High-Performance Data Converters

by

Jesper Steensgaard-Madsen

(Steensgaard@ieee.org)

A thesis submitted in partial fulfillment of
the requirementments for the Ph.D. degree

The Technical University of Denmark
Department of Information Technology
DK-2800, Lyngby Denmark

January 20, 1999
(Revised March 8, 1999)

© by Jesper Steensgaard-Madsen, 1999

Copyright ©
by Jesper Steensgaard-Madsen, 1999
All rights reserved

Abstract

Novel techniques for multi-bit oversampled data conversion are described. State-of-the-art oversampled data converters are analyzed, leading to the conclusion that their performance is limited mainly by low-resolution signal representation. To increase the resolution, high-performance, high-resolution internal D/A converters are required. Unit-element mismatch-shaping D/A converters are analyzed, and the concept of mismatch-shaping is generalized to include scaled-element D/A converters. Several types of scaled-element mismatch-shaping D/A converters are proposed. Simulations show that, when implemented in a standard CMOS technology, they can be designed to yield 100 dB performance at 10 times oversampling.

The proposed scaled-element mismatch-shaping D/A converters are well suited for use as the feedback stage in oversampled delta-sigma quantizers. It is, however, not easy to make full use of their potential, because that requires a high-resolution loop quantizer which introduces only a small delay. Generally, it is not acceptable to design the loop quantizer as a high-resolution flash quantizer because they require a large chip area and high power consumption. Pipeline techniques are proposed to circumvent this problem. This way, the delta-sigma quantizer's feedback signal is obtained by a multiple-stage quantization, where the loop quantizer (low-resolution and minimum-delay) implements only the last-stage quantization. Hence, high-speed, high-resolution delta-sigma quantization is feasible without using complex circuitry.

An improved version of the MASH topology is also proposed. A delta-sigma quantizer is used to quantize the input signal into an oversampled digital representation of low-to-moderate resolution. The delta-

sigma quantizer's truncation error is estimated either directly, or as the first-order difference of the output signal from the loop filter's first integrator stage. This technique avoids the need for accurate matching of analog and digital filters that characterizes the MASH topology, and it preserves the signal-band suppression of quantization errors. Simulations show that quantizers of this type can yield 100 dB performance at 10 times oversampling. There are no requirements for high-resolution flash quantizers or other hard-to-implement circuitry.

Acknowledgements

The author wishes to acknowledge several people for their contributions to this work.

In a category by himself, Professor Gabor C. Temes has helped me in countless ways during the almost two years I have been fortunate enough to work with him. I thank him for his friendship, encouragement, advice, enthusiasm, ideas, feedback, editing, our discussions, and for providing an environment where electrical engineering can be practiced in a fun and rewarding way. He truly deserves the “IEEE Graduate Teaching Award,” which was awarded to him in 1998. He has my highest respect and appreciation in every aspect.

At Oregon State University I have worked with many wonderful people with whom I have had fruitful discussions. They have included Professor Un-Ku Moon, Professor Richard Schreier, Luis Hernandez, Andreas Wiesbauer, Paul Ferguson, Yunteng Huang, Bo Wang, Tao Sun, and many more.

I particularly wish to thank the President of MEAD Microelectronics Inc. of Switzerland, Vlado Valence, and Gabor and Ibi Temes, from U.S. MEAD, for inviting me to participate in the outstanding courses in electrical engineering which they arrange in Switzerland and in the United States. I believe I have obtained some of my most useful and practical information from these courses, and I am very grateful for that. I need to thank the outstanding course lecturers who teach these courses, many of whom I have been fortunate to encounter in private and fruitful discussions. In arbitrary order, I particularly wish to thank Bob Adams, Eric Vittoz, Paul Brokaw, Tim Schmerbeck, Christian Enz, Todd Brooks, Paul Ferguson, Bob Blauschild, Ian Galton, Berrie Gilbert, Stephen Jantzi and several others.

I also extend my thanks to the design team at MEAD Microelectronics Inc., particularly Fabien and Phillip Duval.

For making my life pleasant in many ways, I wish to thank my brother, Bjarne, his wife, Tamara, and my girlfriend, Patrice. Also, I wish to thank Ms. Ibi Temes for her gracious hospitality at the Temes' home on several occasions.

Finally, I wish to thank my supervisor, Erik Bruun, and the Danish educational system for its financial support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Modern society relies on signal processing. It is applied in communication equipment, medical devices, automated production facilities, computers, weapons, navigation equipment, tools and toys, etc.. Most human-designed signal processing is performed by electronic circuits, and the range of applications is broadened as these circuits are perfected and their costs reduced.

The majority of the signals of interest are found in the world that surrounds us, whether it relates to monitoring a heart or guiding a missile. The first step performed by a signal-processing system is to convert a considered signal into a form that can be processed by an electronic circuit. Sometimes a dedicated electro-mechanical system (called a *sensor*) will be required to sense the signal and convert it into a voltage, charge, or current signal, and sometimes the signal is readily available in one of these forms. An electronic circuit will then process the electric signal in a specified way, and the outcome will often be applied to a nonelectronic task, such as displaying an image of the heart, or adjusting the missile's direction of flight.

The signal processing that needs to be performed can vary from very simple operations (e.g., amplification) to extremely complex ones involving computation of several parameters, such as standard deviation, spectral composition, correlation coefficients, etc.. A fundamental property of analog electric signal processing is that each operation will be associated with a degradation of the signal-to-noise ratio (SNR).

Hence, if substantial analog signal processing (ASP) is performed, stochastic artifacts (noise) will accumulate, and the resulting signal may not represent the desired signal with the required significance. Furthermore, the accuracy of ASP is inherently limited; the linearity of supposedly linear operations is not ideal, multiplication of signals is poorly implemented, etc..

A wide range of applications require substantial amounts of highly accurate signal processing. High-accuracy electronic signal processing can generally be implemented only when the signals are represented in digital form. In digital form, signals can be processed with arbitrary resolution and accuracy and without noticeably degrading the SNR. However, many thousands of transistors are required to implement a circuit that performs only simple digital signal processing (DSP). Hence, the feasibility of DSP is mainly a matter of circuit density and power consumption.

Thanks to CMOS integrated circuit technology, DSP has experienced explosive growth during the last couple of decades. CMOS technology has become widely available and it is characterized by an outstanding cost-to-performance ratio which is improved steadily (Moore's Law). CMOS technology's many advantages include its low cost, high speed, high circuit density[1], low power consumption per operation, and the availability of software for the semi-automated design of DSP circuits. The cost and efficiency of CMOS-based DSP is actually so competitive that it is often used for the implementation of even simple signal processing systems where ASP could potentially be used instead. The fields that remain dominated by ASP include high-frequency (radio) signal processing and applications that are characterized by low resolution and a high degree of parallelism (for example, finger-print sensors).

Data converters are the missing link needed for the implementation of a DSP-based electronic circuit. Although digital signals can be processed with arbitrary resolution and accuracy, the system's overall performance cannot exceed the resolution or accuracy by which the considered analog signals can be converted into digital form (A/D conversion), or by which the processed digital signal can be reconverted into analog form (D/A conversion). Obviously, data conversion is not a new discipline in circuit design, but huge industrial investments are still being made, and there is a tremendous research activity continuing in this technical field. This clearly shows that there is a great demand for CMOS-based data converters that combine high speed, high resolution, and low cost.

[1] Several hundred million transistors can be employed in the same circuit.

1.1 The Class of Data Converters Considered

This work focuses almost exclusively on delta-sigma modulation as the chosen technique for A/D and D/A conversion. Delta-sigma ($\Delta\Sigma$) converters have gained popularity during the last decade because they trade an increased requirement for DSP for a relaxed requirement for high-performance analog circuit blocks. Single-bit $\Delta\Sigma$ converters have usually been preferred because they avoid the requirement for accurately matched electrical parameters that characterize most other high-resolution data converters.

Delta-Sigma Modulation. Any signal is uniquely characterized by its spectral composition. This work is dedicated to the large range of applications that characterize signals by their spectral composition in only a selected frequency band (the signal band). Nyquist's sampling theorem states that the maximum bandwidth that can be represented by a uniformly-sampled digital signal is half the signal's sampling frequency (the Nyquist bandwidth), in which case there is a one-to-one correspondence between the signal's spectral composition and the value of its samples. However, if the signal is oversampled, i.e., if it is characterized by its spectral composition in a signal band which is narrower than the Nyquist bandwidth, then the value of each sample is not uniquely defined, and the flexibility[2] can be used (for example) to truncate the signal's samples to values from a finite set of selected values. More precisely, there is a tradeoff between the signal's oversampling ratio[3] (OSR) and the tolerance allowed in each sample's value. The samples' truncation errors must be correlated to preserve the signal-band spectral composition, and the process somewhat resembles interpolation. The tradeoff between resolution and bandwidth is considered good. For example, the same signal can be represented by truncation to 65,536 uniformly-spaced values using only negligible oversampling, or by truncation to only 2 values using 32 times oversampling[4].

[2]Two signals are considered to be equivalent if their spectral composition in the signal band is identical (or if the difference is smaller than a chosen threshold, say -100 dB full scale). The flexibility reflects that spectral variations outside the signal band are allowed.

[3]The Nyquist bandwidth divided by the signal's bandwidth.

[4]The minimum required oversampling ratio expressed as a function of the signal and the selected set of truncation levels is not known, but would probably be of little practical interest. The numbers provided herein characterize circuits that can perform the discussed translation from one representation to another.

In essence, $\Delta\Sigma$ modulators are circuits that can translate a signal between representations of different resolutions and sampling rates.

Single-Bit Delta-Sigma Converters. Single-bit (two-level) signal representation is useful because it facilitates linear A/D and D/A conversion without relying on accurate matching of electrical parameters [1]. This is why the technique has become popular. Although signals can be $\Delta\Sigma$ modulated into a two-level representation using only 32 times oversampling, there are several practical reasons why this is rarely done. Usually, oversampling ratios in the order of 128 are used, which, unfortunately, considerably constrains the system's bandwidth because the maximum sampling frequency cannot be increased arbitrarily. Single-bit $\Delta\Sigma$ converters have, therefore, been used mainly for audio and other high-performance applications which have a fairly low bandwidth.

Multi-Bit Delta-Sigma Converters. A $\Delta\Sigma$ data converter's linearity is constrained by the linearity of a D/A converter employed internally. The inherent linearity of time-invariant single-bit D/A converters (DACs) is the key to single-bit $\Delta\Sigma$ converters' superb linearity.

Multi-bit $\Delta\Sigma$ converters can operate at a substantially lower OSR than their single-bit counterparts, even if the signal is represented by only a few bits of resolution [2]. They are, therefore, more suitable for wide-bandwidth data conversion, which is required by a wide range of applications. Unfortunately, a DAC's full-scale linearity is essentially independent of its resolution (except for single-bit DACs), hence $\Delta\Sigma$ modulation does not directly offer any advantages for non-single-bit data converters. However, it is indeed simpler to calibrate a low-resolution DAC than it is to calibrate a high-resolution one, and multi-bit $\Delta\Sigma$ modulation has successfully been used for calibrated systems [1, 3, 4].

The introduction of mismatch-shaping DACs marked a major breakthrough in multi-bit $\Delta\Sigma$ data conversion. The fundamental principle employed by these DACs is that they are allowed to produce inaccurate analog output values, as long as they interpolate between the errors and the output signal's signal-band spectral composition remains intact. This operation is very similar to $\Delta\Sigma$ modulation.

The basic requirement for mismatch-shaping DACs is that they must be able to interpolate between the mismatch errors without knowing the actual value of each error. This operation can, for example, be ob-

tained when using a digital state machine to control a unit-element DAC[5] [1,2,5–18]. The complexity of unit-element mismatch-shaping DACs increases considerably with their resolution, hence the technique is suitable only for DACs with a resolution of up to (say) 6 bits. In other words, a $\Delta\Sigma$ modulator is required to reduce the signal's resolution to a level where a mismatch-shaping DAC can be implemented using only circuitry of reasonable complexity.

High-Resolution Mismatch-Shaping Data Converters. Through the development of digital state machines that can implement scaled-element DACs[6] with mismatch shaping, this work extends the possibilities for high-resolution data conversion. The circuit complexity of the proposed state machines is low, and the mismatch-shaping DACs' resolution can be made arbitrarily high. Because the signal is not interpolated to a low-resolution representation, large spectral components outside the signal band (representing the truncation error) will not occur, and the specifications of the filters that are normally required to remove such spectral components can, therefore, be relaxed considerably. Hence, the proposed techniques facilitate the implementation of high-speed D/A converters that are characterized by an unpreceded simplicity and level of performance (100 dB performance at 10 times oversampling is feasible using an inexpensive standard CMOS technology with no post-production calibration).

The proposed mismatch-shaping DACs need not cause substantial delay, hence they are well suited for use in multi-bit $\Delta\Sigma$ A/D converters. Usually, the D/A converter employed internally in $\Delta\Sigma$ A/D converters has been the limiting factor for the overall performance, but when a scaled-element mismatch-shaping DAC is used for this purpose, the performance can be improved to the level where it is only the complexity of the internal loop quantizer that will limit the performance. This work also proposes techniques that solve this complexity problem. Using the proposed techniques, the achievable performance reaches a level where only device noise, clock jitter, and other unavoidable effects will constrain the performance.

[5] Unit-element DACs generate the analog output signal by adding analog sources of the same nominal value.

[6] Scaled-element DACs generate the analog output signal by adding analog sources of scaled nominal values. Binary-weighted DACs (for which the analog sources are proportional to $1, 2, 4, 8, \ldots$) are an example which illustrates that the resolution of scaled-element DACs can be vastly higher than the resolution of unit-element DACs based on the same number of analog sources.

1.2 The Structure of This Thesis

Following this Introduction, Chapter 2 begins by defining the class of signals considered and the main mathematical tool used to characterize and analyze them (the Fourier transform). Methods for estimating a signal's spectral composition are also discussed. The chapter includes only material that should be common knowledge for all trained electrical engineers, so it may be considered as optional reading.

Chapter 3 is a discussion of the basic aspects of data conversion. It discusses the basic steps and the topologies in which most A/D and D/A converters are implemented (but it is not comprehensive). It provides several definitions, and it points out some of the many effects that are likely to limit a data converter's performance. The reader is advised to be familiar with this material.

Chapter 4 is an overview of state-of-the-art $\Delta\Sigma$ quantizers, and it includes a thorough discussion of mismatch-shaping unit-element DACs. It outlines the advantages of multi-bit $\Delta\Sigma$ modulation, and it points out the drawbacks of the so-called MASH quantizers. It also includes an evaluation of the best-case noise performance, which ultimately will limit the overall performance. Even the reader with good insight in $\Delta\Sigma$ data conversion is advised to read this chapter carefully.

Chapters 5, 6, 7, 8, and 9 constitute the main part of this work, and at least 90% of the material contained in them is believed to be novel.

Chapter 5 is a discussion of how dynamic errors can be avoided in current-mode DACs. Current-mode DACs are important because they facilitate the implementation of data converters with a very low noise floor (discussed in Chapter 4).

Chapter 6 is a discussion of idle tones in mismatch-shaping DACs. Idle tones are a very unpleasant (and hence important) phenomenon which has received little attention in the open literature. Several techniques to prevent idle tones are proposed.

Chapter 7 is a discussion of the design of scaled-element mismatch-shaping DACs and possibly the most important part of this thesis. Several techniques are proposed.

Chapter 8 is a discussion of what is required to make full use of the proposed scaled-element mismatch-shaping DACs when they are used for the implementation of high-resolution $\Delta\Sigma$ quantizers. Pipeline

techniques are proposed as a way to avoid the need for high-resolution flash quantizers.

Chapter 9 takes a different approach for the design of high-performance $\Delta\Sigma$ quantizers. The technique is based on a multiple-stage quantization, which somewhat resembles MASH-topology $\Delta\Sigma$ quantizers. The major advantage of the proposed technique (as opposed to MASH quantizers) is that it does *not* rely on accurate matching of analog and digital filters, therefore, high-performance low-complexity quantizers can be implemented robustly.

1.3 Intellectual Property Rights

This serves as a public notice that several U.S. and international patents are pending for substantial parts of this work. The reader is advised to contact the author (Steensgaard@ieee.org) for licensing information before employing the discussed techniques in commercial products.

Chapter 2

Characterization of Signals

This chapter will define the class of signals that are relevant for this work. The properties of analog and digital signals are discussed, and an important distinction between continuous-time and discrete-time signals is made.

Although the considered signals are defined as functions of a time index, they are often better analyzed in the frequency domain. A mathematical tool, the Fourier Transformation, is used as the fundamental link between the time domain and the frequency domain. As this technique is assumed to be common knowledge for all properly-trained electrical engineers, the main results will merely be summarized. Unfortunately, the process of estimating a signal's spectral composition on the basis of its time-domain representation in a finite-duration period of time is not always well understood. Because this work makes extensive use of such estimates, this process and its tradeoffs will be discussed in some detail.

2.1 Time-Domain Representation of Signals

A signal's properties can be described in many ways – too many to be discussed in this context. In this thesis, signals are assumed to be defined by their relation to the time variable.

2.1.1 Analog Signals

An analog signal shall mean a physical phenomenon described by a single-variable measure, which is a continuous function of time.

A typical example of an analog signal is the electrostatic potential (voltage) at a specified location relative to a selected reference location (ground). Another typical example is current, which is defined as the first derivative of the charge passing through a specified oriented surface[1]. Analog signals can, in principle, be measures of almost anything: angle, velocity, acceleration, temperature, energy, reflection, weight, resistance, capacitance, inductance, etc.. However, notice that because it can be described only by at least three parameters, color is not considered to be an analog signal; whereas, the light intensity at a specified wave length *is* an analog signal.

Analog signals will (as usual) be characterized by the measure, rather than by the physical phenomenon that the measure evaluates. Thus, analog signals are simply continuous mathematical functions of a variable called time. The provided circuit examples will, however, use voltage, current, and charge as examples of analog signals.

Continuous-Time Analog Signals. A continuous-time analog signal is an analog signal that is defined and evaluated with respect to a continuum of time values.

Discrete-Time Analog Signals. Although all analog signals, or at least the described physical phenomena, are defined for a continuum of time values; some analog signals are evaluated only at discrete time instances. Such analog signals are called discrete-time signals. Notice that whether an analog signal should be characterized as a continuous-time or a discrete-time signal depends only on the application to which the signal is applied, hence it is not a property that can be extracted from the signal itself.

[1] Strictly speaking, in the classical description of charge being discretely distributed in space, current is not an analog signal according to the above definition (as current would be a sequence of impulses, and hence not a continuous function of time). However, as this thesis addresses macroscopic problems, such inconsistencies will be allowed without further notice.

Although not fully comprehensive, this thesis will only consider discrete-time signals that are uniformly-sampled. More precisely, a discrete-time analog signal $a_d(k)$ is in this thesis defined by an analog signal's $a_c(t), t \in R$, values at discrete time instances, which are equidistantly spaced by a chosen constant time unit T_s. Thus, a discrete-time analog signal can be described as

$$a_d(k) = a_c(kT_s), \qquad k \in Z = \{\dots, -3, -2, -1, 0, 1, 2, 3, \dots\} \tag{2.1}$$

The chosen time unit T_s is called the *sampling period*, and the reciprocal of the sampling period is called the *sampling frequency* $f_s = 1/T_s$.

Switched-capacitor (SC) circuits are a typical example of circuits that evaluate analog signals in discrete time. SC circuits consist of one or more cells designed to settle towards a stable and well-defined equilibrium. By evaluating the signals only at time instances when the cells have settled to values very close to their equilibria, the signal processing provided by these circuits can be made accurate[2].

2.1.2 Digital Signals

This thesis will only consider a subset of the large class of signals that generally are considered to be digital signals. In the following, a digital signal shall mean a member of this subset.

Digital signals are always discrete-time signals, and their characteristics are, from a mathematical point of view, equivalent to those of discrete-time analog signals. The difference between a digital and a discrete-time-analog signal is that a digital signal is a sequence of numbers, whereas a discrete-time-analog signal is a sequence of samples/evaluations of some physical phenomenon. In other words, a digital signal $d(k)$ is simply a sequence of numeric values that are a function of an integer variable k, which is considered to represent a sequence of equidistant time values $t = kT_s$.

In a physical system, a digital signal will be represented by a set of one or more analog signals (often called *bits* or *bit signals*) that, when evaluated jointly at $t = kT_s$, are considered to represent one of a finite number of possible states (called *codes*), each of which are considered to represent a numeric

[2] In this way, the performance of SC circuits can be made insensitive to nonlinear settling effects etc. , which can cause substantial errors in circuits operating on continuous-time signals.

value. A fundamental property of all digital systems is that each code is easily distinguishable from the other possible codes, hence the correct numeric value will be represented/detected even if the bit signals are subject to a considerable amount of noise. The *noise margin* is defined as the level of noise which can be tolerated at a given (very high) stochastic significance level of the representation of the codes. Usually, each bit signal is considered to represent one of only two possible states, high (1) or low (0), therefore the noise margin is typically very good. Using this practice, a collection of N bit signals can represent up to 2^N different codes. If P evaluations of the individual bit signals are used to represent each code (serial data representation), then N bit signals can represent as many as 2^{N+P} different codes. Using a sufficient number of bit signals/evaluations, digital signals with arbitrary high dynamic range can easily (and robustly) be represented in a digital system.

2.2 Frequency-Domain Representation of Signals

As an alternative to the time-domain representation, continuous-time as well as discrete-time signals can be described in the frequency domain. The two representations are complementary because some of a signal's properties are best described/analyzed in the time domain, whereas others are best described/analyzed in the frequency domain.

It is of particular importance that the frequency-domain representation of signals allows for the definition of the *signal-band* part of a signal, which, in essence, is the only part of the signal that is important for a given application.

The Fourier Transformation will be used as the fundamental link between the two domains. In essence, the Fourier transform of a signal represents the coefficients and angles in a uniquely-defined linear combination of sinusoids in a continuum of frequencies, this linear combination being equal to the signal.

2.2.1 Fourier Transformation of Continuous-Time Signals

The Fourier transform $G(f)$ of a continuous-time signal $g(t)$ is defined mathematically as in (2.2), where convergence is assumed.

$$G(f) = \int_{-\infty}^{\infty} g(t)e^{-j2\pi ft}dt \tag{2.2}$$

The inverse relation, the *Inverse Fourier Transformation*, describes that $G(f)$ simply represents the complex coefficients to the set of signals $e^{j2\pi ft}, f \in R$

$$g(t) = \int_{-\infty}^{\infty} G(f)e^{j2\pi ft}df \tag{2.3}$$

Because $e^{j2\pi ft}$ is an orthogonal basis[3], we can define the signal's *energy E_{ab} in the frequency band* $0 < f_a < f < f_b$ as

$$E_{ab}[g(t)] = 2\int_{f_a}^{f_b} |G(f)|^2 df \tag{2.4}$$

Parseval's Theorem (2.5) is a special case

$$E[g(t)] = \int_{-\infty}^{\infty} g^2(t)dt = \int_{-\infty}^{\infty} |G(f)|^2 df \tag{2.5}$$

The Fourier transformation will be denoted by $\mathcal{F}\{\cdot\}$, whereas the Fourier transformed signal will be denoted by capitalization of the time-domain symbol and change of the argument from t to f, i.e. $X(f) = \mathcal{F}\{x(t)\}$. A signal and its Fourier transform will be denoted as $x(t) \leftrightarrow X(f)$.

Using the Fourier Transformation. Probably the most powerful feature of the Fourier Transformation is that when a signal $g(t) \leftrightarrow G(f)$ is applied as input to a (settled and stable) linear system with the impulse response $h(t) \leftrightarrow H(f)$, the output $y(t)$ is described by

$$y(t) = g(t) * h(t) = \int_{-\infty}^{\infty} g(\lambda)h(t-\lambda)d\lambda \leftrightarrow Y(f) = G(f)H(f) \tag{2.6}$$

The inverse property is used less often, but it is also important

$$x(t)y(t) \leftrightarrow X(f) * Y(f) = \int_{-\infty}^{\infty} X(\lambda)Y(f-\lambda)d\lambda \tag{2.7}$$

[3]With respect to the scalar product calculating the average value of the product of two functions.

2.2.2 Fourier Transformation of Discrete-Time Signals

In reality, it is the same definition of the Fourier Transformation, Equation (2.2), which is applied to continuous-time as well as discrete-time signals. For this to make sense, it is necessary to define a continuous-time equivalent $g_{\text{eqv}}(t)$ of the discrete-time signal $g_d(k)$, for which the Fourier transformed can be calculated using the definition.

More precisely, the Fourier transform $\mathcal{F}_d\{\cdot\}$ of a discrete-time signal $g_d(k)$, sampled with the sampling period T_s, is defined as

$$G_d(f) = \mathcal{F}_d\{g_d(k)\} = \mathcal{F}\{g_{\text{eqv}}(t)\} \tag{2.8}$$

The discrete-time to continuous-time (DT/CT) transformation to be applied in (2.8) is defined as

$$g_{\text{eqv}}(t) = T_s \sum_{k \in Z} g_d(k)\delta(t - kT_s) \tag{2.9}$$

The equivalent continuous-time signal $g_{\text{eqv}}(t)$ is thus a sequence of impulses scaled according to $g_d(k)$, and occurring at the respective sampling instances.

Notice that $g_{\text{eqv}}(t)$ is a mathematical abstraction, which does not represent a real-world analog signal. The DT/CT conversion (2.9) is illustrated in Figure 2.1, which also shows the DT/CT conversion (3.21) that, in general, is approximated in real-world implementations (discussed later).

Calculating the Fourier Transformed. An advantage of the above definition of the Fourier Transformation of discrete-time signals is that it is simple to calculate the Fourier transform $G_d(f)$ of the sequence $g_d(k)$ that is sampled from a continuous-time signal $g(t)$, for which the Fourier transform $G(f)$ is known. The simple relation is

$$G_d(f) = \sum_{n \in Z} G(f - nf_s) \tag{2.10}$$

The relation (2.10) is shown graphically in Figure 2.2. At the top is shown the magnitude of the assumed Fourier spectrum $|G(f)|$ of the continuous-time signal $g(t)$. In the center are shown a few (for $n \in \{-2, -1, 0, 1, 2\}$) of the infinitely many spectra summed in Equation (2.10). At the bottom is shown the magnitude of the sum $|G_d(f)|$.

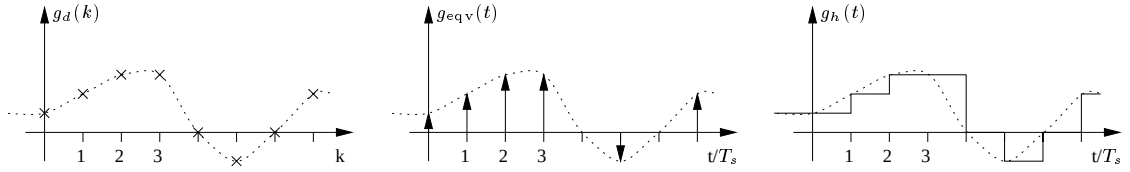


Figure 2.1: Two DT/CT conversions that are commonly used for signal analysis.

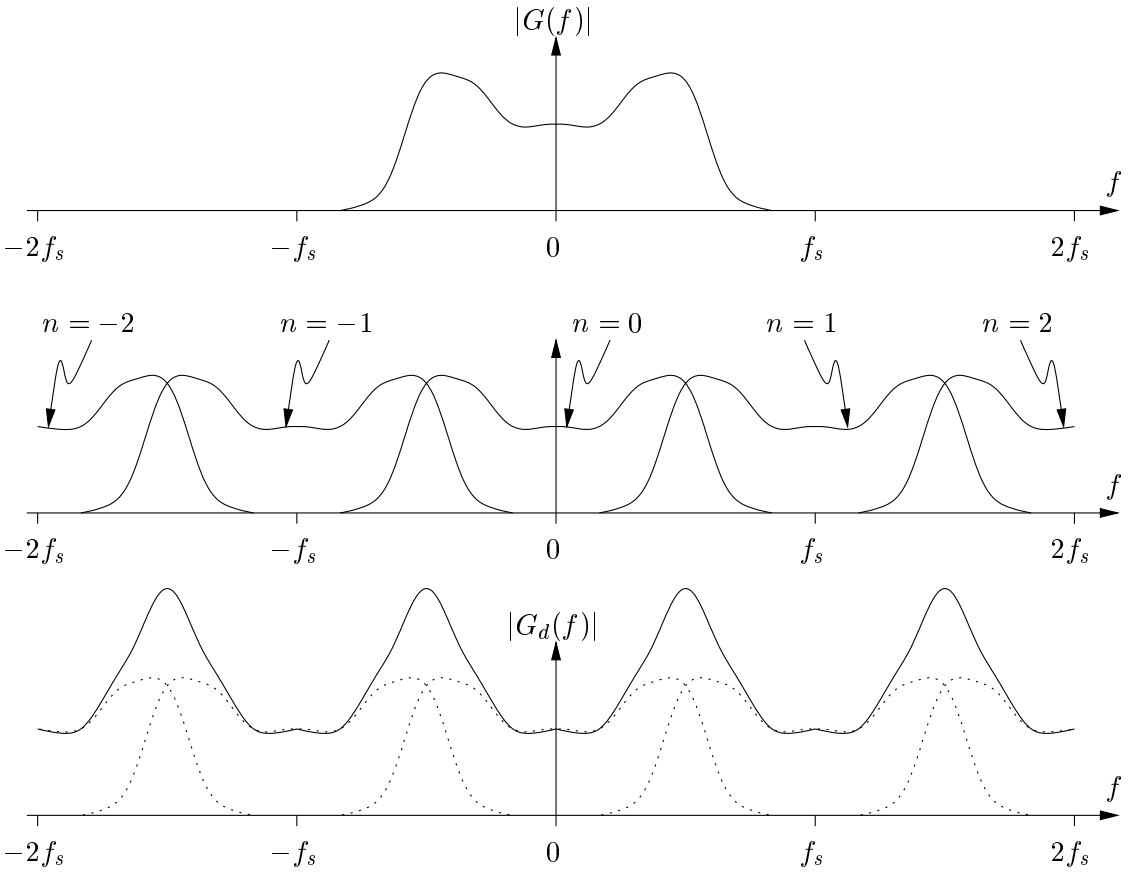


Figure 2.2: A graphic interpretation of equation 2.10 illustrating aliasing.

According to Parseval's Theorem (2.5), the energy of $g_{\mathrm{eqv}}(t)$ is infinite, which is in agreement with the signal including impulses.

Because the Fourier Transformation for discrete-time signals is based on the Fourier Transformation for continuous-time signals, Equations (2.6) and (2.7) have trivial generalizations for discrete-time signals.

2.2.3 Definition of the Signal Band

The Fourier Transformation is, in principle, just a mathematical technique to separate a signal into trigonometric functions, but it should be understood that its scope reaches far beyond the mathematical level.

Several applications evaluate an applied signal by its spectral composition, rather than by its waveform or derivatives at a given time (i.e. , by its time-domain representation). For example, the human ear is a highly-sophisticated electro-mechanical spectrum analyzer [19] [20], which is able to detect the spectral composition of an air-pressure signal (sound) in the frequency range from about 20 Hz to 20 kHz. Although most sensors are much less delicate than the human ear, it is invariably the case that many applications are indifferent to even very significant variations in the applied signal's spectral composition outside a certain frequency range. The frequency range in which the signal is evaluated is called the application's *signal band*.

Notice that the signal band is defined with respect to the application to which the signal is applied, and hence it is not a property that can be extracted from the signal itself. The signal band can be very different from application to application. Seismological detectors are, for example, often designed with a signal band from 0 Hz to (say) 5 Hz. Audio applications have somewhat wider signal bands (20 Hz to 20 kHz), whereas video applications and high-speed modems have signal bands which extend into the low MHz range. The signal bands of cellular phones and other wireless communication equipment are typically quite narrow frequency ranges centered around some high frequencies (the carriers, say, 900 MHz).

An application's *bandwidth* is considered to be the width of the signal band.

2.2.4 Nyquist's Sampling Theorem

Signals are sampled because it is generally easier to process them when they are represented in this format. This is true for discrete-time analog signals as well as, and in particular, for digital signals. However, there is no point in sampling a signal, unless it is possible to accurately reconstruct at least the signal-band part of the signal.

Notice that for any chosen sampling period T_s and origin of the time variable, a sampled sequence is uniquely described by the continuous-time signal from which it is generated, but that the sampling of two different continuous-time signals may result in the same discrete-time signal. In other words, the process of sampling a continuous-time signal may represent a loss of information, and the continuous-time signal can in general be reconstructed from its sampled sequence only if certain conditions are met. Nyquist's Sampling Criteria is an example of such conditions.

Reconstructing a continuous-time signal from a discrete-time signal will involve some kind of DT/CT conversion. Without loss of generality, only DT/CT conversions, which can be modeled as applying $g_{\mathrm{eqv}}(t)$ to a linear filter[4], will be considered.

Traditional Version. As expressed by (2.10) and illustrated in Figure 2.2, the process of sampling a signal is nonlinear. However, if the signal $g(t) \leftrightarrow G(f)$ being sampled is characterized by *Nyquist's Sampling Criteria*,

$$G(f) = 0 \quad \text{for} \quad |f| > f_s/2 \tag{2.11}$$

it follows that the sampled signal $g_d(k) \leftrightarrow G_d(f)$ is equivalent to $g(t) \leftrightarrow G(f)$ in *the Nyquist Range* $|f| < f_s/2$. In other words, by filtering $g_{\mathrm{eqv}}(t)$ with a linear filter having the transfer function $H(f)$

$$H(f) = \begin{cases} 1 & \text{for} \quad |f| < f_s/2 \\ 0 & \text{for} \quad |f| > f_s/2 \end{cases} \tag{2.12}$$

the result will be $g(t)$.

[4] $g_{\mathrm{eqv}}(t)$ was defined by Equation (2.9).

Nyquist's Sampling Theorem simply states that a signal, which fulfills Nyquist's Sampling Criteria, can be ideally reconstructed from its sampled sequence.

2.2.5 Aliasing

If a signal is sampled at a sampling frequency f_s, which does not fulfill Nyquist's Sampling Criteria (2.11), *aliasing* will occur. Aliasing simply means that the individual terms in the sum described by Equation (2.10) overlap, i.e. , that two terms both are non-zero at some frequency f_0. Figure 2.2 illustrates a situation in which significant aliasing takes place at frequencies around $f_s/2 + nf_s, n \in Z$.

Aliasing can sometimes be allowed. For example, if a signal only needs to be reconstructed in an application's signal band, it is sufficient to require that aliasing does not take place at signal-band frequencies.

Modified Sampling Criteria. For an application which is characterized by the signal band, $|f| < f_b < f_s/2$, the signal-band part of a signal $g(t)$ can be ideally reconstructed from the sequence $g_0(k)$ sampled from $g(t)$ at the sampling frequency f_s, if and only if

$$G(f) = 0 \quad \text{for} \quad |f| > f_s - f_b \tag{2.13}$$

For the maximum signal-band width ($f_b = f_s/2$), this modified sampling criteria is equivalent to Nyquist's Sampling Criteria.

2.3 Estimation of a Signal's Fourier Spectrum

The mathematical definition of the Fourier Transformation, Equation (2.2), is used mainly for theoretical derivations. In practice, a method is needed to estimate the Fourier spectrum $G(f)$ of a signal $g(t)$, which has been obtained by simulations or through experiments. The main incentive to discuss this method in some detail is that it is used extensively to evaluate the performance of data converters. For a more detailed and coherent discussion of this topic, the reader is referred to [21] and [22].

2.3.1 Estimation Based on a Finite-Duration Signal

Obviously, it will generally be impossible to calculate the actual Fourier Transformation (2.2), because any estimation process can only be based on finite-duration signals. Only in special cases, e.g. when the signal is known to be periodic and an exact representation of a period of the signal is available, can the Fourier transform be calculated accurately. In the following, it will be assumed that $G(f) \leftrightarrow g(t)$ will be estimated on the basis of $g(t)$ known for $|t| < T_{\rm obs}/2$, where $T_{\rm obs}$ is some chosen *observation time*.

A typical way to estimate $G(f)$ is to calculate $G_{\rm obs}(f) \leftrightarrow g_{\rm obs}(t)$, where $g_{\rm obs}(t)$ is defined as $g_{\rm obs}(t) = w_{\rm sq}(t)g(t)$ and $w_{\rm sq}(t)$ is the chosen *time window*, which in the following is defined as

$$w_{\rm sq}(t) = \begin{cases} 1 & \text{for} \quad |t| < T_{\rm obs}/2 \\ 0 & \text{otherwise} \end{cases} \tag{2.14}$$

The feasibility of this technique follows from calculating the Fourier transform $W_{\rm sq}(f)$ of $w_{\rm sq}(t)$ as

$$W_{\rm sq}(f) = \int_{-T_{\rm obs}/2}^{T_{\rm obs}/2} e^{-j2\pi ft} dt = T_{\rm obs} \frac{\sin(\pi f T_{\rm obs})}{\pi f T_{\rm obs}} \tag{2.15}$$

The Fourier transform $W_{\rm sq}(f)$ is illustrated in Figure 2.3, where it can be seen that the window's *main lobe* is twice as wide as the reciprocal of the observation time, and as high as the observation time.

Equation (2.7) implies that

$$G_{\rm obs}(f) = G(f) * W_{\rm sq}(f) \tag{2.16}$$

This agrees with the observation that $W_{\rm sq}(f)$ is an impulse approximation when $T_{\rm obs} \to \infty$, in which case $G(f) = G_{\rm obs}(f)$. It is, however, more important to consider the case where $T_{\rm obs}$ is finite.

Properties of Finite-Duration Estimates. Figure 2.4 shows on the left the Fourier spectrum $G(f)$ of a two-tone periodic signal, and to the right the Fourier spectrum $G_{\rm obs}(f)$ of the observed signal. Notice that $G_{\rm obs}(f)$ can easily be predicted on the basis of the Fourier transform $G(f)$, Figure 2.3, and Equation (2.16).

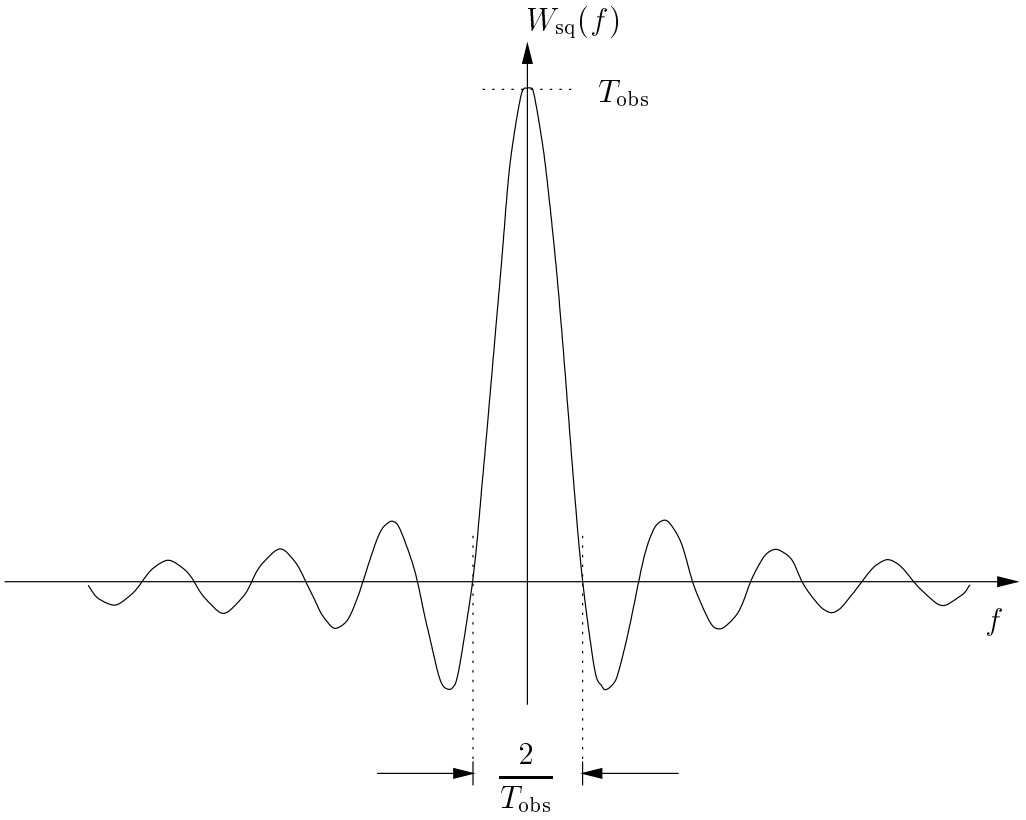


Figure 2.3: The Fourier transform of the rectangular window $w_{\rm sq}(t)$.

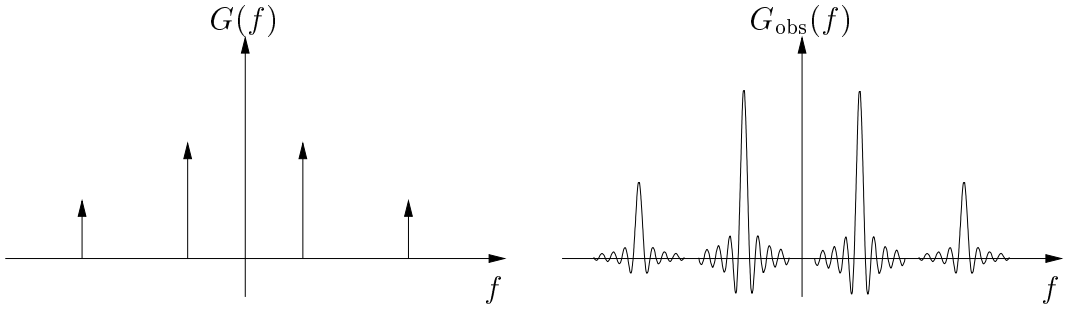


Figure 2.4: The actual and the observed frequency spectrum of a two-tone periodic signal.

Clearly, when estimating a signal's Fourier spectrum on the basis of a finite-duration representation, the energy located at any one frequency, say f_0, will be smeared over a range of frequencies in the neighborhood of f_0. This effect is called *spectral leakage*.

The range of frequencies to which the energy leaks is in principle infinitely wide. However, the width of the frequency range, in which any given fraction (say 99.9%) of the energy is represented, will be inversely proportional to the duration T_{obs} of the observed signal. The width of the window's $W_{\mathrm{sq}}(f)$ main lobe will in the following be called the window's *spectral aperture*. Notice that the spectral composition of a signal can be estimated with any accuracy, simply by using a sufficiently long observation time T_{obs}.

In the example illustrated in Figure 2.4, $G_{\mathrm{obs}}(f)$ represents only a fraction of the energy of $G(f)$; this is a simple implication of $g(t)$ being periodic, and hence of infinite energy, whereas $g_{\mathrm{obs}}(t)$ is of finite duration and energy.

2.3.2 Estimation Based on Assumed Periodicity

All real-world analog signals are continuous and of finite duration, and hence they can be analyzed using the basic Fourier transform without encountering any convergence problems. However, it is often preferable to estimate a signal's spectral composition in terms of power rather than energy. One reason is that the typical test setup will evaluate a system's steady-state response to an input that is periodic in the observation period. In that case, the observed signal will supposedly consist of a periodic deterministic component (the signal) and a stationary stochastic component (noise).

The following is based on the fundamental assumption that the analyzed signal $g(t)$ can be approximated by an periodic extension $g_{\mathrm{per}}(t)$ of the observed signal $g_{\mathrm{obs}}(t)$

$$g_{\mathrm{per}}(t) = \sum_{n=-\infty}^{\infty} g_{\mathrm{obs}}(t - nT_{\mathrm{obs}}) = \sum_{n=-\infty}^{\infty} g(t - nT_{\mathrm{obs}}) w_{\mathrm{sq}}(t - nT_{\mathrm{obs}}) \tag{2.17}$$

Assuming that T_{obs} spans exactly an integer number of periods, the deterministic component will be represented with great accuracy, but it should be obvious that a periodic extension of a stochastic component will not represent the actual stochastic process.

A stationary stochastic process is best described by its autocorrelation function and the Fourier transform thereof, i.e. , the *Spectral Power Density* (SPD) (cf. [23]); but in practice, the noise signal's SPD can be estimated only on the basis of a periodic extension of an observed finite-duration sequence.

In the following, the *average value* shall refer to averaging in the period of time in which the considered signal is observed.

The Band-Pass-Filter Method. Figure 2.5 shows the fundamental elements of the Band-Pass-Filter (BPF) method.

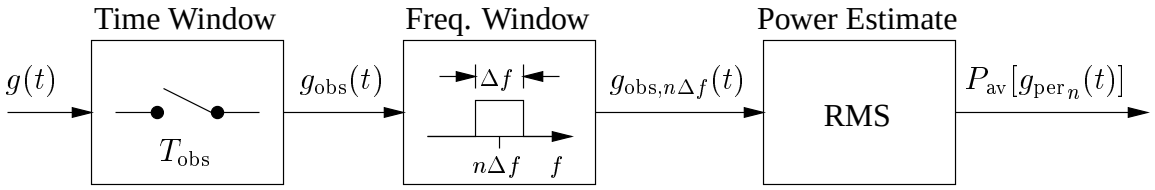


Figure 2.5: Fundamental steps in the Band-Pass-Filter (BPF) method.

Using Parseval's Theorem (2.5) and the definition of the time window (2.14), it follows that the average power of the periodic extension $g_{\rm per}(t)$ of $g_{\rm obs}(t)$ can be calculated as

$$P_{\rm av}[g_{\rm per}(t)] = \frac{1}{T_{\rm obs}} \int_{-\infty}^{\infty} |G_{\rm obs}(f)|^2 df \tag{2.18}$$

The BPF method uses a tunable band-pass filter to isolate individual parts of the frequency spectrum of $G_{\rm obs}(f)$. For simplicity, the bandpass filter is assumed to be ideal with a single-sided tunable center frequency of $n\Delta f$ and a bandwidth of Δf. Equation (2.18) can, therefore, be written in the form

$$\begin{aligned} P_{\rm av}[g_{\rm per}(t)] &= \frac{1}{T_{\rm obs}} \sum_{n=-\infty}^{\infty} \left[\int_{\Delta f(n-0.5)}^{\Delta f(n+0.5)} |G_{\rm obs}(f)|^2 df \right] \\ &= \frac{1}{T_{\rm obs}} \sum_{n=-\infty}^{\infty} \left[\int_{-\infty}^{\infty} |G_{{\rm obs},n\Delta f}(f)|^2 df \right] \end{aligned} \tag{2.19}$$

Using Parseval's Theorem (2.5) and (2.19), it follows that $g_{\rm per}(t)$ thereby is separated in terms of power

$$\begin{aligned} P_{\rm av}[g_{\rm per}(t)] &= \frac{1}{T_{\rm obs}} \sum_{n=-\infty}^{\infty} \left[\int_{-\infty}^{\infty} g_{{\rm obs},n\Delta f}(t)^2 dt \right] \\ &= \sum_{n=-\infty}^{\infty} P_{\rm av}[g_{{\rm obs},n\Delta f}(t)] \end{aligned} \tag{2.20}$$

When estimating a noise signal, it makes the most sense to assume that $|G_{\rm obs}(f)|$ is constant in the narrow frequency ranges: $(n-0.5)\Delta f < f < (n+0.5)\Delta f$, and hence that $|G_{\rm obs}(f)|$ should be estimated by

$$\begin{aligned} |G_{\rm obs}(f)| &= \sqrt{\frac{1}{\Delta f} \int_{-\infty}^{\infty} g_{{\rm obs},n\Delta f}(t)^2 dt} \\ &= \sqrt{T_{\rm obs}} \sqrt{\frac{P_{\rm av}[g_{{\rm obs},n\Delta f}(t)]}{\Delta f}} \quad \text{for} \quad |f - n\Delta f| < 0.5\Delta f \end{aligned} \tag{2.21}$$

However, when estimating a deterministic and assumed periodic signal component, it makes more sense to approximate the estimated power by assuming a tone at $n\Delta f$, and hence that $|G_{\rm per}(f)|$ should be estimated as

$$|G_{\rm per}(f)| = \delta(f - n\Delta f) \frac{\sqrt{P_{\rm av}[g_{{\rm obs},n\Delta f}(t)]}}{2} \quad \text{for} \quad |f - n\Delta f| < 0.5\Delta f \tag{2.22}$$

Using equations (2.15) and (2.16), it follows that the respective portion of $|G_{\rm obs}(f)|$ can be estimated as

$$|G_{\rm obs}(f)|_{n\Delta f} = W_{\rm sq}(f - n\Delta f) \frac{\sqrt{P_{\rm av}[g_{{\rm obs},n\Delta f}(t)]}}{2} \tag{2.23}$$

For real-world implementations of the BPF method, the window's spectral aperture will typically be much smaller than the band-pass filter's band-width, and hence the individual terms of (2.23) will usually not overlap significantly.

2.3.3 The Discrete Fourier Transformation

The Discrete Fourier Transformation (DFT) is, in principle, just an implementation of the BPF method. Hence, all the above comments on the assumed periodicity etc. also apply to the DFT.

The DFT is a numeric technique, which is implemented using DSP[5], therefore it is applied only to digital signals. Thus, to obtain an estimate of the Fourier spectrum of a continuous-time signal, it must first be sampled according to Nyquist's Criteria, Equation (2.11), to avoid aliasing. Because the DFT can be implemented with arbitrary accuracy, both the real and the imaginary part of the Fourier spectrum can be estimated.

Using the DFT is the natural approach in a simulation environment, but it can also be used for laboratory measurements[6].

Fundamental Properties of the DFT. The DFT is calculated on the basis of a finite sequence of N samples, which are assumed to result from a uniform sampling with the sampling period T_s. The observation period T_{obs} is therefore

$$T_{\text{obs}} = NT_s \tag{2.24}$$

The DFT can be considered to be an implementation of the BPF method using an array of N band-pass filters with center frequencies $\frac{n}{N} f_s, n \in \{0, 1, 2, \ldots, (N-1)\}$, each with a bandwidth of f_s/N. Because the analyzed signal $g(k)$ is sampled, the Fourier spectrum will be periodic with period f_s, and hence the DFT provides an estimate of the entire Fourier spectrum.

As discussed with respect to the BPF method, it is a choice whether the estimated power in a given frequency range is assumed to be equally distributed at all frequencies in the frequency range, or concentrated at a single frequency in the range. The DFT cannot possibly make an intelligent choice, and hence it always assumes that the power is located only at the band-pass filter's center frequencies (which in the following will be called the *fundamental frequencies*). Indeed, this is the correct choice if the analyzed signal is periodic with the period[7] T_{obs}. In other words, the DFT assumes that $g(k)$ can be

[5]The DFT is available in many dedicated software packages, such as MATLAB.

[6]Measurement equipment which is based on the DFT, however, requires a dedicated highly-accurate ADC [24].

[7]It does not have to be the shortest period.

written in the form

$$\begin{aligned} g_{\mathrm{obs}}(k) &= \sum_{n=0}^{N-1} c_n e^{j2\pi(f_s n/N)k} \\ &= \sum_{n=0}^{N-1} a_n \cos(2\pi(f_s n/N)k) + jb_n \sin(2\pi(f_s n/N)k) \end{aligned} \tag{2.25}$$

The vector DFT(n) is simply the N complex coefficients in

$$\mathrm{DFT}(n) = c_{n+1}, \quad n \in \{1, 2, 3, \ldots, N\} \tag{2.26}$$

Windowing. The DFT's method of allocating the estimated power is only correct if the analyzed signal $g(k)$ is periodic with the period T_{obs}. In a simulation environment, it is often possible to assure that the signal component of the analyzed signal fulfills this requirement, but errors that are not in harmonic relation to the signal component, such as the quantization "noise" in delta-sigma modulators, will usually not be periodic with period T_{obs}. As for the BPF method, spectral leakage will occur if the periodicity requirement is not fulfilled.

The Fourier transform $W_{\mathrm{sq}}(f)$ of the basic rectangular time window (2.14) was shown in Figure 2.3. The zero crossings of this transform are equidistant with the same distance as the spacing between the DFT's fundamental frequencies. An implication of this property is that $G_{\mathrm{obs}}(f)$ in (2.16) equals the analyzed signal's Fourier spectrum $G(f)$ at the frequencies of interest if, but only if, $g(k)$ is periodic with period T_{obs}. In other words, assuming periodicity, spectral leakage will not occur in the DFT. However, if $g(k)$ is not periodic with period T_{obs}, spectral leakage will occur, and it may cause very misleading results, especially when using the DFT to estimate the Fourier spectrum of a signal with substantial out-of-band power[8].

Fortunately, it is fairly simple to avoid the deleterious effects of spectral leakage. The technique is to avoid substantial areas of the side lobes in the time window's Fourier transformed (see Figure 2.3). Side-lobe suppression can be obtained by scaling the time window's coefficients (2.14), such that, even

[8]This is especially important for single-bit delta-sigma modulators where the out-of-band power in general will be greater than the signal power.

in the lack of periodicity, the periodic extension of the observed windowed representation of the analyzed *continuous-time* signal $g_{\mathrm{per}}(t)$ is continuous and has continuous derivatives. An outstanding and comprehensive tutorial on windowing, including the theory and the advantages and disadvantages of various windows, is provided by Harris [22].

All DFTs that are presented in the following have utilized the Hanning Window. The Hanning Window is characterized by a fairly narrow main lobe, a good suppression of spectral components away from the main lobe, and a worst-case *processing loss* of approximately 3 dB. A window's processing loss is comparable to the noise factor of an analog circuit. The results presented in this thesis have *not* been corrected for the window's processing loss, therefore all the results may be up to 3 dB on the pessimistic side. Power located at any one of the fundamental frequencies will leak to the two neighboring fundamental frequencies. Hence, the power of a sinusoid signal must be estimated as the power represented by three coefficients in DFT(n).

An Example of Windowing. Figure 2.6 shows six DFTs, which are all calculated from the same signal provided by an ideal four-bit delta-sigma modulator. Each of the plots show the DFT coefficients in dB versus frequency normalized with respect to $f_s/2$. The estimated signal consists of a signal component, a sinusoid at the normalized frequency 0.16; and a pseudo-stochastic component, the quantization noise that supposedly has very little energy in the signal band, i.e. in the normalized frequency range from 0 to 0.2.

The DFTs in the left-hand column are based on 512 samples, whereas the DFTs in the right-hand column are based on 4096 samples. Hence, the bandwidth of the longer DFTs band-pass filters is eight times smaller than the bandwidth of the short DFTs band-pass filters. This property is reflected by the DFT coefficients that represent the noise components, e.g. , at the normalized frequencies from 0.4 to 1, which are 9 dB lower for the longer DFTs. The coefficients of the DFTs that represent the signal component are, however, unaffected by the length of the DFT, because the energy is concentrated at a single frequency, which will be part of only one band-pass filter's pass band[9]. Hence, the modulator's signal-to-noise ratio (SNR) cannot be estimated as "the vertical distance between the signal bin and the noise floor," which is

[9] When not encountering spectral leakage.

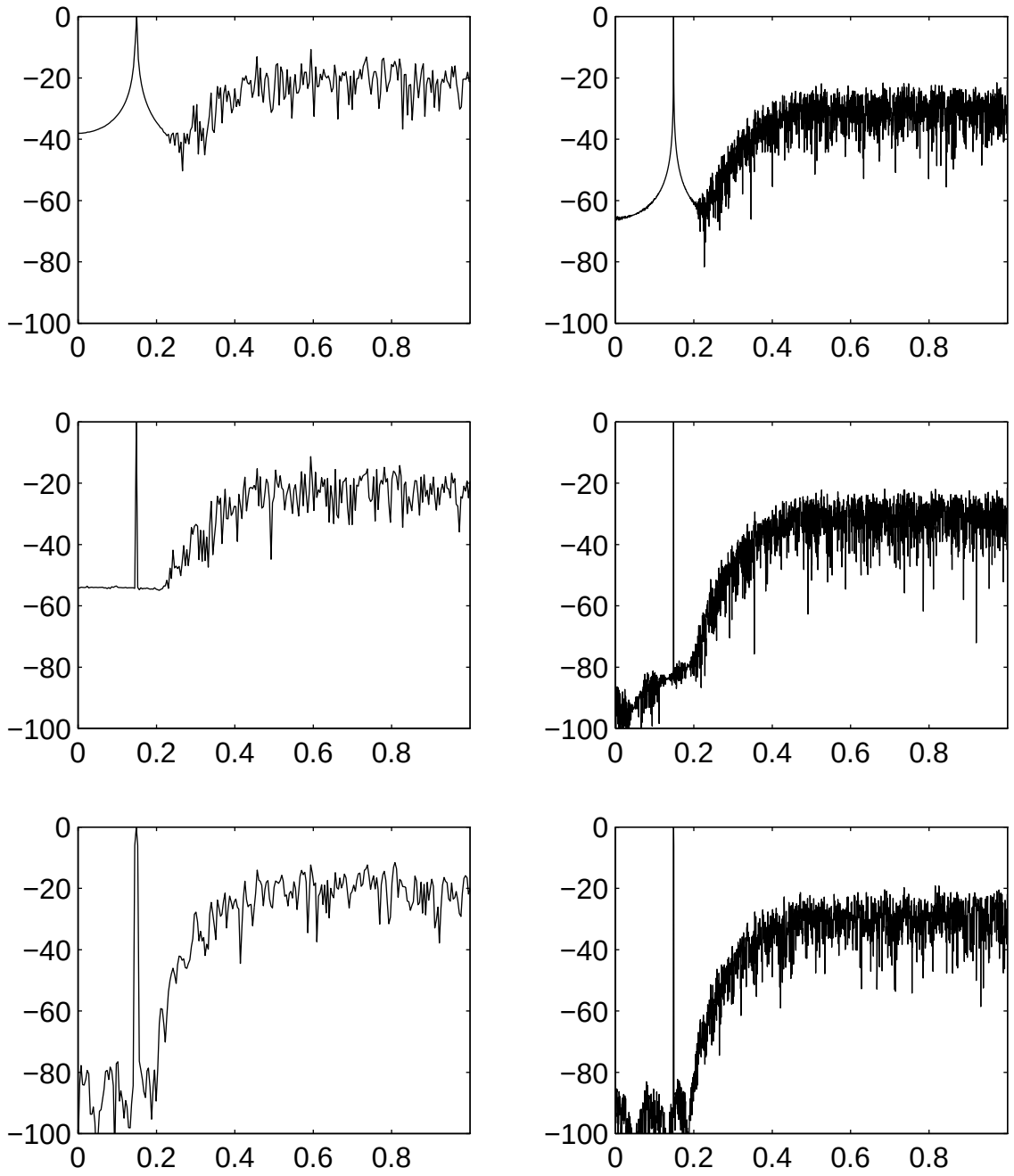


Figure 2.6: Examples of some of the misleading results that the DFT may cause when windowing is not used correctly. The two lower plots are "correct."

a common misconception.

The two DFTs in the top row illustrate what can happen when the signal component is not located exactly at a fundamental frequency. The rectangular window (2.14) was used for these two DFTs. Here, spectral leakage masks the huge difference in power contents at neighboring frequencies around 0.16, hence the total signal-band noise power cannot be observed correctly from these two DFTs.

The two DFTs in the middle row are also based on the rectangular window, but here the signal component is placed exactly at a fundamental frequency; consequently, spectral leakage does not occur from the signal component. The DFTs are, however, still corrupted from spectral leakage from the noise components, as can be observed in the signal band.

The two DFTs in the bottom row are based on a periodic signal, which has been applied to the Hanning window. Spectral leakage of the noise components is now greatly suppressed, therefore the total power of the signal-band noise can be estimated fairly accurately. Notice that the three characteristic[10] signal-band notches in the quantization noise's power density can be observed only in these two plots, which is an effect of the improved detection. Also, notice that the signal component is spread to three coefficients in the DFT (controlled spectral leakage). This can best be observed for the short DFT, but the effect is also present in the longer DFT.

[10] They are characteristic of this specific modulator.

Chapter 3

Basic Aspects of Data Conversion

This chapter will discuss the fundamental elements in A/D and D/A converters. D/A converters are discussed mainly because they are an integral part of most A/D converters, and as such, they are a very important part of the problem.

To facilitate meaningful discussion of the various problems involved in implementing data converters, the basic terminology and the ideal operation are defined. The potential problems associated with the individual stages are identified, and the associated errors are modeled. The common measures for both static and dynamic errors are also described.

An important aspect of this chapter is that it points to several system-level reasons as to why most data converters necessarily will operate with somewhat oversampled signals. This implies that the proposed data converters do not impose any real limitations only because they require the same degree of over-sampling for yet another reason.

3.1 Fundamental Steps in A/D Conversion

Figure 3.1 shows the basic steps of a typical A/D conversion process. The operation of some ADCs cannot be separated into the illustrated three-step sequence, but the method is general enough to serve

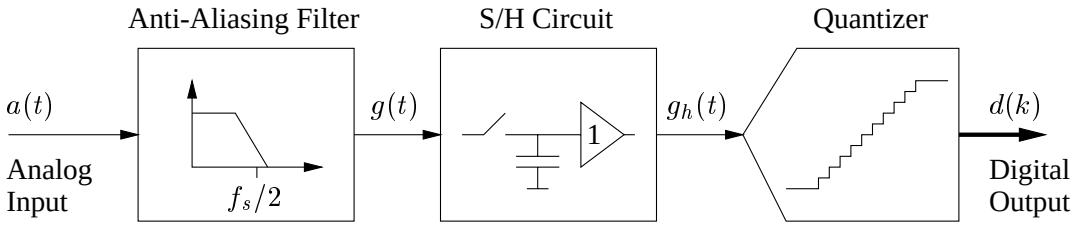


Figure 3.1: Fundamental steps in A/D conversion: anti-aliasing filtering, sampling, and quantization.

as background for the following discussion.

Anti-Aliasing Filter. Generally, the analog input signal will be a continuous-time signal. Since the digital output signal is a discrete-time signal, a sampling process will take place. As any sampling process is subject to aliasing (cf. page 18), it is usually necessary to use an anti-aliasing filter to reduce the consequences of this nonlinear effect to the required level.

Sample-and-Hold Circuit. Many A/D conversion structures make use of a dedicated sample-and-hold (S/H) circuit. This circuit receives the continuous-time signal $g(t)$, samples it, and provides as its output the most recently sampled value. The reason for this operation is that many quantizers require a minimum set-up time[1], or that the input must be held constant for a synchronized minimum period of time (multi-step quantizers).

Quantizer. The quantizer performs the actual analog-to-digital conversion (quantization). Ideally, a linear function will describe the relation between the analog signal $g_h(kT_s)$ and the digital output $d(k)$.

3.1.1 Errors Caused by the Anti-Aliasing Filter

The anti-aliasing filtering process must take place prior to the sampling; consequently the filter can only be implemented as a continuous-time filter. It can be an RLC filter, but for integrated-circuit applications

[1]Set-up time is the amount of time that the analog value must be provided (i.e., as a constant) to the quantizer before the quantization can be initiated.

it will typically be an active-RC or a transconductor-capacitor (g_mC) filter.

All analog filters – especially those operating in continuous time – are subject to errors due to analog imperfections. These errors, which include thermal noise, flicker noise, distortion, etc., can very well turn out to be the limiting factors in a high-performance A/D converter system. To minimize the influence of such errors, the anti-aliasing filter should be designed with great care, and it should preferably have a simple transfer function.

This work does not address this design issue, but it is worthwhile to notice that the usual biquad topology may not be the optimal design approach. Some valuable information on this aspect can be obtained from [25]. Also, some interesting aspects on simplicity, linearity, and high-speed operation are discussed in [26].

Estimation of Aliasing Errors. Aliasing errors are determined uniquely by the spectral composition of the signal $g(t)$ that is undergoing sampling. To avoid aliasing errors, Nyquist's Sampling Theorem (cf. page 17) requires that

$$G(f) = A(f)H_{\text{alias}}(f) = 0, \quad \text{for} \quad |f| > f_s/2 \tag{3.1}$$

This is, however, impossible to obtain because any finite-duration signal has infinite bandwidth.

Fortunately, it is more reasonable to require that the power of the spectral components that alias back into the signal band, $E_{\text{alias}}[g(t)]$, be dominated by other errors, such as noise. In the following, it will be assumed that the signal band is described by $|f| < f_b$. The requirement can then be expressed as

$$\begin{aligned} E_{\text{alias}}[g(t)] &= \sum_{n\neq 0}\left[\int_{nf_s-f_b}^{nf_s+f_b} |G(f)|^2 df\right] \\ &< \frac{[E_{\text{inband}}[g(t)]]_{\max}}{\text{SNR}} \\ &= \frac{\left[\int_{-f_b}^{f_b} |G(f)|^2 df\right]_{\max}}{\text{SNR}} \end{aligned} \tag{3.2}$$

Even for a modest signal-to-noise ratio (SNR), this requirement is nearly impossible to fulfill if the signal band is as wide as the Nyquist Range, and if it is assumed that the signal-band power may be distributed *anywhere* in the signal band.

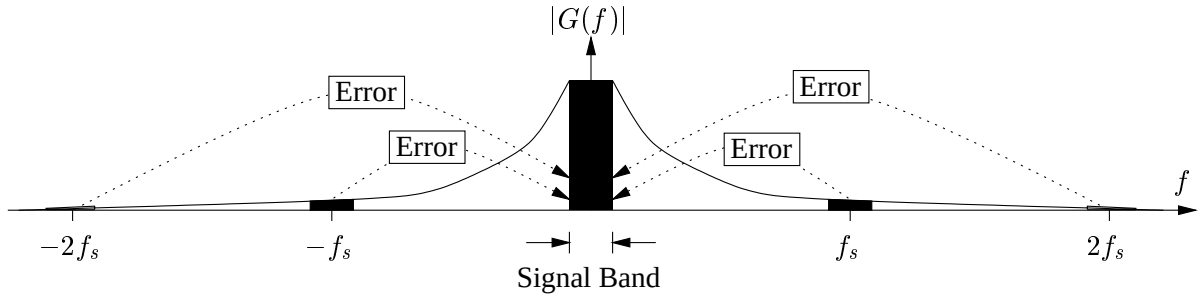


Figure 3.2: Error that alias back into the signal band for a six-times oversampled system.

To fulfill (3.2) for a high SNR, it is in general necessary to *oversample* the signal $g(t)$, i.e., to sample $g(t)$ at a frequency somewhat higher than $f_s = 2f_b$. The *oversampling ratio* (OSR) is defined as

$$\text{OSR} = \frac{f_s}{2f_b} \tag{3.3}$$

Figure 3.2 is a representation of the expression for $E_{\text{alias}}[g(t)]$ in Equation (3.2) for a six-times oversampled system.

Choosing the Anti-Aliasing Filter's Characteristic. As the Fourier spectrum $G(f) \leftrightarrow g(t)$ is the product of the input signal's Fourier spectrum $A(f) \leftrightarrow a(t)$ and the anti-aliasing filter's transfer function $H_{\text{alias}}(f)$, no closed-form specification for the anti-aliasing filter can exist; the required properties of $H_{\text{alias}}(f)$ are highly dependent on the signal $A(f)$.

It can be derived from Figure 3.2 that the anti-aliasing filter should be designed as a low-pass filter with the signal band as its pass band, and with a stop band for frequencies higher than $f_s - f_b$. Assuming that the filter has 0 dB gain in the pass band and a minimum attenuation R_{stop} in the stop band, the aliasing criteria (3.2) can be written in the form

$$\begin{aligned} R_{\text{stop}} &> \text{SNR} \cdot \frac{\sum_{n\neq 0}\left[\int_{nf_s-f_b}^{nf_s+f_b} |A(f)|^2 df\right]}{\left[\int_{-f_b}^{f_b} |A(f)|^2 df\right]_{\max}} \\ &= \text{SNR} \cdot \frac{P_{\text{alias}}[a(t)]}{[P_{\text{inband}}[a(t)]]_{\max}} \end{aligned} \tag{3.4}$$

Equation (3.4) expresses that the anti-aliasing filter's stop-band attenuation must be at least the ratio of the required SNR divided by the worst-case signal-to-aliasing-error ratio for a full-scale input signal

$a(t)$.

Unfortunately, (3.4) does not provide an absolute design criteria because whereas the full-scale signal-band power $[P_{\text{inband}}[a(t)]]_{\max}$ is usually well-specified, the worst-case power of the spectral components that may alias back into the signal band is not. Hence, the specification of the the anti-aliasing filter can only be based on a (qualified) assumption.

For low values of OSR, the ratio $P_{\text{alias}}[a(t)]/\left[P_{\text{inband}}[a(t)]\right]_{\max}$ should probably be estimated as at least -40 dB, which may even be too optimistic. However, using this estimate in the specification for a high-performance application with a required SNR of (say) 100 dB, it follows that the anti-aliasing filter must have a stop-band attenuation of at least 60 dB.

It is well understood that, for relaxed pass-band requirements, the complexity of continuous-time low-pass filters depends largely on the required stop-band attenuation[2] and the relative width of the transition band

$$\frac{f_s - f_b}{f_b} = \frac{f_s}{f_b} - 1 = 2\text{OSR} - 1 \tag{3.5}$$

To obtain 60 dB stop-band suppression with a reasonably simple (say, third-order) filter, the OSR must be at least 4 to 5.

It is important to notice that the above observation on the required OSR is general, and that it is *not* a consequence of a particular choice of quantizer. Hence, quantizer structures that require a similar degree of oversampling to operate, such as those proposed in this work, do not have strict limitations for their applicability.

3.1.2 Errors Caused by the Sample-and-Hold Circuit

The S/H circuit performs the sampling of the signal, and thus it inflicts aliasing errors. These errors were discussed in detail in the previous section 3.1.1.

[2] Relative to the pass-band gain.

As any other analog circuit, the S/H circuit will cause errors such as noise, offset, and nonlinearity. In particular, the sampling switch is a very critical design aspect, as it will cause charge injection, clock feedthrough, and nonlinearity. Details on these aspects can be found in [27–32].

The S/H circuit must meet the overall design specification, so it is difficult to design for high-performance applications. This work, however, does not address this design aspect.

Clock Jitter. All sampling circuits will be subject to clock-jitter problems. Jitter means that the signal is sampled at slightly incorrect time instances (3.6)

$$g_{\text{jitter}}(k) = g(kT_s + \Delta T(kT_s)) \tag{3.6}$$

where, for simplicity, it is assumed that $|\Delta T(t)| \ll T_s$.

The simplification implies that jitter can modeled as a distortion of $g(t)$ prior to the sampling instance. More precisely, $g_{\text{jitter}}(k)$ is considered to be generated by sampling $g_{\text{jitter}}(t)$, which is a distorted representation of $g(t)$

$$g_h(k) = g_{\text{jitter}}(kT_s) = g(kT_s) + g_{\text{error}}(kT_s) \tag{3.7}$$

where

$$g_{\text{error}}(kT_s) \simeq \Delta T(kT_s)\frac{d}{dt}(g(kT_s)) \tag{3.8}$$

A model for the jitter-induced error is shown in Figure 3.3.

A low-jitter clock signal can be generated using a crystal-based oscillator. As this work focuses on the baseband as the signal band, the magnitude of the first-order derivative $dg(t)/dt$ will be moderate, and $g_{\text{error}}(t)$ will have a comparably low power. Hence, clock jitter is not considered to be a major problem in this context. However, for band-pass applications (where $dg(t)/dt$ may be large), clock-jitter errors from the sampling process are of great concern. This aspect is discussed in more detail in Section 3.2.3.

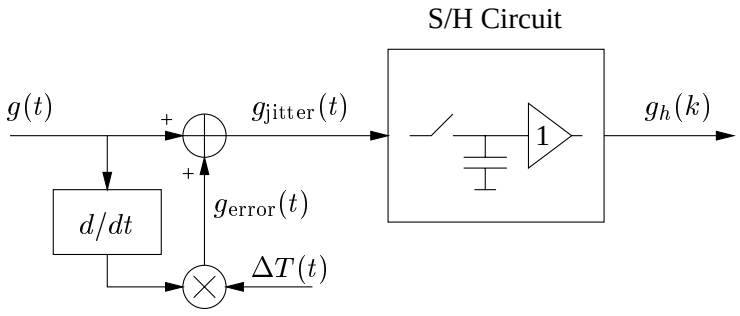


Figure 3.3: Model for clock-jitter-induced errors in the sampling process.

3.1.3 Characterization of the Ideal Quantizer

A fast high-performance quantizer is very difficult to design. The remaining part of this thesis is dedicated in part to the optimization of this portion of the A/D converter system. This section will define the ideal operation.

Basic Assumption. It is assumed that the quantizer's input signal $g_h(t)$ is supplied by a sample-and-hold circuit, such that dynamic effects are avoided. In other words, from the quantizer's point of view, the input is a constant (dc) signal.

Resolution. The quantizer will provide a digital output signal that at any time will attain one value from a finite set of possible values. The number of possible output values will, in this thesis, be called the quantizer's *resolution*. The resolution may be expressed as the number of levels, e.g., *an eight-level ADC*, or as a number of bits, e.g., *a three-bit ADC* (which is equivalent).

For simplicity, it will be assumed that the set of possible output values is a set of uniformly spaced integers, but the spacing (i.e. the step size) need not be one. This flexibility is practical when discussing systems involving multiple quantizers, but (for simplicity) the *overall* quantizer, such as the one illustrated in Figure 3.1, will usually be characterized by a step size of one.

The quantizer's output will occasionally be characterized by binary codes rather than the represented numeric value; e.g., the codes "000," "001," "010," "011," "100," "101," "110," "111" could represent

the eight possible states of a 3-bit ADC. Whenever necessary for clarity, the step size will be identified with the code as "101@8," i.e., code "five" with a step size of eight. Similarly, a 3-bit@8 ADC means a 3-bit ADC with a step size of eight. All ADCs will have a minimum and a maximum code.

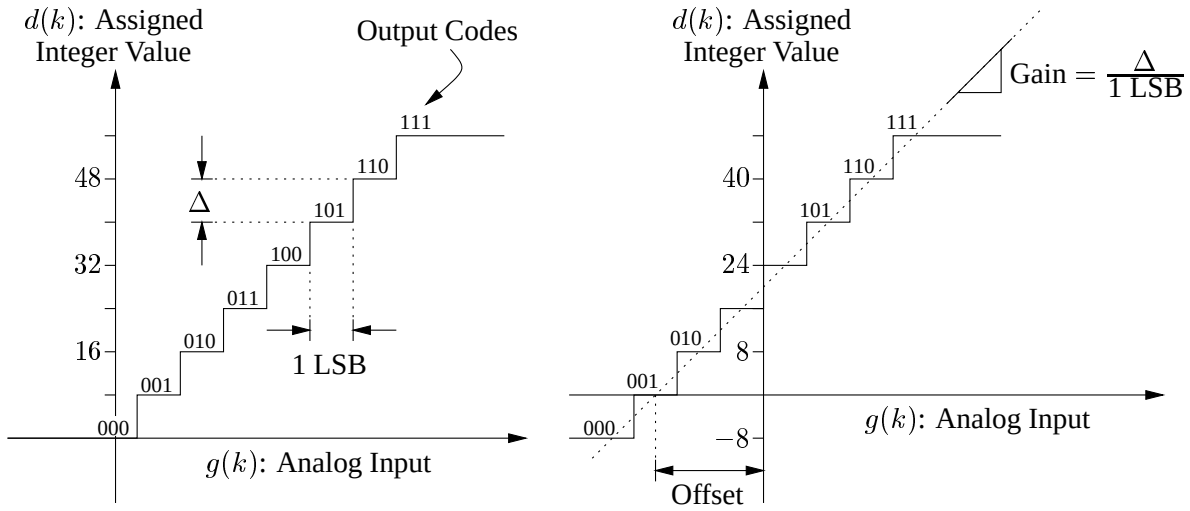


Figure 3.4: Linear 3-bit@8 quantizers. Left: Definition of the step size (Δ) and of 1 LSB. Right: Definition of offset and gain.

Ideal Operation. Figure 3.4 shows two linear 3-bit@8 quantizers. The quantizer shown to the left is a *proportional* quantizer, which (except for the truncation) exhibits proportionality between the analog input and the digital output (i.e., the assigned integer values). Notice that the unit "1 LSB" is defined as an analog quantity, namely the variation of the analog input that will force a transition between two neighbor output codes.

Although it is not a proportional quantizer, the quantizer shown to the right is also considered to be linear. It illustrates that the assigned integer values can be chosen from any set of uniformly spaced integers, for example, as shown $\{-8, 0, 8, 16, 24, 32, 40, 48\}$.

The quantizer's *gain* is defined as the step size divided by 1 LSB. The quantizer's *offset* is defined as the analog value added to the analog input before it becomes a proportional quantizer; the shown quantizer

has an offset of 2 LSB.

Truncation. Because the analog input $g(k)$ can attain values from a continuum of values, and because the digital output $d(k)$ must attain values from a finite set of values, the quantizer will necessarily perform truncation. The truncation error, which often is called the "quantization error," is simply the residue $r(k)$ of the quantization. If the quantizer is characterized by a gain K and an offset g_{offset}, the truncation error, i.e., the residue $r(k)$, is described as

$$r(k) = [(g(k) + g_{\text{offset}}) - d(k)/K] \tag{3.9}$$

As $d(k)$, in principle, is a deterministic function of $g(k)$, the residue $r(k)$ will be a function $T(\cdot)$ of only $g(k)$

$$r(k) = T[g(k)] \tag{3.10}$$

The truncation caused by an ideal 3-bit quantizer is shown in Figure 3.5. As shown, the magnitude of

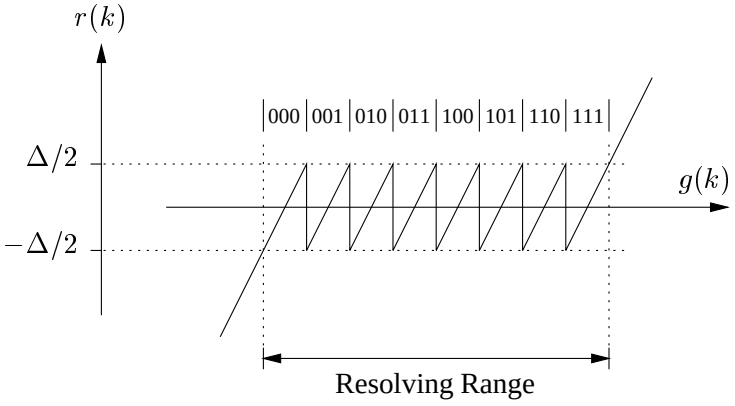


Figure 3.5: Residue of an ideal 3-bit@Δ quantizer.

$r(k)$ will be less than half the step size Δ, as long as the analog input $g(k)$ is within the resolving range.

Based on the nonlinear relationship $T(\cdot)$ between $g(k)$ and $r(k)$, the quantizer model shown in Figure 3.6 can be derived.

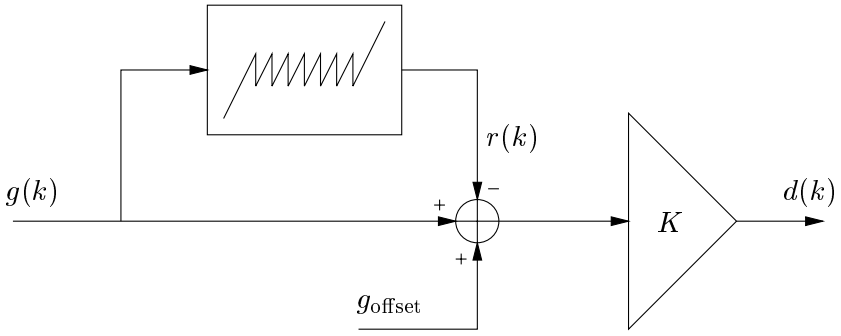


Figure 3.6: Quantizer model.

For simplicity, the truncation error $r(k)$ is often modeled as a white-noise error signal. The validity of this model is highly dependent on the situation. The function $T(\cdot)$ (3.10) can be approximated by polynomials. A simple coarse approximation of $T(\cdot)$ in the resolving range for a $(2P+1)$-level quantizer would be

$$\frac{T(y)}{1\text{ LSB}} \simeq x(x^2-1^2)(x^2-2^2)(x^2-3^2)\cdots(x^2-P^2), \text{ where } x = \frac{y}{1\text{ LSB}} \tag{3.11}$$

The point is that, since $T(\cdot)$ is a very nonlinear function, (3.11) involves high powers of x, hence the Fourier spectrum $R(f) \leftrightarrow r(k)$ will be a sum of many terms of $G(f) \leftrightarrow g(k)$ convolved with itself the same high number of times (cf. Equation (2.7)). As self convolution tends to flatten a function, it may be reasonable to model $R(f)$ as white noise.

There are, however, some situations where it is *not* reasonable to model $R(f)$ as white noise. First, if the resolution of the quantizer is low, the order of (3.11) will be comparably low. Because $G(f)$ is then convolved with itself only a few times, the result will generally not be a Fourier spectrum of almost uniform power density, and hence $r(k)$ should not be modeled as white noise. Second, if $r(k)$ is periodic, $G(f)$ will be a line spectrum, which even after many self-convolution operations will be very tonal. For example, if $g(k)$ is periodic with period N, then $r(k)$ will be periodic with period N as well[3]. Third, even if $R(f)$ does have an approximately uniform spectral power density, modeling it as white noise lacks the correlation information of $g(k)$ and $r(k)$, which sometimes is very important.

In conclusion, $r(k)$ should be modeled as white noise only if

[3]Because $T(\cdot)$ is a deterministic function.

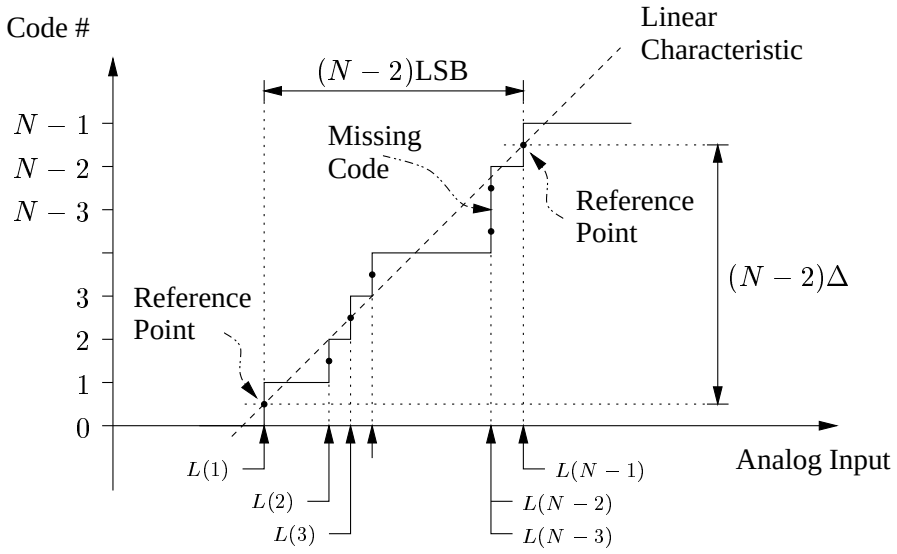


Figure 3.7: Characteristic of a N-level nonideal quantizer with step size Δ.

- the properties of the correlation of $g(k)$ and $r(k)$ are not important,
- the quantizer has a high resolution (and many levels are excited), and
- the input $g(k)$ includes substantial broad-band spectral components.

