CONTINUOUS COMPENSATION OF
BINARY-WEIGHTED DAC NONLINEARITIES IN
BANDPASS DELTA-SIGMA MODULATORS

By
Ghyslain Gagnon

A thesis submitted to the Faculty of Graduate Studies and Research
in partial fulfillment of the requirements for the degree of
DOCTOR OF PHILOSOPHY

Ottawa-Carleton Institute for Electrical and Computer Engineering
Department of Electronics
Carleton University

Ottawa, Ontario, Canada
© Ghyslain Gagnon, 2008

Library and
Archives Canada

Published Heritage
Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et
Archives Canada

Direction du
Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

ISBN: 978-0-494-47463-1

Our file Notre référence

ISBN: 978-0-494-47463-1

NOTICE:

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protègent cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

Abstract

This thesis introduces and discusses the implementation of a novel calibration technique to compensate for DAC element mismatches in bandpass multibit delta-sigma ($\Delta\Sigma$) modulators. The compensation is accomplished in the digital domain using the output bitstream of the $\Delta\Sigma$ modulator; only a minor modification to the analog portion of the $\Delta\Sigma$ modulator loop is needed. The technique is compatible with binary weighted element DACs and the storage requirements for the calibrated coefficients increases only linearly with the number of quantizer bits. The calibration is performed without breaking the loop, which allows continuous tracking of environmental drifts. Furthermore, the proposed technique is shown to be also applicable to lowpass $\Delta\Sigma$ modulators with minor modifications to the system.

Simulation results show a peak signal to noise ratio (SNR) of 73.9 dB after calibration for a DAC with 1% mismatches, a sinusoid input signal near 1/4 of the sampling frequency and an oversampling ratio of 14. This 73.9 dB peak SNR result represents a 26 dB improvement over the non-calibrated case and it is only 0.4 dB lower than an ideal-DAC case.

The proposed compensation technique is applied to a fourth-order $\Delta\Sigma$ modulator fabricated in 0.13 μm CMOS. The peak SNR of the tested chip, without compensation, is measured at 57.3 dB. With the implemented compensation technique, the peak SNR increases to 64.5 dB. The peak SNR after calibration drops by 1.8 dB when severe mismatch is forced on the DAC elements while the same mismatch causes the SNR to drop by 10.1 dB if no calibration is used.

Acknowledgments

I would like to begin by thanking my supervisor, Dr. Leonard MacEachern. Leonard, five years ago, although you had never heard my name (and then had a hard time figuring out how to pronounce it), although you were as busy as professors can be, you took the time to meet with me and then allowed me to join your team. Thank you for giving me that opportunity, and thank you for your precious help and wise guidance throughout this adventure.

Many thanks to Ecole de Technologie Supérieure, Université du Québec, for believing in me and providing financial support which allowed me to complete those doctoral studies at Carleton University. I now realize how important it was to force me out of my comfort zone, go get shaken up a little bit in a new environment, with new people, in a different language.

It was a real pleasure to meet such interesting and clever colleagues. Particularly, Mark, Karim, Sinisa, Till, Greg, Lado and Igor, I really enjoyed spending this time with you.

An unusual thank goes to Full Throttle® energy drinks who took the lead when coffee wasn't doing it anymore, a few days before chip tapeout.

Finally, thank you Marianne for being who you are, for being there in my life. I love you. I promise, it's over now. No more studies.

Table of Contents

Abstract	ii
Acknowledgments	iii
Table of Contents	iv
List of Tables	viii
List of Figures	ix
List of Abbreviations and Symbols	xiii
1 Introduction	1
1.1 Motivation	1
1.2 Document Outline	2
1.3 Publications	3
2 $\Delta\Sigma$ Modulation	5
2.1 Important Definitions	5
2.2 Basic Concepts	9
2.3 High-Order Modulation and Stability	12
2.4 Loop Filter Architectures	13
2.5 Quantizer	19
2.6 Summary	21
3 State-of-the-Art in Compensation of DAC Nonlinearities	23
3.1 Dynamic Element Matching (DEM)	23
3.1.1 Randomization	26
3.1.2 Data Weighted Averaging (DWA)	26
3.1.3 Higher-Order Mismatch Shaping	28
3.1.4 Bandpass DEM	31
3.2 Digital Compensation	32

3.3	Other Techniques	38
3.4	Summary	39
4	Proposed Solution for Compensation of DAC Nonlinearities	40
4.1	The importance of binary-weighted DAC compensation	40
4.2	System Description	42
4.2.1	Nonlinear Transfer Function	46
4.2.2	Lowpass Decimation Filter and Power Estimator	46
4.2.3	Minimization Algorithm	48
4.2.4	DC Offset and Noise	49
4.3	Simulations	50
4.3.1	Simulation Parameters	50
4.3.2	Simulation Results	51
4.4	Compensation of Lowpass $\Delta\Sigma$ Modulators	51
4.5	Summary	53
5	Minimization Algorithm	55
5.1	Effect of Each Coefficient on the Noise Power in the Observation Band	55
5.2	The Probability of Error	59
5.3	Description of the Algorithm	62
5.3.1	Step One	66
5.3.2	Step Two	67
5.3.3	Step Three	67
5.3.4	Step Four	68
5.3.5	Additional Precaution	68
5.4	Simulation Results	69
5.4.1	Calibration Time	69
5.5	Summary	71
6	System-Level Design	74
6.1	SNR Target	74
6.2	Number of Quantizer Levels	76
6.3	Center Frequency of Signal and Observation Bands	77
6.4	NTF Order and OSR	77
6.5	Exact Signal and Noise Transfer Functions	78
6.6	CRFB Gains	82
6.7	Simulation	84
6.8	Parameters for the Digital Calibration Chain	85
6.8.1	The Probability of Error	85
6.8.2	Coefficient Resolution	88

6.8.3	Lowpass Filter Cutoff Frequency	90
6.8.4	Lowpass filter order	95
6.8.5	D , the Lowpass filter decimation factor	95
6.8.6	K , the number of samples for estimation	95
6.8.7	Adaptive Parameters	97
6.9	Summary	98
7	IC and PCB Design	100
7.1	Choice of Technology	101
7.2	Continuous or Discrete-Time	101
7.3	Differential Signaling	102
7.4	Sampling Frequency	102
7.5	Thermal Noise and Capacitor Sizing	103
7.6	Circuit Overview	104
7.6.1	Fully Differential Amplifier (<code>diffamp</code>)	105
7.6.2	Clock buffers (<code>clkbuf</code>)	107
7.6.3	Bias circuit (<code>bias</code>)	107
7.6.4	Sample and hold (<code>sah_3phases</code>)	108
7.6.5	Integrator (<code>integrator_no_delayx2</code>)	109
7.6.6	Resonators (<code>resonator1x2</code> and <code>resonator2x2</code>)	110
7.6.7	Summing Node (<code>sum2x2</code>)	112
7.6.8	Quantizer (<code>adc_5b_no_ff</code>)	112
7.6.9	DAC (<code>dac_5b</code>)	112
7.7	Design Iterations and Noise Problems	115
7.8	Simulation Results for the Complete System	116
7.9	IC Layout	118
7.10	Chip Bonding and Packaging	119
7.11	PCB Design	119
7.12	Summary	122
8	Experimental Results	125
8.1	Test Setup and Equipment	125
8.2	Bias Optimization and Power Consumption	127
8.3	Input Signal	128
8.4	Waveforms	131
8.5	Modifications to the Digital Calibration Chain	133
8.6	Results	135
8.6.1	The Implemented NTF	135
8.6.2	The Effect of Calibration on the Fabricated Chip	137
8.7	Summary	145

9 Summary of Thesis	147
9.1 Objective Evaluation of the Novel DAC Mismatch Compensation Technique	150
9.2 Contributions	151
9.3 Future Work	152
A Description of Matlab Functions Used in the Thesis	156
B Matlab Code for the Minimization Algorithm	159
C Paper Presented at NEWCAS 2007	163
D Deriving the Probability of Error	168
E Schematics	171
Bibliography	191

List of Tables

6.1	Summary of $\Delta\Sigma$ modulator System-Level Parameters.	98
7.1	Description of IC pins.	124
8.1	Summary of experimental results.	146

List of Figures

2.1	Hypothetical modulator with different DR and SNR_{max} values. . .	8
2.2	Linear mathematical model of a $\Delta\Sigma$ modulator.	10
2.3	Conceptual output spectra for lowpass and bandpass $\Delta\Sigma$ modulators. . .	11
2.4	Empirical peak SNR limit for 1-bit quantizers.	13
2.5	Empirical peak SNR limit for 2-bit quantizers.	14
2.6	Empirical peak SNR limit for 3-bit quantizers.	14
2.7	NTF with all zeros at DC as well as NTF with optimized zeros, for a fifth-order modulator with $OSR = 32$. The integrated noise power in the signal band is 18 dB lower for the optimized-zeros curve.	16
2.8	CIFB structure.	17
2.9	CRFB structure.	18
2.10	$\Delta\Sigma$ modulator employing the error feedback structure.	19
2.11	SNDR versus oversampling ratio for a 4 th -order 4-bit $\Delta\Sigma$ modula- tor to show the effect of DAC nonlinearity.	22
3.1	Static response of a 2-bit DAC composed of 3 mismatched unit- element cells.	25
3.2	General block diagram of DEM.	25
3.3	Simulation results comparing DWA and Bi-DWA.	28
3.4	General block diagram of high-order DEM.	29
3.5	$\Delta\Sigma$ modulator employing the error feedback structure.	29
3.6	Block diagram of bandpass DEM.	32
3.7	Block diagram of a modified tree-structured DEM.	33

3.8	Simplified block diagram of a compensated nonlinear system. . . .	34
3.9	Block diagram of an analog self-calibrated $\Delta\Sigma$ modulator.	35
3.10	Selection multiplexer to calibrate one DAC element at a time while the other elements perform normal conversion.	37
3.11	Output spectrum showing the calibration tone at $f_s/2$ and the notch in the NTF.	38
4.1	Simplified block diagram of a compensated nonlinear system. . . .	43
4.2	Modulator output spectrum with NTF zero at DC to show the noise floor increase due to DAC nonlinearity	44
4.3	Bode diagram of NTF with and without the observation band (at DC).	44
4.4	Detailed model of the digitally compensated $\Delta\Sigma$ modulator. . . .	45
4.5	Lowpass filter cell.	47
4.6	Power spectrum at modulator output before (a) and after (b) the calibration process.	52
4.7	Simulation results showing SNR vs. input amplitude.	53
4.8	Power spectrum at modulator output before (a) and after (b) the calibration process using a two-tone input.	54
5.1	Effect of each coefficient on the noise in the observation band and SNR.	57
5.2	Effect of C_1 and C_5 on the frequency spectrum.	58
5.3	Noise in the observation band while varying coefficient C_5	60
5.4	Probability of error when setting coefficients.	63
5.5	Main steps of the proposed minimization algorithm.	65
5.6	Noise power in the observation band and SNR throughout calibra- tion process.	70
5.7	Calibration time versus $\Delta\Sigma$ modulator sampling frequency.	72
6.1	Block diagram of the implemented $\Delta\Sigma$ modulator.	79

6.2	NTF and STF magnitude response with two different values for the fifth pole p_5	81
6.3	Noise and signal transfer functions of the $\Delta\Sigma$ modulator to be implemented.	83
6.4	Simulink model of the $\Delta\Sigma$ modulator.	85
6.5	Frequency spectrum of the Simulink model.	86
6.6	Frequency spectrum of the Simulink model showing the STF. . . .	86
6.7	Simulation results showing the SNR vs input amplitude for the Simulink model.	87
6.8	SNR vs coefficient offsets.	89
6.9	Noise in the observation band while varying C_1	92
6.10	Probability of error for C_1 and C_5 for different values of A	94
7.1	Block diagram of the circuit-level schematics.	105
7.2	Schematic of the fully-differential amplifier.	106
7.3	Schematic of the sample and hold.	109
7.4	Transient simulation results for the sample and hold.	110
7.5	Schematic of the integrator.	111
7.6	Transient simulation results for the integrator.	111
7.7	Schematic of the DAC.	113
7.8	Transient simulation results for the DAC.	114
7.9	Intentional mismatch.	115
7.10	Circuit-level simulation results.	117
7.11	Layout of the IC with overlay showing the major blocks.	119
7.12	Block diagram of the PCB and external components.	120
7.13	Top and bottom layers of the PCB (actual size).	122
8.1	Picture of the tested board.	126
8.2	Oscilloscope screen capture showing both phases of the differential input signal.	129
8.3	Spectrum analyzer screen capture showing the differential input signal.	130

8.4	Oscilloscope screen capture showing the $\Delta\Sigma$ modulator DAC output.	132
8.5	Oscilloscope screen capture showing the $\Delta\Sigma$ modulator first integrator output (differential).	132
8.6	$\Delta\Sigma$ modulator time-domain output samples.	133
8.7	Power spectrum at modulator output.	136
8.8	Power spectrum at modulator output after modifications.	137
8.9	Power spectrum at modulator output before (a) and after (b) the calibration process.	139
8.10	Power spectrum at modulator output with intentional 1% DAC mismatch.	141
8.11	SNR vs input amplitude with and without intentional mismatch. .	142
8.12	Peak SNR for various OSRs.	145
9.1	Possible implementation of the digital calibration chain using parallel computing for faster calibration.	155
D.1	Probability of error of two normally distributed variables.	169

List of Abbreviations and Symbols

Δ_q	Quantization step
δ_i	The offset that is applied to coefficients during calibration
$\Delta\Sigma$	Delta-sigma (modulator)
ADC	Analog-to-digital converter
$C_{1..n}$	Set of coefficients stored in memory for the nonlinear transfer function block
CIFB	Cascade of integrators with distributed feedback and distributed coupling
CMC	Canadian microelectronics corporation
CMFB	Common-mode feedback
CMOS	Complementary Metal Oxide Semiconductor
CRFB	Cascade of resonators with distributed feedback and input
DAC	Digital-to-analog converter
dBFS	dB below full-scale
DC	Direct current. Also refers to a frequency of 0 Hz in digital systems
DEM	Dynamic element matching
DWA	Data weighted averaging
ESL	Element selection logic
f_b	Input signal bandwidth
f_c	Input signal center frequency
f_s	Sampling frequency

FET	Field Effect Transistor
FS	Full-scale
FPGA	Field programmable gate array
$H(z)$	Loop filter transfer function
HDL	Hardware description language
IIR	Infinite impulse response
K	Number of symbols used for noise power estimation
L	Modulator order
L	Transistor length (in Chapter 7)
m	Number of bits for storage of coefficients $C_{1..n}$
MIM	Metal-to-metal (capacitor)
MSA	Maximum stable amplitude
n	Number of bits in the quantizer
NTF	Noise transfer function
OSR	Oversampling ratio
P_{qf}	Filtered quantization noise power
P_q	Quantization noise power
PCB	Printed circuit board
PSD	Power spectral density
Q	Lowpass filter output signal in the digital calibration chain
R	Noise power estimate signal
SNDR	Signal-to-noise-and-distortion ratio
SNR	Signal-to-noise ratio
$\text{SNR}_{\max}$	Peak signal-to-noise ratio
SQNR	Signal-to-quantization noise ratio
STF	Signal transfer function
$T(z)$	Lowpass filter transfer function in the digital calibration chain
U	Modulator input signal
UUT	Unit Under Test
V	Modulator output signal
W	Transistor width (in Chapter 7)

Chapter 1

Introduction

1.1 Motivation

Bandpass multibit delta-sigma ($\Delta\Sigma$) modulators are attractive for communication systems because they allow digitization early in the receiver while achieving high resolution at low oversampling ratios (OSR) [1]. Also, for lowpass continuous-time modulators, multibit designs are less sensitive to clock jitter and pulse shape since the voltage gap between two consecutive samples at the digital-to-analog (DAC) output is usually smaller [2]. However, it is well known [2, 3] that the required linearity of the feedback DAC must be in the vicinity of the overall desired modulator resolution. Practical layout techniques with current CMOS processes limits the DAC linearity to 10-12 bits, unless trimming is employed [2, 4].

As will be reviewed in Chapter 3, several techniques exist to overcome the inherent nonlinearity of the DAC; they fall in two categories: dynamic element matching (DEM) and calibration. While DEM techniques with bandpass noise shaping are cumbersome, on the other hand, most calibration techniques must

take the modulator offline and thus preclude continuous tracking of environmental changes. Moreover, the majority of reported calibration techniques and DEM methods known to the author work with n -bit DACs built from $2^n - 1$ unit element cells. The number of DAC elements and the storage requirement for the coefficients in previously reported works increases according to 2^n , typically limiting the number of bits to $n = 4$.

In this thesis, a continuous calibration technique is proposed that requires only a minor addition to the modulator loop, does not inject any signal in the modulator loop, and works with DACs composed of binary weighted cells. Most of the added circuitry is digital and will benefit from the advantages (area, power) of MOSFET feature size scaling.

1.2 Document Outline

In Chapter 2, a coverage of $\Delta\Sigma$ modulation theory is presented. After basic noise-shaping concepts are introduced, high-order loop filter structures are described. Advantages and disadvantages of using a multibit quantizer are emphasized.

Some problems associated with multibit $\Delta\Sigma$ modulators can be overcome. Important references in this research area, including recently published papers, are presented in Chapter 3 to establish the state of the art in this domain.

Chapter 4 and 5 describe in detail the proposed idea for digital compensation of DAC nonlinearities. The theory behind the proposed technique is developed and then several simulation results are included, demonstrating that the proposed

technique is effective.

The simulation results presented in Chapter 4 are validated with experimental results. A chip was fabricated for this purpose. The next two chapters describe the design steps that led to the fabrication of the chip. Chapter 6 describes the design procedure at the system level and Chapter 7 documents the circuit-level design of the chip.

Chapter 8 contains the experimental results demonstrating that the proposed compensation technique works with the fabricated integrated circuit (IC).

A summary of the entire thesis, comprising a list of contributions and ideas for future work is included as the last chapter of the thesis.

1.3 Publications

At the time of depositing the final copy of this thesis, the work described in the thesis resulted in three accepted publications [5–7]:

- G. Gagnon and L. MacEachern, “Digital compensation of DAC mismatches in multibit delta-sigma ADCs,” *Electronics Letters*, vol. 44, no. 12, pp. 721722, May 2008.
- G. Gagnon and L. MacEachern, “Continuous compensation of binary-weighted DAC nonlinearities in bandpass delta-sigma modulators,” in *Proc. IEEE ISCAS 2007*, May 2007, pp. 253256.
- G. Gagnon and L. MacEachern, “System-level design aspects of a DAC

calibration technique for bandpass delta-sigma modulators,” in *Proc. IEEE NEWCAS 2007*, Aug. 2007, pp. 14811484.

Chapter 2

$\Delta\Sigma$ Modulation

This chapter concisely treats the basic theory of $\Delta\Sigma$ modulation. Some amount of knowledge concerning analog-to-digital conversion and oversampling converters is required. The reader is referred to [1, Chapter 1] for a more explanatory coverage of those topics. This chapter is written so that it applies to both lowpass and bandpass $\Delta\Sigma$ modulators.

2.1 Important Definitions

There is a particular terminology used in the fields of $\Delta\Sigma$ modulators and data converters in general. Plus, the definition of some terms changes for different reference materials. To avoid confusion, the most important terms and acronyms are defined here. Note that some of these terms will become clearer throughout the theory presented in this chapter.

$\Delta\Sigma$ *modulator*: The feedback system that quantizes the input signal and shapes the quantization noise.

$\Delta\Sigma$ converter: The complete data conversion system : either an analog-to-digital converter (ADC) or a digital-to-analog converter (DAC). It comprises, *inter alia*, a $\Delta\Sigma$ modulator and either a digital decimation filter (for a $\Delta\Sigma$ ADC) or an analog reconstruction filter (for a $\Delta\Sigma$ DAC).

OSR: Oversampling ratio. The ratio of the modulator sampling frequency, f_s , over twice the input signal bandwidth (band of interest). For a lowpass $\Delta\Sigma$ modulator with an input signal maximum frequency equal to f_{max} ,

$$\text{OSR} = \frac{f_s}{2f_{max}} \quad (2.1)$$

For a bandpass $\Delta\Sigma$ modulator with an input signal bandwidth ranging from frequency f_1 to f_2 [1, Chapter 9],

$$\text{OSR} = \frac{f_s}{2(f_2 - f_1)} \quad (2.2)$$

SNR: Signal-to-noise ratio. The ratio of the input signal power over the total noise in the defined signal bandwidth. The noise power includes the noise power spectral density (PSD) integrated over the signal bandwidth *and* the power of the harmonics of the input signal (caused by distortion) that fall in the signal bandwidth. Such a definition of SNR is equivalent to signal-to-noise and distortion ratio (SNDR). Some authors¹ exclude the power of harmonics in SNR calculation. Therefore, we must always specify that harmonic distortion is included in SNR calculation, or refer explicitly to SNDR. Otherwise, SNR performances can be

¹Especially in papers describing modulators with poor harmonic distortion performances.

interpreted as signal-to-quantization noise ratio (SQNR, see below), an overly optimistic calculation in some cases.

The SNR calculation is commonly performed by taking the FFT (Fast Fourier Transform) of the modulator output. A Hann window is used to avoid tone leakage. The noise power is the sum of FFT power bins (excluding the signal bins²). This calculation assumes that a perfect brickwall filter follows the $\Delta\Sigma$ modulator. Although optimistic, this calculation method is implemented in the “delsig” Matlab toolbox [9] and used in the majority of reported papers in the area of $\Delta\Sigma$ modulation.

SNDR: Signal-to-noise and distortion ratio. In this work, equivalent to SNR. See explanations above.

SQNR: Signal-to-quantization noise ratio. The ratio of the input signal power over the quantization noise power only, integrated over the defined signal bandwidth.

SNR_{max} or peak SNR: the maximum SNR that is measured when the input signal amplitude is swept over the allowable range. It usually occurs at the limit where saturation starts happening in the loop, i.e. within 0 to 3 dB below full scale.

DR (Dynamic Range): the DR of a $\Delta\Sigma$ modulator is often said to be equivalent to the resolution of the modulator. It can be converted to bits of a conventional

²Simply removing the signal bins in the noise power calculation creates a 0.4 dB favorable bias in the SNR value when a 4096-point FFT is used [8]. This bias diminishes when the number of FFT points increases.

Figure 2.1: Hypothetical modulator with different DR and SNR_{max} values.

converter using

$$\text{DR}(\text{bits}) = \frac{\text{DR}(\text{dB}) - 1.76}{6.02} \quad (2.3)$$

The DR is obtained by first plotting the SNR versus the input signal amplitude. The DR is the input amplitude range which gives $\text{SNR} \geq 0$. The DR value is often very close to SNR_{max} but in some cases, both performance measures can differ to a great extent, as shown by the example in Fig. 2.1. Both modulators shown in Fig. 2.1 would have the same DR, hence the designer of the modulator with $\text{SNR}_{\text{max}} = 40$ dB could claim it has near 10-bit resolution. To avoid confusion, whenever possible, this thesis quantifies the performance of $\Delta\Sigma$ modulators using the peak SNR measure instead of the dynamic range.

Full-scale (FS): in this work, a full scale input signal corresponds to the full

range of the feedback DAC. For example, if a 1-bit quantizer is used and the DAC levels are $\pm 0.2V$, a full scale input signal would be a sinusoid of peak amplitude $0.2V$.

dBFS: dB below full-scale. Obtained by:

$$\text{dBFS} = 20 \log \left(\frac{A_i}{FS} \right) \quad (2.4)$$

where A_i is the input signal amplitude.

2.2 Basic Concepts

Whenever analog-to-digital conversion takes place, the quantization process adds noise to the input signal. The quantization noise power is [10]:

$$P_q = \frac{\Delta_q^2}{12} \quad (2.5)$$

where Δ_q is the quantization step size. The above equation assumes that the quantization error is uniformly distributed in the range $(-\Delta_q/2, \Delta_q/2)$. This assumption is valid when the quantization step size is small and the input signal spans several quantization levels between samples. If no noise shaping is used (as in flash converters), the quantization noise PSD is uniformly distributed from 0 to $f_s/2$, where f_s is the converter sampling frequency. The quantization noise PSD is thus

$$S_q(f) = \frac{P_q}{f_s/2} = \frac{\Delta_q^2}{6f_s} \quad (2.6)$$

If the sampling frequency is greater than twice the bandwidth of the input signal to be converted, quantization noise is present outside the signal band. Digital

Figure 2.2: Linear mathematical model of a $\Delta\Sigma$ modulator [8].

filtering can be used to remove the quantization noise outside the signal band. Assuming the use of a perfect brickwall filter with a bandwidth equal to f_b (the input signal bandwidth), the excess quantization noise can be completely removed³. After filtering, the quantization noise power becomes

$$P_{qf} = \left(\frac{\Delta_q^2}{12} \right) \left(\frac{f_s}{2f_b} \right) \quad (2.7)$$

The ratio $f_s/2f_b$ is referred to as the oversampling ratio (OSR). From (2.7), there is a SNR improvement of 3 dB for each doubling of the sampling frequency.

