KTH Electrical Engineering

Characterization and Correction of Analog-to-Digital Converters

HENRIK F. LUNDIN

Doctoral Thesis
Stockholm, Sweden 2005

TRITA–S3–SB-0575
ISSN 1103-8039
ISRN KTH/SB/R--05/75--SE
ISBN 91-7178-210-9

KTH School of Electrical Engineering
SE-100 44 Stockholm
SWEDEN

Akademisk avhandling som med tillstånd av Kungl Tekniska högskolan framlägges till offentlig granskning för avläggande av teknologie doktorsexamen i signalbehandling fredagen den 16 december 2005 klockan 13.00 i D3, Kungl Tekniska högskolan, Lindstedtsvägen 5, Stockholm.

Tryck: Universitetsservice US AB

Abstract

Analog-to-digital conversion and quantization constitute the topic of this thesis. Post-correction of analog-to-digital converters (ADCs) is considered in particular. ADCs usually exhibit non-ideal behavior in practice. These non-idealities spawn distortions in the converters output. Whenever the errors are systematic, it is possible to mitigate them by mapping the output into a corrected value. The work herein is focused on problems associated with post-correction using look-up tables. All results presented are supported by experiments or simulations.

The first problem considered is characterization of the ADC. This is in fact an estimation problem, where the transfer function of the converter should be determined. This thesis deals with estimation of quantization region midpoints, aided by a reference signal. A novel estimator based on order statistics is proposed, and is shown to have superior performance compared with the sample mean traditionally used.

The second major area deals with predicting the performance of an ADC after post-correction. A converter with static differential nonlinearities and random input noise is considered. A post-correction is applied, but with limited (fixed-point) resolution in the corrected values. An expression for the signal-to-noise and distortion ratio after post-correction is provided. It is shown that the performance is dependent on the variance of the differential nonlinearity, the variance of the random noise, the resolution of the converter and the precision of the correction values.

Finally, the problem of addressing, or indexing, the correction look-up table is dealt with. The indexing method determines both the memory requirements of the table and the ability to describe and correct dynamically dependent error effects. The work here is devoted to state-space–type indexing schemes, which determine the index from a number of consecutive samples. There is a tradeoff between table size and dynamics: more samples used for indexing gives a higher dependence on dynamic, but also a larger table. An indexing scheme that uses only a subset of the bits in each sample is proposed. It is shown how the selection of bits can be optimized, and the exemplary results show that a substantial reduction in memory size is possible with only marginal reduction of performance.

Sammanfattning

Denna avhandling behandlar analog–digitalomvandling. I synnerhet behandlas postkorrektion av analog–digitalomvandlare (A/D-omvandlare). A/D-omvandlare är i praktiken behäftade med vissa fel som i sin tur ger upphov till distorsion i omvandlarens utsignal. Om felen har ett systematiskt samband med utsignalen kan de avhjälpas genom att korrigera utsignalen i efterhand. Detta verk behandlar den form av postkorrektion som implementeras med hjälp av en tabell ur vilken korrektionsvärden hämtas.

Innan en A/D-omvandlare kan korrigeras måste felen i den mätas upp. Detta görs genom att estimera omvandlarens överföringsfunktion. I detta arbete behandlas speciellt problemet att skatta kvantiseringsintervallens mittpunkter. Det antas härvid att en referenssignal finns tillgänglig som grund för skattningen. En skattare som baseras på sorterade data visas vara bättre än den vanligtvis använda skattaren baserad på sampelmedelvärde.

Nästa huvudbidrag visar hur resultatet efter korrigering av en A/D-omvandlare kan predikteras. Omvandlaren antas här ha en viss differentiell olinjäritet och insignalen antas påverkad av ett slumpmässigt brus. Ett postkorrektionssystem, implementerat med begränsad precision, korrigerar utsignalen från A/D-omvandlaren. Ett utryck härleds som beskriver signal–brusförhållandet efter postkorrektion. Förhållandet visar sig bero på den differentiella olinjäritetens varians, det slumpmässiga brusets varians, omvandlarens upplösning samt precisionen med vilken korrektionstermerna beskrivs.

Till sist behandlas indexering av korrektionstabeller. Valet av metod för att indexera en korrektionstabell påverkar såväl tabellens storlek som förmågan att beskriva och korrigera dynamiska fel. I avhandlingen behandlas i synnerhet tillståndsmodellbaserade metoder, det vill säga metoder där tabellindex bildas som en funktion utav flera på varandra följande sampel. Allmänt gäller att ju fler sampel som används för att bilda ett tabellindex, desto större blir tabellen, samtidigt som förmågan att beskriva dynamiska fel ökar. En indexeringsmetod som endast använder en delmängd av bitarna i varje sampel föreslås här. Vidare så påvisas hur valet av indexeringsbitar kan göras optimalt, och experimentella utvärderingar åskådliggör att tabellstorleken kan reduceras avsevärt utan att fördenskull minska prestanda mer än marginellt.

De teorier och resultat som framförs här har utvärderats med experimentella A/D-omvandlardata eller genom datorsimuleringar.

Acknowledgements

As a PhD student, you sometimes get the response 'That must be difficult!' after having told what you do for a living. Well, I'll tell you what the really difficult part of it all is: writing the acknowledgements. In this thesis that you now hold in you hands, I've spent some 190 pages elaborating at great length on a subject so narrow that almost nobody has ever heard of it. Meanwhile, I have less than two pages to express feelings such as joy and gratitude, and to declare the importance of friends and family and of great colleagues. And this is the one part that everyone reads—if you do not flip this pages to see what's coming next, you will be in good company, but will perhaps miss out on some interesting results...

First of all I'd like to express my deepest gratitude to my supervisors Peter Händel and Mikael Skoglund. From the first day of my Master's thesis project, which is where my researching days started, I've always felt your support and your confidence in me. Without your skillful supervision and deep knowledge in the various aspects of signal processing, this thesis would not have been what it is today. I thank you for all this.

The Signal Processing group, with its past and present members, has been a never-ending source of knowledge, inspiration and, most important, great fun. The number of lunches and coffee breaks that have extended way beyond reasonable time, simply because of interesting and amusing discussions, are uncountable. Although all of you contribute to the unique and warm atmosphere, I must extend my special thanks to a few of you:

The first one out is Tomas Andersson, for putting up with me on trips around the globe, for joint research and also for being a great office neighbor. I know you are doing your best to "keep up that good Swedish tradition."

Although Niclas Björsell spends his days in Gävle, to me he is a significant part of the Signal Processing group. Thanks for good company on several conference trips, and for your valuable feedback on my work.

I thank the computer support group, Nina Unkuri and Andreas Stenhall, for splendid computer support. Together with Tomas Andersson we have also shown that shouting is the best form of intra-office communication between neighboring rooms.

Karin Demin deserves recognition for her impeccable administration of this research group – whatever you need to know, Karin finds it in her binders within

seconds.

Reaching outside the department, I'd like to thank Professor Pasquale Daponte, Associate Professor Sergio Rapuano, Dr. Luca De Vito and the rest of the people at L.E.S.I.M., the Laboratory of Signal and Measurement Information Processing at the University of Sannio in Benevento, Italy. They make me feel very welcome when visiting Benevento, and the collaboration has lead to joint publications.

I thank Professor Paolo Carbone, Associate Professor Ana Rusu, Dr. Jan-Erik Eklund and Professor Håkan Johansson for acting opponent and grading committee during my public defense of this thesis.

I would like to thank my friends for keeping my mind off my work from time to time. Of course, I thank my parents for their support and for patiently waiting the twenty-two years it took me to finish school.

Finally, I thank my wife Therese for her infinite support and love. I would not have made it without you.

Henrik Fahlberg Lundin
Stockholm, November 2005

Contents

Chapter 1

Introduction

This thesis deals with the subject of digital post-correction of analog-to-digital converters. The fundamental observation, upon which the work is motivated, is that practical analog-to-digital converters are prone to exhibit errors. To be more precise, we say that a practical converter is likely to exhibit deviations from the ideal operation of sample, hold and quantization. The term 'post-correction' indicates that the correction methods considered are applied *after* the converter, thus operating on the digital signal provided from the output. One of the fundamental constraints within this work is therefore that the internal signals and states of the analog-to-digital converter under consideration are unavailable to us. The goal of the correction is, naturally, to make the corrected output from the converter more true to the ideal output, in some sense; as we will see later on, there are many ways to measure the performance of a converter. Further background and motivation to using post-correction is provided in Section 1.1.

This chapter introduces the *analog-to-digital converter*, henceforth also referred to as A/D converter or simply ADC. The properties and theory of operation of the ideal ADC are introduced in Section 1.2, where also some common non-idealities encountered in practical ADCs are discussed. In Section 1.3, three frequently used ADC implementations are briefly described; these are (pipelined) flash, successive approximation and sigma-delta type ADCs. In Section 1.4, some performance measures that will be used throughout this thesis are defined.

Section 1.5 gives a short introduction to the problems associated with post-correction. The specific problems where this thesis provides contributions are pointed out in particular.

The remaining parts of the thesis are outlined in Section 1.6. Finally, Sections 1.7 and 1.8 list some of the notations and acronyms used in this work.

1.1 Background

Today, ADCs are widely used in many different applications. Analog-to-digital conversion in radio receivers, for instance, impose special demands on the converter, and a trend in receiver design has been to move the digitization closer to the receiving antenna. Flexibility in configuration and lower cost are two reasons for striving in this direction. Meanwhile, the carrier frequency as well as the bandwidth are increasing, calling for higher sampling rates and increasing analog input bandwidth. The linearity of the ADC is also a key characteristic, and the specifications of the system in which the ADC is a part, e.g., required signal-to-noise ratio, impose requirements on linearity of the converter (see for example [SNM99]). Currently the performance of all-digital receivers, also known as software radio receivers, is limited by the distortion produced by the ADC, and typical dynamic range requirements are hard to meet – if at all possible – with present commercially available converters [Hum02]. This is one application where digital post-correction can be beneficial, simply because there is no product available that meets the demands of a certain application.

Another rationale for applying post-correction is that in some applications it can be beneficial, or even unavoidable, to use a converter with inferior characteristics (e.g., a less expensive ADC). Digital post-correction can then be used to compensate for the shortcomings of the selected converter. A similar case arises when integrating many functions on one chip. An example can be where a sensor (e.g., pressure or acceleration) is to be integrated on the same chip as an ADC, converting the sensor output into digital form. It is not far fetched that the ADC will face design constraints that restrict the performance—supply voltage, power consumption and even choice of manufacturing process can be such factors. The converter is then designed at a non-optimum design point, and post-correction can be applied to mitigate the errors associated with the design. Another example, which is far more common, is where an ADC is integrated on the same chip as a digital signal processor (DSP). In this case there is often a tradeoff between what is good for the performance of the DSP and for the ADC. The DSP would typically be manufactured using a chip process with smaller geometry an lower supply voltage than what is beneficial for the ADC, mainly in order to keep down power consumption and facilitate higher computational power. The ADC would then, again, suffer from manufacturing parameters that are less suited for high-precision analog design—post-correction could also here be a measure to reduce the distortions of the converter.

Looking at an ADC in a larger system perspective, we can regard one or several sensors, the analog signal conditioning and the digitization as one system, where the physical quantities to be measured are input signals and the digital signal from the converter is the output. The distortions introduced in any of the components in the measurement chain will of course affect the output signal, and can be seen as a non-ideality in the path from physical input to digital output. Again, a digital post-correction after the ADC can be a feasible solution to reduce the errors in the

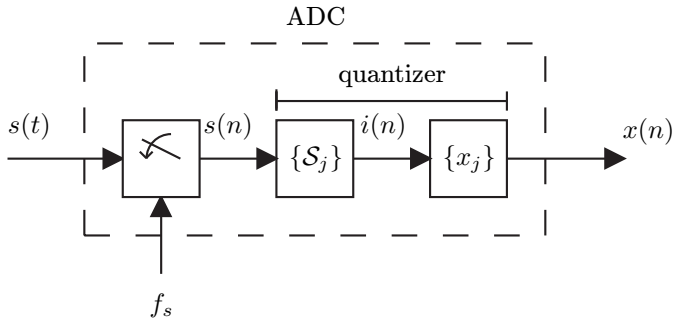


Figure 1.1: A model for the ideal AD converter with the quantization represented as a two-step operation.

measurement system. Thus, even if the ADC as such is without considerable flaws, a digital correction scheme can be beneficial to apply anyway.

1.2 The Analog-to-Digital Converter

In this section, the theory of operation of an ADC is described, starting with the ideal ADC.

Ideal Converter

An A/D-converter is an intricate electronic device. The theory of operation for an *ideal* b-bit converter is nevertheless straightforward to explain. The converter has a continuous-time input signal, say $s(t)$. This signal can assume any real value, possibly confined to be within some limits, i.e., $s(t)$ is continuous both in amplitude and in time. The analog-to-digital conversion is then a quantization in time and amplitude, such that for every time instant $n\,T_s$, where T_s is the sampling period and n is the running sample index (integer), an output x_i from a finite set $\{x_j\}_{j=0}^{M-1}$ is produced. Here, $M = 2^b$ is the number of quantization levels or possible output codes. The A/D-converted version of the input $s(t)$ can be represented with the discrete-time signal $x(n) \in \{x_j\}_{j=0}^{M-1}$. It is common to divide the ideal ADC device into two parts: an ideal sample-and-hold device and an ideal quantizer, see Figure 1.1. This structure is not only mathematically convenient, it is also consistent with some practical ADC architectures (cf. Section 1.3).

The ideal sample-and-hold (S/H) device is merely a device that samples the input signal at given time instants. Usually these time instants occur periodically at rate f_s, corresponding to a sampling period $T_s = 1/f_s$, thus sampling the input at time $n\,T_s$ for all integers n. The output from the S/H is $s(n\,T_s)$ for $n\,T_s \leq t < (n+1)\,T_s$, until the next sampling instant, when it changes to the new value

$s((n+1)\,T_s)$. The output can be represented by a discrete-time signal[1] $s(n) \triangleq s(n\,T_s)$. The operation of an ideal S/H is illustrated in Figure 1.2. The well-known mathematical theory of sampling (e.g., [PM96, OWN97]), including the sampling theorem and the concept of aliasing, applies to the ideal S/H device.

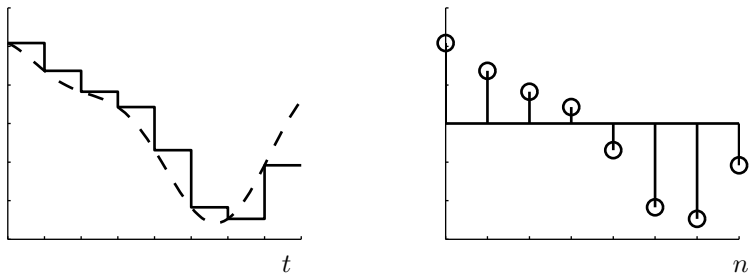


Figure 1.2: The operation of an ideal S/H device (left). The value of the input signal $s(t)$ (dashed) at the sampling instant is held until the next sampling instant, producing a piecewise constant output signal (solid). The output signal can be represented with a discrete-time signal $s(n)$ (right).

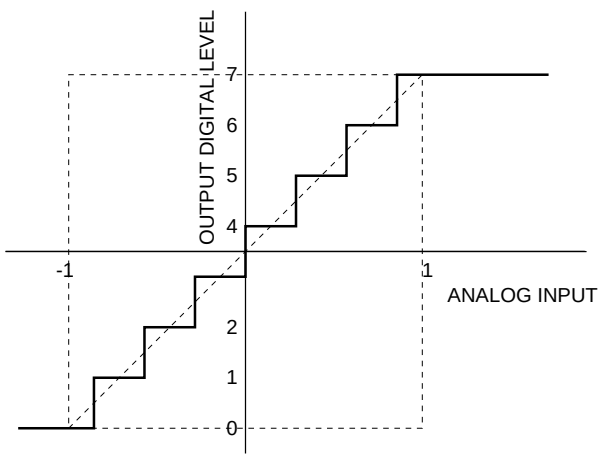


Figure 1.3: Ideal 3-bit quantizer.

[1] It can be argued that this notation is ambiguous; does '$s(n)$' mean $s(t)$ for $t = n\,T_s$ or $t = n$? Alternative notations exist, including square brackets or subscript indices for discrete-time signals. However, throughout this thesis we will use the notation introduced here, and it should be clear from the context whether continuous-time or discrete-time signals are considered.

Figure 1.3 illustrates the operation of a quantizer, in this case a 3-bit quantizer. In general, b-bit quantization is often represented as a two-step mapping, as shown in Figure 1.1. In the first step the sampled signal value $s(n)$ is mapped through a noninvertible mapping into an index $i \in \{0, 1, \ldots, M-1\}$. The index production is defined by a partition of the real line into disjoint sets $\{\mathcal{S}_j\}_{j=0}^{M-1}$, or *quantization regions*. A quantization region is defined by two neighboring transition levels as

$$\mathcal{S}_i = \{s : T_i \leq s < T_{i+1}\}, \tag{1.1}$$

where $T_0 = -\infty$, $T_M = +\infty$ and $T_{i-1} < T_i$ for all i. If the input signal $s(n\,T_s)$ falls into the region $\mathcal{S}_i$, then the index i is produced. The vast majority of practical ADCs are designed to have quantization regions of equal size, save the semi-infinite end regions, as in Figure 1.3. These quantizers are referred to as *uniform* or *linear*. The difference between two neighboring code transition levels is denoted the *code bin width* Δ. That is, T_k is the code transition level between codes $k-1$ and k, and $W_k = T_k - T_{k-1}$ is the width of the k-th bin. In the second step the index i is mapped through a (possibly invertible) mapping to an output value $x_i \in \{x_j\}_{j=0}^{M-1}$, where $\{x_j\}_{j=0}^{M-1}$ is the set of all possible output values. Sometimes the output value x_i is denoted *reconstruction level* and $\{x_j\}$ the *codebook*.

It is convenient to denote the b-bit quantization operation with the operator $\mathcal{Q}_b[\cdot]$. The operator can be defined as $s \in \mathcal{S}_i \Rightarrow \mathcal{Q}_b[s] = x_i$, with the notation introduced above. Using this operator, the ADC output signal can be written in a compact manner as $x(n) = \mathcal{Q}_b[s(n\,T_s)]$.

Quantization Error

In an ideal A/D converter, the only "error"[2] in the output signal is the quantization error, here denoted $e_q(n)$. The quantization error is defined as

$$e_q(n) = x(n) - s(n) = \mathcal{Q}_b[s(n)] - s(n). \tag{1.2}$$

When analyzing the quantization error it is common to regard the error as random [PM96, WKL96, Wal99, Gra90]. In general the quantization error is correlated with the quantizer input $s(n)$, especially when $s(n)$ is constant or slowly varying. However, under certain conditions $e_q(n)$ will assume more tractable properties.

In [PM96] it is claimed (but not proved) that when the code bin width is small and when the input sequence $s(n)$ traverses several transition levels between two successive samples, then the following properties apply:

1. Each sample of $e_q(n)$ is uniformly distributed over $(-\Delta/2, \Delta/2)$, thus having zero mean and variance $\Delta^2/12$.

[2] It has been argued that the quantization error should not be denoted 'error', since the ideal quantizer is intentionally designed to have this discrepancy between the input and the output. Hence, it is not an error.

2. The error sequence $\{e_q(n)\}$ is a stationary white sequence, that is, the autocorrelation $\mathrm{E}\{e_q(n)\, e_q(m)\} = 0$ for $n \neq m$.

3. The error $e_q(n)$ is uncorrelated with the signal $s(n)$.

On the other hand, in [Gra90] it is argued that, for a *random* input $s(n)$ with probability density function (PDF) $f_s(s)$, the approximation that $e_q(n)$ is uniform and white is valid only when all of the following conditions are fulfilled:

1. The signal is uniformly distributed.

2. The signal is kept within the full-scale range of the quantizer.

3. The number of quantization levels is large.

4. The code bin width is small.

5. The PDF $f_s(s)$ is smooth.

In [WKL96], yet alternative conditions are stated. These conditions are based on the PDF $f_s(s)$ of the input $s(n)$ and the characteristic function (CF) $\Phi_s(u) = \mathcal{F}\{f_s(s)\}$, where $\mathcal{F}\{\cdot\}$ denotes the Fourier transform, in this case with respect to s. It is shown[3] that the quantization noise $e_q(n)$ is exactly uniform, with zero mean and variance $\Delta^2/12$, if the CF is 'band-limited' so that $\Phi_s(u) = 0$ for $u > \pi/\Delta$. Also, when the same condition is met, the quantization noise is uncorrelated with the input. The results are further developed to a *pseudo quantization noise* model, where quantization is *approximated* with addition of an independent uniformly distributed noise $e_{\mathrm{pqn}}(n)$ (again with zero mean and variance $\Delta^2/12$); these two models are *not equal*, but under the above condition it can be shown that all moments and joint moments correspond exactly for $\mathcal{Q}_b[s(n)]$ and $s(n) + e_{\mathrm{pqn}}(n)$. The ideal b-bit quantizer is contrasted with the pseudo quantization noise model in Figure 1.4.

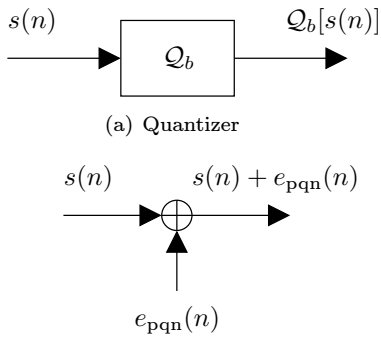
Regarding the whiteness of the quantization error it is shown in [WKL96] that when the *joint* CF of $s(0)$, $s(1)$, ..., $s(N-1)$,

$$\Phi_{s(0),\, s(1),\, \ldots,\, s(N-1)}(s_0,\, s_1,\, \ldots,\, s_{N-1}),$$

is 'band-limited', i.e., zero when $|s_k| > \pi/\Delta$ for all $0 \leq k \leq N-1$, the samples of $e_q(n)$ are independent of each other over time. Hence, the quantization error is a uniformly distributed, white, random process with zero mean and variance $\Delta^2/12$, independent of the input process.

The quantization noise $e_q(n)$ is in general *not* statistically independent of the input signal $s(n)$, since the (ideal) quantizer provides a deterministic mapping from $s(n)$ to $x(n)$, and thus also to $e_q(n)$. However, in [Gra90] it is proved that when an

[3] An alternative, but closely related, condition is also given in [WKL96].



(a) Quantizer

(b) Pseudo quantization noise model

Figure 1.4: A statistical model for quantization. The ideal b-bit quantizer (top figure) is contrasted with the pseudo quantization noise model (bottom figure). The noise $e_{\text{pqn}}(n)$ is uniformly distributed, white and zero-mean with variance $\Delta^2/12$, and is independent of $s(n)$. The two models are not equal, but under some conditions all moments and joint moments correspond exactly.

independent, identically distributed process $w(n)$ is added to a *quasi-stationary*[4] input process before quantization, then the error $e_q(n)$ is indeed independent of the input signal $s(n)$.

The matters of statistical quantization error and dithering are discussed in more depth in Chapter 4.

In addition to the approaches mentioned above, a deterministic derivation is possible [PM96], at least for sinusoidal inputs. The calculations conclude that the mean-square error power of the quantization error is $\Delta^2/12$, which is consistent with the probabilistic results above.

Coding

In practical A/D converters, the output from the quantizer, x_i, is coded in some way. The process of coding assigns a unique binary number to each possible output state of the quantizer. It is in the coding process that the actual 'bits' of the converter comes in. For a converter with $M = 2^b$ quantization levels, we need (at least) a b-bit binary number to associate each level with a unique code. A number of different binary coding systems exists [PM96], e.g., sign-magnitude, one's and two's complement. In this thesis we will not deal with coding of the quantized samples, except for Chapter 10 where bit-allocation is analyzed and binary coding

[4]See e.g., [Gra90] or [Lju99] for definition. The class of quasi-stationary processes includes both deterministic processes with convergent and bounded sample means and autocorrelations, as well as stationary random processes.

and bits comes in naturally. Elsewhere it suffices for our purposes to consider the quantized values (reconstruction levels) $\{x_j\}$.

Non-Ideal Converters

Practical converters are of course impaired by various types of errors and flaws. The sources of errors are numerous, and only a subset is mentioned here.

The first, and perhaps most obvious error is *erroneous code transition levels*, e.g., when a quantizer designed to be uniform becomes non-uniform. This is illustrated in Figure 1.5. Transition level errors often occur in converters utilizing a resistance ladder (e.g., flash converters, see Section 1.3) when the resistances are mismatched.

For converters incorporating an S/H device, this can be an origin of errors. The ideal S/H device is assumed to have zero switching time, so that the signal is sampled during an infinitely short sampling instant. However, practical circuits always have a certain switching time, over which the input signal is averaged. This is referred to as the *aperture time*. Also related to the S/H device is the *aperture uncertainty*, or *timing jitter*. This is the random deviation of the sampling instant from the nominal time. Let us assume that a certain sampling instant deviates from the nominal instant t_s by some (small) time δ, and that we are sampling a sinewave

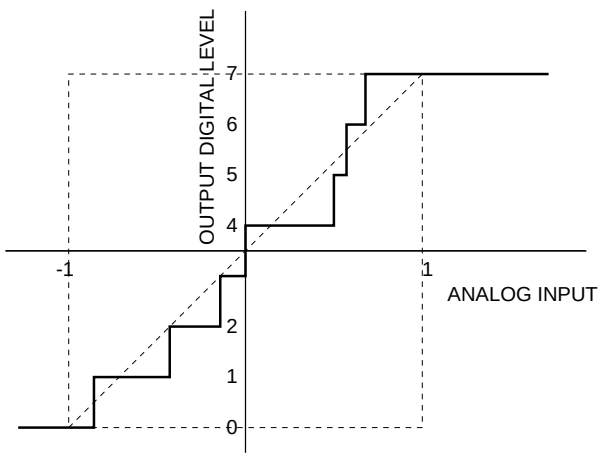


Figure 1.5: A 3-bit quantizer with non-uniform code bins.

with amplitude A and frequency f. Then, our sample, before quantization, will be

$$\begin{aligned} s &= A\sin(2\pi f(t_s+\delta)) = A\sin(2\pi f t_s)\underbrace{\cos(2\pi f\delta)}_{\approx 1} + A\cos(2\pi f t_s)\underbrace{\sin(2\pi f\delta)}_{\approx 2\pi f\delta} \\ &\approx A\sin(2\pi f t_s) + \underbrace{2\pi f\delta A\cos(2\pi f t_s)}_{\text{jitter induced error}}. \end{aligned} \tag{1.3}$$

The approximations $\sin(x) \approx x$ and $\cos(x) \approx 1$ for small x were used here. Thus, the effect of timing jitter will be a signal dependent noise, increasing in amplitude with the signal frequency f.

Other examples of error sources in A/D converters are distortion in the analog signal path, charge injection from the switch transistors [Jes01] and comparator thermal storage of signal history [WKN75].

Another way of classifying errors in the ADC output is to designate them as either *systematic* or *random*. Systematic errors are those that are deterministic or deterministically correlated to, for instance, the input signal, while an example of random errors is thermal noise.

1.3 ADC Architectures

In this section a brief description of three widespread ADC types is given. These are the flash, the successive approximation and the sigma-delta converters. The following descriptions are only intended as introductory examples to the field of A/D converters, and are in no way comprehensive; the reader should turn to the literature for an exhaustive theory, e.g., [Jes01, PM96, vdP94, Jon00, NST97]. The theory given here is however more than sufficient for the material to be presented in the following chapters. The theories for ADC correction in this work are in fact applicable to any ADC architecture, although the level of success will probably vary depending on converter type.

Flash and Pipelined Flash Converters

The flash converter is the only converter described here that truly operates in one sample clock cycle. In Figure 1.6 a fully parallel flash converter structure is shown. The converter consists of a resistance ladder, an array of $2^b - 1$ comparators and an encoder. The endpoints of the resistance ladder defines the input range of the ADC, while the intermediate points define the code transition levels. The input voltage V_{in} is simultaneously compared with *all* code transition levels in the comparators array. If V_{in} exceeds the k-th transition level, the k-th comparator output will be '1' (high, set, etc.), else it will be '0'. In the ideal case, i.e., when the resistance ladder and comparators array is without flaws, the output from the comparators will be *thermometer-coded* [Jes01] in 2^b bits. Finally, the encoder maps the highly redundant thermometer code to a b-bit binary-coded word.

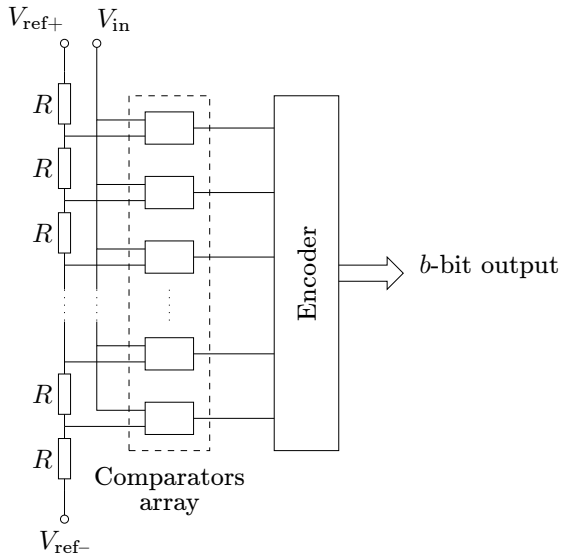


Figure 1.6: Flash converter.

Flash converters can be made to work very fast; sample rates up to 24 gigasamples per second (GSPS) and 3 bits have been reported [GDG04, N$^+$04]. The major drawback, on the other hand, is that the number of comparators grow exponentially with the number of bits b. This implies an exponential growth on both chip area and power consumption, putting a practical constraint on pure flash converters at about 10 bits, although most contemporary constructions typically have 5 or fewer bits.

Several ways to mitigate this problem have been proposed, of which probably the most common is the *pipelined* flash architecture, depicted in Figure 1.7. The pipelined flash ADC is formed by several flash ADCs in cascade, each having a rather low resolution. The quantized output from each stage is *digital-to-analog* converted back to an analog voltage, and the *residue* is formed. This residue is amplified and passed on to the next pipeline stage. Since each stage begins with an S/H device the data is propagated one stage every clock period. Hence, the conversion time is longer than for a fully parallel flash, but, the sample rate is the same; every clock cycle a new sample is produced. Pipeline converters of 16 bits and 100 MSPS (e.g., [AD9446]) are commercially available today.

In the non-ideal case, imperfections in the internal ADC stages, DAC stages or interstage gain will result in different kinds of error characteristics. Due to the recursive structure of a pipelined converter, the transfer function can display repetitive error patterns and shifted regions. It is common to implement pipelined converters with internal correction. By extending the range of the internal ADCs and DACs, mismatch errors in the ADC stages are intercepted. The output bits

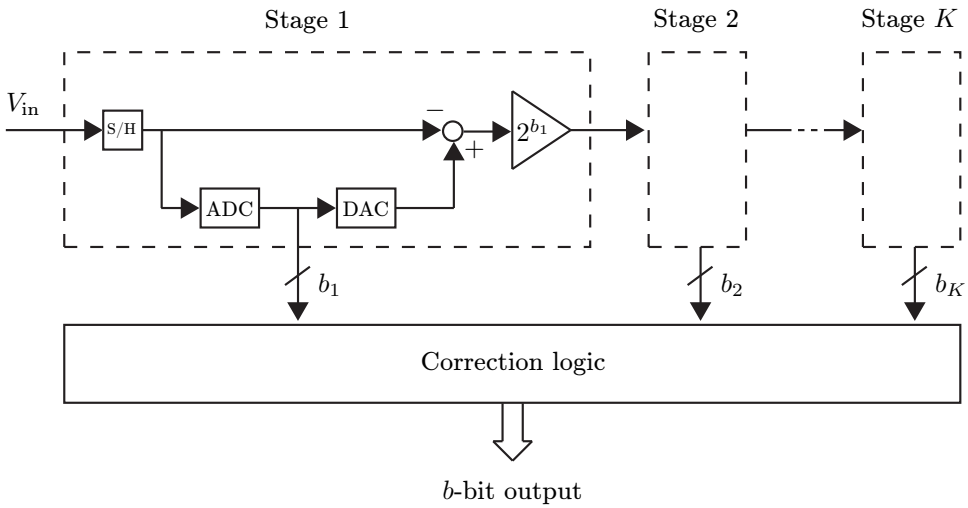


Figure 1.7: A K-stage pipelined flash converter with correction logic.

of the internal stages are augmented with overlapping bits; a 10-bit converter can for example be implemented as 4–3–3–3, i.e., using four stages with four bits in the first and three bits in the subsequent stages. Refer to [Jes01] for an elucidating discussion on error behavior and internal correction strategies.

Successive Approximation

Successive approximation converters (SA-ADC) determine the digital output by the *successive approximation algorithm*. Figure 1.8 shows the layout of an SA-ADC consisting of a sample-and-hold circuit, a comparator, a slave D/A converter and control logic. The control logic alters the output code, starting with the most significant bit (MSB) and continuing through to the least significant bit (LSB), so that the difference between the output of the slave DAC and the analog input is minimized. In each clock cycle one more output bit is determined. Thus, a b-bit SA-ADC requires b clock cycles to complete the conversion. Under the assumption of an ideal comparator, the performances of the SA-ADC are identical to those of the DAC [Jes01].

Variations of the successive approximation converter exists, including the *sub-ranging* SA-ADC and *pipelined* SA-ADC [Elb01].

Sigma-Delta Converters

A special form of ADC converters are sigma-delta converters. Figure 1.9 shows an outline of a generic sigma-delta converter. This special type of ADCs work

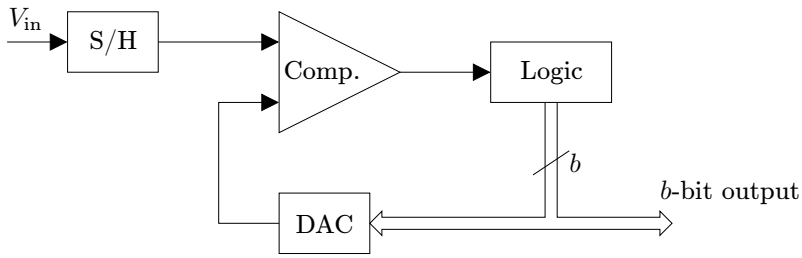


Figure 1.8: Typical successive approximation ADC.

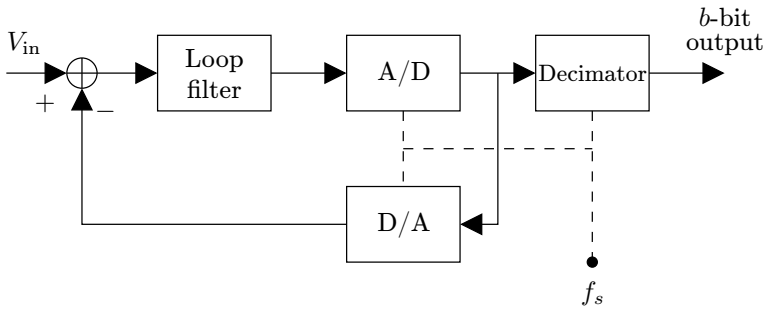


Figure 1.9: A generic sigma-delta ADC. The A/D and D/A converters are in the simplest form of sigma-delta converters implemented as one-bit converters.

with oversampling so that the bandwidth of the input signal is (much) less than $f_s/2$. In its most basic form, the sigma-delta converter contains a one-bit ADC and DAC. The decimator then produces a higher-resolution digital output word by averaging several one-bit samples, in addition to reducing the data rate. The loop filter provides 'noise shaping' and can be used to improve stability of the feedback loop. Thorough descriptions and analyses are found in the literature, e.g., [NST97] and [Jes01].

Sigma-delta converters are highly linear converters but applications are limited to rather narrow-band signals, owing to the oversampling.

1.4 Describing A/D Converter Performance

In order to be able to tell how "good" an ADC is, we must have measures and figures of merit. We are mainly concerned with the linearity of the ADC. At first glance it seems reasonable to believe that it suffices to consider the error introduced by the sampling and quantization, i.e., to look at $x(n) - s(nT_s)$, or some function thereof. This is the case when calculating the *signal-to-noise and distortion ratio* (SINAD). However, other characteristics can be more important, depending on the application, converter architecture, etc.

The IEEE Standard 1241, "Standard for Terminology and Test Methods for Analog-to-Digital Converters", [Std1241], provides a consistent set of terms, definitions and test methods for ADCs. The standard is intended to be applied when specifying, characterizing and evaluating ADCs. The measures below are all defined in the standard.

In the definitions below we will use the following notation:

$s(t)$ Continuous-time input to the ADC. This will typically be a spectrally pure, large amplitude (near full-scale) sinewave, $s(t) = A\sin(2\pi f_0 t + \phi) + C$, with C and A chosen so that the signal is centered within and spans a major part of the ADC input range (FSR). The phase ϕ is arbitrary.

$x(n)$ The output sequence from the ADC under test. This sequence will often be assumed to be of finite length N, so that we have a batch of data $x(0)$, $x(1)$, ..., $x(N-1)$.

$\check{s}(n)$ A least-squares sinewave fit to the data $x(n)$. This is our estimate of the actual input signal. (Standardized sinewave fitting methods are described in the literature [Std1241, Std1057, Hän00, And05].)

$X(f_d)$ The discrete Fourier transform (DFT) of the ADC output $x(n)$ for $n = 0, 1, \ldots, N-1$. [5]

Figure 1.10 shows a typical ADC output power spectral density (PSD). The PSD is based on experimental data from an Analog Devices AD9430, 12-bit, 210 MSPS pipelined flash converter [AD9430], exercised with a 60.12 MHz sinewave input. (See Appendix B.3 for a detailed description of the experimental data.) The measures pointed out in the figure, and some additional measures, will be described below.

Integral Nonlinearity (INL)

The standard [Std1241] states that "the integral nonlinearity is the difference between the ideal and measured code transition levels after correcting for static gain and offset." If T_k^o, $k = 1, 2, ..., M-1$, are the ideal transition levels and T_k, $k = 1, 2, ..., M-1$, are the actual code transition levels, then

$$\mathrm{INL}_k = \frac{GT_k + O - T_k^o}{\Delta} \tag{1.4}$$

is the INL for the k-th transition level in least significant bits. Here, G and O are the static gain and offset, respectively, and they are found by fitting a straight line to the actual transfer curve; cf. [Std1241] for the procedure. Two different versions are defined in the standard: independently based and terminal based INL. They differ in how the parameters G and O are found.

[5] In [Std1241] the quantity $X_{\mathrm{avm}}(f_d)$ is used, and denotes the average magnitude of the DFT at the discrete frequency f_d. The averaging is performed by taking several N-point data records, under the same test conditions, and taking the average of the magnitude of the DFT at f_d, that is $|X(f_d)|$.

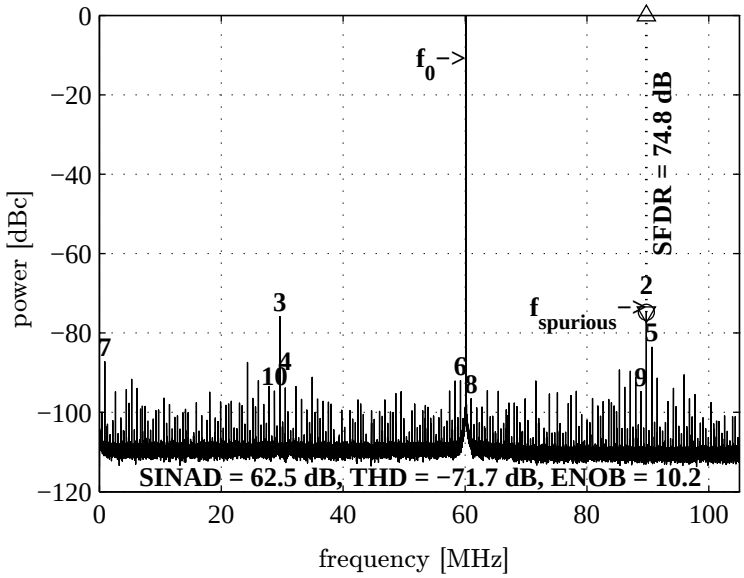


Figure 1.10: Typical output power spectral density of an ADC, this being from an Analog Devices AD9430. The fundamental frequency f_0 and the nine first harmonics (after aliasing) are labeled. Various performance measures are also stated in the figure.