3.1.4 Characterization of Quantizer Errors

Very accurate quantizers are hard to realize. The spacing of the assigned integer values will (obviously) be ideal, but the accuracy of their relation to the analog signal is prone to errors. Figure 3.7 shows the static characteristic for an N-level nonideal quantizer with step size Δ. This section will classify the various deviations from the ideal operation in terms of some well-known measures.

Linear Errors. It is not always obvious how a quantizer's gain and offset should be defined. In fact, the correct definitions often depend on the application.

For example, consider a system that requires a quantizer with an absolute specification on its gain and offset. Since the specification is absolute, there is no need to distinguish between linear and nonlinear

errors. Instead, the specification can be used as the reference, with respect to which all deviations are considered to be errors. Errors can then be defined by (3.9) and modeled as shown in Figure 3.6.

Quite often, however, offset and gain errors are of little importance compared to the quantizer's linearity. In that case, it sometimes makes sense to define a *best linear characteristic*, with respect to which nonlinear errors (including truncation) are defined. The problem is that the best linear characteristic may be highly dependent on the analog input signal. Consider the quantizer shown in Figure 3.7. If the input, for example, is such that only codes 2 and 3 are used, then the best linear characteristic would have a higher gain and a different offset than the best linear characteristic for a signal where all codes are used.

To avoid inconsistency, the linear characteristic will be defined uniquely. For any N-level quantizer, where N is greater than 2, there will be an maximum input value, $L(1)$, for which all lower values will be quantized to the minimum code. Similarly, there will be a minimum input value, $L(N-1)$, for which all higher values will be converted to the maximum code. The linear characteristic is defined with respect to the two reference points shown in Figure 3.7.

Accordingly, the unit 1 LSB is defined as

$$1 \text{ LSB} = \frac{L(N-1) - L(1)}{N-2} \tag{3.12}$$

and the quantizer's gain K is defined as

$$K = \frac{\Delta}{1 \text{ LSB}} = \frac{(N-2)\Delta}{L(N-1) - L(1)} \tag{3.13}$$

The offset is also defined with respect to the defined linear characteristic.

Nonlinear Errors. Truncation and other nonlinear errors are defined with respect to the defined linear characteristic. For an N-level quantizer, an $(N-1)$-element vector $L(n)$ is defined. The nth element in $L(n)$ represents the lowest analog input[4], for which code n occurs (see Figure 3.7).

Ideally, $L(n)$ will consist of $(N-1)$ uniformly-spaced analog values. To characterize deviations from the ideal behavior, two measures are defined with respect to $L(n)$: the *differential nonlinearity*, DNL(n), and the *integral nonlinearity*, INL(n).

[4]With the exception of $L(N-1)$, which is defined as previously described.

Each horizontal step of the quantizer's characteristic should be 1 LSB wide. DNL(n) is a $(N-2)$-element vector representing each step's deviation from the ideal width

$$\text{DNL}(n) = [L(n+1) - L(n)] - 1 \text{ LSB} \tag{3.14}$$

The transitions (marked with black dots in Figure 3.7) should occur at input values uniformly spaced between $L(1)$ and $L(N-1)$. The INL(n) is a $(N-1)$-element vector representing the transitions' deviation from the ideal locations

$$\begin{aligned} \text{INL}(n) &= [L(n) - L(1)] - (n-1) \text{ LSB} \\ &= \left[\frac{L(n) - L(1)}{L(N-1) - L(1)} - \frac{n-1}{N-2}\right](N-2) \text{ LSB} \end{aligned} \tag{3.15}$$

Notice that (by definition)

$$\text{INL}(1) = \text{INL}(N-1) = 0 \tag{3.16}$$

Often a quantizer is characterized by the largest elements of the two measures

$$\text{INL} = \max_n |\text{INL}(n)| \quad \text{and} \quad \text{DNL} = \max_n |\text{DNL}(n)| \tag{3.17}$$

A quantizer's DNL and INL performance are typically expressed in LSBs. A quantizer described by a good DNL has a smooth characteristic[5], but the overall linearity, i.e., the INL, need not be comparably good [24]. On the other hand, a good INL implies a good DNL, because

$$\text{DNL}(n) = \text{INL}(n+1) - \text{INL}(n) \tag{3.18}$$

Effective Number of Bits. It should be understood that a quantizer can be very linear without having a high resolution, and also that a quantizer can have a high resolution without being very linear. It may be relatively simple to increase a quantizer's resolution, whereas it is almost always hard to increase its relative linearity. In many publications, the relative linearity of data converters is expressed in terms of a number of bits.

[5] Good local linearity.

A quantizer's DNL expressed as an *effective number of bits* (ENOB) is defined as the fictitious resolution for which the DNL is 1 LSB. In other words, a quantizer's DNL expressed in ENOB is defined as

$$\text{DNL} = \log_2 \left[\frac{[L(N-1) - L(1)][N/(N-2)]}{\max_n |\text{DNL}(n)|} \right] \text{ ENOB} \tag{3.19}$$

Similarly, a quantizer's INL expressed in ENOB is defined as

$$\text{INL} = \log_2 \left[\frac{[L(N-1) - L(1)][N/(N-2)]}{\max_n |\text{INL}(n)|} \right] \text{ ENOB} \tag{3.20}$$

3.2 Fundamental Steps in D/A Conversion

Many A/D converters make use of D/A converters in the quantization process, and quite often the performance of such ADCs is limited by the DAC's (typically good) performance. Hence, a good insight into some aspects of D/A conversion is required for a meaningful evaluation of A/D converter structures.

3.2.1 Basic Voltage-Mode Implementation

Figure 3.8 shows the fundamental steps of a typical voltage-mode D/A conversion process.

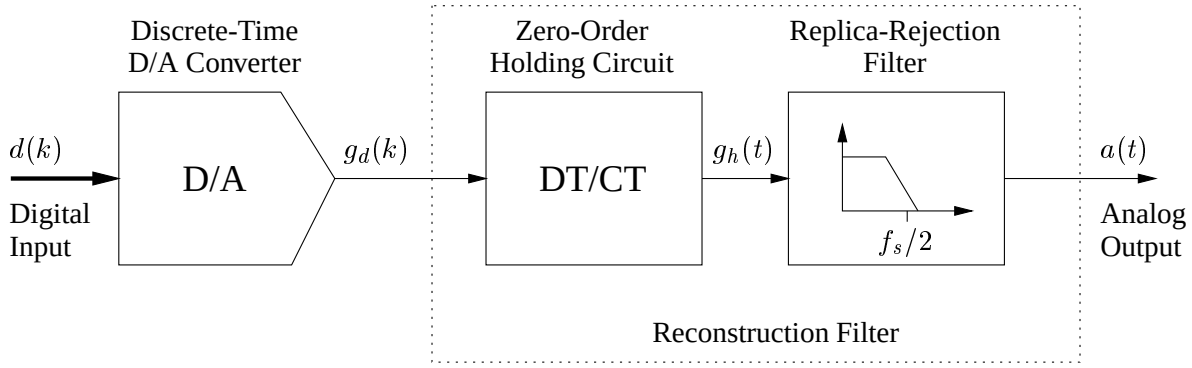


Figure 3.8: Basic elements in a voltage-mode D/A converter system.

First, the digital input $d(k)$ is converted into a discrete-time voltage signal $g_d(k)$. The advantage of this approach is that the voltage signal is evaluated only at discrete time instances, so that slew-rate limitation

and other nonlinear settling effects of this circuit can be accepted. The discrete-time voltage signal $g_d(k)$ is then DT/CT converted by a reconstruction filter.

The first stage of most reconstruction filters is a zero-order holding filter, which performs the DT/CT conversion described by

$$g_h(t) = g_d(k) \quad \text{for} \quad kT_s < t < (k+1)T_s \tag{3.21}$$

The circuit is, in principle, a S/H circuit, for which the output is evaluated in continuous time. The operation can be modeled in the frequency domain as a filter with the transfer function[6] (3.22) [33]:

$$H_{\mathrm{DT/CT}}(f) = T_s\, e^{-j\pi f T_s} \left[\frac{\sin(\pi f T_s)}{\pi f T_s}\right] \tag{3.22}$$

This filter will suppress the replica images (2.10) somewhat, but quite often, a dedicated replica-rejection filter is needed to obtain further suppression.

Linearity of the Reconstruction Filter. The continuous-time evaluation of the output signal $g_h(t)$ from the DT/CT converter is a very critical aspect. Ideally, $g_h(t)$ should be a staircase signal, but this is impossible to obtain because it is not a continuous signal. Any implementation of a zero-order holding circuit will include some degree of low-pass filtering, e.g., caused by the frequency response of an employed op-amp. The main design aspect is, however, to make the circuit behave linearly. For example, if the step size (i.e., the sample-to-sample variation) of $g_d(k)$ is large, the circuit is likely to become slew-rate limited, resulting in a performance deterioration. An analysis will show that, to preserve linearity, the discrete-time analog signal $g_d(k)$ should be oversampled at least ten times and include only little out-of-band power.

Single-bit delta-sigma D/A converters represent the signal-band information in the form of a highly-oversampled single-bit signal $d(k)$. The advantage of this approach is that it may avoid nonlinearity in the discrete-time D/A conversion process. However, because the analog equivalent is a rapidly-varying two-level signal, it is nearly impossible to DT/CT convert it linearly. In the frequency domain, the

[6] Except for the phase shift factor $e^{-j\pi f T_s}$ and the substitution of T_{obs} and T_s, the transfer function $H_{\mathrm{DT/CT}}(f)$ is identical to $W_{\mathrm{sq}}(f)$ defined as (2.15) and illustrated in Figure 2.3.

problem can be detected as the signal having substantial power at frequencies outside the signal band. To facilitate linear DT/CT conversion, the two-level voltage signal is typically filtered with a *discrete-time* low-pass filter before it is DT/CT converted; see Figure 3.9 and [25].

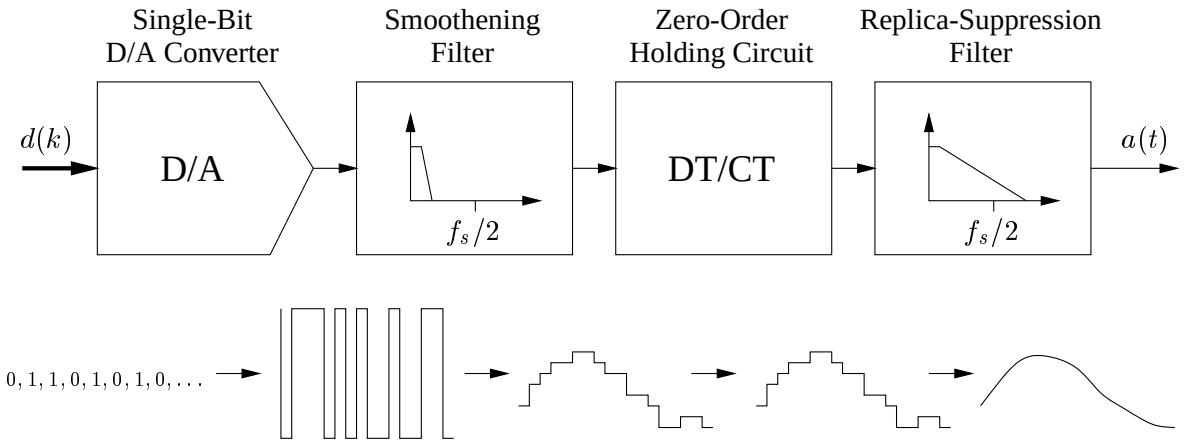


Figure 3.9: Output stage of a single-bit delta-sigma D/A converter.

Notice that the main purpose of the smoothing filter is to reduce the step size, and not in particular to suppress the replica images (2.10). Whether it is necessary to suppress replica images more than by the zero-order holding filter depends on the application. However, even if the application is indifferent to the received signal's spectral composition outside the signal band, the above discussion illustrates that filtering may be necessary to avoid nonlinearity of the channel, which transmits the signal to the application. Another example of this aspect could be a loudspeaker that distorts high-frequency energy, thereby transforming it into signal-band energy that corrupts the signal detected by the human ear. In this case, however, the long-term effects on the human ear's sensitivity should also be considered.

Assuming that the step size somehow has been made sufficiently small, the DT/CT converter can be implemented as shown in Figure 8 in [25].

3.2.2 Basic Current-Mode Implementation

Due to the absence of a simple high-performance current-mode sample-and-hold circuit for the implementation of a zero-order holding circuit, current-mode DACs do usually not employ discrete-time analog signals. Figure 3.10 shows a model of a current-mode D/A converter structure, where the usual three-stage separation is indicated. It should, however, be understood that the separation is mainly theoretical, as it does not generally represent the implementation's topology.

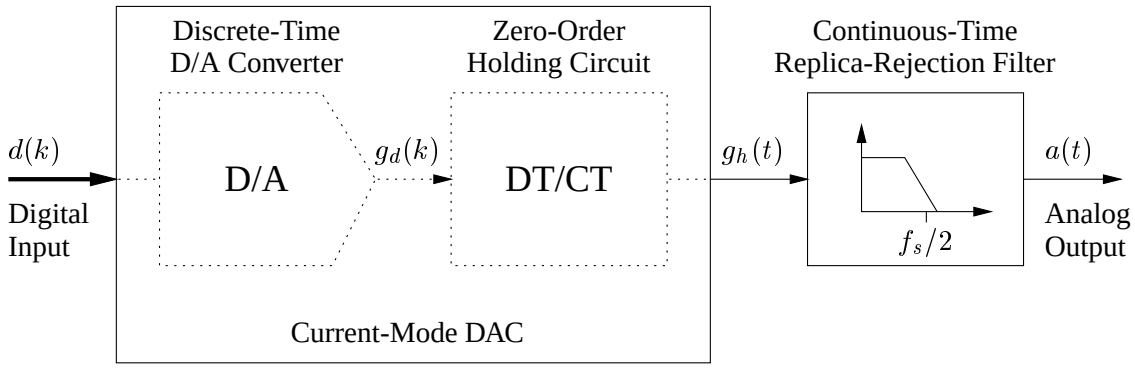


Figure 3.10: Current-mode D/A converter system.

Although other structures exist, such as R/2R and M/2M current splitters, the following will, without loss of generality, assume that the current-mode DAC is implemented using the current-steering principle shown in Figure 3.11. At the onset of each new sample, the array of switches is controlled according to the input code $d(k)$, thereby guiding the current from each element in the array of current sources to either the output terminal or to a current-dump node at the same potential. The switches remain in the same position until the onset of the next sample, therefore the output current will be a staircase current signal. The output current is typically processed as a continuous-time signal, possibly by a continuous-time replica-rejection filter.

Dynamic Errors. As discussed for voltage-mode D/A converters, the DT/CT conversion is a very critical process. Assuming temporarily that the static linearity of the DAC is ideal, the focus will now be on dynamic errors. Dynamic errors are caused by imperfect switching (glitches) when the output current is updated from one value to the next.

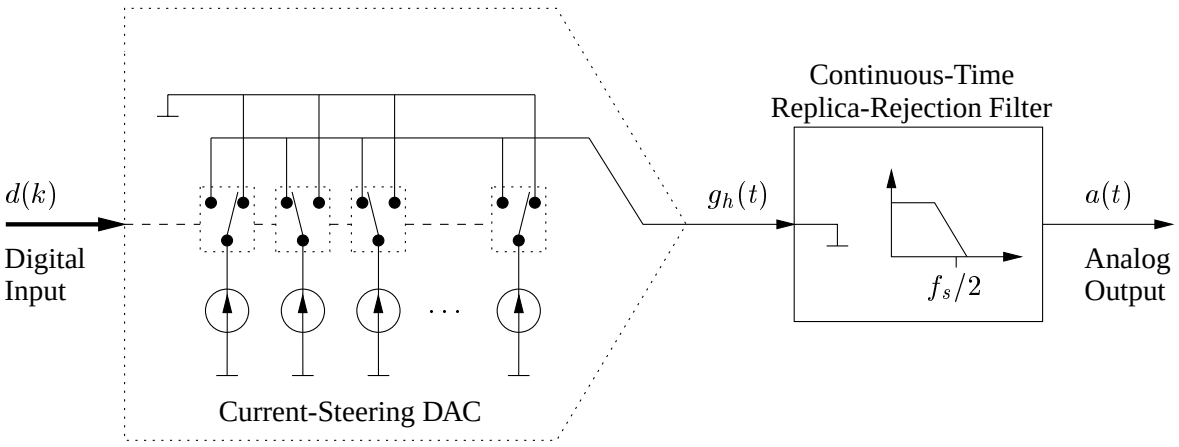


Figure 3.11: Typical current-steering D/A converter.

As dynamic errors occur only in the switching instances, their influence is highly dependent on the duration of the switching period relative to the sampling period T_s. Current-mode D/A converters are typically employed for high-speed and/or high-dynamic-range applications[7], so the switching period either has to be short, relative to a already short period T_s; or extremely short, relative to a (hopefully) somewhat longer period. Hence, dynamic errors are often a main concern when designing a current-mode DAC. Delta-sigma data converters that employ current-mode DACs are no exception to this point because they usually aim for a very high dynamic-range performance, and because a high degree of oversampling often will require the current-mode DAC to operate at high speed.

Dynamic Errors: Timing. One type of dynamic errors occurs if the timing is inaccurate, i.e., if the switches are updated at slightly shifted time instances. For example, if in a transition a current source is switched from the current-dump node to the output node slightly after the other switches have been updated, the output current will temporarily be too small.

Large timing-related dynamic errors are in general caused by poor layout, or by not including an array

[7]Current-steering switches can be made to operate *very* fast; hence high-speed operation is feasible. Assuming a small signal bandwidth (say less than 100kHz), the noise performance of current-mode DACs can generally be made significantly better than that of DACs which sample the signal as a voltage on a capacitor C_{sampl}, and thereby have an inband thermal-noise power of at least $\frac{1}{\mathrm{OSR}} \cdot \frac{kT}{C_{\mathrm{sampl}}}$

of latches to isolate the switches from the digital circuitry decoding the switch-control signals, but even the best current-steering DACs will (to some extent) be subject to this problem.

Dynamic Errors: Nonlinearity. The glitch caused by switching a current source is in often not perfectly proportional to the current that is switched. If the current-steering DAC is implemented with an array of binary-weighted current sources, the switching of each current source will be a nonlinear function of the input $d(k)$; hence the glitch in the output current will have a nonlinear relationship to the input. This effect can be avoided by using an array of equally-sized current sources [34].

Dynamic Errors: Intersymbol Interference. Stochastic variations in the production of integrated circuits will cause circuit imperfections, which imply that even fully-differential circuits will be subject to asymmetric switching. A simple example is shown in Figure 3.12, where the difference in the rise and fall times is shown exaggerated for the purpose of illustration.

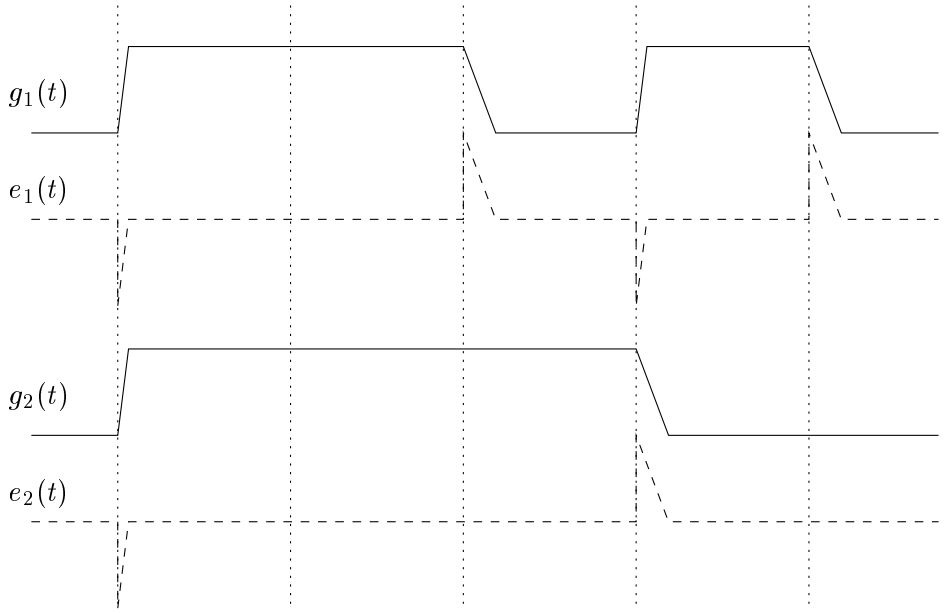


Figure 3.12: Output current obtained by switching the same current source for two different sequences: $\{0, 1, 1, 0, 1, 0\}$ and $\{0, 1, 1, 1, 0, 0\}$.

The Figure illustrates the current provided by a single switched current source. The current $g_1(t)$ results

from a digital input $d(k)$, where the corresponding switch is controlled by the sequence $\{0, 1, 1, 0, 1, 0\}$, whereas $g_2(t)$ results from the sequence $\{0, 1, 1, 1, 0, 0\}$. The glitch currents $e_1(t)$ and $e_2(t)$ represent for each of the two sequences the difference between the provided currents and the ideal two-level current signals.

When the rise time and the fall time are identical, the glitch currents are proportional to the first-order difference of the switching sequences, and hence the phenomenon can be made linear[8]. However, when the switching is asymmetric, nonlinearity will result.

For simplicity, consider just the average value of the signals. The two switching sequences have the same average value (one half), therefore linearity will require the two glitch currents to have the same average value. However, due to asymmetric switching, the average value of each glitch current will instead be linearly dependent on the number of rising and falling edges (cf. Figure 3.12). As $g_1(t)$ is switched more often than $g_2(t)$, the two glitch currents do not have the same average value, consequently the system is nonlinear. This effect is often referred to as *intersymbol interference*.

In order to avoid intersymbol interference, return-to-zero (RTZ) switching schemes have been introduced [35] [36] [37]. The basic principle is that the switches of a current-steering DAC are controlled according to the input $d(k)$ only during the first fraction, say 3/4, of each period T_s, and according to some fixed code in the remaining portion of each period. Figure 3.13 shows the output from the current source when a RTZ switching scheme is employed for the two signal sequences that were considered above. Notice that the average values of the two glitch currents, $e_1(t)$ and $e_2(t)$, are now the same. More important, notice that the glitch current will always be proportional to the signal that is being converted by the considered current source.

When using the RTZ switching scheme, the glitch current can be modeled simply as a small deviation of the static value of the considered current source, and hence the technique is an efficient remedy for dynamic errors caused by intersymbol interference. Phrased differently, the RTZ switching scheme assures that the DT/CT conversion is described by an impulse response, which (by definition) assures that the current source's control signal (bit signal) is D/A converted linearly.

[8] For example, by employing only unit-sized current sources as discussed above.

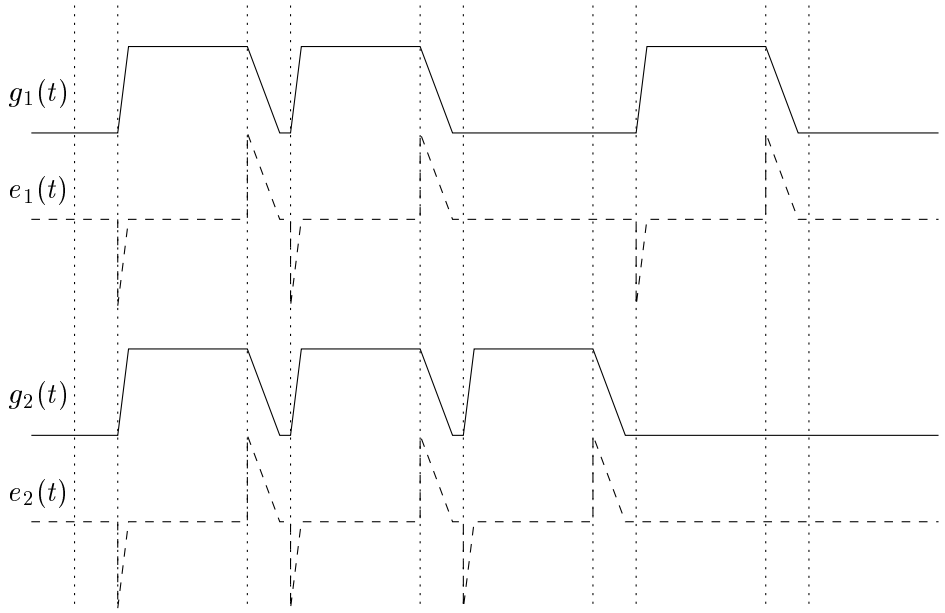


Figure 3.13: Output of a current source when using the return-to-zero switching scheme.

3.2.3 Clock Jitter in D/A Converters

Clock jitter does usually not cause dominating errors because high-quality clock signals can be generated using crystal-controlled clock generators. However, a D/A converter's robustness to clock jitter is highly dependent on the actual implementation, and for some high-performance DACs, clock-jitter-induced errors can easily become dominating.

Modeling Clock Jitter Errors. D/A converters are sensitive to clock jitter only in the DT/CT conversion process. Assuming at this point that the DT/CT converter is a zero-order holding circuit, the jitter-distorted continuous-time output signal $g_{\mathrm{jitter},h}(t)$ can be described as

$$g_{\mathrm{jitter},h}(t) = g_d(k), \quad \text{for} \quad kT_s + \Delta T(k) \leq t < (k+1)T_s + \Delta T(k+1) \tag{3.23}$$

where, for simplicity, it has been assumed that $\Delta T(k), k \in Z$ is a zero-mean stochastic process characterized by a standard deviation much less than the sampling period T_s.

The jitter-induced error signal $g_{\mathrm{jitter,error}}(t)$, i.e., the difference between $g_{\mathrm{jitter},h}(t)$ and the ideal output

$g_h(t)$ described by (3.21), is shown in Figure 3.14, and can be described as

$$g_{\text{jitter,error}}(t) = \begin{cases} -[g_d(k) - g_d(k-1)] & \text{for } 0 < t - kT_s < \Delta T(k) \\ [g_d(k) - g_d(k-1)] & \text{for } 0 > t - kT_s > \Delta T(k) \\ 0 & \text{otherwise} \end{cases} \tag{3.24}$$

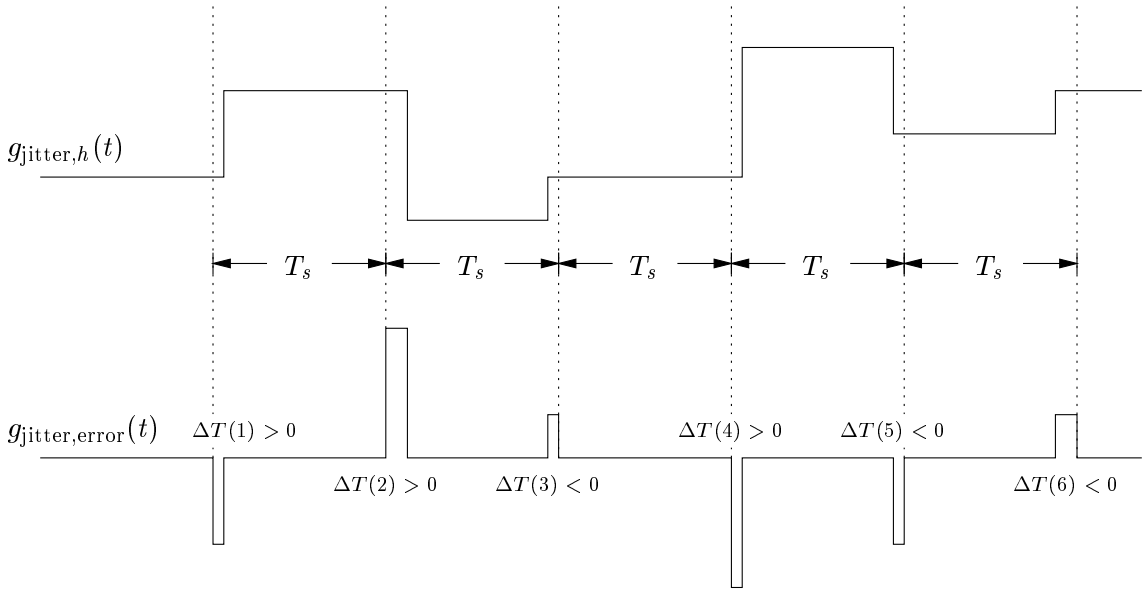


Figure 3.14: Clock jitter on output from zero-order holding circuit.

Clearly, the polarity of $g_{\text{jitter,error}}(t)$ depends on the polarity of $\Delta T(k)$, as well as the polarity of the sample-to-sample variation of $g_d(k)$. As $\Delta T(k)$, by assumption, is a zero-mean noise signal with a standard deviation much less than T_s, the error signal $g_{\text{jitter,error}}(t)$ will be a sequence of short-duration spikes, which for many purposes can be modeled as a sequence of impulses[9]. The strength of each impulse will equal the enclosed area; i.e., the jitter-induced error signal can be expressed as

$$g_{\text{jitter,error}}(t) = \sum_{k=-\infty}^{\infty} [g_d(k-1) - g_d(k)]\Delta T(k)\delta(t - kT_s) \tag{3.25}$$

[9] The impulse approximation (3.25) can be used to estimate the spectral composition of $g_{\text{jitter,error}}(t)$, only for frequencies which are somewhat lower than the reciprocal of the standard deviation of the stochastic process $\Delta T(k)$. In less-than-extreme cases, this model will be valid in the signal band. Hence, the approximation can be used to estimate the signal-band power of $g_{\text{jitter,error}}(t)$, but obviously, not to estimate the total power.

Notice that $g_{\text{jitter,error}}(t)$ is the time-domain equivalent[10] of

$$g_{\text{jitter,error}}(k) = [g_d(k-1) - g_d(k)]\frac{\Delta T(k)}{T_s} \tag{3.26}$$

and hence, the spectral composition of $g_{\text{jitter,error}}(t)$ can be found by convolving the Fourier spectrum of the first-order difference of $-g_d(k)$ with the Fourier spectrum of the relative-clock-jitter signal $\Delta T(k)/T_s$.

Clock jitter will include a flicker-noise component and a thermal-noise component. However, it is generally the jitter's thermal-noise component that may cause problems, and hence the flicker-noise component will be neglected in the following.

Given the assumption that $\Delta T(k)/T_s$ is a white-noise signal, i.e that it is a sequence of identically-distributed independent stochastic events, it follows that $g_{\text{jitter,error}}(k)$ will be a signal with uniform power density[11]. The total power of $g_{\text{jitter,error}}(k)$ will be the power of $\Delta T(k)/T_s$ multiplied by the average step size of $g_d(k)$.

Without taking extreme measures and for a sampling frequency in the few-MHz range [38], the stochastic element $\Delta T(k)/T_s$ can be made to have a standard deviation in the order of $50\text{pS} \cdot 2\text{MHz} = 10^{-4}$. This value will be used in the following evaluations to get a feeling of the problem's significance.

Clock Jitter in Voltage-Mode D/A Converters. Voltage-mode D/A converters will typically employ a zero-order holding circuit for the DT/CT conversion, and hence the above derivation will apply directly. The worst-case[12] average value of $|g_d(k-1) - g_d(k)|$ is Δ_{pp}/OSR, where Δ_{pp} is the full-scale peak-to-peak signal swing.

Knowing that the clock-jitter-induced error signal $g_{\text{jitter,error}}(t)$ has a uniform spectral power density,

[10] As defined by (2.9).

[11] Although $g_{\text{jitter,error}}(k)$ has uniform power density, it is not a white-noise signal. The individual samples are modulated with $[g_d(k-1) - g_d(k)]$, so it is not a stationary stochastic process.

[12] Assuming a full-scale signal at the highest signal-band frequency.

the worst-case *signal-band* power can be calculated as

$$\begin{aligned} P[g_{\text{jitter,error}}(t)] &< \widehat{\left(\frac{\Delta T(k)}{T_s}\right)^2} \left(\frac{\Delta_{pp}}{\text{OSR}}\right)^2 \frac{1}{\text{OSR}} \\ &\simeq 10^{-8} \cdot \frac{\Delta_{pp}^2}{\text{OSR}^3} \end{aligned} \tag{3.27}$$

where $\widehat{\left(\frac{\Delta T(k)}{T_s}\right)^2} \simeq 10^{-8}$ represents the above assumption of a fair estimate for what can be achieved without taking extreme measures, i.e. for what is considered to be a reasonable requirement for well-designed mass-produced electronic equipment. As the full-scale signal power is $\Delta_{pp}^2/8$, it follows that the signal-to-noise ratio (SNR) will be in the order of 10^7OSR^3. As discussed on page 43, voltage-mode DACs will usually operate on at least ten times oversampled signals, in which case the SNR estimate will be 100 dB.

The conclusion is that clock-jitter-induced errors typically can be made non-dominating in voltage-mode DACs of up to 16-bits of resolution (audio quality). Notice, however, that this is not the case if the signal is significantly less oversampled. It is quite difficult to meet the clock-jitter requirement for a 16-bit DAC operating at Nyquist rate, i.e., for OSR $= 1$. In this case, the clock-jitter requirement is just a few pS for a 2MHz clock signal – a level of performance which normally will require a stabilized laboratory test setup. This is yet another reason for why high-performance DACs must operate on somewhat oversampled signals.

Clock Jitter in Current-Mode D/A Converters. In principle, the clock-jitter sensitivity of current-mode DACs is the same as that of voltage-mode DACs. In reality, however, the situation is often much less favorable.

Consider a current-mode DAC, which is designed to meet the same specifications as those of a good voltage-mode DAC. High-performance current-mode DACs are usually designed to operate with a RTZ switching scheme in order to avoid intersymbol interference errors (cf. page 47). A problem arises because the RTZ switching scheme produces a signal which (for each sample) has two clock-jitter-prone edges instead of just one, and because the average step size (per sample) will be Δ_{pp} independent of the OSR. Hence, to meet the same specifications, the standard deviation of $\Delta T(k)/T_s$ must be a factor of

OSR smaller for the current-mode implementation than for the voltage-mode implementation. Because the improved thermal noise performance often is the main incentive to implement a DAC in current mode, the system specifications can be quite tough, therefore, the clock-jitter requirement can become extreme (a requirement of less than 1 pS of clock-jitter is not uncommon, but it is extremely difficult to fulfill).

In conclusion, the only way to avoid very strict clock-jitter requirements is to implement the current-mode DAC without RTZ switching, in which case other techniques must be employed to avoid inter-symbol-interference errors. Such alternative techniques are discussed in the second part of this thesis and in [35].

3.2.4 Static Performance of D/A Converters

The fundamental requirement of any D/A converter is that it should generate an analog signal which is proportional to the digital input. This section will define the DAC's static characteristic, its gain and offset, some terminology, and some measures for static DAC nonidealities.

Static Output Value. The static performance of a DAC is evaluated when the input is held constant. The static analog output corresponding to a considered code is defined as the average value of the provided output.

Resolution and Step Size. The set of permitted digital input codes is assumed to represent a set of integers, which are uniformly distributed with the step size Δ. The resolution is defined as the number of permitted input codes. The notation, "3-bit@Δ DAC," represents a DAC with eight permitted input codes representing integers spaced with the step size Δ.

Static Characteristic. The static characteristic is defined as the relation between each represented integer and the corresponding static analog output. The static characteristic of an N-level DAC can be described by a vector $S(n)$, where $S(1)$ is the static analog output for the smallest represented integer, and $S(N)$ is the static analog output for the largest represented integer (see Figure 3.15).

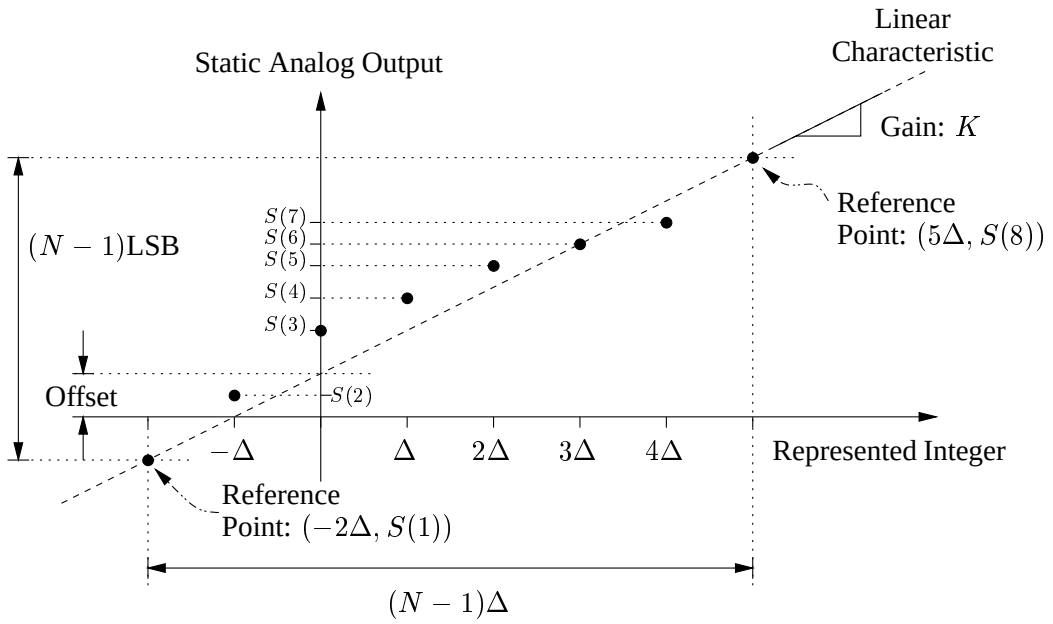


Figure 3.15: Static characteristic of an nonideal 3-bit@Δ DAC.

Linear Characteristic. The linear characteristic of an N-level@Δ DAC is defined on the basis of the two reference points shown in Figure 3.15. The analog quantity 1 LSB is defined as

$$1\text{ LSB} = \frac{S(N) - S(1)}{N-1} \tag{3.28}$$

and the DAC's gain K is defined as

$$K = \frac{1\text{ LSB}}{\Delta} = \frac{S(N) - S(1)}{(N-1)\Delta} \tag{3.29}$$

Measures for Static Nonlinearity. Equivalent to the definitions provided on page 40 for ADC DNL and INL performance, the following definitions apply for N-level DACs:

$$\text{DNL}(n) = [S(n+1) - S(n)] - 1\text{ LSB} \tag{3.30}$$

$$\text{INL}(n) = [S(n) - S(1)] - (n-1)\text{ LSB} \tag{3.31}$$

$$\text{DNL} = \log_2\left[\frac{[S(N) - S(1)][1 + 1/N]}{\max_n |\text{DNL}(n)|}\right]\text{ ENOB} \tag{3.32}$$

$$\text{INL} = \log_2\left[\frac{[S(N) - S(1)][1 + 1/N]}{\max_n |\text{INL}(n)|}\right]\text{ ENOB} \tag{3.33}$$

3.2.5 Linearity Limitations

This section will discuss the factors that limit a DAC's static linearity. The discussion will only address DACs that generate the analog output signal as a sum of one or more concurrent analog signals, such as, for example, the current-steering DAC shown in Figure 3.11.

Basic Principle for D/A Conversion. The basic principle for many D/A converters is shown in Figure 3.16.

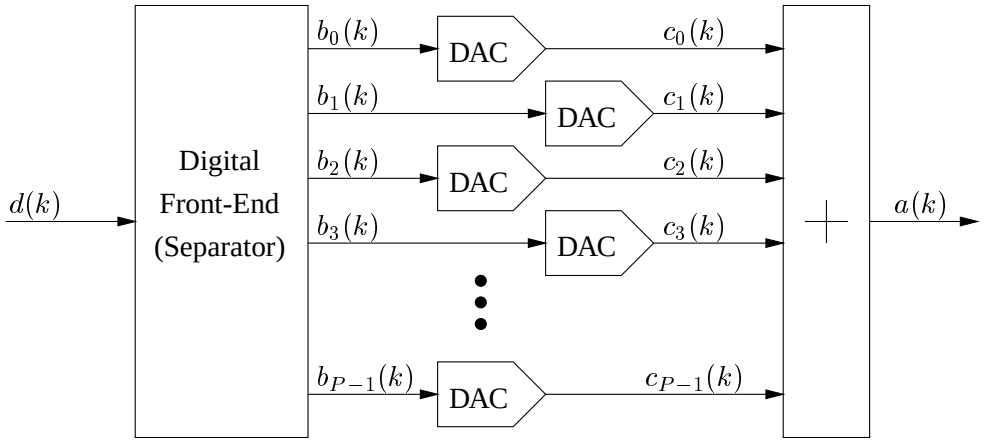


Figure 3.16: Topology of most D/A converters.

The digital front-end separates the digital input signal $d(k)$ into a sum of P digital signals:

$$d(k) = \sum_{i=0}^{P-1} b_i(k) \tag{3.34}$$

These P signals will in general be of low resolution, such that they are easy to D/A convert individually. The P DACs all have the same nominal gain K, therefore the analog output $a(k)$ can be formed simply by adding the generated analog signals $c_i(k)$

$$a(k) = \sum_{i=0}^{P-1} c_i(k) = \sum_{i=0}^{P-1} K b_i(k) = K d(k) \tag{3.35}$$

The digital front end can be implemented with arbitrary accuracy. Any nonideality of this type of D/A converter must, therefore, originate from either the individual D/A conversions or from the summing operation.

If the analog signals $c_i(k)$ are represented as currents, the summing operation can be implemented simply by connecting them to one common node (shown in Figure 3.11). Kirchoff's current law assures that the summation will be ideal for all practical purposes. However, because voltage signals are difficult to add accurately, voltage-mode DACs are typically implemented as switched-capacitor (SC) circuits, in which case the analog signals $c_i(k)$ are represented as charge pulses (which can be added as ideally as currents). The issue of accurately summing $c_i(k)$ is thereby replaced by the problem of accurately converting the discrete-time summed-charge signal into a discrete-time voltage signal. This design aspect is well understood, and circuits, the linearity of which is limited mainly by the linearity of the available capacitors, are known [39]. Although highly dependent on the technology used, poly-to-poly, metal-to-poly, and metal-to-metal capacitors will be up to 14-18 bits linear[13].

The sum of the individual DACs' offsets will be the overall DAC's offset, but they will not cause any other errors. As the offset of the overall DAC usually is not considered to be a critical parameter, offsets in the individual DACs are of only little concern.

In conclusion, the static linearity of DACs, which are implemented in the topology shown in Figure 3.16, is affected only by inaccurate gain and nonlinearities of the individual DACs.

Binary-Weighted D/A Converters. Usually, the digital input signal $d(k)$ is coded in the binary format, which makes binary-weighted DACs especially simple to implement. For such DACs, the digital front end is just a simple separation, where $b_0(k)$ is the most-significant bit of $d(k)$, $b_1(k)$ the second-most-significant bit of $d(k)$, etc.. Assuming that $d(k)$ is of five-bit resolution, $b_0(k)$ will attain only two values: 0 and 16; also $b_1(k)$ will attain only two values: 0 and 8, etc.. Each of the five DACs will, therefore,

[13]The linearity of a capacitor/resistor is defined as the linearity of the exploited part of the function describing the charge-to-voltage/voltage-to-current relationship. The DNL and INL linearity of a function $f(x)$, in a range of values $x \in \mathcal{J}$, is defined using the expressions (3.32) and (3.33). Here, $S(n) = f(x_n)$, where x_n is a set of $N \to \infty$ uniformly distributed values in $\mathcal{J}$. The text refers to the estimated INL linearity.

consist of only one analog source, which is turned on and off according to the corresponding signal $b_i(k)$. As the input signals $b_i(k)$ are binary scaled, so will the analog sources constituting the five DACs be binary scaled.

Now, consider the static linearity. Each of the five internal DACs is a two-level DAC, hence individually, they are all inherently linear[14]. In other words, each of the separated signals $b_i(k)$ will be D/A converted linearly, implying that only differences of the individual DACs' gain may cause static nonlinearity. Assume that the nominally identical gains of the five DACs are K_0, K_1, K_2, K_3, and K_4. The gain of the overall DAC will, therefore, be

$$\widehat{K} = \frac{16K_0 + 8K_1 + 4K_2 + 2K_3 + 1K_4}{16 + 8 + 4 + 2 + 1} \tag{3.36}$$

which leads to the following description of the output-referred nonlinearity error

$$\begin{aligned} a_{\text{error}}(k) &= a(k) - \widehat{K}d(k) - a_{\text{offset}} \\ &= \sum_{i=0}^{4} b_i(k)(K_i - \widehat{K}) \end{aligned} \tag{3.37}$$

Using Schwartz's inequality, it follows that

$$|a_{\text{error}}(k) - a_{\text{error}}(k-1)| \leq \sum_{i=0}^{4} |b_i(k) - b_i(k-1)|(K_i - \widehat{K}) \tag{3.38}$$

As each of the separated signals $b_i(k)$ are related to $d(k)$ by modulo functions, even small changes in $d(k)$ can result in large changes in $b_i(k)$, and consequently in large changes $a_{\text{error}}(k)$. In other words, binary-weighted DACs will typically exhibit a poor DNL performance, so they are not suitable for D/A conversion of large-dynamic-range signals.

Unit-Element DACs. A DAC with a good DNL is preferable for D/A conversion of large-dynamic-range signals. Unit-element DACs have this property.

Still referring to Figure 3.16, the digital front end of a N-bit@1 unit-element DAC separates the digital input into $(2^N - 1)$ signals $b_i(k)$, which all attain only two values: 0 and 1. An array of $(2^N - 1)$

[14]Because the static characteristic consists of only two points, which will always define a straight line.

nominally-identical DACs, each consisting of only one analog source, is used to generate the analog output $a(k)$ by turning on as many of these analog sources as the digital input $d(k)$ prescribes.

Assuming that the gains of the individual DACs are described by $K_0, K_1, \ldots, K_{(2^N-2)}$, the overall DAC's gain $\widehat{K}$ will be

$$\widehat{K} = \frac{\sum_{i=0}^{2^N-2} K_i}{2^N - 1} \tag{3.39}$$

which leads to the following description of the output-referred nonlinear error

$$\begin{aligned} a_{\text{error}}(k) &= a(k) - \widehat{K} d(k) - a_{\text{offset}} \\ &= \sum_{i=0}^{2^N-1} b_i(k)(K_i - \widehat{K}) \end{aligned} \tag{3.40}$$

If, as usual, the digital front end is of the thermometer-coding type

$$b_i(k) = 1 \quad \text{for} \quad i < d(k), \quad \text{and} \quad b_i(k) = 0 \quad \text{otherwise} \tag{3.41}$$

it implies that a small variation in $d(k)$ will cause a comparably small variation in $a_{\text{error}}(k)$ because only as many as $|d(k) - d(k-1)|$ terms are altered in the summation in (3.40).

More precisely, the DAC's DNL is simply the maximum absolute deviation from the average value of the analog sources. This will generally be 3-5 times the standard deviation of the individual analog source, which for modern technologies is typically less than 0.01 LSB.

Sectored DACs. Clearly, because the digital front end and especially the interconnection of unit-element DACs can easily become quite involved[15], they are more difficult to implement than their binary-weighted counterparts. Using the unit-element-DAC topology is often unnecessary, because a simple analysis will show that the stochastic properties[16] of the INL performance are the same as those of a binary-weighted DAC, and because only few applications will require a DNL performance as good

[15]Unit-element DACs are, however, not as complicated to implement as they may appear at first sight. Some details on this aspect can be found in [34].

[16]The stochastic properties of the INL will typically only depend on the output value, which will be the same.

as 0.01 LSB. Sectored DACs represent a compromise between complexity and DNL performance [40]. In these converters, the least-significant bits are D/A converted with binary-weighted DACs, whereas the most-significant bits are first thermometer coded and then D/A converted with unit-element DACs.

Matching of Analog Sources. The relative matching of electrical properties spatially distributed on an integrated-circuit surface depends somewhat on the layout technique. Any on-chip electrical parameter will usually be a function $f(x, y)$ of the on-chip coordinates as well as subject to a stochastic variation.

For a given area $\mathcal{A}$ used for the implementation of supposedly matched devices, the variation of $f(x, y)$ over $\mathcal{A}$ will typically be larger than the stochastic process' standard variation due to systematic errors and gradients in processing. Hence, to obtain good matching, one should strive to cancel the effect of $f(x, y)$.

As $f(x, y)$ is generally unknown, it has become common practice to cancel only the linear trend in $f(x, y)$ using the common-centroid layout principle, i.e., to lay out each matched element on an area $\mathcal{A}_i \subset \mathcal{A}$, such that the centroid (x_{cc}, y_{cc}) is the same for all elements

$$(x_{cc}, y_{cc}) = \left(\iint_{\mathcal{A}_i} x \, dxdy, \iint_{\mathcal{A}_i} y \, dxdy, \right) \tag{3.42}$$

Obviously, the common-centroid layout technique will provide the best cancellation when the matched devices are laid out on a small area $\mathcal{A}$. However, to assure a small standard deviation of the matching process' stochastic element, the matched devices should preferably be large. These two requirements are not necessarily contradictory because each device in a set of matched devices can be implemented as a parallel combination of several small devices, each small device being a matched element in one of several arrays of matched devices, each array being implemented in a small area.

In principle, arbitrarily good matching can be obtained using the above technique, but in reality, considerations such as chip area (cost), yield, power consumption, dynamic effects and/or stray capacitance will limit the obtainable relative matching.

Conclusion. The static linearity of the discussed types of DACs is singularly dependent on the relative accuracy with which an array of analog sources can be implemented. Whereas the DNL performance

can be made almost arbitrarily good by using a (partly) thermometer-coding digital front end, the INL performance can only be improved by improving the relative matching of the analog sources. When using a modern technology and good layout techniques, the INL performance can be made as good as 10–12 bits. Laser trimming and other once-and-for-all post-processing calibration techniques can be used to obtain better matching, but they are expensive to perform and they lack the ability to track effects due to, for example, aging and temperature. Better alternatives include power-up and background calibration, but these techniques tend to require a relatively large chip area for their implementation, and they are subject to leakage effects if the compensation parameter is stored in analog form on a capacitor.

3.3 Measuring Dynamic Performance

Unless a data converter is used for ultra-low-bandwidth applications, its linearity may not be fully described by its static performance measures, i.e., its DNL and INL description. For high-speed data conversion, it is the dynamic performance rather than the static performance which is of interest.

Several factors may limit the dynamic performance. Examples include nonlinearities of the filters, S/H circuits, and DT/CT converters used; clock-jitter effects; device noise and noise coupling; etc.. Because static nonlinearity also will manifest itself as dynamic nonlinearity, a thorough description of a converter's dynamic performance avoids the need for a static linearity description.

The dynamic performance is best described in the frequency domain. The data converter is operated under certain test conditions, and the performance is evaluated as the output-referred ratio of the applied signal's power relative to the power of the considered error. To ease the process of separating the output signal's power in terms of "signal" and "error," the applied test signal will typically consist of one or more sinusoids.

As the dynamic performance may depend significantly on the test conditions, the measurements should be repeated for many sets of test conditions, and the results presented in a graphical format. Obviously, full particulars of the measurement technique and test conditions used are an essential part of the results.

3.3.1 Signal-to-Noise Ratio

Noise is the term often used to indicate those errors for which the source is assumed to be of stochastic nature. Because noise in general will have to be considered in its wide sense, it need not be independent of the input signal. Noise sources include thermal and flicker device noise, substrate- and capacitively-coupled noise, clock-jitter-induced noise, and quantization noise.

The signal-to-noise ratio (SNR) is usually measured for a sinusoid input signal and plotted as a function of the input signal's amplitude. Harmonic distortion and other errors, which are described by separate measures, are not included in the estimated error power[17].

If the noise power is signal-independent, which is the case for device noise, the SNR will be linearly dependent on the input signal's magnitude. If the SNR saturates a given level, it indicates that clock-jitter-induced noise dominates from that level and up. For single-bit delta-sigma modulators, the SNR will not only saturate but drop for input signals above a certain level. This occurs when the internal one-bit quantizer becomes overloaded, in which case the quantization-noise power increases faster than the signal power.

The signal-band noise power is calculated by integration of the estimated spectral power density over the signal band, hence a wide-bandwidth application will be relatively more sensitive to noise than a narrow-bandwidth application[18]. The lower limit (in frequency) of flicker noise is a somewhat tricky issue, but it is not important for the following discussion.

3.3.2 Dynamic Range

The dynamic range (DR) is defined on basis of the SNR. The DR is the ratio of the magnitude of the largest sinusoid input that does not cause clipping or other coarse nonlinear effects, relative to the magnitude of the sinusoid input for which the SNR is 0 dB.

[17]The signal-to-noise-and-distortion ratio (SNDR) is a composite measure, for which the error consist of all signal-band spectral components which are not in linear relation to the applied input signal.

[18]Also, as the total signal-band power of the sampling thermal noise (kT/C noise) in switched-capacitor circuits is inversely proportional to both the capacitor size and the oversampling ratio, large capacitors are required for SC circuits with low OSR.

3.3.3 Spurious-Free Dynamic Range

A full-scale sinusoid of frequency f_0 is applied as test input, and the output signal is evaluated at all frequencies in the signal band. The ratio of the estimated power density of the output signal's spectral component centered at f_0 and the largest other spectral component is called the spurious-free dynamic range (SFDR). Often, the SFDR is close to the reciprocal of the total harmonic distortion (THD), but in some implementations the largest spurious spectral component occurs at a frequency which is not in a harmonic relationship to f_0.

3.3.4 Intermodulation Distortion

Simple harmonic-distortion measurements are not suitable to evaluate high-frequency linearity. The problem is that the dominating harmonic may occur at a frequency outside the signal band, therefore, if out-of-band spectral components are rejected, for example, by low-pass filtering, the measurement may provide misleadingly optimistic results. To avoid this scenario, the linearity should be evaluated as a signal-to-distortion ratio where the distortion is the intermodulation product occurring in the signal band.

3.4 Quantizer Topologies

Again addressing the implementation of A/D converters, this section will provide an overview of some commonly used quantizer topologies. Each topology is associated with advantages and disadvantages in terms of linearity, speed, power consumption, complexity, and latency.

Although quantizers can be classified in several ways, the following will consider quantizers to be separated into two main categories: data quantizers and signal quantizers. A data quantizer is defined as a quantizer that evaluates each input sample as accurately as possible, and which does not take any sample-to-sample correlation into account. In other words, a data quantizer's output approximates the time-domain description of the received signal (cf. Section 2.1). A signal quantizer, on the other hand,

is defined as a quantizer that evaluates the input signal's spectral composition as accurately as possible. Each sample may or may not be quantized accurately, but the quantizer has memory and it assures that the sequence of errors, i.e., the error signal, has very little energy in the signal band. In other words, a signal quantizer approximates the signal-band frequency-domain description of the received signal (cf. Section 2.2).

Because the Fourier transformation is a bijective[19] relation between the time domain and the frequency domain, an ideal data quantizer will yield exactly the same output signal as an ideal signal quantizer. The difference will show only in the presence of quantizer nonidealities, where a data quantizer can produce a substantial amount of signal-band quantization noise and harmonic distortion, whereas a signal quantizer will suppress such errors in the signal band by moving the energy to other frequencies outside the signal band. Clearly, a signal quantizer will require a minimum degree of oversampling. Elsewhere, they are sometimes referred to as oversampling quantizers.

It is the application that determines which type of quantizer should be used. For example, if the application is to measure the resistance of a large number of produced resistors to facilitate estimation of the production's standard deviation, the resistance of each resistor represents data which should be evaluated using a data quantizer and not a signal quantizer. However, the majority of applications process signals rather than data; in such cases a signal quantizer may be the better choice.

3.4.1 Direct-Comparison Data Quantizers

Data quantizers will be classified as one of two types, one of which is called *Direct-Comparison* quantizers. Direct-Comparison quantizers are all characterized by the property that the (possibly sampled-and-held) input signal is quantized by multiple comparisons with analog signals generated by D/A conversion.

[19] I.e. there is a one-to-one correspondence.

Counting Quantizers. Extremely good linearity[20] can be obtained using quantizers of the counting type, such as dual-slope and incremental [41] quantizers, but the achievable bandwidth is so low that they are useful mainly for instrumentation and calibration purposes, which nonetheless are important applications for data converters.

Successive-Approximation Quantizers. Higher bandwidth and good resolution can be obtained using successive-approximation quantizers, where each input sample is digitized by generating a sequence of successively improved approximations. Each approximation is D/A converted, and the output is compared with the sampled-and-held input. The result of each comparison is used as the basis for the next supposedly-improved approximation.

In principle, the uncertainty of each new approximation in the sequence of approximations is reduced by a factor of two[21], but (obviously) the uncertainty of any approximation cannot be reduced to less than the uncertainty by which the approximation is D/A converted, that is, the employed DAC's INL performance. In other words, the linearity of a successive-approximation quantizer is limited by the linearity of the employed DAC, which typically will be in the order of 10 bit. Assuming that only 1 bit is resolved for each new approximation, ten cycles will be necessary to resolve an input sample for 10-bit conversion, which, combined with a minimum degree of oversampling[22] (see section 3.1.1), will limit the bandwidth to less than one percent of the cycle frequency, i.e. , typically well below 1MHz.

Flash Quantizers. Flash quantizers are among the fastest quantizers available. By means of (for example) a resistor string, N supposedly equidistant analog reference signals (say voltages) are generated. At the sampling instances, the input signal is simultaneously compared to each and all of these N reference signals, thereby producing N one-bit digital signals, which jointly represent the quantized value in the thermometer-coded format. A digital back end is in general used to recode the digital signal into the

[20] In excess of 20 bits.

[21] This is the typical case, but more than one bit can be resolved in each cycle.

[22] This, of course, only makes sense if the quantizer is used for a signal-quantizer application.

more dense binary-coded format[23].

As N comparators are required for the implementation of an N-level quantizer, flash quantizers typically have a fairly low resolution, say less than 8 bits. The main sources of nonlinearity include non-equidistant spacing of the generated analog signals (DAC nonlinearity) and the offsets of the individual comparators. Interpolation techniques can, however, be used efficiently to reduce the offset-induced nonlinearity. Depending on the actual implementation[24], the INL performance is typically in the order of 5–10 bits.

Flash quantizers can be designed for high-speed quantization. Sampling frequencies well beyond 100 MHz are achievable, but their main limitations are high circuit complexity and power consumption, low resolution, and somewhat poor linearity.

Subranging Quantizers. Subranging quantizers represent a compromise between conversion speed and circuit complexity. The sampled-and-held input signal is first compared to N voltages equidistantly spaced in the full range, whereafter the same N comparators are used to compare the input signal with N voltages equidistant spaced in a smaller range centered around the voltage estimated in the first quantization. In this way, an N^2-level quantizer can be implemented using only N comparators. These quantizers can be designed to have an INL linearity in the order of 8–10 bits, and they are often employed in portable video equipment.

3.4.2 Residue-Calculating Data Quantizers

The second type of data quantizers is called *Residue-Calculating* quantizers. They are, in principle, all based on the simple residue-calculating stage shown in Figure 3.17. The stage's input signal $g(t)$ is quantized by a (typically flash-type direct-comparison) quantizer providing the digital output signal $d(k)$. The signal $d(k)$ is D/A converted and the result is subtracted from the stage's input signal $g(t)$ to

[23]The digital back end may also include some logic to prevent the deleterious effects from "bubbles" in the thermometer codes.

[24]Although it may seem unnecessary, a good S/H circuit is often required to obtain good linearity.

produce $\widehat{r}(k)$. This operation can be described as

$$g(k) = \widehat{r}(k) + d(k)K_{\rm DAC} + O_{\rm DAC} + {\rm INL}_{\rm DAC}[d(k)] \tag{3.43}$$

where $K_{\rm DAC}$ is the DAC's gain, $O_{\rm DAC}$ is the DAC's offset, and ${\rm INL}_{\rm DAC}[d(k)]$ represents the DAC's INL error for the codes representing $d(k)$. Notice that $\widehat{r}(k)$ is an estimate of the residue $r(k)$ of the quantization $d(k)$ of $g(k)$ *calculated with respect to the linear characteristic defined by the DAC* (cf. Equation (3.9) where $g_{\rm offset} = -O_{\rm DAC}$ and $K = 1/K_{\rm DAC}$).

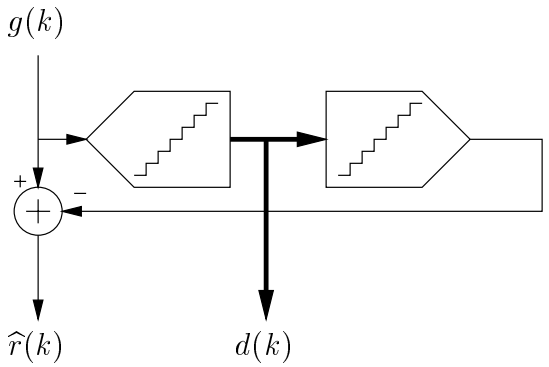


Figure 3.17: Basic residue stage used in most residue-calculating quantizers.

The overall point is that, if $\widehat{r}(k)$ is quantized by a quantizer with gain $K_1 = 1/K_{\rm DAC}$, so that $d_1(k) = \widehat{r}(k)/K_{\rm DAC}$, it follows from (3.43) that

$$\begin{aligned} g(k) &= d_1(k)K_{\rm DAC} + d(k)K_{\rm DAC} + O_{\rm DAC} + {\rm INL}_{\rm DAC}[d(k)] \\ &= [d_1(k) + d(k)]K_{\rm DAC} + O_{\rm DAC} + {\rm INL}_{\rm DAC}[d(k)] \end{aligned} \tag{3.44}$$

Equation (3.44) implies that, given the above assumption, the linear characteristic and the nonlinearity of the quantizer, evaluated from $g(k)$ to $[d(k) + d_1(k)]$, are defined only by the employed DAC. Thus, a residue-calculating quantizer can, in principle, be implemented with the same linearity as that of a DAC.

Obviously, it is preferable to match the quantizer's linear characteristic to the DAC's linear characteristic[25], in which case $|\widehat{r}(k)| < 0.5\ {\rm LSB}_{\rm DAC}$, but this is actually not a strict requirement, because mismatch only implies that the magnitude of $\widehat{r}(k)$ will increase. In other words, assuming that the

[25]Such that the combined circuit is a quantizing unity-gain element.

resolving range of the quantizer quantizing $\widehat{r}(k)$ is large enough to resolve $\widehat{r}(k)$ with the required accuracy, (3.44) will hold independent of any nonidealities of the employed quantizer quantizing $g(k)$. Such residue-calculating quantizers, for which $\widehat{r}(k)$ is resolvable in a range larger than $\pm 0.5\,\mathrm{LSB}_{\mathrm{DAC}}$, are said to employ *digital correction*. Notice that digital correction does not refer to the use of calibration techniques, but only that the residue-calculating quantizer is made somewhat insensitive to nonidealities of the internally employed quantizers.

Two-Step Flash Quantizers. The simplest residue-calculating quantizer is the two-step flash quantizer shown in Figure 3.18 for 7-bit conversion. The residue stage provides a four-bit quantization $d_0(k)$ of $g(k)$ with residue $r_0(k)$, defined with respect to the DAC's linear characteristic.

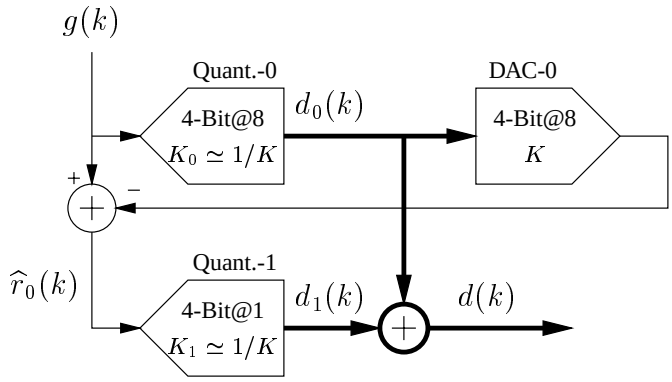


Figure 3.18: Two-step flash quantizer.

To minimize the magnitude of $\widehat{r}_0(k)$, the first quantizer's nominal gain should match the reciprocal of the DAC's gain K. As this cannot be obtained ideally, the magnitude of $\widehat{r}_0(k)$ will exceed $4K$. Digital correction is used to compensate for this problem, which is why the second quantizer is designed with a resolving range from $-8K$ to $+8K$. Thus, although two 4-bit quantizers are used, this structure only provides 7-bit@1 quantization.

Roughly speaking, the two-step quantizer will have an INL performance[26] described by

$$\mathrm{INL}_{\mathrm{Two-Step}}[d(k)] = \mathrm{INL}_{\mathrm{DAC-0}}[d_0(k)] + \mathrm{INL}_{\mathrm{Quant.-1}}[d_1(k)] + (1/K_1 - K)\cdot d_1(k) \qquad (3.45)$$

[26] Defined with respect to the DAC's linear characteristic.

The first term in (3.45) represents the nonlinearity induced by calculating the residue $\widehat{r}_0(k)$; the second term represents the nonlinearity of the second quantization; and the last term represents the nonlinearity induced by mismatch of the DAC's gain and the reciprocal of the second quantizer's gain[27].

The INL of the second quantizer is expected to dominate in (3.45), because it reflects stochastic offsets in an array of comparators. This term can, however, be suppressed by using the scaling technique shown in Figure 3.19. The idea is to multiply the estimated residue $\widehat{r}_0(k)$ by a factor of (say) eight, and then convert it with a quantizer whose gain is the same factor eight smaller.

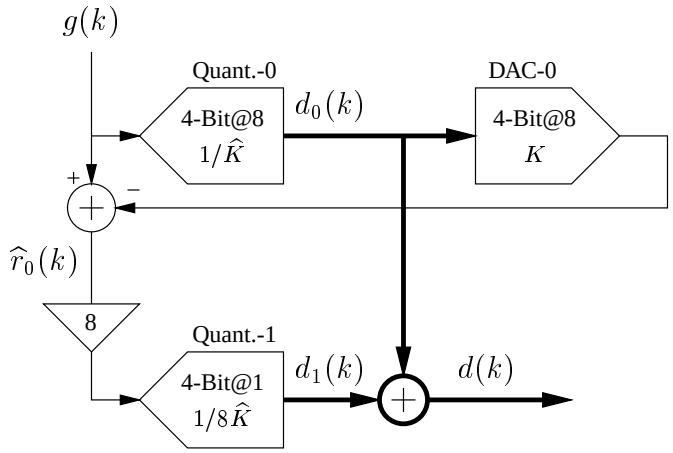


Figure 3.19: Scaled two-step flash quantizer.

The two-step quantizer's nonlinearity will then be described by

$$\mathrm{INL}_{\mathrm{Two-Step}}[d(k)] = \mathrm{INL}_{\mathrm{DAC-0}}[d_0(k)] + \frac{\mathrm{INL}_{\mathrm{Quant.-1}}[d_1(k)]}{8} + \Delta_{\mathrm{gain}} \cdot d_1(k) \tag{3.46}$$

As expected, the two-step quantizer is now less sensitive to nonlinearity of the second quantizer. The gain error $\Delta_{\mathrm{gain}} \cdot d_1(k)$ is determined by the inaccuracy of the gain from $\widehat{r}_0(k)$ to $d_1(k)$, i.e. the term includes the inaccuracy of the analog-domain gain block.

Pipeline Quantizers. Two-step quantizers can be generalized to become multiple-step quantizers, simply by employing multiple residue stages. This is, however, seldom implemented, because of timing con-

[27] Although it is a simple gain mismatch, it is indeed a nonlinear error, because $d_1(k)$ is a nonlinear (modulo) function of $g(k)$.

siderations. Clearly, after the first quantizer is strobed, the circuit must be allowed a minimum amount of time to calculate the scaled residue before the second quantizer is strobed. Thus, multiple-step quantizers will require a significant period of time to perform the quantization, and hence the maximum sampling frequency will be greatly reduced. Pipelining is an efficient technique to avoid this scenario.

The concept of pipelined residue-calculating quantizers is illustrated in Figure 3.20, which shows a four-stage 12-bit@1 pipeline quantizer. The key point to notice is that delay blocks are inserted between each of the four stages. In the analog domain, the delay elements are implemented as S/H circuits, whereas in the digital domain they are implemented as latches. The input signal $g(k)$ is quantized in four stages, and as the digital output $d(k)$ cannot be calculated before the fourth quantization is performed, this pipeline quantizer will cause at least 3 samples of latency. The sampling frequency is, however, independent of the number of stages, because all the internal quantizers are strobed simultaneously.

The pipeline quantizer shown employs digital correction, because each stage's residue is resolved in a range which is 25% wider than strictly necessary. Scaling is used to reduce the quantizer's sensitivity to the nonidealities of the second, third, and fourth stage. The INL performance is limited mainly by the nonlinearity of the first-stage DAC and the gain error of the three-stage quantizer which resolves the first stage's residue. Optimization of a pipeline quantizer's topology will make its linearity dependent almost exclusively on the first-stage DAC's INL. This can be obtained by increasing the resolution of the first-stage quantizer[28].

Other Residue-Calculating Data Quantizers. The so-called algorithmic quantizers are very similar to residue-calculating pipeline quantizers (cf. Figure 3.20), where the only difference is that the same residue stage is used for all of the partial quantizations. Algorithmic quantizers will be significantly slower than their pipeline counterparts because the conversion of each input sample must be completed before the conversion of the next sample can be initiated.

Folding-and-interpolating quantizers form a special class of the two-step-flash quantizers, which is not based directly on the residue-calculating stage. Although they have some significant advantages, they

[28]The required resolution is dependent on the technology's matching performance, but 5-bit resolution is a good starting point, see [42].

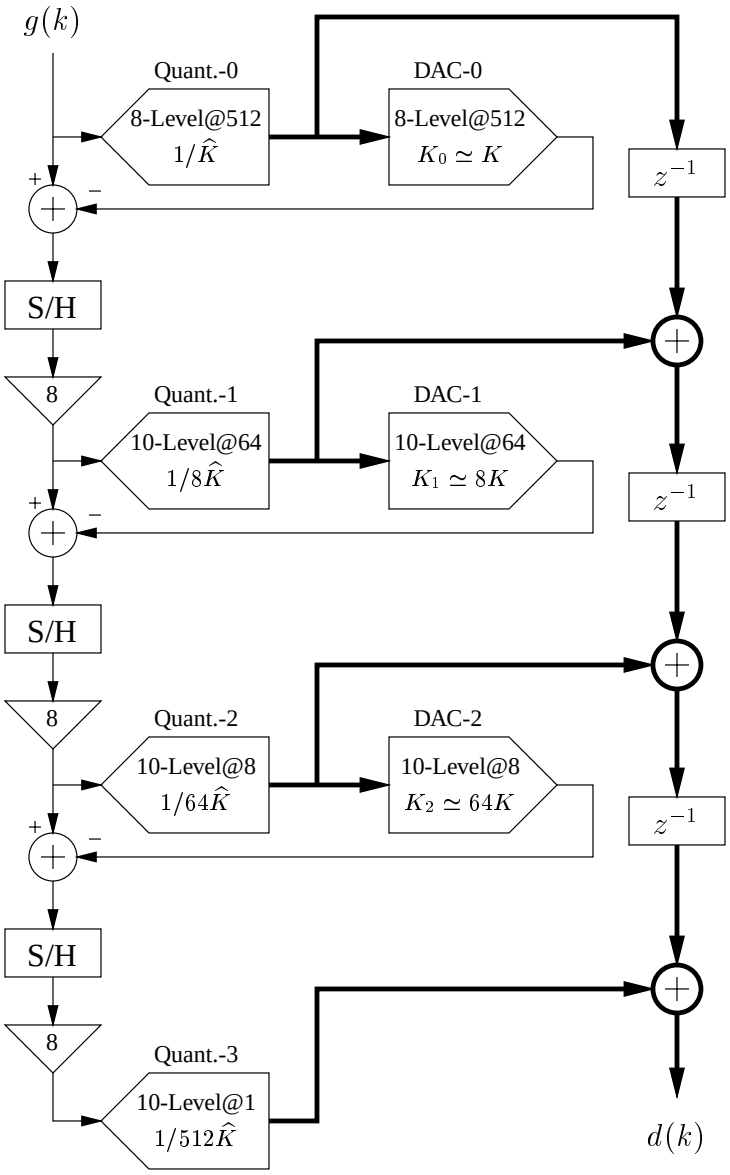


Figure 3.20: Four-stage pipeline quantizer.

will not be discussed in detail, because they aim at high-speed rather than high-performance quantization. Roughly speaking, the design tradeoffs are similar to those of two-step-flash quantizers.

3.4.3 Introduction to Signal Quantizers

Signal quantizers will always require a minimum amount of oversampling[29], and they will always be of the residue-calculating type, see Figure 3.21.

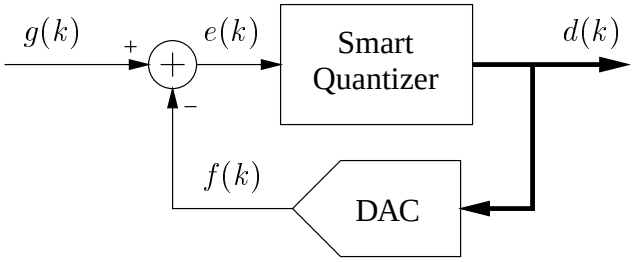


Figure 3.21: Fundamental principle of signal quantizers.

Assume that the digital output signal $d(k)$ has been generated somehow by the smart quantizer. The residue of this quantization, which for signal quantizers (of historic reasons) will be called the *error signal* $e(k)$, is calculated by D/A conversion of $d(k)$ and by subtracting the result $f(k)$ from the analog input $g(k)$. The fundamental concept, common for all signal quantizers, is that a smart quantizer selects the output signal $d(k)$ in such a way that the error signal's signal-band power is made very small, ideally zero. Assuming that the smart quantizer is successful in this aspect, it is thereby assured that the frequency spectra of $g(k)$ and $f(k)$ are equivalent in the signal band. Further, assuming that the DAC is ideal, i.e., $f(k) = d(k)K_{\mathrm{DAC}}$ where K_{DAC} is the DAC's gain, it follows that the frequency spectra of $g(k)$ and $d(k)K_{\mathrm{DAC}}$ are equivalent, and consequently that the signal quantizer is ideal (in the signal band) and described by a gain of $K = 1/K_{\mathrm{DAC}}$.

[29]It is, however, not quite clear how low that minimum is. Even if it would be possible, it would be meaningless to reduce the oversampling ratio to (say) 1.5, because other elements in an A/D converter system will require the OSR to be higher (cf. Section 3.1.1). An OSR in the order of ten is assumed to be sufficiently low not to impose intolerable restrictions on the circuit's application range.

Nonideal DAC Effects Because all signal quantizers are of the residue-calculating type, they are sensitive to DAC nonidealities. Figure 3.22 shows a model of the signal quantizer, where DAC nonidealities are taken into account.

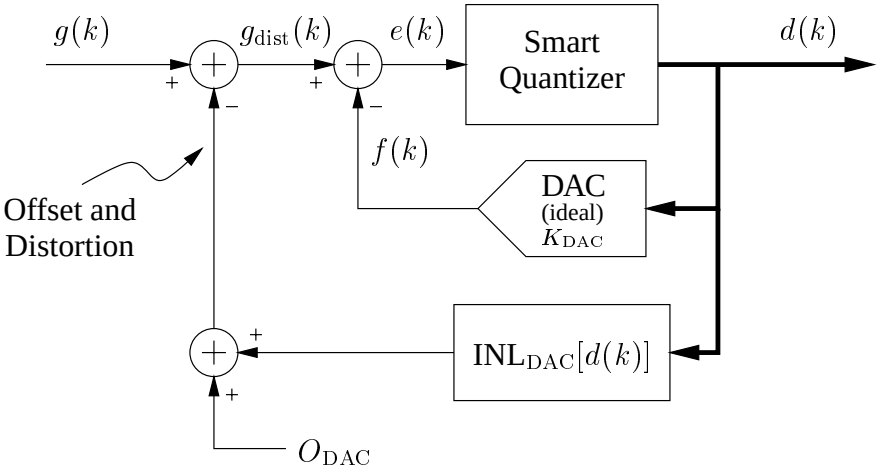


Figure 3.22: Nonidealities caused by the feedback DAC.

As shown in Figure 3.22, the DAC's offset and nonlinearity adds offset and distortion directly to the input signal $g(k)$. It is the sum $g_{\text{dist}}(k)$ which is quantized by the assumed ideal signal quantizer. In other words, any signal quantizer will be as nonlinear as its internal DAC which is used to calculate the residue of the quantization. This property is shared by all residue-calculating quantizers, including residue-calculating data quantizers. A main advantage of signal quantizers is, however, that the error signal $e(k)$ does not have to be minimized in each sample because the smart quantizer can relocate the corresponding error power to frequencies outside the signal band. In other words, $d(k)$ can be a low-resolution signal, and in particular, it can be a two-level signal.

The use of two-level output signals has for a long time been the preferred choice, because the DAC's INL error thereby will be zero[30]. In this way, DAC nonidealities will cause only gain and offset errors,

[30]For $N = 2$, the $\text{INL}(n)$ consists of only those two points that define the DAC's linear characteristic, i.e., $\text{INL}(1) = \text{INL}(2) = 0$. The same conclusion also follows from a simple manipulation of equations (3.28) and (3.31). Although the DAC's static performance will be perfectly linear, certain practical rules must be observed to obtain good dynamic performance (of the one-bit DAC), which is a fundamental requirement in order to obtain good static and dynamic performance of the quantizer (see [1]).

whereby the implementation of inherently-linear quantizers is possible.

Nonideal Smart Quantizer Effects. While still allowing the smart quantizer to be somewhat unspecified, the consequences of its potential nonidealities will be considered.

The main purpose of the smart quantizer is to minimize the signal-band power of the error signal $e(k)$. The success of this operation can be estimated from

$$P_{\text{inband}}[e(k)] = \lim_{N\to\infty}\left[\frac{1}{NT_s}\int_{-f_b}^{f_b}|E_{\text{obs}}(f)|^2 df\right] \tag{3.47}$$

where $E_{\text{obs}}(f)$ is the Fourier spectrum of $e(k)$ observed in a period of N samples, and where the signal band is assumed to be the baseband $|f| < f_b$.

Because signal quantizers need not have a good sample-to-sample equivalence between the analog input $g(k)$ and the digital output $d(k)$, they are best evaluated using measures of their dynamic performance (cf. Section 3.3). Thus, signal quantizers are typically described by their signal-to-error ratio (SER), which can be estimated from

$$\text{SER} = \frac{P_{\text{inband}}[g(k)]}{P_{\text{inband}}[e(k)]} = \lim_{N\to\infty}\left[\frac{\int_{-f_b}^{f_b}|G_{\text{obs}}(f)|^2 df}{\int_{-f_b}^{f_b}|E_{\text{obs}}(f)|^2 df}\right] \tag{3.48}$$

where $g(k)$ and $e(k)$ are observed in the same period of N samples.

For every 6 dB increase in the SER, the effective resolution of the signal quantizer is said to increase by one bit. Proportionality can be assumed for high-resolution quantizers, i.e., a quantizer described by a SER of 96 dB is said to have an effective resolution of 16 bits.

The SER measure is elsewhere called the quantizer's signal-to-noise ratio (SNR). This practice has not been adopted herein because it tends to lead to confusion. The source of $e(k)$ is in general well known, and its stochastic nature is, at best, only pseudo-random. Even when the smart quantizer is designed such that $e(k)$ does not include any detectable harmonics of $g(k)$ (see [43]), the spectral composition of $e(k)$ is quite likely to be tonal. Furthermore, the time-domain equivalent of the signal-band part of $e(k)$ can have a peak-to-peak value which is significantly larger than 1 LSB, defined with respect to the quantizer's

effective resolution[31] (see Chapter 3 in [1]). These problems are most serious when $d(k)$ is a single-bit signal, in which case the smart quantizer should be designed for a much better SER performance than the SNR performance (limited by device noise). Although truly-stochastic noise cannot mask that the error signal is tonal, this approach tends to make such quantizers more suitable for critical applications (e.g. audio).

Design of Smart Quantizers. This section will discuss some of the properties that the smart quantizer must possess, and how it can be implemented.

As discussed in Section 2.3, the accuracy with which a signal's spectral composition can be estimated is highly dependent on the duration of the observed signal. Thus, to effectively minimize the signal-band power of $e(k)$, the smart quantizer must choose each sample of $d(k)$ on the basis of $e(k)$ observed for many (typically several thousand) samples. In other words, the smart quantizer must have memory.

In order to allow the error signal $e(k)$ to be nonzero, the smart quantizer should minimize only the signal-band spectral components of $e(k)$, i.e. , it must allow $e(k)$ to have substantial out-of-band power. Equivalent to the band-pass-filter estimation method (see page 22), a filter $H(f)$ is used to focus on the spectral components of interest, i.e., the signal-band spectral components. The requirement of a long observation time, combined with the preference for simple implementations, leads to the conclusion that the filter should be of the infinite-impulse-response (IIR) type.

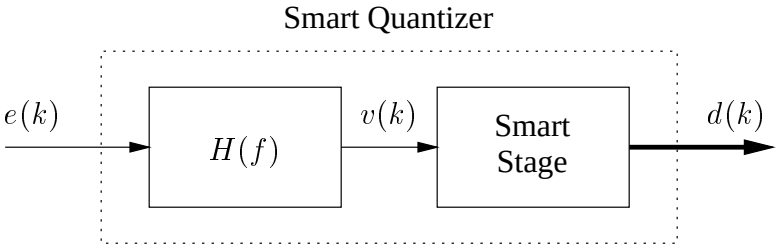


Figure 3.23: Fundamental elements in a smart quantizer.

Figure 3.23 shows the basic implementation of a smart quantizer, where the error signal $e(k)$ is filtered

[31] In other words, the signal-band part of $e(k)$ is often described by a large peak-to-rms ratio (it includes spikes), which reflects that $e(k)$ is not modeled well by a Gaussian stochastic process.

by the filter $H(f)$, and a "smart stage" evaluates the filter's output signal $v(k)$ (and possibly more) in the process of choosing the digital output signal $d(k)$. By noticing that $V(f) = H(f)E(f)$ and revisiting the measure for the smart quantizer's performance (3.47), the implementation can be evaluated using

$$\begin{aligned} P_{\text{inband}}[e(k)] &= \lim_{N\to\infty}\left[\frac{1}{NT_s}\int_{-f_b}^{f_b}|E_{\text{obs}}(f)|^2 df\right] \\ &= \lim_{N\to\infty}\left[\frac{1}{NT_s}\int_{-f_b}^{f_b}\frac{|V_{\text{obs}}(f)|^2}{|H(f)|^2} df\right] \end{aligned} \tag{3.49}$$

Assuming that the filter $H(f)$ has some large minimum signal-band gain

$$H_{\min} = \min_{|f|<f_b}|H(f)| \tag{3.50}$$

it follows that (3.49) can be evaluated by the inequality

$$\begin{aligned} P_{\text{inband}}[e(k)] &< \frac{\lim_{N\to\infty}\left[\frac{1}{NT_s}\int_{-f_b}^{f_b}|V_{\text{obs}}(f)|^2 df\right]}{{H_{\min}}^2} \\ &= \frac{P_{\text{inband}}[v(k)]}{{H_{\min}}^2} \end{aligned} \tag{3.51}$$

Equation (3.51) reflects that in order to minimize the signal-band power of $e(k)$, one could use a filter $H(f)$ with a large signal-band gain and attempt to keep the signal-band power of $v(k)$ small (or at least finite). These conditions will, for example, be fulfilled if the feedback loop is stable and $d(k)$ is generated simply by quantizing $v(k)$. This most typical structure is shown in Figure 3.24.