$\Delta\Sigma$ modulators allow greater SNR improvements when the sampling frequency is increased, by shaping the quantization noise out of the signal band. This is done by placing the quantizer in a feedback loop, as shown in Fig. 2.2. Since the quantization noise is inserted at a different point in the loop than the input signal, it undergoes a different transfer function. We thus distinguish the signal transfer function (STF) from the noise transfer function (NTF):

$$STF(z) = \frac{V(z)}{U(z)} = \frac{H(z)}{1 + H(z)} \quad (2.8)$$

³The actual digital filters used for this purpose come very close to the brickwall filter approximation.

Figure 2.3: Conceptual output spectra for lowpass and bandpass $\Delta\Sigma$ modulators [2].

$$NTF(z) = \frac{V(z)}{E(z)} = \frac{1}{1 + H(z)} \quad (2.9)$$

The loop filter $H(z)$ is designed so that its gain is high in the signal frequency band, i.e. $|H(e^{j2\pi f_c})| \gg 1$, where f_c is the center frequency of the input signal⁴. Then, $|STF(e^{j2\pi f_c})| \simeq 1$ and $|NTF(e^{j2\pi f_c})| \simeq 0$. In other words, around the band of interest, the output signal contains a faithful copy of the input signal along with attenuated quantization noise, as depicted in Fig. 2.3 [2].

The order of the $\Delta\Sigma$ modulator, L , refers to the number of zeros in the NTF. Using a simple integrator $H(z) = 1/(z - 1)$ as a loop filter yields a first-order lowpass $\Delta\Sigma$ modulator. In that case,

$$STF(z) = z^{-1} \quad (2.10)$$

$$NTF(z) = 1 - z^{-1} \quad (2.11)$$

The STF is just a delay, while the NTF has a highpass shape. The quantization noise is thus said to be *pushed away* from the signal band.

⁴When $f_c = 0$, the $\Delta\Sigma$ modulator is said to be a lowpass modulator; a bandpass modulator has its center frequency $f_c \neq 0$

More complex loop filters architectures can result in better noise shaping characteristics, as will be explained in the next sections.

2.3 High-Order Modulation and Stability

Adding integrator stages in the loop filter increases the order of the modulator. If all NTF zeros are placed at DC and all NTF poles at infinity, we find the well-known equation for the maximum SNR, which occurs when the input signal is a full-scale sinusoid [1, 11]:

$$\text{SNR}_{\max} = 10 \log \left(\frac{3}{2} 2^{2n} \right) + 10 \log \left(\frac{2L+1}{\pi^L} \text{OSR}^{2L+1} \right) \quad (2.12)$$

where n is the number of bits in the quantizer.

From (2.12), the SNR increases by $3(2L - 1)$ dB for every doubling of the OSR. However, starting from $L > 2$, the modulator is unstable unless some care is taken in positioning the NTF poles: as the NTF poles get closer to the NTF zeros, the modulator becomes more stable. However, the side-effect is an increase of the in-band noise. Also, as the order increases, the input signal maximum stable amplitude (MSA) decreases, which limits the peak SNR ($\text{SNR}_{\max}$). Moreover, for 1-bit quantizers, the gain of the linearized model of the quantizer is dependent on the input signal characteristics, which can make the $\Delta\Sigma$ modulator behavior depart from the linear model.

Because of the above-mentioned factors, (2.12) is not very accurate and thus of limited use, especially for high-order modulators. An empirical study based on

Figure 2.4: Empirical peak SNR limit for 1-bit quantizers [2].

simulations was conducted in [2], and reproduced in part here in Figs. 2.4-2.6, to find the maximum achievable SNR for $n = 1, 2, 3$ and $L = 1, 2, \dots, 8$. The NTF had optimized zeros (see Section 2.4). These simulation results take into account the reduction of the MSA for high-order modulators as well as the nonlinear behavior of $\Delta\Sigma$ modulators. Hence, these results are more accurate than those predicted by (2.12).

2.4 Loop Filter Architectures

The loop filter determines the NTF and the STF. The NTF zeros should be placed in the signal band in order to get useful noise shaping. It is not necessary that all of the NTF zeros be coincident. In fact, spreading the NTF zeros results in better SNR performances, as shown in Fig. 2.7 for a fifth-order lowpass modulator. As explained in the previous section, the NTF poles are chosen as a tradeoff between

Figure 2.5: Empirical peak SNR limit for 2-bit quantizers [2].

Figure 2.6: Empirical peak SNR limit for 3-bit quantizers [2].

stability and noise shaping performances. As a guide for positioning the NTF poles, the Lee criterion [12] states that a $\Delta\Sigma$ modulator with a binary quantizer is likely to be stable if $\max |NTF(e^{j\omega})| < 1.5$. Note that the Lee criterion is neither necessary nor sufficient for stability; it needs to be confirmed by simulations. Nonetheless, it is a good indicator and it is widely used.

In most loop filter architectures, the STF and NTF must share the same poles. The NTF poles must result in a flat STF in the signal band. Since $\Delta\Sigma$ modulation theory is based on a linear model of a nonlinear system, there is no analytic solution as to the ideal placement of NTF poles [2]. Software tools, such as the “delsig” toolbox for Matlab [9], exist to help the designer in this task. Appendix A describes the Matlab functions that were used in this thesis.

The STF acts as a prefilter on the input signal. The STF poles and zeros must thus be placed so that the STF magnitude is flat in the signal band. An obvious choice is $|STF(z)| = 1$ over the entire band. Depending on the loop filter architecture, the STF can also be chosen to provide beneficial prefiltering on the signal. For example, an STF with a bandpass frequency response can help removing adjacent channels in a bandpass $\Delta\Sigma$ modulator used in communication systems [13].

The first loop architecture presented here is built from a cascade of integrators with distributed feedback and distributed coupling (CIFB). It is shown in Fig. 2.8. The loop filter can be extended to give an L^{th} order modulator by having a total

Figure 2.7: NTF with all zeros at DC as well as NTF with optimized zeros, for a fifth-order modulator with $OSR = 32$. The integrated noise power in the signal band is 18 dB lower for the optimized-zeros curve.

of L integration stages. Note that by making $a_4 = b_4 = 1$ and all other coefficients equal to zero, we find the simple first order modulator presented in the previous section.

Using the CIFB configuration, the NTF is [2],

$$NTF(z) = \frac{(z-1)^L}{D(z)} \quad (2.13)$$

where

$$D(z) = a_1 + a_2(z-1) + \dots + a_L(z-1)^{L-1} + (z-1)^L \quad (2.14)$$

The STF is,

$$STF(z) = \frac{b_1 + b_2(z-1) + \dots + b_{L+1}(z-1)^L}{D(z)} \quad (2.15)$$

From (2.13)-(2.15), the following observations can be made:

Figure 2.8: CIFB structure [2].

- Coefficients $a_{1..L}$ simultaneously position the poles of the NTF and the STF;
- Coefficients $b_{1..L+1}$ position the zeros of the STF. By choosing $b_i = a_i$ for all $i \leq L$ and $b_{L+1} = 1$, the STF is 1 over the entire band [14];
- The circuit can be simplified by choosing $b_1 = 1$ and $b_{2..L+1} = 0$. From (2.15), $STF(z)$ becomes $1/D(z)$. The coefficients $a_{1..L}$ must be adjusted to give a flat STF magnitude in the signal band;
- All of the NTF zeros lie at $z = 1$ (DC).

From Fig. 2.7, having all of the NTF zeros at DC leads to lower SNR performances than spreading the zeros in the signal band. In order to move the NTF zeros away from DC, the loop filter must contain resonators instead of integrators. A modification of the CIFB loop filter structure that comprises resonators is the

Figure 2.9: CRFB structure [2].

CRFB, shown in Fig. 2.9. This structure implements NTF zeros as conjugate complex pairs at frequencies $|\omega_1| = \cos^{-1}(1 - g_1/2)$ and $|\omega_2| = \cos^{-1}(1 - g_2/2)$.

Another possible configuration is the error feedback structure, shown in Fig. 2.10. Using this structure, the STF is 1 and the NTF is $1 - H(z)$. A first order $\Delta\Sigma$ modulator can thus be realized with $H(z) = z^{-1}$ (a simple delay), while making $H(z) = z^{-1}(2 - z^{-1})$ implements a second order $\Delta\Sigma$ modulator.

The error feedback structure is rarely used for ADCs because it is sensitive to any imperfection in the loop filter. However, it is a hardware-efficient structure for $\Delta\Sigma$ DACs. As will be discussed in Chapter 3, it can also be used to shape the mismatch noise of a multibit DAC in $\Delta\Sigma$ ADC loops.

Other structures are also possible [2]. Since they do not offer more flexibility in NTF pole and zero placement than the CRFB structure, they are not presented

Figure 2.10: $\Delta\Sigma$ modulator employing the error feedback structure [2].

here.

The designer usually relies on software tools such as the “delsig” Matlab toolbox to adequately position the NTF poles and zeros for a given modulator specifications: order, number of bits, OSR, etc. Once a loop filter architecture has been selected, such software tools can also calculate the value of the loop filter coefficients used to implement the desired NTF. The stability and SNR performances can be verified in simulation before the modulator is implemented.

2.5 Quantizer

Early $\Delta\Sigma$ converters used a single-bit quantizer in the loop [15–18]. First of all, for circuit simplicity: a 1-bit quantizer can be built around a simple comparator. Moreover, the feedback DAC can be embedded within the quantizer since it only needs to regenerate two reference voltages. The 1-bit quantizer offers another undeniable advantage: it is inherently linear [19]. Besides, early $\Delta\Sigma$ converters were mainly used for low-bandwidth signals such as audio, so high OSRs were *relatively* easy to achieve. While recent implementations for specific applications

still use 1-bit quantizers [20–22], the majority of recently reported $\Delta\Sigma$ modulators use multibit quantization [23–29].

There are clear advantages in using a multibit quantizer in a $\Delta\Sigma$ loop:

1. There is a 6 dB SNR improvement for each additional quantizer bit. A given SNR target can be reached using lower OSR, i.e. a lower sampling frequency [2];
2. A multibit $\Delta\Sigma$ modulator behaves more like the linearized mathematical model than the 1-bit modulator. This results in more stable high-order modulators; which in turns allows a higher peak SNR for a given OSR [2];
3. The requirements on the circuit components, especially the operational amplifiers, are relaxed [30];
4. For continuous-time lowpass modulators, multibit designs are less sensitive to clock jitter and pulse shape since the voltage gap between two consecutive samples at the DAC output is usually smaller [31]. This is less true for bandpass modulators because the sampling frequency is not much higher than the center frequency.

The above first two advantages combine to give more than 6 dB improvements in SNR_{max} for each additional quantizer bit. For example, refering back to Fig. 2.4-2.6, for a fourth order ($L = 4$) modulator and $\text{OSR}=32$, the peak SNR increases from 85 dB to 125 dB when going from a 1-bit to a 3-bit quantizer; a 40 dB

improvement!

Unfortunately, as soon as we cross the barrier of 1-bit quantization, the inherent linearity of both the quantizer and the DAC is lost. Nonlinearities in the quantizer are not problematic because the quantizer is preceded in the loop by large-gain integration stages; the input-referred errors are usually negligible [32]. On the other hand, since the DAC output is connected to the summing node at the $\Delta\Sigma$ modulator input, any misplaced level in the DAC will have a proportional effect on the error signal, and, consequently, on the output signal. This is why it is often said that the DAC linearity must be equivalent to the overall $\Delta\Sigma$ converter linearity. For example, if a $\Delta\Sigma$ converter is designed to achieve a peak SNR of 70 dB with a quantizer with $n = 3$ bits, the 3-bit DAC in the feedback path must have the linearity of a 12-bit DAC. Otherwise, the harmonic distortion and the noise floor will increase and limit the maximum achievable peak SNR. This is illustrated in Fig. 2.11 [3]. The next chapter treats reported methods of overcoming the DAC nonlinearity effects in $\Delta\Sigma$ modulators.

2.6 Summary

In this chapter, the basic theory of $\Delta\Sigma$ modulation was covered. It was shown that a $\Delta\Sigma$ modulator comprises a quantizer in a feedback loop, so that the quantization noise is shaped away from the signal band. A filter usually follows any $\Delta\Sigma$ modulator to remove the quantization noise outside the signal band. $\Delta\Sigma$ converters

Figure 2.11: SNDR versus oversampling ratio for a 4th-order 4-bit $\Delta\Sigma$ modulator to show the effect of DAC nonlinearity [3].

make use of oversampling and noise shaping instead of high-accuracy analog components to achieve high resolution.

Simulation results from an empirical study of $\Delta\Sigma$ modulators were reported and show that high resolution can be achieved at low OSR when a multibit quantizer and high order loop with optimized NTF zeros are used. The CRFB loop filter architecture, which allows freedom in positioning the NTF poles and zeros was analyzed.

Finally, it was shown that multibit $\Delta\Sigma$ modulators pose stringent linearity requirements on the feedback DAC. The following chapter describes some state-of-the-art methods of limiting the unwanted effects of inevitable DAC nonlinearities.

Chapter 3

State-of-the-Art in Compensation of DAC Nonlinearities

In the previous chapter, it was shown that using a multibit quantizer in a $\Delta\Sigma$ modulator loop had significant advantages. It also had a major drawback: since the DAC errors are input-referred, the DAC linearity must be in the vicinity of the $\Delta\Sigma$ modulator desired SNR. Several techniques exist to overcome the inherent DAC nonlinearity. Important previous works in this area are summarized in this chapter.

3.1 Dynamic Element Matching (DEM)

Let us consider an n -bit DAC built from $N = 2^n - 1$ unit element cells fed by a thermometer-coded digital input signal $V_{1..N}$. Those cells could be implemented using N identical current sources, enabled by the signal V , and summed to create an output current V' .

- The minimum DAC output value V'_{min} is generated when all input bits

$V_{1..N} = 0$. All unit-element cells are disabled.

- The full-scale DAC output value V'_{max} is generated when all input bits $V_{1..N} = 1$. All unit-element cells are enabled.
- For all other desired output values, there are multiple possible configurations. For example, the desired DAC output value equivalent to 1 LSB is generated when only one of the N input bits is 1. There are thus N possible configurations to generate $V' = 1 \text{ LSB}$.

If the unit-element cells are not identical, e.g. due to limitations in IC fabrication, the static DAC response will deviate from its ideal response, as shown in Fig 3.1. It can be mathematically shown [2] that for each input word, the average of all possible output values falls exactly on an ideal line drawn from V'_{min} to V'_{max} . So, instead of always choosing the same configuration for a given desired output value, if an element selection logic (ESL) block controls the selection of the unit elements so that all elements are used equally often, as in Fig 3.2, the *averaged over time* DAC transfer function is linear, with each sample deviating from the (linear) average response. In the frequency domain, this has the effect of transforming the harmonic spurs caused by static DAC nonlinearity into pseudo-random noise. Depending on the choice of the ESL, the spread noise can either be flat or shaped away from the signal band. The following subsections describe various ESL schemes.

Figure 3.1: Static response of a 2-bit DAC composed of 3 mismatched unit-element cells.

Figure 3.2: General block diagram of DEM.

3.1.1 Randomization

The simplest ESL scheme is to randomly choose the unit elements at each clock cycle [33]. The DAC errors are no longer correlated with the input signal; this converts the harmonic distortion spurs into frequency-flat noise. Since the resulting noise is spread from DC to $f_s/2$, a portion (relative to the OSR) of this excess noise will be eliminated by the decimation filter at the $\Delta\Sigma$ modulator output. In some applications, the reduction in SNR performances caused by this excess noise is more acceptable than the harmonic distortion spurs caused DAC nonlinearity [2]. For such applications, random DEM is a simple solution to DAC nonlinearity.

3.1.2 Data Weighted Averaging (DWA)

The DWA algorithm [34] for ESL is also called the element rotation algorithm. Instead of selecting the DAC elements in a random fashion, the DAC elements are *sequentially* selected. For example, for a 3-bit DAC (built from 7 elements), if the first 3 input words are 2, 3 and 6, then elements 1 and 2 will first be enabled (input word = 2), then elements 3, 4 and 5 will be enabled (input word = 3), and then elements 6, 7, 1, 2, 3 and 4 will be enabled (input word = 6). Each element is thus used in turn and as often as possible.

This ESL scheme shapes the mismatch noise like a first-order highpass filter. The mismatch noise is thus shaped away from the signal band (for a lowpass $\Delta\Sigma$ modulator), and an important portion can be removed by the same process

by which the quantization is removed outside the signal band.

The DWA algorithm suffers from the same pathology as a first-order $\Delta\Sigma$ modulator: it is prone to tone generation. To understand why, consider a DC input of value 3 to the DAC. The DWA algorithm will pick the first 3 consecutive DAC elements for the first sample. Then, it will pick the next 3 elements, and so on. When it reaches the last element, it starts over with the first. After a few iterations, the same 3 elements that were first selected will be picked up again, and the exact same sequence will start over. Since there will be a repetition of the element selection sequence, there is a periodicity on the mismatch error. This causes frequency spurs at the DAC output. The location and amplitude of the spurs depend on the length of the periodicity, i.e. the value of the DC input signal and the number of DAC elements. The same mechanism happens with low-frequency input signals.

As a remedy to the tone-generation related problems of DWA, the bi-DWA was proposed in [35]. At each cycle, the rotation direction is changed. This lengthens the periodicity. In some cases, it completely removes the spurs caused by DAC nonlinearity, while still shaping the mismatch noise by a first-order highpass filter like the DWA. The price paid is an increase in the noise floor, as the simulation results of Fig. 3.3 [2] illustrate.

Other modifications were proposed to remove the cyclic behavior of DWA

Figure 3.3: Simulation results comparing DWA and Bi-DWA [35].

in [36–38]. All of these techniques rely on the basic idea of adding some randomization to DWA. They can remove the tones, but, to different degrees, result in an increase of the noise floor [39].

3.1.3 Higher-Order Mismatch Shaping

The random DEM and the DWA DEM result respectively in zero and first order mismatch noise shaping. The efficacy of those technique increases with the OSR. Depending on the mismatch severity, they are only effective at high OSRs. This is unfortunate, since multibit $\Delta\Sigma$ modulation is often employed precisely to achieve a certain SNR target at reduced OSR. It was demonstrated in [40] that higher-order mismatch shaping is possible by using N digital $\Delta\Sigma$ modulators (one for each unit element).

The high-order DEM scheme is shown in Fig. 3.4. It is based on the error

Figure 3.4: General block diagram of high-order DEM [1].

Figure 3.5: $\Delta\Sigma$ modulator employing the error feedback structure [2].

feedback $\Delta\Sigma$ structure, presented in Chapter 2, and reproduced in Fig. 3.5 for convenience.

In Fig. 3.4, all bold signals are vectors of size N . The loop filter $H(z)$ also needs to be implemented N times. The vector quantizer block provides coupling between the N $\Delta\Sigma$ modulators. The input signal sy of the vector quantizer represent the desired usage of each of the N unit elements. It uses this information to select which elements to enable by selecting those elements with the largest sy components. It can be shown [40] that using this technique, the output of the

DAC becomes

$$V'(z) = V(z) + H(z)(SE(z) \cdot de) \quad (3.1)$$

where $SE(z)$ is a vector representing the z-transform of the vector signal se , and de is a (static) column vector representing the difference between the actual element values and the average of all elements. The term $(SE(z) \cdot de)$ is thus noise due to element mismatch. Depending on the choice of $H(z)$, the element mismatch noise can be spectrally shaped away from the signal band. Using $H(z) = 1 - z^{-1}$ yields a behavior similar to DWA [1], while using $H(z) = (1 - z^{-1})^2$ results in stable second-order noise shaping. Even bandpass noise shaping can be obtained by using $H(z) = 1 - z^{-1} + z^{-2}$.

A comparison of various DEM techniques was done in [41], specifically for low OSRs. It was found that DWA is the best in terms of SNDR, while other techniques such as Bi-DWA are better in terms of Spurious-Free Dynamic Range (SFDR). They also state that, for low OSRs, second-order DEM schemes are not better than first-order DEM. This is explained in more detail in [39], where it is shown that second order DEM using the structure of Fig. 3.4 can never achieve perfect $(1 - z^{-1})^2$ noise shaping. As a remedy, [39] proposes to modify $H(z)$ so that both NTF zeros are no longer at DC, but at an optimal frequency α . The loop filter is now $H(z) = 1 - 2z^{-1} \cos \alpha + z^{-2}$. For an OSR of 8, $\alpha = 2\pi f_s/25$. Using this technique, an SNR improvement of 6 dB is reported when compared to $H(z) = (1 - z^{-1})^2$. The price paid is increased circuit complexity to implement

the modified $H(z)$ (remember that $N = 2^n - 1$ loop filters are required in high-order DEM).

3.1.4 Bandpass DEM

For multibit bandpass $\Delta\Sigma$ modulators, DEM must shape the mismatch noise away from the signal band, centered at a nonzero frequency f_c . Of course, random DEM can be used. As explained earlier, random DEM does not shape the mismatch noise at all; it simply spreads the spurious tones into wideband noise which can cause significant SNR reductions, especially for low OSRs. DWA cannot be used with bandpass $\Delta\Sigma$ modulators because it shapes the mismatch noise away from DC.

One possible solution is to use the structure of Fig. 3.4 and make $H(z)$ resonators centered at f_c . This has been successfully implemented in [42] for a multistage (MASH) 2-2 $\Delta\Sigma$ modulator with a 9-level DAC ($N = 8$), to achieve a measured peak SNDR of 77.4 dB. The block diagram of the resulting bandpass DEM scheme is shown in Fig. 3.6.

A completely different technique was proposed in [29] for quadrature $\Delta\Sigma$ modulators. A complex digital mixer is inserted before the DAC to downconvert the signal centered at f_c to DC. Lowpass DEM is performed. Another mixer is inserted after the DAC to upconvert the signal from DC to f_c . When $f_c = f_s/4$, the second input of the mixers is the simple sequence $e^{j\pi n/2}$; no oscillator nor multiplier are required. This technique is specific to complex $\Delta\Sigma$ modulators because

Figure 3.7: Block diagram of a modified tree-structured DEM [44].

Figure 3.8: Simplified block diagram of a compensated nonlinear system.

Fig. 3.8 [45], where N_1 , N_2 and N_3 are respectively the transfer functions of the ADC, the DAC and a digital compensation block. If the loop gain is high enough in the band of interest, then $V' \simeq U$ in this band. If N_3 can be made equivalent to N_2 , then $V'' = V' \simeq U$, and the unwanted effects of the DAC nonlinearity are compensated at the modulator output V'' . The challenge becomes setting $N_3 = N_2$.

In [30] and [45], N_3 is found at power-up by rearranging the loop so that it becomes a 1-bit $\Delta\Sigma$ ADC. This 1-bit ADC is used to characterize the output of the DAC (N_2) while it is fed by a slow counter from 0 to $2^n - 1$. Since 1-bit $\Delta\Sigma$ ADCs are known to be highly linear, the analog output corresponding to each DAC input can be precisely measured and stored in a flash memory. Each DAC input code must be maintained long enough so that the 1-bit $\Delta\Sigma$ ADC can measure its corresponding analog output value with enough precision. The $2^n - 1$ measured values are stored using m bits; the choice of m depends on the desired linearity of the overall $\Delta\Sigma$ converter. The method in [30] is more hardware-efficient than [45]

Figure 3.9: Block diagram of an analog self-calibrated $\Delta\Sigma$ modulator [3].

since it stores only the offset from the ideal values.

A similar technique, shown in Fig. 3.9, presented in [3], also relies on a 1-bit $\Delta\Sigma$ ADC to accurately measure the DAC output. The main difference with [30,45] is that instead of correcting the DAC errors in the digital domain, an additional *fine DAC* is added. At each clock cycle, the *fine DAC* is fed with the previously measured and stored error of the *coarse DAC* for the current corresponding input word. The output of both DACs are summed. The *coarse DAC* errors are thus corrected in the analog domain.

A major inconvenience in [3, 30, 45] is that calibration is only performed at power-up. Any changes due to thermal drift cannot be tracked. The technique proposed in [46] allows continuous (background) calibration of the DAC elements. Here is how this is accomplished:

- The DAC is built from $N + 1$ elements (one extra element is used);

- N elements are used for normal DAC operation, while the extra element is being calibrated;
- When the calibration of the extra element is complete, it is used as part of the DAC, and another element is taken out for calibration.

This basic idea was first proposed in [47], where the calibration of each cell was based upon the charge storage on the gate-source capacitance of CMOS transistors to create multiple copies of a reference current, and implemented in a 16-bit DAC. In [46], the novel idea is in the way that each element is calibrated:

- The input of the element being calibrated is connected to a *calibration sequence* signal, while its output is still connected to the $\Delta\Sigma$ loop, as shown in Fig 3.10.
- The calibration sequence could be any signal that can easily be identified at the $\Delta\Sigma$ modulator output. In [46], a simple alternating 0,1,0,1,... signal is used. This creates a single tone at $f_s/2$, away from the signal band.
- The amplitude of the tone at $f_s/2$ is measured at the $\Delta\Sigma$ modulator output and stored in memory.

This is done for each of the $N+1$ DAC elements. At the $\Delta\Sigma$ modulator output, each output bit is replaced by its corresponding stored value. The calibration sequence creates a tone at $f_s/2$, where the quantization noise PSD is high. The

Figure 3.10: Selection multiplexer to calibrate one DAC element at a time while the other elements perform normal conversion [46].

tone can be buried in noise, and the calibration time can thus be long. This is circumvented in [46] by modifying the loop filter to add an NTF zero at $f_s/2$, as shown in Fig. 3.11.

