/s	0-45 GS/s	2.1 GS/s	16 GS/s
Core area (mm ²)	0.55	0.073	0.004	0.033
SNDR (dB)	58	20.3	48	22.6
Total power, P (mW)	1.6 mW (average)	6.2 mW (average)	24	16
# of taps, N_{taps}	16	6	4	8
DSP figure of merit ^b (fJ/sample)	3300	30	15	5
Sampler requires antialiasing filter?	No	No	Yes	Yes

^bDSP figure of merit is defined as: $FOM_{DSP} = \frac{P}{2f_{BW}2^{ENOB}N_{tans}}$.

Table 1.3. Comparison of prior CT DSP work with state-of-the-art DT DSPs.

parallelization and adopting hybrid processing domains, which have already been exploited in Ref. [10], that can improve energy efficiency at the DSP end. It is thus the contention of this thesis that, for a CT DSP to improve further towards attaining its full potential, significant improvements are needed in the performance of the accompanying CT ADCs. With this in perspective, this thesis presents techniques for CT A/D conversion/encoding that achieve two primary goals:

- 1. achieve high energy efficiency (energy/conversion-step) in the A/D encoder itself; and
- 2. drastically relax some of the constraints of the CT DSP, vis-à-vis NTPS, T_{GRAN} , and the encoding format—single-/multi-bit.

Taken together, these two goals aim to improve the energy efficiency of the composite CT

ADC/DSP system over existing ones, bringing it closer to state-of-the-art DT DSP systems. The proposed principles—three in all—exploit one or more of the design considerations discussed in Sec. 1.2.3.

Chapter 2 presents an adaptive-resolution CT encoding scheme that achieves first-order reconstruction with a very simple reconstruction circuit, thereby allowing, under certain conditions, a drastic relaxation of NTPS and T_{GRAN} over that in conventional LCS of similar specifications.

Chapter 3 presents a novel CT modulator/ADC architecture, developed for wake-up radio receiver applications, that achieves an energy efficiency comparable to that of state-of-the-art DT ADCs with similar specifications. As will be shown, the CT modulator also relaxes the constraints of the subsequent CT DSP by adopting a 2-bit encoding. The CT DSP itself is further optimized for energy efficiency by choosing operation-specific hybrid signal processing domains, as described in Sec. 1.2.3. The integrated circuit design, implementation, and measured/simulation results are presented for the composite CT ADC/DSP system.

Chapter 4 discusses the possibility of implementing CT A/D conversion using voltage-controlled oscillators. What emerges eventually is a highly-digital architecture where both the CT ADC and DSP can, in principle, be implemented using the same asynchronous digital delay. Chapter 5 describes the integrated circuit design and implementation of a CT ADC/DSP/DAC system based on this principle. Furthermore, operation-specific hybrid signal-processing domains are also chosen to improve the energy efficiency of the CT DSP.

Chapter 6 concludes the thesis and makes suggestions for future work.

Chapter 2

Adaptive Derivative Level-Crossing Sampling

2.1 Introduction

As electronic devices become ubiquitous, several applications demand signal processing and transmission with as little power dissipation as possible. For example, wireless sensor networks consist of a number of sensor nodes that sense, process, transmit, and receive information wirelessly; they must often do so under severe constraints in terms of energy usage, whether such energy is derived from a small, difficult-to-replace battery, or through energy harvesting techniques. In applications such as these, it is essential to minimize the power budget in sampling, processing, and transmission [26]. In some cases, communication may take place locally, from sensor node to sensor node in a network, and then the power budget at the receiving end is important as well. Such cases provide the context of the work reported in this chapter.

Conventional sampling and processing occur at a fixed, worst-case sampling rate, as dictated by the Shannon theorem. However, some signals have spectral properties that change significantly with time; thus a fixed sampling rate needlessly wastes samples and results in wasted energy in processing, transmission, and reception. A variety of non-uniform sampling techniques [27] can be considered for addressing this problem. Level-crossing sampling (LCS) [4], described in Chap. 1, is one such technique. In LCS, sampling is performed each time a signal crosses a threshold (Fig. 1.3). This type of sampling scales the inter-sample interval automatically depending

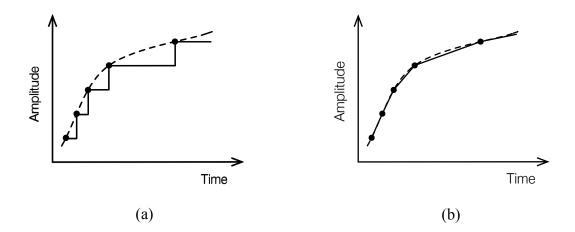


Fig. 2.1. Comparison of (a) zero-order and (b) first-order reconstruction.

on the slope of the signal; when the input is idle, no samples are wasted, without the need for elaborate power-down scenarios. We thus say that LCS automatically achieves data compression.

LCS has better quantization error properties than conventional sampling and does not suffer from aliasing [17], [1]. Techniques to process the resulting signal digitally without a clock, in continuous time, have been demonstrated, with the resulting event-driven processors offering certain advantages complementary to those of conventional processors [1].

Practical schemes for achieving LCS have been described [3], [5], [9], [23], [28]–[30]. Most are based on zero-order-hold (ZOH) reconstruction at the receiver, as shown in Fig. 2.1(a). Other schemes employ computationally intensive reconstruction techniques [22], [27]; however, in applications where the receiver is on a very tight power budget, such techniques constitute a serious overhead.

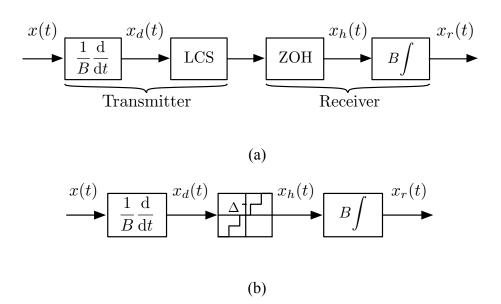


Fig. 2.2. Principle of derivative level-crossing sampling scheme: (a) actual scheme in a communication system; (b) conceptually equivalent system for analysis purposes.

One can then consider a compromise, namely piecewise-linear reconstruction as shown in Fig. 2.1(b). As can be expected from a comparison between the plots in Fig. 2.1, this can result in significantly smaller error; e.g., using a sinusoidal input and 8-bit resolution results in a signal-to-error ratio (SER) of 49 dB in the case of ZOH, and 73 dB in the case of first-order reconstruction. Conversely, for a given SER requirement, first-order reconstruction can achieve compression in the data produced by the encoder. Unfortunately, first-order reconstruction is non-causal; to know the signal value at a given instant between two samples, one needs to know the value of the sample following that instant. The corresponding storage need and computational effort can result in significant hardware overhead. First-order prediction techniques can be used to avoid the above non-causality, but those can result in discontinuities and they, too, imply a significant computational overhead.

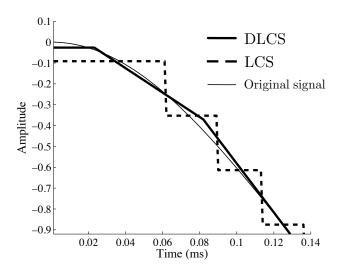


Fig. 2.3. Blow-up of DLCS (first-order) and LCS (zero-order) reconstruction for a full-scale sinusoidal input signal at 2 kHz.

This chapter discusses an LCS technique, described in Ref. [31], that automatically results in piecewise-linear reconstruction in real time, with no need for storage, meant for applications in which both the transmitter and the receiver are on a tight power budget.

2.2 Derivative Level-Crossing Sampling

The principle of the proposed system, termed Derivative Level-Crossing Sampling (DLCS), was first proposed by Pablo Martinez-Nuevo at Columbia University. However, it was refined further to its companded and adaptive-resolution forms (discussed later) by this author, resulting in a joint publication [31]. The DLCS principle is shown in Fig. 2.2(a). At the transmitter, the input is scaled and differentiated, and the result is level-crossing-sampled. At the receiver, the samples are zero-order-held and integrated, thus compensating for the differentiation. Thanks to integration, the scheme inherently achieves first-order reconstruction, leading to a lower reconstruction error, in real time, without the need of any linear predictor or non-causal techniques.

Fig. 2.3 compares the output of the system to that of an LCS system with zero-order reconstruction and to the original signal.

We note that, although the technique in Fig. 2.2(a) makes use of the signal derivative, it is very different from other schemes using derivatives for performing sampling expansions to achieve perfect reconstruction, for example in Refs. [32] and [33]. An implementation for DLCS can use an LCS quantizer [3], [9], [34] preceded by an OPAMP-based differentiator circuit. Reconstruction can be accomplished using an OPAMP-based integrator circuit.

Assuming that the direction of level crossing is taken into account as proposed in Refs. [3], [35], the operations of LCS and ZOH in Fig. 2.2(a) together are conceptually equivalent to quantization [1], [17], as shown in Fig. 2.2(b). We assume that the input signal x(t) satisfies a zero initial condition, x(0) = 0 (as in delta-modulated systems [20]), and is bandlimited to B rad/s, bounded so that $|x(t)| \le M$, where M is a positive number; using Bernstein's inequality [36] (Th. 11.1.2), we conclude that |dx/dt| is bounded by BM. Therefore, the quantizer has an input range of [-M, M]. We use a mid-tread quantizer.

Consider a signal x(t) and its reconstructed version, $x_r(t)$. The mean square error (MSE) in $x_r(t)$ can be found by comparing it to x(t), while at the same time not penalizing for amplitude, DC offset, and delay errors. Thus, the MSE can be found by minimization:

MSE =
$$\min_{a,b,\tau} (\overline{x_r(t) - (ax(t-\tau) + b)})^2$$
 (2.1)

where the overline denotes time average. Then the SER, with both signal and error as rms quantities, is given by the square root of the corresponding power ratio:

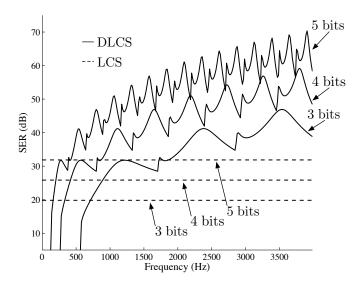


Fig. 2.4. Signal-to-error ratio (SER) for full-scale sinusoids using DLCS reconstruction; dashed lines correspond to the LCS SER: 6.02*N*+1.76 dB for *N* bits of resolution.

$$SER = \sqrt{\frac{\overline{x(t)^2}}{MSE}}$$
 (2.2)

The corresponding number of decibels is given by $20\log_{10}(SER)$.

For sinusoidal inputs, the SER can be equivalently calculated using the FFT, as the square root of the ratio of the power in the fundamental to the total power in the rest of the components (excluding DC), thanks to Parseval's theorem. Fig. 2.4 shows the SER of DLCS (solid lines) vs. frequency, for a full-scale sinusoidal input in the voice band and for different resolutions. The overall drop in SER as frequency is lowered is due to the decreasing amplitude of the derivative, $x_d(t)$, resulting in coarser quantization. The non-monotonicity of the curves is due to the fact that when the peak of the input to the quantizer changes around a quantization threshold, large local variations in the SER occur, as is the case with normal quantization [23], [37]. Classical LCS, assuming zero-order reconstruction, is equivalent to quantization [1], [17] and its SER has the

well-known value of 6N + 1.76 dB over infinite bandwidth independent of frequency [37], as shown by the broken lines in Fig. 2.4. We can see that DLCS outperforms classical LCS of the same resolution significantly over most of the frequency range.

2.3 Companded DLCS

To improve the performance at low frequencies, we explore non-uniform quantization, in which the low values of the derivative that occur at such frequencies are quantized with higher resolution. This is as is done in companding (compressing-expanding) used in telephony [37], but with two differences: we use this approach for the signal derivative, not the signal itself; and we use it only for low derivative values (using the so-called A-law, with parameter *A* set to 87.6 [37], for derivative values up to 0.18× the full scale); at higher derivative values, uniform quantization is used, in order not to sacrifice the achievable SER. Simulation results for the SER obtained with this approach, termed companded DLCS, are included in Fig. 2.7. However, this approach will not be considered further in this chapter, in view of the higher performance obtained by adaptive DLCS, which is described next.

2.4 Adaptive-Resolution (AR) DLCS

2.4.1 System Description

Going one step further from the above approach, we have made the resolution of the quantizer adaptive. As a starting point, we considered the work in Ref. [23], where the quantizer resolution is made to depend on the first derivative of the signal being quantized; it was found that this reduces the number of samples per second, while not affecting the in-band error. Since in our case we quantize the derivative of the signal, the resolution must be made to depend on the second

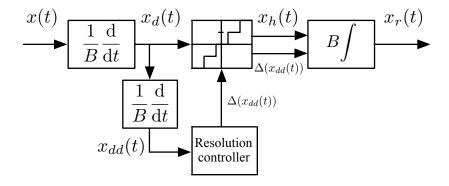


Fig. 2.5. Adaptive-resolution derivative level-crossing sampling and reconstruction principle. $\Delta(x_{dd}(t))$ denotes the variable quantization step size, which depends on the value of $x_{dd}(t)$.

derivative, as shown in Fig. 2.5. When the magnitude of the input second derivative is small, a small quantization step is used; the step size is increased as the absolute value of the second derivative increases. The algorithm used for this is different from the one used in the above reference and will be described in the next subsection. This approach results in fine resolution during intervals in which $x_d(t)$ is relatively flat, which is important because the corresponding difference in (2.1) lasts longer during such intervals, and a coarse resolution in them would deteriorate the MSE. In the resulting adaptive-resolution DLCS (AR DLCS), the resolution varies significantly; however, the average resolution can be significantly lower than the highest resolution.

In the scheme of Fig. 2.5 we need to transmit information on the quantization step size, $\Delta(x_{dd}(t))$, along with the quantized signal, $x_h(t)$, for reconstruction purposes at the receiver. As a consequence, this scheme adds some information overhead to what is being transmitted, albeit with significant benefits, as will be seen. No general discussion of the overhead caused by this can be given, as the details will depend on the protocol used. However, packet overheads in general

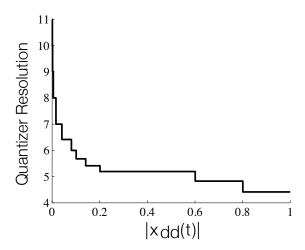


Fig. 2.6. Quantizer resolution versus the magnitude of the second derivative of the input for the system depicted in Fig. 2.5. We consider full-scale sinusoids from 0 to 4 kHz.

represent a substantial portion of the content of a packet, thus reducing the relative overhead of adding the step size information [38].

2.4.2 System Design Procedure

Using the above qualitative considerations, we have arrived at a qualitative empirical procedure for determining the law that needs to be obeyed by the resolution controller in the system of Fig. 2.5. We use sinusoidal inputs as an example; however, the procedure can be extended to other types of inputs. We now describe this design procedure.

Using the lowest-frequency input, we determine the resolution needed in order to achieve our SER target at the receiver (recall that the reconstruction is 1st-order, this being inherent to DLCS). With such lowest-frequency input, the second derivative is small throughout, hence demanding the highest resolution. We then increase the input frequency, and for portions of the input where the second derivative is large, we increase the quantization step while ensuring that the SER stays above our target. A quantization-step-versus-second-derivative-value characteristic

curve can thus be developed and locked into the system. An example is shown in Fig. 2.6; this plot was developed using the above procedure, with 60 dB SER as a target. It will be seen in the next section that this system achieves the target specification, and requires a maximum resolution of 11 bits. However, its resolution varies from 4.5 to 11 bits over the frequency band, with an average resolution of around 5.7 bits. The average resolution is calculated by first measuring the average quantization step size during quantization of sinusoids at each input frequency, and then calculating the mean of these average quantization step sizes over the entire input band.

2.5 SER Comparison

We now present simulations results for a full-scale sinusoidal input for (a) classical LCS (5- and 10-bit); (b) DLCS (5-bit); (c) companded DLCS (5-bit); and (d) AR DLCS (average 5.7-bit) using the law shown in Fig. 2.6. The type of reconstruction assumed is the one that inherently occurs in each technique, without special reconstruction algorithms, namely sample-and-hold for LCS, and first-order for DLCS and its variants. (We caution the reader that proposed systems and the presented results are relevant when the metric for accuracy measurement is MSE. Should inband SER be important, the differentiator and integrator transfer functions need to be adjusted such that differentiator amplifies and the integrator attenuates in the band of interest. This condition is not satisfied in the presented results as MSE was the metric for accuracy comparison.) Fig. 2.7 shows the SER-versus-frequency plot for the above cases. The improvement afforded by companded DLCS, compared to plain DLCS, at low frequencies is clearly visible; however, at higher frequencies the two techniques exhibit essentially the same performance, which is highly frequency dependent. In contrast to this, AR DLCS maintains a superior SER at all frequencies, and meets the 60 dB target mentioned above in conjunction with Fig. 2.6.

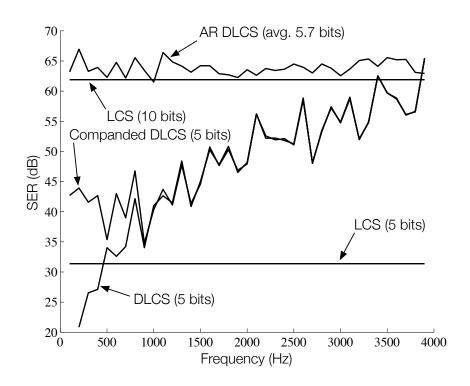


Fig. 2.7. Signal-to-error ratio (SER) for DLCS, companded DLCS, AR DLCS—where quantizer resolution varies from 4.5 to 11 bits with an average resolution of around 5.7 bits—, and LCS for full-scale sinusoidal inputs.

2.6 Sample Generation Rate and Figure of Merit

2.6.1 FOM Definition

SER alone does not provide a complete picture of performance. The rate at which samples are generated, N_s (in samples/s), is equally important as, in event-driven systems, every generated sample has an energy cost for processing and transmission. The dynamic power dissipation of the entire system is directly proportional to N_s . This, of course, does not include static power dissipation, which depends on the details of the circuit implementation. However, the power saved by transmitting fewer packets in DLCS and its variants is expected to far outweigh the static power overhead, which today can be minimized using a variety of techniques (see, for example, [9]).

Consider LCS with a sinusoidal input as a starting point, for which it is known that SER doubles for each bit of resolution increase [37], [39]. N_s is proportional to SER and to the input frequency, f. Thus, one can define the following figure of merit, FOM:

$$FOM = \frac{N_s}{SER \times f}$$
 (2.3)

For LCS, this gives a constant value independent of quantizer resolution and input frequency. For other systems, the same FOM can be used to compare them against LCS and against each other. Since we can expect the power dissipation to be roughly proportional to N_s , the above FOM is qualitatively consistent with a common FOM used to compare analog-to-digital converters [40]. The lower the FOM, the better.

For non-sinusoidal inputs, the frequency f is not well defined. For periodic inputs, it can be the inverse of the input period, but some interesting signals (see below) are not periodic; a conceivable f in such cases is the upper frequency limit of the band of interest. Since the appropriate f to be used depends on the application, we prefer to leave it as a factor in the denominator in what follows, without assigning a value to it. This will not interfere with comparisons of systems with the same input signal; f is then a common factor in their FOM values (see below).

2.6.2 Simulation Results

In Table 2.1 we compare the SER, N_s , and FOM for 6b LCS, 6b DLCS, and AR DLCS (signal-dependent average resolution) systems for several types of inputs. A considerable advantage of DLCS, and especially of AR-DLCS, over LCS, is seen in most cases. For the electrocardiogram (ECG) and speech input signals, DLCS shows very little improvement over LCS. This is because both these signals contain strong components at very low frequencies, which

Test Signal	System	SER (dB)	N_s (S/s)	$FOM = \frac{N_s}{SER \times f}$
One tone @100Hz	LCS	37.6	12,400	1.63
	DLCS	21.3	400	0.34
	AR DLCS	63.1	5,400	0.04
One tone @3.9kHz	LCS	37.6	483,600	1.63
	DLCS	66.5	483,600	0.06
	AR DLCS	62.9	226,200	0.041
	LCS	34.7	126,400	2316/f
Two tones @200Hz and 2kHz	DLCS	42.7	63,200	463.1/f
	AR DLCS	57.5	71,000	94.7/f
4kHz-bandlimited	LCS	29.1	86,000	3016/f
random Gaussian (HPF cutoff: 100Hz)	DLCS	38.2	125,100	1535/f
	AR DLCS	49.2	107,500	374/f
ECG (HPF cutoff: 0.5Hz)	LCS	27	139.3	6.22/f
	DLCS	29.1	166.1	5.79/f
	AR DLCS	50.8	180.7	0.52/f
Speech (HPF cutoff: 300Hz)	LCS	23.1	11,930	835/f
	DLCS	27	15,530	693.7/f
	AR DLCS	49.8	28,500	91.7/f

Table 2.1. Performance comparison of LCS (6b), DLCS (6b), and AR DLCS (signal-dependent average resolution) for different input signals. All inputs are full scale.

DLCS fails to quantize with enough fidelity. We have verified that for sufficiently higher quantization resolution, DLCS does become better than LCS. AR DLCS, on the other hand, shows significant improvement over LCS. All results shown in the table are for full-scale inputs. When the input amplitude is lowered, the FOM of DLCS and AR DLCS stays above that of LCS for comparable resolutions for most of the input range, except for extremely small input amplitudes, at which no levels are crossed for DLCS quantization; AR DLCS, however, continues to be better than LCS even at very low amplitudes.

2.7 Practical Considerations

Simulations show that the numbers in Table 2.1 are representative even in the presence of band-limited noise at the input, as long as the input noise power is less than the quantization error power. Addition of hysteresis reduces number of samples by avoiding excess triggering, at the expense of SER. At very high input noise, the output SER approaches that of the input SER. In a practical implementation, the differentiator should be bandlimited to avoid amplification of highfrequency noise. Comparator noise, which will limit the highest resolution in AR DLCS, can be mitigated by designing to keep it below the smallest quantization step, at the expense of power. A power-efficient alternative is the quantization-step-dependent comparator biasing scheme in Ref. [9]. Hysteresis can be introduced in the comparators in order to limit excess triggering in the presence of noise [3]. If the input to the integrator has a DC offset or very low-frequency components not present in the original signal, it can result in a local drift in the reconstructed signal that may cause a locally-escalating reconstruction error. Such a problem may be caused by asymmetries in the input signal, or comparator offsets/DAC nonlinearities which result in an asymmetric quantizer. To avoid such issues, a high-pass filter needs to precede the integrator in order to limit the low-frequency components of the quantized signal before reconstruction; such a filter was used in the above simulations (see information in the first column of the Table). With such a filter used, simulations show less than 10% degradation of SER for up to 20% offset in comparators or up to 20% deviation in the DAC output levels. Auto-zero techniques can also be used.

2.8 Conclusions

We have presented a signal-dependent sampling and reconstruction technique for bandpass signals called derivative level-crossing sampling, which inherently includes first-order reconstruction without the need of complex reconstruction schemes. Improvements to this scheme have been discussed, one using companding and another using adaptive resolution. Simulation results indicate that for certain inputs, the schemes presented can provide a significant reduction in the number of samples generated per unit time, compared to schemes based on zero-order hold reconstruction, for a given signal-to-error ratio.

Chapter 3

An Error-Shaping Alias-Free CT ADC/DSP/DAC

System

3.1 Introduction

In Chap. 1, we discussed the advantages of CT DSP over conventional schemes, along with limitations of existing CT DSP systems. In this chapter, we present a novel CT ADC architecture that allows a significant improvement over prior CT ADCs in terms of energy efficiency of conversion. The CT ADC produces a unique encoding, which relaxes the constraints of the CT DSP, which is also described. Details of integrated implementation and measurement results are presented.

While the principles underlying the resulting composite CT ADC/DSP/DAC system are general, the system itself was developed in the context of an ultra-low-power (ULP) receivers (RXs) application [41]. Such ULP RXs are characterized by extremely tight power budgets (e.g. only 100 µW in wake-up receivers [41]). This power constraint limits the RX multichannel capabilities and blocker robustness [41]. To enable scenario-dependent power and performance scalability, programmability is desirable. Consequently, ULP receivers need filtering capabilities that are programmable in terms of response type (e.g. band-pass/low-pass etc.), performance (e.g. number of taps), and specifications (e.g. passband width).

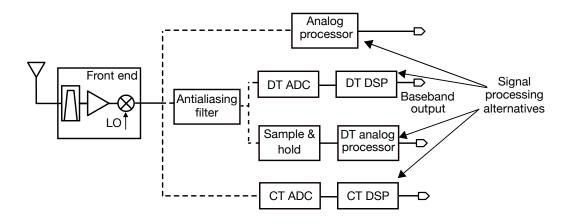


Fig. 3.1. Processing alternatives based on signal domain type in the ULP RX system context.

We now evaluate the different processor types discussed in Chap. 1 with respect to the ULP RX application (see Fig. 3.1). Analog signal processing, while power efficient, does not offer the desired programmability. DT DSP allows a high degree of programmability. However, it requires a DT ADC, which, in the context of ULP RX, needs to digitize signals in the 10 MHz - 50 MHz intermediate frequency (IF) bandwidth (bounded by the 1/f corner on the lower end and by the LO drift on the upper one [41]) with a modest resolution of about 5 bits. The power budget is limited to only few 10s of μ W. A Nyquist DT ADC may meet this constraint, but it suffers from aliasing and requires an antialiasing filter with stringent specifications, which cannot be met by a passive implementation, requiring a power-hungry active one. Oversampling can simplify the filter specifications, but the high sampling rate results in a major power overhead for the ADC and the DSP. The DT analog representation, too, suffers from aliasing, and requires an antialiasing filter. Thus, power-efficient handling of DT digital and DT analog signals is a challenge.

We then consider the CT digital domain, where a CT ADC/DSP/DAC system can process an analog input without sampling in time, i.e. in continuous time. As there is no sampling in time, no aliasing occurs and no antialiasing filter is required [3]. Therefore, unlike its DT counterparts,

analog to CT digital conversion—performed by a clockless CT ADC—can be power-efficient and with improved spectral properties [3]. The ADC output is at a non-uniform, signal-dependent rate, which can be low enough to keep the power dissipation of the subsequent blocks low. It can be processed *directly* by a clockless event-driven CT DSP [3], which preserves the timing details of the CT digital signals as they evolve in CT.

While the prospect of an alias-free CT and an event-driven digital processor is interesting, as discussed in Chap. 1, a survey of prior work in such processors reveals that these systems are limited due to the CT ADC, whose energy efficiency lags considerably in comparison with their DT counterparts. For a CT DSP to attain its potential, significant improvements are thus needed in the performance—power consumption for a given SNDR and input bandwidth—of the accompanying CT ADCs. To address this, we present a novel CT ADC architecture that enables a very power-efficient implementation [42]. In this chapter, we describe the integrated CT ADC architecture, give measurement results, and compare it with other DT/CT ADCs. Finally, the CT DSP is also described. We start by discussing the prior CT ADC art.

3.2 Overview of Existing Medium-Resolution CT ADC

Architectures

Medium-resolution CT ADCs have so far been implemented using asynchronous CT delta modulators (Fig. 3.2) [3], [9]. The comparators detect the crossings of the input with the comparison levels; the feedback DAC generates one of 2^N comparison levels each time, as needed to track the input. A CT digital signal is generated at the counter output. The scheme implemented is level-crossing sampling (LCS) [4], but in CT, without time discretization, unlike the case in

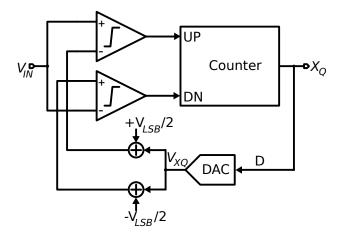


Fig. 3.2. The asynchronous delta modulator architecture used in prior work.

Refs. [4] and [5]. The comparators handle a rail-to-rail input swing; nonidealities like offsets make their design challenging. Improvements to this architecture are discussed elsewhere [9].

The loop delay in an asynchronous delta modulator needs to be smaller than the minimum time between two consecutive level crossings, the granularity, T_{GRAN} . For a single-tone input, $T_{GRAN} = (2^N \pi f_{in,max})^{-1}$ [3]. For a 5-bit ADC with a 50-MHz $f_{in,max}$, $T_{GRAN} = 200$ ps. This delay needs to be further divided between the comparators, the digital logic, and the feedback DAC of the ADC, making their design challenging under a low power budget. Therefore, although CT ADCs have been improving [3], [9], they are not as power efficient as DT ADCs.

In a uniform-resolution LCS CT ADC, any two consecutive level crossings are spaced in amplitude by one quantization step. Ref. [12] exploits this and replaces the *N*-bit DAC with a 1-bit DAC, resulting in a compact, low-power ADC. However, the implementation has substantial circuitry in the feedback path, which may cut into the loop delay and make it unsuitable for IF applications. The approach presented in this paper exploits the 1-bit feedback DAC concept while keeping the feedback path extremely simple, thereby lowering the loop delay significantly.

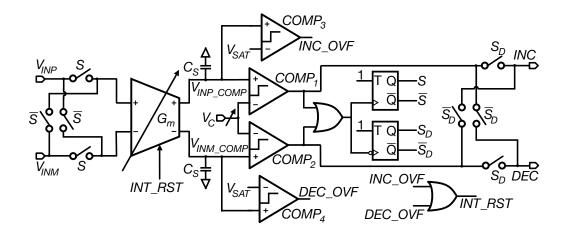


Fig. 3.3. Proposed CT ADC architecture.

3.3 Proposed CT ADC Architecture

3.3.1 Operation

In our approach, we replace the *N*-bit feedback DAC with chopping, resulting in the architecture shown in Fig. 3.3 (the relation to LCS schemes will become clearer shortly). The fully differential input (when *S* is 1) or its negative version (when *S* is 0) is fed to a G_m -C integrator through chopping switches. The comparators $COMP_1$ and $COMP_2$ detect when the outputs of the integrator, V_{INP_COMP} and V_{INM_COMP} , cross the threshold V_C . Each comparator output is connected to the INC or DEC output depending on S_D . Assume that $S = S_D = 1$. As the input V_{INP} rises from the common mode (see initial part in Fig. 3.4), the integrator output V_{INP_COMP} increases. When the latter crosses V_C , $COMP_1$ outputs a 1. The comparator output is connected to a T flip-flop through an OR gate, and this $0 \rightarrow 1$ transition on it, toggles S to 0. This flips the input switches, and hence, the polarity of the input to the integrator, causing it to charge the capacitances, C_S , in the opposite direction. This folds the integrator outputs such that now V_{INP_COMP} decreases, moving away from V_C (whereas V_{INM_COMP} starts rising towards it). Once V_{INP_COMP} goes below V_C , the output of

 $COMP_I$ (which was 1) becomes 0. This $1\rightarrow 0$ transition in the output of $COMP_I$ ($V_{INP_COMP} < V_C$) follows the $0\rightarrow 1$ transition ($V_{INP_COMP} > V_C$) after a "loop" delay due to the delays in the comparator, digital blocks, switches, and the transconductor. Therefore, the output of $COMP_I$ is a narrow pulse, which appears on the INC output to which it is connected ($S_D = 1$). The $1\rightarrow 0$ transition on the output of $COMP_I$ toggles S_D to 0 through another T flip flop, so that comparator output connections to INC and DEC are reversed (e.g. output of $COMP_2$ now gets connected to INC). Next, when a rising V_{INM_COMP} crosses V_C , $COMP_2$ makes a $0\rightarrow 1$ transition, causing another flip in S. A similar process as above generates a narrow pulse at the output of $COMP_2$, and hence, at the INC output. The cycle thus repeats. The ADC output is 2-bit pulse train—every 2-bit pulse defines an output token—and it represents the difference between INC and DEC signals, shown in Fig. 3.4. This output is CT digital and is not synchronized to any clock. The input analog information is thus encoded in the timing and polarity of the output pulses (INC/DEC).

A non-zero pulse width in the ADC output results due to a non-zero loop delay, and it is not constant, as the loop delay is not constant either. In order to ensure that no threshold crossings

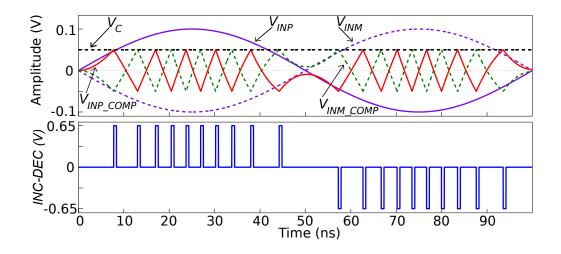


Fig. 3.4. Example waveforms of the differential input, output and some key internal signals.

are missed due to the non-zero pulse width, the latter (and hence the loop delay) needs to be lower than the minimum inter-sample time, T_{GRAN} . This condition defines the loop delay constraint for the ADC. The variable loop delay/pulse width, which will be PVT dependent, can be a source of nonlinearity [9][43]. To avoid this, we ensure that the pulse width is not an essential part of the coding scheme; as can be deduced from the description above, it is the pulses' rising edges, which represent the crossing instant, that matter. Those edges can be preserved in further pulse shaping. We, in fact, ensure this in pulse shaping for measurement purposes, as described later. Similar pulse shaping can be used if further on-chip processing of the pulses is desired.

The feedback path in the ADC is greatly simplified compared to that in CT delta modulators [3], [9], [12], as it is composed of switches and digital logic. This significantly lowers the loop delay—now primarily determined by the comparator—and allows a high speed of operation.

3.3.2 Model

We will now present a simple model for the ADC. Consider the case of a sinusoidal input. The comparators (in Fig. 3.3) detect when $V_{INP_COMP}/V_{INM_COMP}$ cross V_C , and generate an output token—a narrow pulse on INC/DEC. The polarity of the integrator input is then flipped, and the integrator outputs switch directions, resulting in the folded $V_{INP_COMP}/V_{INM_COMP}$ waveforms shown in Fig. 3.5. By mentally "unflipping" the folded waveforms, we can reconstruct the "unfolded" integrator output corresponding to V_{INP_COMP} , as shown in Fig. 3.5. This represents what the integrator output would have been, had there been no folding. Quantization levels for a mid-tread LCS quantizer [3] with $V_{LSB} = 2V_C$ and the staircase LCS-quantized version (not actually implemented) of the unfolded V_{INP_COMP} signal are shown. It is seen that the quantized version switches from one step to the next, exactly at the times the actual waveform, V_{INP_COMP} , flips.

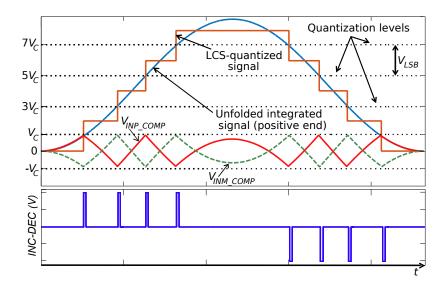


Fig. 3.5. Illustrating the development of a model for the ADC of Fig. 3.3, with a sinusoidal input (not shown). The two upper waveforms are fictitious ones (see text). The ADC generates pulses at instants where the LCS-quantized version of the unfolded integrated signal makes step-transitions.