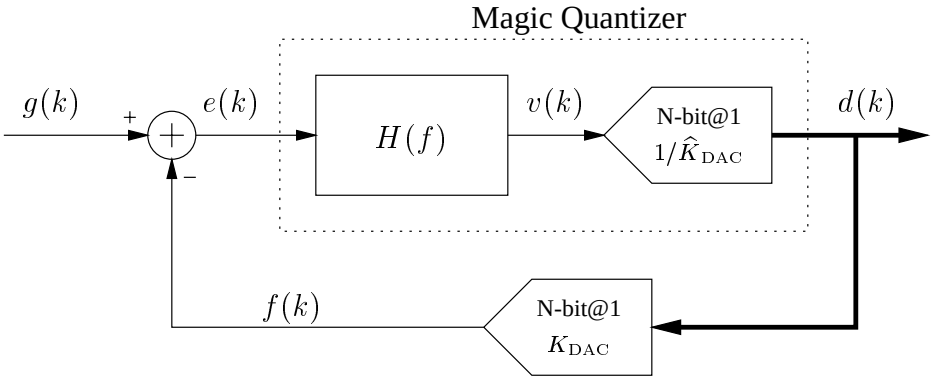


Figure 3.24: Typical implementation of signal quantizers.

In Figure 3.24, the quantizer and the DAC in the feedback loop are (for convenience) assumed to have reciprocal gains. This assumption is useful, because the smart quantizer's properties are not altered if

the filter and quantizer are scaled by reciprocal coefficients; also, the performance measures (Equations (3.49) and (3.51)) are not affected by this assumption. Thus, assuming that the resolving range of the internal quantizer is sufficiently wide, the system's stability can be evaluated using any traditional stability criteria, e.g. Nyquist's Stability Criterion. The traditional design approach is, however, to design

$$\mathrm{NTF}(f) = \frac{1/K_{\mathrm{DAC}}}{1 + H(f)} \tag{3.52}$$

as a high-pass filter[32] with some pre-calculated characteristic, for example a Chebychev Type-II high-pass-filter response. $\mathrm{NTF}(f)$ is called the signal quantizer's noise transfer function, because it determines the output-referred linear signal processing of the internal quantizer's truncation error[33].

Although Figure 3.24 shows the most common implementation for signal quantizers, it is not the optimal way to implement the smart quantizer. Obviously, it is not an optimal design[34] when $g(k)$ must be included in $e(k)$ and filtered before it can appear in the output $d(k)$. An improvement can be obtained simply by directly informing the smart quantizer about $g(k)$, so that it can be taken into account immediately. Figure 3.25 shows a simple signal quantizer implemented according to this principle.

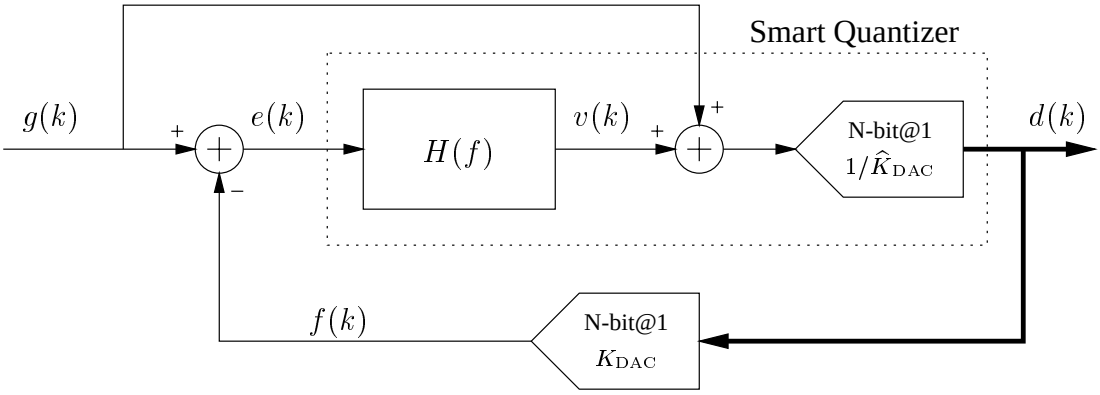


Figure 3.25: Improved implementation of signal quantizers.

Figure 3.26 shows the same signal quantizer as shown in Figure 3.25 redrawn. It is included to shed

[32] More generally, as a filter that suppresses the signal-band.

[33] Which too often is modeled as noise (cf. the discussion on page 38).

[34] Because the object is to minimize $e(k)$.

some light on the fundamental operation; compare Figures 3.17, 3.25, and 3.26.

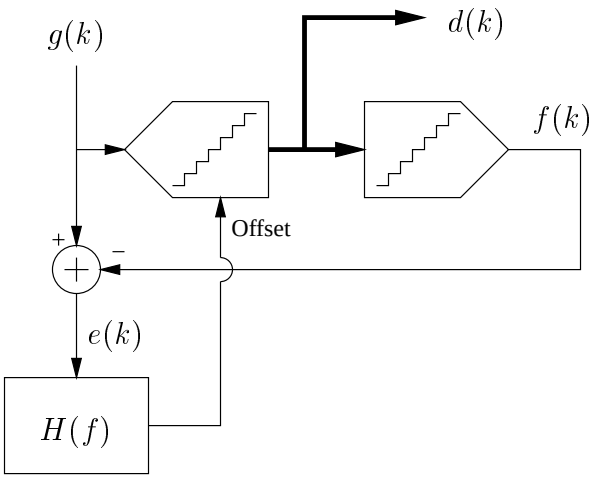


Figure 3.26: Model of the signal quantizer shown in Figure 3.25.

It is worthwhile to notice that the topology shown in Figure 3.25 also assures that the signal transfer function becomes independent of the filter $H(f)$

$$\text{STF}(f) = \frac{\mathcal{F}\{d(k)\}}{\mathcal{F}\{g(k)\}} = \frac{1/\widehat{K}_{\text{DAC}}[1 + H(f)]}{1 + [K_{\text{DAC}}/\widehat{K}_{\text{DAC}}]H(f)} \simeq \frac{1}{K_{\text{DAC}}} \tag{3.53}$$

The advantage of designing the signal quantizer as shown in Figure 3.25, rather than using the traditional topology shown in Figure 3.24, is only significant if $g(k)$ has substantial sample-to-sample variation. That is, however, always the case when the oversampling ratio is low, which is the general assumption in this work.

This concludes the introduction to signal quantizers. They will be discussed in greater detail in Chapter 4.

Chapter 4

State-of-the-Art Signal Quantizers

This chapter will provide an overview of state-of-the-art techniques used for the implementation of signal quantizers. The advantages and shortcomings of the techniques will be discussed and serve as background for the discussion in the subsequent chapters. The reader is assumed to be familiar with Chapters 2 and 3, especially the introduction to signal quantizers provided in Section 3.4.3.

First, it will be explained why single-bit signal quantizers are not suitable for high-resolution wide-bandwidth quantization. Aspects, such as stability and technology constraints, will be considered to reach this conclusion.

Second, signal quantizers implemented in the so-called MASH topology will be discussed. In principle, this technique can improve the performance to an almost unlimited extent, but in reality, circuit imperfections will significantly limit the achievable improvement.

Third, it will be argued that mismatch-shaping DACs facilitate the implementation of linear multi-bit signal quantizers. In this way, the desired high-resolution wide-bandwidth signal quantizers can be implemented. The basic techniques will be described, and some shortcomings will be identified.

Terminology. The remaining part of this work will discuss several topologies for the implementation of data converters[1] that attempt to minimize the signal-band power of nonlinear errors. Figures 3.24 and 3.25 illustrate only the most basic principle for such signal quantizers, and to distinguish these topologies easily from the new topologies to be discussed, it is convenient to have a separate name for these two structures.

For historic reasons, signal quantizers of the type shown in Figures 3.24 and 3.25 will be called delta-sigma ($\Delta\Sigma$) quantizers[2]. If the digital output signal $d(k)$ has a resolution of N bits, the structures are said to be N-bit delta-sigma quantizers. Notice that N refers to the actual resolution, i.e., not the effective resolution, which typically is much higher (cf. page 73).

4.1 Single-Bit Delta-Sigma Quantizers

A single-bit DAC can provide only two output levels. Assuming that these two levels are time-invariant and that dynamic errors are avoided[3], such DACs are inherently linear. This is because any static characteristic $S(n)$ consisting of only two elements will define a straight line (the linear characteristic) with respect to which the DAC has an ideal static performance. As with any other DAC, single-bit DACs are subject to linear errors (offset and gain errors), but such errors are generally tolerable.

General Structure of a Delta-Sigma A/D Converter System. Figure 4.1 illustrates the implementation of a typical delta-sigma A/D converter system. The $\Delta\Sigma$ quantizer employs a single-bit DAC in the feedback path, which ensures that the quantizer is not subject to nonlinear DAC errors (cf. page 72).

In brief, the operation can be described as follows. The analog front end filters and samples the input signal $a(t)$ and provides the discrete-time analog signal $g(k)$. The single-bit $\Delta\Sigma$ quantizer converts $g(k)$

[1] In principle, such data converters should be called signal converters, but this name has not been widely adopted.

[2] Digital circuits implemented in the same topology are called delta-sigma modulators. Digital delta-sigma modulators are used to reduce the resolution of a signal while preserving its signal-band spectral composition.

[3] This is feasible for switched-capacitor implementations, for which good circuit techniques have been developed. Similar techniques for current-mode DACs are under development.

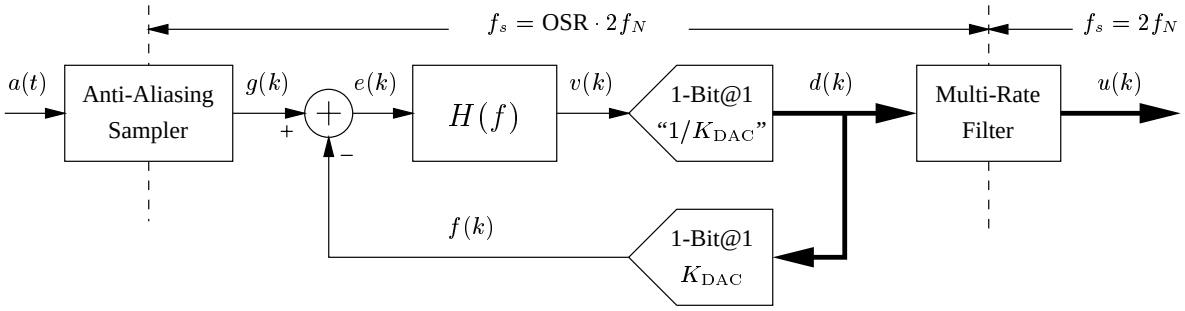


Figure 4.1: A/D converter system based on a single-bit $\Delta\Sigma$ quantizer.

into $d(k)$ in such a way that the signal-band spectral composition of the two signals are linearly related. Outside the signal band, the spectral compositions will differ substantially. A digital filter will reject the out-of-band spectral components of $d(k)$, thereby generating a highly oversampled signal, which is decimated to a signal-band-equivalent signal $u(k)$ sampled at a much lower sampling frequency.

Evaluation of the Topology. The main advantages of the approach shown in Figure 4.1 are that the analog front end is simple to implement, and that the quantization process is inherently linear despite possible analog circuit imperfections. The analog portion of the system is highly oversampled; hence the anti-aliasing filter can be implemented as a simple filter of low order (cf. Equation (3.5)). The $\Delta\Sigma$ quantizer is a simple circuit, the performance of which depends mainly on the linearity of the input-stage integrator used for the implementation of the loop filter $H(f)$ (see [43]).

The need for the somewhat complex digital multi-rate filter is a disadvantage, but as the layout density has increased and the power consumption of digital circuits has been reduced over time, this tradeoff for avoiding complex analog circuitry has become quite acceptable. Hence, single-bit $\Delta\Sigma$ ADCs are suitable for the implementation of many high-resolution applications in modern technologies, which typically are optimized for the design of digital circuits.

4.1.1 Obtaining and Preserving Stability

Useful stability criterions for the closed-loop operation of single-bit $\Delta\Sigma$ quantizers are very hard to derive. This problem is caused by the single-bit loop quantizer (the internal single-bit data quantizer),

which is so nonlinear that its linear characteristic is not even defined (cf. Section 3.1.4).

A single-bit data quantizer is just a polarity detector that assigns one of two integers to $d(k)$. Hence, if the input signal $v(k)$ has a small standard deviation[4] (relative to the difference of the two levels provided by the DAC), the quantizer is, in a sense, described by a large gain (relative to $1/K_{\rm DAC}$); whereas if the standard deviation of $v(k)$ is large, the quantizer's "gain" is low. Despite this highly nonlinear property of single-bit data quantizers, many attempts to analyze single-bit $\Delta\Sigma$ quantizers' stability have been based on the model shown in Figure 4.2, where it is assumed that $q(k)$ is an externally-applied uniformly-distributed white-noise signal, and where the single-bit loop quantizer's gain is assumed to be $1/K_{\rm DAC}$, i.e., independent of $v(k)$.

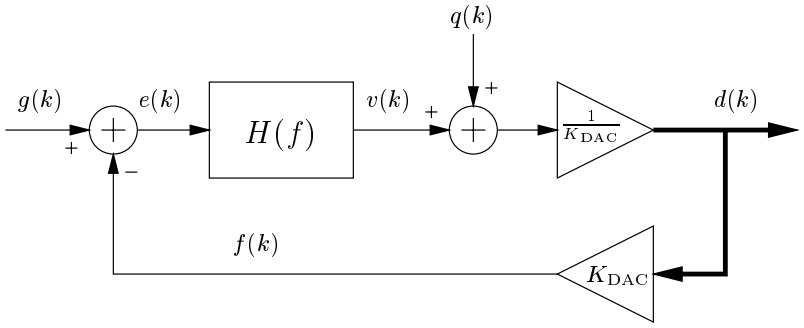


Figure 4.2: Linear model, which often has been used to model single-bit $\Delta\Sigma$ quantizers.

Not surprisingly, the linear model has proven to be so oversimplified that in many situations it is almost useless. An example in which the model leads to the incorrect conclusion is the closed-loop system's stability evaluated using Nyquist's Stability Criterion.

Figure 4.3 shows a correct nonlinear model of the closed-loop system, where $q(k)$ is modeled as a deterministic function T_1 of $v(k)$. Without going into detail, notice that stability should be construed as $v(k)$ having the property of being bounded in magnitude. The stability properties are often analyzed/evaluated by considering a pseudo-probability-density-function (PPDF) description of $v(k)$, i.e., a histogram of $v(k)$ obtained from a finite-duration simulation. "Stability" is indicated if most (say 99.9%) of the PPDF$[v(k)]$ is well within the single-bit loop quantizer's resolving range (cf. Figure 4.3), but it is not an accurate measure that will guarantee stability. The problem is, in part, complicated by

[4]The use of this measure should not be construed to be that $v(k)$ is a stochastic signal; it is not.

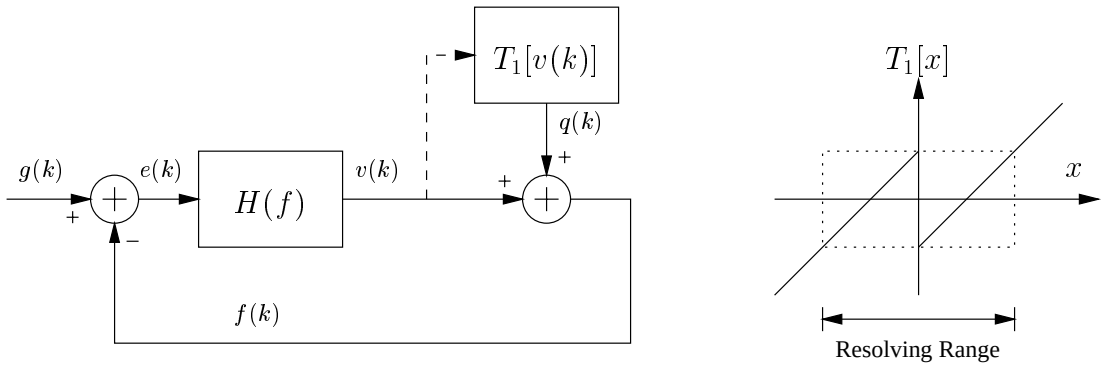


Figure 4.3: Nonlinear system modeling the closed-loop operation of a single-bit $\Delta\Sigma$ quantizer.

the property that the PPDF$[v(k)]$ is affected by the input signal $g(k)$, which implies that stability only can be obtained if the input signal is restricted to a certain range. The main complication is, however, the deterministic correlation of $v(k)$ and $q(k)$; if $v(k)$ attains large[5] values for some reason in a short period of time, the deterministic relationship T_1 implies that $q(k)$ also will attain large values, which may increase $v(k)$ and thereby possibly initiate a positive-feedback sequence that leads the $\Delta\Sigma$ quantizer to instability. The problem is indeed very complex, and it is perhaps best described using chaos theory [44]. Several papers discussing the stability issue have been published; the interested reader is referred to [1] as a starting point.

Design of Stable Single-Bit Delta-Sigma Quantizers. A simple rule has been experimentally found for the design of single-bit $\Delta\Sigma$ quantizers. The rule, which was first stated by Lee [45], claims that a single-bit $\Delta\Sigma$ quantizer will be stable if its noise transfer function (3.52) has a maximum gain of less than $1.5/K_{\mathrm{DAC}}$, i.e., if

$$\max_{f\in R}\left|\frac{1}{1+H(f)}\right| < 1.5 \tag{4.1}$$

Lee's Rule (4.1) cannot be proved; and in fact, some single-bit $\Delta\Sigma$ quantizers designed according to it are quite unstable. However, most single-bit $\Delta\Sigma$ quantizers that fulfill Lee's Rule tend to remain stable for a substantial number of samples (millions), so the rule should be used only with caution.

A sound design approach is to:

[5]Relative to the resolving range.

- design a $\Delta\Sigma$ quantizer according to (4.1),
- simulate the $\Delta\Sigma$ quantizer extensively and re-design it if necessary, and
- assure the stability by making the loop filter nonlinear, for example as described below.

A Toolbox [46] can be used to simplify the first two steps of this design procedure.

Nonlinear Loop Filters. The stability of a single-bit $\Delta\Sigma$ quantizer can be assured by incorporating simple nonlinearities in an otherwise linear loop filter $H(f)$. Linear loop filters can be implemented in several ways, but the following discussion will consider only the fourth-order structure shown in Figure 4.4. This type of loop filter is a good choice for analog $\Delta\Sigma$ quantizers [43], and the reader should find no difficulty in generalizing the discussion to other filter orders or structures.

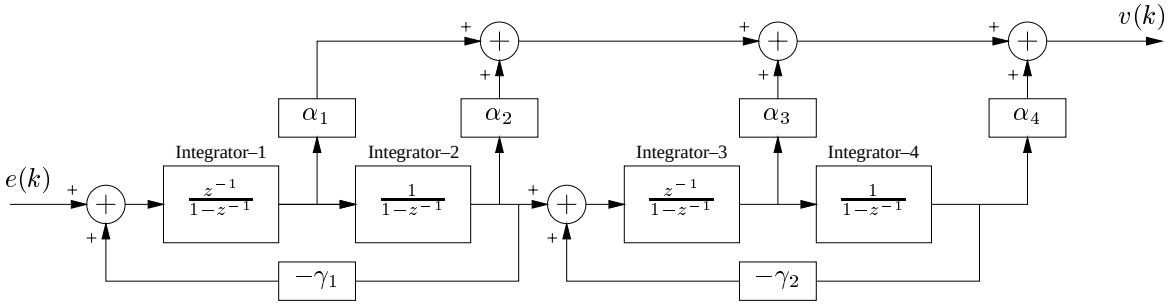


Figure 4.4: Example of a loop-filter structure.

The loop filter's poles, which will become the noise transfer function's (3.52) zeroes, will be located on the z-plane unit circle[6] at angles (frequencies) determined by the coefficients γ_i that are used to tune the two resonators. The feed-forward coefficients α_i are used to control the noise transfer function's poles.

A $\Delta\Sigma$ quantizer, which is based on this loop filter, will be stable only if the output of all integrators remain bounded. By simple inspection of the loop filter's topology, it should be observed that the output of each integrator is a function of $e(k)$ only.

[6]The z-plane refers to the z-transform, i.e., the Fourier transform, for which the variable substitution $z = e^{j2\pi f T_s}$ is used.

Preserving stability is somewhat comparable to backing a truck with as many trailers as the number of integrators less one, where the output of each integrator is modeled as the orientation of each trailer. Obviously, backing the trailer train is a difficult task; hence instability is quite likely to occur. The visualization corresponding to Lee's Rule is that the trailers' lengths must be an increasing function of their distance from the driver. Although this model is somewhat inconsistent mathematically, it may aid the understanding of how and why the following very efficient stabilization technique works.

The stabilization technique is quite simple. If and when instability occurs, i.e., if $v(k)$ becomes a sequence of large values, the last-stage (the fourth) integrator is reset. If this action does not restore stability, then the two last integrators are reset shortly afterwards. This process is continued (more and more integrators are reset) until stability is restored. In practice, it will never be necessary to reset the two first integrators, and the effective noise shaping will always be of at least second order[7].

Achievable Performance. The implication of Lee's Rule may not be obvious, but a study will show that the OSR that is required to obtain a signal-band Signal-to-quantization-Error-Ratio (SER) of (say) 100 dB is only a slowly-decreasing function of the loop filter's order. This is illustrated in Figure 4.14 in [1], where the derivation also can be found. Table 4.1 summarizes the result expressed as the OSR required to obtain 100 dB SER. Clearly, 10 times oversampling is not enough to achieve this level of effective resolution when using a single-bit $\Delta\Sigma$ quantizer with a loop filter of any reasonable order. In fact, at 10 times oversampling, the SER will be around 40 dB for all well-designed single-bit $\Delta\Sigma$ quantizers of high order.

Filter Order	1	2	3	4	5	6	7	8
OSR	$>$1000	220	90	56	45	36	32	30

Table 4.1: OSR required to obtain 100 dB SER for single-bit $\Delta\Sigma$ quantizers.

[7]The term "noise shaping" refers to the filtering performed on $q(k)$ in the linear model, Figure 4.2. The effective order of noise shaping is (in this thesis) defined as the maximum number of signal-band unit-circle poles the linear filter can have while obtaining a bounded output.

4.1.2 Bandwidth Limitation

The signal bandwidth equals the sampling frequency divided by twice the OSR (3.3); for this reason wide-bandwidth $\Delta\Sigma$ quantization requires either a high sampling frequency or a low OSR.

As discussed above, the stability properties of single-bit $\Delta\Sigma$ quantizers restricts the choice of the loop filter such that high-resolution $\Delta\Sigma$ quantization cannot be obtained using these quantizers at less than approximately 30 times oversampling. Furthermore, for high-order $\Delta\Sigma$ quantizers, the digital multi-rate filter must have a narrow transition band, which drastically complicates its implementation. At high speed, the power consumption of the digital circuitry becomes more of an issue, so medium-order $\Delta\Sigma$ quantizers operating at an OSR of about 50 will typically be preferred.

Increasing the sampling frequency implies an increase in the power consumption of both the analog and digital circuitry. Assuming that the $\Delta\Sigma$ quantizer is implemented as a switched-capacitor circuit in a modern technology[8], the sampling frequency is hard to increase beyond 20–50 MHz, even if the power consumption is allowed to be as large as 100–500 mW. The speed/power tradeoff is of course dependent on the technology of choice, but improvements from this source do not seem to keep pace with market requirements. It is unlikely that high-performance switched-capacitor circuits that are implemented in standard technologies will be clocked much faster than (say) 100 MHz in the coming few years.

By combining the estimates made above, it follows that the bandwidth of a high-resolution single-bit $\Delta\Sigma$ ADC is unlikely to exceed 1 MHz within a reasonable time frame. Furthermore, to obtain a competitive advantage, bandwidth improvements should be obtained either by lowering the OSR or by increasing the sampling frequency without increasing the power consumption or the production cost.

Continuous-Time Loop Filter. An interesting approach, which can be used to increase the sampling-frequency/power-consumption ratio in almost any technology, is to implement the $\Delta\Sigma$ quantizers with continuous-time loop filters. These so-called *continuous-time* (CT) $\Delta\Sigma$ quantizers have several advantages (noise, power, and bandwidth), but also substantial drawbacks; they are very sensitive to the

[8] As of 1998.

dynamic errors of the DAC in the feedback path, and particularly they are very sensitive to clock-jitter-induced errors (cf. page 52). Hence the resolution reported thus far has been quite modest. Some exotic continuous-time $\Delta\Sigma$ quantizers have been implemented in highly specialized technologies and operated at sampling frequencies in the GHz range [47]. Such high-speed $\Delta\Sigma$ quantizers will obviously require the use of extremely fast and power-consuming digital circuits. The SNR performance reported was approximately 30 dB less than the theoretical SER, which indicates that clock-jitter and other dynamic errors were the limiting factors.

4.2 MASH Topology

The limiting factor of single-bit $\Delta\Sigma$ quantizers operating at low oversampling ratios is that the loop filter's $H(f)$ minimum signal-band gain cannot be increased beyond a certain level without encountering stability problems; consequently, the error signal's $e(k)$ inband spectral power density cannot be lowered to less than a certain minimum. Now referring to the nonlinear model shown in Figure 4.3, it can easily be shown that $d(k) = f(k)/K_{\mathrm{DAC}}$ can be calculated from

$$\begin{aligned} D(f) &= G(f)\frac{1}{K_{\mathrm{DAC}}} \cdot \frac{H(f)}{1+H(f)} + Q(f)\frac{1}{K_{\mathrm{DAC}}} \cdot \frac{1}{1+H(f)} \\ &= G(f)\mathrm{STF}(f) + Q(f)\mathrm{NTF}(f) \end{aligned} \tag{4.2}$$

where the two terms represent the signal and the error spectra, respectively. Although $Q(f)$ often has an approximately uniform spectral power density in the entire frequency spectrum (which is why it is often modeled as white noise) it should be understood that it is not noise, but merely the single-bit loop quantizer's truncation error. Because $q(k)$ is not noise, it can be estimated, A/D converted, filtered by a digital filter that imitates $\mathrm{NTF}(f)$, and the result subtracted from $d(k)$ to compensate for the error term $Q(f)\mathrm{NTF}(f)$ in (4.2). This so-called MASH[9] quantizer is shown in Figure 4.5.

[9]The name MASH is claimed to be an abbreviation for Multi-Stage Noise Shaping, but it is also claimed that the name was chosen from a popular TV show (for whatever reasons).

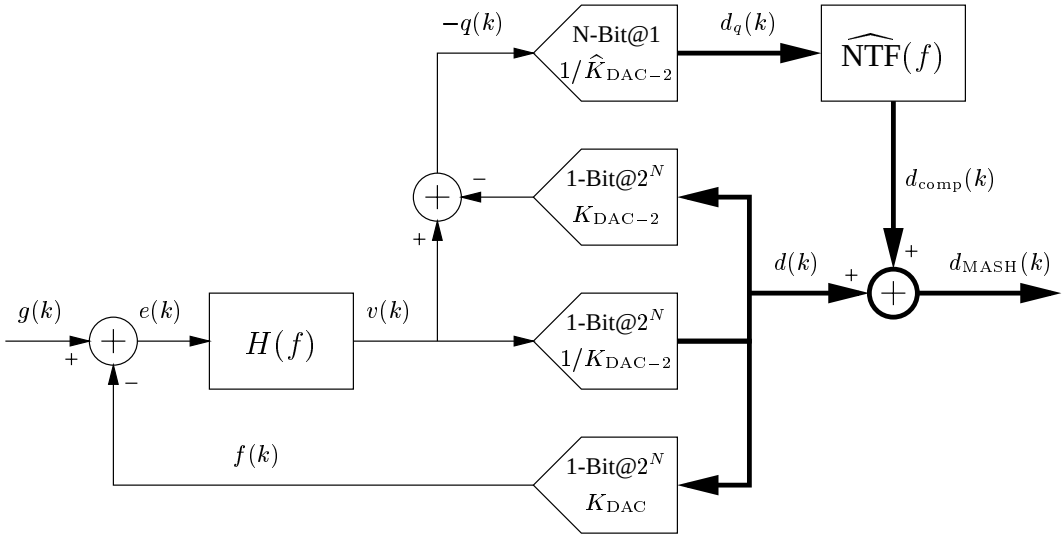


Figure 4.5: MASH-topology signal quantizer.

4.2.1 Analysis of the MASH Topology

The MASH topology includes a traditional residue stage (cf. Figure 3.17) for the calculation of the residue of the single-bit quantization of $v(k)$. As discussed in Section 3.4.2, the gain of a residue stage's quantizer is determined uniquely by the gain $K_{\mathrm{DAC}-2}$ of the residue stage's DAC; consequently, the single-bit loop quantizer will – by definition – have a gain of $1/K_{\mathrm{DAC}-2}$. This definition does not alter the single-bit $\Delta\Sigma$ quantizer's stability properties, but it uniquely determines the noise transfer function NTF(f), by which $q(k)$ is filtered in its appearance in $d(k)$. The noise transfer function is calculated from

$$\mathrm{NTF}(f) = \frac{1/K_{\mathrm{DAC}-2}}{1 + H(f)\frac{K_{\mathrm{DAC}}}{K_{\mathrm{DAC}-2}}} \tag{4.3}$$

Estimation of the Residue. Because $d(k)$ is a single-bit signal, the residue, $-q(k)$, of the single-bit quantization of $v(k)$ can be calculated with very good accuracy (except for an unimportant offset). A multi-bit quantizer[10] A/D converts $-q(k)$, and the result $d_q(k)$ is filtered by a digital filter $\widehat{\mathrm{NTF}}(f)$ and

[10]Originally, this quantizer was a single-bit $\Delta\Sigma$ quantizer, but the generalization to include data quantizers is trivial, and a high-resolution, say pipeline, quantizer is typically a better choice [48].

then added to the single-bit output $d(k)$.

Nonlinearity of the Multi-Bit Quantizer. Nonlinearity of the multi-bit quantizer is only a minor issue. Nonlinearity will cause an error in $d_q(k)$, which contains harmonic distortion of $q(k)$. For this purpose, $q(k)$ can be modeled very well as a white-noise signal, and hence $d_q(k)$ will include a white-noise error with a total power determined by the linearity of the quantizer, i.e., typically 40–60 dB less than the power of $q(k)$. Considering that $\widehat{\mathrm{NTF}}(f)$ will efficiently suppress this error in the signal band, it will be understood that the quantizer's nonlinearity will not be the limiting factor unless more than 40–60 dB improvement (relative to Equation (4.2)) needs to be obtained.

Gain Error of the Multi-Bit Quantizer. Using the same kind of argument as above, it can be understood that a slight inaccuracy of the multi-bit quantizer's gain is acceptable. Assuming that $\widehat{K}_{\mathrm{DAC}-2}$ matches $K_{\mathrm{DAC}-2}$ with (say) -40 dB relative accuracy, it follows that the error term $Q(f)\mathrm{NTF}(f)$ in (4.2) is suppressed by 40 dB, which would be a significant improvement relative to the performance when $d_{\mathrm{comp}}(k) = 0$.

Mismatch of Transfer Functions. Mismatch of the single-bit $\Delta\Sigma$ quantizer's noise transfer function (4.3) and the digital filter $\widehat{\mathrm{NTF}}(f)$ meant to implement it[11] is the only (but highly) critical factor in the MASH topology.

The error term in the digital output $d_{\mathrm{MASH}}(k)$ can be described as

$$D_{\mathrm{error}}(f) = Q(f)\left[\frac{1/K_{\mathrm{DAC}-2}}{1+\frac{K_{\mathrm{DAC}}}{K_{\mathrm{DAC}-2}}H(f)} - \frac{1}{\widehat{K}_{\mathrm{DAC}-2}}\widehat{\mathrm{NTF}}(f)\right] \tag{4.4}$$

Assuming (for the reasons discussed above) that $K_{\mathrm{DAC}-2}$ and $\widehat{K}_{\mathrm{DAC}-2}$ match, the error term can be simplified to

$$D_{\mathrm{error}}(f) \simeq \frac{Q(f)}{\widehat{K}_{\mathrm{DAC}-2}}\left[\frac{1}{1+\frac{K_{\mathrm{DAC}}}{K_{\mathrm{DAC}-2}}H(f)} - \widehat{\mathrm{NTF}}(f)\right] \tag{4.5}$$

[11] Except for the gain factor $1/K_{\mathrm{DAC}-2}$, which is implemented by the multi-bit quantizer

In general, K_{DAC} and $K_{\mathrm{DAC}-2}$ will have the same nominal value, therefore the digital filter $\widehat{\mathrm{NTF}}(f)$ will be designed to have the same transfer function as the nominal value of $1/(1+H(f))$.

Ideally, the difference in Equation (4.5) will be zero. The success of the MASH topology is in general evaluated as the resulting improvement of the SER, which can be estimated as

$$\frac{\int_{-f_b}^{f_b} |\widehat{\mathrm{NTF}}(f)|^2 df}{\int_{-f_b}^{f_b} \left| \frac{1}{1+\frac{K_{\mathrm{DAC}}}{K_{\mathrm{DAC}-2}}H(f)} - \widehat{\mathrm{NTF}}(f) \right|^2 df} \tag{4.6}$$

Considering that the numerator in (4.6) is a very small number, it follows that the difference in (4.5) must be *very* small to obtain a significant improvement.

Minimization of the denominator of (4.6) is, in theory, a matter of matching two rational functions, one having slightly inaccurate coefficients due to the imperfections of the analog circuitry. The imperfections will manifest themselves as a dislocation of the noise transfer function's poles and zeroes, where each dislocation will result in a nonzero term in the denominator of (4.6). Without going into great detail, the generally accepted[12] conclusion is that the achievable improvement becomes less and less as the order of the loop filter is increased; hence the MASH topology is mainly of interest for low-order $\Delta\Sigma$ quantizers.

Stability concerns are actually another reason why the MASH topology is useful for low-order $\Delta\Sigma$ quantizers only. High-order $\Delta\Sigma$ quantizers will generally require that nonlinearities be incorporated in the loop filter (cf. page 84), and it will be very complicated to incorporate the "inverse" nonlinearities in $\widehat{\mathrm{NTF}}(f)$ to cancel the difference in (4.5).

Typical Design Approach. For the reasons discussed above, MASH quantizers are almost always based on a second-order first-stage $\Delta\Sigma$ quantizer. The classical design technique is to choose the nominal value of $H(f)$ as

$$H(f) = \frac{z^{-1}(2-z^{-1})}{(1-z^{-1})^2}, \quad z = e^{j2\pi f T_s} \tag{4.7}$$

[12] According to the author's own survey, this is the general understanding of most experts. Because a much more robust technique has been developed (which is discussed in the third part of this thesis), the author has not directly attempted to optimize the design of MASH quantizers.

in which case the digital compensation filter is a simple second-order FIR filter:

$$\widehat{\mathrm{NTF}}(f) = (1 - z^{-1})^2, \;\; z = e^{j2\pi f T_s} \tag{4.8}$$

This design, however, does not fulfill Lee's Rule (4.1), and hence the single-bit loop quantizer's input signal $v(k)$ will significantly exceed the quantizer's resolving range. Although the $\Delta\Sigma$ quantizer will remain stable[13], this overload operation implies that the multi-bit quantizer must have a resolving range that is significantly (15-20 dB) larger than what normally would be expected. The corresponding drawback is that the system becomes the same 15-20 dB more sensitive to errors caused by nonlinearity of the multi-bit quantizer, which therefore must be designed to have a quite high performance. If the OSR is low, say 10, the digital compensation filter (4.8) will only suppress signal-band white-noise errors by about 40 dB, which (because the single-bit loop quantizer is overloaded) implies that the multi-bit quantizer must have the same linearity and resolution as the overall system less 3-4 bits. A main advantage is, however, that the multi-bit quantizer's nonlinearity will not cause harmonic distortion of the input signal, but instead, a colored pseudo-noise error.

Improved Second-Order MASH Quantizer? Although it has not yet been attempted[14], it seems that a substantial advantage can be obtained simply by scaling $H(f)$ with a factor (say 0.25) or change it otherwise to conform with Lee's Rule, in which case the multi-bit quantizer can be designed to have a smaller resolving range. Because another much improved topology has been discovered (presented in the third part of this thesis), this approach has not been analyzed thoroughly.

Conclusion. MASH quantizers are, in principle, wonderful signal quantizers; but in reality, they suffer greatly from imperfections of the analog circuitry.

Dislocation of the noise transfer function's poles and zeroes require the quantizer to be of low order. If the oversampling ratio is fairly high, a better performance can generally be obtained using a simpler-to-implement higher-order single-bit $\Delta\Sigma$ quantizer. If the oversampling ratio is low, the MASH quantizer

[13] Second-order quantizers will always be stable, but they are not necessarily well-behaved, i.e., although $v(k)$ will be bounded, it can attain large values.

[14] As far as the author is aware.

will become somewhat sensitive to nonlinearity of the multi-bit quantizer, which possibly can be avoided by scaling the loop filter properly.

Some successful implementations of wide-bandwidth MASH quantizers have been reported [48, 49], but the effective resolution is barely 15-16 bits. This resolution is probably close to the limit of what can be obtained, because these quantizers were designed with great care by highly-qualified designers, who employed many circuit tricks to make the analog circuitry behave as ideally as possible.

In an attempt to improve the cancellation of the truncation error, a technique to adaptively adjust the digital compensation filter has been developed [50]. However, the effective resolution obtained thus far does not exceed 12-13 bits at 8 times oversampling.

4.3 Multi-Bit Delta-Sigma Quantizers

To obtain inherent linearity, $\Delta\Sigma$ quantizers have traditionally been restricted to have a single-bit output signal. The low resolution is, however, associated with numerous disadvantages, among which are the quantizer's poor stability and the large magnitude of $q(k)$. Section 4.4 will discuss the so-called mismatch-shaping D/A converters, which can provide an inherently-linear multi-bit D/A conversion with a very good signal-band performance. Mismatch-shaping D/A converters should be considered as a major breakthrough, because they facilitate the use of multi-bit $\Delta\Sigma$ quantizers, which (in the absence of DAC errors) can be designed to have a much better performance than their single-bit counterparts. The following discussion of multi-bit $\Delta\Sigma$ quantizers refers to Figure 3.25.

Brute-Force Improvement. Because the magnitude of the multi-bit loop quantizer's truncation error is inversely proportional to the resolution of $d(k)$, every doubling of the resolution will result in 6 dB improvement of the SER. Hence, assuming that the loop filter is designed in the same way as for a single-bit $\Delta\Sigma$ quantizer (cf. page 83), $d(k)$ will need to be of 10-bit resolution to obtain 100 dB SER at 10 times oversampling[15]. Fortunately, significantly better results can be obtained if the loop filter is re-designed.

[15] As discussed in Section 4.1, single-bit $\Delta\Sigma$ quantizers of any high order will provide a SER of only about 40 dB.

4.3.1 Stability Properties

A main advantage of multi-bit $\Delta\Sigma$ quantizers is that their stability properties are much better than for their single-bit counterparts, and hence their design need not be restricted as much as proposed by Lee's Rule (4.1).

In essence, $\text{NTF}_{\max} = \max_f |1/(1+H(f))|$ is the main factor that controls the rms value[16] of $v(k)$, and Lee's Rule is merely an empirical estimate of when $v(k)$ is expected to be small enough not to overload a single-bit data quantizer. However, for a multi-bit $\Delta\Sigma$ quantizer, where it will be assumed that the input signal $g(k)$ is somewhat smaller (say by ± 5 LSB) than the loop quantizer's resolving range, $v(k)$ may have significant fluctuation (± 5 LSB) without causing overload of the loop quantizer, and hence the loop filter can be designed to have a larger $\text{NTF}_{\max}$ than imposed by Lee's Rule.

An Example. Figure 4.6 shows the sinusoid input signal $g(k)$ and the staircase output signal $d(k)$ for two quite different designs of a 6th-order 17-level $\Delta\Sigma$ quantizer operating at eight times oversampling[17]. The upper plot corresponds to an aggressively-designed quantizer, for which $\text{NTF}_{\max} = 8$, whereas the lower plot corresponds to a more conservatively-designed quantizer, for which $\text{NTF}_{\max} = 2$.

Clearly, $d(k)$ is more "busy" for the aggressive design than it is for the conservative design. This property is in good agreement with the prediction made above. Although the total power of the error signal $e(k)$ is larger for the aggressive $\Delta\Sigma$ quantizer, it is actually 41 dB less is the signal band (assuming eight times oversampling) than for the conservative $\Delta\Sigma$ quantizer. However, Figure 4.6 indicates that the aggressive $\Delta\Sigma$ quantizer's input signal should be considered to be full-scale, whereas the conservative $\Delta\Sigma$ quantizer supposedly will remain stable for inputs that are as much as 5 dB larger[18]. Hence, "only" 36 dB improvement is obtained by using the aggressive loop filter.

[16] More precisely, the standard deviation of the PPDF$[v(k)]$ (cf. Section 4.1.1).

[17] OSR $= 8$ is used because the Figures are based on simulations that constitute a partial reproduction of work performed by Richard Schreier [2].

[18] The quantizer becomes unstable when the loop quantizer's resolving range (± 8.5 LSB) is exceeded.

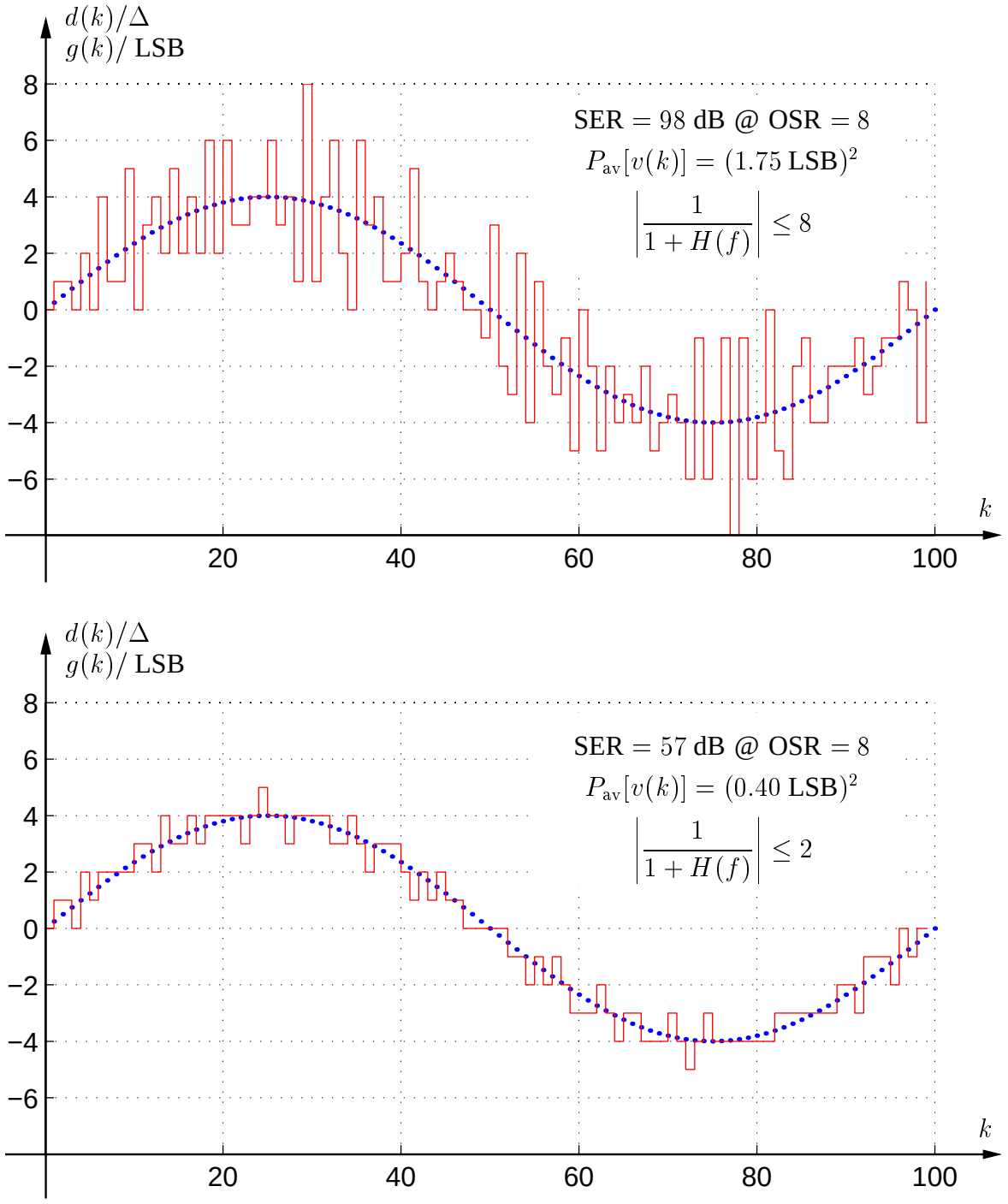


Figure 4.6: Simulation result for two 6th-order multi-bit $\Delta\Sigma$ quantizers. Top: an aggressive design. Bottom: a more conservative design.

Achievable Performance. The advantages that can be obtained by increasing the value of $\mathrm{NTF}_{\max}$ are discussed in [2], in which Figure 5 illustrates the SER performance that can be obtained at eight times oversampling using quantizers of various orders and for various values of $\mathrm{NTF}_{\max}$.

The result is summarized here in Table 4.2. $\mathrm{SER}_{1\,\mathrm{LSB}}$ is the SER obtained when the input is a sinusoid with a peak-to-peak magnitude of 1 LSB. The minimum resolution of $d(k)$ must be approximately $6\sqrt{P_{\mathrm{av}}[v(k)]}/$ LSB to assure stability. In this limit, the considered 1- LSB input signal $g(k)$ is full-scale, and the loop quantizer's resolving range is used mainly for fluctuations in $v(k)$. If $d(k)$ is of higher resolution, the wider resolving range can be utilized to increase the input signal's magnitude, which will result in a 6 dB improvement for every factor of two increase in the magnitude of $g(k)$ (cf. Equation (4.9)).

Quantizer Order (OSR $= 8$)	2	3	4	5	6	7	8
$\mathrm{SER}_{1\mathrm{LSB}}$/dB for $\sqrt{P_{\mathrm{av}}[v(k)]} < 1$ LSB	38	49	56	62	65	68	70
$\mathrm{SER}_{1\mathrm{LSB}}$/dB for $\sqrt{P_{\mathrm{av}}[v(k)]} < 1.5$ LSB	38	52	62	70	76	81	85
$\mathrm{SER}_{1\mathrm{LSB}}$/dB for $\sqrt{P_{\mathrm{av}}[v(k)]} < 2$ LSB	38	52	65	74	82	88	94
$\mathrm{SER}_{1\mathrm{LSB}}$/dB for $\sqrt{P_{\mathrm{av}}[v(k)]} < 3$ LSB	38	52	65	78	87	95	101
$\mathrm{SER}_{1\mathrm{LSB}}$/dB for $\sqrt{P_{\mathrm{av}}[v(k)]} < 5$ LSB	38	52	65	78	91	100	108

$$\mathrm{SER} \simeq \mathrm{SER}_{1\,\mathrm{LSB}} + 6\ \mathrm{dB}\log_2\left[N - \left(\frac{6\sqrt{P_{\mathrm{av}}[v(k)]}}{\mathrm{LSB}}\right)\right] \tag{4.9}$$

Table 4.2: The SER performance that can be obtained from a N-level $\Delta\Sigma$ quantizer operating at eight-times oversampling.

As will be discussed in the Section 4.4, mismatch-shaping D/A converters will typically be of fairly low resolution, say 4 to 5 bits or even less.

Consider a design where the resolution of $d(k)$ is 5 bits and the loop filter is designed such that

$$\sqrt{P_{\mathrm{av}}[v(k)]} = 5\ \mathrm{LSB}$$

Although the $\Delta\Sigma$ quantizer's resolving range is quite small (a few LSBs), the maximum SER can still be made as high as 108–120 dB (using a high-order loop filter). However, a characteristic of this design

is that the $\Delta\Sigma$ quantizer's feedback signal $f(k)$ is many times larger than the input signal $g(k)$, therefore, the system's performance is very sensitive to the feedback DAC's nonlinearity and noise performance. In general, this should not be considered a good design; usually it is preferable to have a (relatively) larger resolving range for $g(k)$, even if a lower SER is an unavoidable tradeoff.

Conclusion. A fluctuation in $v(k)$ of about $\pm 3\sqrt{P_{\rm av}[v(k)]}$ is expected, and the loop quantizer's resolving range must be at least this wide to preserve stability. An increment in the resolution of $d(k)$ beyond this minimum will affect (improve) only the $\Delta\Sigma$ quantizer's resolving range[19], and (as for any other quantizer) the maximum SER will improve 6 dB for every doubling of the system's resolving range.

The SER is not the only design aspect that needs to be considered, and the circuit designer should, in general, choose an entry in Table 4.2, for which the feedback DAC's resolution N is at least twice as large as $6\sqrt{P_{\rm av}[v(k)]}$. For example, if the feedback DAC's resolution is (say) $N = 16$, then the loop filter should be designed according to the second entry row in Table 4.2. The $\Delta\Sigma$ quantizer's resolving range will be approximately $(16 - 9)$ LSBs $= 7$ LSBs, therefore, the second term in Equation (4.9) will be approximately 17 dB. Accordingly, to obtain a 100 dB SER, the loop filter must be of at least the 8th order.

Because the resolution N in general will be low, the second factor in Equation (4.9) will be less than 30 dB, and hence higher-order[20] loop filters will be required to obtain 100 dB SER performance.

For continuous-time implementations, the feedback DAC is usually a current-mode DAC, and clock jitter is another (important) reason not to use a loop filter for which $6\sqrt{P_{\rm av}[v(k)]/\ \rm LSB}$ is too close to N (cf. Section 3.2.3).

[19] The $\Delta\Sigma$ quantizer's resolving range can be estimated as $N \cdot$ LSB $- 6\sqrt{P_{\rm av}[v(k)]}$ (cf. Equation (4.9)).

[20] Fifth order will be the minimum, and 6th- and 7th-order loop filters will be typical.

4.4 Mismatch-Shaping DACs

As described in the previous section, many advantages can be obtained if an inherently-linear more-than-two-level DAC is available. Considering that the multi-bit DAC will be used as the feedback element in a multi-bit $\Delta\Sigma$ quantizer, it should be understood that it need not be ideal in the entire frequency spectrum; the $\Delta\Sigma$ quantizer's performance is fairly insensitive to errors outside the signal band, and hence (errors cannot generally be avoided) the DAC should preferably optimize only the signal-band performance. DACs that have this characteristic will in the following be referred to as *mismatch-shaping* DACs.

Mismatch-shaping DACs assume that their output will be evaluated as a *signal*[21], and they attempt to keep the signal-band part of the generated signal free of errors by allowing mismatch-induced errors to occur at other frequencies. Accordingly, mismatch-shaping DACs and signal quantizers alike are based on the same fundamental philosophy, namely, that the error signal is applied to a filter that has high gain in the signal band, and that the system is controlled such that the filter's output remains bounded in magnitude.

A main difference between signal quantizers and mismatch-shaping DACs is in the way they calculate the error signal. Signal quantizers make use of a supposedly-ideal[22] DAC to calculate the error signal (cf. Figure 3.21). Mismatch-shaping DACs, on the other hand, cannot use the same technique because that will require the existence of an ideal reference DAC, as shown in Figure 4.7. In the absence of an ideal reference DAC, it will be impossible to calculate and filter the mismatch-shaping DAC's error signal $m(k)$ in the analog domain, but nevertheless (as it will be explained in the following) it is actually possible to use digital techniques to control certain properties of the error signal $m(k)$.

[21] Meaning that the output is evaluated by its spectral composition, and not sample by sample.

[22] The performance is evaluated with respect to the signal band only.

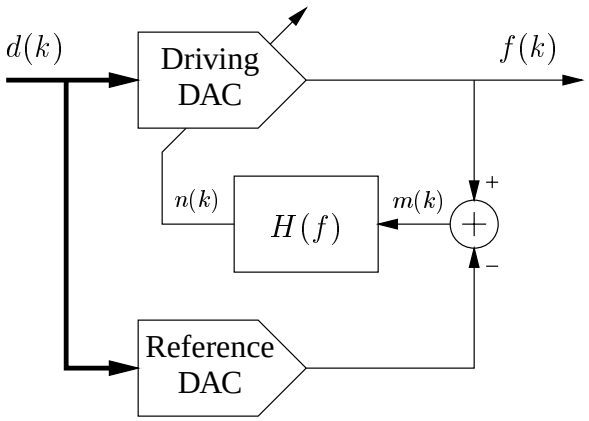


Figure 4.7: Conceptual mismatch-shaping DAC.

4.4.1 Estimation of the Error Signal

In the following, it will be assumed that the DAC is implemented in the topology shown in Figure 4.8, where a digital front end separates $d(k)$ into a set of signals $b_i(k)$ such that

$$d(k) = \sum_{i=0}^{P-1} b_i(k) \tag{4.10}$$

The composite DAC's linear-characteristic gain K_d and offset f_{offset} are defined as shown in Figure 3.15. When calculated with respect to this linear characteristic, the DAC's nonlinearity $m(k) = \text{INL}_\text{d}[d(k)]$ can be expressed with respect to the characteristics of the individual DACs as follows:

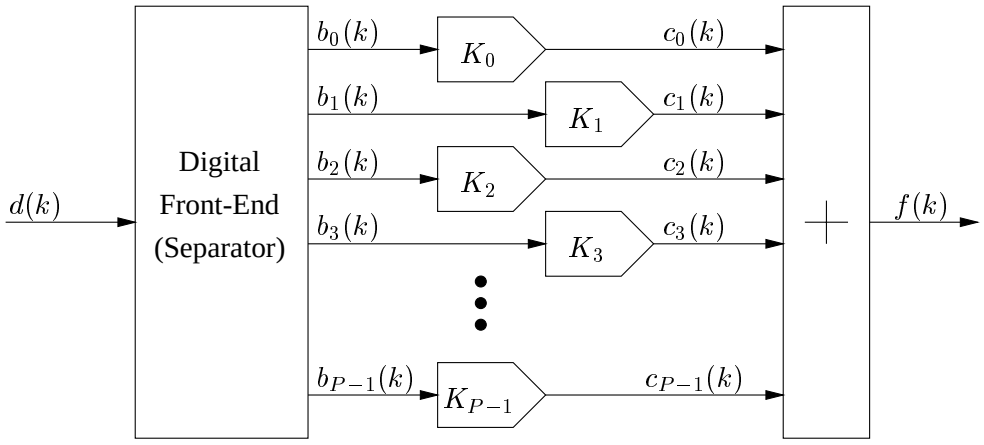


Figure 4.8: Fundamental topology of a typical mismatch-shaping DAC.

$$
\begin{aligned}
m(k) &= f(k) - [K_d d(k) + f_{\text{offset}}] \\
&= \sum_{i=0}^{P-1} [K_i b_i(k) + c_{i,\text{offset}} + \text{INL}_i[b_i(k)]] - [K_d d(k) + f_{\text{offset}}] \\
&= \sum_{i=0}^{P-1} [K_d b_i(k) + [K_i - K_d] b_i(k) + c_{i,\text{offset}} + \text{INL}_i[b_i(k)]] - [K_d d(k) + f_{\text{offset}}] \\
&= \underbrace{\left[\sum_{i=0}^{P-1} [K_d b_i(k)] - K_d d(k)\right]}_{\text{These terms cancel } (=0)} + \underbrace{\left[\sum_{i=0}^{P-1} [c_{i,\text{offset}}] - f_{\text{offset}}\right]}_{\text{These terms cancel } (=0)} \\
&\quad + \sum_{i=0}^{P-1} [[K_i - K_d] b_i(k) + \text{INL}_i[b_i(k)]] \\
&= \sum_{i=0}^{P-1} [b_i(k)[K_i - K_d]] + \sum_{i=0}^{P-1} \text{INL}_i[b_i(k)]
\end{aligned}
\tag{4.11}
$$

Accordingly, the DAC's nonlinearity $\text{INL}_\text{d}[d(k)]$ can be considered to consist of two terms:

- the error term $\sum_{i=0}^{P-1} [b_i(k) [K_i - K_d]]$ will be called the *gain mismatch errors*, because each term in the sum reflects an error that is caused by mismatch of the respective DAC's gain relative to the composite DAC's gain.

- the error term $\sum_{i=0}^{P-1} \mathrm{INL}_i[b_i(k)]$ will be called the *local nonlinearity errors*, because each term in the sum reflects an error caused by the respective DAC's nonlinearity with respect to its own (local) linear characteristic.

Notice that if the individual DACs consist of only one element, the local nonlinearity errors become zero, and the composite DAC's nonlinearity $\mathrm{INL}_d[d(k)]$ will consist of the gain mismatch errors only. Hence, Equation (4.11) agrees with the previously derived expressions (3.37) and (3.40) for the nonlinearity $\mathrm{INL}_d[d(k)]$ of binary-weighted DACs and unit-element DACs, respectively.

Unknown Parameters. Equation (4.11) is in reality not an estimation of the error signal $m(k)$, but merely a description based on several unknown parameters and functions:

$$K_0, K_1, \ldots, K_{P-1} \quad \text{and} \quad \mathrm{INL}_0, \mathrm{INL}_1, \ldots, \mathrm{INL}_{P-1} \tag{4.12}$$

Obviously, one could attempt to measure these parameters as accurately as possible, and then estimate $m(k)$ on the basis of (4.11). That would, however, be a very tedious approach. It would also be a somewhat silly approach, because one could instead employ digital correction [1] [3] [4], which would be much simpler to implement and also provide better results.

Mismatch-shaping DACs avoid the complexity of estimating analog parameters using digital techniques (that are not based on knowledge of the parameter's (4.12) value) for the minimization[23] of the signal-band power of $m(k)$. Hence, mismatch-shaping DACs have an advantage in that they are very robust with respect to low-frequency variations[24] of the parameters (4.12).

4.4.2 First-Order Unit-Element Mismatch-Shaping DACs

The so-called unit-element mismatch-shaping (UE-MS) DACs [2, 5–18] are based on an array of nominally-identical analog sources, which are controlled by each one of the single-bit signals $b_i(k)$ (for simplicity, the following will assume that the signals $b_i(k)$ attain only the two values 0 and 1).

[23] The use of this term should not be construed to be that the mismatch-shaping DAC actually finds a minimum, but merely that it tries to reduce the signal-band power of $m(k)$.

[24] For example, caused by aging of the circuit or by changes in the ambient temperature, humidity, etc..

The linear-characteristic gain K_d was calculated in Equation (3.39), and according to (4.11), the nonlinearity $m(k) = \text{INL}_d[d(k)]$ can be described as

$$\begin{aligned} m(k) &= \sum_{i=0}^{P-1} [b_i(k)(K_i - K_d)] + \sum_{i=0}^{P-1} \text{INL}_i[b_i(k)] \\ &= \sum_{i=0}^{P-1} [b_i(k)(K_i - K_d)] \end{aligned} \tag{4.13}$$

The main characteristic of all first-order-shaping systems – whether it is a $\Delta\Sigma$ quantizer/modulator or a mismatch-shaping DAC – is that the integral (sum) of the error signal remains bounded in magnitude. To evaluate this property for unit-element DACs, the integrated error signal $n(k)$ can be calculated as

$$\begin{aligned} n(k) &= \sum_{j=0}^{k} m(j) \\ &= \sum_{j=0}^{k} \left[\sum_{i=0}^{P-1} [b_i(j)(K_i - K_d)] \right] \\ &= \sum_{i=0}^{P-1} \left[\sum_{j=0}^{k} [b_i(j)(K_i - K_d)] \right] \\ &= \sum_{i=0}^{P-1} \left[(K_i - K_d) \sum_{j=0}^{k} [b_i(j)] \right] \end{aligned} \tag{4.14}$$

Because the composite DAC's gain K_d is the average value of the individual DACs' gains' K_i (cf. Equation (3.39)), it follows that

$$\sum_{i=0}^{P-1} (K_i - K_d) = 0 \tag{4.15}$$

This fundamental property (4.15), which is valid for all unit-element DACs, can be used to bring Equation (4.14) to the form

$$n(k) = \sum_{i=0}^{P-1} \left[(K_i - K_d) \left[\sum_{j=0}^{k} [b_i(j)] - q \right] \right] \quad \text{for all real } q \tag{4.16}$$

By defining the set $\mathcal{U} = \{0, 1, 2, \ldots, (P-1)\}$ and applying Schwartz's inequality to (4.16), it follows

that

$$\begin{aligned} |n(k)| &\leq \sum_{i=0}^{P-1} \left[|K_i - K_d| \left| \sum_{j=0}^{k} [b_i(j)] - q \right| \right] \\ &\leq \max_{i \in \mathcal{U}} \{|K_i - K_d|\} \sum_{i=0}^{P-1} \left| \sum_{j=0}^{k} [b_i(j)] - q \right| \\ &\leq \max_{i \in \mathcal{U}} \{|K_i - K_d|\} \cdot P \cdot \max_{i \in \mathcal{U}} \left\{ \left| \sum_{j=0}^{k} [b_i(j)] - q \right| \right\} \quad \text{for all real } q \end{aligned} \tag{4.17}$$

The term $P \cdot \max_{i \in \mathcal{U}} \{|K_i - K_d|\}$ is a constant for each implementation[25], therefore, the unit-element DAC will perform first-order mismatch-shaping, i.e., $|n(k)|$ will be bounded in magnitude, if $s(k)$ is bounded

$$s(k) = \min_{q \in R} \left\{ \max_{i \in \mathcal{U}} \left\{ \left| \sum_{j=0}^{k} [b_i(j)] - q \right| \right\} \right\} \tag{4.18}$$

Considering that $\sum_{j=0}^{k} [b_i(j)]$ represents the number of times that the i'th DAC has been used to convert $b_i = 1$, it follows that $s(k)$ simply represents the difference in usage

$$s(k) = \frac{\max_{i \in \mathcal{U}} \left\{ \sum_{j=0}^{k} [b_i(j)] \right\} - \min_{i \in \mathcal{U}} \left\{ \sum_{j=0}^{k} [b_i(j)] \right\}}{2} \tag{4.19}$$

Hence, the simple condition required for baseband first-order mismatch shaping is that all of the unit-element DACs must be employed equally often (on average).

Simple Explanations of How and Why First-Order Mismatch-Shaping Works. The basic concept on which all baseband first-order mismatch-shaping DACs are based, can be expressed as follows:

> Baseband mismatch-shaping DAC's will emphasize the performance at low frequencies (in particular the average value of the output). When all elements are used the same number of times, their corresponding errors will have canceled, and hence the average value will be correct.

[25] More precisely, it is a stochastic variable, which will attain a constant value for each independent implementation.

Alternatively, the operation can be expressed as:

> Let a unit-element DAC consist of the set $\mathcal{A}$ of unit elements. When a sample is D/A converted using the set $\mathcal{B}_0 \subseteq \mathcal{A}$ of unit elements, an error δ will occur. To assure a good low-frequency performance, this error must be corrected as soon as possible by an error $-\delta$. The error δ is unknown, but (as expressed by Equation (4.15)) the error $-\delta$ can be committed by using the set $\overline{\mathcal{B}_0} = \mathcal{A} \setminus \mathcal{B}_0$ of unit elements. Hence, the set $\overline{\mathcal{B}_0}$ of unused elements should be used before any of the already-used elements $\mathcal{B}_0$ are used again; when all elements $\mathcal{A}$ are used, the next (of the remaining part of a) sample can be D/A converted using any new set $\mathcal{B}_1 \subseteq \mathcal{A}$ of unit elements, and the procedure is repeated.

Element-Rotation Scheme. The element-rotation scheme (ERS) is possibly the simplest algorithm for the implementation of baseband first-order UE-MS mismatch-shaping DACs.

As illustrated in Figure 4.9, the unit elements are arranged in a fixed order and used sequentially. For example:

> If the first sample is of value 4, then the first four unit elements $b_0 \sim b_3$ are turned on. If the next sample is of value 3, then the *next* three elements $b_4 \sim b_6$ are turned on. This procedure continues until all the elements are used once, at which time the pointer "wraps around" as shown in Figure 4.9, where the third sample, of value 5, causes $b_7 \sim b_8$ and $b_0 \sim b_2$ to be turned on.

Figure 4.10 shows the implementation of a digital front end that provides the ERS switching function illustrated in Figure 4.9. The digital input $d(k)$ is assumed to be binary coded and attain only integer values in the range from 0 to 8. A rotation pointer (cf. Figure 4.9) is generated by accumulating $d(k)$ modulo 8. The input $d(k)$ is binary-to-thermometer encoded (cf. Equation (3.41)), and the thermometer-coded representation of $d(k)$ is rotated stepwise (in steps of 4,2,1) according to the rotation pointer, which will always attain an integer value in the range from 0 to 7.

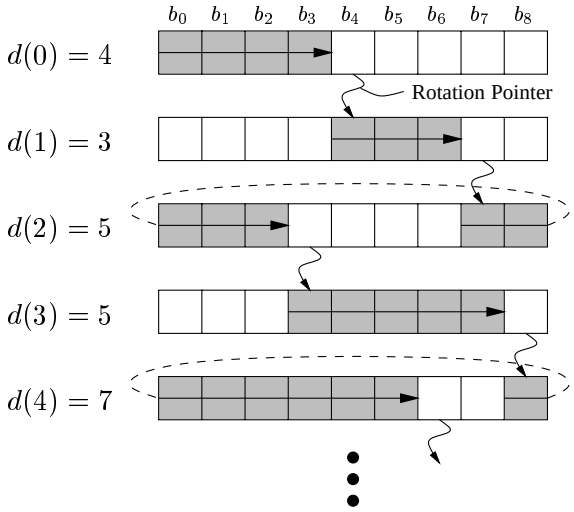


Figure 4.9: The element-rotation scheme, which is used for the implementation of baseband first-order unit-element mismatch-shaping DACs.

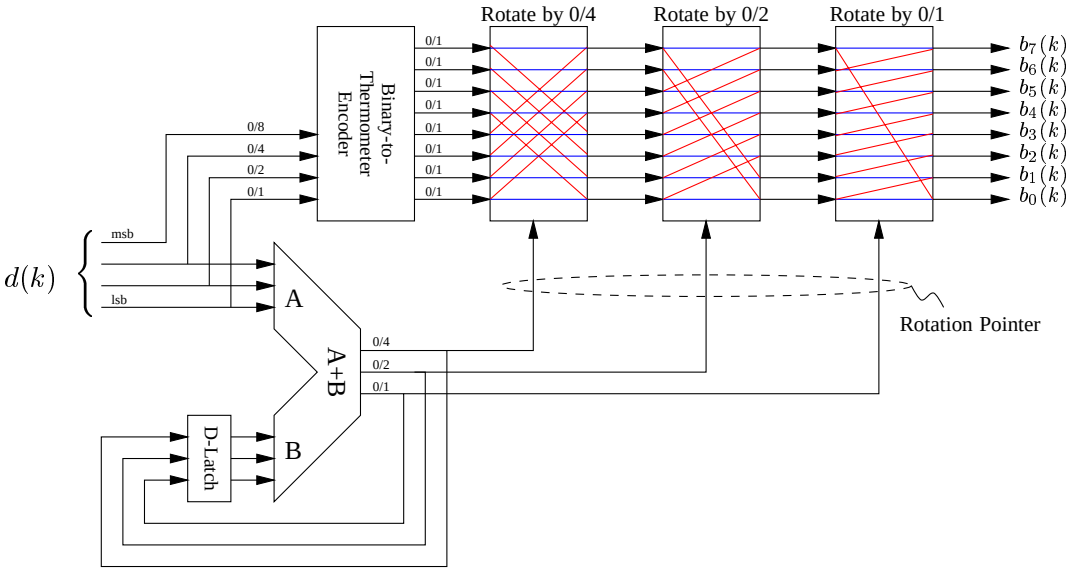


Figure 4.10: An implementation of the ERS algorithm.

Other Implementations. The term Data-Weighted-Averaging (DWA) DACs will refer to all baseband first-order mismatch-shaping D/A converters, which can be implemented in infinite variations. The differences lie typically in the complexity of the digital front end, in how small they manage to keep $s(k)$ in (4.19), and in the spectral composition of $n(k)$. Descriptions of several implementations can be found in [2, 5–18].

The ERS algorithm was discussed in this context because it was the first known implementation [5], and because it is fairly simple to implement.

4.4.3 Performance of First-Order Mismatch-Shaping DACs

The term *baseband first-order mismatch-shaping* comes from the property that $n(k) = \sum_{j=0}^{k} m(j)$ will be of finite power, and hence that the spectral properties of the error signal $m(k)$ can be calculated from

$$M(f) = N(f) \cdot (1 - z^{-1}), \quad \text{where} \quad z = e^{2\pi f T_s} \tag{4.20}$$

Equation (4.20) implies that $M(0) = 0$, but otherwise it is not straightforward to estimate $M(f)$ accurately. Often $N(f)$ is assumed to be white noise, and although simulations provide some justification for this assumption, it should be understood that this model is typically oversimplified and that it may lead to very inaccurate estimates of the system's performance.

Idle Tones. Idle tones are often encountered in single-bit $\Delta\Sigma$ quantizers when the truncation error $q(k)$ (cf. Figure 4.2) attains pseudo-periodic patterns, in which case $q(k)$ is poorly modeled as white noise. The consequence is that the $\Delta\Sigma$ quantizer's actual SER may be significantly less than the SER predicted on the basis of the white-noise assumption for $q(k)$. Even worse, idle tones may imply that a $\Delta\Sigma$ quantizer is less suitable for critical applications such as audio (cf. page 74). The idle-tone problem is known to be quite troublesome for low-order $\Delta\Sigma$ quantizers, and in particular for first-order quantizers. Dither, a noise signal that is deliberately added to $v(k)$ (cf. Figure 4.2), is known to be an efficient means to prevent[26] idle tones (cf. Chapter 3 in [1]).

[26] Unfortunately, the dither signal must be so large that it tends to cause instability in single-bit $\Delta\Sigma$ quantizers, so (in reality) it is very hard to efficiently avoid idle tones in single-bit $\Delta\Sigma$ quantizers.

Idle tones are also a serious problem in many DWA DACs [51]. Consider for example, the ERS DWA DAC illustrated in Figure 4.9. If the input is constant (say $d(k) = 3$), the output will be generated by the periodic use of three groups of three unit elements each, and the error signal $m(k)$ will be of the form

$$m(k) = \ldots, \delta_1, \delta_2, \delta_3, \delta_1, \delta_2, \delta_3, \delta_1, \delta_2, \delta_3, \ldots \tag{4.21}$$

where $\delta_1 + \delta_2 + \delta_3 = 0$. Hence, the error signal $m(k)$ will be a tone at the frequency $f_s/3$. If the tone is not located in the signal-band, then the behavior may be acceptable; but if a tone *is* in the signal-band, the behavior is highly unacceptable because the DAC functions partly as an in-band oscillator.

For idle tones to become a problem for baseband DACs, the pseudo-periodic patterns in $m(k)$ must have period(s) which are at least $2 \cdot \text{OSR}$ long, and that is actually quite likely to occur [51].

A DWA DAC's digital front end can be designed (more or less successfully) to avoid idle tones. The so-called butterfly scrambler [9] is, for example, claimed to be less tonal than the simpler ERS front ends, partly because it uses more combinations to generate each nominal value. The general idea is to assure that the signals $b_i(k)$ are "complicated" functions of the input signal $d(k)$. An example of how to implement "intentional complexity" was presented by Williams [39], who designed a DWA front end comprising three ERS encoders for the control of the same array of 9 unit elements. For each input sample, only one of the three ERS encoders is activated, i.e., clocked and used to control the unit elements, depending on which of the three controlling sets $\mathcal{B}_0 = \{0, 3, 6\}$, $\mathcal{B}_1 = \{1, 4, 7\}$, $\mathcal{B}_2 = \{2, 5, 8\}$ the input sample $d(k)$ is a member of. Williams claims that this technique efficiently prevents idle tones, but this author claims that some tones are still likely[27] to occur.

Some improved techniques to avoid idle tones are described in the second part of this thesis.

Estimation of the Error Signal's Signal-Band Power. Assuming that the digital front end efficiently avoids idle tones, i.e., that it is reasonable to model $N(f)$ in (4.20) as white noise, it is fairly simple to estimate the signal-band performance of DWA DACs.

[27] The previous example, where $d(k) = 3$ will result in the same performance; signals where $d(k)$ (for example) has the period $\{1, 4, 7, 7, 4, 1\}$ will generate tones at $f_s/6$; etc.

Assume that the ERS algorithm[28] is used for a DWA DAC with P unit elements, and that the the rotation pointer is equally likely to attain any of the P values: $\{0, 1, 2, \ldots, (P-1)\}$. Assume further that each of the unit elements $X_i, i \in \{1, 2, \ldots, P\}$ represents a stochastic event of the same (Gaussian) $\mathcal{N}(1, \sigma^2)$ process (normalized with respect to 1 LSB).

When the rotation pointer attains the value q, the standard deviation $\sigma_n(q)$ of $n(k) = \sum_{j=0}^{k} m(k) = \sum_{i=1}^{q}[X_i] - \frac{q}{P}\sum_{i=1}^{P}[X_i]$ can be calculated from

$$\sigma_n(q) = \sigma \text{ LSB}\sqrt{\frac{q(P-q)}{P}} \tag{4.22}$$

implying that the average power of $n(k)$ can be estimated roughly (for large P) as[29]

$$P_{\text{av}}[n(k)] = \frac{\sigma^2 \text{ LSB}^2}{P}\sum_{q=0}^{P-1}\frac{q(P-q)}{P} \simeq \frac{P\sigma^2 \text{ LSB}^2}{6} \tag{4.23}$$

Because $m(k)$ is the first-order difference of $n(k)$, the signal-band power of $m(k)$ can be estimated as

$$\begin{aligned} P_{\text{av,inband}}[m(k)] &= \frac{P\sigma^2}{6}\left[\frac{1}{\pi}\int_0^{\pi/\text{OSR}}(1-e^{-j\omega})^2 d\omega\right] \\ &= \frac{P\sigma^2 \text{ LSB}^2}{6} S_h(\text{OSR}, 1) \end{aligned} \tag{4.24}$$

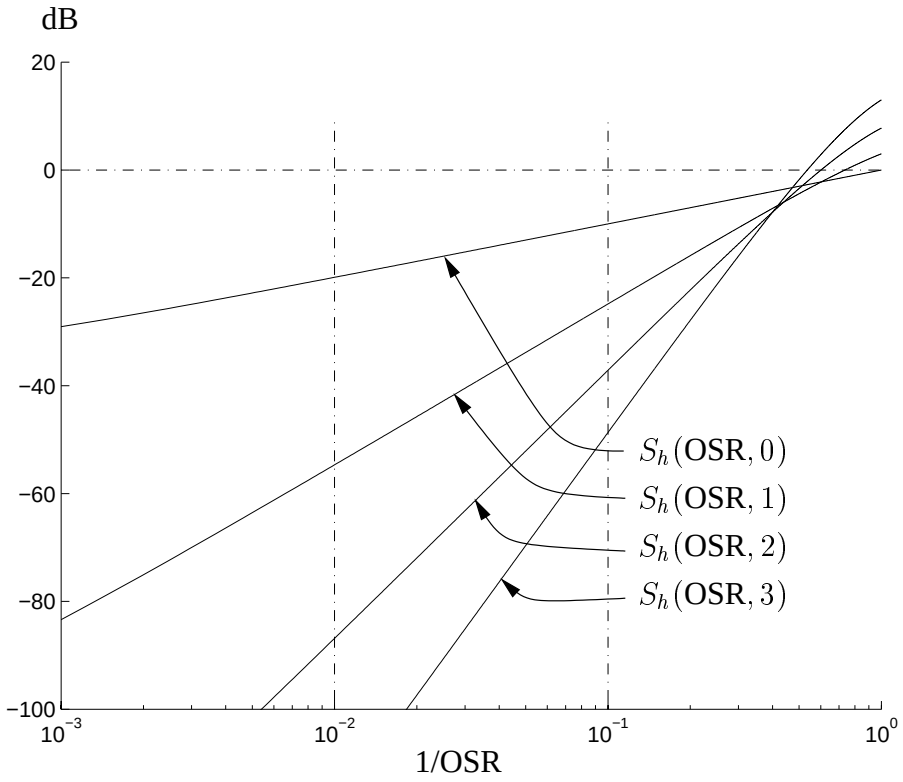
where $S_h(\text{OSR}, 1)$ is the first-order-shaping signal-band gain factor, which is shown graphically in Figure 4.11.

Estimating the Signal-Band Signal. To evaluate the DWA DAC's performance, the error signal's signal-band power should be related to a full-scale signal. Assuming that the DWA DAC is used as the feedback element in a multi-bit $\Delta\Sigma$ quantizer, it makes the most sense to relate the error signal to a full-scale *input* signal $g(k)$.

Recall from Section 4.3.1 that to obtain a high SER from a multi-bit $\Delta\Sigma$ quantizer with a low-resolution feedback DAC, the loop filter $H(f)$ must be designed with a fairly high $\text{NTF}_{\max}$ value, in which case

[28] This assumption is justifiable. Idle-tone-free ERS front ends will be discussed in the second part of this thesis.

[29] This is the *average* value for multiple implementations of the same circuit. The reader is referred to [52] for an analysis of the yield aspect.



$$S_h(\text{OSR}, Q) = \left[\frac{1}{\pi} \int_0^{\pi/\text{OSR}} (1 - e^{-j\omega})^{2Q} d\omega \right] \tag{4.25}$$

Figure 4.11: Plot of the signal-band gain factor $S_h(\text{OSR}, Q)$ as a function of the oversampling ratio (OSR) and the order of differentiation (Q).

the $\Delta\Sigma$ quantizer's full-scale input range (i.e., the resolving range) will be somewhat smaller than the feedback DAC's full-scale output range (cf. Figure 4.6). Hence, a full-scale input signal will occupy only a fraction $\alpha_{\rm quant}$ of the feedback DAC's full-scale output range.

The maximum signal power of a input sinusoid will therefore be

$$P_{\rm av}[g(k)] = \frac{[0.5\alpha_{\rm quant} P \ \text{LSB}]^2}{2} = \frac{[\alpha_{\rm quant} P \ \text{LSB}]^2}{8} \tag{4.26}$$

Conclusion. The DAC's maximum signal-to-error ratio $\text{SER}_{\rm DAC}$ can be estimated using Equations (4.24) and (4.26) as

$$\text{SER}_{\rm DAC} = \frac{P_{\rm av}[g(k)]}{P_{\rm av,inband}[m(k)]} = \frac{6}{\sigma^2 \ \text{LSB}^2 S_h(\text{OSR}, 1)} \cdot \frac{P\alpha_{\rm quant}^2 \ \text{LSB}^2}{8} \tag{4.27}$$

Assuming that a certain fixed (independent of P) chip area will be used for the implementation of the unit elements[30], σ will equal $\sqrt{P} \cdot \sigma_0$, where σ_0 is the technology's matching index for the area used for the implementation of the unit elements. The $\text{SER}_{\rm DAC}$ can then be expressed as

$$\text{SER}_{\rm DAC} = \frac{3\alpha_{\rm quant}^2}{4\sigma_0^2} \cdot \frac{1}{S_h(\text{OSR}, 1)} \tag{4.28}$$

The first factor in (4.28) represents the usual mismatch-induced linearity limitation (the normal THD level), which typically will be in the order of $60-75$ dB (for $\alpha_{\rm quant} = 1$) when a reasonable large area of a modern-technology chip is used for the implementation of the DAC's unit elements. The second factor in (4.28) represents the improvement obtained by the use of oversampling and first-order mismatch-shaping. Figure 4.11 shows that, at 10 times oversampling, the improvement will be about 24 dB. In other words, 4 extra bits of resolution can be achieved by means of 10 times oversampling and first-order mismatch-shaping; hence 16-bit performance can be obtained only if the technology's matching performance allows for the implementation of 12-bit linear DACs (at a given yield level). However, if $\alpha_{\rm quant} \simeq 0.5$, which will be a somewhat typical design approach[31] (cf. Section 4.3.1), one bit of effective

[30] This is a quite reasonable assumption. To obtain good matching and noise performance, a large chip area will have to be used even for a small number P of elements, and multiple unit elements will typically be implemented as a sub-division of the same area.

[31] A quite typical case if the DAC's absolute resolution is low and the $\Delta\Sigma$ quantizer is implemented with an aggressive loop filter to achieve a good SER performance.

resolution will be lost, and it can only be re-gained by using a slightly higher oversampling ratio or by providing better matching of the unit elements.

Several options are available to avoid the requirement for an intolerably low relative matching index σ_0. First of all, the quantizer can (and often should) be designed such that α_{quant} is fairly close to one, which (however) may require the DAC to be of fairly high resolution. Generally, at most 6-bit resolution will be of interest for UE-MS DACs employed in $\Delta\Sigma$ quantizers, because a flash quantizer of the same resolution is required for the implementation of the loop quantizer, and because the complexity of the digital DWA front end and of the routing to the unit elements increases with the resolution. If even higher resolution is needed, the use of a higher oversampling ratio may be an option (provides 9 dB improvement for each doubling of the OSR), but this should be used only as a last resort. A simple post-processing or power-up calibration may be an alternative worth considering because the required degree of matching (say $\sigma_0 \leq 10^{-4}$) usually can withstand aging of the circuit and/or changes in the ambient temperature, etc.. Also, a change of the topology may be a feasible option. Considering that the $\Delta\Sigma$ quantizer's signal-band performance has been successfully improved by increasing the order of the loop filter, it seems plausible that higher-order mismatch-shaping techniques may yield an even better signal-band suppression of the mismatch errors. $S_n(\mathrm{OSR}, 2)$ in Figure 4.11 shows that, if $m(k)$ can be made the second-order difference of a white-noise signal $n_2(k)$ with the same power as $n(k)$, approximately 6 bits (as opposed to just 4 bits) extra performance relative to the normal THD level can be obtained at OSR=10. The next section will investigate the possibility and evaluate the performance.

4.4.4 Second-Order Mismatch-Shaping DACs

The object of this section is to analyze digital front ends (encoders), which can provide second-order mismatch shaping of an array of unit elements. Such an encoder was first published by Yasuda [53] and other techniques have been published by Schreier [12] and Galton [14]. The following discussion will focus on the tree-structure encoder proposed by Galton, because it is the simplest known implementation (complexity is a drawback for these encoders).

Definitions. The following discussion will assume that the mismatch-shaping encoder controls an array of 2^N unit elements $X_i,\ i \in \{1, 2, 3, \ldots, 2^N\}$, where N is an integer. Let $\mathcal{A}_0$ denote the set of these unit elements

$$\mathcal{A}_0 = \{X_1, X_2, \ldots, X_{2^N}\} \tag{4.29}$$

and let K_i denote the gain of the single-bit DAC implemented by the unit element X_i.