Differential Nonlinearity (DNL)

Differential nonlinearity is defined as

$$\mathrm{DNL}_k = \frac{T_{k+1} - T_k - \Delta}{\Delta} \tag{1.5}$$

for $k = 1, \ldots, M-2$ (the semi-infinite end regions are omitted). That is, it is the difference between the actual and the ideal code bin width, expressed in least significant bits. For an ideal converter, DNL is zero. A DNL of -1 implies a missing code, i.e., the upper and lower transition level of a quantization region coincide, so that the width of the region is zero, and the corresponding output can never be produced—the code is missing.

Signal-to-Noise and Distortion Ratio (SINAD)

The signal-to-noise and distortion ratio is the ratio of the root-mean-square (rms) signal amplitude of the sinusoid test signal to the rms noise. Thus, with

$$P_{\mathrm{noise}} = \left(\frac{1}{N} \sum_{n=0}^{N-1} (x(n) - \check{s}(n))^2 \right)^{1/2} \tag{1.6}$$

being the rms noise, we have

$$\text{SINAD} = 20\log_{10}\frac{A}{\sqrt{2}P_{\text{noise}}} \quad [\text{dB}]. \tag{1.7}$$

For an *ideal* ADC, the only origin of noise is the amplitude quantization. We saw in Section 1.2 that for a uniform quantizer the quantization noise can (under certain conditions) be modeled as a uniformly distributed random process with power $\Delta^2/12$, so that the ideal rms quantization error is $\Delta/\sqrt{12}$. Inserting this into (1.7), the SINAD for an ideal uniform ADC is

$$\text{SINAD}_{\text{ideal}} = 20\log_{10}\frac{\sqrt{6}A}{\Delta} = 20\log_{10}\frac{\sqrt{6}A\,2^b}{\text{FSR}} \quad [\text{dB}]. \tag{1.8}$$

In the last equality, the relation $\Delta = \text{FSR}/2^b$ was used, where FSR denotes the full-scale range of the ADC. From (1.8) it is apparent that every additional bit increases the SINAD of an ideal ADC by $20\log_{10}2 \approx 6$ dB, which is a well known rule of thumb [Wal99, Wep95] (see also Section 4.1 and (4.2)).

Effective Number of Bits (ENOB)

The effective number of bits is related to the SINAD of the ADC. In (1.8) the relation between the number of bits b and the SINAD of an *ideal* ADC was given. For a practical converter the SINAD will be lower. Thus, by solving for b in (1.8), we get

$$\text{ENOB} = \frac{\log_2 10}{20}\,\text{SINAD} + \log_2\frac{\text{FSR}}{\sqrt{6}A}. \tag{1.9}$$

The effective number of bits should be interpreted as the number of bits required in an ideal converter to achieve a certain SINAD.

Spurious Free Dynamic Range (SFDR)

While the first two performance measures have been related to the total noise, the last two will be related to certain spectral components of the ADC output signal. The converter under test is still supposed to be exercised with a spectrally pure, large amplitude sinewave. Furthermore, the fundamental frequency f_0 of the test signal is now supposed to align with a DFT bin frequency and to be concurrent with the conditions for *coherent sampling*, as defined in [Std1241]. These are given in [Std1241, §4.1.4.5] as

$$f_0 = \frac{J}{N}f_s, \tag{1.10}$$

where J is an integer which is relatively prime to the record size N. The condition of J and N being relatively prime means that J and N have no common factors; when N is a power of 2, then any odd J meets this condition.

The spurious free dynamic range is the ratio of the magnitude of the DFT at the fundamental signal frequency and the magnitude of the DFT at the largest harmonic *or* spurious signal component observed over the full Nyquist range, that is,

$$\text{SFDR} = 20\log_{10}\frac{|X(f_0/f_s)|}{\max\limits_{f_d \neq f_0/f_s}|X(f_d)|} \quad [\text{dB}]. \tag{1.11}$$

The SFDR is in most practical cases dependent on both signal frequency and amplitude as well as sampling frequency. Therefore, these parameters should be specified along with the SFDR. In most cases the amplitude of the test signal is near full-scale, typically around -0.5 dBFS.

Spurious free dynamic range indicates the usable dynamic range of an ADC. In the likely scenario of detecting two (or more) separate signals of different amplitude, the SFDR indicates the maximum feasible ratio between the stronger and the weaker signal; with a ratio grater than the SFDR the weaker signal will probably be lost in the spurs generated by the converter itself from the stronger signal.

Total Harmonic Distortion (THD)

The total harmonic distortion is defined as

$$\text{THD} = \frac{1}{N}\sqrt{\sum_{f\in\{f_k\}}|X(f)|^2} \tag{1.12}$$

where $\{f_k\}$ is the set of the H first harmonics to f_0. The harmonic frequencies should be aliased so that $f_k \in [0, 0.5]$, which is described in [Std1241]. As with SFDR, the THD is dependent on the input signal parameters and possibly also on the sampling frequency, wherefore these should be specified. Like SFDR, THD is usually tested using a near full-scale sinewave.

1.5 The Post-correction Problem

This thesis is – as was stated already in the beginning of this chapter – dedicated to correction techniques for A/D converters. The contributions made are for the specific case of post-correction using look-up tables. Chapter 3 starts out by dissecting look-up table methods, but we will anticipate events and list a few specific problems of interest.

What is a good correction? This question is directly linked to what distortion measure that is being used. The vast majority of the results in this work are based on the mean-squared error criterion. Chapter 6 discusses this criterion, and a few other, and the implications of the criterions are stated.

How can the correction value be obtained? Knowing the optimal correction value in theory is one thing – obtaining a good set of correction values for a

practical ADC is a different matter. In Chapter 7 this problem is dealt with from an estimation point of view.

Which value in the table should be used? Or, if each entry has a unique index (address), how shall the index corresponding to a certain ADC output be determined? This is determined by something that is defined as an indexing scheme in Chapter 3. The indexing scheme determines the dynamic behavior – if any – of the post-correction system. The state of the art is reviewed in Chapter 3, and novel contributions are made in Chapter 9.

How can the memory requirements be reduced? One frequent problem in look-up table based post-correction systems is that they can be quite memory consuming. That is, the number of table entries and the (word) length of each entry simply amounts to a too large memory size. The contributions of Part III aim at reducing the number of entries, while theories presented in Chapter 8 analyzes the effects of reducing the word length.

1.6 Outline and Contributions

This thesis is organized into three parts, which are here briefly summarized.

Part I: ADC Correction Methods: An Overview The first part of the thesis provides an overview to the art of ADC correction. This part does not only deal with post-correction, but also discusses the concept of dithering, a technique where random noise is added to the input of the converter in order to obtain better quantization results.

After a short introduction in Chapter 2, Chapter 3 provides an overview to the art of look-up table based correction, considering indexing schemes, correction values and calibration. Chapter 4 deals with different forms of dithering. The key results within statistical quantization theory are reviewed, and the dithering theories based upon these results are given. Dithering for slowly varying signals and dithering for non-ideal quantizers is also considered. The last chapter in Part I, Chapter 5, deals with correction methods based on a mathematical model describing the ADC. The correction is found by inverting the model and connecting the ADC to be corrected and the model inverse in tandem, hopefully producing a system which cancels out the non-idealities.

Part II: Signal Processing in ADC Post-correction In the second part, the results akin to signal processing and information theory are given.

Chapter 6 provides different distortion measures that can be used when assessing the performance of systems in general, and quantizers in particular. Different choices of distortion measure yield different strategies for choosing the re-

construction levels of a quantizer. These theories provide the foundation for which values are considered good correction values in a post-correction system.

Chapter 7 addresses the problem of estimating the correction values for a practical ADC. The results presented are based on a calibration scheme where some kind of reference signal is obtained during a calibration phase. The reference is assumed to be the true input perturbed by some noise. Two different scenarios are considered and they result in two different estimators. In the first scenario, a mathematically tractable Gaussian assumption is made. The optimal estimator in the mean squared sense is shown to be the sample mean. The second scenario conforms with the classical staircase quantizer transfer function, and a non-linear estimator base on order statistics is found to be better than the sample mean. Also, the Cramér–Rao bound for the estimation problem at hand is derived.

Chapter 8 elaborates on the theoretical performance limits of an ADC after post-correction. In particular, the effect of using limited-resolution correction terms is investigated. A prediction of the resulting SINAD and ENOB after correction is provided. The prediction is based on four parameters: the number of bits, the variance of the random ADC input noise, the variance of the intrinsic DNL and the precision with which the correction terms are represented.

Part III: Generalized Look-Up Table Post-correction In the last part of this work, a generalized indexing scheme for look-up table based ADC correction is proposed and analyzed.

Chapter 9 defines the novel multidimensional look-up table correction strategy. In the context of the proposed method, the concept of *bit-masking* is introduced; bit masks are fundamental for the subsequent analysis of correction results. The performance of the method is illustrated with experimental ADC data for a few exemplary cases.

Chapter 10 is dedicated to the derivation of an analysis framework for the generalized correction method proposed in Chapter 9. The derivations are based on the Hadamard transform, which is also described in the chapter. It is shown that the outcome of many different correction schemes can be compared through matrix relations. In particular, a special *reduction matrix* is introduced in the chapter.

Chapter 11 illustrates the applicability of the analysis framework – in particular the reduction matrix – derived in the preceding chapter. The reduction matrix is employed in an optimization problem where the objective is to find the best allocation of a limited number of bits in the bit mask associated with an ADC correction system. Two different criterions are applied, viz. SINAD and THD.

Chapter 12 gives a suboptimal method for finding a solution to the rather complex optimization problems arrived upon in Chapter 11. The algorithm is a type of deflation algorithm.

Chapter 13 proposes another structure for look-up table correction schemes. In the proposed method, the frequency of a supposedly narrow-band signal is estimated using a frequency estimator based on look-up tables. The estimate is then used to decide which correction table to use to correct an ADC.

Publications

Part I was compiled as an overview of the state-of-the-art in ADC correction, published in

[Lun05] H. Lundin. An introduction to ADC error correction. Technical report, Royal Institute of Technology (KTH), May 2005. Course literature for the *5th Summer School on Data Acquisition Systems*, Italy, 2005. Available from: `http://www.ee.kth.se`.

The work in Chapter 7 was published in:

[LSH05b] H. Lundin, M. Skoglund, and P. Händel. On the estimation of quantizer reconstruction levels. In *Proceedings IEEE Instrumentation And Measurement Technology Conference*, volume 1, pages 144–149, Ottawa, Canada, May 2005.

and was submitted as

[LSH05c] H. Lundin, M. Skoglund, and P. Händel. On the estimation of quantizer reconstruction levels. *IEEE Transactions on Instrumentation and Measurements*, July 2005. Submitted.

The majority of the results of Chapter 8 was published in:

[LSH05a] H. Lundin, M. Skoglund, and P. Händel. ADC post-correction using limited resolution correction values. In *Proceedings IMEKO 10th Workshop on ADC Modelling and Testing*, volume 2, pages 567–572, Gdynia/Jurata, Poland, September 2005.

The material in Chapter 8 has also been submitted as

[LHS06] H. Lundin, P. Händel, and M. Skoglund. Accurate prediction of analog-to-digital converter performance after post-correction. In *Proceedings IMEKO XVIII World Congress*, Rio de Janeiro, Brazil, September 2006. Submitted.

The methods and results of Part III have been published in:

[LSH04] H. Lundin, M. Skoglund, and P. Händel. A criterion for optimizing bit-reduced post-correction of AD converters. *IEEE Transactions on Instrumentation and Measurements*, 53(4):1159–1166, August 2004.

[LSH05d] H. Lundin, M. Skoglund, and P. Händel. Optimal index-bit allocation for dynamic post-correction of analog-to-digital converters. *IEEE Transactions on Signal Processing*, 53(2):660–671, February 2005.

[LSH01] H. Lundin, M. Skoglund, and P. Händel. On external calibration of analog-to-digital converters. In *IEEE Workshop on Statistical Signal Processing*, pages 377–380, Singapore, August 2001.

[LSH02] H. Lundin, M. Skoglund, and P. Händel. A framework for external dynamic compensation of AD converters. In *Proceedings 7th European Workshop on ADC Modelling and Testing*, pages 135–138, Prague, Czech Republic, June 2002.

[LASH02] H. Lundin, T. Andersson, M. Skoglund, and P. Händel. Analog-to-digital converter error correction using frequency selective tables. In *RadioVetenskap och Kommunikation (RVK)*, pages 487–490, Stockholm, Sweden, June 2002.

[LSH03b] H. Lundin, M. Skoglund, and P. Händel. Optimizing dynamic post-correction of AD converters. In *Proceedings IEEE Instrumentation And Measurement Technology Conference*, pages 1206–1211, Vail, Colorado, USA, May 2003.

[LSH03a] H. Lundin, M. Skoglund, and P. Händel. Minimal total harmonic distortion post-correction of ADCs. In *International Workshop on ADC Modelling and Testing*, pages 113–116, Perugia, Italy, September 2003.

The material in Chapter 13 was also published in

[And05] T. Andersson. *Parameter Estimation and Waveform Fitting for Narrowband Signals.* PhD thesis, Royal Institute of Technology (KTH), June 2005. SB-0540.

A concise summary of the main results of Parts II and III is published in

[LSH05e] H. Lundin, M. Skoglund, and P. Händel. Signal processing results on dynamic ADC post-correction. In *GigaHertz, Proceedings of*, Uppsala, Sweden, November 2005.

Further results on ADC post-correction – including experimental results for MIMO communication systems – that are not included in this thesis have been published in:

[LSZ+04] H. Lundin, P. Svedman, X. Zhang, M. Skoglund, P. Händel, and P. Zetterberg. ADC imperfections in multiple antenna wireless systems—an experimental study. In *9th European Workshop on ADC Modelling and Testing*, pages 808–813, Athens, Greece, September 2004.

[ZJL+04] P. Zetterberg, J. Jaldén, H. Lundin, D. Samuelsson, P. Svedman, and X. Zhang. Implementation of SM and rxtxIR on a DSP-based wireless MIMO test-bed. In *The European DSP Education and Research Symposium EDERS*, November 2004.

[DVLR05b] L. De Vito, H. Lundin, and S. Rapuano. A Bayesian filtering-approach for calibrating a look-up table used for ADC error correction. In *Proceedings IEEE Instrumentation And Measurement Technology Conference*, volume 1, pages 293–297, Ottawa, Canada, May 2005.

[ZSLZ05] X. Zhang, P. Svedman, H. Lundin, and P. Zetterberg. Implementation of a smart antenna multiuser algorithm on a DSP-based wireless MIMO test-bed. In *Proceedings IEEE International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*, September 2005.

In addition, the work in [DVLR05b] was submitted for publication as a journal article:

[DVLR05a] L. De Vito, H. Lundin, and S. Rapuano. Bayesian calibration of a look-up table for ADC error correction. *IEEE Transactions on Instrumentation and Measurements*, June 2005. Submitted.

Parts of the work in Part III have also contributed to two patents:

[LHSP02] H. Lundin, P. Händel, M. Skoglund, and M. Pettersson. Adaptively calibrating analog-to-digital conversion. U.S. Patent 6,445,317, September 2002.

[LHS04] H. Lundin, P. Händel, and M. Skoglund. Adaptively calibrating analog-to-digital conversion with correction table indexing. U.S. Patent 6,690,311 B2, February 2004.

1.7 Notation

$\mathbb{R}$ The real numbers.

$\mathbb{R}^k$ The k-dimensional real space.

$\mathbb{B}^k = \{0, 1\}^k$ The k-dimensional binary space.

$\mathbb{Z}$ The integer numbers.

$\mathbb{Z}^k$ The k-dimensional integer space.

j The imaginary unit, defined through $j^2 = -1$.

$\mathbf{A}$ A matrix.

$\mathbf{a}$ A column vector.

$\mathbf{a}_i$ The i-th element of the vector $\mathbf{a}$. The notation $[\mathbf{a}]_i$ is used to avoid ambiguities.

$\mathbf{A}^{\mathrm{T}}$, $\mathbf{a}^{\mathrm{T}}$ The transpose of a matrix and a vector, respectively.

$\mathbf{A}^*$, $\mathbf{a}^*$ The complex conjugate transpose, or Hermitian adjoint, of a matrix and a vector, respectively.

$\mathbf{A}^{-1}$ The matrix inverse of $\mathbf{A}$.

$\mathrm{diag}\{\mathbf{A}\}$ The vector consisting of the main diagonal elements of the matrix $\mathbf{A}$.

$\mathrm{diag}\{\mathbf{a}\}$ The diagonal matrix with the elements of the vector $\mathbf{a}$ on the main diagonal.

$\mathrm{Tr}\{\mathbf{A}\}$ The trace of a matrix, i.e., the sum of the elements on the main diagonal.

$\mathrm{vec}\{\mathbf{A}\}$ The columns of $\mathbf{A}$ stacked in a vector.

$\mathbf{A} \otimes \mathbf{B}$ The Kronecker matrix product (see Section 10.1 for a definition).

$\mathbf{A} \circ \mathbf{B}$ The Hadamard (entrywise) matrix product (see Appendix 11.A).

$\mathbf{I}_N$ The N-by-N identity matrix.

$\mathbf{1}$, $\mathbf{1}_k$ A column vector of all ones, in the second case with length k specified.

$\mathbf{0}$, $\mathbf{0}_k$ A column vector of all zeros, in the second case with length k specified.

Δ The code bin width of the quantizer. For a uniform quantizer the relation $\Delta = \mathrm{FSR}/2^b$ holds, i.e., the full range of the ADC is divided into 2^b bins of equal width.

$I \xrightarrow{\mathbf{q}} J$ Mapping of an integer I through a 'bit mask' $\mathbf{q}$ to produce the integer J (see Section 9.1 and Definition 2 on page 117).

$\mathcal{F}$ The Fourier transform operator.

$\arg\min_x f(x)$ The value of x for which $f(x)$ is minimized.

δ_i The Kronecker delta function, defined for all integers i as

$$\delta_i \triangleq \begin{cases} 1, & i = 0, \\ 0, & i \neq 0. \end{cases}$$

Also, $\delta_{ij} \triangleq \delta_{i-j}$.

$abcd_2 = (a, b, c, d)_2$ Binary representation where a and d are the most and least significant bits, respectively.

E The expected value of a stochastic variable or function of a stochastic variable.

E_X The expected value taken with respect to the PDF of the stochastic variable X.

var The variance of a stochastic variable or function of a stochastic variable.

$\mathcal{N}(m, \sigma^2)$ The Gaussian or normal distribution with mean m and variance σ^2 (standard deviation σ).

$\{x_i\}_{i=1}^{N}$ A set with members $x_1, x_2, \ldots, x_N$.

1.8 Abbreviations

ADC Analog-to-digital converter.

CF Characteristic function. The Fourier transform of a PDF.

DAC Digital-to-analog converter.

dBFS dB ratio with respect to full scale.

DNL Differential nonlinearity.

ENOB Effective number of bits.

GSPS Giga-samples per second; billion samples per second.

FSR Full-scale range. The difference between the most positive and most negative analog inputs of a converter's operating range.

i.i.d. Independent identically distributed.

IMD Intermodulation distortion.

INL Integral nonlinearity.

LSB Least significant bit.

MAE Mean absolute error.

MLE Maximum likelihood estimator.

MMAE Minimum mean absolute error.

MMSE Minimum mean squared error.

MSB Most significant bit.

MSE Mean squared error.

MSPS Mega-samples per second; million samples per second.

MVU Minimum variance unbiased (estimator).

NPR Noise power ratio.

PDF Probability density function.

PMF Probability mass function.

PSD Power spectral density.

rms Root-mean-square.

SFDR Spurious free dynamic range.

SINAD Signal-to-noise and distortion ratio.

s.v. Stochastic variable.

THD Total harmonic distortion.

Part I

ADC Correction Methods
An Overview

Chapter 2

Introduction to Part I

This part gives an introduction to correction methods for analog-to-digital converters. The material is mainly intended to provide an overview, and the motivated reader is encouraged to pursue deeper knowledge in the references given herein. The work is divided into three chapters, each covering a special form of ADC correction. The classification into different families of methods follows that of [BDR05] to a large extent.

In Chapter 3, methods that are implemented using look-up tables are reviewed. Post-correction using look-up tables is a very common way of diminishing ADC errors, and extensive research has been conducted within this field. As a natural consequence, Chapter 3 only gives a brief summary of some of the most common methods, and should not in any way be seen as a complete description of the subject.

Chapter 4 covers the method known as dithering. The word dithering is used for a group of methods that all add noise to the input signal, prior to sampling and quantization. Chapter 4 starts with the fundamental theories of (ideal) quantization in order to facilitate the understanding of how additional noise can be beneficial. In addition to improving ideal quantizers, dithering can also be useful in randomizing the error patterns of non-ideal converters, as well as providing increased resolution (through averaging) for slowly varying signals.

In Chapter 5 methods that are based on a mathematical model of the ADC are presented. In particular, the chapter is focused on inverting Volterra models.

Chapter 3

Look-Up Table Based Methods

ADC post-correction using look-up tables (LUTs) is probably the most frequently proposed method for ADC correction in the literature, and is the post-correction method that was first introduced.[1] The outline of a generic LUT correction system is shown in Figure 3.1. The basic idea of the method is that the output samples from the ADC are used as index, or address, into the table – possibly using some indexing function. The index points out a specific entry value in the table, and the value is either added to or used to replace the current ADC output sample.

In theory, any post-correction methods that operate on a finite sequence of ADC output samples can be represented as an LUT. However, implementation issues limit the feasible methods to those of limited dynamic dependence, that is, only those methods that directly use a few subsequent samples for indexing can be successfully implemented as LUTs. Methods targeted at mitigating highly dynamic error effects must be implemented using some kind of arithmetic on-line computation of the correction values (cf. Chapter 5).

3.1 Classification of LUT Methods

Returning to Figure 3.1, we will in this chapter classify various LUT methods depending on how they implement the various blocks of the figure. In particular, we will address the following parts:

Indexing scheme Determines how the table index I is generated from the sequence of output samples $\{x(n)\}$. Static, state-space, and phase-plane correction methods can all be incorporated into this framework through proper design of the indexing function.

Correction vs. replacement The look-up table can either be used to store correction values to be added to the ADC output sample ($\hat{s}(n) = x(n) + e_I$), or

[1] Dithering methods were proposed earlier, but they do not fall into the class of post-correction methods.

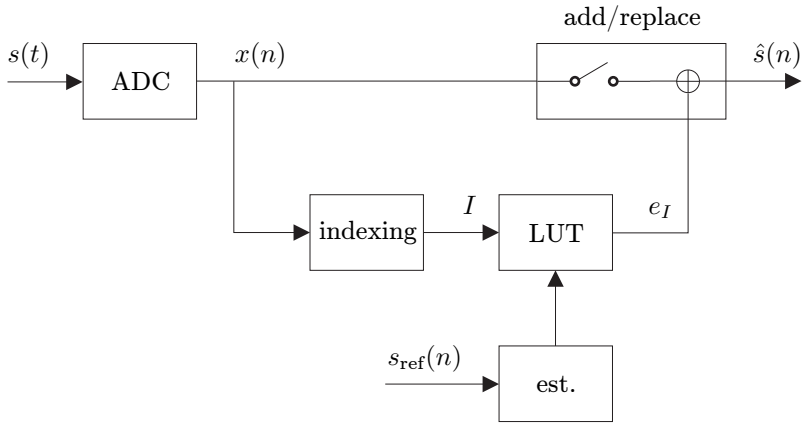


Figure 3.1: A generic look-up table correction system.

replacement values so that the output is simply replaced with a value from the table ($\hat{s}(n) = e_I$).

Nominal value An obvious question when considering ADC post-correction is with what values the table should be loaded. Different views on this issue results in slightly different strategies.

Reference signal Calibration of the LUT is a nontrivial task indeed, and the choice of calibration signal has proven to be of paramount importance. Different choices of calibration signal also give different possibilities of how to obtain a reference signal $s_{\text{ref}}(n)$ in the digital domain, which is needed for calibration of the LUT. (The definitions of calibration and reference signals are provided in Section 3.4 below.)

Estimation methods Different strategies on how to obtain the table values from the reference signal have been proposed in the literature.

The above issues are all treated in the following sections.

3.2 Indexing Schemes

The indexing scheme is perhaps the most significant part of the LUT system, and also the part that determines the size and structure of the actual table. Generally speaking, the indexing function operates on a vector of output samples $\begin{bmatrix} x(n-K_a) & x(n-K+1) & \dots & x(n) & \dots & x(n+K_b) \end{bmatrix}^{\mathrm{T}}$ ($\cdot^{\mathrm{T}}$ denotes the transpose of a vector) and produces a nonnegative integer index I associated with sample index n. The indexing function is in most cases causal, so that $K_b = 0$ and

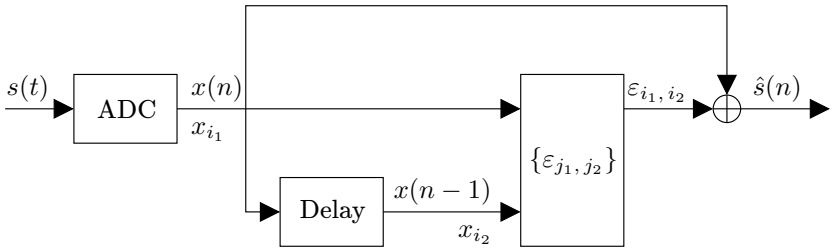


Figure 3.2: Two-dimensional state-space correction table.

$K \triangleq K_a \geq 0$. How the available input samples are mapped to an index is what differs from one indexing scheme to another.

The size of the table is determined by the range of possible indices $I \in \{0, 1, \ldots, I_{\max}\}$. In the vast majority of the cases, the size of the table is a power of 2, say 2^B with B being a positive integer, implying that the index I can be represented in a binary format using B bits, and $I_{\max} = 2^B - 1$.

In the following we will give a short résumé of the most commonly used indexing schemes.

Static Indexing

A static look-up table correction scheme maps the present sample $x(n)$ into an index I,

$$x(n) \rightarrow I, \tag{3.1}$$

that is, the index depends neither on past nor on future samples. In its most basic form, the index I is simply the binary b-bit word given by the ADC, so that $I = x(n)$ where $x(n)$ is in binary format. It is also possible to reduce the index space by further quantizing the ADC output, i.e., discarding one or several of the least significant bits (LSBs) in $x(n)$ providing an index of $B < b$ bits, as proposed for instance in [DC90].

It is obvious that this scheme will produce the same index I for a given ADC output *regardless of the signal dynamics* (e.g., regardless of signal history). Thus, it is of significant importance that the errors of the ADC stay constant in the intended range of operating signals for the ADC, and do not change depending on which input signal is being applied.

This is the method proposed, for example, in [Iro86] and [HSP00]. In the latter it was demonstrated that static correction may improve performance for some frequencies, while deteriorating it for other frequencies—this is a typical indication that the ADC possesses some significant dynamic error mechanism.

State-Space Indexing

One way to introduce dynamics into the correction scheme is to adopt a state-space structure. The current sample $x(n)$ and the previous sample $x(n-1)$ are used to build the index:

$$(x(n),\, x(n-1)) \;\rightarrow\; I. \tag{3.2}$$

This method is referred to as *state-space indexing*, and is illustrated in Figure 3.2. The basic form is when the b bits from $x(n)$ and $x(n-1)$ are concatenated to form an index of $B = 2b$ bits. The indexing is undoubtedly dependent on signal dynamics, since the index for a sample $x(n) = x_i$ is potentially different for different values of the previous sample $x(n-1)$. This scheme can be equivalently described as a two-dimensional LUT where $x(n)$ and $x(n-1)$ are used to index the two dimensions, respectively. State-space ADC correction is proposed in, for example, [IHK91] and [TL97].

The two-dimensional state-space method generalizes to an indexing scheme utilizing K delayed samples in conjunction with the present sample for indexing:

$$(x(n),\, x(n-1),\, \ldots,\, x(n-K)) \;\rightarrow\; I. \tag{3.3}$$

Again, the basic form would just take all the samples and concatenate all the bits into an index of $B = (K+1)b$ bits. This extended scheme was alluded to by Tsimbinos in [Tsi95].

An immediate problem with extending the dimension of the table is that the memory required to store the table becomes unwieldy very fast. The number of table entries is $M \triangleq 2^B = 2^{(K+1)b}$ and we see that it grows exponentially in K. The number of ADC bits, b, of course comes in to the equation, but it is reasonable to say that for resolutions common in high-speed ADCs – some 8 to 14 bits in general – it is not practical to have K greater than 2.

In order to tackle the memory problem, measures must be taken to reduce the index space. One way to accomplish this is to apply further quantization (or truncation) to the delayed samples, so that they are represented with *less* than b bits resolution (a method used for state-space indexing in [TMBSL02], and in the context of phase-plane correction in [RI87] and [DVBS92]). In Chapter 9 of the present work, this approach is generalized to say that a number less than or equal to b bits are used from the sample $x(n-k)$ (for $k \in \{0,\, 1,\, \ldots,\, K\}$). However, these are not necessarily the most significant bits but can be selected from *all* b bits of $x(n-k)$. That is, some of the bits in the sample $x(n-k)$ are discarded, and the remaining bits are used for addressing.

Phase-Plane Indexing

As an alternative to state-space indexing, the phase-plane indexing, described in, for example, [RI87], [Mou89], [Hum02], and [Ber04], may be used; sometimes the

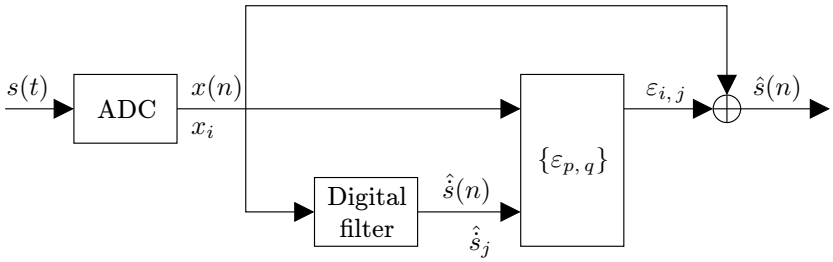


Figure 3.3: Two-dimensional phase-plane correction table. The filter calculates an estimate of the signal slope for each sample. The estimate is represented with a binary word $\hat{\hat{s}}_j$ of finite precision (say b_2 bits, not necessarily equal to b) and is used as part of the table index.

term code-slope indexing is used. The table index is constructed from the present sample $x(n)$ and an estimate of the slope (derivative) of the input signal $\hat{\hat{s}}(n)$:

$$\Big(x(n),\, \hat{\hat{s}}(n)\Big) \;\rightarrow\; I. \tag{3.4}$$

The slope can either be estimated from the output samples, using for instance the backward difference $x(n) - x(n-1)$ or an FIR differentiator filter [Mou89, HICP94, TMBSL02], or using an analog differentiator and a separate (possibly low resolution) ADC sampling the output of the differentiator [IHK91]—the former variant is exemplified in Figure 3.3. Just as in the state-space case, the indexing can be generalized to higher order

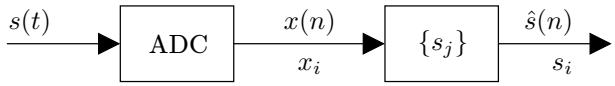
$$\Big(x(n),\, \widehat{s^{(1)}}(n),\, \ldots,\, \widehat{s^{(K)}}(n)\Big) \;\rightarrow\; I, \tag{3.5}$$

where $\widehat{s^{(k)}}$ denotes the estimate of the k-th derivative. Addressing with higher order derivatives has been reported in [DVBS92].

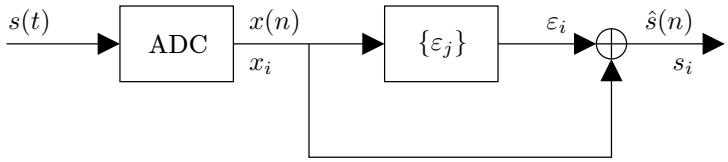
3.3 Correction Values

While the indexing regime determines which table entries will be used at all times, the actual value of the table entry is still not settled. In this section we will review the most common approaches, and also touch upon the issue of finite precision in the correction values.

First, however, we go back to the distinction between correction versus replacement values. In the former case, the table is filled with correction terms that are added to the output from the ADC, while the output is replaced by the LUT values in the latter case. In other words, if the ADC produces an error e_I (difference between some nominal and the actual output) for some index I, then a correction scheme would store e_I in the table, while the corresponding replacement scheme



(a) Replacement correction



(b) Additive correction

Figure 3.4: Two types of static correction tables. In 3.4(a) the ADC output $x(n) = x_i$ is replaced by the table entry s_i, while in 3.4(b) the corrected output is produced by adding a correction term ε_i to the ADC output x_i.

would store $e_I + x$, where x is equal to the current output of the ADC. The two alternatives are depicted in Figure 3.4(a) and Figure 3.4(b), respectively.

From an implementation point of view, the replacement scheme is beneficial, since no addition is needed during correction, which is not the case in the correction approach. Any correction system implemented as a replacement system can be equivalently replaced using a correction type system, while the converse is not true. It is in fact only in the case when the table index $I(n)$ is unique for distinct current samples $x(n)$ that the correction based system can be replaced with a replacement system, i.e., if $x(n) \neq x(m) \rightarrow I(n) \neq I(m)$. Looking back at Section 3.2 we see that it is only those schemes where all the bits from the current sample $x(n)$ go straight into the index that fulfill this criterion, e.g., the basic forms of static, state-space and phase-plane corrections. The methods using further quantization of $x(n)$ mentioned at page 32 does not qualify, since if any of the bits in $x(n)$ are omitted, then different values of $x(n)$ that differ only in the bits removed may give the same index I. The same applies for the generalized scheme presented in Chapter 9.

Nominal Value

The goal with ADC post-correction is of course to produce an output that is better than before correction. A few approaches are listed here. More detailed descriptions are provided in Chapter 6.

Midpoint Correction The midpoint correction strategy is based on the assumption that the ADC acts as a staircase quantizer with M quantization regions $\{\mathcal{S}_j\}_{j=0}^{M-1}$. The i-th region is delimited below and above by the transition levels T_i and T_{i+1}, respectively (cf. Section 1.2). The table value for an additive

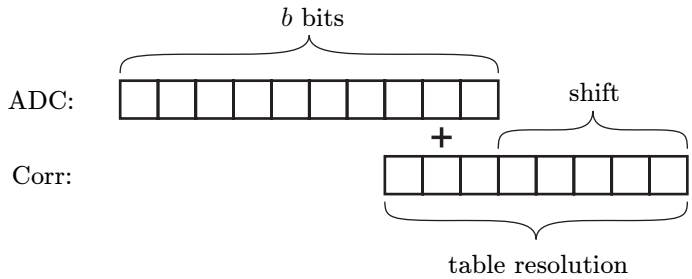


Figure 3.5: Addition of the ADC output with a correction value. The bits of the table value are shifted in order to enhance the precision of the corrected ADC.

correction type system using midpoint correction should be

$$e_i = \frac{T_i + T_{i+1}}{2} - x_i. \tag{3.6}$$

MMSE Correction In the minimum mean squared error approach the input $s(n)$ is considered to be drawn from a stochastic variable S with PDF $f_S(s)$. Then, the MMSE correction value, i.e., the correction that minimizes $\mathrm{E}[(\hat{s}(n) - s(n))^2]$, is

$$e_i = \frac{\int_{s \in \mathcal{S}_i} s\, f_S(s)\, ds}{\int_{s \in \mathcal{S}_i} f_S(s)\, ds} - x_i. \tag{3.7}$$

Minimum Harmonic Correction A specialized method that assigns correction values so that the harmonic distortion after correction is minimized.

Precision of Correction Values

In a practical post-correction application the correction values are stored with fixed-point precision. This does, of course, affect the outcome of the correction.

An ADC can in general be assumed to have zero errors in a number of the most significant bits. Hence, these bits will never be in need of correction, and if an additive correction system is applied (cf. Figure 3.4) no memory have to be used to store any correction for these bits. We can instead focus on the least significant bits, and even use the excessive word length in the LUT for sub-LSB correction. For example, if a 10-bit converter is known to have errors only in the 3 least significant bits, then an 8-bit correction word could be shifted 5 steps so that 5 correction bits are sub-LSB bits, or binary decimal bits. Figure 3.5 gives a graphical explanation for this. Note that the shifting is not possible when implementing a replacement-value correction system, since the replacement value stored in the LUT must have all bits.

The choice of correction word length and number of bits to shift are both design parameters to be fixed when implementing a post-correction system, and depend on the ADC to be corrected. It should be noted here that if the correction value is not shifted, so that the LSBs of the ADC and the correction align, then the correction system can only mitigate errors that are $\geq 1/2$ LSB.

In Chapter 8 of this thesis, these matters are dealt with in greater detail.

3.4 Calibration of LUTs

Prior to first use the correction table must be calibrated. The ADC under test is calibrated experimentally, i.e., a signal is fed to the input of the ADC and the transfer characteristics of the ADC are determined from the outcome. Many methods require that a *reference signal* is available in the digital domain, this being the signal that the actual output of the ADC is compared with. This reference signal is in the ideal case a perfect, infinite resolution, sampled version of the signal applied to the ADC under test. In a practical situation, the reference signal must be estimated in some way. This can be accomplished by incorporating auxiliary devices such as a reference ADC, sampling the same signal as the ADC under test [EM04], or a DAC feeding a digitally generated calibration signal to the ADC under test [TL97, Hum02]. Another alternative is to estimate the reference signal by applying signal processing methods to the output of the ADC under test. Special cases of this exist; in [Elb01] methods for blind calibration, assuming only a smooth but otherwise unknown probability density function for the reference signal, are presented, while [HSP00] proposes sinewave reference signals in conjunction with optimal filtering techniques to extract an estimate of the reference signal. The latter method was further developed in [LSH01, Lun00] to include adaptive filtering techniques, adapting to a calibration signal that is switching between a number of frequencies.

When the reference signal has been acquired, the table values should be estimated from it. When using the MMSE approach (3.7) above, the following method is frequently used. Since neither the regions $\{\mathcal{S}_j\}$ nor the PDF $f_S(s)$ is known in general, we are forced to use a more practical method. Assume that the reference samples collected during calibration results in a set $\mathcal{C}$ of N samples. Each sample in $\mathcal{C}$ belongs to one, and only one, quantization region $\mathcal{S}_j$. Hence, we can split $\mathcal{C}$ into $M = 2^b$ subsets, $\{\mathcal{C}_j\}_{j=0}^{M-1}$, such that $s_{\text{ref}}(n) \in \mathcal{C}_i \Rightarrow s(n) \in \mathcal{S}_i$. Here, $s(n)$, $n = 0, \ldots, N-1$, are the calibration samples input to the ADC under test and $s_{\text{ref}}(n)$, $n = 0, \ldots, N-1$, are the corresponding reference samples. It is assumed that the sample-and-hold of the ADC under test is ideal, so that the entire error behavior is captured in the following quantizer and the discrete-time signal $s(n)$, $n = 0, \ldots, N-1$, can be considered to be the exact calibration signal.

To summarize, $\mathcal{C}_i$ contains all reference samples in $\mathcal{C}$ collected when the index i was produced in the ADC under test. Each subset $\mathcal{C}_j$ has N_j samples, and naturally $\sum_{j=0}^{M-1} N_j = N$. Since the actual PDF $f_S(s)$ is unknown, the collected reference

samples $\mathcal{C}$ is all information at hand. We assign to each sample in $\mathcal{C}$ the probability $1/N$, i.e., all samples in $\mathcal{C}$ are equally probable, and the probabilities sum up to one. Now we can approximate the integrals in (3.7) with

$$\int_{s\in\mathcal{S}_j} s\, f_S(s)\, ds \approx \sum_{s\in\mathcal{C}_j} s\, \frac{1}{N} = \frac{1}{N} \sum_{s\in\mathcal{C}_j} s = \frac{N_j}{N} \bar{\mathcal{C}}_j \tag{3.8}$$

$$\int_{s\in\mathcal{S}_j} f_S(s)\, ds \approx \sum_{s\in\mathcal{C}_j} \frac{1}{N} = \frac{N_j}{N}, \tag{3.9}$$

so that

$$x_{j,\,\mathrm{opt}} \approx \bar{\mathcal{C}}_j, \tag{3.10}$$

where $\bar{\mathcal{C}}_j$ is the arithmetic mean of all samples in $\mathcal{C}_j$. Chapter 7 provides further results on correction value estimation using the sample mean. Moreover, an alternative estimator, based on order statistics, is derived and analyzed.