It is interesting to note that basically the same calibration technique, including tone generation at $f_s/2$ and the additional NTF zero at $f_s/2$ was later presented in [48], without any reference to [46]. The experimental results presented in [48] show an improvement of 7 dB in SNDR performances (from 59 to 66 dB using calibration). The major improvement was in the spurious-free dynamic range (SFDR), which went from 61 to 87 dB using calibration. A second order $\Delta\Sigma$ modulator with $\text{OSR}=64$ and $n=4$, as was used in [48], should theoretically provide more than 100 dB SNR. The experimental chip was probably dominated by thermal noise, which could explain why the SNR is limited to 66 dB after

Figure 3.11: Output spectrum showing the calibration tone at $f_s/2$ and the notch in the NTF.

calibration.

Finally, the solution presented in [49] also relies on pseudo-random noise injection and LMS (least mean square) adaptation to characterize the $\Delta\Sigma$ loop. Unfortunately, the hardware realization of the proposed technique had major flaws and its efficacy could not be verified.

3.3 Other Techniques

There are several additional techniques available in the literature to compensate for DAC mismatches in $\Delta\Sigma$ modulators. While not every technique can be described in this thesis, the interested reader in the subject is strongly encouraged to read [50–54].

3.4 Summary

This chapter reviewed important publications in the area of compensation of DAC nonlinearity in $\Delta\Sigma$ modulators. First, it was shown that the effects of DAC nonlinearities can be attenuated by using DEM. DEM relies on shuffling the selection of the DAC unit elements to convert the spurs due to nonlinearity into spread mismatch noise. Depending on the algorithm used for the element selection, the mismatch noise can be shaped away from the signal band. Since DEM rely on mismatch noise shaping, in general, DEM is gradually less efficient as the mismatch increases and as the OSR diminishes.

The second family of DAC nonlinearity compensation techniques involves calibration. All the calibration techniques that were presented in this chapter rely on measuring each of the elements in the DAC and using this information to compensate for the mismatch between the DAC elements. One possible strategy to achieve this is to perform calibration at power-up; the $\Delta\Sigma$ loop is temporarily rearranged so that it becomes a highly-linear 1-bit $\Delta\Sigma$ ADC and then this 1-bit ADC is used to measure the value of each DAC element. Analog and digital calibration are possible using this technique. A background calibration technique that injects an out-of-band signal using an extra DAC element was also described in this chapter. The out of band signal is measured at the output for each DAC element.

Chapter 4

Proposed Solution for Compensation of DAC Nonlinearities

This chapter presents the proposed idea for compensation of DAC nonlinearities.

It is an expanded version of a paper presented at the 2007 IEEE International Symposium on Circuits and Systems (ISCAS 2007) [7].

4.1 The importance of binary-weighted DAC compensation

As was explained in Chapter 2, adding bits to the quantizer pays off more than 6 dB per bit because of the combined effect of increased quantizer resolution and more stable operation for high-order modulators. It was also shown that as soon as we trade the 1-bit quantizer for a multibit quantizer, we must be concerned about the linearity of the DAC in the feedback path of the $\Delta\Sigma$ modulator. The DAC linearity must be in the vicinity of the overall desired $\Delta\Sigma$ modulator resolution. This means that the *absolute* precision requirements of the DAC levels are the

same for a 2-bit, 3-bit or 8-bit DAC. With that in mind, it seems like a good design strategy to either use a 1-bit quantizer and a very high OSR, or, if we are to use a multibit quantizer, push the number of bits high and use a low OSR. This allows achieving high resolution at lower OSRs, while preserving the advantages of $\Delta\Sigma$ converters over flash ADCs. Those advantages were described in Chapter 2.

In most reported multibit $\Delta\Sigma$ modulator implementations, the DAC is implemented using $2^n - 1$ unit element cells. This is done because: 1 – matching between cells is easier, 2 – it interfaces well with a flash-ADC quantizer and 3 – conventional DEM schemes work with this DAC architecture. However, with a unit-element cells DAC, the number of elements increases following 2^n , with n being the number of bits. The coefficient storage requirement of the compensation technique also increases exponentially with n , typically limiting the number of bits to 4.

The quantizer complexity can also limit the number of quantizer bits, n . Most $\Delta\Sigma$ modulators use a flash ADC as the quantizer, so the complexity and power consumption of the quantizer will be problematic if n is overly increased. $\Delta\Sigma$ modulators employing different architectures for the quantizer have been recently reported [23, 24, 55, 56]. Therefore, a non-flash ADC can be used as the quantizer along with a binary-weighted DAC, so that n can be increased to achieve high SNR performances at very low OSRs.

The calibration technique described in this chapter works with DACs composed of binary-weighted cells. Therefore, the complexity of the DAC and the compensation technique increases only linearly with the number of quantizer bits. To the knowledge of the author, this is the first reported continuous (background) DAC compensation technique for $\Delta\Sigma$ modulators using a binary weighted DAC.

4.2 System Description

It was explained in Chapter 3 that the system shown in Fig. 4.1 can compensate nonlinearities in N_2 if N_3 can be made equivalent to N_2 and the loop gain is high enough in the band of interest. The challenge resides in setting $N_3 = N_2$.

When the feedback DAC N_2 is nonlinear, not only does harmonic distortion increase, but the noise floor also increases due to folding of quantization noise, gradually filling in the notches of the noise transfer function (NTF) [2]. Any calibration algorithm that minimizes the noise power in those NTF notches will also reduce harmonic distortion, leading to enhanced signal to noise and distortion ratio (SNDR) performances¹.

During normal operation, the input signal frequency range covers the NTF notches, so minimization of the noise in the signal band is difficult to perform on-line. If an additional zero outside the signal band is added to the NTF, it creates what will be called an *observation band*. The noise due to DAC nonlinearity fills

¹Unless otherwise specified, all SNR results reported in this thesis include the effect harmonic distortion, so $\text{SNDR} = \text{SNR}$.

Figure 4.1: Simplified block diagram of a compensated nonlinear system.

the notch in the observation band in the same fashion as it does in the signal band. This is depicted in Fig. 4.2 where the observation band was created by adding a NTF zero at DC. It is demonstrated in this chapter that if the noise power near DC is minimized through calibration of nonlinear block N_3 , the noise power and harmonic distortion in the signal band are *ipso facto* minimized. With judicious placement of NTF poles, the SNR penalty due to the creation of the observation band can be less than 1 dB, as shown in Fig. 4.3

A detailed model of the proposed solution is shown in Fig. 4.4. The selected loop filter structure is the CRFB (see Chapter 2) because it allows placement of the NTF zeros optimally, which improves SNR, especially for a low OSR. The architecture can be simplified by making $b_0 = 1$ and $b_{1..5} = 0$, which causes the signal transfer function (STF) to be maximally flat.

The digital calibration chain consists of a nonlinear transfer function, a lowpass filter, a power estimator and a minimization algorithm. Each block is described in the following subsections.

Figure 4.2: Modulator output spectrum with NTF zero at DC to show the noise floor increase due to DAC nonlinearity. An exaggerated $\pm 10\%$ mismatch in the binary weighted DAC element values is used to clearly show the notch noise-filling due to DAC nonlinearity.

Figure 4.3: Bode diagram of NTF with and without the observation band (at DC). The additional NTF zero allows the calibration algorithm to find an optimized nonlinear transfer function by minimizing the noise near DC.

Figure 4.4: Detailed model of the digitally compensated $\Delta\Sigma$ modulator. The supplementary circuitry is enclosed in the two dashed boxes.

4.2.1 Nonlinear Transfer Function

The nonlinear transfer function performs the following operation:

$$P = \sum_{i=1}^n V_i \cdot C_i \quad (4.1)$$

where n is the number of bits at the modulator output, V_i is the value (either 0 or 1) of the i^{th} bit of the modulator output signal V , and C_i is one of the n stored value to represent the binary weighted element of the DAC. The $C_{1..n}$ values are stored using m bits, where $m > n$, and m is in accordance with the desired overall resolution of the modulator. For example, if the ADC and DAC within the modulator are implemented using $n = 4$ bits and the desired resolution of the ADC is around 70 dB, then the nonlinear transfer function block will need to store 4 words of $m = 12$ bits. This is in contrast with previous calibration methods where 2^n words of m bits are stored [45, 46]. To further reduce circuit complexity, only the offset from the ideal values of $C_{1..n}$ can be stored, as was done in [30]. This will reduce the layout size of the implemented modulator.

Chapter 6 provides more elaborate indications for setting the value of m .

4.2.2 Lowpass Decimation Filter and Power Estimator

The lowpass decimation filter is required to estimate the noise power in the observation band. Since it does not have to process the input signal U , its performance requirements such as magnitude ripple and linear phase response are relaxed. An efficient implementation uses a cascade of simple Infinite Impulse Response (IIR)

Figure 4.5: Lowpass filter cell realizing $T_c(z)$.

cells having the transfer function:

$$T_c(z) = \frac{2^{-A}}{z - (1 - 2^{-A})} \quad (4.2)$$

Each cell $T_c(z)$ is a first-order lowpass filter with a pole at $z = (1 - 2^{-A})$ and a gain of 1 in the passband. The cutoff frequency of the lowpass filter does not need to be precisely controlled, so A can be restricted to integers for an efficient multiplier-free implementation. The block diagram showing a possible implementation of eq. (4.2) is shown in Fig. 4.5.

The power of the signal at the output of the lowpass filter is:

$$P_{ob} = \frac{1}{K} \sum_{k=1}^K Q_k^2 \quad (4.3)$$

where K is the number of samples used to do the calculation and Q is the signal at the output of the lowpass filter (see Fig. 4.4). While the power estimator block could perform this operation and send the result to the minimization algorithm, eq. (4.3) has both division and multiplication operations which are more complex to implement in hardware than additions and subtractions. If K is a constant, the $1/K$ term is just a constant gain factor and can be removed without weakening the calibration process. To remove the multiplication operation Q_k^2 , the following

power estimator can be used:

$$R_j = \sum_{k=1}^K |Q_k| \quad (4.4)$$

While eq. (4.4) does not result in the exact value for the noise power in the observation band, it does provide an estimate that is proportional to the square root of that quantity. Since it is sufficient for the power estimator to give a proportional estimate of the noise power for the minimization algorithm to work and since the hardware implementation of $|x|$ is simpler than x^2 , the simple integrate and dump defined by eq. (4.4) is favored to implement the power estimator. Note that the power estimator operates on the decimated signal Q and is thus clocked at a lower frequency than the modulator. It outputs a power estimate value R_j every K samples of the lowpass decimation filter.

The actual values of A and K and the method to determine them are in Chapter 6.

4.2.3 Minimization Algorithm

The minimization algorithm must find the optimum set of $C_{1..n}$ values in the non-linear transfer function block that minimizes the noise estimates R_j . It becomes an n -variable optimization problem. The optimum solution could be found using various multi-variable (gradient-based or not) optimization techniques [57]. One must remember that the computation has to be performed by dedicated hardware on the ADC chip, not by a CPU or a DSP chip. Some popular and efficient

optimization techniques are simply not suitable for this application. Chapter 5 describes a hardware-efficient minimization algorithm that was used to obtain the experimental results reported in Chapter 8. For now, let us just take for granted that an algorithm (and many different solutions would work) will find the set of coefficients that minimizes the noise in the observation band.

4.2.4 DC Offset and Noise

This calibration technique relies on minimization of the noise near DC. If the input signal U contains a certain DC offset, the optimization process will converge to a solution where the low-frequency noise becomes insignificant when compared to the DC component. The DAC nonlinearity will not be fully compensated and there will be some residual noise in the signal band. The amount of residual noise will be proportional to the DC offset magnitude.

A similar behavior happens if the input signal U contains significant noise near DC. In such a case, the optimization algorithm will converge to the solution where the input noise is the major noise source.

The preceding two situations may not be of primary concern with practical implementations where the modulator is preceded by a bandpass filter and differential circuits are used. But, in any case, DC offset and noise related problems are easily circumvented by adding a zero at DC in the STF. This is done through proper selection of coefficients in the loop filter, and adds beneficial filtering capabilities to the $\Delta\Sigma$ modulator.

4.3 Simulations

The system of Fig. 4.4 has been simulated in Matlab to evaluate the performances of the calibration technique.

4.3.1 Simulation Parameters

The system parameters are the same as the fabricated chip whose result are reported in Chapter 8. Chapter 6 describes the methodology used to set those parameters.

The center frequency of the bandpass modulator is $f_s/4$, where f_s is the modulator operating frequency. The loop filter coefficients have been found using the Matlab “delsig” toolbox [9], for a fourth order bandpass modulator with optimized NTF zeros, $OSR = 14$ and one STF zero at DC. The resulting NTF has zeros at $f/f_s = \{0, 0.238, 0.262\}$. The modulator internal quantizer and DAC are $n = 5$ bits. The resolution for the calibrated coefficients $C_{1..5}$ is $m = 13$ bits. The decimation factor in the lowpass filter is $D = 16$. The value of A , which sets the pole frequency of the lowpass filter, is set to 9. The number of samples for noise power estimation is $K = 2^{16}$. Simulation was performed by forcing errors of $\pm 1\%$ in the DAC binary weighted element values; the resulting DAC element values are: $\{0.99, 2.02, 3.96, 8.08, 15.84\}$, scaled and offset to give a full-scale range of ± 1 . White noise is added to the input signal to model thermal noise and limits the peak SNR to approximately 74 dB.

4.3.2 Simulation Results

Fig. 4.6 shows the frequency spectrum before and after calibration. Once calibration is complete, the noise floor at the modulator output dropped by approximately 10 dB and the harmonic distortion improved by 37 dB. A plot of SNR² for various input amplitudes is shown in Fig. 4.7. The peak SNR after calibration is 73.9 dB. This is 0.4 dB less than the ideal DAC case (74.3 dB) and more than 26 dB better than the non-calibrated case (47.1 dB). The calibrated results are within 1.5 dB of the ideal-DAC case over the entire range of input amplitudes.

Simulations were also run using a two-tone input signal to verify the performances of the proposed calibration technique when the input signal is not a simple sinusoid. The output spectra with and without calibration are shown in Fig. 4.8. The calibration algorithm performed equally well in this situation, with the highest third-order intermodulation component (IM3) at -80 dBFS, an improvement of 29 dB over the non-calibrated case.

The calibration time is dependent on the minimization algorithm. The proposed minimization algorithm is presented in the next chapter, along with the calibration time.

4.4 Compensation of Lowpass $\Delta\Sigma$ Modulators

The compensation technique described in this thesis has also been successfully applied to lowpass $\Delta\Sigma$ modulators in [5] with two modifications: the observation

²The SNR is obtained with the Matlab command `calculateSNR` from the “delsig” toolbox.

(a) Without calibration.

(b) With calibration.

Figure 4.6: Power spectrum at modulator output before (a) and after (b) the calibration process. The odd-numbered harmonics are aliased into the signal band due to the sampling process. The input signal frequency is slightly offset from $f_s/4$ to clearly show the harmonics. The input signal amplitude is -1 dBFS.

Figure 4.7: Simulation results showing SNR vs. input amplitude.

band is at $f_s/4$ instead of DC and the lowpass decimation filter is replaced by a bandpass filter. The interested reader is invited to consult [5] for additional information.

4.5 Summary

This chapter presented a novel digital calibration technique for bandpass multibit $\Delta\Sigma$ modulators. An NTF zero is added at DC to create an observation band, which allows minimization of the noise while the modulator is in operation. The noise is minimized by digitally calibrating the coefficients of a nonlinear function. Simulation results showed that it converges rapidly with significant improvements in SNR performances.

(a) Without calibration.

(b) With calibration.

Figure 4.8: Power spectrum at modulator output before (a) and after (b) the calibration process using a two-tone input.

Chapter 5

Minimization Algorithm

The previous chapter provided a high-level description of the block “Minimization Algorithm” in Fig. 4.4, simply stating that it was an algorithm that has to solve a multivariable optimization problem for which the variables are the coefficients $C_{1..n}$ of the nonlinear transfer function, and the objective function to minimize is the noise power in the observation band. It was also said that many different algorithms could accomplish this task.

This chapter presents in detail the minimization algorithm that was used to calibrate the coefficients of the implemented $\Delta\Sigma$ modulator. This algorithm is suitable for this task because it is hardware-efficient; no multiplication or division are required.

5.1 Effect of Each Coefficient on the Noise Power in the Observation Band

Using the same system-level parameters of the chip to be fabricated (see Table 6.1), the system was simulated without DAC mismatch, and then the noise power in

the observation band was plotted as each coefficient of the nonlinear function in the digital calibration chain is swept around its nominal value¹. The results are depicted in Fig. 5.1, which also includes the effect on the $\Delta\Sigma$ modulator SNR when sweeping those coefficients.

Fig. 5.1 reveals crucial information in determining an adequate minimization algorithm. First, one can see that C_5 is the coefficient that has the weakest effect on the amount of noise in the observation band. However, its effect on the SNR is not any weaker than the other coefficients. One must remember that C_5 is a multibit digital value representing the implemented value of the most significant bit (MSB) element of the DAC. This element is turned on when the input signal crosses midscale. Therefore, with a sinewave input signal and no offset, mismatch on the DAC MSB element will cause nonlinearity only when the input signal crosses zero. This results in nonlinearity which is strongly correlated with the input signal, giving rise to harmonic distortion. On the other hand, the LSB DAC element constantly changes value (0 or 1) throughout the sinewave cycle, its effect is more similar to adding uncorrelated noise, which increases the noise floor. The frequency spectrum when either C_1 or C_5 is offset from its optimal value is plotted in Fig. 5.2.

The limited sensitivity of the noise power to C_5 indicates that it would be difficult to optimize C_5 if the other coefficients are away from their optimum

¹While this figure shows the results when there is no mismatch and each coefficient is swept around its nominal value, the same results are obtained when mismatch is present and each coefficient is swept around its optimum value.

(a) Effect on the noise power in the observation band, obtained directly using eq. (4.4).

(b) Effect on the SNR

Figure 5.1: Effect of each coefficient on the noise in the observation band and SNR. The nominal value for coefficients is $2^0, 2^1, \dots, 2^4$ for $C_1, C_2, \dots, C_5$.

Figure 5.2: Effect of C_1 and C_5 on the frequency spectrum.

value. Therefore, it has to be optimized last. The order in which the coefficients will be optimized is defined in Section 5.3.

Fig. 5.1 also indicates that the effect on the SNR of each coefficient is in the same vicinity. To obtain an SNR after calibration that is within 2 dB of the maximum achievable SNR, the coefficient resolution is set to 2^{-8} . The coefficient nominal values being from 2^0 to 2^4 and stored using 13-bit words, 2^{-8} represents the LSB of those words. Chapter 6 gives more detailed information on how this resolution was selected.

5.2 The Probability of Error

In order to fully compensate the effect of DAC mismatches, the minimization algorithm needs to be able to change the coefficient being optimized by one LSB and reliably determine if the new value results in a lower amount of noise in the observation band than the previous coefficient value. Since the quantity to be estimated is noise, a random process in nature, there always is a probability that the algorithm takes a wrong decision. This is illustrated in Fig. 5.3 which plots the noise power in the observation band when coefficient C_5 is swept². As indicated on the figure caption, each point is the average of 50 simulation-runs. While the coefficient value that minimizes the noise is clearly identifiable, there is a significant variance for each point (due to the random nature of the process). This is this variance that can cause the minimization algorithm to mistakenly consider one coefficient value as being better than its 1-LSB-neighbor at reducing the noise in the observation band.

What happens when an error occurs? The consequences depend on the particular minimization algorithm being used. The effect could be an increase in calibration time, a lack of precision on the optimum coefficient values found... it could even neutralize an algorithm which was not designed with this eventuality in mind. For the minimization algorithm that is used in this thesis (and

²The curve shown in Fig. 5.3 is strongly dependent on system-level parameters such as A , D and K . The value for those parameters are set in the following chapter so that the focus of the current chapter remains the description of the minimization algorithm.

Figure 5.3: Noise in the observation band while varying coefficient C_5 . Each point is the average of 50 simulation-runs. The confidence interval shows the $\pm 2\sigma$ range, which includes 95% of the results.

described in the following pages), the consequence of an error depends on *when* the error occurs. If the calibration process is at its early stages, when it shuffles the coefficients to find the optimum set, an error will temporarily send the minimization algorithm the wrong way. This will usually be quickly corrected at the next iteration. In the worst case, the coefficient being optimized will be set to a non-optimum value, which will affect the optimization of the other coefficients. This will be eventually corrected in subsequent iterations, but it can mislead the minimization algorithm for a longer time.

Let us define P_e , the probability that given the coefficient values C_{i-} , C_i and C_{i+} which are separated by one LSB, the minimization algorithm makes an error

when trying to sort them in the correct order of the amount of noise they generate in the observation band. It can be shown (see Appendix D), that:

$$P_e \simeq 1 - (Q(\alpha))^2, \quad \alpha = \frac{\mu_{C_i} - \mu_{C_{i+}}}{\sqrt{2\sigma_{C_i}^2}} \quad (5.1)$$

where $\mu_{C_{i+}}$ and μ_{C_i} are the average values of the noise power in the observation band when coefficients C_{i+} and C_i are selected, $\sigma_{C_i}^2$ is the variances of the noise power estimates when C_i is selected, and $Q(x)$ is the normal cumulative distribution function.

The actual values of μ and σ can be obtained analytically, as described in [6]. However, the equations to obtain μ and σ require simulations to set the variable $S_{f,C}^2(f)$, the folded noise power spectral density. In this thesis, values of μ and σ are calculated directly from simulation results. Still, the equations found in [6] are useful to evaluate the effect of several important system-level parameters on the probability of error, and as such, reference [6] is reproduced in Appendix C.

The probability of error is plotted in Fig. 5.4. Those results confirm that C_5 is indeed the most difficult coefficient to set. It also shows that except for C_1 and no mismatch, the probability of error is minimum when coefficients are away from their optimum value. Moreover, the minimum probability of error occurs at a different value for a given coefficient whether the other coefficients are optimized or not. This happens because when the other coefficients are not optimized, the noise floor is raised. Therefore, when the coefficient being optimized is near its optimum value, it is not the main contributor to the noise in the observation

band. It is thus difficult to set it precisely. However, when sufficiently away from its optimum value, it becomes the main noise contributor and changing its value by 1-LSB creates significant differences in the noise power in the observation band. These effects are taken in consideration when designing the minimization algorithm.

5.3 Description of the Algorithm

The proposed minimization algorithm calibrates each coefficient one at a time. Fig. 5.1 a) revealed that coefficients C_1 to C_3 had greater effects on the noise in the observation band than C_4 and C_5 . Therefore, it would be difficult to calibrate C_4 and C_5 if the other coefficients are not yet optimized. Also, Fig. 5.4 indicates that the probability of error of coefficients C_2 and C_3 is generally lower than C_1 . Therefore, the order in which the coefficients are calibrated is: C_2 , C_3 , C_1 , C_4 and C_5 , and then over again.

The minimization algorithm takes advantage of the fact that the probability of error generally occurs when the coefficient being optimized is away from its optimum value (so that it becomes the major noise contributor). It also takes advantage of the fact that the noise power in the observation band is the same whether the coefficient offset from its optimum value is negative or positive. In other words, it takes advantage of the fact that Fig. 5.1 is symmetrical around the optimum point.

The basic idea behind the minimization algorithm is thus to find a point on

(a) Without mismatch.

(b) With mismatch on all coefficients except the one being plotted.

Figure 5.4: Probability of error when setting coefficients. $A = 9$, $K = 2^{16}$, $D = 2^4$, as set in Chapter 6.

the left side of the curve, in the trustworthy region, and then jump to the other side of the curve and find the point whose amplitude is the closest to the one picked on the left side of the curve. Then, the optimum coefficient value is the average coefficient value of both points. As an added benefit, because the average of two points is taken, there is a gain of two in the resolution of the optimum point found. These steps are illustrated in Fig. 5.5 and explained in the following pages. The Matlab code implementing the minimization algorithm is included in Appendix B.

(a) Step one.

(b) Step two.

(c) Step three.

(d) Step four.

Figure 5.5: Main steps of the proposed minimization algorithm.

5.3.1 Step One

The first step is to move the coefficient being optimized, C_i , to the left side of the curve, in a region where the probability of error is low for that coefficient. An offset value, δ_i , is set *a priori* for each coefficient. The first operation of step one is thus to set $C_i = C_i - \delta_i$. The values of δ_i are $\{2, 3, 3, 4, 8\} \times 2^{-7}$ (see Fig. 5.4).

Because the trustworthy region of each coefficient is farther away from the optimum point when the other coefficients are not yet optimized, on the very first iteration (the first time every coefficient is optimized, at startup), the coefficient being optimized is set to $C_i = C_i - 2\delta_i$.

Depending on the mismatch and the outcome of the previous iteration, the coefficient value after this offset may still be on the right side of the curve. This would seriously compromise the minimization algorithm, so precautions are taken to verify that the coefficient is now on the left side of the curve. Two noise estimates when the coefficient value is increased are compared. If the noise increases when C_i increases, the coefficient is either on the right side of the curve or near the bottom of the curve, none of which are good. Should this event occur, the coefficient is offset again by $-\delta_i$ and the verification is redone. Note that the larger offset one the first iteration also helps making sure the coefficient is sent to the left side of the curve.