We thus see from Fig. 3.5 that the ADC produces an output token (a pulse) at its output every time the unfolded integral of the input signal crosses a quantization level of the LCS quantizer. We can thus model the ADC as a cascade of an integrator, an LCS quantizer, and a " Δ " block that generates a narrow pulse for every transition in the quantizer output, as shown in Fig. 3.6. The polarity of the pulse depends on that of the transition: a rising transition results in a positive (*INC*) pulse whereas a falling one results in a negative (*DEC*) pulse. The Δ block thus behaves like a differentiator (with narrow pulses replacing the theoretical impulses). Note that the cascade of an LCS quantizer and the Δ block is a delta encoder [20]. Thus, the ADC produces a delta-modulated version of the input integral.

As shown in Fig. 3.6, the input signal and noise undergo an integrator transfer function (TF), with a 20 dB/decade roll-off. Note that this TF can be designed such that there is amplification in the signal bandwidth (i.e. by making its unity gain bandwidth, $f_0 > f_{in,max}$). The

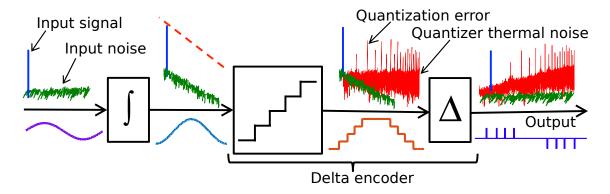


Fig. 3.6. The proposed ADC is modeled as a cascade of an integrator, a level-crossing sampling quantizer, and a Δ block (which behaves like a differentiator). The input signal and input noise components pass through an integrator-differentiator cascade and come out without frequency shaping. The quantizer adds quantization error and thermal noise, which are first-order shaped by the differentiator transfer function of the Δ block. Time waveforms are shown below the corresponding spectra.

quantizer produces a staircase LCS-quantized version of the integrated signal. In the process, it adds quantization error—the difference between the quantized and the integrated signals—and thermal noise to the integrated signal. This quantizer, since it is not accompanied by sampling, does not suffer from aliasing; the quantization error it adds thus consists of only distortion components with no spectral components in-between [1], [3]. In a DT ADC, however, sampling causes aliasing of these distortion components, resulting in additional spectral components [1]; the resulting error is often termed "quantization noise" [1]. The reader is cautioned to not confuse the thermal noise shown with such "quantization noise". The Δ block acts as a differentiator to the quantizer output and shapes it accordingly. In the overall system, the signal component and input noise go through a cascade of an integrator and an effective differentiator, coming out with no net attenuation or amplification. The quantization error, however, only goes through a differentiator transfer function and undergoes first-order shaping, which keeps the power of the baseband error components low. This fact, combined with alias-free operation, improves the baseband SNDR of the ADC. This spectral behavior has been confirmed through simulations and measurements.

Typical output spectra can be found in Sec. V (e.g. see Fig. 3.14). Such shaping is also seen in VCO-based DT ADCs [44], which are based on a different principle. The original signal can be reconstructed using a low-pass filter.

3.3.3 Design Considerations

- 1. <u>Performance tradeoffs</u>: The power of the distortion components produced by quantization (before shaping) in Fig. 3.6 [3] depends on the quantizer resolution, which is set through the threshold, V_C (recall that $V_{LSB} = 2V_C$), and the quantizer input amplitude, which depends on the transconductance, G_m . In order to enhance SNDR, V_C must be reduced and G_m increased. Both result in a lower minimum inter-sample time, T_{GRAN} , and require higher power dissipation in order to satisfy a tighter loop delay constraint. A higher G_m , depending on design, may also increase the input transistor nonlinearities in the G_m block and increase distortion (the output nonlinearities are not an issue as the G_m output swing is limited to $[-V_C, V_C]$ due to flipping; note that the swing of the mentally constructed "unfolded" integrator output in Fig. 3.5 can go beyond the supply if the integrator has enough gain). Also, the noise of the G_m stage dominates the total input-referred noise and limits SNDR. In short, the ADC performance is set through V_C and/or G_m (or with programmable capacitors, C_s), resulting in a direct trade-off with power dissipation.
- 2. <u>Overflows</u>: In a practical implementation, the non-zero loop delay results in an overshoot of the integrator output above the threshold after every crossing (Fig. 3.7). A special case of an overshoot is when the input signal changes polarity (crosses the common mode) before the integrator output can go below V_C , as indicated by "overflow" in Fig. 3.7. The change of input polarity reverses the direction of integration such that the integrator outputs move away from the comparison window. To bring the signal back, comparators $COMP_3$ and $COMP_4$ in Fig. 3.3 detect when the integrator

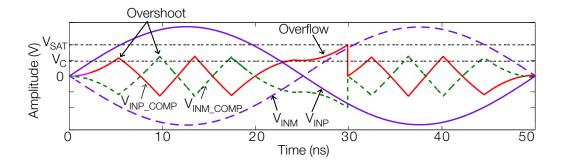


Fig. 3.7. Example showing overshoots and an overflow situation when the integrator input changes sign while its output has exceeded the comparison window set by V_C . A crossing of V_{SAT} is detected and integrator outputs are reset.

output crosses V_{SAT} (= $2V_C$) and reset its output ($IN_RST=1$) by shorting the integrating capacitors through switches (not shown).

For a given V_C , as the loop delay is increased relative to the minimum inter-sample time, T_{GRAN} , the duration for which the integrator output is outside the comparison window due to the overshoot increases, and so does the likelihood of an overflow. The same effect results as the input amplitude increases or as V_C is reduced for a given loop delay, in which case T_{GRAN} becomes smaller for a given loop delay. This imposes a constraint on the loop delay. To investigate this, we applied a full-scale (worst case) 50-MHz-bandlimited random Gaussian input to the ADC set up with a typical value of V_C (80 mV). The loop delay was then varied and the number of overflows was measured over a 100- μ s duration. No overflow was observed as long as the loop delay was under 1.4 ns. This is a rather relaxed requirement as the T_{GRAN} (~2 ns) requires a much lower loop delay. Excessive overflows result when the loop delay is close to T_{GRAN} . In such a case, for the worst-case two-tone input, the SFDR falls drastically, as will be seen in Section V (Fig. 3.18(a) for $V_C = 44$ mV). In order to avoid this, we designed the system to have a worst-case loop delay lower

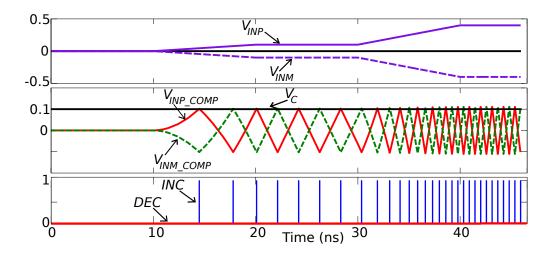


Fig. 3.8. The output token rate of the ADC increases in proportion to the input signal amplitude. A higher amplitude results in faster integration, and hence, a higher rate of threshold crossings. The polarity of the output (*INC/DEC*) depends on that of the input.

than $T_{GRAN}/5$. This, however, required a very fast CT comparator, whose design will be discussed later.

3. <u>Output token rate</u>: In a V_{LSB} -step LCS quantizer [3], the number of tokens produced per second, NTPS, for a single-tone input at frequency, f_{in} , with $A_{in,p-p}$ amplitude is given by $NTPS = f_{in} \times \left[\frac{2A_{in,p-p}}{V_{LSB}}\right]$. In the presented ADC, the input to the quantizer is the integrator output (see Fig.

3.6), whose amplitude is given by $A_{q,p-p} = A_{in,p-p} \times \frac{f_0}{f_{in}}$. Also, $V_{LSB} = 2V_C$. Therefore,

$$NTPS = f_{in} \times \left[\frac{2A_{q,p-p}}{V_{LSB}} \right] \approx \left[\frac{f_0 \times A_{in,p-p}}{V_C} \right]$$
 (3.1)

Thus, the ADC's *NTPS*—and hence its power dissipation—is independent of frequency. Rather, it increases with the input amplitude; this is seen in Fig. 3.8, where the output pulse density increases with the input signal value. This makes the circuit partially behave like a voltage-to-frequency converter (VFC) [45]. However, unlike a VFC, our ADC produces no pulses, and hence

no output tokens, when the input is zero. The resulting modulation scheme is similar to that of an asynchronous sigma-delta modulator with a three-level quantizer [46][47]. However, the proposed architecture is more power efficient than the integrate-and-reset structure of the latter (and also the VFC), as no charge is lost during flipping. The minimum inter-sample time of the ADC can be expressed as $T_{GRAN} = \frac{2V_C}{A_{In,p-p,max}\pi f_0}$.

4. <u>System-level considerations</u>: An LCS quantizer requires a pre-filter to limit the input bandwidth in order to avoid slope overload and the resulting high quantization error. In the presented ADC, the integrator acts as a pre-filter, obviating the need for a separate filter. Once the ADC is designed to handle a single-tone input at an amplitude $A_{in,max}$, it can handle a $A_{in,max}$ -amplitude single-tone input at any frequency without slope overload. The integrator can also serve as an IF gain stage in a receiver, in which case the only blocks that will require additional power will be the comparators and the logic.

3.3.4 CT ADC Integrated Implementation

A proof-of-concept chip was designed using ST's 28 nm Ultra Thin Body and BOX (UTBB) FDSOI technology. FDSOI [48] allows the use of transistor backgate bias—or the "backbias"—to lower the threshold voltage, and enables a low- V_{DD} (0.65 V) implementation. The implementation details of the transconductor and the CT comparator are discussed now.

Transconductor

The transconductor is an actively-loaded differential stage (Fig. 3.9; transistor sizing is given in Table 3.1). Its G_m can be programmed through the tail current, I_{GM} . Transistors are biased in the subthreshold regime for good g_m/I_D . The degraded linearity is mitigated through

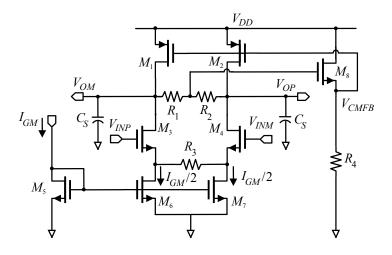


Fig. 3.9. Transconductor (G_m) -C integrator circuit.

degeneration. The CMFB circuit has a resistor common-mode voltage sensor and a source follower, whose output controls the active load gates.

Comparators

The comparator is the most critical block in the architecture as its delay dominates the loop delay. We use the inverter-based comparator architecture from Ref. [10] (Fig. 3.10(a)), which is particularly suited for a low- V_{DD} implementation. This comes at the cost of a poor power supply rejection ratio, requiring a clean supply.

Component	t Sizing/Value	
M_{1-2}	1.28 μm/400 nm	
M_{3-4}	1.8 μm/190 nm	
M_{5-7}	1.5 μm/500 nm	
M_8	4 μm/100 nm	
C_S	30 fF	
R_{1-2}	350 kΩ	
R_3	60 kΩ	
R_4	220 kΩ	

Table 3.1. Sizing of transistors and values of other components in the transconductor circuit in Fig. 3.9.

The comparator thresholds are controlled by a single, external current reference, I_{TH} . They are set in a two-phase process shown in Fig. 3.10, controlled by signals S_{1-3} . The process is initiated

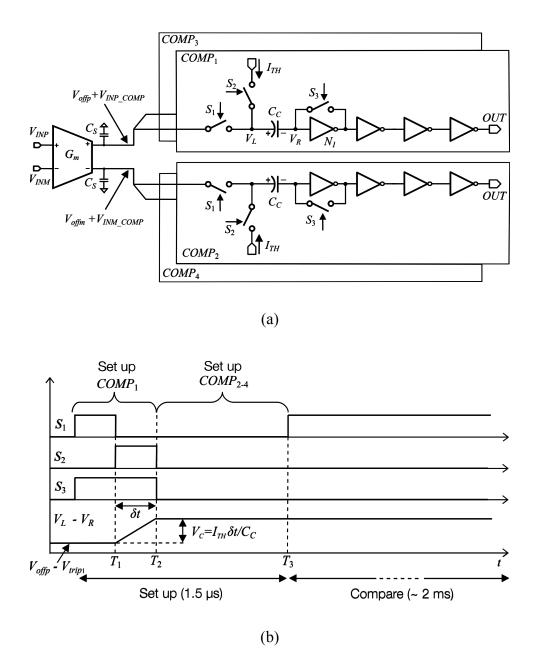


Fig. 3.10. (a) Threshold setting scheme with the comparator architecture. (b) Waveforms illustrating the threshold setting mechanism.

following a reset command, and also performs automatic on-chip calibration of the offsets at the comparator inputs and at the transconductor (G_m) outputs. We describe it now.

Let us consider comparator $COMP_1$. Phase 1 (active for $t < T_1$) is the reset phase during which the comparator and G_m offsets are stored in the voltage across the capacitor C_C . The latter will equal $(V_L - V_R)$, where V_L and V_R respectively represent the node voltages at the left and right plates of C_C (see Fig. 3.10(a)). In phase 1, control signals S_1 and S_3 are made 1, turning on the corresponding switches shown in Fig. 3.10(a), while S_2 stays 0 (see time waveforms in Fig. 3.10(b)). In the resulting circuit, V_R is shorted to the output of the inverter N_1 (in reset due to a short between its input and output nodes) with its voltage equal to the trip point of N_1 , V_{trip1} (which includes the offset of N_1), while V_L is connected to the positive end of the G_m stage output, which will ideally be at voltage V_{INP_COMP} . However, in the presence of offsets in the G_m stage, the latter will instead be at $V_{offp} + V_{INP_COMP}$ as shown in Fig. 3.10(a), where V_{offp} (V_{offm}) is the offset voltage at the positive (negative) end of the G_m output. The inputs to the G_m stage are set to zero during this phase, and thus, V_{INP_COMP} will be zero too. Consequently, the positive end of the G_m output, and hence node V_L (to which it is connected), will be at V_{offp} . The net voltage across C_C is ($V_L - V_R$), and its value at the end of phase 1 (using the above) will then be

$$(V_L - V_R)(T_1^-) = V_{offp} - V_{trip1}$$
(3.2)

This is marked in Fig. 3.10(b). The offset at the positive end of the G_m stage output (V_{offp}) and that at the comparator input (included in V_{trip1}) are thus stored in the voltage across C_C .

Phase 2 (active for $T_1 < t < T_2$) follows next; the threshold is actually set now. During this phase: S_1 goes to 0, disconnecting the G_m stage output from node V_L ; S_2 goes to 1, thereby connecting this node V_L to the terminal that sends in a (constant) current I_{TH} (see Fig. 3.10(a)); S_3

continues to remain 1, thereby keeping N_1 in reset and ensuring that V_R is a low impedance node with voltage equal to V_{trip1} . S_2 stays 1 for duration δt , and current I_{TH} charges C_C during it. Consequently, with V_R fixed at V_{trip1} , the voltage across C_C , $(V_L - V_R)$, rises (see Fig. 3.10(b)) on top of its initial value, $(V_L - V_R)(T_1^-)$, given by (3.2). Its net value at the end of phase 2 is then given by

$$(V_L - V_R)(T_2^-) = (V_L - V_R)(T_1^-) + V_C = V_{offp} - V_{trip1} + V_C$$
(3.3)

where the expression for $(V_L - V_R)(T_1^-)$ from (3.2) is used; V_C is the amount by which $(V_L - V_R)$ changes during the charging duration δt and is expressed as

$$V_C = \frac{I_{TH}\delta t}{C_C} \tag{3.4}$$

As we will see, this value represents the comparator threshold.

Phase 3 (active for $t > T_3$) represents normal operation. Here, S_1 is 1 while S_{2-3} are 0. The former connects the node V_L (the left plate of C_C) to the positive end of the G_m stage output. When an input is now applied to the G_m stage (i.e. $V_{INP} \neq 0$, $V_{INM} \neq 0$), the voltage at node V_L will be

$$V_L(t) = V_{offp} + V_{INP_COMP}(t)$$
(3.5)

where the RHS of the expression represents the voltage at the positive end of the G_m output (see Fig. 3.10(a)). Also, S_3 is 0 in this phase; this pulls inverter N_1 out of reset so that its input and output are not shorted anymore. This makes the node V_R (the right plate of C_C) floating (if we ignore the parasitic capacitor at the input of N_1). Therefore, in phase 3, the voltage across C_C , $(V_L - V_R)$, cannot change from its value at the end of phase 2, $(V_L - V_R)(T_2^-)$, given by (3.3). Given

this voltage across the C_C and that at its left plate (node V_L), the resulting voltage at its right plate (node V_R) will be

$$V_R(t) = V_L(t) - (V_L - V_R)(T_2^-)$$
(3.6)

Substituting (3.4) and (3.5) in (3.6), we get

$$V_R(t) = V_{trip1} + (V_{INP_COMP}(t) - V_C)$$
 (3.7)

This is the voltage applied to the input of inverter N_1 in the comparator (see Fig. 3.10(a)). When $V_{INP_COMP} = V_C$, from (3.7), $V_R = V_{trip1}$, and thus, inverter N_1 of $COMP_1$ will be at its trip point; its output will go towards 0 (or 1) as V_{INP_COMP} —which equals the voltage at the positive end of G_m stage output (see Fig. 3.10(a)) *minus* its offset, V_{offp} —goes above V_C (or goes below V_C). V_C thus represents the comparator threshold, and it is thus not affected by the value of the G_m stage offset, V_{offp} , or by V_{trip1} , which includes the offset of inverter N_1 , which in turn dominates the comparator offset. Therefore, we can conclude that these offsets are effectively cancelled.

The calibration mechanism described above is sequentially repeated for all comparators using the same current I_{TH} (100s of nA) and the same on-chip timing control block that sets δt . This ensures that there are no timing-skew and current-mismatch errors. Threshold accuracy is thus limited by the C_C mismatch in the comparators. In order to mitigate this, the capacitor C_C was implemented as an 80 fF MOM capacitor. Monte-Carlo simulations show a 1σ comparator offset of less than 1% of the nominal V_C value (80 mV). This mechanism thus guarantees good matching of all comparator thresholds irrespective of PVT variations.

The entire threshold setting phase takes only 1.5µs and happens once every few ms. This can be a limitation for some applications and arises due to the comparator architecture used. We

note, however, that most wireless communication systems have to be periodically calibrated to handle an always-evolving channel. A ms-range operation between calibration pauses is sufficiently long, and the thresholds can be set during the pauses. In cases where this is a concern, a larger C_C can be used to reduce the reset frequency or other comparator architectures that do not face this limitation can be considered (e.g., those in [3], [9]), while using the presented ADC architecture.

From (3.7) we see that as the comparator input $(V_{INP_COMP}/V_{INM_COMP})$ moves closer to its threshold, V_C , the input to the first inverter in the comparator gets closer to the inverter's trip point, V_{trip1} , and the crowbar current in the latter, and hence the comparator power dissipation, increases. Conversely, farther the comparator input from its threshold, lower the comparator power dissipation. The latter thus goes to a low value for a zero input (this is also why $COMP_{3-4}$ in Fig. 3.3 do not add a major power overhead); in such a case, the transconductor dissipates most of the power in the ADC.

The comparator delay decreases with an increasing input slope [9]. Post-layout simulations for a ramp input show a 480 ps - 190 ps delay drop for a 5 V/ μ s - 50 V/ μ s rise in ramp slope (Fig.

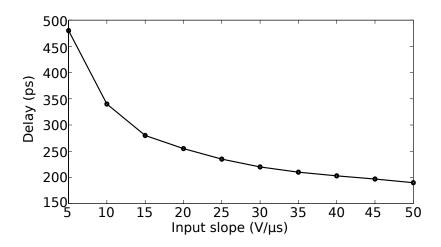


Fig. 3.11. The comparator delay drops as the slope of its input increases.

3.11). The rest of the components in the feedback path (digital logic and input switches) incur a delay of only 50 ps, resulting in a worst-case system loop delay of 240 ps.

3.3.5 Measurement Results

The core area of the chip (Fig. 3.12), implemented using ST's 28 nm FDSOI technology, including that of the threshold setting circuit, is only $45\times72~\mu\text{m}^2$ (0.0032 mm²). In this test chip, transconductor- and threshold-setting current sources are external for testing purposes. Backbiases of $\pm2~\text{V}$ and $\pm0.75~\text{V}$ (a negative value is used for PMOS transistors; see Ref. [48] for information

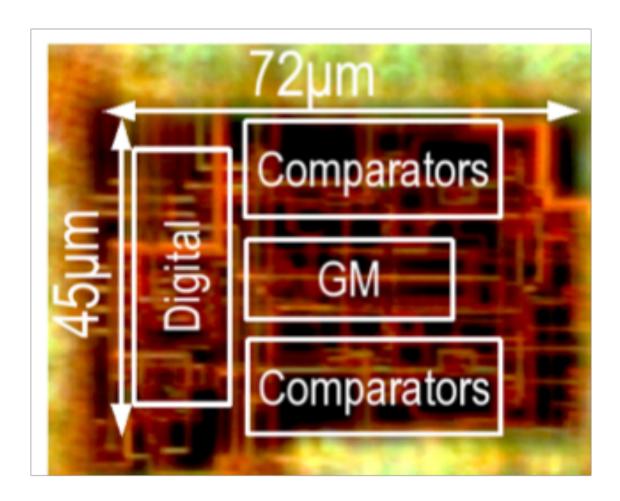


Fig. 3.12. CT ADC chip micrograph.

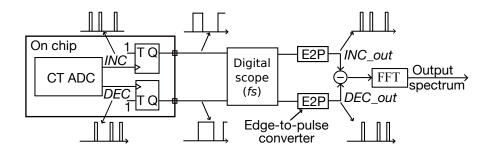


Fig. 3.13. Output extraction and reconstruction. The output spectrum is obtained by performing an FFT on the difference between *INC_out* and *DEC_out* signals.

on back-bias in FDSOI) were used for the FDSOI transistors in digital and analog sections respectively, chosen such that the transistor thresholds were lowered enough to meet the loop delay constraint. While the back-bias generators were also external, note that no current is drawn from them and the precision required of them is low, simplifying their potential IC implementation (e.g. see [48] where a low-power charge pump implements the back-bias generator on chip; note that in [48] the back-bias varies dynamically, whereas in our case it stays constant throughout). Back-bias generators can be shared between circuits on a larger chip, of which the ADC would be a part. For example, a wake-up radio containing the ADC would be integrated with a complete transceiver, which would use body biasing to advantage; we would thus benefit from the existence of the body bias generator.

The ADC output consists of narrow pulses, whose rising edge encodes the desired timing information (the falling edge, and hence, the pulse width can thus be ignored in principle). To extract them out of the chip for measurement purposes, we connected the ADC outputs to T flip-flops, which toggle for every rising edge of *INC/DEC* signals (Fig. 3.13). This extends the pulse width to the time between the rising edges of two *INC* or two *DEC* signals, and makes extraction

out of the chip possible. We used on-chip digital buffers to drive them out. Once outside, every edge (rising/falling) is converted to a pulse of fixed width, T_{PW} , thereby reconstructing the ADC

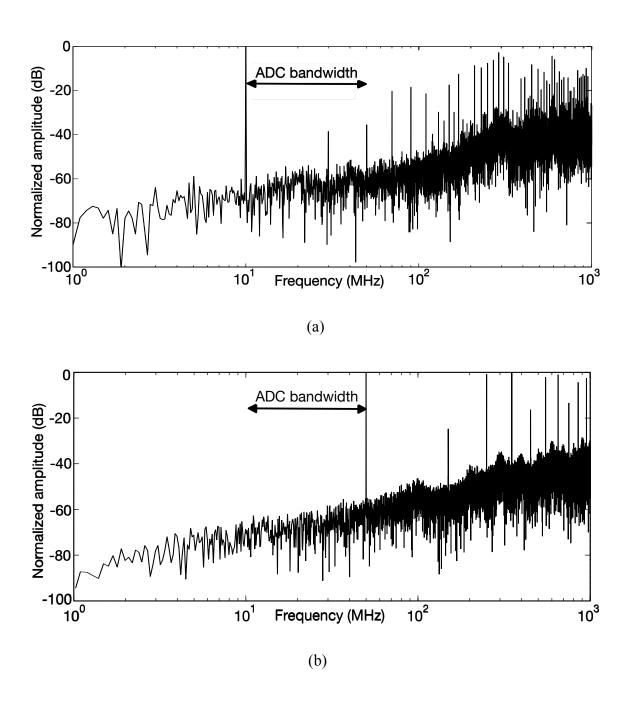


Fig. 3.14. Measured output spectra for -3 dBFS single-tone inputs at (a) 10 MHz and (b) 50 MHz, with $V_C = 80$ mV and $I_{GM} = 4$ μ A. It contains the signal component and its first-order-shaped harmonics.

outputs. This edge-to-pulse conversion is equivalent to passing the ADC's impulse output through an analog filter with an impulse response given by $h(t) = rect(\frac{t}{T_{PW}} - \frac{1}{2})$, which corresponds to a TF of $H(f) = T_{PW} sinc(\pi f T_{PW}) e^{-j\pi f T_{PW}}$. Thus, T_{PW} can be set as per the bandwidth specifications. For example, if $f_{in,max} = 50$ MHz, a $T_{PW} < 20$ ns should be used so that the in-band components will be preserved. This requirement, however, is quite relaxed as the pulse width is limited by T_{GRAN} , which is ~2 ns. To minimize timing errors, the output pulses were captured in real time with a high-speed scope (40 GS/s). An FFT was performed on the difference of the oversampled INC out and DEC out signals (Fig. 3.13) to get the output spectra.

Measured output spectra for 150 mV_{p-p} (-3 dBFS) single-tone inputs at 10 MHz and 50 MHz are given in Fig. 3.14. These tests were carried out using $V_C = 80$ mV and $I_{GM} = 4$ μ A. As expected, the output spectrum contains the signal component, along with its first-order-shaped odd harmonics and thermal noise. Alias-free operation is confirmed through an out-of-band test tone at 60 MHz; the output spectrum (Fig. 3.15) shows no degradation due to noise or aliasing. The

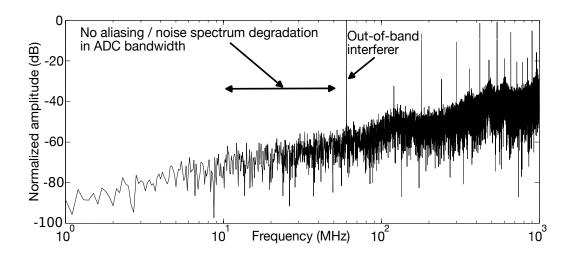


Fig. 3.15. An out-of-band test tone at 60 MHz does not result in any degradation due to aliasing or increased noise ($V_C = 80 \text{ mV}$ and $I_{GM} = 4 \mu \text{A}$).

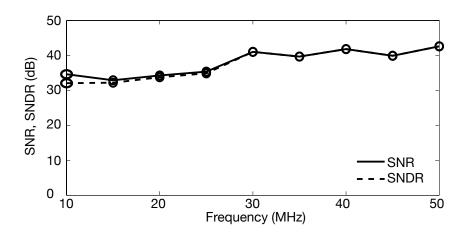


Fig. 3.16. Plot of single-tone SNR/SNDR versus input frequency. The input amplitude is -3 dBFS, and $V_C = 80$ mV and $I_{GM} = 4$ μ A.

SNR (SNDR) is measured by integrating the noise (and distortion) over the 10 MHz - 50 MHz band. The SNDR-vs.-input-frequency plot for -3 dBFS single-tone inputs is given in Fig. 3.16.

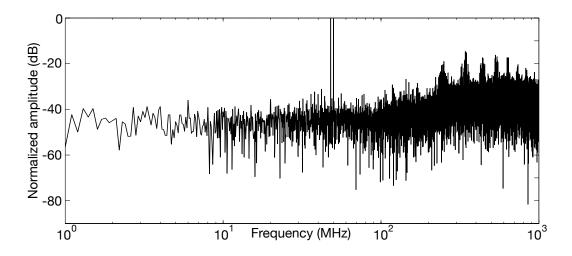


Fig. 3.17. Two-tone output spectrum; the input tones are at 48 MHz and 50 MHz; $V_C = 80$ mV and $I_{GM} = 4 \mu A$. The output consists of signal components and IM products. The low-frequency noise floor does not show first-order shaping, and is attributed to the input noise from the two-tone signal generator, which was different from the one used for single-tone tests.

The constant input amplitude results in a constant power consumption of 24 μ W, independent of input frequency. The maximum measured ADC output token rate is a modest 200 MS/s.

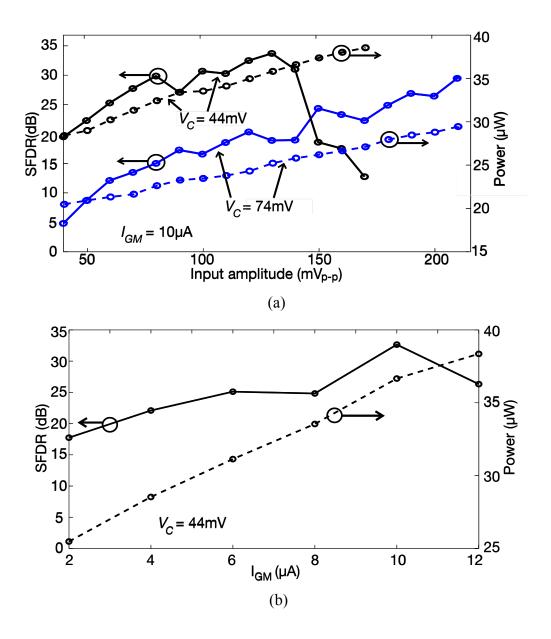


Fig. 3.18. SFDR measurements for a two-tone input with two tones at 48 MHz and 50 MHz. V_C and I_{GM} are changed from their nominal values to demonstrate programmability. (a) SFDR and power dissipation vs. the input amplitude for different V_C values with $I_{GM} = 10 \mu A$; (b) SFDR and power dissipation for different values of I_{GM} with $V_C = 44 \text{ mV}$ and a 130 mV_{p-p} input.

For a two-tone input, the output spectrum consists of signal components and first-order shaped intermodulation products (Fig. 3.17). In order to demonstrate programmability, the threshold, V_C , and the transconductor bias current, I_{GM} , are changed from their nominal values. The SFDR and power dissipation for a two-tone input is plotted in Fig. 3.18(a) for two different V_C values. We see that V_C can be programmed, at the expense of power, to maintain an SFDR >30 dB over a wide amplitude range, potentially easing the IF AGC in a wake-up receiver. The fall in the SFDR plot for $V_C = 44$ mV at high input amplitudes is due to excessive overflows due to a lower T_{GRAN} for a fixed loop delay. Such programmability can also be obtained through I_{GM} (Fig. 3.18(b)). Performance can thus be traded off for power dissipation based on signal conditions. Power dissipation decreases with decreasing input amplitude. Power consumed for a zero input is 8 μ W.

The two-tone test was repeated for different back-biases of the digital section transistors (Fig. 3.19). A 0 to ± 2 V change in the backbias drops the delay of the digital section from 83 ps

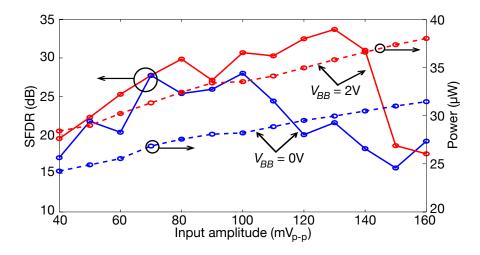


Fig. 3.19. Effect of back-bias (V_{BB}) used for digital circuits on ADC performance. A higher back-bias lowers delay and offers better linearity at the expense of power dissipation. Test set up is the same as that used to generate Fig. 3.18(a), but with different values of V_{BB} (V_{BB} is the absolute value of the back-gate bias, the latter being positive for NMOS and negative for PMOS).

	Yoshioka [74]	Tsai [75]	Van der Plas [76]	Brooks [77]	This Work (27 °C)
Technology	40 nm CMOS	90 nm CMOS	90 nm CMOS	180 nm CMOS	28 nm UTBB FDSOI CMOS
Supply (V)	0.7	1	1	1.8	0.65
Input bandwidth	12.3 MHz	20 MHz	75 MHz	100 MHz	40 MHz
Sampling rate	24.6 MS/s	40 MS/s	150 MS/s	200 MS/s	No sampling
Core area (mm ²)	0.0058	0.055	0.0625	0.05	0.0032
SNDR (dB)	44.2	44.5	40	40.3	32-42
Total power (µW)	54.6ª	113ª	133ª	8500 ^{a,b}	24° (8 µW standby)
Figure of Merit (fJ/conv-step)	17	20	10.9	503.3	3-10
$P/f_s(pJ)$	2.2	2.8	0.88	42.5	0.3 ^d
Antialiasing filter required?	Yes	Yes	Yes	Yes	No

^aDoes not include the power dissipation of the antialiasing filter, and that required for clock generation.

Table 3.2. Comparison of the proposed ADC with sampled ADCs with bandwidths ≤100 MHz and modest SNDR values.

to 40 ps, per post-layout simulations. The faster feedback reduces loop delay and increases the input amplitude at which overflows cause the SFDR to fall drastically.

3.3.6 Comparison of CT ADC with the State of the Art

Table 3.2 compares our ADC with other state-of-the-art DT ADCs that have a bandwidth ≤ 100 MHz and similar modest SNDR values, and Table 3.3 does it with other CT ADCs (FOM figures are discussed below). The presented ADC occupies a very low area, owing to both the

^bDoes not include the power dissipation for the generation of reference currents/voltages for biasing and threshold setting.

^cDoes not include the power dissipation for the generation of reference current (used to generate I_{GM} (<12 μ A) and I_{TH} (100s of nA)) and backbiases, which are assumed shared with other circuits on the same chip.

 $^{{}^{\}rm d}f_s = 2 \times \text{Input bandwidth}.$

	Schell [3]	Kurchuk [10]	Weltin-Wu [9]	This Work (27 °C)
Technology	90 nm CMOS	65 nm CMOS	130 nm CMOS	28 nm UTBB FDSOI CMOS
Supply (V)	1	1.2	1	0.65
Input bandwidth	10 kHz	2.4 GHz	20 kHz	40 MHz
Core area (mm ²)	0.06	0.0036	0.36	0.0032
SNDR (dB)	58	20.3	47-54	32-42
Total power (µW)	50 ^a	2700 ^a	2-8 ^a	24 ^b (8 μW standby)
Figure of Merit (fJ/conv-step)	3769	66	200-850	3-10
$P/f_s(pJ)^c$	2500	0.56	200	0.3

^aDoes not include the power dissipation for the generation of reference currents/voltages for biasing and threshold setting.

Table 3.3. Comparison of the proposed ADC with other CT ADCs.