Calibration methods that do not rely on any digital reference signal has also been proposed in the literature. In [EM04], a method is proposed that estimates the integral nonlinearity (INL) from the output code histogram and subsequently builds an LUT from the INL sequence.

Daponte et al. proposes a hybrid correction system in [DHH$^+$02]. The correction comprises an LUT using the minimum mean-squared approach followed by a low-pass filter. The filtering is possible since the system is aimed at over-sampling applications, so that the signal of interest only can reside in the lower part of the spectrum. The LUT is calibrated using the sinewave histogram method and Bayesian estimation.

Chapter 4

Dithering Based Methods

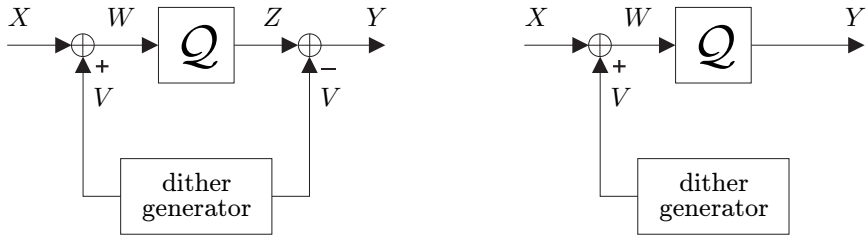
Distortion resulting from quantization – both ideal and non-ideal quantization – can often be reduced using a technique called *dithering*. The method can be divided into subtractive and non-subtractive dithering. Figure 4.1 shows the two different structures. The somewhat counterintuitive basic idea of dithering is to add some kind of noise to the signal prior to quantization. The same noise signal is subsequently subtracted after quantization in the subtractive method, while this is obviously not the case in the non-subtractive one. There are three main purposes for adding noise to the signal:

1. Break up statistical correlations between the quantization error and the input signal, and make the popular pseudo quantization noise model valid.
2. Randomize the DNL pattern of a non-ideal uniform quantizer.
3. Increase the resolution for slowly varying signals.

The three approaches will be briefly explained in this chapter. The basics of statistical quantization theory are given in Section 4.1 while Section 4.2 explains how dithering can reduce the distortion by randomization. Finally, dithering in conjunction with low-pass post-processing is dealt with in Section 4.3.

4.1 Statistical Theory of Quantization and Dithering

In this section we will give a short historical overview of the development of statistical quantization theory and provide the key results, intentionally focused on dithering applications. The motivated reader will find a comprehensive insight into the topic with numerous references and an exhaustive historical résumé in [GN98], which is the main reference for the historical overview given here. We will restrict ourselves to fixed-rate scalar quantization, and refrain from dealing with deeper information theoretical concepts such as variable-rate quantization and vector quantization. Following the quantization theory are the theories for dithering.



(a) Subtractive (b) Non-subtractive

Figure 4.1: Figures of subtractive and non-subtractive dither.

In this section we are mainly concerned with small-scale dithering for ideal uniform quantizers; large-scale dithering and dithering intended to mitigate DNL errors is considered in Sections 4.2 and 4.3.

Perhaps the oldest example of quantization is rounding off, first analyzed by Sheppard [She98] in his work on histograms. The real starting point for quantization theory, however, was the invention of pulse-code modulation (PCM), patented by Reeves in 1938 and accurately predicted to become a ubiquitous part of communication, both for audio and video. Oliver, Pierce and Shannon [OPS48] provided the first general contribution to statistical quantization theory in their analysis of PCM for communications. One of their contributions is the classical result that for a high resolution quantizer the average distortion, in terms of squared-error (i.e., quantization noise power), is

$$\sigma_Q^2 \approx \frac{\Delta^2}{12}, \tag{4.1}$$

where Δ is the width of the quantization regions. A uniform quantizer with $M = 2^b$ quantization regions and an input range V would then have $\Delta = V/2^b$ and (4.1) can be applied to yield the classical "6-dB-per-bit" result for the signal-to-noise and distortion ratio (SINAD)

$$\text{SINAD} = 10 \log_{10} \frac{\text{signal power}}{\frac{V^2}{2^{2b}\,12}} \approx 6.02b + \text{constant}. \tag{4.2}$$

In response to the need for a simple linear model for the effects of quantization, the *pseudo quantization noise* model was introduced. The model replaces a quantizer with an additive noise source, independent of the input signal – see Figure 1.4 on page 7. The model was popularized by Widrow [Wid56, Wid61, WKL96], who also gave conditions for when it is valid.

The pioneering work on dithering was carried out by Roberts in his work on image quantization [Rob62]. Roberts argued that adding noise to an image before quantization and subtracting it before reconstruction could mitigate the quantiza-

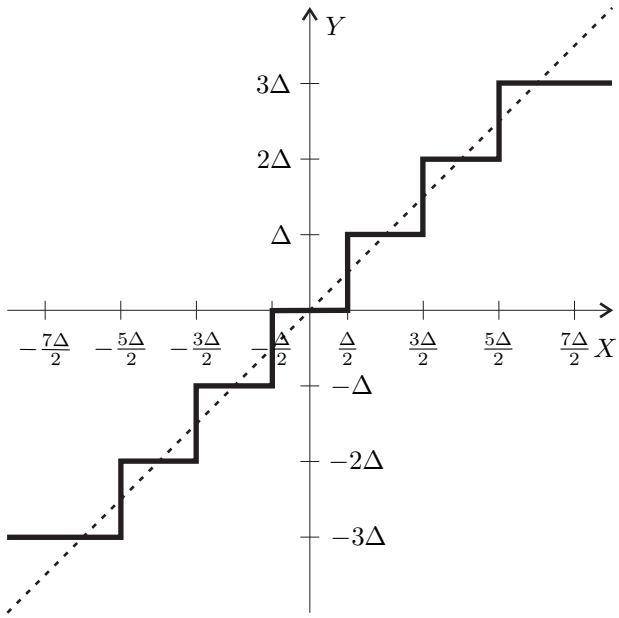


Figure 4.2: The input–output transfer function of a uniform, scalar, mid-tread quantizer with quantization step size Δ.

tion effects on the image, seen as regular, signal dependent, patterns. The general theory of dithering was then further developed by Schuchman [Sch64].

Quantization Theory

The following results, many of them originally due to Widrow, give the amplitude quantization counterpart to the widespread Nyquist's sampling theorem (note that sampling can be seen as time quantization, and conversely, amplitude quantization can be seen as sampling the amplitude). The following results and discussion applies to uniform, scalar, mid-tread quantization with a quantization step size Δ. Figure 4.2 shows the input–output transfer function of such a quantizer.

Assume that the input to a quantizer is a stochastic variable (s.v.) X with probability density function (PDF) $f_X(x)$. The output from the quantizer is an s.v. denoted Y. Figure 4.3 shows the signal relations, including the quantization error $E \triangleq Y - X$. The Fourier transform of the input PDF, usually referred to as

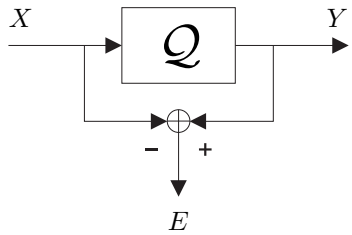


Figure 4.3: The quantizer $\mathcal{Q}$ with input X, output Y and error E defined.

the characteristic function (CF), is[1]

$$\Phi_X(u) = \int_{-\infty}^{\infty} f_X(x) e^{jux}\, dx = \mathrm{E}\left[e^{juX}\right]. \tag{4.3}$$

The statistical expected value is denoted $\mathrm{E}[\cdot]$. Two quantization theorems based on the CF were provided by Widrow, and they are recapitulated here:

Theorem 1 (QT I)**.** *If the CF of X is bandlimited, so that*

$$\Phi_X(u) = 0 \quad \textit{for} \quad |u| > \frac{\pi}{\Delta}, \tag{4.4}$$

where Δ is the width of the quantization regions, then the CF (PDF) of X can be derived from the CF (PDF) of Y.

Theorem 2 (QT II)**.** *If the CF of X is bandlimited, so that*

$$\Phi_X(u) = 0 \quad \textit{for} \quad |u| > \frac{2\pi}{\Delta} - \varepsilon, \tag{4.5}$$

with ε positive and arbitrarily small, then the moments of X can be calculated from the moments of Y.

The proofs are for instance provided in [WKL96] and are based on impulse-train sampling and calculation of moments from characteristic functions. Theorem 1 is in direct analogy with Nyquist's sampling theorem: if the Fourier transform of the continuous-time signal is bandlimited within a maximum angular frequency $\omega_{\max}$, then the continuous-time signal can be perfectly reconstructed from the sampled signal if the samples are taken at an angular frequency at least $2\omega_{\max}$. This is because quantization is in fact a discretization in the PDF domain, just as sampling

[1] As usual there are different definitions of the Fourier transform and care must be taken when interpreting results based on characteristic functions. In [WLVW00], for instance, the definition is $\Phi_X(u) = \int_{-\infty}^{\infty} f_X(x) \exp\{-j2\pi ux\}\, dx = \mathrm{E}\left[\exp\{-j2\pi uX\}\right]$. The definition (4.3) is user throughout this work, and results from sources where another definition has been used are rewritten to match the current notation.

is a discretization in the time domain. We see that Δ – the distance between two adjacent quantization points – is the quantization counterpart of the sampling period – the distance between two adjacent samples.

It is of less importance for our purposes here to know whether we can reconstruct the input PDF from the output PDF, which is the primary result of the quantization theorems above. However, the theorems can be used to say when a pseudo quantization noise (PQN) model can be used with success. The PQN model (sometimes also called the classical model of quantization) models a quantizer as an additive, white noise source, independent of the input signal, and with a zero-mean uniform distribution with variance $\Delta^2/12$ (i.e., uniform in $[-\Delta/2, \Delta/2]$). This model can of course never hold true, since the quantization error, defined as the difference between the output and the input of the quantizer, is a *deterministic* function of the input signal. However, when the conditions for Theorem 1 or Theorem 2 are met, all moments and joint moments correspond exactly for quantization and the addition of said independent noise [WKL96], and the PDF of the quantization error is exactly zero-mean uniform with variance $\Delta^2/12$ [Wid56, Wid61]. Under the same conditions it can also be shown [Wid56, Wid61] that the quantization error is uncorrelated with the input to the quantizer.

Sripad and Snyder [SS77] gave a weaker sufficient and necessary condition for the quantization error:

Theorem 3. *The PDF of the quantization error is uniform in* $[-\Delta/2, \Delta/2]$ *if and only if the CF of the input satisfies*

$$\Phi_X\left(\frac{2\pi n}{\Delta}\right) = 0 \quad \textit{for all integers } n \neq 0. \tag{4.6}$$

It is obvious that this condition is milder than that of Theorem 2. In particular, this is no longer a band-limiting constraint. In the same paper it is also shown that under the condition (4.6), the correlation between the input X and the quantization error E is

$$\mathrm{E}[X \cdot E] = \frac{\Delta}{2\pi} \sum_{k \neq 0} \frac{(-1)^k}{k} \left.\frac{\partial}{\partial u} \Phi_X(u)\right|_{u=2\pi k/\Delta}. \tag{4.7}$$

A sufficient condition for (4.7) to equate to zero is $\left.\frac{\partial}{\partial u}\Phi_X(u)\right|_{u=2\pi k/\Delta} = 0$ for all integers $k \neq 0$.

The above results have only considered the case of quantizing one s.v., that is, one single sample, and does not provide any temporal information such as the color of the quantization noise. Corresponding results for quantization of a sequence of samples were derived by Widrow [WKL96] using joint PDFs and CFs, and again a milder sufficient and necessary condition was provided by Sripad and Snyder [SS77]. We only provide the latter:

Theorem 4. *The joint PDF of two quantization errors E_1 and E_2 is uniform, i.e.,*

$$f_{E_1\,E_2}(e_1,\,e_2) = \begin{cases} \dfrac{1}{\Delta^2}, & |e_1| < \dfrac{\Delta}{2},\ |e_2| < \dfrac{\Delta}{2}, \\ 0, & \textit{otherwise,} \end{cases} \tag{4.8}$$

if and only if the joint CF of the two input stochastic variables X_1 and X_2 satisfies

$$\Phi_{X_1\,X_2}\left(\frac{2\pi k}{\Delta},\,\frac{2\pi \ell}{\Delta}\right) = 0 \quad \textit{for all } k \neq 0 \textit{ and } \ell \neq 0. \tag{4.9}$$

The proof is outlined in [SS77]. An important implication follows from the results given so far. When (4.9) is satisfied, the covariance of the quantization noise process $e(n)$ is $\mathrm{E}\,[e(n)\,e(m)] = \delta_{m\,n}\Delta^2/12$ ($\delta_{m\,n}$ is the Kronecker δ-function). That is, the quantization error is indeed white under Theorem 4. Next, we will see how the input signal to the quantizer can be forced to fulfill some or all of the conditions in the preceding theorems.

Dithering Theory

We have now established conditions under which the quantization noise is uniform, zero mean, white and uncorrelated with the input. In this section we will see how the input to the quantizer can be manipulated so that certain conditions are met. The material presented here is to a large extent compiled from [WLVW00]. As before, we consider a uniform, scalar, mid-tread quantizer with quantization step size Δ (cf. Figure 4.2). The quantizer is assumed to be infinite, but the practical implication is that the input should be such that the quantizer never saturates. The theoretical results of dithering – both subtractive and non-subtractive – applied to such a quantizer are provided in the sequel.

The general framework for both cases is shown in Figure 4.1. The input to the *system* is an s.v. X to which a dither V is added forming the *quantizer* input W. In the non-subtractive case, the output $Y = \mathcal{Q}(W)$ from the quantizer is also the system output, while $Y = \mathcal{Q}(W) - V$ is the system output in the subtractive case. In both cases, the total error E is defined as

$$E = Y - X = \begin{cases} \mathcal{Q}(X+V) - X & \text{non-subtractive dithering;} \\ \mathcal{Q}(X+V) - (X+V) & \text{subtractive dithering.} \end{cases} \tag{4.10}$$

The two different topologies are now treated separately.

Subtractive Dither

It can be argued that subtractive dithering is more powerful in some sense than non-subtractive dithering. The main results of subtractive dithering are summarized in the following theorem [WLVW00]:

Theorem 5 (Subtractive dithering)**.** *The total error E induced by a subtractive dithering system can be made uniformly distributed in $[-\Delta/2,\, \Delta/2]$ for arbitrary input distributions if and only if the CF $\Phi_V(u)$ of the dither obeys*

$$\Phi_V\left(\frac{2\pi k}{\Delta}\right) = 0 \quad \textit{for all integers } k \neq 0. \tag{4.11}$$

Moreover, the total error E is statistically independent of the system input X if and only if the same condition (4.11) holds. Finally, two total error samples, E_1 and E_2 say, with a non-zero separation in time, will be statistically independent of each other if and only if the joint CF $\Phi_{V_1 V_2}(v_1,\, v_2)$ of the dither satisfies

$$\Phi_{V_1 V_2}\left(\frac{2\pi k_1}{\Delta},\, \frac{2\pi k_2}{\Delta}\right) = 0 \quad \textit{for all } (k_1,\, k_2) \in \mathbb{Z}_0^2. \tag{4.12}$$

The set $\mathbb{Z}_0^n$ is defined as all integer component vectors of length n with the exception of the vector of all zeros, i.e., $\mathbb{Z}_0^n = \mathbb{Z}^n \setminus \mathbf{0}_n$. The first part of the theorem (uniform distribution) was proven by Schuchman [Sch64], and the condition (4.11) is also referred to as Schuchman's condition. The second and third parts (independence and temporal characteristics) have been shown in various publications, e.g., [SS77]. Note that Schuchman's condition is satisfied if V is an s.v. with uniform density in $[-\Delta/2,\, \Delta/2]$, and (4.12) is satisfied for any independent identically distributed (i.i.d.) dither sequence satisfying Schuchman's condition. That is, selecting the dither to be white and uniform in $[-\Delta/2,\, \Delta/2]$ renders a total error that is white, uniform and statistically independent of the system input.

A final remark on subtractive dithering is that when the dithering is selected as stipulated in Therorem 5, the quantization noise power equals $\Delta^2/12$ and the PQN model is in fact valid. Thus, properly designed *subtractive* dithering can make a quantizer behave according to the PQN model for arbitrary (non-saturating) input signals.

Non-subtractive Dither

Subtractive dithering has obvious advantages, being able to give an overall quantization system that behaves according to the idealized PQN model. However, in many practical cases, it is impossible to subtract the same signal that was added prior to quantization, simply because the dither signal is not known in the digital domain. The following results – most of them due to Wannamaker, et al. [WLVW00], but relying on the quantization theories presented by Widrow – summarizes the properties of non-subtractive dithering.

The first result states the fundamental limitations of non-subtractive dithering:

Theorem 6. *In a non-subtractive dithering system it is not possible to make the total error E either statistically independent of the system input X or uniformly distributed for arbitrary system input distributions.*

That is, we can never expect to get such good results as with a perfect subtractively dithered system. Careful design of a non-subtractive dither can nevertheless improve a quantization system considerably. The following results tell how, and what results can be expected.

The next theorem on non-subtractive dithering concerns the dependence between the quantization error and the input signal [LWV92, WLVW00]:

Theorem 7. *The m-th moment of the total error, $\mathrm{E}[E^m]$ is independent of the distribution of the system input if and only if*

$$G_V^{(m)}\left(\frac{2\pi k}{\Delta}\right) = 0 \quad \textit{for all integers } k \neq 0. \tag{4.13}$$

Further, when (4.13) is fulfilled, the m-th moment is

$$\mathrm{E}[E^m] = (-j)^m G_V^{(m)}(0). \tag{4.14}$$

Here, the function $G_V(u)$ is

$$G_V(u) \triangleq \operatorname{sinc}\left(\frac{\Delta}{2\pi}u\right)\Phi_V(u), \tag{4.15}$$

the notation $^{(m)}$ denotes the m-th derivative, and

$$\operatorname{sinc}(x) \triangleq \frac{\sin(\pi x)}{\pi x}. \tag{4.16}$$

Since $\mathrm{E}[V^m] = (-j)^m \Phi_V^m(0)$, we can use (4.14) to express the moments of E in the moments of V; for instance,

$$\mathrm{E}[E] = \mathrm{E}[V] \tag{4.17}$$

and

$$\mathrm{E}[E^2] = \mathrm{E}[V^2] + \Delta^2/12 \tag{4.18}$$

when (4.13) holds. An immediate result of (4.18) is that using non-subtractive dither to make the quantization noise power independent of the input signal results in an increase in the quantization noise power, over that of a PQN model, by an amount equal to the dither variance. The quantization noise power can be made smaller, but that comes at the expense of making the noise power dependent of the input signal.

It can also be shown that when condition (4.13) in Theorem 7 is satisfied for some m, then E^m is uncorrelated with the input X. In fact, $\mathrm{E}[E^m \cdot X^n] = \mathrm{E}[E^m] \cdot \mathrm{E}[X^n]$ for any positive integer n. Finally, before moving on to the temporal properties of non-subtractive dithering, a stronger version of condition (4.13) is provided:

Corollary 8. *When using non-subtractive dithering, all moments* $\mathrm{E}[E^\ell]$ *for* $\ell = 1, 2, \ldots, m$ *are independent of the distribution of the system input if and only if*

$$\Phi_V^{(r)}\left(\frac{2\pi k}{\Delta}\right) = 0 \quad \textit{for all integers } k \neq 0 \textit{ and all } r = 0, 1, \ldots, m-1. \tag{4.19}$$

For the temporal properties of non-subtractive dithering, we give one theorem and an important corollary:

Theorem 9. *The joint moment* $\mathrm{E}[E_1^{m_1} E_2^{m_1}]$ *of two total errors* E_1 *and* E_2*, with a non-zero separation in time, is independent of the system input for arbitrary input distributions if and only if*

$$G_{V_1 V_2}^{(m_1, m_2)}\left(\frac{2\pi k_1}{\Delta}, \frac{2\pi k_2}{\Delta}\right) = 0 \quad \textit{for all } (k_1, k_2) \in \mathbb{Z}_0^2. \tag{4.20}$$

Here, the function $G_{V_1 V_2}(u_1, u_2)$ is defined as

$$G_{V_1 V_2}(u_1, u_2) = \operatorname{sinc}\left(\frac{\Delta}{2\pi}u_1\right) \operatorname{sinc}\left(\frac{\Delta}{2\pi}u_2\right) \Phi_{V_1 V_2}(u_1, u_2) \tag{4.21}$$

and $^{(m_1, m_2)}$ denotes differentiating m_1 and m_2 times with respect to u_1 and u_2, respectively. The proof is provided in [WLVW00]. Finally, an important implication of Theorem 9 is pointed out:

Corollary 10. *Any i.i.d. non-subtractive dither signal that satisfies (4.19) for* $m = \max(m_1, m_2)$ *will provide*

$$\mathrm{E}\left[E_1^{m_1} \cdot E_2^{m_2}\right] = \mathrm{E}\left[E_1^{m_1}\right] \cdot \mathrm{E}\left[E_2^{m_2}\right] \tag{4.22}$$

for two error values E_1 *and* E_2 *with a non-zero separation in time.*

When this holds true, the moments $\mathrm{E}\left[E_1^{m_1}\right]$ and $\mathrm{E}\left[E_2^{m_2}\right]$ are given by (4.14). From (4.17) we have that when Corollary 10 holds and the dither signal is zero-mean, the total error signal is uncorrelated in time, i.e., $\mathrm{E}\left[E_1 E_2\right] = 0$.

In summary, non-subtractive dithering can neither make the total error statistically independent of the system input, nor ensure that the distribution of the total error is uniform. However, clever design of the dither signal can make arbitrary powers of the error uncorrelated with the input signal, and also make the error signal temporally uncorrelated – in particular it can be made white.

Two Common Dither Distributions

Two very common dither distributions are rectangular and triangular distributions. In the following, we will briefly asses their merits in the light of the theory provided for subtractive and non-subtractive dithering.

Rectangular Distribution

A frequently proposed dither signal is an i.i.d. uniform noise signal with a 1 LSB range, that is producing samples uniformly in $[-\Delta/2, \Delta/2]$. This dither signal is zero-mean and has a variance $\Delta^2/12$.

As noted before, this dither satisfies all conditions of Theorem 5. Hence, a *subtractively* dithered system using such a rectangular noise totally obeys the PQN model.

In the *non-subtractive* case, we see that Theorem 7 is satisfied only for $m = 1$, implying that the mean error $\mathrm{E}[E] = \mathrm{E}[V] = 0$, while the noise power (and higher moments of E) varies with the input; despite our dithering effort, we still have so called *noise modulation*. From Corollary 10 we conclude that since (4.19) is satisfied for $m = 1$ and since the dither is i.i.d. the error signal is temporally uncorrelated; $\mathrm{E}[E_1 \cdot E_2] = \mathrm{E}[E_1] \cdot \mathrm{E}[E_2] = 0$.

Triangular Distribution

For subtractive dithering the rectangular noise turned out to be quite sufficient, but in the non-subtractive topology rectangular dither fell short in breaking the correlation between the input signal and the quantization noise power. Our knowledge from Theorem 7 tells us that we need a dither distributed such that (4.13) is fulfilled for $m = 1, 2$.

One such distribution is the symmetric triangular distribution ranging from $-\Delta$ to Δ:

$$f_V(v) = \begin{cases} \dfrac{\Delta - |v|}{\Delta^2} & |v| < \Delta, \\ 0 & \text{otherwise.} \end{cases} \tag{4.23}$$

Triangular dither can easily be generated by adding two independent zero-mean rectangular variables, each with variance $\Delta^2/12$. The mean of the triangular dither is zero, and the variance is $\Delta^2/6$. This distribution satisfies Theorem 7 for both $m = 1$ and $m = 2$ and also Corollary 8 for $m = 2$. Thus, $\mathrm{E}[E] = 0$ from (4.17) and $\mathrm{var}[E] = \mathrm{E}[E^2] = \Delta^2/4$ from (4.18). Again, if the individual dither samples are i.i.d. the error sequence is white.

4.2 Randomizing INL and DNL Patterns with Dithering

Until now, we have only discussed dithering methods aimed at mitigating the unwanted quantization effects of an ideal quantizer, mainly the deterministic nature of the quantization error giving rise to phenomenons such as noise modulation. But dithering can also be a powerful tool for diminishing the distortion induced by quantizer non-idealities.

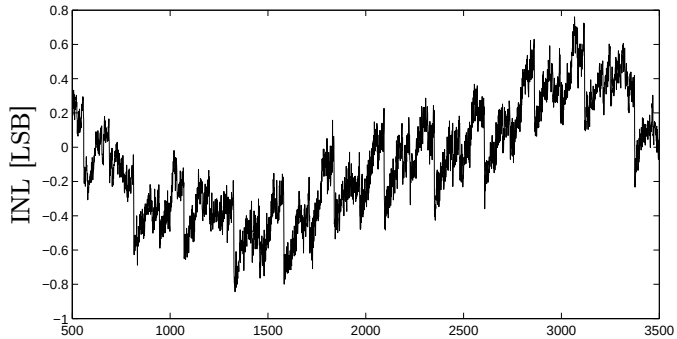
While the purpose of the dither investigated in Section 4.1 was to break the statistical dependence between the input and the quantization error, we will now see how dithering can randomize the non-linearity pattern of an ADC (or quantizer).

Consider the INL and DNL curves in Figure 4.4, showing respectively the deviation from an ideal transfer function and the actual bin widths' deviation from the ideal width (Δ), as a function of input value. A given input value x will always face the same INL and DNL – provided that these do not change – giving rise to a deterministic distortion in the output from the ADC. The deterministic distortion will manifest itself as unwanted spurs in the power spectral density of the output signal. However, if we add a dithering signal v to the input x prior to quantization, we might shift the input signal to a value that renders another INL and DNL. Hence, if the dither is independent of the input and large enough to shift the input signal to another quantization bin (at least occasionally), then the distortion inflicted for a certain input value will be different from time to time. The amplitude of the dither must typically be on the order of a few LSBs, rather than fractions of one LSB as in Section 4.1. The result is that the deterministic nature of the distortion is broken.

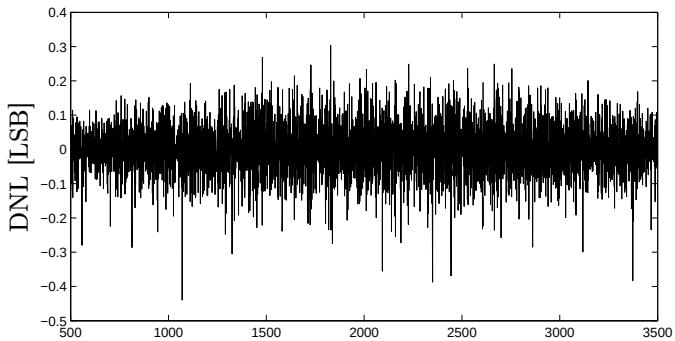
An early contribution in this direction was given by De Lotto and Paglia in [DLP86], where the authors show how dithering can smooth out the errors of the different quantization regions. A more recent publication – and also more relevant for contemporary ADCs – is [AN410], where the basic methods of dithering are explained. Also, exemplary results for an Analog Devices AD9042 with non-subtractive, large-scale (> 1 LSB), Gaussian dither is given. In the same reference the out-of-band dithering method is mentioned. In this particular method, the dither is band-pass filtered to fit into a band which is not occupied by the signal of interest. Two such bands are proposed, viz. near DC and near Nyquist. The experimental results using the AD9042 12-bit converter shows that the strongest unwanted spur is decreased from -81 dBFS to -103 dBFS using an out-of-band dither occupying the lowest 5% (near DC) of the Nyquist range. Meanwhile, the overall noise floor is increased by approximately 5 dB; the spurs are converted into noncoherent noise.

4.3 Increasing the Resolution for Slowly Varying Signals

The last dithering application considered here is dithering in combination with low-pass post-processing. Numerous aspects and versions of this has been treated in the literature, e.g., [AH98b, AH98a, Car97, CP00, SØ05]. The fundamental idea is the following. An ADC is sampling a slowly varying signal—the signal can be considered constant for a number of samples, N, say. The signal falls within a specific quantization region for all of these samples, with a resulting quantization error dependent on where within this region the signal is situated. Averaging the output samples will not reduce this error, because the samples are all the same. However, if a dither signal, with large enough amplitude, is added to the input prior to quantization, then the output will no longer be constant for all N samples, but will fluctuate around the previously constant output. Taking the mean of the N output samples now has a meaning, and might in fact yield a result with a higher



(a) INL



(b) DNL

Figure 4.4: INL and DNL from an Analog Devices AD9430.

resolution than that of the quantizer itself. We have thus traded bandwidth for resolution (since the averaging in effect is an LP filter). The dither signal must however possess certain properties for the dithering to be successful.

In [CP00], a uniform random dither is compared with a periodic, self-subtractive, deterministic dither. Self-subtractive means that the sum of the dither samples over one period is zero. The benefit of using this type of dither signal is that when an N-sample average is applied, where N is the period of the dither signal, the dither is automatically subtracted from the output, mimicking a subtractively dithered system. The periodicity of the dither does however give rise to some spurs in the output spectrum, but this can be mitigated using a random permutation of the samples within each period. The simulation results in [CP00] indicate that an increase of up to 3 effective bits can be obtained using dithering and averaging.

In [AH98b], different types of dither signals are investigated in conjunction

with averaging. Both deterministic and stochastic uniform dithering is considered, as well as Gaussian and mixed dither densities. For the uniform dithers it is stated that the number of effective bits can be increased to

$$\mathrm{ENOB}_{\mathrm{U\ determ}} = b + \log_2 N \tag{4.24}$$

in the case of deterministic uniform dither in the range $[-\Delta/2,\, \Delta/2]$, and

$$\mathrm{ENOB}_{\mathrm{U\ stoch}} = b + \frac{1}{2}\log_2 \frac{N}{1+k^2}, \tag{4.25}$$

when the dither is stochastic and uniform in $[-k\Delta/2,\, k\Delta/2]$. The number of samples averaged is N. Experimental results in [AH98b] indicates that the effective number of bits for a practical 12-bit converter can be increased to 16 bits using non-subtractive stochastic uniform dithering in $[-\Delta,\, \Delta]$ and $N = 16\,384$, and even further for subtractive dither.

4.A A Critical Note on Dithering and Quantization Noise

The purpose of this appendix is to refute a common misconception on dithering and correlation between the quantization error and the quantizer input or output. It is concluded that one possible source of confusion is an error in a frequently cited reference on quantization theory.

Problem Formulation

We consider a b-bit quantizer $\mathcal{Q}$ having the stochastic variable (s.v.) X as input and the s.v. Y as output. The quantization error is defined as

$$E = Y - X, \tag{4.26}$$

inherently also an s.v. Figure 4.3 on page 42 shows the model. The quantization is defined by a partition $\{\mathcal{S}_i\}$ and a set of reconstruction levels $\{\gamma_i\}$ as in Section 1.2. The partition consists of $M = 2^b$ disjunct regions $\mathcal{S}_0$ through $\mathcal{S}_{M-1}$, which together covers the entire input range (usually $\mathbb{R}$). A quantization region is defined by two neighboring transition levels as

$$\mathcal{S}_i = \{x :\ T_i \le x < T_{i+1}\}, \tag{4.27}$$

where $T_0 = -\infty$, $T_M = +\infty$ and $T_{i-1} < T_i$ for all i. Each quantization region $\mathcal{S}_i$ is associated with one reconstruction level γ_i which is assumed to be fixed but otherwise arbitrary. The quantization $\mathcal{Q}(\cdot)$ is defined such that $Y = \gamma_i$ if $X \in \mathcal{S}_i$.

Further, if we let the quantizer be uniform, we have that $T_i = T_{i-1} + \Delta$ for all i except the end points $i = 1$ and $i = M$. Here, Δ is the quantization step-size, or grain-size.

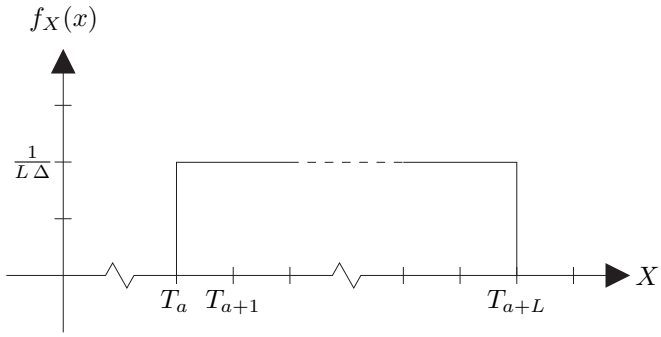


Figure 4.5: The PDF for the input X.

The problem is illustrated using a simple example. Let the quantizer be uniform. Assume that the input X has a probability density function (PDF)

$$f_X(x) = \begin{cases} \frac{1}{L\,\Delta} & T_a \le x < T_a + L\,\Delta, \text{ for some integer } a \in [0,\, M - L - 1]; \\ 0 & \text{otherwise.} \end{cases} \tag{4.28}$$

In other words, the variable s is uniform over exactly L consecutive quantization regions $\{\mathcal{S}_i\}_{i=a+1}^{a+L}$. Figure 4.5 illustrates the PDF. We define the set of indices for which the PDF is non-zero in the corresponding quantization regions as

$$\mathcal{I}_X = \{i \in \mathbb{Z} :\ f_X(x) > 0 \text{ for all } x \in \mathcal{S}_i\}. \tag{4.29}$$

We are now interested in investigating the statistical relationship, the correlation in particular, between the quantization error E and the other two variables. Two different approaches will be used, and these will yield two different results which in turn implies a paradox.

Lloyd's Resolution-Constrained Quantizer

We begin with the "source-coding approach" to the problem. Lloyd provided the optimal reconstruction levels $\{\gamma\}$ for an arbitrary quantizer partition and source PDF under a mean-squared distortion criterion. The i-th reconstruction level should according to [Llo82] (see also Chapter 6) be

$$\gamma_i = \mathrm{E}[X|X \in \mathcal{S}_i] = \frac{\int_{\mathcal{S}_i} x\, f_X(x)\, dx}{\int_{\mathcal{S}_i} f_X(x)\, dx} = \frac{T_i + T_{i-1}}{2}, \quad i \in \mathcal{I}_X, \tag{4.30}$$

where the last equality comes after inserting (4.27) and (4.28). For those levels that are outside the support of X, i.e., γ_i for $i \notin \mathcal{I}_X$, we can set any level we want, since they are never used. From (4.30) we can derive that

$$\int_{\mathcal{S}_i} x\, f_X(x)\, dx = \gamma_i \int_{\mathcal{S}_i} f_X(x)\, dx, \tag{4.31}$$

which will be used below.

We are now interested in calculating the correlation between the quantization error and the output. Straightforward calculations give

$$\begin{aligned}\mathrm{E}[E\,Y] = \mathrm{E}[(Y-X)\,Y] &= \int (y(x)-x)\,y(x)\,f_X(x)\,dx \\ &= \sum_i \int_{\mathcal{S}_i} (y(x)-x)\,y(x)\,f_X(x)\,dx = \sum_i \int_{\mathcal{S}_i} (\gamma_i - x)\,\gamma_i\,f_X(x)\,dx \\ &= \sum_i \gamma_i \left(\gamma_i \int_{\mathcal{S}_i} f_X(x)\,dx - \int_{\mathcal{S}_i} x\,f_X(x)\,dx\right) = 0.\end{aligned} \tag{4.32}$$

The fact that all quantization regions are disjoint was used to split the integration into the sum of integrals, and (4.31) was used in the last equality. The conclusion drawn is that *the output and the quantization noise are uncorrelated* when the reconstruction levels are selected as in (4.30).

Sripad and Snyder's Condition

A necessary and sufficient condition for the quantization error to be uniform and white was presented by Sripad and Snyder in [SS77], in the present work also given in Theorem 3. The result is valid for uniform "roundoff" quantizers and is in fact a milder version of the sufficient condition provided by Widrow (e.g., [WKL96] and also Theorems 1 and 2). A roundoff quantizer can be described in our framework above as one where the reconstruction levels are situated exactly in the middle of their corresponding quantization regions, i.e., $\gamma_i = (T_i + T_{i-1})/2$ which is exactly the same as (4.30) in our example (although not true in general with an arbitrary PDF $f_X(x)$). We refrain from dealing with the semi-infinite end regions by assuming the rather pragmatic standpoint that we do not care about them since our signal s will never occupy these regions.

The conditions in [SS77, WKL96] are based on the characteristic function of the source, defined in (4.3). In [SS77] it is shown that the quantization noise is uniform zero-mean with variance $\Delta^2/12$ if and only if

$$\phi_X\left(\frac{2\pi n}{\Delta}\right) = 0 \quad \text{for all integers } n \neq 0, \tag{4.33}$$

as stated in Theorem 3.

More interesting for us is the statement in [SS77, prop. 4, rem. 3] that under the same condition (4.33), the input and the quantization error are uncorrelated, that is $\mathrm{E}[EX] = 0$.

Returning to our case with the PDF (4.28), we can calculate the corresponding characteristic function to

$$\phi_s(u) = e^{ju\left(T_a + \frac{L\Delta}{2}\right)} \frac{\sin\left(u\frac{L\Delta}{2}\right)}{u\frac{L\Delta}{2}}, \tag{4.34}$$

from which it is easy to see that condition (4.33) is satisfied for this PDF. Hence, *the input and the quantization noise should be uncorrelated.*

The Paradox

Summarizing the results arrived upon above we know that $\mathrm{E}[E\,Y] = 0$ and $\mathrm{E}[E\,X] = 0$. It is also safe to say that the variance of the quantization error must be strictly positive for any non-zero quantization step-size Δ; it was in fact pointed out above that the variance is $\Delta^2/12$ under the present conditions. Also, the mean of the quantization error is zero. But then

$$\begin{aligned} 0 < \mathrm{var}[E] &= \mathrm{E}[(E - \mathrm{E}[E])^2] = \mathrm{E}[E^2] = \mathrm{E}[E\,(Y - X)] = \mathrm{E}[E\,Y] - \mathrm{E}[E\,X] \\ &= 0 - 0 = 0. \end{aligned} \tag{4.35}$$

This is truly a strange result.

The Explanation

The explanation to the phenomenon described above is quite simple: Sripad and Snyder's paper [SS77] contains an error.

It is derived in Equation (19) of [SS77] that, under condition (4.33) above, the correlation between the input and the quantization error is[2]

$$\mathrm{E}[XE] = -\frac{\Delta}{2\pi} \sum_{k \neq 0} \frac{(-1)^k}{k} \dot{\phi}_X \left(\frac{2\pi k}{\Delta} \right), \tag{4.36}$$

which is correct. (Here, $\dot{\phi}_X$ is the first derivative of $\phi_X(u)$ with respect to the transform variable u.) However, in the succeeding lines the faulty statement that condition (4.33) will guarantee $\mathrm{E}[XE] = 0$ is made—this is simply not true.

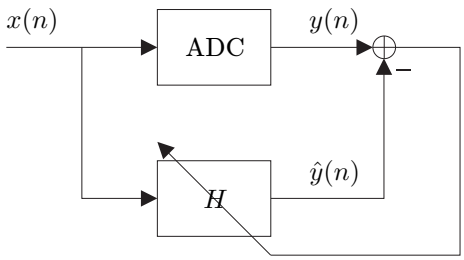
The correct conclusion in our example above is that the output and the quantization error are indeed uncorrelated, as shown above, while the correlation between input and quantization error equates to $\mathrm{E}[XE] = -\Delta^2/12$ from (4.36).

[2]The initial minus sign does not appear in the equation found in [SS77], simply because the error is defined as $E \triangleq X - Y$ in that paper.