The object of the following is to use the set $\mathcal{A}_0$ of unit elements to D/A convert the digital signal $d(k)$ in such a way that the D/A conversion's error signal $m(k)$ can be considered to be generated by filtering a finite-power signal $n(k)$ with an appropriate filter $1/H(f)$. More particularly, the desired mismatch-shaping operation is defined by the relations

$$\begin{aligned} f(k) &= \mathcal{D}\{d(k), \mathcal{A}_0, H(f)\} \\ &= K[\mathcal{A}_0]d(k) + m(k) \\ &= K[\mathcal{A}_0]d(k) + n(k) * h^{-1}(k) \end{aligned} \tag{4.30}$$

where $K[\mathcal{A}_0]$ denotes the average value (i.e. the linear-characteristic gain) of the elements in $\mathcal{A}_0$; $n(k) * h^{-1}(k)$ denotes the convolution of $n(k)$ and $h^{-1}(k)$; $n(k)$ is required to be bounded in magnitude; and $h^{-1}(k)$ is the impulse response of $1/H(f)$.

Figure 4.12 shows the symbol that will be used to represent a mismatch-shaping DAC based on the set $\mathcal{A}_0$ of unit-elements. The number of oriented arcs encircling $\mathcal{A}_0$ represents the order of the mismatch-shaping encoder (second-order mismatch shaping is indicated).

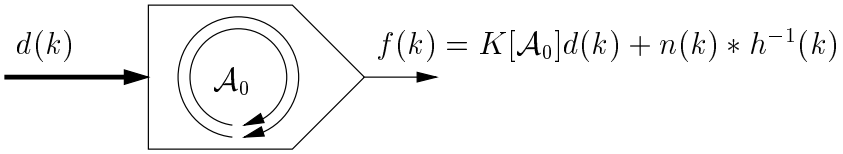


Figure 4.12: Symbol representing a second-order unit-element mismatch-shaping DAC, based on the set $\mathcal{A}_0$ of unit elements.

Tree-Structured Mismatch-Shaping Encoder. The basic idea of tree-structured mismatch-shaping encoders is that they transform a single complex problem into several simpler problems[32]. More particularly, the problem of implementing a mismatch-shaping DAC with 2^N unit elements is transformed into the problem of implementing two mismatch-shaping DACs (each with 2^{N-1} unit elements) and a *node separator* as shown in Figure 4.13.

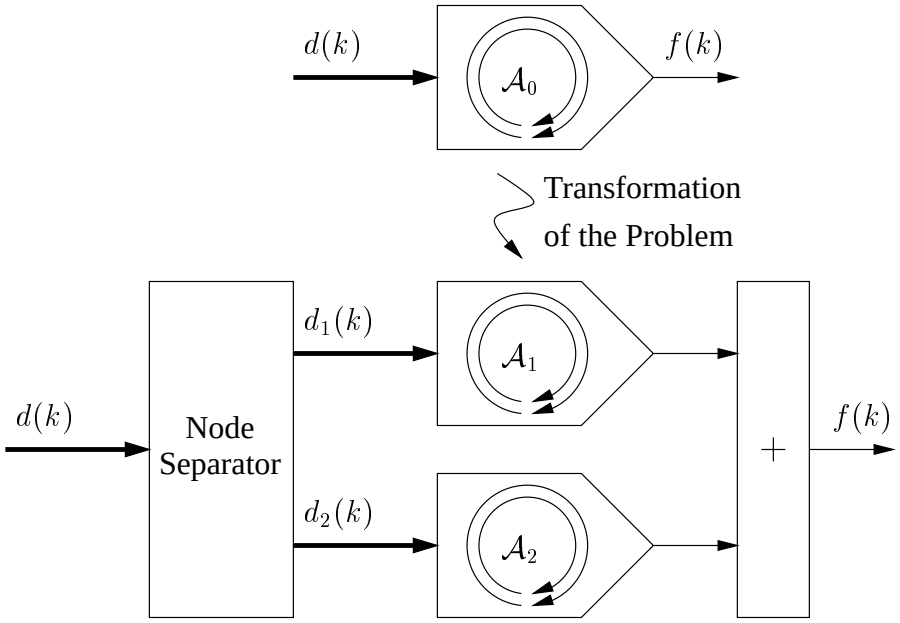


Figure 4.13: Fundamental transformation used for the implementation of tree-structure mismatch-shaping encoders.

The two smaller mismatch-shaping DACs each control one of the two sets $\mathcal{A}_1$ and $\mathcal{A}_2$ of unit elements defined as[33]

$$\mathcal{A}_1 = \{X_1, X_2, \ldots, X_{2^{(N-1)}}\} \quad \text{and} \quad \mathcal{A}_2 = \mathcal{A}_0 \setminus \mathcal{A}_1 \tag{4.31}$$

The node separator must fulfill

$$d(k) = d_1(k) + d_2(k) \tag{4.32}$$

[32] Divide et impera!

[33] I.e. , $\mathcal{A}$ is split in two sets of equal size.

which can be written in the form

$$d_1(k) = \frac{d(k) + t(k)}{2} \quad \text{and} \quad d_2(k) = \frac{d(k) - t(k)}{2} \tag{4.33}$$

where $t(k)$ is chosen such that $d_1(k)$ and $d_2(k)$ are integers smaller than or equal to 2^{N-1}.

Using definition (4.30), it follows that

$$\begin{aligned} f(k) &= \mathcal{D}\{d_1(k), \mathcal{A}_1, H(f)\} + \mathcal{D}\{d_2(k), \mathcal{A}_2, H(f)\} \\ &= K[\mathcal{A}_1]d_1(k) + n_1(k) * h^{-1}(k) + K[\mathcal{A}_2]d_2(k) + n_2(k) * h^{-1}(k) \\ &= K[\mathcal{A}_1]d_1(k) + K[\mathcal{A}_2]d_2(k) + [n_1(k) + n_2(k)] * h^{-1}(k) \end{aligned} \tag{4.34}$$

Because $K[\mathcal{A}_0] = 0.5[K[\mathcal{A}_1] + K[\mathcal{A}_2]]$, it follows that (4.34) can be expressed as

$$f(k) = d(k)K[\mathcal{A}_0] + \frac{t(k)\,(K[\mathcal{A}_1] - K[\mathcal{A}_2])}{2} + [n_1(k) + n_2(k)] * h^{-1}(k) \tag{4.35}$$

The considered transformation (cf. Figure 4.13) is permissible only if (4.35) complies with the definition (4.30) for mismatch shaping. By comparing the two equations, it is concluded that the transformation is permissible only if $t(k)$ can be written in the form

$$t(k) = n_3(k) * h^{-1}(k) \tag{4.36}$$

where $n_3(k)$ is required to be bounded in magnitude. For simplicity, $t(k)$ is usually chosen as a signal that attains only the values -1, 0, and 1.

Single-bit DACs are inherently linear and can be considered to perform mismatch shaping of arbitrarily high order for any signal band. Hence, recursive use of the considered transformation, until $d(k)$ becomes single-bit signals, will result in a DAC for which the error signal has the same spectral properties as the signals $t_i(k)$ generated by the node separators. Figure 4.14 illustrates an eight-unit-element mismatch-shaping DAC implemented using this technique.

Tree-Structured First-Order Mismatch-Shaping DAC. To implement a baseband first-order UEMS DAC, each of the signals $t_i(k)$ generated by the node separators, should be of the form (4.36),

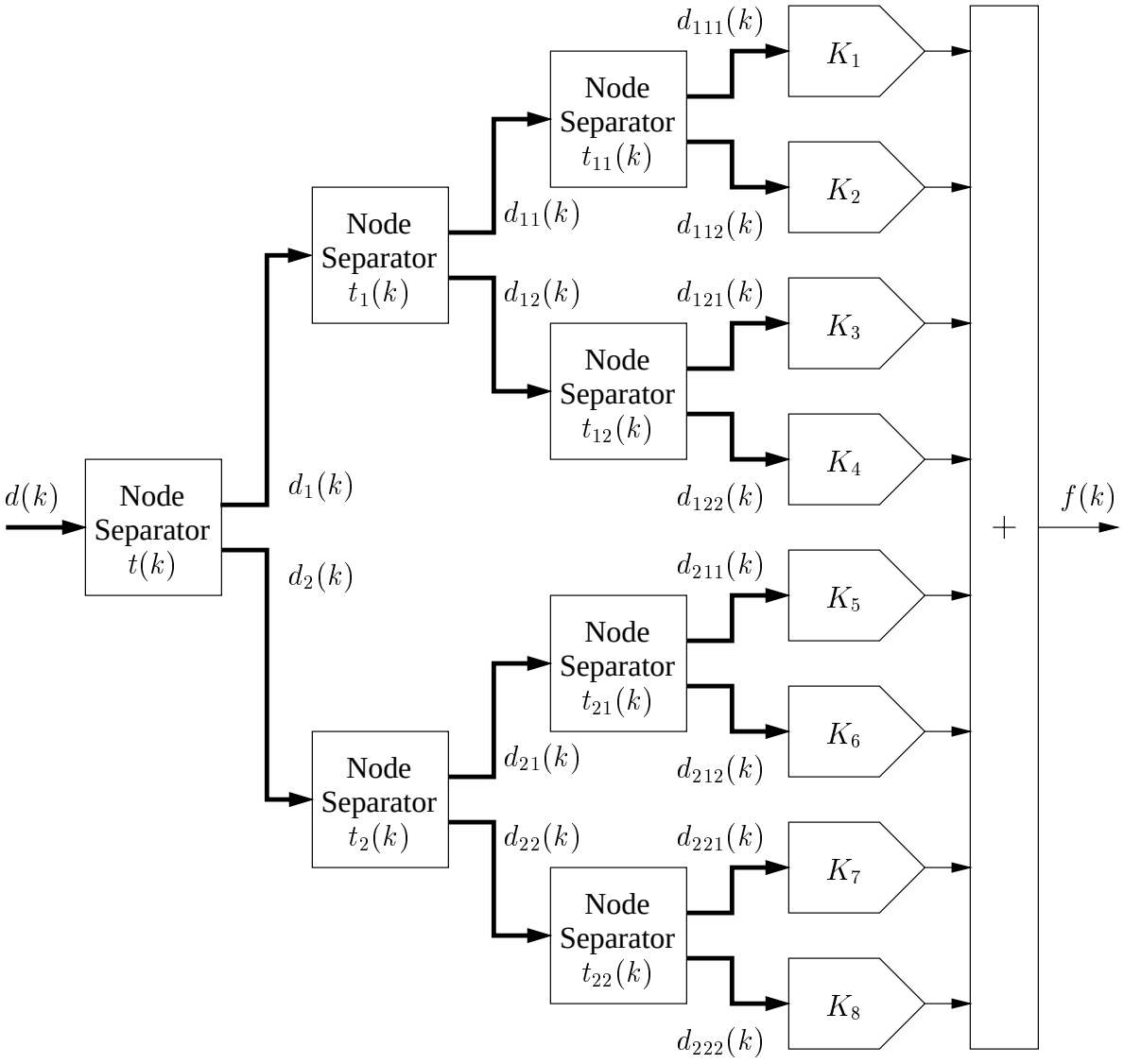


Figure 4.14: Implementation of an eight-unit-elements tree-structured mismatch-shaping DAC.

where $h^{-1}(k) = \{1, -1, 0, 0, 0, \dots\}$. In other words, the signals $t_i(k)$ must each fulfill

$$\exists W \quad \text{for which} \quad \left| \sum_{j=0}^{k} t_i(j) \right| < W \quad \text{for all } k \tag{4.37}$$

Equation (4.37) can be fulfilled for $W = 1$. Consider, for example, the first node separator. When $d(k)$ is an even number, then $t(k)$ is chosen as 0 because $d(k)$ can be split evenly into $d_1(k)$ and $d_2(k)$. However, when $d(k)$ is odd, either $d_1(k)$ or $d_2(k)$ must be one larger than the other, i.e., $|t(k)| = 1$. By choosing $t(k)$ of the opposite polarity to that of $\sum_{j=0}^{k-1} t(j)$, it follows that (4.37) will be fulfilled for $W = 1$; consequently first-order mismatch-shaping is obtained. This operation can be implemented using only a few logic gates and a toggle flip-flop for each node separator.

Tree-Structured Band-Pass Mismatch-Shaping DAC. Assume that the signal band is a small frequency range centered around $f_s/4$. With respect to this signal band, a first-order UE-MS DAC can be implemented as described above, except that now $h^{-1}(k) = \{1, 0, 1, 0, 0, 0, \dots\}$.

Tree-Structured Second-Order Mismatch-Shaping DAC. The implementation of a second-order mismatch-shaping tree-structured DAC is, in principle, the same as that of the first-order one described above. The only difference is in the generation of the signals $t_i(k)$. For a baseband DAC, the object is to assure that the second-order sum of $t(k)$ remains bounded, a task which can be implemented by means of a $\Delta\Sigma$ quantizer (shown in Figure 4.15).

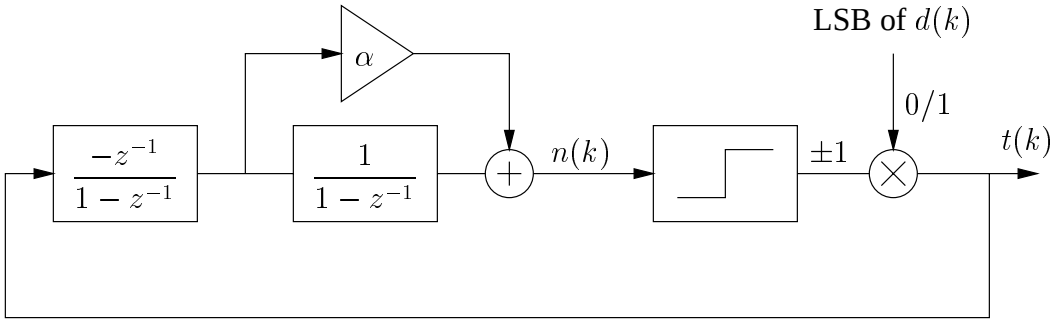


Figure 4.15: Node separator for use in a tree-structured second-order mismatch-shaping DAC.

The output of the single-bit truncation element decides the polarity of the next nonzero $t(k)$ value. The multiplier at the truncator's output assures that $t(k)$ will be nonzero only when $d(k)$ is odd.

The multiplier is, in reality, the main problem for UE-MS encoders of orders higher than 1 because long sequences of even numbers will deactivate the quantizer's feedback and possibly thereby cause large values of $n(k)$. Consider, for example, a situation where the outputs of both integrators equal 1 and the following 10 values of $d(k)$ are even. During this period of even values, the output of the first integrator will remain constant 1, whereas the second integrator's output will increase to 11. Only then, when $d(k)$ finally attains an odd value, can the quantizer attempt to control the state variables.

A second-order node separator is in principle unstable[34], but usually $d(k)$ will attain odd values once in a while, in which case stability can be preserved[35]. In a best-case scenario, $d(k)$ will be odd/even according to a stochastic process for which both outcomes are equally likely to occur, and in which there is no sample-to-sample correlation.

4.4.5 Performance of Second-Order Mismatch-Shaping DACs

The following will estimate the performance of first-order and second-order tree-structured UE-MS DACs on the basis of simulations. The simulations will be based on the assumption that for each sample and for each node separator, it is entirely random whether $|t_i(k)|$ is one or zero. This assumption favors the second-order mismatch-shaping DACs, but even then, the conclusion turns out to be in favor of first-order systems; hence the assumption is acceptable for this purpose.

Estimating the Error Signal. The performance is determined by the spectral properties of the DACs' error signal, which can be expressed as (cf. Equation (4.35))

$$m(k) = t(k)\delta + [n_1(k) + n_2(k)] * h^{-1}(k) \quad \text{where} \quad \delta = \frac{(K[\mathcal{A}_1] - K[\mathcal{A}_2])}{2} \tag{4.38}$$

[34] In the strict sense.

[35] Reset operations will most likely be necessary on certain occasions.

Considering that $n_1(k) * h^{-1}(k)$ and $n_2(k) * h^{-1}(k)$ are the error signals from the two D/A conversions $\mathcal{D}\{d_1(k), \mathcal{A}_1, H(f)\}$ and $\mathcal{D}\{d_2(k), \mathcal{A}_2, H(f)\}$, which are implemented by completing the tree (cf. Figures 4.13 and 4.14), it follows that

$$\begin{aligned} m(k) &= t(k)\delta \\ &\quad +[t_1(k)\delta_1 + t_2(k)\delta_2] \\ &\quad +[t_{11}(k)\delta_{11} + t_{12}(k)\delta_{12} + t_{21}(k)\delta_{21} + t_{22}(k)\delta_{22}] \\ &\quad +\dots \\ &= \sum_i t_i(k)\delta_i \end{aligned} \tag{4.39}$$

Provided the above assumption, each of the signals $t_i(k)$ will have the same spectral power density, hence the performance can be simulated and evaluated on the basis of the spectral properties of only a single $t(k)$ signal.

Simulation Results. A node separator was simulated for $H(f) = \frac{1}{1-z^{-1}}$ (first order) and for $H(f) = \frac{5-3z^{-1}}{(1-z^{-1})^2}$ (second-order[36]). The estimated spectral power densities of the two $t(k)$ signals obtained are shown in Figure 4.16. The DFTs are based on each 16384 samples; and for clarity, they have been smoothed with a 10-tap moving-average filter.

For the first-order mismatch-shaping encoder, the spectral power density of $t(k)$ is approximately proportional to the frequency, which is the expected performance (cf. Equation (4.20)). Ideally, the spectral power density of the $t(k)$ signal generated by the second-order encoder should be proportional to the second power of the frequency, which can actually be observed at low frequencies. However, at higher frequencies ($f > f_s/10$), the spectral power density is fairly constant. Hence, the OSR must be high to take advantage of the more efficient suppression of low-frequency errors.

The different behavior of first- and second-order encoders can be explained as follows. A first-order encoder is in "standby" mode when $t(k) = 0$, and it will make use of the first available chance (i.e., as soon as $|t(k)| = 1$) to correct the accumulated error. A second-order encoder, on the other hand, is not in

[36]This seems to be a good design, compared to simulation results obtained for other α values (cf. Figure 4.15).

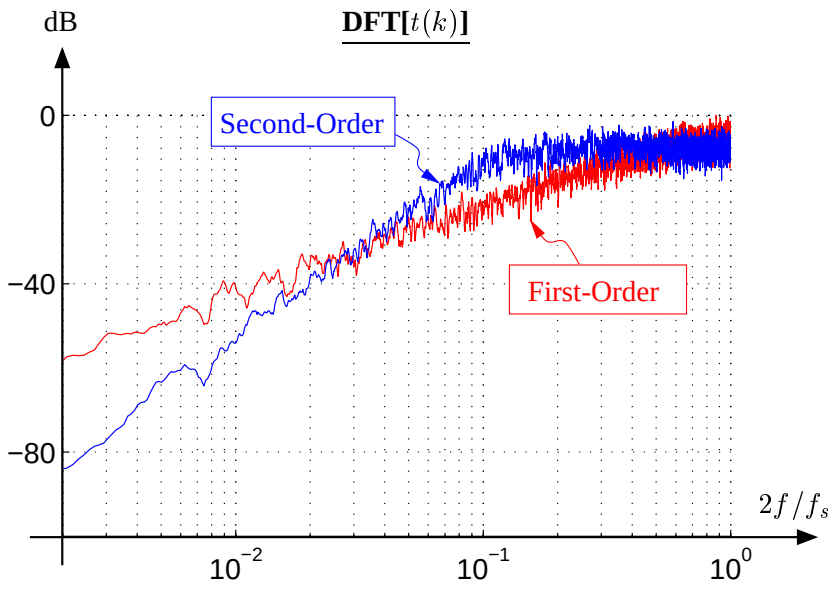


Figure 4.16: Spectral composition of the control signal $t(k)$ when $d(k)$ is equally likely to be odd/even and when there is no sample-to-sample correlation.

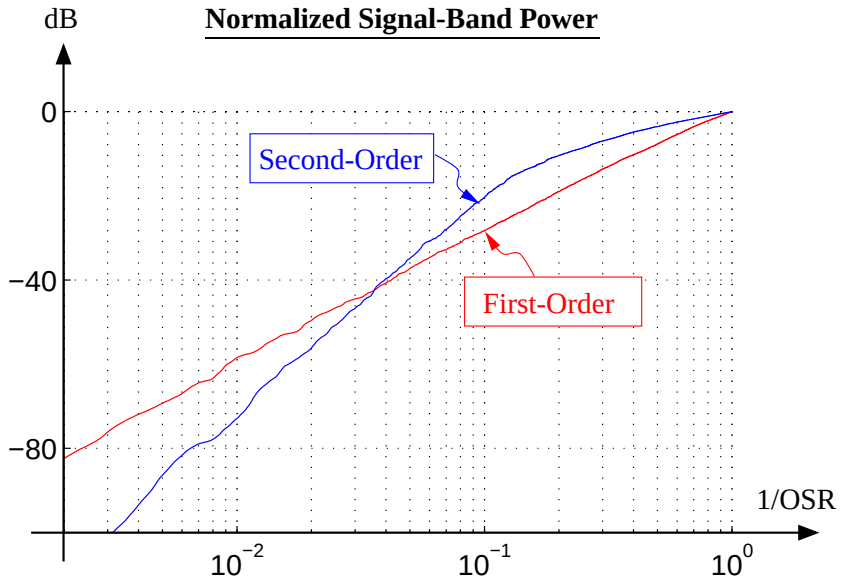


Figure 4.17: The signal-band power of $t(k)$ normalized with respect to its Nyquist-band power estimated from (4.40).

standby mode when $t(k) = 0$, because the second integrator is clocked for every sample and it may have a nonzero input. Hence, a second-order encoder will often have to step in the "wrong" direction with respect to the first-order sum of $t(k)$ in order to control the second accumulator's output. In essence, this implies that a second-order encoder will not be able to adjust for high-frequency variations in the first-order sum, which is why the spectral power density is nearly uniform for high frequencies. In other words, a second-order encoder is so busy correcting for low-frequency errors that it neglects the high-frequency performance.

Estimation of the Performance. To properly evaluate the performance, the DACs' SER should be calculated. It is important to understand that the error signal's Nyquist-band power will be *independent*[37] of the order of the mismatch-shaping encoder which is used to control the unit elements; consequently, it is only the shape of the error signal's spectral power density that affects the signal-band performance. Figure 4.17 shows the error signal's signal-band power normalized with respect to its Nyquist-band power as a function of the oversampling ratio. By making use of the property that the error signal's Nyquist-band power is independent of the encoder used, and by calculating from (4.25) that $S_i(1,1) = \sqrt{2}$, it follows from (4.28) that the Nyquist-band power of $m(k)$ can be estimated as

$$P_{\text{av}}[m(k)] \simeq 2\frac{\sigma_0^2}{\alpha_{\text{quant}}^2} P_{\text{av,max}}[g(k)] \tag{4.40}$$

where α_{quant} and σ_0 are defined in the same way as in (4.28). In other words, the Nyquist-band SER_{DAC} is approximately $\frac{\alpha_{\text{quant}}^2}{2\sigma_0^2}$, and the improvement, which can be obtained by means of oversampling and mismatch shaping, can be estimated from Figure 4.17.

Conclusion. Whether or not second-order mismatch-shaping encoders are useful depends somewhat on the performance required and on the oversampling ratio. Figure 4.17 shows that first-order encoders actually yield a *better* performance than second-order encoders if the oversampling ratio is less than about 25, at which point the SER will be

$$\text{SER}_{\text{DAC@OSR}=25} \simeq \frac{\alpha_{\text{quant}}^2}{2\sigma_0^2} + 40 \text{ dB} \tag{4.41}$$

[37] Because the δ parameters in (4.39) are the same and the power of each $t_i(k)$ signal is independent on the encoder.

The first term in (4.41) can typically be made as large as 60–75 dB without employing calibration; hence, second-order mismatch-shaping encoders are usually not necessary for applications with an effective resolution of less than 16–18 bits. In fact, the prospect of a significantly improved performance will, in general, be required to justify the extra circuit complexity that is associated with the implementation of a second-order encoder; they will rarely be used for oversampling ratios of less than 100. However, at OSR=100 the SER will typically be so high that it is the SNR (device noise) that limits the performance. Hence, the conclusion is that second-order UE-MS encoders are very interesting from an academic point of view, but they are, in general, not useful or necessary for the implementation of high-performance circuits. Although this conclusion is not reached in either [12] or [14], it is worthwhile to notice that both references provide simulation results that support the above conclusion[38].

To justify the above derivation further, notice that it is the last layer of node separators in a tree-structured UE-MS encoder that contributes the most noise; consequently, the encoder's performance cannot be improved significantly by allowing $t(k)$ to also attain the values ± 2 (which cannot be allowed in the last layer). Another implication is that local matching is more important than global matching, e.g., in Figure 4.14 it is preferable to emphasize the matching of X_1 to X_2 and X_7 to X_8, rather than to emphasize the matching of X_1 to X_8 and X_2 to X_7. This observation is very useful in the layout procedure of tree-structure first-order UE-MS DACs, because the matching accuracy depends on the physical distance (cf. page 59).

4.4.6 Mismatch-Shaping Encoders in Perspective

Considering that they are generally the simplest and yield the best performance, it makes sense to attempt to optimize the design of first-order UE-MS DACs. In particular, it is important to develop simple ways to implement high-resolution first-order UE-MS DACs, and also to that these DACs do not produce idle tones.

[38] Their crossover point (cf. Figure 4.17) may occur at a slightly lower oversampling ratio (say 20), but this slight improvement does alter the provided conclusion. Their results are based on the simulation of a full DAC, and not just a single node separator.

4.5 Noise Limitation

All analog systems are subject to performance limitations due to noise and nonlinear behavior of the analog circuit elements. As discussed in [43], a $\Delta\Sigma$ quantizer can be implemented to be very robust with respect to the nonlinearity of the analog loop filter; hence, device noise is the main factor to consider.

4.5.1 Discrete-Time Delta-Sigma Quantizers

Thus far, most $\Delta\Sigma$ quantizers have been implemented with a (discrete-time) switched-capacitor loop filter[39]. As discussed in [43], it is mainly the first integrator (cf. Figure 4.4) that determines the $\Delta\Sigma$ quantizer's noise performance.

In the best-case scenario, which can be obtained if the first integrator is implemented as shown in Figure 4.18, only one capacitor will contribute noise[40].

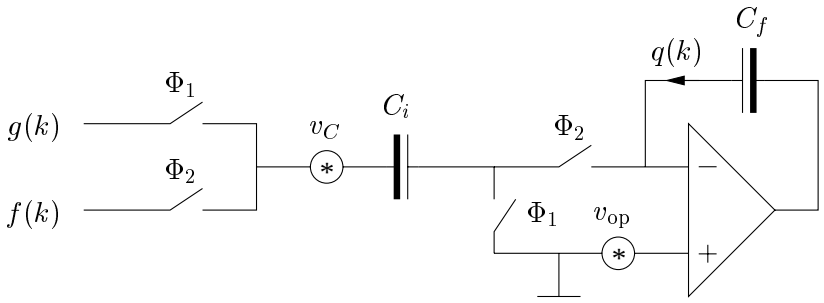


Figure 4.18: Best-case implementation (with respect to noise) of a switched-capacitor $\Delta\Sigma$ quantizer's input stage.

Noise Analysis. In clock phase Φ_1, the input signal $g(k)$, and a thermal-noise component $v_c(k, \Phi_1)$ are sampled on the input capacitor C_i. In the following clock phase Φ_2, a charge portion $q(k)$ is transferred

[39] Switched-current techniques have also been employed, but the performance did not measure up to that of switched-capacitor implementations.

[40] If the $\Delta\Sigma$ quantizer employs a multi-bit DAC in the feedback path, then C_i is sectioned in a number of smaller capacitors in parallel. Because the noise performance will be the same, C_i is (for simplicity) modeled as only one capacitor of the total capacitance.

from C_i to the integrating capacitor C_f. The charge $q(k)$ is proportional to the voltage variation across C_i and can be expressed as

$$\begin{aligned} q(k) &= C_i\left[[g(k) + v_C(k,\Phi_1)] - [f(k) + v_C(k,\Phi_2) - v_{\rm op}(k,\Phi_2)]\right] \\ &= C_i\left[e(k) + [v_C(k,\Phi_1) - v_C(k,\Phi_2) + v_{\rm op}(k,\Phi_2)]\right] \end{aligned} \tag{4.42}$$

where $v_C(\Phi_2)$ is the thermal noise during Φ_2[41], $av_{\rm op}(k,\Phi_2)$ is the opamp's input-referred noise during Φ_2, and $e(k) \triangleq g(k) - f(k)$.

Accordingly, the $\Delta\Sigma$ quantizer's signal-to-noise ratio (SNR) can be estimated from

$$\mathrm{SNR}_{\max} \leq \frac{P_{\rm av,max}[g(k)]}{P_{\rm inband}[v_C(k,\Phi_1) - v_C(k,\Phi_2) + v_{\rm op}(k,\Phi_2)]} \tag{4.43}$$

The maximum signal power $P_{\rm av,max}[g(k)]$ can be estimated from

$$P_{\rm av,max}[g(k)] \leq \frac{V_{\rm supply}^2}{8} \tag{4.44}$$

where $V_{\rm supply}$ is the circuit's supply voltage difference. Assuming that the three noise contributions are stochastically independent, it follows that

$$P_{\rm inband,noise} = P_{\rm inband}[v_C(k,\Phi_1)] + P_{\rm inband}[v_C(k,\Phi_2)] + P_{\rm inband}[v_{\rm op}(k,\Phi_2)] \tag{4.45}$$

Flicker Noise. The switched-capacitor integrator can easily be designed to incorporate correlated double sampling (CDS) techniques, which efficiently suppress the opamp's offset and flicker noise [54], and hence leaves only thermal noise to be considered.

Thermal Noise. Thermal noise is by nature continuous-time and wide-band. When thermal noise is sampled, the total noise power will alias into the Nyquist range (cf. Figure 2.2 and [29]), and (to a very good approximation) the sampled noise will have a uniform spectral power density.

As discussed in [29], the Nyquist-band power of $v_C(k,\Phi_x)$ is kT/C_i, where k is Boltzmann's constant, T is the absolute temperature, and C_i is the capacitance of the input capacitor. Because the spectral

[41] More precisely: at the time instance when the switches that are controlled by Φ_2 are opened.

power density is uniform, the input-referred signal-band noise power can be calculated from

$$P_{\text{inband}}[v_C(k,\Phi_1)] = P_{\text{inband}}[v_C(k,\Phi_2)] = \frac{1}{\text{OSR}}\frac{kT}{C_i} \tag{4.46}$$

The opamp's thermal noise depends on its topology. For an OTA[42], the input-referred signal-band noise power can be estimated as [55]

$$P_{\text{inband}}[v_{\text{op}}(k,\Phi_2)] = \frac{1}{\text{OSR}}\frac{4kT}{3C_i} \tag{4.47}$$

whereas, for a two-stage opamp, the input-referred signal-band noise power can be estimated as [55]

$$P_{\text{inband}}[v_{\text{op}}(k,\Phi_2)] = \frac{1}{\text{OSR}}\frac{4kT}{3C_c} \tag{4.48}$$

where C_c is the opamp's internal frequency-response compensation (Miller-effect) capacitor. Considering that the maximum capacitance is constrained by the available chip area, it is reasonable to assume that $C_c \simeq C_i$. When CDS is used, the opamp's thermal noise power will be doubled (because it is sampled twice), and hence the $\Delta\Sigma$ quantizer's input-referred signal-band noise power can be estimated (roughly) as

$$P_{\text{inband,thermal}} \simeq \frac{5}{\text{OSR}}\frac{kT}{C_i} \tag{4.49}$$

By combining Equations (4.43), (4.44), and (4.49) it follows that

$$\text{SNR} \leq V_{\text{supply}}\sqrt{\frac{\text{OSR}\cdot C_i}{40kT}} \tag{4.50}$$

Assuming that the supply voltage is 5 V and that the input capacitance is at most 10 pF, it is found that the Nyquist-band SNR is limited to about 92 dB. For each decade of oversampling, 10 dB improvement will result, so the SNR is limited to about 102 dB at 10 times oversampling. Fully differential implementations may have approximately 3 dB better performance.

Conclusion. The assumptions made for the supply voltage and the input capacitance are slightly optimistic, so it may be concluded that it is not possible to obtain substantially more than about 100 dB

[42] Operational Transconductance Amplifier

performance when the oversampling ratio is as low as 10. The bandwidth *per se* does not influence the noise performance, but the power consumption will increase with both the capacitance C_i and the sampling frequency. Hence, the power consumption, bandwidth, and noise performance are very much interrelated [56].

4.5.2 Continuous-Time Delta-Sigma Quantizers

Figure 4.19 shows the basic implementation of a continuous-time (CT) $\Delta\Sigma$ quantizer[43]. Notice that the sampling operation is performed only at the loop quantizer, and hence that all aliasing errors (including noise aliasing) that occur at this point are efficiently suppressed by the loop filter's large signal-band gain when they are referred to the input. This implies that an improved noise performance can be obtained, and that a separate anti-aliasing filter is not needed for CT $\Delta\Sigma$ quantizers (a considerable advantage).

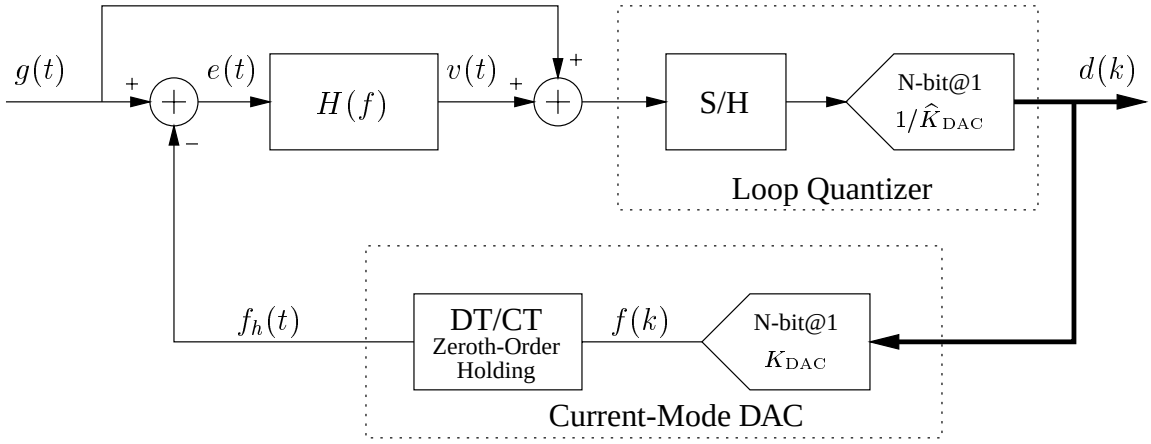


Figure 4.19: Continuous-time $\Delta\Sigma$ quantizer.

Another advantage is that the loop filter can be implemented with opamps having a lower gain-bandwidth product[44] than that required for opamps employed in SC loop filters of comparable performance, and hence a higher sampling frequency or a lower power consumption can be obtained.

The main problem of CT $\Delta\Sigma$ quantizers is that they are very sensitive to the dynamic errors of the

[43] The loop quantizer is (of course) a discrete-time block; the name refers to the loop filter only.

[44] Switched-capacitor filters require opamps with a gain-bandwidth product of at least 4–5 times the sampling frequency [29].

feedback DAC.

The Input Stage. Figure 4.20 shows the basic implementation of a typical input stage for CT $\Delta\Sigma$ quantizers. The resistor R will contribute thermal noise; and the opamp, as well as the current-mode DAC, will contribute both thermal and flicker noise.

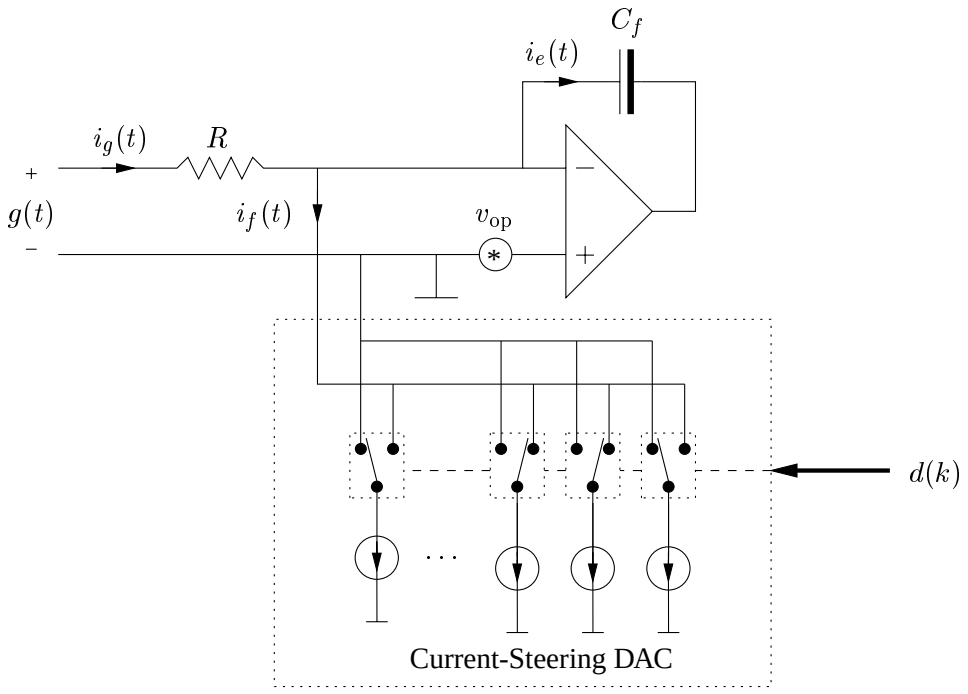


Figure 4.20: Input stage of a continuous-time $\Delta\Sigma$ quantizer.

Noise from the Feedback DAC. Insight in this case may best be obtained by considering the fully-differential implementation shown in Figure 4.21. In practice, most implementations will be fully differential.

The DAC is modeled as a differential current source providing a differential reference current $I_{\rm ref}$ which is converted according to $d(k)$ into the differential feedback current $i_f(t) = d(k)I_{\rm ref}$. In the actual implementation, the differential current source will be implemented as multiple differential current sources operating in parallel, and they will be switched individually (i.e. multiplied by ± 1) by the current splitter

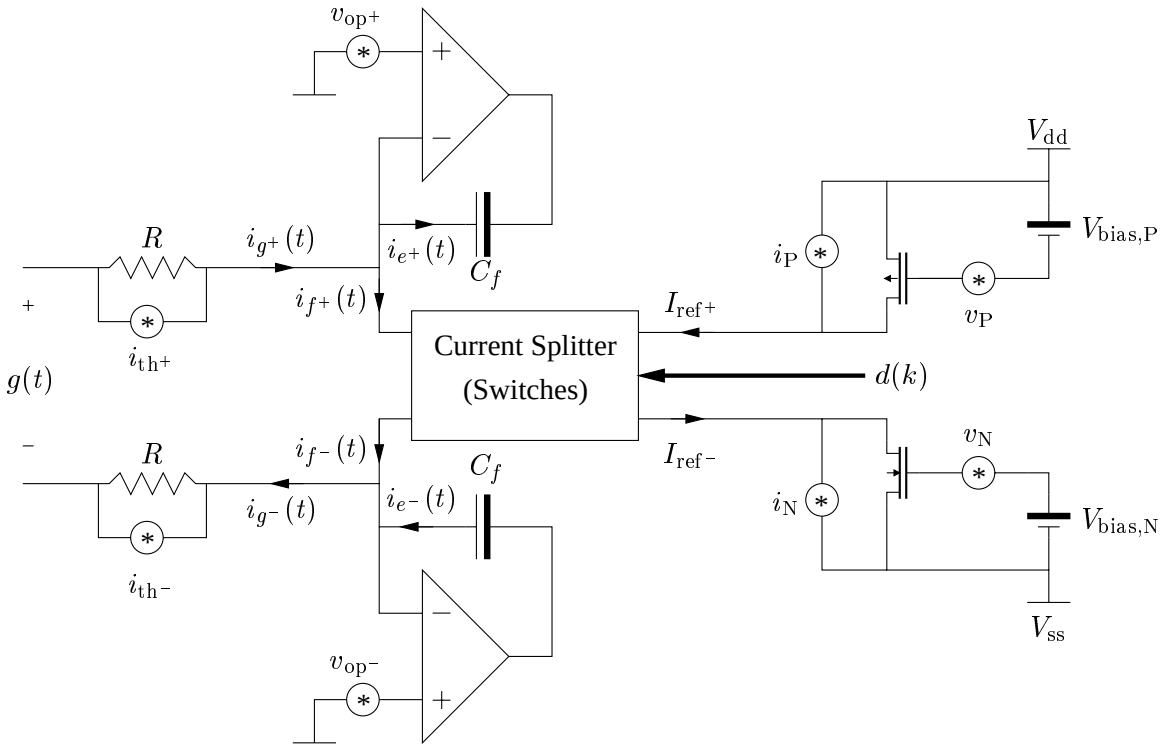


Figure 4.21: Noise model for a fully-differential continuous-time $\Delta\Sigma$ quantizer.

(cf. Figure 4.20). The operation can be described in the form

$$i_{\mathrm{f}+}(t) + i_{\mathrm{f}-}(t) = d(k)[I_{\mathrm{ref}+} + I_{\mathrm{ref}-}] \quad (4.51)$$

$$i_{\mathrm{f}+}(t) - i_{\mathrm{f}-}(t) = I_{\mathrm{ref}+} - I_{\mathrm{ref}-} \quad (4.52)$$

Because the current splitter is merely an array of data-controlled switches, it will not produce noise.

Flicker noise is mainly caused by fluctuations in the current-source MOSFETs' effective threshold voltage, and so it can be modeled by the gate-referred noise voltages v_{N} and v_{P}. When the flicker noise is referred to the input $g(t)$, it will become modulated by $d(k)$ and magnified by $R \cdot g_m$, where g_m is the respective current source's transconductance. Since the reference current is switched and not dumped, the modulation by $d(k)$ will not change the flicker noise's power, but merely its spectral composition. Hence the input-referred flicker noise can be calculated as

$$\begin{aligned} v_{\mathrm{flicker,input,DAC}} &= \sqrt{v_{\mathrm{P}}^2 g_{m,\mathrm{P}}^2 R^2 + v_{\mathrm{N}}^2 g_{m,\mathrm{N}}^2 R^2} \\ &= R\sqrt{v_{\mathrm{P}}^2 g_{m,\mathrm{P}}^2 + v_{\mathrm{N}}^2 g_{m,\mathrm{N}}^2} \end{aligned} \quad (4.53)$$

Because $d(k)$ presumably will have most of its power in the signal band[45], and because flicker noise has a low bandwidth, $v_{\mathrm{flicker,input,DAC}}$ should be considered to be signal-band noise.

The total power of the flicker-noise voltages v_{N} and v_{P} can be controlled independently by changing the transistor area while preserving the aspect ratio, therefore (for simplicity), the following discussion will be based on the assumption that $v_{\mathrm{N}} \simeq v_{\mathrm{P}}$, and hence

$$v_{\mathrm{flicker,input,DAC,inband}} = v_{\mathrm{N,P}} \left[R\sqrt{g_{m,\mathrm{P}}^2 + g_{m,\mathrm{N}}^2} \right] \quad (4.54)$$

The current sources' thermal noise can (assuming strong inversion) be represented by the (signal-band) noise currents

$$i_{\mathrm{N}} = \sqrt{4kT\Delta f g_{m,\mathrm{N}}} \quad (4.55)$$

$$i_{\mathrm{P}} = \sqrt{4kT\Delta f g_{m,\mathrm{P}}} \quad (4.56)$$

[45] Assuming that $d(k)$ is the output from a *multi-bit* $\Delta\Sigma$ quantizer.

where Δf is the width of the signal band. When referred to the input, the (signal band) thermal noise voltage takes the form

$$v_{\mathrm{thermal,input,DAC,inband}} = \sqrt{4kTR\Delta f}\left[R\sqrt{[g_{m,\mathrm{N}}^2+g_{m,\mathrm{P}}^2]}\,\right] \tag{4.57}$$

Notice that both the flicker noise and the thermal noise (cf. (4.54) and (4.57)) are proportional to the gain factor $R\sqrt{[g_{m,\mathrm{N}}^2+g_{m,\mathrm{P}}^2]}$, and hence the MOSFETs' transconductance should be minimized.

The transconductances can be expressed as

$$g_{m,\mathrm{P}} = \frac{2I_{\mathrm{ref}+}}{V_{\mathrm{eff,P}}} \quad \text{where} \quad V_{\mathrm{eff,P}} = V_{\mathrm{bias,P}} - |V_{\mathrm{th,P}}| \tag{4.58}$$

$$g_{m,\mathrm{N}} = \frac{2I_{\mathrm{ref}-}}{V_{\mathrm{eff,N}}} \quad \text{where} \quad V_{\mathrm{eff,N}} = V_{\mathrm{bias,N}} - V_{\mathrm{th,N}} \tag{4.59}$$

To optimize the input signal swing while avoiding a common-mode component in the reference currents[46], it is preferable that the opamps' virtual ground potential be $V_{\mathrm{supply}}/2$, and hence (to keep the MOSFETs operating in saturation) it is a requirement that[47]

$$V_{\mathrm{eff,N}} \leq V_{\mathrm{supply}}/2 \quad \text{and} \quad V_{\mathrm{eff,P}} \leq V_{\mathrm{supply}}/2 \tag{4.60}$$

Because the reference current must be able to balance the input current, it follows that

$$I_{\mathrm{ref}+} = I_{\mathrm{ref}-} \geq \frac{V_{\mathrm{supply}}}{2R} \tag{4.61}$$

By combining the Equations (4.58) through (4.61) it can be shown that

$$R\sqrt{[g_{m,\mathrm{N}}^2+g_{m,\mathrm{P}}^2]} > R\sqrt{\left(\frac{2}{R}\right)^2+\left(\frac{2}{R}\right)^2} = \sqrt{8} \tag{4.62}$$

and in conclusion, Equations (4.57) and (4.64) can be evaluated as

$$v_{\mathrm{thermal,input,DAC,inband}} \geq \sqrt{4kTR\Delta f}\sqrt{8} \tag{4.63}$$

$$v_{\mathrm{flicker,input,DAC}} \geq v_{\mathrm{N,P}}\sqrt{8} \tag{4.64}$$

[46] Preferable in order to minimize the transconductance (cf. Equations (4.58) and (4.59)).

[47] The current-source MOSFETs will typically be cascode-coupled, and hence they can potentially be operated in the triode region. However, to avoid an additional noise contribution from the cascode transistors, the output impedance of the current-source MOSFETs should be reasonable high, i.e., they should be operated in saturation.

Overall Thermal Noise Performance. For simplicity, assume that the opamps are designed to have an equivalent thermal-noise resistance of approximately R. Because the opamp's gain-bandwidth product must be higher than the highest signal-band frequency, it follows that the opamps and the resistors will have the same spectral power density $4kTR\Delta f$. The overall thermal-noise performance is therefore (cf. (4.63))

$$\begin{aligned} P_{\text{inband,thermal,continuous,differential}} &= 4kTR\Delta f\,[8+1+1+1+1] \\ &= 48kTR\Delta f \end{aligned} \tag{4.65}$$

The result (4.65) will now be compared to the derived thermal-noise performance estimate (4.49) for DT $\Delta\Sigma$ quantizers. Considering that the input capacitance C_i has an equivalent resistance of

$$R_{\text{eqv}} = \frac{1}{C_i f_s} = \frac{1}{C_i \cdot \text{OSR} \cdot 2\Delta f} \tag{4.66}$$

it follows that DT $\Delta\Sigma$ quantizers' thermal-noise performance can be expressed as

$$P_{\text{thermal,inband,discrete}} = 10kTR_{\text{eqv}}\Delta f \tag{4.67}$$

which, for a differential implementation yields

$$P_{\text{thermal,inband,discrete,differential}} = 20kTR_{\text{eqv}}\Delta f \tag{4.68}$$

From this point of view, a DT $\Delta\Sigma$ quantizer's thermal noise performance is approximately 4 dB better than that of a CT $\Delta\Sigma$ quantizer. However, this conclusion is somewhat misleading, because generally it will be possible (for the same power consumption and using the same technology) to choose the physical resistor R significantly smaller than R_{eqv}, therefore, CT $\Delta\Sigma$ quantizers can be designed to have a better noise performance. This is especially true for low-frequency (audio) quantizers, where $R = \frac{20}{48}R_{\text{eqv}} = 1$ kΩ would require C_i to be approximately 1 nF for $\Delta f = 20$ kHz and OSR $= 10$. In other words, assuming that OSR $= 10$ is sufficient to obtain the required SER performance, a DT $\Delta\Sigma$ quantizer will have to be operated at a much higher OSR (in the order of 1000) using practical capacitors to obtain the SNR performance of a CT $\Delta\Sigma$ quantizer with $R = 1$ kΩ. Hence, the CT $\Delta\Sigma$ quantizer is highly preferable from a power-consumption point of view. Even when the bandwidth is so large that the same OSR is required, the CT $\Delta\Sigma$ quantizer can be designed to have a lower power consumption, because the opamps are not subject to the same high requirements.

To get a better feeling for the numbers quoted, it may be interesting to note that for $R = 1\,\text{k}\Omega$ and $\Delta f = 1$ MHz, the signal-band thermal noise of a CT $\Delta\Sigma$ quantizer is -97 dBV, and hence for a 5 V supply voltage 108 dB SNR is within reach.

As a verification of the evaluations made above, one may compare them to the measurements made for the design presented in [35]. The bandwidth was 20 kHz and the resistors were designed with a nominal resistance of 1.8 kΩ. The PMOS current source was not switched, which implies that the gain factor (4.62) is doubled, and hence that $10\log_{10}(36/12)$ dB $= 4.8$ dB more noise than predicted by the above estimate is to be expected. Assuming that the full-scale output swing was defined as $\pm 5\text{V}_{\text{pp}}$ (for the differential circuit operating on a 5 V supply voltage), the above estimate predicts the SNR performance to be no better than 117 dB. The actual SNR performance measured was 113 dB. The small 4 dB difference is very realistic for a good design, considering that the measured SNR also includes flicker noise, quantization noise, clock-jitter-induced noise, etc., and that the current sources probably were not designed quite as aggressively as assumed in (4.60).

Overall Flicker Noise Performance. The flicker noise component is hard to estimate accurately, because it depends highly on the technology used for the implementation of the circuit. The following calculations are based on measurements[48] for a 3 μm technology, for which the gate-referred noise voltage is estimated to be approximately 300 [nV/$\sqrt{\text{Hz}}$]@10 Hz for a 1500 μm^2 transistor of either polarity. However, because flicker noise voltage is believed to be proportional to the gate oxide's thickness, less flicker noise can be expected when using a modern sub-micron technology.

Assuming that the lowest frequency of interest is f_{low}, the gate-referred flicker noise voltage for the

[48] Performed by Dr. Christian Enz and provided in his lecture notes from a short course on low-noise amplifiers at EPFL in Lausanne, Switzerland, August 1996.

considered technology can be calculated (for $\Delta f \gg f_{\text{low}}$) from

$$\begin{aligned} v_{\text{flicker,MOS}} &= \sqrt{\frac{(300 \cdot 10^{-9}\text{V})^2 1500\mu\text{m}^2}{W \cdot L} \int_{f_{\text{low}}}^{\Delta f} \frac{10}{f} df} \\ &= \sqrt{\frac{10(300 \cdot 10^{-9}\text{V})^2 1500\mu\text{m}^2}{W \cdot L} \ln\left(\frac{\Delta f}{f_{\text{low}}}\right)} \\ &\simeq 1\mu\text{V}\sqrt{\frac{1500\mu\text{m}^2}{W \cdot L}}\sqrt{\ln\left(\frac{\Delta f}{f_{\text{low}}}\right)} \end{aligned} \tag{4.69}$$

An opamps' input-referred flicker noise depends on the topology in which it is implemented; but assume (for simplicity) that four equally-sized (and equal-transconductance) transistors contribute to v_{p}. In this case, the opamps' input-referred flicker noise can be evaluated as

$$v_{\text{flicker,opamp,input}} \simeq 2.8\mu\text{V}\sqrt{\frac{1500\mu\text{m}^2}{W_{\text{opamp}} \cdot L_{\text{opamp}}}}\sqrt{\ln\left(\frac{\Delta f}{f_{\text{low}}}\right)} \tag{4.70}$$

The DAC's flicker noise contribution will be (cf. (4.64))

$$v_{\text{flicker,input,DAC}} \simeq 2.8\mu\text{V}\sqrt{\frac{1500\mu\text{m}^2}{W_{\text{DAC}} \cdot L_{\text{DAC}}}}\sqrt{\ln\left(\frac{\Delta f}{f_{\text{low}}}\right)} \tag{4.71}$$

To obtain the required degree of matching, the DAC's current sources will typically be implemented on a much larger chip area than that of the opamps' input differential pair; consequently, the overall flicker noise performance can be estimated roughly as

$$v_{\text{flicker,input}} \simeq v_{\text{flicker,input,opamp}} \simeq 2.8\mu\text{V}\sqrt{\frac{1500\mu\text{m}^2}{W_{\text{opamp}} \cdot L_{\text{opamp}}}}\sqrt{\ln\left(\frac{\Delta f}{f_{\text{low}}}\right)} \tag{4.72}$$

For $f_{\text{low}} = 10$ Hz, $\Delta f = 1$ MHz, and a transistor area of $1500\mu\text{m}^2$, the input-referred flicker noise will be approximately -100 dBV, which implies that the flicker noise (in this situation) is dominated by the thermal noise, even when the input resistor R is as small as 1 kΩ. Considering that the gate-oxide thickness (and hence the flicker-noise coefficient) is a decreasing function of the technology's minimum feature size, it follows that good flicker-noise performance can be obtained with a modern sub-micron technology, even when using transistors of a more moderate size (10 μm^2).

4.5.3 Conclusion

Continuous-time systems generally have a better thermal noise performance than discrete-time systems because the sampling also causes aliasing of the broadband noise, whereby the signal-band noise power increases. Quantizers always have a discrete-time (digital) output signal; but, as discussed above, the impact of noise aliasing depends highly on where the sampling is performed. CT $\Delta\Sigma$ quantizers sample the signal in the last stage of the (high-gain) closed-loop structure, and hence aliasing errors are efficiently suppressed. CT $\Delta\Sigma$ quantizers can be made so robust with respect to aliasing errors that not only noise aliasing but also signal aliasing is tolerable. In other words, a separate anti-aliasing filter is not required.

Because the noise performance of CT $\Delta\Sigma$ quantizers is limited mainly by the noise from the feedback DAC, and because they can be designed to have a low power consumption, these quantizers will probably dominate the market if and when (cf. Chapter 5) the industry learns to design highly-linear current-mode DACs.

This section has only considered device noise, i.e., thermal and flicker noise; but a different kind of noise – substrate noise – is usually the main problem in mixed-mode circuits, which become ever more important, and which usually require high-performance A/D and D/A converters in the analog portion of the circuit. In these circuits, high-resolution discrete-time analog signals simply cannot be allowed because the digital-switching noise's high-power high-frequency spectral components alias into the signal band in the sampling process. For the above reasons, CT $\Delta\Sigma$ quantizers may represent the only way by which high-resolution quantizers can be implemented.

Chapter 5

Improved Current-Mode DACs

In Chapter 4, it was discussed that CT $\Delta\Sigma$ quantizers have several advantages compared to DT $\Delta\Sigma$ quantizers in terms of speed, power consumption, and noise performance. However, as discussed in Section 3.2.2, dynamic errors from the feedback DAC is the main obstacle that prevents the successful design of high-performance CT $\Delta\Sigma$ quantizers. It is well understood that intersymbol-interference errors (cf. page 47) generally are the main source of distortion, but also that return-to-zero (RTZ) switching (cf. page 48) can suppress these errors to a very low level. Unfortunately, the (classic) RTZ switching scheme is associated with an increased sensitivity to clock-jitter-induced noise (cf. page 52), which makes it nearly impossible to implement commercial high-resolution wide-bandwidth DACs. This chapter will discuss techniques for the implementation of current-mode DACs, which are robust with respect to both intersymbol-interference and clock-jitter-induced errors. The best technique proposed will also avoid the timing and nonlinear-switching errors discussed on page 46.

5.1 Dual Return-to-Zero Current-Mode DAC

The RTZ switching scheme represents a good school of thought in that it relies neither on matching of nor the absolute value of electrical parameters. However, it does rely on the ability to accurately

reproduce the exact same waveform, and this imposes the discussed problem of its high sensitivity to clock jitter.

To ease the sensitivity to clock jitter while preserving the good dynamic linearity of RTZ DACs, Adams [35] has proposed that a current-mode DAC be implemented using two RTZ[1] current-mode (sub) DACs operated time-interleaved as shown in Figure 5.1. Such DACs will be called Dual-RTZ DACs. Assuming static linearity, the RTZ switching assures that each sub DAC's operation is accurately described by an impulse response. Because $f(t) = f_1(t) + f_2(t)$, the composite DAC will also be described by an impulse response[2]; consequently, it will be linear. Uncertainty and mutual mismatch of the sub DACs' impulse responses[3] is acceptable, because it will cause neither static nor dynamic nonlinearity. The main advantage of the system is that *if the two DACs are clocked by the same clock signal*[4], the DAC's clock-jitter-induced error can be expressed as (cf. Equation (3.26) and Footnote 9 on page 50):

$$f_{\text{jitter,error}}(t) = K_{\text{DAC}} \sum_{k=-\infty}^{\infty} [d(k) - d(k-1)]\Delta T(k)\delta(t - kT_s) \tag{5.1}$$

which, for a multi-bit signal $d(k)$, is as good as what can be obtained when using a discrete-time voltage-mode DAC. Thus, the clock-jitter sensitivity of multi-bit Dual-RTZ DACs operating with a minimum degree of oversampling will be modest (cf. Section 3.2.3); consequently, they will be suitable for use in commercial products.

The error included in $f(t)$ will be a linear combination[5] of the errors[6] included in $f_1(k)$ and $f_2(k)$, so, if the sub DACs are individually mismatch-shaping, the Dual-RTZ DAC will be mismatch-shaping as well. The feasibility and effectiveness of this technique is verified by a state-of-the-art design [35], which is also described in more detail in [57].

Although the performance reported in [35] is outstanding, it should be understood that it is not easily obtained. Each of the sub DACs consist of several current sources, and the timing of the signals control-

[1] Each operating with a 50% duty-cycle.

[2] The duration of the combined impulse response will typically slightly exceed the sampling period.

[3] Including, but not limited to, mismatch of their linear-characteristic gain.

[4] In which case, the two sub DACs are subject to the same clock jitter signal $\Delta T(k)$.

[5] Approximately the average value of the two error signals.

[6] Each error is defined with respect to the linear characteristic of the respective sub DAC.

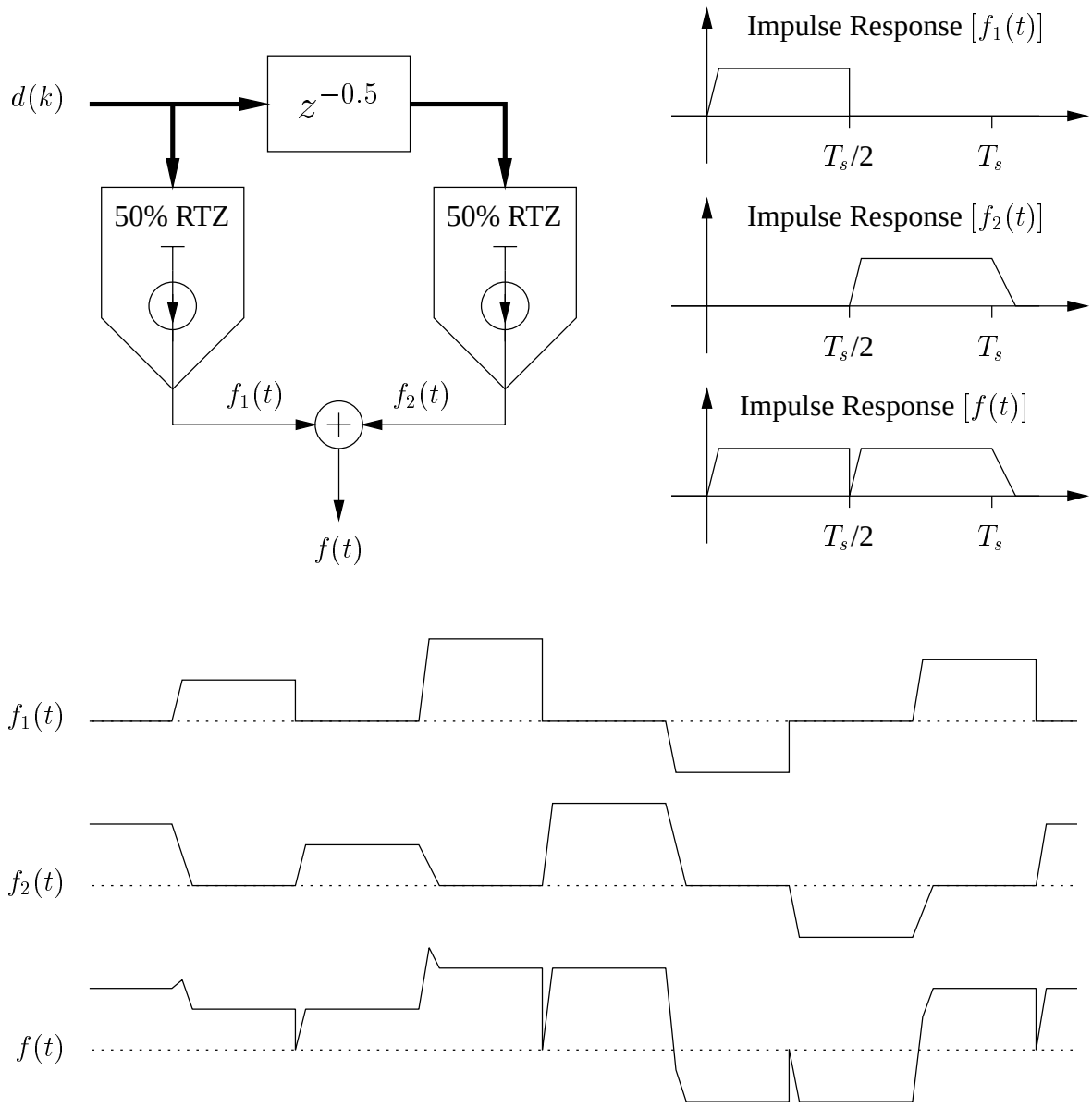


Figure 5.1: Linearization using two time-interleaved RTZ current-mode DACs.

ling these many current sources must be synchronized not only within each sub DAC, but also mutually between the two sub DACs. In other words, the success of the design depends significantly on the experience of the actual designer(s) and layout person(s), because timing errors and nonlinear switching errors are still major concerns (cf. page 46).

5.1.1 A Variation

The fundamental element in the RTZ switching scheme is that no current source is used in any two samples without being brought to a fixed status (the "zero" in "return-to-zero") in between. When used in combination with mismatch-shaping techniques, reduced clock-jitter sensitivity can be obtained without using the Dual-RTZ switching scheme.

First notice that the fundamental property of the element-rotation mismatch-shaping encoder is that, with respect to *any* origin in time, no unit element is used twice before the other unit elements have been used in between. Hence, if an ERS UE-MS DAC is implemented with twice as many unit-element current sources as the maximum code to be D/A converted, it follows that the encoder will automatically perform RTZ switching[7]. The advantage of this approach in comparison to the Dual-RTZ switching scheme is that the currents are switched at half the frequency, therefore, timing errors and nonlinear switching errors are suppressed by 3 dB.

5.2 Time-Interleaved Current-Mode DAC

This section will describe a simple technique for the implementation of current-mode DACs with a very good dynamic linearity/performance. As for Dual-RTZ DACs, the output current is generated by switching between two (or more) DACs, but dynamic linearity is obtained using another technique not employing RTZ switching.

Dynamic nonlinearity is avoided by operating two DACs time interleaved, and by connecting the DACs to the output only after they have settled from being updated. Thereby, all switching effects become

[7] It is assumed that each unit-element current source operates with a 100% duty cycle.

invisible seen from the output terminal. The switching between the two DACs is performed at one centralized node, whereby the DAC becomes as insensitive to clock-jitter as Dual-RTZ DACs, but it is also insensitive to nonlinear-switching, intersymbol-interference, and timing errors. Hence, the proposed technique is useful not only for the implementation of high-performance oversampled data converters, but also for high-speed DACs with bandwidths of, say, 100 MHz or more.

5.2.1 Basic Topology and Operation

Figure 5.2 shows the basic implementation of the proposed DAC structure. DAC 1 receives the new input value $d(k)$ slightly after the onset of Φ_2, whereas DAC 2 receives it half a sampling period later (slightly after the onset of Φ_1). The switching block consisting of the four PMOS transistors is controlled such that the output $f_1(t)$ from DAC 1 is connected to the output, i.e., $f(t) = f_1(t)$ in clock phases Φ_1, whereas the output $f_2(t)$ from DAC 2 is connected to the output, i.e., $f(t) = f_2(t)$ in clock phases Φ_2.

The point is that when the DACs are updated, the respective DAC is connected to a through-away point (i.e., the current is dumped to an arbitrary low-impedance terminal, preferably having the same virtual-ground potential as the load terminal), and the potential switching errors – including timing, intersymbol-interference, and nonlinear-switching errors – will not affect the output signal $f(t)$ in any way. The only requirement is that when the DACs are updated, they must settle to their static value within approximately half a sampling period, at which point they are connected to the output terminal.

The four PMOS transistors constituting the switching block may either be operated as CMOS switches, in which case the two active transistors operate in their triode region, or as hard-driven differential pairs, in which case the two active transistors operate in their saturation region. Either way, it is imperative that the differential driver is implemented such that the switching block implements a make-before-break switching function. Such differential drivers are well known and widely used (see [58]). For the implementation of high-speed DACs, the switching block may be driven such that all four transistors always are active and operate in saturation, and such that only one transistor of each differential pair carries the majority (say 99%) of the tail current $f_x(t)$.

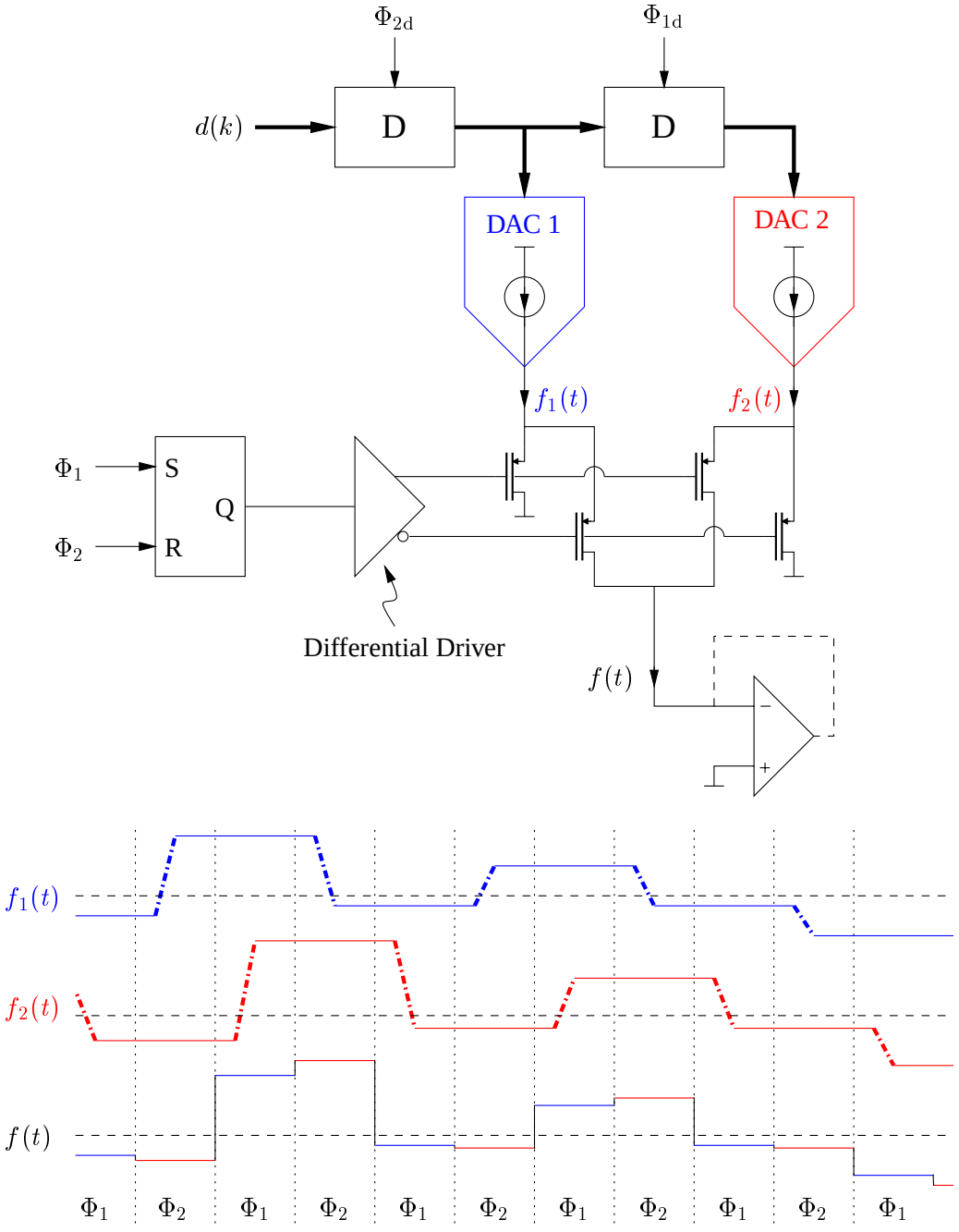


Figure 5.2: Basic topology of the time-interleaved current-mode DAC.

5.2.2 Analysis

Whereas deleterious dynamic effects associated with the update of each DAC are efficiently prevented, the switching block can potentially cause switching errors. The situation is, however, much improved compared to stand-alone current-mode DACs as well as Dual-RTZ DACs.

Timing errors will obviously not be a problem. The mere fact that the switching function is centralized assures that all current sources in each DAC are updated at exactly the same instance with respect to $f(t)$. Glitches may occur in the transition from one clock phase to the next, but assuming that the differential driver provides a consistent switching waveform[8], these glitches will be *linearly* related to the input signal $d(k)$, and hence they do not pose a problem.

Nonlinear-switching errors[9] can occur if the switching waveform provided by the differential driver is not consistent. Consistency is, however, reasonably simple to obtain because it refers to only one signal, and any reminiscence of the potential nonlinear error can be suppressed efficiently by making the differential driver fast. Notice that only the differential driver and the switching block "need" to be fast; the two sub DACs may be significantly slower (as shown in Figure 5.2), and that is a property which can be used to minimize the digital circuit elements' power consumption (of concern in high-speed DACs).

Charge injection can sometimes cause nonlinear-switching errors, but this implementation is indeed very robust with respect to charge-injection errors. If the active PMOS transistors are operated in saturation, the charge-injection signal will be proportional to the switched signal, because the inversion charge of a saturated MOSFET is proportional to the conducted current, and hence it is a linear effect. On the other hand, if the active PMOS transistors are operated in the triode region, the charge-injection signal will be independent of the switched signal because the opamp provides a virtual ground, and hence the

[8] Any waveform is acceptable because linearity is dependent on repeatability only. "Switching waveform" refers to the provided waveform locally around (i.e. from shortly before to shortly after) the switching instances, and hence it is independent of clock jitter. It is acceptable if the switching waveform $\Phi_1 \rightarrow \Phi_2$ differs from the switching waveform $\Phi_2 \rightarrow \Phi_1$, as long as they are consistent individually.

[9] This type of error should more precisely be considered to be noise rather than nonlinear errors. It can be assured that the differential driver is not affected by the signal $d(k)$, and hence the error will not be deterministically related to the signal (which is the fundamental property of nonlinear errors).

gate-to-channel potential will be constant. This is also a linear effect.

Conclusion. The proposed time-interleaved DACs are characterized by the same advantages as the Dual-RTZ DACs, namely reduced clock-jitter sensitivity, insensitivity to mismatch of the sub DACs' gain and offset, and that a time-interleaved DAC will be mismatch-shaping if the sub DACs are individually mismatch-shaping[10]. However, as discussed above, time-interleaved DACs are also characterized by an improved immunity to timing errors as well as nonlinear-switching errors, and hence they are much less critical to implement, compared to Dual-RTZ DACs, for example. The improved robustness can be used constructively to reduce the overall chip area, or to employ more modular layout techniques, which potentially can help reduce a product's turn-around time.

In conclusion, time-interleaved current-mode DACs are very suitable for use in commercial products, whether they are high-speed, high-performance, or short-design-time applications.

5.3 Conclusion

The successful implementation [35] of a mismatch-shaping Dual-RTZ DAC has verified that it is feasible to implement current-mode DACs with a very good dynamic performance. The proposed time-interleaved DACs represent an improved concept, which combines the advantages of Dual-RTZ DACs with insensitivity to timing errors, thereby providing an even more robust architecture for the implementation of current-mode DACs.

[10]The property that the time-interleaved DAC's error signal is the average value of the sub DACs' error signals implies that the requirement of two (or more) sub DACs need not imply that the time-interleaved DAC will require twice the chip area for its implementation (compared to a stand-alone DAC of the same static linearity). A DAC will occupy a minimum area $\mathcal{A}$ which cannot be reduced because the area is related to the stochastic element in the technology's matching properties. To avoid systematic errors, the minimum area $\mathcal{A}$ is often divided into (say) two areas $\mathcal{A}_1$ and $\mathcal{A}_2$ of equal size, and chosen according to the common-centroid layout principle (cf. page 59). It is easily identified that if DAC 1 is laid out on $\mathcal{A}_1$, and DAC 2 is laid out (symmetrically) on $\mathcal{A}_2$, the composite DAC will fulfill the common-centroid principle even if the individual DACs do not. In other words, in terms of stochastic properties, time averaging is equivalent to coordinate averaging, and hence the total area $\mathcal{A} = \mathcal{A}_1 \cup \mathcal{A}_2$ can be the same as if only one DAC had been implemented.

High-performance current-mode DACs are, in general, more difficult to implement than discrete-time voltage-mode DACs. However, in critical design situations (low-power, low-noise, or high-speed) the extra difficulty may well be worth the trouble because the achievable improvement in performance can be as much as a factor of 10.

Chapter 6

Dithering of Mismatch-Shaping DACs

Unit-element mismatch-shaping (UE-MS) DACs are by themselves important circuits, and they are also important for the implementation of scaled-element mismatch-shaping DACs which are discussed in Chapter 7. Unfortunately (as discussed in Section 4.4.3), UE-MS DACs have a tendency to produce idle tones[1] that can ruin the performance of an otherwise well designed data converter [51].

Fortunately, idle tones can be prevented using dithering techniques. This chapter will discuss and analyze a prior-art dithering technique dedicated for tree-structure UE-MS encoders, and two novel dithering techniques for use with the simpler ERS encoders will be proposed. As discussed in Section 4.4.5, first-order UE-MS DACs are usually preferable[2] in comparison to higher-order UE-MS DACs; hence, the discussion will center around only first-order encoders.

6.1 Idle Tones in Deterministic UE-MS Encoders

Idle tones in mismatch-shaping DACs are caused by periodic/systematic use of the unit elements. The following discussion will consider the idle-tone behavior of deterministic UE-MS encoders, i.e., en-

[1] Spurious tones caused by periodic use/shuffling of the unit elements.

[2] In terms of simplicity, but also in terms of performance for OSR less than 25.

coders in which the unit elements for the conversion of $d(k)$ are selected as a function of only the signal $d(k)$ and an initial condition. These encoders represent the majority of the published/known encoders.

6.1.1 Idle Tones in ERS UE-MS Encoders

Element-Rotation-Scheme (ERS) UE-MS encoders select the unit elements for the D/A conversion of $d(k)$ according to the very simple rotation rule described on page 103, and hence they have a tendency to produce idle tones (cf. page 105). The following discussion refers to Figures 4.9 and 4.10.

Assume that the input $d(k)$ is periodic with the minimum period P. A certain number

$$Q = \sum_{i=1}^{P} d(i) \tag{6.1}$$

of elements will be used for the conversion of one period of $d(k)$.

Considering the nature of the ERS algorithm, it follows that the conversion of one period of $d(k)$ is characterized by a certain increment (modulo N, where N is the number of unit elements) of the rotation pointer $r(k)$

$$r(k+P) = \text{modulo}[r(k) + Q, N] \tag{6.2}$$

The rotation pointer $r(k)$ will be periodic signal with a period of NP samples (and possibly also with a shorter period)

$$r(k+NP) = \text{modulo}[r(k) + NQ, N] = \text{modulo}[r(k), N] = r(k) \tag{6.3}$$

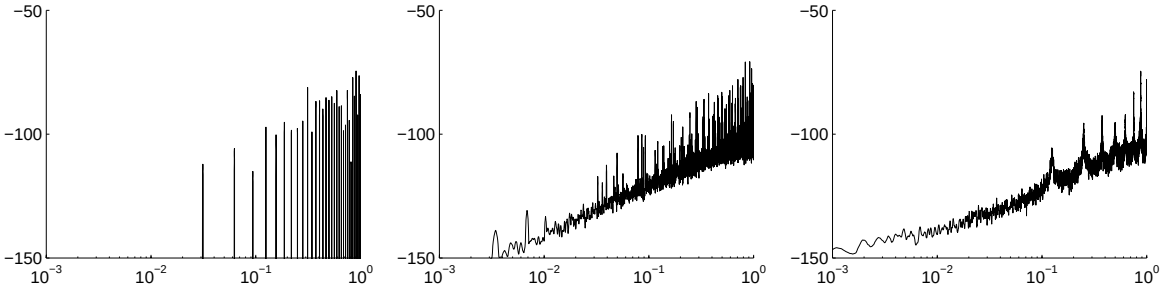
Because the signal $d(k)$ and the rotation pointer $r(k)$ are periodic with the same period NP, the use of the unit elements (and hence the error signal $m(k)$) will be periodic with the period NP as well. Consequently, the error signal $m(k)$ will consist of *only* idle tones, which are located at integer multiples of the frequency $f_s/(NP)$.

Performance Evaluation. The above analysis indicates that the ERS encoder is indeed very likely to produce idle tones, and that there is no reason to believe that they will not appear in the signal band.

The assumption made that $d(k)$ is periodic does, however, not represent the typical case. Almost always, $d(k)$ will be the output of a $\Delta\Sigma$ quantizer/modulator, and hence $d(k)$ will *not* be periodic because it includes the quantizer/modulator's truncation error signal, which is an Autoregressive Moving-Average (ARMA) pseudo-stochastic process (popularly called "shaped noise"). The included ARMA pseudo-stochastic signal will generally improve the idle tone behavior, but because it has only little power in the baseband, the maximum-likelihood estimator of $r(k + PN)$ is still described by Equation (6.3) (now with a nonzero standard deviation). Hence, intuitively, the power of the persistent idle tones in the UE-MS DAC's error signal will (mainly) be smeared locally.

Figure 6.1 shows three examples of the error signal generated by a 16-element UE-MS DAC driven by an ERS encoder. The input signal $d(k)$ was in all three cases generated from an 8-times oversampled sinusoid.

Figure 6.1a shows the result when $d(k)$ is generated by simple truncation of the sinusoid, in which case $d(k)$ is periodic with period 16. As expected, idle tones are observed at frequencies that are integer multiples of $4\frac{f_s}{16\cdot 16} = \frac{f_s}{64}$, where the factor of 4 reflects that, for the parameters used in this simulation, modulo$[Q, N] = 4$ (cf. Equation (6.2)). Clearly, idle tones are indeed a problem that need to be dealt with.



(a) Periodic (b) $\Delta\Sigma$-driven (c) ARMA-stochastic dither

Figure 6.1: Spectral power density (DFT) of the error signal from a 16-element ERS UE-MS DAC. The units are dBFS versus frequency normalized to the Nyquist frequency.

The high-resolution input sinusoid can, of course, only be represented accurately if the 4-bit truncation is performed by a $\Delta\Sigma$ quantizer/modulator, in which case the error signal will be as shown in Figure 6.1b. It can be observed that the ARMA pseudo-stochastic truncation error breaks the periodicity

and thereby spreads the error signal's power to, presumably, all frequencies, but idle tones are still a major problem. Notice that the ARMA pseudo-stochastic truncation error does *not* spread the error signal's spectral power density as predicted above, but that idle tones rather appear at discrete frequencies in an unpredictable pattern. The discrepancy reflects the "pseudo" in the ARMA pseudo-stochastic truncation error, i.e., that the truncation error is *correlated* with the sinusoid, and hence does not originate from a truly stochastic process. To support and investigate the validity this explanation, a third simulation was performed where $d(k)$ was generated by truncating the sinusoid (as in Figure 6.1a) and adding to it the output from an *independent* $\Delta\Sigma$ modulator with zero input achieving a realistic non-pseudo ARMA stochastic process. The spectral power density of the error signal produced in this way is shown in Figure 6.1(c). Now, the expected local spreading of the persistent idle tones can be observed.

Conclusion. Figure 6.1b shows the spectral power density of a typical-case error signal $m(k)$ generated by a UE-MS DAC driven by an ERS encoder (Figures 6.1a and 6.1c do not correspond to useful systems). Strong idle tones are observed, and they should be expected whenever the simple ERS encoder is used. The performance is unacceptable for a wide range of applications, and hence better UE-MS encoders are needed.

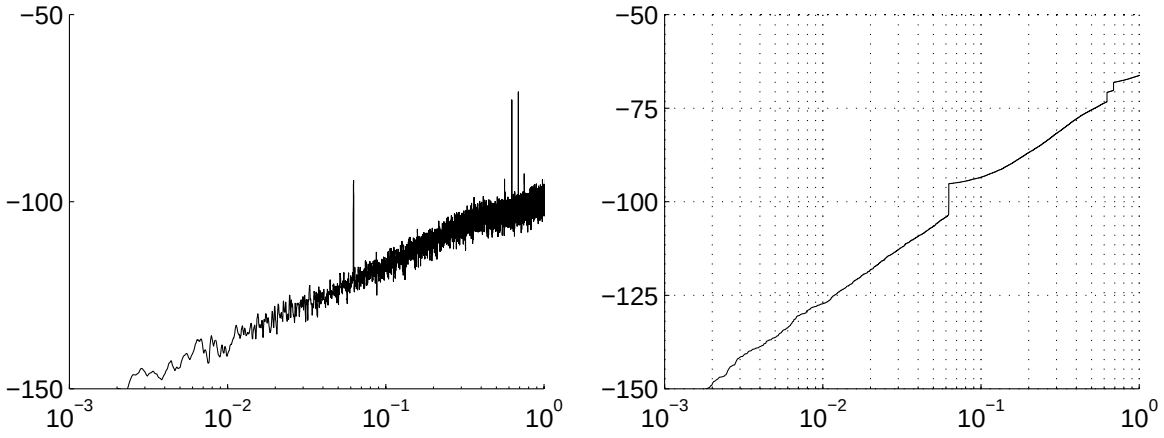
6.1.2 Idle Tones in Complex UE-MS Encoders

It is often claimed – and there is some truth to it – that the idle-tone behavior can be improved by selecting the unit elements in a pattern which is more "complex" than the simple rotation pattern used by the ERS encoders. Several techniques have been proposed [9] [14] [39], but neither of them is actually recommendable unless a stochastic element[3] (dither) is appropriately incorporated. Indeed, idle tones are less likely to occur when using these encoders, but as long as the encoders are deterministic digital state machines, idle tones can (and usually do) exist. The problem is that the idle tones may be so hard to find (by means of time-consuming and therefore sparse simulations) that the designer is

[3]Truly stochastic processes cannot be implemented using only digital circuitry, but a pseudo-random process is in general sufficient for all practical purposes. By assuring that the pseudo-random sequence has a sufficiently long (minimum) period, the resultant idle tones can be made arbitrarily dense in frequency.

inclined to assume that they will not occur. Unfortunately, idle tones tend to reveal themselves primarily after the circuit is implemented (under testing), at which point a redesign of the circuit may prove to be necessary. Even *very* experienced designers have been deceived by "hiding" idle tones, and hence it is usually worthwhile to eventually overdesign the circuit and thereby avoid the extra cost and time which is associated with a redesign.