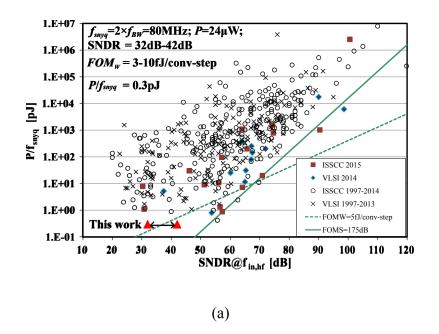
advanced technology used and its simple circuitry. It does not require an antialiasing filter, unlike DT ADCs. Its power consumption is quite competitive compared to the DT ADCs, despite the latter not including the power dissipation of the antialiasing filter. We do not include the power dissipation for the generation of the back-biases and the reference currents I_{GM} (<12 μ A) and I_{TH} (100s of nA), as they will be shared by the complete CT ADC/DSP/DAC system.

To compare the ADCs [49], we use two metrics: the Walden FOM for the core ADC, $FOM_W = P/(2^{ENOB} f_{snyq})$ and the energy per sample, P/f_{snyq} , where P is the power dissipation of the core ADC; ENOB is the effective number of bits, calculated as (SNDR - 1.76)/6; and f_{snyq} is the Nyquist sampling frequency. CT ADCs do not have a sampling frequency; thus, for comparison, we define $f_{snyq} = 2f_{BW}$, where f_{BW} is the ADC input bandwidth (40 MHz). The presented ADC

^bDoes not include the power dissipation for the generation of reference current (used to generate I_{GM} (<12 μ A) and I_{TH} (100s of nA)) and backbiases, which are assumed shared with other circuits on the same chip.

 $^{^{}c}f_{s} = 2 \times \text{Input bandwidth}.$

achieves a core FOM of 3-10 fJ/conv-step and a P/f_{snyq} of 0.3 pJ. The FOM improvement over the CT ADCs in [3], [9], [10] is respectively $>300\times$, $>20\times$ and $6-22\times$. Fig. 3.20 compares the



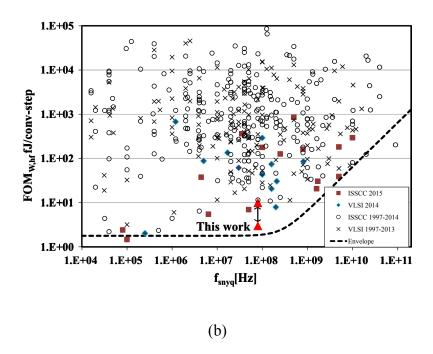


Fig. 3.20. Comparison of the presented CT ADC with state-of-the-art ADCs in the Murmann survey w.r.t. (a) energy per conversion; and (b) Walden figure of merit.

proposed ADC with state-of-the-art ADCs from the Murmann survey [49]. Clearly, the ADC also achieves competitive performance relative to DT ADCs in terms of core FOM, P/f_{smyq} , and area. It thus presents a significant step in the development of CT data conversion.

The high power dissipation of CT ADCs has in the past been a bottleneck in the development of CT DSP systems. We have proposed a CT ADC architecture that allows a highly power-efficient and compact implementation. The ADC is alias-free with first-order quantization error spectral shaping; has power dissipation that scales automatically with input amplitude; and has a low output token rate that will ensure low power dissipation in a subsequent event-driven CT DSP. Its programmability allows performance to be traded off for power depending on signal conditions. Overall, the proposed ADC presents a major advance in the development of CT ADCs and paves the way for consideration of CT DSP as an interesting mode of flexible signal processing, and it will be discussed next.

3.4 CT DSP

We now consider the CT DSP of the output of the CT ADC (see Fig. 3.1 for system-level view) described in the previous section, in the context of providing interferer rejection in ultra-low-power radios [41] in the intermediate frequency band of [10 MHz, 50 MHz]. The specifications of such a filter have been derived in detail in Ref. [50], and according to it, we need a bandpass filter with a:

- Tunable center frequency, f_c , in the [10 MHz, 50 MHz] band;
- Passband width of 2 MHz;
- Stopband starting at 2 MHz from the center frequency with a stopband rejection >30 dB.

These specifications are summarized in Fig. 3.21(a), with 50 MHz as an example f_c . While a

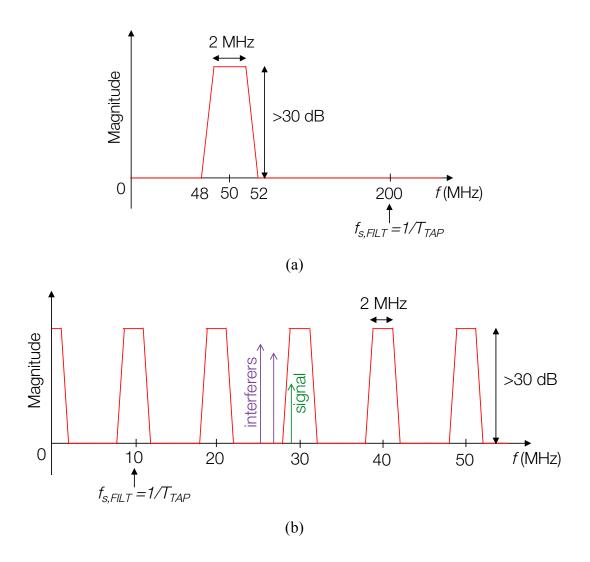


Fig. 3.21. (a) Specifications for the desired filter frequency response with an example center frequency of 50 MHz and an $f_{s,FILT}$ of 200 MHz; (b) frequency response of (a) for an $f_{s,FILT}$ of 10 MHz.

stopband rejection of 20 dB will suffice for the application [50], we aim for 30 dB in order to account for the circuit-level non-idealities. These non-idealities have been thoroughly simulated as described in Ref. [50], and a set of CT DSP constraints have been derived. We use them and design the different CT DSP blocks within the bounds set by them.

The specifications described were fed into the fdatool in MATLAB along with the tap

delay, T_{TAP} to synthesize the desired filter transfer function of FIR (traversal) form and to arrive at the required number of filter taps, N_{taps} . From (1.1) we know that both N_{taps} and T_{TAP} determine the power dissipation of the DSP. Recall from Chap. 1 that T_{TAP} defines the repetition frequency of the filter frequency response, given by

$$f_{s,FILT} = \frac{1}{T_{TAP}} \tag{3.8}$$

As an initial choice, T_{TAP} was chosen to be 5 ns so that $f_{s,FILT}$ (200 MHz) is higher than the input bandwidth of 50 MHz. The filter center frequency, f_c , was chosen to be 50 MHz. A direct synthesis of a bandpass transfer function using fdatool then reveals that the filter order required to satisfy these specifications would be 165. This is a very high number and will impose an overwhelming power penalty. The required filter order is high on account of the high value of the ratio of $f_{s,FILT}$ to the filter passband (2 MHz). Therefore, in order to lower the required filter order for the given passband, $f_{s,FILT}$ has to be lowered.

To achieve this, we make the following observation. In an uncertain-IF wake-up radio receiver [41][50], the front-end BAW filter limits its input signal (which contains the interferers) to a 10-MHz bandwidth around the carrier frequency; this input is then mixed down to an uncertain IF bandwidth in the [10 MHz, 50 MHz] range. While the signal and the interferers reside somewhere in this range (hence the name "uncertain"), their bandwidth is only 10 MHz. Now, if we synthesize a low-pass filter with $f_{s,FILT}$ = 10 MHz (or T_{TAP} = 100 ns), -3dB cut-off frequency of 1 MHz (equal to half of the desired passband width of the bandpass filter: 2 MHz) and the same stopband frequency (2 MHz) and attenuation (>30 dB), the required filter order is 9 (which will have 10 taps), and the resulting frequency response is shown in Fig. 3.21(b). The repetition of the frequency response every $f_{s,FILT}$ (= 10 MHz) creates a bandpass frequency response with multiple passbands centered at integer multiples of 10 MHz, each with a passband width of 2 MHz and with

Parameter		Value	
	Order	9	
Filter	Response type	Tunable bandpass	
	Center frequency	10 MHz – 50 MHz	
	Passband width	2 MHz	
	Stopband rejection	> 30 dB	
	Delay range	66 ns – 100 ns	
Tap delay	Delay mismatch, 1σ	< 0.7%	
	Delay jitter, 1σ	< 0.3%	
Multiplion	Coefficient resolution	3 bits	
Multiplier	Coefficient mismatch, 1σ	< 9%	

Table 3.4. Table summarizing the specifications of the FIR filter and those of its individual blocks.

>30 dB stopband attenuation. The presence of multiple passbands in the [10 MHz, 50 MHz] bandwidth is not an issue since any two passbands are separated along the frequency axis by at least 8 MHz (see Fig. 3.21(b)) and, thanks to the front-end bandlimiting filter, the signal and interferers can be ensured to not occupy a bandwidth more than that of one lobe (including its passband and stopband). By tuning T_{TAP} from 100 ns to 66 ns, the center frequencies of the lobes can be made to cover the entire [10 MHz, 50 MHz] bandwidth. Therefore, a practical filter order is achieved while guaranteeing sufficient degree of interferer rejection over the entire bandwidth. Ref. [50] further details system-level simulations to derive individual specifications for different blocks in the CT DSP: tap delays, multipliers etc. They are summarized in Table 3.4. Now that we know these, we consider the filter's integrated design.

⁹ In some cases, a high-pass filter (instead of a low-pass one) may be necessary to ensure the lobes cover the entire signal band.

3.4.1 Integrated Implementation

In this section, we discuss the integrated implementation of a programmable 9^{th} -order CT FIR DSP (N_{taps} = 10) that can interface with the CT ADC described previously in this chapter. The tap delay can be tuned from 100 ns to 66 ns, and the filter order can be programmed by selectively turning off unwanted delay taps.

The CT ADC outputs a 2-bit pulse stream (see Fig. 3.4), which will be processed by the CT DSP: it will be delayed along a tapped delay line, multiplied with coefficients, and then added (see Fig. 1.6). We know that the ADC output has an average token rate, \overline{NTPS} , of 200 MS/s, and a minimum intersample time, T_{GRAN} , of 2 ns. The latter would mean that, in order to preserve the timing details of the ADC output, every 100-ns tap delay in the DSP will be implemented as a cascade of 50 delay cells, each with a delay of 2 ns (see Fig. 3.22). Using (1.1) and assuming from Ref. [3] that a delay cell dissipates E_{Del} =35 fJ for every delay operation, we can estimate that the delay line alone in such a 10-tap filter will consume 3 mW. This is clearly way too high for the given application, which allows a meagre power budget of about 100 μ W. Therefore, to meet such

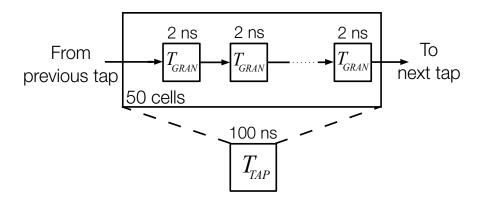


Fig. 3.22. For a minimum intersample time, T_{GRAN} , of 2 ns, each tap delay, T_{TAP} , of 100 ns is implemented using a cascade of 50 2-ns digital delay cells.

a challenging specification, we adopt two solutions: (a) parallelization in the delay line to relax T_{GRAN} [10]; and (b) improvement in the energy efficiency (i.e. lower E_{Del}) of the delay cell over that in Ref. [3]. They will be discussed next.

Delay line parallelization

As discussed in Chap. 1, every tap delay in the CT DSP is implemented as a cascade of unit delay cells, each with a delay of T_{GRAN} . Each ADC output token (or each pulse) that is fed into the delay line in the DSP then goes through each cell, and the resulting power dissipation increases in proportion to the number of such cells. This power consumption can thus be reduced by lowering the number of delay cells in any given tap. This is achieved by using parallelization in the delay line [51] as shown in Fig. 3.23.

Let N_P be the number of parallel paths in the delay line. Consider that at a certain point in time, the CT ADC output token is sent along the first (upper-most) delay line path in Fig. 3.23. The next ADC token will then be sent along the second path; the one after that will be sent along

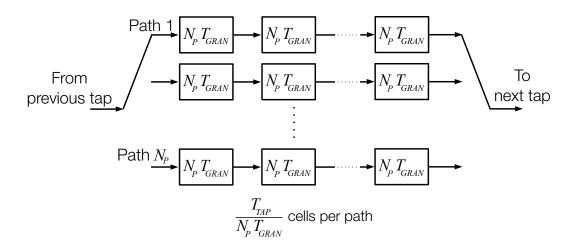


Fig. 3.23. A tap delay in a parallelized delay line with N_P parallel paths. Each path of the tap delay is implemented as a cascade of delay cells, each with a delay of $N_P T_{GRAN}$. The input to the tap is connected to one path at a time and circulated in a round-robin fashion.

the third path, and the process will continue until we reach the $(N_P+1)^{th}$ token, which will be sent along the first path, and the cycle repeats. The ADC tokens will thus be fed into the parallel paths in a round-robin fashion. A demultiplexer (not shown) performs this task. At each tap, the parallel delay line outputs are combined using an OR gate (not shown) and fed into the multiplier.

Due to the round-robin rotation of the input connection, for any path in the parallelized delay line, the minimum time between any consecutive input tokens it receives is increased by N_P times compared to that for the single-path delay line in Fig. 3.22. For instance, if $N_P = 5$, this minimum time will increase from 2 ns to 10 ns. The number of delay cells required to implement one path in a given parallelized tap delay will then be lower by N_P times compared to that in the single-path system in Fig. 3.22. For example, in Fig. 3.23, if $N_P = 5$, and hence $N_P T_{GRAN} = 10$ ns, each 100-ns tap delay can be implemented using 5 parallel delay line paths, where each path is in turn implemented as a cascade of 10 unit delay cells, each with a delay of 10 ns. With 10 cells per path, and 5 paths in all, each parallelized tap delay of Fig. 3.23 will have a total of 50 cells, just like the single-path one in Fig. 3.22. The difference, however, is that in the former every input token goes through only 10 delay cells (10-ns each) to undergo a 100-ns tap delay, as against 50 of them (2-ns each) in the latter for the same tap delay. Assuming the energy per delay operation, E_{Del} , is equal for the 10-ns and 2-ns unit delay cells¹⁰, the total energy dissipated in delaying a token along the parallelized version of the 100-ns tap delay (Fig. 3.23) will then be $50/10 = 5 \times$ lower compared to that in the single-path version of the same (Fig. 3.22). For a given average input token rate and a given E_{Del} , we then estimate from (1.1) that the parallelized delay line will

¹⁰ The energy dissipated by the delay cell per token is assumed independent of the delay value [25]. This happens when the delay is tuned by changing the charging current, while keeping the capacitor fixed. We will confirm this in simulations for our delay cell design later.

dissipate $5 \times (N_P \text{ times})$ lower power than the single-path one in Fig. 3.22. Using the same numbers for \overline{NTPS} and E_{Del} as before, the parallelized delay line is estimated to consume a power of 630 μ W. This, too, is much higher than the system power budget. We will lower it further by using an energy-efficient delay cell architecture (that will lower E_{Del}), described later.

While parallelization lowers the power dissipation of the delay line in proportion to the number of parallel paths, N_P , it does so at the expense of an increased sensitivity to mismatch and a higher jitter¹¹. Therefore, there exists an optimal value of N_P that maximizes energy efficiency without compromising the jitter performance (mismatch is handled via calibration, described later). Thorough system-level simulations were carried out (described in Ref. [50]) and it was concluded that a choice of $N_P = 5$ is optimal. The delay line in the CT DSP thus has 5 delay line paths for each ADC output (10 paths for *INC* and *DEC* combined); each path has 9 tap delays, with each composed of 10 10-ns unit delay cells (900 cells in all). At each tap, the outputs of all five paths are combined and fed into the multiplier at that tap. We next discuss the design of the unit delay cell that implements a delay of 10 ns.

Delay cell design

The asynchronous digital delay cell architecture is based on the one from Ref. [25] with a few important modifications done to improve the energy efficiency of the cell. A conceptual schematic (with signal waveforms) for the delay cell architecture presented in Ref. [25] is shown

¹¹ Under the above assumption of the energy/operation in the delay cell being independent of the delay value, the area of the unit delay cell does not change between the single-path and the parallelized delay line. Then the total delay line area also remains unchanged [50], [70].

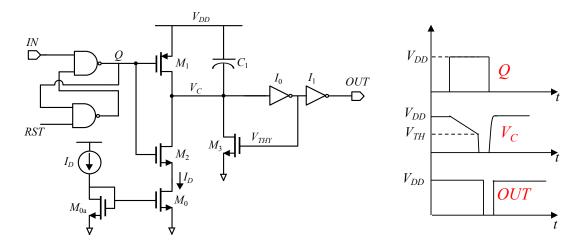


Fig. 3.24. Principle of the delay cell presented in Ref. [25] along with example time waveforms.

in Fig. 3.24. During reset, Q is 0, and thus, M_1 is on, M_2 is off, and the capacitor C_1 is completely discharged. The cell output, OUT, is 1 (at V_{DD}). Following a low input pulse (not shown), Q becomes 1 as shown, and thus, M_1 turns off, and M_2 turns on. The current in transistor M_0 , I_D , then charges C_1 so that node V_C falls as shown. Once V_C reaches a certain threshold, V_{TH} , a positive feedback loop composed of M_3 and I_0 is triggered causing V_C to drop quickly to ground. The cell output consequently becomes 0 and triggers the next cell in the delay line. An acknowledge signal (not shown) from this next cell, adopted as part of a handshaking protocol, then resets the current cell by discharging C_1 through M_1 . The total delay, T_D , of the cell is given by

$$T_D = \frac{C_1(V_{DD} - V_{TH})}{I_D} \tag{3.9}$$

The energy consumed by the delay cell per operation is composed of two parts: (a) the energy dissipated during the discharging of capacitor C_1 , from a voltage of V_{DD} across it (to 0), during reset—it is equal to $C_1V_{DD}^2$; and (b) the energy dissipated due to switching (and crowbar) in the digital gates in the circuit, which is also proportional to V_{DD}^2 . Since both components of the

energy consumption depend quadratically on the supply voltage, V_{DD} , our choice of $V_{DD} = 0.65$ V allows a lowering of overall energy consumption over that in Ref. [25], which has a 1-V supply. Typically, C_1 is larger than the parasitic capacitances in the digital gates; (a) thus dominates the total energy per event of the cell. During every delay operation, each node in the delay cell undergoes the same amount of voltage change. For instance, V_C always goes from V_{DD} to V_{TH} (and only the slope of its fall changes; see Fig. 3.24). The net amount of charge drawn from the supply for every delay operation is thus constant and is independent of the value of the charging current, I_D . Thus, the total energy dissipated by the cell per delay operation is independent of the I_D . If the latter is then used to tune the delay of the cell (using (3.9)), as do we in our cell, the energy per delay operation will also stay constant across the range of delay values achieved [25].

The delay cell architecture used in the proposed system is shown in Fig. 3.25 (transistor sizing given in Table 3.5; all transistors have their back-bias terminals connected to ground). Capacitor C_1 is implemented using a MOM capacitor. Example waveforms for some internal signals are shown in Fig. 3.26. A NAND SR-latch holds the state of the delay cell (either reset or

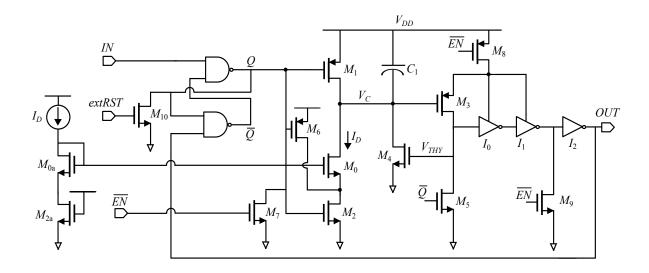


Fig. 3.25. Architecture of the delay cell used in the proposed CT DSP.

Component	Sizing/Value	
M_0 , M_{0a}	200 nm/2000 nm	
$M_{1-2}, M_{4-10}, M_{2a}$	80 nm/30 nm	
M_{5-7}	1.5 μm/500 nm	
C_1	2.11 fF	

Table 3.5. Sizing/values of different components in the delay cell shown in Fig. 3.25.

delay). A low pulse at the input triggers the cell into delay mode, and V_C starts falling (as shown in Fig. 3.26); when it hits the threshold (not shown), buffers I_{0-2} pull the output, OUT, to ground. The latter triggers the next delay cell in the delay line and also resets the current one through the SR latch. No handshaking is thus adopted. Following such reset, V_C is pulled back up to V_{DD} and M_0 goes into the cut-off region.

In contrast to what is done in Ref. [25], V_C does not go all the way to ground before this pull up (contrast the plots of V_C in Figs. 3.24 and 3.26). This happens because, in the proposed

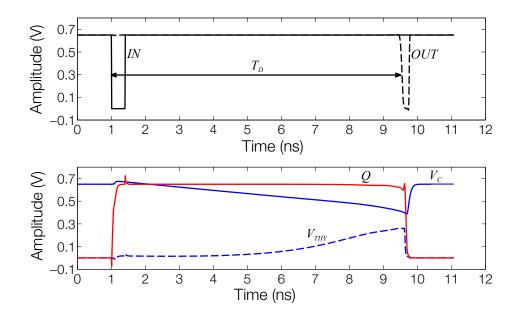


Fig. 3.26. Time waveforms for some key signals in the delay cell in Fig. 3.25.

cell, M_4 is intentionally made weak enough so that the positive feedback loop between M_3 and M_4 is never triggered¹². Due to this, the fall of V_C (shown in Fig. 3.26) after it crosses the threshold continues at the same slow rate as that before, until reset when it is pulled back to V_{DD} ; it does not sharpen as that in the cell in Fig. 3.24 [25], where the positive feedback loop formed by M_3 and I_0 is triggered and V_C undergoes a fast fall to ground. Note that during the slow fall of V_C , no crowbar current flows from the drain of M_3 to the source of M_5 , as the latter is off during this time.

The choice of not letting V_C fall all the way to ground is intentional as now the energy consumed during the discharge of the C_1 , from a voltage close to $(V_{DD} - V_{TH})$ across it (to 0), is about $C_1(V_{DD} - V_{TH})^2$, which will be lower than $C_1V_{DD}^2$ —the value for the delay cell in Fig. 3.24. The delay value of the cell is also given by (3.9), and it can be tuned through the charging current, I_D . As discussed above, if delay tuning is done using the latter, the energy dissipated by the cell

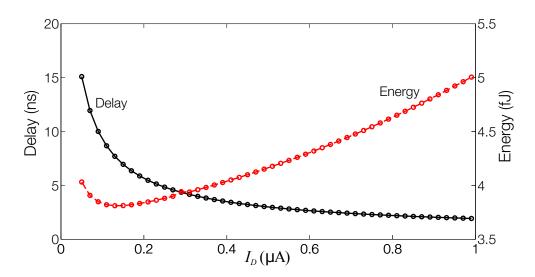


Fig. 3.27. Plot of the delay and energy/operation versus the charging current, I_D , for the delay cell in Fig. 3.25.

¹² In hindsight, M_4 is thus redundant and can be removed. We correct this in the delay cell described in chapter 5.

_

per delay operation is fairly independent of the delay value [25]. This is confirmed by simulations. Shown in Fig. 3.27 is the plot of the delay value and energy per operation of the delay cell against the charging current, I_D . The latter varies from 3.8 fJ to 5 fJ (a $\pm 13\%$ variation around the average: 4.4 fJ) over a delay variation of 2 ns to 15 ns. The low value of the energy dissipation is due to a low supply voltage of 0.65 V and the architectural choices described above. The leakage power of the delay cell for an inactive input is 4.6 nW; it can be disabled using transistors M_{7-9} , following which it will dissipate 0.6 nW.

The performance of this delay cell is summarized and compared with prior work in similar asynchronous delay cells in Table 3.6. Relative to the delay cells in Refs. [3], [10], [16], its per token energy dissipation is improved by $8\times$, $2.7\times$, and $11\times$, respectively. With this value of E_{Del} (4.4 fJ), the estimated power dissipation of the parallelized delay line for the specifications we have been discussing so far will be 79 μ W, which is 38× lower than our previous estimate for the single-path delay line and $8\times$ lower than our previous estimate for the parallelized delay line, both

	Schell [3]	Kurchuk [10]	Vezyrtzis [16]	This Work
Technology	90 nm CMOS	65 nm CMOS	130 nm CMOS	28 nm CMOS
Supply (V)	1	1.2	1	0.65
Delay range	22 ns – 280 ns	100 ps - 300 ps	15 ns – 500 ns	2 ns – 15 ns
Energy/token	35 fJ	11 fJ – 13 fJ	50 fJ	3.8 fJ - 5 fJ
Power dissipation with inactive input	-	-	-	4.9 nW
Power dissipation when disabled	14.2 nW	-	2.3 nW	0.6 nW
Delay mismatch, 1σ	-	-	-	7.4%

Table 3.6. Summary of the delay cell performance and comparison with other similar delay cells in prior art.

One path (of the 5) in a delay tap From previous tap 10 ns 10 ns

Fig. 3.28. One path (of the 5) in a delay tap along with the additional calibration circuitry: 2 10-ns (coarse) delay cells, 5 1-ns (fine) delay cells, and multiplexers for programmability.

assuming the delay cell was based on the one in Ref. [3]. More important is the fact that this will be within the system power budget. Besides, the leakage power dissipation is also very low. However, mismatch in the delay cell will impair the performance, and it needs calibration. The delay calibration scheme is described next.

Tap delay calibration

System-level simulations require that the variations in the 100-ns tap delay due to mismatch should have a 1σ value lower than 0.7%, or 700 ps, (see Table 3.4) in order to maintain desired performance [50]. This cannot be achieved by a delay tap composed of the proposed cell by default. Calibration is thus necessary.

As we saw in Fig. 3.27, the delay of an individual cell can be tuned by changing the bias current I_D . However, individually calibrating every delay cell in every tap (900 cells in all) using the bias current is impractical. Therefore, we only use a common bias current I_D to set the bias currents in all delay cells in all taps. Tap delays are instead calibrated by adding extra coarse/fine calibration delay cells to them as shown in Fig. 3.28. The coarse delay cell is identical to the 10-ns delay cell described above. The fine delay cell has a nominal delay of 1 ns. Its architecture is similar to that of the 10-ns delay cell shown in Fig. 3.25, with the exception that the MOM

capacitor is removed (capacitor C_1 is then equal to the total parasitics at node V_C); the current source transistor M_0 is sized with a W/L of 200 nm/600 nm; and the charging current is 180 nA (as against 190 nA in the 10-ns delay cell). While the value of the RMS delay jitter relative to the nominal delay is expected to be worse for the 1-ns delay cell (2.6% in simulations) compared to that of the 10-ns one (1.7% in simulations), the absolute contribution of the former to the overall tap delay jitter will be much smaller than that of the latter. This is because (a) there are fewer 1-ns cells compared to 10-ns ones, and (b) with a smaller delay value (1 ns), the absolute value of the RMS delay jitter of the 1-ns cell will be lower than that of the 10-ns delay cell.

Each tap has two extra coarse and five extra fine delay cells for calibration (in addition to the 10 cells already present). The resulting delay tap is shown in Fig. 3.28. During calibration, each tap delay is measured by sending a test pulse along the associated delay line path. The bias current I_D is adjusted until all tap delays are less than or equal to 100 ns, and the calibration delay cells (coarse/fine) are selectively added (using a scan-chain (not shown) which set the select bits of the MUXes shown in Fig. 3.28) to taps that have a total delay less than 100 ns, until their delay reaches 100 ns. Simulations show that with this, every 100-ns tap delay can be calibrated to have a 1σ delay variation of 640 ps (less than 0.7%) [50]. The average energy dissipated by the tap delay per input token is 42 fJ. The average power overhead of the additional calibration blocks is about 10%. The RMS value of the tap jitter is 316 ps. Therefore, the tap delay satisfies the specifications set in Table 3.4.

The calibration scheme can be implemented on chip and automated. It can be turned off after completion and occasionally turned on to correct for delay variations due to temperature drifts. However, the system was designed as part of a test chip, and we keep the calibration off chip for simplicity and flexibility.

Design of arithmetic blocks

The high energy efficiency of the delay cell (and, hence, the delay tap) needs to be maintained in the arithmetic blocks—multiplier and adder—too. We will discuss their design now.

The power dissipation of arithmetic operations done in CT digital form can be quite high. For instance, the CT DSP in Ref. [16] has CT digital multiplier and adder blocks that together dissipate 150 pJ for every input token to the CT DSP. For an average token rate of 200 MS/s in our case, this would imply, from (1.1), a power dissipation of 30 mW in the arithmetic blocks! This is clearly too high for our application. Therefore, to lower the power dissipation, we exploit hybrid processing domains (mentioned in Chap. 1) [10].

The multiplier coefficients at a tap in the CT DSP are 3-bit signed numbers, $b<0:2>(b_2)$ indicates sign), as given in the specification summary in Table 3.4. Those for the *INC* signal path have a polarity opposite to that of those for the *DEC* signal path in the DSP¹³. As both *INC* and *DEC* are individually 1-bit digital pulses, multiplying them with a coefficient can be accomplished using a simple pass gate: Every *INC/DEC* pulse input to such a multiplier results in a 3-bit output equal to the 3-bit tap coefficient for the duration of the pulse. Therefore, thanks to the 2-bit modulation scheme of the CT ADC, the energy dissipated by the multiplier will be small.

From Ref. [10] (and as discussed in Chap. 1) we know that addition is far more power efficient when done in CT analog domain as against the CT digital domain. Besides, unlike a CT digital adder, a CT analog one does not suffer from metastability issues due to very closely spaced

_

¹³ Recall from Fig. 3.4 that the output is represented by the *difference* of *INC* and *DEC* signals. As discussed above, they have separate parallel delay line paths.

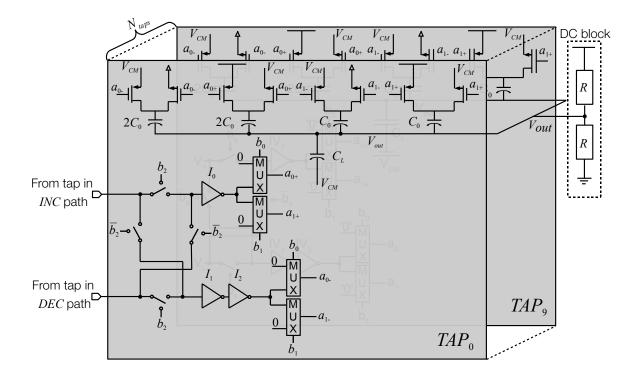


Fig. 3.29. Schematic of the arithmetic unit (multipliers and adder) [50].

input tokens. The downside of the latter is that now the output of the processor is in the analog domain. This, however, is not an issue for the ultra-low-power radio application we target; the output of the processor is to be fed to an energy detector, which does not necessitate a digital input. The adder used in the processor is capacitive in nature. Fig. 3.29 shows the arithmetic unit that houses the adder and the multiplier. It was developed by Alin Ratiu at CEA-LETI, France. We will only give a short description here for completion; more details can be found in Ref. [50].

The composite arithmetic unit of Fig. 3.29 is composed of a number of "slices" of multiplier and capacitive-adder units; there is one such slice at every tap. The multiplier at each tap is implemented using pass-gates and merged with the adder unit of that tap using multiplexers as shown. Consider a single slice of the adder corresponding to TAP_0 , shown in Fig. 3.29. Following an input pulse, the voltage at node V_{out} at the tap is set by capacitive division between

 V_{DD} , V_{CM} (= V_{DD} /2), and ground. The division ratio is set through switches, implemented using transistors controlled using the multiplier outputs, $a_{0+/-}$ and $a_{1+/-}$, which in turn depend on the tap coefficient as described above. The V_{out} nodes in all adder slices in Fig. 3.29 are shorted to generate the summed adder output. A resistor divider composed of two very large (500 k Ω) resistors (R in Fig. 3.29) is used to set the DC level of the node V_{out} .

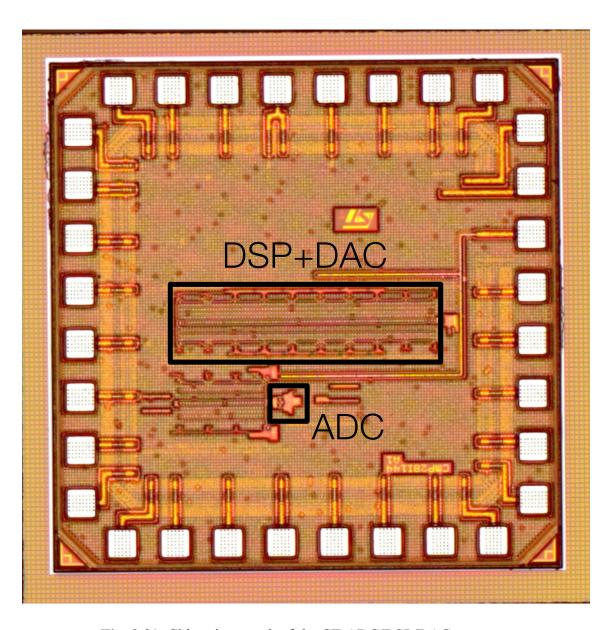


Fig. 3.30. Chip micrograph of the CT ADC/DSP/DAC system.

Each slice of the arithmetic unit (the adder unit and the multiplier) dissipates between 3 fJ to 14.7 fJ per token as the multiplier coefficient goes from 00 to 11. Therefore, the worst-case power dissipation of the adder for an average token rate of 200 MS/s for a 9^{th} -order (10-tap) filter can be estimated to be 29 μ W. This is three-orders-of-magnitude lower than that estimated for the system with a CT digital arithmetic unit. Clearly, an appropriate choice of signal processing domains allows a drastic lowering of power dissipation of the system.

3.4.2 Simulation Results

The CT DSP chip has been fabricated (Fig. 3.30 shows the die photo), but has not been tested yet. Therefore, only simulation results are provided here.

The filter transfer function can be programmed by changing the tap coefficients, $c_{0.9}$ (each represented using three bits, b<0.2> in Fig. 3.29), or the tap delay, T_{TAP} . Simulations were performed to verify both options. These were at the transistor level for all CT DSP blocks, except the digital delays, which were ideal. This was done to allow exhaustive simulations with a realistic simulation time.

First, T_{TAP} was set to 100 ns and the tap coefficients were programmed to achieve different filter response types. From (3.7), we can then say that the filter magnitude response will repeat every 10 MHz. We thus show it for inputs in the [40 MHz, 45 MHz]/ [40 MHz, 50 MHz] band only, as either will completely define the filter response. Three different magnitude response plots obtained from such simulations are shown in Fig. 3.31(a): lowpass, highpass, and bandpass.

Next, the tap coefficients were fixed to achieve a highpass response, and T_{TAP} was varied so that center frequency of the bandpass filter that effectively results (thanks to the response repetition) can be tuned. Three different magnitude response plots, obtained from such simulations

for $T_{TAP} \in [71 \text{ ns}, 77 \text{ ns}, 83 \text{ ns}]$, are shown in Fig. 3.31(b). The plots from Fig. 3.31 demonstrate that the filter transfer function can be successfully programmed by changing T_{TAP} or the tap coefficients. The desired passband width and stopband rejection is also confirmed.