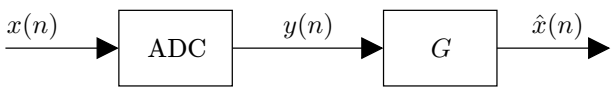
Chapter 5

Model Inversion Methods

ADC nonlinearity correction through model inversion has been proposed in the literature on several occasions (cf. [BDR05]). This family of correction schemes is based on some mathematical system model and its inverse. Typically, a model is identified for the ADC considered. The model gives an approximation of the input–output signal relationship. An inverse – possibly approximate – of the model is calculated thereafter. The model inverse is used in sequence after the ADC, hence operating on the output samples, in order to reduce or even cancel the unwanted distortion. Figure 5.1 shows the general concept of ADC correction with inverse models.



(a) Model identification



(b) Post-distortion

Figure 5.1: ADC correction using inverse system models. In (a) a model H for the ADC is found by identification. In (b) the inverse $G = H^{-1}$ of the identified model is used in sequence with the ADC, applying post-distortion to reduce the nonlinearities of the converter.

The vast majority of the references proposing a model inversion method are based on the Volterra model, but the use of other models have been reported, e.g., orthogonal polynomials and Wiener models in [Tsi95] and Chebyshev polynomials in [AAGS02]. In this chapter we will concentrate on the Volterra model.

5.1 Volterra Model

The causal discrete-time M-th order Volterra model (or filter) is an extension of the linear time-invariant discrete-time filter

$$y(n) = \sum_{k=0}^{\infty} h_k \, x(n-k), \tag{5.1}$$

where $x(n)$ is the input signal and $y(n)$ is the output from the filter. The discrete-time M-th order Volterra extension then is

$$y(n) = H_0 + H_1\left(x(n)\right) + \cdots + H_M\left(x(n)\right) \tag{5.2}$$

where

$$H_m\left(x(n)\right) = \sum_{k_1=0}^{\infty} \sum_{k_2=k_1}^{\infty} \cdots \sum_{k_m=k_{m-1}}^{\infty} h_m(k_1,\, k_2,\, \ldots,\, k_m) \times x(n-k_1)\, x(n-k_2) \cdots x(n-k_m). \tag{5.3}$$

The Volterra kernels h_m are assumed to be symmetric, hence the indexing in (5.3) where subsequent summation indices start at the index of the preceding sum. Finally, the kernels are truncated to finite length sums

$$H_m\left(x(n)\right) = \sum_{k_1=0}^{K_1} \sum_{k_2=k_1}^{K_2} \cdots \sum_{k_m=k_{m-1}}^{K_m} h_m(k_1,\, k_2,\, \ldots,\, k_m) \times x(n-k_1)\, x(n-k_2) \cdots x(n-k_m). \tag{5.4}$$

that can be implemented in practical applications.

Volterra models have long been used in general system modeling. One of the first contributions specifically targeted at modeling and correcting ADCs with Volterra models was made by Tsimbinos and Lever [TL93], in which they propose to use both Volterra and Wiener models for ADC modeling. They also show how to obtain the inverse of a fifth-order Volterra model, which is a nontrivial task. In their subsequent publications they also determine the computational complexity of Volterra based correction [TL96a] and make a comparison between a look-up table correction and a Volterra based correction [TL96b]. Gao and Sun [GS94] also made an early publication on ADC modeling using a Volterra model, although they did not explicitly propose any correction scheme based on the model.

5.2 Volterra Inverse

First we must ask ourselves what the inverse of a nonlinear system with memory, such as the Volterra model, actually is. The following definition is commonly used [Sch76, Tsi95]

Definition 1. *A p-th order inverse G of a given nonlinear system H is a system that when connected in tandem with H results in a system in which the first order Volterra kernel is the unity system and the second through p-th order Volterra kernels are zero. That is, if the H and G in tandem constitute a Volterra model denoted F, then*

$$F(x(n)) = x(n) + \sum_{m=p+1}^{\infty} F_m(x(n)). \tag{5.5}$$

In particular, we are interested in post-inverses, i.e., the inverse system G is to be used posterior to the original system H. However, as a matter of curiosity we note that it has been shown in [Sch76] that the p-th order post-inverse of a system H is in fact identical to the p-th order pre-inverse; only the Volterra operators of order higher than p of the tandem system are affected by which comes first of G and H.

Two different techniques can be used in finding the inverse Volterra model to be used as a correction:

1. A Volterra model H is identified for the ADC under test. From this model, an analytical inverse G is derived and used as corrector. Figure 5.1 shows this approach.

2. The inverse system G is identified directly as in Figure 5.2, minimizing a suitable function of the error signal, such as the mean squared error.

When finding the analytical inverse we are of course concerned about the stability of the inverse. It has been shown, again in [Sch76], that the p-th order inverse of H will be stable and causal if and only if the inverse of H_1 is stable and causal. That is, we only have to check that the linear part of the system does not have any zeros outside the unit circle.

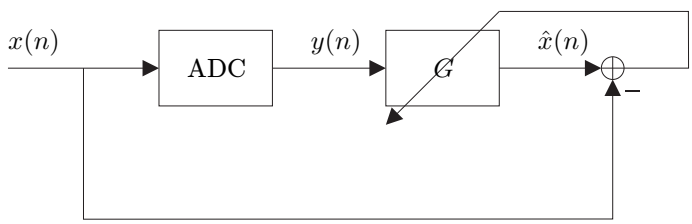


Figure 5.2: Direct identification of the inverse system from experiments.

The two different approaches above would typically yield slightly different results. The computational burden of the two methods when engaged in correction differ significantly. The second approach – sometimes referred to as adaptive Volterra inverse – is far more computationally heavy than the analytical inverse [TL96a]. The difference stems from the fact that the adaptively identified inverse generates a more general inverse system while the analytical inverse gains from the structure given by the original model H.

One of the key features of the Volterra model correction method is its ability to capture highly dynamic error effects at a moderate cost. A look-up table can hardly be used in practice to successfully model errors that depend on more than a few subsequent input samples—the size (memory requirements) of the table grows exponentially in the number of samples K used for addressing. Meanwhile, a Volterra model, or an FIR-filter in particular, can easily use tens of input samples in the calculation of the correction values, at a moderate computational cost. Quite opposite, the computational complexity of the Volterra model rapidly becomes too heavy when the nonlinearity order M is increased (see [TL96a]) while a look-up table has a very low and constant computational cost for any nonlinearity order.

Another issue for Volterra models is the identification process, both when the system model H is identified first, and if the inverse G is identified directly. Imposing further structure on the Volterra model can ease the burden of identification. This was proposed in [MŠ02], where the Volterra model was compared with an error model for an integrating ADC. The coefficients of the Volterra model that are most important in modeling such a converter could thus be isolated, and the identification process was significantly simplified.

Part II

Signal Processing in ADC Post-correction

Chapter 6

Distortion Criteria and Optimal Correction

Before embarking on correcting an ADC, we must decide upon a measure of error or distortion. In other words, we must have a mechanism to judge what is good and what is not in terms of ADC performance. Our choice of distortion criterion will affect what correction values to use. These questions are within the topic of this chapter.

6.1 Distortion Criteria

A distortion criterion, or distortion measure, is generally speaking a function that given the input s and output x to a system calculates the distortion inflicted by the system. See Figure 6.1. Most common distortion measures are based on the distance between the output and the input, and are therefore denoted distance measures. That is, the distortion is a function of the absolute difference between output and input, and not of the input or output themselves. In the multi-dimensional case this corresponds to the distortion being a function of the norm (length) of the difference vector. Two commonly used distance measures are the absolute error

$$d(s,\, x) = |x - s| \qquad d(\mathbf{s},\, \mathbf{x}) = \|\mathbf{x} - \mathbf{s}\|\,, \tag{6.1}$$

and the squared error

$$d(s,\, x) = (x - s)^2 \qquad d(\mathbf{s},\, \mathbf{x}) = \|\mathbf{x} - \mathbf{s}\|^2\,, \tag{6.2}$$

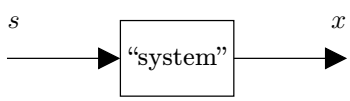


Figure 6.1: A system with input s and output x.

where both the scalar and the vector notations are shown in each case.

Mean-Squared Error

The squared error criterion is by far the most popular, mainly because it (a) represents the energy in the error signal, and (b) usually yields tractable mathematical expressions and solutions.

In many cases it is more interesting to consider the expected distortion in a statistical sense. The mean-squared error (MSE) is then defined as

$$\mathrm{MSE}(X - S) = \mathrm{E}\left[(X - S)^2\right] \tag{6.3}$$

where S and X are stochastic variables. In most cases the output X is a function of the input S, so that the expected value is taken with respect to S only.

Mean-Absolute Error

Analogously with the mean-square error above, the mean-absolute error is defined as

$$\mathrm{MAE}(X - S) = \mathrm{E}\left[|X - S|\right]. \tag{6.4}$$

Again, S and X are stochastic variables representing input and output, respectively.

ADC Specific Criteria

There are many specific measures for describing the performance of an ADC – SINAD, SFDR, ENOB, THD, NPR and IMD, to mention a few (see Section 1.4, Section 1.8 and [Std1241]). Since these measures are used to assess the precision and quality of A/D converters, it would be natural to use them as optimality criteria when designing post-correction systems. However, most of the specialized measures result in fairly complicated expressions that do not provide results of practical use. Exceptions are SINAD and ENOB which are both closely related to the MSE. Therefore, most results on MSE can be transferred to results on SINAD and ENOB.

6.2 MMSE Optimal Correction Values

In Chapter 1 a quantizer – or the quantization operation – was represented by a two-step operation, as in Figure 1.1 on page 3. The design of such a quantizer incorporates both defining the quantization regions $\{\mathcal{S}_j\}_{j=0}^{M-1}$ and assigning suitable values $\{x_j\}_{j=0}^{M-1}$ to represent each level with – the reconstruction points. Optimal reconstruction points[1] for minimizing the mean-square error $\mathrm{E}\left[(X - S)^2\right]$ have been derived in [Llo82]. Since X is a function of S, $\mathrm{E}[\cdot]$ is the expected value

[1] In [Llo82] the reconstruction values are denoted 'quanta'.

operator taken with respect to S. The key results in [Llo82] and their implications are briefly reviewed here.

The scheme used here is in fact half of the popular Lloyd two-step algorithm for optimal quantizer design; see for instance [GG92]. This algorithm iteratively trains the transition levels and the reconstruction points until a (local) optimum in terms of MSE has been reached. In the case of ADC correction, the transition levels are not possible to change, wherefore only the reconstruction points are optimized. The procedure is single-step and iteration is not needed in this case.

In [Llo82] a quantizer operating on a value s is considered; the quantization region model of Section 1.2 is used. The value s is regarded to be drawn from a stochastic variable S with a probability density function (PDF) $f_S(s)$. If the quantization regions $\{\mathcal{S}_j\}$ are assumed fixed, then it is proved that the optimal reconstruction values $\{x_j\}$, in the mean-squared sense, are given by

$$x_{j,\,\mathrm{opt}} = \arg\min_x \mathrm{E}\left[(x-S)^2 \middle| S \in \mathcal{S}_j\right] = \frac{\int_{s\in\mathcal{S}_j} s\, f_S(s)\, ds}{\int_{s\in\mathcal{S}_j} f_S(s)\, ds}, \tag{6.5}$$

i.e., the optimal reconstruction value for each region is the "center of mass" of the region.

The optimal *additive correction values* are found directly from (6.5) as

$$e_j = x_{j,\,\mathrm{opt}} - x_j, \tag{6.6}$$

where x_j is the output originally produced by the quantizer.

6.3 Minimum Mean-Absolute Error Correction Values

As mentioned above, the mean-absolute error is closely related to the MSE. However, they do yield different reconstruction values when applied to the quantizer design problem. As above, the input s is regarded to be drawn from a stochastic variable S with a probability density function (PDF) $f_S(s)$. We consider the quantization region $\mathcal{S}_j$ with lower and upper boundaries T_j and T_{j+1}, respectively. The optimal reconstruction point is defined by

$$x_{j,\,\mathrm{opt}} = \arg\min_x \mathrm{E}\left[|x-S| \,\middle|\, S \in \mathcal{S}_j\right] = \arg\min_x \int_{T_j}^{T_{j+1}} |x-s|\, f_S(s)\, ds. \tag{6.7}$$

Differentiating the integral in (6.7) with respect to x and equating the result to zero, yields that the optimal reconstruction point $x_{j,\,\mathrm{opt}}$ should satisfy

$$\int_{T_j}^{x_{j,\,\mathrm{opt}}} f_S(s)\, ds = \int_{x_{j,\,\mathrm{opt}}}^{T_{j+1}} f_S(s)\, ds. \tag{6.8}$$

That is, the optimal reconstruction point $x_{j,\,\mathrm{opt}}$ for the j-th quantization region in the mean-absolute error sense is the *median* of the input S given that $S \in \mathcal{S}_j$. Just as in Section 6.2, the optimal additive corrections are found by applying (6.6).

6.4 Other Correction Strategies

Although the MMSE correction strategy is very commonly used, especially when considering ADC correction from a signal processing point of view, other correction strategies can be considered. The most popular alternative is midpoint correction described here.

Midpoint Correction

From an ADC characterization point of view one might want to consider the reconstruction levels to be an inherent parameter of the ADC under test, and not of the input signal as was the case in the MMSE and MMAE strategies above.

The midpoint correction strategy is based on the assumption that the ADC acts as a staircase quantizer. That is, the quantizer part acts as described in Section 1.2 with quantization regions $\{\mathcal{S}_j\}_{j=0}^{M-1}$. The midpoint correction strategy is based on the intuitively pleasing assumption that the reconstruction value associated with a specific quantization region should be the midpoint of that region. That is, if the i-th region is delimited below and above by T_i and T_{i+1}, respectively, then the ideal x_i should be the midpoint in between, i.e., $(T_i + T_{i+1})/2$. If the quantization regions should deviate from the ideal ones, then the output values should be changed accordingly. Thus, in the case of additive midpoint correction the correction term should nominally be

$$e_i = \frac{T_i + T_{i+1}}{2} - x_i. \tag{6.9}$$

It is easy to see that the midpoint approach in fact is consistent with Lloyd's MMSE approach (6.5) *and* with the mimimum mean-absolute error approach (6.8) above if the variable S is assumed to be symmetric within each quantization region. Two such signals are the uniform noise and the deterministic ramp, which provide symmetric PDFs within each quantization region, save the regions at the extremes of the signal range where the signal may occupy only part of the region.

Minimum Harmonic Correction

Hummels, et al. [HICP94, Hum02] have provided a method where the correction values are selected such that the harmonic distortion generated by the ADC is minimized. The method uses single sinewaves for calibration and the correction tables are built using error basis functions, usually two-dimensional Gaussian basis functions in a phase-plane indexing scheme. The basis function coefficients are selected using minimization of the power in the H first harmonics to the test frequency.

6.5 Remarks

The two choices of reconstruction values presented in Sections 6.2–6.3 above are evidently dependent not only on the characteristics of the ADC under test, but

also on the test signal itself (through the PDF of the signal). Thus, we take into account what we know about the signal as well. It is therefore of vital importance that the calibration routine is carefully designed. It is indeed very easy to end up with a correction system that is heavily biased towards a specific signal, since the correction values were trained using that signal type. On the other hand, if prior knowledge says that the ADC will be used to convert signals of a specific class, it makes good sense to calibrate the post-correction system using the same class of signals. Using calibration signals with a uniform PDF can be considered to yield unbiased calibration results. We also saw that in this case, the MMSE, the MMAE and the midpoint corrections all coincided.

In this chapter we have presented and discussed a few distortion criteria and optimal correction values. We have not, however, touched upon how to obtain these correction values in practice. Applying the formulae directly is in most cases impossible since our knowledge is sufficient neither of the actual quantization regions nor of the PDF of the test signal. This matter will be dealt with in depth in Chapter 7.

Chapter 7

Characterization of Quantizer Transfer Functions

Within the art of post-correction lies an estimation problem, namely how to find the best correction values for a specific ADC. In this chapter we will analyze this estimation problem. The estimation is assisted by some reference signal, as was discussed in Section 3.4. A probabilistic ADC model is used. The model was introduced in the pioneering work of Giaquinto et al. [GST96b], in which it was used to find the optimal correction terms in relation to the model.

The estimation problem posed here is targeted towards finding suitable correction values for an ADC post-correction system. Therefore, we will consider the problem of estimating the optimal reconstruction points rather than finding the transfer function in terms of identifying the transition levels, which is often the case in ADC characterization.

Two different scenarios are investigated in this chapter, and estimators are derived for them both. Also, a fundamental lower limit on the variance – the Cramér–Rao bound – is calculated and compared with the performance of the derived estimators. In particular, we show that in scenarios with a staircase transfer function and with an accurate reference signal available, a simple estimator taking the mean of the smallest and the largest sample significantly outperforms the traditionally used sample mean.

7.1 Prior Art

This chapter deals with calibration of an ADC post-correction system from a reference signal. Traditionally, the reconstruction point for the k-th quantization level is estimated as the sample mean of the reference samples that produced the k-th output, as described for instance in [HSP00].

This chapter also presents variance expressions for the reconstruction point estimation, and compares the variances with the Cramér–Rao lower bound. Some work

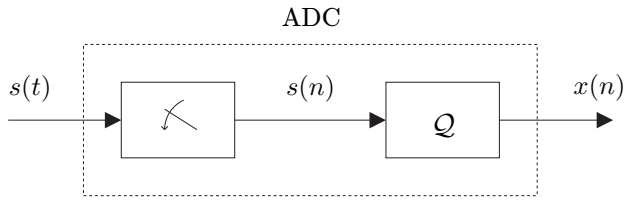


Figure 7.1: A model of an ADC. The sample-and-hold is assumed to be ideal while the quantizer $\mathcal{Q}$ can introduce errors.

in this direction has been presented in [CNP02], but only for the case of sinewave histogram tests. Also, in [MCP03] and [BH04] similar results were derived for the Gaussian and truncated Gaussian histogram tests, respectively.

7.2 Probabilistic Quantizer Model

In this chapter we model the ADC to consist of an ideal sample-and-hold circuit followed by a non-ideal but static quantizer, see Figure 7.1. We disregard from the sample-and-hold in the forthcoming analysis, and consider only the quantizer with discrete-time input and output signals. The quantizer is assumed to be b-bit, hence having $M = 2^b$ quantization levels and M possible output values $\{x_k\}_{k=0}^{M-1}$. Thus far, the model is identical to that of Chapter 1 (Figure 1.1).

The difference from the model in Chapter 1 lies in how the quantizer part $\mathcal{Q}$ is modeled in this chapter. Here, the probabilistic model presented in [GST96b] is adopted. The model for $\mathcal{Q}$, henceforth denoted $\mathcal{M}$, has a probabilistic transfer function where the input and output of the quantizer are modeled as random variables S and X, respectively. The two variables are linked through the probabilistic quantizer, resulting in a joint probability distribution $f_{S,\,X}(s,\,x)$. The probability distribution captures the stochastics of both the input S and of the quantizer $\mathcal{Q}$ itself. The transfer function is defined by the conditional probability mass function (PMF) $f_{X|S}(x_k|s) = \Pr\{X = x_k|S = s\}$. Note that the model of Chapter 1, where the behavior is governed by the quantization regions $\{\mathcal{S}_j\}$, is in fact a special case of the probabilistic model; simply let the probability distribution $f_{S,\,X}(s,\,x_k) = 0$ for $s \notin \mathcal{S}_k$.

7.3 Optimal Reconstruction Levels

In Chapter 6 the optimal reconstruction points were calculated in the case of mean-squared error and mean-absolute error. In particular, we saw that when the quantization regions $\{\mathcal{S}_j\}$ are assumed fixed, then the optimal reconstruction values $\{\gamma_j\}$,

in the mean-squared sense, are given by

$$\gamma_k = \arg\min_{\gamma} \mathrm{E}[(\gamma - S)^2 | S \in \mathcal{S}_k] = \frac{\int_{s\in\mathcal{S}_k} s\, f_S(s)\, ds}{\int_{s\in\mathcal{S}_k} f_S(s)\, ds}. \tag{7.1}$$

The extension of this result to the non-deterministic transfer function of model $\mathcal{M}$ is that the optimal estimator is given by the conditional expectation of S given x_k, that is [GST96b]

$$\gamma_k = \int_{-\infty}^{+\infty} s\, f_{S|X}(s|x_k)\, ds = \int_{-\infty}^{+\infty} s\, \frac{f_{X|S}(x_k|s)\, f_S(s)}{p_X(x_k)}\, ds, \tag{7.2}$$

where $p_X(x_k)$ is the probability mass function for the output random variable X and the last equality is obtained using Bayes' rule. The problem of finding the optimal reconstruction levels for the probabilistic ADC model used here has a strong connection to the problem of quantizer design for noisy channels, e.g., [KW69].

In this section we have formulated the exact expressions (7.1) and (7.2) for the optimal reconstruction levels, under certain model assumptions. However, the first expression requires that we know the PDF $f_S(s)$ of the input signal and the limits of the quantization region $\mathcal{S}_k$, and the second expression depends on exact knowledge of the transfer statistics $f_{X|S}(x_k|s)$ together with $f_S(s)$. Neither of these cases are likely when characterizing an ADC. On the contrary, finding the unknown transfer function is the goal of the characterization procedure, wherefore we must find a way to estimate the reconstruction levels from a finite number of measurements. This is the objective of the next section.

7.4 Estimating the Reconstruction Levels

In this section the problem of estimating the optimal reconstruction levels of a quantizer from measurements is considered. Assume that a signal $s(n)$ is connected to the input of the quantizer under test and that N_{tot} output samples $x(n)$, $n = 1, 2, \ldots, N_{\mathrm{tot}}$ are recorded. The input samples are for now assumed to be independent and identically distributed realizations of an s.v. with PDF $f_S(s)$. It is further assumed that a positive number $N_k \leq N_{\mathrm{tot}}$ of samples result in the specific ADC output $x(n) = x_k$ (the k-th output value). As a consequence, $\sum_{k=0}^{M-1} N_k = N_{\mathrm{tot}}$ – typically $N_k \ll N_{\mathrm{tot}}$.

An estimate of the input signal $s(n)$ is also obtained. The estimate is denoted $z(n)$ and is a perturbed version of the true input signal. The reference signal $z(n)$ can for example be an estimate based on the measured signal $x(n)$, $1 \leq n \leq N_{\mathrm{tot}}$, or a measurement from a reference device. The reference signal is modeled as the true input $s(n)$ with an additive perturbation $u(n)$. The perturbation can then be used to account for measurement errors, (reference device) quantization errors, reconstruction errors or estimation errors, as appropriate.

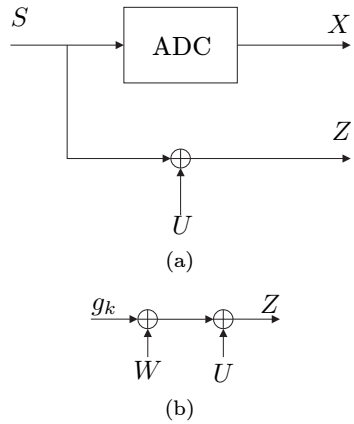


Figure 7.2: The probabilistic ADC setup. Figure (a) shows the original ADC characterization problem with an input stimuli S, the resulting output X and a reference signal Z. Figure (b) shows the equivalent estimation problem setup of the type "DC level in noise."

A record of N_{tot} samples of $z(n)$ is taken. The subset of N_k samples for which the output $x(n) = x_k$ is of special interest. Comparing with the sets defined in Section 3.4, these samples form the set denoted $\mathcal{C}_k$. The samples are collected in a column vector $\mathbf{z}^{\{k\}} = \left[z_1^{\{k\}}\ z_2^{\{k\}}\ \ldots\ z_{N_k}^{\{k\}}\right]^{\mathrm{T}}$ of length N_k, where the notation $^{\{k\}}$ denotes that the sample corresponds to an instant when the ADC produced x_k as output. In the same manner we denote the corresponding input samples in the column vector $\mathbf{s}^{\{k\}} = \left[s_1^{\{k\}}\ s_2^{\{k\}}\ \ldots\ s_{N_k}^{\{k\}}\right]^{\mathrm{T}}$. That is, when the sample $s_n^{\{k\}}$ was input, the the reference signal was estimated to be $z_n^{\{k\}}$. Note that $\mathbf{s}^{\{k\}}$ is unknown in the estimation problem.

The input to the quantizer is drawn from an s.v. S, as before (cf. Chapter 6). Assume further that the perturbation samples are independent realizations of an s.v. U with PDF $f_U(u)$, then the reference estimate can be modeled as an s.v. $Z = S + U$. Figure 7.2(a) illustrates the signal relationship. The reference is a sum of two independent stochastic variables. The resulting PDF of a sum of two independent variables is the convolution of the PDFs of the two terms. Thus, the PDF of Z is

$$f_Z(z) = (f_S * f_U)(z) = \int_{-\infty}^{+\infty} f_S(z-\zeta)\, f_U(\zeta)\, d\zeta \tag{7.3}$$

if independence between S and U is assumed.

In the sequel the problem of estimating the optimal reconstruction level γ_k given the observation $\mathbf{z}^{\{k\}}$ is considered.

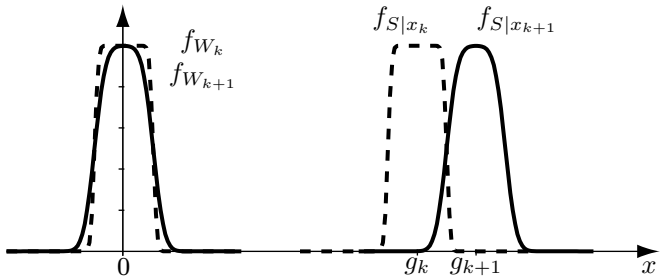


Figure 7.3: Illustration of the relocation of PDFs around the origin.

Reformulating the estimation problem

Since we only consider the sample instances where the quantizer output was a specific x_k, the PDF for our observation of Z can be formulated as a conditional PDF $f_{Z|X}(z|x_k)$. The variable Z is still the sum of S and U, but now we can describe S using the conditional statistics $f_{S|X}(s|x_k)$. Thus, we have

$$f_{Z|X}(z|x_k) = (f_{S|X} * f_U)(z) = \int_{-\infty}^{+\infty} f_{S|X}(z-\zeta|x_k)\, f_U(\zeta)\, d\zeta. \tag{7.4}$$

Assume that the conditional PDF $f_{Z|X}(z|x_k)$ is symmetric around a location parameter g_k. That is, $f_{Z|X}(z|x_k)$ depends on g_k only as a shift with g_k along the z-axis.

Define the stochastic variable

$$W_k = S - g_k, \tag{7.5}$$

which then of course has the conditional PDF

$$f_{W_k|X}(w|x_k) = f_{S|X}(w+g_k|x_k). \tag{7.6}$$

That is, the location parameter of $f_{S|X}$ is removed, repositioning the PDF at the origin. Figure 7.3 illustrates this relationship. The s.v. W_k is thus independent of g_k. With these premises it is easy to see from (7.2) that $\gamma_k = g_k$. Thus, estimating the optimal reconstruction levels is in this case equivalent to estimating the location parameter of $f_{Z|X}(z|x_k)$. We can now reformulate our observation Z as a constant (DC) level g_k in two additive noises W_k and U,

$$Z = g_k + W_k + U, \quad \text{when } x(n) = x_k. \tag{7.7}$$

Figure 7.2(b) depicts the estimation problem setup.

The problem class now arrived to – estimation of a constant (DC-level) in additive noise – is a classical estimation problem, found in the literature (e.g., [Kay93]).

The optimal solution to the problem differs depending on (a) the optimality criterion and (b) the probability distribution of the noise.

Two different assumptions for the distribution of W_k will be considered. In the first case, W_k is assumed to be zero-mean Gaussian with variance σ_k^2, and in the second case a zero-mean uniform distribution with a width Δ_k (variance $\Delta_k^2/12$) is assumed for W_k. In both cases U is assumed to be zero-mean Gaussian with variance σ_U^2. Since we are now only considering one specific quantization level, namely the k-th level, both the index k and the explicit conditioning of the PDFs will be omitted for brevity. Thus, g, σ^2, Δ, $\mathbf{z}$ and z_i will in the sequel represent g_k, σ_k^2, Δ_k, $\mathbf{z}^{\{k\}}$ and $z_i^{\{k\}}$, respectively.

Gaussian W and U

The first scenario, motivated by its simplicity, is where $W \in \mathcal{N}(0,\, \sigma^2)$ and $U \in \mathcal{N}(0,\, \sigma_U^2)$. Hence, the observed variable $Z = g + W + U$ is also Gaussian with PDF

$$f_Z(z) = \frac{1}{\sqrt{2\pi(\sigma^2 + \sigma_U^2)}} \exp\left(-\frac{(z-g)^2}{2(\sigma^2+\sigma_U^2)}\right). \tag{7.8}$$

We know from the literature (e.g., [Kay93]) that the sample mean is both the minimum variance unbiased (MVU) estimator *and* the maximum likelihood estimator (MLE) of g in this case. That is,

$$\hat{g}_{\mathrm{sm}}(\mathbf{z}) = \frac{1}{N}\sum_{i=1}^{N} z_i \tag{7.9}$$

is the estimator with the lowest variance and zero bias *and* the estimator that minimizes the likelihood function $p(\mathbf{z};\, g) = \prod_{i=1}^{N} f_Z(z_i)$ with respect to g (the index 'sm' denotes sample mean). Moreover, this is the same estimator as (3.10) devised in Section 3.4. The variance is also straightforward to calculate:

$$\mathrm{var}\left[\hat{g}_{\mathrm{sm}}(\mathbf{z})\right] = \frac{\sigma^2 + \sigma_U^2}{N}. \tag{7.10}$$

In fact, this is the variance for the sample mean of N i.i.d. realizations of any two independent variables with variances σ^2 and σ_U^2, regardless of the distributions.

In this case, the total variance $\bar{\sigma}^2 = \sigma^2 + \sigma_U^2$ can be estimated from the recorded data, but not the individual variances σ^2 and σ_U^2. From literature (e.g., [Kay93]) we find that the MVU estimator for $\bar{\sigma}^2$ is

$$\widehat{\bar{\sigma}^2}_{\mathrm{MVU}} = \frac{1}{N-1}\sum_{i=1}^{N}\left(z_i - \hat{g}_{\mathrm{sm}}(\mathbf{z})\right)^2, \tag{7.11}$$

with estimator variance $2\bar{\sigma}^4/(N-1)$. Meanwhile, the MLE for the same case is

$$\widehat{\bar{\sigma}^2}_{\mathrm{MLE}} = \frac{1}{N}\sum_{i=1}^{N}\left(z_i - \hat{g}_{\mathrm{sm}}(\mathbf{z})\right)^2, \tag{7.12}$$

with a resulting estimator variance of $2(N-1)\bar{\sigma}^4/N^2$, The MLE has evidently less variance than the MVU, but at the expense of non-zero bias.

Further, one may note that superefficient estimators do exist for estimating $\bar{\sigma}^2$ [Bey98, SO96]. (A superefficient estimator possesses a mean-square error lower than the Cramér–Rao bound (cf. Section 7.5) at the expense of non-zero bias.)

Uniform W and Gaussian U

The second case results from modifying the model $\mathcal{M}$ in Section 7.2 to coincide with the classic staircase transfer function and using an input signal whose PDF $f_S(s)$ is uniform within each step of the transfer function[1] (e.g., a ramp signal or a uniform noise, but also when the step size is small compared with the variability of the PDF $f_S(s)$). In this case, the distribution of the input S given the output X ($f_{S|X}$) is uniform with some width Δ (possibly dependent on k, again omitted for brevity). Therefore, the PDF for W in the equivalent model of Figure 7.2(b) is in this case

$$f_W(w) = \begin{cases} \dfrac{1}{\Delta} & |w| \le \dfrac{\Delta}{2}, \\ 0 & \text{otherwise.} \end{cases} \tag{7.13}$$

The PDF for U is still assumed to be zero-mean Gaussian with variance σ_U^2, so the resulting PDF for $W+U$ becomes

$$f_{W+U}(v) = \int_{-\frac{\Delta}{2}}^{\frac{\Delta}{2}} \frac{1}{\Delta\sqrt{2\pi\sigma_U^2}} \exp\left(-\frac{(v-\tau)^2}{2\sigma_U^2}\right) d\tau. \tag{7.14}$$

Define the stochastic variable $V \triangleq (W+U)/\Delta$. Straightforward derivations from (7.14) give that the PDF for V becomes

$$\begin{aligned} f_V(v) &= \int_{-\frac{1}{2}}^{\frac{1}{2}} \frac{1}{\sqrt{2\pi}\frac{\sigma_U}{\Delta}} \exp\left(-\frac{(v-\tau)^2}{2\left(\frac{\sigma_U}{\Delta}\right)^2}\right) d\tau \\ &= \frac{1}{2}\operatorname{erf}\left(\frac{v+\frac{1}{2}}{\sqrt{2}\rho}\right) - \frac{1}{2}\operatorname{erf}\left(\frac{v-\frac{1}{2}}{\sqrt{2}\rho}\right), \end{aligned} \tag{7.15}$$

where

$$\rho \triangleq \frac{\sigma_U}{\Delta}. \tag{7.16}$$

We see from (7.15) that V depends on Δ and σ_U only through ρ, which can be interpreted as a shape parameter. See Figure 7.4. When $\rho \to 0$ the distribution for V approaches a uniform distribution with unit width, while for large ρ, V approaches a Gaussian distribution with the variance ρ^2. In fact, it has been claimed [BNPP00] that when ρ is approximately larger than 0.35 the distribution for V can be considered Gaussian, a claim that is further investigated at the end of this section.

[1] This case can also be obtained by using the traditional deterministic staircase transfer function instead of $\mathcal{M}$.

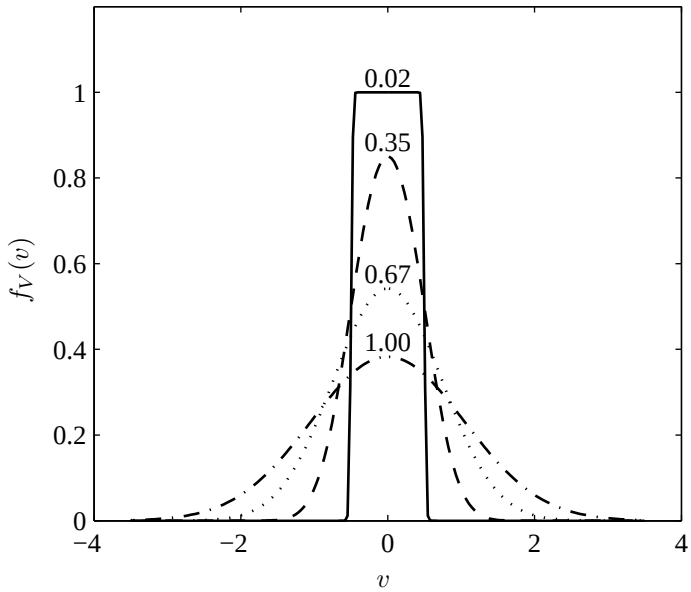


Figure 7.4: The distribution for V for four different values of ρ.

So far, we have reformulated the estimation problem as follows: estimate g from N independent observations of $Z = \Delta V + g$, where the PDF of V depends on the parameter ρ. This setting was considered in [Llo52] where an estimator for g was derived using order statistics, i.e., an estimator working on a batch of samples arranged by their magnitude. The theory and the resulting estimator are described in the sequel.

Order Statistics Estimator Theory

Lloyd[2] considered the use of order statistics for estimation of location and scale parameters in [Llo52]. The setting is as follows.

A stochastic variable X has location and scale parameters g and Δ, respectively. These are not necessarily the mean and standard deviation of X. Consider N independent identically distributed variables $X_1, \ldots, X_N$, from which our observation is taken. The observations are arranged in ascending order. The ordered stochastic variables are denoted $X_{(1)} \leq \cdots \leq X_{(N)}$. The estimator sought for should be linear in the ordered observation $\mathbf{y} = \begin{bmatrix} x_{(1)} & \cdots & x_{(N)} \end{bmatrix}^{\mathrm{T}}$. Now, introduce the

[2]N.B., this is E. H. Lloyd, not to be confused with Stuart P. Lloyd who has authored numerous papers on quantization, e.g., [Llo82].

standardized ordered stochastic variables $V_{(1)} \leq \cdots \leq V_{(N)}$, where

$$V_{(i)} = \frac{X_{(i)} - g}{\Delta}. \tag{7.17}$$

Let $\mathbf{m}_{(V)} = [\mathrm{E}[V_{(1)}] \ \ldots \ \mathrm{E}[V_{(N)}]]^{\mathrm{T}}$ be the vector of expected values and $\mathbf{R}_{(V)}$ be the covariance matrix of the ordered stochastic variables $V_{(1)}, \ldots, V_{(N)}$. That is

$$\mathbf{R}_{(V)} = \begin{bmatrix} \mathrm{E}[V_{(1)}^2] & \mathrm{E}[V_{(1)}V_{(2)}] & \cdots & \mathrm{E}[V_{(1)}V_{(N)}] \\ \mathrm{E}[V_{(2)}V_{(1)}] & \mathrm{E}[V_{(2)}^2] & \cdots & \mathrm{E}[V_{(2)}V_{(N)}] \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{E}[V_{(N)}V_{(1)}] & \mathrm{E}[V_{(N)}V_{(2)}] & \cdots & \mathrm{E}[V_{(N)}^2] \end{bmatrix}. \tag{7.18}$$

It is shown in [Llo52], that when the PDF of X is symmetric, the least-squares estimator for the location parameter becomes

$$\hat{g} = \frac{\mathbf{1}^{\mathrm{T}}\mathbf{R}_{(V)}^{-1}\mathbf{y}}{\mathbf{1}^{\mathrm{T}}\mathbf{R}_{(V)}^{-1}\mathbf{1}}, \tag{7.19}$$

where $\mathbf{1}$ is the column vector of all ones. The variance of the estimator is shown to be

$$\mathrm{var}[\hat{g}] = \frac{\Delta^2}{\mathbf{1}^{\mathrm{T}}\mathbf{R}_{(V)}^{-1}\mathbf{1}}. \tag{7.20}$$

It is further proven that the least-squares estimator for the scale parameter Δ is

$$\hat{\Delta} = \frac{\mathbf{m}_{(V)}^{\mathrm{T}}\mathbf{R}_{(V)}^{-1}\mathbf{y}}{\mathbf{m}_{(V)}^{\mathrm{T}}\mathbf{R}_{(V)}^{-1}\mathbf{m}_{(V)}}, \tag{7.21}$$

with estimator variance

$$\mathrm{var}[\hat{\Delta}] = \frac{\Delta^2}{\mathbf{m}_{(V)}^{\mathrm{T}}\mathbf{R}_{(V)}^{-1}\mathbf{m}_{(V)}}. \tag{7.22}$$

These estimators are the estimators that provide the lowest mean-squared error out of all estimators that are linear in the ordered observations. Note that the ordinary sample mean belongs to this class of estimators.

Both estimators are dependent on statistical information of the ordered stochastic variables. Appendix 7.A provides basic theory for order statistics. In particular, the correlations in $\mathbf{R}_{(V)}$ are given by (7.42) and (7.43), and the expected values of $\mathbf{m}_{(V)}$ are found in (7.40).

The Estimator

Consider again the batch of observation samples $\mathbf{z}$. Let $z_{(i)}$, $i = 1, \ldots, N$ be the ordered samples with $z_{(1)}$ the smallest and $z_{(N)}$ the largest, that is $z_{(1)} \leq \cdots \leq z_{(i)} \leq \cdots \leq z_{(N)}$. The estimator should be linear in the ordered samples[3], that is,

$$\hat{g}_{\text{os}}(\mathbf{z}) = \sum_{i=1}^{N} \alpha_i z_{(i)}, \tag{7.23}$$

where $\boldsymbol{\alpha} \triangleq [\alpha_1 \ \ldots \ \alpha_N]^{\text{T}}$ are the filter coefficients to be determined. Applying the theories presented above, the best (in the mean-square sense) unbiased estimator, linear in the ordered samples, is given by the coefficient vector

$$\boldsymbol{\alpha} = \frac{\mathbf{R}_{(V)}^{-1}\,\mathbf{1}}{\mathbf{1}^{\text{T}}\,\mathbf{R}_{(V)}^{-1}\,\mathbf{1}}, \tag{7.24}$$

with a resulting variance

$$\text{var}[\hat{g}_{\text{os}}] = \frac{\Delta^2}{\mathbf{1}^{\text{T}}\,\mathbf{R}_{(V)}^{-1}\,\mathbf{1}}. \tag{7.25}$$

The correlation matrix $\mathbf{R}_{(V)}$ is difficult to compute analytically for the PDF (7.15). Therefore, it has been calculated numerically for some exemplary values of ρ and N. The integrals (7.42) and (7.43) in Appendix 7.A have been solved using a Monte–Carlo technique to obtain numerical values for $\mathbf{R}_{(V)}$. From these, the optimal filter coefficients were calculated using (7.24). Figure 7.5 shows exemplary solutions for $N = 10$ and different values of ρ.