When the verification is made that the coefficient is now on the left side of the curve, the coordinates are stored in a register. The coefficient value is denoted x_i

and the noise power estimate using this coefficient value is denoted y_l .

5.3.2 Step Two

The second step is to send the coefficient value to the right side of the curve. This is done by applying an offset of $2\delta_i$ ($4\delta_i$ on the first iteration) to the coefficient:

$$C_i = C_i + 2\delta_i.$$

It is safe to verify that this new coefficient is indeed on the right side of the curve. The algorithm checks if the noise power estimates increases when the coefficient value increases. If not, the coefficient value is offset further by δ_i and the verification is redone.

5.3.3 Step Three

The minimization algorithm now needs to find the coefficient value on the right side of the curve that is equivalent to the value found on the left side of the curve. The algorithm searches which coefficient value generates noise estimates that are the closest to y_l .

The algorithm starts by taking three measures of noise for three different values of C_i : C_{i-} , C_i and C_{i+} , which are different by one LSB (the precision required on coefficients). This generates three values for noise estimates: y_{r-} , y_r and y_{r+} . Those three noise estimates are compared to y_l by taking the absolute value of the difference $|y_l - y_r|$. If the smallest difference is from C_{i-} , the new coefficient value is $C_i = C_{i-}$ (the algorithm starts searching towards the left side of the curve),

and three new noise measures are taken³. If the smallest difference is from C_{i+} , the algorithm searches towards the right side of the curve. When the smallest difference is associated with C_i , that means no further improvement is possible by searching in either direction, and the coefficient value of the point on the right side of the curve is set to $x_r = C_i$.

5.3.4 Step Four

The previous steps found coefficient values x_l and x_r from both sides of the curve that generate the same amount of noise in the observation band. Because the curve is symmetrical around the optimum value, the optimized value is the average of both x_l and x_r . The coefficient being optimized is set to $C_i = (x_l + x_r) / 2$ and the next coefficient gets optimized (step one).

5.3.5 Additional Precaution

Step three searches for the ideal calibration value on the right side of the curve. Simulations showed that depending on the noise floor, the algorithm might get lost, especially for coefficients C_4 and C_5 . As a remedy, an additional safety measure is added to the algorithm.

Before going through step three, the algorithm verifies how far right the coefficient value is. If the coefficient value is more than $4\delta_i$ away from x_l (the coefficient value taken on the left side of the curve), it is assumed that the algorithm is lost.

³For faster calibration, the algorithm can reuse the value of y_{r-} and take two new measures of noise (instead of three) after it has moved C_i by one LSB.

The algorithm stops searching, the optimized value is set to $x_l + \delta_i$ (which should be near the coefficient value at the beginning of the iteration) and the next coefficient gets optimized following the pre-established order. The next time the problematic coefficient gets optimized, the noise floor will likely be lower and the coefficient can now be optimized as it should.

5.4 Simulation Results

The system of Chapter 4 was simulated using the minimization algorithm proposed in this chapter. The evolution of noise power in the observation band and SNR throughout calibration is shown in Fig. 5.6 for DAC mismatch of 0.1%, 1% and 5%. These results indicate that the DAC mismatch are compensated when the mismatch is in the order of 0.1% and 1%. When the mismatch is 5% (the DAC element values were 0.95, 2.10, 3.80, 8.40 and 15.20), the algorithm converged to a local minimum which is not the global minimum. As a result, the SNR after calibration is 5.8 dB lower than without mismatch. There is a 36.9 dB improvement over the non-calibrated case, but the DAC mismatches are not fully compensated when the mismatch is in the order of 5%.

5.4.1 Calibration Time

The calibration time is dependent on the severity of the DAC mismatch. Fig. 5.6 reveals that for this example, the calibration was complete (noise was minimized) after respectively one, two and three iterations when the mismatch was 0.1%, 1%

(a) Noise in observation band.

(b) SNR.

Figure 5.6: Noise power in the observation band and SNR throughout calibration process for DAC mismatch of 0.1%, 1% and 5%. An iteration is considered complete when all coefficients have been sequentially set to their optimum value; fractions of an iteration thus represent each individual coefficient being calibrated. The dashed lines show the values for an ideal modulator without mismatch. The input signal is -1 dBFS.

and 5%.

Each iteration requires several estimations of the noise power in the observation band. Each estimation requires $K \times D$ samples from the $\Delta\Sigma$ modulator. The Matlab code implementing the minimization algorithm takes count of the number of times the function that estimates the noise power in the observation band was called. For this particular simulation, with 1% DAC mismatch, the number of estimations after two iterations was 88. Since $K = 2^{16}$ and $D = 2^4$, the total number of samples required for calibration was around 92×10^6 . For example, a $\Delta\Sigma$ modulator running at 100 MHz would be calibrated in approximately 0.92 seconds. Fig. 5.7 shows the relation between calibration time and sampling frequency according to the simulation results.

Tradeoffs between calibration time and hardware complexity are possible. Chapter 9 discusses those tradeoffs.

5.5 Summary

A minimization algorithm that is adapted to the various particularities of the optimization problem was proposed in this chapter. The algorithm takes into account the fact that the objective function is noisy but has a symmetrical shape around the optimum value of each variable. It is a robust algorithm that converged when submitted to different types of DAC mismatch. The number of samples required for calibration depends on the DAC mismatch severity. It was around 92×10^6 samples when the DAC mismatch is in the order of 1%. It should be emphasized

Figure 5.7: Calibration time versus $\Delta\Sigma$ modulator sampling frequency.

that this is the time it takes to calibrate the coefficients at power-up when all coefficients start from their nominal value. Once calibrated, slight changes in the mismatch will be quickly tracked because the noise in the observation band is already low, which means the noise power estimates are reliable (the probability of error is low).

The DAC mismatch effects were not completely compensated for mismatch in the order of 5%. However, as was explained in Chapter 3, current typical CMOS fabrication processes have mismatch in the order of 0.1% (10-bit), meaning mismatch as severe as 5% is unlikely.

The algorithm could be implemented in CMOS using a finite state machine,

registers to store values of various quantities (δ , x_l , y_l , y_r , etc.) and a few mathematical operations (sums and differences).

Chapter 6

System-Level Design

The following chapters describe the design of a $\Delta\Sigma$ modulator IC to report experimental results demonstrating the validity of the proposed compensation technique. Early in the process of designing a $\Delta\Sigma$ modulator, several important system-level parameters must be set. These parameters include the order of the $\Delta\Sigma$ modulator, the number of bits of the quantizer and the OSR. These are usually set to attain a target SNR while optimizing circuit area, power consumption and/or yield. The calibration technique proposed in this thesis requires additional circuitry which calls for extra design choices.

All system-level parameters for the implemented IC are reported in this chapter, along with their respective rationales.

6.1 SNR Target

Usually, the performance targets are set to comply with the requirements of a specific application. Since the proposed compensation technique applies to bandpass $\Delta\Sigma$ modulators with a low OSR (large bandwidth), it makes it very well suited

for digitization of the received signal at the intermediate frequency (IF) in radios. However, no specific communication standard was picked out from the panoply of available standards. Instead, it was decided that the primary objective of fabricating the IC was to demonstrate the validity of the proposed compensation technique. As such, performance targets are set to highlight its advantages. If the implemented modulator exhibits performances that compares it advantageously to other state-of-the-art implementations, the design could be slightly adapted to comply with the requirements of a specific application.

It was explained in Chapter 1 that the accuracy of IC fabrication limits the resolution of multibit $\Delta\Sigma$ modulators to 10-12 bits if no compensation is used. Therefore, a target SNR of less than 74 dB (~ 12 bits) would likely not show the benefits of the proposed compensation technique. On the other hand, achieving significantly more than 12 bits of dynamic range in a bandpass $\Delta\Sigma$ modulator with a low OSR requires a complex system. For example, to achieve a peak SNR of 96 dB (~ 16 bits) at an OSR of 14, a sixth order modulator is required if a 5-bit quantizer is used¹. Designing a sixth order $\Delta\Sigma$ modulator poses serious design challenges and increases the risk of having a non-functional chip. Also, taking in consideration thermal noise, implementing a circuit with such a high dynamic range in a technology with 1.2V supply voltage (see Chapter 7) is a challenge on its own. Finally, as the target SNR of the modulator increases, the requirements

¹Unless otherwise specified, in this chapter, SNR results for a given set of parameters are found using the “delsig” Matlab Toolbox. Note that the results comply with those found in [2].

on the accuracy of the coefficients in the digital calibration chain increases. At the time of the design, there was no certainty that the proposed technique could set the coefficients with enough precision to allow 12-bit resolution when thermal noise and other real-world artifacts would come into play, much less 14 or 16-bit resolution.

For all the above-mentioned reasons, it was decided that aiming for a target SNR of 74 dB was a reasonable choice. Provided that the mismatch in the implemented chip limits the SNR to around 60 dB (that number would change from sample to sample), there is room for 14 dB of SNR improvement using the proposed compensation technique. As a safety measure in case the mismatch was not severe enough to demonstrate the improvements due to the compensation technique, a mechanism to apply intentional mismatch using jumpers was included in the chip (see Chapter 7).

6.2 Number of Quantizer Levels

One advantage of the proposed compensation technique is that it works with binary-weighted DACs. The circuit complexity of the DAC and digital calibration chain increases linearly with the number of quantizer bits, n . Unfortunately, the complexity of flash quantizers increases exponentially with n . Also, the output of flash quantizers is a thermometer code (2^{n-1} bits), so a digital circuit is required to interface a flash quantizer with a binary-weighted DAC. But, if a quantizer such as a successive approximation converter – whose complexity does not increase

exponentially with n and readily outputs binary-weighted codes – is used, the number of bits can be increased to a higher number usually seen in $\Delta\Sigma$ modulators. This would allow achieving high SNRs at low OSRs, which could lead to an overall lower power consumption for a given target SNR and bandwidth.

In order to highlight this advantage, the number of quantizer bits has been set to $n = 5$. While this is more than the majority of reported $\Delta\Sigma$ implementations (see Chapter 3), the complexity should remain manageable.

6.3 Center Frequency of Signal and Observation Bands

The $\Delta\Sigma$ modulator to be implemented will have its signal band centered at $f_s/4$. One advantage of using this frequency is that the digital mixer required to down-shift the converter output signal from $f_s/4$ to DC is easily implemented. The observation band center frequency is placed at DC. Any other choice of frequency than DC for the observation band would require to add a complex pair of zeros to the NTF instead of a single zero at $z = -1$.

6.4 NTF Order and OSR

In order to fulfill the SNR requirement of 74 dB, either a fourth order modulator with an OSR of 11 or a sixth order modulator with an OSR of 7 could be used. Although the second choice would allow operating the modulator at a 35% lower frequency than the first, its complexity does not make it an attractive choice for

a design in which risks must be minimized. A fourth order design was selected.

While achieving 74 dB of SNR using a fourth order modulator with an OSR of 11 and 5 bit of quantization is theoretically possible, this assumes the only noise source is quantization noise. It is more exact to say that this modulator would achieve a signal to *quantization* noise ratio (SQNR) of 74 dB. Obviously the circuits will be subjected to thermal noise. A larger OSR than the theoretical minimum is a safer choice. An OSR of 14 has been chosen. It allows a peak SQNR of 79 dB, which provides a 5 dB margin for thermal noise and the SNR penalty due to the observation band (see next section).

It is worthwhile to note that using an OSR of 14 and a sampling frequency of 280 MHz, the bandwidth is 10 MHz while the signal center frequency is 70 MHz; a common IF for satellite and terrestrial microwave systems. However, the primary objective of fabricating the $\Delta\Sigma$ modulator is only to verify that the proposed compensation technique does work with real-world circuits. Aiming for such a high operating frequency and bandwidth would have significantly increased the risk of having a non-functioning chip, so a lower sampling frequency was picked (see Chapter 7).

6.5 Exact Signal and Noise Transfer Functions

The $\Delta\Sigma$ modulator is implemented using the CRFB structure shown in Fig. 6.1. This structure realizes five NTF poles and zeros, as well as five STF poles and zeros. Using this structure, the NTF and STF can have independently placed

Figure 6.1: Block diagram of the implemented $\Delta\Sigma$ modulator.

zeros, but share the same poles [2].

Four NTF zeros are used to shape the quantization noise away from the signal band, as in a typical $\Delta\Sigma$ modulator. They are placed as two conjugate pairs around the signal band, at frequencies (ω_1 and ω_2) that maximizes the SNR for a given OSR. The fifth NTF zero is placed at DC ($z = 1$) to create the observation band.

Four NTF poles are placed at infinity ($z = 0$). Typically, NTF poles are placed closer to the NTF zeros to reduce the out-of-band NTF gain and thus improve stability [2]. However, for the system under study, extensive simulations showed that stability was not jeopardized by placing four poles at $z = 0$, while it maximizes the peak SNR. The fifth NTF pole, labeled p_5 , plays an important role in the system performances and will be discussed shortly.

The NTF can thus be expressed as:

$$NTF(z) = \frac{(z-1)(z-e^{\pm j\omega_1})(z-e^{\pm j\omega_2})}{z^4(z-p_5)} \quad (6.1)$$

where ω_1 and ω_2 are the notch frequencies around the signal band. They are

found using the `synthesizeNTF` command from the “delsig” toolbox in Matlab (see Appendix A for a description of this function). For a fourth order bandpass modulator with an OSR of 14, $\omega_1 = 2\pi \times 0.238$ and $\omega_2 = 2\pi \times 0.262$.

The STF can be made maximally flat by letting loop filter coefficients $b_{1..5} = 0$. As was explained in Chapter 4, if the input signal contains significant power in the observation band, e.g. if it has a DC component, the calibration performances will be altered. Since there are no guarantees that the signal connected to the $\Delta\Sigma$ converter input is clean in the observation band, it is safe practice to filter the input signal by adding a notch in the STF in the middle of the observation band. This ensures that no interferer from the input signal is present in the observation band. This is done by placing one STF zero at DC. The four remaining STF zeros are placed at $z = 0$ to be canceled by the STF poles, also placed at $z = 0$. The STF is then:

$$STF(z) = \frac{z - 1}{z - p_5} \quad (6.2)$$

This leaves only one remaining NTF and STF pole (shared) to position. As this pole moves towards DC, it tends to cancel the extra NTF zero at DC. As shown in Fig. 6.2, this has three important consequences: 1 – it lowers the NTF magnitude in the signal band (which improves the SNR), 2 – it flattens the STF magnitude in the signal band and 3 – it boosts the NTF magnitude in the observation band (which affects the calibration accuracy and speed).

In most applications, the STF magnitude flatness in the signal bandwidth

Figure 6.2: NTF and STF magnitude response with two different values for the fifth pole p_5 . As p_5 moves towards DC, the characteristics of both transfer functions are more desirable in the signal band, at the expense of increased quantization noise in the observation band.

is a crucial specification. Therefore, the fifth pole p_5 will be positioned at the highest frequency that still meets the application constraints in terms of frequency response slope in the signal band. This slope is obtained by:

$$\Delta STF(p_5) = \frac{|STF(e^{j\omega_{\max}})|^2}{|STF(e^{j\omega_{\min}})|^2} \quad (6.3)$$

where $\omega_{\min}$ and $\omega_{\max}$ are the lower and upper frequency of the signal band. For example, to obtain $\Delta STF \leq 0.1$ dB for the system under study, $p_5 \geq 0.625$ ($f/f_s = 0.075$).

The SNR penalty, ΔSNR , due to the extra NTF zero at DC is obtained by:

$$\Delta SNR(p_5) = \frac{\int_{\omega_{\min}}^{\omega_{\max}} |NTF(e^{j\omega})|^2 d\omega}{\int_{\omega_{\min}}^{\omega_{\max}} |NTF(e^{j\omega})|^2 d\omega |_{p_5 = 1}} \quad (6.4)$$

Using $p_5 = 0.625$, the SNR penalty is $\Delta SNR = 1.6$ dB. Again, lowering the frequency of p_5 would reduce this SNR penalty, but it would increase the amount of quantization noise in the observation band. As a consequence, the lowpass filter cutoff frequency would have to be set to a lower value and the number of samples used for noise power estimation would have to be increased to generate significant estimates. More information can be found in [6], reproduced in Appendix C.

The $\Delta\Sigma$ modulator noise and signal transfer functions are now completely defined:

$$NTF(z) = \frac{(z-1)(z^2 - 0.1295z + 1)(z^2 + 0.1295z + 1)}{z^4(z - 0.625)} \quad (6.5)$$

$$STF(z) = \frac{z-1}{z-0.625} \quad (6.6)$$

Both transfer functions are plotted in Fig. 6.3.

6.6 CRFB Gains

In the last sections, the signal and noise transfer functions were determined. The next step is to transpose those transfer functions to fit the architecture that was chosen to implement the $\Delta\Sigma$ modulator, the CRFB. There are 23 gains to set. There is not a unique solution for the set of gains that will realize the wanted NTF and STF. For any given set of interstage gains $c_{0..4}$, the other gains (a , b and g) can be adjusted to realize the NTF and STF. What changes among the various possibilities is the signal swing at the output of each integrator. Fortunately, the Matlab delsig toolbox provides functions to find those gains for specified NTF and

(a) Logarithmic frequency plot.

(b) Zoom on signal band.

Figure 6.3: Noise and signal transfer functions of the $\Delta\Sigma$ modulator to be implemented.

STF, along with the desired signal swing at each integrator. Using the functions `realizeNTF` and `scaleABCD`, the 23 gains were found. Those values had to be refined during the circuit-level design because some of these gains required capacitor ratios that were impracticable.

The final values of the CRFB gains are given in Table 6.1.

6.7 Simulation

There are numerous places where mistakes can happen when designing a $\Delta\Sigma$ modulator. For instance, simply using a delay-free integrator (with transfer function $z/(z - 1)$) where the architecture called for a delayed integrator (with transfer function $1/(z - 1)$) will significantly change the NTF and have a major impact on the SNR; that is, if the modulator is stable and works at all.

Fig. 6.4 shows the Matlab Simulink model of the $\Delta\Sigma$ modulator to be implemented. Several transient simulations were run to verify its functionality by confirming that the NTF, STF and SNR_{max} were close to the predicted values.

The rigorous reader will have noticed that the interstage gains, $c_{0..4}$ are not included in the model. In fact, they are specified as the gain parameter of each integrator. This parameter is hidden in the printout of the model.

The “Band-Limited White Noise” block models the thermal noise. Its power is adjusted to limit the peak SNR to 74 dB.

The simulation results are reported in Figs. 6.5-6.7. The peak SNR, at 74.2 dB, occurs at an input amplitude of -0.4 dBFS. Without thermal noise, the peak

Figure 6.4: Simulink model of the $\Delta\Sigma$ modulator.

SQNR is 76.7 dB. The noise and signal transfer functions are as predicted.

6.8 Parameters for the Digital Calibration Chain

There are four parameters to set in the digital calibration chain: the precision of stored coefficients $C_{1..5}$, the lowpass filter cutoff frequency, the lowpass filter decimation factor and the number of samples used by the power estimator. Most of the parameters to set in the digital calibration chain are chosen to minimize the probability of error, which is described next.

6.8.1 The Probability of Error

The concept of probability of error was introduced in Section 5.2. Because the minimization algorithm works towards minimization of noise (caused by DAC mismatch), the objective function has a random component. Moreover, there are

Figure 6.5: Frequency spectrum of the Simulink model. The input signal is a sine wave at -1 dBFS.

Figure 6.6: Frequency spectrum of the Simulink model showing the STF. The input signal is frequency-flat noise at -66.4 dB. The quantizer is bypassed to remove quantizations noise.

Figure 6.7: Simulation results showing the SNR vs input amplitude for the Simulink model.

other noise sources (quantization noise and thermal noise) that act as interferers. This means that even if the coefficients are kept constant, two successive noise power estimates will not have the same value. This also means that if the minimization algorithm sweeps one coefficient, the randomness may cause errors when deciding if this new coefficient value is better than the previous one.

Using the minimization algorithm described in Chapter 5, when an error occurs in the early stages of calibration, the consequence is a more or less severe augmentation of the overall calibration time. On the other hand, if an error occurs when calibration is considered complete and the $\Delta\Sigma$ modulator is in normal operation, if that error occurs at the wrong time, a coefficient will be set to a non-optimal value. As will be seen in the next subsection, this can have a substantial

impact on the modulator SNR until that error is corrected.

6.8.2 Coefficient Resolution

Each coefficient C_i in the digital calibration chain is stored using m bits. While increasing m increases chip area, perhaps more importantly it also increases the calibration time. As m increases, changing one coefficient by one LSB results in gradually smaller effects on the noise power in the observation band. In order to detect a change of one LSB on the coefficient being optimized, more samples have to be crunched by the power estimator which results in longer calibration times. Therefore, m must be chosen just large enough for enough precision to attain the target SNR, but no more.

Fig. 6.8 shows the $\Delta\Sigma$ modulator SNR sensitivity to coefficient offsets. A quick look at this figure reveals that the coefficients that have the most drastic effect on the SNR are C_3 and C_4 . Varying any of these two coefficients by 2^{-8} from their ideal value drops the SNR_{max} by 1.8 dB, to 72.2 dB. The finite coefficient resolution has an effect which is similar to quantization noise : the worst error happens when the desired signal is halfway between two quantization steps. For example, if the mismatch on the DAC is such that the optimal value of C_3 is 3×2^{-9} (which would result in a SNR of 74 dB) and the maximum resolution of coefficients is set to 2^{-8} , the minimization algorithm will set C_3 to either 2×2^{-9} or 4×2^{-9} . This results in an error of approximately 1 dB in either case.

The penalty due to coefficient precision adds-up for each coefficient, but C_1 ,

(a) Offsets from -15 to 15 ($\times 2^{-7}$).

(b) Zoom on offsets from -3 to 3 ($\times 2^{-7}$).

Figure 6.8: Simulation results showing the SNR versus coefficient offsets. The nominal coefficient values are $\{1, 2, 4, 8, 16\}$. The simulation includes thermal noise limiting $\text{SNR}_{\max}$ to 74 dB.

C_2 and C_5 are less sensitive, which means that if the precision on coefficients is 2^{-8} , the peak SNR when calibration is complete can be anywhere between around 72 and 74 dB, depending on the mismatch of the DAC elements on the chip.

Although better performances could be achieved using a resolution of 2^{-9} , a larger number of samples is required for each power estimation. Therefore, the resolution of the coefficients has been set to 2^{-8} , meaning that the calibration algorithm will allow a peak SNR that is within 2 dB of an ideal modulator without mismatch. The nominal coefficient values are 2^0 to 2^4 . The coefficients will thus be stored using $m = 13$ bits.

Since the minimization algorithm proposed in Chapter 5 takes the average of two coefficient values (with equal offsets on each side of the optimum), there is a gain of two in the resolution of the coefficient being optimized. Therefore, the minimization algorithm needs to adequately sort two adjacent coefficient values with a precision of 2^{-7} .

6.8.3 Lowpass Filter Cutoff Frequency

The lowpass decimation filter determines the observation band. Its value must be set to optimize the significance of each noise power estimate, taking in consideration those two factors:

- The noise in the observation band due to DAC mismatch is relatively flat while the quantization noise increases with the frequency. The quantization

noise being independent of the DAC mismatches, it is considered an interferer. Widening the observation band lets more quantization noise in, which blurs the noise power estimates.

- For flat noise with a given variance, reducing the bandwidth is equivalent to reducing the number of samples used to estimate its variance.

There is an optimal value for the lowpass filter cutoff frequency that maximizes the observation band while not letting too much quantization noise in. Intuitively, the cutoff frequency should be around the point where the flat noise due to DAC mismatch meets the quantization noise PSD.

For its hardware-efficient realization, the following lowpass transfer function was favored (see Chapter 4):

$$T(z) = \left(\frac{2^{-A}}{z - (1 - 2^{-A})} \right)^2 \quad (6.7)$$

It is a second order lowpass filter with two poles at $z = 1 - 2^{-A}$. It can be realized by cascading two identical cells $\sqrt{T(z)}$. By restricting A to integers, the gain can be implemented by a simple bit shift operation and no multiplier is required.

Fig. 6.9 shows the noise power in the observation band when C_1 is swept around its nominal value for two different values of A . This figure puts in evidence the effect of A on the confidence interval of each point. Using $A = 9$, the variance

(a) $A = 6$. (b) $A = 9$

Figure 6.9: Noise in the observation band while varying C_1 . Each point is the average of 50 simulation-runs. The confidence interval shows the $\pm 2\sigma$ range, which includes 95% of the results.

of the estimates is lower when near the optimum value. The variance of the estimates is higher when away from the optimum value. Using $A = 6$ is the other way around.

Inspecting Fig. 6.9 alone is not sufficient to determine the optimal value of A . Another way of looking at this is by inspecting P_e , the probability of error.

Fig. 6.10 shows the probability of error for different values of A (which sets the lowpass filter cutoff frequency). The first observation is that the probability of error is generally much lower when calibrating C_1 than C_5 . As explained in Chapter 5, C_5 has a lesser impact on the noise power in the observation band than the other coefficients. Unfortunately, it does not have a lesser impact on the SNR, so it has to be set with the same accuracy.

The second observation from Fig. 6.10 is that as A decreases (the cutoff frequency increases), the probability of error is low over a wider range of coefficient offsets. Also, Fig. 6.10 indicates that the ideal value of A might be different for C_1 and C_5 . In this work, the value of A remains constant throughout calibration, so a single value of A must be picked. Since in general, C_5 shows a higher probability of error than any other coefficients (see Chapter 5), A is set optimally for C_5 .