Next, in order to demonstrate interferer rejection achieved by the processor, we apply the

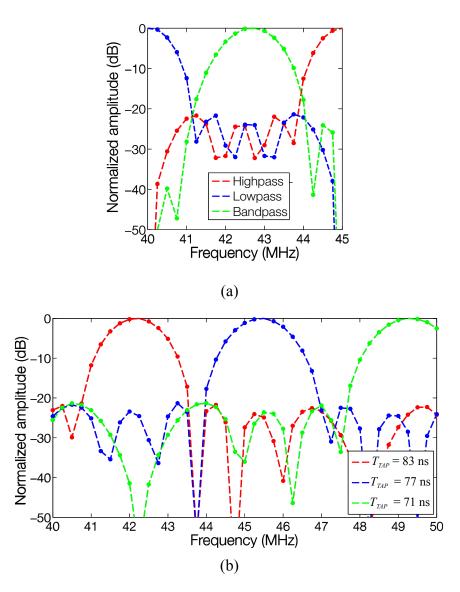


Fig. 3.31. The CT DSP is configured in simulations to implement different frequency responses (a) by changing tap coefficients, c_{0-9} ; and (b) by tuning the tap delay, T_{TAP} .

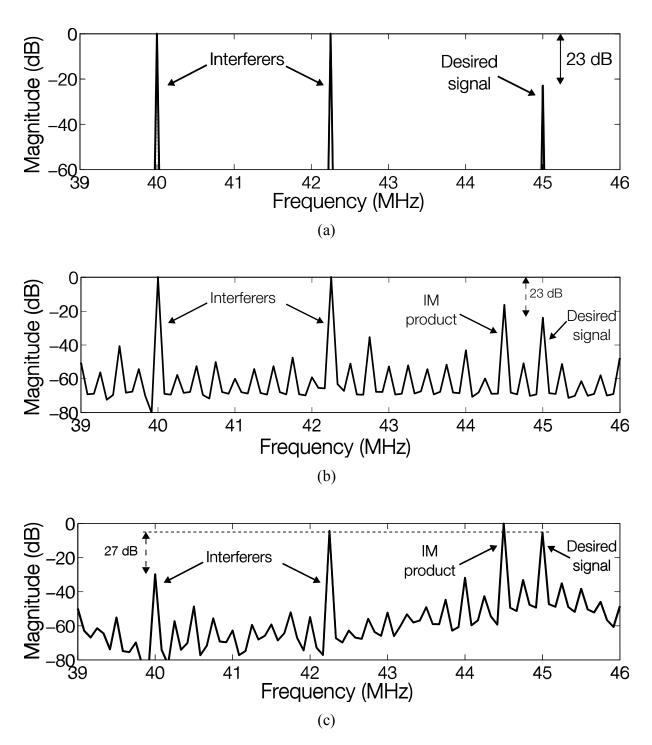


Fig. 3.32. Interferer rejection using the CT/ADC/DSP/DAC system: (a) Input spectrum, with a weak signal and strong interfering components; (b) spectrum at the CT ADC output; and (c) that at the CT DSP output.

at 40 MHz and 42.25 MHz. The input spectrum is shown in Fig. 3.32(a). The amplitude of the signal tone is 23 dB lower than that of each of the interferers, resulting in an input signal-to-interferer (SIR) ratio of -23 dB. The filter is configured to have a bandpass response with a center frequency of 45 MHz. The simulation was performed with the entire system (including the delay cells) at the transistor level. The CT ADC output spectrum is shown in Fig. 3.32(b). The two interfering tones stay intact and create a strong intermodulation product that falls very close to the signal component. The spectrum at the output of the CT DSP is shown in Fig. 3.32(c). The interferer at 42.25 MHz is at par with the signal component at 45 MHz, while that at 40 MHz is about 27 dB below it. Given than we started with an input SIR of -23 dB, we can say that the interferers at 42.25 MHz and 40 MHz are respectively attenuated by 23 dB and 40 dB by the filter. For the given input scenario, the observed ADC output token rate was 256 MS/s. The total system power dissipation is 122 μ W, which can be split as: 26 μ W in the CT ADC; 79 μ W in the CT DSP delay line; and 17 μ W in the CT DSP arithmetic unit.

While the filter rejects out-of-band interferers, the poor input SIR of -23 dB results in a strong in-band intermodulation distortion at the CT ADC output¹⁴, which is not rejected by the filter (since it falls in its passband) and appears at the processor output. Techniques to address this intermodulation distortion using the CT DSP are presented in Ref. [50] and are not detailed here.

We know that the power dissipation of the CT ADC adapts automatically with the input signal amplitude, as seen in Fig. 3.18. Since the ADC output token rate also varies in proportion

 $^{^{14}}$ The intermodulation product will be much smaller for a better SIR, as confirmed by the ADC two-tone measurement results shown in Figs. 3.17-3.18.

to the input amplitude (from (3.1)), we surmise that the event-driven CT DSP, too, should show a scaling of power dissipation with input amplitude. To verify this, a variable-amplitude single-tone input at 50 MHz was applied to the system with the CT DSP configured to have a passband at 50 MHz. The power dissipation of the entire system (ADC and DSP included), simulated at the transistor level, is plotted versus input amplitude in Fig. 3.33. As can be seen, the power dissipation scales automatically with input amplitude; that for zero input is 15.7 μ W: 11.1 μ W in the CT ADC and 4.6 μ W in the CT DSP (dominated by leakage in the 900 delay cells¹⁵). The total power dissipation thus automatically scales from 15.7 μ W to 163 μ W as the input amplitude scales from 0 to 160 mV_{p-p}.

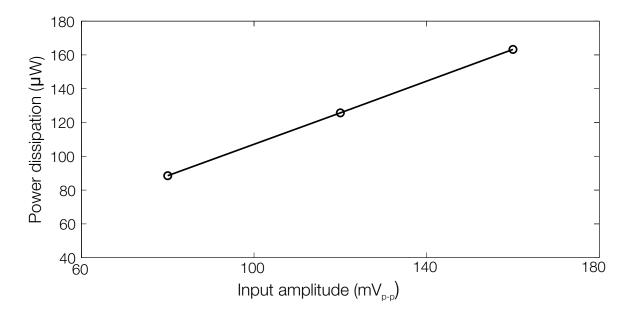


Fig. 3.33. Power dissipation of the entire CT ADC/DSP/DAC system, configured as a bandpass filter with a passband center frequency of 50 MHz, versus input amplitude for a single-tone input at 50 MHz.

¹⁵ The leakage power of each delay cell that is *not* disabled, for an inactive input is 4.6 nW from Table 3.6. If the cell is disabled (e.g. if it is in a disabled tap) the leakage power is only 0.6 nW.

3.4.3 Comparison with the State of the Art

Table 3.7 summarizes the performance of the processor (including the ADC and DSP) and compares it with relevant state-of-the-art FIR CT/DT DSPs and analog FIR filters. They are compared using a normalizing figure of merit, FOM_{DSP} , given by

$$FOM_{DSP} = \frac{P_{SYS}}{2^{ENOB} \times 2f_{BW} \times N_{tans}}$$
(3.10)

where P_{SYS} is the power dissipation of the entire CT ADC/DSP system and N_{taps} is the number of taps in the FIR filter (other terms have been defined before). Unlike the DT counterparts, the proposed system (like the other CT DSP systems reported) is alias free and does not require an

	Ma, JSSC'10 [53]	O'hAnnaidh, ISSCC'10 [52]	Schell, JSSC'08 [3]	Kurchuk, JSSC'12 [10]	This Work (Simulations)
Technology	130 nm CMOS	45 nm CMOS	90 nm CMOS	65 nm CMOS	28 nm UTBB FDSOI CMOS
Supply (V)	0.36	1.1	1	1.2	0.65
Nature	DT FIR DSP	Analog FIR	CT ADC + DSP	CT ADC + mixed-domain DSP	CT ADC + mixed- domain DSP
Input bandwidth	93.5 MHz	800 MHz	10 kHz	2.4 GHz	40 MHz
Sampling rate	187 MHz	3.2 GHz	No sampling	No sampling	No sampling
# of taps, N_{taps}	14	16	16	6	10
Core area (mm ²)	0.38 ^a	0.15	0.06	0.0036	0.093
SNDR (dB)	49.7 ^a	33	58	20.3	32
Total power (µW)	5900 ^a	48	1600 (average)	6200 (average)	16-163 (89 μW average)
FOM _{DSP} (fJ/sample)	9	51	3300	30	3.3
Sampler requires antialiasing filter?	Yes	Yes	No	No	No

^aDoes not include DT ADC. Input is assumed 8-bit; SNDR is thus assumed 49.7 dB.

Table 3.7. Comparison of proposed CT ADC/DSP/DAC system with state-of-the-art CT/DT DSPs and analog FIR filters.

anti-aliasing filter before the ADC. The FOM improvement over CT DSP systems in Refs. [3] and [10] is respectively 1000× and 9× and that over the analog FIR filter in Ref. [52] is 15×. At the same time, the processor achieves performance at par with the state-of-the-art DT DSP in Ref. [53]. It thus presents a significant advance in CT DSP systems in general.

3.5 Conclusions

In this chapter, we discussed the design, implementation, and measurement/simulation results of an energy-efficient CT ADC/DSP/DAC system. The proposed principles, while general, have been developed with an eye towards an application in ultra-low-power radio receivers, which, we surmise, will benefit from the alias-free, event-driven nature of CT DSP systems. A number of principles have been proposed to improve the energy efficiency of the system in order to meet the challenging specification of a meagre 100-µW power budget required by the application. Measurement—and in some cases, simulation—results show that the implemented system will beat state-of-the-art CT DSP systems and will bring it closer to state-of-the-art DT DSP systems.

Chapter 4

Continuous-Time Data Conversion and DSP Using

Voltage-Controlled Oscillators

4.1 Introduction

Voltage-controlled oscillators (VCOs) have an integral relationship between the input voltage and output phase. Recently, there has been a lot of interest in exploiting this relationship to implement the analog function of integration, which forms an important building block in many analog and mixed-signal systems like ADCs, amplifiers, and filters. Most notably, the highly

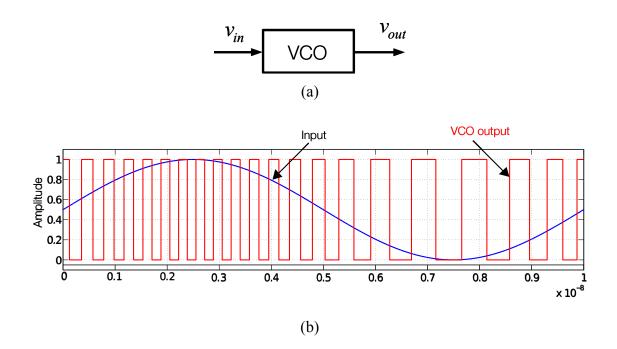


Fig. 4.1. (a) A general VCO; and (b) its terminal waveforms.

digital and technology-scaling-friendly nature of voltage-controlled ring oscillators (VCROs) has made them an attractive option for implementing analog integrators [54]. VCROs have been successfully used to implement analog filters [54], amplifiers [55], and DT ADCs [56][57].

Given the mostly-digital nature of the resulting systems, there is a strong motivation to pursue VCO-based mixed-signal systems. However, much of the prior work in this regard has been restricted to CT analog or DT digital systems. The possibility of a VCO-based CT ADC/DSP system has not yet been explored. This is surprising since the output of any typical VCO¹⁶, shown in Fig. 4.1, is inherently CT digital—the transitions in the discrete-amplitude (or digital) output are not synchronized to any clock and can occur at any point in time. Processing this CT digital output of a VCO in continuous time using a CT DSP should, therefore, be a proverbial "lowhanging fruit". In this chapter, we explore this possibility [58]. In the process, two well-known modulation schemes are revisited, but in the context of a VCO-based implementation: pulse width modulation (PWM) [18] and pulse frequency modulation (PFM) [19]. In the systems discussed, a VCO-based CT ADC encodes the analog input in PWM or PFM form, and a CT DSP processes its output. Both modulation schemes fall in the broad category of analog-to-digital conversion via duty cycle modulation [59], but in continuous time, i.e. without sampling in time. We compare the two schemes with each other and with LCS for a given set of specifications, in terms of NTPS and T_{GRAN} , and hence, potential DSP power dissipation (see (1.1)). Advantages and limitations of both approaches are also discussed. Unless specified as such, the principles discussed are general and

¹⁶ We consider here, without loss of generality, VCOs that produce a binary signal at their output. We assume that VCOs that produce sinusoidal waveforms, can produce a similar binary signal too, provided their sinusoidal output is passed through a zero-crossing comparator.

not specific to a particular type of VCO (e.g. ring or LC VCO). The choice of an appropriate VCO will depend on the targeted applications and desired specifications. We will see one such case in the next chapter.

4.2 Pulse Width Modulation Using a VCO

At the heart of VCO-based analog and mixed-signal systems (e.g. filters and amplifiers) is a pseudo-differential VCO-based system shown in Fig. 4.2(a) [54]. A fully differential input is applied to two VCOs, which produce an oscillatory voltage output whose frequency is proportional

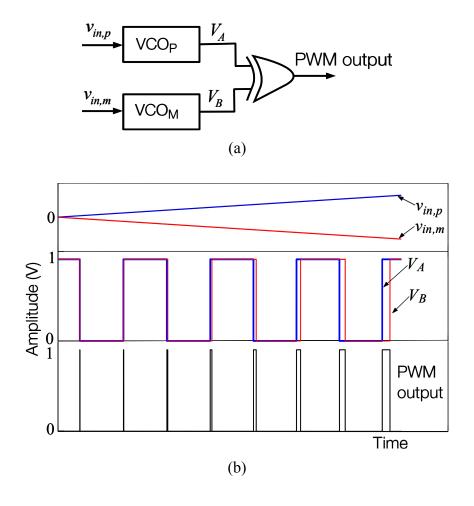


Fig. 4.2. (a) Pseudo-differential voltage-controlled oscillators implementing an analog integrator; (b) example waveforms.

to the applied input voltage. The differential analog input creates a proportional difference in the frequency of oscillation—and hence in the phase (relative positioning of the edges)—of one oscillatory output relative to that of the other, as can be clearly seen in Fig. 4.2(b). This relative phase difference is captured by a phase detector. An XOR gate (Fig. 4.2(a)) is one possible phase detector (see Ref. [54] for other possibilities) as it converts the phase difference into the pulse width of a digital signal (Fig. 4.2(b)). The phase at the output of a VCO is related to its input voltage through an integration [54]. Thus, the circuit encodes the *integral* of the analog input into a pulse-width modulated signal, thereby implementing an integrator (discussed later in more detail).

The open-loop architecture allows high-speed operation, but is limited by VCRO nonidealities like drift and nonlinearity. These issues associated with the PWM encoder have been a subject of work elsewhere [54]–[56], with a number of interesting solutions, including ones with feedback [55]. The encoder and its issues are, however, not the focus here. We instead assume an ideal encoder and focus on the DSP of these signals produced by a general VCO-based PWM encoder (Fig. 4.2(a)), without time discretization—i.e. in continuous time.

The PWM signal produced by the phase detector in Fig. 4.2(a) is CT digital: it is continuous in time and discrete in amplitude [3]. Therefore, the PWM signals generated by a VCO-based PWM encoder can be directly processed by a CT DSP. As we will see such an encoding method greatly relaxes the CT DSP constraints relative to LCS, thereby enabling a potential implementation with low power and small chip area.

4.2.1 System Architecture

As an example, we consider the open-loop pseudo-differential VCO structure of Fig. 4.2(a) to be the PWM encoder (the principles presented can also be applied to a feedback structure [55] with suitable modifications). In this case, the PWM representation encodes the integral of the differential input. Therefore, in order to restore the original signal, the output needs to be differentiated—only in the signal band—using a bandlimited CT digital differentiator, so as to implement a CT ADC (Fig. 4.3(a)). As this is a linear operation, it can be interchanged/merged with the subsequent CT DSP (which is the case in the following presented simulations), or even after the following reconstruction CT DAC, which generates the analog output. The resulting system is shown in Fig. 4.3(a); the system with details of the CT DSP block is shown in Fig. 4.3(b). If the VCOs are implemented using a VCRO, the system will consist of only inverters, digital delays, and other logic blocks, making it highly scalable and amenable to low-supply implementations. The CT ADC does not require power-hungry CT comparators, which have so far been a bottleneck in LCS-based CT ADCs [3].

Depending on the phase detector used, the minimum output pulse width of the PWM signal can be arbitrarily low, resulting in an arbitrarily low minimum intersample time, T_{GRAN} . Preserving every ADC output sample—every edge of the digital output—will then necessitate an extremely long delay-line, with individual delay cells of a very small delay value equal to the minimum intersample time, T_{GRAN} . This will make delaying of these signals along a sufficiently-long delay line (Fig. 4.3(b)) impractical. An alternative must thus be found.

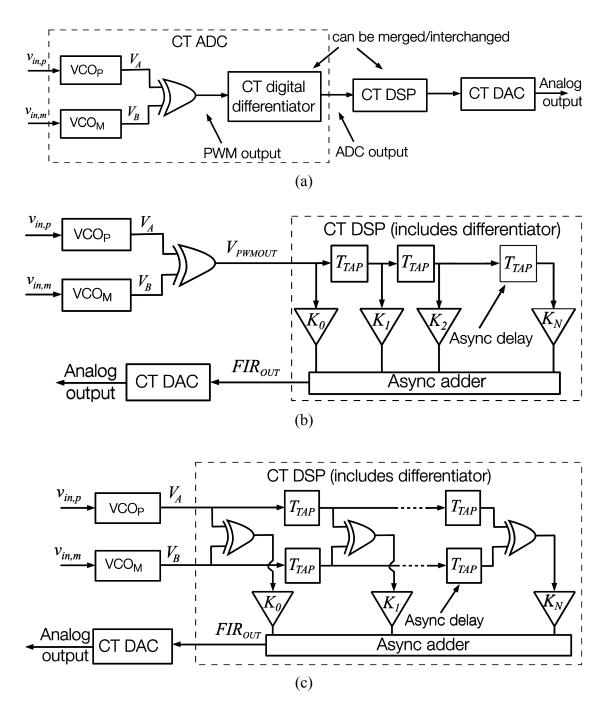


Fig. 4.3. CT ADC/DSP/DAC systems based on pseudo-differential VCOs: (a) general system; (b) general system with details of CT DSP block; (c) practical CT DSP implementation for avoiding very narrow pulses.

Let V_A and V_B be the outputs of the VCOs, and let \oplus represent the XOR operation. The PWM output is:

$$V_{PWM,OUT}(t) = V_A(t) \oplus V_B(t) \tag{4.1}$$

The input to the multiplier at the n^{th} tap in Fig. 4.3(b) is $V_{PWM,OUT}(t - nT_{TAP})$. As the XOR operation is time invariant, we can write:

$$V_{PWM,OUT}(t - nT_{TAP}) = V_A(t - nT_{TAP}) \oplus V_B(t - nT_{TAP})$$
(4.2)

This implies that the input to the multiplier at the n^{th} tap can also be obtained by directly delaying the two VCO outputs along a tapped delay line and performing an XOR operation on the two at the n^{th} tap. The resulting system is shown in Fig. 4.3(c). It is, in principle, equivalent to that in Fig. 4.3(b) as the signals at the multiplier inputs in the two are identical. Two delay lines are required to delay the outputs of the two VCOs. Despite this doubling of the delay line, there is no power dissipation penalty; in fact, the latter will be lower. This is because with two delay lines, the constraints of each of those delay lines are significantly more relaxed compared to that of the single delay-line implementation [10]. Now, the minimum inter-sample time at the input of each delay line, T_{GRAN} , is greatly relaxed to the minimum time between two rising or two falling edges of the PWM pulses (or the minimum VCO output pulse width). This makes propagation of the pulses along the delay line feasible.

We note that the PWM output in Fig. 4.2(b) differs from that produced by a classical asynchronous sigma-delta modulator (ASDM) [21][59], which is also of PWM form and is CT digital [60], in two important aspects. Unlike ASDM, the VCO-based system produces no limit-cycle oscillations for a zero input. Besides, the digital and scalable nature of the encoder based on VCOs makes it more attractive than an ASDM.

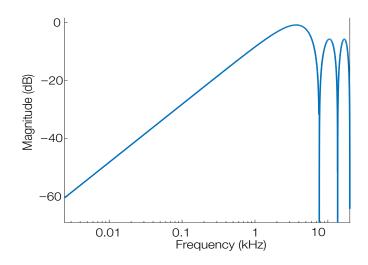


Fig. 4.4. Differentiator transfer function.

4.2.2 Simulation Results

The system in Fig. 4.3(c) was simulated in MATLAB using behavioral code. As an example, input signals with frequency, f_{in} , in the range [200 Hz, 4 kHz] and amplitude in the range [-1, 1] were considered. The zero-input oscillation frequency of the VCO, f_c , was set to 10 kHz. The range of the VCO output frequency, f_{out} , set through its gain, K_{VCO} (= $\frac{df_{out}}{dv_{in}}$), was chosen such that the phase difference between the two VCRO outputs (time between two rising or two falling edges) never exceeds π (corresponding to half the period). Otherwise, excessive distortion results due to output overflow in the phase detector [54]. This is a critical requirement and results in a small value for K_{VCO} (100 Hz/V). The 4-kHz-band-limited differentiator was implemented using a 6th-order CT FIR filter with a 25 μ s tap delay (transfer function shown in Fig. 4.4).

1. Spectral characteristics: Fig. 4.5 shows the ADC output spectrum in the proposed scheme for a full-scale single-tone input at 200 Hz. The spectrum consists of the signal component and modulation products at $2kf_c \pm nf_{in}$ $(k, n \in I)$ that roll off at high frequencies. Unlike the case with an LCS ADC [3], there are no distortion components in the signal bandwidth. A high in-band

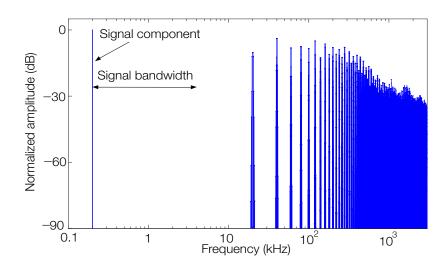


Fig. 4.5. Spectrum of the output of the VCO-based PWM encoder (post differentiation) for a full-scale single-tone input at 200 Hz.

SER is achieved, limited only by noise. The out-of-band components can be rejected using a lowpass filter, possibly after the CT DAC. If the VCOs are implemented using an architecture that produces multiple phases, N_{phi} , (as is the case, e.g., in a ring VCO), the latter can be XORed using N_{phi} XOR gates to generate N_{phi} parallel PWM-encoded signals as shown in Fig. 4.6 (CT DSP slices are discussed later). The composite output can then be obtained by summing these N_{phi} signals (not shown). This output will then have a zero-input oscillation frequency of $N_{phi} \times 2f_c$, and its spectrum will have modulation products at integer multiples of $N_{phi} \times 2f_c$ [54]— N_{phi} times higher than that in the case with one phase. For these simulations, however, we consider only one phase.

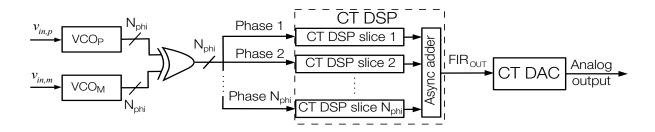


Fig. 4.6. System implementation for multiphase operation—each DSP slice has the CT DSP architecture shown in Fig. 4.3(c).

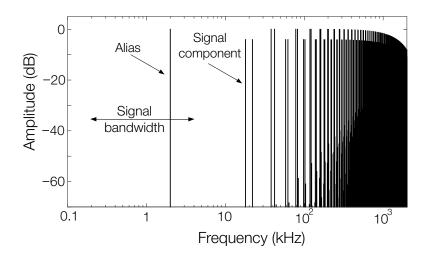


Fig. 4.7. Example spectrum demonstrating aliasing in the proposed system.

For a two-tone input, with the two tones at 200 Hz and 2 kHz, the ADC output spectrum shows no significant distortion components in the 4 kHz bandwidth (see below in Fig. 4.8). In contrast, such an input to an LCS system will fill the baseband of the output spectrum with intermodulation distortion. This demonstrates the spectral superiority of the scheme over LCS. However, it has one major drawback compared to LCS. The ADC output spectrum shows modulation products at $2kf_c \pm nf_{in}$ ($k,n \in I$). Therefore, if f_{in} is close to $2f_c$, it will be "aliased" back to the baseband. For instance, for $f_c = 10$ kHz, an 18-kHz single-tone input to the ADC will create a component at 2 kHz in the ADC output spectrum, as shown in Fig. 4.7. This spectral aliasing is equivalent to what one would get in a DT ADC with a sampling rate, f_s , of $2f_c = 20$ kHz. Therefore, unlike LCS, this system will require an antialiasing filter.

1. Example CT DSP: As an example, a 67^{th} -order low-pass FIR CT DSP with an f_{-3dB} of 500 Hz was implemented with a 25 µs tap delay (filter frequency response repeats every 40 kHz). A two-tone input with two equal-amplitude tones at 200 Hz and 2 kHz was applied at the input of the ADC/DSP system. The spectra for the ADC and DSP outputs is shown in Fig. 4.8. As can be seen, the signal component at 2 kHz in the DSP output spectrum is attenuated (by >70 dB) while

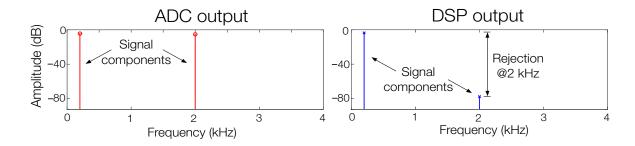


Fig. 4.8. Spectrum of the output of the proposed ADC and that of the following DSP for a two tone input with two tones at 200 Hz and 2 kHz. The DSP is a 67^{th} -order low-pass FIR filter with an f_{-3dB} of 500 Hz.

the one at 200 Hz is maintained relative to the corresponding components in the ADC output spectrum. This confirms that the CT DSP techniques from Refs. [3], [10], [16] can interface well with the VCO-based PWM encoding approach. Note that if the VCOs, and hence the encoder, produces multiple output phases N_{phi} (as discussed above), a CT DSP slice can be used to process signals at each phase (N_{phi} slices in all) as shown in Fig. 4.6; the outputs of all slices can finally be summed to generate the composite output [10].

2. Output token rate (NTPS) and granularity (T_{GRAN}): Every edge of the PWM output constitutes an encoder output token. The encoder's number of tokens—or the number of PWM output pulse edges (rising and falling)—generated per second for the single-tone-input case is independent of the input frequency, and is approximately equal to $N_{phi} \times 4f_c$, or 40 kS/s in the presented simulations ($N_{phi} = 1$). This is because a pulse (of variable width) is generated at every rising/falling edge of the outputs of the VCOs (Fig 4.1(b)), with four such edges/cycle; also, the average frequency of oscillation of the VCO output is f_c (10 kHz). The T_{GRAN} , as described above, is equal to the minimum pulse width of the VCO outputs, and, thanks to the low K_{VCO} , is approximately $(2 \times N_{phi} \times f_c)^{-1} = 50 \,\mu s$.

Parameter		LCS system (8-bit)	VCO PWM system
NTPS		102.4 kS/s-2 MS/s (200 Hz to 4 kHz)	40 kS/s
T_{GRAN}		300 ns	50 μs
DSP power (P_{DSP})	P _{Delay-line}	P_1	$P_1/434$ to $P_1/8500$
	P_{Adder}	P_2	$P_2/2.5$ to $P_2/50$
DSP delay-line area $(A_{Delay-line})$		A	A/85
In-band quantizaton distortion?		Yes	No

Table 4.1. Comparison of the VCO-based PWM encoder system with an LCS CT ADC/DSP system for identical CT DSP specifications.

The proposed system is compared with a CT ADC/DSP/DAC system with 8-bit LCS [3] in Table 4.1. As can be seen, the proposed system will achieve a significantly lower NTPS (by 2.5-50×) and a greatly relaxed T_{GRAN} (by 170×). Using these results and equations (1.1)-(1.2), we can estimate and compare the potential power dissipation and chip area of the CT DSP that will handle the outputs of each of the two systems. In these comparisons, we will assume that the energy/token and chip area of the delay cell in the CT DSP remains independent of the delay value [25]. For reasons discussed in Sec. 3.4.1, this assumption is valid provided the delay value of the delay cell is controlled using the charging current and not the charging capacitor [25]. Using this, we conclude that the relaxation of NTPS and T_{GRAN} afforded by the proposed system has the potential to drastically lower the CT DSP chip area by decreasing the size of its delay line by 85×, and to lower its power dissipation in the delay line by 434-8500× and that in the arithmetic blocks by 2.5-50×. This is achieved with a much higher in-band SER than that in LCS ADCs due to the latter's in-band distortion and the former's lack of it, but at the expense of aliasing.

4.2.3 Non-Idealities and Practical Considerations

The system in Fig. 4.3(c) does not have to delay very narrow pulses. However, the narrow pulses eventually re-appear at the XOR outputs and need to be handled by the multipliers and the adder. To handle this (especially at high input frequencies) one can use a semi-digital approach as in Ref. [10].

VCO nonlinearity, modeled based on Ref. [57], results in in-band distortion, the power of which depends on the K_{VCO}/f_c ratio. Since this value is small in our simulations (due to the phase detector), the distortion power remains negligible. In cases where this is not true, feedback-based structures [55] or those with calibration [57] can be used, while processing the PWM output using the proposed approach.

The phase noise of the VCOs directly limits the SNDR. When modeled as 4-kHz-bandlimited (voltage) white noise at the input of the VCOs for the system described in Sec. 4.2.2,

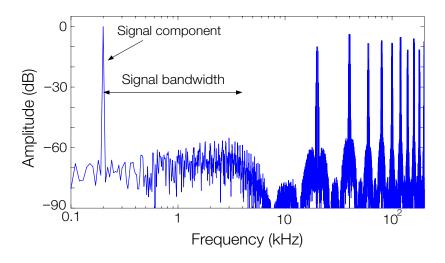


Fig. 4.9. Output spectrum from Fig. 4.5 repeated with 4-kHz-bandlimited (voltage) white noise added at the input of the VCOs.

the spectrum of the ADC output (post differentiation) is shown in Fig. 4.9. A comparison with the output spectrum for the noiseless case in Fig. 4.5 shows how such noise limits the in-band SNDR.

Any drift in the VCO output frequency due to, say, temperature variations, will affect the conversion linearity; this may necessitate calibration and some form of feedback in the PWM encoder, thereby limiting the frequency of operation.

The systems in Figs. 4.3(b)-(c) are equivalent only in the ideal case where no mismatch exists between the two parallel delay lines. In the presence of mismatch, this equivalence is invalid. Consider (for now) that in any delay cell there is no mismatch in the delay for rising and falling edges of the pulse input (i.e. pulse width at the input of the tap delay input equals that at its output). Two cases can then be considered:

Case I: Some or all delay cells in any given delay line in Fig. 4.3(c) are mismatched, but delay cells in the two delay lines in Fig. 4.3(c) at the same tap level—or in the same tap column—match with each other. In such a case, the pulses at each tap are delayed by mismatched values, but the widths of each of these pulses remain equal to what they would have been had there been no mismatch. Therefore, such mismatch will only modify the filter frequency response, without adding any new spectral distortion components (or affecting SER).

Case II: All delay cells in Fig. 4.3(c), including those in the two delay lines at the same tap level—or in the same tap column—are mismatched with each other. Due to the mismatched tap delays, the tap outputs in the two delay paths at each tap level will have undergone different delays, resulting in pulse-width distortion at the XOR outputs. This manifests itself as in-band distortion in the output spectrum. Simulations indicate that a 1% tap delay mismatch (achieved post calibration in Ref. [16]) limits the worst-case CT ADC's effective number of bits (ENOB =

(SNDR – 1.76)/6) to only 6-7 bits. To reach at least a 10-bit ENOB, the mismatch needs to be below 0.2%. At this level of mismatch, the filter frequency response described in Fig. 4.7 also remains essentially unaffected. Note, however, that to avoid in-band distortion, we only need satisfy requirements of Case I. This means that high-accuracy matching is only required in the two delay lines in Fig. 4.3(c) at the same tap level—or in the same tap column; up to 1% mismatch in any other delay cells then does not lower the ADC ENOB below 10 bits.

Another imporant source of mismatch is that between the delay value for rising and falling edges of input pulses in any delay cell [10]. It is this mismatch that primarily limits the resolution of the CT DSP in Ref. [10] to a mere 3 bits, and it will limit that of this system too. A distinction has to be made between: a) the delay mismatch between delaying any two rising-edge (or two falling-edge) inputs; and b) the delay-value discrepancy between rising and falling edges of inputs due to mismatch. The mechanism for delaying a rising input edge is inherently different from that for delaying a falling input edge. For instance, in Ref. [10], a rising edge was delayed by discharging a capacitor using an NMOS current source, while a falling edge was delayed by charging the same capacitor using a PMOS current source. Because the charging and discharging mechanisms involve two inherently different devices—PMOS and NMOS—they will never match perfectly under global and local variations, thereby affecting (b). On the other hand, the effect of global variations on (a) can be minimized through careful layout, as the mechanisms for delaying any two rising (or any two falling) edges of the input are usually identical; in that case, only local variations affect (a). Therefore, (b) is usually the more dominant factor. This is even more worrisome for PWM encoding as all signal information is encoded in the width of the output pulses, and any discrepancy between the delays for rising and falling edge of the input directly alters the pulse width at the output and creates distortion.

Overall, it is thus clear that, while the system in Fig. 4.3(c) relaxes T_{GRAN} , it does this at the expense of an increased sensitivity to mismatch, which is only aggravated as multiple phases, N_{phi} , of the VCOs are considered (see Fig. 4.6). This makes intuitive sense as the phase difference between the two VCO outputs, which encodes the analog input and defines the PWM pulse width, has to be maintained along the processing chain in order to preserve information. This is a major drawback of the system. One silver lining is that the proposed system requires very few delay cells per tap (e.g. only 1 cell/tap in the example discussed, compared to >83cells/tap in the LCS case for the same specifications). Therefore, while mismatch specifications will be tight, calibration complexity need not be as severe as that in the LCS case [3] and can be handled.

4.2.4 Conclusions

VCO-based analog integrators produce a unique PWM encoding of an analog input. We considered direct CT DSP of signals encoded using this method and found that such a scheme presents a major advantage over traditional approaches to CT data conversion and CT DSP in terms of power dissipation, chip area, spectral quality, and affinity to technology scaling. However, these advantages come at the cost of aliasing and an increased sensitivity to mismatch. While much of the latter can potentially be tackled through calibration, the delay discrepancy for rising and falling edges of PWM-encoded inputs can be tough to tackle [10]. We thus abandon the prospect of an integrated implementation for this system, and instead focus on the one introduced next that keeps much of the advantages of this system and resolves the issues associated with it.