Figure 6.2 shows a signal-band idle tone which has been observed in a 16-element tree-structure UE-MS DAC (cf. page 115).



(a) Spectral power density versus frequency

(b) Signal-band power versus 1/OSR

Figure 6.2: Signal-band idle tone observed in a tree-structure first-order UE-MS DAC

6.2 Dithered UE-MS Encoders

Avoiding idle tones is essentially a matter of avoiding patterns in how the unit elements are selected. A good technique to prevent patterns is to use an encoder that encounters equilibrium states, in which the unit elements for the conversion of the next sample can be selected in two or more (almost) equally good ways, and to let a stochastic process choose how to proceed. Idle tones will be efficiently prevented if such equilibrium states are reached fairly often, and if the future selection of the unit elements will differ substantially depending on the outcome of the stochastic process. Such techniques will be called *dithering* techniques.

6.2.1 Dithered Tree-Structure UE-MS Encoders

Tree-structure UE-MS encoders were discussed in Section 4.4.4, and a specific first-order implementation was discussed on page 115. The overall operation relies on the operation of the node separators, which must each generate a signal $t(k)$ of values $\{-1, 0, 1\}$ such that the sum $|\sum_{i=1}^{k} t(i)|$ is bounded/minimized for all k. Whenever a node separator receives an odd signal, $t(k)$ must be chosen as ± 1, otherwise as 0. Obviously, $t(k)$ should always be chosen of the opposite polarity as that of $\sum_{i=1}^{k-1} t(i)$; but when the sum is zero, any choice may be equally good (i.e., its an equilibrium state). Hence, dithering a tree-structure UE-MS encoder is a matter of, in all the node separators[4], *randomly* choosing the polarity of $t(k)$ when $\sum_{i=1}^{k-1} t(i) = 0$.

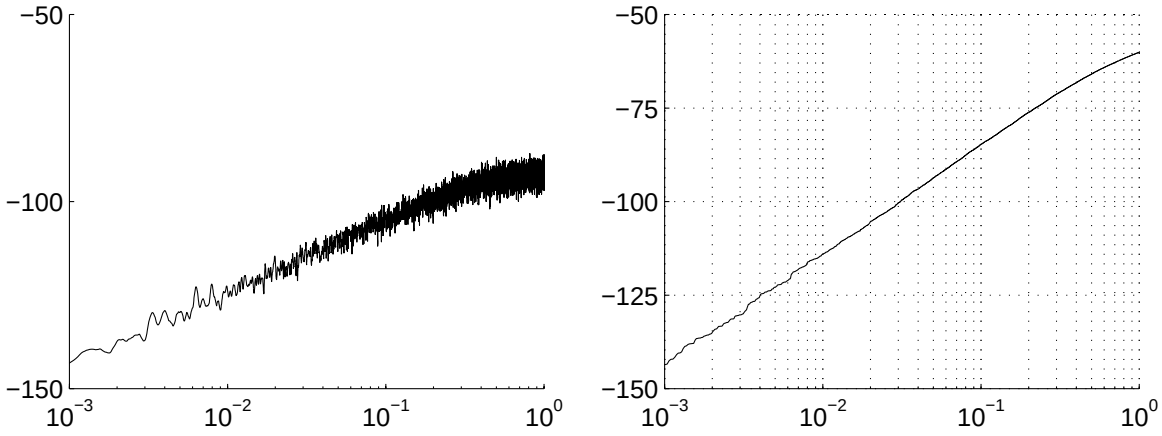
Figure 6.3 shows the characteristics of the error signal that was generated by a 16-element fully-dithered tree-structure first-order UE-MS DAC when the input signal $d(k)$ is a -12 dBFS 64-times oversampled $\Delta\Sigma$-quantized sinusoid.

The dithering technique (which was first proposed in [14]) appears to be very efficient, but in Figure 6.2a it is observed that the error signal's spectral power density flattens around the Nyquist frequency. This is not a major problem, but it does imply that the signal-band performance is less than what it perhaps could be (compare Figures 6.2b and 4.17). The slight degradation is caused by a combination of two effects:

1. In the dithered tree-structure encoder, $\sum_{i=1}^{k} t(i)$ will oscillate between the three values {-1,0,1} instead of only the two values used by the deterministic encoder. Because the frequency of $|t(k)| = 1$ events is the same, the dithered encoder will need more samples to correct for previous errors, and so it will not perform as well at high frequencies.

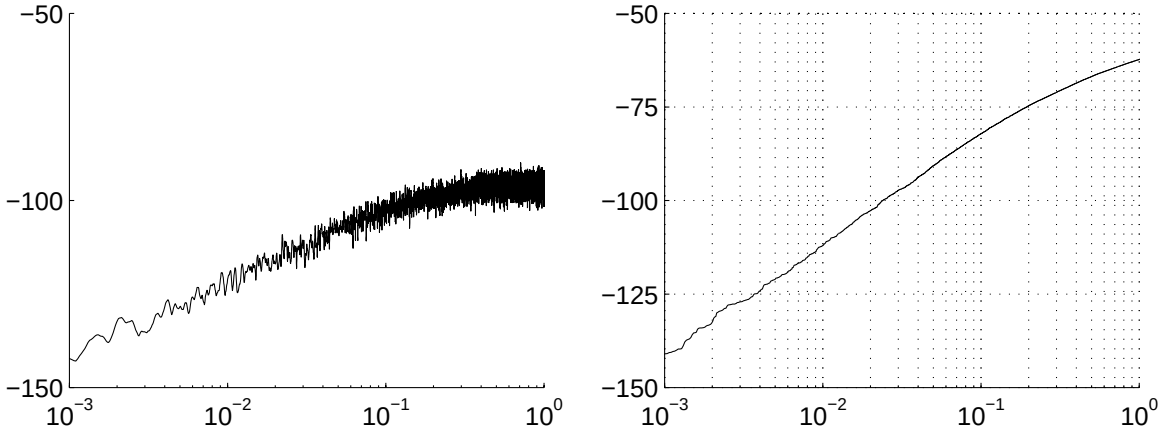
2. Midscale input $d(k)$ (i.e., when half the unit elements are selected in each sample) is the optimum operating condition for the encoder. This is because midscale input corresponds to the highest

[4]To efficiently avoid idle tones, a separate random bit should be generated for *each* node separator because idle tones have been observed when the same stochastic process is shared by all the node separators. This observation supports the second rule for how to obtain efficient dithering (cf. the introduction to this chapter), namely that the stochastic process must be allowed to choose among *substantially* different patterns in the use of the unit elements.



(a) Spectral power density versus frequency (b) Signal-band power versus 1/OSR

Figure 6.3: Error signal from a dithered tree-structure UE-MS DAC. The input signal $d(k)$ was a -12 dBFS 64-times oversampled $\Delta\Sigma$-quantized sinusoid.



(a) Spectral power density versus frequency (b) Signal-band power versus 1/OSR

Figure 6.4: Error signal from a dithered tree-structure UE-MS DAC. The input signal $d(k)$ was a -1.5 dBFS 64-times oversampled $\Delta\Sigma$-quantized sinusoid.

probability ($p = 0.5$) for $|t(k)| = 1$ events, which is proportional to the closed-loop's feedback factor and thereby also the error signal's controllability (cf. the discussion on page 115). If the input signal includes long sequences for which nearly all/no unit elements are used, the error signal's controllability will deteriorate (in those periods), and hence the encoder's high-frequency performance will be degraded. Such sequences will be called *low-control* sequences. Low-control sequences are, e.g., a characteristic of highly-oversampled full-scale sinusoid input signals. This effect is dominating in Figure 6.4, where the magnitude of the $\Delta\Sigma$-quantized sinusoid input signal is increased to -1.5 dB full scale.

Low-Control Sequences. The following will study the effect of low-control sequences. For simplicity, $\widetilde{d(k)}$ is defined as $d(k)$ scaled such that a full-scale $\widetilde{d(k)}$ spans the range from -1 to $+1$.

For a constant input $d(k)$, each node separator in a tree-structure UE-MS encoder will (for each sample) have a constant probability $p = \frac{1-|\widetilde{d(k)}|}{2}$ for $|t(k)| = 1$. The probability p is proportional to the control system's loop gain (cf. Figure 4.15 with a first-order loop filter), and hence the transfer function of the filtering process to which the error signal is subject can be estimated as (for some empiric constant $a \in R$)

$$H_{\mathrm{MS}}(f) = \frac{1}{1 + ap\frac{z^{-1}}{1+z^{-1}}} = \frac{1 - z^{-1}}{1 + (ap-1)z^{-1}} \quad \text{where} \quad z = e^{j2\pi f/f_s} \tag{6.4}$$

In Equation 4.22 it was estimated that the error signal's power density is related to $\widetilde{d(k)}$ as $\sqrt{p(1-p)}$, and hence – for all constant inputs $d(k)$ – the spectral power density can presumably be expressed as

$$\mathrm{PSD}(f) \propto \frac{1 - z^{-1}}{1 + (ap-1)z^{-1}}\sqrt{p(1-p)} \quad \text{where} \quad z = e^{j2\pi f/f_s} \tag{6.5}$$

To investigate the validity and accuracy of Equation (6.5), a fully-dithered 16-element tree-structure UE-MS DAC was simulated for 8 constant inputs, for which $8, 9, 10, \ldots, 15$ unit elements were used in average[5]. The signal-band power of the obtained error signals versus $1/\mathrm{OSR}$ is shown in Figure 6.5

[5] As usual, the UE-MS DAC was driven by a $\Delta\Sigma$ modulator, and hence the neighbor values were used in a small fraction of the time. This was mainly done to conform with the subsequent simulations, where a small amount of noise is required.

(solid lines), and the corresponding estimates obtained by integration of PSD(f) (for $a = 2$) are shown with dashed lines. The correspondence is reasonably good.

In particular, it can be observed that when low-control sequences are encountered (i.e., when p decreases), the signal-band suppression $H_{\mathrm{MS}}(f)$ becomes less efficient *and* the error signal's total power is decreased. When the OSR is around 30, the two effects somewhat cancel, and an only minor deterioration of the performance is encountered even when p becomes quite small. This relationship can also be observed in the simulation results shown in Figures 6.3 and 6.4.

6.2.2 Dithered ERS UE-MS Encoders

Dithered tree-structure UE-MS encoders offer efficient mismatch shaping without producing idle tones. The circuit complexity is, however, fairly high, and the technique is commercially protected by a U.S. patent[6]. It was, therefore, found to be worthwhile to develop alternative dithering techniques which can be used also for the simpler ERS encoders (which possibly are the simplest known UE-MS endoders).

To obtain dithering, it is necessary to allow the encoder to choose among two or more ways to select the unit elements. Williams has proposed a technique where the unit elements are selected by only one of several alternating ERS encoders [39]. When using his technique, the ERS encoders are activated as a nonlinear function of the input signal $d(k)$, but that will not guarantee the absence of idle tones (cf. Footnote 27 on page 106). Instead, to obtain efficient dithering of the composite encoder, a stochastic process can be used to randomly choose which encoder is used in which samples. The principle is shown schematically in Figure 6.6 where only two ERS encoders are employed[7]. This dithering technique will be called *dual-ERS* dithering (or, more generally, for *dual-encoder* dithering if the encoders are not necessarily ERS encoders). The underlying mechanics of dual-encoder dithering is that the stochastic process assures that the input to each of the (sub) encoders will be free of patterns and of uniform spectral

[6] U.S. patent 5,684,482, filed March 6, 1996, issued November 4, 1997. A PCT application does not appear to have been filed, but other patents may be pending.

[7] Obviously, this technique will work with any type and any number of UE-MS encoders. Two ERS encoders are shown because this probably is the simplest implementation, yet effective to prevent idle tones.

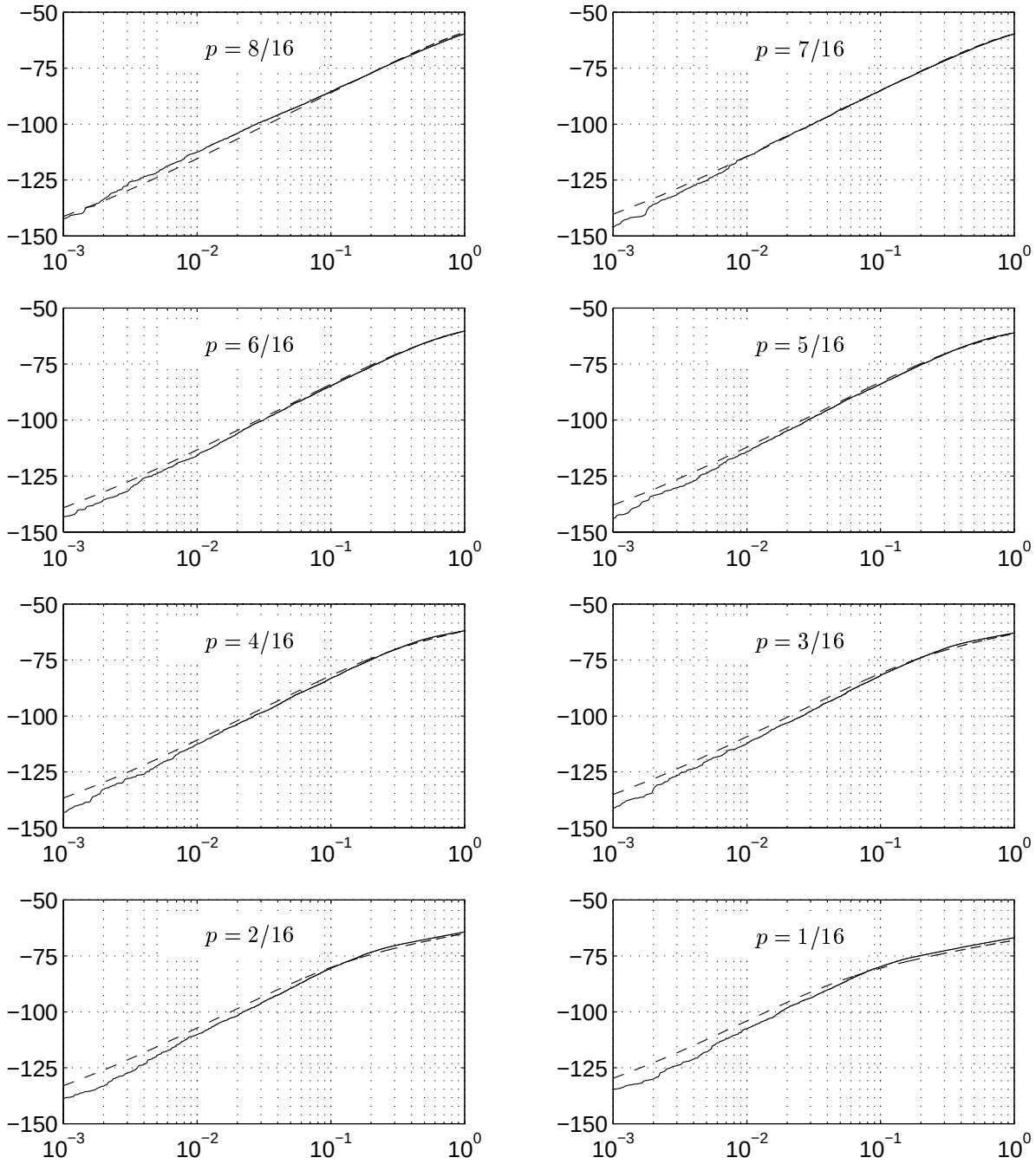


Figure 6.5: Static performance of a dithered tree-structure UE-MS encoder. The plots show the signal-band power versus 1/OSR for each constant value of $d(k)$.

power density. It is simple to show that the mismatch-shaping property is preserved (provided that only one set of unit elements is used).

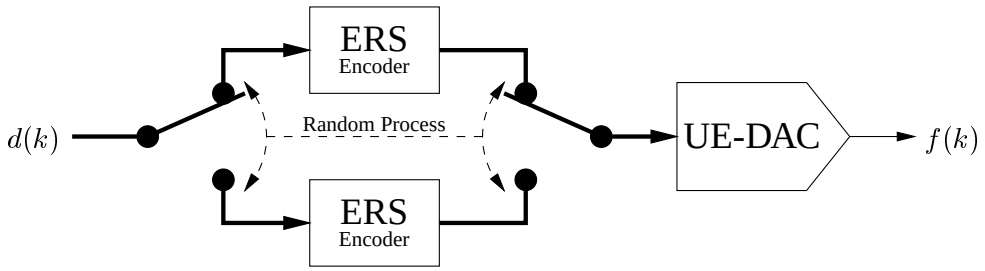


Figure 6.6: Basic principle for a dithered ERS encoder

Simulation Results and Evaluation. The dynamic performance of a dual-ERS dithered UE-MS encoder is illustrated in part by Figures 6.7 and 6.8, which correspond to Figures 6.3 and 6.4, respectively, in that they are based on the same operating conditions. The performance is quite similar to that of fully dithered tree-structure UE-MS encoders.

The mismatch-shaping property of ERS encoders is obtained because the error cancels every time the rotation pointer wraps around (i.e., when a full rotation is completed, cf. Figure 4.9). When the input $d(k)$ is shared by two encoders, wrap-around events will occur at only half the frequency, and hence the high-frequency performance is degraded as the number of sub encoders is increased (two is the optimum). This effect is quite similar to the first degrading effect for dithered tree-structure encoders (described on page 148). The other degrading effect for dithered tree-structure encoders (described on page 148) also has its equivalent for dual-ERS dithered encoders. When nearly all/no unit elements are used for the conversion of each sample, the two rotation pointers will rotate (backwards/forwards) only relatively slowly, whereby the error-cancellation process is slowed and the high-frequency performance degraded. To investigate this effect, a series of simulations with constant input $d(k)$ was performed (equivalent to the series of simulations presented in Figure 6.5). The quite promising[8] results are shown in Figure 6.9. It appears that, for some input signals ($9/16$, $13/16$, $14/16$, $15/16$), there is (relatively-

[8]When evaluated with respect to the OSR required to suppress the error signal's signal-band power to -100 dBFS, the performance is slightly better than that of dithered tree-structure encoders.

speaking) a slightly increased spectral power density around the frequencies $f_s/16$ and/or $f_s/32$. These frequencies correspond to the average period for the rotation pointers, and the locally increased spectral power density can be interpreted as spread idle tones. These idle tones are well-spread (dithered) and they are related only to the number N of unit elements and to the number of encoders (two), and hence they cannot move to low frequencies ($f_s/2N = f_s/32$ is the lowest possible frequency). Thus, if the signal-band does not include these well-defined frequencies, the spread idle tones do not pose any threat of deteriorating the system's performance. In essence, the phenomenon is due to low number[9] (N) of combinations by which any given code can be converted, which is directly related to the implementation's simplicity. The performance can possibly to be improved if the unit elements are ordered differently depending on which of the two encoders is activated (it has not been tested yet).

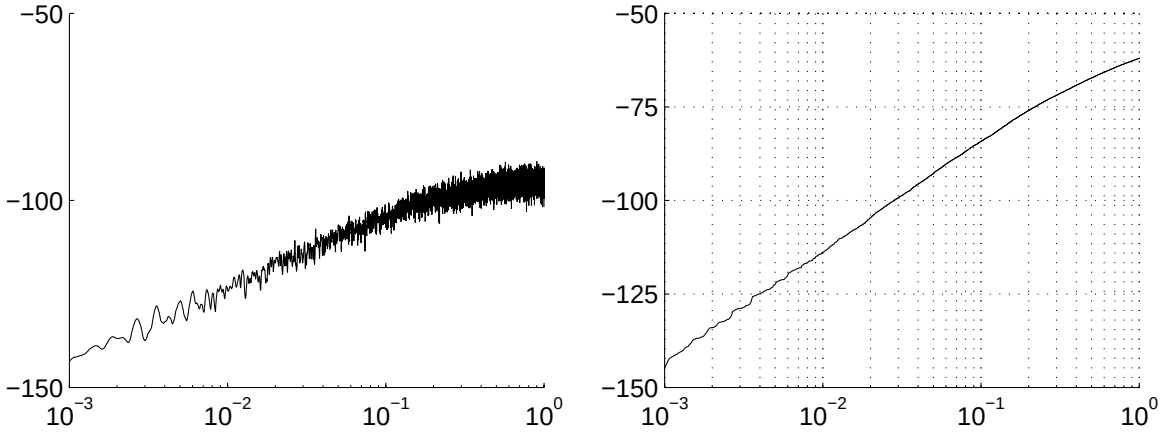
Proposed Implementation. Figure 6.10 shows a simple implementation where the two ERS encoders share most of the hardware; only the distinctive elements (namely the management of the rotation pointers) are implemented separately. A single-bit pseudo-random signal Φ_{rd} is generated to select which of the two rotation pointers is to be used for the encoding of each sample of $d(k)$, and the rotation pointers are updated accordingly.

When comparing this implementation to the usual implementation of ERS encoders (cf. Figure 4.10), it is observed that the extra hardware required to implement dual-ERS dithering is of negligible complexity. In conclusion, the overall complexity of the proposed dual-ERS dithered encoders is significantly less than that of dithered tree-structure encoders, and they yield essentially the same performance.

6.3 Random-Orientation Dithered ERS Encoder

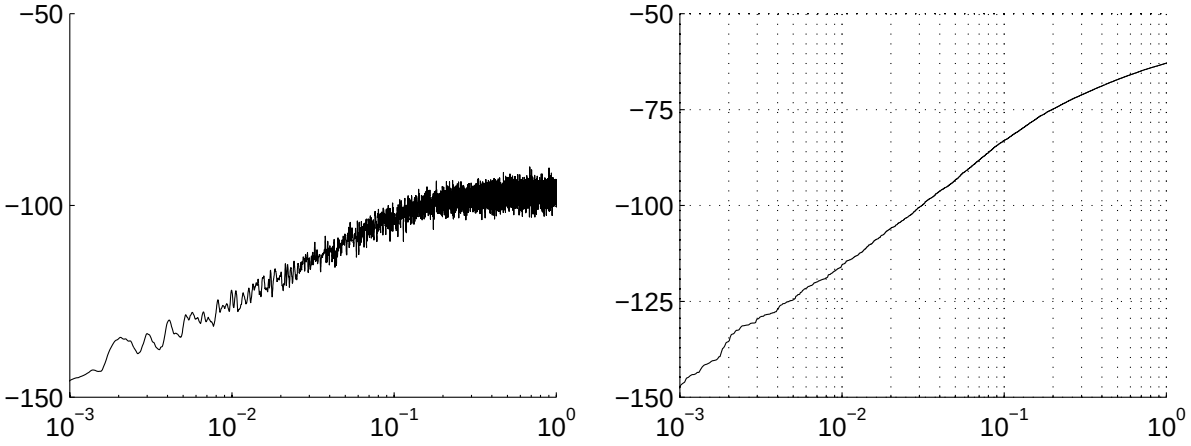
The dithering of tree-structure encoders as well as dual-ERS encoders has the side effect of decreasing the frequency by which the errors are canceled. In an attempt to design a dithered encoder with a better high-frequency performance, the random-orientation (RO) dithering technique described below was developed. The result is an ERS encoder for which the errors are canceled with the same (high)

[9] Compared to tree-structure encoders, for example.



(a) Spectral power density versus frequency (b) Signal-band power versus 1/OSR

Figure 6.7: Error signal from a dual-ERS dithered UE-MS DAC. The input signal $d(k)$ was a -12 dBFS 64-times oversampled $\Delta\Sigma$-quantized sinusoid.



(a) Spectral power density versus frequency (b) Signal-band power versus 1/OSR

Figure 6.8: Error signal from a dual-ERS dithered UE-MS DAC. The input signal $d(k)$ was a -1.5 dBFS 64-times oversampled $\Delta\Sigma$-quantized sinusoid.

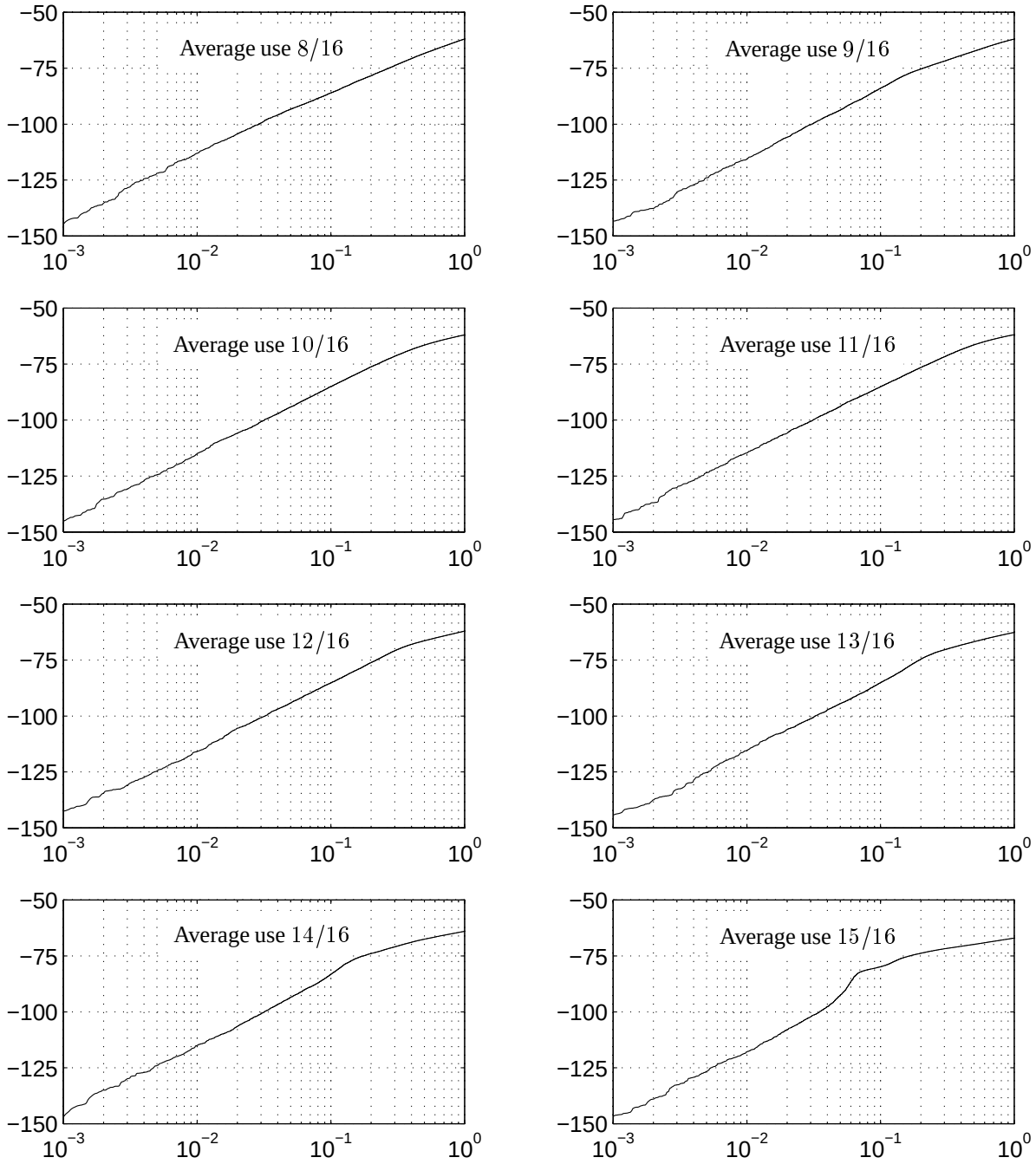


Figure 6.9: Static performance of a dual-ERS dithered UE-MS encoder. The plots show the signal-band power versus 1/OSR for each constant value of $d(k)$. Compare with Figure 6.5.

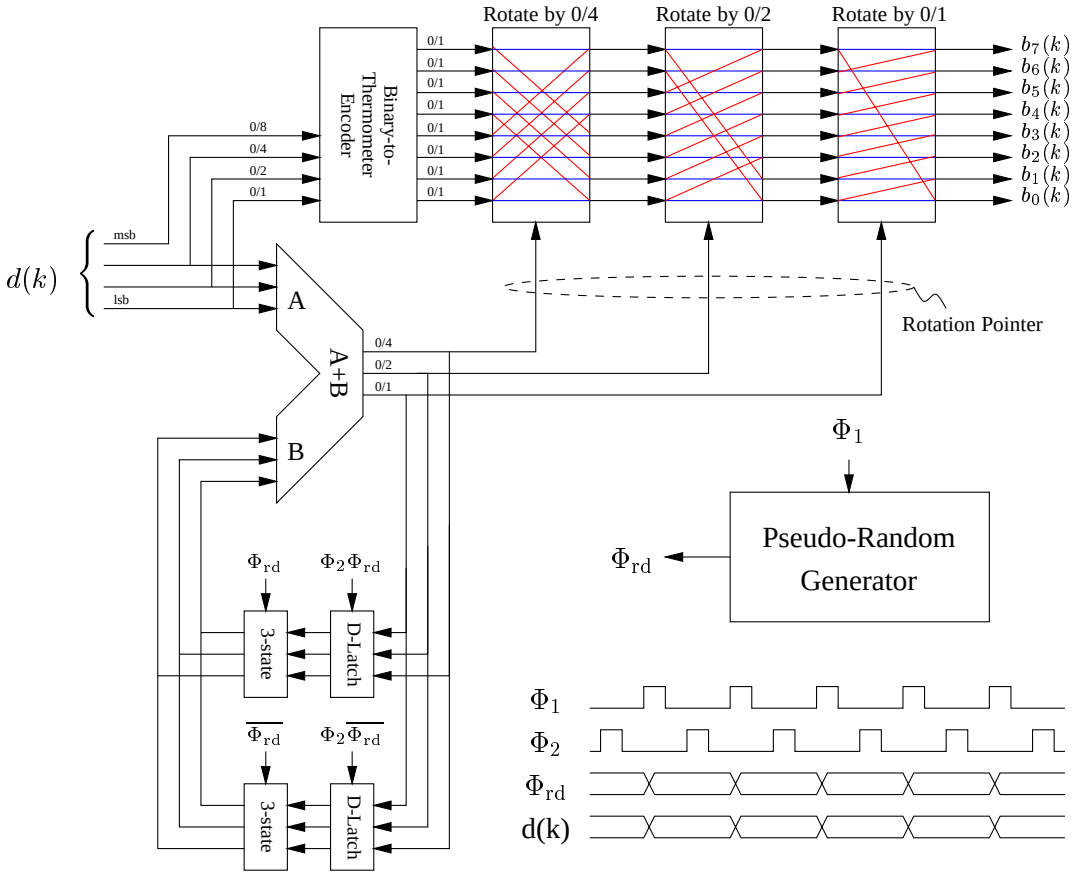


Figure 6.10: Simple implementation of a dithered ERS encoder

frequency as for un-dithered ERS encoders (the average value of $d(k)$ divided by the number N of unit elements).

6.3.1 A Family of Dithering Techniques

The following will investigate the possibility of dithering ERS encoders without allowing that any unit element is used twice before all the other elements have been used in between[10]. Dithering requires that equilibrium states are encountered, and that the unit elements then can be selected in at least two different ways.

Figure 6.11 shows a general technique for the implementation of dithered ERS encoders. With respect to an arbitrary initial condition (for $k = 0$), the unit elements are used sequentially[11] until they all have been used once. At this point, an equilibrium state is reached. The equilibrium state is likely to occur in the middle of a sample, i.e., when the encoder must select more unit elements than there are unused elements left to choose from. Instead of implementing a simple "wrap-around" event and selecting the extra unit elements from "the other end" (as for ERS encoders), the extra elements are chosen arbitrarily among the elements that are not already selected for the present sample. For simplicity, the extra elements are chosen as a sequence of elements, such that the ERS algorithm can be used. The ERS algorithm is employed (starting with respect to the new origin) until all the elements have been used twice, at which point the second equilibrium state is reached. Notice that it is acceptable to alter the orientation of the rotation between the equilibrium states. This process is continued; every time an equilibrium state is reached, a new origin and the orientation of rotation are chosen (randomly), whereby the desired dithered mismatch-shaping characteristic is obtained.

[10] More precisely, the unit elements must be selected such that there exists an origin in time, for which the outlined condition is fulfilled.

[11] Obviously, the unit elements may be used in any order, the ERS algorithm is chosen only to simplify the implementation.

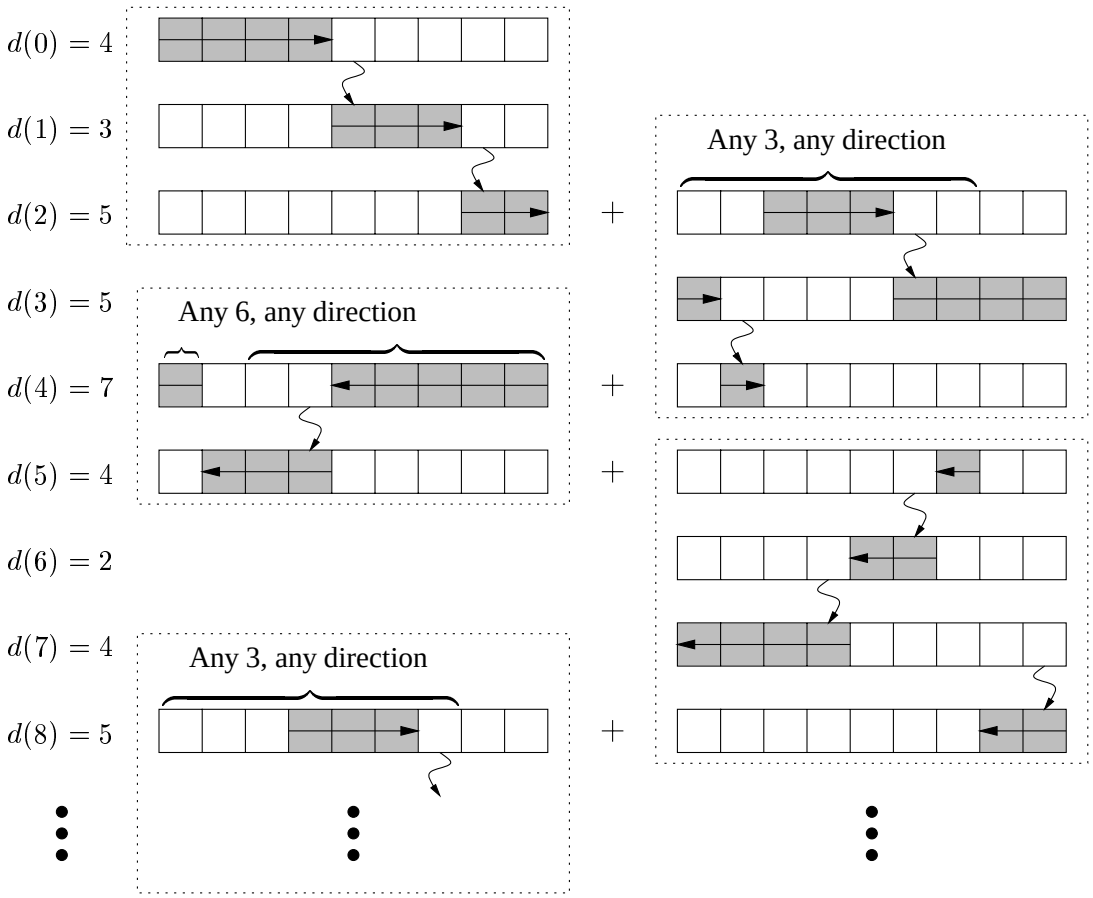


Figure 6.11: Identification of equilibrium states.

6.3.2 Random-Rotation-Scheme Dithering

Simple implementations are in general preferable, and the selection scheme shown in Figure 6.11 is perhaps a little more general than it needs to be. The unit elements used for $d(3)$, $d(5)$, and $d(8)$ are, for example, not sequences of neighbor elements, and hence two ERS encoders operating simultaneously, an array of logic-and gates, etc., will be required for the implementation. Sufficiently effective dithering can, however, be obtained even if only the orientation of rotation is chosen randomly when an equilibrium state is reached, and then the selection process can be designed to choose only sequences of elements. This simplified dithering technique – called *random-orientation* dithering – is illustrated in Figure 6.12. The rotation pointer is marked with an "$\times$" and the target pointer, which (when the element is used) identifies that an equilibrium state has been reached, is marked with a "$\circ$."

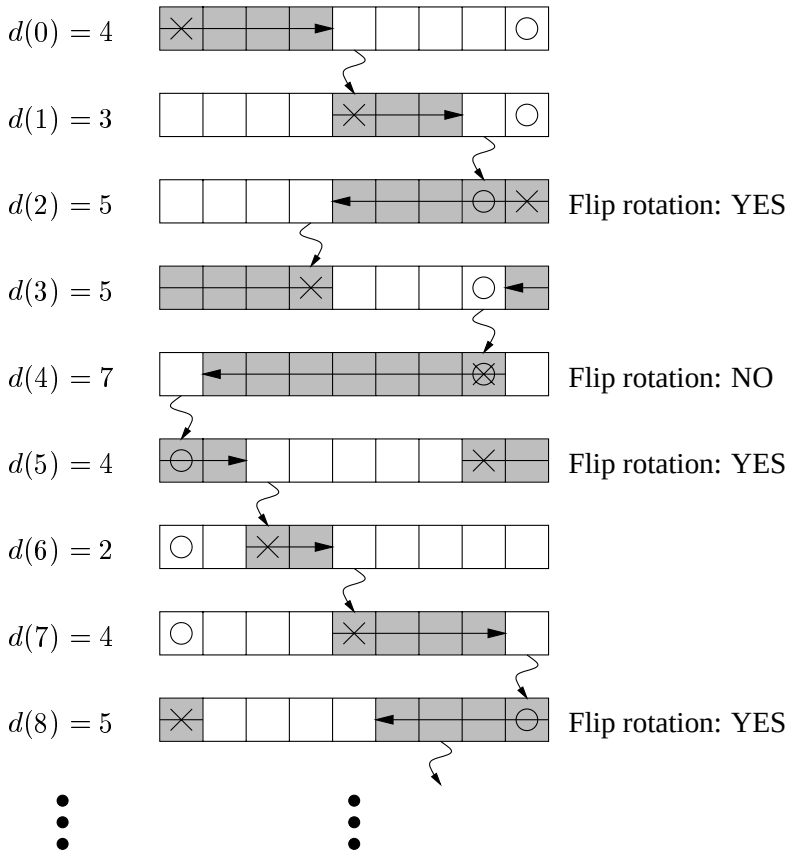


Figure 6.12: Random-rotation dithering of an ERS encoder.

Notice that it is quite simple to keep track of the two pointers. While the orientation of rotation is maintained, the target pointer remains constant whereas the rotation pointer is updated according to the ERS algorithm. When the orientation of rotation is altered (occurs randomly when an equilibrium state is reached), the two pointers are simply interchanged. The dithering effect is obtained because the target pointer will be a random signal.

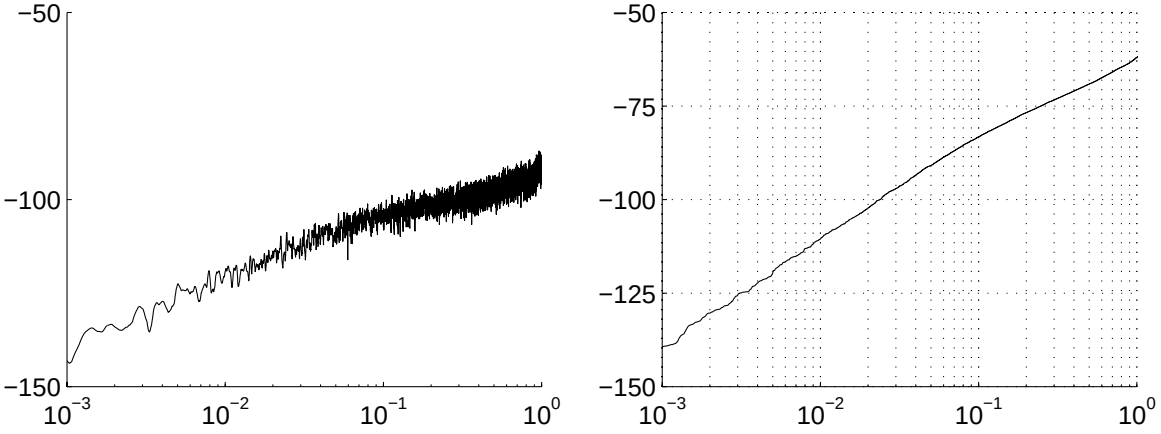
The encoder can, for example, be implemented using the topology shown in Figure 4.10 when the latch is replaced by a small digital state machine (the counterclockwise rotation can be mimicked by adding a signal-dependent offset to “B”).

Simulated Performance. The proposed random-orientation dithered encoder has been simulated using the same signals and conditions used for the dithered tree-structure encoder and the dual-ERS encoder. The results are shown in Figures 6.13, 6.14, and 6.15. The performance is, unfortunately, not better than the performance of dual-ERS encoders, which in general are simpler to implement. It is believed that the unexpected poor performance is due to the relatively slow change in the target pointer. If, for example, the probability of altering the orientation of the rotation is reduced from 0.5 to (say) 0.3, then the relatively flat region (10 dB per decade) for OSR less than 10 in Figure 6.14a is extended to much lower frequencies (not shown).

The performance can probably be improved by also choosing a new origin for every new equilibrium state, but then the hardware complexity will increase considerably (although it may be worthwhile for small encoders). A simpler way to improve the performance, for example, would be to constrain the random process such that the rotation’s orientation is altered for at least every second equilibrium state, but not more than two times in a row (not tested).

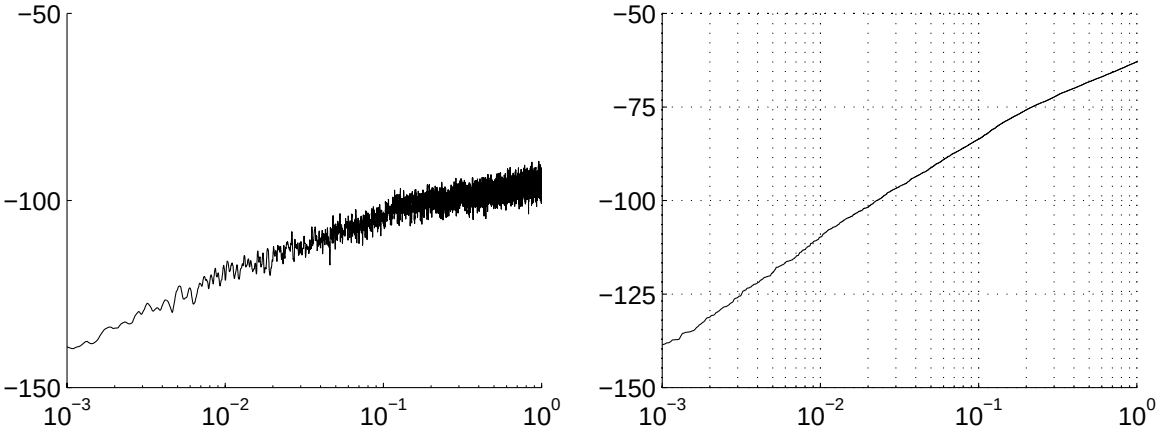
6.4 Conclusion

Idle tones are a serious problem for mismatch-shaping DACs, and it is generally necessary to use some kind of dithering technique to break the patterns/tones. The tree-structure UE-MS encoder can be efficiently dithered, but the hardware complexity is considerable. The proposed dual-ERS encoders are



(a) Spectral power density versus frequency (b) Signal-band power versus 1/OSR

Figure 6.13: Error signal from a random-orientation dithered ERS UE-MS DAC. The input signal $d(k)$ was a -12 dBFS 64-times oversampled $\Delta\Sigma$-quantized sinusoid. Compare with Figures 6.3 and 6.7.



(a) Spectral power density versus frequency (b) Signal-band power versus 1/OSR

Figure 6.14: Error signal from a random-orientation dithered ERS UE-MS DAC. The input signal $d(k)$ was a -1.5 dBFS 64-times oversampled $\Delta\Sigma$-quantized sinusoid. Compare with Figures 6.4 and 6.8.

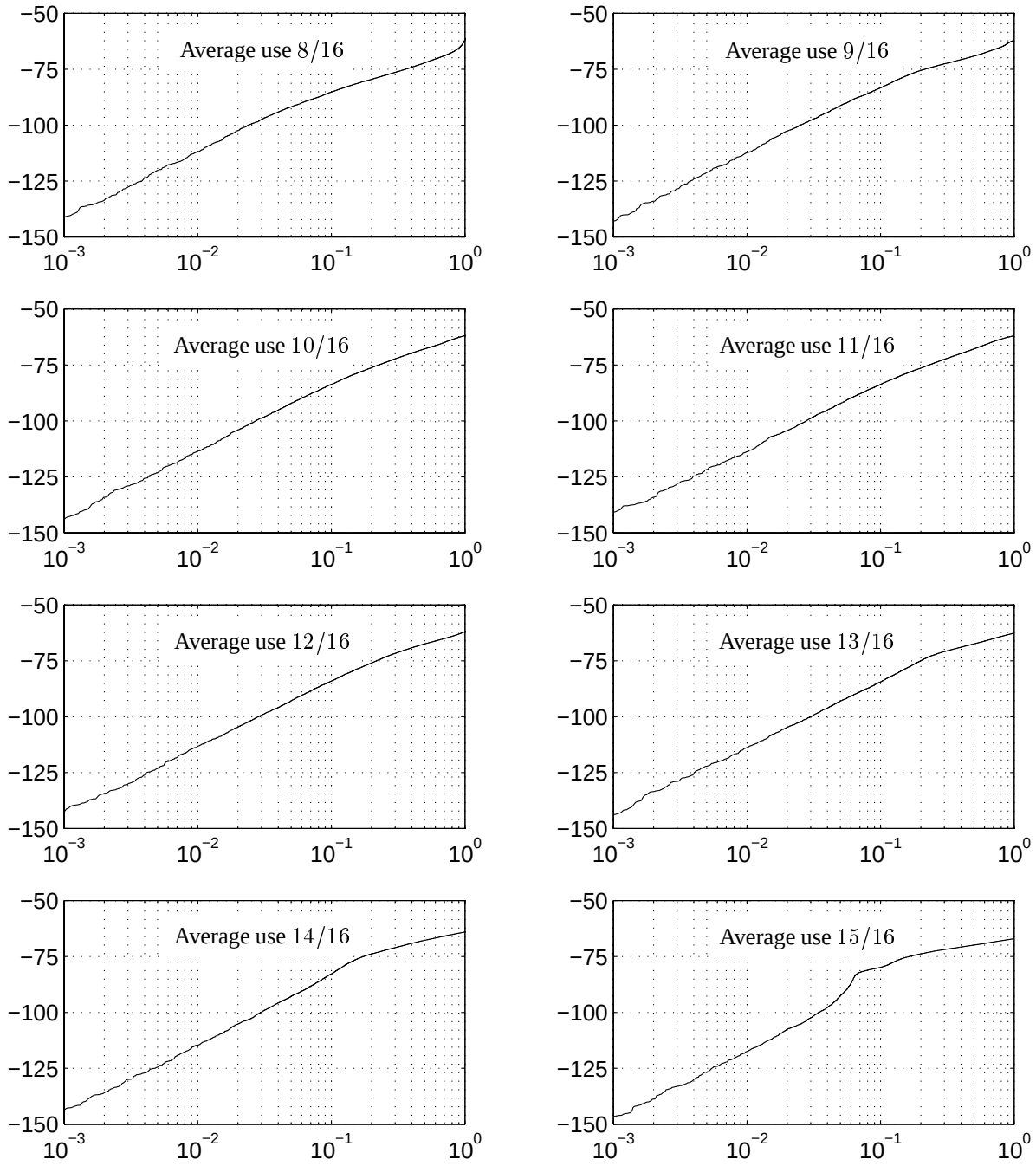


Figure 6.15: Static performance of a random-orientation dithered ERS UE-MS encoder. The plots show the signal-band power versus 1/OSR for each constant value of $d(k)$. Compare with Figures 6.5 and 6.9.

significantly simpler to implement, and the performance is comparable. They have a slight tendency to produce spread (partially dithered) idle tones, but these "tones" will only appear at high frequencies and they are seldom a problem. The lowest frequency at which these tones may appear is inversely proportional to the number of unit elements, and they are hardly noticeable for encoders with only 4 of 8 unit elements. The next chapter will discuss a technique which voids the need for UE-MS DACs of resolution higher than a few bits.

It is a key point that the shape of the error signal's spectral power density is dependent on the relative drive of the encoder. This implies that the OSR cannot be made as low as expected/desired unless the encoder is designed to have a larger-than-minimum output range. However, for several reasons (cf. Section 4.4), the average drive will typically be only 50% to 75% of the full range, and hence the effect is usually not too troublesome[12].

[12]The lower right and left plots in Figures 6.5 6.9, and 6.15 can generally be neglected.

Chapter 7

Scaled-Element Mismatch-Shaping D/A Converters

Mismatch-shaping DACs are the key element for the implementation of high-resolution wide-bandwidth A/D and D/A (signal) converters. In Section 4.4.3, it was found that the SER performance of a unit-element mismatch-shaping (UE-MS) DAC depends mainly on the technology's matching index (cf. Equation (4.28)), and not on the inherent resolution of the DAC. The hardware complexity of the digital UE-MS encoder is, however, only reasonable if the inherent resolution is fairly low, say 5 bits or less, and hence a $\Delta\Sigma$ quantizer/modulator is necessary to encode/interpolate the signal to an oversampled low-resolution representation.

In Section 4.3.1, it was found that high-performance low-oversampled $\Delta\Sigma$ quantizers can be implemented using an only 5-bit representation of $d(k)$, but notice that high-order loop filters and complex digital multi-rate filters (to decimate the output) are required for the overall implementation. Similarly, high-performance low-oversampled D/A converter systems can be implemented using an only 5-bit signal representation; but again, high-order $\Delta\Sigma$ modulators and high-performance high-order analog filters are required for the implementation. In other words, the inherent resolution represents a tradeoff between the complexity of the encoder and the complexity of the rest of the system. Clearly, the development of simple high-resolution mismatch-shaping DACs will improve this tradeoff considerably, thereby facili-

tating the implementation of much simpler high-performance data converter systems.

7.1 High-Resolution Mismatch-Shaping DACs

Simple high-resolution DACs can be implemented using an array of scaled elements (e.g., the binary-weighted DAC discussed on page 56); the problem is to make them mismatch-shaping. In fact, it is impossible to implement a mismatch-shaping DAC using only an encoder and an array of (inaccurately) binary-scaled elements. To facilitate mismatch-shaping, the array of elements must fulfill the necessary but not sufficient criteria that each nominal output value can be generated in at least two different ways (i.e., the DAC must have *sub levels* [59]). Neither the analog nor the digital portions of the DAC, however, need be very complex The following discussion will include an example of a mismatch-shaping DAC based on only *two* arrays of binary-scaled elements.

7.1.1 General Aspect of the Design of Mismatch-Shaping Encoders

The main issue in the design of scaled-element mismatch-shaping (SE-MS) DACs is, of course, the design of the digital encoder, which preferably should be made as simple as possible. A good starting point to solve this problem was provided in Section 4.4.1, where Equation (4.11) expresses the error signal $m(k)$ produced by a generic DAC of the considered topology (cf. Figure 4.8). The error signal was expressed as the sum of the *gain mismatch errors* $\left(\sum_{i=0}^{P-1}[b_i(k)[K_i - K_d]]\right)$ and the *local nonlinearity errors* $\left(\sum_{i=0}^{P-1}\mathrm{INL}_i[b_i(k)]\right)$, which were defined with respect to the linear characteristics of the (sub) DACs

$$m(k) = \sum_{i=0}^{P-1}[b_i(k)[K_i - K_d]] + \sum_{i=0}^{P-1}\mathrm{INL}_i[b_i(k)] \tag{7.1}$$

To make the composite DAC mismatch-shaping, it must be assured that $m(k)$ has only negligible power in the system's signal band. The errors can be considered individually.

Controlling the Local Nonlinearity Errors. The local nonlinearity errors can be suppressed in the signal band if the individual sub DACs are designed as either single-bit or mismatch-shaping DACs.

In the general case[1], most of the P sub DACs will be implemented as (low-resolution) UE-MS DACs. Notice, however, that the UE-MS DACs need not mutually employ unit elements of the same size; hence, the number of possible output levels can be vastly higher than the total number of elements. The simplest implementation will use a binary-weighted array of UE-MS DACs, each comprising only two elements.

Controlling the Gain Mismatch Errors. The nature of the system is that the parameters $[K_i - K_d]$ are small[2], but unknown constants. To obtain the mismatch-shaping property (of the composite DAC), each of the signals $b_i(k)$ must be of the form

$$b_i(k) = \mathcal{L}_i\{d(k)\} + h^{-1}(k) * n_i(k) \tag{7.2}$$

where $\mathcal{L}_i$ are *linear* operators, $h^{-1}(k)$ is the impulse response of a filter[3] that suppresses the signal band (cf. Section 4.4.4), and $n_i(k)$ are (preferably non-tonal) bounded signals. Provided that (7.2) is fulfilled for all $b_i(k)$, the gain mismatch error $m_{\text{gain}}(k)$ can be written in the form

$$\begin{aligned} m_{\text{gain}}(k) &= \left[\sum_{i=0}^{P-1}[K_i - K_d]\left[\mathcal{L}_i\{d(k)\} + h^{-1}(k) * n_i(k)\right]\right] \\ &= \left[\sum_{i=0}^{P-1}[K_i - K_d]\mathcal{L}_i\{d(k)\}\right] + h^{-1}(k) * \left[\sum_{i=0}^{P-1}[K_i - K_d]n_i(k)\right] \\ &= \mathcal{L}_{\text{DAC}}\{d(k)\} + h^{-1}(k) * n_{\text{DAC}}(k) \end{aligned} \tag{7.3}$$

[1]New types of (e.g., serial) mismatch-shaping DACs are currently being developed; they can often substitute for the more typical UE-MS DACs.

[2]The sub DACs are designed to have the same nominal gain, and $[K_i - K_d]$ expresses the ith DAC's gain's deviation from the composite DAC's gain K_d. K_d may be either the average value of the sub DACs' gains (as for UE-MS DACs), or it may be the average gain of a subset of the sub DACs (explained in the main text). The constants, $[K_i - K_d]$, are stochastic variables with zero as the expected value. The standard deviations σ_i will in general be $\sigma_i = \sigma_{\text{process}}\sqrt{\frac{||b_i(k)||}{||d(k)||}}$, where $||\cdot||$ is an appropriate norm (say, the peak-to-peak value), and σ_{process} is the technology's matching index (with respect to the assigned chip area).

[3]It need not be the same type/order of filter for all $b_i(k)$, but Equation (7.3) becomes confusing if this extra flexibility is included. The main text will provide several examples of systems where $h^{-1}(k)$ is not the same for all $b_i(k)$.

where $\mathcal{L}_{\rm DAC}\{d(k)\}$ is an implementation-specific linear operator, which usually will be zero if the DAC's gain K_d is defined with respect to the actual implementation, and not as an absolute value[4]. Notice that $\mathcal{L}_{\rm DAC}$ will be small because the coefficients $[K_i - K_d]$ are small. Hence, $m_{\rm gain}(k)$ will have the properties required for mismatch-shaping DACs.

7.1.2 Mismatch-shaping Unit-Element DACs – Revisited

To shed some light on the implications of Equations (7.2) and (7.3), consider once again a mismatch-shaping unit-element DAC with P elements. The UE-MS encoder produces P signals $b_i(k)$ with the same average value, hence $\mathcal{L}_i\{d(k)\} = d(k)/P$ for all i. Unless $d(k) = 0$ or $d(k) = P$, a truncation will occur (because $b_i(k) \in \{0, 1\}$), which is performed such that $b_i(k) - \mathcal{L}_i\{d(k)\}$ has the desired property that $b_i(k) - d(k)/P = h^{-1}(k) * n_i(k)$.

It may be observed that this point of view leads directly to the *parallel* UE-MS encoder that was proposed independently by Richard Schreier [12] and Akira Yasuda [53] (shown in a slightly modified form[5] in Figure 7.1). The parallel UE-MS encoder is based on P individual $\Delta\Sigma$ modulators (similar to tree-structure UE-MS encoders), and the correct number of unit elements are selected by varying the truncators' threshold value[6]. Unfortunately, this is, computationally, a complex operation which involves use of the "sort" function.

7.1.3 Complicated Scaled-Element Mismatch-Shaping Encoders

A scaled-element mismatch-shaping encoder can, in principle, be implemented in a topology similar to that shown in Figure 7.1, where the individual $b_i(k)$ signals control elements of non-uniform values,

[4] More precisely, the DAC's signal transfer function is $K_d + \mathcal{L}_{\rm DAC}$; hence, $\mathcal{L}_{\rm DAC} = 0$ if the DAC is free of dynamics and K_d is defined with respect to the actual implementation. That this is not the only option is exemplified by $\Delta\Sigma$ DACs implemented in the topology shown in Figure 3.24, which will have a signal transfer function close to, but not exactly, K_d. In that case, $\mathcal{L}_{\rm DAC}$ will be a high-pass filter function with only very little gain in the signal band.

[5] For simplicity and graphical purposes, the variable-threshold truncator is not suitable for actual implementations (cf. the original publication [12]).

[6] Professor Gabor Temes introduced the threshold-control element/idea during a private meeting.

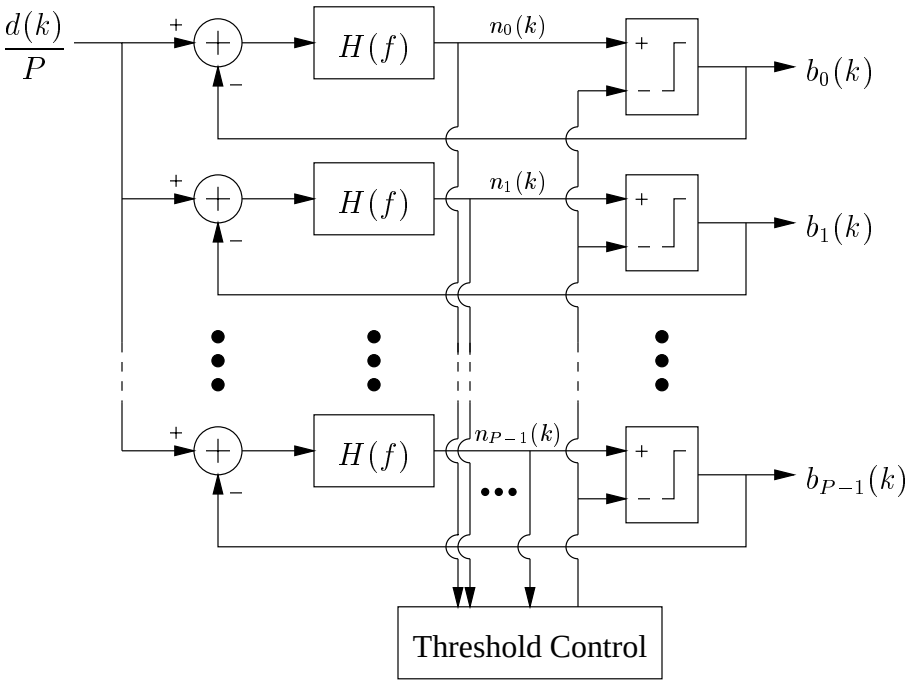


Figure 7.1: Parallel UE-MS encoder.

and where each $\Delta\Sigma$ modulator is fed a fraction of $d(k)$ which corresponds to the nominal value of the respective element. The threshold-control mechanism, however, must be replaced by a vastly more complex selection mechanism, which assures that the output of the loop filters remains bounded, and that the sum of the selected elements' nominal values equals $d(k)$. Overall, the scheme becomes quite complicated. The interested reader is encouraged to study [59].

7.1.4 Simple Scaled-Element Mismatch-Shaping Encoders

Simple encoders are generally characterized by decentralized calculations which can be performed in parallel. A useful technique is, for example, to gather elements of the same nominal value in groups, and let separate UE-MS encoders maintain the mismatch-shaping operation among each group of elements. The UE-MS encoders can be construed either as part of the sub DACs, which is the approach taken in the following discussion, or as part of the overall encoder, in which case there are as many sub DACs as elements. Notice that, either way, this concept complies with the requirement to control local

nonlinearity errors (cf. page 166).

Introducing the Master DAC. Another important technique, which can be used to simplify the encoder, is to allow only one of the linear operators $\mathcal{L}_i$ to be nonzero. In other words, when using this technique, one of the signals (say $b_0(k)$) is defined as the *master signal*, which is the only signal that is correlated[7] to the input signal $d(k)$; consequently, the composite DAC's gain is defined by the *master DAC's* gain, i.e., $K_d = K_0$.

Notice that this technique complies with the fundamental operation of a $\Delta\Sigma$ modulator, where the low-resolution output signal is the master signal. The new element (cf. Figure 7.2) is that the truncation signal $\epsilon(k)$ is not discarded, but separated into one or more *compensation* signals $b_i(k), i \neq 0$, which are D/A converted individually and added to the D/A converted master signal. This way, the input signal $d(k)$ is *not* truncated and only partially represented by an oversampled lower-resolution signal, but instead re-encoded and fully represented by the set of *spectrally-coded signals* $b_i(k)$, which are better suitable for direct D/A conversion with an array of inaccurately matched elements. In other words, the analog output signal $f(k)$ will not include a large truncation error signal (often called "the shaped quantization noise"), but will include only the mismatch error signal $m(k)$, which typically is quite small (cf. Equation (4.28)).

The master signal $b_0(k)$ automatically fulfills the requirement (7.2), because

$$\begin{aligned} b_0(k) &= d(k) - e(k) \\ &= d(k) - h^{-1}(k) * n_0(k) \end{aligned} \tag{7.4}$$

Hence, the only requirements that need be fulfilled to obtain the mismatch-shaping property are that the compensation signals be of the form

$$\epsilon(k) = \sum_{i=1}^{P-1} b_i(k) \tag{7.5}$$

$$b_i(k) = h^{-1}(k) * n_i(k), \quad i \neq 0 \tag{7.6}$$

[7] In the signal band, cf. Equation (7.2).

and that the set of spectrally-coded signals $b_i(k)$ be D/A converted using single-bit and/or mismatch-shaping DACs. Digital systems that fulfill Equations (7.4), (7.5), and (7.6) will hereinafter be called *spectral encoders*.

The simplest way to fulfill (7.6) is to omit the separator and let $b_1(k) = \epsilon(k)$. This technique will be investigated in the next section, and a discussion of some more advanced separation techniques will follow.

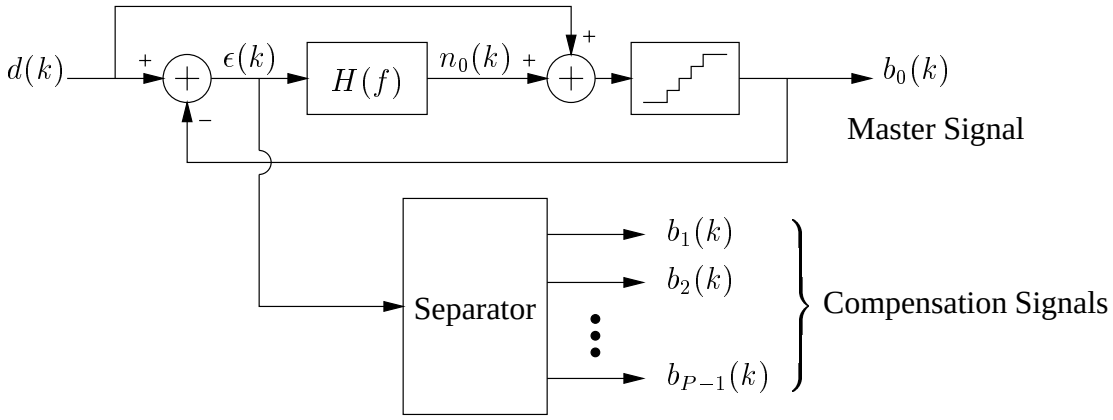


Figure 7.2: Spectral encoder that separates $d(k)$ into a master signal and one or more compensation signals with only little signal-band power.

7.2 A Dual-Type-Element Mismatch-Shaping DAC

Figure 7.3 shows a mismatch-shaping DAC controlling elements $\mathcal{A}_0$ and $\mathcal{A}_1$ of two different nominal nominal values (with a ratio of 8). The digital input signal $d(k)$ is separated into $b_0(k)$ and $b_1(k)$ by the spectral encoder, which is implemented as shown in Figure 7.2, where $b_1(k) = \epsilon(k)$. For graphic simplicity, the subtraction block is shown outside the $\Delta\Sigma$ modulator, although it will naturally be implemented as a part of the $\Delta\Sigma$ modulator.

Resolution of the Spectrally-Encoded Signals. The resolution of the three signals $d(k)$, $b_0(k)$, and $b_1(k)$ is an important aspect to consider. This is important because it determines how many elements

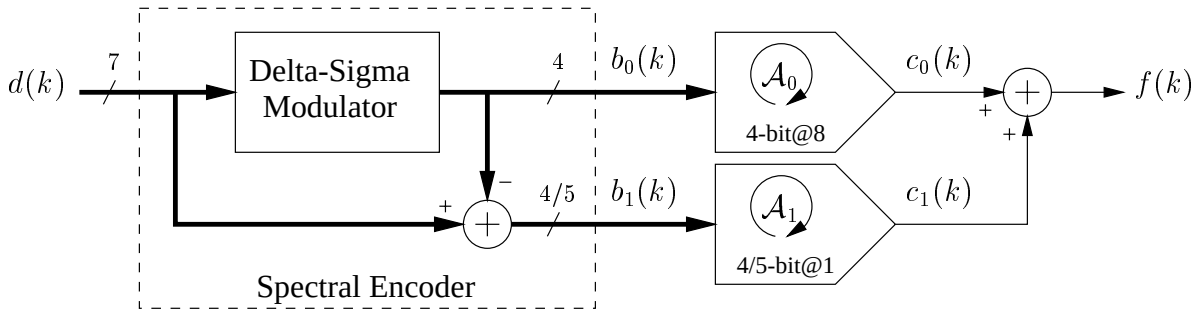


Figure 7.3: Simple dual-type-element mismatch-shaping DAC.

(in $\mathcal{A}_0$ and $\mathcal{A}_1$) are required for the conversion of $d(k)$ with a given resolution. Preferably, for a simple encoder, the full-scale range of $b_1(k)$ should be small relative to the full-scale range of $d(k)$.

As always for $\Delta\Sigma$ modulators, the master signal $b_0(k)$ must span at least the same range of values as the input signal $d(k)$, and usually the range must be a little wider to allow some fluctuation in $n_0(k)$ (cf. Section 4.3.1 and Figure 7.2). In the ideal case, $b_0(k)$ will for all k attain one of the two values that are the closest to the value of $d(k)$, in which case the full-scale range of $b_1(k)$ will be about twice as wide as the step size of $b_0(k)$, i.e., a 7-bit input signal can e.g. be spectrally encoded into two 4-bit signals whereby the unit elements in $\mathcal{A}_1$ will be 8 times smaller than the unit elements in $\mathcal{A}_0$, and the 128 levels be represented using only 32 elements. The full-scale range of $b_1(k)$ will, however, depend highly on the design of the $\Delta\Sigma$ modulator's loop filter $H(f)$.

7.2.1 Designing the Delta-Sigma Modulator

The full-scale range of $\epsilon(k)$ (cf. Figure 7.2) is a main concern in the design of the spectral encoder's $\Delta\Sigma$ modulator. To avoid unnecessary[8] large values of $\epsilon(k)$, the feed-forward branch (from $d(k)$ to the input of the truncation element) should always be included in the topology (for reasons discussed on page 76). Assuming, therefore, that the $\Delta\Sigma$ modulator's topology is as shown in Figure 7.2, the only free parameters are the loop filter's transfer function $H(f)$ and the master signal's step size and resolution.

[8]Unless the feed-forward branch is implemented, the full-scale range of $\epsilon(k)$ may, for any loop filter, be at least as large as the full-scale range of $d(k)$, assuming that $d(k)$ is not spectrally constrained.

If the performance is evaluated *only* with respect to the full-scale range of $\epsilon(k)$, it would be optimal to use a first-order $\Delta\Sigma$ modulator

$$H(f) = \frac{z^{-1}}{1+z^{-1}}, \quad \text{where } z = e^{j2\pi f/f_s} \tag{7.7}$$

for which the full-scale range of $\epsilon(k)$ is only twice the master signal's step size. A reason why this may *not* always be the best design is that first-order $\Delta\Sigma$ modulators are extremely tonal, and hence the gain error

$$m_{\text{gain}}(k) = b_1(k)[K_1 - K_0] \tag{7.8}$$

will include *gain-error* idle tones[9], because $b_1(k) = \epsilon(k)$ includes many strong idle tones that are not perfectly canceled if $[K_1 - K_0]$ is nonzero (it is usually in the order of -60 dB).

Avoiding Gain-Error Idle Tones. Essentially, gain-error idle tones can be prevented (suppressed) in one of two ways. Either $\epsilon(k)$ can be separated into several compensation signals $b_i(k), i \neq 0$ in a nonlinear and aperiodic way (which may turn out to be a risky business), or they can be removed at the source by designing the $\Delta\Sigma$ modulator to be idle-tone free[10] using dither [1], chaos [44], or any other of the well-know techniques. The latter approach is usually much preferable, although it may increase the magnitude of $\epsilon(k)$.

For dither[11] to be effective, it should have a uniform probability density function in a range which is as wide as the master signal's step size. It is often claimed that the use of dither in multi-bit $\Delta\Sigma$ modulators is "harmless" because it is small relative to full-scale output and hence does not noticeably affect the stability. This is often true, but dither is actually quite harmful in this context, because the dither's magnitude will add linearly to the magnitude of $\epsilon(k)$. For example, in a fully dithered first-order $\Delta\Sigma$ modulator, the full-scale range of $\epsilon(k)$ will be *four* times the master signal's step size, and hence the compensation DAC $\mathcal{A}_1$ is required to have twice an many unit elements as in the un-dithered system.

[9] In this case they do not originate from the UE-MS DACs, but directly from the spectral encoder.

[10] This is usually not too difficult when the master signal is multi-bit.

[11] Noise added to $n_0(k)$, see [1] for details.

Another technique to suppress idle tones is to use a higher-order (i.e., second-order or higher) loop filter and possibly make it slightly chaotic. It is well understood that idle tones cannot be fully avoided in this way [1], but considering that the tones will be further suppressed by the gain-matching factor $[K_1 - K_0]$, cf. Equation (7.8), they need only be suppressed to (say) -80 dBFS which is quite possible. The high-order loop filter should *not* be designed with a high $\mathrm{NTF}_{\max}$ value (cf. Section 4.3.1), because that will vastly increase the magnitude of $\epsilon(k)$. The loop filter can, however, easily be of high order without increasing the magnitude of $\epsilon(k)$ noticeably. For example, Figure 4.6 shows that a 6th order $\Delta\Sigma$ modulator with $\mathrm{NTF}_{\max} = 2$ will have a full-scale range of $\epsilon(k)$, which is only slightly larger than twice the master signal's step size. Notice that the compensation DAC need not employ a power-of-two number of unit elements; $\mathcal{A}_1$ can very well include (say) 22 unit elements. The use of high-order $\Delta\Sigma$ modulators also has the advantage that $\epsilon(k)$ will be a high-order shaped signal, hence even a substantial gain error $[K_1 - K_0]$ is acceptable with respect to the signal-band performance. In other words, local matching (i.e., within $\mathcal{A}_0$ and $\mathcal{A}_1$) is more important than global matching (i.e., K_0 relative to K_1), which is an important observation to make use of when the circuit is to be laid out (cf. page 120).

7.2.2 Parallel Work Published

Independent of this work, Robert Adams from Analog Devices Inc. has invented and published [35] a mismatch-shaping DAC identical to that shown in Figure 7.3. The published system was a high-resolution DAC where the 20-bit input signal was first interpolated to a 6-bit representation using a traditional second-order (Candy-structure) $\Delta\Sigma$ modulator. This 6-bit signal was D/A converted using a scaled-element mismatch-shaping DAC of the type shown in Figure 7.3, where the spectral encoder was of first order. The resolution of the master DAC was 3 bits and the resolution of the compensation DAC was 4 bits.

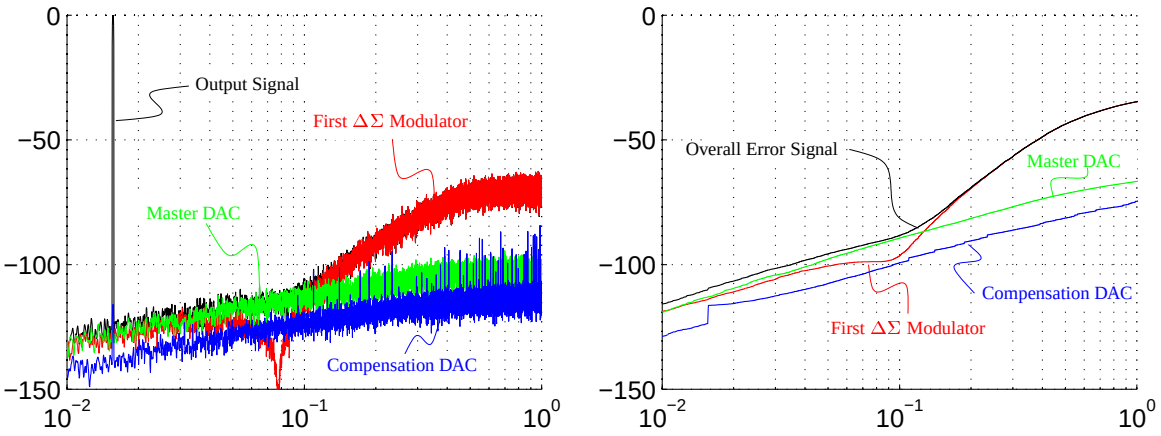
Measured Performance. The system worked very well (113 dB performance was reported), although the used low-order spectral encoder very well could have caused problems (discussed above). During the question period after the presentation, the speaker's attention was drawn to the tones that were measured at the -110 dBFS level. They were claimed to be idle tones from the interpolating $\Delta\Sigma$ modulator, and

that they had since been avoided by dithering the modulator (second silicon). This *may* be the correct explanation, but they could also have been idle tones from the used spectral encoder's first-order $\Delta\Sigma$ modulator. The speaker, however, specifically pointed out that idle tones cannot originate from this source, because the interpolated signal will include an aperiodic component (the truncation error). That is incorrect.

Simulation Results for a Similar System. Figure 7.4 shows the simulation results from a similar system that was designed for operation at OSR $= 10$. The red traces show the truncation error from the fully-dithered third-order $\Delta\Sigma$ modulator $(\mathrm{NTF}_{\max} = 4)$ used to interpolate the signal to 8-bit representation. The green traces show the nonlinearity error from the master DAC, which was designed as a 20-element dual-ERS dithered mismatch-shaping DAC. This error dominates the signal-band performance (which it should for a good design). The blue traces show the combined gain and nonlinearity error from the compensation DAC, which was designed as a 40-element dual-ERS dithered mismatch-shaping DAC. Notice the many spurious tones that are part of this error signal. These tones originate from idle tones in the spectral encoder's first-order $\Delta\Sigma$ modulator; they disappear if the encoder is just partially (0.3) dithered. The compensation DAC's error signal is somewhat smaller than the master DAC's error signal, because the master signal is about 20 dB larger than the compensation signal.

How to Improve the System. Consider again the high-resolution DAC system discussed above. In Figure 7.4 it can be observed that the interpolating $\Delta\Sigma$ modulator's truncation error is dominating in the Nyquist band, and it will typically be necessary to use an analog filter to smooth out the output waveform. Clearly, it would be preferable if the high-resolution input signal could be D/A converted directly, such that the master DAC's error signal would be the dominating error in the entire frequency spectrum. This way, the analog filter could be omitted and the system's performance improved.

Figure 7.3 shows a good concept for the implementation of mismatch-shaping DACs, but if $d(k)$ is of more than 7 to 8 bits of resolution, the UE-MS encoders will have to be large and the complexity of the digital circuitry becomes of great concern. Hence, simpler mismatch-shaping encoders are sometimes required. To be commercially interesting, the mismatch-shaping encoders' complexity should be only



(a) Spectral power density versus frequency

(b) Signal-band power versus 1/OSR

Figure 7.4: Simulated performance of a D/A converter system similar to that presented at ISSCC 98. Notice the idle tones in the compensation DAC's error signal.

linearly dependent on the resolution (in bits), and the UE-MS encoders should each be of less than 3 to 4 bits of resolution. Several techniques will be proposed in the following.

7.3 Tree-Structure Scaled-Element Mismatch-Shaping DACs

Consider the DAC structure shown in Figure 7.3. The two DACs converting $b_0(k)$ and $b_1(k)$ are required to be mismatch-shaping, but they need not be UE-MS DACs. Either of them can be a scaled-element mismatch-shaping (SE-MS) DAC, implemented e.g. as shown in Figure 7.3. In other words, the structure can be used recursively in a symmetrical tree structure as shown in Figure 7.5.

The first $\Delta\Sigma$ modulator separates $d(k)$ into a (say) 7-bit internal master signal and a corresponding (say) 7-bit truncation signal $\epsilon(k)$. The internal master signal is D/A converted with a SE-MS DAC, here consisting of the second $\Delta\Sigma$ modulator and two scaled 4-bit UE-MS DACs. The truncation error signal is also D/A converted with a SE-MS DAC, here consisting of the third $\Delta\Sigma$ modulator and two scaled 4-bit UE-MS DACs.

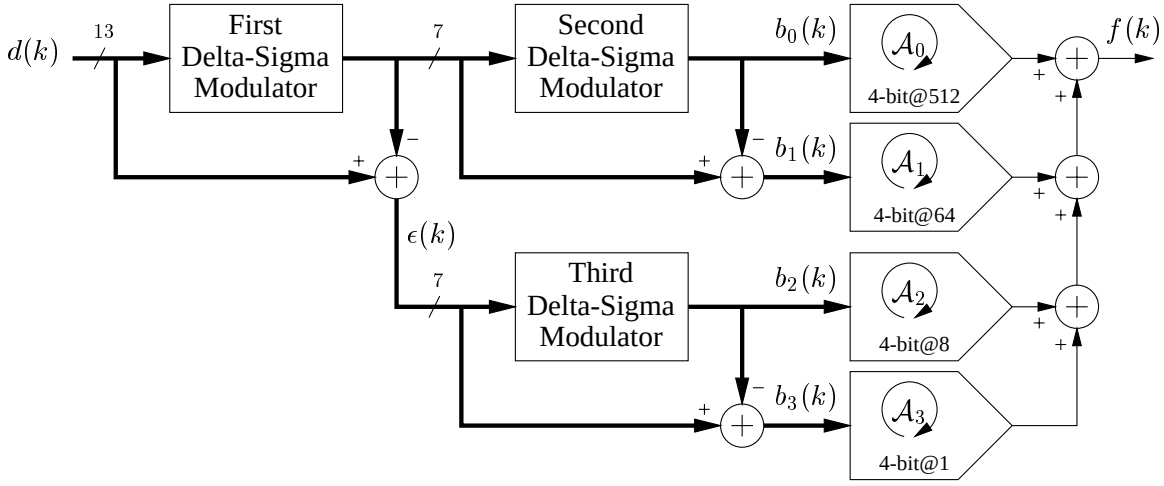


Figure 7.5: Symmetrical tree-structure scaled-element mismatch-shaping DAC

Verification of the Mismatch-Shaping Property. The fundamental property of the $\Delta\Sigma$ modulators is that $\epsilon(k)$, $b_1(k)$, and $b_3(k)$ will be of the form (7.6), and since $b_2(k) = \epsilon(k) - b_3(k)$, it follows that $b_2(k)$ will be of the same form (7.6). Hence, because $d(k) = \sum_{i=0}^{3} b_i(k)$, the three $\Delta\Sigma$ modulators in combination can be construed as a spectral encoder that separates $d(k)$ into the master signal $b_0(k)$ and the three compensation signals $b_1(k)$, $b_2(k)$, and $b_3(k)$. Notice that the topology is somewhat similar to the high-resolution DAC discussed on page 174. The red traces in Figure 7.4 show the discarded truncation signal $\epsilon(k)$, which now is compensated for by including the third $\Delta\Sigma$ modulator and the two extra 4-bit UE-MS DACs shown in Figure 7.5.

Designing the Delta-Sigma Modulators. To avoid gain-error idle tones, $\epsilon(k)$ should be nearly idle-tone-free, which (as discussed on page 172) is best obtained by designing the first $\Delta\Sigma$ modulator with a higher-order loop filter with a low $\mathrm{NTF}_{\max}$ value. However, because the internal master signal represents the input signal $d(k)$, gain-error idle tones can also originate from the second $\Delta\Sigma$ modulator (cf. Figure 7.4); hence, it should also be designed with a higher-order loop filter. On the other hand, the truncation signal $\epsilon(k)$ consists of *only* shaped quantization noise (an ARMA pseudo-stochastic process), which is usually sufficiently "random" to prevent idle tones in the third $\Delta\Sigma$ modulator, even if the loop filter is of only first order (7.7).

7.3.1 Asymmetrical Tree Structures

SE-MS encoders can also be designed in asymmetrical tree structures. In particular, a *master* $\Delta\Sigma$ modulator can be used to interpolate $d(k)$ directly to the resolution desired for the master signal $b_0(k)$, in which case internal master signals (as in Figure 7.5) are avoided. The advantage of this approach is that only the master $\Delta\Sigma$ modulator will be likely to produce gain-error idle tones, hence all but that $\Delta\Sigma$ modulator can be designed with first-order loop filters.

Figure 7.6 shows a SE-MS DAC based on the above concept. The master $\Delta\Sigma$ modulator can be based on a (say) second-order loop filter with a $\mathrm{NTF}_{\max}$ value of (say) 1.5. The 4-bit master signal $b_0(k)$ is D/A converted with a dithered UE-MS DAC, and the truncation signal $\epsilon(k)$ is D/A converted with a symmetrical SE-MS DAC of the type shown in Figure 7.5. The first, second, and third compensation $\Delta\Sigma$ modulators can be implemented as first-order modulators without encountering significant idle-tone problems.

The ratio of the unit elements in $\mathcal{A}_0$ and $\mathcal{A}_4$ is as large as 2048, which is more than sufficient, because the master DAC's error signal $m_0(k)$ generally[12] will be larger than the analog equivalent of $b_4(k)$. In other words, the least-significant DAC usually need not be mismatch-shaping, and quite often it makes sense to simply discard is. This is particularly the case when the third compensation $\Delta\Sigma$ modulator is of second order (although it is simpler to implement a small binary-weighted non-mismatch-shaping DAC for the D/A conversion of $b_4(k)$).