Taking in consideration the above-mentioned observations, A was set to 9. The probability of error for C_5 and $A = 9$ has the lowest minimum, and is lower than the other curves for a wide range of coefficient offsets².

²However, using $A = 8, 9$ or 10 wouldn't be terribly bad choices either, as long as the minimization algorithm is set to work in the area where the probability of error is minimum for a given value of A .

(a) C_1 . (b) C_5 .

Figure 6.10: Probability of error for C_1 and C_5 for different values of A . The number of samples for noise power estimation is 2^{20} ($K = 2^{16}$, $D = 2^4$).

6.8.4 Lowpass filter order

The lowpass filter order has been set to 2 because it was found [6] that the probability of error did not significantly improve with higher order filters.

6.8.5 D , the Lowpass filter decimation factor

Using $A = 9$ results in a pole frequency of $3.1 \times 10^{-4} f_s$ in the lowpass filter. At the center of the signal band, $f_s/4$, the filter attenuation is 114 dB. Because there is no significant power at the output of the lowpass filter at higher frequencies, the signal can be downsampled before it is sent to the noise power estimator, with little or no effect on the estimates. This allows running the power estimator at a lower frequency and simplifies its design since the integrate and dump requires less bits to quantify the signals.

The adequate decimation factor can be determined by analyzing its impact on the probability of error. It was found that using a value of $D = 2^4$ had little to no impact when compared to no decimation ($D = 1$); both curves were indistinguishable. It is very likely that a larger decimation factor can be used without interfering with the calibration process. Nonetheless, a decimation factor of 16 is selected as a prudent choice.

6.8.6 K , the number of samples for estimation

The parameter K , which dictates how many samples from the lowpass filter are used to estimate the noise power in the observation band, has a strong influence

on the probability of error. Obviously, increasing K increases the precision of each estimation which in turns reduces the probability of error. The main drawback of increasing K is that it takes longer to obtain an estimate.

As was stated in section 5.2, one of the consequences of an error event is that the minimization algorithm is temporarily sent in the wrong search direction which increases calibration time. So on one side, setting K to a low value increases the rate at which estimates are available, but on the other side, the reliability of those estimates is lower which results in errors and ultimately a longer calibration time. The values of K must thus be set as a compromise to minimize the overall calibration time.

Setting K to the value that minimizes the overall calibration time is a painstaking job. First, a cost cannot be attributed to an error event because the impact of an error event on the calibration time depends on when this error occurs, on which coefficient, on the severity of the DAC mismatch, etc. The value of K cannot be found analytically; simulations are required. When K gets higher and the probability of error gets arbitrarily low, the number of error events changes from one simulation to another even if nothing is changed. Therefore, for a given value of K , the calibration time must be the average of several simulations, each of which takes considerable CPU time.

Fortunately, the value of K can be made a configurable parameter. The word lengths in the power estimator must be made large enough to avoid overflow

when the highest forecasted value of K is used. By inspecting Fig. 6.10, which was obtained using 2^{20} samples ($K = 2^{16}$, when taking the decimation factor of 16 in consideration), intuitively, K cannot be much lower than 2^{16} . On the other hand, using this value, the toughest coefficient to optimize, C_5 , can be found with a probability of error of less than 10^{-3} when all the other coefficients are optimized.

K was thus set to 2^{16} , with room to grow up to $K = 2^{20}$.

6.8.7 Adaptive Parameters

In this chapter, parameters were considered fixed values throughout the calibration process. The overall calibration time and the final accuracy (and SNR) can be optimized by adapting several parameters as the calibration progresses. For example, at the beginning, the precision on coefficients can be made less stringent than 2^{-8} , use a lower value of K and (optionally) increase the observation band just to set each coefficient near its optimum value and quickly lower the noise floor in the observation band. As the calibration process advances, those values can be adjusted to favor accuracy instead of speed.

While this has not been done in this thesis, it is an interesting topic to investigate to further increase the performance of the proposed compensation technique.

Parameter	Value
Order	$L = 4$
Number of Quantizer Bits	$n = 5$
Oversampling Ratio	$OSR = 14$
Noise Transfer Function	$NTF(z) = \frac{(z-1)(z^2-0.1295z+1)(z^2+0.1295z+1)}{z^4(z-0.625)}$
Signal Transfer Function	$STF(z) = \frac{z-1}{z-0.625}$
Maximum SQNR	$SQNR_{max} = 76.7 \text{ dB}$
Target SNR	$SNR_{max} = 74.0 \text{ dB}$
CRFB Gains	$a_{0..4} = \{1.333, -2.230, -1.127, -1.513, 0.387\}$ $b_{0..5} = \{0, -1.989, -1.553, -1.910, 0, 0.853\}$ $c_{0..4} = \{0.500, 0.898, 1.200, 0.540, 0.833\}$ $g_{0..1} = \{2.050, 3.997\}$
Coefficient precision	2^{-8} (stored using $m = 13$ bits)
Lowpass filter cutoff frequency	$3.1 \times 10^{-4} f_s$ (using $A = 9$)
Decimation factor	$D = 2^4$
Number of samples for noise power estimation	$K = 2^{16}$ (configurable parameter to be fine tuned later)

Table 6.1: Summary of $\Delta\Sigma$ modulator System-Level Parameters.

6.9 Summary

This section went through all the parameters that define the $\Delta\Sigma$ modulator to be designed at the circuit-level in the next chapter. Explanations were given for setting each parameter to its value. Table 6.1 summarizes those parameters.

First, the decision was made that the primary objective of fabricating the chip was to demonstrate that the technique to compensate DAC mismatches in multibit $\Delta\Sigma$ modulators proposed in this thesis can be applied successfully to real-world circuits. This chip being the first IC realization of the author, it seemed

important not to be caught by an excess of enthusiasm and try to achieve stellar performances on every aspect of the design.

A fourth order bandpass $\Delta\Sigma$ modulator architecture with an OSR of 14 and 5 bits of quantization was picked. Simulation results confirm that this architecture can achieve a peak SQNR of 77 dB. The target is to attain a peak SNR of 74 dB once thermal noise is factored in. By comparison, a 1-bit fourth order $\Delta\Sigma$ modulator would require an OSR of 64 to achieve the same SQNR performances.

The methodology to position each pole and zero for the noise and signal transfer functions was described. The creation of the observation band (which is essential for the compensation technique) also adds a pole to the NTF and STF. This additional pole was set so that the slope of STF gain in the signal band is 0.1 dB.

Finally, several parameters that are particular to the proposed compensation technique were also set. The effect of each parameter was thoroughly analyzed and ultimately, they were set to values that seem to offer the best compromise between calibration accuracy and speed.

Chapter 7

IC and PCB Design

No matter how realistic simulations get, the most convincing method of proving the concept of a novel idea in the field of microelectronics is still to implement it in silicon. An integrated circuit (IC) chip was designed and fabricated for this purpose. In Chapter 6, the system-level parameters of this chip were set. The circuit-level design of the chip is covered in this chapter, as well as the design of the printed circuit board (PCB) to support it.

The IC contains the fourth-order $\Delta\Sigma$ modulator with the additional integrator to create the observation band, i.e. it realizes the system shown in Fig. 6.1. The digital calibration chain is not included in the chip to allow modifications to the various parameters; the operations of the digital calibration chain are done in Matlab. This decision is discussed in more depth in the next chapter.

There was no intention to bring any novelty to the circuit design of the $\Delta\Sigma$ modulator. The standard techniques found in textbooks [2, 11, 58] were applied and verified by simulation.

Cadence (ICFB5141) was used to do schematics, simulations and layout.

7.1 Choice of Technology

The chip was fabricated using the services of the Canadian Microelectronics Corporation (CMC). At the time of the design (first half of 2007), the mainstream technology offered by CMC was CMOS 0.18 μm , but 0.13 μm was emerging to replace it shortly as CMC's mainstream technology. CMC's 0.13 μm CMOS process was selected to implement the chip. That decision was mostly based on the research team's past successes using this process and the good experience they had using the design kit.

The 0.13 μm process offered by the CMC is IBM's 8RF-DM CMOS process manufactured through MOSIS services. It is an 8-layer metal structure with nominal 1.2 V core and 2.5 V I/O supply voltages [59]. However, it allows up to 1.6 V power supply voltage (V_{dd}) operation when using thin oxide field effect transistors (FET) [60]. It also supports Metal-to-Metal (MIM) capacitors.

7.2 Continuous or Discrete-Time

There are several advantages to continuous-time $\Delta\Sigma$ modulators. For instance, they inherently provide anti-alias filtering and they generally allow higher clock frequencies [8]. However, the exact realization of the noise and signal transfer functions usually rely on the precision of gm - C cells, which is typically less than

the precision of capacitor ratios in switched-capacitor designs. Also, continuous-time $\Delta\Sigma$ modulators are more affected than their discrete-time counterparts by the DAC pulse shape, excess loop delay and clock jitter. There are documented techniques to address those issues, but for the purpose of this implementation, it was safer to go with the more common discrete-time switched-capacitor architecture.

7.3 Differential Signaling

Fully differential signals are used everywhere in the signal path in the implemented chip. Differential signaling rejects common-mode noise, cancels second-order distortion and increases dynamic range [11]; all of which are welcome especially when the supply voltage is low. Also, to implement a negative value coefficient, the two signal phases only need to be swapped. The main drawback is increased circuit complexity, mostly because a common-mode feedback (CMFB) circuit must be used to force the common-mode voltage close to $V_{dd}/2$.

7.4 Sampling Frequency

No sampling frequency specification has been set. To the author's point of view, as long as the fabricated IC can demonstrate that the novel compensation technique proposed in this thesis can be implemented successfully, the goal will be considered achieved. The technique being digital, it would certainly work similarly at higher frequencies as long as the digital circuits are designed using the corresponding

clock constraint.

The IC was designed and simulated using a sampling frequency of 10 MHz.

7.5 Thermal Noise and Capacitor Sizing

The modulator architecture yields a peak SQNR of 76.7 dB. To attain the 74 dB target peak SNR of the $\Delta\Sigma$ modulator, the thermal noise power at the output of the modulator should be approximately -77 dBFS. The relative value of capacitors in switched-capacitor circuits determines the gain factor of each signal path. The absolute value of capacitors, on the other hand, limits the bandwidth of the signal and noise. The value of capacitors is set to comply with the noise power specification, and then other parameters like bias currents and transistor sizes are set to comply with speed requirements.

There are many thermal noise sources in the circuit. It is well known [1] that the first stage in a $\Delta\Sigma$ modulator is the most important contributor to the noise power at the modulator output. However, in this case, the first stage is the integrator that creates the observation band. It does not have a high gain in the signal band. Therefore, the first resonator is as important as the first integrator for the noise perspective. The noise introduced by the sample and hold block also contributes directly to the noise at the modulator output.

Typically, the noise power of a sampled circuit is kT/C [11], with $kT = 4.1 \times 10^{-21}$ at room temperature. That noise is spread from DC to $f_s/2$. The noise

that contributes to the noise power in the signal band is thus:

$$P_N = \frac{kT}{C \cdot OSR} \quad (7.1)$$

For a noise power of -77 dBFS, a peak-to-peak signal amplitude of 600 mV at 0 dBFS, the noise power is:

$$P_N = \left(\frac{0.3}{\sqrt{2}} \right)^2 \times 10^{-77/10} \quad (7.2)$$

For an OSR of 14, this translates into a capacitor value of 0.3 pF. Of course, this quick calculation only gives a rough estimate of the capacitor sizes. For one, there will be many noise sources that add-up. Also, the peak-to-peak signal amplitude of 600 mV was an intuitive choice. A signal amplitude which is too large will generate harmonic distortion. The calculation also does not take into account the fact that differential circuits are used, which adds 3 dB to the SNR. Simulations will be required to refine the value of capacitors. Nonetheless, 0.3 pF is a reasonable capacitor value to implement in an IC.

7.6 Circuit Overview

Fig. 7.1 shows the block diagram of the IC. Each block is described in this section, along with schematics and simulation results of the most important blocks. The complete schematics of the IC are available in Appendix E.

The words in `type writer font` indicate the name of the blocks in the schematics.

Figure 7.1: Block diagram of the circuit-level schematics.

7.6.1 Fully Differential Amplifier (diffamp)

The whole system requires several operational amplifiers (13 in total). Because these amplifiers are at the heart of almost every block described in the next pages, they are presented first.

The amplifiers used in this design need to be fully differential. The folded-cascode architecture was picked due to its high performances with limited power supply voltages and capacitive loads. The schematic of the differential amplifier that was implemented in the IC is shown in Fig. 7.2. Even though requirements (linearity, speed, gain) are not as stringent for each of the 13 instances of the amplifier, they are identical copies, for simplicity.

The length of each transistor is 360 nm, which is three times the minimum

Figure 7.2: Schematic of the fully-differential amplifier.

length; a common approach in analog IC design. The value for bias currents were initially selected so that the output voltage had time to settle within half a clock period when a 1 pF capacitive load was switched. The width of the transistors were set so that they would place the various transistors in saturation. Lots of simulations were required to refine those values.

The CMFB block sets the gate voltage of M_8 and M_9 to keep the common mode voltage to a predefined value. The CMFB circuit is the simple switched-capacitor CMFB circuit from [11]. Signals V_{b1} , V_{b3} , V_{b4} , V_{bcmfb} , V_{com} are bias voltages that are set off-chip (see Section 7.6.3).

“Clamp” transistors are sometimes used to improve the slew rate performances

of the folded-cascode amplifier [11]. Those transistors, though not shown in Fig. 7.2 were included in the design. However, an error was made when connecting them: instead of using diode-connected NFETs between the gate and the source of Q_3 and Q_4 , diode-connected PFETs were connected between the source and the drain of Q_3 and Q_4 (see Appendix E). Those PFETs have their gates connected to V_{dd} and are thus turned off all the time. That mistake was flagged after tapeout. Fortunately, other than consuming area and adding load capacitance, those useless transistors do not cause any major problem.

7.6.2 Clock buffers (clkbuf)

The switched-capacitor circuits in the $\Delta\Sigma$ modulator require two non-overlapping clock signals ϕ_1 and ϕ_2 . Those two clock signals are generated off-chip by an FPGA (see Section 7.11).

It is important that the edges of the clock signals are steep; otherwise, sampling errors can occur [11]. The `clkbuf` block consists of two chains (one for each clock input signal) of three inverters each, of increasing W/L (width over length) ratios. The quantizer clock signal is the same as the ϕ_2 clock signal. To make sure the load on ϕ_1 and ϕ_2 is the same, an additional inverter was included to drive the quantizer, hence the third output `quant_clk`.

7.6.3 Bias circuit (bias)

No advanced biasing circuits were included in the chip.

It was found through simulations that the performances were strongly dependent on the biasing voltages fed to the various differential amplifiers. So instead of using scaled current mirrors to extrapolate the various bias voltages from a bandgap circuit or a single external current source, the decision was taken to use several independent external voltage or current sources. While this pushes the complexity towards the PCB and requires more equipment, it allows setting bias voltages independently to get the best performances out of the chip. The block `bias` is mainly 1:1 current mirrors to set the bias voltages for a given bias current generated from an off-chip current source.

7.6.4 Sample and hold (`sah_3phases`)

The input signal to the $\Delta\Sigma$ modulator is a continuous differential signal. The `sah_3phases` block samples this continuous input signal using the clock signals ϕ_1 and ϕ_2 . Some blocks in the modulator need the signal on the ϕ_1 phase while some others require it on the ϕ_2 phase. Also, the summing node before the quantizer needs the input signal with a full clock cycle in advance (see Fig. 6.1). The sample and hold block outputs three (differential) signals, all of them being sampled versions of the modulator input signal delayed by half a clock signal each.

The circuit for the sample and hold is shown in Fig. 7.3. It was picked for its simplicity and accuracy [11]. Its main weakness is its speed; during the sampling phase (ϕ_1), the input and output of the amplifier are connected together which means the output is held close to zero. When switching to the holding phase (ϕ_2),

Figure 7.3: Schematic of the sample and hold.

the amplifier must resolve the sampled voltage which poses big requirements on the amplifier slew rate for fast operation (which was not the case).

Transient simulation results are shown in Fig. 7.4. As stated above, the sample and hold must provide the same samples on three consecutive clock phases. To do that, the same circuit is instantiated three times with interchanged clock phases between stages.

7.6.5 Integrator (integrator_no_delayx2)

This block is a non-delaying switched-capacitor integrator. It implements the first integrator in Fig. 6.1, i.e. the NTF zero at DC to create the observation band. Since $b_0 = 0$, it has only one input, coming from the DAC.

The schematic for the integrator is shown in Fig. 7.5. It samples when ϕ_1 is active, and the output is available at the end of ϕ_1 and during all of ϕ_2 . The

Figure 7.4: Transient simulation results for the sample and hold.

capacitor ratio C_1/C_2 implements the coefficient $a_0 = 1.333$. Transient simulation results (Fig. 7.6) confirm that the integrator performs as expected.

7.6.6 Resonators (resonator1x2 and resonator2x2)

There are two resonators in the $\Delta\Sigma$ modulator. Each resonator is composed of two integrators with feedback from the output of the second integrator to the input of the first. Both resonators are essentially identical, except for the values of capacitors implementing the various gains a , b , c and g . Also, because $b_4 = 0$, there is one less path from the sample and hold block to the second resonator.

The resonators are built using the same circuit architecture as the integrator shown in Fig. 7.5. The inputs are summed at the amplifier input simply by connecting them in parallel much in the same fashion as in standard summing circuits using opamps. When a delaying integrator is required, the clock phases of

Figure 7.5: Schematic of the integrator.

Figure 7.6: Transient simulation results for the integrator. The input signal is from the DAC output, which is valid during ϕ_1 only.

the switches are rearranged as in [11]. The circuits, which tend to get cumbersome, is not included in this chapter but can be found in Appendix E.

7.6.7 Summing Node (sum2x2)

This block implements the summing node before the quantizer. It adds the signal from the output of the second resonator to the input signal. It is a switched-capacitor adder without integration and proper capacitor ratios to realize c_4 and b_5 .

7.6.8 Quantizer (adc_5b_no_ff)

The quantizer is a 5-bit flash ADC. There are two reference voltages. They are connected to pins to be set externally from voltage sources. It has a string of 31 identical resistors in series to set the 31 quantization step voltages which are connected to 31 comparators. The quantizer block also includes a thermometer-code to binary-code converter, implemented using standard NAND and NOR gates.

7.6.9 DAC (dac_5b)

The DAC converts the digital signal from the quantizer back to analog form. It is basically composed of binary-scaled current mirrors. The output current is converted to voltage via resistors and buffers.

The simplified schematic is shown in Fig. 7.7. The current sources are implemented using cascode transistors (PFETs for the I_{op} branch and NFETs for the

Figure 7.7: Schematic of the DAC.

I_{on} branch). One drawback of this architecture is that when a current source is disabled, the output transistor drain is disconnected and left open. When that current source is re-enabled, the current flowing through it must step from zero to its nominal value. This limits the operating frequency of the DAC. A better design would deviate the current to a dummy load when the controlling bit is zero. Although the speed was not a primary concern in this design, it was found through simulation that the speed limitation of the DAC quickly degraded the performances of the modulator and a second version of the chip would certainly include a better DAC.

The transient response of the DAC to a ramp is shown in Fig. 7.8. Those simulation results show the spikes due to the behavior explained in the previous paragraph.

The entire thesis is devoted to a DAC mismatch compensation technique. The

Figure 7.8: Transient simulation results for the DAC. The input signal is a voltage ramp fed to the quantizer, creating a digital ramp for the DAC.

mismatch is due to the limited precision of the IC fabrication technology. The mismatch will vary from one chip sample to the other. It is thus possible that the chip being tested has very small mismatch, and that the compensation technique is of no use for that particular sample. Testing another chip involves several meticulous tasks (bonding, soldering). So instead of getting in a quest for finding a chip sample with large mismatch to show that the compensation technique works in such conditions, a mechanism to apply intentional mismatch was included in the DAC. That mechanism is shown in Fig. 7.9.

A current source which is 1% the nominal value of the element is put in parallel with the main source to offset its value. The mismatch current source is enabled

Figure 7.9: Intentional mismatch.

by a binary control signal which is tied to a pin to be controlled externally via a jumper. The mechanism has been included for the LSB and MSB elements of the DAC.

7.7 Design Iterations and Noise Problems

Each block was refined through transient simulations including thermal noise. The bias currents, transistor and capacitor sizes were modified to get low harmonic distortion (< 80 dB) and good SNR (~ 80 dB). Those values had to be refined again when all the blocks were connected together to form the $\Delta\Sigma$ modulator.

There was a major problem with thermal noise. Achieving 74 dB SNR for the entire system turned out to be a much greater challenge than initially anticipated. The switches (NFETs) were substantial noise sources. To lower noise, larger capacitor values were used. This caused settling time problems, so bias and transistor sizes had to be modified. This, in turn, often increased noise. A design

solution could not be found in time for the fabrication deadline¹.

The following section presents the simulation results of the design submitted for fabrication and discusses the implications of not complying with the SNR target.

7.8 Simulation Results for the Complete System

The frequency spectrum of the transient simulation results for the complete system are shown in Fig 7.10. The sampling frequency is 10 MHz. The input signal is 380 mV differential which is 1 dB lower than the full-scale quantizer range. The accuracy of the simulator is set to conservative, the highest setting. Thermal noise is enabled, but no Monte Carlo statistical deviation is enabled.

The simulation results indicate that the NTF is close to the exact equation, but not exactly the same. The center frequency is not $f/f_s=0.25$, but around $f/f_s=0.24$. The problem is probably caused by an insufficient gain in the differential amplifiers which makes the effective gain factors (a , b , c and g) of the switched-capacitor integrators different from their nominal values. This problem is discussed in more detail in Chapter 8, but it is worthwhile to note that the SNR

¹The author was caught in a (very frustrating) loop where he kept increasing the capacitor values, play with bias, play with transistor sizes, adjust the quantizer step size, reduce sampling frequency, increase supply voltage (...) to attain the 74 dB target. For each modification, transient simulations had to be run. Given the complexity of the system, these simulations took days to run, even on a very-high performance Linux box. And sometimes, what seemed like a favorable change which caused a 1-2 dB SNR improvement was just due to the variance of the SNR measure resulting from the limited number of samples.

At the same time, the deadline for submitting the design to CMC for chip fabrication kept approaching so at some point, this design loop had to stop and layout had to begin. Unfortunately, the 74 dB SNR goal could not be reached, even using capacitors in the 10-20 pF range.

Figure 7.10: Circuit-level simulation results. The input signal amplitude is -1 dBFS. The input signal frequency is outside the signal band to get a better view of the noise in the signal band since the number of samples is limited (2000 samples).

penalty caused by this frequency shift of the signal band is cancelled by properly adjusting the center frequency parameter when calling the function to calculate the SNR.

The SNR of the system is approximately² 66.8 dB using an OSR of 14 and a center frequency of $f/f_s = 0.241$. While 67 dB is 7 dB short of the SNR objective for the system, the fabrication of the IC is not in vain. First, there was still hope that the performances could improve using the fabricated chip by tweaking the bias voltages. Measuring the effect of changing bias voltages on SNR could be

²The number of samples used for SNR calculation is 2000. Using 2000 samples gives a standard deviation of 0.7 dB, which means this SNR estimation is within ± 1.4 dB of the actual SNR value 95% of the time.

done in a few seconds instead of a few days (unfortunately, no improvement could be done this way as will be discussed in the next chapter). Also, the compensation technique could still be validated using the fabricated chip. Assuming typical DAC mismatch, the SNR would be limited to approximately 60 dB, leaving room for 7 dB improvement using the proposed DAC mismatch compensation technique.

7.9 IC Layout

The area allocated for the chip die by CMC is 2 mm^2 . While this seemed like more than enough at first, it turned out to be a challenging task to fit the circuit within that space, especially given the large capacitor values used abundantly in the circuit.

MIM capacitors were used to implement every switched capacitor because of the linearity requirements. MOSCAP (an NFET with source and drains shorted) were used for on-chip power supply decoupling capacitance.

Whenever possible, resistors, capacitors and transistors were laid out using the interdigitated technique [58] to increase matching between the two differential pairs. Dummy elements were also consistently used to increase matching of the border elements. Extra care was taken to ensure the metal wires of the differential pairs had matched length and resistance.

The layout of the circuit is reproduced in Fig. 7.11, mostly to show the relative size of each laid out block. The size, including bond pads is $1 \text{ mm} \times 2 \text{ mm}$. Without bond pads, the circuit is approximately $1.6 \text{ mm} \times 0.6 \text{ mm}$ (0.96 mm^2).

Figure 7.11: Layout of the IC with overlay showing the major blocks.

The circuit has 34 bond pads. The purpose of each pin is found in Table 7.1 at the end of the chapter.

7.10 Chip Bonding and Packaging

The fabricated chip was bonded to a MLP6X6-40-0P-01 package bought from the company Sempac (www.sempac.com). It is a 40 pin QFN (quad flat no-lead) open-cavity plastic package measuring $6\text{ mm} \times 6\text{ mm} \times 0.8\text{ mm}$. The lead pitch is 0.5 mm and the die pad size is $4.24\text{ mm} \times 4.24\text{ mm}$.