4.3 Pulse Frequency Modulation Using a VCO

A pulse frequency modulator [19], [59] converts an analog input into a stream of fixed-width pulses at its output, whose repetition rate—which we will call the "pulse frequency" from

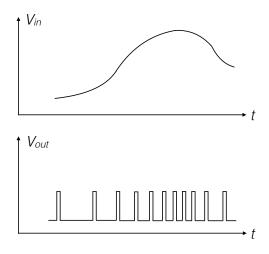


Fig. 4.10. An example of pulse frequency modulation.

here on (more on this terminology later)—varies linearly with the applied input amplitude (Fig. 4.10). The analog input thus modulates the pulse frequency at the output, resulting in the name "pulse frequency modulation" (PFM). While this technique is old [19], [61], [62], we revisit it from the point of view of a VCO-based implementation and in the context of processing its output using CT DSP. Such processing makes sense because, just like the PWM-encoded output discussed in the first part of the chapter, the PFM output is CT digital: the 1-bit digital pulses at the output (Fig. 4.10) and their transitions are not synchronized to any clock and can occur at any point in time. Therefore, they can, in principle, be directly interfaced with a CT DSP. In the following sections, we will consider VCO-based PFM and also study the effect of the corresponding encoding on the CT DSP constraints. Comparisons will be drawn against VCO-based PWM and LCS encoding.

4.3.1 System Architecture

There are a number of ways to implement a pulse frequency modulator using a VCO. A

straightforward implementation¹⁷ is shown in Fig. 4.11(a) along with example waveforms in Fig. 4.11(b). An analog input is applied to a VCO, which produces a binary CT digital (voltage) output whose frequency of oscillation varies in proportion to the applied input voltage. For instance, it can be observed in Fig. 4.11(b) that the VCO output oscillation frequency is higher at the positive

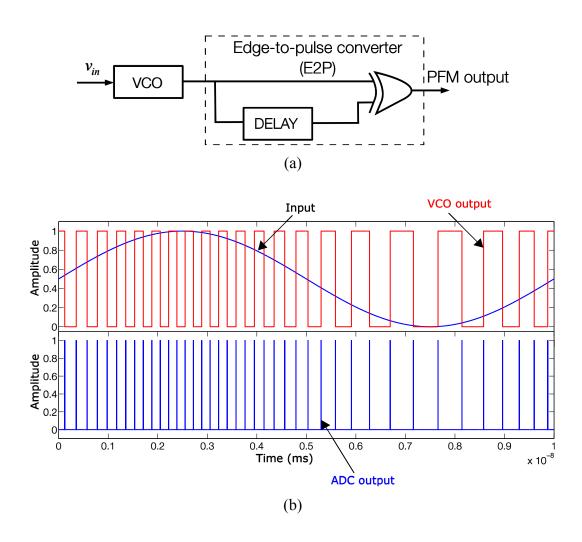


Fig. 4.11. (a) A VCO-based PFM encoder with (b) example waveforms.

¹⁷ We choose the current implementation so that the reader can easily relate it with the VCO-based PWM encoder of Sec. 4.2. In chapter 5, we will consider a different implementation that interfaces better with the CT DSP that follows it

peak of the analog input compared to its negative peak. An edge-to-pulse converter (E2P) then converts every edge—rising/falling—of the VCO output into a pulse of fixed width, T_{PW} (which will be smaller than the minimum time between any two edges), resulting in a unipolar pulse train, at the ADC output, as shown in Fig. 4.11(b). The time origin, t = 0, is marked using a point on the input signal; the time between this origin and the nearest output pulse that precedes it (not shown in Fig. 4.11(b)) is then a random variable, α [19]. The random nature of this variable indicates the fact that, in PFM, the occurrence of an output pulse is a random event w.r.t. the time origin [19]. The output pulse train is then represented as $p(t, \alpha)$.

The fixed-width 1-bit pulse train at the PFM encoder output is CT digital, and its repetition rate varies in linear proportion to the applied analog input. This can be seen in Fig. 4.11(b) where the pulse train becomes dense around the positive peak of the applied input—i.e., when it has a high value—and gets sparse around the negative peak of the input—i.e., when it has a low value. The cascade of the VCO and the E2P in Fig. 4.11(a) forms a CT VCO-based PFM ADC, which thus encodes the analog input in the relative occurrence rate of the 1-bit CT digital pulses at its output. This output can then be fed into the delay line of a CT DSP FIR filter (discussed later). We note the following important points about this modulation scheme:

—Unlike the VCO-based PWM encoder, the PFM encoder in Fig. 4.11(a) produces a pulse train at its output even when the input is zero (or at its common mode value). For a zero input, this pulse train has a fixed frequency of oscillation, termed the unmodulated pulse frequency, f_0 , equal to twice the oscillation frequency of the VCO output for a zero input, f_c . The factor of 2 occurs

because the E2P converts both rising and falling edges of the VCO output¹⁸. Similar to case discussed for the VCO-based PWM encoder in Fig. 4.6, if the VCO architecture produces multiples phases (e.g. as in a ring VCO), N_{phi} , each of them can be fed into an E2P (N_{phi} E2Ps in all), producing N_{phi} parallel output pulse trains; these can then be summed to generate the composite output (not shown). The zero-input oscillation frequency of this composite output will then be, $f_0 = N_{phi} \times 2f_c$.

—We now discuss the input dependence of the instantaneous pulse frequency at the PFM output. Strictly speaking, referring to the pulse repetition rate as the "pulse frequency" is not rigorous as the term "frequency" assumes a fixed periodicity of the pulses, whereas in the PFM encoder the time between consecutive output pulses is varying with the input signal. The periodicity of the output pulse train can then be very different from this inter-pulse time. Besides, if the pulse frequency is defined using the inter-pulse time, it will have well-defined values only at non-uniform, discrete time instants that mark the rising edges in the output pulse train. Therefore, the pulse frequency will not be a CT function and cannot be expressed as a simple function of the CT voltage input function, $v_{in}(t)$. To avoid this, the pulse frequency has to be defined as a CT function and that requires invoking the phase at the VCO output. We will do this in the phase-domain model discussed in the next section. For now, we assume that the instantaneous output pulse frequency, f_{out} , will be a CT function, which can be defined based on the time rate at which the rising edges

¹⁸Had only the rising (or only the falling) edge of the VCO output been converted to a pulse by the E2P, the PFM encoder would produce a pulse train with a fixed frequency of oscillation, $f_0 = f_c$.

of the output pulses occur, and that its input dependence can then be expressed as 19:

$$f_{out}(t) = f_0 + K_{VCO}v_{in}(t)$$
 (4.3)

where v_{in} , \in [-A, A] (unit: Volt), is the input amplitude; K_{VCO} (unit: Hz/V) is termed the "gain" of the modulator/VCO; and f_0 (unit: Hz), as described above, is the zero-input oscillation frequency of the modulator output. $\Delta f_p = K_{VCO}A$ is the maximum frequency deviation (unit: Hz) of the output oscillation frequency from what it is for a zero input (f_0). It is an important parameter in the system design.

Note that this form of encoding can also be achieved using a classical voltage-to-frequency converter [45]. However, we assume that the encoder is implemented using the VCO-based implementation in Fig. 4.11(a) and evaluate it in the context of CT DSP. A comparison is made against systems based on LCS and VCO-based PWM encoders discussed in the previous section.

4.3.2 System Model and Spectral Description

1. Spectral description: The spectral description of this form of PFM for a sinusoidal input and rectangular pulses, $p(t, \alpha)$, at the output has been derived in Ref. [19]. We show it here to understand the scheme better for a potential design implementation. Let the sinusoidal input be

$$v_{in}(t) = A\cos\left(2\pi f_{in}t + \theta\right) \tag{4.4}$$

The output PFM pulses can be expressed as a sum of infinite cosine components [19] as

¹⁹ The VCO output oscillation frequency has the following relation: $f_{VCO}(t) = f_c + K_{VCO}v_{in}(t)$.

$$p(t,\alpha) = V_{PUL}T_{PW}f_0 + V_{PUL}T_{PW}\Delta f_p \frac{\sin(\pi T_{PW}f_{in})}{\pi T_{PW}f_{in}} \cos(2\pi f_{in}t - \pi T_{PW}f_{in} + \theta)$$

$$+2V_{PUL}T_{PW}f_0 \sum_{k=1}^{\infty} \sum_{n=-\infty}^{\infty} J_n \left(\frac{k\Delta f_p}{f_{in}}\right) \frac{\sin(\pi T_{PW}(kf_0 + nf_{in})}{\pi T_{PW}kf_0}$$

$$\times \cos\left[2\pi (kf_0 + nf_{in})t - \pi T_{PW}(kf_0 + nf_{in}) + 2\pi kf_0\alpha\right]$$

$$+n\theta - \frac{k\Delta f_p}{f_{in}} \sin(\theta - 2\pi f_{in}\alpha)$$

$$(4.5)$$

where V_{PUL} and T_{PW} are respectively the amplitude and width of the PFM output pulses, f_0 is the unmodulated pulse frequency, $\Delta f_p = K_{VCO}A$ represents the maximum frequency deviation, and J_n is a Bessel function of the first kind of order n.

In order to simplify (4.5), if the pulses, $p(t, \alpha)$, are approximated as impulses by taking the limit as $V_{PUL} \to \infty$ and $T_{PW} \to 0$, and assuming the resulting impulses have strength $S_I = V_{PUL}T_{PW}$, the resulting expression is [19]:

$$p(t,\alpha) = S_I f_0 + S_I \Delta f_p \cos(2\pi f_{in} t + \theta)$$

$$+2S_I f_0 \sum_{k=1}^{\infty} \sum_{n=-\infty}^{\infty} J_n \left(\frac{k\Delta f_p}{f_{in}}\right) \left(1 + \frac{n f_{in}}{k f_0}\right)$$

$$\times \cos\left[2\pi (k f_0 + n f_{in})t + 2\pi k f_0 \alpha\right]$$

$$+n\theta - \frac{k\Delta f_p}{f_{in}} \sin(\theta - 2\pi f_{in} \alpha)$$

$$(4.6)$$

Eq. (4.6) is approximately valid independent of the pulse shape, provided the area of the pulse is concentrated over a duration that is much smaller than the total period [19]. The components in (4.5)/(4.6) represent the frequency spectrum components of $p(t, \alpha)$, and they directly give the magnitude and frequency of the terms in its Fourier transform [19]. These components are shown in Fig. 4.12 (they are confirmed via simulations in Sec. 4.3.3).

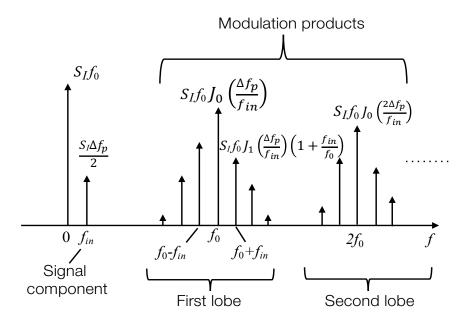


Fig. 4.12. A typical output spectrum of a pulse-frequency modulated signal. The component magnitudes are from the expression in (4.6).

The DC component in the spectrum, which represents the average of the output pulse train, has an amplitude of $S_L f_0$. The signal component, located at f_{in} , has a magnitude given by

$$v_{out,sig} = \frac{S_I \Delta f_p}{2} \tag{4.7}$$

which is proportional to the frequency deviation, Δf_p , and the impulse strength, S_I .

The remaining terms are the modulation products at $kf_0 \pm nf_{in}$ ($k,n \in I, f_0 = N_{phi} \times 2f_c$). They form "lobes" centered at kf_0 . The individual modulation product components have amplitudes given by Bessel functions of the first kind and order n, $J_n\left(\frac{k\Delta f_p}{f_{in}}\right)$. Assume that the maximum frequency deviation and the input frequency are fixed and that k=1. Then, $J_n\left(\frac{\Delta f_p}{f_{in}}\right)$ quickly drops as the absolute value of n increases. This can be seen in Fig. 4.12, where in any given modulation-

product lobe, the amplitude of the components drops as one moves away from the lobe center (kf_0) . The rate of this drop depends on the argument of the Bessel function, $\frac{\Delta f_p}{f_{in}}$: the higher its value, the slower the drop. An example is shown in Fig. 4.13, where $J_n\left(\frac{\Delta f_p}{f_{in}}\right)$ is plotted versus n for $\frac{\Delta f_p}{f_{in}}=2$ and $\frac{\Delta f_p}{f_{in}}=4$. Another way of putting this is: a higher $\frac{\Delta f_p}{f_{in}}$ results in a wider "essential bandwidth" of the modulation-product lobe²⁰. Therefore, in order to not let the modulation products from the first lobe span a high bandwidth and create significant in-band distortion, this ratio has to be restricted. Consequently, for a given f_{in} , the maximum frequency deviation, Δf_p , has to be carefully chosen to have a sufficiently small value, which when exceeded will result in high in-band distortion.

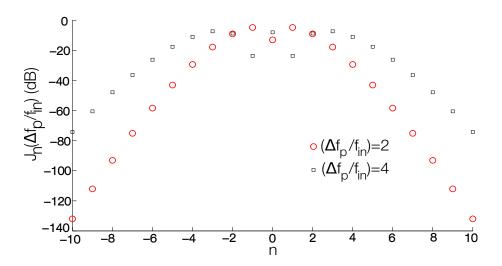


Fig. 4.13. Plot of $J_n\left(\frac{\Delta f_p}{f_{in}}\right)$ (in dB) versus n for two different values of $\left(\frac{\Delta f_p}{f_{in}}\right)$; the higher the latter's value, the slower the fall in the value of the Bessel function w.r.t. n.

²⁰ A good heuristic is that the first modulation product lobe contains relatively strong components (which defines its essential bandwidth) in the frequency range spanning $f_0 \pm \Delta f_p$ (which is exactly the range of frequencies spanned by the PFM encoder output); the second lobe in the range $f_0 \pm 2\Delta f_p$ and so on. In general that for the k^{th} lobe will be: $f_0 \pm k\Delta f_p$.

However, the amplitude of the signal component is proportional to Δf_p and needs to be sufficiently large to overcome the random noise level in the system (thermal, flicker etc.) by a suitable margin. This trade-off between in-band distortion power and that of the signal component informs the choice of the maximum frequency deviation for a given input signal bandwidth.

The first lobe (k = 1) is the most critical one as it is the closest one to the baseband. Therefore, in the following discussion, we assume k = 1. If $f_0 >> f_{in}$, the modulation products that fall in the signal band have negligible amplitudes; the signal band thus practically only consists of the signal component at f_{in} , without any of its harmonics. Such a practically-distortion-free signal band then allows demodulation with a low-pass filter that can reject the out-of-band high-frequency modulation products.

The PFM output with rectangular pulses, expressed in (4.5), can be thought of as one obtained by passing the impulse-form output in (4.6) through a filter that converts an input impulse into a pulse of fixed width, T_{PW} . Such a filter has an impulse response given by $h(t) = rect(\frac{t}{T_{PW}}) - \frac{1}{2}$ and presents a low-pass sinc transfer function²¹ given by: $H(f) = T_{PW} sinc(\pi f T_{PW}) e^{-j\pi f T_{PW}}$, which provides spectral nulls at integer multiples of $1/T_{PW}$. Therefore, a non-zero pulse width gives a low-pass filtering effect that can be used to limit the bandwidth of the output spectrum. The higher the pulse width, T_{PW} , the better the attenuation at high frequencies. However, T_{PW} has to be kept smaller than the minimum time between two consecutive output pulses.

²¹ This explains the sinc terms in (4.5).

A comparison with conventional frequency modulation (FM) [63] is now in order. Like PFM, the output spectrum of an FM signal also contains an infinite number of modulation products. However, unlike PFM, it does not contain a component in the signal band; in fact, all the modulation products together represent the *desired* signal in FM, whereas in PFM they are undesired. In the latter, we thus focus on getting the baseband component and on rejecting all the modulation components. Demodulation in PFM thus only requires low pass filtering unlike in FM, where a discriminator is needed [63]. It is worth noticing that the voltage signal at the VCO output in Fig. 4.11 is frequency modulated, whereas that at the E2P output is pulse frequency modulated.

We conclude this section by noting the equivalence of PFM encoding with that found in an important biological system—the brain. In PFM, analog information modulates the repetition rate of a unipolar pulse train; the latter's average then represents the encoded information and can be obtained by low-pass filtering. This form of encoding is used by the brain for communication in the nervous system [19]: Neural spikes encode analog information in their repetition rate; the response of a synapse to a nerve impulse input has low-pass characteristics necessary for demodulation [19]. Given that natural selection has forced biological systems to evolve to optimal states vis-à-vis certain criteria²² [19], there is an added motivation for us to consider this form of encoding.

2. *Phase-domain model*: In order to connect the VCO-based PFM encoder with VCO-based DT ADCs, we consider the phase-domain model of the PFM encoder. A general bandpass signal can be represented by a function of the form

²² These criteria could be coding efficiency, robustness, distortion etc. [19].

$$x(t) = a(t)\cos(\omega_c t + \theta(t)) \tag{4.8}$$

where a(t) is the time function for the amplitude, ω_c is a constant, and $\theta(t)$ is the excess phase of the signal. Assume that a(t) is constant and equal to a_m . Using these assumptions and (4.8), the general VCO output can then be expressed as²³:

$$x(t) = a_m \cos(\omega_c t + \theta(t)) \tag{4.9}$$

We define the term inside the bracket, $(\omega_c t + \theta(t))$, as the complete phase, $\phi(t)$, of the VCO output. The instantaneous angular frequency, ω_i , of the VCO output is then defined as the derivative of the complete phase; i.e. $\omega_i = \frac{d\phi(t)}{dt}$. Notice that this angular frequency of the VCO output is a function of continuous time because the phase, $\phi(t)$, is also one. Therefore, by extension, we can also consider the frequency of the PFM output as a function of continuous time, and (4.3) will be valid.

Conversely, we can express the output phase using the instantaneous angular frequency as

$$\phi(t) = \int_{-\infty}^{t} \omega_i(r) dr \tag{4.10}$$

As described in the previous section, the instantaneous angular frequency of the VCO is related to its applied input, v_{in} . It can be expressed as

$$\omega_i(t) = 2\pi [f_c + K_{VCO} v_{in}(t)] \tag{4.11}$$

²³ For the sake of this analysis, we consider a VCO that has a sinusoidal output, without loss of generality and without affecting the equation for the phase signal below. The binary signal can be obtained by passing the sinusoidal signal through a zero-crossing detector.

From (4.10) and (4.11) we get

$$\phi(t) = 2\pi \int_{-\infty}^{t} [f_c + K_{VCO} v_{in}(r)] dr$$
 (4.12)

$$\phi(t) = \varphi_0 + 2\pi \int_0^t [f_c + K_{VCO} v_{in}(r)] dr$$
 (4.13)

where φ_0 is the initial value of the phase, ϕ , of the VCO output, at time t = 0. Assuming it is 0 for simplicity, we can say that, the phase of the VCO output represents the integral of the scaled (by $2\pi K_{VCO}$) version of the input signal with an offset equal to $2\pi f_c$.

For an example test tone input (not shown) the PFM encoder output (amplified by $10\times$ for better visibility) is shown in Fig. 4.14. The VCO output phase signal from (4.13) normalized to 2π is also plotted²⁴. Note that this signal is not explicitly observed anywhere in the system. It is implicit and can only be interpreted through the number of oscillations completed by the VCO output: every oscillation of the VCO output represents a phase change of 2π ($\frac{\phi}{2\pi}$ increases by 1); a phase change of π corresponds to half an oscillation ($\frac{\phi}{2\pi}$ increases by 0.5); and so on. If the phase signal were quantized along the phase axis with a quantization step of π , the quantized signal, $\phi_q/2\pi$, would have a waveform shown in Fig. 4.14. We next observe that the pulses at the PFM encoder output, shown in Fig. 4.14, occur exactly at the instants where the quantized phase signal, $\phi_q/2\pi$, makes a step transition. We can then model the PFM encoder as a system that takes an analog input, implicitly generates the phase signal of (4.13), quantizes it with a step of π , and then converts the quantized signal into a stream of fixed-width pulses, with the timing of the pulses

²⁴ Thanks to the offset $2\pi f_c$, this signal increases in a monotonic fashion, as can be seen in Fig. 4.14.

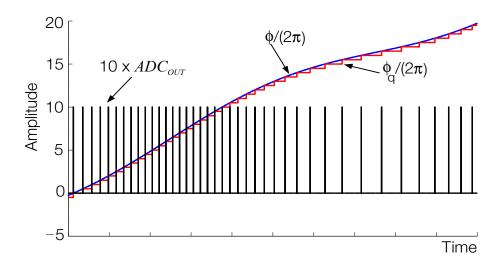


Fig. 4.14. Example waveforms to demonstrate the phase-domain model of the PFM encoder. The encoder output is amplified by $10 \times$ for better visibility.

coinciding with that of the transition of quantized phase signal. The latter can be obtained by passing the quantized phase signal through a Δ block (similar to that used to model the CT ADC in Chap. 3) that converts every transition in the quantized signal to a pulse of fixed width. It thus has properties similar to a differentiator. The complete model is shown in Fig. 4.15. Comparing it with the model in Fig. 3.6, developed for the CT ADC in Chap. 3, we observe that the PFM encoder can, in principle, be derived from the CT ADC in Chap. 3, by scaling its input, v_{in} , by $2\pi K_{VCO}$ and by applying a DC offset equal to $2\pi f_c$ at its input.

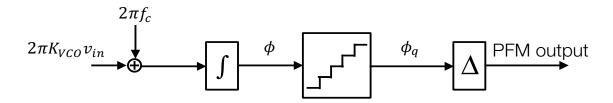


Fig. 4.15. The PFM encoder is modeled as one that integrates the input signal with an offset to generate the phase signal of (4.13), then quantizes this signal, and produces a pulse at every step of the quantized signal. The latter operation is achieved through the Δ block.

3. Connection with DT VCO ADCs: Having developed a phase-domain model for the VCO-based PFM encoder, we can now easily relate it to DT VCO-based ADCs. The latter, too, have all the blocks shown in the model in Fig. 4.15 and in addition, have a sampler. In fact, in Ref. [64], it was shown that a DT VCO ADC can be modeled as a cascade of a PFM encoder and a sampler (Fig. 4.16). Therefore, the CT VCO ADC we have proposed in this section can be thought of as the system one gets by removing the sampler from the DT VCO ADC model²⁵.

4.3.3 Simulation Results

The system in Fig. 4.10(a) was simulated in MATLAB using behavioral code. Similar to case with the PWM encoder, input signals with frequency, f_{in} , in the range [200 Hz, 4 kHz] and with amplitude in the range [-1, 1] were considered, and the zero-input oscillation frequency of the VCO, f_c , was set to 10 kHz. The VCO output frequency range needs to be set such that the modulation products in the output spectrum that fall inside the signal band (i.e. < 4 kHz) have

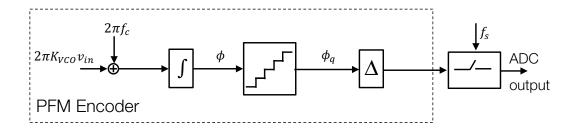


Fig. 4.16. A DT VCO ADC can be modeled as a cascade of a PFM encoder and a clocked sampler.

²⁵ Note the symmetry of this line of thought with that for LCS CT ADCs: An LCS CT ADC is what one gets by removing the sampler from a general DT ADC (leaving only the quantizer) [1].

121

negligible magnitudes. For these simulations, the VCO output frequency range was [9 kHz, 11 kHz] ($K_{VCO} = 1$ kHz/V; $\Delta f_p = 1$ kHz).

1. Spectral characteristics: The PFM-based system has spectral properties described in Sec. 4.3.2. These are similar to that of the VCO-based PWM encoder discussed in Sec. 4.2. Fig. 4.17 shows the ADC output spectrum for the PFM scheme for a full-scale single-tone input at 200 Hz for two cases of the ADC output: (a) one with a rectangular pulse with width, $T_{PW} = 40 \, \mu s$ (Fig. 4.17(a)); and (b) one with a small pulse width, thereby approximating the pulse to an impulse of finite strength²⁶ (Fig. 4.17(b)). In both cases, the output spectrum consists of the signal component and modulation products at $mf_0 \pm nf_{in}$ ($m, n \in I, f_0 = N_{phi} \times f_c = 20 \, \text{kHz}$), but they roll off at high frequencies in case (a) and remain strong in case (b). This is due to the low-pass sinc magnitude response of having a pulse of non-zero pulse width described in Sec. 4.3.2.

Similar to the VCO-based PWM encoder, there are no distortion components in the signal bandwidth. A high in-band SER is achieved, limited only by noise. The out-of-band components can be rejected using a low-pass filter. Like the multi-phase VCO-based PWM encoder in Fig. 4.6, N_{phi} phases of the VCO can be considered in the PFM encoder too, so as to push the modulation products to even higher frequencies (integer multiples of $N_{phi} \times f_c$) [54]. For these simulations, however, we consider only two phases (i.e. rising and falling edges). For a two-tone input, with the two tones at 200 Hz and 2 kHz, the ADC output spectrum shows no significant distortion components in the 4 kHz bandwidth, just like the case with the PWM encoder (in-band spectrum

²⁶ The exact strength of the impulse is immaterial as the magnitude spectrum is normalized to its value at DC.

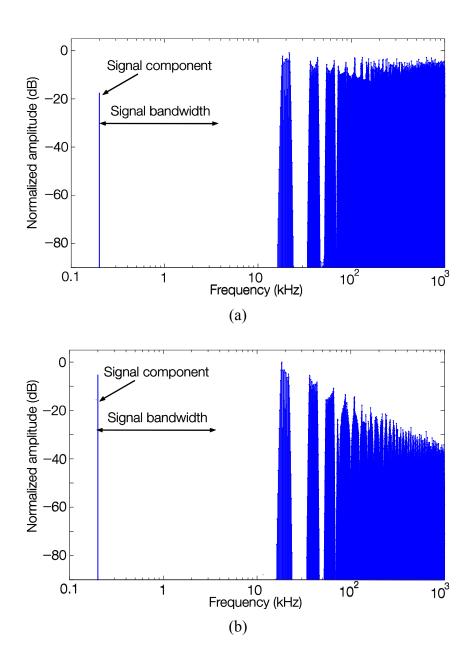


Fig. 4.17. Spectra of the PFM-encoded output for a single-tone input at 200 Hz with the output (a) in impulse-form and (b) with $T_{PW} = 40 \mu s$.

is identical to that in Fig. 4.8). Finally, we note that, similar to the VCO-based PWM encoder, the PFM encoder, too, suffers from aliasing; a spectrum depicting this for the VCO-based PWM encoder was shown in Fig. 4.7.

- 2. Example CT DSP: The transversal structure of an FIR CT DSP can be directly interfaced with the VCO-based PFM encoder. The frequency response and in-band output spectrum is similar to that shown in Fig. 4.8 for the VCO-based PWM encoder in Fig. 4.3(b)-(c). The practical considerations of interfacing the VCO-based PFM encoder with an FIR CT DSP are discussed in Sec. 4.3.4.
- 3. Output token rate (NTPS) and granularity (T_{GRAN}): Every pulse at the output of the PFM encoder represents an output token (one pulse is one token). This is in contrast to the PWM encoder, where every pulse edge is considered a token (one pulse is then two tokens). This is because in PWM the analog information is encoded in the width of the pulses, and therefore, every pulse edge, being central to signal representation, has to be precisely preserved. In PFM, on the other hand, the analog signal is encoded in the density of the fixed-width pulses. Therefore, only rising (or only falling) edges of the encoder output need to be preserved; once a rising (falling) edge triggers the delay line in the DSP, an appropriate pulse width can be ensured at the output of the delay cells, without necessitating the preservation of every falling (rising) edge of the input pulse along the delay-line.

The *NTPS* of the VCO-based PFM encoder for a single-tone-input case is independent of the input frequency and is approximately equal to $f_0 = N_{phi} \times 2f_c$. This is because a pulse is produced following every edge of the VCO output (two edges per cycle), and the output has an average oscillation frequency of f_c (cycles per second). The minimum inter-sample time between encoder output pulses, T_{GRAN} , occurs when the VCO output frequency is at its highest (pulses have the highest density), and is equal to $1/(f_0 + \Delta f_p)$. Since we consider a single phase of the VCO output, $N_{phi} = 1$; also, $f_c = 10$ kHz and $f_0 + \Delta f_p = 11$ kHz. This results in an *NTPS* of 2×10 kS/s = 20 kS/s, and $T_{GRAN} = 45$ μ s.

Parameter		LCS system (8-bit)	VCO PWM system	VCO PFM system
NTPS		102.4 kS/s-2 MS/s (200 Hz to 4 kHz)	40 kS/s	20 kS/s
T_{GRAN}		300 ns	50 μs	45 μs
DSP power	$P_{Delay ext{-}line}$	P_1	$P_1/434$ to $P_1/8500$	$P_1/868$ to $P_1/17000$
(P_{DSP})	P_{Adder}	P_2	$P_2/2.5$ to $P_2/50$	$P_2/5$ to $P_2/100$
DSP delay-line area ($A_{Delay-line}$)		A	A/85	A/85
In-band quantizaton distortion?		Yes	No	No

Table 4.2. Comparison of the VCO-based PFM encoder system with an 8-bit LCS CT ADC/DSP system and the VCO-based PWM encoder system for identical CT DSP specifications.

The PFM-encoder-based CT ADC/DSP system is compared against the PWM-encoder-based one from Sec. 4.2 and with an 8-bit LCS CT ADC/DSP system in Table 4.2, for identical input characteristics and CT DSP specifications. Since comparison between PWM and LCS was made in Sec. 4.2 and the benefits of PWM over LCS were already highlighted there, here we focus on a comparison between PFM and PWM. Neither of the latter two will ideally produce any significant in-band distortion in the spectrum of their respective encoded outputs. The SNDR will thus be limited by the random noise produced by the encoders. As can be seen in Table 4.2, compared to the PWM encoder, the PFM encoder will achieve a $2\times$ reduction in the *NTPS*, with about the same T_{GRAN} . Based on this and (1.1)-(1.2), we conclude that, thanks to the reduction of *NTPS*, a PFM encoder can potentially lower the power dissipation of the subsequent CT DSP by $2\times$ compared to what it would be if a PWM encoder were used. Just as before, in these comparisons, we assume that the energy/token and chip area of the delay cell in the CT DSP remain

independent of the delay value [25]. The area of the delay-line would be similar in the two. The improvements over LCS are thus sustained even in the case of a PFM encoder.

4.3.4 Practical Considerations

The VCO-related nonidealities like its nonlinearity, phase noise, and drift are common to both the PWM encoder and the PFM encoder. They have been discussed in Sec. 4.2.3 and will not be discussed here as their effect on the PFM encoder is similar. We instead consider the practical considerations from the point of view of an integrated implementation of a VCO-based CT ADC/DSP/DAC system.

PFM results in a unipolar encoding of the analog input. This makes delaying the 1-bit CT digital output of the encoder along an asynchronous digital delay-line of the subsequent CT DSP relatively easy. Any mismatch in the delay cells of the delay-line only affects the filter transfer function implemented by the CT DSP and does not create any distortion products in the signal band. This is in contrast to the PWM encoder discussed in Sec. 4.2, which is quite sensitive to mismatch in the parallel delay-lines. The PFM output is 1-bit CT-digital-encoded, making multiplication in the CT DSP simple, as it will be implemented using pass-gates. In order to maximize energy efficiency, addition can then be implemented in the analog domain using an ON/OFF current source with the current value set based on the multiplication coefficient [10] (more on this in the next chapter). The high-frequency components in the output spectrum of the ADC need to be rejected using a post-filter. Such a filter can be a part of the CT DSP or also can be placed after the CT DAC. We conclude that PFM retains most of the benefits of the PWM encoder, while, unlike the latter, being inherently more robust to mismatch.

The need for an anti-aliasing filter, unlike in LCS, in the VCO-based PFM (or PWM) encoder can incur a significant power dissipation penalty. Besides, unlike LCS, the PFM (or PWM) encoder is not event driven and produces output samples even for a zero input. This results in a waste of energy for encoding and processing even when the input is absent. Therefore, the choice of the PFM encoder over LCS needs to be made by carefully considering the application at hand. For low-power applications that deal with burst-like input signals with long periods of inactivity, LCS may continue to be the better choice²⁷. PFM can instead be considered for applications that involve signals with high degree of activity and where anti-aliasing constraints are not too tight.

For instance, consider the case of feedforward equalization (FFE) in wireline systems [65]. High data-rate signals undergo frequency-dependent (usually low-pass) attenuation over a wire. In some cases, this loss is corrected at the receiver end using an FFE, implemented using a FIR²⁸ filter, before a 1/0 decision about the received symbol is made. In such an application, an antialiasing filter is not necessary as the wireline channel acts as one [65]. Besides, given the high activity of the signals involved, the system not being event-driven is not necessarily a show-stopper for the application. The only important target is to equalize the channel loss with a high energy efficiency. This makes the proposed system appropriate for the application.

-

²⁷ After all, all comparisons made in this chapter have been for tone-based inputs, which do not help make the case of LCS.

²⁸ The coefficients of the FIR taps are defined by the desired impulse response of the filter. The latter is chosen such that the convolution of the impulse response of the channel and that of the filter is the delta function.

4.3.5 Conclusions

VCO-based PFM encoding has the following distinct advantages over existing LCS-based CT DSP systems:

- It allows 1-bit CT digital encoding and can significantly lower the *NTPS* and relax the T_{GRAN} ; this can lower the power dissipation and chip area of the CT DSP drastically;
- Superior spectral qualities: the in-band spectrum contains much lower distortion components than that in LCS; and
- If the VCO is implemented using a VCRO, the entire system is composed of inverters, digital delays, flip flops and combinational logic circuits, making it highly digital and technology scaling friendly.

The above advantages are obtained at the expense of aliasing. Besides, the encoder is also not event driven. We have seen that this can be a non-issue in some applications.

4.4 Chapter Summary

In this chapter, we studied two different modes of CT A/D conversion using VCOs. The study revealed that a VCO-based PFM encoder can potentially achieve a drastic reduction in the power dissipation and chip area of a CT ADC/DSP/DAC system compared to existing LCS-based ones. We thus consider an integrated implementation of a CT ADC/DSP/DAC system using a VCO-based PFM encoder in the next chapter. We will use the benefits of the latter, to achieve an energy efficiency that is significantly better than existing LCS-based systems.

Chapter 5

A Delay-Based CT ADC/DSP/DAC System

5.1 Introduction

Much of the work in this thesis is based on the contention that while CT DSP has great potential, it is severely constrained in its energy efficiency due to the preceding CT A/D encoder. As discussed in Chap. 1, much of the prior work in CT DSP systems is based on LCS encoders, which show an exponential worsening of CT DSP constraints—NTPS and T_{GRAN} —as encoder resolution increases. Consequently, once the CT encoder is fixed, there are very few options other than brute-force parallelization [10] left to the designer to optimize the CT DSP. This has restricted prior CT DSP work to either low resolution [10] or low bandwidth [3]. While the energy-efficient encoder proposed in Chap. 3 addresses this issue by adopting a novel 2-bit modulation scheme, increasing the input bandwidth it can handle (currently 40 MHz) while retaining its resolution (5-7 bits) is not trivial.