We see that for large values of ρ all α_i go to $1/N$ (0.1 in this case), i.e., the ordinary sample mean. For low ρ, on the other hand, we see that the solution approaches $\alpha_1 = \alpha_N = 1/2$ and the remaining coefficients zero. That is, in the limiting case when $\rho \to 0$ the best estimate of g based on ordered samples is simply the mean of the smallest and the largest sample. These two asymptotic results align perfectly with the results in [Llo52] where the limiting cases – purely uniform and purely Gaussian distribution, respectively – were investigated. Finally, the white line visible in the plot where the two "fins" disappear into the surface marks where $\rho = 0.35$ (or $20\log_{10}\rho \approx -9$). We see that for ρ larger than this, the surface is almost flat at $1/N$, which supports the results from [BNPP00] that the distribution for V is approximately Gaussian in that region, implying that the sample mean should be optimal.

7.5 Midpoint Estimation Cramér–Rao Bound

The Cramér–Rao Bound (CRB) is a theoretical lower bound on the variance of any unbiased estimator for the parameter of interest, given the considered problem

[3]Note, however, that this estimator is *nonlinear* in the observed samples $\mathbf{z}$ since determining the sample order is a nonlinear operation.

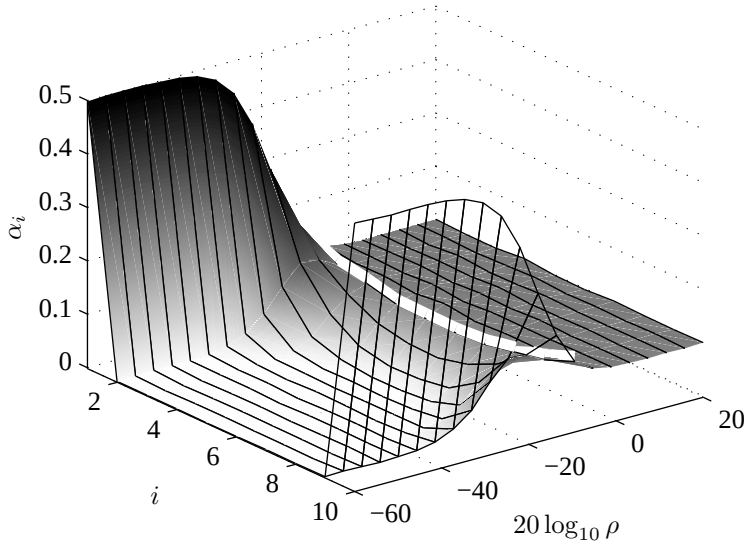


Figure 7.5: The filter coefficients α_i as a function of ρ for $N = 10$.

formulation and model. In this section we will pursue the CRB for the estimation problem described in the previous section.

Assume that the ADC under test follows the model $\mathcal{M}$ introduced earlier. The observation at hand is the N_k samples of the reference signal $z(n)$, $\mathbf{z}^{\{k\}} = \mathbf{z} = [z_1 \ z_2 \ \ldots \ z_{N_k}]^{\mathrm{T}}$, as described above. We seek to find the probability of this observation $\mathbf{z}$ given that we are considering samples when $x(n) = x_k$ only, i.e., $f(\mathbf{z}|\mathbf{x} = x_k \cdot \mathbf{1})$. Under the model assumption $\mathcal{M}$, the input S and the output X of the ADC are related through their joint PMF, from which we can derive the conditional PDF $f_{S|X}(s|x_k)$. Under the assumption that the disturbance U is independent from both S and X, the PDF $f(z_i|x_i = x_k)$ for the single sample z_i was given in (7.4). The total PDF for the observation $\mathbf{z}$, when the samples are assumed to be independent over time, is found by taking the product over $i = 1, 2, \ldots, N_k$, i.e.,

$$f(\mathbf{z}|\mathbf{x} = x_k \cdot \mathbf{1}) = \prod_{i=1}^{N_k} f(z_i|x_i = x_k). \tag{7.26}$$

The CRB for an unbiased estimator $\hat{g}_k(\mathbf{z})$ can now be found from (7.26) as [Kay93]

$$\mathrm{var}[\hat{g}_k(\mathbf{z})] \geq \left(\mathrm{E}\left[\frac{\partial^2 \ln f(\mathbf{z}|\mathbf{x} = x_k \cdot \mathbf{1})}{\partial g_k^2} \right] \right)^{-1}, \tag{7.27}$$

provided that the regularity condition

$$\mathrm{E}\left[\frac{\partial \ln f(\mathbf{z}|\mathbf{x} = x_k \cdot \mathbf{1})}{\partial g_k}\right] = 0 \tag{7.28}$$

holds for all values of g_k. That is, an estimator that estimates g_k from the observation $\mathbf{z}$ with zero bias can never have a variance that is lower than (7.27). An estimator that attains the bound is referred to as *efficient*.

In the following two subsections the CRB for the two different cases introduced in Section 7.4 will be investigated. In the all-Gaussian case we will obtain a closed form analytic solution, while in the case of uniform and Gaussian distributions we will again resort to numerical methods.

Gaussian W and U

We make the same assumptions as in Section 7.4, i.e., $W \in \mathcal{N}(0, \sigma^2)$ and $U \in \mathcal{N}(0, \sigma_U^2)$. The PDF for a single observation of Z is then as in (7.8). As pointed out earlier, this is a problem of estimating a constant in additive zero-mean Gaussian noise with variance $\sigma^2+\sigma_U^2$. Again, this is a standard problem found in the literature (e.g., [Kay93]) and the CRB for this problem is easily calculated using (7.27) to

$$\mathrm{var}[\hat{g}_k(\mathbf{z})] \geq \mathrm{E}\left(\frac{\partial^2 \ln f(\mathbf{z}|\mathbf{x} = x_k \cdot \mathbf{1})}{\partial g_k^2}\right)^{-1} = \frac{\sigma^2 + \sigma_U^2}{N}. \tag{7.29}$$

We see that the estimator $\hat{g}_{\mathrm{sm}}(\mathbf{z})$ in Section 7.4 has a variance that attains the CRB, and the sample mean is thus efficient in this case.

Uniform W and Gaussian U

The assumptions in Section 7.4 are made once more. However, because of the rather complicated PDF for $W + U$ in (7.14) we again have to resort to numerical solutions.

The normalized variable V with PDF (7.15) is considered. If the CRB for a parameter g (the index k is as before omitted for brevity) observed in the presence of V is calculated, the CRB when V is replaced with ΔV is obtained simply by scaling the bound by Δ^2—this is in analogy with the way that the variance of a stochastic variable changes with a scaling factor.

First, the regularity condition (7.28) is verified for this problem. The PDF for a single observation z of $Z = V + g$ is in this case $f_V(z - g)$, and inserting into

(7.28) yields

$$\begin{aligned} \mathrm{E}\left(\frac{\partial \ln f_V(z-g)}{\partial g}\right) &= \int_{-\infty}^{\infty} \frac{\partial \ln f_V(z-g)}{\partial g} f_V(z-g)\, dz \\ &= \int_{-\infty}^{\infty} \frac{f_V(z-g)}{f_V(z-g)} \frac{\partial f_V(z-g)}{\partial g}\, dz \\ &= \frac{\partial}{\partial g} \int_{-\infty}^{\infty} f_V(z-g)\, dz = \frac{\partial}{\partial g} 1 = 0, \end{aligned} \tag{7.30}$$

since $f_V(z-g) > 0$ for all z and $\rho > 0$ (otherwise the limits of integration would depend on g and the order of integration and differentiation cannot be changed).

After verifying that the regularity condition is met, the CRB should be calculated from (7.27). As mentioned above, the CRB has only been calculated numerically for the PDF $f_V(z-g)$ due to the heavy integration required to solve it analytically. The resulting CRB is presented in the next section along with the simulation results in Figures 7.6–7.7.

7.6 Simulations

In order to verify the results obtained above some simulations have been carried out. The purpose of the experiment was to evaluate the estimators obtained in Section 7.4 and compare them with the Cramér–Rao bounds obtained in Section 7.5. However, the results from the case where both W and U are Gaussian are omitted, since these only verify the standard results reported numerous times in the literature. The emphasis is instead on the scenario with uniform W and Gaussian U.

More specifically, we are interested in assessing the difference between the sample mean ($\hat{g}_{\mathrm{sm}}$), the order statistic estimator ($\hat{g}_{\mathrm{os}}$) and a third estimator ($\hat{g}_{\mathrm{mm}}$) which is the mean of the smallest and the largest sample (equivalent to $\hat{g}_{\mathrm{os}}$ when $\rho \to 0$). The experiment was set up according to Figure 7.2(b), i.e., an unknown constant g is corrupted by two independent additive white noises W and U, with the former being uniform in $[-\Delta/2, \Delta/2]$ and the latter zero-mean Gaussian with variance σ_U^2. In each experiment N samples were taken and the following estimators were calculated:

1. Order statistics estimator $\hat{g}_{\mathrm{os}}$ in (7.23) with $\boldsymbol{\alpha}$ from (7.24).
2. Sample mean $\hat{g}_{\mathrm{sm}}$ in (7.9).
3. Min-max estimator $\hat{g}_{\mathrm{mm}} = (\mathbf{z}_{(1)} + \mathbf{z}_{(N)})/2$.
4. Maximum likelihood estimator (MLE), calculated using numerical maximization of the likelihood function $f_{\mathbf{Z}|\mathbf{X}}(\mathbf{z}|\mathbf{x} = x_k \cdot \mathbf{1}; g)$ with respect to g, under the Gaussian-uniform assumption of Section 7.4.

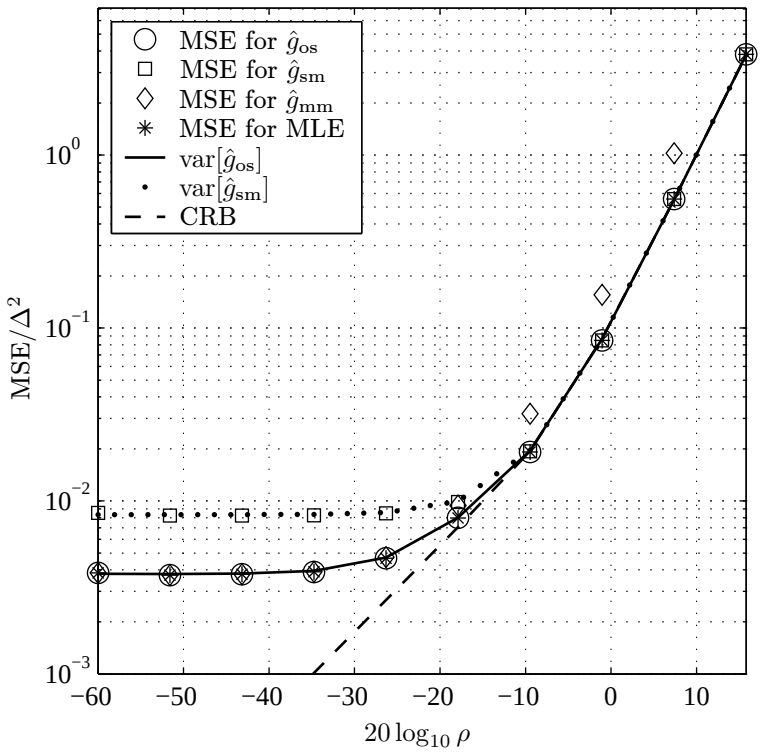


Figure 7.6: Mean squared error results from simulations. The performance in terms of MSE is plotted for four estimators of g: optimized order statistics $\hat{g}_{\mathrm{os}}$ ('◯'), sample mean $\hat{g}_{\mathrm{os}}$ ('□'), the mean of the minimum and maximum samples $\hat{g}_{\mathrm{mm}}$ ('◇'), and the MLE ('∗'). Also, the theoretical variance for $\hat{g}_{\mathrm{os}}$ and $\hat{g}_{\mathrm{sm}}$, and the CRB are plotted using solid, dash-dotted and dashed lines, respectively. This plot shows the results for $N = 10$ samples.

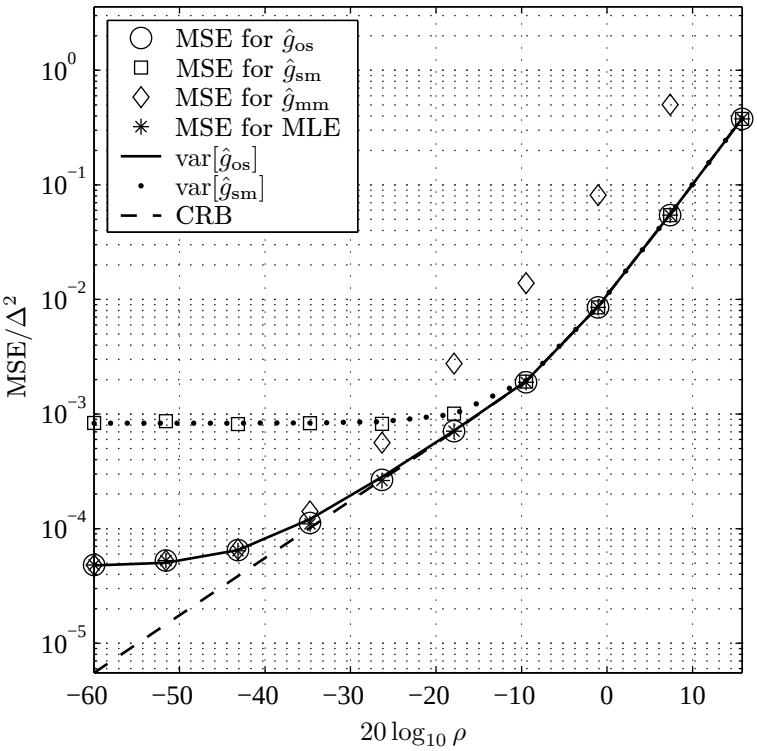


Figure 7.7: Mean squared error results from simulations. This plot shows the results for $N = 100$ samples.

The parameter ρ was varied (as before $\rho \triangleq \sigma_U/\Delta$), and for each value of ρ the experiment was repeated 10 000 times. The mean square error (MSE) between the estimated values and the true value was calculated. The results are plotted in Figures 7.6–7.7.

The first observation is that the order statistics estimator always has the lowest MSE coinciding with the theoretical value (7.25) (dashed line). The maximum likelihood estimator (MLE) of g also results in the same MSE as $\hat{g}_{\text{os}}$ for all ρ.

We see that for $20\log_{10}\rho > -10$ the MSE of the sample mean estimator ($\hat{g}_{\text{sm}}$) is identical to that of the order statistic estimator, and also attains the CRB. Again, this verifies the result from [BNPP00] that for ρ large enough the distribution for V is approximately Gaussian, implying that the sample mean is optimal in the mean-square sense for those values of ρ.

For low ρ, on the other hand, the estimator based on the largest and the smallest samples ($\hat{g}_{\text{mm}}$) actually approaches the performance of the order statistics estimator. This is not surprising, since we know from Section 7.4 that the coefficients

of the order statistics estimator approaches $\alpha_1 = \alpha_N = 1/2$ (with the remaining coefficients zero), which is equivalent to $\hat{g}_{\mathrm{mm}}$.

Another observation is that the order statistics estimator $\hat{g}_{\mathrm{os}}$ attains the CRB for lower ρ when N is increased from 10 to 100 (cf. Figures 7.6–7.7). Also, we know from estimation theory (e.g., [Kay93]) that the MLE – if it exists – attains the CRB when $N \to \infty$. Since the numerical results indicate that the order statistics estimator performs equally well as the MLE, we conjecture that the order statistics estimator $\hat{g}_{\mathrm{os}}$ is asymptotically efficient (in the number of samples N). That is, it seems reasonable to believe that the order statistics estimator attains the CRB for all $\rho > 0$ when $N \to \infty$.

7.7 Roadmap to an Order Statistics Calibration System

The foundation for a novel method for calibrating a look-up table based post-correction system has been presented in this chapter. The method is based on a nonlinear estimator (7.23)–(7.24) using order statistics. However, several issues must be dealt with before a practical correction system using this technique can be implemented. This section points out the main problems that must be solved, and in some cases possible solutions are suggested.

Calibration Signal

It was mentioned in the preamble of Section 7.4 that the PDF of the input signal should preferably be uniform for the uniform-Gaussian assumption to be valid. It can of course be argued that the PDF is approximately uniform within each quantization region if the quantization regions are "small". This means that they should be so small that the PDF of the input signal changes negligibly within one quantization region; in other words, the conditions for high-rate quantization (e.g., [Gra90]) are fulfilled.

Sinewaves have been used extensively in ADC calibration. The benefits of using sinewave signals are several: They are (fairly) easy to generate with high purity, and that standardized sinewave fit methods can be used to re-create the reference signal from the recorded signal (cf. [Std1057, Std1241]). The drawback, on the other hand, is that a single sinusoid has a PDF that is far from uniform. Another disadvantage is that the calibration signal is narrow-banded, so that the calibration is only valid in a narrow band close to the calibration frequency. Many applications in which ADCs are used today are wide-band, wherefore it is of special interest to have a calibration that excites the converter under test in a wide frequency band. A wide-band uniform noise seems like a plausible candidate, but we then loose the ability to estimate the reference signal from the recorded samples.

Schoukens and Dobrowiecki [SD98] presented a method to design a broadband excitation signal, consisting of a sum of sinusoids (so called *multisine*), where the amplitude distribution can be tailored to meet certain demands. This method could

be useful in ADC calibration, since it would provide excitation in a wide frequency range, and can be designed to have uniform PDF. A reference signal can then be estimated from the recorded output, since we are again dealing with sinewaves. In particular, Andersson and Händel have presented two methods for estimating the parameters of a multisine. The two methods – suitable for instrumentation and measurement purposes – are presented in [AH03] and [AH04], respectively, and both are described in [And05]. Combining the design method for multisines with multisine estimation could form a powerful tool for calibrating ADCs in a wide frequency band.

Knowledge of the Shape Parameter ρ

The new estimator presented in this chapter cannot be fully utilized without knowledge of the parameter ρ, defined in (7.16). The parameter can be interpreted as a shape parameter, where small ρ gives an approximately rectangular PDF, while larger ρ leads to a more Gaussian distribution.

We have not presented any methods for estimating the parameter ρ in this chapter. One possible solution is to derive an approximate value based on prior knowledge of the calibration system, such as (nominal) quantization bin width, reference signal variance, etc. Another option is to devise an estimator based on the recorded samples.

Calculating the Filter Coefficients

It was mentioned in Section 7.4 that the filter coefficients $\boldsymbol{\alpha}$ in (7.24) were quite difficult to calculate. The reason for this was that the integrals (7.42) and (7.43) in Appendix 7.A are difficult to solve analytically. The results presented in Figure 7.5 and in the simulations of Section 7.6 were found by using numerical methods to solve the integrals.

In a practical system, this problem must be overcome. One reason for this is that it can be expected that the operating point of the estimator, in terms of values for ρ and N, will change from one quantization level to another. This since it is likely that (a) the reference signal has different accuracy for different levels and/or (b) the number of samples recorded in each level vary. The best solution is of course to find an exact, or possibly approximate, closed form solution to the α-parameters. However, in the likely case that this is not feasible, we must resort to other methods.

Approximating the α-coefficients

When examining the shape of Figure 7.5, it is evident that the surface can be described as two extreme cases—viz. $1/N$ for large ρ and $\alpha_1 = \alpha_N = 1/2$ for ρ close to zero—and a gradient region in between. One way to acquire approximate

coefficients in an efficient manner could be to find a mixture between these two extreme cases, where ρ acts as a mixing parameter.

Switching Between the Extremes

Consider Figures 7.6–7.7. Besides the fact that the order statistics estimator $\hat{g}_{\text{os}}$ is best in all cases, it is interesting to note that for every value of ρ, either the sample mean ($\hat{g}_{\text{sm}}$) or the min-max estimator ($\hat{g}_{\text{sm}}$) is very close to optimal. Thus, if we out of these two estimators always use the one with lowest variance, the resulting estimates will be not far from optimal.

It can be seen from Figures 7.6–7.7 that there is a ρ at which the MSEs for the two estimators intersect. It is also clear that this point changes when the number of samples N is changed. Thus, with knowledge of this cross-over point, the proper selection of estimator can be made. The benefit of this approach is of course that these two estimators are far more easier to calculate, since we no longer use the integrals (7.42) and (7.43) in Appendix 7.A.

Now, calculating the cross-over point, that is, the point at which $\text{var}[\hat{g}_{\text{sm}}] = \text{var}[\hat{g}_{\text{mm}}]$, is of interest. The variance for $\hat{g}_{\text{sm}}$ is

$$\text{var}[\hat{g}_{\text{sm}}] = \frac{\frac{\Delta^2}{12} + \sigma_U^2}{N} = \frac{\Delta^2}{12N}(1 + 12\rho^2) \tag{7.31}$$

when taking the mean of N samples of $Z = \Delta V + g$. For $\hat{g}_{\text{mm}}$, the variance is

$$\begin{aligned} \text{var}[\hat{g}_{\text{mm}}] &= \text{var}\left[\frac{Z_{(1)} + Z_{(N)}}{2}\right] = \text{var}\left[\frac{\Delta V_{(1)} + g + \Delta V_{(N)} + g}{2}\right] \\ &= \frac{\Delta^2}{4}\left(\text{E}\left[V_{(1)}^2\right] + \text{E}\left[V_{(N)}^2\right] + 2\,\text{E}\left[V_{(1)}V_{(N)}\right]\right). \end{aligned} \tag{7.32}$$

Again, the order statistics are defined in Appendix 7.A. Also this time, the integrals are difficult to calculate analytically, and we resort to numerical methods.

The equation $\text{var}[\hat{g}_{\text{sm}}] = \text{var}[\hat{g}_{\text{mm}}]$, i.e.,

$$\frac{1}{12N}(1 + 12\rho^2) = \frac{1}{4}\left(\text{E}\left[V_{(1)}^2\right] + \text{E}\left[V_{(N)}^2\right] + 2\,\text{E}\left[V_{(1)}V_{(N)}\right]\right), \tag{7.33}$$

is solved with respect ρ for different N using numerical methods. The results are plotted in Figure 7.8. The circles mark the solutions to (7.33). Also, a least-squares fit is shown with a solid line. The fit resulted in the function

$$N = 0.03839\,\rho^{-2.868} \tag{7.34}$$

which is the line in the plot. The rightmost point ($20\log_{10}\rho \approx -16$, $N = 5$) was omitted from the data while doing the fit, since this point is heavily affected by border effects.

From the plot it is possible to tell, for a certain combination of ρ and N, whether the sample mean or the mean of the largest and smallest sample should be used

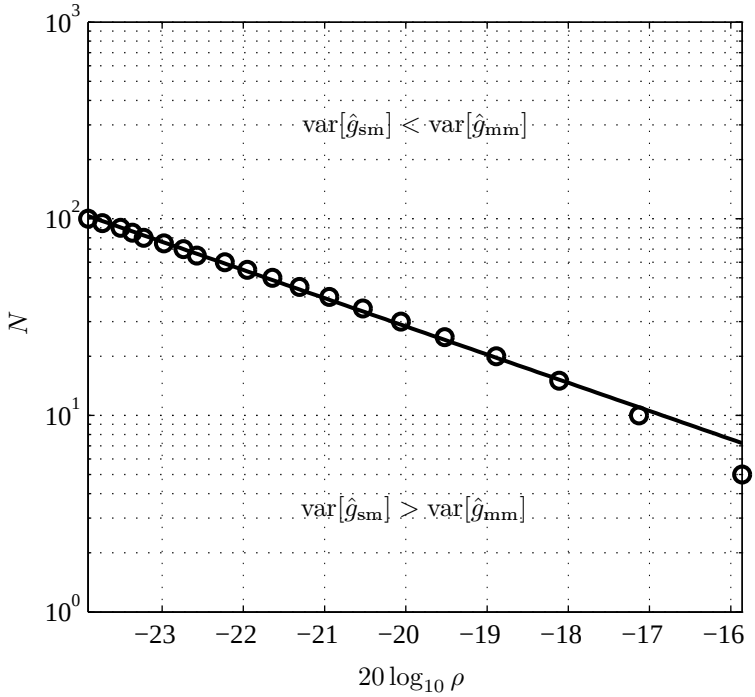


Figure 7.8: The circles show points in the ρ-N-plane where the equation $\mathrm{var}[\hat{g}_{\mathrm{sm}}] = \mathrm{var}[\hat{g}_{\mathrm{mm}}]$ holds, i.e., where $\hat{g}_{\mathrm{sm}}$ and $\hat{g}_{\mathrm{mm}}$ are equally good. Above these points, $\hat{g}_{\mathrm{sm}}$ is better, while the opposite is true under. The solid line represents the least-squares fit to the data points.

to estimate g. If the pair (ρ, N) is located above the line, the variance of the sample mean is the smallest, wherefore $\hat{g}_{\mathrm{sm}}$ should be selected in favor of $\hat{g}_{\mathrm{mm}}$. The opposite is of course true below the line.

7.A Probability Distributions of Order Statistics

In this appendix, the fundamentals of order statistics are reviewed. The results can be found in the literature, e.g., [AK97].

Let $X_1, X_2, \ldots, X_N$ be i.i.d. stochastic variables with a common continuous distribution function $F(x)$ and a PDF $f(x)$. The random vector $\bar{\mathbf{X}} \triangleq [X_{(1)}\ X_{(2)}\ \cdots\ X_{(N)}]$, where the random variables have been arranged in ascending order so that $X_{(1)} \leq X_{(2)} \leq \cdots \leq X_{(N)}$, is referred to as the *order statistics.* The joint PDF of $\bar{\mathbf{X}}$ is

$$f_{\bar{\mathbf{X}}}(x_{(1)}, x_{(2)}, \ldots, x_{(N)}) = N! f(x_{(1)}) f(x_{(2)}) \ldots f(x_{(N)}) \tag{7.35}$$

in the region $-\infty < x_{(1)} < x_{(2)} < \cdots < x_{(N)} < \infty$ and zero outside.

The PDF for the k-th element of $\bar{\mathbf{X}}$ is

$$f_{X_{(k)}}(x_{(k)}) = \frac{N!}{(k-1)!\,(N-k)!}\, F(x_{(k)})^{k-1} \left(1 - F(x_{(k)})\right)^{N-k} f(x_{(k)}). \tag{7.36}$$

Furthermore, the PDF for a subset of K elements in $\bar{\mathbf{X}}$ is

$$\begin{aligned} f_{X_{(k_1)}, X_{(k_1+k_2)}, \ldots, X_{(k_1+k_2+\cdots+k_K)}}&(x_{(k_1)}, x_{(k_1+k_2)}, \ldots, x_{(k_1+k_2+\cdots+k_K)}) \\ &= \frac{N!}{(k_1-1)!(k_2-1)!\cdots(k_K-1)!(N-k_1-k_2-\cdots-k_K)!} \times \\ &\quad F(x_{(k_1)})^{k_1-1} \left(F(x_{(k_1+k_2)}) - F(x_{(k_1)})\right)^{k_2-1} \cdots \times \\ &\quad \left(1 - F(x_{(k_1+k_2+\cdots+k_K)})\right)^{N-k_1-k_2-\cdots-k_K} \times \\ &\quad f(x_{(k_1)}) \cdots f(x_{(k_1+k_2+\cdots+k_K)}) \end{aligned} \tag{7.37}$$

in the region $-\infty < x_{(k_1)} < x_{(k_1+k_2)} < \cdots < x_{(k_1+k_2+\cdots+k_K)} < \infty$ and zero outside.

From these PDFs, we can derive different properties of the order statistics. For instance, the PDF of the smallest element in $\bar{\mathbf{X}}$, $X_{(1)}$ is

$$f_{X_{(1)}}(x_{(1)}) = N \left(1 - F(x_{(1)})\right)^{N-1} f(x_{(1)}), \tag{7.38}$$

the PDF for the largest element is

$$f_{X_{(N)}}(x_{(N)}) = N F(x_{(1)})^{N-1} f(x_{(1)}), \tag{7.39}$$

and the expected value for the k-th element is

$$\begin{aligned} \mathrm{E}[X_{(k)}] = \frac{N!}{(k-1)!\,(N-k)!} \times \\ \int_{-\infty}^{\infty} F(x_{(k)})^{k-1} \left(1 - F(x_{(k)})\right)^{N-k} f(x_{(k)}) x_{(k)}\, dx_{(k)}. \end{aligned} \tag{7.40}$$

Finally, the correlation between two elements in $\bar{\mathbf{X}}$ is of interest. The joint PDF for the i-th and j-th elements is

$$\begin{aligned} f_{X_{(i)}X_{(j)}}(x_{(i)}x_{(j)}) = \frac{N!}{(i-1)!\,(j-i-1)!\,(N-j)!} \times \\ F(x_{(i)})^{i-1} \left(F(x_{(i)}) - F(x_{(j)})\right)^{j-i-1} \left(1 - F(x_{(j)})\right)^{N-j} f(x_{(i)}) f(x_{(j)}) \end{aligned} \tag{7.41}$$

for $x_{(i)} < x_{(j)}$ and zero otherwise. This gives that the correlation between two elements is

$$\begin{aligned} \mathrm{E}\left[X_{(i)}X_{(j)}\right] = \int_{-\infty}^{\infty} \int_{-\infty}^{x_{(j)}} \frac{N!}{(i-1)!\,(j-i-1)!\,(N-j)!} \times \\ F(x_{(i)})^{i-1} \left(F(x_{(i)}) - F(x_{(j)})\right)^{j-i-1} \left(1 - F(x_{(j)})\right)^{N-j} \times \\ f(x_{(i)}) f(x_{(j)})\, dx_{(i)} dx_{(j)} \end{aligned} \tag{7.42}$$

for $i \neq j$, and

$$\mathrm{E}\left[X_{(i)}^2\right] = \frac{N!}{(i-1)!\,(N-i)!} \times \int_{-\infty}^{\infty} F(x_{(i)})^{i-1} \left(1 - F(x_{(i)})\right)^{N-i} f(x_{(i)}) x_{(i)}^2 \, dx_{(i)} \quad (7.43)$$

for $i = j$.

Chapter 8

Theoretical Limits for ADC Correction

This thesis has thus far dealt with post-correction – manly applying look-up tables – in the optimal case. In a practical post-correction application it is very likely that the correction values will be stored with fixed-point precision. However, most of the evaluations and experiments reported in the literature have been conducted with infinite precision in the representation of the correction values stored in the LUT. One of few exceptions is [IC86], where experimental results indicated that the precision of the correction values strongly affect the outcome of the correction.

In this chapter the relationship between the precision of the correction values and the resulting ADC performance after correction is investigated. First, the quantization and correction model is repeated, introducing also a few new notations. Then, the outcome of the best possible post-correction is derived, in terms of MSE, SINAD and ENOB. Finally, the effects of nonideal correction values – due to fixed-point precision – is investigated. The results are verified by simulations in Section 8.6, and also using experimental ADC data in Section 8.7. The outcome of the experiments are discussed in the concluding Section 8.8.

8.1 ADC and Correction System Model

The ADC is assumed to operate as described in Section 1.2, with an ideal sample-and-hold circuit. Thus, the sampled signal $s(n)$ is regarded as input to the system. The quantizer has b bits, resulting in $M = 2^b$ quantization levels. The output, denoted $x(n)$, is a quantized version of $s(n)$. The notation $x(n) = \mathcal{Q}(s(n))$ is used to denote the quantization operation. Note that $\mathcal{Q}(\cdot)$ does not necessarily have to be a uniform quantization, but represents the actual transfer function of the ADC at hand.

It is assumed that the input value $s(n)$ is drawn from a stochastic variable S with probability density function (PDF) $f_S(s)$. The temporal properties for S are

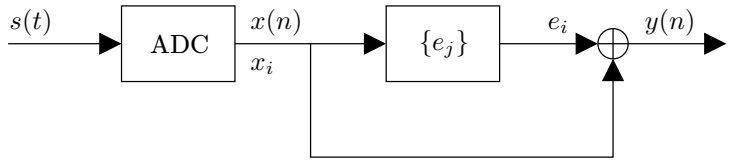


Figure 8.1: Additive correction system.

immaterial since the quantizer is assumed to be non-dynamic, i.e., the output of the quantizer at time n depends only on the input at the same instant. Let MSE_Q denote the MSE for the quantizer without correction, i.e.,

$$\begin{aligned}\mathrm{MSE}_Q &= \mathrm{E}[(S-x)^2] = \int (s-\mathcal{Q}(s))^2 \, f_S(s)\, ds \\ &= \sum_i \int_{s\in\mathcal{S}_i} (s-x_i)^2 \, f_S(s)\, ds.\end{aligned} \tag{8.1}$$

A static additive correction as described in Chapter 3 is employed. Figure 8.1 depicts the correction system. The corrected value y is produced by adding a correction term $e(x)$ to the output x so that $y = x + e(x)$. Every possible output value $x \in \{x_j\}_{j=0}^{M-1}$ is associated with a correction term $e(x) \in \{e_j\}_{j=0}^{M-1}$.

Optimal correction values for minimizing the mean-square error $\mathrm{E}[(S-y)^2]$ are used (note that y is a function of S). In Section 6.2 the MMSE optimal correction values were derived. It was shown that if the quantization regions $\{\mathcal{S}_j\}$ are assumed fixed, the optimal correction values are given by

$$e_{j,\,\mathrm{opt}} = \arg\min_y \mathrm{E}[(y-S)^2 | S\in\mathcal{S}_j] = \frac{\int_{s\in\mathcal{S}_j} s\, f_S(s)\, ds}{\int_{s\in\mathcal{S}_j} f_S(s)\, ds} - x_j. \tag{8.2}$$

When representing the correction values with infinite precision, i.e., using the values (8.2), the resulting MSE after correction is

$$\begin{aligned}\mathrm{MSE}_o &= \mathrm{E}[(S-y)^2] = \mathrm{E}\left[(S-x-e(x))^2\right] \\ &= \mathrm{E}\left[(S-x)^2 - 2(S-x)\, e(x) + e(x)\right] \\ &= \mathrm{MSE}_Q + \mathrm{E}\left[e(\mathcal{Q}(S))^2\right] - 2\,\mathrm{E}\left[(S-\mathcal{Q}(S))\, e(\mathcal{Q}(S))\right],\end{aligned} \tag{8.3}$$

where the last equality comes from applying (8.1). In order to simplify the expression, we use (8.2) to obtain

$$\int_{s\in\mathcal{S}_i} s\, f_S(s)\, ds = (e_i + x_i) \int_{s\in\mathcal{S}_i} f_S(s)\, ds, \tag{8.4}$$

and use it in the last term of the expression (8.3) above. Thus,

$$\begin{aligned}
\mathrm{E}[(s-\mathcal{Q}(s))\,e(\mathcal{Q}(s))] &= \int (s-\mathcal{Q}(s))\,e(\mathcal{Q}(s))\,f_S(s)\,ds \\
&= \sum_i \int_{s\in\mathcal{S}_i} (s-x_i)\,e_i\,f_S(s)\,ds \\
&= \sum_i \left(e_i \int_{s\in\mathcal{S}_i} s\,f_S(s)\,ds - x_i\,e_i \int_{s\in\mathcal{S}_i} f_S(s)\,ds \right) \\
&= \sum_i \left(e_i\,(e_i+x_i) \int_{s\in\mathcal{S}_i} f_S(s)\,ds - x_i\,e_i \int_{s\in\mathcal{S}_i} f_S(s)\,ds \right) \\
&= \sum_i e_i^2 \int_{s\in\mathcal{S}_i} f_S(s)ds = \mathrm{E}\left[e(\mathcal{Q}(S))^2\right],
\end{aligned} \tag{8.5}$$

which is in fact nothing but the variance of the correction value e. Reapplying this in (8.3) yields

$$\mathrm{MSE}_o = \mathrm{MSE}_Q - \mathrm{E}\left[e(\mathcal{Q}(S))^2\right]. \tag{8.6}$$

Since $\mathrm{E}\left[e(\mathcal{Q}(S))^2\right]$ always is a non-negative quantity, we can immediately see that the MSE after MMSE-optimal correction is never higher than the MSE before correction, or $\mathrm{MSE}_o \leq \mathrm{MSE}_Q$.

In the next section we will derive a more specific expression for the resulting MSE after correction based on certain assumptions on the quantizer behavior.

8.2 Optimal Correction Results Under Random DNL

In this section we will pursue a limit on how good a quantizer can be after correction. The nonidealities of the quantizer in terms of DNL are modeled as a random process. The maximum achievable MSE after MMSE optimal correction is derived and found to be dependent on the variance of the DNL process. The problem considered in this section was also studied in [GST96a], and similar results as (8.18) were found then.

Assume that the ADC, or quantizer, suffers from a certain differential nonlinearity DNL (cf. Section 1.4). We will here describe the DNL statistically in the following way. The ideal code bin width of the quantizer is denoted Δ. The actual code bin width for the k-th code bin is

$$W[k] = \Delta + d_\Delta[k]. \tag{8.7}$$

The DNL naturally becomes $\mathrm{DNL}_k = d_\Delta[k]/\Delta$ in accordance with (1.5). The differences $d_\Delta[k]$, $k = 1,\, 2,\, \ldots,\, M-2$, are considered to be independent realizations of a stochastic variable D with probability density function $f_D(d)$. It is assumed that $f_D(d)$ is an even function – implying zero-mean – and that the variance of D is σ_D^2.

Figure 8.2 shows a comparison between data from a real ADC and synthetic DNL as in (8.7). The real ADC is an Analog Devices AD9430, 12-bit converter. The simulated DNL was generated from a Gaussian random variable, having the same variance as estimated from the measured DNL, viz. $\sigma_D^2 = 0.0068$. The first row shows the DNL plots (the shaded areas of Figure (a) are omitted from the statistics, due to poor accuracy in the estimated DNL). The second row shows histograms over the DNLs, and the last row shows estimates of the auto-correlation functions. It is seen that the synthetic DNL quite accurately models that of the experimental data. The largest discrepancy is in the auto-correlations, where the measured auto-correlation has a large negative correlation at lags –1 and 1. This is not the case in the synthetic DNL since it is white.