7.11 PCB Design

The PCB serves several purposes: it holds the UUT (unit under test, i.e. the $\Delta\Sigma$ modulator), it converts the single-ended input signal to differential signals, it provides bias and supply voltages to the various active components, it adjusts

Figure 7.12: Block diagram of the PCB and external components.

digital signal levels for communication with the FPGA, etc.

Fig 7.12 shows how the main blocks of the PCB are connected together. The complete schematics are found in Appendix E. There is a brief description of each block in the next pages.

The $\Delta\Sigma$ modulator is the unit under test: the fabricated chip implementing the $\Delta\Sigma$ modulator described in this chapter.

The single-ended to differential converter is a circuit built around an Analog Device AD8132 differential amplifier with its negative input tied to ground, as recommended in its datasheet. One of the target application of the AD8132AR is single-ended to differential conversion for ADCs.

Two 4-channel 16-bit DACs (Texas Instruments DAC8565) are used to provide bias voltages to the $\Delta\Sigma$ modulator chip. The DACs are controlled using a serial interface by the FPGA. The desired DAC output values can be changed from

a Matlab command on a PC which communicates with the FPGA. This allows optimizing the bias voltages to maximize the SNR at the $\Delta\Sigma$ modulator output.

The FPGA block is a Xilinx XC2VP100 mounted on a PCI card (AP1000 FPGA Development Board from Amirix) inserted in a PC. Initially, the main purpose of the FPGA was to implement the digital calibration chain. It was later decided that the digital calibration chain could be implemented in Matlab for reasons discussed in Chapter 8. The FPGA generates the two clock phases for the $\Delta\Sigma$ modulator derived from the external clock source and also manages the communication with the bias DACs.

The buffers translates the digital signal levels between the $\Delta\Sigma$ modulator and the FPGA. They also isolate both components and make sure the long cable connecting the FPGA to the PCB is driven with enough strength.

Several jumpers define the operating mode of the modulator. They are symbolized by the block called controls. For instance, this block includes the jumpers to enable the intentional mismatch on the DAC elements.

Not shown in the block diagram are the decoupling capacitors. One 0.1 μF surface-mount (0603) capacitor was placed as close as possible to each supply pin (including bias pins) for local decoupling. One 10 μF capacitor was placed near each integrated circuit, and two 220 μF capacitors were used for global decoupling. Ferrites in series with the supply voltage inputs were also used for enhanced filtering.

Figure 7.13: Top and bottom layers of the PCB (actual size).

The PCB measures 3 inches $\times$ 3 inches. It has four layers: the top and bottom are used for signal routing, the middle layers are planes for supplies. The minimum trace width and the minimum clearance are both 7 mil. The printout of the PCB showing the top and bottom layers is found in Fig. 7.13.

7.12 Summary

The chapter outlines the design of the IC that was fabricated to verify that the compensation technique described in Chapter 4 can be applied successfully to a real circuit.

The chip to be fabricated implements the $\Delta\Sigma$ modulator only. The digital calibration is done off-chip for more flexibility. The modulator mainly uses switched-capacitor circuits. The technology used is 0.13 μm with 8 layers of metallization. The chip occupies 2 mm² including bond pads (approximately half of

that without bond pads) and has 34 pins.

To increase the probability of a successful design and to allow fine-tuning, the bias voltages are set off-chip using DACs controlled by a PC and an FPGA.

Despite the efforts spent to attain the 74 dB SNR target, the circuit-level simulations indicate a peak SNR of 67 dB. That does not mean the DAC mismatch compensation technique cannot be validated using this chip; but that the proposed technique can only be shown to improve the SNR up to 67 dB.

The experimental results are reported in the following chapter.

Pin	Type	Name	Purpose
1	In	p1	Clock phase ϕ_1
2	In	p2	Clock phase ϕ_2
3	Bias	vref_n	Lower voltage reference for the quantizer
4	Power	VDD1	Positive supply voltage (V_{dd})
5	Bias	vref_p	Higer voltage reference for the quantizer
6	Out	d0	$\Delta\Sigma$ modulator output bit 0 (LSB)
7	Out	d1	$\Delta\Sigma$ modulator output bit 1
8	Out	d2	$\Delta\Sigma$ modulator output bit 2
9	Out	d3	$\Delta\Sigma$ modulator output bit 3
10	Out	d4	$\Delta\Sigma$ modulator output bit 4 (MSB)
11	Power	VSS_buffers	Negative supply voltage for output buffers only
12	In	test_quant	Control signal to switch to test mode
13	In	vip	$\Delta\Sigma$ modulator differential input signal (positive)
14	In	vin	$\Delta\Sigma$ modulator differential input signal (negative)
15	Power	VDD2	Positive supply voltage (V_{dd})
16	Power	VSS	Negative supply voltage (V_{ss})
17	Out	dac3p	DAC output (for testing purposes)
18	Out	x1n	First integrator positive output (for testing purposes)
19	Out	x1p	First integrator negative output (for testing purposes)
20	In	reset	Reset signal
21	Power	VSS	Negative supply voltage (V_{ss})
22	Bias	vbdac	Bias current for the DAC elements
23	Power	VDD_dac	Positive supply voltage (V_{dd})
24	Power	VSS	Negative supply voltage (V_{ss})
25	In	mismatch4	Control enabling mismatch on the DAC MSB
26	In	mismatch0	Control enabling mismatch on the DAC LSB
27	Bias	vbcmfb	Bias current for the CMFB
28	Bias	vb1	Bias current for the differential amplifiers
29	Bias	vbp	Bias current for the differential amplifiers
30	Bias	vb3	Bias current for the differential amplifiers
31	Power	vcom	Common-mode voltage
32	Power	VDD_buffer	Positive supply voltage for output buffers only
33	Power	VSS	Negative supply voltage (V_{ss})
34	In-Out	vq-p	Quantizer input signal (for testing purposes)

Table 7.1: Description of IC pins.

Chapter 8

Experimental Results

This chapter reports and analyses the results taken from the $\Delta\Sigma$ modulator IC, fabricated to experimentally test the novel DAC mismatch compensation technique. Screen captures from the test equipment are provided to analyze important waveforms and frequency spectrums. The SNR at the chip output is calculated with and without compensation to quantify its effectiveness. A table at the end of the chapter summarizes the measured results.

8.1 Test Setup and Equipment

The PCB block diagram shown in Fig. 7.12 provides an overview of the test setup. The input signal fed to the UUT (the $\Delta\Sigma$ modulator), as well as the clock signal are both generated from an Agilent 81150A arbitrary function generator.

The input signal from the 81150A is a single-ended sinusoidal wave lying in the $\Delta\Sigma$ modulator signal band, i.e. near 1/4 the sampling frequency. The signal is converted to differential by a circuit built around an Analog Device AD8132 fully differential amplifier (see Chapter 7).

Figure 8.1: Picture of the tested board.

The 5-bit digital output of the $\Delta\Sigma$ modulator is connected to an FPGA. Upon a command from the PC, the FPGA buffers 2^{18} samples from the $\Delta\Sigma$ modulator using block RAM and transfers those samples to the PC via the PCI bus, for analysis. Matlab is used to analyze the output data from the UUT, i.e. to plot the frequency spectrum and calculate the SNR.

The FPGA is also utilized to generate the two non-overlapping clock signals needed by the UUT from the square wave clock signal generated by the 81150A.

A 80-pin connector allows communication between the PCB and the FPGA.

The supply voltages are generated by an Agilent 6624A four-channel DC power supply. It provides V_{dd} , V_{com} (the common-mode voltage) and ground for the UUT, as well as 3.3 V for the buffers to communicate with the FPGA.

A picture of the tested chip and the PCB is shown in Fig. 8.1.

8.2 Bias Optimization and Power Consumption

As stated in the preceding chapter, the various bias voltages for the UUT are set by DACs which are controlled by a PC. This includes the bias voltages fed to the 13 differential amplifiers in the $\Delta\Sigma$ modulator, both reference voltages setting the quantizer dynamic range and the bias voltage setting the reference current of the DAC in the $\Delta\Sigma$ loop.

When the system was first powered on, the DACs were programmed to set the bias voltages to the same values as in the circuit-level simulations. While it made the $\Delta\Sigma$ modulator work – showing a familiar output frequency spectrum¹ – the SNR results were disappointing (below 60 dB).

A Matlab script was developed to optimize the bias voltages by maximizing the SNR (after calibration) for a given operation point (input signal amplitude, clock and signal frequencies, supply voltages). Given the size of the optimization problem (seven bias voltages), the time it takes to transfer data from the $\Delta\Sigma$ modulator to the PC and then calibrate the coefficients, the bias voltage optimization process can take several days to complete. And that is for one operation point.

¹Which drew a large smile on the author's face...

Changing one parameter (the input signal amplitude, for instance) changes the optimal bias voltages and the resulting peak SNR. Moreover, what sometimes seemed like a good set of bias voltages yielding a high peak SNR resulted in an awkward-shaped SNR-vs-input amplitude curve with the SNR dropping by up to 8 dB when the input signal amplitude was reduced by 1 dB from the amplitude over which the bias optimization was done.

The results that are reported in this chapter are arguably the best results (highest SNR for a decent SNR-vs-input amplitude curve) that could be achieved after several weeks of tweaking the bias voltages. The system is *possibly* capable of better SNR performances or faster operation, but a set of bias voltages that would result in such improved performances could not be found over the time spent optimizing it.

The results are obtained using $V_{dd} = 1.6$ V, $V_{com} = 0.75$ V, $f_s = 1$ MHz and the full-scale input amplitude is 1.15 V pk-pk on each phase (2.30 V pk-pk differential). The total DC current drawn by the V_{dd} power supply is 23 mA, for a 37 mW DC power consumption.

8.3 Input Signal

No matter the $\Delta\Sigma$ modulator performances, the SNR at its output can only be as good as the SNR at the input. Fig. 8.2 shows the measured waveforms at the $\Delta\Sigma$ modulator input. The differential signal amplitude is 916 mV pk-pk. This

Figure 8.2: Oscilloscope screen capture showing both phases of the differential input signal.

corresponds to -8 dBFS^2 , i.e. near the region where the peak SNR is obtained (see section 8.6). No obvious clipping can be observed. A 10 mV DC offset is present on the differential signal (the common voltage is not the same on both phases) due to component tolerances on the PCB. This DC offset will be removed by the STF of the modulator (see Chapter 6).

A better evaluation of the input signal quality is done using a spectrum analyzer. A screen capture is found in Fig. 8.3. The measure was taken using a high-impedance differential probe connected to a probe amplifier to properly drive the spectrum analyzer 50Ω input. The 20 dB discrepancy with the actual input signal power is due to the 10:1 probe attenuation.

The input signal spectrum shows that the third harmonic is 73 dB lower than

²The full-scale voltage (0 dBFS) is defined here as the full-scale range of the quantizer in the $\Delta\Sigma$ loop.

(a) From DC to f_s .

(b) Zoom on signal band.

Figure 8.3: Spectrum analyzer screen capture showing the differential input signal. The signal amplitude is -8 dBFS.

the input signal. The odd-numbered harmonics will fall in the signal band (around $f_s/4$) after the sample and hold, so this third harmonic could be problematic for SNR performances in the > 70 dB range. However, as will be seen shortly, the peak SNR was limited to 64 dB, so the harmonic distortion of the input signal is not the limiting factor of the tested system.

The noise around the input signal tone is also significant. The noise power integrated over the signal band is estimated at approximately -80 dBm. The input signal tone being measured at -16 dBm, this indicates the SNR of the input signal is around 64 dB. It is difficult to tell which proportion of this noise is due to the test equipment itself, the single-ended to differential conversion circuit or the signal source. Nonetheless, the SNR of the input signal is one possible limiting factor of the SNR at the system output.

8.4 Waveforms

The signals at the output of the first integrator and the DAC in the $\Delta\Sigma$ modulator are connected to pads on the IC for monitoring purposes. Figs. 8.4 and 8.5 show those waveforms during normal operation, captured by an oscilloscope. These results show that the settling of the integrator is fast enough for this operating frequency, while the DAC output is more problematic due to its higher capacitive load. Reducing the operating frequency did not improve the SNR performances which indicates that the DAC settling is adequate.

The digital signal taken from the $\Delta\Sigma$ modulator output is plotted in Fig. 8.6

Figure 8.4: Oscilloscope screen capture showing the $\Delta\Sigma$ modulator DAC output (one phase only due to pad limitation). The signal returns to zero on each clock cycle due to the circuit architecture of the switched-capacitor buffer.

Figure 8.5: Oscilloscope screen capture showing the $\Delta\Sigma$ modulator first integrator output (differential).

Figure 8.6: $\Delta\Sigma$ modulator time-domain output samples. The maximum range is from 0 to 31 (5-bit output). The input signal amplitude is -8 dBFS.

for a -8 dBFS input signal amplitude. While not a lot of relevant information can be extracted from this figure, it shows that the output signal is centered at midscale and it relates the input signal amplitude to the full-scale range of the quantizer.

8.5 Modifications to the Digital Calibration Chain

Despite the notch at DC in the STF of the $\Delta\Sigma$ modulator, the (digital) output of the $\Delta\Sigma$ modulator contains a slight DC offset. As was explained in Chapter 4, since the implemented compensation technique needs to minimize the noise in an observation band around DC, a DC offset compromises the effectiveness of the technique.

After inspection of the system, it was found that this DC offset was caused by the interface between the summing block in the $\Delta\Sigma$ modulator and the quantizer. If the midscale voltage of the quantizer is not set exactly to the common-mode voltage of the summing block differential output, a DC offset occurs at the modulator output. Unfortunately, adjusting the quantizer reference voltages to correct this situation affected the SNR performances of the system.

As an alternative solution to this problem, the digital calibration chain was modified to remove any DC offset at the $\Delta\Sigma$ modulator output. A zero at DC was introduced in transfer function of the lowpass decimation filter. This is achieved by multiplying the transfer function by $(1 - z^{-1})$; i.e. each sample is subtracted from the previous one thus creating a digital differentiator. Since this added zero at DC effectively removes an order to the lowpass filter, another cell needs to be cascaded. The modified lowpass³ filter transfer function becomes:

$$T(z) = \left(\frac{z-1}{z} \right) \left(\frac{2^{-A}}{z - (1 - 2^{-A})} \right)^3 \quad (8.1)$$

This modification proved to work well: the average value at the output of the filter is zero and the minimization algorithm could find the optimum set of coefficients, as can be seen in the next section.

The calibration was performed by setting $A = 8$ and $K = 2^{17}$, which gave better results than the values previously set in Chapter 6 using simulations ($A = 9$ and $K = 2^{16}$). This means the observation band and the number of samples used

³Technically, the filter becomes a bandpass filter.

for noise power estimation are effectively doubled.

8.6 Results

This section presents and analyzes the experimental results obtained by the $\Delta\Sigma$ modulator. The digital calibration chain, part of the DAC mismatch compensation technique described in the previous chapters, is implemented in Matlab synthesizable code and applied to the $\Delta\Sigma$ modulator output to verify its effectiveness.

A table is included at the end of this chapter to summarize the experimental results.

8.6.1 The Implemented NTF

Fig. 8.7 shows the frequency spectrum of the modulator to compare it to its intended NTF.

The quantization noise shape is similar to the theoretical NTF, with notable differences. First, the position of the NTF zeros are different: they are measured at $f/f_s = \{0.234, 0.258\}$ while they should have been at $f/f_s = \{0.238, 0.262\}$. This shifts the center frequency of the signal band from $f/f_s = 0.25$ to $f/f_s = 0.246$. This behavior was observed in circuit-level simulations too, so it's not due to component tolerance but to the circuit architecture itself.

The position of the NTF zeros are mainly determined by the resonator feedback gain factors g_1 and g_2 . A simple mathematical analysis reveals that a 2.5% error on those gain factors will shift the NTF zeros to the values obtained by the fabricated

Figure 8.7: Power spectrum at modulator output (after calibration) for a -9 dBFS input signal.

chip, as illustrated in Fig. 8.8. Given the large value of the capacitors used to implement those gains (≥ 3 pF), parasitic capacitances are likely not the cause of this shift. A more probable cause is the limited gain of the differential amplifiers. Fortunately, system-level simulations using the shifted NTF zeros demonstrates that if the center frequency of the signal band is modified when calculating the SNR, there is no SNR penalty due to this shift in frequency.

The implemented NTF also diverges from the theoretical NTF at frequencies around $f/f_s = 0.06$ and $f/f_s = 0.5$. The most probable hypothesis explaining this behavior is that the bias voltages result in a quantization step which is different for the quantizer and the DAC which has the same effect as changing the feedback gain factors $\alpha_{0..4}$. Fig. 8.8 shows the measured power spectrum once the bias

Figure 8.8: Power spectrum at modulator output (after calibration) for a -9 dBFS input signal. The bias voltages fed to the chip are modified to fit the theoretical NTF. The gain factors g_1 and g_2 are also attenuated by 2.5% when plotting the theoretical NTF.

voltages were tweaked to fit the measured NTF to the theoretical NTF. While the NTF is closer to its ideal shape, this new set of bias voltages also increased distortion and the net effect was lower SNR performances. Therefore, the bias voltages were reverted to the ones used in Fig. 8.7.

8.6.2 The Effect of Calibration on the Fabricated Chip

The power spectrum of the modulator output, zoomed on the signal band, is plotted in Fig. 8.9 before and after calibration of the coefficients. The number of samples used to compute the FFT is 2^{18} as in Fig. 4.6 (simulation results). The calibration allows a 9 dB improvement in harmonic distortion and lowers the noise

floor by approximately 8 dB. Without calibration, the SNR is measured at 56 dB.

After calibration, the SNR is 64 dB.

When the SNR is calculated without including the third and fifth harmonics of the input signal, the SNR is 67 dB. This is better than the SNR of the input signal which was estimated to 64 dB in Section 8.3, revealing that a significant portion of the input noise observed with the spectrum analyzer was due to the test equipment.

The third harmonic, at -76 dBFS after calibration, indicates that the calibration algorithm found the optimum solution up to the resolution of the coefficients (see Chapter 6).

The spectrums are plotted using a -8 dBFS input signal because the SNR begins to drop at higher amplitudes, as will be seen shortly.

To check how the implemented compensation technique would perform with severe DAC mismatch, intentional 1% mismatch can be applied to the LSB and MSB elements of the DAC using configuration jumpers on the PCB (see Chapter 7). Fig. 8.10 shows the frequency spectrum of the modulator output, before and after calibration, when DAC mismatch is enabled.

The SNRs are 45 and 62 dB before and after calibration respectively. The minimization algorithm converged to a local minimum which is different from the solution found without the intentional DAC mismatch. Interestingly, the set of coefficients found by the minimization algorithm results in lower harmonic

(a) Without calibration.

(b) With calibration.

Figure 8.9: Power spectrum at modulator output before (a) and after (b) the calibration process. The input signal amplitude is -8 dBFS. The “skirt” around the input signal tone is an FFT artifact and did not impact the SNR calculation. The odd-numbered harmonics are aliased into the signal band due to the sampling process.

distortion than in Fig. 8.9, but a higher noise floor. The net effect is a SNR which is 2 dB lower than without intentional mismatch. Still, the calibration allowed a 17 dB improvement over the non-calibrated modulator.

Reported experimental results of $\Delta\Sigma$ modulators usually include a plot showing the SNR as a function of the input amplitude. Those results are depicted in Fig. 8.11, with and without intentional DAC mismatch.

Before discussing those results, a few words on the curve “optimized for SNR” are necessary. To plot this curve, Matlab calls an optimization function which finds the coefficients (the variables) that maximizes the SNR (the objective function). Obviously, this optimization technique could hardly be used in a commercial $\Delta\Sigma$ ADC for two reasons: for each iteration, the SNR needs to be calculated, which requires significant computation power. Also, the SNR is not trivial to obtain when the input signal is not a single sinusoid, making the optimization of the coefficients based on SNR is thus hard to perform during normal operation. However, it shows the absolute best SNR performances that can be achieved by the tested $\Delta\Sigma$ modulator for the current point of operation. It is therefore used as a tool to evaluate performance of the implemented compensation technique rather than a possible implementation of the coefficient optimization technique.

From Fig. 8.11, the peak SNR after calibration is 64.5 dB⁴. It occurs at an input amplitude of -9 dBFS (818 mV pk-pk differential). Without calibration,

⁴If no specific mention is made, the discussion applies to the results without intentional DAC mismatch.

(a) Without calibration.

(b) With calibration.

Figure 8.10: Power spectrum at modulator output with intentional 1% DAC mismatch.

(a) Without intentional mismatch.

(b) With intentional mismatch.

Figure 8.11: SNR vs input amplitude with and without intentional mismatch. The input amplitude corresponding to -33 dBFS is the lower amplitude limit of the equipment used to generate the input signal. See the text for a description of the curve “Optimized for SNR”.

the peak SNR is 57.3 dB, and it occurs at a higher input amplitude: -7 dBFS. The optimized for SNR curve shows that the best peak SNR that could have been achieved by the calibration technique is 65.6 dB, at -8 dBFS.

The 2 dB difference between the optimized for SNR curve and the theoretical modulator curve is due to thermal noise; the theoretical modulator curve, obtained from system-level simulations includes quantization noise only.

A quick look at Fig. 8.11 tells what is preventing the fabricated $\Delta\Sigma$ modulator from attaining the 74 dB peak SNR objective: the SNR starts to drop at amplitudes higher than -8 dBFS. If the curve for the calibrated modulator is extrapolated to higher amplitudes, the SNR at -1 dBFS is 73 dB. This early SNR drop is probably caused by clipping of the signal in the $\Delta\Sigma$ modulator. A better design of the differential amplifier cell would have allowed a larger dynamic range of the input signal. Although the biasing of the chip makes it possible to change the input voltage associated with 0 dBFS (by reducing the quantizer dynamic range), this resulted in an overall lower peak SNR because thermal noise becomes a more dominant noise source.

Yet another observation from Fig. 8.11 is that the SNR-vs-input amplitude curve after calibration does not follow a straight 1 dB/dB line. For instance, at -21 dBFS input amplitude, the SNR is 6.3 dB lower than the ideal modulator curve whereas it is within 3.2 dB of that same curve at lower amplitudes. The fact that the optimized for SNR curve has a more linear shape shows that there was a

better set of coefficients, in terms of SNR, than the one found by the minimization algorithm. It was verified that the minimization algorithm *did* find the coefficients that minimized the noise in the observation band for that point, however, for this input amplitude, minimizing the noise in the observation band *did not* maximize the SNR in the signal band. This behavior could not be recreated in system-level simulations. Moreover, when using different bias voltages, larger bumps than those reported in Fig. 8.11 were measured. In light of those observations, it seems that nonlinearities in the analog blocks forming the $\Delta\Sigma$ modulator can increase the noise in the observation so that the minimization algorithm cannot fully optimize the coefficients. The nature of those nonlinearities is dependent on the biasing and the signal amplitude. Which component causes those nonlinearities could not be identified at this point.

With intentional DAC mismatch, the peak SNR after calibration is 62.7 dB, again at an input amplitude of -9 dBFS. Enabling the intentional DAC mismatch caused a 1.8 dB peak SNR loss whereas this number increased to 10.1 dB without calibration. It should also be noted that the optimized for SNR curve also suffered a 0.8 dB peak SNR drop when the intentional DAC mismatch was enabled which indicates that the maximum achievable SNR with calibration had dropped by 0.8 dB with intentional DAC mismatch.

Finally, Fig. 8.12 plots the measured peak SNR over the range of OSR from 8 to 20. This figure shows that the implemented DAC mismatch compensation

Figure 8.12: Peak SNR for various OSRs. For $\text{OSR} \geq 18$, the fifth harmonic of the input signal lies outside the signal band.

technique allowed the peak SNR of the fabricated chip to be within 1.6 dB of the maximum achievable SNR for this range of OSRs. As explained earlier, those results are short of the 74 dB peak SNR target due to implementation problems which led to dynamic range limitations.

8.7 Summary

The fabricated $\Delta\Sigma$ modulator chip was put to test in this chapter. The UUT peak SNR is limited to 57.3 dB without compensation. When the proposed DAC compensation technique is applied to the modulator output, the peak SNR increases to 64.5 dB. This is within 2 dB of the maximum achievable SNR for the current UUT. The peak SNR is below the 74 dB target for which the modulator

was designed due to implementation flaws which limits the dynamic range of the input signal and causes the SNR to start dropping at 8 dB below the full-scale range of the quantizer.

The compensation technique was found to be efficient when larger (1%) DAC mismatch was intentionally enabled to model the behavior of possible weaker fabricated chip samples. Under those conditions, the proposed compensation technique was also shown to be efficient, bringing the peak SNR from 47.1 dB to 62.7 dB, again within 2 dB of the maximum achievable SNR for the current UUT.

Table 8.1 summarizes the experimental results obtained from the fabricated chip.

Parameter	Value
Technology	0.13 μ m CMOS
Supply voltage (V_{dd})	1.6 V
Common-mode voltage (V_{com})	0.75 V
DC power consumption	37 mW
Clock frequency	1 MHz
Oversampling ratio (OSR)	14
Full-scale input signal amplitude (0 dBFS)	2.30 V pk-pk differential
Peak SNR without compensation	57.3 dB (at -7 dBFS)
Peak SNR with novel compensation technique	64.5 dB (at -9 dBFS)
Maximum achievable peak SNR of the tested chip (see text)	65.6 dB (at -7.5 dBFS)
Peak SNR with intentional 1% mismatch, without compensation	47.2 dB (at -7 dBFS)
Peak SNR with intentional 1% mismatch, with novel compensation technique	62.7 dB (at -9 dBFS)
Maximum achievable peak SNR of the tested chip with intentional 1% mismatch	64.8 dB (at -8.5 dBFS)

Table 8.1: Summary of experimental results.