This motivates the need for the VCO-based PFM encoder proposed in the previous chapter. This encoder promises (a) a significant reduction in NTPS and a relaxation of T_{GRAN} , and as a result, lower CT DSP power; (b) superior in-band spectral properties; and (c) a highly-digital implementation with affinity to scaling and amenability to a low-supply implementation. While such an encoder does suffer from aliasing and is not event driven, applications exist where these are not issues. For instance, feed-forward equalization in wireline receivers, where antialiasing filters are not required and the system input is rather active, was discussed in the previous chapter.

In this chapter, we aim to demonstrate the principle of VCO-based PFM encoding and its associated advantages vis-à-vis CT DSP. We thus describe the implementation of an integrated CT ADC/DSP/DAC system based on VCO-based PFM encoding. The principle is general, and the chosen specifications (see below) are for proof-of-concept. In the process, we will show how such a system can achieve an order-of-magnitude improvement over existing state-of-the-art CT DSP systems [10] and be on par with state-of-the-art DT DSP systems.

5.2 Top-level Architecture

Fig. 5.1 shows the top-level architecture of the proposed system. It consists of a VCO-based PFM encoder—the CT ADC in the system—that converts an analog input signal into a train of fixed-width 1-bit CT digital pulses, whose repetition rate—or frequency—varies in proportion

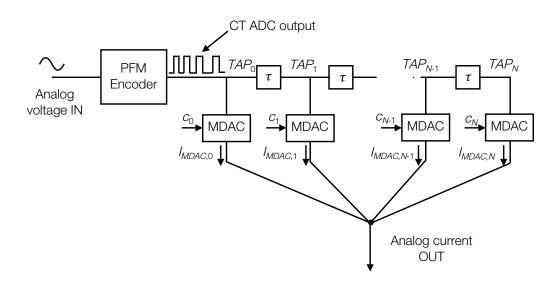


Fig. 5.1. Top-level architecture of the PFM-encoder-based CT ADC/DSP/DAC system. The PFM encoder produces a 1-bit pulse train at its output, and the CT DSP delays it along a tapped delay line composed of asynchronous delays (labelled τ). The multiplying DAC (MDAC) multiplies the pulses at each tap output with a coefficient c_i and outputs a proportional current. The output currents of all MDACs are summed by shorting their outputs together and connecting to a low-impedance node.

to the analog input amplitude: the higher the amplitude, the higher the frequency and vice versa. This pulse train, which constitutes the CT ADC output, is then fed into an FIR CT DSP. A typical FIR CT DSP consists of a delay line, implemented as a cascade of asynchronous digital delays (block represented by τ , the tap delay, in Fig. 5.1); coefficient multipliers; and an adder. In a general CT ADC/DSP/DAC system (see, for example, Fig. 1.2), the adder output is then fed to a CT DAC, which generates a CT analog output.

In our system, thanks to the 1-bit encoding, the coefficient multiplier at each FIR tap is implemented as a simple pass gate and combined with a DAC to form a multiplying DAC (MDAC in Fig. 5.1), similar to that in Ref. [10]. The DAC at each tap outputs a current proportional to the multiplier output, which is the set filter tap coefficient. The output nodes of all DACs are shorted to perform addition in the current domain, thereby generating a current output, which is then taken off-chip²⁹. The system thus has a CT analog voltage input and a CT analog current output.

All the blocks in the system in Fig. 5.1 other than the PFM encoder present straightforward design choices, drawn from considerable work in the past [3], [10], [16]. In Chap. 4, we saw one possibility of implementing the PFM encoder: a VCO followed by an edge-to-pulse converter. In this chapter, we will consider a different possibility—one that will interface better with the following CT DSP. Before that, however, we will discuss the choice of the tap delay, τ , in the DSP delay line, as it will inform the design choices for the PFM encoder.

²⁹ Bringing current-mode output off-chip is easier than doing the same with a voltage-mode output due to pad parasitics etc.

5.2.1 Choice of Tap Delay, τ

The CT DSP serves two filtering functions: (a) To implement the desired transfer function for in-band signals (e.g. low-pass, band-pass etc.) and (b) to reject the strong out-of-band modulation products that result due to PFM (discussed in Chap. 4). A single composite filter transfer function is then synthesized by co-designing to achieve both the desired filtering functions. The first step in such design is the choice of tap delay, τ , in the CT DSP. Once τ is chosen, the desired filter coefficients can be obtained using the *fdatool* in MATLAB.

Let us consider a PFM encoder with a zero-input pulse repetition frequency of f_0 and a maximum frequency deviation Δf_p (see Chap. 4 for definitions). Let the input signal bandwidth be f_{BW} . A representative spectrum of a PFM-encoded signal is shown in Fig. 5.2. It shows out-of-band modulation product lobes centered at kf_0 ($k \in I$); each lobe has strong components in the

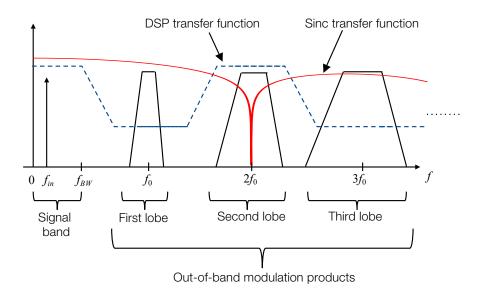


Fig. 5.2. The out-of-band modulation products are rejected using a combination of the CT DSP transfer function and the sinc transfer function created by using a non-zero pulse width for the PFM output.

bandwidth $kf_0 \pm k\Delta f_p$. The first lobe (k = 1) is the most critical one, as it is the closest to the baseband (marked with f_{BW} in Fig. 5.2). The CT DSP can reject this lobe if its composite transfer function has a stop-band³⁰ placed between $f_0 \pm \Delta f_p$. Such a transfer function is shown in Fig. 5.2. Since the transfer function of the FIR filter repeats every $1/\tau$, it is clear from Fig. 5.2 that for the stopband to be centered around f_0 , we would need $f_0 = \frac{1}{2\tau}$, or

$$\tau = \frac{1}{2f_0} \tag{5.1}$$

The repetition of the FIR transfer function every $1/\tau$ results in rejection of the modulation products centered in the filter stopbands, i.e. f_0 , $3f_0$, $5f_0$, ...; those centered in the filter passbands, i.e. $2f_0$, $4f_0$, $6f_0$, ... are not rejected. The latter can be rejected as follows. Recall from Sec. 4.3.2 that a non-zero pulse width, T_{PW} , in the PFM output pulses results in a sinc shaping of the output spectrum for an impulse-form output. Such a sinc filter has spectral nulls at integer multiples of $1/T_{PW}$. If the pulse widths of the PFM pulses are made equal to τ (i.e. $T_{PW} = \tau$), these nulls will fall at $1/\tau$, $2/\tau$, $3/\tau$, ..., or from (5.1) at $2f_0$, $4f_0$, $6f_0$, ..., as shown in Fig. 5.2. Therefore, by satisfying (5.1) and by choosing PFM pulses to have widths equal to τ , we can reject most of the out-of-band modulation products to a good extent.

Process, voltage, and temperature (PVT) variations will result in corresponding static/dynamic variations in the tap delay, τ , in an integrated implementation. The center frequencies of the filter-transfer-function stopbands and the spectral nulls in the sinc filter are

 $^{^{30}}$ Note that this stopband falls outside the signal bandwidth, f_{BW} .

proportional to $1/\tau$. Both of them will thus track variations in τ . The modulation products centered at kf_0 ($k \in I$), however, need not necessarily track them. They will do so only if f_0 is derived using the same tap delay, τ . We will discuss this next.

5.2.2 PFM Encoder Architecture

The PFM encoder can be implemented using a VCO followed by an edge-to-pulse converter as described in the previous chapter. However, in such a case, f_0 will not track variations in the tap delay, τ . For that the PFM encoder needs to be implemented using the same (or a very similar) tap delay as in the delay line. We will present such a scheme now.

The tap delay, τ , is typically [24], [25] implemented using a voltage-controlled asynchronous digital delay, which, as shown in Fig. 5.3, takes an input CT digital pulse and

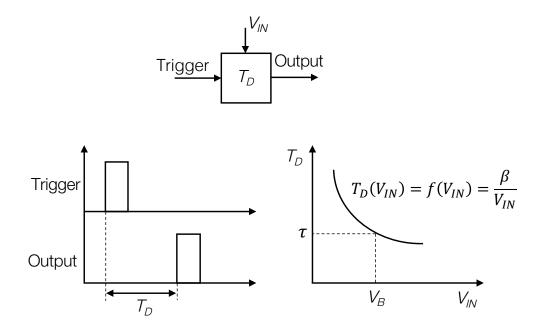


Fig. 5.3. The asynchronous digital delay cell; example time waveforms; and its $T_D(V_{IN})$ characteristic. When input equals V_B , the delay of the cell is τ .

produces a similar output pulse after a delay T_D . This delay can be tuned using the voltage control terminal, V_{IN} . Typically, T_D is inversely proportional to V_{IN} :

$$T_D(V_{IN}) = \frac{\beta}{V_{IN}} \tag{5.2}$$

where β is a constant.

A corresponding plot is shown in Fig. 5.3. When $V_{IN} = V_B$, the nominal delay τ is obtained; such a delay is used in the delay line in the CT DSP.

In order to use this same delay in the PFM encoder, we need to make a VCO using it. Fig. 5.4(a) shows an implementation where two such asynchronous digital delays (D_1 and D_2) are connected in feedback to implement a PFM encoder (the CT DSP is also shown for reference). The voltage control terminals of the delay cells in the DSP delay line are connected to V_B , thereby resulting in a delay of τ . Those of the delay cells in the encoder are connected to $V_B + v_{in}(t)$, where v_{in} (\in [-A, A]) is the analog input that is to be encoded. The thus-formed PFM encoder does not produce a pulse train by default; it needs to be triggered by an external START launch pulse applied to D_1 (see Fig. 5.4(b)). Let v_{in} be 0 for now, so that the control terminal of the cells in the encoder are at V_B , resulting in their delay being $T_D(V_B) = \tau$. The initial START pulse triggers cell D_1 , which produces a pulse at its output, V_1 , after a delay³¹ of τ . This new pulse then triggers cell D_2 , and the latter, in turn, generates another pulse at its output, V_2 after a delay of τ . The pulse at

we will consider a way to implement that in detail later.

³¹ The width of this pulse is immaterial at this stage, provided it is sufficiently large to ensure that the subsequent cell is triggered. The rising edge is what matters to us. This edge then needs to be converted into a pulse of width τ , and

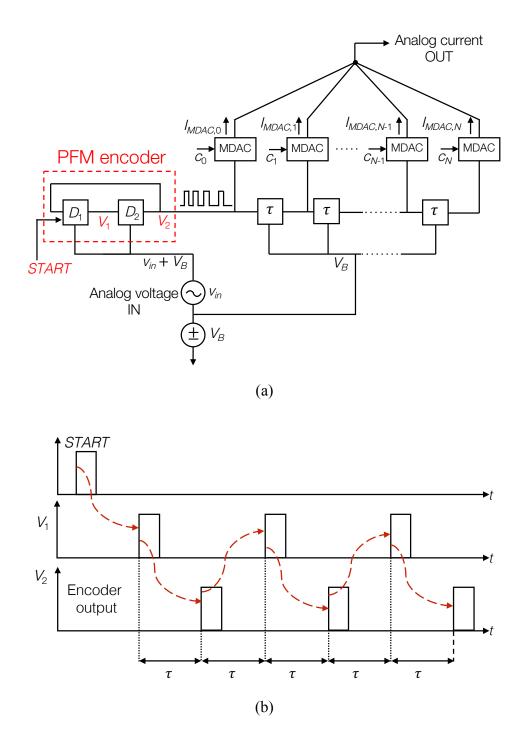


Fig. 5.4. (a) A PFM encoder made out of two asynchronous digital delays identical to the ones in the following CT DSP (also shown); and (b) example waveforms at the PFM encoder terminals following a *START* signal trigger.

formed by the two delay cells, and a unipolar pulse train is produced at the output node, V_2 , which also forms the encoder output (see Fig. 5.4(b)). This output is then fed to the delay line in the DSP.

From the input of D_1 to the output of D_2 , every pulse undergoes a delay of two cells: $2T_D(V_B) = 2\tau$, resulting in a 2τ spacing in between the pulses at output V_2 . This results in a zero input (since $v_{in} = 0$) pulse repetition frequency at output V_2 given by (using (5.2))

$$f_0 = \frac{1}{2\tau} = \frac{1}{2T_D(V_B)} = \frac{V_B}{2\beta}$$
 (5.3)

This relation between f_0 and τ is mandated by (5.1) to ensure the out-of-band modulation products can be satisfactorily rejected by the DSP transfer function.

When an input is applied (i.e. $v_{in} \neq 0$), the pulses at output V_2 are separated in time by $2T_D(V_B + v_{in}(t))$, and the corresponding output pulse frequency will be:

$$f_{out}(t) = \frac{1}{2T_D(V_B + v_{in}(t))}$$
 (5.4)

Using (5.2) and (5.3), we can then write

$$f_{out}(t) = \frac{V_B + v_{in}(t)}{2\beta} = f_0 + \frac{v_{in}(t)}{2\beta}$$
 (5.5)

We can now see that the two-delay encoder in Fig. 5.4(a) will convert an input analog signal into a train of unipolar CT digital pulses, whose frequency/repetition-rate will vary in linear proportion (with an offset) to the input through the relation given by $(5.5)^{32}$. Therefore, it will be a PFM

-

³² Note how an inverse relation between T_D and V_{IN} results in a linear relation between f_{out} and v_{in} (ignoring the offset in (5.5)).

encoder. Comparing (5.5) with (4.3), we also see that the modulator gain is $K_{VCO} = \frac{1}{2\beta}$ and the maximum frequency deviation is $\Delta f_p = \frac{A}{2\beta}$, where A is the input amplitude.

We note the following features about the composite system in Fig. 5.4(a):

- The system, in principle, can be designed using a single delay line composed of a cascade of identical³³ voltage-controlled asynchronous digital delay units: the first two of these are connected in feedback and have their control terminal at $V_B + v_{in}(t)$, thereby forming the PFM encoder; the remaining cells are biased at V_B , and form the tap delays in the CT DSP. Due to the event-driven nature of the delay cell, even two such cells connected in a feedback loop can oscillate: every input trigger to the delay cell will result in an output pulse, which will then circulate in the delay-cell loop forever and produce an oscillatory output waveform (see Fig. 5.4(b)). An odd number (≥ 3) is not necessary as in ring oscillators, which need to satisfy the Barkhausen criterion to generate sustained oscillations. In fact, even a single such cell when connected in a similar feedback loop can oscillate. We choose two cells in order satisfy (5.1).
- As the PFM encoder is composed of digital delays, it falls in the class of delay-based ADCs [66]. The proposed encoder, however, is unique in that its output is CT digital with no sampling in time, unlike other delay-based ADCs [66]. Besides, the notion of using a single digital delay cell to make both the ADC and the DSP delay line promises to simplify the design of the system.
- As τ varies with PVT variations, f_0 will now track it due to the identical delay cells in the ADC and the DSP delay line (and from (5.1)). Therefore, out-of-band modulation products that

-

³³ The delay cells in the ADC can be (and, as we shall see, will be) slightly different from those in the delay line; the important thing is for them to be similar enough so that they track each other with PVT variations.

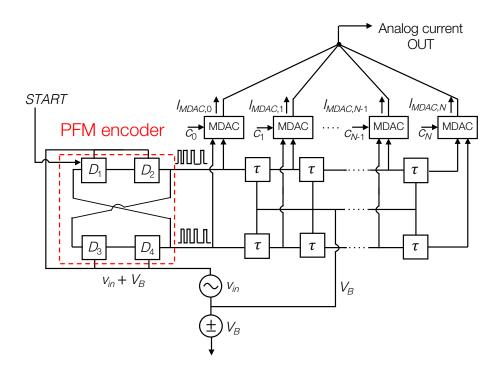


Fig. 5.5. Parallelized version of the PFM-encoder-based CT ADC/DSP/DAC system from Fig. 5.4(a).

result due to the PFM encoding (see example spectrum in Fig. 5.2) also track such variations. We have already discussed how the stopbands of the CT DSP and the nulls of the sinc filter (both of which attenuate the out-of-band modulation products) also track variations in τ . Therefore, the rejection of the out-of-band modulation products that result due to the PFM encoding will be robust to PVT variations.

• In Fig. 5.4(a), every delay cell is either in delay mode or in reset mode. In the encoder, when cell D_1 delays, D_2 is in reset mode and vice versa. The reset operation in one cell, say D_1 (D_2), needs to be completed before the other cell, D_2 (D_1), finishes the delay operation and triggers the former, i.e. D_1 (D_2). Therefore, the amount of time allotted to the reset operation has to be less than or equal to the delay of one cell. In some cases, which we will see later in the chapter, this is not sufficient, especially with PVT variations. To allow for a greater reset duration, parallelization

can then be adopted: the delay line (ADC+DSP) in the system in Fig. 5.4 is duplicated to create the system shown in Fig. 5.5. Four cells, instead of two, now implement the PFM encoder; a START pulse launched at the input of cell D_1 is circulated in a loop of delay cells following the sequence: $D_1 \rightarrow D_2 \rightarrow D_3 \rightarrow D_4 \rightarrow$ (back to) D_1 and so on. Now the reset duration is increased to the delay of three cells and leaves a sufficient margin. The DSP now has two delay lines, as shown in Fig. 5.5. The pulse rate at the input to each DSP delay line is half, and the minimum time between any two consecutive input pulses is twice, that of what it would be in the case of the system in Fig. 5.4(a) with a single delay line. The tap outputs in the two DSP delay lines in Fig. 5.5 are combined in the MDAC to form a single tap output, equivalent to what it would be in the system in Fig. 5.4(a). Such parallelization ensures robustness at the cost of doubling the area and the static power and an increased sensitivity to mismatch between the two parallel paths. Note that the dynamic power dissipation remains the same as, while the delay lines are doubled, the input event rate of each of them is halved compared to that in the single-path system and the energy per event is constant. We will discuss these issues later in the chapter.

We conclude this section by noting an important point. We have made a deliberate attempt to have identical delay units in the PFM encoder and the DSP delay line to ensure that the delay units track each other across PVT variations. This in turn will guarantee that the rejection of out-of-band modulation products in the output spectrum is robust to PVT variations. However, the design choice we make is not necessary to ensure the latter. It could be possible to have a PFM encoder made out of delay units that are not similar to those in the DSP delay line, provided each of the delay units can be calibrated to ensure robustness to PVT variations. The choice we make is thus one of the many possibilities and is certainly not restrictive.

5.3 Integrated Implementation

Now that the top-level architecture has been established, we will go into the details of the integrated implementation.

5.3.1 Specifications and Targets

Our goal is to demonstrate the advantages of the PFM-encoder-based CT ADC/DSP/DAC processor system. To do that we choose the following specifications:

- Input: Bandwidth, $f_{BW} = 600$ MHz; Amplitude, A = 0.2 V ($V_{DD} = 1.2$ V)
- >30 dB in-band CT ADC SNDR (ENOB: 5-6 bits)
- 16-tap ($N_{taps} = 16$) FIR filter with 7-bit programmable filter coefficients

The system will be designed with a 1.2 V supply in ST's 28 nm FDSOI technology. The processor will take a CT analog input voltage signal and produce a CT analog current signal at the output (Figs. 5.4(a) and 5.5). Its number of taps can be programmed by setting the unused tap coefficients to zero. Asynchronous design necessitates calibration circuitry to calibrate the tap delays and other system parameters (discussed later) in order to ensure the desired performance across PVT variations. The system will include an on-chip automatic calibration set up to achieve this. While FDSOI allows the use of the back-bias of the transistor to reduce the threshold, we will try to avoid using it so that the back-bias voltage does not have to be generated separately on chip. Wherever possible, we will connect the back gate of all PMOS transistors to ground and that of all NMOS transistors to the supply voltage.

The above specifications can be appropriate for applications like feedforward equalization in wireline receivers [65] (data rates up to 1.2 Gb/s, for the listed specifications), where SNDR

requirements are modest, but high energy efficiency (power/data-rate/tap) is critical. We do not intend to restrict the proposed processor to this application, but we will try to maximize its energy efficiency so that it can be considered for it in the future.

Power estimation

For this section, we consider the system in Fig. 5.4(a), while noting that the analysis that follows also applies to the system in Fig. 5.5.

The power dissipation of the system can be written as:

$$P_{SYS} = P_{ADC} + P_{Delay-Line} + P_{MDAC} (5.6)$$

The proposed system consists of two major blocks: the delay cell (which defines the power dissipation of the ADC and the DSP delay line) and the MDAC. While both of these blocks dissipate static power, as we will see, their dynamic power consumption is relatively much larger. Therefore, they can be considered to be event driven, dissipating a certain energy per input event. Overall, given the fairly active nature of the input and the event-driven nature of these individual blocks, the total dynamic power dissipation of the system will be much larger than its static counterpart. Therefore, in this analysis, we will only consider the former.

Let the energy/event of the delay cell be E_{DC} (assumed independent of the delay value for reasons discussed in Chap. 3 [25]) and that of the MDAC be E_{MDAC} . We know that the input to each of these blocks is the encoder output, whose event rate³⁴ goes from $f_0 - \Delta f_p$ to $f_0 + \Delta f_p$, with

³⁴ One event is one pulse.

an average rate of f_0 . The power dissipation of each unit block can then be calculated as a product of its energy/event and the average input event rate. Eq. (5.6) can then be written as:

$$P_{SYS} = (N_{DC,ADC} \times E_{DC} \times f_0) + (N_{DC,DL} \times E_{DC} \times f_0) + (N_{MDAC} \times E_{MDAC} \times f_0)$$
(5.7)

where $N_{DC,ADC}$ and $N_{DC,DL}$ are the number of delay cells in the ADC and the delay line respectively, and N_{MDAC} is the number of MDACs in the system. But,

$$N_{DC,DL} = N_{taps} - 1; N_{MDAC} = N_{taps}$$
 (5.8)

Substituting this in (5.7) and simplifying, we get,

$$P_{SYS} = \left[\left(N_{DC,ADC} + N_{taps} - 1 \right) \times E_{DC} + N_{taps} \times E_{MDAC} \right] \times f_0$$
 (5.9)

Since, $N_{DC,ADC} \ll N_{taps}$, $N_{DC,ADC} + N_{taps} - 1 \approx N_{TAPS}$. Eq. (5.9) then becomes

$$P_{SYS} \approx N_{taps} \times (E_{DC} + E_{MDAC}) \times f_0$$
 (5.10)

The system power dissipation per tap is then given as

$$\frac{P_{SYS}}{N_{tans}} = (E_{DC} + E_{MDAC}) \times f_0 \tag{5.11}$$

Dividing both sides by $2 \times f_{BW}$, the equivalent Nyquist sampling frequency for this clockless encoder, we then get the energy per tap:

$$\frac{P_{SYS}}{2f_{BW} \times N_{taps}} = (E_{DC} + E_{MDAC}) \times \frac{f_0}{2f_{BW}}$$
 (5.12)

 $\frac{f_0}{2f_{BW}}$ can be thought of as an oversampling ratio, OSR. Once the OSR, E_{DC} , and E_{MDAC} are known, the power dissipation of the system can be reasonably predicted for a given bandwidth and N_{taps} .

For example, if OSR = 4, and $E_{DC} = E_{MDAC} = 20$ fJ/event (both reasonable numbers), the system will consume 160 fJ/tap from (5.12). When (5.12) is normalized with 2^{ENOB} , we get the figure of merit (FOM) used to compare CT DSP systems (also given in (3.10)) [10]:

$$FOM_{DSP} = \frac{P_{SYS}}{2^{ENOB} \times 2f_{BW} \times N_{taps}} = (E_{DC} + E_{MDAC}) \times \frac{f_0}{2f_{BW}} \times \frac{1}{2^{ENOB}}$$
(5.13)

The lower the FOM the better. For the numbers given above and a targeted *ENOB* of 6 bits, the FOM for the proposed system will be 2.5 fJ/conversion-step/tap. This would be 12× better than the state-of-the-art CT DSP system in Ref. [10]. Note that this analysis is true for the parallelized system in Fig. 5.5 as well since doubling the delay line halves the throughput to each path, keeping the dynamic power dissipation the same.

5.3.2 Delay Cell Design for Delay Line

Operation

The delay cell used in the delay line implements the (constant) tap delay τ shown in Fig. 5.5. Its architecture is based on the one described in Chap. 3 with some modifications and is shown in Fig. 5.6(a) (sizing of the transistors is given in Table 5.1). The delay cell can be in two stable states: delay or reset. These states are held by the NOR SR latch made of gates N_0 and N_1 . Transistor M_C implements a MOS capacitor between the supply, V_{DD} , and the node V_C . Its capacitance in addition to the parasitic capacitances at node V_C define the net charging capacitance of the delay cell. Let this capacitance be C_1 .

During reset, the output Q of the SR latch is 0. The switch made of PMOS transistor M_1 is thus on. There is thus a direct path for the drain current, I_B , of transistor M_0 to flow from V_{DD} . This is in contrast to prior work [10], [25] and the delay cell described in Chap. 3, where such a direct

current path does not exist and all static current is eliminated during reset. We avoid this in order to obviate the need to turn M_0 on/off in a short period of time, especially considering that the delay is already in the ps range. This current, I_B , of transistor M_0 is set through a current-mirror biasing

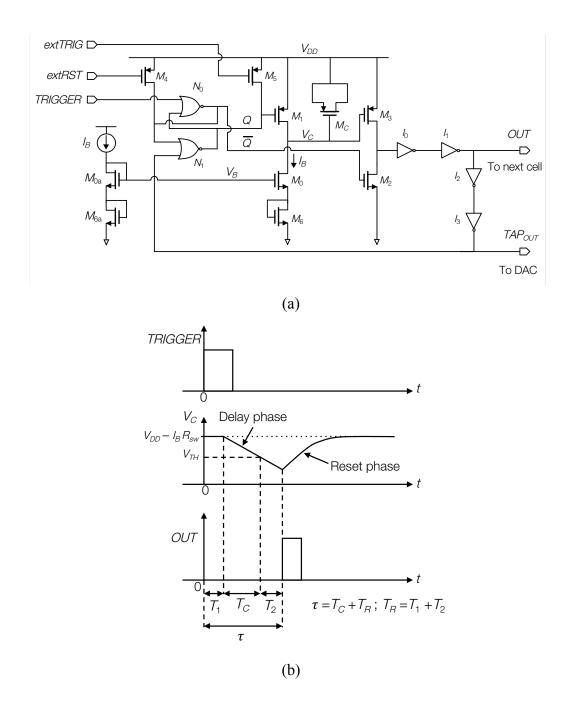


Fig. 5.6. (a) Architecture of the delay cell used in the delay line; and (b) time waveforms depicting operation.

Transistor	Width, $W(nm)$	Length, L (nm)
M_0	366	90
M_1	97	30
M_2	80	30
M_3	1000	30
M_4	400	30
M_5	400	30
M_6	366	30
$M_{ m C}$	980×6	30

Table 5.1. Sizes of transistors in the delay cell used in the delay line, shown in Fig. 5.6(a).

scheme³⁵ which generates the bias voltage, V_B , of M_0 , and is implemented using transistors M_{0a} and M_{6a} (the need for M_6 will be discussed later).

The voltage at node V_C at this point in the reset phase also forms its initial condition and is given by

$$V_C(0^-) = V_{DD} - I_B R_{sw} (5.14)$$

where R_{sw} is the on resistance of the transistor M_1 . This quantity is marked in the waveforms in Fig. 5.6(b). V_C being close to V_{DD} turns off PMOS M_3 . During this reset state, \bar{Q} is 1, thereby turning on M_2 and pulling its drain and, consequently, the delay cell output, OUT, to 0.

When a pulse appears at the *TRIGGER* input of the cell, it sets the SR latch so that now Q becomes 1 (\bar{Q} becomes 0), and M_{1-2} turn off. The current in the transistor M_0 now starts charging

-

 $^{^{35}}$ The delay of the cell is calibrated by adjusting the bias current, I_B , through a 6-bit current DAC, as we will see later.

the capacitance at node V_C , C_1 . As a result, V_C starts falling as shown in Fig. 5.6(b). When it drops to the threshold voltage, V_{TH} , M_3 turns on, charging its drain towards V_{DD} and, following a short delay due to inverters I_{0-1} , making the cell output, OUT, 1. This cell output is connected to the TRIGGER terminal of the next delay cell, and by becoming 1, it triggers the latter out of reset and into delay mode. After a short delay due to inverters I_{2-3} , it also resets the SR latch, so that Q goes back to 0 (\overline{Q} becomes 1) and M_{1-2} turn back on. The buffered output of I_3 , TAP_{OUT} , represents the tap delay output, which is fed onto the MDAC (see Fig. 5.5)³⁶. The capacitor is now discharged so that V_C goes to the value given by (5.14), M_3 turns off, and the cell output goes back to 0, thereby creating a pulse as shown in Fig. 5.6(b). The cell is now in reset mode. The output pulse is ensured to have enough width so that the next delay cell is triggered before the current one goes to reset mode.

An external active-low reset, applied to the gate of M_4 , can be used to force the cell (through the SR latch) into reset mode. The reset pins of all delay cells are shorted together, and connected to an active-low global reset pin. The global reset can then force all cells into reset mode during the initial set up. Similarly, an active-low external trigger, applied to the gate of M_5 , can be used to force a cell (through the SR latch) into the delay mode. This external trigger is useful during calibration when a test trigger is to be applied to the delay line.

 $^{^{36}}$ The delay cell in Fig. 5.6(a) thus has two different outputs: OUT, which triggers the next delay cell, and its buffered version, TAP_{OUT} , which goes to the MDAC. However, in Fig. 5.5 (and also Fig. 5.4(a)), the delay cell in the DSP delay line is shown to have a single output that connects to both the next delay cell and the MDAC. This is done to avoid clutter in the system diagram.

The total delay, τ , of the cell from the rising edge of the input pulse to that of the output pulse (see Fig. 5.6) is given by:

$$\tau = T_C + T_R \tag{5.15}$$

where T_C is the delay due to the charging of the capacitor and T_R is the total propagation delay of the digital blocks given by

$$T_R = T_1 + T_2 (5.16)$$

where T_1 is the propagation delay in the SR latch, from the time of the input pulse's rising edge to the time the capacitor-charging operation starts; and T_2 is the total propagation delay in the inverters I_{0-1} , from the time V_C crosses the threshold V_{TH} to the time OUT becomes 1 (both are marked in Fig. 5.6(b)). Typically, T_R is about 40 ps. T_C is the time taken to charge the capacitor from $V_C(0^-)$ to V_{TH} with current I_B in transistor M_0 . It can be expressed as

$$T_C = \frac{C_1(V_C(0^-) - V_{TH})}{I_B} \tag{5.17}$$

Substituting for $V_C(0^-)$ from (5.14) we can write:

$$T_C = \frac{C_1(V_{DD} - V_{TH})}{I_R} - R_{sw}C_1$$
 (5.18)

Substituting this in (5.15) we get

$$\tau = \frac{C_1(V_{DD} - V_{TH})}{I_R} + T_{RST} \tag{5.19}$$

where

$$T_{RST} = T_R - R_{SW}C_1 \tag{5.20}$$

defines the net reset delay. Eq. (5.19) gives the expression for the nominal delay τ of the delay cell, which will implement the tap delay in the delay line in the system in Fig. 5.5. The factor $R_{sw}C_1$ in (5.20) cancels the propagation delay in the digital blocks, T_R , to some extent, so that the net delay in (5.19) primarily depends only on the first term.

Simulation Results

We choose an $f_0 = 4.2$ GHz (OSR=3.5). From (5.1), the required nominal tap delay will then be $\tau = 119$ ps. It can be achieved with the presented delay cell with a 5 fF capacitor (implemented using the MOS capacitor M_C), $I_B = 13 \mu A$, and $(V_{DD} - V_{TH}) = 0.3$ V. Table 5.2 lists the performance numbers for the delay cell. The static power dissipation in the reset phase on account of not turning off M_0 is 16 μ W. The total active power dissipation varies from 33 μ W to 57 μ W, as the time spacing between input tokens, T_{in} , goes from 850 ps to 320 ps. The cell dissipates 20 fJ/token. Local variations result in a 2σ delay variation of 15% of the nominal delay. The RMS value of the delay jitter is 0.6% of the nominal delay.

Paran	neter	Value
Nominal	delay, $ au$	119 ps
Energy/to	ken, E_{DC}	20 fJ
Delay misr	natch (2σ)	15% of τ
RMS delay	jitter (1σ)	0.6% of τ
Static _j	power	16 μW
Average active power for input tokens spaced by:	320 ps	57 μW
	480 ps	43 μW
	850 ps	33 μW

Table 5.2. Performance summary of the asynchronous digital delay cell in the delay line.

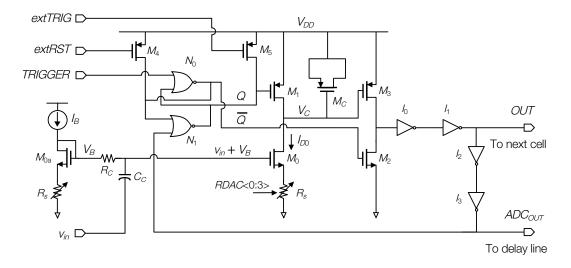


Fig. 5.7. Architecture of the delay cell used in the ADC.

5.3.3 Delay Cell Design for the ADC

The delay cell used in the ADC is shown in Fig. 5.7 (transistor sizes are given in Table 5.3). It is identical to the one in the delay line, shown in Fig. 5.6(a), with only two differences: (a) the diode-connected NMOS M_6 in the latter is replaced with a programmable degeneration resistor, R_s ; and (b) in addition to the bias voltage, V_B , which sets the bias current, I_B , the analog input, v_{in} , is also applied through ac coupling, resulting in a net analog voltage of $V_B + v_{in}$ at the gate of M_0 , as shown in Fig. 5.7. AC coupling is one way of applying the input v_{in} to the ADC. Other possibilities, like DC coupling the input with a DC correction feedback loop, exist and may, in fact, be suitable for certain applications. The proposed principle is general and can be integrated well with such schemes. However, for our purpose of demonstrating the principle, we choose AC coupling without loss of generality. The OUT terminal of the cell is connected to the TRIGGER terminal of the next delay cell in the ADC. The buffered signal ADC_{OUT} is connected to the input of the delay line in the DSP. The external trigger input, extTRIG, similar to the one in delay cell in

Transistor	Width, $W(nm)$	Length, L (nm)
M_0	366	90
M_1	97	30
M_2	80	30
M_3	1000	30
M_4	400	30
M_5	400	30
$M_{ m C}$	980×6	30
RDAC unit resistance (2.44 kΩ)	200	400

Table 5.3. Sizes of transistors in the delay cell used in the ADC, shown in Fig. 5.7.

the DSP delay line, can be used to force the cell into delay mode. This signal is used to provide the START signal to the ADC (as shown in Fig. 5.4(b)) to start the conversion process. To do this, the extTRIG pin in one of the delay cells in the ADC (e.g. D_1 in the systems shown in Figs. 5.4(a) and 5.5) is connected to an external START signal. In all other cells this trigger pin is connected to V_{DD} .