The quantizer is still fed with a signal modeled as a stochastic variable S, with a PDF $f_S(s)$. The MSE is written as

$$\begin{aligned} \mathrm{MSE} &= \mathrm{E}_S[(S-y)^2] = \int (s-y)^2 \, f_S(s)\, ds \\ &= \sum_{k=0}^{M-1} \int_{\mathcal{S}_k} (s-y_k)^2 \, f_S(s)\, ds \triangleq \sum_{k=0}^{M-1} \mathrm{MSE}(k), \end{aligned} \tag{8.8}$$

where y_k is the corrected output for the k-th level: $y_k = x_k + e_k$. The mean-squared error in the k-th quantization region $\mathcal{S}_k$ as a function of $d_\Delta[k]$ is then

$$\begin{aligned} \mathrm{MSE}(k;\, d_\Delta[k]) &= \mathrm{E}_S[(S-y_k)^2 | S \in \mathcal{S}_k;\, d_\Delta[k]] \\ &= \int_{\mathcal{S}_k} (s-y_k)^2 f_S(s)\, ds. \end{aligned} \tag{8.9}$$

Note that the dependence on $d_\Delta[k]$ is in $\mathcal{S}_k$. Assume that the quantization region $\mathcal{S}_k$ is sufficiently small, and that $f_S(s)$ is sufficiently smooth, so that $f_S(s)$ can be considered a constant C_k within $\mathcal{S}_k$. Then, the MSE becomes

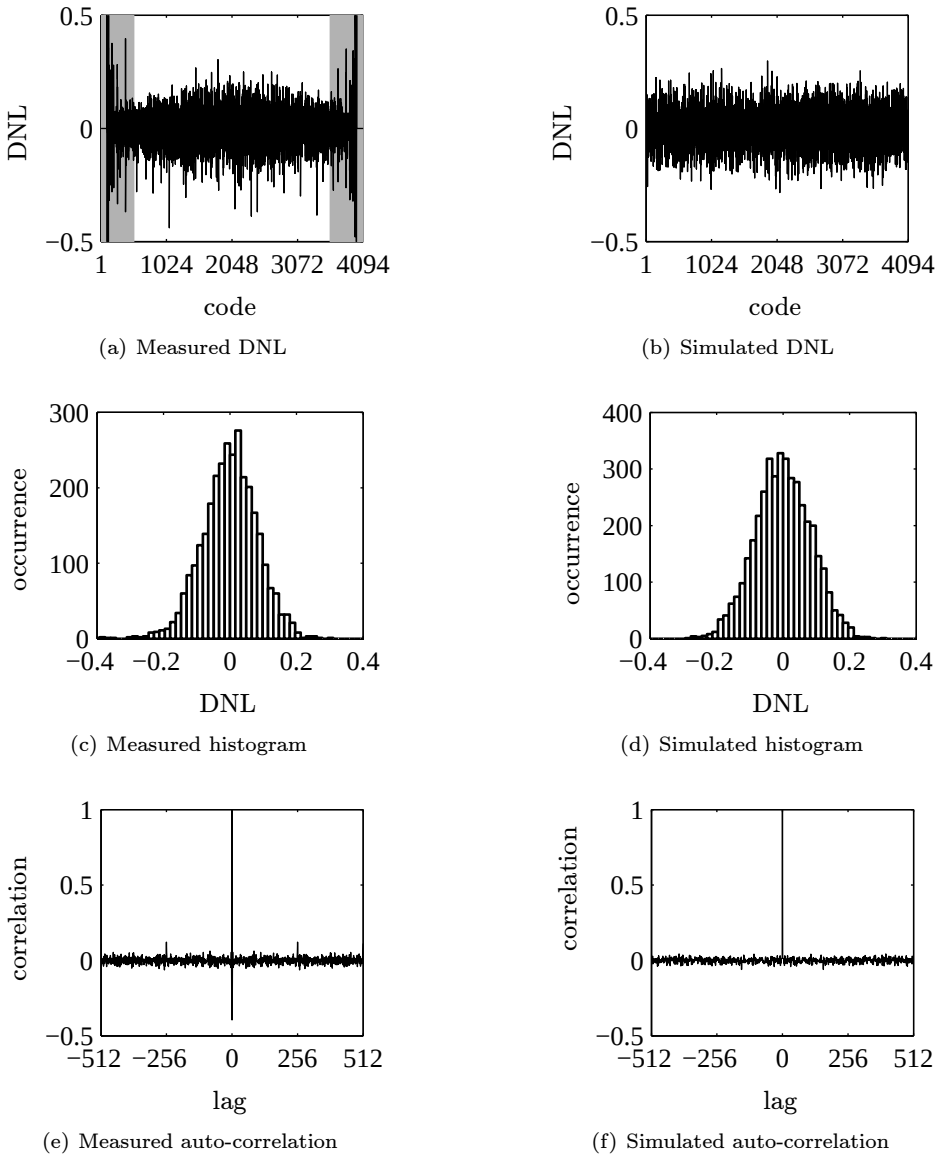
$$\mathrm{MSE}(k;\, d_\Delta[k]) = C_k \int_{\mathcal{S}_k} (s-y_k)^2 \, ds. \tag{8.10}$$

We also know in this case that the MMSE-optimal y_k is the midpoint of $\mathcal{S}_k$ (cf. Section 6.4). With the substitution $t = s - y_k$ the integral can be written as

$$\mathrm{MSE}(k;\, d_\Delta[k]) = 2C_k \int_0^{\frac{\Delta + d_\Delta[k]}{2}} t^2 \, dt = \frac{C_k}{12} \left(\Delta + d_\Delta[k]\right)^3. \tag{8.11}$$

By taking the expected value of $\mathrm{MSE}(k;\, D)$ with respect to D, the MSE for the k-th quantization region is:

$$\begin{aligned} \mathrm{MSE}(k) &= \mathrm{E}_D\left[\mathrm{MSE}(k;\, D)\right] = \mathrm{E}_D\left[\frac{C_k}{12}\left(\Delta + D\right)^3\right] \\ &= \frac{C_k}{12}\left(\Delta^3 + 3\Delta\,\mathrm{E}[D^2] + \mathrm{E}[D^3]\right). \end{aligned} \tag{8.12}$$



(a) Measured DNL

(b) Simulated DNL

(c) Measured histogram

(d) Simulated histogram

(e) Measured auto-correlation

(f) Simulated auto-correlation

Figure 8.2: Comparison between the DNL of a real ADC (left column) and simulated DNL (right column). The shaded areas of Figure (a) are omitted from the statistics, due to poor accuracy in the estimated DNL.

The fact that D is zero-mean was used in the last equality. Further, $\mathrm{E}[D^3] = 0$, since $f_D(d)$ is even. Thus, the MSE becomes

$$\mathrm{MSE}(k) = \frac{C_k}{12}\left(\Delta^3 + \Delta\sigma_D^2\right) = \frac{C_k}{12}\left(1 + \frac{\sigma_D^2}{\Delta^2}\right)\Delta^3 \tag{8.13}$$

Upon inserting this into (8.8) the overall MSE is obtained as

$$\mathrm{MSE}(\sigma_D^2) = \frac{\Delta^3}{12}\left(1 + \frac{\sigma_D^2}{\Delta^2}\right)\sum_{k=0}^{M-1} C_k. \tag{8.14}$$

It is perhaps more interesting to consider the relative MSE. In particular, we consider the MSE related to the MSE for an ideal quantizer, fed with the same signal. The MSE for an ideal quantizer, denoted MSE_Q as before, was expressed in (8.1). By making the smoothness assumption again, the MSE can be approximated as

$$\mathrm{MSE}_Q = \sum_{k=0}^{M-1}\int_{s\in\mathcal{S}_k'}(s - x_k)^2 f_S(s)\,ds \approx 2\sum_{k=0}^{M-1} C_k' \int_0^{\frac{\Delta}{2}} t^2\,dt = \frac{\Delta^3}{12}\sum_{k=0}^{M-1} C_k', \tag{8.15}$$

where $'$ is used to denote that $\mathcal{S}_k'$ and C_k' are not necessarily equal to $\mathcal{S}_k$ and C_k, respectively. The potential discrepancy between $\mathcal{S}_k'$ and $\mathcal{S}_k$ does of course come from the deviation $d_\Delta[k]$ in the regions. Since the regions may end up at different places, the assumed constant value C_k' for the PDF $f_S(s)$ may consequently change to C_k.

Define the ratio between $\mathrm{MSE}(\sigma_D^2)$ and MSE_Q as

$$\kappa \triangleq \frac{\mathrm{MSE}(\sigma_D^2)}{\mathrm{MSE}_Q} = \frac{\frac{\Delta^3}{12}\left(1 + \frac{\sigma_D^2}{\Delta^2}\right)\sum_{k=0}^{M-1} C_k}{\frac{\Delta^3}{12}\sum_{k=0}^{M-1} C_k'}. \tag{8.16}$$

The assumption that

$$\sum_{k=0}^{M-1} C_k \approx \sum_{k=0}^{M-1} C_k' \tag{8.17}$$

is rather reasonable—multiplying each sum by Δ gives an approximation of the integral $\int f_S(s)\,ds = 1$. Under that assumption, κ simplifies to

$$\kappa \approx \left(1 + \frac{\sigma_D^2}{\Delta^2}\right). \tag{8.18}$$

This is the increase in MSE that is inflicted by the DNL *after* MMSE optimal correction is applied.

SINAD and ENOB

The result (8.18) above can be directly translated to a degradation in SINAD and ENOB. The SINAD is defined as (1.7)

$$\text{SINAD} = 20\log_{10}\frac{\text{RMS}_{\text{signal}}}{\text{RMS}_{\text{noise}}}, \tag{8.19}$$

where we can use $\text{RMS}_{\text{noise}} = \sqrt{\text{MSE}} = \sqrt{\kappa\text{MSE}_Q}$. This is of course assuming that there are no other noise sources affecting the quantizer. Thus, the difference between the SINAD with optimally corrected DNL errors and the SINAD of an ideal converter is

$$\Delta\text{SINAD} = \text{SINAD}_\kappa - \text{SINAD}_{\text{ideal}} = -10\log_{10}\left(1 + 3\frac{\sigma_D^2}{\Delta^2}\right). \tag{8.20}$$

The ENOB is directly linked with the SINAD (cf. (1.9)), and we can therefore state the difference between the ENOB with DNL errors and the ENOB of an ideal converter as

$$\Delta\text{ENOB} = -\frac{1}{2}\log_2\left(1 + 3\frac{\sigma_D^2}{\Delta^2}\right), \tag{8.21}$$

again provided that the errors are corrected using (8.2).

8.3 Fixed-Point Resolution for Correction Values

It is not always feasible, let alone practical, to implement an ADC post-correction system, such as the one in Figure 8.1, using floating-point[1] representation for the stored correction values $\{e_j\}_{j=0}^{M-1}$. It is natural to settle for a specific precision with which the correction terms are stored, e.g., a certain number of bits. Obviously, the performance of the corrected ADC will depend on which precision that is used.

The precision of digitally stored values is often stated as a number of bits. Assume that the table is stored using τ bits and that the ADC to be corrected converts the signal into b-bit values. If we know that the ADC only has error in the lower bits, then we can "shift" the bits of the correction table and obtain a correction with higher effective precision. For example, if the ADC has 10 bits, but only the 2 LSBs need correction, then the remaining bits of the correction values (minus the sign bit) can be used to get a better precision.

After the correction with a shifted correction value we have an ADC with a b-bit resolution but a supra-b bit precision. That is, the ADC still has got 2^b quantization regions, but the reconstruction levels after correction are represented with more than b bits.

The problem gets easier to analyze if the resolution η, being the smallest possible difference between two different correction values, is used instead of the actual

[1] Floating-point representation of numbers does not have infinite precision, but it is the closest to infinite precision we can muster in a digital implementation.

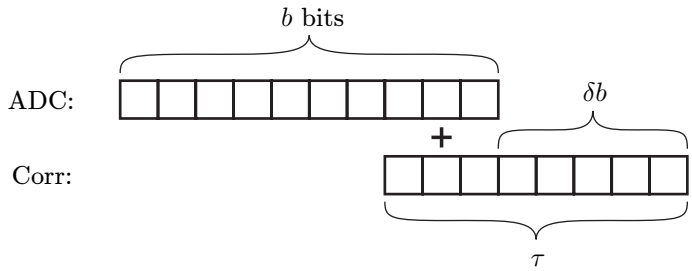


Figure 8.3: Addition of the ADC output with a correction value. The bits of the table value are shifted in order to enhance the precision of the corrected ADC.

number of bits τ. Moreover, we are here only interested in the number of extra bits δb added by the correction term. See Figure 8.3 for an illustration. The relationship between δb and η is straightforward:

$$\eta = 2^{-\delta b} \text{ LSBs.} \tag{8.22}$$

It is assumed that the correction values never exceed the largest number that can be represented by the τ correction bits, meaning that saturation of the correction values does not happen.

Let $\tilde{e}_i$ be the (uniformly) quantized version of the table entry e_i, i.e.,

$$\tilde{e}_i = \mathcal{Q}_\eta(e_i). \tag{8.23}$$

This is the correction value that would be used in a post-correction system with a fixed-point resolution η. The notation $\mathcal{Q}_\eta$ is used to distinguish this quantization from the one performed in the converter ($\mathcal{Q}$). We assume that one of the quantization cells is centered at zero. That is, if a certain correction term is within the interval $[-\eta/2, \eta/2]$ it will be quantized to zero, and, since the quantization of correction terms is uniform (ideal round-off), all other possible quantized values are located at an integer multiple of η. Hence, we can say that

$$\tilde{e}_i \in \{k\eta : \ k = \ldots, -2, -1, 0, 1, 2, \ldots\}. \tag{8.24}$$

Also let

$$\delta_i = \tilde{e}_i - e_i \tag{8.25}$$

be the difference between the fixed-point and the infinite-precision correction terms. The notation $\delta(x)$ denotes the correction term quantization error associated with a specific x, i.e., $\delta(x) = \delta_i$ if $x = x_i$.

MSE

The MSE obtained using the quantized correction terms becomes

$$\begin{aligned}\mathrm{MSE}_\eta &\triangleq \mathrm{E}\left[(S-x-\tilde{e}(x))^2\right] = \mathrm{E}\left[(S-x-e(x)-\delta(x))^2\right] \\ &= \mathrm{E}\left[(S-x-e(x))^2\right] + \mathrm{E}[\delta(x)^2] - 2\,\mathrm{E}[(S-x-e(x))\,\delta(x)] \\ &= \mathrm{MSE}_o + \mathrm{E}[\delta(x)^2] - 2\,\mathrm{E}[(S-x-e(x))\delta(x)].\end{aligned} \tag{8.26}$$

The error term $\delta(x)$ ultimately depends on the stochastic variable S. Analyze the last term to find that

$$\begin{aligned}\mathrm{E}[(S-x-e(x))\,\delta(x)] &= \sum_i \int_{s\in\mathcal{S}_i} (s-x_i-e_i)\,\delta_i\, f_S(s)\,ds \\ &= \sum_i \delta_i \left(\int_{s\in\mathcal{S}_i} (s-x_i)\, f_S(s)\,ds - \int_{s\in\mathcal{S}_i} e_i\, f_S(s)\,ds \right) = 0,\end{aligned} \tag{8.27}$$

where, again, the (modified) relationship in (8.4) has been used in the last term. This results in that the MSE when using quantized correction terms is

$$\mathrm{MSE}_\eta = \mathrm{MSE}_o + \mathrm{E}[\delta(x)^2]. \tag{8.28}$$

The second moment of the error $\mathrm{E}[\delta(x)^2]$ can further be written as

$$\begin{aligned}\mathrm{E}[\delta(x)^2] &= \int \delta(x)^2\, f_S(s)\,ds = \sum_i \int_{s\in\mathcal{S}_i} \delta_i^2\, f_S(s)\,ds \\ &= \sum_i \int_{s\in\mathcal{S}_i} f_S(s)\,ds \frac{\int_{s\in\mathcal{S}_i} \delta_i^2\, f_S(s)\,ds}{\int_{s\in\mathcal{S}_i} f_S(s)\,ds} \\ &= \sum_i \int_{s\in\mathcal{S}_i} f_S(s)\,ds\, \mathrm{E}[\delta^2 | S\in\mathcal{S}_i].\end{aligned} \tag{8.29}$$

Under the assumption that the quantization error δ_i is uniformly distributed in $[-\eta/2,\, \eta/2]$, then each $\mathrm{E}[\delta^2|S\in\mathcal{S}_i] = \eta^2/12$ for all i, and (8.29) becomes

$$E[\delta(x)^2] = \frac{\eta^2}{12} \sum_i \int_{s\in\mathcal{S}_i} f_S(s)\,ds = \frac{\eta^2}{12}. \tag{8.30}$$

The MSE in (8.28) then boils down to

$$\mathrm{MSE}_\eta = \mathrm{MSE}_o + \frac{\eta^2}{12}. \tag{8.31}$$

Since η is expressed in LSBs, so should also MSE_o be. Alternatively, the second term can be scaled by Δ^2 to get the result in input units, e.g., V^2 (volts squared).

It is reasonable to believe that the assumption leading up to (8.30) is valid for small η, i.e., when the quantization is assumed to be "high-rate". (See [WKL96] or Section 4.1 for a thorough discussion and precise conditions for the uniformity of the quantization noise.) However, as η grows large the assumption will become invalid, motivating the asymptotic analysis.

Asymptotic MSE

Recall that one of the quantization cells is centered at zero and that all table values e_i that fall within $[-\eta/2, \eta/2]$ will be quantized to $\tilde{e}_i = 0$. When we enlarge the quantization step, i.e., when $\eta \to \infty$, all $\tilde{e}_i$ will inevitably be zero, since all table values will fall into the expanding center region at zero. Consequently, the resulting MSE becomes

$$\mathrm{MSE}_\eta = \mathrm{E}[(S - x - 0)^2] = \mathrm{E}[(S - x)^2] = \mathrm{MSE}_Q \tag{8.32}$$

when the resolution tends to zero. The interpretation is straightforward: since no correction is effected, the MSE is that of the uncorrected quantizer. The MSE will increase with η according to (8.31), but not any further than to MSE_Q.

8.4 Random Input Noise

Errors in the transfer function that are deterministic can be compensated for. The results of the previous sections show how successful this compensation is, taking the DNL of the quantizer and the precision of correction values into account. One error effect of a practical ADC that *cannot* be compensated for using look-up tables[2] is random noise. Since the noise is truly random, it is impossible to say anything about it, even with knowledge of the resulting output signal. Hence, it cannot be compensated for.

The random noise is modeled as an additive noise, with variance σ_n^2, added to the input of the quantizer. As a natural consequence, the MSE of the output is increased by σ_n^2.

8.5 Combining the Theories

The performance description provided in (8.31) above is dependent on MSE_o – a quantity that is dependent on the actual transfer characteristics of the ADC under test, the accuracy of the calibration and correction schemes, and on the signal considered. One way to obtain MSE_o is of course to test it in practice – that is calibrate an infinite-precision LUT and use it to evaluate the resulting MSE after correction. This is, however, cumbersome in many situations. It would therefore be interesting to find an expression for MSE_o that could be used to estimate the resulting performance of a post-corrected ADC before it was calibrated.

In this section, the theories presented in Sections 8.2–8.4 are combined to form a joint formula for the resulting performance after correction, as a function of DNL, LUT resolution and random noise.

[2]Random noise can be compensated using oversampling and averaging techniques, but this limits the input frequency range.

Random DNL

Recall from (8.18) that the MSE after correction using perfect (infinite-precision) correction values could be expressed as

$$\mathrm{MSE}(\sigma_D^2) = \kappa\,\mathrm{MSE}_Q = \left(1 + 3\frac{\sigma_D^2}{\Delta^2}\right)\mathrm{MSE}_Q. \tag{8.33}$$

Here, σ_D^2 was the variance of the DNL, and Δ was the nominal quantization bin width.

In order to get a value for MSE_Q we make the assumption that the quantization step size Δ of the ADC is small compared with the variability of the source PDF and that the input signal does not overload the quantizer – i.e., the requirements for high-rate quantization are fulfilled. Then, MSE_Q is the classical result

$$\mathrm{MSE}_Q = \mathrm{MSE}_{\text{uniform}} = \frac{\Delta^2}{12}. \tag{8.34}$$

Now, inserting (8.34) into (8.33), and this in turn into (8.31), gives the resulting MSE after correction with a fixed-point LUT. We also account for a random noise with variance σ_n^2 as in Section 8.4. The result is

$$\begin{aligned}\mathrm{MSE}_\eta(\sigma_D^2, \sigma_n^2) &= \left(1 + 3\frac{\sigma_D^2}{\Delta^2}\right)\frac{\Delta^2}{12} + \frac{\eta^2\Delta^2}{12} + \sigma_n^2 \\ &= \frac{\Delta^2}{12} + \frac{\sigma_D^2}{4} + \frac{\eta^2\Delta^2}{12} + \sigma_n^2.\end{aligned} \tag{8.35}$$

The MSE consists of four terms: the first term is the MSE of the ideal uniform quantization, the second term is the error inflicted by the DNL, the third term is the effect of limited-resolution correction values, and the fourth term is the input random noise. Note that the result is in squared input units, e.g., V^2. The resulting MSE in LSB^2 is obtained simply by dividing the equation by Δ^2.

SINAD

When characterizing ADCs the SINAD is more frequently used than the MSE. It is therefore interesting to state the results obtained above in terms of SINAD instead of MSE. The SINAD is defined as (1.7)

$$\mathrm{SINAD} = 20\log_{10}\frac{\mathrm{RMS}_{\text{signal}}}{\mathrm{RMS}_{\text{noise}}} \quad [\mathrm{dB}]. \tag{8.36}$$

The SINAD is in most cases tested using a sinewave signal. Let the amplitude of the test signal be A_{dBFS}, expressed in dB relative full scale. Hence, the RMS value is then

$$\mathrm{RMS}_{\text{signal}} = \frac{\Delta\, 10^{\frac{A_{\mathrm{dBFS}}}{20}}\, 2^{b-1}}{\sqrt{2}}. \tag{8.37}$$

The RMS noise amplitude is obtained from the MSE expression (8.35) above so that

$$\mathrm{RMS}_{\mathrm{noise}} = \sqrt{\mathrm{MSE}_{\eta}(\sigma_D^2, \sigma_n^2)}. \tag{8.38}$$

We obtain the expression

$$\begin{aligned}\mathrm{SINAD}_{\mathrm{DNL}} = {} & 20\,b\,\log_{10} 2 + 10\,\log_{10}\frac{3}{2} + A_{\mathrm{dBFS}} \\ & - 10\,\log_{10}\left(1 + 3\frac{\sigma_D^2}{\Delta^2} + \eta^2 + 12\frac{\sigma_n^2}{\Delta^2}\right)\end{aligned} \tag{8.39}$$

for the resulting SINAD in dB. Note that A_{dBFS} must be negative for this expression to be valid. If not, the quantizer is overdriven, and the MSE in (8.35) is no longer accurate.

ENOB

The effective number of bits (ENOB) is defined in (1.9) as a function of the SINAD. Let the amplitude $A = 10^{A_{\mathrm{dBFS}}/20}\,\mathrm{FSR}/2$ and insert (8.39) into (1.9) to obtain

$$\mathrm{ENOB}_{\mathrm{DNL}} = b - \frac{1}{2}\log_2\left(1 + 3\frac{\sigma_D^2}{\Delta^2} + \eta^2 + 12\frac{\sigma_n^2}{\Delta^2}\right). \tag{8.40}$$

Optimal Quantization

The results above are based on the assumption that the deviations from the ideal quantizer transfer function all deteriorate the performance. However, if we want to be rigorous we must consider the (unlikely) possibility that the quantization regions actually deviate from the uniform quantizer to a configuration which is *more beneficial for the considered test signal*. We will therefore resort to information theory and the results on optimal quantization to derive a true lower bound for the MSE of a perfectly corrected ADC. We will derive a result based on the *Shannon lower bound* which is a bound on the so called *rate–distortion function*.

From information theory (see e.g., [Gra90]) we learn that the rate–distortion function tells us how small the resulting distortion can be when describing the outcomes of a certain random variable with a specific rate (resolution). The inverse *distortion–rate function* provides the reverse relation[3]. In numerous situations the rate–distortion function is inherently difficult to calculate, therefore, the Shannon lower bound on the rate–distortion function is frequently used. The lower bound has the advantage that it is often easier to compute.

If we are quantizing a random variable S and using a squared-error criterion (MSE), the Shannon lower bound is defined as

$$R_{\mathrm{SLB}}(D) = h(S) - \frac{1}{2}\log_2(2\pi e D), \tag{8.41}$$

[3]The distortion–rate function is the inverse of the rate–distortion function whenever the latter is strictly decreasing.

where $h(S)$ is the differential entropy of S, D is the squared-error distortion and R_{SLB} is the rate in bits. The result says that it is impossible to represent a random variable S with less than R_{SLB} bits if the MSE should be no more than D.

Now, since (8.41) is a lower bound on the rate–distortion function and is strictly decreasing, the inverse of (8.41) – D as a function of R – is a lower bound on the distortion–rate function, which is of greater interest to us. We get

$$D = \frac{1}{2\pi e} 2^{2h(S)-2R}. \tag{8.42}$$

The differential entropy

$$h(S) \triangleq -\int f_S(s)\, \log f_S(s)\, ds = -\,\mathrm{E}[\log f_S(S)] \tag{8.43}$$

is a function of the distribution of S. Therefore we cannot say anything more about the lower bound before we choose a PDF for S. In this case, we let S be a sample function of a sinusoid with amplitude A, mainly because it is the predominant test signal in ADC testing and because the result obtained can be compared with (8.39). The PDF of S is then given by

$$f_S(s) = \frac{1}{\pi A \sqrt{1 - \left(\frac{s}{A}\right)^2}}, \quad |s| < A, \tag{8.44}$$

and the differential entropy (in bits) can be shown to be

$$h(S) = \log_2\left(\frac{\pi A}{2}\right). \tag{8.45}$$

Inserting this result into (8.42) we get the lower bound

$$D = \frac{\pi A^2}{16} 2^{-2R}. \tag{8.46}$$

Using this last result we can obtain a lower bound on the distortion when quantizing sinusoids. For instance, when quantizing a sinusoid, having A_{dBFS} amplitude, with a b-bit quantizer, the amplitude is $A = \Delta\, 10^{\frac{A_{\mathrm{dBFS}}}{20}}\, 2^{b-1}$ input units (cf. (8.37)) and the rate is $R = b$ (bits). The squared-error distortion can in this case never be lower than

$$D_{\mathrm{SLB}} = \frac{\pi}{64}\, \Delta^2\, 10^{\frac{A_{\mathrm{dBFS}}}{10}}. \tag{8.47}$$

Note that D_{SLB} is a *lower bound* on the distortion of a quantizer *tailored* for a sinusoid input.

Finally, inserting D_{SLB} from (8.47) as MSE_o in (8.31), we obtain

$$\mathrm{MSE}_{\eta,\,\mathrm{SLB}} = \frac{\pi}{64}\, \Delta^2\, 10^{\frac{A_{\mathrm{dBFS}}}{10}} + \frac{\eta^2}{12}. \tag{8.48}$$

This result is a lower bound on MSE for $\eta = 0$, but is also likely to be lower than the practically achievable MSE for $\eta > 0$. Note, however, that the discussion on asymptotic results in Section 8.3 is valid in this case too. Therefore, the practical MSE will not increase above MSE_Q, although the formula (8.48) does, implying that $\mathrm{MSE}_{\eta,\,\mathrm{SLB}}$ can for sure *not* be a lower bound for all η. It can also be noted that the random input noise of Section 8.4 was not added in (8.48). If random noise at the input of the quantizer should be accounted for in this case, the distribution of the noise must be known. The differential entropy must then be calculated, not for the sinusoid only, but for the resulting PDF after adding the sinusoid to the random noise.

SINAD

Once again, the MSE result obtained is transferred to a SINAD expression. This time, the MSE from (8.48) is inserted into (8.38) to obtain the expression

$$\mathrm{SINAD}_{\mathrm{SLB}} = 20\,b\,\log_{10} 2 + A_{\mathrm{dBFS}} - 10\,\log_{10}\left(\frac{\pi}{8}\,10^{\frac{A_{\mathrm{dBFS}}}{10}} + \frac{2\eta^2}{3}\right), \qquad (8.49)$$

also in dB. Comparing with (8.39), we see that $\mathrm{SINAD}_{\mathrm{SLB}} > \mathrm{SINAD}_{\mathrm{DNL}}$ for all $A_{\mathrm{dBFS}} < 0$, arbitrary σ_n^2 and σ_D^2, and for all η.

8.6 Simulations

The results derived in this chapter has been tested in simulation experiments. Two different mathematical models for ADCs have been used to verify the results. The experiments and the outcome of them are described in the following two sections.

Both experiments have the same overall structure, which is described first. The resolution of the quantizers are in both cases $b = 10$ bits. The following steps are common to both simulations:

1. Calibrate a LUT. A sinusoid with amplitude –1 dBFS and random initial phase is used as input to the ADC model. The normalized frequency (f/f_s) is selected to $3001/16384 \approx 0.1832$, i.e., the conditions for coherent sampling are fulfilled for a record of 16 384 samples, which is the size of the record taken.

2. A sinewave fit is made to the recorded data, as per the IEEE standard [Std1241]. The fitted sinewave is used as reference signal and a correction table is built (cf. Section 7.4). The correction table is static, i.e., having one correction term per ADC code level. The correction terms are stored in floating-point precision.

3. Evaluate the correction. 16 sinusoid test signals are generated, each having amplitude and frequency as above and random initial phase. These are used

as input to the ADC model and the resulting output is corrected using the LUT. The performance in terms of mean SINAD over the 16 sequences is calculated.

4. The LUT correction values are quantized to lower precision η, and the evaluation step 3 is repeated for different values of η.

Random DNL Model

In this first experiment, the model was a simple quantizer where the widths of the quantization regions where randomly altered from an ideal uniform configuration. The input signal was first perturbed by additive Gaussian noise, with zero mean and variance $\sigma_n^2 = 0.05$ LSB^2. The subsequent quantizer had a DNL that for each code level was generated as an independent observation of a zero-mean Gaussian random variable with variance $\sigma_D^2 = 0.002$ LSB^2. The resulting DNL is shown in Figure 8.4.

The evaluation procedure outlined above is performed on the quantizer model. Figure 8.5 shows the output spectrum of the uncorrected quantizer, evaluated using the 16 sinusoids in step 3. Finally, Figure 8.6 shows the resulting SINAD after correction with varying precision η. The graph shows the experimental results ('o'), the theoretical result $\text{SINAD}_{\text{DNL}}$ as predicted in (8.39) (solid line), and the upper limit $\text{SINAD}_{\text{SLB}}$ in (8.49) (dashed line). The graph also shows two horizontal lines, where the upper ('$\triangle$') shows the SINAD after correction with infinite precision, and the lower ('∇') shows the SINAD of the uncorrected ADC model.

We see from the results that the theoretical line $\text{SINAD}_{\text{DNL}}$ aligns well with the experimental results, up to $\delta b = 0$. For poorer resolution than that, i.e., for $\delta b < 0$ implying $\eta > 1$ LSB, the experimental SINAD approach that of the uncorrected ADC. This is in perfect accordance with the discussion in Section 8.3, where it was argued that the performance would not be worse than that of the uncorrected

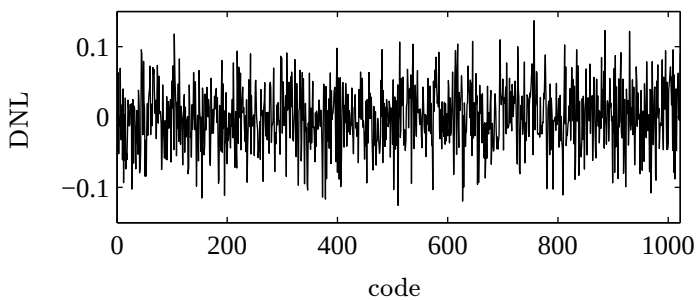


Figure 8.4: DNL of the random DNL model. The DNL curve was generated as independent realizations of a zero-mean Gaussian random variable with variance $\sigma_D^2 = 0.002$ LSB^2.

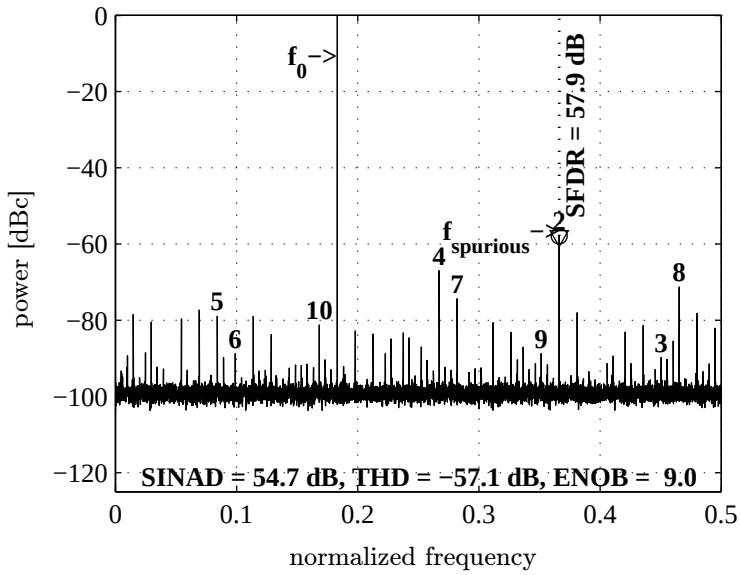


Figure 8.5: The average output spectrum of the 16 test signals before correction.

ADC. It does also make sense that a table resolution $\eta \geq 2$ LSBs does not provide much improvement, since the vast majority of the table values in this case had a magnitude less than 1.

It is also evident that $\text{SINAD}_{\text{SLB}}$ is an upper bound in the "active" region, i.e., the region where the experimental SINAD is larger than the uncorrected SINAD.

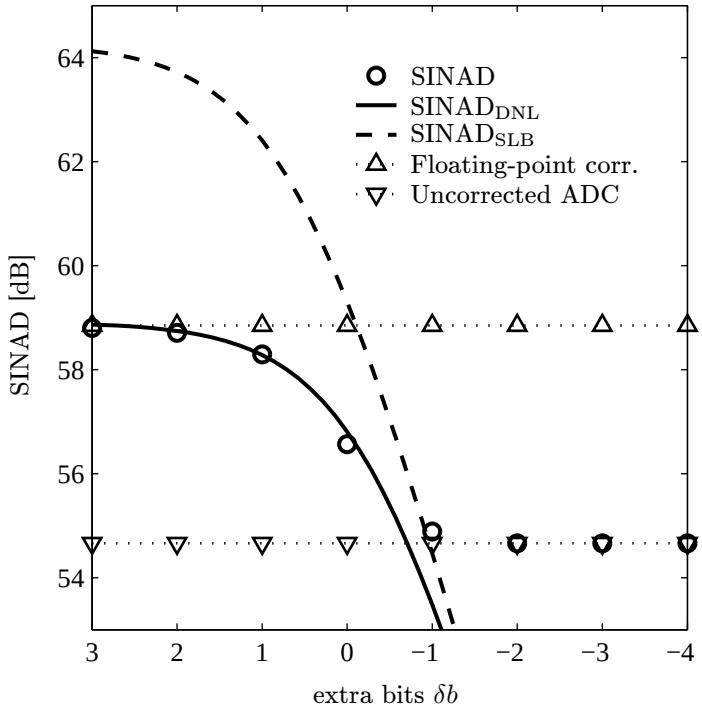


Figure 8.6: The figure shows SINAD results for different correction value resolutions η. The experimental results using the random DNL model are shown using circles ('o'), while the solid line shows the predicted results according to (8.39). The upper bound from (8.49) is shown in dashed line. Finally, '$\triangle$' and '∇' show the results of infinite precision correction and no correction, respectively.

Polynomial Model

In this model, the transition levels of the quantizer have been altered according to a fourth degree polynomial. The rationale for simulating with this model is to test whether the whiteness assumption for the DNL made in Section 8.2 is mandatory.

The model is defined as following: If the ideal value for the k-th transition level is T_k^o, then the actual k-th transition level is

$$T_k = T_k^o + C_1 k^4 + C_2 k^3 + C_3 k^2, \quad k = 1, 2, \ldots, 2^b - 1. \tag{8.50}$$

The coefficients were set to

$$C_1 = 2.958 \cdot 10^{-10},$$
$$C_2 = -4.512 \cdot 10^{-7},$$

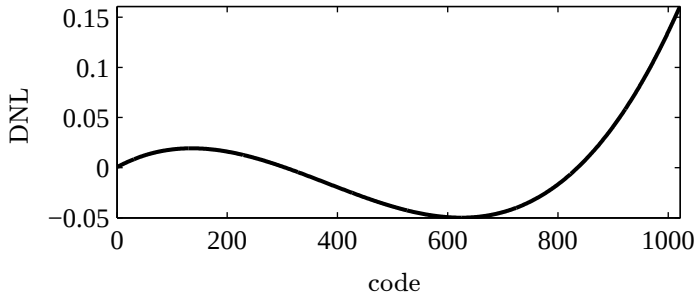


Figure 8.7: DNL of the polynomial model.

and

$$C_3 = 1.520 \cdot 10^{-4},$$

where T_k and T_k^o is expressed in LSBs. The resulting DNL is shown in Figure 8.7. The values were chosen so that the mean-squared DNL is equal to 0.002 LSB^2, i.e., the same as the variance σ_D^2 in the random DNL model above. Also in this model, the input signal is perturbed by an additive Gaussian noise, with zero mean and variance $\sigma_n^2 = 0.05$ LSB^2

The evaluation procedure outlined on page 102 is again performed, now with the polynomial model. Figure 8.8 shows the output spectrum of the uncorrected quantizer, evaluated using the 16 sinusoids in step 3. Figure 8.9 shows the resulting SINAD after correction with different precision η.

Again, the results show a very good match between the theoretical values $\text{SINAD}_{\text{DNL}}$ and the experimental SINAD. Just as with the random DNL model, the experimental SINAD levels out at the SINAD of the uncorrected ADC. This time, however, the performance of the uncorrected ADC is worse. An effect of this is that even a resolution as coarse as $\eta = 4$ LSBs gives a significant increase in performance. This is of course because the values of the correction table have a larger span in this case. We can also conclude that the whiteness assumption on the DNL seems to be unnecessary, since the agreement between theory and simulation is just as good as in the simulation with random DNL.

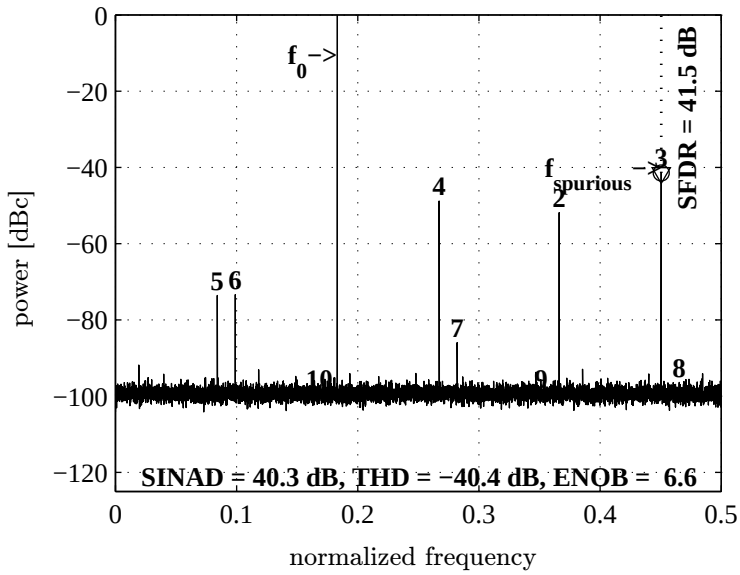


Figure 8.8: The average output spectrum of the 16 test signals before correction.

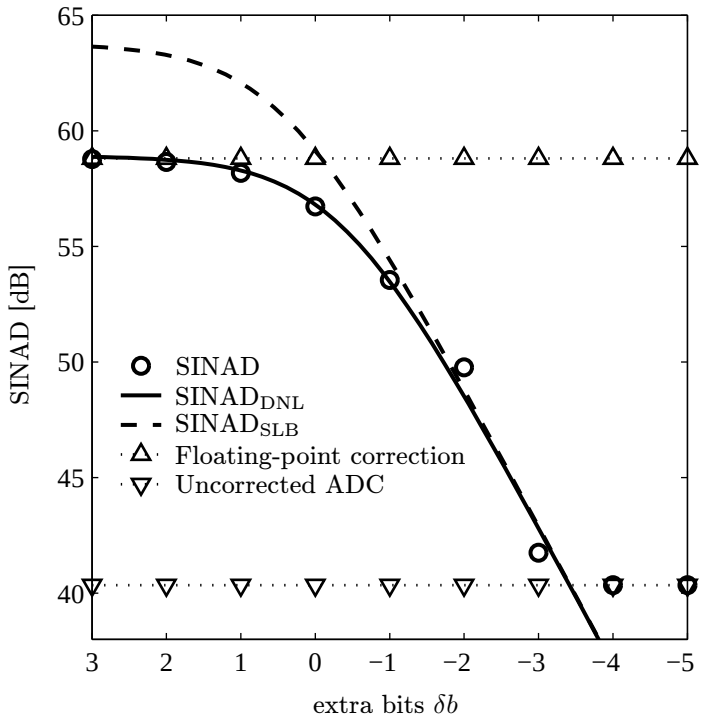


Figure 8.9: Same contents as in Figure 8.6, but here using the polynomial ADC model.