Chapter 9

Summary of Thesis

In this thesis, a novel technique for compensation of DAC nonlinearities in multibit bandpass $\Delta\Sigma$ modulators was proposed, analyzed and implemented.

After presenting the basic theory of $\Delta\Sigma$ modulation, the advantages of multibit quantization were highlighted. Briefly, multibit $\Delta\Sigma$ modulators allow high SNRs at low OSRs while putting less stringent requirements on the circuit components such as the operational amplifiers. This makes multibit $\Delta\Sigma$ converters suitable in applications with larger bandwidths than typical $\Delta\Sigma$ applications. On the other hand, it is well documented in the literature that the SNR is limited by the linearity of the feedback DAC. The mismatch between its constituent elements, caused by the finite precision of IC lithography, typically limits multibit $\Delta\Sigma$ converters to approximately 10 bit resolution if no compensation technique is used.

The most common compensation technique is dynamic element matching (DEM), where the DAC unit elements are shuffled on each clock cycle. This randomizes

the mismatch errors, transforming the spurs into wideband noise. The main advantage of DEM is its simplicity and it works well in typical $\Delta\Sigma$ application with high OSRs. However, since the noise due to mismatch is not removed but spread into wideband noise, its effect diminishes as the bandwidth increases, i.e. for low OSR modulators. Also, the amount of noise due to mismatch is dependent on the mismatch severity, which means DEM might not be sufficient if an IC fabrication process with lower precision is used. Finally, DEM requires DACs made of 2^n-1 unit elements, which means the number of its constituent elements increases exponentially with the number of bits.

Published calibration techniques, which precisely measure the DAC elements at power-up and use this information to compensate for the DAC mismatch during normal operation, were also presented in Chapter 3. Since the DAC elements are measured only once, if mismatch changes due to thermal variations, those changes cannot be tracked. The novel compensation technique introduced in this thesis compares advantageously to those state-of-the-art calibration techniques by continuously tracking the DAC mismatches.

The novel compensation technique was introduced in Chapter 4. It was shown that the technique necessitates that an NTF zero is added to a frequency away from the signal band. This NTF zero creates an observation band, which allows minimizing the noise due to DAC nonlinearity while the modulator is in operation. The noise is minimized by digitally calibrating coefficients of a nonlinear

function. Simulation results showed that using this technique and a DAC with 1% mismatch¹, the SNR improved by 26 dB. The SNR after calibration was 73.9 dB, which is only 0.4 dB less than the ideal-DAC case.

The proposed technique was analyzed in greater detail in Chapters 5 and 6. Information concerning the minimization algorithm and guidelines to set important system-level parameters such as the ideal noise and signal transfer functions, lowpass filter cutoff frequency and the number of samples used for noise power estimations were given. Most parameters of the digital calibration chain are chosen to minimize the probability of error, i.e., the probability that the minimization algorithm goes in the wrong search direction during calibration.

The DAC mismatch compensation technique was validated by applying it to a fourth-order, 5-bit $\Delta\Sigma$ modulator fabricated in 0.13 μm CMOS. The modulator was designed to attain a 74 dB peak SNR (12 bit resolution) at an OSR of 14. Although the typical recommended IC layout techniques to minimize mismatch was used, the peak SNR of the tested sample was 57.3 dB. When the compensation technique was applied, the peak SNR increased to 64.5 dB. The peak SNR occurs at an amplitude of -9 dBFS. The SNR starts dropping early, at any amplitude larger than -8 dBFS, due to flaws in the analog section of the IC design that limits the dynamic range of the implemented $\Delta\Sigma$ modulator. An extrapolation of the SNR curve shows that the peak SNR could have been around 73 dB at -1 dBFS, as in simulation, with the compensation technique.

¹This value is taken as a severe mismatch factor using current CMOS process fabrication.

9.1 Objective Evaluation of the Novel DAC Mismatch Compensation Technique

The novel technique implemented in this thesis has the following advantages:

- It is mainly digital. The only modification to the $\Delta\Sigma$ loop is the addition of an integrator to create the observation band. The calibration is done by digital gates using the $\Delta\Sigma$ output bitstream. As such, the technique will benefit from any future MOSFET feature size scaling. For instance, it will gradually use a smaller chip area and consume less power.
- It continuously runs in the background so it tracks the mismatch variations due to environmental changes. This was verified by applying intentional mismatch while the modulator was in operation.
- It applies to both bandpass and lowpass $\Delta\Sigma$ modulators.
- It applies to binary-weighted DACs whose number of elements increases linearly with the number of bits.
- Its effectiveness was shown to be essentially independent of the DAC mismatch severity up to 1% mismatch.

On the other hand, it has the following weaknesses:

- As implemented, calibration is relatively slow², requiring around 10^8 clock cycles for 1% DAC mismatch. This is because the number of samples used

²Compared to DEM, for example.

to estimate the noise in the observation band must be relatively high to get reliable estimates.

- Because the quantity to minimize is noise, the algorithm can temporarily set one coefficient to a non-optimal value which will affect the SNR. While the occurrence frequency can be attenuated by properly adjusting system-level parameters, the probability of error is not null.
- The digital calibration chain tends to be complex. For extra flexibility, the digital calibration chain was implemented in Matlab rather than in silicon in this thesis. So until the work continues (see section 9.3), it is hard to tell the area and power overhead associated with the compensation technique.
- Because it applies to binary-weighted DACs, if a flash quantizer is used, a thermometer-code to binary-code digital conversion circuit is required.

9.2 Contributions

The present thesis brings the following contributions to the state of the art:

- A novel digital technique to compensate for DAC mismatches in multibit $\Delta\Sigma$ modulators was invented. To the author's knowledge, this is the first reported multibit $\Delta\Sigma$ modulator digital calibration technique that runs in the background, doesn't require a pilot signal, and applies to binary-weighted DACs. The novel technique applies to both lowpass and bandpass $\Delta\Sigma$ modulators.

- The novel technique was analyzed. The probability of error was derived, which helps setting the various system-level parameters of the digital calibration chain.
- A new minimization algorithm that suits the particularities of the given optimization problem was described and implemented. The minimization algorithm does not require multiplication or division operations, so its physical implementation should be area efficient.
- The novel compensation technique was successfully applied to an experimental $\Delta\Sigma$ modulator fabricated in $0.13\ \mu\text{m}$ CMOS.

9.3 Future Work

The following list provides ideas of areas that would benefit from additional research:

- Another $\Delta\Sigma$ modulator chip should be fabricated, with better analog performances, to confirm that the novel compensation technique can be successfully applied to a modulator with $>70\ \text{dB}$ SNR.
- The digital calibration chain should be synthesized to evaluate its area and power consumption.
- The possibility of using adaptive parameters in the digital calibration chain should be investigated. This could allow the coefficient values to be quickly

set near their optimum value (with a lower resolution) to lower the noise in the observation band, and then adjust the parameters to gradually favor precision over speed.

- Alternative minimization algorithms should be researched. For example, the implementation of the minimization algorithm could benefit from parallel computing. As depicted in Fig. 9.1, instead of using new samples from the $\Delta\Sigma$ modulator to estimate the noise in the observation band on each iteration, the same samples could be used to compare different sets of coefficients. This could be done by instantiating several (three in the example in Fig. 9.1) nonlinear transfer functions, with different coefficients, in parallel, working on the same output samples from the $\Delta\Sigma$ modulator. This would reduce calibration time for two reasons. First, several operations are done at the same time, instead of one after the other. Second, because the same samples are used to estimate the noise with different coefficient sets, the probability of error is reduced.
- A quantizer that readily outputs a thermometer-code signal, such as a successive approximation ADC, could be used instead of a flash quantizer. This would eliminate the need for a conversion circuit to drive the binary-weighted DAC and could result in a more area-efficient implementation.
- The architecture could be modified to apply the compensation technique to

DACs composed of identical unit elements.

- As explained in Chapter 5, some DAC elements are more difficult to calibrate because they tend to generate harmonic distortion with a lesser impact on the noise floor than other DAC elements. It would be interesting to try to find a way to combine the novel compensation technique to DEM. The DEM would spread the harmonic spurs into wideband noise that would be more easily minimized by calibration.
- The effects of CMOS process scaling on DAC mismatch could be researched to evaluate if the proposed compensation technique would become more desirable in the future.
- Additional research could also be done to find new applications to the minimization of the noise in the observation band. If it is found that the noise in the observation band is correlated to other circuit nonidealities (dynamic errors, incorrect gains in the $\Delta\Sigma$ loop, etc.), new techniques could be used to reduce the impact of these nonidealities by minimizing the noise in the observation band.

Figure 9.1: Possible implementation of the digital calibration chain using parallel computing for faster calibration.

Appendix A

Description of Matlab Functions Used in the Thesis

The thesis refers to several Matlab functions without much description of what these functions do to avoid disrupting the flow of the thesis.

The most important Matlab functions are described in this appendix.

delsig toolbox: The **delsig** toolbox refers to a collection of Matlab functions pertaining to $\Delta\Sigma$ modulation, developed by Richard Schreier, and freely available to download on the Mathworks web site [9]. Among other tools, the toolbox provide functions to find the optimal NTF for specified system-level parameters (order, number of bits, etc.), find the gain factors to implement the NTF to a desired architecture and scale those coefficients for specified internal node signal swing. It also includes functions to simulate $\Delta\Sigma$ modulators and calculate the SNR at the modulator output. Since there are different ways of calculating the SNR [8], the use of the **delsig** toolbox in the majority of published works in this field standardizes the SNR calculation method.

synthesizeNTF: Returns the NTF for specified parameters. Those parameters are: modulator order, oversampling ratio, optimized or coincident NTF zeros, center frequency and maximum NTF gain.

realizeNTF: Returns the gain factors of a specified NTF (and optionally, a specified STF) for a given architecture.

scaleABCD: Scales the loop filter gains of a general $\Delta\Sigma$ modulator according to the maximum desired signal swing at the output of each integrator.

calculateSNR: Estimates the SNR of the $\Delta\Sigma$ modulator output. The input parameter is the in-band bins of the signal FFT. Another parameter, *nsig*, specifies how many bins on each side of the signal tone must be considered as part of the input signal. Those bins are removed from the calculation of the noise power.

calculateSNR_time_samples: This function, not part of the *delsig* toolbox, was written to directly obtain the SNR value from time samples. It simply formats the signal to fit the **calculateSNR** function requirements and then calls **calculateSNR**.

hann: Returns the coefficients of a Hann window. The use of a Hann window when computing the FFT of a signal prevents the tone from smearing into several FFT bins. To avoid tone smearing, the input signal frequency must also be in the middle of an FFT bin.

bode: Returns (and optionally plots) the magnitude of a given transfer function over a range of specified frequencies.

`pwelch`: The `pwelch` function calculates the power spectral density of a signal via Welch's method. One parameter specifies how many averages are performed. Averaging smoothes the response of noisy signals at the expense of lower frequency resolution. When a high number of signal samples are available, averaging is useful to identify the noise spectral shape. The `pwelch` function was used to plot the frequency power spectrum of signals in this thesis, while the common `fft` function was used to calculate the SNRs.

Appendix B

Matlab Code for the Minimization Algorithm

This appendix contains the Matlab code implementing the minimization algorithm proposed in Chapter 5. The same code was used to provide the experimental results, the `run_sim` command was just replaced by the command `get_data` which downloads data from the FPGA.

08-10-20 14:27 D:\Ghyslain\work\icgcuggg\minimization.m

1 of 3

```

H = tf(2^-9,[1,-1+2^-9],1)^2; % Lowpass filter transfer function (A=9)
coefs = 2.^(0:4); % Initial values for coefficients (nominal values)
sim_count = -1; % Register to count the number power estimations
iter = 1; % iteration number
step = 2^(-7); % coefficient resolution
delta = [2 2 2 3 7] * step * 2; % x2 for the first iteration only
coef = 2; % Coefficient being optimized

% Open new figures to plot results as calibration progresses
hnoise = figure;
hsnr = figure;

% Before starting calibration, simulate once and get the value of noise in
% the observation band and SNR with nominal coefficient values
run_sim % Simulate that modulator !
snr = calculateSNR_time_samples(dactf(simout,coefs),.25,14,1,5,0.25);
noise = get_noise_near_dc(coefs,simout,H);

while(iter < 11)
    if coef == 2
        fprintf('After %d iterations, %d calls\n',iter-1,sim_count);
    end
    % Step one: Go to left side of the curve
    coefs(coef) = coefs(coef) - delta(coef);
    run_sim
    noise1 = get_noise_near_dc(coefs,simout,H);
    coefs(coef) = coefs(coef) + delta(coef);
    run_sim
    noise2 = get_noise_near_dc(coefs,simout,H);
    if noise2 < noise1 % Are we on the left side of the curve ?
        % Yes, take note of the coordinates
        coef_left = coefs(coef) - delta(coef);
        noise_left = noise1;

        % Step two: jump to the right side of the curve
        coefs(coef) = coefs(coef) + delta(coef);

        % Step three: find the coef value on the right side of the
        % curve that generates noise power that it is closest to the
        % noise power generated from the left side of the curve
        calib_done = false;
        first_time = true;
        while(not(calib_done))
            % First thing before measuring noise, are we good ?
            if (coefs(coef) - coef_left) > 4*delta(coef)
                % Houston, we have a problem. The coef is way too far right
                % Let's just put the value of coef_left + delta and
                % optimize next coefficient. Maybe noise floor was too high
                % to optimize this coefficient.
                coefs(coef) = coef_left + delta(coef);
                fprintf('Problem with coef %d\n',coef);
            end
        end
    end
    iter = iter + 1;
end

```

08-10-20 14:28

D:\Ghyslain\work\icgcuggg\minimization.m

2 of 3

```

        calib_done = true;
    else
        % We're good
        % Start by taking three readings of noise for coef values
        % offset by 1 LSB

        coefs(coef) = coefs(coef) - step;
        run_sim
        noise1 = get_noise_near_dc(coefs,simout,H);

        coefs(coef) = coefs(coef) + step;
        if first_time
            run_sim
            noise2 = get_noise_near_dc(coefs,simout,H);
        end

        % First, make sure we're on the right side of the curve
        if noise1 > noise2
            % Oops! Probably not (or at least, not far enough on the right)
            coefs(coef) = coefs(coef) + delta(coef);
        else
            % We're on the right side of the curve, take 3rd noise measure
            coefs(coef) = coefs(coef) + step;
            run_sim
            noise3 = get_noise_near_dc(coefs,simout,H);
            coefs(coef) = coefs(coef) - step; % Go back in the middle

            % Then, check which noise reading is closest to the left value
            if abs(noise_left - noise1) < abs(noise_left - noise2)
                % The noise1 value is the closest
                % Move coef by -1 LSB and start over
                coefs(coef) = coefs(coef) - step;
                noise2 = noise1;
                first_time = false;
            elseif abs(noise_left - noise3) < abs(noise_left - noise2)
                % The noise3 value is the closest
                % Move coef by +1 LSB and start over
                coefs(coef) = coefs(coef) + step;
                noise2 = noise3;
                first_time = false;
            else
                % The noise2 value is the closest: coefficient is optimized
                coefs(coef) = (coefs(coef) + coef_left)/2;
                calib_done = true;
            end
        end
    end
end

if calib_done
    noise(end+1) = get_noise_near_dc(coefs,simout,H);
    figure(hnoise)
end

```

08-10-20 14:30 D:\Ghyslain\work\icgcuggg\minimization.m 3 of 3

```

plot(noise,'b.-');grid on; drawnow;
% Change coef to be optimized (order is 2,3,1,4,5)
switch coef
case 1
    coef = 4;
case 2
    coef = 3;
case 3
    coef = 1;
case 4
    coef = 5;
case 5
    if iter == 1 % We just completed the first iteration
        delta = 0.5*delta;
    end
    coef = 2;
    iter = iter + 1;
    snr(end+1) = calculateSNR_time_samples(dactf(simout,coefs),.✓
25,14,1,5,0.25);

    figure(hsnr)
    plot(snr);grid on; drawnow;
    otherwise % That should never happen, but just in case...
        coef = 2;
    end
end
end
else
    % We seem to be on the right side of the curve
    % Move coefficient towards left side of the curve
    % and redo the same coef
    coefs(coef) = coefs(coef) - delta(coef);
end
end
end

```

Appendix C

Paper Presented at NEWCAS 2007

A paper [6] that was presented at the *IEEE 2007 North West Circuits and Systems International Conference (NEWCAS 2007)* gives supplemental information that is not found in this thesis. This 4-page paper is reproduced in this appendix for the interested reader convenience.

System-Level Design Aspects of a DAC Calibration Technique for Bandpass Delta-Sigma Modulators

Ghyslain Gagnon
Department of Electronics
Carleton University
Ottawa, Canada
Email: ggagnon@ieee.org

Leonard MacEachern
Department of Electronics
Carleton University
Ottawa, Canada
Email: leonard@ieee.org

Abstract—This paper gives relevant system-level design information for a previously reported technique that digitally compensates for DAC nonlinearities in bandpass delta-sigma modulators. A method is described to optimally determine the noise and signal transfer functions, the lowpass decimation filter cutoff frequency as well as the number of samples required for power estimation. Simulation results confirm the validity of the analysis.

I. INTRODUCTION

A new digital calibration technique for compensation of DAC nonlinearities in bandpass delta-sigma modulators was recently published in [1]. This calibration technique relies on the fact that when the transfer function of the feedback DAC in a $\Delta\Sigma$ modulator is nonlinear, the noise floor of the output signal increases. This noise floor increase is due to shaped quantization noise folding; it affects the entire band of the $\Delta\Sigma$ modulator output. By monitoring the noise floor in a frequency band outside the signal band (herein called the observation band), the DAC nonlinearities can be digitally compensated outside the $\Delta\Sigma$ loop, while the modulator is operating. This is accomplished by creating an additional NTF (noise transfer function) zero in the observation band. Digital signal processing is then used to minimize the noise floor in this observation band while the $\Delta\Sigma$ modulator is in operation.

It was shown in [1] that when the noise floor is minimized in the observation band, the signal to noise and distortion ratio (SNDR) in the signal band is improved. This calibration technique is accurate, it is mainly digital, it requires only a minor modification to the $\Delta\Sigma$ loop and allows continuous compensation of environmental drifts. It has the added benefit that it is compatible with binary-weighted element DACs and the storage requirements for the calibrated coefficients increase only linearly with the number of quantizer bits.

This paper gives essential novel information regarding system-level design aspects of the compensation technique based on both mathematical analysis and simulation results.

II. SYSTEM DESCRIPTION

The block diagram of the $\Delta\Sigma$ modulator with its digital calibration loop is shown in Fig. 1. It's a fourth order bandpass modulator with an extra integrator to implement the required additional NTF zero in the observation band (around DC). The

selected loop filter structure is the CRFB [2] because it allows placement of the NTF zeros optimally, which improves SNR, especially for a low oversampling ratio (OSR).

As stated above, DAC nonlinearities increases the noise floor, thus filling the notches in the NTF. The additional NTF zero, at DC, allows the digital calibration loop to monitor and minimize the noise power around DC.

The digital calibration loop is composed of a nonlinear transfer function, a lowpass decimation filter, a power estimator and a minimization algorithm block. The nonlinear transfer function comprises n coefficients (representing the n -bit DAC binary-weighted elements), stored in memory. These coefficients are adjusted one at a time by the minimization algorithm. The combination of the lowpass filter and the power estimator continuously gauges the power contained in the observation band. The minimization algorithm could be any search algorithm that can adjust the coefficients so that the power contained in the observation band is minimum. Details concerning the minimization algorithm is outside the scope of this paper. After a few iterations, the n coefficients are fixed at the values that best compensates for the DAC nonlinearities.

The system under study in this paper has an oversampling ratio of $OSR = 10$. The sampling frequency is normalized to $f_s = 1$. The input signal has a normalized peak-to-peak amplitude of ± 1 . The quantizer and DAC have $n = 5$ bits and the nonlinear function coefficients are stored using $m = 12$ bits for a maximum achievable peak SNDR around 70 dB.

III. SYSTEM-LEVEL DESIGN ASPECTS

A number of design choices must be done to obtain a calibration system which is accurate, reliable and fast, without sacrificing the performances of a regular fourth order bandpass $\Delta\Sigma$ modulator. These design choices include optimum placement of the NTF and STF (signal transfer function) poles and zeros, taking into account the additional NTF zero, as well as the lowpass filter cutoff frequency and the number of samples used for power estimation.

A. Placement of Poles and Zeros

The $\Delta\Sigma$ modulator loop shown in Fig. 1 realizes five NTF poles and zeros, as well as five STF poles and zeros. Using this structure, the NTF and STF can have independently placed

Fig. 1. Detailed model of the digitally compensated $\Delta\Sigma$ modulator. The supplementary circuitry is enclosed in the two dashed boxes.

zeros, but share the same poles [2]. Four NTF zeros are used to shape the quantization noise away from the signal band, as in a typical $\Delta\Sigma$ modulator. They are placed as two conjugate pairs around the signal band, at frequencies (ω_1 and ω_2) that maximizes the SNDR for a given oversampling ratio (OSR). The fifth NTF zero is placed at DC ($z = 1$) to create the observation band.

Four NTF poles are placed at infinity ($z = 0$). Typically, NTF poles are placed closer to the NTF zeros to reduce the out-of-band NTF gain and thus improve stability [2]. However, for the system under study, extensive simulations showed that stability was not jeopardized by placing four poles at $z = 0$, while it maximizes the peak SNDR. The fifth NTF pole, labelled p_5 , plays an important role in the system performances and will be discussed shortly.

The STF can be made maximally flat by letting loop filter coefficients $b_{1..5} = 0$. As explained in [1], if the input signal contains significant power in the observation band, e.g. if it has a DC component, the calibration performances will be altered. Using an STF with a highpass response provides prefiltering to ensure that no interferer is present in the input signal in the observation band. This is done by placing one STF zero at DC. The four remaining STF zeros are placed at $z = 0$ to be canceled by the STF poles, also placed at $z = 0$.

The NTF can thus be expressed as:

$$NTF(z) = \frac{(z-1)(z-e^{\pm j\omega_1})(z-e^{\pm j\omega_2})}{z^4(z-p_5)} \quad (1)$$

while the STF is:

$$STF(z) = \frac{z-1}{z-p_5} \quad (2)$$

where ω_1 and ω_2 are the notch frequencies around the signal band.

This leaves one remaining NTF and STF pole (shared) to position. As this pole moves towards DC, it tends to cancel the extra NTF zero at DC. As shown in Fig. 2, this has three important consequences: 1-it lowers the NTF magnitude in the signal band (which improves the SNR), 2-it flattens the STF magnitude in the signal band and 3-it boosts the NTF magnitude in the observation band (which affects the calibration accuracy and speed).

In most applications, the STF magnitude flatness in the signal bandwidth is a crucial specification. Therefore, the fifth pole p_5 will be positioned at the highest frequency that still meets the application's constraints in terms of frequency response slope in the signal band. This slope is obtained by:

$$\Delta STF(p_5) = \frac{|STF(e^{j\omega_{\max}})|^2}{|STF(e^{j\omega_{\min}})|^2} \quad (3)$$

where $\omega_{\min}$ and $\omega_{\max}$ are the lower and upper frequency of the signal band. For example, to obtain $\Delta STF \leq 0.1$ dB for the system under study, $p_5 \geq 0.677(f/f_s = 0.062)$.

The SNR penalty, ΔSNR , due to the extra NTF zero at DC is obtained by:

$$\Delta SNR(p_5) = \frac{\int_{\omega_{\min}}^{\omega_{\max}} |NTF(e^{j\omega})|^2 d\omega}{\int_{\omega_{\min}}^{\omega_{\max}} |NTF(e^{j\omega})|^2 d\omega, p_5 = 1} \quad (4)$$

Using $p_5 = 0.677$, the SNR penalty is $\Delta SNR = 1.4$ dB. The NTF notch frequencies, ω_1 and ω_2 , are found using the Matlab "delsig" toolbox [3]; $\omega_1 = 2\pi \times 0.236$ and $\omega_2 = 2\pi \times 0.264$. The effect of p_5 on the calibration process is analyzed in the next subsection.

B. Digital Calibration Loop Parameters

Two important system-level design choices must be made for the digital calibration loop: f_c , the lowpass filter cutoff frequency and K , the number of samples used by the power estimator for each estimation. The lowpass filter cutoff frequency determines the observation band while K determines the reliability of the power estimates. To ease the mathematical analysis, it will be assumed here that the lowpass filter has a brickwall response and that the decimation factor is $D = 1$ (no decimation). These assumptions are discussed later.

Fig. 3 shows the expected value of R , the output of the power estimator as coefficient C_5 is swept around its optimal value. Since the nature of signals to be estimated is noise, for each estimation, the result can deviate from the expected value. When the estimated value is significantly different from its expected value, the minimization algorithm can temporarily go in the wrong search direction, which ultimately increases the amount of time required for complete calibration. The

Fig. 2. NTF and STF magnitude response with two different values for the fifth pole p_5 . As p_5 moves towards DC, the characteristics of both transfer functions are more desirable in the signal band, at the expense of increased quantization noise in the observation band.