Transistor M_0 together with R_s converts the applied input voltage, $V_B + v_{in}$, at its gate, into a proportional current at its drain given by

$$I_{D0} = I_B(1 + f(v_{in})) (5.21)$$

where the function $f(v_{in})$ defines the V-to-I conversion characteristic. This current will be the charging current of the capacitor C_1 in the delay cell. As the input v_{in} varies around the bias value V_B , so does the drain current of M_0 , I_{D0} , around its bias value, I_B . The degeneration resistor, R_S , helps improve the V-to-I conversion linearity and, as we will see, to control the maximum frequency deviation. It is implemented as an unsilicided poly resistor. If the V-to-I relationship is perfectly linear,

$$f(v_{in}) = \frac{G_m}{I_B} v_{in} \tag{5.22}$$

where G_m , the transconductance, is constant.

The analysis for finding the delay of this cell is similar to that for the one in the delay line with one difference: Whereas the charging current for the delay cell in the delay line is a constant, I_B , for the cell in the ADC it will be I_{D0} , which is not constant, but a function of the input given by (5.21). Eq. (5.14), which defines the initial condition on node V_C , can then be modified as

$$V_C(0^-) = V_{DD} - I_{D0}(0^-)R_{sw}$$
(5.23)

where $I_{D0}(0^-)$ is the value of the drain current of M_0 at the start of the delay phase. Eqs. (5.15)-(5.16), which define the different components (T_C and T_R) of the total delay in the cell in the delay line, apply to the delay cell in Fig. 5.7 as well.

If the input signal has a period (e.g. 100 ns for a 10 MHz input) that is much larger than T_C (e.g. 119 ps), it will not change much over a time duration of T_C and can be assumed constant. Then the drain current, I_{D0} , of M_0 will also not change much over this duration and can be assumed to be constant and equal to its value at the start of the delay phase, $I_{D0}(0^-)$. The time, T_C , taken to charge the capacitor at node V_C from $V_C(0^-)$ to V_{TH} in this cell can then be obtained by analogy to (5.18) as

$$T_C = \frac{C_1(V_{DD} - V_{TH})}{I_{D0}} - R_{sw}C_1 \tag{5.24}$$

The total delay of the cell, from (5.15), will then be

$$T_D = T_C + T_R = \frac{C_1(V_{DD} - V_{TH})}{I_{D0}} - R_{sw}C_1 + T_R$$
 (5.25)

where T_R is the net delay in the digital logic blocks in the cell, and was defined in (5.16). Eq. (5.25) can then be written as

$$T_D = \frac{C_1(V_{DD} - V_{TH})}{I_{D0}} + T_{RST}$$
 (5.26)

where, the net reset delay, is expressed in (5.20). Notice the similarity between (5.19) and (5.26).

To find the expression for T_D in terms of v_{in} , we next substitute in (5.26) the expression for I_{D0} in terms of v_{in} from (5.21) and get

$$T_D(V_B + v_{in}) = \frac{C_1(V_{DD} - V_{TH})}{I_B(1 + f(v_{in}))} + T_{RST}$$
(5.27)

Therefore, the total delay of the cell, $T_D(V_B + v_{in})$, is composed of two components: one that is dependent on the input (v_{in}) , and the other, the reset delay, T_{RST} , which is independent of the input. For a zero input, i.e. $v_{in} = 0$, $f(v_{in}) = 0$, and the resulting delay will be (from (5.27))

$$T_D(V_B) = \frac{C_1(V_{DD} - V_{TH})}{I_B} + T_{RST} = \tau$$
 (5.28)

which equals the nominal tap delay of the delay cell in the delay line, τ , from (5.19) and (5.28).

Now that we know the delay of an individual cell used in the ADC, we can find an expression for the output pulse repetition rate, $f_{out}(t)$, of the ADC, using (5.28) in (5.4), as

$$f_{out}(t) = \frac{1}{2T_D(V_B + v_{in}(t))} = \frac{1}{2} \frac{1}{\frac{C_1(V_{DD} - V_{TH})}{I_B(1 + f(v_{in}))} + T_{RST}}$$
(5.29)

Using (5.28), (5.29) can be simplified as

$$f_{out}(t) = \frac{1}{2} \frac{1 + f(v_{in})}{\tau + T_{RST} f(v_{in})}$$
 (5.30)

For a zero input $(v_{in} = 0, f(v_{in}) = 0)$, the output pulse repetition frequency, f_0 , will be (from (5.30))

$$f_0 = \frac{1}{2\tau} {(5.31)}$$

which matches the expression for f_0 in (5.3). After some mathematical manipulation, (5.30) can be simplified to

$$f_{out}(t) = f_0 \frac{1 + f(v_{in}(t))}{1 + \delta f(v_{in}(t))}$$
(5.32)

where

$$\delta = \frac{T_{RST}}{\tau} \tag{5.33}$$

is the ratio of the reset delay to that of the total delay for a zero input, τ , given by (5.28).

Two different cases can now be considered.

Case I: If $T_{RST} = 0$, $\delta = 0$, and (5.32) simplifies to

$$f_{out}(t) = f_0[1 + f(v_{in}(t))]$$
 (5.34)

If the V-to-I relationship of (5.21) in is perfectly linear, from (5.22), we can write,

$$f_{out}(t) = f_0 \left[1 + \frac{G_m}{I_R} v_{in}(t) \right]$$
 (5.35)

Comparing this expression for the output pulse repetition frequency with the general one from (4.3), we conclude that the maximum frequency deviation of this encoder is proportional to the transconductance efficiency, $\frac{G_m}{I_B}$, and is given by $\Delta f_p = f_0 \frac{G_m}{I_B} A$, where A is the input amplitude.

For the chosen value of $f_0 = 4.2$ GHz (OSR=3.5), we choose a maximum frequency deviation of $\Delta f_p = 1.3$ GHz. As discussed in Sec. 5.3.2, the zero-input delay is $\tau = 119$ ps and can be achieved with the same parameter values as chosen in Sec. 5.3.2 for the delay cell in the delay line. For an input amplitude of A = 0.2 V, the desired maximum frequency deviation can be obtained by setting $\frac{G_m}{I_B} = 1.6$, or $G_m = \frac{g_{m0}}{1+g_{m0}R_S} = 21$ µS, where g_{m0} is the transconductance of transistor M_0 . The output pulse repetition frequency then goes from 2.9 GHz to 5.5 GHz, with an average value of 4.2 GHz. The resulting minimum time between two consecutive output pulses, T_{GRAN} , is 180 ps for the single-path system in Fig. 5.4, and 360 ps for the parallelized two-path one in Fig. 5.5. A 6-bit LCS system for the same specifications would have resulted in a T_{GRAN} of 10 ps, which would clearly be significantly worse (by $18 \times$ or $36 \times$) than the proposed system. R_S is nominally set to 37 k Ω . It is implemented using a 4-bit R-string DAC (RDAC) to allow programmability and calibration.

Case II: If $T_{RST} \neq 0$, $\delta \neq 0$, and we go back to (5.32). We surmise by observing this equation that a "right" choice of the value of δ can help cancel some of the non-linearity created by $f(v_{in})$. We thus aim to determine this specific value of δ . Achieving this using transistor-level simulations of the ADC can be quite cumbersome and slow. An analytical approach can instead be used to quickly grasp the effect of δ on the output distortion. We do this now.

Any non-linearity in the *V*-to-*I* characteristic, modeled through $f(v_{in})$, will result in a non-linear f_{out} - v_{in} relationship in (5.32) and will cause in-band distortion in the output spectrum. To estimate the latter, $f(v_{in})$ was modeled as a third-order polynomial³⁷:

$$f(v_{in}) = av_{in} + bv_{in}^{2} + cv_{in}^{3}$$
(5.36)

The coefficients of this polynomial depend on the drop across the resistor R_S , V_{RS} —the higher the drop, the lesser the effect of the nonlinearity³⁸. The coefficients were found using the V-to-I characteristic of (5.21) obtained from transistor-level simulations. Eq. (5.32) was then simplified as follows.

Let $f \equiv f(v_{in})$. We first expand (5.32) using the Taylor expansion of $\frac{1}{1+\delta f}$ and, assuming that $\delta f \ll 1$, keep only the first three terms in the expansion. The resulting expression is

$$f_{out} = f_0[1+f][1-\delta f + (\delta f)^2 - (\delta f)^3]$$
(5.37)

Substituting (5.36) into (5.37) and simplifying the resulting expression to the third order we get

$$f_{out} = f_0[1 + dv_{in} + ev_{in}^2 + gv_{in}^3]$$
 (5.38)

where

$$d = a(1 - \delta) \tag{5.39}$$

³⁷ Note that $f(v_{in})$ is unit-less (see 5.20).

³⁸ In order to restrict the effect of this non-linearity, V_{Rs} has to be controlled, especially across PVT. This is done through a calibration scheme described later in the chapter.

$$e = (1 - \delta)(b - \delta a^2) \tag{5.40}$$

$$g = c(1 - \delta) - 2\delta ab + 2\delta^2 ab + \delta^2 a^3 - \delta^3 a^3$$
 (5.41)

We thus started with a non-linear V-to-I characteristic for the source-degenerated transistor M_0 in the delay cell and modeled it using (5.36). We then used transistor-level simulations to obtain the coefficients a, b, and c in this model. Using these coefficients and the analytical expressions in (5.39)-(5.41), the coefficients—d, e, and g—for the model of the output pulse repetition frequency as a function of v_{in} , given by (5.38), can be obtained for a given δ . Using these coefficients, which will be functions of δ , the HD₂ and HD₃ can be estimated for single-tone inputs. The latter will also be functions of δ , and are plotted in Fig. 5.8, along with the resulting Total Harmonic Distortion (THD). As can be seen, there are is a *non-zero* value of δ that minimizes HD₂ and

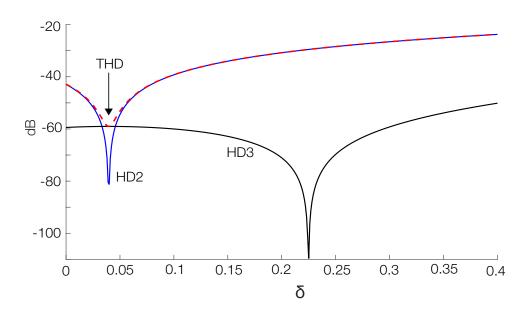


Fig. 5.8. Plot of HD2, HD3, and THD versus δ .

another (distinct) one that minimizes HD₃. THD is minimized when HD₂ is minimized, i.e. the second harmonic is cancelled; the THD is then limited by HD₃.

It is impossible to perfectly cancel the second harmonic across PVT variations. Instead we observe that if $\delta < 0.1$ —i.e. the reset delay, T_{RST} , is less than 10% of the total zero-input delay, τ —the THD is guaranteed to be better than -40 dB. In such a case, noise, and not the distortion due to these non-idealities, will limit the SNDR. We design to achieve this goal. For a zero-input delay of $\tau = 119$ ps, this requires $T_{RST} < 12$ ps, which is quite challenging. Recall from (5.20) that, $T_{RST} = T_R - R_{sw}C_1$, and T_R , the total propagation delay in the digital gates in the delay cell, can be close to 40 ps. Fortunately, the $R_{sw}C_1$ term cancels T_R to some extent, thereby lowering T_{RST} . The resistance, T_{sw} , of the PMOS switch T_{sw} 0 and be adjusted to ensure that the resulting T_{rs} 1 guarantees T_{rs} 2 quarantees T_{rs} 3 or T_{rs} 4 quarantees the desired SNDR, and verify through simulations that it is so across local and global variations. We will see these simulation results later.

Unfortunately, during reset, the capacitance at node V_C in the delay cell discharges through the on resistance R_{sw} of M_1 , with a first-order settling response, with settling time proportional to $R_{sw}C_1$. Increasing R_{sw} to cancel T_R (from (5.20)) also increases this settling time necessary for a complete reset. This is why the top-level system architecture adopted is that of the parallel one in Fig. 5.5, as it allows extra reset time margins that will ensure robustness and also a guarantee that the desired SNDR will be maintained in the face of nonlinearities of the kind discussed above.

Simulation Results

For the delay cell in the DSP delay line (Fig. 5.6), the linearity of V-to-I conversion is not important. Therefore, unlike the delay cell in the ADC (Fig. 5.7) there is no resistor R_s at the source of its current source transistor, M_0 . Removal of R_s from the delay line cell saves area and also the wiring complexity of the 4-bit control of the R-string DAC that implements R_s . The latter's replacement in the delay cell in the DSP delay line is the diode-connected transistor M_6 (Fig. 5.6), which helps to keep the drain voltage of M_0 in the delay line cell close to what it will be in the ADC delay cell (Fig. 5.7). This minimizes the V_{DS} , and hence, current, mismatch between the two cells, resulting in their delays matching very well. Simulations show less than 1 ps mismatch in the nominal delay (that for zero input) between the two types of cells across PVT variations.

The plot of the cell delay, T_D , versus $v_{in} + V_B$ is shown in Fig. 5.9, confirming the inverse relationship defined in (5.2). Table 5.4 lists the performance numbers of the delay cell used in the ADC. They are very similar to those of the delay cell designed for the delay line, listed in Table

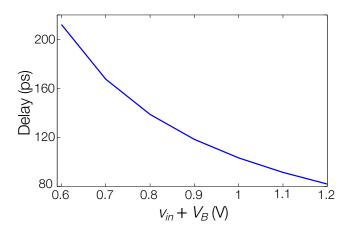


Fig. 5.9. Plot of delay of the asynchronous delay cell in the ADC versus input voltage, $v_{in} + V_B$.

Parameter		Value	
Delay range		82 ps – 212 ps	
Energy/token, E_{DC}		20 fJ	
Delay mismatch (2σ)		9% of τ	
RMS delay jitter (1σ)		0.5% of τ	
Static power		16 μW	
Average active power for input tokens spaced by:	320 ps	57 μW	
	480 ps 43 μW		
	850 ps	33 μW	

Table 5.4. Performance summary of the asynchronous digital delay cell in the ADC.

5.2. The only difference is that local variations result in a 2σ delay variation of 9% of the nominal delay and the RMS delay jitter value is slightly lower (0.5% of τ).

5.3.4 MDAC Design

The MDAC performs the function of coefficient multiplication. As shown in Fig. 5.10(a), at each tap i in the CT DSP, there is a 7-bit current-mode MDAC, which, when triggered by a pulse from the delay line tap output, produces an output current corresponding to the tap coefficient, c_i , for the duration of the pulse. There are 16 taps, and hence, 16 coefficients and 16 MDACs. The exact value of the current is not important. Instead, the ratio of the currents output at different taps in the CT DSP are important for accuracy. Positive coefficients are implemented using an NMOS DAC (NDAC) and negative ones are implemented using a PMOS DAC (PDAC); the 7^{th} bit defines the sign (i.e. whether or not the NDAC/PDAC is on). The sum of the absolute values of all coefficients is proportional to the DAC full-scale current, I_{FS} . In our design, we choose $I_{FS} = 64 \, \mu A$. Then, $I_{LSB} = 1 \, \mu A$.

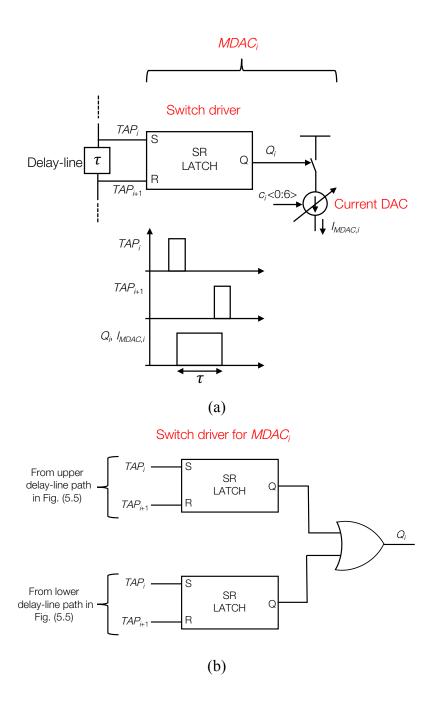


Fig. 5.10. (a) The switch driver for $MDAC_i$ is made of an SR latch, which is set by TAP_i and reset by TAP_{i+1} ; its resulting output, Q_i , then controls the switches of the 7-bit current DAC. (b) For the two-path delay line structure, each $MDAC_i$ has two SR-latches, whose outputs are combined using an OR gate to generate the final control output Q_i .

As discussed in Sec. 5.2, we wish to make the duration of the output current pulse equal to

the tap delay, τ , in order to present a sinc filter to reject the out-of-band modulation products in the ADC output spectrum (see Fig. 5.2). We do this using the tap delay as shown in Fig. 5.10(a). The MDAC at each tap consists of an SR-latch-based switch driver. The stream of pulses produced by the PFM CT ADC moves along the delay line in the CT DSP (see Fig. 5.5). When a pulse arrives at TAP_i (see Fig. 5.10(a)), it sets the SR latch of the corresponding $MDAC_i$, resulting in its output Q_i becoming 1. When the same pulse passes through the following delay cell in the delay line and arrives after a delay of τ at TAP_{i+1} , it resets the SR latch of $MDAC_i$ (and sets the one in $MDAC_{i+1}$), thereby making Q_i go to 0 (and making Q_{i+1} , the output of the SR latch in $MDAC_{i+1}$, go to 1). The resulting pulse on Q_i thus has a duration of τ . When Q_i is 1, the current DAC draws/supplies (depending on the sign of the coefficient) a current, I_{MDAC_i} , set by the coefficient, c_i , from/into a low-impedance output node. Since there are two parallel delay line paths in the top-level system, shown in Fig. 5.5, we use two different SR latches (one for the tap on each path) at each tap and combine the Q outputs of the two SR latches using an OR gate, before connecting it to the switches in the current DAC³⁹, as shown in 5.10(b).

The PDAC/NDAC at each tap is implemented using a current-mode architecture shown in Fig. 5.11 (transistor sizes are given in Table 5.5). It is similar to the architecture in Ref. [10]. As shown in Fig. 5.11, a 6-bit coefficient code, b_{0-5} , defines the output current, I_{DACP}/I_{DACN} , of a binary-weighted DAC, which is copied to transistor M_0 using a current mirror. A pulse on Q_i turns on transistor M_{sw} , thereby producing an equal current, I_{OUT} , that flows into/from the low-impedance output node, normally held at the common-mode voltage, V_{CM} (equal to 0.6 V, half the supply

³⁹ While we could, in principle, use two different DACs for a given tap (one for each path), we choose a single common DAC in order to avoid mismatch issues.

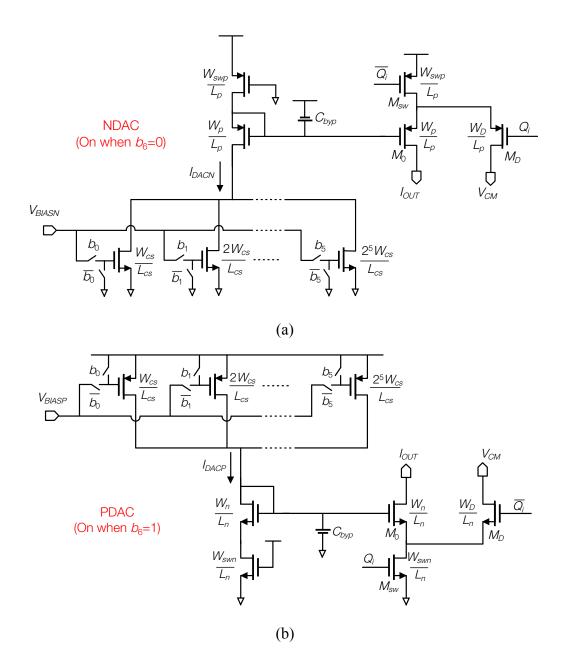


Fig. 5.11. Current-mode (a) NMOS DAC (NDAC) and (b) PMOS DAC (PDAC), 6-bit each, together implement a 7-bit DAC. Both DACs can never be on at the same time.

voltage, V_{DD}). When Q_i is 0, M_{sw} is off, and M_D shorts the source of M_0 to V_{CM} , thereby ensuring that the V_{DS} of M_0 is 0 and I_{OUT} is zero. The 7th bit, b_6 , determines if the PDAC/NDAC is on and, hence, the sign of the output current. Bypass capacitors, C_{byp} , on the bias nodes are implemented

Parameter	Value (nm)
W_{cs}	300
W_n, W_{swn}	120
W_p	750
W_{swp}	150
W_D	80
L_{cs}	90
L_p , L_n	30

Table 5.5. Values for width and length parameters for the current DACs shown in Fig. 5.11.

using MOS capacitors. This architecture is preferred over the current-steering architecture to minimize the effect of charge injection in the steering switches of the latter into the output node.

We will now compute the energy/token, E_{MDAC} , of the DAC by considering its static and dynamic power dissipation. When all the 16 MDACs are taken together, the total worst-case static power dissipation will depend on the sum of the absolute values of all coefficient currents. This sum is proportional to the DAC full-scale current, I_{FS} , and can be expressed as αI_{FS} , where $\alpha \leq 16$ is a proportionality factor. The total worst-case static power dissipation will then equal $V_{DD} \times \alpha I_{FS} = 1.2 \times \alpha 64 \ \mu\text{A} \approx \alpha 77 \ \mu\text{W}$. The digital portion in each MDAC, on the other hand, consumes, 17 fJ/token. For an average token rate of 4.2 GS/s, the 16 MDACs will together, from (5.7), have an average digital power of $16 \times 17 \ \text{fJ} \times 4.2 \ \text{GS/s} = 1.14 \ \text{mW}$. For small values of α , the dynamic power dissipation due to the switching of digital logic nodes can thus significantly exceed the static power; the two are comparable for $\alpha = 16$. From the total (static plus dynamic) power dissipation, we then estimate that each MDAC consumes an energy/token, of $18-35 \ \text{fJ/token}$ for $1 \leq \alpha \leq 16$. Table 5.6 summarizes the performance of the NDAC/PDAC in the system.

Parameter	Value		
Architecture	Current-steering		
Resolution (bits)	6		
Full-scale current, I_{FS}	64 μΑ		
LSB current, I_{LSB}	1 μΑ		
Analog static power	Code-dependent (1.2 μW-77 μW)		
Digital energy/token	17 fJ		
Total DAC energy/token	18-35 fJ		

Table 5.6. Performance summary of a unit NDAC/PDAC.

5.3.5 Calibration

PVT robustness requires calibration. Two different parameters have to be calibrated to ensure desired performance:

- 1. The delay of the delay cells in both the ADC and the delay line needs to be calibrated to ensure the DSP transfer function and ADC performance is robust to PVT variations.
- 2. The degeneration resistor in the delay cell used in the ADC needs to be calibrated to ensure the drop across it, V_{Rs} , is not too high to risk transistor M_0 in Fig. 5.7 going out of saturation or too low to increase the non-linearity in the V-to-I conversion in the delay cell in the ADC.

Both calibration loops are implemented on-chip and are automatically executed in a sequential manner—delay calibration first (Fig. 5.12), followed by calibration of V_{Rs} (Fig. 5.13). They require a clock with a low-frequency of oscillation, f_{CK} , which can be turned off after calibration is done. The delay of the delay cells is calibrated by measuring it using a replica delay line of 16 delay cells connected in a feedback loop and then generating an appropriate calibration code IDAC<0:5> for a 6-bit current DAC that sets the bias current of the delay cells (in both the ADC and the DSP

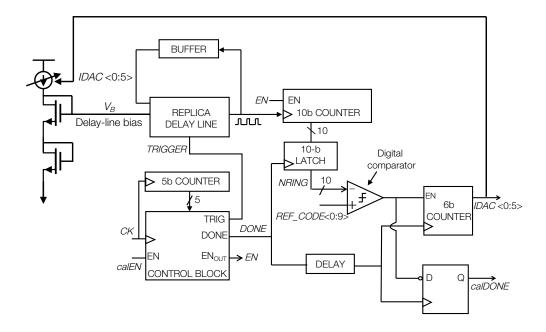


Fig. 5.12. Calibration loop for calibrating the delay of the cells in the delay line.

delay line) (see Fig. 5.12). The drop across the degeneration resistor is calibrated by adjusting the resistance through a 4-bit R-string DAC (RDAC) present in all the four delay cells in the ADC.

Both calibration schemes are described as follows:

- 1. First, an external reset signal resets all delay cells, resetting the codes for both calibration DACs (RDAC and IDAC) in Figs. 5.12 and 5.13 to all 0s.
- 2. Control signal *calEN* (Fig. 5.12) is then asserted and on the first rising edge of the external clock following it, the delay calibration loop is started. The delay calibration loop, shown in Fig. 5.12, is similar to the one in Ref. [10].
- When the delay calibration loop starts, the replica delay line is triggered with a TRIGGER input pulse generated by the control block in Fig. 5.12. The delay line produces a train of pulses at its output with a fixed time spacing, T_{RING} , given by

$$T_{RING} = 16 \times T_D + T_{LOGIC} \tag{5.42}$$

where T_{LOGIC} is the propagation delay of the digital logic blocks necessary to complete the feedback loop in the replica delay line (e.g. a couple of inverters for buffering the feedback path), and T_D is the average delay of a cell in the delay line.

- A 10-bit counter counts the number of pulses produced by the replica delay line; a 5-bit counter counts the number of clock cycles.
- When the clock cycle counter value reaches a threshold value N_{CK} , the control block asserts the *DONE* signal, thereby sampling the value of the 10-bit counter and resetting the delay cells. Let this sampled value be N_{RING} . The value of N_{RING} is given by

$$N_{RING} = \left\lfloor \frac{T_{CK} N_{CK}}{T_{RING}} \right\rfloor \tag{5.43}$$

where T_{CK} is the clock period. Substituting (5.42) in this we get,

$$N_{RING} = \left| \frac{T_{CK} N_{CK}}{16 \times T_D + T_{LOGIC}} \right| \tag{5.44}$$

The sampled N_{RING} value is compared using a digital comparator against a desired value, set through $REF_CODE<0:9>$, and if it is lower than it, it would indicate that the delay of the cell is too large. The 6-bit IDAC code, IDAC<0:5>, is then incremented by 1 (using a 6-bit counter), thereby increasing the bias current in the delay cell; all other counters are reset and the cycle repeats. Since the current is increased, the delay in the next iteration will be lower than that in the current one, and the number of pulses produced by the replica delay line will be higher, resulting in a higher N_{RING} in the next iteration.

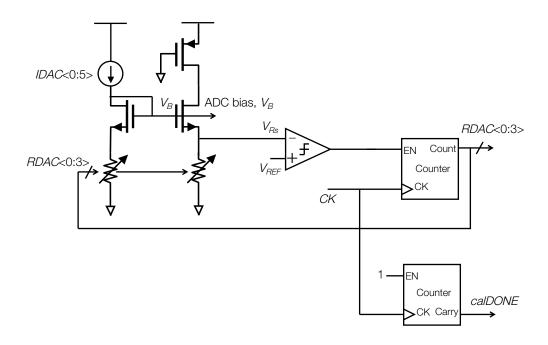


Fig. 5.13. Calibration loop for calibrating V_{Rs} .

- The above cycle repeats while the measured N_{RING} is lower than the desired value. Once it becomes greater than or equal to it, it would imply that the delay is close to the desired value. The output of the 6-bit counter, IDAC<0:5>, which represents the IDAC code, is locked. The delay calibration is complete. The delay value is calibrated with an error of $\pm \frac{1}{N_{RING}} \times 100\%$. A 10-bit counter is thus chosen to allow a large count on N_{RING} , and thus, a delay calibration resolution of < 1ps.
- 3. Once the delay calibration loop is complete, it asserts a *calDONE* signal (see Fig. 5.12) and on the first rising edge of the external clock following it, the V_{Rs} calibration loop is started. This calibration loop is shown in Fig. 5.13.
- At this point in the calibration scheme, the IDAC code is set, but the RDAC code is all 0s, resulting in a small degeneration resistance and a rather low value of V_{Rs} . This value is compared against a desired reference voltage, V_{REF} , using an analog comparator. If it is lower than the V_{REF}

at every clock edge, the comparator output is 1 and the count of a 4-bit counter is incremented by 1. The output of this counter is the RDAC code, *RDAC*<0:3>.

- The above cycle repeats and the RDAC code increments at each clock edge and raises V_{RS} slowly. Eventually the latter exceeds V_{REF} . Once this is done, the comparator output, which is connected to the EN pin of the counter, stays 0, and is not affected by any subsequent clock edges; the RDAC code is thus locked. Once the clock completes 16 cycles, counted through the lower counter in Fig. 5.13, the latter's Carry output becomes 1. This output is the signal calDONE and its assertion indicates completion of calibration. This resets the calibration circuit so that they are turned off thereby saving power.
- Note that the reference voltage V_{REF} is not required to be precise as V_{Rs} itself takes steps of 20 mV, resulting in a similar calibration error, which is fine for the purpose of ensuring linearity of the V-to-I conversion. In this chip, we keep V_{REF} external for simplicity and flexibility. It can be easily implemented on-chip in an industrial version.
- Hysteresis is added to the loop through the analog comparator to avoid unnecessary oscillation.

In the worst-case, the calibration scheme will take 80 clocks cycles (64 to go through all the IDAC codes, and 16 to go through all the RDAC codes). Both calibration loops can be sped up by using binary search algorithm if desired.

5.3.6 System-Level Simulation Results and Comparisons

The system has so far been completely implemented at the schematic level. Here we present transistor level simulations results.

ADC results

The ADC, implemented using four delay cells in a loop (see Fig. 5.5), was simulated at the transistor level. The outputs of the two paths were combined using an OR gate, and the pulse width

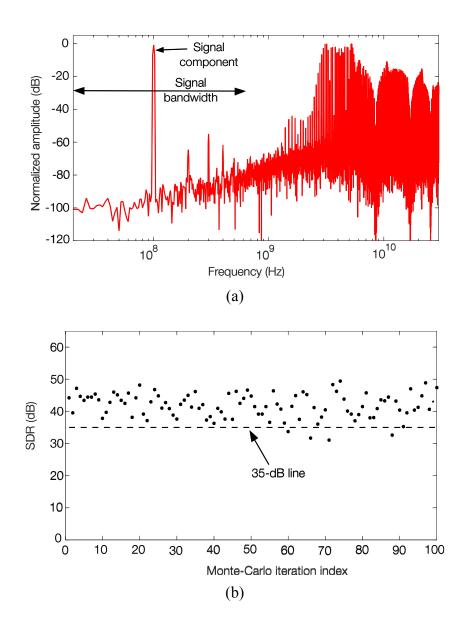


Fig. 5.14. (a) ADC output spectrum obtained from a noiseless transient simulation for a full-scale single-tone input at 100 MHz. (b) SDR vs. iteration number for a Monte-Carlo simulation.

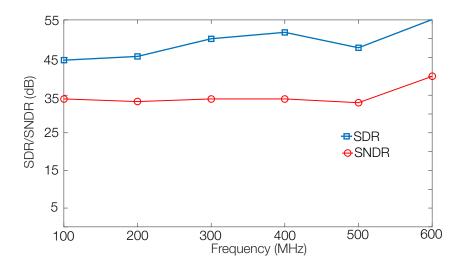


Fig. 5.15. Plot of in-band SDR and SNDR of the PFM encoder output versus single-tone input frequency.

of the OR output was set to equal the tap delay τ in MATLAB. An FFT was then performed using a Hann window. An example output spectrum for a full-scale single-tone input at 100 MHz, simulated with no noise, is shown in Fig. 5.14(a). The out-of-band modulation products are, as expected, centered at integer multiples of $f_0 = 4.2$ GHz. The in-band spectrum consists of second and third harmonics, with the latter dominating and limiting the in-band (0-600 MHz) SDR to 44 dB. To see the effect of mismatch on this, a Monte-Carlo simulation with 100 iteration runs was performed. The SDR for each run was then obtained by performing an FFT as mentioned above. The obtained SDR spread is plotted in Fig. 5.14(b). The SDR is >35 dB in 96% of the runs. Besides, the calibration scheme described in Sec. 5.3.5. will guarantee robustness to PVT variations.

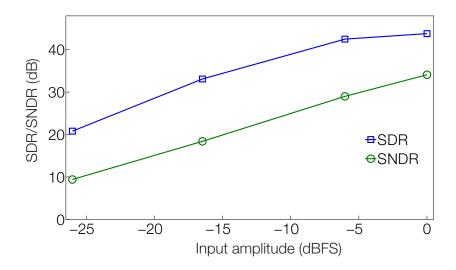


Fig. 5.16. Plot of in-band SDR and SNDR of the PFM encoder output versus input amplitude for a single-tone input at 100 MHz.

The plot of in-band (0-600 MHz) SDR and SNDR of the PFM encoder output versus input frequency for full-scale single-tone inputs is shown in Fig. 5.15. As can be seen, noise⁴⁰, and not distortion due to the non-idealities discussed above, limits the SNDR. The plot of in-band SDR and SNDR of the encoder output versus input amplitude for a single-tone input at 100 MHz is shown in Fig. 5.16. Simulations also show that the 1-dB compression point of the ADC lies beyond its full scale.

Next, a two-tone input with equal-amplitude tones at 450 MHz and 500 MHz was applied, and the resulting output spectrum (obtained from a simulation with noise) is shown in Fig. 5.17. The in-band SFDR is 49 dB.

⁴⁰ Noise in asynchronous digital delay cells has been analyzed in detail in Refs. [70]–[72]; the relation between the phase noise of a delay line and that of an oscillator made by connecting the delay line in feedback is derived in Ref. [73].

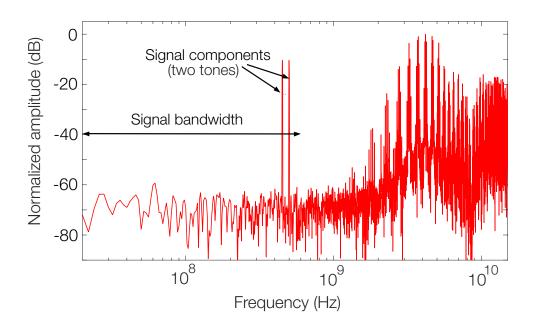


Fig. 5.17. ADC output spectrum for a two-tone input with two equal-amplitude tones at 450 MHz and 500 MHz obtained from a transient noise simulation.

The ADC performance is summarized and compared with other prior CT ADCs in Table 5.7. The power dissipation of the ADC is 176 μ W. This results in a $P/(2 \times f_{BW})$ of 0.15 pJ/sample and a Walden FOM of 2-4 fJ/conv-step. The improvement over CT ADCs in [3], [9], [10], [42] is respectively 16×, 940×, 50×, and 2.5×. Its placement in the energy and FOM plots of the Murmann survey [49] is shown in Fig. 5.18. That the proposed ADC beats the state-of-the-art CT and DT ADCs shows its promise.