Implementation of the Spectral Encoder. Figure 7.7 shows an implementation of the spectral encoder used for the compensation DAC in Figure 7.6 . The three first-order $\Delta\Sigma$ modulators are implemented in the so-called *error-feedback* topology [1], whereby only two adders and a delay element are needed for the implementation of each modulator (the rectangles with a diagonal line across represent a hardwired separation in most- and least-significant bits).

[12] Unless the technology's matching index (total-area relative standard deviation) is smaller than about 1/2000.

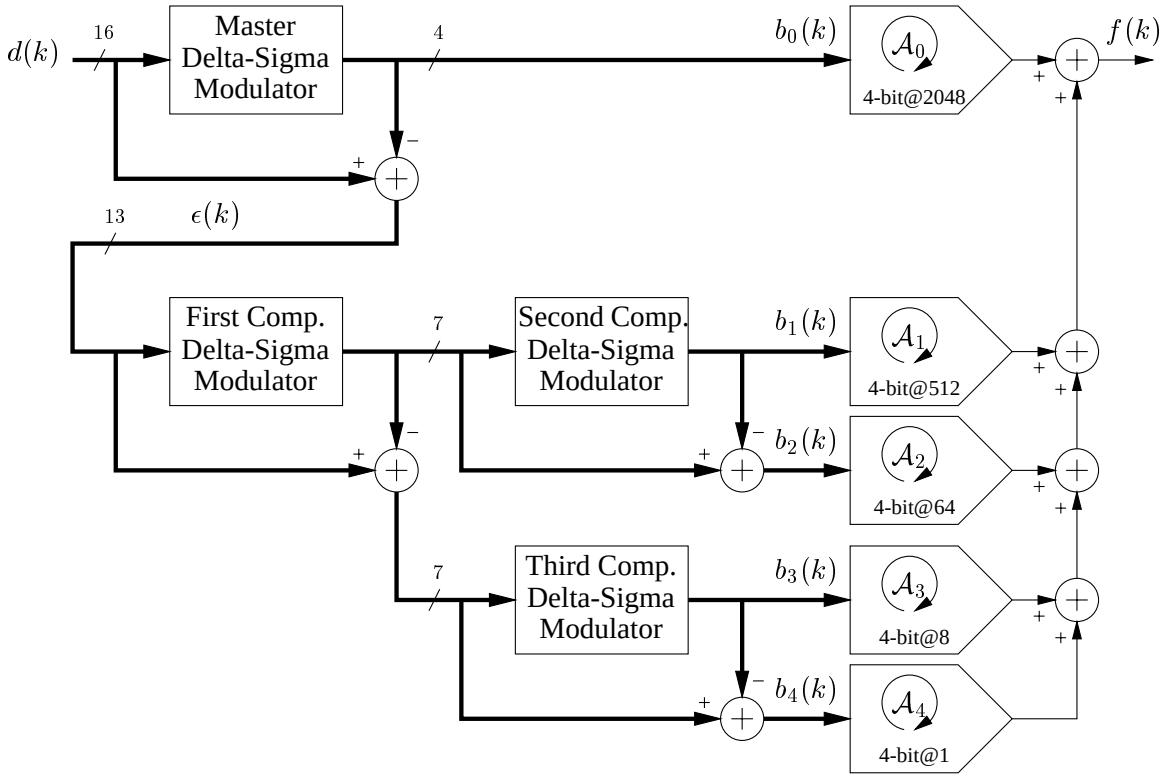


Figure 7.6: Asymmetrical tree-structure scaled-element mismatch-shaping DAC.

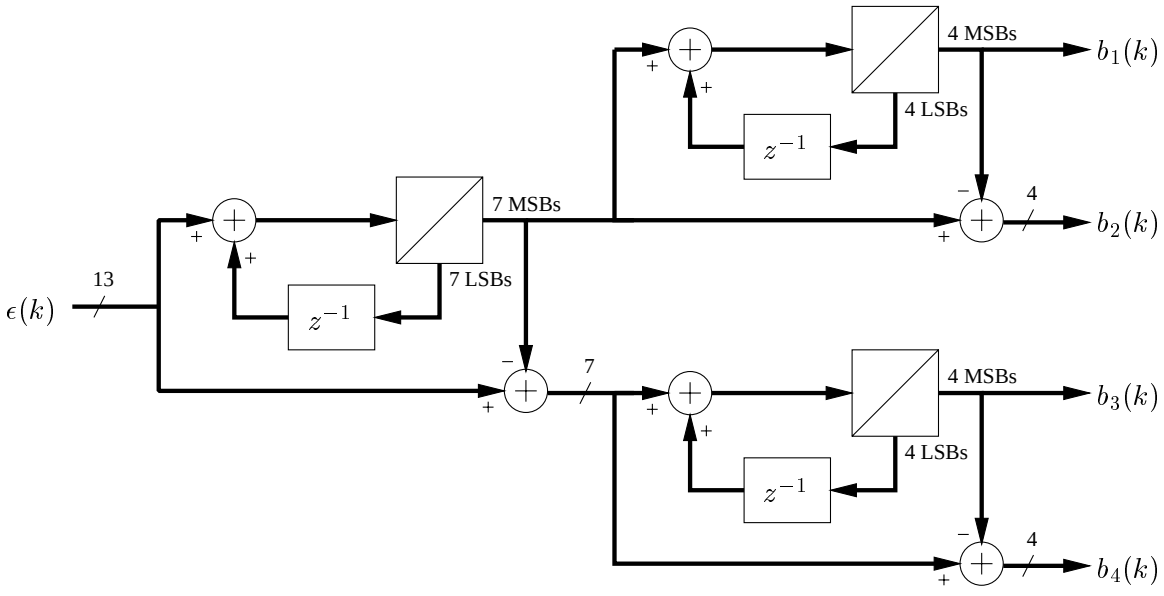


Figure 7.7: Implementation of the compensation DAC's spectral encoder (cf. Figure 7.6).

7.3.2 One-Sided Tree-Structure

Figure 7.8 illustrates yet another way to design spectral encoders. In this case, only one side of the tree structure has been "expanded." Notice the very regular topology, and that the signals' resolution is reduced gradually.

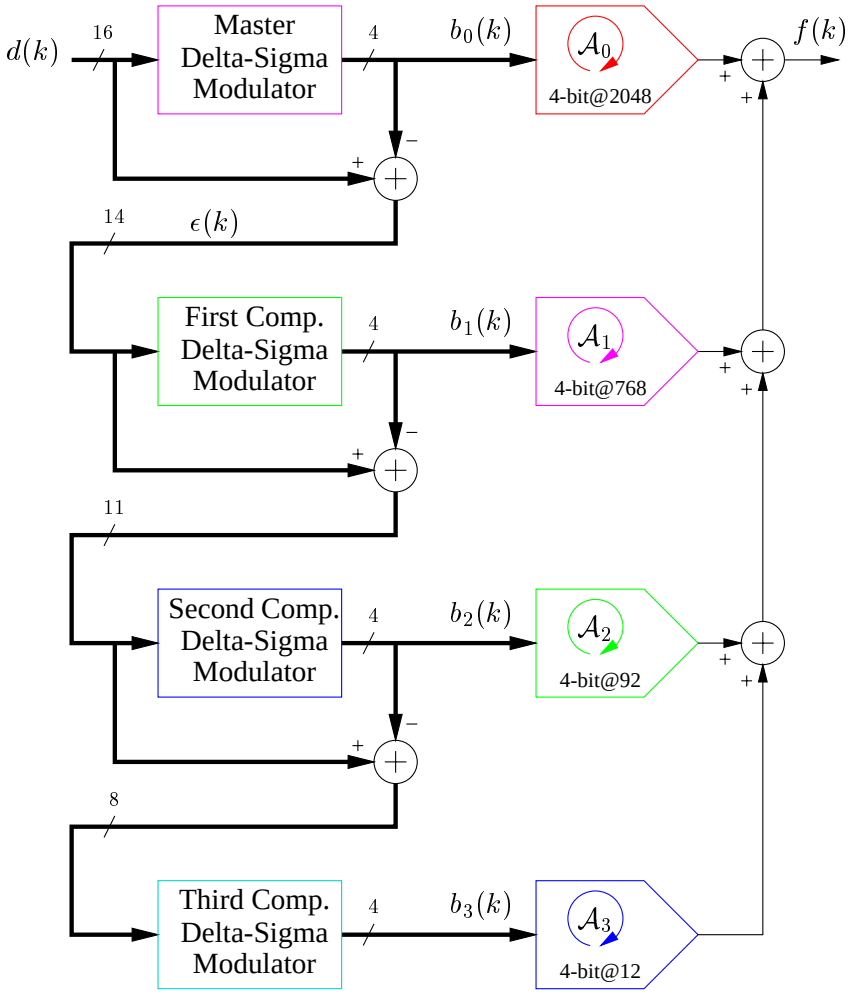
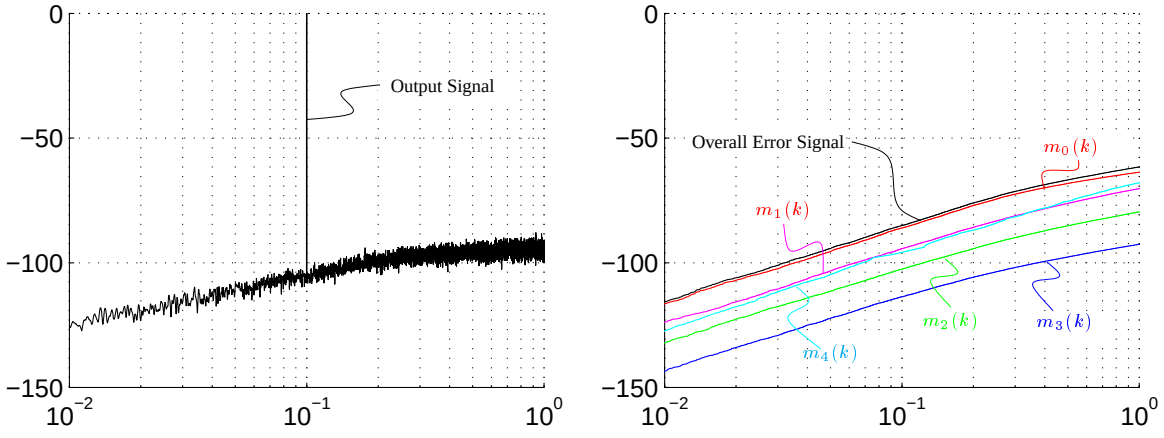


Figure 7.8: One-sided tree-structure scaled-element mismatch-shaping DAC.

Simulated Performance. The one-sided tree-structure has been simulated for the following conditions. The input was a 10-times oversampled sinusoid of magnitude $\pm 0.7 \cdot 2^{15}$, which defines 0 dBFS in Figure 7.9. The total-area matching index was assumed to be 0.1%, and each element's variance was

assumed to be inversely proportional to the element's value relative to the sum of all the elements. The master $\Delta\Sigma$ modulator was of second order ($\text{NTF}_{\max} = 1.5$) and the three other $\Delta\Sigma$ modulators were of first order ($\text{NTF}_{\max} = 2$). $\epsilon(k)$ was represented by 14 bits to avoid unintentional truncation, but since it would not span the whole range, the elements in $\mathcal{A}_1$ were chosen 6 and not 4 times smaller than the elements in $\mathcal{A}_0$[13].

The black traces in Figure 7.9 show the simulated overall performance, whereas the colored traces show the combined gain and nonlinearity errors from the individual DACs (the colors are chosen according to the color code used in Figure 7.8). The master DAC's nonlinearity error limits the performance in the entire frequency spectrum, and 100 dB performance was obtained at about 30 times oversampling. Notice that the truncation error caused by the third compensation $\Delta\Sigma$ modulator is not dominating (cyan trace $m_4(k) = b_4(k)K_0$, where $b_4(k) = d(k) - \sum_{i=0}^{3} b_i(k)$).



(a) Spectral power density versus frequency

(b) Signal-band power versus 1/OSR

Figure 7.9: Performance of the one-sided tree-structure SE-MS DAC shown in Figure 7.8.

[13]The simplest way to implement this technique is to multiply $\epsilon(k)$ by 3/2 (or 5/4 for a more aggressive design) and set the gain of the compensation DAC to 2/3 (or 4/5) of the master DAC's gain K_0, simply by scaling the elements accordingly.

7.4 Filtering Scaled-Element Mismatch-Shaping DACs

The following refers to Figure 7.2. The design of the master DAC is a separate issue, but generally it is preferable to implement the compensation DAC as simple as possible, which usually implies that the compensation signals should be of very low resolution. A tree-structure spectral encoder can be used to separate the truncation signal $\epsilon(k)$ to almost any low resolution (two bits), but doing so is a tradeoff between the spectral encoder's complexity (cf. Figure 7.7) and the UE-MS encoders' complexity (cf. Figures 4.10 and 6.10). This section will propose an even simpler *filtering* spectral encoder, where the separation of $\epsilon(k)$ into 3-level compensation signals is very simple to implement.

The Filtering Principle. In section 7.1.4 it was discussed that the compensation signals should be of the form (cf. Equation (7.6))

$$b_i(k) = h^{-1} * n_i(k), \quad i \neq 0 \tag{7.9}$$

and in Figure 7.2 it can be observed that $\epsilon(k) = h^{-1}(k) * n_0(k)$, where $h^{-1}(k)$ is the impulse response of $1/H(f)$. Because filtering is a *linear* operation, it follows that

$$n_0(k) = \sum_{i=1}^{P-1} n_i(k) \qquad \text{and} \qquad \epsilon(k) = h^{-1}(k) * n_0(k) \tag{7.10}$$

$$\Downarrow$$

$$\epsilon(k) = \sum_{i=1}^{P-1} b_i(k) \qquad \text{and} \qquad b_i(k) = h^{-1}(k) * n_i(k), \quad i \neq 0 \tag{7.11}$$

which means that if $n_0(k)$ is separated *in any way* into a set of signals $n_i(k), i \neq 0$, and these signals are filtered individually by $1/H(f)$, then the outcome will be a set of compensation signals $b_i(k)$ with the desired property[14] (7.9). To simplify the circuit, the compensation signals should preferably be of low resolution. This can be obtained, for example, if $n_i(k), i \neq 0$, are single-bit signals and $h^{-1}(k)$ is a first-order difference filter. The following discussion will provide several examples of the use of this very powerful concept.

[14]For the system to be implementable, $h^{-1}(k)$ must be a causal filter, which requires that $H(f)$ be non-delaying. The feedback loop, however, must include one sample of delay for the $\Delta\Sigma$ modulator to be stable. This delay is best inserted after the loop filter $H(f)$ and before $n_0(k)$ is added to $d(k)$. See the main text for several examples.

7.4.1 Minimalist Scaled-Element Mismatch-Shaping Encoder

The simplest approach to use the filtering principle is to design the master $\Delta\Sigma$ modulator with a first-order loop filter and separate $n_0(k)$ bitwise. Figure 7.10 shows this system implemented using the error-feedback topology. The master signal $b_0(k)$ is here chosen as a single-bit signal, hence mismatch errors will originate only from the compensation DAC. The compensation signals $b_i(k), i \neq 0$, are 3-level signals generated as the first-order difference of the bit-wise separated signal $n_0(k)$. Figure 7.11 shows how each branch of the compensation DAC can be implemented; clearly, the overall complexity is low. Notice that the DAC elements consist of two nominally identical binary-scaled arrays of analog sources (less one element).

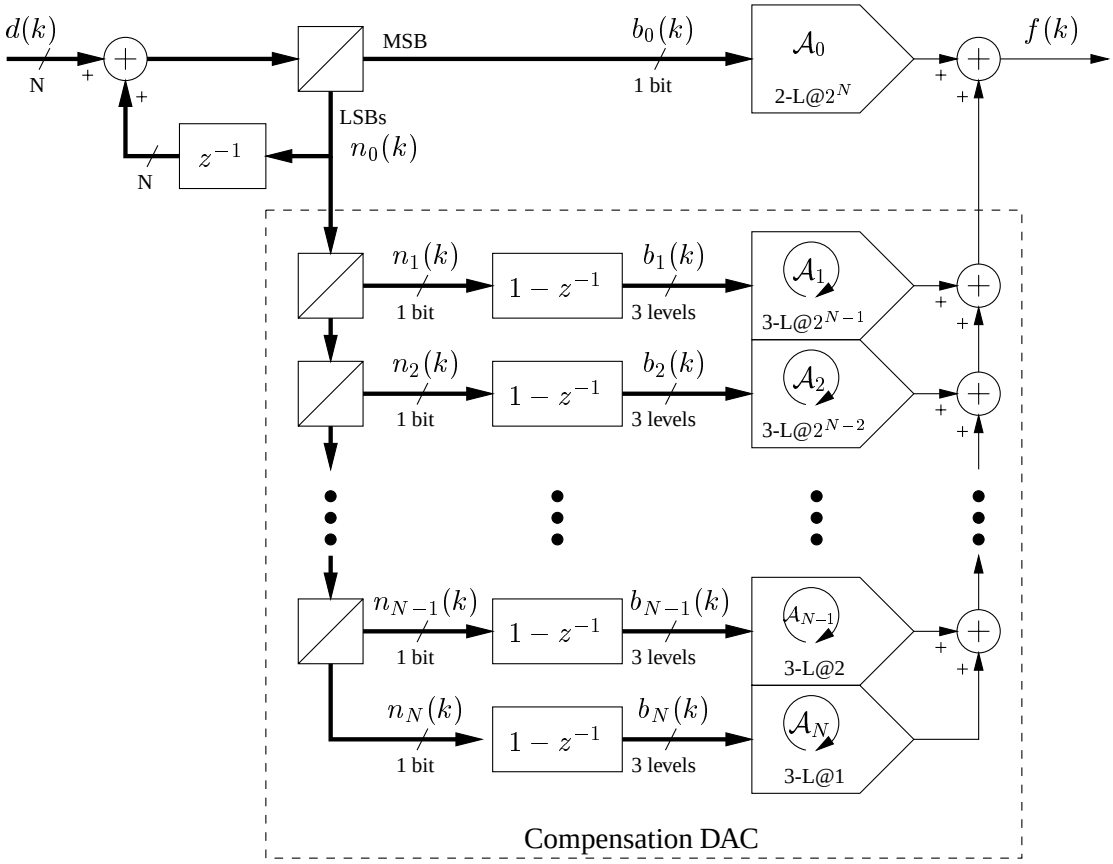


Figure 7.10: Minimalist first-order binary-scaled-elements mismatch-shaping DAC.

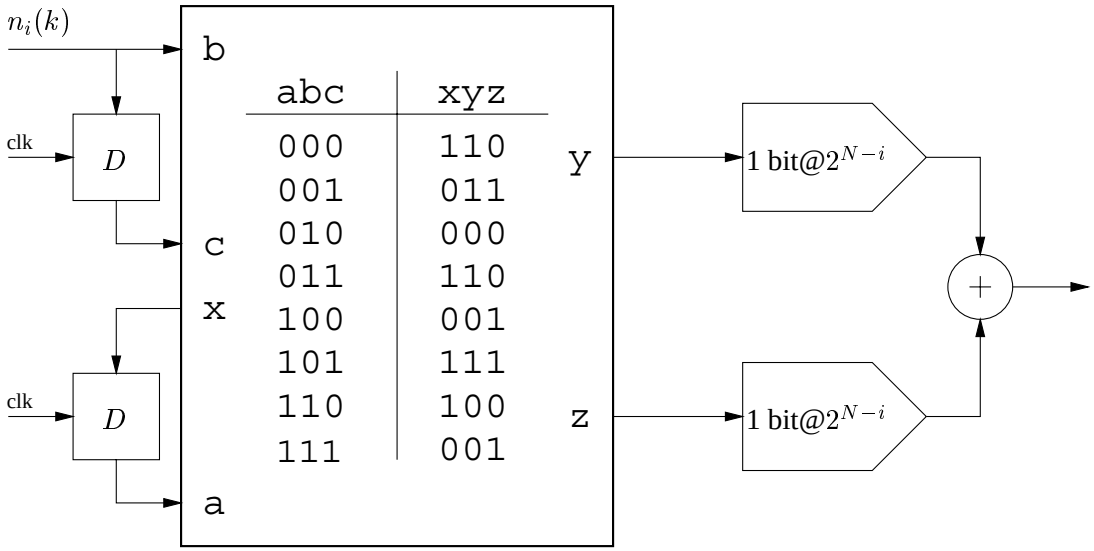


Figure 7.11: State machine implementing first-order differentiation and mismatch shaping.

7.4.2 Practical Filtering Scaled-Element Mismatch-Shaping DACs

The SE-MS DAC shown in Figure 7.10 may be simple, but the topology is not the most suitable for the implementation of high-performance DACs. The problem is that the first-order master $\Delta\Sigma$ modulator is almost certain to generate idle tones, and gain error will make them leak to the output $f(k)$. As usual, this problem can be avoided by increasing the order of the loop filter; but to minimize the resolution of the compensation signals $b_i(k), i \neq 0$, first-order difference filtering of $n_i(k)$ is preferable. These two requirements can be fulfilled simultaneously when the spectral encoder is designed as shown in Figure 7.12. The basic idea is that the loop filter is realized as a cascade of two stages, where the first stage has the desired transfer function and the second stage is designed to keep the $\Delta\Sigma$ modulator stable. The loop filter's $\mathrm{NTF}_{\max}$ value should be fairly low (say 1.5) to assure that the full-scale range of $n_0(k)$ is reasonably small. The full-scale range of $n_0(k)$ is proportional to the step size of the master DAC. For a first-order loop filter (cf. Figure 7.10) $n_0(k)$ will be as large as the master DAC's step size, but for higher order loop filter's it may be up to about 1.5 times larger (for low values of $\mathrm{NTF}_{\max}$). Although it usually is not necessary, the presented design incorporates a safety factor (to prevent unintentional truncation) by using elements of the same value in $\mathcal{A}_0$ and $\mathcal{A}_1$. The design can be improved slightly by increasing the master DAC's step size by about one third to $\frac{4}{3}2^{N-1}$. The $\Delta\Sigma$ modulator must then be

modified accordingly, but that is a simple matter.

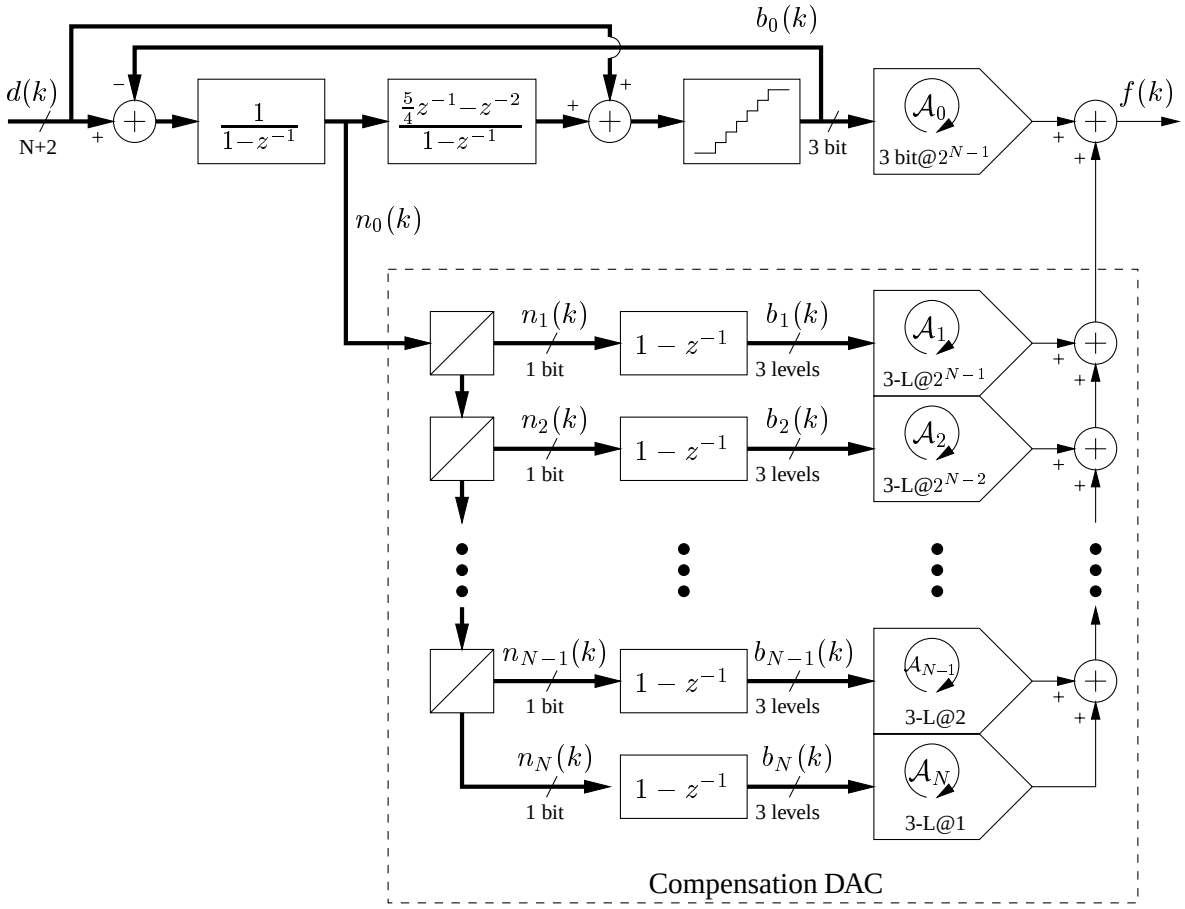


Figure 7.12: Improved SE-MS DAC that is less subject to gain-error idle tones.

7.4.3 Reducing the Gain-Error Sensitivity

Usually, it is the master DAC's error signal that limits the performance. This is, however, not the case when the master DAC is linear, e.g., if $b_0(k)$ is a single-bit signal, in which case the compensation DAC's error signal will dominate. Because this design approach bears some interest (discussed later), it is worth while to consider how the compensation DAC's error signal can be minimized.

The compensation DAC's error signal is the sum of the internal DACs' error signals $m_1(k)$, $m_2(k)$, ... , $m_N(k)$. Each of the error signals consist of two parts: the gain error $b_i(k)[K_i - K_0]$ and the local

mismatch error. The local mismatch errors will almost always be first-order shaped (when first-order UE-MS encoders are used). The gain errors are, however, likely to be larger[15], because they depend on global matching of electrical parameters, and because the individual DACs often will be implemented with some physical distance. It is, therefore, preferable if the compensation signals $b_i(k)$ are more-than-first-order shaped, i.e., of the form (7.9), where $h^{-1}(k)$ is a higher-order filter (cf. Figure 4.11).

Clearly, second-order shaped compensation signals $b_i(k)$ can be obtained by designing the master $\Delta\Sigma$ modulator's first filter stage as a second-order integrator (cf. Figure 7.12) and employing second-order difference filters in the compensation DAC. This is, however, not a good idea. The problem is that the magnitude of $n_0(k)$ will increase (even for the same $\mathrm{NTF}_{\max}$ value), and the second-order difference filters will further increase the magnitude of the compensation signals by a factor of two. This way, the reduction of the gain error's signal-band power, due to a higher-order shaping of $b_i(k)$, is lost because the Nyquist-band power of the error signals is increased. This is especially a problem for the local mismatch errors, which will remain first-order shaped.

Fortunately, there are other and better ways to higher-order shape the compensation signals. In Figure 7.12, $n_0(k)$ is first-order shaped (because it is the input signal to an integrator for which the output signal is bounded), but the spectral composition is immediately destroyed by splitting the signal bitwise. Figure 7.13 shows how the spectral encoder can be modified such that $n_1(k)$ inherits the first-order-shaped property of $n_0(k)$. The property is inherited, because $n_1(k)$ is the difference between $n_0(k)$ and the first-order difference of $n_0^*(k)$, which are both first-order shaped[16].

The Nyquist-band powers of the error signals are largely proportional to the square root of the corresponding DACs' step size. Hence, if $\mathcal{A}_1$ includes (say) 8 elements, it is mainly the gain error of $m_1(k)$ that is of concern. In other words, for simplicity, it would be better if the secondary compensation signals, $b_2(k), b_3(k), \ldots, b_N(k)$, could be of only 3-level resolution, even if that were to imply that they would be only first-order shaped. This simplification can be obtained, for example, by differentiating $n_0^*(k)$ *before* it is split bitwise (not shown). An even simpler technique to obtain the same result is shown

[15] Nyquist-band power.

[16] Except for the differentiation of $n_0(k)$, the compensation DAC is identical to the minimalist SE-MS DAC shown in Figure 7.10).

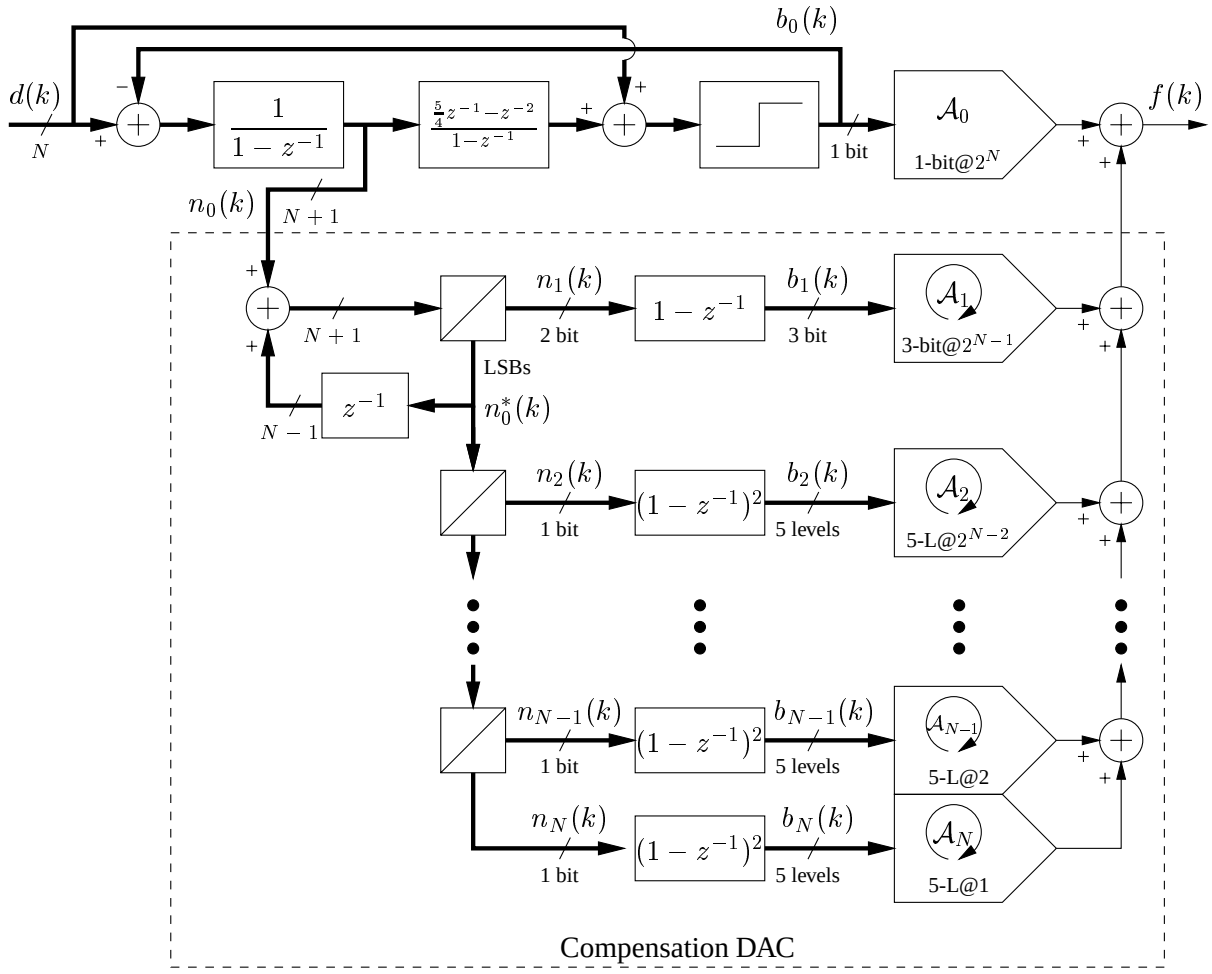


Figure 7.13: SE-MS DAC where the gain-errors are second-order shaped.

in Figure 7.14. In essence, the topology is equivalent to that shown in Figure 7.3, where the compensa-

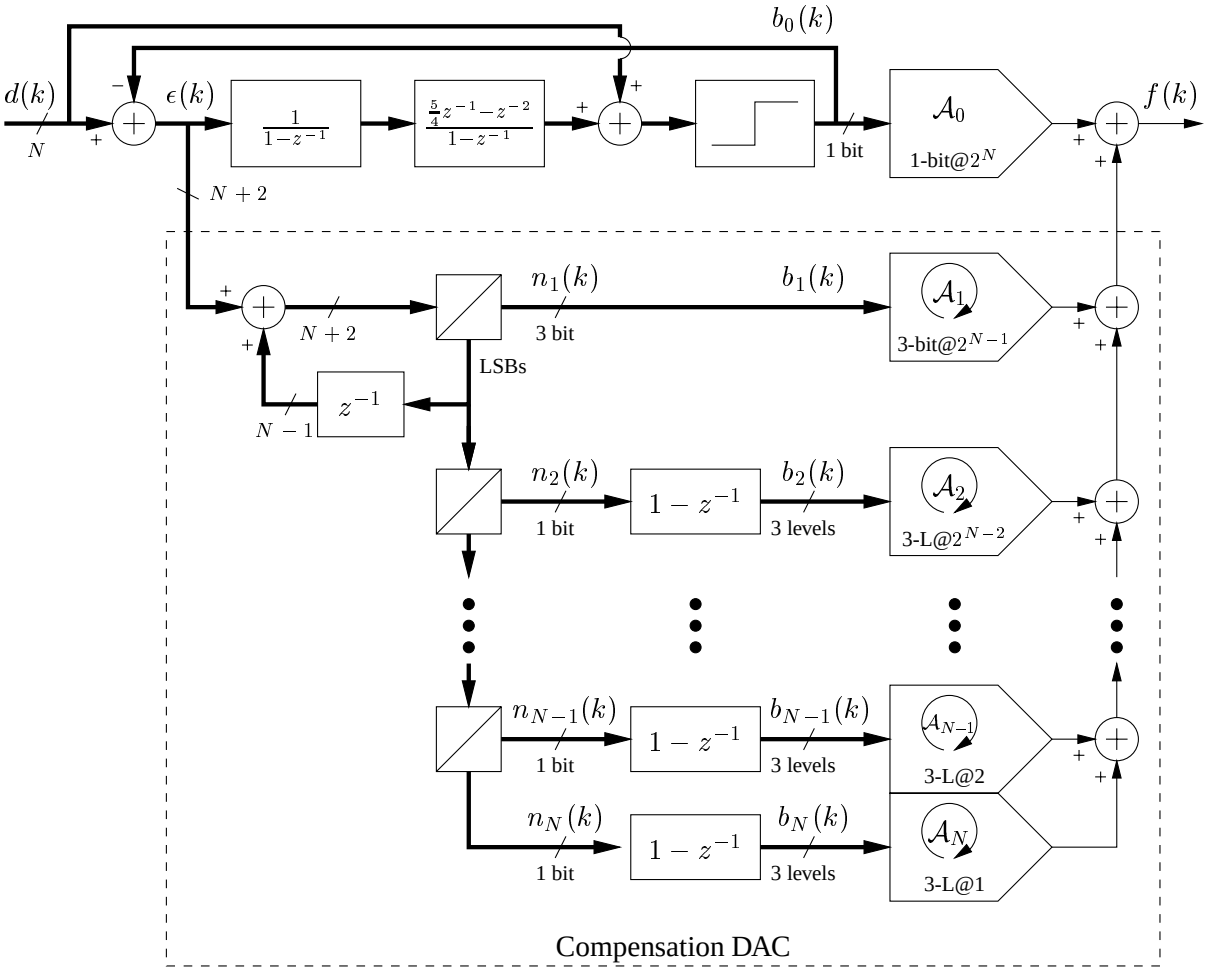


Figure 7.14: Alternative to Figure 7.13 where the compensation signals are shaped according to their magnitude.

tion DAC is implemented as the minimalist SE-MS DAC shown in Figure 7.10. The main compensation signal $n_1(k)$ will inherit the spectral composition[17] of $\epsilon(k)$ because gain errors within the compensation DAC can be referred to the secondary compensation signals $b_2(k), b_3(k), \ldots, b_N(k)$. In principle, a few of the least significant compensation signals can be gathered in one signal (of say 4-bit resolution) and

[17] In this case, it is second-order shaped.

D/A converted with a non-mismatch-shaping DAC. The local mismatch error from this DAC will not be shaped, but be a very small pseudo-white-noise signal (the gain error remains first-order shaped).

7.5 Second-Order Scaled-Element Mismatch-Shaping DACs

In principle, it is very simple to design second-order SE-MS DACs; just use second-order UE-MS DACs and assure that the compensation signals are at least second-order shaped (e.g., using any tree-structure topology with second-order $\Delta\Sigma$ modulators, or an appropriate filtering structure, such as that shown in Figure 7.13). This approach is, however, not too attractive, because second-order UE-MS shaping encoders are complicated to implement and only truly effective if the OSR is 25 or larger. The purpose of increasing the order of the mismatch shaping is typically to allow the use of lower OSR, hence the second-order structures discussed above are only relevant in extreme situations (or if better and simpler second-order UE-MS encoders are invented). The following will discuss a technique by which effective second-order mismatch shaping can be obtained *without* using second-order UE-MS encoders.

7.5.1 The Generalized Filtering Principle

Consider again the SE-MS DAC shown in Figure 7.13. If, for example, the master signal $b_0(k)$ is of 1-bit resolution, the master DAC can be made linear. In that case, it is entirely the compensation DAC that limits the system's performance. The compensation DAC's gain errors can easily be made second- or even higher-order shaped (cf. Section 7.4.3), but in the absence of good second-order mismatch-shaping encoders, the compensation DAC's local nonlinearity errors will be only first-order shaped. If, however, the compensation DAC's output is filtered (differentiated) *in the analog domain*, the local nonlinearity errors will be shaped also by this filter's transfer function. This way, efficient second- or higher-order mismatch-shaping is made feasible; it can be implemented as shown in Figure 7.15.

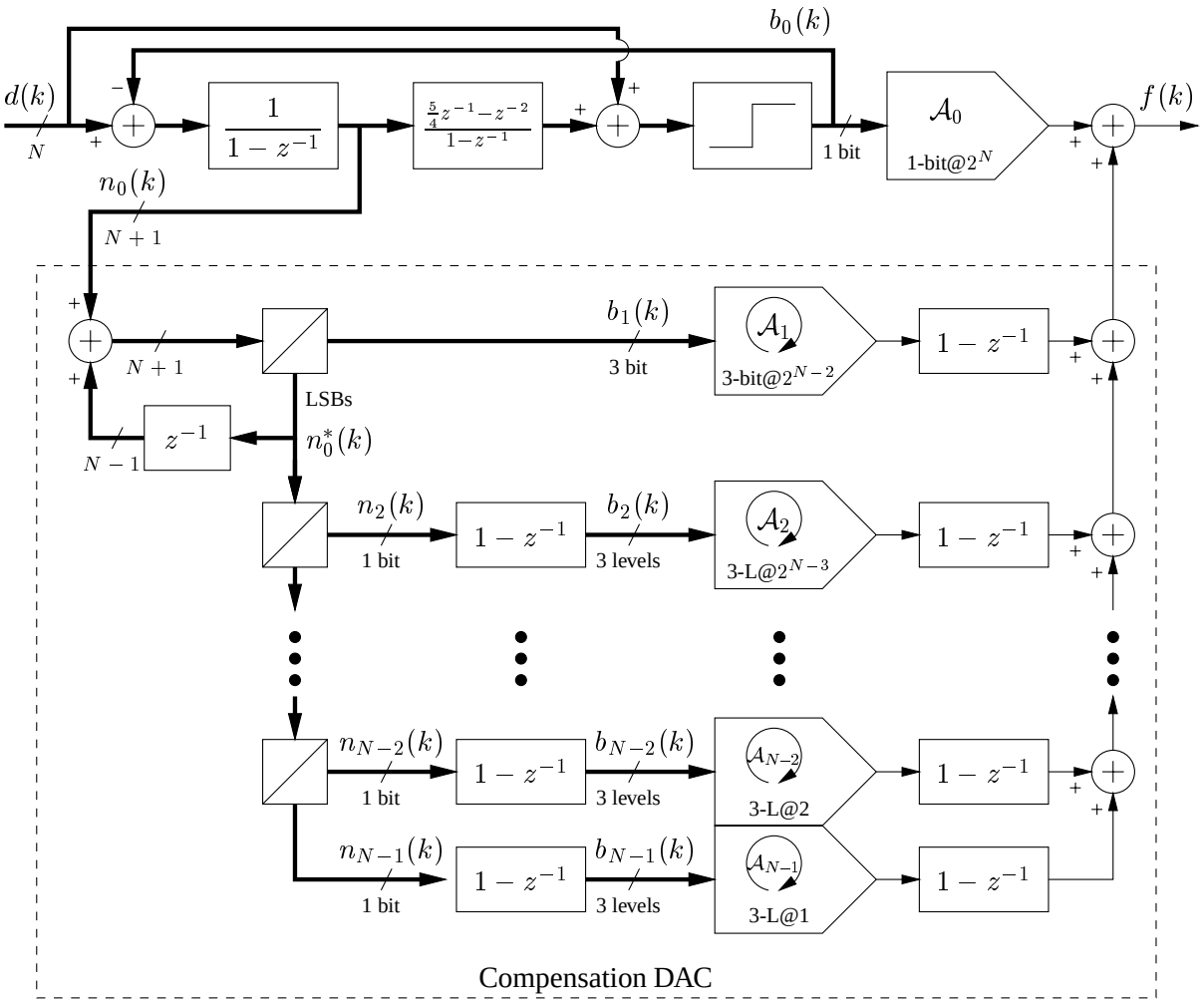


Figure 7.15: Second-order filtering SE-MS DAC employing analog filters.

7.5.2 The Filter-Mismatch Problem

The generalized filtering principle has similarities to the MASH structure [1] and its cousin – the dual-quantization DACs [60]. Since the performance of these circuits is critically dependent on matching an analog filter's transfer function to the (master) $\Delta\Sigma$ modulator's (noise) transfer function, a discussion of this issue is found to be appropriate.

The MASH structure was discussed[18] on page 89, and it was concluded that the $\Delta\Sigma$ modulator generally should not be chosen of an order higher that two, because the impact of mismatch of the analog/digital filters' transfer functions on the system's performance will increase with the filters' order. The argument that filter mismatch is a problem[19] even for only second-order systems is supported by the many circuit-level techniques developed to improve the matching [60].

The reason why filter mismatch is much less of a problem when using the generalized filtering principle is that the performance, even for high-order master $\Delta\Sigma$ modulators, does not rely on matching of high-order filters. Consider Figure 7.15. It can be observed that the relation $d(k) = b_0(k)+[n_0(k)-n_0(k-1)]$ is a consequence only of $n_0(k)$ is the accumulation of $d(k) - b_0(k)$. In other words, the relation is *completely independent* of the second filter stage of the master $\Delta\Sigma$ modulator[20]. Hence, the analog filter is to match only the master $\Delta\Sigma$ modulator's first filter stage (which typically is of low order), and *not* the modulator's high-order noise transfer function. As opposed to MASH and dual-quantization DACs, the signal-band performance of generalized filtering DACs will *improve* when the master $\Delta\Sigma$ modulator's order is increased[21], because $n_0(k)$ will have increasingly less signal-band power. Consequently, the cancellation process need not be relied upon in the signal band. In the extreme case, such as for very poor filter matching and master $\Delta\Sigma$ modulators of high order, the compensation DAC can be construed as a simpler substitute for the smoothening filter shown in Figure 3.9. However, usually it is possible to obtain quite good matching of low-order filters, whereby state-of-the-art performance can be obtained

[18] Only the MASH *quantizer* structure was discussed, but the filter-matching issue is the same for MASH DACs.

[19] It is only a problem if you aim at high (say more than 85 dB) performance.

[20] The second filter stage affects mainly the magnitude (through $\mathrm{NTF}_{\max}$) and the spectral composition (through the filter's order) of $n_0(k)$.

[21] Assuming that the master $\Delta\Sigma$ modulator's first filter stage and the analog filter remains the same.

even when using low-order master $\Delta\Sigma$ modulators.

7.5.3 Variations

Clearly, the signals from the compensation DAC's sub DACs can be filtered either before or after they are summed[22]. In a minimalist filtering DAC, see Figure 7.16, $n_0(k)$ will be converted by an ordinary (e.g., binary-weighted) DAC and the output filtered and added to the output of the D/A converted master signal. Because $n_0(k)$ is almost uncorrelated with $d(k)$, and because $n_0(k)$ has a broadband spectral composition, nonlinearity of the compensation DAC will cause a white-noise-like error signal which will be filtered by $1/H_1(f)$ and which, therefore, will appear shaped in the output signal $f(k)$. Hence, the minimalist filtering DAC can be interpreted as a (UE-MS-encoder-free) mismatch-shaping DAC, and it can in principle be implemented of any high order. Ultimately, the reason why it makes most sense to implement the generalized filtering DAC with a mismatch-shaping compensation DAC (i.e. as shown in Figure 7.15) is because it is very simple to do so, and because the magnitude of $n_0(k)$ and the required order of $H_1(f)$ will be lower (and the circuit, therefore, better and simpler).

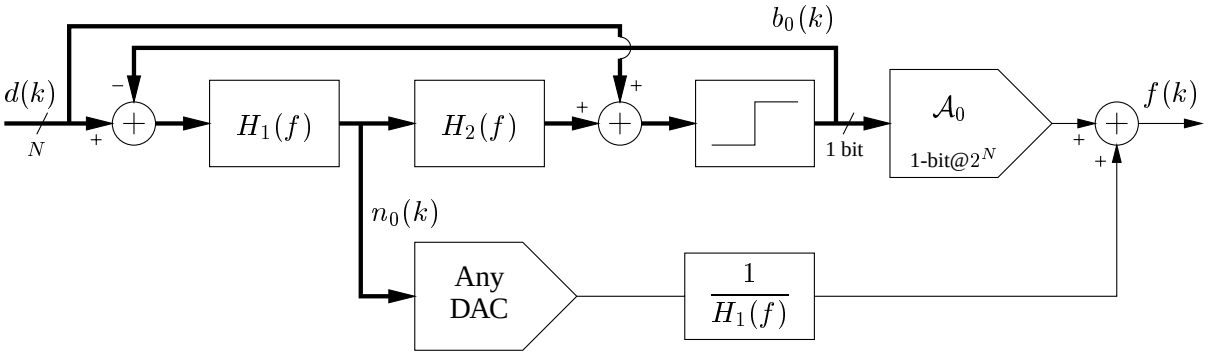


Figure 7.16: Minimalist DAC implemented according to the generalized filtering principle.

[22]Or in groups, if that for some reason is preferable. For switched-capacitor implementations, it is almost always preferable to implement the first differentiation separately for each DAC element (cf. Figure 7.17).

7.5.4 Switched-Capacitor Implementation

Figure 7.17 shows a switched-capacitor (SC) implementation of the second-order SE-MS DAC shown in Figure 7.15 (the spectral encoder is not shown).

The analog output signal $f(k)$ is represented as charge pulses dumped to the opamp's virtual-ground node, and it is low-pass filtered and converted into a voltage signal $v_{\rm out}(k)$ by the switched-RC feedback network. The linearity of the D/A conversion does not particularly depend on the ideality of the provided virtual ground potential, but (of course) the SC circuit must meet the overall performance requirements. The summation of the individual charge-pulse signals $c_i(k)$ will be ideal (charge conservation).

The master DAC will (in clock phases Φ_1 and as a function of $b_0(k)$) provide charge pulses of $\pm V_{\rm ref}C$, hence the system's gain is

$$K_d = K_0 = \frac{2V_{\rm ref}C}{2^N} = \frac{V_{\rm ref}C}{2^{N-1}} \tag{7.12}$$

It is important that $c_0(k)$ is in linear relation with $b_0(k)$, hence the usual precautions for the implementation of single-bit SC DACs' should be observed. It is particularly important that the load seen by the reference voltage is *independent* of $b_0(k)$ (see [1] for a detailed discussion). Also, the Haigh-and-Singh delayed-clock-phases clocking scheme should (as always for high-performance SC circuits) be employed to prevent signal-dependent charge injection and clock feedthrough [29].

The primary compensation DAC is implemented by a UE-MS driver and 8 nominally-identical capacitors[23] of capacitance $C/4$. The capacitor terminals that are connected to the virtual-ground node are not switched, hence the charge-pulse signal $c_1(k)$ will be the first-order difference of the weighted voltage signal provided by the UE-MS driver. Each capacitor is driven by a single-bit logic signal (with the step size of 2^{N-2}) which is buffered to $\pm V_{\rm ref}$ (as shown for $b_0(k)$ in the master DAC). Hence, each capacitor will provide charge pulses of either $-2V_{\rm ref}C/4$, 0, or $2V_{\rm ref}C/4$, and the gain of this DAC will be

$$K_1 = \frac{2V_{\rm ref}C/4}{2^{N-2}} = \frac{V_{\rm ref}C}{2^{N-1}} \tag{7.13}$$

[23] 6 capacitors would in principle be enough, because $b_1(k)$ does not span the full range. 8-element UE-MS encoders are, however, particularly simple to implement.

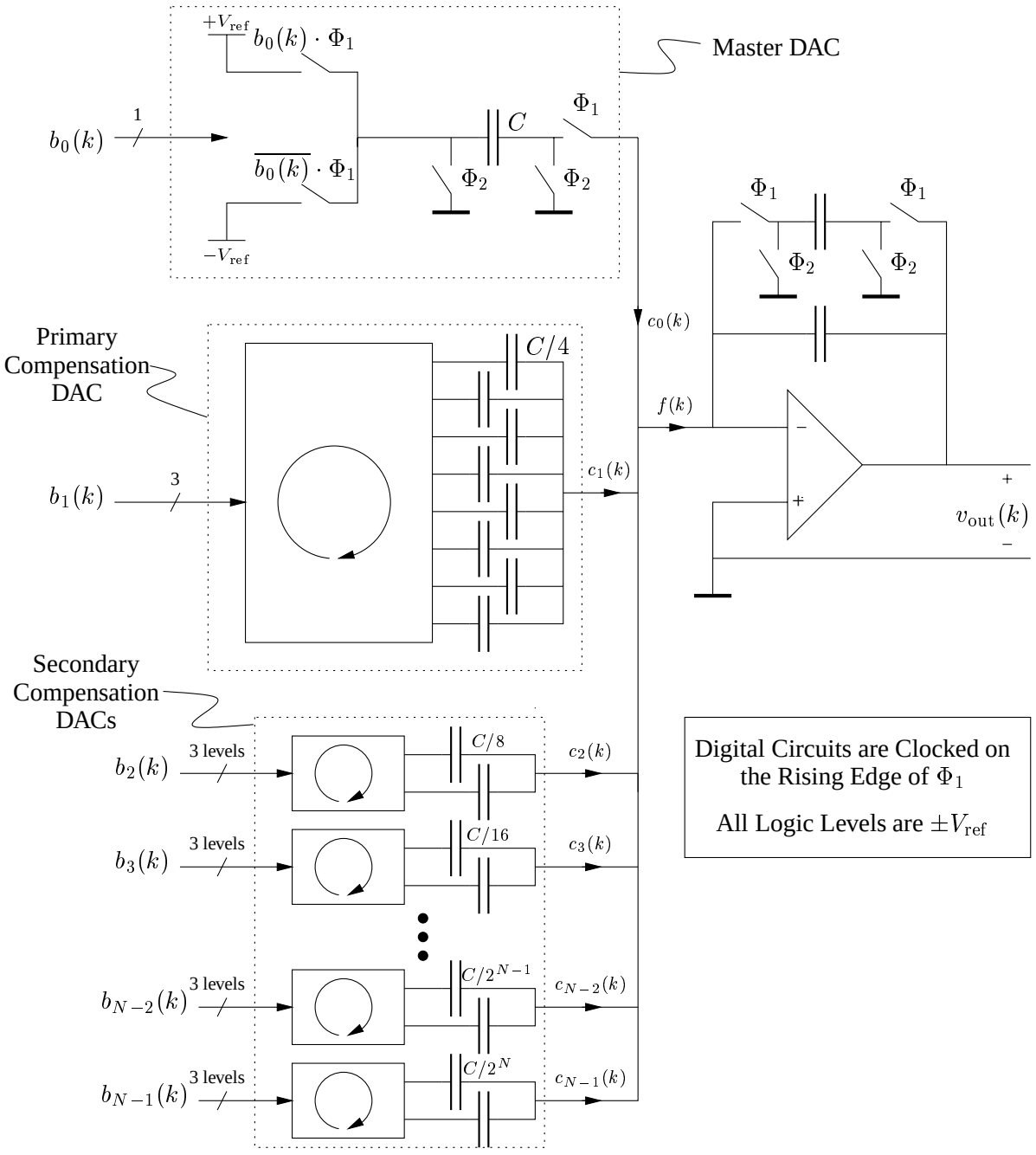


Figure 7.17: Switched-capacitor implementation of the analog part of the second-order filtering SE-MS DAC shown in Figure 7.15.

which conforms with the master DAC's gain.

The analog-domain differentiation will match the digital-domain integration exceptionally well, because the ideality of the analog filter relies only on the electrical insulation of each capacitor's two terminals. In other words, unless the sampling frequency is *extremely* low and the capacitors' leakage current large (e.g., caused by high-temperature operation or a very bad technology), the analog filter can be assumed ideal for all practical purposes.

The secondary compensation DACs are implemented similar to the primary compensation DAC.

Simulation Results. The circuit was simulated (with MATLAB) using the assumptions outlined on page 180. The results are shown in Figure 7.18. The black traces show the overall performance. The error signal is second-order shaped (the signal-band error power decreases 50 dB per decade of oversampling), and it is observed that 100 dB performance is obtained at 10 times oversampling (the target for this work). Because the error signal is higher-order shaped, the OSR required for 100 dB performance will be less dependent on the technology's matching index.

The master DAC is linear, hence it produces no local nonlinearity error. The master DAC, however, defines the system's gain K_d, which should carefully be matched to the (primary) compensation DAC's gain to minimize gain errors. The primary compensation DAC's combined gain and local nonlinearity error is shown with green traces. This is the dominating error source, which it should be because the primary compensation signal $c_1(k)$ is somewhat larger than the secondary compensation signals. The combined error signal from the secondary compensation DACs is shown with blue traces (only the 8 most-significant secondary compensation DACs were included). Notice the small gain-error idle tones that occur at high frequencies. These tones are so small that they usually do not represent a problem, but it is interesting to observe that they indeed do exist.

In comparison with the previous shown simulation results (Figures 7.4 and 7.9), the error signal's Nyquist-band power is approximately 6 dB larger. This is because the primary compensation DAC, which produces the dominating error signal, can produce a signal which is 6 dB larger than a full-scale output signal. This is in itself not a major problem (because the error signal is second-order shaped);

but since the magnitude of $c_1(k)$ is proportional to the master DAC's step size, 6 dB improvement (in the entire frequency spectrum) can be obtained if a linear 3-level master DAC can be realized (discussed below).

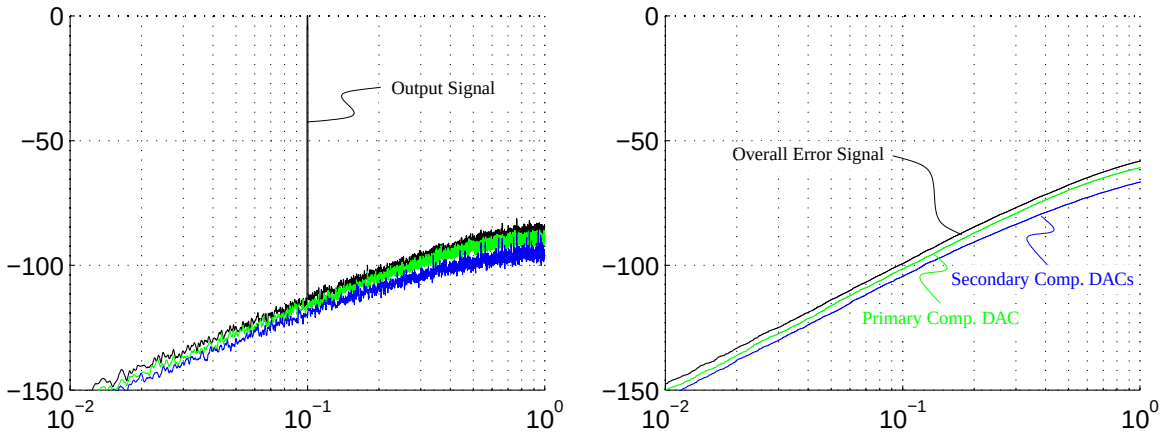
To facilitate comparison, Figure 7.19 shows the equivalent simulation results for the system shown in Figure 7.13. Here, the first-order-shaped local nonlinearity errors are dominating, hence the performance is not as good as in Figure 7.18. The error signal's Nyquist-band power is equally high, so the performance is not even as good as for the other proposed first-order shaping systems (compare Figures 7.9 and 7.19). As discussed above, this first-order system may, however, perform better than the other proposed first-order systems because the gain errors are second-order shaped. In actual implementations, gain errors are likely to dominate the local nonlinearity errors (because of physical distance in the layout), but this parameter was not included in the stochastic model used for the simulations.

Figure 7.20 shows the second-order SE-MS DAC's time-domain output signal $f(k)$ for a saw-tooth input signal $d(k)$. Clearly, the error signal is very small, and there is hardly a need for a filter dedicated to suppress it further (for example, for audio applications, the human ear can maintain this function). It should, however, by understood that the signal shown is *highly* oversampled, and that there may be a need for a filter dedicated to suppress replica spectral images (cf. Section 3.2.1).

Experimental Work. A fully-differential version of the SC DAC shown in Figure 7.17 has been implemented in cooperation with MEAD Microelectronics in Switzerland.

The spectral encoder and the primary compensation DAC's UE-MS encoder was implemented using an external FPGA, and the SC circuit was implemented in a 5-Volt 0.8μm CMOS technology. Because the DAC's error signal has only little Nyquist-band power, the voltage signal $v_{\rm out}(k)$ was DT/CT converted directly with a zero-order holding circuit (cf. Figure 3.9). A replication-rejection filter was not implemented.

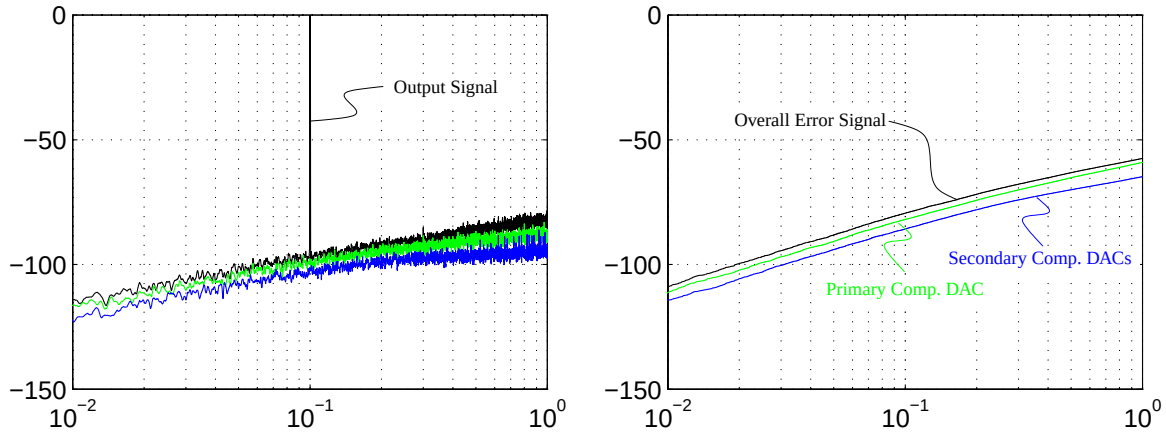
The target performance was 100 dB SNDR at 10 times oversampling with a sampling frequency in the range from 400 kHz to 1 Mhz (i.e., a signal bandwidth of 20 kHz to 50 kHz). The targeted low OSR and high SNR requires that large signal capacitors be used to avoid thermal-noise limitation; the master



(a) Spectral power density versus frequency

(b) Signal-band power versus 1/OSR

Figure 7.18: Simulated performance of the second-order SE-MS DAC shown in Figures 7.15.



(a) Spectral power density versus frequency

(b) Signal-band power versus 1/OSR

Figure 7.19: Simulated performance of the first-order SE-MS DAC shown in Figure 7.13.

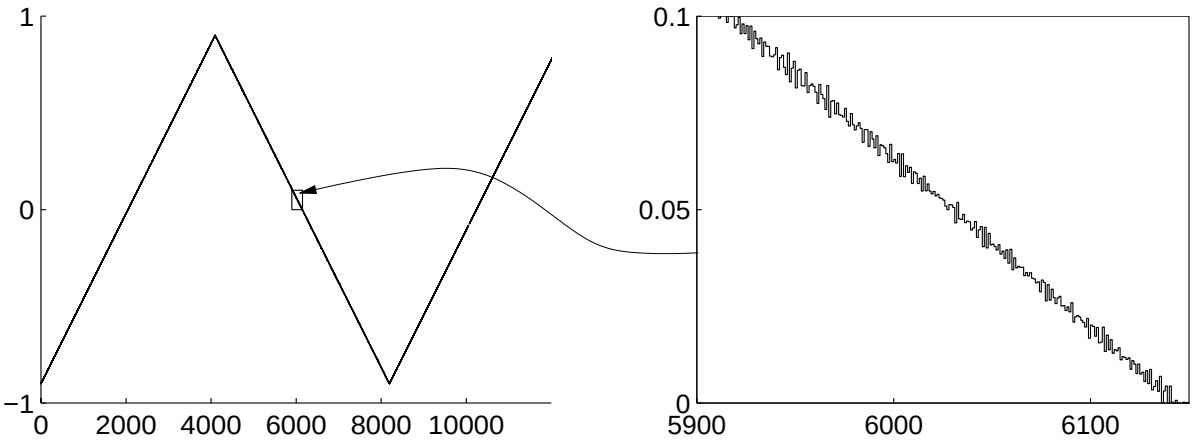


Figure 7.20: Time-domain performance of the second-order SE-MS DAC shown in Figure 7.15.

DAC's capacitor C was designed with a nominal value of 11.25 pF. Although noise calculations for the system lead to a result somewhat similar to that found in Section 4.5.1, there is a main difference due to the compensation DAC's relatively large capacitive load of the opamp's virtual-ground node (cf. Figure 7.17). This implies that the compensation DAC's voltage reference $V_{\rm ref}$ must be very clean[24], which is assured by means of an external passive low-pass filter. The compensation DAC's capacitive load will also cause some magnification of the opamp's input-referred noise, so the opamp had to be designed very carefully.

The layout was optimized with respect to matching the gain of the master DAC to the gain of the primary compensation DAC. However, because the error signal is second-order shaped, matching of the capacitors was one of the least difficult problems to solve; for a 20kHz bandwidth, the OSR can be increased from 10 to 25, which will improve the SER performance by 20 dB. Besides the difficulty in achieving the target noise performance, the most difficult aspect of the design was the DT/CT conversion. At 10 times oversampling, the maximum step size of a full-scale sinusoid at the edge of the signal band will be approximately one third of the supply voltage. It is *very* difficult to design CT analog circuits that are able to handle step transients of this magnitude linearly, but a good solution is believed to have been

[24]The opamp's integrating feedback capacitor was omitted (to facilitate verification of the error signal's assumed second-order-shaped spectral composition), hence the noise on $V_{\rm ref}$ is not shaped or suppressed otherwise and will be subject to aliasing. If this integrating capacitor is included, the noise requirement to $V_{\rm ref}$ will be less stringent.

found. A thorough discussion of this aspect would, however, require a separate volume of this thesis.

The fabrication of the test chip is expected to be completed around January 22, 1999. Test results may be available by the time this work will be defended. Figure 7.21 shows the layout of the test chip.

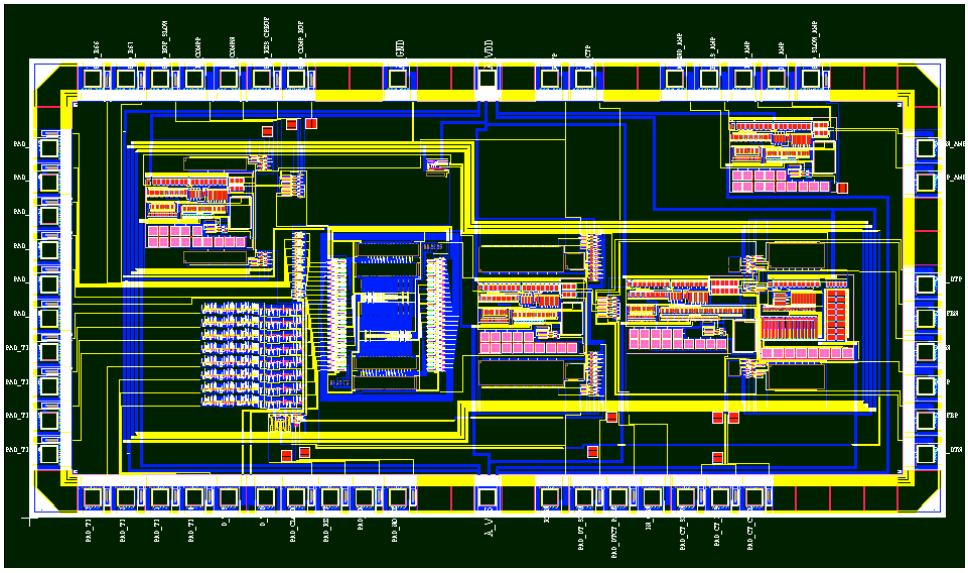


Figure 7.21: Layout of test chip (second-order SE-MS SC DAC).

Improving the Noise Performance. As discussed above, the SC implementation shown in Figure 7.17 is sensitive to the large capacitive load (posed by the compensation DAC) of the opamp's virtual ground node. The opamp can be designed such that its settling time is as good/fast as if the capacitive load had been removed, but the degradation of the noise performance is unavoidable. If the noise performance is required to be as good as 100 dB, the capacitors become impractically large (the active area of the test chip was about 5 mm^2), and the power consumption will increase accordingly. It should be understood that 100 dB performance from any 10-times oversampled sub-5-Volt switched-capacitor circuit will always require use of large signal capacitors (cf. Section 4.5.1), but since the capacitive load of the opamp increases the circuit's sensitivity to the opamp's noise[25] by about 6 dB, it should be minimized if possible.

[25]For a two-stage opamp, the input-referred thermal noise is approximately $\frac{4kT}{3C_i}$, where C_i is the internal frequency-response-compensating (Miller) capacitor. Although C_i can be implemented as a MOSCAP, it will still require a quite large

Because the compensation DAC's relative capacitance[26] is inversely proportional to the master DAC's resolution, increasing the master signal's resolution is a simple way to improve the system's noise performance. If the master signal is multi-bit, the master DAC will generally be nonlinear, and a UE-MS encoder (or other means) will be required to suppress the local nonlinearity error in the signal band. If the master DAC is implemented as a first-order UE-MS DAC (in the absence of efficient second-order UE-MS encoders), the system will be only first-order mismatch-shaping and the OSR, therefore, higher (30 is a typical value). The higher OSR will be acceptable (and sometimes even preferable) for many applications because the capacitors can be made comparably smaller, and the compensation DAC's relative capacitance can be made negligible[27]. The power consumption need not be higher. Other applications, however, are designed to have a high bandwidth, and they rely on a low OSR. Alternatives are required for such applications[28].

It is highly preferable if the master DAC can be designed as a *linear* multi-bit DAC. Several U.S. patents [61, 62] describe the implementation of low-resolution high-linearity DACs, some of which may actually work, but their operation is characterized by averaging a sequence of D/A conversions, which is equivalent to increasing the OSR by a factor of the length of the sequence, and they are, therefore, not of particular interest. However, without using calibration or averaging techniques, it is possible to design linear 3-level DACs, which can be used to bring the compensation DAC's relative capacitance down by a factor of two, and that is enough to improve the performance considerably. The design of 3-level DACs will be discussed in the next section.

---

area to fulfill $C_i > 10C \simeq 100$ pF (to compensate for the 6 dB degeneration). A large C_i capacitance (30 pF was used) will also affect the opamp's slew-rate performance and its power consumption. To minimize the power consumption, the test chip was designed such that the period in which the opamp is slewing may as long as half the settling period.

[26] I.e., the compensation DAC's capacitance (in Figure 7.17, it is approximately 10/4 C) relative to the master DAC's capacitance (in Figure 7.17, it is C).

[27] Notice that the simplest method is to use the generalized filtering principle, and that it may not be necessary to use UE-MS encoders for the compensation DAC (or perhaps only for the primary compensation DAC). In the simplest case, the DAC can be implemented as shown in Figure 7.16, where $b_0(k)$ is multibit and $1/H_1(f) = 1 - z^{-1}$.

[28] The problem is easy to solve if the master DAC can be calibrated (which is a perfectly good solution), but in the following, that will not be considered as an option.

7.5.5 Linear Three-Level DACs

Linear three-level DACs represent a variety of options to improve the discussed DAC designs. For example, the minimalist filtering DAC (cf. Figure 7.10) can be implemented without UE-MS encoders[29], and so can the improved version shown in Figure 7.12 if the master signal is a two- or three-level signal. As discussed above, 3-level DACs can also be used to improve the noise performance of the SC circuit shown in Figure 7.17, and it will be discussed in the next section that they are very useful for current-mode second-order mismatch-shaping DACs. Hence, linear 3-level DACs are important circuits.

Feasibility of 3-Level DACs. The basic operation/idea in 3-level DACs is very simple. In a fully-differential circuit, an analog signal is dumped to either the positive path (conversion of "$+1$"), the negative path (conversion of "-1"), or not at all (conversion of "0").

An example of this principle is shown in Figure 7.22, which is from a paper presented by Ka Y. Leung at ISSCC in 1997 [63]. The performance relies slightly on the matching of $C_{\mathrm{ref,p}}$ and $C_{\mathrm{ref,m}}$ and/or the

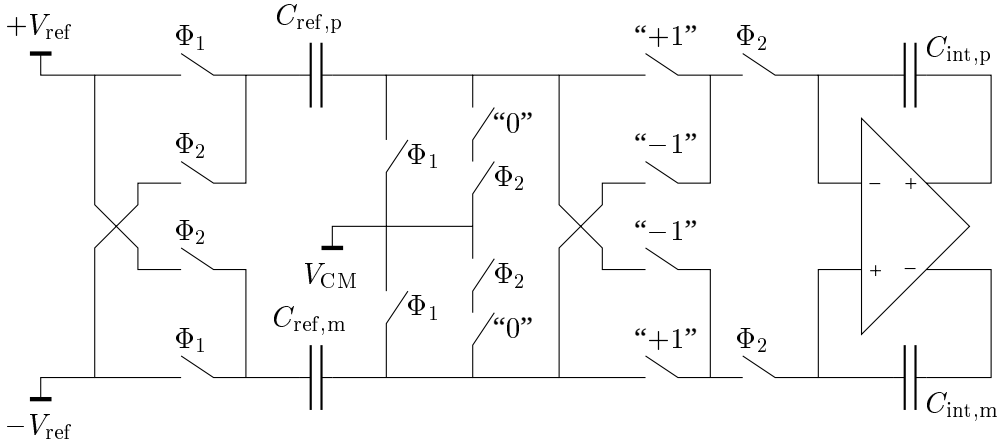


Figure 7.22: Almost-linear 3-level DAC presented at ISSCC 1997.

matching of $C_{\mathrm{int,p}}$ and $C_{\mathrm{int,m}}$, because it can be shown that the DAC's output levels[30] are proportional

[29] Academically, this is quite an interesting system, because it would implement a scaled-element mismatch-shaping DAC, where each nominal output value could be generated in only two ways (one for each value of $b_0(k)$). This is indeed a minimalist mismatch-shaping DAC.

[30] The DAC's output is here the *variation* of the opamp's output voltage, i.e., the signal is being integrated (which is generally

to -1, 0, and $+1$ with a relative inaccuracy of

$$\delta_{\mathrm{rel}} = \left(\frac{C_{\mathrm{ref,p}} - C_{\mathrm{ref,m}}}{C_{\mathrm{ref,p}} + C_{\mathrm{ref,m}}}\right)\left(\frac{C_{\mathrm{int,p}} - C_{\mathrm{int,m}}}{C_{\mathrm{int,p}} + C_{\mathrm{int,m}}}\right) \tag{7.14}$$

However, assuming that the technology's matching index is in the order of 0.1%, this error will not be dominating unless the performance is 120 dB or better. The reported performance was "only" about 100 dB, but by means of calibration, the DAC was linearized to the 118 dB level. In this case, the error sources are easy to identify: clock feedthrough and charge injection (charge errors) are well-known (and quite large) "signals", which will cause a signal-dependent error if the switches do not match perfectly (and they never do). Another (larger) error results because the opamp's offset only affects the output (by $V_{\mathrm{offset}}\frac{C_{\mathrm{ref}}}{C_{\mathrm{int}}}$) in the samples that are either "$+1$" or "-1" (effectively, this moves the "0" away from the DAC's linear characteristic, and by adjusting the opamp's offset, the DAC's linearity can be adjusted). Thermal-noise considerations were probably the reason why the circuit was designed this way.

Linear Three-Level DAC. Three-level DACs should preferably not rely on matching or cancellation of any kind (in Figure 7.22, the assumed matching of the switches can cause nonlinearity), and certainly not on the offset of opamps. Allowing the circuit to rely only on the second-order matching expressed by Equation 7.14, a linear 3-level DAC can be implemented as shown in Figure 7.23[31].

It is important that the Haigh-and-Singh delayed-clock-phases clocking scheme is used, i.e., that the switches controlled by Φ_1 and Φ_2 are opened slightly before the switches controlled by $\Phi_{1\mathrm{d}}$ and $\Phi_{2\mathrm{d}}$ [29]. An alternative clocking scheme, which is equally good, is discussed in [64]. This clocking scheme may provide a better understanding of the operation's ideality.

Either clocking scheme will assure that it is only the switches controlled by Φ_1 and Φ_2 that will cause charge errors. Because these switches are controlled independent of the digital signal, these charge errors will not cause nonlinearity[32]. As discussed in [64], the DAC's output signal is determined exclusively

the case for $\Delta\Sigma$ quantizer front ends).

[31] This implementation was proposed by Professor Un-Ku Moon (Oregon State University) during a private discussion. He claims that it is used frequently in pipeline ADCs (which are one of his specialties).

[32] The charge errors depend slightly on the surrounding impedance levels. This can potentially cause signal-dependent errors (nonlinearity), but it can be brought to a *very* low level.

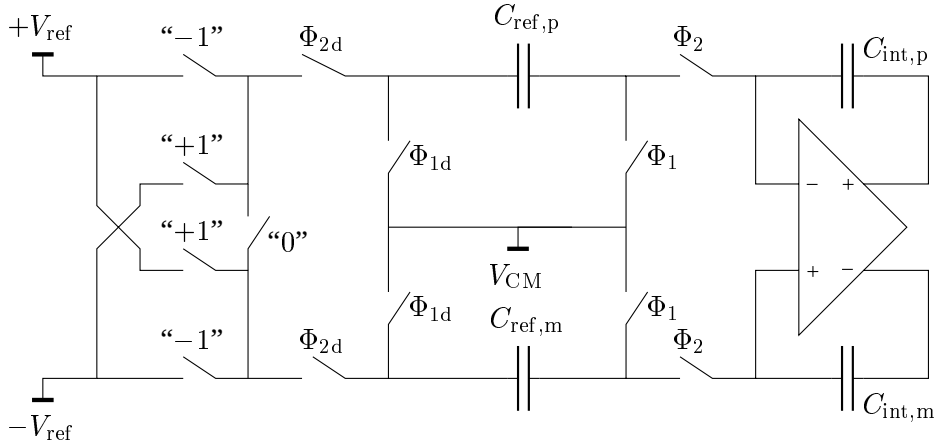


Figure 7.23: Linear 3-level DAC often used in pipeline ADCs.

by *voltage variation*[33] on the input side of the $C_{\rm ref}$ capacitors. Clearly, the *differential* voltage variation will be *perfectly* proportional to the digital signal, hence only common-mode variations of the input may cause errors. Nominally, $+V_{\rm ref}$ and $-V_{\rm ref}$ are of opposite polarity (with respect to $V_{\rm CM}$), but in reality, there will always be a small mismatch. Common-mode signals, however, can only "leak" to the output due to the opamp's finite common-mode-rejection-ratio (CMRR) (which can be made *very* good) and/or due to capacitor mismatch, in which case the suppression factor is $\delta_{\rm rel}$, as expressed by Equation (7.14). In conclusion, this 3-level DAC can be designed to be linear well beyond the 100 dB level.

Fully-Linear 3-Level DAC. Figure 7.24 shows a 3-level DAC which is "fully-linear" in the sense that the linearity does not in any way rely on capacitor matching, common-mode rejection, or on symmetric voltage signals. To emphasize this point, the circuit is shown in a single-ended version, although it should normally be implemented as a fully-differential circuit. The DAC delays the signal by one full sample[34], which may be a disadvantage when it is used as the feedback stage of a $\Delta\Sigma$ quantizer. Otherwise, the operation is quite similar to that of the DAC shown in Figure 7.23. It is made insensitive to charge errors by employing one of the clocking schemes described in [64], and the DAC's linearity relies only on the fact that the converted signal is proportional to the voltage variation on the input side of $C_{\rm ref}$,

[33] From when Φ_1 is opened until Φ_2 is opened.

[34] The input signal must be available in clock phase Φ_1 and held constant (from that point on) for one full clock cycle.

which in this implementation, is perfectly symmetric (assuming only that the reference potential $V_{\rm ref}$ is time-invariant).

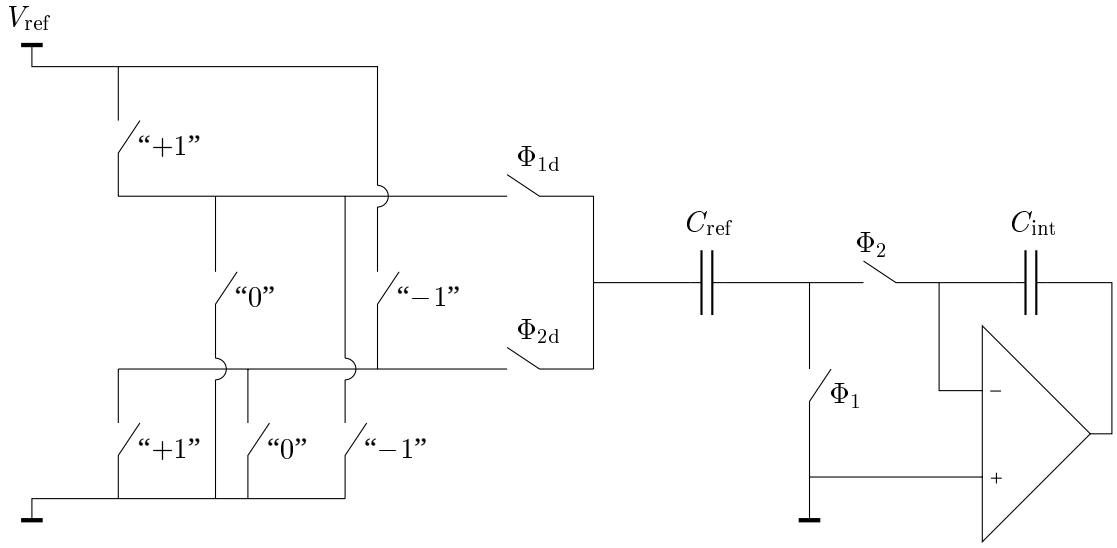


Figure 7.24: Fully-linear – but delaying – 3-level DAC, which also can be implemented differentially.

Protecting the Reference Voltage. To avoid a signal component on the reference voltage, it is often important that the load on the voltage reference is signal-independent (because the reference's output impedance will be nonzero). The 3-level DAC shown in Figure 7.22 already has this feature, but the DACs shown in Figures 7.23 and 7.24 do not. Hence, when using these DACs, a dummy load should be implemented to make the load on the voltage reference signal independent (in Figure 7.23, this can be implemented by connecting a discharged dummy capacitor $C_{\rm ref}/2$ between $\pm V_{\rm ref}$ when the digital signal attains the value "0").

Linear 3-Level Current-Mode DAC. The next Section will make good use of 3-level current-mode DACs, hence an implementation example is required (cf. Figure 7.25).

To avoid dynamic errors, the proposed DAC is based on a fully-differential implementation of the time-interleaved switching concept discussed in Section 5.2, whereby the linearity will depend only on static errors. Similar to the discussed fully-differential SC DACs, it can be shown that the static linearity will

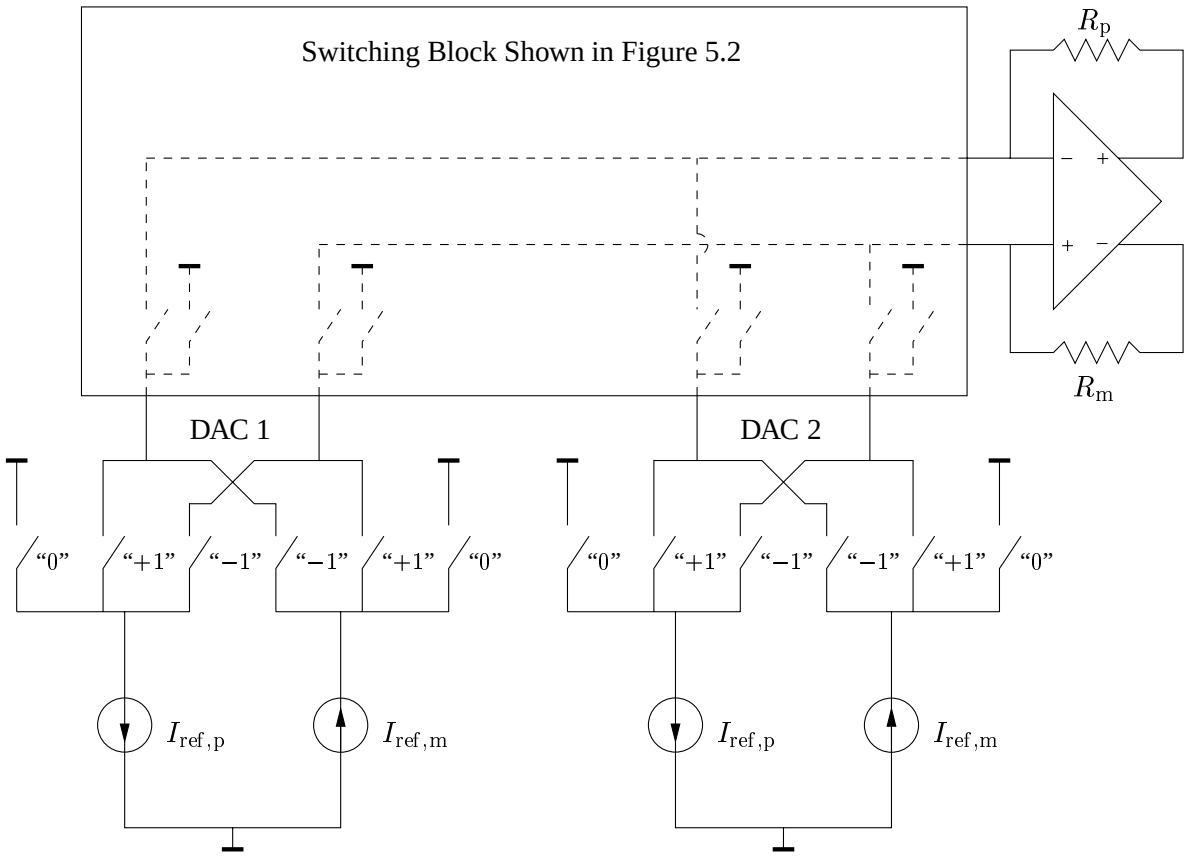


Figure 7.25: Linear 3-level current-mode DAC.

be limited by

$$\delta_{\mathrm{rel}} = \left(\frac{R_{\mathrm{p}} - R_{\mathrm{m}}}{R_{\mathrm{p}} + R_{\mathrm{m}}}\right)\left(\frac{I_{\mathrm{ref,p}} - I_{\mathrm{ref,m}}}{I_{\mathrm{ref,p}} + I_{\mathrm{ref,m}}}\right) \tag{7.15}$$

If the resistors cannot be assured to match well, then the current sources must match accordingly better, and vice versa. In general, to obtain 100 dB performance, δ_{rel} must be less than -100 dB, which usually is feasible.