8.7 Results Using Experimental ADC Data

The theories presented in this chapter have also been evaluated using experimental ADC data, acquired from an Analog Devices AD9430, 210 MSPS, 12 bit converter. The data is described in detail in Appendix B.3.

From the measurements the following parameters were estimated:

- Random noise variance: $\sigma_n^2 \approx 0.5374$ LSB^2.
- Variance of DNL: $\sigma_D^2 \approx 0.004206$ LSB^2.

With these values we can estimate the performance in terms of SINAD of the converter after correction using the formula (8.39).

An LUT is calibrated and used for correction. The procedure is quite similar to that used in the simulations of Section 8.6:

1. Calibrate a LUT. A sinusoid with amplitude –0.5 dBFS is used as input to the ADC model. The frequency is selected to 60 124 547 Hz, so that the conditions for coherent sampling are fulfilled for a record of 65 536 samples. 31 such sequences are recorded. (The samples are in fact recorded in one long sequence of 2 097 152 samples, subsequently split into 31 sequences.)

2. A three-parameter sinewave fit is made to the recorded data, as per the IEEE standard [Std1241]. The fitted sinewave is used as reference signal and a correction table is built (cf. Section 7.4). The correction table is static, i.e., having one correction term per ADC code level. The correction terms are stored in floating point precision.

3. Evaluate the correction. The same sinusoid signal as in step 1 is applied to the ADC. 31 sequences with 65 536 samples each are recorded. The resulting output is corrected using the LUT. The performance in terms of mean SINAD over the 31 sequences is calculated.

4. The LUT correction values are quantized to lower precision η, and the evaluation step 3 is repeated for different values of η.

Figure 8.10 shows the results using the experimental ADC data. It is clear that the formula (8.39) overestimates the resulting SINAD after correction. It seems that either the noise variance or the DNL variance is estimated too low. Therefore, an alternative method was used to estimate the random noise parameter σ_n^2. The method is described in Appendix B.3 on page 181, and the resulting estimate was $\sigma_n^2 \approx 0.8092$ LSB^2. Using this new value of the random noise variance, the results of Figure 8.11 were obtained. We see now a good match between the experimental results and the predicted value from (8.39).

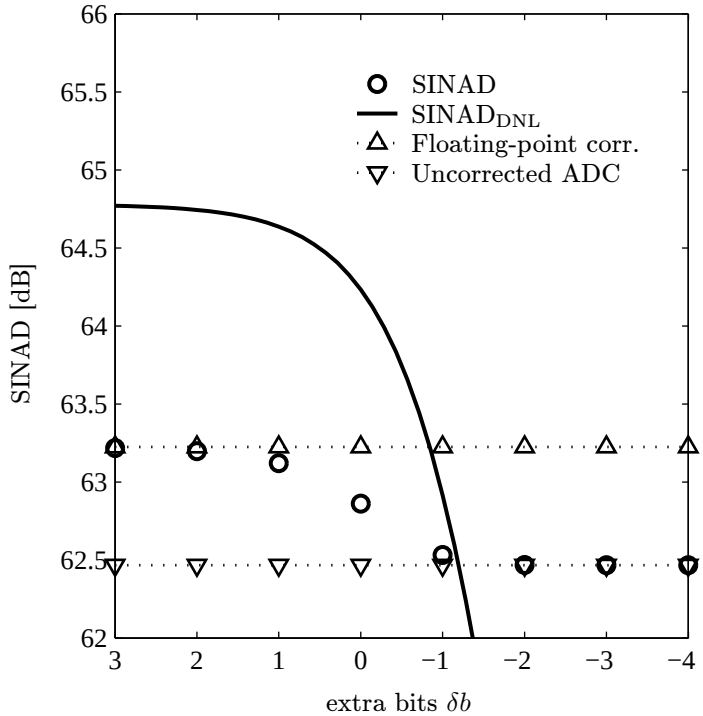


Figure 8.10: The figure shows SINAD results for different correction value resolutions η. The results using data from an AD9430 are shown using circles ('o'), while the solid line shows the predicted results according to (8.39). Finally, '$\triangle$' and '∇' show the results of infinite precision correction and no correction, respectively.

8.8 Discussion

The results obtained in the simulations of Section 8.6 seemed to verify the theories derived in this chapter. It seemed that knowing the random noise variance σ_n^2 and the variance of the DNL σ_D^2 was sufficient to accurately predict the outcome of a post-correction, in terms of SINAD or ENOB. The first results obtained using experimental ADC data did, however, not verify this theory. The results of Figure 8.10 indicate that either σ_n^2 or σ_D^2 were too low. On the other hand, estimating the noise using a 60 MHz –0.5 dBFS sinewave as input – instead of a 20 kHz triangle wave with an amplitude of approximately –54 dBFS – resulted in a higher noise estimate for the random noise σ_n^2. The results shown in Figure 8.11 indicate that this estimate is more consistent with the actual noise corrupting the input signal.

The most likely explanation for this is that σ_n^2 in (8.39) should not be the variance of the random input noise (or thermal noise) only, but should also include

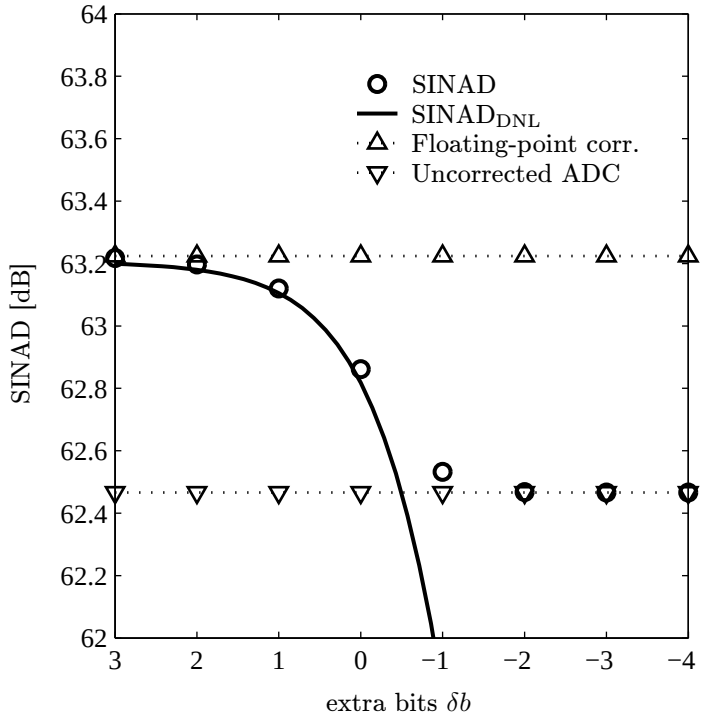


Figure 8.11: The figure shows SINAD results for different correction value resolutions η. The setting is the same as for Figure 8.10, but with an alternative estimate for the variance of the random noise σ_n^2 resulting in a different curve for $\text{SINAD}_{\text{DNL}}$.

other random errors in the sampled signal. One significant source of random errors is aperture jitter, especially as the signal frequency increases. Thus, it is likely that the higher noise estimate obtained using a high-frequency large-scale sinewave signal also incorporates errors induced by aperture jitter.

Part III

Generalized Look-Up Table Post-correction

Chapter 9

Results on ADC Correction Using LUTs

In Chapter 3, an overview of the most common LUT methods was given. The methods were (cf. Section 3.2):

Static The LUT is addressed using only one sample from the ADC output. Figure 3.4 shows two implementations of static correction.

State-space The table is addressed using a combination of two consecutive output samples from the ADC. Figure 3.2 depicts a state-space correction system.

Phase-plane (Also called code-slope.) The LUT is addressed using one output sample and (an estimate of) the slope corresponding to this sample. Figure 3.3 shows this type of correction.

In this second part of the thesis we will focus most of our attention to a specific generalized look-up table post-correction system. In this chapter, the method is defined and exemplary results using experimental ADC data are presented. In Chapter 10 an analysis tool related to the proposed method is derived. The analysis tool is applied in a few optimization scenarios in Chapter 11. Finally, a suboptimal method to solve the optimization problem posed is discussed in Chapter 12.

The motivation for the proposed scheme is twofold: First, the errors of an ADC can exhibit substantial dependence on the dynamics of the input signal, wherefore a static or even a state-space or phase-plane correction method might fall short. Introducing more samples into the table index will provide more information about the signal dynamics. Second, the method proposed in this chapter will provide efficient means of controlling the amount of memory required by a LUT.

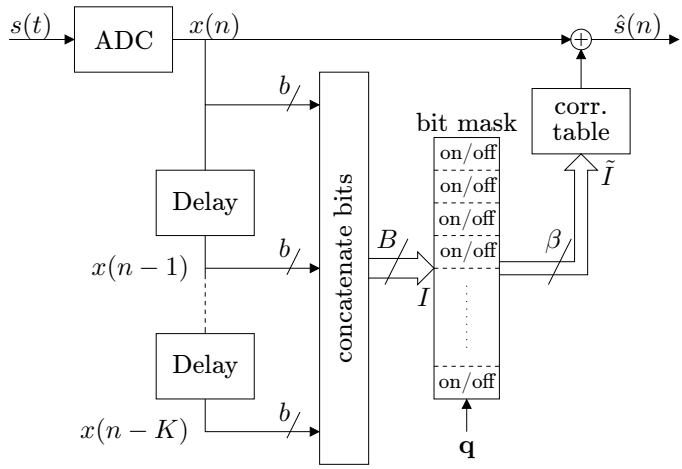


Figure 9.1: Dynamic post-correction system outline. Since the ADC errors sought to mitigate are dependent on signal history, the correction is also history dependent through the use of K delay elements. In order to reduce the index size (and thereby the memory requirements), a subset of β samples are selected to address the correction table. The bit mask vector $\mathbf{q}$ is a mathematical construct included to facilitate the performance analysis.

9.1 The Generalized LUT Post-correction Method

The method proposed in this thesis is a generalized extension of the look-up table correction methods listed above. The correction system is outlined in Figure 9.1. Previous work has alluded to the possibility of extending the look-up table to more than two dimensions, e.g., [RI87], [DVBS92] and [Tsi95]. In the state-space method in Section 3.2, the present and one-step delayed samples are used to address a two-dimensional table. Here, the present sample, $x(n)$, is used together with K delayed samples, $x(n-1)$ through $x(n-K)$, to address a $(K+1)$-dimensional table. This is illustrated in Figure 9.1 with the $K+1$ address samples being concatenated together to form the table index I of size $B = K\,b + b$ bits. With more table dimensions, it is presumed that a better estimate of the present error can be produced. This is based on the assumption that the ADC error depends on the signal dynamics, and that a higher-order state-space structure can describe dynamic features more accurately.

One major drawback, also identified in the previous work where multidimensional tables have been suggested, is that the size of the table, equating to 2^B, quickly grows out of hand with increasing K. For a 10-bit converter and 8-bit correction words, a static correction requires a modest 2^{10} bytes, or 1 kB. Meanwhile, a 2-dimensional state-space table ($K = 1$) requires 1 MB, and expanding yet another

dimension to $K = 2$ amounts to 1 GB. The proposed countermeasure is to reduce the address space by using only a subset of the b available bits in each sample. One way to accomplish this is to apply further quantization to the delayed samples, so that they are represented with *less* than b bits resolution (used in [TMBSL02] and similar to the method used in [RI87] and [DVBS92] in the context of phase-plane correction). Here, this approach is *generalized* to say that a number less than or equal to b bits are used from the sample $x(n-k)$ (for $k \in \{0, 1, \ldots, K\}$). However, these are not necessarily the most significant bits but can be selected from *all* b bits of $x(n-k)$. That is, some of the bits in the sample $x(n-k)$ are masked away, and the remaining bits are used for addressing. This is illustrated in Figure 9.1 with the B-bit concatenated address I being bit-masked into a β-bit address $\tilde{I}$, where β is an integer less than (or equal to) B.

In the bit-masking process the (column) vector $\mathbf{q} \in \{0, 1\}^B$ is introduced. It has exactly β ones and the remaining $B - \beta$ entries are zero. A '1' in the i-th position dictates that the i-th bit of I should be propagated to the reduced-size index $\tilde{I}$, while a '0' implies that the corresponding bit should be masked out. That is, $\mathbf{q}$ can be viewed as a selection vector. This is illustrated in Figure 9.1 with 'on' and 'off' in the bit mask block. The order of the bits is preserved. The notation $I \xrightarrow{\mathbf{q}} \tilde{I}$ is used to denote that the index $\tilde{I} \in [0, 2^\beta - 1]$ is constructed from the β bits in I selected by the bit mask $\mathbf{q}$:

Definition 2. *Let* $\mathbf{q}$ *be a vector with elements* $\mathbf{q}_k$, $k = 1, 2, \ldots, B$, *out of which exactly* β *elements are equal to 1, and the remaining* $B - \beta$ *elements are 0. Let* $\mathbf{i}_1 < \mathbf{i}_2 < \cdots < \mathbf{i}_\beta$ *be the indices of those elements that are 1, i.e.,* $\mathbf{q}_{\mathbf{i}_j} = 1$ *for* $j = 1, 2, \ldots, \beta$. *Let* I *be an integer in* $[0, 2^B - 1]$, *with binary representation* $I = (I_B, I_{B-1}, \ldots, I_1)_2$. *Then, the mapping* $I \xrightarrow{\mathbf{q}} \tilde{I}$ *defines an integer* $\tilde{I}$ *with binary representation*

$$\tilde{I} = (I_{\mathbf{i}_\beta}, I_{\mathbf{i}_{\beta-1}}, \ldots, I_{\mathbf{i}_1})_2. \tag{9.1}$$

The integer $\tilde{I}$ *is in the range* $[0, 2^\beta - 1]$.

Elucidating example Consider a 4-bit quantizer and an index-building structure as in Figure 9.1. Assume that $K = 1$ and $\mathbf{q} = [1\,1\,0\,0\,1\,1\,0\,0]^{\mathrm{T}}$. Hence, $\beta = 4$. Assume that at time n we have the output $x(n) = 11 = 1011_2$ from the quantizer, and the previous sample was $x(n-1) = 4 = 0100_2$. Concatenating the bits gives the full 8-bit index $I = 1011\,0100_2 = 180$. The bitmask defined by $\mathbf{q}$ would now select the two most significant bits from the two samples, so that the 4-bit index $\tilde{I} = 1001_2 = 9$.

To conclude, the proposed method reduces the memory size by using only a subset of the available bits for addressing, but still takes information from K delayed samples into account.

Vector Quantizer Interpretation

The production of the concatenate index I can be described in a framework similar to that of *vector quantization* (e.g., [GG92]). Assume that the ADC is ideal, thus operating as described in Section 1.2. Define the $(K+1)$-dimensional vector of input samples

$$\mathbf{s}(n) = [s(n)\ s(n-1)\ \ldots\ s(n-K)]^{\mathrm{T}}. \tag{9.2}$$

Now, the production of the index I in Figure 9.1 can be described using a partition of the (K+1)-dimensional real space into disjoint sets $\{\boldsymbol{\mathcal{S}}_J\}_{J=0}^{2^B-1}$. The partition is such that the index I is produced if $\mathbf{s} \in \boldsymbol{\mathcal{S}}_I$, where I is the concatenation of the b-bit quantized values[1] $x(n)$ through $x(n-K)$. This description is a straightforward extension into higher dimension of the first step in the quantizer of Figure 1.1, i.e., the mapping from s to i where the index i is decided based on the sets $\{\mathcal{S}_j\}_{j=0}^{2^b-1}$. In the light of this framework, the bit-masking $I \xrightarrow{\mathbf{q}} \tilde{I}$ is simply the result of merging the sets $\boldsymbol{\mathcal{S}}_J$ into larger, and obviously fewer, sets and assigning them a new index $\tilde{I}$. This will be thoroughly discussed in Chapter 10, where also the higher-dimension representation introduced here will become useful.

Selecting a "good" bit mask $\mathbf{q}$ for a given delay K and a fixed table size 2^β is a non-trivial problem, indeed. A solution to this problem is one of the main contributions in Chapter 10. It should be made clear at this point that the bit mask vector $\mathbf{q}$ is a mathematical construct included to facilitate the performance analysis. It is likely that in a typical implementation of the correction structure, the address bit selection would be hardwired, after deciding upon a beneficial configuration.

Calibration of the Generalized LUT

The results of Section 3.4 can now be applied to the generalized post-correction system described above. The suggested calibration structure is depicted in Figure 9.2, where an external reference signal is used (cf. Chapter 3 for a discussion on different ways to obtain a reference signal). In this case we are interested in finding the MMSE optimal correction values as opposed to the optimal reconstruction values. (Figures 3.4(a) and 3.4(b) illustrate the difference.) In Section 6.2 the optimal correction values were given in (6.5)–(6.6). These results were derived for a quantizer without memory, i.e., where the produced index i at sample time n depends only on the present sample $s(n)$, but they are readily extended to the case where the table index I is a function of present and past samples, as in Figure 9.2.

By incorporating the multidimensional description of quantization introduced above, it is possible to express the optimal value for the correction table entry $\mathbf{e}_I$.

[1] In order to follow the presentation of the ideal quantizer in Section 1.2, and Figure 1.1 in particular, the integer index I should be the concatenation of the *intermediate* quantization indices $i(n)$ through $i(n-K)$. However, since the mapping from $i(n)$ to $x(n) = x_i$ is uniquely invertible for practical ADCs, it does not matter which notation we use. We can simply assume that the output samples $x(n)$ are represented as b-bit binary values, which is indeed the case in a practical ADC.

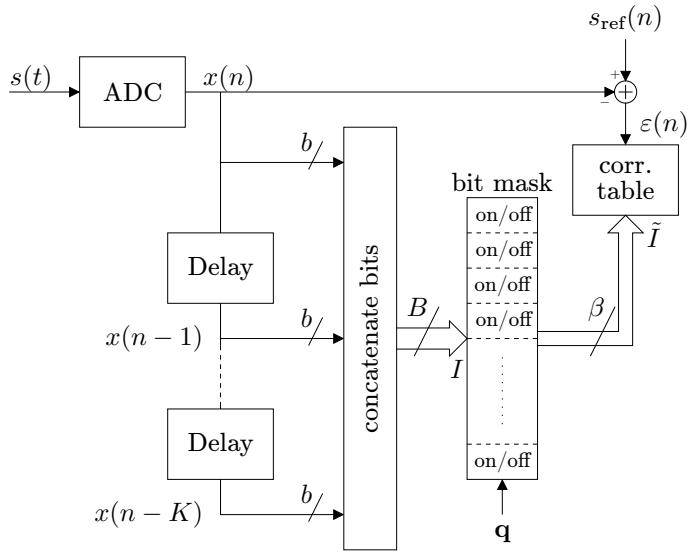


Figure 9.2: Outline of calibration system. The table is built by exercising the ADC with a calibration signal and comparing a reference measurement or an estimate of the calibration signal, $s_{\text{ref}}(n)$, with the output $x(n)$.

Let $f_{\mathbf{s}}(\mathbf{s})$ be the joint probability density function for the $(K+1)$-dimensional vector in (9.2). It is here assumed that the underlying signal is such that the joint PDF is independent of n; consequently, the n is omitted. Since the aim is to find the *correction* value, the function $e(\mathbf{s})$ (from $\mathbb{R}^{K+1}$ to $\mathbb{R}$) is introduced. The function is the error produced by the ADC, $s(n) - x(n)$, when $\mathbf{s}(n) = \mathbf{s}$. Ideally, this error is only the quantization error, but in a practical ADC it is more involved. These extensions to (6.5) result in the equation

$$\mathbf{e}_{I,\,\text{opt}} = \frac{\displaystyle\int_{\mathbf{s}\in\boldsymbol{\mathcal{S}}_I} e(\mathbf{s})\, f_{\mathbf{s}}(\mathbf{s})\, d\mathbf{s}}{\displaystyle\int_{\mathbf{s}\in\boldsymbol{\mathcal{S}}_I} f_{\mathbf{s}}(\mathbf{s})\, d\mathbf{s}} \tag{9.3}$$

for the optimal table entries. Recall that the disjoint sets $\{\boldsymbol{\mathcal{S}}_J\}_{J=0}^{M-1}$ define the mapping $\mathbf{s} \to I$ by partitioning the space $\mathbb{R}^{K+1}$.

Equation (9.3) gives the optimal value for the correction table entries, but requires full knowledge of both the actual quantization regions $\{\mathcal{S}_k\}_{k=0}^{M-1}$ and the PDF $f_{\mathbf{s}}(\mathbf{s})$ of the calibration signal. The situation is analogous to that of Chapter 7, where two estimators for the optimal correction terms, based on a reference signal $s_{\text{ref}}(n)$, were proposed. In this chapter we use the sample mean (7.9) of Section 7.4. The estimator was shown to be optimal when the reference signal associated with a certain output code was modeled as a Gaussian random variable. In Section 3.4

it was shown that the sample mean can also be motivated even without knowledge of the reference signal distribution. The derivation was made in (3.8)–(3.10) by approximating the integrals of (3.7) (or (6.5)). Here, the sample mean approach is extended to multiple dimensions. Similar approximations as in (3.8)–(3.10) can be made to (9.3), resulting in the conclusion that $\mathbf{e}_{I,\,\mathrm{opt}}$ should be estimated as the mean of all errors observed while the index I was produced. Observed error is simply the difference between a reference signal $s_{\mathrm{ref}}(n)$ and the ADC output $x(n)$. In the bit-masked case, we simply say that the $\tilde{I}$-th correction value $\mathbf{e}_{\tilde{I}}$ is estimated as *the mean of all errors observed at times when the output from the index building was* $\tilde{I}$. For now we make this statement as a heuristic conclusion drawn from the above results, but the validity of this statement will be proved in Chapter 10.

The calibration of the $\tilde{I}$-th table entry can be implemented as a running average: Let $\tilde{\mathbf{e}}_{\tilde{I}}$ be assigned the initial value zero for all $\tilde{I}$; when no calibration information is available it makes sense to estimate the error with zero. Let the calibration signal $s_{\mathrm{cal}}(t)$ and an estimate thereof, $s_{\mathrm{ref}}(n)$, be available for $n \geq 0$. Define $\mathbf{a}_{\tilde{I}}(m)$ as the number of times the index has been equal to $\tilde{I}$ for $0 \leq n < m$. Then, assuming that the index is $\tilde{I}$ at time n, update

$$\tilde{\mathbf{e}}_{\tilde{I}} \to \frac{\mathbf{a}_{\tilde{I}}(n)\,\tilde{\mathbf{e}}_{\tilde{I}} + s_{\mathrm{ref}}(n) - x(n)}{\mathbf{a}_{\tilde{I}}(n) + 1}. \tag{9.4}$$

After calibration is completed, the vector $\mathbf{a}$ will represent the distribution, or 'hit-rate', of the calibration signal over the correction table. Referring to the notation introduced in Section 3.4 we can conclude that the i-th element $\mathbf{a}_i$ will be the number of elements in the set $\mathcal{C}_i$, i.e., $\mathbf{a}_i = N_i$.

9.2 Exemplary Results

In this section the generalized post-correction method presented in Section 9.1 is evaluated. The evaluation is performed using experimental ADC data from an Analog Devices AD876 commercially available converter. The converter is a 10-bit pipelined flash converter designed to operate at 20 MSPS. The ADC data was acquired using a test bed which is described in Appendix A.1, and the properties of the recorded data is described in Appendix B.1.

Fixed-Point Representation of Table Entries

Naturally, the precision with which the correction table entries are represented will affect the performance of the correction. This was thoroughly investigated in Chapter 8. All forthcoming results in this work have been obtained using 8 extra bits of resolution in the correction values; the correction value resolution is $2^{-8} = 1/256$ LSBs. According to the results of Chapter 8 this will yield virtually the same results as an infinite-precision correction would.

Table 9.1: Correction results for some exemplary configurations. The results are presented as mean improvement over the entire Nyquist range. Each configuration represent a unique bit mask.

	Configuration	Improvement [dB]		
		SFDR	SINAD	THD
1	$K = 0$, 10-bit index	10.6	4.2	−10.0
2	$K = 1$, 20-bit index	12.4	2.6	−9.5
3	$K = 1$, 10-bit index	13.6	4.4	−10.6
4	$K = 1$, 5-bit index	11.3	4.1	−9.6
5	$K = 1$, 10-bit index	11.1	4.0	−9.8
6	$K = 1$, 5-bit index	10.6	3.9	−9.4
7	$K = 1$, 10-bit index	11.9	4.1	−9.7
8	$K = 1$, 5-bit index	10.5	3.8	−8.7
9	$K = 4$, 10-bit index	12.9	3.9	−9.3
10	$K = 4$, 18-bit index	20.3	5.4	−17.9

Correction Results

Some results of the correction method described above are presented here. The vast amount of different configuration possibilities when varying the number of delay elements K and the bit mask $\mathbf{q}$ makes it virtually impossible to make an exhaustive evaluation. Moreover, the results of the various configurations are most certainly different for different ADC models. Therefore, only a few exemplary configurations have been evaluated. For each configuration tested the table $\tilde{\mathbf{e}}$ has been calibrated according to the structure in Figure 9.2, using several near full-scale sinewave calibration signals at different frequencies. Then, the performance was evaluated with near full-scale sinewaves at several frequencies, separate from those used for calibration in order to evaluate the wide-band performance of the converter. Three performance measures have been used: SFDR, SINAD and THD (see Section 1.4). The results are presented in Table 9.1.

The first configuration is a standard static table, using all 10 bits in $x(n)$ and no delay elements. The second configuration is a state-space table, i.e., one delay element and 20 index bits. Configurations 3–8 also utilize one delay element, but select only a subset out of the 20 available bits in $x(n)$ and $x(n-1)$. Finally, configurations 9 and 10 are higher order tables with four delay elements and 10 and 18 index bits, respectively. When considering the 10-bit tables (1, 3, 5, 7 and 9) we see that the performance can be improved from that of a static correction without increasing the index size, i.e., without increasing the memory size. The SFDR is in fact increased with up to 3 dB. Configuration 10 is outstanding in all three measures; in this configuration the 18 bits are distributed over $x(n)$ through $x(n-4)$, selecting all 10 bits from $x(n)$ and the two most significant bits from the delayed samples.

Another observation is that the improvement in SINAD is less than the improvement in SFDR and THD (note that improving performance corresponds to *decreasing* THD). This is likely an effect of the SINAD being more dependent on *stochastic* errors (noise) than the SFDR and THD, since the latter two measure harmonic distortion and spurs often resulting from deterministic errors.

From the results in Table 9.1 we can conclude that the configuration employed has a significant impact on the performance of the corrected ADC, and that the allocation of bits in the bit mask is a non-trivial problem (even though one skilled in the art may have a partial intuition on the topic). This conclusion motivates the theoretical analysis to be performed in Chapter 10.

Chapter 10

Bit Allocation Analysis

In this chapter we focus our attention on the generalized correction scheme for ADCs presented in Chapter 9. We saw in the exemplary results of Section 9.2 that the actual choice of the number of delay elements and bit mask greatly influenced on the performance after correction. In this chapter, an analysis of the effect of a specific bit mask is derived. The analysis is based on the *Hadamard transform* of a vector, which is introduced in Section 10.1. In Section 10.2, the analysis is performed, and Section 10.3 concludes the chapter by explaining how the analysis tool is utilized in the correction table problem.

10.1 The Hadamard Transform

We begin this chapter by reviewing the *Hadamard transform* of a vector. The transform is useful because it provides an efficient means of analyzing the influence of a single bit in the vector index.

First, we introduce the Hadamard matrix (see e.g., [YH97, Lüt96]):

Definition 3 (Hadamard matrix)**.** *The Sylvester-type Hadamard matrix of order B is recursively defined through*

$$\mathbf{H}_B = \mathbf{H}_1 \otimes \mathbf{H}_{B-1}, \quad B > 1, \tag{10.1}$$

$$\mathbf{H}_1 = \begin{bmatrix} +1 & +1 \\ +1 & -1 \end{bmatrix}, \tag{10.2}$$

where $\otimes$ is the Kronecker matrix product.

The Kronecker product of two matrices $\mathbf{A}$ and $\mathbf{B}$, where $\mathbf{A}$ is of size m-by-n with elements a_{ij} and $\mathbf{B}$ of size p-by-q, is an mp-by-nq matrix defined as (see

e.g., [Lüt96, HJ91])

$$\mathbf{A} \otimes \mathbf{B} \triangleq \begin{bmatrix} a_{1\,1}\,\mathbf{B} & a_{1\,2}\,\mathbf{B} & \cdots & a_{1\,n}\,\mathbf{B} \\ a_{2\,1}\,\mathbf{B} & a_{2\,2}\,\mathbf{B} & \cdots & a_{2\,n}\,\mathbf{B} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m\,1}\,\mathbf{B} & a_{m\,2}\,\mathbf{B} & \cdots & a_{m\,n}\,\mathbf{B} \end{bmatrix}. \tag{10.3}$$

Accordingly, the matrix $\mathbf{H}_B$ in (10.1) is M-by-M square and symmetric, consisting of only ± 1. As before, $M = 2^B$. Furthermore, let $\mathbf{h}_i$, $i = 0, 1, \ldots, M-1$, be the columns of $\mathbf{H}_B$. Any two columns $\mathbf{h}_i$ and $\mathbf{h}_j$, $i \neq j$, are orthogonal, i.e., the inner product $\mathbf{h}_i^{\mathrm{T}}\,\mathbf{h}_j = 0$, and $\mathbf{h}_i^{\mathrm{T}}\,\mathbf{h}_i = M$. This results in $\mathbf{H}_B\,\mathbf{H}_B = M\,\mathbf{I}_M$, which implies that the inverse of $\mathbf{H}_B$ is $\mathbf{H}_B^{-1} = \frac{1}{M}\,\mathbf{H}_B$.

Another useful feature of the Sylvester-type Hadamard matrix is that each column $\mathbf{h}_i$ of the matrix can be explicitly calculated from the column index i. To make this to work out properly, the integer index $i \in [0,\, M-1]$ must be represented in a special binary format. Let the 'standard' binary (base-two) representation of i be $(i_B, i_{B-1}, \ldots, i_1)_2$, with all i_j either 0 or 1. The LSB is i_1 and the MSB is i_B so that $i = \sum_{j=1}^{B} i_j\, 2^{j-1}$. Now, define an *alternative* binary representation $(\bar{\imath}_B, \bar{\imath}_{B-1}, \ldots, \bar{\imath}_1)$ of i, where we let

$$\bar{\imath}_j = 1 - 2i_j = \begin{cases} 1, & i_j = 0, \\ -1, & i_j = 1. \end{cases} \tag{10.4}$$

That is, logical 'zero' is represented by $+1$ and logical 'one' is represented by -1. This is nothing but a re-mapping of the standard $0/1$ binary representation to a ± 1 representation. With this special binary representation, the following relationship, easily proven by verification, between i and the column $\mathbf{h}_i$ holds:

Lemma 11. *Let $(\bar{\imath}_B, \bar{\imath}_{B-1}, \ldots, \bar{\imath}_1)$ be the special ± 1-representation of the integer $i \in [0, M-1]$, such that*

$$i = \frac{2^B - 1}{2} - \sum_{j=1}^{B} \bar{\imath}_j 2^{j-2}. \tag{10.5}$$

Let $\mathbf{H}_B$ be the Hadamard matrix of order B with columns denoted $\mathbf{h}_i$; $\mathbf{h}_0$ is the first column and $\mathbf{h}_{M-1}$ is the last. Then,

$$\mathbf{h}_i = \begin{bmatrix} 1 \\ \bar{\imath}_B \end{bmatrix} \otimes \begin{bmatrix} 1 \\ \bar{\imath}_{B-1} \end{bmatrix} \otimes \cdots \otimes \begin{bmatrix} 1 \\ \bar{\imath}_1 \end{bmatrix}. \tag{10.6}$$

We are now ready to define the Hadamard transform:

Definition 4 (Hadamard transform of a vector)**.** *Let $\mathbf{e} = [\mathbf{e}_0\ \mathbf{e}_1\ \ldots\ \mathbf{e}_{M-1}]^{\mathrm{T}}$ be a column vector of size $M = 2^B$. The Hadamard transform of $\mathbf{e}$ is defined as*

$$\mathbf{t} \triangleq \frac{1}{M}\,\mathbf{H}_B\,\mathbf{e}. \tag{10.7}$$

Straightforward calculations of the Hadamard transform as it is written in (10.7) requires $\mathcal{O}(2^{2B}) = \mathcal{O}(M^2)$ arithmetic operations (not counting those required to build the matrix $\mathbf{H}_B$). However, there are faster methods for calculating (10.7), requiring only $\mathcal{O}(B\,2^B) = \mathcal{O}(M\,\log M)$ arithmetic operations, e.g., [YH97].

It follows directly from the properties of the Hadamard matrix $\mathbf{H}_B$ that upon pre-multiplying the transform (10.7) with $\mathbf{H}_B$ we get the vector $\mathbf{e}$ back. Breaking it down to a single vector element, we have that

$$\mathbf{e}_i = \mathbf{h}_i^{\mathrm{T}}\,\mathbf{t}, \quad i = 0,\, 1,\, \ldots,\, M-1. \tag{10.8}$$

Thus, if we represent the vector index i with the special binary representation above, we can write $\mathbf{e}_i = \mathbf{h}_i^{\mathrm{T}}\,\mathbf{t}$, with $\mathbf{h}_i$ as in (10.6). This is in fact $\mathbf{e}_i$ written as a (nonlinear) function of the bits in i. To see how it works, we consider a small example.

Simple Example of Hadamard Transform Let $\mathbf{e} = [\mathbf{e}_0\ \mathbf{e}_1\ \mathbf{e}_2\ \mathbf{e}_3]^{\mathrm{T}}$ be a vector of length $M = 4$ (implies $B = 2$), say $\mathbf{e} = [6\ 0\ 10\ -4]^{\mathrm{T}}$. From (10.1) we obtain

$$\mathbf{H}_2 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}. \tag{10.9}$$

Then, using (10.7), we have $\mathbf{t} = [3\ 5\ 0\ -2]^{\mathrm{T}}$. We would now like to calculate the element $\mathbf{e}_2$ (the third element) from $\mathbf{t}$. Thus, $i = 2$ with the corresponding standard binary representation $(1,\, 0)_2$ mapping through (10.4) to the special binary representation $(-1,\, +1)$. From (10.6) we have

$$\mathbf{h}_2 = \begin{bmatrix} 1 \\ -1 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ -1 \\ -1 \end{bmatrix} \tag{10.10}$$

(indeed equal to the third column of $\mathbf{H}_2$) and upon inserting this and our $\mathbf{t}$ in (10.8) we obtain $\mathbf{e}_2 = 3 + 5 - 0 - (-2) = 10$, which is the correct value.

From this small example it might seem as if the Hadamard transform is merely a cumbersome method for extracting an element from a vector. On the contrary, the transform will prove itself useful in the remainder of this chapter, when it is applied in the analysis of bit mask assignments. In a similar way as the discrete Fourier transform can be used to transform a vector to the frequency domain, perform manipulations and transform it back again, we will use the Hadamard transform to manipulate vectors in the 'bit domain'.

10.2 Allocating the Index Bits

It is clear from the results presented in Section 9.2 that the choice of bit mask configuration, i.e., the allocation of ones and zeros in $\mathbf{q}$, has a significant effect

on the corrected ADC performance. In this section we will derive a mathematical analysis tool, based on Hadamard transforms, for the bit allocation problem.

The analysis below will have its starting point in a calibrated correction table $\mathbf{e}$, indexed with a full B-bit index I, thus having 2^B entries. This table is then *reduced* by deselecting index bits from I.

Correction Table Hadamard Representation

Consider again the setup of Figure 9.1 on page 116. Assume that K delay elements are used, and that *no bit-masking* is performed; this is of course the same as assigning a *transparent* bit mask $\mathbf{q} = \mathbf{1}$. Accordingly, the table is indexed with $B = K\,b + b$ bits, resulting in a table size $M = 2^B$. Denote this table $\mathbf{e}$ and let it be represented as a column vector. The table entry corresponding to an index I is denoted $\mathbf{e}_I$, and the vector $\mathbf{e}$ consists, thus, of entries $\mathbf{e}_0$ through $\mathbf{e}_{M-1}$.

Using (10.7), the Hadamard transform $\mathbf{t}$ of $\mathbf{e}$ is readily calculated. Recall also that a certain table entry $\mathbf{e}_I$ can be calculated from the bits of I through (10.6) and (10.8).

Deselecting Index Bits

Let us now employ a bit mask on the index I to deselect bits so that only β bits remain ($0 \leq \beta \leq B$). As in Chapter 9, we use a bit mask vector $\mathbf{q}$ to define which bits from (the binary representation of) the index I to propagate to the β-bit index $\tilde{I}$. The vector $\mathbf{q}$ contains ones and zeros with a '1' in the j-th position dictating that the j-th bit I_j should be propagated to $\tilde{I}$. The notation $I \xrightarrow{\mathbf{q}} \tilde{I}$ will be used to denote the mapping from I to $\tilde{I}$ through the bit mask $\mathbf{q}$. See Definition 2 on page 117. We can define a new table, $\tilde{\mathbf{e}}$, of length 2^β, which is indexed with $\tilde{I}$. Naturally, the table $\tilde{\mathbf{e}}$ will differ with the choice of bit mask $\mathbf{q}$, even if exactly the same calibration signal is applied. Consequently, the performance of the corrected ADC will also depend on the choice of $\mathbf{q}$.

The problem of interest is *which* β bits to select from I. This is of course dependent on which figure of merit we choose; different figures of merit will be discussed and evaluated in Chapter 11. In this chapter, we derive a framework to compare the outcome of different bit masks. Below, two different derivations are presented – one probabilistic and one deterministic – yielding the same result.

Probabilistic Derivation

Assume, as above, that a correction table $\mathbf{e}$ of length $M = 2^B$ has been calibrated. Furthermore, assume that the calibration signal applied is such that the calibration samples are distributed over all possible indices I according to some probability mass function (PMF), say $p_I(i)$. Thus, we can say that the probability of the index I being equal to i is $p_I(i)$. A straightforward estimate of $p_I(i)$ is to use the counter

a_I introduced in the calibration scheme in Section 9.1; simply set

$$\hat{p}_I(i) = \frac{\mathbf{a}_i}{\sum_j \mathbf{a}_j}. \tag{10.11}$$

This is in fact the maximum-likelihood estimate if the samples used for calibration are considered independent and identically distributed.

Next, a set $\mathcal{Q}_{\tilde{I}}$ of indices is defined in order to simplify the notation in the forthcoming derivations. Since the mapping $I \xrightarrow{\mathbf{q}} \tilde{I}$ omits a number of bits in I, there will be several – in fact $2^{B-\beta}$ – values for I that map to one and the same $\tilde{I}$. Those integers I constitute the set $\mathcal{Q}_{\tilde{I}}(\mathbf{q})$:

Definition 5. *Let* $\mathbf{q}$ *be a bit mask, consisting of the elements* $\mathbf{q}_k$, $k = 1, 2, \ldots, B$. *The bit mask defines a mapping* $I \xrightarrow{\mathbf{q}} \tilde{I}$ *as in Definition 2. Then, define the set*

$$\mathcal{Q}_{\tilde{I}}(\mathbf{q}) = \{I : \ I \xrightarrow{\mathbf{q}} \tilde{I}\}, \tag{10.12}$$

that is, the set of all indices I *which map to the same* $\tilde{I}$ *through the bit mask defined by* $\mathbf{q}$.

Each choice of $\mathbf{q}$ defines 2^β different sets. It is easy to see that all sets will have exactly $2^{B-\beta}$ members and that all sets are disjoint. In fact, the operation $\xrightarrow{\mathbf{q}}$ defines an *equivalence relation* on the set of B-bit integers $\{0, 1, \ldots, 2^B - 1\}$. Two integers I_1 and I_2 are said to be equivalent (with respect to $\xrightarrow{\mathbf{q}}$) if $I_1 \xrightarrow{\mathbf{q}} \tilde{I}$ and $I_2 \xrightarrow{\mathbf{q}} \tilde{I}$. Hence, the set of B-bit integers is partitioned into disjoint *equivalence classes*. Each class is the set of all B-bit integers mapping to the same β-bit integer through $\xrightarrow{\mathbf{q}}$, leading to the conclusion that $\mathcal{Q}_{\tilde{I}}(\mathbf{q})$ is an equivalence class.