Fig. 3. Noise power in the observation band as a function of the offset from the ideal value of coefficient C_5 , along with the 95% confidence interval. The other coefficients to their ideal value. $f_c = 3.5 \times 10^{-4}$ and $K = 2^{22}$.

vertical bars in Fig. 3 shows the 95% confidence intervals. The standard deviation depends on the noise power, observation bandwidth and number of samples used for estimation. The following paragraphs derives the expected value and standard deviation for the signal R . Then, f_c can be chosen as to minimize the probability that the minimization algorithm goes in the wrong search direction, herein called the probability of error. Afterwards, K is fixed at a value that results in acceptable probabilities of errors, given all of the other parameters.

Ignoring thermal noise, the signal P , at the lowpass decimation filter input, contains the following components:

- a copy of the input signal (with distortion due to DAC nonlinearity);

Fig. 4. Folded noise unilateral power spectral density as a function of the offset from the ideal value of each coefficient $C_{1..n}$. All other coefficients to their ideal value.

- quantization noise shaped by the NTF;
- folded noise due to DAC nonlinearity;

The component that contains useful information for calibration is the folded noise. All other components are considered as interferers for the calibration process. To ease the analysis, it is assumed that the lowpass filter provides enough attenuation in the signal band that the power of the input signal components at the lowpass filter output is negligible when compared to the noise power in the observation band. The optimal observation bandwidth is thus determined by the folded noise and the quantization noise power.

The folded noise power depends on the DAC nonlinearity severity and the value of the coefficients $C = C_{1..n}$. Its value is difficult to obtain analytically; it was found by simulation. The results are shown in Fig. 4.

Within the observation band, the folded noise power spectral density (PSD), $S_{f,C}^2(f)$, is flat. The folded noise power, at the lowpass filter output, is thus:

$$\sigma_{f,C}^2 = \int_0^{f_c} S_{f,C}^2(f) df = f_c S_{f,C}^2(f) \quad (5)$$

The value of $S_{f,C}^2(f)$ and consequently the value of $\sigma_{f,C}^2$ are valid only for one point of operation, i.e. for a set of coefficients C and a given DAC transfer function.

The quantization noise PSD, $S_q^2(f)$ is assumed to be white Gaussian noise (caused by the signal quantization) shaped by the NTF [2]:

$$S_q^2(f) = \left(\frac{\Delta_q^2}{6f_s} \right) |NTF(e^{j2\pi f})|^2 \quad (6)$$

where Δ_q is the quantization step: $\Delta_q = 2/2^n$. After a few algebraic manipulations, the quantization noise PSD in the observation band (at low frequencies) is found to be:

$$S_q^2(f) \simeq \frac{2^{(-2n+7)}\pi^2}{3(1-p_5)^2} f^2 \quad (7)$$

The quantization noise power, at the lowpass filter output, is thus:

$$\sigma_q^2 = \int_0^{f_c} S_q^2(f) df = \frac{2^{(-2n+7)}\pi^2}{9(1-p_5)^2} f_c^3 \quad (8)$$

The power estimator is an integrate and dump, and performs the following operation:

$$R_j = \sum_{k=jK+1}^{jK+K} Q_k^2 \quad (9)$$

From the central limit theorem [4], the signal R is a normally distributed random signal with mean value:

$$\mu_R = K\sigma_Q^2 \quad (10)$$

and standard deviation:

$$\sigma_R = \sqrt{K/f_c} \sigma_Q^2 \quad (11)$$

where σ_Q^2 is the variance of signal Q :

$$\sigma_Q^2 = \sigma_f^2 + \sigma_q^2 \quad (12)$$

Gathering eq. (5, 8, 10, 11 and 12), the expected value and standard deviation of the signal R , for any set of coefficients C , is:

$$\mu_{R,C} = K \left(f_c S_{f,C}^2(f) + \frac{2^{(-2n+7)}\pi^2}{9(1-p_5)^2} f_c^3 \right) \quad (13)$$

$$\sigma_{R,C} = \sqrt{K/f_c} \left(f_c S_{f,C}^2(f) + \frac{2^{(-2n+7)}\pi^2}{9(1-p_5)^2} f_c^3 \right) \quad (14)$$

In order to function properly, the minimization algorithm must be able to identify the lower value of two power estimates when it varies one coefficient C_i by one LSB. If we define C_{i+} as a set of coefficients for which the i^{th} coefficient has been increased by one LSB, we can then express $P_{e,C}$, the probability of error:

$$P_{e,C} = P(R_C > R_{C_{i+}} | \mu_{R,C} < \mu_{R,C_{i+}}) \quad (15)$$

It can be shown [4] that:

$$P_{e,C} = \frac{1}{\sqrt{2\pi}} \int_{\alpha}^{\infty} e^{-z^2/2} dz, \alpha = \frac{\mu_{R,C_{i+}} - \mu_{R,C}}{\sqrt{\sigma_{R,C}^2 + \sigma_{R,C_{i+}}^2}} \quad (16)$$

The optimal lowpass filter cutoff frequency is the value that minimizes $P_{e,C}$. After a few manipulations, it is found that:

$$f_{c,\text{opt}} \simeq \frac{1-p_5}{2^{(-n+4)}\pi} \sqrt{3(S_{f,C}^2 + S_{f,C_{i+}}^2)} \quad (17)$$

An important point to remember is that eq. (17) gives the optimal lowpass filter cutoff frequency for a given point of operation. For example, for two given implementations, the one that has more severe DAC nonlinearity will have greater

Fig. 5. Probability of error as a function of K , the number of samples used for power estimation. The simulations have been performed with a second and fourth order Butterworth lowpass decimation filter with cutoff frequency $f_c = 3.5 \times 10^{-4}$. The set of coefficients C consists of all coefficients to their ideal values, except for C_5 which is offset by 8 LSB (9 LSB for C_{1+}). These results are from a Matlab system-level simulation.

folded noise at power up; the optimal value of $f_{c,\text{opt}}$ would be lower than $f_{c,\text{opt}}$ for the implementation with a better DAC. Since the DAC characteristics are not known a priori, the actual value of $f_{c,\text{opt}}$ will be chosen as a compromise which depends on the fabrication process mismatch statistics.

Fig. 5 shows the probability of error for the system under study. These results show the validity of eq. (16). It also shows that the use of a second order lowpass filter gives similar results to the brickwall filter assumption.

As stated above, when an error occurs, the minimization algorithm can temporarily go in the wrong search direction. As K increases, the error probability diminishes, but the time it takes to perform each power estimation increases. As a consequence, K should be fixed to a value that minimizes the overall average calibration time, which is tributary to the particular minimization algorithm used.

In this analysis, the decimation factor has been assumed to be $D = 1$. To reduce power dissipation, a higher decimation factor can be used. As long as its value results in negligible signal aliasing, eq. (16) and (17) are still valid, but the power estimator will only use K/D samples at each estimation.

REFERENCES

- [1] G. Gagnon and L. MacEachern, "Continuous compensation of binary-weighted DAC nonlinearities in bandpass delta-sigma modulators," *Accepted for Publication in IEEE ISCAS 2007*, 2007.
- [2] R. Schreier and G. C. Temes, *Understanding Delta-Sigma Data Converters*. Hoboken, NJ: John Wiley & Sons, 2005.
- [3] R. Schreier. (2004) The delta sigma toolbox version 7.1. [Online]. Available: <http://www.mathworks.com/matlabcentral/fileexchange>
- [4] W. Feller, *An Introduction to Probability Theory and Its Applications, Volume 2*. J. Wiley & Sons, 1971.

Appendix D

Deriving the Probability of Error

To ease up the reading of the thesis, eq. (5.1) was given directly. This appendix shows how this equation was derived.

Let A and B be two normally distributed random variables with mean values μ_A and μ_B , and standard deviations σ_A and σ_B .

The first step is to determine the probability P_{AB} that a single experiment results in $A > B$ even though $\mu_A < \mu_B$. This is illustrated in Fig. D.1. Mathematically,

$$P_{AB} = P(A > B | \mu_A < \mu_B) = P(A - B > 0) \quad (\text{D.1})$$

If we define $C = A - B$, from [61], C is also a normally distributed random variable with mean value:

$$\mu_C = \mu_A - \mu_B \quad (\text{D.2})$$

and standard deviation:

$$\sigma_C = \sqrt{\sigma_A^2 + \sigma_B^2} \quad (\text{D.3})$$

Figure D.1: Probability of error of two normally distributed variables.

Then,

$$P(A > B) = P(C > 0) \quad (\text{D.4})$$

Again, from [61]:

$$P(C > 0) = Q(-\mu_C/\sigma_C) \quad (\text{D.5})$$

where

$$Q(\alpha) = \frac{1}{\sqrt{2\pi}} \int_{\alpha}^{\infty} e^{-z^2/2} dz \quad (\text{D.6})$$

Then,

$$P_{AB} = P(A > B) = Q(\alpha), \quad \alpha = \frac{\mu_B - \mu_A}{\sqrt{\sigma_A^2 + \sigma_B^2}} \quad (\text{D.7})$$

The minimization algorithm needs to differentiate a sample from its two neighbors, i.e. when the coefficient is offset by -1 LSB and +1 LSB. If the difference in the average values are similar on each side, and if the standard deviations are

also similar¹, determining the probability of error is equivalent to repeating the experiment twice and having no error in both experiments. This is calculated using the binomial coefficients. For two experiments,

$$P_2 = 1 - (1 - P_1)^2 \quad (\text{D.8})$$

where P_2 is the probability of obtaining 0 success in 2 experiments when the probability of success for each experiment is P_1 (the probability of success being defined as the probability of making an error).

Combining eq. (D.7) and (D.8), and since $1 - Q(x) = Q(-x)$, the probability of error is found:

$$P_e \simeq 1 - (Q(\alpha))^2, \quad \alpha = \frac{\mu_A - \mu_B}{\sqrt{2\sigma^2}} \quad (\text{D.9})$$

where σ^2 is the variance of A or B , which, as stated above, should be equivalent.

Note that $Q(x)$ is obtained in Matlab using the command `1-normcdf(x)`.

¹Those conditions are valid when not near the optimum value.

Appendix E

Schematics

This appendix includes the schematics of the $\Delta\Sigma$ modulator IC and the schematics of the PCB to support it.

sah_3phases

The sampled value is the input voltage at the end of p2.
On vop1, the output is only valid slightly before the end of p1.
On vop2, the output is only valid slightly before the end of p2
vop1d is the same as vop1 with 1 full (p1+p2) clock delay
As soon as p1 (p2 for vop2) is 0, the output is no longer valid.

integrator_no_delayx2

Samples when p1 is high
 Also integrates when p1 is high
 The output is valid at the end of p1
 and during all of p2, i.e., the output
 is valid whenever p1 is low.

resonator1x2

resonator2x2

sum2x2

Samples on p1.

The output is available without any clock delay on p1.

Valid until the next rising edge of p1.

adc_comparators

`dac_l_sources_mult`

difamp_cmb

buffer

Samples on p1.

The output is available without any clock delay on p1.

Valid until the next rising edge of p1.

clkbuf

TG_sample_2

Bibliography

- [1] S. R. Norsworthy, R. Schreier, and G. C. Temes, *Delta-Sigma Data Converters: Theory, Design, and Simulation*. New York: IEEE Press, 1997.
- [2] R. Schreier and G. C. Temes, *Understanding Delta-Sigma Data Converters*. Hoboken, NJ: John Wiley & Sons, 2005.
- [3] R. T. Baird and T. S. Fiez, "A low oversampling ratio 14-b 500-khz $\Delta\Sigma$ ADC with a self-calibrated multibit DAC," *IEEE Trans. Circuits Syst.*, vol. 31, pp. 312–320, Mar. 1996.
- [4] M. R. Miller and C. S. Petrie, "A multibit sigma-delta ADC for multimode receivers," *IEEE Trans. Circuits Syst.*, vol. 38, pp. 475–482, Mar. 2003.
- [5] G. Gagnon and L. MacEachern, "Digital compensation of DAC mismatches in multibit delta-sigma ADCs," *Electronics Letters*, vol. 44, no. 12, pp. 721–722, May 2008.
- [6] G. Gagnon and L. MacEachern, "System-level design aspects of a DAC calibration technique for bandpass delta-sigma modulators," in *Proc. IEEE NEWCAS 2007*, Aug. 2007, pp. 1481–1484.
- [7] G. Gagnon and L. MacEachern, "Continuous compensation of binary-weighted DAC nonlinearities in bandpass delta-sigma modulators," in *Proc. IEEE ISCAS 2007*, May 2007, pp. 253–256.
- [8] J. A. Cherry and W. M. Snelgrove, *Continuous-Time Delta-Sigma Modulators for High-Speed A/D Conversion: Theory, Practice, and Fundamental Performance Limits*. Norwell, MA: Kluwer, 2000.
- [9] R. Schreier. (2004) The delta sigma toolbox version 7.1. [Online]. Available: <http://www.mathworks.com/matlabcentral/fileexchange>
- [10] J. G. Proakis and D. G. Manolakis, *Digital Signal Processing: Principles, Algorithms, and Applications*, 3rd ed. New Jersey: Prentice Hall, 1996.
- [11] D. A. Johns and K. Martin, *Analog Intergrated Circuit Design*. John Wiley & Sons, 1997.
- [12] K. C.-H. Chao, S. Nadeem, W. L. Lee, and C. G. Sodini, "A higher order topology for interpolative modulators for oversampling A/D converters," *IEEE Trans. Circuits Syst.*, vol. 37, pp. 309–318, Mar. 1990.
- [13] S. Jantzi, K. Martin, and A. Sedra, "Quadrature bandpass $\Delta\Sigma$ modulation for digital radio," *IEEE Trans. Circuits Syst.*, vol. 32, pp. 1935–1950, Dec. 1997.
- [14] P. Benabes, A. Gauthier, and D. Billet, "New wideband sigma-delta convertor," *IEE Electronic Letters*, vol. 29, no. 17, pp. 1575–1577, Aug. 1993.
- [15] R. Koch, B. Heise, F. Eckbauer, E. Engelhardt, J. A. Fisher, and F. Parzefall, "A 12-bit sigma-delta analog-to-digital converter with a 15-MHz clock rate," *IEEE Trans. Circuits Syst.*, vol. 21, pp. 1003–1010, Dec. 1986.

- [16] Y. Matsuya, K. Uchimura, A. Iwata, T. Kobayashi, and M. Ishikawa, "A 16b oversampling A/D conversion technology using triple integration noise shaping," in *Proc. IEEE ISSCC 1987*, Feb. 1987, pp. 48–49.
- [17] P. J. A. Naus, E. C. Dijkmans, E. F. Stikvoort, D. J. Holland, and W. Bradinal, "A CMOS stereo 16-bit D/A converter for digital audio," *IEEE Trans. Circuits Syst.*, vol. 22, pp. 390–395, June 1987.
- [18] M. Rebeschini, N. R. van Bavel, P. Rakers, R. Greene, J. Caldwell, and J. R. Haug, "A 16-b 160-kHz CMOS A/D converter using sigma-delta modulation," *IEEE Trans. Circuits Syst.*, vol. 25, pp. 431–440, Apr. 1990.
- [19] R. Schreier, "An empirical study of high-order single-bit delta-sigma modulators," *IEEE Trans. Circuits Syst. II*, vol. 40, pp. 461–466, Aug. 1993.
- [20] M. Safi-Harb and G. W. Roberts, "Low power delta-sigma modulator for ADSL applications in a low-voltage CMOS technology," *IEEE Trans. Circuits Syst. I*, vol. 52, pp. 2075–2089, Oct. 2005.
- [21] S. Lee and C. Cheng, "A low-voltage and low-power adaptive switched-current sigma-delta ADC for bio-acquisition microsystems," *IEEE Trans. Circuits Syst. I*, Accepted for future publication (2006).
- [22] T. Ndjountche, F.-L. Luo, and R. Unbehauen, "A high-frequency double-sampling second-order $\Delta\Sigma$ modulator," *IEEE Trans. Circuits Syst. II*, vol. 52, pp. 841–845, Dec. 2005.
- [23] Y. Cheng, C. Petrie, B. Nordick, and D. Comer, "Multibit delta-sigma modulator with two-step quantization and segmented DAC," *IEEE Trans. Circuits Syst. II*, vol. 53, pp. 848–852, Sept. 2006.
- [24] G. Bernardinis, F. Borghetti, V. Ferragina, A. Fornasari, U. Gatti, P. Malcovati, and F. Maloberti, "A wide-band 280-MHz four-path time-interleaved bandpass sigma-delta modulator," *IEEE Trans. Circuits Syst. I*, vol. 53, pp. 1423–1432, July 2006.
- [25] L. Dorrer, F. Kuttner, P. Greco, P. Torta, and T. Hartig, "A 3-mW 74-dB SNR 2-MHz continuous-time delta-sigma ADC with a tracking ADC quantizer in 0.13 μ m CMOS," *IEEE Trans. Circuits Syst.*, vol. 40, pp. 2416–2427, Dec. 2005.
- [26] K. Nam, S.-M. Lee, D. K. Su, and B. A. Wooley, "A low-voltage low-power sigma-delta modulator for broadband analog-to-digital conversion," *IEEE Trans. Circuits Syst.*, vol. 40, pp. 1855–1864, Sept. 2005.
- [27] A. Rusu, D. R. L. Gonzalez, M. Ismail, and H. Tenhunen, "The design of a low-distortion sigma-delta ADC for WLAN standards," in *Proc. IEEE ISSCC 2005*, vol. 1, July 2005, pp. 151–154.
- [28] R. Hezar and O. Altun, "Digitally tuned bandpass sigma delta modulator for wireless communications," in *Proc. IEEE ISCAS 2005*, vol. 4, May 2005, pp. 3668–3671.
- [29] R. Maurino and C. Papavassiliou, "Quadrature $\Sigma\Delta$ modulators with a dynamic element matching scheme," *IEEE Trans. Circuits Syst. II*, vol. 52, pp. 771–775, Nov. 2005.
- [30] M. Sarhang-Nejad and G. C. Temes, "A high-resolution multibit $\Sigma\Delta$ ADC with digital correction and relaxed amplifier requirements," *IEEE J. Solid-State Circuits*, vol. 28, pp. 648–660, June 1993.
- [31] S. Luschas and H.-S. Lee, "High-speed $\Sigma\Delta$ modulators with reduced timing jitter sensitivity," *IEEE Trans. Circuits Syst. II*, vol. 49, pp. 712–720, Nov. 2002.

- [32] M. W. Hauser and R. W. Brodersen, "Circuit and technology considerations for MOS delta-sigma A/D converters," in *Proc. IEEE ISCAS 1986*, 1986, pp. 1310–1315.
- [33] L. R. Carley, "A noise-shaping coder topology for 15+ bit converters," *IEEE J. Solid-State Circuits*, vol. 24, pp. 267–273, Apr. 1989.
- [34] R. T. Baird and T. S. Fiez, "Improved $\Delta\Sigma$ DAC linearity using data weighted averaging," in *Proc. IEEE ISCAS 1995*, vol. 1, 1995, pp. 13–16.
- [35] I. Fujimori, L. Longo, A. Hairapetian, K. Seiyama, S. Kosic, J. Cao, and S.-I. Chan, "A 90 dB SNR, 2.5 MHz output rate ADC using cascaded multibit $\Delta\Sigma$ modulation at 8x oversampling ratio," in *Proc. IEEE ISSCC 2000*, 2000, pp. 338–339.
- [36] A. A. Hamoui and K. W. Martin, "High-order multibit modulators and pseudo data-weighted-averaging in low-oversampling $\Delta\Sigma$ ADCs for broad-band applications," *IEEE Trans. Circuits Syst. I*, vol. 51, pp. 72–85, Jan. 2004.
- [37] Y.-I. Park, S. Karthikeyan, W. M. Koe, Z. Jiang, and T.-C. Tan, "A 16-bit, 5MHz multi-bit sigma-delta ADC using adaptively randomized DWA," in *Proc. IEEE Custom Integrated Circuits Conference*, 2003, pp. 115–118.
- [38] M. Vadipour, "Techniques for preventing tonal behavior of data weighted averaging algorithm in $\Sigma - \Delta$ modulators," *IEEE Trans. Circuits Syst. II*, vol. 47, pp. 1137–1144, Nov. 2000.
- [39] A. Gupta, E. Sanchez-Sinencio, S. Karthikeyan, W. M. Koe, and Y.-I. Park, "Second order dynamic element matching technique for low oversampling delta sigma adc," in *Proc. IEEE ISCAS 2006*, May 2006, pp. 2973–2976.
- [40] R. Schreier and B. Zhang, "Noise-shaped multibit D/A convertor employing unit elements," *Electron. Lett.*, vol. 31, pp. 1712–1713, Sept. 1995.
- [41] L. Zhimin and T. S. Fiez, "Dynamic element matching in low oversampling delta sigma ADCs," in *Proc. IEEE ISCAS 2002*, vol. 4, 2002, pp. 683–686.
- [42] T. Ueno, A. Yasuda, T. Yamaji, and T. Itakura, "A fourth-order bandpass $\Delta - \Sigma$ modulator using second-order bandpass noise-shaping dynamic element matching," *IEEE J. Solid-State Circuits*, vol. 37, pp. 809–816, July 2002.
- [43] I. I. Galton, "Spectral shaping of circuit errors in digital-to-analog converters," *IEEE Trans. Circuits Syst. II*, vol. 44, pp. 808–817, Oct. 1997.
- [44] E. N. Aghdam and P. Benabes, "Higher order dynamic element matching by shortened tree-structure in delta-sigma modulators," in *Proc. IEEE ISCAS 2005*, vol. 1, 2005, pp. 201–204.
- [45] T. Cataltepe, A. Kramer, L. Larson, G. Temes, and R. Walden, "Digitally corrected multibit $\Sigma\Delta$ data converters," in *Proc. IEEE ISCAS 1989*, vol. 1, May 1989, pp. 647–650.
- [46] C. Petrie and M. Miller, "A background calibration technique for multibit delta-sigma modulators," in *Proc. IEEE ISCAS 2000*, vol. 2, May 2000, pp. 29–32.
- [47] D. W. J. Groeneveld, H. J. Schouwenaars, H. A. H. Termeer, and C. A. A. Bastiaansen, "A self-calibration technique for monolithic high-resolution D/A converters," *IEEE Trans. Circuits Syst.*, vol. 24, pp. 1517–1522, Dec. 1989.
- [48] A. Fornasari, F. Borghetti, P. Malcovati, and F. Maloberti, "On-line calibration and digital correction of multi-bit sigma-delta modulators," in *2005 Symposium on VLSI Circuits Digest of Technical Papers*, 2005, pp. 184–187.

- [49] X. S. Wang, "Characterization and digital correction of multi-bit delta-sigma modulators," Master's thesis, Carleton University, Ottawa, 1999.
- [50] X. Wang, P. Kiss, U. Moon, J. Steensgaard, and G. Temes, "Digital estimation and correction of dac errors in multibit sigma-delta adcs," *Electronics Letters*, vol. 37, no. 7, pp. 414–415, Mar 2001.
- [51] J. Arias, P. Kiss, V. Boccuzzi, L. Quintanilla, L. Enriquez, J. Vicente, D. Bisbal, J. Pablo, and J. Barbolla, "Nonlinearity correction for multibit delta-sigma dacs," *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 52, no. 6, pp. 1033–1041, June 2005.
- [52] J. Chen and Y. P. Xu, "A novel noise-shaping dac for multi-bit sigma-delta modulator," *Circuits and Systems II: Express Briefs, IEEE Transactions on*, vol. 53, no. 5, pp. 344–348, May 2006.
- [53] Z. Cao and S. Yan, "Robust analogue background calibration technique for multi-bit switched capacitor dacs," *Electronics Letters*, vol. 42, no. 11, pp. 618–620, May 2006.
- [54] D.-H. Lee and T.-H. Kuo, "Advancing data weighted averaging technique for multi-bit sigma-delta modulators," *Circuits and Systems II: Express Briefs, IEEE Transactions on*, vol. 54, no. 10, pp. 838–842, Oct. 2007.
- [55] L. Samid, M. Ortmanns, Y. Manoli, and F. Gerfers, "A new kind of low-power multibit third order continuous-time lowpass $\Sigma\Delta$ modulator," in *Proc. IEEE ISCAS 2002*, vol. 3, 2002, pp. 293–296.
- [56] L. Samid and Y. Manoli, "A multibit continuous time sigma delta modulator with successive-approximation quantizer," in *Proc. IEEE ISCAS 2006*, 2006, pp. 2965–2968.
- [57] P. Venkataraman, *Applied Optimization with Matlab Programming*. New York: John Wiley & Sons, 2002.
- [58] R. J. Baker, *CMOS: Circuit Design, Layout, and Simulation*, 2nd ed. New Jersey: J. Wiley & Sons, 2005.
- [59] MOSIS. (2008) IBM 0.13 micron 8RF-DM process. [Online]. Available: <http://www.mosis.com/ibm/8rf-dm/>
- [60] *CMOS8RF (CMRF8SF) Design Manual*. Confidential. Mixed Signal Technology Development, IBM Microelectronics Division, 2007.
- [61] W. Feller, *An Introduction to Probability Theory and Its Applications*. J. Wiley & Sons, 1971, vol. 2.