Notice that the switching block's impedance seen from the opamp is signal-dependent. This may cause nonlinearity, because the opamp's offset will affect the output as a linear function of this impedance. The simplest way to alleviate the problem is to make the impedance level high, e.g., by cascode coupling the reference current sources (the switches can be used for this purpose). As the ratio of the opamp's offset relative to full-scale output usually will be -40 dB or less, the reference current sources' output impedance relative to the opamp's feedback resistors need only be about 60 dB to obtain 100 dB performance (usually feasible).

7.5.6 Current-Mode Implementation

Continuous-time current-mode DACs can typically be designed to have a better noise performance, higher speed, and lower power consumption than their switched-capacitor counterparts. Hence, first-order mismatch-shaping will generally be sufficient to obtain the required performance, but there may be some applications where second-order mismatch shaping is preferable or needed.

Second-order mismatch-shaping DACs are best and simplest implemented using the generalized filtering principle, e.g., as shown in Figure 7.15. However, the current-mode analog filters are not as simple to implement as switched-capacitor circuits are (cf. Figure 7.17). The analog filters should, by definition, match a discrete-time filter function, e.g., $1/H_1(f) = 1 - z^{-1},\ \ z = e^{j2\pi f/f_s}$, but part of the advantage of current-mode DACs is lost if switched-current filters are used. One option to avoid this scenario is to design a CT filter

$$H_{\mathrm{CT}}(f) = \frac{\sum_{i=0}^{Q_N} b_i s^i}{\sum_{i=0}^{Q_D} a_i s^i} \tag{7.16}$$

such that $H_{\mathrm{CT}}(f) \simeq 1/H_1(f)$ in the signal band; but it is typically difficult to obtain a reasonably good matching of the filters, and even in the best case, the filters will not match in the entire frequency range[35]. Another and typically much better option is to use pseudo-digital filters.

Pseudo-Digital Filters – Type I. Considering again the SC implementation shown in Figure 7.17, it can be observed that each of the single-bit digital signals generated by the UE-MS encoders are D/A converted and filtered individually by the non-reset capacitors. Hence, each capacitor can be construed as a linear 3-level DAC converting the first-order difference of the respective single-bit digital signals. The shown implementation is particularly suitable for SC circuits, but the underlying principle can also be used for CT current-mode implementations, where the 3-level DACs are implemented differently. This is shown in Figure 7.26, where the overall D/A conversion is performed by an array of 3-level DACs[36].

The spectral encoder (not shown) is assumed to be designed as a variation of the spectral encoder shown in Figure 7.15, where the master signal $b_0(k)$ is of 3-level resolution, and the primary compensation signal $b_1(k)$ is of only 2-bit resolution (because the magnitude of $n_0(k)$ is proportional to the master signal's step size, i.e., it is decreased by a factor of two). If the 3-level DACs are as linear as the required performance, the composite DAC will perform second-order mismatch-shaping, and 100 dB performance can be obtained at around 10 times oversampling.

Pseudo-Digital Filters – Type II. Figure 7.27 shows an alternative technique for the implementation of pseudo-digital filters suitable for use in combination with the generalized filtering principle. For simplicity, only the D/A conversion of one of the compensation signals is shown. Each of the two single-bit signals from the UE-MS encoder is fed to a digital delay line, where each tap controls a linear single-bit DAC. The filtering (second-order differentiation is shown) of these signals is perfectly linear.

[35] A filter will be needed to smoothen the DAC's output signal

[36] The system is assumed to be implemented as a fully-differential circuit, where the 3-level DACs are implemented as shown in Figure 7.25. Notice that only one time-interleaving switching block (cf. Figure 5.2) should be implemented for the composite DAC.

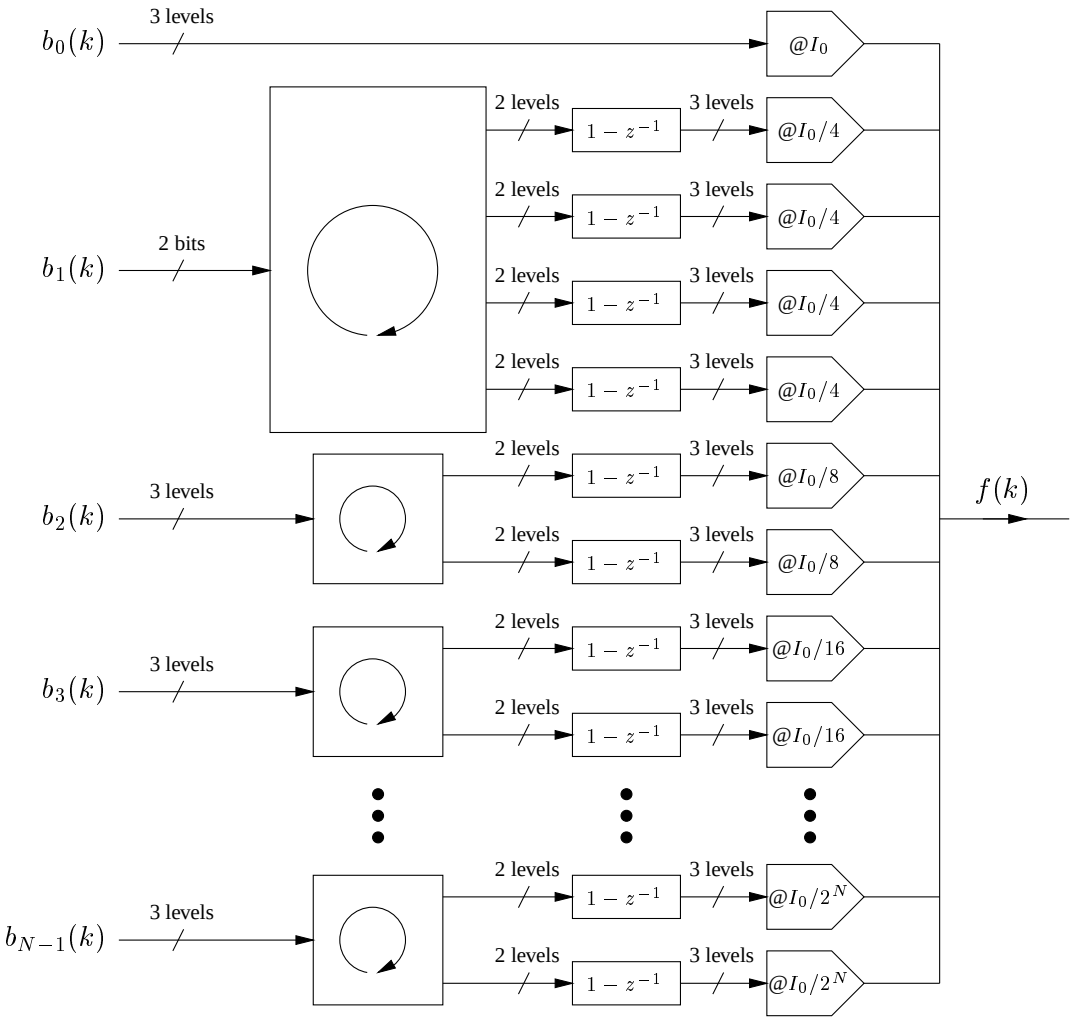


Figure 7.26: Continuous-time current-mode second-order mismatch-shaping DAC.

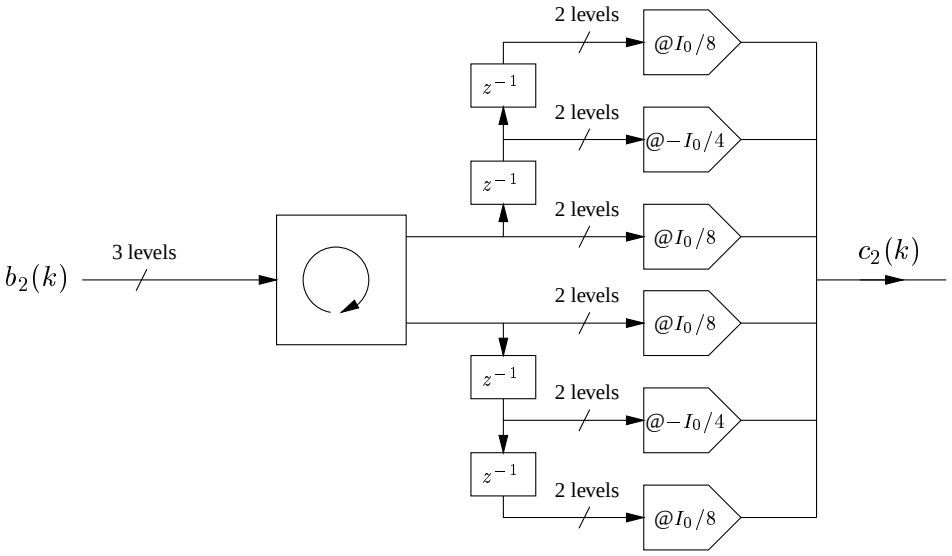


Figure 7.27: Second-order differentiating compensation DAC.

Although each FIR filter individually is perfectly linear, this does not imply that the overall operation will be ideal. If the FIR filter coefficients do not match their nominal value, the filter's transfer function will not be ideal, and the compensation of the master $\Delta\Sigma$ modulator's truncation error $\epsilon(k)$ (cf. Figure 7.2) will not be ideal. Although mismatch of the filter's transfer function can be tolerated to some extent (cf. Section 7.5.2), this effect somewhat limits the maximum length of the FIR filter. First-order filters are generally sufficient, but in this case the 3-level-DAC technique proposed above (Figure 7.26) is likely to provide the best performance.

The analog-FIR filter technique is useful mainly if second-order filtering is required. In that case, the numeric stability of the transfer function's zeroes should be considered carefully. For example, the two nominal zeroes at $z = 1$ for the circuit shown in Figure 7.27 are *extremely* sensitive to coefficient inaccuracy (because it is a double zero), and the circuit should never be implemented as shown. The filter matching can be improved either by designing the master $\Delta\Sigma$ modulator's first filter stage as a resonator (resonating at a high signal-band frequency), or by designing the second-order differentiation as shown in Figure 7.28

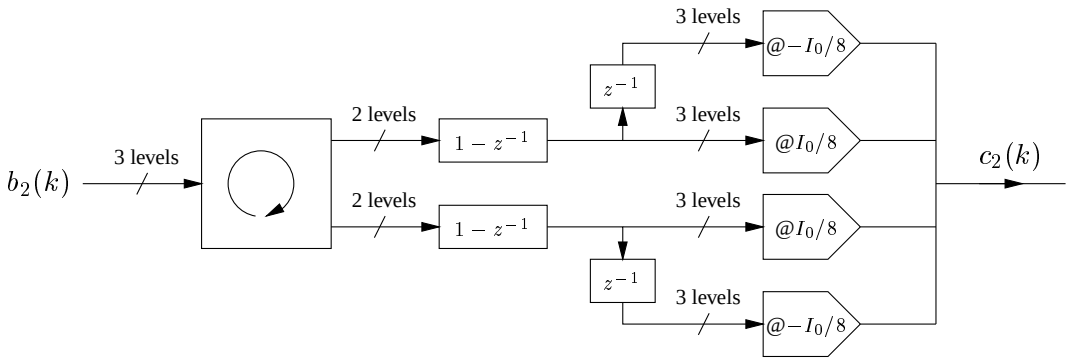


Figure 7.28: Improved second-order differentiating compensation DAC.

7.5.7 Mismatch-Shaping Bandpass DACs

Bandpass unit-element mismatch-shaping encoders are feasible[37] and it should be understood that all the discussed techniques can be implemented as bandpass systems. Bandpass DACs are particularly simple to implement when the center frequency is $f_s/4$.

Notice, in particular, that the current-mode DAC shown in Figure 7.26 can be implemented as a bandpass system because a single-bit signal, which is filtered by $1 + z^{-2}$, will result in a 3-level signal.

[37]For example, for tree-structure encoders, choose $H(f) = \frac{1}{1+z^{-2}}$. Although the filter is of second order, the complexity, stability and performance is equivalent to that of a first-order encoder.

Chapter 8

High-Resolution Delta-Sigma Quantizers

A general characteristic for state-of-the-art quantizers – and particularly for signal quantizers – is that their performance is limited mainly by the linearity of an internal DAC (cf. Section 3.4 and Chapter 4). However, with the development (and use) of scaled-element mismatch-shaping DACs (cf. Chapter 7), other factors will typically turn out to be limiting in the design of a signal quantizer.

Consider, for example, Figure 8.1, which shows the topology in which most multi-bit $\Delta\Sigma$ quantizer's have been implemented thus far[1]. The loop quantizer may cause only little delay, hence it is practically

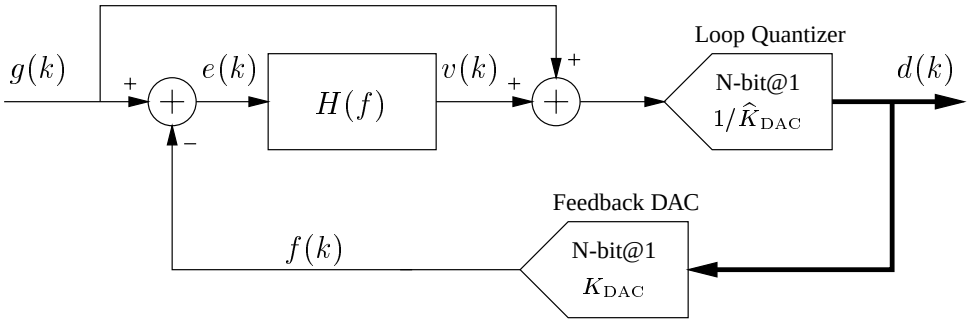


Figure 8.1: Traditional multi-bit delta-sigma quantizer.

always designed as a flash quantizer (cf. Section 3.4.1). Typically, to avoid complex circuitry and a high

[1] In the best case; quite often the feed-forward branch (from $g(k)$ to the loop quantizer) is omitted and/or $d(k)$ is injected at internal nodes of $H(f)$. These are undesirable variations that may cause substantial problems [43].

power consumption, the loop quantizer's resolution, and hence the resolution of $d(k)$, will be restricted to less than (say) 5 bits. In other words, it is now the loop quantizer, and not the feedback DAC, which is the limiting element[2]. This Chapter will consider some options for how the resolution of $d(k)$ can be increased without using high-speed high-resolution quantizers.

8.1 Choosing the Optimal Resolution

Consider Figure 8.1. An optimally designed signal quantizer will represent the analog input signal $g(k)$ by a high-resolution digital approximation $d(k)$. It is, however, useless to increase the resolution of $d(k)$ beyond a certain limit. This limit is attained when the feedback DAC's error signal $m(k)$ dominates the truncation error $r(k)$[3]

$$e(k) = g(k) - f(k) = \underbrace{[g(k) - K_{\mathrm{DAC}} d(k)]}_{\text{Truncation Error}} - \underbrace{[\ m(k)\ ]}_{\text{DAC Error}} \tag{8.1}$$

at which point the signal-to-noise ratio of $e(k)$ is 0 dB, and there is only little information left for the loop filter to detect. In other words, for a typical technology for which the relative matching index is (say) 0.1%, the highest meaningful resolution of $d(k)$ is in the order of 10 bits, in which case the analog loop filter $H(f)$ needs to be of only the same order as the mismatch-shaping DAC. Sometimes $H(f)$ may be designed one order higher to make the truncation error fully negligible.

The objective of this chapter is to design signal quantizers that provide an output signal $d(k)$ with a resolution of approximately 10 bits, without requiring use of high-speed high-resolution quantizers (to minimize the quantizer's complexity and power consumption).

[2]When the signal is represented with only 5-bit resolution, a sixth- or higher-order loop filter $H(f)$ is required to obtain 100 dB performance at a OSR of 10 (cf. Section 3.4.3). If the signal is represented by (say) 10-bit resolution, a second-order loop filter is sufficient.

[3]Ideally, only the truncation error will be detected by $H(f)$.

8.1.1 Fundamental Principle for High-Resolution Quantization

High-resolution low-complexity quantizers generally function by adding the results from several low-resolution quantizations of signals of very different effective magnitudes[4]. This fundamental principle can also be used for the multi-bit $\Delta\Sigma$ quantizer shown in Figure 8.1.

In Section 4.3.1, it was derived that the magnitude of $v(k)$ will be only a small factor of (say) 1 to 10 times larger than the loop quantizer's step size. Hence, for high-resolution $\Delta\Sigma$ quantizers, the full-scale range of $g(k)$ will be significantly larger than that of $v(k)$, whereby $d(k)$ can possibly be generated as the sum of a low-resolution quantization of $g(k)$ (the MSBs of $d(k)$) and a low-resolution quantization of $v(k)$ (the LSBs of $d(k)$). This arrangement is shown in Figure 8.2. It can be construed as a digital-domain implementation of the feed-forward path.

8.2 Two-Stage Delta-Sigma Quantizers

Figure 8.3 shows an alternative and very interesting interpretation of the circuit shown in Figure 8.2 (it is the same circuit, but drawn differently), which will be called a *two-stage* $\Delta\Sigma$ *quantizer*. This Figure clearly shows that the quantizer, in reality, is a two-stage residue-calculating quantizer (cf. Section 3.4.2), where the first stage is a traditional residue stage (cf. Figure 3.17), and where the second stage is a $\Delta\Sigma$ quantizer of the type shown in Figure 3.24. A peculiarity of this circuit is, however, that the two stages share the same feedback DAC, which is essential in order to avoid unshaped DAC errors.

The $\Delta\Sigma$ quantizer's input signal is the first stage's residue signal $r_0(k) = g(k) - d_0(k)K_{\mathrm{DAC}}$. Preferably, this residue signal should be added to $v(k)$[5] (cf. Figure 8.4), but this would require that the loop quantizer be clocked *after* the first-stage quantizer (timing issues are discussed later).

[4] For example, pipeline quantizers (cf. Figure 3.20).

[5] In order to implement the $\Delta\Sigma$ quantizer in the preferred topology (cf. Section 3.4.3 and Figure 3.25).

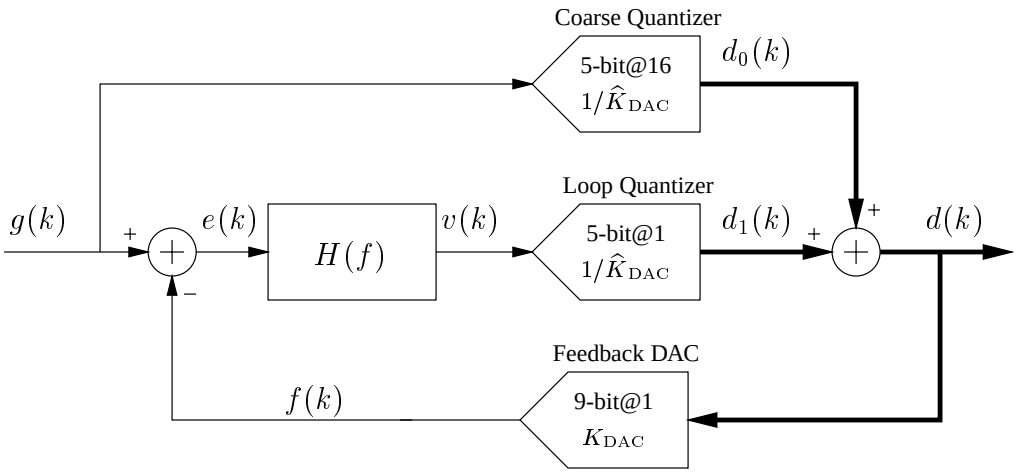


Figure 8.2: $\Delta\Sigma$ quantizer with a digital-domain feed-forward path.

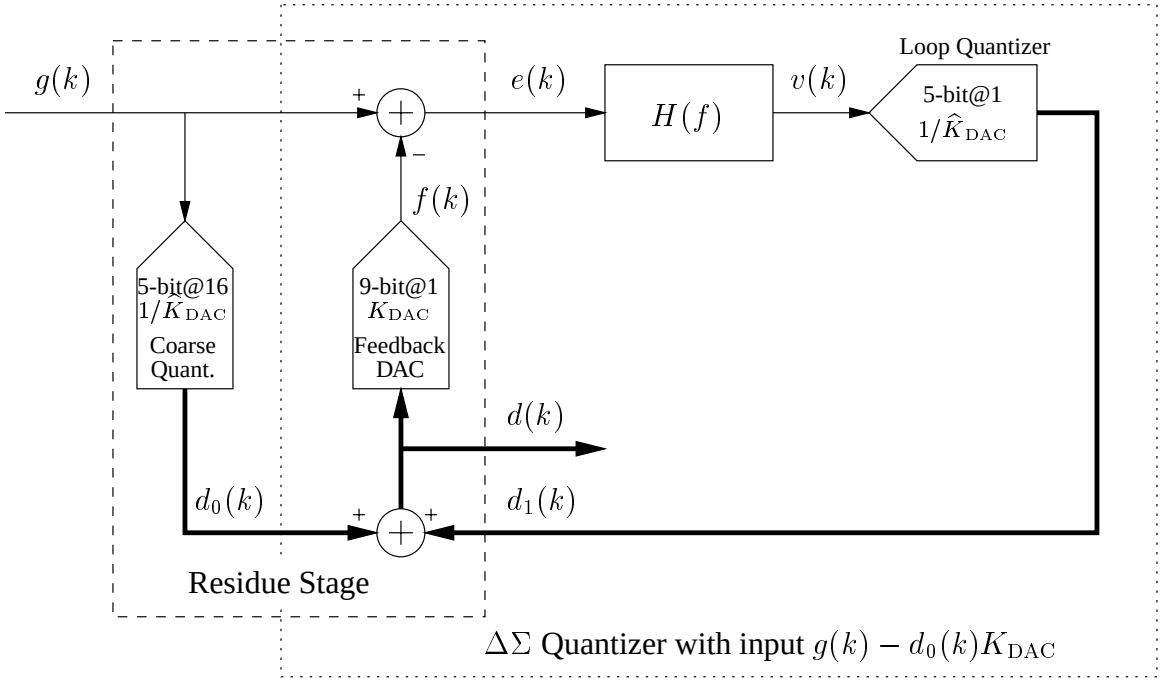


Figure 8.3: The same $\Delta\Sigma$ quantizer as shown in Figure 8.2, but here drawn to emphasize its residue-calculating topology.

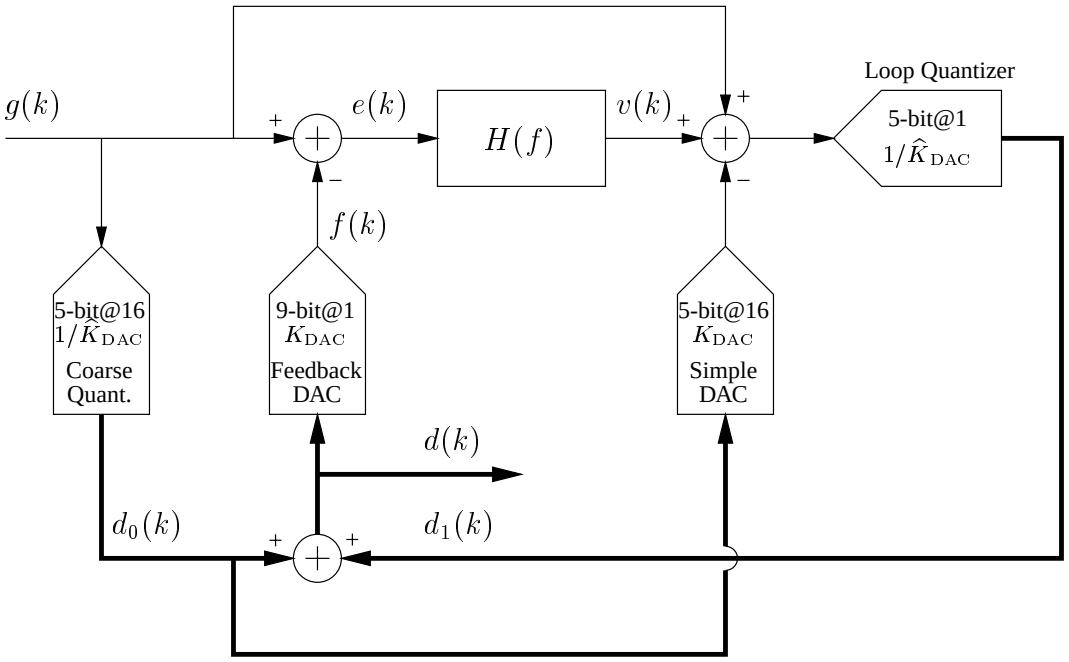


Figure 8.4: Two-stage $\Delta\Sigma$ quantizer with implemented feed-forward path.

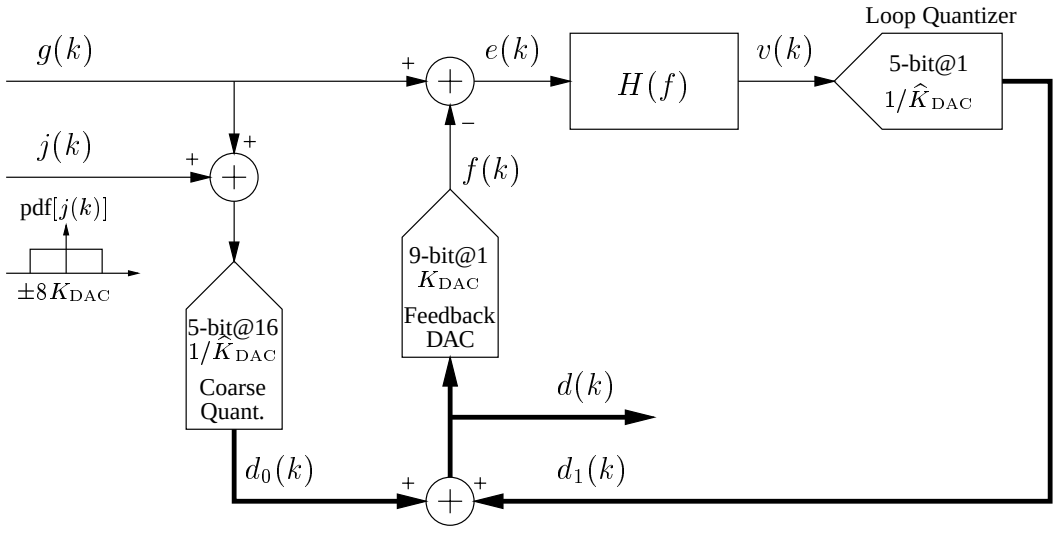


Figure 8.5: Two-stage $\Delta\Sigma$ quantizer with fully-dithered first stage.

8.2.1 Preventing Nonlinearity

A perhaps not-so-obvious difference between the two-stage $\Delta\Sigma$ quantizers shown in Figures 8.3 and 8.4 is that the one without the feed-forward path is nonlinear, even if the feedback DAC and the two quantizers are ideal. The problem arises because the first stage (the residue stage) has a nonlinear static transfer characteristic[6] with unity gain, whereas the second stage (the $\Delta\Sigma$ quantizer) has a *dynamic* (frequency-dependent) signal transfer function, which is unity only if the feed-forward path is implemented (cf. Figure 8.4)[7].

The nonlinearity can be avoided in two ways; either the two stages must be designed to have the same signal transfer function, or the first stage must be made linear.

Matching the Transfer Functions. The signal transfer functions can be matched either by designing the $\Delta\Sigma$ quantizer with the feed-forward path[8], or by matching the first stage to the second by computing $d(k)$ as the sum of $d_1(k)$ and $d_0(k)$ filtered by $\frac{H(f)}{1+H(f)}$. The latter technique, however, is of only limited interest because it involves matching of analog/digital filters, and because the complexity of the digital filter is considerable.

Linearizing the Residue Stage. Linearization of the residue stage is a matter of decorrelating the residue signal from the input signal, which can be obtained by dithering the first-stage quantizer as shown in Figure 8.5. The dither signal $j(k)$ should preferably be white noise with a uniform probability-density function (pdf) spanning one LSB of the first-stage quantizer. By identifying that this two-stage $\Delta\Sigma$ quantizer is equivalent to the traditional multi-bit $\Delta\Sigma$ quantizer shown in Figure 8.1 (with a 9-bit loop quantizer) where $j(k)$ is added directly to $v(k)$, linearity can be concluded. Notice, however, that the dither signal $j(k)$ is much larger than what would normally be used in a 9-bit $\Delta\Sigma$ quantizer, hence the performance will not be comparable.

[6] Truncation to 5-bit resolution.

[7] The signal transfer function is $\frac{H(f)}{1+H(f)}$, which can differ substantially from unity when the OSR is low.

[8] In which case, the $\Delta\Sigma$ quantizer's signal transfer function is $\frac{1+H(f)}{1+H(f)} = 1$. See Section 3.4.3 and Equation (3.53).

8.2.2 Simulation Results

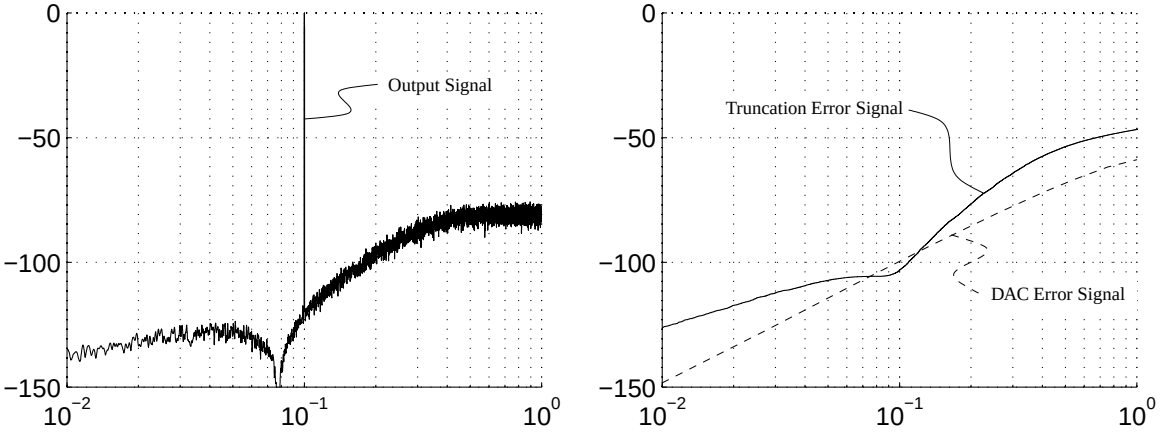
The simulation results presented below are generated using an ideal feedback DAC. The actual (total) error signal will be the sum of the shown truncation error signal and the feedback DAC's error signal (cf. Figure 3.22 and Chapter 7). The dashed line in the Figures shows an estimate of the feedback DAC's error signal, assuming that it is second-order shaped and that the technology's matching index is 0.1% (cf. Figure 7.18).

Reference Performance. Figure 8.6 shows the performance of a traditional multi-bit $\Delta\Sigma$ quantizer with a 9-bit loop quantizer (cf. Figure 8.1). The loop filter is of third order and the $\mathrm{NTF}_{\max}$ value is three[9]. It can be observed that the truncation error signal dominates the feedback DAC's error signal in large parts of the frequency spectrum, but also that the DAC's error signal limits the performance at the target OSR (as it should for an optimally designed quantizer). The performance of this system will represent the reference to which the performance of the proposed systems will be compared (the performance cannot be improved unless a better DAC is available).

Performance of the Two-Stage Delta-Sigma Quantizer Shown in Figure 8.4. Figure 8.7 shows the simulated performance of the two-stage $\Delta\Sigma$ quantizer shown in Figure 8.4. The performance of this quantizer is virtually indistinguishable from the reference performance. This result is very encouraging, because two-stage $\Delta\Sigma$ quantizers are significantly simpler to implement than traditional single-stage single-loop multi-bit $\Delta\Sigma$ quantizers (cf. Section 8.3).

Performance of the Two-Stage Delta-Sigma Quantizers Shown in Figures 8.3 and 8.5. Figure 8.8 shows the performance of the two-stage $\Delta\Sigma$ quantizer shown in Figure 8.3. Clearly, this quantizer is quite nonlinear (partly due to the high signal frequency), and hence it is not suitable for high-performance

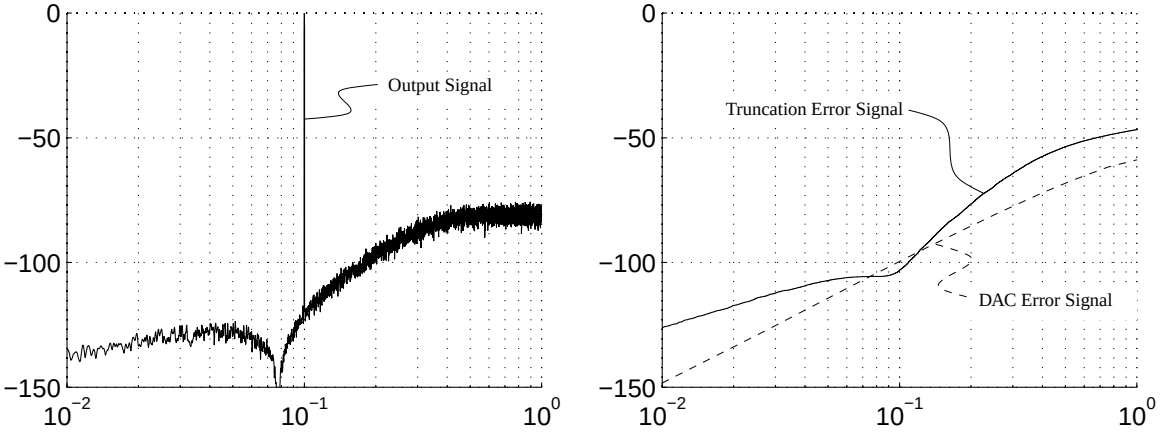
[9] At 10 times oversampling, the signal-band power of the truncation error can be reduced by about 10 dB by increasing the loop filter's $\mathrm{NTF}_{\max}$ value, but in this case there is little point in doing that, because the DAC's error signal dominates the signal-band performance and the out-of-band performance would be degraded. 100 dB performance cannot be obtained at 10 times oversampling when using a second-order loop (the resolution of $d(k)$ is too low).



(a) Spectral power density versus frequency

(b) Signal-band power versus 1/OSR

Figure 8.6: Traditional 9-bit $\Delta\Sigma$ quantizer (cf. Figure 8.1).



(a) Spectral power density versus frequency

(b) Signal-band power versus 1/OSR

Figure 8.7: Two-stage $\Delta\Sigma$ quantizer with 5-bit quantizers and feed-forward signal path (cf. Figure 8.4).

applications. The system can be linearized by dithering the first-stage quantizer, as shown in Figure 8.5, in which case the performance will be as shown in Figure 8.9. Although this quantizer is linear, it is observed that the performance does not measure up to the reference. The same performance can be obtained using a traditional $\Delta\Sigma$ quantizer with a 5-bit loop quantizer, which often will be easier to implement[10].

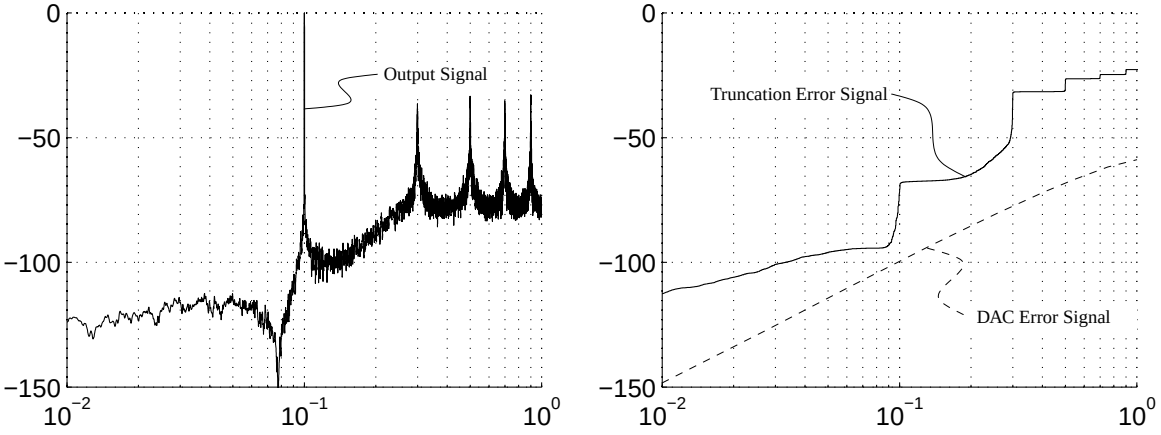
Conclusion. The two-stage $\Delta\Sigma$ quantizer structure is very useful and efficient, but only if the feed-forward signal path is implemented (as shown in Figure 8.4). In that case, the quantizer's linearity does not depend on the linearity of the first-stage quantization $d_0(k)$, which may be generated by actual quantization of $g(k)$, by prediction on the basis of recent values of $d(k)$, or in almost any other way[11]. For each bit of correlation between $g(k)$ and $d_0(k)$, the Nyquist-band power of the truncation error signal will decrease by 6 dB. Hence, in the ideal case, the first-stage quantizer would be a high-resolution quantizer (say, a pipeline quantizer), such that the loop quantizer need have only a few bits of resolution. However, this is only implementable if the required *timing* can be obtained (discussed in Section 8.3).

8.3 Implementation of Two-Stage Delta-Sigma Quantizers

As discussed in Section 3.4.3, the fundamental property of signal quantizers is that they attempt to minimize the error signal $e(k)$. In essence, the first stage of the two-stage $\Delta\Sigma$ quantizer (cf. Figure 8.3) provides an estimate $d_0(k)$ of $g(k)$, and the second stage generates $d_1(k)$ to correct for the inaccuracy of the signal $d(k)$ generated thus far (which is expressed by $v(k)$). If $d_0(k)$ is a good (say 10-bit accurate) estimate of $g(k)$, then $e(k)$, $v(k)$, and hence $d_1(k)$ will be small, and the loop quantizer can have a small step size with only (say) 2 bits of resolution. On the other hand, if $d_0(k)$ is a less accurate (say, 5-bit)

[10] However, not always. Pipeline techniques will be proposed later, and then the first-stage (say pipeline) quantizer can be designed to have a high resolution. For stability reasons (of the $\Delta\Sigma$ loop), the loop quantizer may have only little delay, hence it generally cannot be designed as a pipeline quantizer.

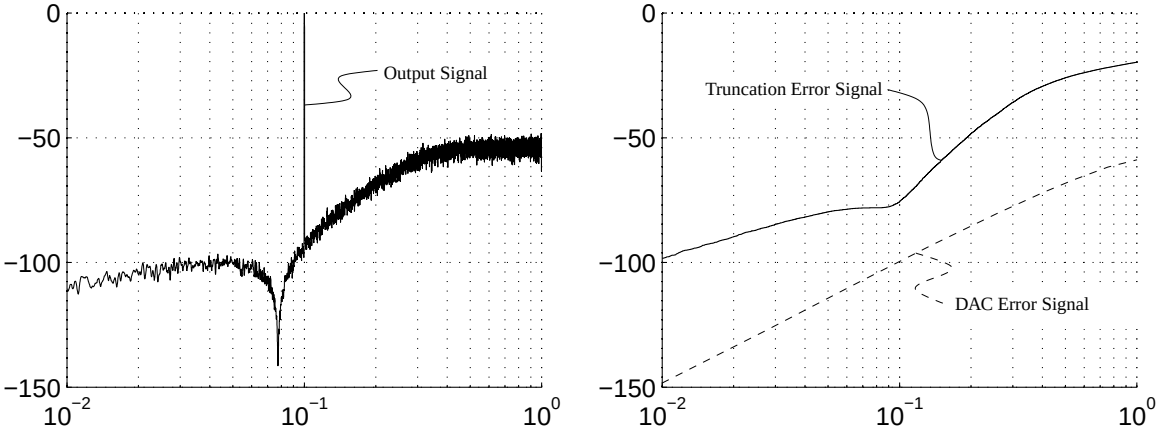
[11] A simple technique is to use linear prediction and define e.g. $d_0(k) = 2d(k-1) - d(k-2)$. Because the *absolute* accuracy of linear prediction is best for small input signals, this technique will result in a system with a good dynamic range performance. Consider also the predictive structure discussed in Section 9.3.1.



(a) Spectral power density versus frequency

(b) Signal-band power versus 1/OSR

Figure 8.8: Two-stage $\Delta\Sigma$ quantizer without the feed-forward signal path (cf. Figure 8.3).



(a) Spectral power density versus frequency

(b) Signal-band power versus 1/OSR

Figure 8.9: Dithered two-stage $\Delta\Sigma$ quantizer without the feed-forward signal path (cf. Figure 8.5).

estimate of $g(k)$, then it is necessary to implement the $\Delta\Sigma$ quantizer's feed-forward path to the loop quantizer (cf. Figure 8.4), because otherwise, the estimation error $g(k) - d_0(k)K_{\rm DAC}$ will be transferred directly to $e(k)$, in which case $v(k)$ will attain large values as an unmistakable sign of the less than ideal operation.

Timing Problems. When thinking of the system in these terms, another problem becomes apparent. Several operations must be performed sequentially and within one clock cycle[12] to assure that $g(k)$ and $f(k)$ are as correlated as possible in order to avoid $e(k)$ and $v(k)$ attaining large values. Clearly, this poses a serious problem for high-speed systems because each operation takes time, thereby possibly limiting the maximum sampling frequency. The problem becomes even worse if the feed-forward path need be implemented, or if the first-stage quantizer is a multi-stage quantizer[13] (continuous-time quantizers are an all-together separate issue).

8.3.1 Introducing Pipeline Techniques to Allow Circuit Delays

As always, it is better to improve the system's topology rather than blindly designing faster circuitry (to which there always will be a limit, and which will be associated with an increased power consumption).

First, it is important to observe that it usually is tolerable if the two-stage $\Delta\Sigma$ quantizer is delaying, i.e., if $d(k)$ represents, e.g., $g(k-2)$ and not necessarily $g(k)$. Second, it should be observed that the main objective is to avoid large values of $e(k)$, which requires a high-resolution (10-bit) data[14] quantization of

[12] The first-stage quantization of $g(k)$, the addition of $d_0(k)$ and $d_1(k)$, the D/A conversion of $d(k)$, and the subtraction of $g(k)$ and $f(k)$.

[13] Which, in reality, is two sides of the same problem. If the first-stage quantizer is a (say) two-step flash quantizer, the resolution of $d_0(k)$ may be high enough to allow the feed-forward branch to be omitted. On the other hand, if the first-stage quantizer is a single-stage flash quantizer, the resolution of $d_0(k)$ will be low and the feed-forward path required, in which case the combined first-stage/loop quantizer operates as a two-stage flash quantizer.

Subranging quantizers may represent a good choice for the implementation of the first-state quantizer. Another technique, which has a lot of potential, is to *predict* $d_0(k)$ on the basis of previous $d(k)$ values. The calculations can typically be performed in advance, whereby the feed-forward path is quite simple to implement. This approach is particularly interesting when the input "$g(k)$" is a continuous-time signal (will be discussed).

[14] The difference between data quantizers and signal quantizers is discussed in Section 3.4.

$g(k)$, which will be associated with substantial delay if the circuit complexity and power consumption is to be reasonable. By combining these two observations, it is found that the two-stage $\Delta\Sigma$ quantizer can be implemented in the topology shown in Figure 8.10, where an analog N-sample delay line has been inserted to (time wise) line up the correlated samples of $g(k)$ and $f(k)$. In the Figure, it is assumed that the loop quantizer and the feedback DAC in combination cause one full sample of delay[15], hence the feed-forward path should delay only $N-1$ samples. This *pipeline* technique can be used extensively to solve most delay problems, but (obviously) it has interest only if high-performance analog delay lines can be implemented.

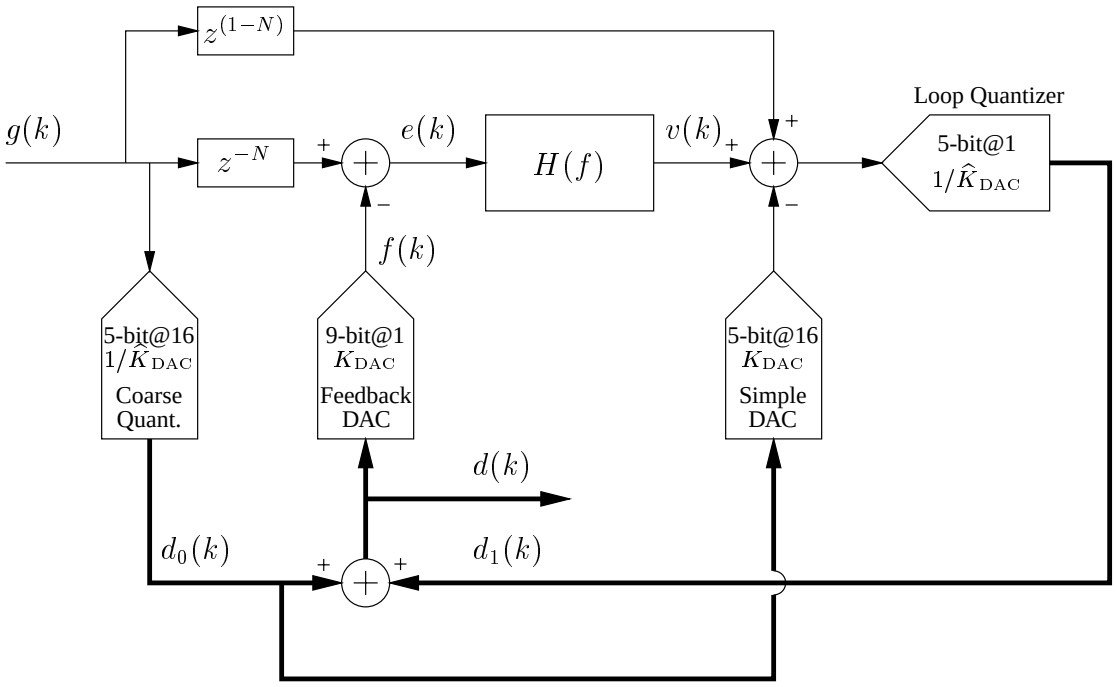


Figure 8.10: Pipelined two-stage $\Delta\Sigma$ quantizer.

8.3.2 Design of Analog Delay Lines

Simple analog delay lines can easily be implemented as switched-capacitor circuits. However, if 100 dB performance is required at 10 times oversampling, noise and several other issues come into play, and

[15]This is generally the maximum allowed in order to preserve the $\Delta\Sigma$ loop's stability.

then the circuit is not that simple to design.

The following discussion will consider the design of a high-performance switched-capacitor delay line, which provides one full clock cycle of delay (and it can easily be extended). A one-sample delay is usually sufficient because $d(k)$ is required to be of only about 10-bit resolution. Half the delay can, for example, be used for a two-step flash quantization, and the other half can be used for the mismatch-shaping feedback DAC's computations. A longer delay line will, however, be required if, for example, the first-stage quantizer is designed as a pipeline quantizer.

Proposed Implementation: a Delay-Line Integrator. Figure 8.11 shows the proposed implementation, which is a delaying integrator, i.e., a combination of the one-sample delay line and the loop filter's first integrator stage.

The input voltage signal $g(k)$ is sampled at the termination of clock phase Φ_2. Capacitor C_1 represents the main (integrating) signal path. It dumps the signal charge $q_g(k) = -g(k)C_1$ to the integrating capacitor $C_{\rm int}$ coupled across the main (high-performance) opamp OP1. The other (typically much simpler) opamp, OP2, implements a sample-and-hold (S/H) operation. The input signal $g(k)$ is sampled on C_3 in clock phases Φ_2, and the capacitor is "flipped around" and used as the opamp's feedback element in clock phases Φ_1. Capacitor C_4 is used only to hold the output voltage while C_3 samples the next sample $g(k+1)$ (uncritical operation, which may be incorporated to prevent undesired effects from stray capacitors). The S/H stage drives C_2 (of the same nominal value as C_1) which in combination with OP1 and $C_{\rm int}$ implements an inverting amplifier from $g(k)$ to $V_1(k)$.

The fundamental idea is that $q_c(k) = (g(k)-g(k-1))C_2$ will cancel the signal charge $q_g(k) = -g(k)C_1$ from C_1, hence $g(k)$ will not affect $V_1(k)$ in the clock phase Φ_1 immediately following the sampling instance. However, in the *following* clock phase Φ_1, the charge provided by C_2 will be withdrawn

$$\begin{aligned} q_c(k+1) &= (g(k+1) - g(k))C_2 \\ &= g(k+1)C_2 - \boxed{g(k)C_2} \end{aligned} \tag{8.2}$$

and the net result is that the signal $-g(k)C_1$ is dumped to the feedback capacitor $C_{\rm int}$ one clock cycle later than it otherwise would have been dumped (if OP2 is omitted). This delay provides the extra time

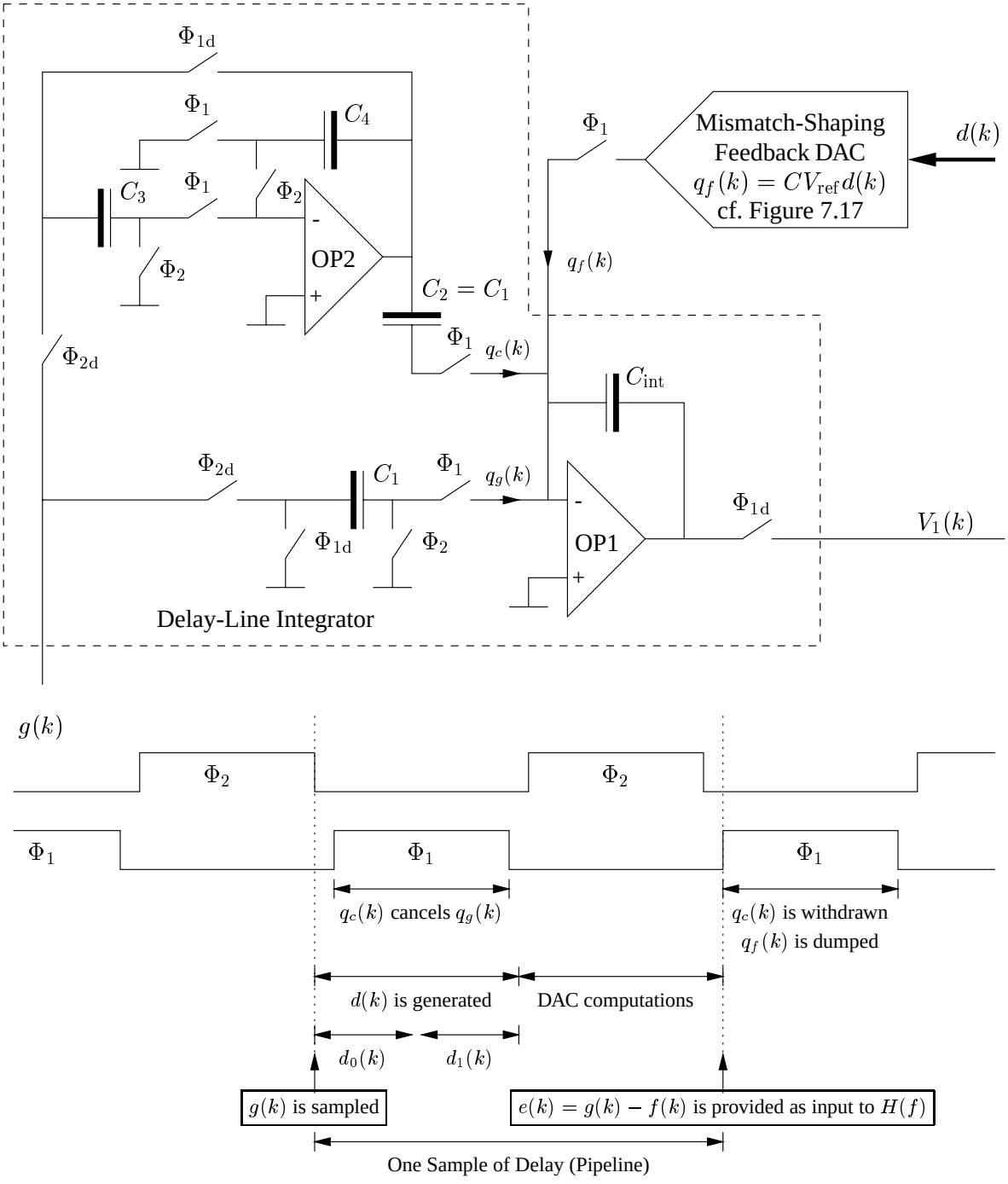


Figure 8.11: Combined delay line and integrator.

required for the generation of a high-resolution mismatch-shaped feedback signal $q_f(k) \simeq q_g(k)$.

A key feature of this implementation is that it is quite robust with respect to circuit imperfections. Mismatch of C_1 and C_2 will, e.g., cause only linear errors, and it is, therefore, of only little concern. Furthermore, because C_2 is not discharged at any time, all errors from OP2 (including noise, nonlinearity, etc.) will be first-order differentiated; hence, C_1 and OP1 will be the dominating error sources (which is the best-case scenario[16]).

Figure 8.12 shows that the principle easily can be generalized and used for the implementation of delay lines of arbitrary length.

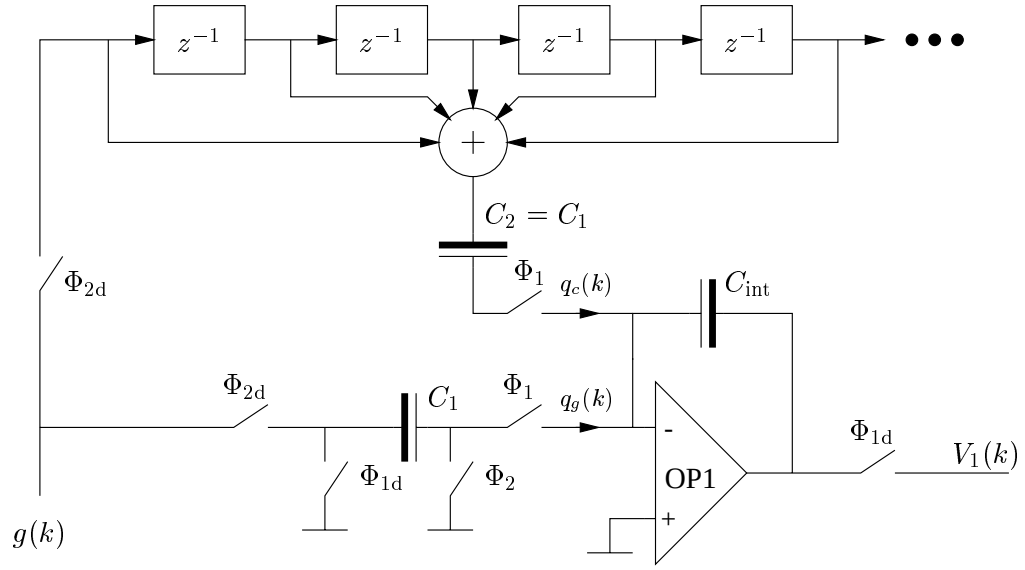


Figure 8.12: Generalized delay-line integrator with first-order shaped error signal.

8.3.3 Avoiding Sequential Settling

Figure 8.13 shows a more detailed (system-level) perspective of the pipelined two-stage $\Delta\Sigma$ quantizer shown in Figure 8.10 (for $N = 1$). The loop filter $H(f)$ has been sectioned in three parts to emphasize the prospective use of the delaying integrator shown in Figure 8.11, and to show that the feed-forward

[16]For further noise improvements, C_1 can be combined with the mismatch-shaping DAC's master DAC.

path can be implemented without introducing new active elements (differentiation is a passive operation in SC circuits).

The Problem. It is well understood that to preserve stability, the $\Delta\Sigma$ loop must have a path in its topology that has exactly one sample of delay [1]. For the shown system topology, this implies that $F(z)$ will be nondelaying because the $\Delta\Sigma$ loop already includes one full sample of delay. For example, to implement the classical second-order loop filter $H(z) = \frac{z^{-1}(2-z^{-1})}{(1-z^{-1})^2}$, the system should be designed with $F(z) = 2 - z^{-1}$, where "2" is the nondelaying part. This is not optimal, because nondelaying SC circuits are subject to sequential settling, which will reduce the maximum sampling frequency [29]. To avoid this scenario, the topology can be modified to incorporate a local feedback path, as shown in Figure 8.14.

The Proposed Solution. The gain $A \cdot K_{\mathrm{DAC}}$ of the local feedback DAC can always be chosen such that the loop filter can be allowed to include one extra sample of delay, here emphasized by defining that the middle part of the loop filter has the transfer function $z^{-1}J(z)$. For example, to implement the classical second-order $\Delta\Sigma$ modulator, A must be 2 and $J(z) = 1$. It should be observed that, by introducing the local feedback path, the $\Delta\Sigma$ quantizer's input signal $g(k) - K_0 K_{\mathrm{DAC}}$ is injected into the loop filter. This effect is highly undesirable, and it should be compensated for by adjusting the feed-forward path as shown [43]. Notice that, for $A = 2$, the feed-forward path's transfer function becomes $1 + z^{-1}$, which in fully-differential SC circuits can be implemented, e.g., by "switching" the two input capacitors in the same way as for double-sampling SC circuits.

8.3.4 Proposed Circuit-Level Implementation

Figure 8.15 shows a detailed schematic for a switched-capacitor implementation of the pipelined two-stage $\Delta\Sigma$ quantizer shown in Figure 8.14. For simplicity[17], the $\Delta\Sigma$ loop is of only second order, and it is designed to have the classical noise transfer function $\mathrm{NTF}(z) = (1 - z^{-1})^2$, which (as discussed

[17] As discussed above, the loop filter should preferably be of third order if the resolution of $d(k)$ is only 9 bits.

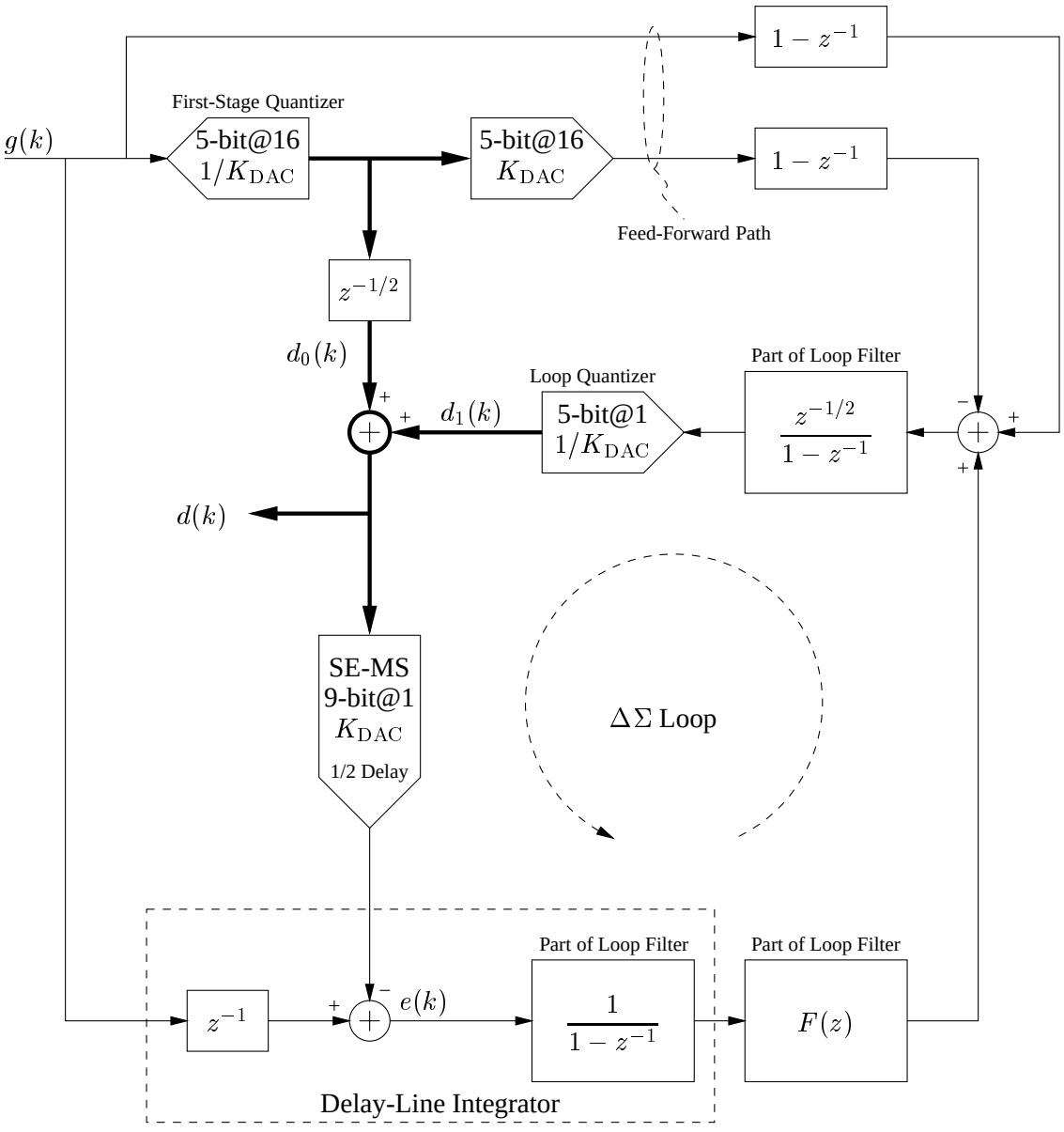


Figure 8.13: System-level implementation of the two-stage $\Delta\Sigma$ quantizer shown in Figure 8.10.

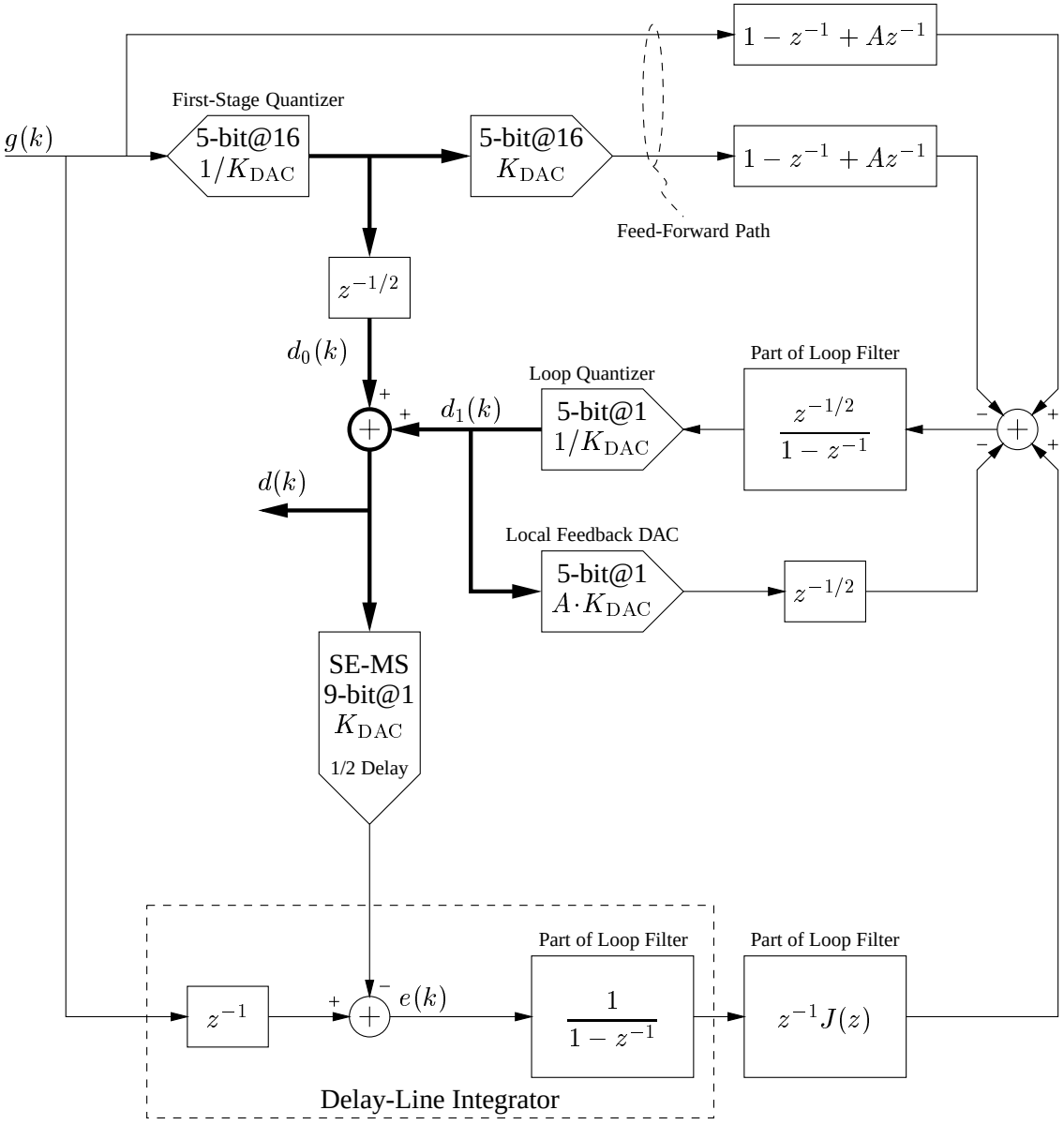


Figure 8.14: Variation of the two-stage $\Delta\Sigma$ quantizer shown in Figure 8.14, which, when implemented as a SC circuit, need not be subject to sequential settling.

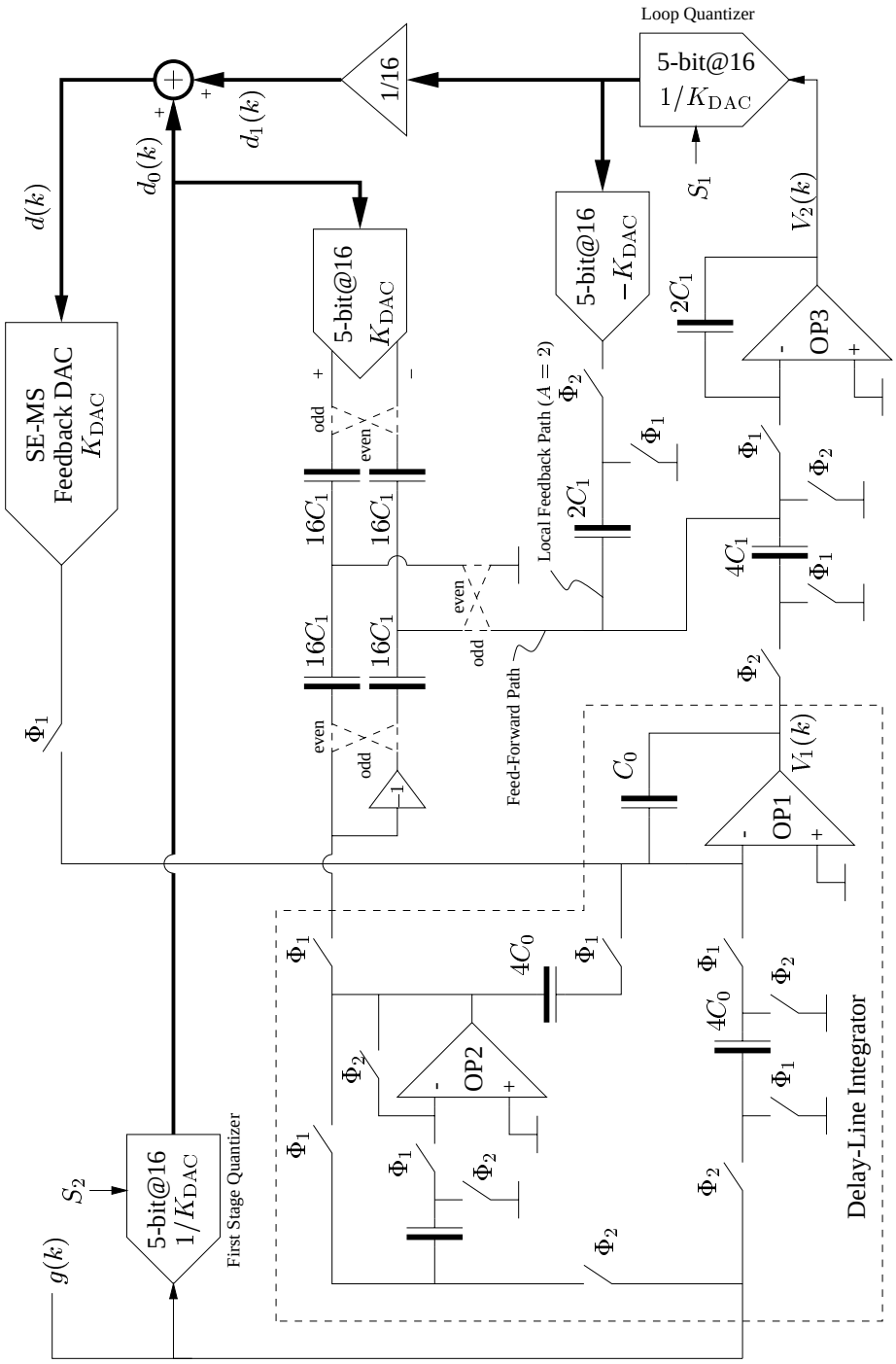


Figure 8.15: SC implementation of the pipelined two-stage $\Delta\Sigma$ quantizer shown in Figure 8.14.

above) is obtained for $J(z) = 1$ and $A = 2$. Although the circuit is shown (pseudo) single-ended, it should preferably be implemented as a fully-differential circuit.

The Schematic. The input stage is implemented as the delay-line integrator shown in Figure 8.11, where also the used clock phases are shown. The first-stage (5-bit flash) quantizer is clocked when clock phase Φ_2 ends, and the result $d_0(k)$ is immediately D/A converted and subtracted from the S/H input signal provided by OP2; the signal is represented as a charge pulse $q_{\rm FF}(k)$ conducted by the feed-forward path leading to OP3's feedback capacitor. The feed-forward path is implemented pseudo differentially, and it makes use of the switching technique commonly employed for differential double-sampling SC circuits (the dashed lines indicate that the two connections are interchanged for every increment of k). Hence, the feed-forward path will provide charge pulses of

$$q_{\rm FF}(k) = 16C_1\left([g(k) + g(k-1)] - K_{\rm DAC}[d_0(k) + d_0(k+1)]\right) \tag{8.3}$$

OP3 implements the loop filter's second and last stage, and it also implements the summation of the feed-forward signal $q_{\rm FF}(k)$, the local feedback signal $q_{\rm LFB}(k)$, and the signal from the loop filter's previous stage, OP1. These signals are represented as charge pulses dumped to the opamp's virtual-ground node, and they will cause a change in its output voltage $V_2(k)$. $V_2(k)$ represents mainly the residue from the first-stage quantization, and it should be quantized as early as possible, i.e., as soon as $V_2(k)$ has settled to 5-bit accuracy. Notice that to obtain 100 dB performance, OP3 must be allowed to settle to at least 10 – 12 bits accuracy; the loop quantizer, however, can be clocked earlier, because settling and truncation errors will be compensated for in the following samples. In general, the loop quantizer can be clocked approximately in the middle of clock phase Φ_1 (see Figure 8.11), which will leave the feedback DAC approximately the duration of clock phase Φ_2 for computations necessary for the mismatch-shaping operation.

Scaling Technique. The schematic (Figure 8.15) also shows a convenient scaling technique. Each of the loop filter's two stages amplifies the signal by a factor of four (not shown in Figure 8.14), which implies that the two flash quantizers operate with the same LSB value. This is convenient, because (this

way) they can be implemented by only one 5-bit flash quantizer multiplexed between the two functions (they are not clocked simultaneously). The scaling technique also assures appropriate full-scale signal levels, and suppression of device noise. Generally, C_1 should be much smaller than C_0 because its thermal noise is suppressed by approximately 30 dB (when referred to the input, and for OSR $= 10$).

Chapter 9

Residue-Compensated Delta-Sigma Quantizers

In Section 8.1 it was found that if the truncation error $r(k) = g(k) - d(k)K_{\mathrm{DAC}}$ is to be dominated by the feedback DAC's error signal $m(k)$, the input signal $g(k)$ must be represented by a signal $d(k)$ of at least 10 bits of resolution[1]. Whereas the complexity of the feedback DAC is practically independent of the resolution of $d(k)$ (cf. Chapter 7), it was found that it may be difficult to quantize $g(k)$ to the required high (10-bit) resolution within the given time frame. Pipeline techniques were proposed as an efficient means to alleviate this problem, but there may be situations (e.g., when designing continuous-time $\Delta\Sigma$ quantizers) where this approach is not the most suitable.

Basic Operation of Residue-Compensated Delta-Sigma Quantizers. Now referring to Figure 8.1, it may be observed that it is not a necessity that the truncation error $r(k)$ be reduced to the same level as that of the feedback DAC's error signal $m(k)$. If $d(k)$ is a coarse representation of $g(k)$ (i.e., if the truncation error is substantial), it merely implies that there is significant[2] information left in $e(k) =$

[1] As discussed in Section 4.3.1, 100 dB performance can be obtained even if the resolution of $d(k)$ is as low as 3 to 4 bits; but that involves the use of high-order (sixth to eighth order) loop filters, and there are several other reasons why this is not a good design approach. When using a second-order loop filter, approximately 10 bits of resolution is required.

[2] I.e., the signal-to-noise ratio (truncation-error to DAC-error ratio) is better than 0 dB (cf. page 212).

$g(k) - f(k) = r(k) - m(k)$, and hence a better estimate $d_g(k)$ of $g(k)$ can be obtained if $e(k)$ is quantized and the result $d_e(k)$ is added to $d(k)$

$$d_g(k) = d(k) + d_e(k) \tag{9.1}$$

Signal quantizers that compensate for the main $\Delta\Sigma$ quantizer's truncation error $r(k)$ will be called *residue-compensated $\Delta\Sigma$ quantizers*.

MASH quantizers (cf. Section 4.2), for example, are residue-compensated $\Delta\Sigma$ quantizers, but the way they estimate $e(k)$ is extremely sensitive to circuit imperfections (cf. Section 4.2.1). This Section will focus on more robust ways to estimate $e(k)$.

9.1 Directly Residue-Compensated Delta-Sigma Quantizers

Consider again the MASH structure shown in Figure 4.5. Things go wrong from the very beginning because the signal $d_q(k)$, i.e., the estimate of $q(k)$ upon which the estimate of $e(k)$ is based, depends on how well the two DACs match[3]. Preferably, $d_e(k)$ should be the simplest possible function of $e(k)$ itself, and *not* some obscure reconstruction that depends on numerous parameters which are only poorly controlled, e.g., matching of high-order analog and digital filters.

Figure 9.1 shows the simplest possible structure for residue-compensated $\Delta\Sigma$ quantizers, which will be called *directly residue-compensated $\Delta\Sigma$ quantizers*.

9.1.1 Analysis and Performance Evaluation

The three data converters' gains are all defined with respect to the same nominal value $K_{\rm DAC}$, but (as always) there will be mismatch errors, etc.. Assume that the quantization $d_e(k)$ of $e(k)$ is associated with an error signal $y(k)$

$$d_e(k) = \frac{1}{K_{\rm DAC2}}(e(k) + y(k)) \tag{9.2}$$

[3]That this is not the dominating problem only emphasizes the poor robustness of the MASH topology.

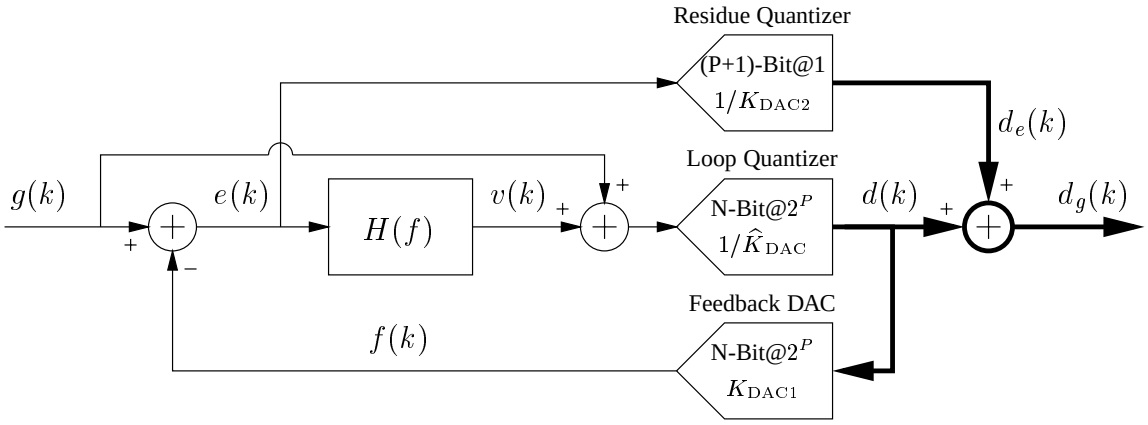


Figure 9.1: Directly residue-compensated $\Delta\Sigma$ quantizer.

which includes truncation, etc.. The residue-compensated $\Delta\Sigma$ quantizer's output $d_g(k)$ can then be calculated as

$$\begin{aligned} d_g(k) &= d(k) + d_e(k) \\ &= \frac{[f(k) - m(k)]}{K_{\mathrm{DAC1}}} + \frac{[e(k) + y(k)]}{K_{\mathrm{DAC2}}} \\ &= \frac{[g(k) - e(k) - m(k)]}{K_{\mathrm{DAC1}}} + \frac{[e(k) + y(k)]}{K_{\mathrm{DAC2}}} \\ &= \frac{[g(k)]}{K_{\mathrm{DAC1}}} + \frac{[e(k)(1 - \frac{K_{\mathrm{DAC2}}}{K_{\mathrm{DAC1}}}) - m(k)\frac{K_{\mathrm{DAC2}}}{K_{\mathrm{DAC1}}} + y(k)]}{K_{\mathrm{DAC2}}} \end{aligned} \tag{9.3}$$

Optimal Design. In Equation (9.3), the first term $\frac{[g(k)]}{K_{\mathrm{DAC1}}}$ is the desired signal, and the second term is the residue-compensated $\Delta\Sigma$ quantizer's error signal. The error signal cannot possibly be made smaller than $\frac{m(k)}{K_{\mathrm{DAC1}}}$, which is the performance that characterizes an optimally designed system. Hence, in the ideal case and when implemented in a technology having a matching index of 0.1%, the residue-compensated $\Delta\Sigma$ quantizer's error signal will have a Nyquist-band power of around -60 dBFS (full-scale), and the signal-band power will be around -100 dBFS at 10-times oversampling (cf. Figure 7.18).

Gain Errors. The factor $(1-\frac{K_{\mathrm{DAC2}}}{K_{\mathrm{DAC1}}})$ depends on the technology's matching index, and it will typically be in the order of -60 dB. If the $\Delta\Sigma$ quantizer is implemented with the feed-forward path and an N-bit

loop quantizer, the Nyquist-band power of $e(k)$ will be in the order of

$$P_{\text{av}}[e(k)] = -6 \text{ dBFS} \cdot (N-2) \tag{9.4}$$

Hence, if the loop quantizer's resolution is just a few (say, 3 to 5) bits, and the loop filter is of at least second order, the gain-error signal $e(k)(1-\frac{K_{\text{DAC2}}}{K_{\text{DAC1}}})$ will not dominate at any frequency. In other words, the performance will generally be limited only by the feedback DAC's error signal $m(k)$ and/or the residue quantizer's error signal $y(k)$[4].

Nonlinearity Errors (Truncation). The Nyquist-band power of $y(k)$ will generally depend on the magnitude of $e(k)$. If, for example, $e(k)$ is -18 dBFS (realistic when using a 5-bit loop quantizer), the target performance can be obtained if the residue quantizer is 13 to 14 bits linear. This level of performance can just barely be obtained from data quantizers (cf. Section 3.4.2), but it can easily be obtained when using signal quantizers (shown in Figure 9.2).

Dual-Loop Directly Residue-Compensated Delta-Sigma Quantizer. When analyzing the structure shown in Figure 9.2, one finds that (because $e(k)$ is small relative to full scale) $H_2(f)$ may be of lower order than $H(f)$, and also that the second feedback DAC need only be first-order mismatch-shaping. The feed-forward path (the dashed line) should preferably be implemented (in which case it may make sense to quantize $e_2(k)$ and add the result to the output), but even if this is not possible, the system will yield a good performance. Notice that even if the two separate $\Delta\Sigma$ quantizers do not have the same signal transfer function, harmonic distortion (similar to that shown in Figure 8.8) will *not* occur.

9.2 Indirectly Residue-Compensated Delta-Sigma Quantizers

Directly residue-compensated $\Delta\Sigma$ quantizers will not be discussed nor analyzed in great detail because they are impractical to implement. The problem is that $e(k)$ itself usually cannot be quantized, and

[4]The design of residue-compensated $\Delta\Sigma$ quantizers is equivalent to the design of the mismatch-shaping DACs discussed in Section 7.1.1. $m(k)$ and $y(k)$ are local nonlinearity errors, and $e(k)(1-\frac{K_{\text{DAC2}}}{K_{\text{DAC1}}})$ is a gain error.