Complete system simulations

To verify the operation of the complete ADC/DSP/DAC system, it was configured to implement a 16-tap FIR decimation filter: all MDAC currents (tap coefficients) were set to the full-scale value, I_{FS} (worst case in terms of DAC static power dissipation). The system output spectrum obtained from a transient noise simulation for a full-scale single-tone input at 100 MHz is shown in Fig. 5.19. The frequency response of the decimation filter is also shown for reference.

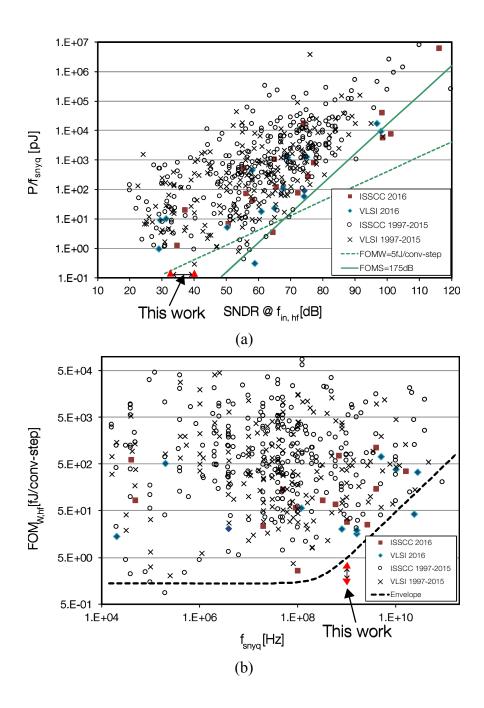


Fig. 5.18. Placement of the proposed ADC in the (a) energy plot and (b) the Walden FOM plot of the Murmann survey [49].

Comparing this spectrum with that of the ADC output without filtering in Fig. 5.14 (though for a noiseless simulation), we notice that the decimation filter attenuates the out-of-band modulation

	Schell [3]	Kurchuk [10]	Weltin-Wu [9]	Patil [42]	This Work (Simulations)
Technology	90 nm CMOS	65 nm CMOS	130 nm CMOS	28 nm UTBB FDSOI CMOS	28 nm UTBB FDSOI CMOS
Supply (V)	1	1.2	1	0.65	1.2
Input bandwidth, f_{BW}	10 kHz	2.4 GHz	20 kHz	40 MHz	600 MHz
Core area (mm²)	0.06	0.0036	0.36	0.0032	-
SNDR (dB)	58	20.3	47-54 ^a	32-42 ^a	33-40 ^a
Total power (µW)	50	2700	2-8	24	176
Figure of Merit (fJ/conv-step)	3769	66	200-850	3-10	2-4
$P/(2 \times f_{BW})$ (pJ)	2500	0.56	200	0.3	0.15
Antialiasing filter required?	No	No	No	No	Yes

^aSNDR varies with input frequency.

Table 5.7. Comparison of the PFM CT ADC with prior-published CT ADCs.

products by more than 20 dB. The output SNDR is 31.2 dB. The average system power dissipation is 4 mW: 0.2 mW in the ADC; 1.5 mW in the delay lines; and 2.3 mW in the DACs. In one input cycle, it goes from 1.9 mW at the trough of the input sinusoid to 6.1 mW at its crest. This is in the spirit of the PFM encoding scheme where the output pulse frequency goes from its lowest value at the input trough to its highest value at the input crest. The entire system being composed of event-driven blocks—delays, MDACs, and digital gates—, the total power dissipation scales in the same manner.

The system was next configured to implement a 16-tap FIR filter with a high-pass transfer function, which would be suitable for an equalizer application. A two-tone input with two tones at 50 MHz and 500 MHz was applied to it in a transient simulation. The spectra at the output of the ADC and that of the complete system along with the filter transfer function are shown in Fig. 5.20. The ADC output has an in-band SFDR of 50 dB. The system output spectrum shows a 15-dB

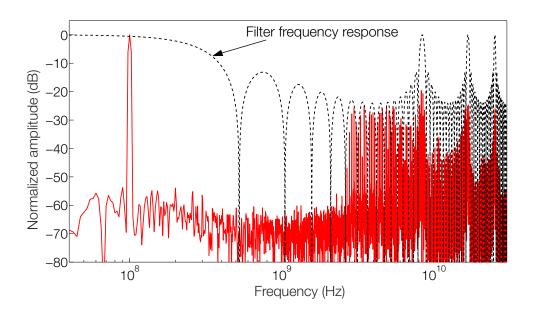


Fig. 5.19. System output spectrum, obtained from a transient noise simulation, for a full-scale single-tone input at 100 MHz when the DSP is configured to implement a 16-tap decimation filter; filter frequency response is also shown (dashed).

attenuation of the component at 50 MHz relative to that at 500 MHz, thereby demonstrating the high-pass nature of the filter transfer function. The average system power dissipation is 3.3 mW.

Comparison with other state-of-the-art processors

Table 5.8 compares the proposed CT ADC/DSP/DAC system with relevant state-of-theart CT DSP, DT DSP, and analog processors. FOM_{DSP} , expressed in (5.13) is used for comparison. Compared to the processors in Refs. [10], [52], [67], the proposed system achieves an FOM_{DSP} improvement of 5.3×, 9×, and 2.7× respectively, while being at par with the state-of-the-art DT DSP in Ref. [68], which will have an FOM_{DSP} of 5 fJ/sample (it is not shown in the table due to its rather high sampling rate (16 GS/s)). This demonstrates the potential of the proposed system. Of course, the numbers for the proposed system are only based on simulation results at the schematic

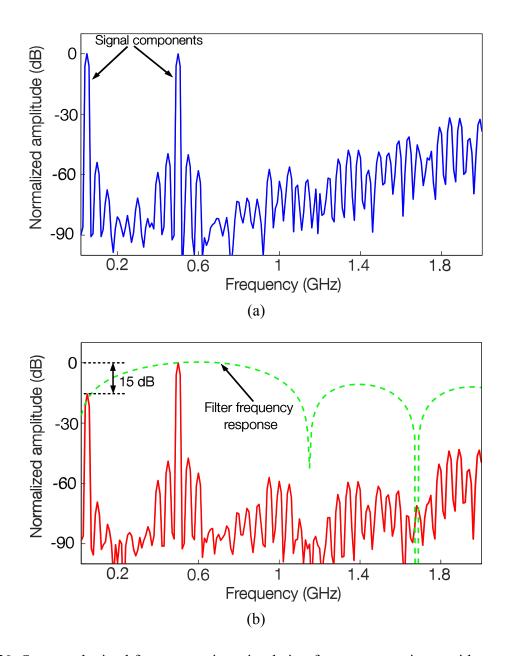


Fig. 5.20. Spectra obtained from a transient simulation for a two-tone input with two tones at 50 MHz and 500 MHz, and with the DSP configured to implement a 16-tap high-pass transfer function. (a) ADC output spectrum; and (b) system output spectrum, showing a 15 dB attenuation of the component at 50 MHz relative to the one at 500 MHz. Filter frequency response is also shown (dashed).

level. Therefore, they may not be taken to reflect the exact degree of improvement the proposed system can achieve. Instead, it should serve as an indication of the interesting possibility the

	Kurchuk [10]	Agarwal [67]	O'hAnnaidh [52]	This Work (Simulations)
Technology	65 nm CMOS	32 nm CMOS	45 nm CMOS	28 nm UTBB FDSOI CMOS
Supply (V)	1.2	1	1.1	1.2
Nature	CT mixed-domain DSP	DT DSP	Analog FIR	CT mixed-domain DSP
Input bandwidth, f_{BW}	2.4 GHz (0.8 GHz-3.2 GHz)	1.05 GHz	800 MHz	600 MHz
Average sample rate	0-45 GS/s	2.1 GS/s	3.2 GHz	4.2 GHz
Core area (mm ²)	0.073	0.004	0.15	-
SNDR (dB)	20.3	48	33 (SNR)	33-40
Total power, P (mW)	6.2 mW (average)	24	48	4
# of taps, N_{taps}	6	4	16	16
FOM _{DSP} (fJ/sample)	30	15	51	5.6
Antialiasing filter required?	No	Yes	Yes, but relaxed	Yes

Table 5.8. Comparison of the proposed CT ADC/DSP/DAC system with relevant state-of-the-art CT DSP, DT DSP, and analog processors.

proposed approach presents towards the goal of realizing an energy-efficient CT DSP.

5.4 Conclusions

In this chapter we described the design of a PFM-encoder-based CT ADC/DSP/DAC system. The resulting system is highly digital, with both the ADC and the DSP delay line implemented by similar asynchronous digital delay cells. This makes the system highly scalable with technology and amenable to a low-supply implementation. Simulations demonstrate that the overall system can achieve a very high energy efficiency that is significantly better than previous CT DSPs and at par with state-of-the-art DT DSPs.

Chapter 6

Conclusions

6.1 Thesis Contributions

The primary goal of this thesis was to develop techniques to improve the energy efficiency of CT DSPs so as to lower their energy-efficiency gap with state-of-the-art DT DSPs. We started by analyzing the design considerations of a CT ADC/DSP/DAC system in Chap. 1 (depicted in Fig. 1.9), where it became clear that the CT ADC, or the encoder, considerably impacts the system energy efficiency in a number of ways. For instance, it defines the number of tokens produced per second, NTPS, and the minimum intersample time, T_{GRAN} , which determine the CT DSP power dissipation (see (1.1)). We observed that once the CT encoder is fixed, so are the constraints of the following CT DSP, and there remain very few options in the designer's toolbox to improve the system energy efficiency. Consequently, the central premise around all the developments presented in this thesis is that for a CT DSP to attain its true potential, significant improvements are necessary in the preceding CT ADC. An appropriate CT encoder can drastically relax the CT DSP constraints, and hence, lower its power dissipation and improve its energy efficiency. Besides, if the ADC is energy efficient, it will also keep its contribution to system power dissipation small. Taken together, this can significantly lower the power dissipation of the composite CT ADC/DSP/DAC system and improve its overall energy efficiency. We thus set out to develop CT ADCs that can achieve this, and the pursuit led us to three principles.

In Chap. 2, an adaptive-resolution technique that achieves superior reconstruction with simple circuitry was proposed. For a given accuracy requirement, it was shown to drastically lower the *NTPS* for some signals, potentially lowering the power dissipation of the subsequent event-driven blocks (e.g. the CT DSP). In Chap. 3, we presented a 2-bit modulation scheme that allows an energy-efficient circuit implementation of the modulator and achieves spectral shaping of the quantization error. The design of the CT DSP that processes the ADC output with high energy efficiency, thanks to the latter's unique encoding, was also discussed. The resulting CT ADC/DSP system was shown to compare favorably with state-of-the-art processors. Finally, in Chap. 4, we considered CT A/D conversion using VCOs, which led to the CT ADC/DSP system composed primarily of asynchronous digital delays, presented in Chap. 5. It was shown that the energy efficiency achieved by the system rivals state-of-the-art DT DSPs. Besides, the highly-digital and technology-scaling-friendly nature of the resulting system makes it particularly attractive from the point of view of technology migration. Each principle has its advantages and limitations, and the right choice will depend on the constraints defined by the targeted application.

6.2 Suggestions for Future Work

Improvements to current work

Fig. 6.1 shows the models of a level-crossing-sampling (LCS) quantizer and the three proposed encoders: Derivative level-crossing sampling (DLCS) from Chap. 2 (Fig. 2.2(b))⁴¹; the error-shaping modulator from Chap. 3 (Fig. 3.6); and the PFM encoder from Chaps. 4-5 (Fig.

⁴¹ The quantizer is shown to have fixed resolution for simplicity; it could as well have been adaptive.

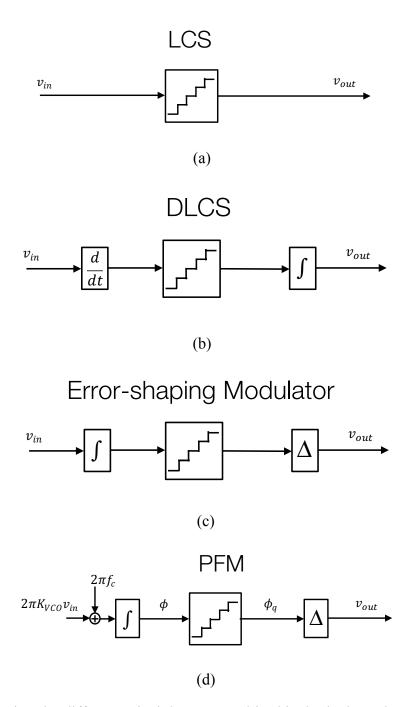


Fig. 6.1. Connecting the different principles proposed in this thesis through models of (a) an LCS quantizer; (b) the DLCS quantizer of Chap. 2; (c) the error-shaping modulator of Chap. 3; and (d) the PFM encoder of Chaps. 4-5.

4.15). We can see that, DLCS is obtained by preceding an LCS quantizer with a differentiator and

following it by an integrator. The model for the error-shaping modulator can be obtained by swapping the differentiator and integrator in the model for the DLCS quantizer⁴²: Now, the integrator precedes the quantizer while the (effective) differentiator follows it. Finally, the model of the PFM encoder is obtained by adding an offset of $2\pi f_c$ to the (scaled) input of the error-shaping modulator from Chap. 3.

It thus clear that, while the work in thesis has made an apparent push away from level-crossing sampling, it is inherently tied to it: The model of each encoder has an LCS quantizer at its heart. This point of view can inform future work in such encoders.

Fig. 6.1 depicts how different encoders with unique characteristics can be developed by placing different blocks around an LCS quantizer. An interesting possibility would be to place a general transfer function H(s) before the quantizer and its inverse, $H^1(s)$, following it as shown in Fig. 6.2. For a given set of signal characteristics, H(s) can then be found to optimize a certain objective function (for example, minimizing *NTPS*). A multitude of interesting possibilities may arise.

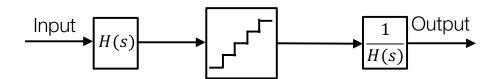


Fig. 6.2. A general CT encoder can be developed by preceding an LCS quantizer with a general transfer function H(s) and by following it with the inverse of the transfer function.

•

 $^{^{42}}$ Recall from Sec. 3.3.2 that the Δ block has properties similar to a differentiator.

On applications

The error-shaping CT ADC/DSP/DAC system presented in Chap. 3 found an interesting application in wake-up radios—it only made sense that an event-driven radio have an event-driven processor in it. The application defined a challenging set of specifications and resulted in the development of a number of interesting architectures throughout the CT ADC/DSP/DAC system.

Inspired by this, one can set out to find other such suitable applications in the hope of creating a new generation of system-/block-level architectures. An obvious way of doing this is to observe discrete-time digital systems and ask the question: What if we removed the clocked sampler from the system? For instance, an LCS CT ADC can be thought of as the system one gets by removing the sample and hold block from a DT ADC [1]; it was shown in Sec. 4.3.2 (Figs. 4.15-4.16) that the PFM encoder is obtained by removing the sampler from a DT VCO ADC⁴³. One obvious case where this can be considered is a digital phase locked loop (D-PLL). Analog PLLs have an analog loop filter, which filters the output of the phase-and-frequency (PFD). This filter typically occupies a large chip area and needs to be off chip. A D-PLL uses a DT digital loop filter in lieu of it. Since this loop filter requires a DT digital input, a D-PLL has a time-to-digital converter (TDC) that quantizes the output of the PFD with very fine time resolution; the latter's digital output is then fed to the DT loop filter.

It can be observed that the output of a PFD in a PLL is inherently CT digital: it is discrete in amplitude (binary) and the transitions in the output are not synchronized to a clock but vary in CT as per the phase error. Therefore, in principle, this PFD output can be processed *directly* by a

⁴³ In fact, that is exactly how the idea was first conceived.

CT DSP, without the need for time quantization⁴⁴ and without having to face the ensuing spectral mess that results due to aliasing in DT systems. The resulting PLL will thus have a "time-based" loop filter that operates in continuous time. Of course, this will require a research effort to find the right CT DSP architecture that can deliver the desired transfer function while keeping power dissipation, chip area, and noise low.

Concluding remarks

In concluding this thesis, we make the following observation: A CT DSP can process *both* CT *and* DT digital signals (see Ref. [16]), while a DT DSP can process *only* DT digital signals. Therefore, for a given application space, one can envision a single CT DSP, which can handle both CT and DT digital signals, as against a DT DSP, which restricts the input to being DT digital ⁴⁵— *provided* the design costs ⁴⁶ of the two DSPs are comparable and the desired functionality is delivered by both. At this stage in its development, however, CT DSP does not match up with DT DSP in terms of functionality and robustness, while it has made significant strides towards improving its energy efficiency for some functions (e.g. transversal filters) as evidenced at points in this thesis. This motivates more research in CT DSP to see if such a vision can be a reality. Even if it turns out to not be so, CT DSP can be used to complement DT DSP in specific cases, thus making the research worthwhile. Irrespective of the outcome, this author believes that the very

⁴⁴ DSP in continuous time can be thought of as DSP in discrete time with an infinite sampling frequency.

⁴⁵ Note that the CT digital output of the CT DSP can be easily converted into DT digital form if necessary; i.e. CT DSP can complement DT DSP.

⁴⁶ These design costs include design time, performance metrics (e.g. energy efficiency), robustness etc.

pursuit of this (initially) baffling signal processing paradigm is bound to lead one to a goldmine of exciting research ideas. The work presented in this thesis only scratches its surface. The next step then is to dig deeper.

Bibliography

- [1] Y. Tsividis, "Event-driven data acquisition and digital signal processing a tutorial," *IEEE Trans. Circuits Syst. II Express Briefs*, vol. 57, no. 8, pp. 577–581, 2010.
- [2] Y. Tsividis, "Continuous-time digital signal processing," *Electron. Lett.*, vol. 39, no. 21, pp. 1551–1552, 2003.
- [3] B. Schell and Y. Tsividis, "A continuous-time ADC/DSP/DAC system with no clock and with activity-dependent power dissipation," *IEEE J. Solid-State Circuits*, vol. 43, no. 11, pp. 2472–2481, 2008.
- [4] J. W. Mark and T. D. Todd, "A nonuniform sampling approach to data compression," *IEEE Trans. Commun.*, vol. 29, no. 1, pp. 24–32, 1981.
- [5] E. Allier, G. Sicard, L. Fesquet, and M. Renaudin, "Asynchronous level crossing analog to digital converters," *Meas. J. Int. Meas. Confed.*, vol. 37, no. 4, pp. 296–309, 2005.
- [6] K. Kozmin, J. Johansson, and J. Delsing, "Level-crossing ADC performance evaluation toward ultrasound application," *IEEE Trans. Circuits Syst. I Regul. Pap.*, vol. 56, no. 8, pp. 1708–1719, 2009.
- [7] Y. Li, D. Zhao, M. N. Van Dongen, and W. A. Serdijn, "A 0.5V signal-specific continuous-time level-crossing ADC with charge sharing," 2011 IEEE Biomed. Circuits Syst. Conf. BioCAS 2011, pp. 381–384, 2011.
- [8] R. L. Grimaldi, S. Rodriguez, and A. Rusu, "A 10-bit 5kHz level-crossing ADC," 2011 20th Eur. Conf. Circuit Theory Des. ECCTD 2011, pp. 564–567, 2011.
- [9] C. Weltin-Wu and Y. Tsividis, "An event-driven clockless level-crossing ADC with signal-dependent adaptive resolution," *IEEE J. Solid-State Circuits*, vol. 48, no. 9, pp. 2180–2190, 2013.
- [10] M. Kurchuk, C. Weltin-Wu, D. Morche, and Y. Tsividis, "Event-driven GHz-range continuous-time digital signal processor with activity-dependent power dissipation," *IEEE J. Solid-State Circuits*, vol. 47, no. 9, pp. 2164–2173, 2012.
- [11] Y. Hong, Z. Xie, and Y. Lian, "Wireless wearable ECG sensor design based on level-crossing sampling and linear interpolation," in *Proceedings of IEEE International Symposium on Circuits and Systems*, 2013, pp. 1300–1303.
- [12] Y. Li, A. L. Mansano, Y. Yuan, D. Zhao, and W. A. Serdijn, "An ECG recording front-end with continuous-time level-crossing sampling," *IEEE Trans. Biomed. Circuits Syst.*, vol. 8, no. 5, pp. 626–635, 2014.
- [13] T. Wang, D. Wang, P. J. Hurst, B. C. Levy, and S. H. Lewis, "A level-crossing analog-to-

- digital converter with triangular dither," *IEEE Trans. Circuits Syst. I Regul. Pap.*, vol. 56, no. 9, pp. 2089–2099, 2009.
- [14] A. Reeves, "The Past Present and Future of PCM," *IEEE Spectr.*, no. May, pp. 58–63, 1965.
- [15] W. R. Bennett, "Spectra of quantized signals," *Bell Syst. Tech. J.*, vol. 27, no. July, pp. 446–471, 1948.
- [16] C. Vezyrtzis, W. Jiang, S. M. Nowick, and Y. Tsividis, "A flexible, event-driven digital filter with frequency response independent of input sample rate," *IEEE J. Solid-State Circuits*, vol. 49, no. 10, pp. 2292–2304, 2014.
- [17] Y. Tsividis, "Mixed-domain systems and signal processing based on input decomposition," *IEEE Trans. Circuits Syst. I Regul. Pap.*, vol. 53, no. 10, pp. 2145–2156, 2006.
- [18] Z. Song and D. V. Sarwate, "The frequency spectrum of pulse width modulated signals," *Signal Processing*, vol. 83, no. 10, pp. 2227–2258, 2003.
- [19] E. J. Bayly, "Spectral analysis of pulse frequency modulation in the nervous systems," *IEEE Trans. Biomed. Eng.*, vol. 15, no. 4, pp. 257–265, 1968.
- [20] R. Steele, *Delta Modulation Systems*. New York: Wiley, 1975.
- [21] C. J. Kikkert and D. J. Miller, "Asynchronous delta sigma modulation," *Proc. IREE*, vol. 36, no. April 1975, pp. 83–88, 1975.
- [22] C. Vezyrtzis and Y. Tsividis, "Processing of signals using level-crossing sampling," *Proc. IEEE Int. Symp. Circuits Syst.*, no. 1, pp. 2293–2296, 2009.
- [23] M. Kurchuk and Y. Tsividis, "Signal-dependent variable-resolution clockless A/D conversion with application to continuous-time digital signal processing," *IEEE Trans. Circuits Syst. I Regul. Pap.*, vol. 57, no. 5, pp. 982–991, 2010.
- [24] M. Kurchuk and Y. Tsividis, "Energy-efficient asynchronous delay element with wide controllability," in *IEEE International Symposium on Circuits and Systems*, 2010, pp. 3837–3840.
- [25] B. Schell and Y. Tsividis, "A low power tunable delay element suitable for asynchronous delays of burst information," *IEEE J. Solid-State Circuits*, vol. 43, no. 5, pp. 1227–1234, 2008.
- [26] L. M. Feeney and M. Nilsson, "Investigating the energy consumption of a wireless network interface in an ad hoc networking environment," in *Proceedings IEEE INFOCOM 2001*. Conference on Computer Communications. Twentieth Annual Joint Conference of the IEEE Computer and Communications Society, 2001, vol. 3, pp. 1548–1557.
- [27] F. A. Marvasti, Nonuniform sampling: theory and practice. New York: Kluwer, 2001.

- [28] W. Tang, C. Huang, D. Kim, B. Martini, and E. Culurciello, "4-Channel asynchronous biopotential recording system," in *2010 IEEE International Symposium on Circuits and Systems*, 2010, pp. 953–956.
- [29] R. Agarwal, M. Trakimas, and S. Sonkusale, "Adaptive asynchronous analog to digital conversion for compressed biomedical sensing," in *2009 IEEE Biomedical Circuits and Systems Conference*, 2009, pp. 69–72.
- [30] Y. Li, D. Zhao, and W. A. Serdijn, "A sub-microwatt asynchronous level-crossing ADC for biomedical applications," *IEEE Trans. Biomed. Circuits Syst.*, vol. 7, no. 2, pp. 149–157, 2013.
- [31] P. Martinez-Nuevo, S. Patil, and Y. Tsividis, "Derivative level-crossing sampling," *IEEE Trans. Circuits Syst. II Express Briefs*, vol. 62, no. 1, pp. 11–15, 2015.
- [32] A. Papoulis, "Generalized sampling expansion," *IEEE Trans. Circuits Syst.*, vol. 24, no. 11, pp. 652–654, 1977.
- [33] M. M. Milosavljević and M. R. Dostanić, "On generalized stable nonuniform sampling expansions involving derivatives," *IEEE Trans. Inf. Theory*, vol. 43, no. 5, pp. 1714–1716, 1997.
- [34] M. Trakimas and S. Sonkusale, "A 0.8 V asynchronous ADC for energy constrained sensing applications," in 2008 Custom Integrated Circuits Conference, 2008, pp. 173–176.
- [35] N. Sayiner, "A level-crossing sampling scheme for A/D conversion," *IEEE Trans. Circuits Syst. II Analog Digit. Signal Process.*, vol. 43, no. 4, pp. 335–339, 1996.
- [36] R. P. Boas, *Entire Functions*. New York: Academic Press, 1954.
- [37] B. Smith, "Instantaneous companding of quantized signals," *Bell Syst. Tech. J.*, vol. 36, no. 3, pp. 653–709, 1957.
- [38] Y. Suh, "Send-on-delta sensor data transmission with a linear predictor," *Sensors*, pp. 537–547, 2007.
- [39] B. Schell and Y. Tsividis, "Analysis and simulation of continuous-time digital signal processors," *Signal Processing*, vol. 89, no. 2009, pp. 2013–2026, 2008.
- [40] B. Murmann, "A/D converter trends: Power dissipation, scaling and digitally assisted architectures," in 2008 IEEE Custom Integrated Circuits Conference, 2008, pp. 105–112.
- [41] N. M. Pletcher, S. Gambini, and J. Rabaey, "A 52 μW wake-up receiver with -72 dBm sensitivity using an uncertain-IF architecture," *IEEE J. Solid-State Circuits*, vol. 44, no. 1, pp. 269–280, 2009.
- [42] S. Patil, A. Ratiu, D. Morche, and Y. Tsividis, "A 3–10 fJ/conv-step error-shaping alias-free continuous-time ADC," *IEEE J. Solid-State Circuits*, vol. 51, no. 4, pp. 908–918, 2016.

- [43] C. Weltin-wu, "Design and optimization of low-power level-crossing ADCs," Ph.D. dissertation, Columbia University, 2012.
- [44] M. Z. Straayer and M. H. Perrott, "Oversampled ADC using VCO-based quantizers," *Multi-Mode/Multi-Band RF Transceivers Wirel. Commun. Adv. Tech. Archit. Trends*, pp. 247–277, 2010.
- [45] J. M. Vandeursen and J. A. Peperstraete, "Analog-to-digital conversion based on a voltage-to-frequency converter," *IEEE Trans. Ind. Electron. Control Instrum.*, vol. IECI-26, no. 3, pp. 161–166, 1979.
- [46] J. G. Harris, C. Principe, J. C. Sanchez, D. Chen, and C. She, "Pulse-based signal compression for implanted neural recording systems," in 2008 IEEE International Symposium on Circuits and Systems, 2008, pp. 344–347.
- [47] Chen Du, J. G. Harris, and J. C. Principe, "Device and methods for biphasic pulse signal coding," US 8139654 B2, 2008.
- [48] D. Jacquet, F. Hasbani, P. Flatresse, R. Wilson, F. Arnaud, G. Cesana, T. Di Gilio, C. Lecocq, T. Roy, A. Chhabra, C. Grover, O. Minez, J. Uginet, G. Durieu, C. Adobati, D. Casalotto, F. Nyer, P. Menut, A. Cathelin, I. Vongsavady, and P. Magarshack, "A 3 GHz dual core processor ARM cortex TM -A9 in 28 nm UTBB FD-SOI CMOS with ultra-wide voltage range and energy efficiency optimization," *IEEE J. Solid-State Circuits*, vol. 49, no. 4, pp. 812–826, 2014.
- [49] B. Murmann, "ADC Performance Survey 1997-2014," 2015. [Online]. Available: http://web.stanford.edu/~murmann/adcsurvey.html.
- [50] A. Ratiu, "Continuous time signal processing for wake-up radios," Ph.D. dissertation, Universite de Lyon, 2015.
- [51] D. Bruckmann, T. Feldengut, B. Hosticka, R. Kokozinski, K. Konrad, and N. Tavangaran, "Optimization and implementation of continuous time DSP-systems by using granularity reduction," in *2011 IEEE International Symposium of Circuits and Systems*, 2011, pp. 410–413.
- [52] E. O'Hannaidh, E. Rouat, S. Verhaeren, S. Le Tual, and C. Garnier, "A 3.2GHz-sample-rate 800MHz bandwidth highly reconfigurable analog FIR filter in 45nm CMOS," in *Digest of Technical Papers, IEEE International Solid-State Circuits Conference*, 2010, pp. 90–91.
- [53] W. H. Ma, J. C. Kao, V. S. Sathe, and M. C. Papaefthymiou, "187 MHz subthreshold-supply charge-recovery FIR," *IEEE J. Solid-State Circuits*, vol. 45, no. 4, pp. 793–803, 2010.
- [54] B. Drost, M. Talegaonkar, and P. K. Hanumolu, "Analog filter design using ring oscillator integrators," *IEEE J. Solid-State Circuits*, vol. 47, no. 12, pp. 3120–3129, 2012.
- [55] C. W. Hsu and P. R. Kinget, "A 40MHz 4th-order active-UGB-RC filter using VCO-based amplifiers with zero compensation," in *ESSCIRC 2014 40th European Solid State Circuits*

- Conference, 2014, pp. 359–362.
- [56] M. Hovin, A. Olsen, T. S. Lande, and C. Toumazou, "Delta-sigma modulators using frequency-modulated intermediate values," *IEEE J. Solid-State Circuits*, vol. 32, no. 1, pp. 13–22, 1997.
- [57] G. Taylor and I. Galton, "A mostly-digital variable-rate continuous-time delta-sigma modulator ADC," *IEEE J. Solid-State Circuits*, vol. 45, no. 12, pp. 2634–2646, 2010.
- [58] S. Patil and Y. Tsividis, "Digital processing of signals produced by voltage-controlled-oscillator-based continuous-time ADCs," in *IEEE International Symposium on Circuits and Systems*, 2016, pp. 1046–1049.
- [59] E. Roza, "Analog-to-digital conversion via duty-cycle modulation," *IEEE Trans. Circuits Syst. II Analog Digit. Signal Process.*, vol. 44, no. 11, pp. 907–914, 1997.
- [60] N. Tavangaran, D. Brückmann, R. Kokozinski, and K. Konrad, "Continuous time digital systems with asynchronous sigma delta modulation," in *Proceedings of the 20th European Signal Processing Conference (EUSIPCO)*, 2012, pp. 225–229.
- [61] A. E. Ross, "Theoretical study of pulse-frequency modulation," *Proc. IRE*, vol. 37, no. 11, pp. 1277–1286, 1949.
- [62] R. W. Rochelle, "Pulse-frequency modulation," *IRE Trans. Sp. Electron. Telem.*, vol. SET-8, no. 2, pp. 107–111, 1962.
- [63] E. H. Armstrong, "A method of reducing disturbances in radio signaling by a system of frequency modulation," *Proc. IRE*, vol. 24, no. 5, pp. 689–740, 1936.
- [64] L. Hernandez and E. Gutierrez, "Analytical evaluation of VCO-ADC quantization noise spectrum using pulse frequency modulation," *IEEE Signal Process. Lett.*, vol. 22, no. 2, pp. 249–253, 2014.
- [65] J. F. Bulzacchelli, "Equalization for electrical links: Current design techniques and future directions," *IEEE Solid-State Circuits Mag.*, vol. 7, no. 4, pp. 23–31, 2015.
- [66] Y. M. Tousi and E. Afshari, "A miniature 2 mW 4 bit 1.2 GS/s delay-line-based ADC in 65 nm CMOS," *IEEE J. Solid-State Circuits*, vol. 46, no. 10, pp. 2312–2325, 2011.
- [67] A. Agarwal, S. K. Mathew, S. K. Hsu, M. A. Anders, H. Kaul, F. Sheikh, R. Ramanarayanan, S. Srinivasan, R. Krishnamurthy, and S. Borkar, "A 320mV-to-1.2V on-die fine-grained reconfigurable fabric for DSP/media accelerators in 32nm CMOS," in *Digest of Technical Papers, IEEE International Solid-State Circuits Conference*, 2010, pp. 328–329.
- [68] T. Toifl, P. Buchmann, T. Beukema, M. Beakes, M. Brändli, P. A. Francese, C. Menolfi, M. Kossel, L. Kull, and T. Morf, "A 3.5pJ/bit 8-tap-feed-forward 8-tap-decision feedback digital equalizer for 16Gb/s I/Os," in ESSCIRC 2014 40th European Solid State Circuits

- Conference (ESSCIRC), 2014, pp. 455–458.
- [69] F. Akopyan, R. Manohar, and A. B. Apsel, "A level-crossing flash asynchronous analog-to-digital converter," in *Proceedings of International Symposium on Asynchronous Circuits and Systems*, 2006, vol. 2006, pp. 12–22.
- [70] C. Vezyrtzis, "Continuous-time and companding digital signal processors using adaptivity and asynchronous techniques," Ph.D. dissertation, Columbia University, 2013.
- [71] M. Kurchuk, "Signal encoding and digital signal processing in continuous time," Ph.D. dissertation, Columbia University, 2011.
- [72] M. A. Ghanad, C. Dehollain, and M. M. Green, "Noise analysis for time-domain circuits," in *Proceedings of IEEE International Symposium on Circuits and Systems*, 2015, pp. 149–152.
- [73] A. Homayoun and B. Razavi, "Relation between delay line phase noise and ring oscillator phase noise," *IEEE J. Solid-State Circuits*, vol. 49, no. 2, pp. 384–391, 2014.
- [74] K. Yoshioka, A. Shikata, R. Sekimoto, T. Kuroda, and H. Ishikuro, "A 0.0058mm2 7.0 ENOB 24MS/s 17fJ/conv. threshold configuring SAR ADC with source voltage shifting and interpolation technique," in *2013 Symposium on VLSI Circuits (VLSI)*, 2013, pp. C266–C267.
- [75] J.-H. Tsai, Y.-J. Chen, M.-H. Shen, and P.-C. Huang, "A 1-V, 8b, 40MS/s, 113μW charge-recycling SAR ADC with a 14μW asynchronous controller," in *Symposium on VLSI Circuits Digest of Technical Papers*, 2011, pp. 264–265.
- [76] G. Van Der Plas and B. Verbruggen, "A 150MS/S 133µW 7b ADC in 90nm digital CMOS using a comparator-based asynchronous binary-search sub-ADC," in *Digest of Technical Papers, IEEE International Solid-State Circuits Conference*, 2008, pp. 242–244.
- [77] L. Brooks and H. S. Lee, "A zero-crossing-based 8-bit 200 MS/s pipelined ADC," *IEEE J. Solid-State Circuits*, vol. 42, no. 12, pp. 2677–2687, 2007.