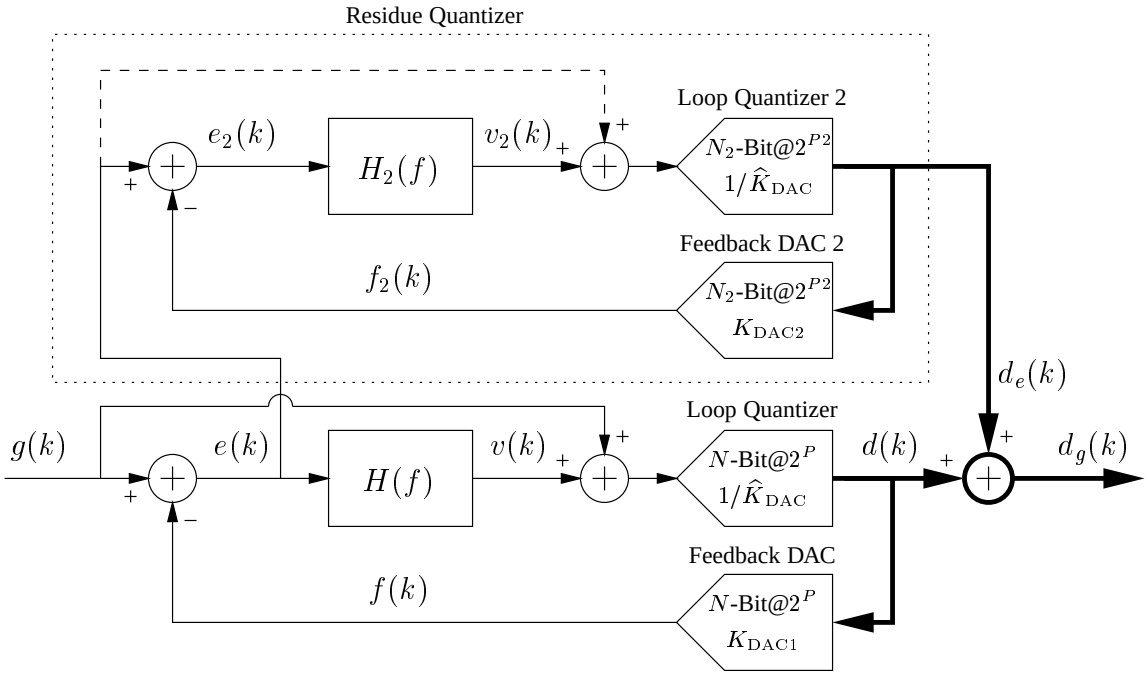


Figure 9.2: Dual-loop directly residue-compensated $\Delta\Sigma$ quantizer.

generating $d_e(k)$ on the basis of a reconstruction of $e(k)$ is usually troublesome (see Footnote 13 on page 248).

The loop filter's $H(f)$ input stage is almost always an integrator, where $e(k)$ is a charge or current signal that is dumped into an integrating capacitor. Because charge and current signals exist in only one copy (which is used for the integrating capacitor), the estimate $d_e(k)$ generally cannot be generated from $e(k)$ itself.

When $e(k)$ is dumped into the integrating capacitor, the outcome is a voltage signal (say $o(k)$) which is proportional to the integral (sum) of $e(k)$. An advantage of voltage signals is that they can be used simultaneously for several purposes, i.e., an arbitrary number of accurate copies are directly available. Hence, the estimate $d_e(k)$ can be obtained by quantizing the first-order difference of $o(k)$, or even better by calculating the first-order difference of an estimate $d_o(k)$ of $o(k)$. This concept, which is similar to the generalized filtering principle (cf. Section 7.5.1), is shown in Figure 9.3. Quantizers implemented in this topology will be called *indirectly residue-compensated* $\Delta\Sigma$ *quantizers*.

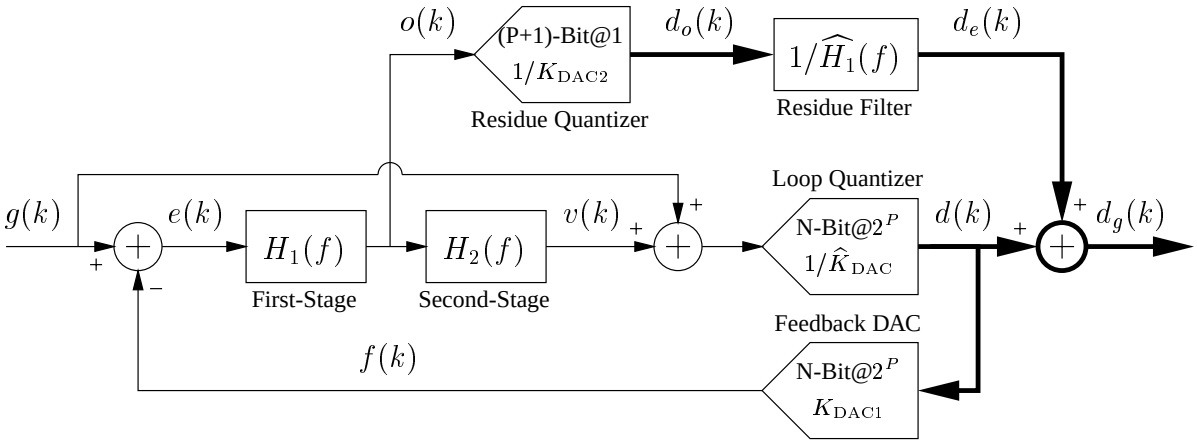


Figure 9.3: Indirectly residue-compensated $\Delta\Sigma$ quantizer.

9.2.1 Analysis and Performance Evaluation

The quantization $d_o(k)$ of $o(k)$ will be associated with an error signal (say, $y_o(k)$)

$$d_o(k) = \frac{o(k) + y_o(k)}{K_{\mathrm{DAC2}}} \tag{9.5}$$

Hence, the estimate $d_e(k)$ of $e(k)$ can be expressed as

$$\begin{aligned} d_e(k) &= \widehat{h_1}^{-1}(k) * d_o(k) \\ &= \frac{\widehat{h_1}^{-1}(k) * (o(k) + y_o(k))}{K_{\mathrm{DAC2}}} \\ &= \frac{\widehat{h_1}^{-1}(k) * (h_1(k) * e(k) + y_o(k))}{K_{\mathrm{DAC2}}} \\ &= \frac{e(k)}{K_{\mathrm{DAC2}}} + \frac{[\widehat{h_1}^{-1}(k) * h_1(k) - 1] * e(k) + \widehat{h_1}^{-1}(k) * y_o(k)}{K_{\mathrm{DAC2}}} \end{aligned} \tag{9.6}$$

The first term in Equation (9.6) corresponds to $\frac{e(k)}{K_{\mathrm{DAC2}}}$ in Equation (9.3), which was found to cause only the nondominating gain error $e(k)(1 - \frac{K_{\mathrm{DAC2}}}{K_{\mathrm{DAC1}}})$ in $d_g(k)$. The second term in Equation (9.6) corresponds to $\frac{y(k)}{K_{\mathrm{DAC2}}}$ in Equation (9.3), and it is, therefore, concluded that indirectly residue-compensated $\Delta\Sigma$ quantizers are optimally designed when the feedback DAC's error signal $m(k)$ dominates

$$\begin{aligned} y(k) &= [\widehat{h_1}^{-1}(k) * h_1(k) - 1] * e(k) + \widehat{h_1}^{-1}(k) * y_o(k) \\ &= y_{\mathrm{match}}(k) + y_{\mathrm{quan}}(k) \end{aligned} \tag{9.7}$$

In Equation (9.7), the first error term $y_{\rm match}(k) = [\widehat{h_1^{-1}}(k) * h_1(k) - 1] * e(k)$ is caused by mismatch of the analog/digital filters, whereas the second error term $y_{\rm quan}(k) = \widehat{h_1^{-1}}(k) * y_o(k)$ is caused by the residue quantizer's nonlinearity. The two terms will be considered separately.

9.2.2 Controlling the Residue-Quantization Error

If the residue quantizer is a data quantizer, then $y_o(k)$ will be a harmonic distortion of $o(k)$. Because $o(k)$ is a "non-tonal" wide-band signal, $y_o(k)$ will be a white-noise-like error signal. Hence, assuming that $1/H_1(f)$ is a Nth order differentiation, the signal-band power of $y_{\rm quan}(k)$ can be estimated using Figure 4.11.

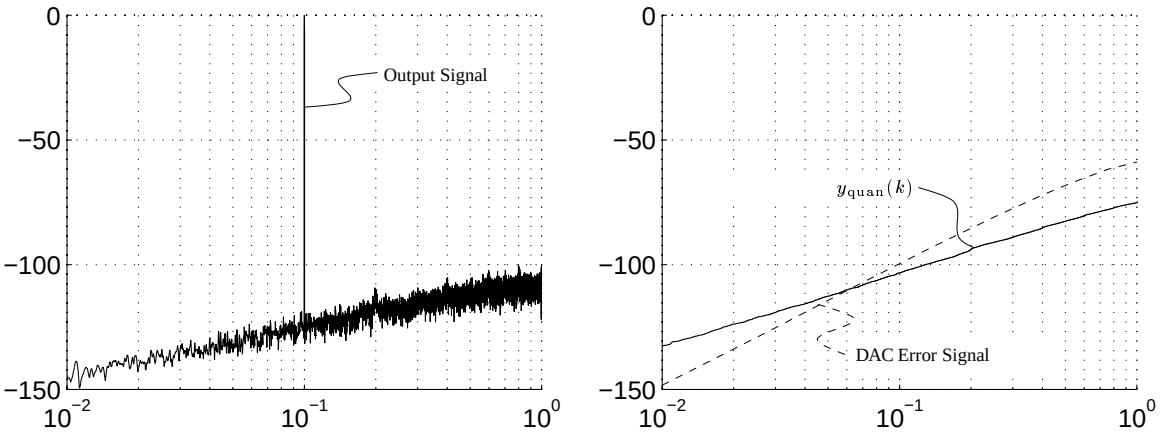
For example, if $1/H_1(f)$ is a first-order differentiation, the signal-band power of $y_{\rm quan}(k)$ will be approximately 25 dB below the Nyquist-band power of $y_o(k)$ (at 10 times oversampling). In general, the Nyquist-band power of $y_o(k)$ can be made at least 50 to 60 dB below full scale of $o(k)$, which, for a 5-bit loop quantizer, already will be about 20 dB below full-scale of $g(k)$. In other words, it can easily be assured that the Nyquist-band power of $y_o(k)$ is in the order of -70 dBFS to -80 dBFS. In conclusion, if the residue quantizer is just 8 to 10 bit linear[5] and $1/H_1(f)$ is a first-order differentiation, the signal-band power of $y_{\rm quan}(k)$ will be approximately -95 to -105 dBFS (at OSR $= 10$, and when using a 5-bit loop quantizer).

If $1/H_1(f)$ is (say) a second-order differentiation, $y_o(k)$ will be suppressed by an extra 12 dB (at OSR $=$ 10), but since the magnitude of $o(k)$ will increase, the overall improvement is only in the order of 6 dB.

Figure 9.4 shows the performance obtained (by simulation) when the residue quantizer is a 10-bit data quantizer. The slight tonality is due to idle tones in $e(k)$, but they can be avoided by dithering the $\Delta\Sigma$ quantizer. The resolution of $d_g(k)$ is high (13 bit), which is reflected by the low Nyquist-band power of $y_{\rm quan}(k)$, but it is dominated by the feedback DAC's error signal.

Using a Delta-Sigma Quantizer as the Residue Quantizer. Alternatively, the residue quantizer can be designed as a $\Delta\Sigma$ quantizer. Then, the resolution of $d_o(k)$ need not be as high as 8 nor 10 bits because

[5] Both the resolution and the linearity must be 8 to 10 bits.



(a) Spectral power density versus frequency

(b) Signal-band power versus 1/OSR

Figure 9.4: Performance of the indirectly residue-compensated $\Delta\Sigma$ quantizer when the residue quantizer is a 10-bit data quantizer.

the truncation error $r_2(k) = o(k) - d_o(k)K_{\mathrm{DAC2}} \simeq y_o(k)$ will be at least first-order shaped, and because $1/H_1(f)$ will provide additional signal-band suppression of $y_o(k)$.

The simplest option is to design the residue quantizer as a first-order $\Delta\Sigma$ quantizer, e.g., as shown in Figure 9.5. The two loop quantizers' resolving ranges overlap by 1 bit, hence the resolution of the output signal $d_g(k)$ is 9 bit. To illustrate the option (which is an important one, because it often makes the circuit *significantly* simpler to implement[6]), the loop filter's first stage is designed to be delaying. Consequently, because the residue filter must be causal[7], $d(k)$ is delayed to align $d(k)$ and $d_e(k)$ time wise[8].

The main $\Delta\Sigma$ quantizer's second loop filter stage can be of almost any order, but it should usually be designed with a modest $\mathrm{NTF}_{\max}$ value to prevent $o(k)$ from attaining large values[9].

[6]Timing issues become less of a concern because the delay "breaks" the row of sequential operations (the general pipeline technique/idea). It is often preferable if the loop filter's first stage delays $e(k)$ by only one half clock cycle, because then, the two loop quantizers can be implemented using only one multiplexed flash quantizer (as suggested in a previous example).

[7]The residue filter cannot be made to implement $\frac{1-z^{-1}}{z^{-1}} = z - 1$.

[8]This "trick" is also used when the residue quantizer is a pipeline quantizer.

[9]There are, however, several design aspects in addition to that of preventing large values of $o(k)$. Sometimes, higher

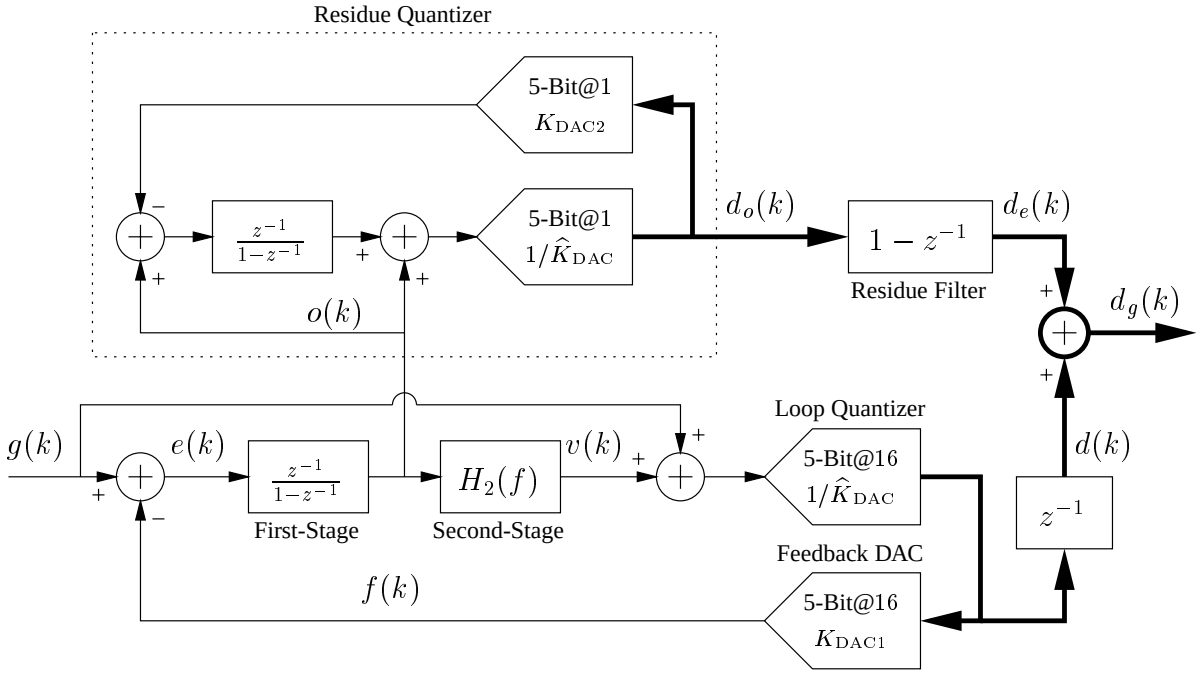
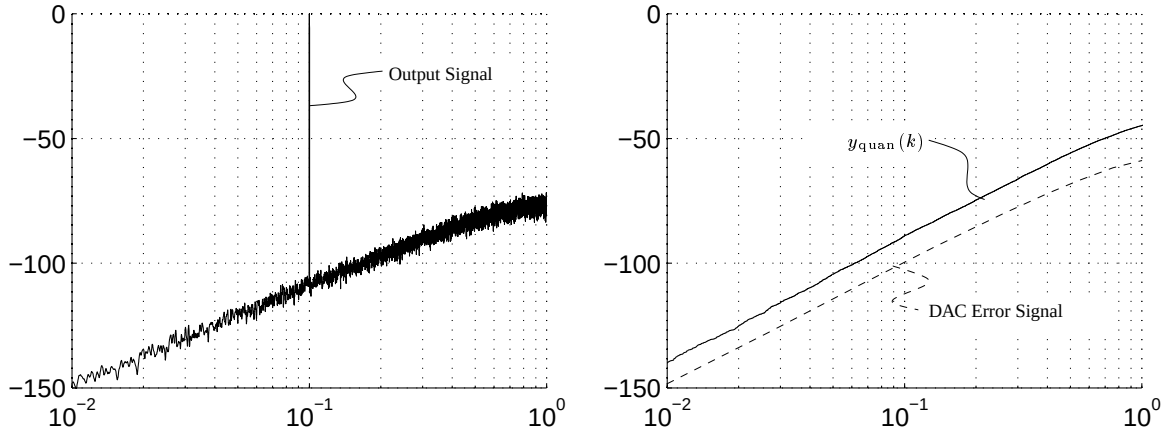


Figure 9.5: Indirectly residue-compensated $\Delta\Sigma$ quantizer for which the residue quantizer is a 5-bit first-order $\Delta\Sigma$ quantizer.

Simulation Results. Figure 9.6 shows the performance obtained (by simulation) when the main $\Delta\Sigma$ quantizer's loop filter is of second order with $\text{NTF}_{\max} = 1.5$. It can be observed that this system is *not* optimally designed because $y_{\text{quan}}(k)$ dominates the DAC's error signal. This is because the resolution of $d_g(k)$ is only 9 bits, and because the truncation error is only second-order shaped (furthermore, the residue quantizer had to be dithered to avoid idle tones.



(a) Spectral power density versus frequency

(b) Signal-band power versus 1/OSR

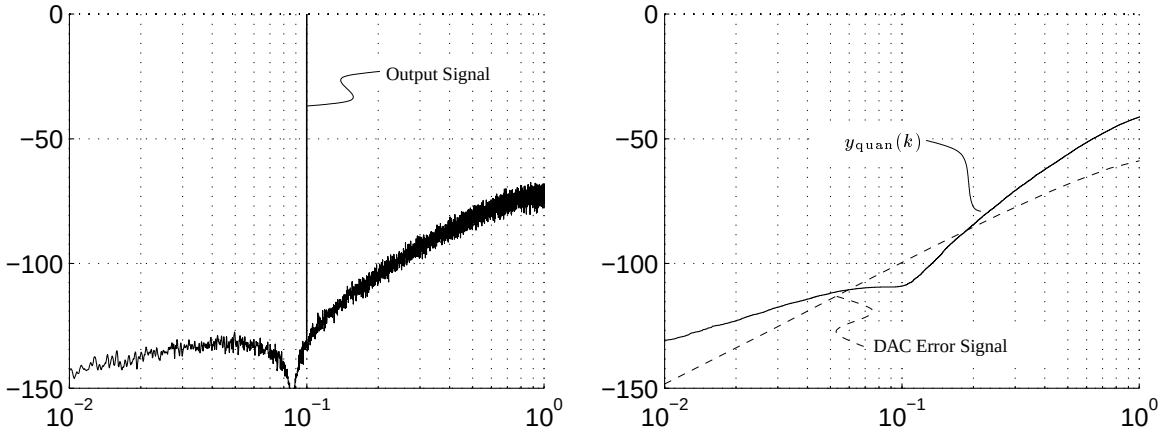
Figure 9.6: Performance of the indirectly residue-compensated $\Delta\Sigma$ quantizer shown in Figure 9.5.

To improve the signal-band performance, the system was also simulated for a design where the residue quantizer is a $\Delta\Sigma$ quantizer with a second-order loop filter. The loop filter was designed with a high $\text{NTF}_{\max}$ value (almost 4), and it was designed to resonate at a high signal-band frequency[10] to efficiently reduce the signal-band power of the residue quantizer's truncation error. The performance obtained (by simulation) from this system is shown in Figure 9.7. Now, the signal-band performance is dominated by the DAC's error signal (optimal design), but the error signal's Nyquist-band power is still fairly large due to the "low" resolution of $d_g(k)$ and to the aggressive design of the residue quantizer's loop filter.

$\text{NTF}_{\max}$ values will be chosen (even as high as 4 or 6) to reduce the gain error's $e(k)(1 - \frac{K_{\text{DAC2}}}{K_{\text{DAC1}}})$ signal-band power by shaping $e(k)$ efficiently (cf. Equation 9.3). The tradeoff is that the residue quantizer must have a larger resolving range, in which case it makes good sense to design the residue quantizer as an indirectly residue-compensated $\Delta\Sigma$ quantizer (recursive use).

[10] More precisely, the loop filter was designed according to Figure 4.4, where $\gamma_1 = 0.0736$.

If desired, the error signal can be reduced further by designing the residue quantizer as an indirectly residue-compensated $\Delta\Sigma$ quantizer employing a simple low-resolution (say, 6 bit) data quantizer for the estimation of $r_2(k)$ (in which case the residue quantizer need only employ a first-order loop filter).



(a) Spectral power density versus frequency

(b) Signal-band power versus 1/OSR

Figure 9.7: Performance of the indirectly residue-compensated $\Delta\Sigma$ quantizer shown in Figure 9.5, where the residue quantizer is designed with an aggressive second-order filter.

Loop Filter Topology. Not all loop filters $H(f)$ can be sectioned in two terms $H_1(f)$ and $H_2(f)$ as shown in Figure 9.3, where $H_1(f)$ is an integrator. Now referring to Figure 4.4, the considered separation can be performed only if $\gamma_1 = 0$, i.e., if the first biquad is designed to not resonate (which implies that the quantizer's noise transfer function will have at least one zero at dc – two if the filter's order is even).

If the indirectly residue-compensated $\Delta\Sigma$ quantizer is designed to have a bandpass characteristic, then $H_1(f)$ should be the first (resonating) biquad. It is important that $e(k)$ is the only input to $H_1(f)$, i.e., that the $\Delta\Sigma$ quantizer's stability is assured by means of the shown feed-forward branches $\alpha_1, \alpha_2, \alpha_3, \alpha_4$, rather than the feedback branches that are often used.

The above comments, of course, also apply to SE-MS DACs implemented according to the generalized filtering principle.

9.2.3 Controlling the Filter-Mismatch-Induced Error

Residue-compensated $\Delta\Sigma$ quantizers implemented in the MASH topology are extremely sensitive to filter-mismatch-induced errors, which often limits the performance considerably (cf. Section 4.2.1). An important advantage of the proposed indirectly residue-compensated $\Delta\Sigma$ quantizers (cf. Figure 9.3) is that they are much more robust with respect to such errors. The following discussion will analyze this aspect.

Calculating the Error's Spectral Composition. The filter-mismatch-induced error $y_{\mathrm{match}}(k)$ was found to be (cf. Equation (9.7))

$$y_{\mathrm{match}}(k) = [\widehat{h_1}^{-1}(k) * h_1(k) - 1] * e(k) \tag{9.8}$$

where $\widehat{h_1}^{-1}(k)$ and $h_1(k)$ are the impulse responses of $1/\widehat{H_1}(f)$ and $H_1(f)$, respectively. In other words, $y_{\mathrm{match}}(k)$ can be considered as being generated by filtering $e(k)$ with the filter

$$H_{\mathrm{match}}(f) = \left(\frac{H_1(f)}{\widehat{H_1}(f)} - 1 \right) \tag{9.9}$$

The spectral composition of $e(k)$ will be modeled as a white-noise signal[11] $q(k) \leftrightarrow Q(f)$ filtered by the $\Delta\Sigma$ quantizer's noise transfer function $\mathrm{NTF}(f) = \frac{1}{1+H_1(f)H_2(f)}$. Hence, the spectral composition of $y_{\mathrm{match}}(k) \leftrightarrow Y_{\mathrm{match}}(f)$ can be calculated from

$$Y_{\mathrm{match}}(f) = H_{\mathrm{match}}(f) \cdot \mathrm{NTF}(f) \cdot Q(f) \tag{9.10}$$

Equation (9.10) shows that the loop quantizer's truncation error $q(k)$ will be suppressed by both $H_{\mathrm{match}}(f)$ and $\mathrm{NTF}(f)$; $H_{\mathrm{match}}(f)$ expresses suppression due to the residue-compensation process, whereas $\mathrm{NTF}(f)$ expresses suppression obtained by noise shaping.

[11] As discussed in Section 4.1.1, the white-noise assumption for $q(k)$ can be hard to justify. However, since the validity of this discussion does not depend on the (lack of) autocorrelation of $q(k)$, the model has been allowed for simplicity. If the multi-bit $\Delta\Sigma$ quantizer is dithered, the model is fully justifiable.

Estimation of the Mismatch Factor $H_{\mathrm{match}}(f)$. When $H_1(f)$ is a simple filter function, $H_{\mathrm{match}}(f)$ can be estimated qualitatively. The following discussion will consider the typical case where $H_1(f)$ nominally is a first-order integrator, i.e., $\widehat{H_1}(f) = \frac{1}{1-z^{-1}}$ (it is irrelevant whether or not the integrator delays $e(k)$).

The transfer function of a switched-capacitor integrator based on a linear opamp with finite-gain is [29]

$$H_1 = \frac{1}{1-(1-\mu)z^{-1}} \tag{9.11}$$

where μ is the reciprocal of the opamp's gain (μ is typically in the range from -40 to -100 dB).

$H_{\mathrm{match}}(f)$ can then be calculated from

$$\begin{aligned} H_{\mathrm{match}}(f) &= \frac{1-z^{-1}}{1-(1-\mu)z^{-1}} - 1 \\ &= \frac{1-z^{-1}-1+(1-\mu)z^{-1}}{1-(1-\mu)z^{-1}} \\ &= \frac{-\mu z^{-1}}{1-(1-\mu)z^{-1}} \end{aligned} \tag{9.12}$$

The frequency response (magnitude) of $H_{\mathrm{match}}(f)$ is shown in Figure 9.8 for a select set of μ values. It can be observed that $H_{\mathrm{match}}(f)$ is a first-order low-pass filter with 0 dB gain at 0 Hz and approximately -6 dB $+ 20\log_{10}(\mu)$ dB gain at the Nyquist frequency.

When using this estimate to evaluate the expression for $Y_{\mathrm{match}}(f)$ in Equation (9.10), the following observations can be made:

- For a first-order $\Delta\Sigma$ quantizer with NTF$(f) = (1-z^{-1})$, the frequency dependence of the two transfer functions $H_{\mathrm{match}}(f)$ and NTF(f) will cancel almost perfectly, i.e, $Y_{\mathrm{match}}(f) \simeq \mu Q(f)$. Thus, $Y_{\mathrm{match}}(f)$ will be a small white-noise-like error signal for which the Nyquist-band power is inversely proportional to the loop quantizer's resolution and the opamp gain.

  Simulations show that to obtain 100 dB performance at OSR $= 10$, the opamp gain must be at least 91 dB $- N \cdot 6$ dB, where N is the loop quantizer's resolution.

- If the opamp gain is low, i.e., if the filters $H_1(f)$ and $1/\widehat{H_1}(f)$ match poorly, the loop filter's order and/or $\mathrm{NTF}_{\max}$ value can be increased to suppress $Y_{\mathrm{match}}(f)$ by means of NTF(f). As discussed

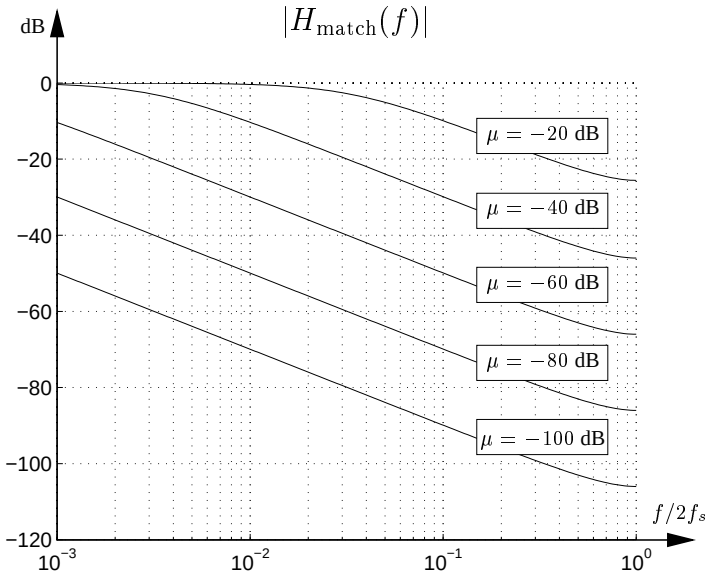


Figure 9.8: The magnitude response of $H_{\mathrm{match}}(f)$ when $H_1(f)$ is a first-order integration. μ is the integration's relative "leakage".

on page 243, NTF(f) will (for baseband systems) always have a zero at 0 Hz. This zero will cancel the pole in $H_{\mathrm{match}}(f)$, and the remaining poles and zeroes in NTF(f) will characterize the spectral composition of $Y_{\mathrm{match}}(f)$. If, for example, NTF(f) is of second order, $Y_{\mathrm{match}}(f)$ will be first-order shaped. Some important details are discussed below.

Evaluation. The filter-mismatch-induced error signal $y_{\mathrm{match}}(k)$ will only rarely be a problem because it is a white-noise signal $q(k)$ filtered by a blocking filter NTF$(f)H_{\mathrm{match}}(f)$, i.e., the high-pass filter NTF(f) is followed by the low-pass filter $H_{\mathrm{match}}(f)$.

The error signal $y_{\mathrm{match}}(k)$ can be suppressed in many ways. Because the power of $q(k)$ is proportional to the loop quantizer's resolution, the attenuation required of the blocking filter NTF$(f)H_{\mathrm{match}}(f)$ is reduced as the resolution of $d(k)$ is increased. For example, by increasing the loop quantizer's resolution from 1 to 5 bits, the attenuation required from NTF$(f)H_{\mathrm{match}}(f)$ is reduced by 24 dB.

The low-pass filter's $H_{\mathrm{match}}(f)$ frequency response depends on stochastic processes, and it cannot be designed directly. However, the filter's maximum gain is always 0 dB, and the cutoff frequency and the

stop-band attenuation are functions of only the opamp gain. Usually it is possible to worst-case evaluate an opamp gain, hence the frequency response $H_{\rm match}(f)$ can be worst-case evaluated as well (using Figure 9.8).

When the loop quantizer's resolution and the worst-case response of $H_{\rm match}(f)$ is known, the $\Delta\Sigma$ quantizer's noise transfer function NTF(f) should be designed such that the feedback DAC's error signal $m(k)$ will dominate $y_{\rm match}(k)$. If the opamp gain is reasonable high, there is usually no problem at all. Table 9.1 shows the opamp gain required to obtain 100 dB performance at OSR $= 10$ when using a 5-bit loop quantizer and when the loop filter is of reasonable low order. It can be observed that almost any loop filter can be used if the opamp gain is only 60 dB or higher. It can also be observed that even very-low-gain opamps can be used if the loop filter's order and NTF$_{\rm max}$ value are increased[12].

Opamp Gain	First Order	Second Order	Third Order	Fourth Order
NTF$_{\rm max} = 1.5$	65 dB	60 dB	59 dB	58 dB
NTF$_{\rm max} = 2.0$	61 dB	53 dB	48 dB	42 dB
NTF$_{\rm max} = 2.5$		50 dB	44 dB	35 dB
NTF$_{\rm max} = 3.0$		48 dB	40 dB	31 dB
NTF$_{\rm max} = 3.5$		47 dB	38 dB	28 dB
NTF$_{\rm max} = 4.0$		46 dB	37 dB	25 dB

Table 9.1: Opamp gain required to suppress the signal-band (OSR $= 10$) power of the filter-mismatch induced error $y_{\rm match}(k)$ to -100 dBFS (5-bit loop quantizer). The suppression is proportional to the opamp gain and the loop quantizer's resolution.

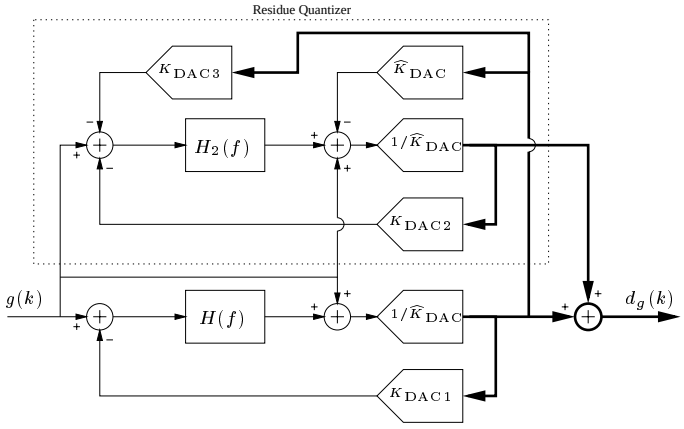
[12]If the loop quantizer is a single-bit device, the NTF$_{\rm max}$ value must be as low as 1.5 to preserve stability. In that case, to obtain the target performance, the opamp must have a gain of at least 85 dB for any order of the loop filter (because noise shaping is not efficient for frequencies higher than $f_s/20$ when NTF$_{\rm max} = 1.5$). This will usually not pose a problem, which reflects that this topology is superior to the MASH topology.

9.2.4 Designing Residue-Compensated Delta-Sigma Quantizers

Residue-compensated quantizers are best implemented in the topology shown in Figure 9.3. The resolution of the output $d_g(k)$ should preferably be fairly high (say, 8 to 10 bits or more) to avoid that $d_g(k)$ includes high-order shaped error signals (to avoid a requirement for complex digital filters). For this and several other reasons, the loop quantizer's resolution should not be too low – 4 to 5 bits seems to be a good choice.

Designing the Loop Filter. To avoid dominating filter-mismatch errors, the loop filter's first stage should be simple. Since $H_1(f) = 1$ usually is not practical[13]; a first-order integrator is the preferred choice for baseband quantizers. If the opamp gain is sufficiently high (cf. Table 9.1), filter-mismatch errors need not be considered when designing the loop filter's second stage. The loop filter $H_1(f)H_2(f)$ should instead be designed such that the gain-mismatch error $e(k)(1-\frac{K_{\rm DAC2}}{K_{\rm DAC1}})$ is sufficiently suppressed. The factor $(1-\frac{K_{\rm DAC2}}{K_{\rm DAC1}})$ can usually be made small (say, -50 to -60 dB). Since the power of $e(k)$ is

[13] It may not be the simplest option, but very good systems can indeed be designed if $e(k)$ is reconstructed and quantized by the residue quantizer. This option is illustrated below. The main reconstruction DAC (DAC3) takes the place of the main $\Delta\Sigma$ quantizer's feedback DAC (DAC1) as the system's most critical element. Hence, DAC3 and possibly also DAC2 should be mismatch-shaping, whereas DAC1 need not be mismatch-shaping. A key point is that $K_{\rm DAC3}$ need not match $K_{\rm DAC1}$ perfectly because mismatch will cause an error which is proportional to $g(k)$ (a linear error). The topology is particularly useful for bandpass quantizers, where designing $H_1(f) = \frac{z^{-1}}{1+z^{-2}}$ may be associated with an intolerable large filter-mismatch-induced error. Hence, on occasion, the circuit's higher complexity may be well worth while.



inversely proportional to the loop quantizer's resolution, it is usually sufficient that NTF(f) suppresses $q(k)$ (in the signal-band) by as little as 20 to 30 dB when using a 5-bit loop quantizer. This can be obtained with a second order loop filter. If the loop quantizer's resolution is low, and/or if $(1-\frac{K_{\mathrm{DAC2}}}{K_{\mathrm{DAC1}}})$ is considerably higher than -50 dB, the loop filter should be of higher (say, third or fourth) order and the $\mathrm{NTF}_{\max}$ value should be increased (to say, 3 or 4), in which case the residue quantizer must be designed with a wider resolving range.

Designing the Residue Quantizer. The residue quantizer's step size must be small enough to assure that $d_g(k)$ has the required resolution. For example, to obtain 9-bit resolution of $d_g(k)$ when using a 5-bit loop quantizer, the resolution of $d_o(k)$ must be at least 5 bits. If the loop filter $H_1(f)H_2(f)$ is designed with a high $\mathrm{NTF}_{\max}$ value, the resolution of $d_o(k)$ must be higher.

Considering that $1/H_1(f)$ is a first-order high-pass filter, $o(k)$ cannot be quantized directly (with a data quantizer) to the discussed minimum resolution of $d_o(k)$. If that is done, the residue quantizer's truncation error will cause a dominating error in $d_g(k)$. This problem can be avoided either by increasing the resolution of $d_o(k)$ using a higher-resolution data quantizer or by shaping the truncation error by using a $\Delta\Sigma$ quantizer.

If the residue quantizer is designed as a data quantizer, the truncation error will be suppressed by only about 24 dB (at OSR $= 10$, cf. Figure 4.11), hence the resolution of $d_g(k)$ must be at least 12 bits. Hence, if the loop quantizer's resolution is (say) 5 bits, the residue quantizer's resolution must be at least 8 or 9 bits. The residue quantizer can, for example, be implemented as a pipeline quantizer, in which case $d(k)$ must be delayed accordingly before it is added to $d_e(k)$ (cf. Figure 9.5).

If the residue quantizer is designed as a $\Delta\Sigma$ quantizer, the truncation error can be efficiently shaped, and the resolution of $d_o(k)$ can be as low as the discussed minimum. If the resolution of $d(k)$ and $d_o(k)$ are both 5 bits, the residue quantizer should be of at least second order (see Figures 9.6 and 9.7). A better/simpler option is to design the residue quantizer as an indirectly residue-compensated $\Delta\Sigma$ quantizer with a first-order loop filter and with a low-resolution data quantizer as the (internal) residue quantizer[14].

[14] A simple design would use a second-order main loop filter, a 4-bit loop quantizer for $d(k)$, and generate $d_o(k)$ as a 4-bit

Timing the System. An important advantage of indirectly residue-compensated $\Delta\Sigma$ quantizers is that they can easily be pipelined – simply by making $H_1(f)$ delaying by one half of a full clock cycle. An example was shown in Figure 9.5, but the concept can be generalized and used throughout the structure (this is especially convenient when the residue quantizer incorporates a pipeline data quantizer).

Conclusion. The conclusion is good. Very robust high-performance signal quantizers can be implemented in the described topologies, and the circuit complexity is modest. Many variations are possible; the topology can be tailored to suit almost any need.

9.3 Continuous-Time Delta-Sigma Quantizers

As discussed in Section 4.5.2, continuous-time $\Delta\Sigma$ quantizers can be designed to yield a better performance than their discrete-time counterparts. However, they are generally more difficult to design because dynamic errors and circuit timing are tricky issues. This section will consider some of the new and interesting design options that are available when using the techniques described in this chapter and in Chapter 8.

9.3.1 High-Resolution Continuous-Time Delta-Sigma Quantizers

Consider Figure 4.19. As discussed above, $e(t)$ should preferably be made as small as possible. There is, however, a limit to how small $e(t)$ can be made, and this limit is independent of how well the system is designed. The problem is that the feedback signal $f_h(t)$ is constant for an entire clock cycle, whereas the input signal $g(t)$ varies continuously. If $g(t)$ is only 10 times oversampled, $e(t)$ may attain values

noise-shaped representation of $o(k)$, which is compensated with a 4-bit-differentiated (which results in a 5-bit) representation of the truncation. This way, the resolution of $d_g(k)$ will be around 10 bits, and the errors will all be small and second-order shaped. The main feedback DAC must (of course) be second-order mismatch-shaping, but the residue quantizer need only employ a first-order mismatch-shaping feedback DAC. In fact, the residue quantizer's feedback DAC may not need to be mismatch-shaping because $o(k)$ is small, and all errors (including the DAC error) in $d_o(k)$ will be first-order shaped by $1/\widehat{H_1}(f)$.

that are as large as one third of full scale; absolutely nothing can be done to prevent that. This is not even the worst case because $g(t)$ may not have been processed by an anti-aliasing filter (this option was emphasized as an advantage of CT $\Delta\Sigma$ quantizers), hence the "sample-to-sample" variation of $g(t)$ may be substantial.

To understand the complexity of designing high-resolution CT $\Delta\Sigma$ quantizers, it should be observed that the feedback DAC usually will be delaying, partly because the spectral encoder will cause delay, and partly because it may be necessary to employ a delaying switching scheme to prevent dynamic DAC errors (cf. Chapter 5 and Figure 5.2). Hence, to minimize the magnitude of $e(t)$, it is necessary to look ahead in time and estimate the median of $g(t)$ for the *next* sample. This is not simple to do because CT analog delay lines can be implemented only as analog all-pass filters with a constant group delay. In other words, the pipeline technique discussed in Chapter 8 is not directly applicable for CT $\Delta\Sigma$ quantizers.

Predictive Quantizers. Figure 9.9 shows an interesting option which can be used to obtain an output signal $d(k)$ of higher resolution than the loop quantizer's resolution[15].

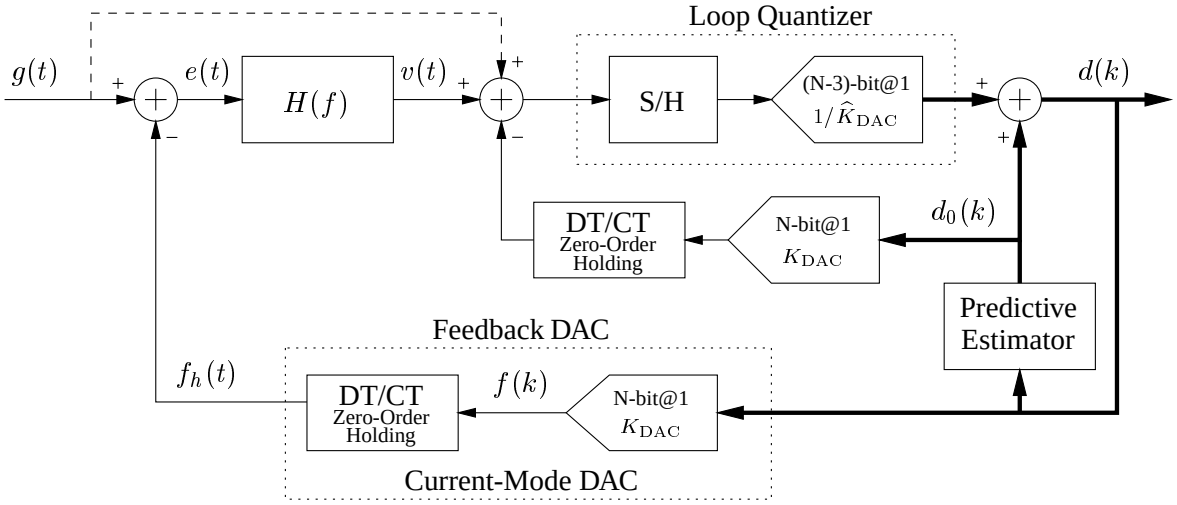


Figure 9.9: Predictive continuous-time $\Delta\Sigma$ quantizer.

[15]This topology can, obviously, also be used for DT $\Delta\Sigma$ quantizers, but the systems discussed in Chapter 8 are usually preferable for that purpose.

The basic idea is to let a predictive estimator predict the next sample as accurately as possible, and use the loop quantizer only to estimate the prediction's error. If the prediction $d_0(k)$ is accurate, the resolution of $d(k)$ can be vastly higher than the loop quantizer's resolution.

Good predictions can, however, only be made for signals that have a low relative bandwidth[16]. A low bandwidth can be mimicked by *omitting* the $\Delta\Sigma$ quantizer's feed-forward path (the dashed line) because, then, the loop filter will act as a low-pass filter (for a baseband quantizer).

A more sophisticated technique[17] which can be used to improve the prediction's accuracy is to design the $\Delta\Sigma$ quantizer with multiple feedback DACs connected to the input of each of the loop filter's integrators (as shown in Figure 5.8 in [1]) and *not* compensate for the injected signal component by omitting the feed-forward paths from $g(t)$, i.e., letting $b_i = 0, i > 1$ in Figures 5.9 and 5.10 in [1][18]. The purpose of this deliberately unusual design is to assure that $g(t)$ will reach the loop quantizer only after some delay. By quantizing not only the loop filter's output signal, but also one or several of the loop filter's internal state variables, a highly-accurate prediction $d_0(k)$ can be obtained (not shown).

An important aspect to consider for predictive $\Delta\Sigma$ quantizers[19] is that the predictive process potentially can become unstable (or never even stabilize), in which case the $\Delta\Sigma$ quantizer will malfunction. To prevent this scenario, the loop quantizer can be designed to have a wide resolving range with a large step size (to assure the stability even if the error in $d_0(k)$ is large) and combine it with a smaller resolving range[20] with a smaller step size (to obtain high performance once the predictive process has converged/stabilized). Many variations/improvements are possible. For example, the loop quantizer can be a high-resolution subranging quantizer, where $d_0(k)$ represents the first "quantization," and where the step size of the second quantization is variable and a decreasing function of a significance factor

[16] This includes, in particular, narrow-band bandpass quantizers, where the relation $d(k+2) \simeq -d(k)$ will hold with a very good accuracy. In this case, linear prediction is a delay combined with a change of polarity.

[17] Which also violates the previously discussed general rules for how to design good $\Delta\Sigma$ quantizers, but that only emphasizes how different DT and CT $\Delta\Sigma$ quantizers are to design.

[18] Which is a direct violation of the rule outlined in [43].

[19] Predictive $\Delta\Sigma$ quantizers are only exemplified by Figure 9.9. This class of quantizers is generalized by allowing the prediction $d_0(k)$ to be based on other and more information than just $d(k)$.

[20] Centered around zero.

that reflects the reliability of $d_0(k)$ (a small digital state machine is required for this purpose). Another interesting option is to use a simple linear prediction (for example $d_0(k+1) = 0$, $d_0(k+1) = d(k)$, or $d_0(k+1) = 2d(k) - d(k-1)$) and design the loop quantizer with logarithmically-spaced quantization levels[21], in which case the quantizer's SER performance will be approximately constant for a wide range of input signals (because the error in $d_0(k)$ largely will be proportional to the magnitude of $g(t)$).

9.3.2 Residue-Compensated Continuous-Time $\Delta\Sigma$ Quantizers

Continuous-time $\Delta\Sigma$ quantizers can also be designed in the residue-compensating topology shown in Figure 9.3, but there are some important differences when the loop filter is a continuous-time filter. The following refers to a CT equivalent of Figure 9.3, i.e., replace the time variable k by t and insert a holding DT/CT converter at the output of the feedback DAC.

The Residue Quantizer's Resolving Range. The fundamental operation of the CT $\Delta\Sigma$ quantizer is that it generates a signal $f_h(t)$ which has approximately the same spectral composition as the input signal $g(t)$. Because $f_h(t)$ is a staircase signal (cf. Figure 2.1), whereas $g(t)$ is a continuous signal, it will not be possible to null the difference $e(t)$. The loop filter will emphasize the signal band in which $e(t)$ will be minimized as much as possible.

Because the feedback path generally is delaying, $g(t)$ and $f_h(t)$ will be out of phase. Hence, $e(t)$ will represent not only the truncation error (from the loop quantizer), but also a signal component which is linearly related to $g(t)$. The magnitude of this signal component will at best[22] be proportional to the signal frequency and the feedback delay[23]. If $H_1(f)$ is a first-order integrator, it follows that the magnitude of $o(t)$ is proportional to the feedback delay t_d only. In other words, to minimize the magnitude of $o(t)$,

[21] I.e. such that the loop quantizer's step size is a decreasing function of the accuracy of $d_0(k)$. This way, highly-accurate estimates (which are expected for low-level input signals $g(t)$) will not be subject to significant truncation error, whereas poor estimates (which are expected only for full-scale input signals $g(t)$) will be subject to a larger truncation error.

[22] I.e. if the feed-forward path is implemented, whereby the delay t_d of $f_h(t)$ is minimized. Notice that the delay t_d should be evaluated as the feedback path's delay *plus* half a clock cycle (cf. Figure 2.1).

[23] The magnitude of $e(t) = \sin(2\pi ft) - \sin(2\pi f(t-t_d))$ is $\sqrt{2(1-\cos(2\pi ft_d))}$, which is approximately $2\pi ft_d$ for $2\pi ft_d \ll 1$. Hence, the magnitude of $e(t)$ is proportional to the signal frequency and the feedback delay.

it is necessary to use a loop quantizer with a small step size *and* to minimize the feedback path's delay as much as possible.

To minimize the residue quantizer's resolving range, it is preferable to compensate for the signal component in $o(t)$. This can be performed as shown in Figure 9.10. Assume that the loop quantizer's delay is referred to the feedback DAC which is described by a delay of $t_d > T_s/2$. Assuming that $H_1(f)$ is a first-order integrator with the time constant $\tau_{\rm int}$, the Fourier transformed of $o_1(t)$ will be

$$\begin{aligned} O_1(f) &= \frac{G(f)}{j\omega\tau_{\rm int}} - D_0(f)K_{\rm DAC}e^{-j\omega t_d}\left(\frac{1}{j\omega\tau_{\rm int}} + \gamma\right) \\ &= \frac{G(f)}{j\omega\tau_{\rm int}} - D_0(f)K_{\rm DAC}e^{-j\omega t_d}\left(\frac{1 + j\omega\gamma\tau_{\rm int}}{j\omega\tau_{\rm int}}\right) \\ &= \frac{G(f)}{j\omega\tau_{\rm int}} - D_0(f)K_{\rm DAC}\sqrt{1+(\omega\gamma\tau_{\rm int})^2}\left(\frac{e^{-j\omega t_d}e^{j\omega\gamma\tau_{\rm int}}}{j\omega\tau_{\rm int}}\right) \\ &= \frac{G(f)}{j\omega\tau_{\rm int}} - \frac{D_0(f)K_{\rm DAC}}{j\omega\tau_{\rm int}}\left(\sqrt{1+(\omega\gamma\tau_{\rm int})^2}\ e^{j\omega(\gamma\tau_{\rm int}-t_d)}\right) \end{aligned} \tag{9.13}$$

Equation (9.13) shows that, if the delay-compensation DAC's gain $\gamma K_{\rm DAC}$ is adjusted such the $t_d = \gamma\tau_{\rm int}$, the signal component in $o_1(t)$ will be efficiently canceled in the frequency band for which $\omega\gamma\tau_{\rm int} = 2\pi f\gamma\tau_{\rm int} \ll 1$. If the delay t_d is reasonable (less than T_s) and if the system is oversampled 10 times or more, the compensation for $g(t)$ will be efficient in the entire signal band. The signal from the delay-compensation DAC may be added to the input $H_2(f)$ (as shown for the residue quantizer), in which case $H_2(f)$ will process only truncation noise [43] and it may be simpler to stabilize the modulator.

The signal component in $o(t)$ can be compensated for in other ways, but the shown technique is believed to be the simplest.

To increase the resolution of $d(k)$, the main $\Delta\Sigma$ quantizer can, for example, be designed with a predictive loop quantizer as shown in Figure 9.9 (with the feed-forward path). The residue quantizer's feed-forward path (the dashed line) should, however, in many cases *not* be implemented (based on anti-aliasing concerns).

The Residue Filter. The residue filter is a discrete-time filter, whereas the linear system which it is to match is a continuous-time one. The sense in which they are to match will now be defined. The

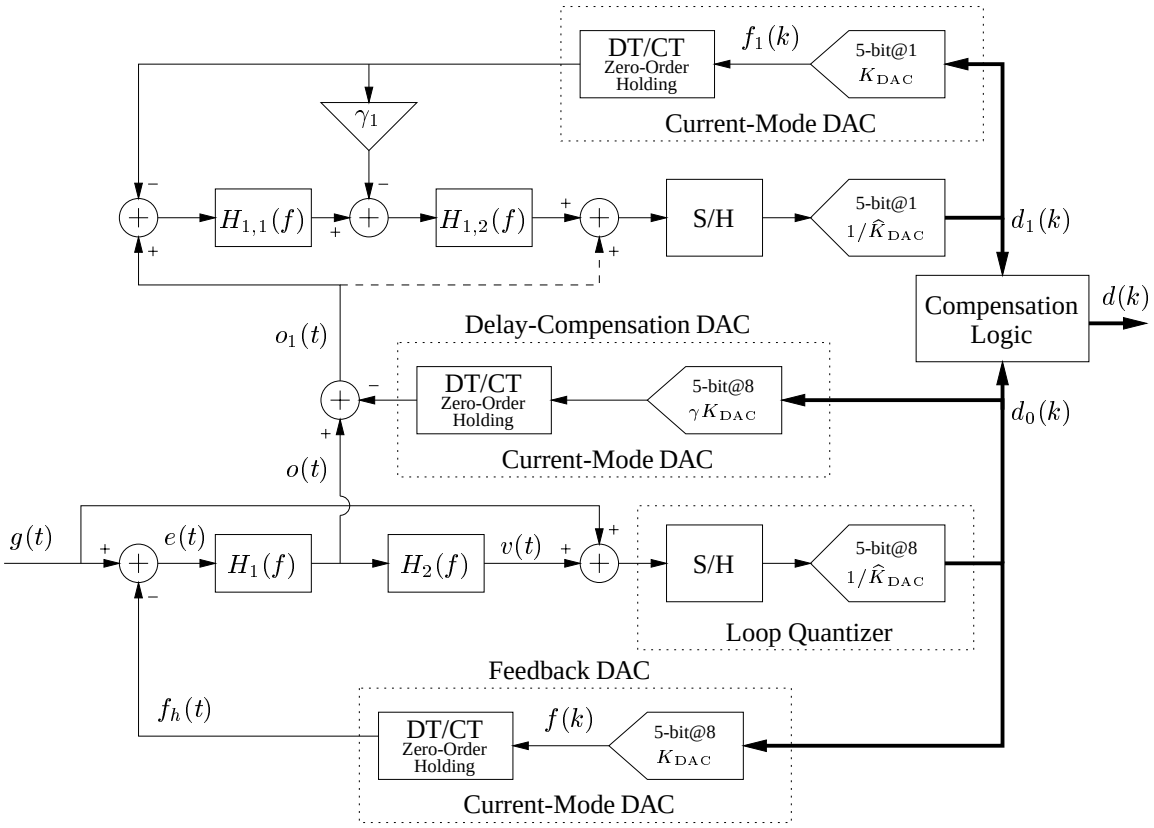


Figure 9.10: Continuous-time residue-compensated $\Delta\Sigma$ quantizer which is compensated for the feedback path's delay.

compensation logic is assumed to generate $d(k)$ according to the rule

$$d(k) = h_{\mathrm{main}}(k) * d_0(k) + h_{\mathrm{res}}(k) * d_1(k) \tag{9.14}$$

where $h_{\mathrm{main}}(k)$ and $h_{\mathrm{res}}(k)$ are the impulse responses of two filters $H_{\mathrm{main}}(z)$ and $H_{\mathrm{res}}(z)$ where $H_{\mathrm{res}}(z)$ is the residue filter). Generally, $H_{\mathrm{main}}(z)$ will be a simple delay, but the technique discussed in the following can also be used in situations when this is not the case (it can also be used for the design of discrete-time residue-compensated $\Delta\Sigma$ quantizers).

It is a fundamental observation that $h_{\mathrm{res}}(k) * d_1(k)$ should compensate for quantization errors as well as any other variation in $h_{\mathrm{main}}(k) * d_0(k)$. Hence, once the entire system except the residue filter is designed, $h_{\mathrm{res}}(k)$ can be determined simply by letting $g(t) = 0$ while forcing $d_0(k)$ to be an digital impluse signal: $1, 0, 0, 0, \ldots$. When the response $d_1(k)$ is know (for example obtained through simulations) it is a simple (linear programming) task to find the coefficients of $h_{\mathrm{res}}(k)$ for which $d(k)$ in Equation 9.14 is zero for all k. A solution can always be found, provided that $H_{\mathrm{main}}(z)$ delays sufficiently (causality problem).

Aliasing Errors. An important advantage of CT $\Delta\Sigma$ quantizers is that a separate anti-aliasing filter may not be necessary. The key point is that the loop filter suppresses the aliasing that occurs in the loop quantizer's sampling process.

The situation is, however, slightly different for residue-compensated $\Delta\Sigma$ quantizers. All errors from the quantizer, including aliasing errors, will be suppressed by $H_{\mathrm{match}}(f)\mathrm{NTF}(f)$ (cf. Section 9.2.2); hence, if the residue filter matches the continuous-time system well (in the sense discussed above), the suppression of aliasing errors does not at all depend on the loop filter. This may sound great, but the pitfall is that a large aliasing error may occur if $o(t)$ is sampled at the residue quantizer's input. This error is suppressed by $1/H_1(f)$ only – not by $\mathrm{NTF}(f)$. Since $H_1(f)$ usually is a low-order filter (in the worst case $o(t) = e(t)$), it is generally *not* acceptable to sample $o(t)$ directly, i.e., the residue quantizer should be designed as a (possibly residue-compensated) CT $\Delta\Sigma$ quantizer.

9.3.3 Conclusions

The proposed type of residue-compensated $\Delta\Sigma$ quantizers combine the advantages of $\Delta\Sigma$ quantizers (high resolution) and pipeline quantizers (simplicity and speed) in an advantageous way. Unlike the well-known MASH quantizers, they do not rely critically on accurate matching of transfer functions.

Continuous-time $\Delta\Sigma$ quantizers are more difficult to design than their discrete-time counterparts, but their better noise performance and the possible omission of a seperate anti-aliasing filter more than compensates for the difficulty. The prospective low power consumption and the good robustness with respect to substrate-coupled noise are other important advantages of CT $\Delta\Sigma$ quantizers; it is the authors belief that they will be used widely in the foreseeable future.

Although several useful techniques for the implementation of high-resolution CT $\Delta\Sigma$ quantizers have been proposed (cf. Section 9.3.1), it is concluded that high-performance low-oversampled CT $\Delta\Sigma$ quantizers are best implemented as residue-compensated systems. For CT $\Delta\Sigma$ quantizers, it may not be imperative to use very low oversampling ratios because CT circuits can be designed to be very fast. A high sampling frequency will mainly affect the digital circuitry, which, soon enough, will turn out to be a limiting factor. Hence, the OSR should be kept moderate if not as low as 10 (at 30 times oversampling, simple first-order mismatch-shaping DACs can provide 100 dB performance).

Continuous-time circuitry is characterized by on-chip-defined time constants, which must be designed with respect to the sampling frequency. To preserve closed-loop stability and to achieve the desired performance, it may be necessary to design a simple master circuit that monitors and controls these time constants; this technique is frequently used in various monolithic CT circuits, such as, for example, transconductance-capacitance g_mC filters.

Chapter 10

Conclusion

Oversampled data conversion is a convenient way to overcome the constraints that are posed by the imperfections and inaccuracies that are inherent in CMOS technology. A delta-sigma ($\Delta\Sigma$) data converter's linearity is constrained mainly by the linearity of an internally employed digital-to-analog converter (DAC). Consequently, because time-invariant single-bit DACs are inherently linear, single-bit $\Delta\Sigma$ converters can have a superb linearity independent of mismatch errors.

The achievable bandwidth of single-bit $\Delta\Sigma$ converters is, however, limited because single-bit signal representation requires a high degree of oversampling. To increase the $\Delta\Sigma$ converter's bandwidth without increasing the sampling frequency or degrading the performance, i.e., to lower the oversampling ratio, multi-bit signal representation is imperative. Mismatch-shaping DACs mark a breakthrough because they facilitate the linear multi-bit D/A conversion required for the implementation of high-performance, low-oversampled, wide-bandwidth $\Delta\Sigma$ converters. The state-of-the-art unit-element mismatch-shaping (UE-MS) DACs suffer two main limitations: their complexity is proportional to their resolution expressed in levels, and they require 25 times oversampling to suppress mismatch errors by 40 dB, which is generally required (cf. Figure 4.17). In other words, circuit complexity constrains the signal representation's resolution to be only a few bits, and the moderately efficient suppression of mismatch-error constrains the data converter's oversampling ratio and bandwidth.

High-Resolution Mismatch-Shaping D/A Converters. In an attempt to improve the tradeoff between performance, cost, and power consumption, this study has focused on the design of high-resolution mismatch-shaping DACs. A major contribution of this work is the development of scaled-element mismatch-shaping (SE-MS) DACs, which are characterized by a low circuit complexity which is only linearly related to the resolution expressed in bits. Because the circuit complexity is low, especially when the so-called filtering principle is used, the signal representation's resolution can be chosen to be so high that the truncation error will be dominated by mismatch errors (10-bit resolution is typical). When using a high-resolution signal representation, the specification of the filters that are normally used to suppress out-of-band errors can be relaxed. Another important advantage of high-resolution signal representation is the generally reduced sensitivity to clock jitter.

To increase the bandwidth, the oversampling ratio must be reduced. However, even in the absence of mismatch errors, high-performance data converters will require a minimum degree of oversampling. Thermal noise, clock-jitter-induced errors, complexity of anti-aliasing filters, etc., are all good reasons why it generally is impractical to reduce the oversampling ratio to less than about 10. Consequently, the target performance, 100 dB for a 0.1% full-scale matching index, was defined for an oversampling ratio of 10. It was shown that to meet this specification the proposed SE-MS DACs cannot be based on the direct use of the filtering principle and an array of UE-MS DACs (because UE-MS DACs do not suppress the mismatch errors sufficiently). To meet the performance requirements, the filtering principle had to be generalized to include the combined use of UE-MS DACs and analog filters. This generalized filtering principle facilitates the implementation of low-complexity DACs with the target bandwidth and performance.

High-Resolution Quantizers. High-performance $\Delta\Sigma$ quantizers can be implemented using a SE-MS DAC as the main feedback stage. However, it is not trivial to increase the signal representation's resolution to the level where the truncation error is dominated by the DAC's mismatch error. The problem arises because the internal loop quantizer may introduce only less than one clock cycle of delay, which for high-speed operation implies that the loop quantizer must be implemented as a flash quantizer with the full resolution. Hence, to avoid high circuit complexity, the signal representation's resolution will

generally be chosen as 6 bits or less. This need not be a problem because the target performance can be obtained if the loop filter is of sufficiently high order.

A pipeline technique was proposed as a way to obtain high-resolution signal representation without compromising the bandwidth or using complex circuitry. The underlying principle is based on the observation that the $\Delta\Sigma$ quantizer's feedback signal will be approximately the same as the input signal if the signal transfer function is unity. Hence, if an analog delay line is used to delay the input signal, an estimate of the input signal can be obtained before it is applied to the $\Delta\Sigma$ quantizer, in which case the loop quantizer need find only the estimate's residue. This way, the feedback signal can have a higher resolution than the loop quantizer. The analog delay line can be designed using only simple circuitry and without significantly degrading the performance.

A high-resolution signal representation can be obtained even if the $\Delta\Sigma$ quantizer operates internally with a coarsely truncated feedback signal. The so-called MASH quantizers estimate the loop quantizer's truncation error, and they process this estimate in an attempt to compensate for the $\Delta\Sigma$ quantizer's truncation error. By compensating for the $\Delta\Sigma$ quantizer's truncation error, a high-resolution output signal can be obtained (cf. Figure 4.5). MASH quantizers are, however, known to be very sensitive to the mismatch of analog and digital filters, which may cause a substantial difference between the $\Delta\Sigma$ quantizer's truncation error and the estimate made thereof. To improve the accuracy by which the truncation error can be estimated and compensated for, the truncation error should be estimated either directly or after only minimal processing. For example, it was shown that a very robust operation can be obtained if the truncation error is estimated on the basis of the output from the loop filter's first integrator stage. The technique is so robust that it can be used successfully in combination with single-bit $\Delta\Sigma$ modulators. It is, however, much preferable to use a multi-bit $\Delta\Sigma$ quantizer because in this case the truncation error can be made small and the cancellation process even more robust. This type of quantizer is characterized by many advantages: low circuit complexity, good robustness, high-resolution signal representation, and a low oversampling ratio.

Concluding Comments. This work facilitates the implementation of oversampled data converters with an unpreceded positive balance between sampling frequency, signal-band and Nyquist-band perfor-

mance, power consumption, circuit complexity, and production cost. Using standard CMOS technology, the achievable performance is mainly limited by device noise, clock jitter, and other unavoidable effects. Hence, these data converters are the state-of-the-art.

Kort beskrivelse på Dansk

(Brief description in Danish)

Denne afhandling omhandler højtydende datakonvertere der bygger på delta-sigma princippet. Dette princip muliggør implementering af højopløsnings datakonvertere, der ikke forudsætter nøjagtig kontrol eller matchning af elektriske parametre. Enkeltbit delta-sigma datakonvertere har igennem de sidste 10–15 år vundet store markedsandele; primært på audio og andre lavfrekvens områder, da deres relative båndbredde er forholdsvis lille (i størrelsorden 1/100 af Nyquist båndbredden).

Flerbit delta-sigma datakonvertere er analyseret med henblik på at opnå en større relativ båndbredde. Sagens kerne ligger i implementeringen af flerbit digital-til-analog (D/A) konvertere, der uanset deres interne opløsning skal være ligeså lineære som det samlede system. Hovedelementet i dette arbejde er udviklingen og beskrivelsen af en ny type D/A konvertere der opfylder dette krav, og som kan implementeres i billige CMOS teknologier hvor den relative kontrol af elektriske parametre ikke kan forventes at være bedre end 0.1%. Disse D/A konvertere er baseret på en digital tilstandsmaskine, der kontrollerer et antal skalerede analoge kilder. Princippet muliggør implementering af datakonvertere med omkring 100 dB linearitet ved en båndbredde på en 1/10 af Nyquist båndbredden. Da denne båndbredde ikke kan forøges nævneværdigt uden at stille upraktisk store krav til andre systemaspekter (såsom støj, clock jitter, frekvensrespons, med videre), konkluderes det at den opnålige ydelse er omtrent optimal for højopløsnings datakonvertere. Det er bemærkelsesværdigt at disse D/A konvertere ydermere er simplere at realisere end enkeltbit delta-sigma D/A konvertere, og de er således velegnede både til lav- og mellemfrekvens formål samt hvor effektforbruget skal være lavt.

De omtalte D/A konvertere er velegnede til brug som tilbagekobling internt i delta-sigma analog-til-digital (A/D) konvertere. Deres potentiale kan imidlertid ikke direkte udnyttes fuldt ud, da det generelt er vanskeligt at kvantisere det analoge signal til en høj opløsning uden at introducere en vis forsinkelse (hvilket kan forårsage ustabilitet). Denne begrænsning kan imidlertid omgås på flere måder. Blandt andet er en ny og forbedret arkitektur for de velkendte kompenserende (MASH) delta-sigma A/D konvertere foreslået. Den nye arkitektur er i praksis ikke afhængig af matchning af overføringsfunktioner.

Bibliography

[1] Steven R. Norsworthy, Richard Schreier, and Gabor C. Temes, Eds., *Delta-Sigma Data Converters: Theory, Design, and Simulation*, IEEE Press, 1996.

[2] Richard Schreier, "Mismatch-Shaping Digital-to-Analog Conversion", An Audio Engineering Society Preprint of paper [4529 (E-1)] presented at the 103rd convention, September 1997, New York.

[3] L. E. Larsen, T. Cataltepe, and G. C. Temes, "Multi-bit Oversampled Σ-Δ A/D Converter with Digital Error Correction", *Electronics Letters*, vol. 24, pp. 1051–1052, August 1988.

[4] L. E. Larsen, A. R. Kramer, T. Cataltepe, G. C. Temes, and R. H. Walden, "Digitally-Corrected Multi-bit ΣΔ Data Converter with", *Proceedings for the 1989 IEEE International Symposium on Circuits and Systems*, pp. 647–650, May 1989.

[5] Michael J. Story, "Digital-to-Analogue Converter Adapted to Select Input Sources Based on A Preselected Algorithm once per Cycle of a Sampling Signal", U.S. Patent 5,138,317, August 1992, Priority date: Feb. 17, 1988 (GB).

[6] L. Richard Carley, "A Noise-Shaping Coder Topology for 15+ Bit Converters", *IEEE Journal of Solid-State Circuits*, vol. 24, no. 2, pp. 267–273, April 1989.

[7] Bosco H. Leung and Sehat Sutarja, "Multibit ΣΔ A/D Converter Incorporating A Novel Class of Dynamic Element Matching Techniques", *IEEE Transactions on Circuits and Systems—II*, vol. 39, no. 1, pp. 35–51, January 1992.

[8] H. Spence Jackson, "Circuit and Method for Cancelling Nonlinearity Error Associated with Component Value Mismatches in a Data Converter", U.S. Patent 5,221,926, June 1993, Filed Jul. 1, 1992.

[9] Robert W. Adams, "Data-Directed Scrambler for Multi-bit Noise Shaping D/A Converters", U.S. Patent 5,404,142, April 1995, Filed Aug. 5, 1993.

[10] Rex.T. Baird and Terri S. Fiez, "Improved $\Delta\Sigma$ DAC Linearity Using Data Weighted Averaging", in *Proceedings for the 1995 IEEE International Symposium on Circuits and Systems*. IEEE Circuits and Systems Society, 1995, vol. 1, pp. 13–16.

[11] Rex T. Baird and Terri S. Fiez, "Linearity Enhancement of Multibit $\Sigma\Delta$ A/D and D/A Converters Using Data Weighted Averaging", *IEEE Transactions on Circuits and Systems—II: Analog and Digital Signal Processing*, vol. 42, no. 12, pp. 753–762, December 1995.

[12] Richard Schreier and B. Zhang, "Noise-shaped multibit D/A converter employing unit elements", *Electronics Letters*, vol. 31, no. 20, pp. 1712–1713, September 1995.

[13] Haiquing Lin, José Barreiro da Silva, Bo Zhang, and Richard Schreier, "Multi-Bit DAC with Noise-Shaped Element Mismatch", in *Proceedings for the 1996 IEEE International Symposium on Circuits and Systems*, Atlanta, May 1996, IEEE Circuits and Systems Society, vol. 1, pp. 235–238.

[14] Ian Galton, "Noise-Shaping D/A Converters for $\Sigma\Delta$ Modulation", in *Proceedings for the 1996 IEEE International Symposium on Circuits and Systems*, Atlanta, May 1996, IEEE Circuits and Systems Society, vol. 1, pp. 441–444, See also U.S. Patent 5,684,482, issued November 4, 1997.

[15] Tom Kwan, Robert Adams, and Robert Libert, "A Stereo Multibit $\Sigma\Delta$ DAC with Asynchronous Master-Clock Interface", *IEEE Journal of Solid-State Circuits*, vol. 31, no. 12, pp. 1881–1887, December 1996.

[16] A. Yasuda and H. Tanimoto, "Noise-shaping dynamic element matching method using tree structure", *Electronics Letters*, vol. 33, no. 2, pp. 130–131, January 1997.

[17] A. Keady and C. Lyden, "Tree structure for mismatch noise-shaping multibit DAC", *Electronics Letters*, vol. 33, no. 17, pp. 1431–1432, August 1997.

[18] Henrik T. Jensen and Ian Galton, "A Reduced-Complexity Mismatch-Shaping DAC for Delta-Sigma Converters", in *Proceedings for the 1998 IEEE International Symposium on Circuits and Systems*, Monterey, June 1998, IEEE Circuits and Systems Society, vol. CDROM, session WAB7–2.

[19] Jont B. Allen and Stephen T. Neely, "Micromechanical models of the cochlea", *Physics Today*, July 1992.

[20] Communication Engineering Laboratory, "Cochlear Mechanics", `http://www.boystown.org/cel/cochmech.htm`.

[21] Athanasios Papoulis, *The Fourier Integral and its Applications*, Electronic Science. McGraw-Hill, 1962.

[22] Fredric J. Harris, "On the Use of Windows for Harmonic Analysis with the Discrete Fourier Transform", *Proceedings of the IEEE*, vol. 66, no. 1, pp. 51–84, January 1978.

[23] Bernard Picinbono, *Random Signals and Systems*, Signal Processing. Prentice Hall, 1993.

[24] Robert Jewett, Ken Poulton, Kuo-Chiang Hsieh, and Joey Doernberg, "A 12b 128MSample/s ADC with 0.05LSB DNL", in *Digest of Technical Papers for the 1997 International Solid-State Circuits Conference*. IEEE Solid-State Circuits Society, February 1997, vol. 40, pp. 138–139.

[25] Nav S. Sooch, Jeffrey W. Scott, T. Tanaka, T. Sugimoto, and C. Kubomura, "18-bit Stereo D/A Converter with Integrated Digital and Analog Filters", An Audio Engineering Society Preprint of paper [3113 (Y-1)] presented at the 91st convention, October 1991, New York.

[26] Kh. Hadidi, K. Eguchi, T. Matsumoto, and H. Kobayashi, "A Highly Linear Second-Order Stage for 500-MHz Third-Order and Fifth-Order Filters", in *Proceedings for the 5th IEEE International Conference on Electronics, Circuits and Systems*. IEEE Circuits and Systems Society, September 1998, vol. 3, pp. 361–364.

[27] D.G. Haigh and B. Singh, "A Switching Scheme for Switched Capacitor Filters which Reduces the Effect of Paracitic Capacitances Associated with Switch Control Terminals", in *Proceedings*

for the 1983 IEEE International Symposium on Circuits and Systems. IEEE Circuits and Systems Society, April 1983, pp. 586–589.

[28] Gabor C. Temes, "Simple Formula for Estimation of Minimum Clock-Feedthrough Error Voltage", *Electronics Letters*, pp. 1069–1070, September 1986.

[29] R. Gregorian and G. C. Temes, *Analog MOS Integrated Circuits for Signal Processing*, Wiley, New York, 1986.

[30] Jose E. Franca and Yannis Tsividis, Eds., *Design of Analog-Digital VLSI Circuits for Telecommunications and Signal Processing*, chapter 2.4, pp. 101–105, Prentice Hall, second edition, 1994, Author: Eric Vittoz.

[31] Todd L. Brooks, David H. Robertson, Daniel F. Kelly, Anthony Del Muro, and Stephen W. Harston, "A Cascaded $\Sigma\Delta$ Pipeline A/D Converter with 1.25 MHz Signal Bandwidth and 89 dB SNR", *IEEE Journal of Solid-State Circuits*, vol. 32, no. 12, pp. 1896–1906, December 1997.

[32] Yunteng Huang, Gabor C. Temes, and Paul F. Ferguson Jr., "Offset- and Gain-Compensated Track-and-Hold Stages", in *Proceedings for the 5th IEEE International Conference on Electronics, Circuits and Systems*. IEEE Circuits and Systems Society, September 1998, vol. 2, pp. 13–16.

[33] Sanjit K. Mitra, *Digital Signal Processing: A Computer-Based Approach*, chapter 5, McGraw-Hill, 1998.

[34] C-H. Lin and Klaus Bult, "A 10b 250MSample/s CMOS DAC in 1mm^2", in *Digest of Technical Papers for the 1998 International Solid-State Circuits Conference*. IEEE Solid-State Circuits Society, February 1998, vol. 41, pp. 214–215.

[35] Robert Adams, Khiem Nguyen, and Karl Sweetland, "A 113dB SNR Oversampling DAC with Segmented Noise-Shaped Scrambling", in *Digest of Technical Papers for the 1998 International Solid-State Circuits Conference*, San Fransisco, February 1998, IEEE Solid-State Circuits Society, vol. 41, pp. 62–62.

[36] R. W. Adams, "Design and Implementation of an Audio 18-bit Analog-to-Digital Converter using Oversampling Techniques", *J. Audio Engineering Society*, vol. 34, no. 3, pp. 153–166, March 1986.

[37] Wai Laing Lee, "A Novel Higher-Order Interpolative Modulator Topology for High Resolution Oversampling A/D Converters", Master's thesis, Massachusetts Institute of Technology, June 1987, Intersymbol Interference: see page 64.

[38] Toshihiko Hamasaki, Yoshiaki Shinohara, Hitoshi Terasawa, Kuo-Ichirou Ochiai, Masaya Hiraoka, and Hideki Kanayama, "A 3-V, 22-mW Multibit Current-Mode $\Delta\Sigma$ DAC with 100dB Dynamic Range", *IEEE Journal of Solid-State Circuits*, vol. 31, no. 12, pp. 1888–1894, December 1996.

[39] Louis A. Williams III, "An Audio DAC with 90dB Linearity using MOS Metal-Metal Charge Transfer", in *Digest of Technical Papers for the 1998 International Solid-State Circuits Conference*, San Fransisco, February 1998, IEEE Solid-State Circuits Society, vol. 41, pp. 58–59.

[40] Germano Nicollini, Sergio Pernici, Pierangelo Confalonieri, Carlo Crippa, Angelo Nagari, Sergio Maniani, Aldo Calloni, Massimo Moioli, and Carlo Dallavalle, "A High-Performance Analog Front-End 14-Bit Codec for 2.7-V Digital Cellular Phones", *IEEE Journal of Solid-State Circuits*, vol. 33, no. 8, pp. 1158–1167, August 1998.

[41] Jacques Robert, Gabor C. Temes, Vlado Valencic, Roger Dessoulavy, and Philippe Deval, "A 16-bit Low-Voltage CMOS A/D Converter", *IEEE Journal of Solid-State Circuits*, vol. SC-22, no. 2, pp. 157–163, April 1987.

[42] Larry A. Singer and Todd L. Brooks, "14-bit 10-MHz Calibration-Free CMOS Pipelined A/D Converter", in *Digest of Technical Papers Proceedings of the 1996 Symposium on VLSI Circuits*, June 1996, pp. 94–95.

[43] Jesper Steensgaard, "Nonlinearities in SC Delta-Sigma A/D Converters", in *Proceedings for the 5th IEEE International Conference on Electronics, Circuits and Systems*, 1998, vol. 1, pp. 355–358.

[44] Lars Risbo, Σ-Δ *Modulators—Stability, Analysis, and Optimization*, PhD thesis, The Technical University of Denmark, DK-2800, Lyngby, Denmark, June 1994.

[45] Wai Laing Lee, "A Novel Higher Order Interpolative Modulator Topology for High Resolution Oversampling A/D Converters", Master's thesis, Massachusetts Institute of Technology, June 1987.

[46] Richard Schreier, "The Delta-Sigma Toolbox", MATLAB code available via anonymous ftp at `ftp://next242.ece.orst.edu/pub/delsig.tar.Z`.

[47] Gopal Raghavan, Joseph F. Jensen, Robert H. Walden, and William P. Posey, "A Bandpass $\Sigma\Delta$ Modulator with 92dB SNR and Center Frequency Continuously Programmable from 0 to 70MHz.", in *Digest of Technical Papers for the 1997 International Solid-State Circuits Conference*. IEEE Solid-State Circuits Society, 1997, vol. 40, pp. 214–215.

[48] Todd L. Brooks, David H. Robertson, Daniel F. Kelly, Anthony Del Muro, and Steve W. Harston, "A 16b $\Sigma\Delta$ Pipeline ADC with 2.5MHz Output Data-Rate", in *Digest of Technical Papers for the 1997 International Solid-State Circuits Conference*. IEEE Solid-State Circuits Society, February 1997, pp. 208–209.

[49] Augusto Manuel Marques, Vicenzo Peluso, Michel S. J. Steyaert, and Willy Sansen, "A 15-b Resolution 2-MHz Nyquist Rate $\Sigma\Delta$ ADC in a 1-μm CMOS Technology", *IEEE Journal of Solid-State Circuits*, vol. 33, no. 7, pp. 1065–1075, July 1998.

[50] Andreas Wiesbauer, Tao Sun, and Gabor Temes, "Adaptive Compensation of Analog Circuit Imperfections for Cascaded Delta-Sigma ADCs", in *Proceedings for the 1998 IEEE International Symposium on Circuits and Systems*, Monterey, June 1998, IEEE Circuits and Systems Society, vol. CDROM, session TPA14–9.

[51] Feng Chen and Bosco Leung, "Some Observations on Tone Behavior in Data Weighted Averaging", in *Proceedings for the 1998 IEEE International Symposium on Circuits and Systems*, Monterey, June 1998, IEEE Circuits and Systems Society, vol. CDROM, session WAB7–1.

[52] Ian Galton and Paolo Carbone, "A Rigorous Error Analysis of D/A Conversion with Dynamic Element Matching", *IEEE Transactions on Circuits and Systems—II: Analog and Digital Signal Processing*, vol. 42, no. 12, pp. 763–772, December 1995.

[53] Akira Yasuda, "Selector", Japanese patent disclosure (Kokai) No. H8-154058, filed September 30, 1994.

[54] Christian Enz and Gabor C. Temes, "Circuit techniques for reducing the effects of op-amp imperfections: autozeroing, correlated double sampling, and chopper stabilization", *Proceedings of the IEEE*, vol. 84, no. 11, pp. 1584–1614, 1996.

[55] Donald Kerth, "Practical Design for Analog Discrete-Time Processing (ADTP)", Notes from an ISSCC Tutorial Lecture, February 1997.

[56] A. Marques, V. Peluso, M. Steyaert, and W. Sansen, "Analysis of the Trade-off between Bandwidth, Resolution, and P ower in $\Delta\Sigma$ Analog-to-Digital Converters", in *Proceedings for the 5th IEEE International Conference on Electronics, Circuits and Systems*, Lisboa, Portugal, September 1998, IEEE Circuits and Systems Society, vol. 2, pp. 153–156.

[57] Robert Adams, Khiem Q. Nguyen, and Karl Sweetland, "A 113-dB SNR Oversampling DAC with Segmented Noise-Shaped Scrambling", *IEEE Journal of Solid-State Circuits*, vol. 33, no. 12, pp. 1871–1878, December 1998.

[58] Lauri Sumanen, Mikko Waltari, and Kari Halonen, "A 10-bit High-Speed Low-Power CMOS D/A Converter in 0.2 mm^2", in *Proceedings for the 5th IEEE International Conference on Electronics, Circuits and Systems*, Lisbon, Portugal, September 1998, IEEE Circuits and Systems Society, vol. 1, pp. 15–18.

[59] Luis Hernandez, "A Model of Mismatch-Shaping D/A Conversion Independent of the DAC Architecture", *IEEE Transactions on Circuits and Systems—I: Fundamental Theory and Applications*, vol. 45, no. 10, pp. 1068–1076, October 1998.

[60] Gabor C. Temes and Shao-Feng Shu, "Dual Quantization Oversampling Digital-to-Analog Converter", U.S. patent # 5,369,403, November 1994, Filed September 1, 1992.

[61] Soenke Mehrgardt and Ulrich Theus, "Monolithic Integrated Digital-to-Analog Converter", U.S. Patent # 4,791,406, December 1988, Filed July 16, 1987.

[62] James W. Everitt and Hiroshi Takatori, "Lor-Resolution, High-Linearity Digital-to-Analog Converter Without Trim", U.S. Patent, # 5,534,863, July 9 1996, Filed January 6, 1994.

[63] Ka Y. Leung, Eric J. Swanson, Kafei Leung, and Sarah S. Zhu, "A 5-V, 118-dB $\Delta\Sigma$ Analog-to-Digital Converter for Wideband Digital Audio", in *Digest of Technical Papers for the 1997 International Solid-State Circuits Conference*, San Fransisco, February 1997, IEEE Solid-State Circuits Society, vol. 40, pp. 218–219.

[64] Jesper Steensgaard, "Clocking Scheme for Switched-Capacitor Circuits", in *Proceedings for the 1998 IEEE International Symposium on Circuits and Systems*, Monterey, California, May 1998, IEEE Circuits and Systems Society, vol. 1 of *CDROM*.