The aim is now to find a good correction value, given that the bit masked index is a specific integer $\tilde{I} \in [0, 2^\beta - 1]$. We saw in (9.3) that the optimal correction values were given by

$$\mathbf{e}_{I,\,\mathrm{opt}} = \frac{\displaystyle\int_{\mathbf{s}\in\boldsymbol{\mathcal{S}}_I} e(\mathbf{s})\, f_{\mathbf{s}}(\mathbf{s})\, d\mathbf{s}}{\displaystyle\int_{\mathbf{s}\in\boldsymbol{\mathcal{S}}_I} f_{\mathbf{s}}(\mathbf{s})\, d\mathbf{s}}. \tag{10.13}$$

Moving on to find the optimal value for $\tilde{\mathbf{e}}_{\tilde{I}}$, we can express this in a similar fashion as (10.13) if the range of integration is extended from $\boldsymbol{\mathcal{S}}_I$ to the union of all sets $\boldsymbol{\mathcal{S}}_J$ such that $J \in \mathcal{Q}_{\tilde{I}}(\mathbf{q})$. The fact that all sets are disjoint will be utilized. Thus, we have

$$\tilde{\mathbf{e}}_{\tilde{I},\,\mathrm{opt}} = \frac{\displaystyle\int_{\mathbf{s}\in\{\cup\boldsymbol{\mathcal{S}}_J:\, J\in\mathcal{Q}_{\tilde{I}}\}} e(\mathbf{s})\, f_{\mathbf{s}}(\mathbf{s})\, d\mathbf{s}}{\displaystyle\int_{\mathbf{s}\in\{\cup\boldsymbol{\mathcal{S}}_J:\, J\in\mathcal{Q}_{\tilde{I}}\}} f_{\mathbf{s}}(\mathbf{s})\, d\mathbf{s}} = \frac{\displaystyle\sum_{J\in\mathcal{Q}_{\tilde{I}}} \int_{\mathbf{s}\in\boldsymbol{\mathcal{S}}_J} e(\mathbf{s})\, f_{\mathbf{s}}(\mathbf{s})\, d\mathbf{s}}{\displaystyle\sum_{J\in\mathcal{Q}_{\tilde{I}}} \int_{\mathbf{s}\in\boldsymbol{\mathcal{S}}_J} f_{\mathbf{s}}(\mathbf{s})\, d\mathbf{s}}. \tag{10.14}$$

Rearranging (10.13), we have

$$\int_{\mathbf{s}\in\boldsymbol{\mathcal{S}}_I} e(\mathbf{s})\, f_{\mathbf{s}}(\mathbf{s})\, d\mathbf{s} = \mathbf{e}_{I,\,\mathrm{opt}} \int_{\mathbf{s}\in\boldsymbol{\mathcal{S}}_I} f_{\mathbf{s}}(\mathbf{s})\, d\mathbf{s}. \tag{10.15}$$

Furthermore, we observe that the integral on the right-hand side of (10.15), and hence the integral in the denominator of (10.13) and (10.14), is in fact the probability of the index I, i.e., $\int_{\mathbf{s}\in\boldsymbol{\mathcal{S}}_I} f_{\mathbf{s}}(\mathbf{s})\, d\mathbf{s} = p_I(I)$. Inserting this and (10.15) into (10.14) yields

$$\tilde{\mathbf{e}}_{\tilde{I}} = \frac{\sum\limits_{J\in\mathcal{Q}_{\tilde{I}}} p_I(J)\, \mathbf{e}_J}{\sum\limits_{J\in\mathcal{Q}_{\tilde{I}}} p_I(J)}, \tag{10.16}$$

provided that the denominator is non-zero; if not, the probability of the index $\tilde{I}$ being produced is zero and we let $\tilde{\mathbf{e}}_{\tilde{I}} = 0$. We have now in (10.16) a relation between the entries of $\mathbf{e}$ and the entries of $\tilde{\mathbf{e}}$. For notational simplicity, the explicit dependence of the set $\mathcal{Q}_{\tilde{I}}$ upon $\mathbf{q}$ is omitted. Furthermore, the subscript 'opt' has been removed, indicating that (10.16) will be used also when merging table entries resulting from an experimental calibration, and thus not being equal to the optimal value of (10.13). Under the assumption that $p_I(I)$ is constant for all $I \in \mathcal{Q}_{\tilde{I}}$, (10.16) reduces to

$$\tilde{\mathbf{e}}_{\tilde{I}} = \frac{1}{2^{B-\beta}} \sum_{I\in\mathcal{Q}_{\tilde{I}}} \mathbf{e}_I. \tag{10.17}$$

That is, the value for $\tilde{\mathbf{e}}_{\tilde{I}}$ is the arithmetic mean of all $\mathbf{e}_I$ for which $I \xrightarrow{\mathbf{q}} \tilde{I}$ when all $\mathbf{e}_I$ have equal probability.

We are now interested in comparing the original correction table $\mathbf{e}$ (of size 2^B) with the reduced-size table $\tilde{\mathbf{e}}$ (of size 2^β) resulting from (10.16). However, these two tables, represented as vectors, are of different sizes (except for the trivial case of $\beta = B$), so a direct, one-to-one comparison is not possible. Instead, we would like to construct a new table, say $\mathbf{f}$, of the same size as $\mathbf{e}$, but with the special property that

$$\mathbf{f}_I = \tilde{\mathbf{e}}_{\tilde{I}} \quad \text{if } I \xrightarrow{\mathbf{q}} \tilde{I}. \tag{10.18}$$

It will become clear in Section 10.3 why this property is desirable. In order to facilitate the bit-allocation analysis, the table $\mathbf{f}$ should have an explicit relationship to $\mathbf{e}$ and the bit mask $\mathbf{q}$. First, however, a special vector and matrix must be introduced:

Definition 6. *Let* $\mathbf{q}$ *be a vector consisting of the elements* $\mathbf{q}_k$, $k = 1, 2, \ldots, B$. *We define the vector*

$$\mathbf{g} \triangleq \begin{bmatrix} 1 \\ \mathbf{q}_B \end{bmatrix} \otimes \begin{bmatrix} 1 \\ \mathbf{q}_{B-1} \end{bmatrix} \otimes \cdots \otimes \begin{bmatrix} 1 \\ \mathbf{q}_1 \end{bmatrix} \tag{10.19}$$

of length $M = 2^B$, and the matrix

$$\mathbf{G} \triangleq \mathrm{diag}\{\mathbf{g}\} = \begin{bmatrix} 1 & 0 \\ 0 & \mathbf{q}_B \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 \\ 0 & \mathbf{q}_{B-1} \end{bmatrix} \otimes \cdots \otimes \begin{bmatrix} 1 & 0 \\ 0 & \mathbf{q}_1 \end{bmatrix} \tag{10.20}$$

of size M-by-M.

Now, an expression for the table (vector) $\mathbf{f}$ as a function of $\mathbf{e}$ and $\mathbf{q}$ is to be derived, first in the case of equal probability, but later generalizing to arbitrary probabilities. In the equal-probability case, we have that $\mathbf{f}_I = \tilde{\mathbf{e}}_{\tilde{I}}$ with $\tilde{\mathbf{e}}_{\tilde{I}}$ as in (10.17) when $I \xrightarrow{\mathbf{q}} \tilde{I}$. The following lemma provides an explicit relationship between $\mathbf{f}$ and $\mathbf{e}$ in this case:

Lemma 12 (Projection matrix $\mathbf{P}(\mathbf{q})$)**.** *Let $\mathbf{e} = [\mathbf{e}_0\ \mathbf{e}_1\ \ldots\ \mathbf{e}_{M-1}]^{\mathrm{T}}$ be a column vector of length $M = 2^B$, were B is a positive integer. Let the integer $I \in [0,\, M-1]$ be represented using a B-bit binary representation $(I_B,\, I_{B-1},\, \ldots,\, I_1)_2$. Let $\tilde{I}$ be an integer resulting from selecting β bits from I, where $0 \le \beta \le B$. Denote this operation $I \xrightarrow{\mathbf{q}} \tilde{I}$ and define the set $\mathcal{Q}_{\tilde{I}}(\mathbf{q})$ as in (10.12). Then, the vector*

$$\mathbf{f} = \mathbf{P}(\mathbf{q})\,\mathbf{e} \tag{10.21}$$

is of length M, with the I-th entry ($I \in [0,\, M-1]$)

$$\mathbf{f}_I = \frac{1}{2^{B-\beta}} \sum_{J \in \mathcal{Q}_{\tilde{I}}} \mathbf{e}_J, \tag{10.22}$$

if the M-by-M matrix $\mathbf{P}(\mathbf{q})$ is

$$\mathbf{P}(\mathbf{q}) = \frac{1}{M}\,\mathbf{H}_B\,\mathbf{G}\,\mathbf{H}_B. \tag{10.23}$$

The matrix $\mathbf{G}$ is defined in (10.20).

The proof is provided in Appendix 10.A, but an intuitive explanation is that $\mathbf{e}$ is transformed to the 'bit domain', where $\mathbf{G}$ nulls out the deselected bits, after which the inverse transform is applied. The matrix $\mathbf{P}(\mathbf{q})$ is in fact an orthogonal projection matrix, since

$$\mathbf{P}(\mathbf{q})^2 = \frac{1}{M^2}\,\mathbf{H}_B\,\mathbf{G}\,\underbrace{\mathbf{H}_B\,\mathbf{H}_B}_{M\cdot\mathbf{I}}\,\mathbf{G}\,\mathbf{H}_B = \frac{1}{M}\,\mathbf{H}_B\,\underbrace{\mathbf{G}\,\mathbf{G}}_{\mathbf{G}}\,\mathbf{H}_B = \mathbf{P}(\mathbf{q}) \tag{10.24}$$

and $\mathbf{P}(\mathbf{q})^{\mathrm{T}} = \mathbf{P}(\mathbf{q})$.

Moving on to the case of arbitrary probabilities $p_I(i)$, the entry $\mathbf{f}_I$ should be set equal to $\tilde{\mathbf{e}}_{\tilde{I}}$ of (10.16). Collect all probabilities in the column vector $\mathbf{p} = [p_I(0)\ p_I(1)\ \ldots\ p_I(M-1)]^{\mathrm{T}}$. The numerator of (10.16) can then be written

$$\begin{aligned} \sum_{J \in \mathcal{Q}_{\tilde{I}}} \mathbf{e}_J p_I(J) &= \sum_{J \in \mathcal{Q}_{\tilde{I}}} \mathbf{e}_J \mathbf{p}_J = /(10.21) \text{ and } (10.22)/ \\ &= 2^{B-\beta}[\mathbf{P}(\mathbf{q})\ \mathrm{diag}\{\mathbf{p}\}\,\mathbf{e}]_I, \end{aligned} \tag{10.25}$$

and the denominator can be written in a similar fashion as

$$\sum_{J \in \mathcal{Q}_{\tilde{I}}} p_I(J) = 2^{B-\beta} [\mathbf{P}(\mathbf{q})\, \mathbf{p}]_I. \tag{10.26}$$

The fact that element-wise multiplication of two column vectors $\mathbf{x}$ and $\mathbf{y}$ can be written $\mathrm{diag}\{\mathbf{x}\}\, \mathbf{y}$ has been used in both the preceding formulas. The vector $\mathbf{f}$ is the element-wise division of the vectors $\mathbf{P}(\mathbf{q})\ \mathrm{diag}\{\mathbf{p}\}\, \mathbf{e}$ and $\mathbf{P}(\mathbf{q})\, \mathbf{p}$ (factors $2^{B-\beta}$ cancel), except where the denominator has a zero element, in which case the corresponding element in $\mathbf{f}$ is set to zero. This can be written using the *Moore-Penrose pseudoinverse* (see e.g., [GVL96]), here denoted †, since $\mathbf{D}^\dagger$, where $\mathbf{D}$ is a diagonal matrix, is simply the nonzero elements replaced by their reciprocals. The following lemma concludes this discussion:

Lemma 13 (Reduction matrix $\mathbf{R}(\mathbf{q}, \mathbf{p})$)**.** *Let* $\mathbf{e} = [\mathbf{e}_0\ \mathbf{e}_1\ \ldots\ \mathbf{e}_{M-1}]^{\mathrm{T}}$ *and* $\mathbf{p} = [\mathbf{p}_0\ \mathbf{p}_1\ \ldots\ \mathbf{p}_{M-1}]^{\mathrm{T}}$ *be column vectors of length* $M = 2^B$*, were* B *is a positive integer. Let the integer* $I \in [0,\, M-1]$ *be represented using a* B*-bit binary representation* $(I_B,\, I_{B-1},\, \ldots,\, I_1)_2$*. Let* $\tilde{I}$ *be an integer resulting from selecting* β *bits from* I*, where* $0 \leq \beta \leq B$*. Denote this operation* $I \xrightarrow{\mathbf{q}} \tilde{I}$ *and define the set* $\mathcal{Q}_{\tilde{I}}(\mathbf{q})$ *as in (10.12). Then, the vector*

$$\mathbf{f} = \mathbf{R}(\mathbf{q},\, \mathbf{p})\, \mathbf{e} \tag{10.27}$$

is of length M*, with the* I*-th entry (*$I \in [0,\, M-1]$*)*

$$\mathbf{f}_I = \begin{cases} \dfrac{\displaystyle\sum_{J \in \mathcal{Q}_{\tilde{I}}} p_I(J) \mathbf{e}_J}{\displaystyle\sum_{J \in \mathcal{Q}_{\tilde{I}}} p_I(J)}, & \displaystyle\sum_{J \in \mathcal{Q}_{\tilde{I}}} p_I(J) \neq 0, \\ 0, & \textit{otherwise}, \end{cases} \tag{10.28}$$

if the M*-by-*M *matrix* $\mathbf{R}(\mathbf{q},\, \mathbf{p})$ *is*

$$\mathbf{R}(\mathbf{q},\, \mathbf{p}) = \mathrm{diag}\{\mathbf{P}(\mathbf{q})\, \mathbf{p}\}^\dagger\, \mathbf{P}(\mathbf{q})\ \mathrm{diag}\{\mathbf{p}\}. \tag{10.29}$$

The proof is in the discussion above. Note that the matrix $\mathbf{R}(\mathbf{q},\, \mathbf{p})$ does not change when the vector $\mathbf{p}$ is scaled with a constant. Moreover, upon inserting a uniformly distributed $\mathbf{p}$ in (10.29), $\mathbf{R}(\mathbf{q},\, \mathbf{p})$ reduces to $\mathbf{P}(\mathbf{q})$, i.e.,

$$\mathbf{R}(\mathbf{q},\, \alpha \mathbf{1}) = \mathbf{P}(\mathbf{q}), \tag{10.30}$$

where $\alpha \neq 0$. We conclude that the matrix $\mathbf{R}(\mathbf{q},\, \mathbf{p})$ in (10.29) provides a linear relation between $\mathbf{e}$ and $\mathbf{f}$, although the matrix is dependent on $\mathbf{q}$ in a nonlinear way.

Deterministic Derivation

In this section, an alternative derivation of the results in the previous section is presented. The same results, in brief Lemmas 12 and 13, are obtained.

Let the table $\mathbf{e}$ be calibrated according to Section 9.1 with a transparent bit mask (all $\mathbf{q}_i = 1$), using a specific set of N calibration samples. Assume that the calibration samples are such that all entries in $\mathbf{e}$ are calibrated with the same (integer) number of samples N/M, i.e., the calibration samples are equally distributed over all possible indices I. Hence, all $\mathbf{a}_I \triangleq \mathbf{a}_I(N)$ are equal. This can be accomplished by using a uniformly distributed noise; the uniformity constraint will be relaxed later on. Assume now that one of the bits in the index I, say the j-th bit, is deselected through bit-masking (cf Figure 9.1) to produce a $(B-1)$-bit index $\tilde{I}$. With B bits it is possible to produce 2^B different indices I, while with $B-1$ bits we can only construct $2^{B-1} = 2^B/2$ indices. Hence, given a $(B-1)$-bit index $\tilde{I}$, we have two possible B-bit indices, say I_1 and I_2, that will map through $\mathbf{q}$ to the same $\tilde{I}$, i.e., both $I_1 \xrightarrow{\mathbf{q}} \tilde{I}$ and $I_2 \xrightarrow{\mathbf{q}} \tilde{I}$ holds. Following the calibration scheme of Section 9.1, samples that would have been used to update either $\mathbf{e}_{I_1}$ or $\mathbf{e}_{I_2}$, will now all be used to update $\tilde{\mathbf{e}}_{\tilde{I}}$. The outcome will be that $\tilde{\mathbf{e}}_{\tilde{I}}$ equals the arithmetic mean of $\mathbf{e}_{I_1}$ and $\mathbf{e}_{I_2}$. A short example is suitable here.

Bit Reduction Example Consider a table $\mathbf{e}$ of $M = 8$ entries, i.e., indexed with a 3-bit index $I \in \{0, 1, \ldots, 7\}$. Assume that the table has been calibrated, in accordance with the method described in Section 9.1, with a set of N calibration samples such that all entries in $\mathbf{e}$ are updated with the same number of samples $(N/8)$. Suppose now that a second table $\tilde{\mathbf{e}}$ is calibrated using the same set of samples, but this time with the second bit deselected, i.e., applying a bit mask $\mathbf{q} = [1\ 0\ 1]^{\mathrm{T}}$ used to map $I \xrightarrow{\mathbf{q}} \tilde{I}$. The new table has 4 entries and $\tilde{I}$ is hence a 2-bit index. Then, both $I = 0$ and $I = 2$ map into $\tilde{I} = 0$, since apart from the second bit they are both equal. In the same way, 1 and 3 map to 1, 4 and 6 map to 2, and 5 and 7 map to 3. The outcome is that $\tilde{\mathbf{e}}_0$ will be updated with those samples previously used to update *both* $\mathbf{e}_0$ and $\mathbf{e}_2$, so that $\tilde{\mathbf{e}}_0 = (\mathbf{e}_0 + \mathbf{e}_2)/2$, and so on.

We are now interested in finding a table $\mathbf{f}$ – of the same size as $\mathbf{e}$, but with $\mathbf{f}_I = \tilde{\mathbf{e}}_{\tilde{I}}$ when $I \xrightarrow{\mathbf{q}} \tilde{I}$ – and an *explicit relationship* between $\mathbf{f}$ and the original $\mathbf{e}$. From the discussion above we learn that those table entries whose indices cannot be distinguished without the j-th bit, are to be replaced by their averages. Returning to the 3-bit example, this implies $\mathbf{f}_0 = \mathbf{f}_2 \equiv (\mathbf{e}_0 + \mathbf{e}_2)/2$, and so forth.

The discussion can be extended to the case of more than one bit being removed. The result is still that if $I \xrightarrow{\mathbf{q}} \tilde{I}$, then $\mathbf{f}_I$ should equal the arithmetic mean of all entries $\mathbf{e}_J$ for which $J \xrightarrow{\mathbf{q}} \tilde{I}$. This is precisely the result of Lemma 12.

The next step is to generalize the result above to the case when the set of calibration samples is *arbitrarily distributed* over all possible indices I, i.e., all $\mathbf{a}_I$ are nonnegative integers summing up to N but otherwise arbitrary. In this case, the averaging above should be replaced by a *weighted averaging*, to conform with

the calibration strategy of Section 9.1. That is, in the 3-bit example above, $\tilde{\mathbf{e}}_0$ would equal $(\mathbf{a}_0\,\mathbf{e}_0 + \mathbf{a}_2\,\mathbf{e}_2)/(\mathbf{a}_0 + \mathbf{a}_2)$ after completed calibration. Consequently, both $\mathbf{f}_0$ and $\mathbf{f}_2$ should equal $(\mathbf{a}_0\,\mathbf{e}_0 + \mathbf{a}_2\,\mathbf{e}_2)/(\mathbf{a}_0 + \mathbf{a}_2)$.

Again, the methods can be extended to the case of more than one bit being deselected. The weighted averaging is extended to weighted arithmetic mean of all those entries whose indices cannot be distinguished without the deselected bits. This agrees with the result of Lemma 13 if we set $\mathbf{p} = \mathbf{a}$.

10.3 Post-correction Equivalents

The reduction matrix $\mathbf{R}(\mathbf{q},\,\mathbf{a})$ can now be used to evaluate the effects of a specific bit allocation in the bit mask $\mathbf{q}$ of Figure 9.1. Retain the assumptions on the table $\mathbf{e}$ being calibrated using the structure of Figure 9.2, employing K delay elements and a *transparent* bit mask, i.e., selecting all $B = K\,b + b$ bits. Still, $\mathbf{e}$ is of size $M = 2^B$ and is indexed with a B-bit index I. The table has been calibrated with a specific set of calibration samples applied to the ADC under test, possibly resulting in nonuniformly distributed elements in $\mathbf{a}$. We could also say that the table has been calibrated with a calibration signal resulting in a specific index PMF $p_I(i)$.

Also, a second table $\tilde{\mathbf{e}}$ has been calibrated with the same set of calibration samples (or a signal with the same statistical properties), still employing K delay elements, but this time with a non-transparent bit mask $\mathbf{q}$ selecting $\beta < B$ bits from I. The table $\tilde{\mathbf{e}}$ is thus of size 2^β and is indexed with a β-bit index $\tilde{I}$. Since $\mathbf{e}$ and $\tilde{\mathbf{e}}$ are calibrated with the same set of calibration samples, the relations of Lemma 13 apply so that the entry $\tilde{\mathbf{e}}_{\tilde{I}}$ equals the weighted average of all entries $\mathbf{e}_I$ whose indices I map into $\tilde{I}$ through the bit mask $\mathbf{q}$.

The performance of the ADC after correction will naturally differ depending on which correction table, $\mathbf{e}$ or $\tilde{\mathbf{e}}$, is being used. Clearly, it would be of great benefit if the outcome of a specific choice of bit mask could be calculated without having to re-calibrate the table. The reduction matrix $\mathbf{R}(\mathbf{q},\,\mathbf{a})$ is the key to relating the results of different bit masks to each other. From (10.29) we have $\mathbf{f} = \mathbf{R}(\mathbf{q},\,\mathbf{a})\,\mathbf{e}$, which is a table of size 2^B. However, through the averaging operation of the matrix $\mathbf{R}(\mathbf{q},\,\mathbf{a})$, the table $\mathbf{f}$ has a $2^{B-\beta}$-fold redundancy, i.e., $\mathbf{f}$ has only 2^β unique entries. The unique entries are exactly those of the 2^β-size table $\tilde{\mathbf{e}}$. Since every index I addresses an entry in $\mathbf{f}$ which is equal to the weighted average of all entries $\mathbf{e}_I$ such that $I \xrightarrow{\mathbf{q}} \tilde{I}$, we have that

$$\tilde{\mathbf{e}}_{\tilde{I}} = \mathbf{f}_I = [\mathbf{R}(\mathbf{q},\,\mathbf{a})\,\mathbf{e}]_I. \tag{10.31}$$

In other words, $\mathbf{f}$ and $\mathbf{e}$ share the same address space, but $\mathbf{f}$ addressed with I yields the same values as $\tilde{\mathbf{e}}$ addressed with $\tilde{I}$ if $I \xrightarrow{\mathbf{q}} \tilde{I}$. Figure 10.1 illustrates this relationship in a signal flowchart.

Bit Reduction Example (continued) Returning once more to the simple 3-bit example in Section 10.2, it is clear that both $\mathbf{e}$ and $\mathbf{f}$ are of size 8, while

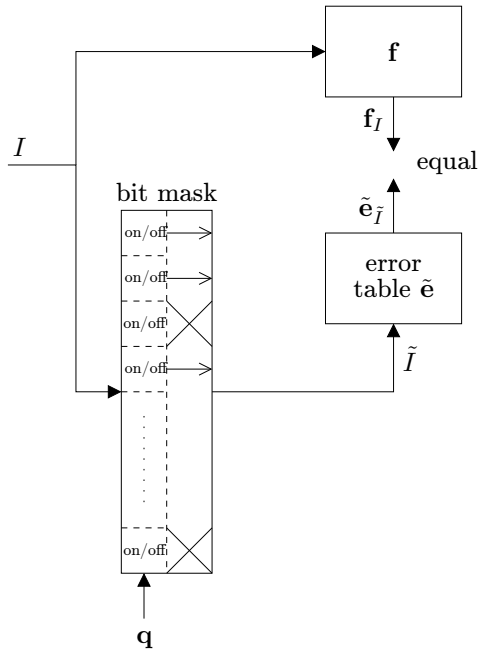


Figure 10.1: An illustration of the relationship between the 2^B-size table $\mathbf{f}$ and and the 2^β-size table $\tilde{\mathbf{e}}$.

$\tilde{\mathbf{e}}$ is of size 4. The table $\mathbf{f}$ has, however, a 2-fold redundancy, since $\mathbf{f}_0 = \mathbf{f}_2 = (\mathbf{a}_0\,\mathbf{e}_0 + \mathbf{a}_2\,\mathbf{e}_2)/(\mathbf{a}_0 + \mathbf{a}_2) = \tilde{\mathbf{e}}_0$ because both $I = 0$ and $I = 2$ maps to $\tilde{I} = 0$.

The observation above can now be used to evaluate the outcome of different bit mask settings *without re-calibrating a new correction table*. For example, if a state-space table $\mathbf{e}$ (that is, $K = 1$ and $B = 2b$) has been calibrated, we can now *mimic* a static table simply by setting the b bits in $\mathbf{q}$ corresponding to the 'previous sample' $x(n-1)$ to zero in (10.29) and (10.31). In other words, set $\mathbf{q} = [\mathbf{1}_b^{\mathrm{T}}\ \mathbf{0}_b^{\mathrm{T}}]^{\mathrm{T}}$. In fact, we can mimic any table $\tilde{\mathbf{e}}$ by using (10.31), as long as $\tilde{\mathbf{e}}$ is indexed with a subset of the bits used to index $\mathbf{e}$.

In the next chapter we will give two examples of how to use the reduction matrix in order to optimize the bit mask.

10.A Proof of Lemma 12

In order to simplify the notation, define the iterated Kronecker products

$$\bigotimes_{k=1}^{K} \mathbf{A}_k \triangleq \mathbf{A}_1 \otimes \mathbf{A}_2 \otimes \cdots \otimes \mathbf{A}_K \tag{10.32}$$

of the matrices $\mathbf{A}_1$ through $\mathbf{A}_K$ of suitable dimensions. Note that the Kronecker product is not commutative. For instance, $\bigotimes_{k=1}^{K} \mathbf{A}_k \neq \bigotimes_{k=K}^{1} \mathbf{A}_k$ in general.

Assume now that one bit, say the j-th bit, is deselected in the B-bit index I, using a bit mask $\mathbf{q}$ (cf. Figure 9.1), i.e., the p-th element of $\mathbf{q}$ is

$$\mathbf{q}_p = \begin{cases} 0, & p = j, \\ 1, & \text{otherwise.} \end{cases} \tag{10.33}$$

Let $\mathbf{e}$ be a vector of size $M = 2^B$ with a Hadamard transform $\mathbf{t}$. With B bits, M different indices $I \in \{0, \ldots, M-1\}$ exist, but with one bit deselected the indices will be partitioned into pairs; both indices in one pair are indistinguishable when the j-th bit is masked, that is, one pair consists of the two indices I such that $I \xrightarrow{\mathbf{q}} \tilde{I}$. Denote one such pair $\{J_1, J_2\}$. Furthermore, let $\bar{J}_k$ be the ± 1-representation (cf (10.4)) of J_k. Then, the average of the two table entries $\mathbf{e}_{J_1}$ and $\mathbf{e}_{J_2}$ is

$$\mathbf{f}_{J_1} = \mathbf{f}_{J_2} = \frac{1}{2}\left(\mathbf{e}_{J_1} + \mathbf{e}_{J_2}\right) = /(10.8)/ = \frac{1}{2M}\left(\mathbf{h}_{J_1}^{\mathrm{T}} + \mathbf{h}_{J_2}^{\mathrm{T}}\right)\mathbf{t}, \tag{10.34}$$

where we get

$$\begin{aligned}
\mathbf{h}_{J_1} + \mathbf{h}_{J_2} &= \sum_{k=1}^{2} \left(\bigotimes_{p=B}^{1} \begin{bmatrix} 1 \\ [\bar{J}_k]_p \end{bmatrix} \right) = \Big/ [\bar{J}_1]_p = [\bar{J}_2]_p,\ p \neq j \Big/ \\
&= \left(\bigotimes_{p=B}^{j+1} \begin{bmatrix} 1 \\ [\bar{J}_1]_p \end{bmatrix} \right) \otimes \left(\begin{bmatrix} 1 \\ [\bar{J}_1]_j \end{bmatrix} + \begin{bmatrix} 1 \\ [\bar{J}_2]_j \end{bmatrix} \right) \otimes \left(\bigotimes_{p=j-1}^{1} \begin{bmatrix} 1 \\ [\bar{J}_1]_p \end{bmatrix} \right) \\
&= \Big/ [\bar{J}_1]_j + [\bar{J}_2]_j = 0 \Big/ \\
&= \left(\bigotimes_{p=B}^{j+1} \begin{bmatrix} 1 \\ [\bar{J}_1]_p \end{bmatrix} \right) \otimes \begin{bmatrix} 2 \\ 0 \end{bmatrix} \otimes \left(\bigotimes_{p=j-1}^{1} \begin{bmatrix} 1 \\ [\bar{J}_1]_p \end{bmatrix} \right) \\
&= 2 \left(\bigotimes_{p=B}^{j+1} \begin{bmatrix} 1 \\ [\bar{J}_1]_p \end{bmatrix} \right) \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \left(\bigotimes_{p=j-1}^{1} \begin{bmatrix} 1 \\ [\bar{J}_1]_p \end{bmatrix} \right)
\end{aligned}$$

$$= /(10.33)/ = 2 \bigotimes_{p=B}^{1} \begin{bmatrix} 1 \\ \mathbf{q}_p [\bar{J}_1]_p \end{bmatrix} = 2 \bigotimes_{p=B}^{1} \left(\begin{bmatrix} 1 & 0 \\ 0 & \mathbf{q}_p \end{bmatrix} \begin{bmatrix} 1 \\ [\bar{J}_1]_p \end{bmatrix} \right)$$

$$= \Big/ \mathbf{A}\,\mathbf{C} \otimes \mathbf{B}\,\mathbf{D} = (\mathbf{A} \otimes \mathbf{B})(\mathbf{C} \otimes \mathbf{D}) \Big/$$

$$= 2 \left(\bigotimes_{p=B}^{1} \begin{bmatrix} 1 & 0 \\ 0 & \mathbf{q}_p \end{bmatrix} \right) \left(\bigotimes_{p=B}^{1} \begin{bmatrix} 1 \\ [\bar{J}_1]_p \end{bmatrix} \right)$$

$$= /(10.20) \text{ and } (10.6)/ = 2\mathbf{G}\,\mathbf{h}_{J_1} = 2\mathbf{G}\,\mathbf{h}_{J_2}. \quad (10.35)$$

The last equality comes from the fact that the only difference between $\mathbf{h}_{J_1}$ and $\mathbf{h}_{J_2}$ lies in the bit masked away by $\mathbf{q}$. Inserting (10.35) in (10.34), we have

$$\mathbf{f}_{J_1} = \mathbf{f}_{J_2} = \mathbf{h}_{J_1}^{\mathrm{T}}\,\mathbf{G}\,\mathbf{t} = \mathbf{h}_{J_2}^{\mathrm{T}}\,\mathbf{G}\,\mathbf{t}, \quad (10.36)$$

and thus for the entire table, i.e., for all $I \in \{0, \ldots, M-1\}$, we obtain

$$\mathbf{f} = \mathbf{H}_B\,\mathbf{G}\,\mathbf{t} = \frac{1}{M}\,\mathbf{H}_B\,\mathbf{G}\,\mathbf{H}_B\,\mathbf{e} \triangleq \mathbf{P}(\mathbf{q})\,\mathbf{e}. \quad (10.37)$$

This result is easily generalized to a bit mask where ℓ bits have been deselected. Through repeated use of (10.37) with bit masks deselecting one bit each, and utilizing the fact that $\mathbf{H}_B\,\mathbf{H}_B = M\,\mathbf{I}$, it can be shown that (10.37) holds for bit masks deselecting an arbitrary number of bits $\ell \in [0, B]$.

Chapter 11

Applications of the Reduction Matrix

In the previous two chapters, a correction method comprising bit-masking was introduced. Also, an analysis tool for analyzing the effect on the correction table of a specific bit mask was derived. The reduction matrix $\mathbf{R}(\mathbf{q}, \mathbf{a})$ in particular was found to be a crucial part of the analysis. In this chapter, the analysis framework is going to be used to optimize the bit mask. Two different criterions will be applied – the total harmonic distortion (THD) and the signal-to-noise and distortion ratio (SINAD). The derivations and results for these are presented in Sections 11.2 and 11.3, respectively. Since the fundamental setup for both criterions is the same, the common ground is laid down in Section 11.1. The optimization problem formulated in Section 11.1 is a nonlinear binary problem. By making certain assumptions, the cost function can be rewritten as a linear function in binary variables. This is shown in Appendix 11.A, where it is also shown how to avoid the Kronecker-type optimization constraints arrived at in the sequel.

11.1 Generic Cost Function

Consider the generalized correction method described in Chapter 9, with correction and calibration structures depicted in Figures 9.1 and 9.2, respectively. In agreement with the analysis of Chapter 10, let the table $\mathbf{e}$ be a table calibrated using the structure of Figure 9.2 and a specific set of calibration samples. Assume that K delay elements are used and that the bit mask is transparent ($\mathbf{q} = \mathbf{1}$), implying that that the table is addressed with $B = K\,b + b$ bits. Since the bit mask is transparent, $\tilde{I} = I$, and we say that the table $\mathbf{e}$ is addressed with the integer index $I \in [0,\, M-1]$, where $M = 2^B$. Thus, the table is a column vector $\mathbf{e} = [\mathbf{e}_0\ \mathbf{e}_1\ \ldots\ \mathbf{e}_{M-1}]^{\mathrm{T}}$.

The general problem posed in this chapter is to find the optimal bit mask $\mathbf{q}$, selecting exactly $\beta < B$ bits out of the B bits in I. That is, if we were restricted to use a look-up table $\tilde{\mathbf{e}}$ of size $2^\beta < M$, but still employing K delay elements and

the same calibration samples, which β bits out of the B bits available in I should be used to form the index $\tilde{I}$? The concept of optimality must of course come with a measure, or a cost function, for which we are interested to find an extremum. As indicated above, we will employ two different measures in this chapter, THD and SINAD, both leading to a special form of cost functions, viz.

$$V(\mathbf{q}) = (\mathbf{R}(\mathbf{q},\, \mathbf{a})\,\mathbf{e} + \mathbf{c})^{*}\mathbf{B}(\mathbf{R}(\mathbf{q},\, \mathbf{a})\,\mathbf{e} + \mathbf{c}). \tag{11.1}$$

Here, $\mathbf{c}$ is a column vector of length M, $\mathbf{B}$ is a Hermitian M-by-M matrix and $*$ denotes complex conjugate transpose. The matrix and vector $\mathbf{B}$ and $\mathbf{c}$ are determined explicitly by which measure is selected, and also on other parameters related to the particular problem. In both cases we will also have the constraint that $\mathbf{q}$ must select exactly β bits. This can be written as

$$\begin{cases} \displaystyle\sum_{i=1}^{B} \mathbf{q}_i = \mathbf{1}^{\mathrm{T}}\mathbf{q} = \beta \\ \mathbf{q}_i \in \{0,\, 1\} \quad i = 1,\, 2,\, \ldots,\, B. \end{cases} \tag{11.2}$$

11.2 Minimizing THD

The total harmonic distortion (THD) is defined in Section 1.4. There it is stated that the ADC under test should be exercised with a spectrally pure, large amplitude (near full-scale) sinewave, $s(t) = A\sin(2\pi f_0 t + \phi) + C$, with C and A chosen so that the signal is centered within and spans a major part of the ADC's analog input range. The fundamental frequency f_0 is in $[0,\, f_s/2]$ and the initial phase ϕ is arbitrary. A record of N successive samples are collected in a vector $\mathbf{x} = [x(0)\; x(1)\; \ldots\; x(N-1)]^{\mathrm{T}}$. The THD is then defined as

$$\mathrm{THD} = \frac{1}{N}\sqrt{\sum_{h=2}^{H} |X(f_h)|^2}, \tag{11.3}$$

where $X(f_h)$ is the discrete Fourier transform (DFT) of the vector $\mathbf{x}$, evaluated at the h-th harmonic of the fundamental frequency f_0. That is,

$$X(f_h) = \sum_{n=0}^{N-1} x(n)\, e^{-\frac{j2\pi n f_h}{f_s}}. \tag{11.4}$$

In order to avoid spectral leakage in the DFT, f_0 should be selected to coincide with a bin frequency $k\, f_s/N$, $k = 0,\, 1,\, \ldots,\, N/2 - 1$, so that the fundamental tone and also the harmonics line up with a DFT bin. Normally the lowest nine harmonics are considered ($H = 10$), and the *aliased* counterpart of those are added in (11.3). An equation for calculating the aliased harmonics is [Std1241]

$$f_h = (h\, f_0 + N\, f_s) \mod f_s, \quad h = \pm\{2,\, 3,\, \ldots,\, H\}, \tag{11.5}$$

although this does not have to be used explicitly in the forthcoming derivations.

We aim at finding a cost function of the form (11.1). Therefore, the THD is expressed in a matrix notation. Let the row vector $\mathbf{w}_h$ be

$$\mathbf{w}_h = \left[e^{-j2\pi h \frac{f_0}{f_s} 0} \quad e^{-j2\pi h \frac{f_0}{f_s} 1} \quad \dots \quad e^{-j2\pi h \frac{f_0}{f_s} (N-1)} \right], \tag{11.6}$$

and form a matrix $\mathbf{W}$ as

$$\mathbf{W} \triangleq \begin{bmatrix} \mathbf{w}_2 \\ \mathbf{w}_3 \\ \vdots \\ \mathbf{w}_H \end{bmatrix}. \tag{11.7}$$

Then, we can write the Fourier transform (11.4) as $X(f_h) = \mathbf{w}_h\,\mathbf{x}$, and the squared magnitude as

$$|X(f_h)|^2 = |\mathbf{w}_h\,\mathbf{x}|^2 = \mathbf{x}^*\,\mathbf{w}_h^*\,\mathbf{w}_h\,\mathbf{x}. \tag{11.8}$$

The sum in (11.3) can then be written as a quadratic form with $\mathbf{W}$, and we obtain the expression

$$\mathrm{THD} = \frac{1}{N}\sqrt{\sum_{h=2}^{H} \mathbf{x}^*\,\mathbf{w}_h^*\,\mathbf{w}_h\,\mathbf{x}} = \frac{1}{N}\sqrt{\mathbf{x}^*\,\mathbf{W}^*\,\mathbf{W}\,\mathbf{x}}. \tag{11.9}$$

This is the THD of the *uncorrected* ADC at the frequency f_0.

Assume now that a correction table $\mathbf{e}$ has been calibrated as described in Section 11.1, i.e., employing K delay elements and a transparent bit mask. The sequence of correction terms for the recorded output $\mathbf{x}$ can also be described with a matrix notation. Let $I(n)$ be the table index produced at time n. Furthermore, let $\mathbf{S}_\mathbf{x}$ be a selection matrix of size N-by-2^B. Each row n of $\mathbf{S}_\mathbf{x}$ corresponds to a sample time index $n-1$, and each column I corresponds to a correction table entry $\mathbf{e}_{I-1}$ (row and column indices of $\mathbf{S}_\mathbf{x}$ start at 1 while the first sample is at time $n = 0$ and the first table entry is $\mathbf{e}_0$). If the table index for the time index n is I, then the element $[\mathbf{S}_\mathbf{x}]_{n+1,\,I+1}$ is set to one and the remaining entries in the same row are zero:

$$[\mathbf{S}_\mathbf{x}]_{n+1,\,I+1} \triangleq \begin{cases} 1, & I(n) = I, \\ 0, & \text{otherwise.} \end{cases} \tag{11.10}$$

The correction term at time n is $\mathbf{e}_{I(n)}$ and, by construction, the $(n+1)$-th row of $\mathbf{S}_\mathbf{x}$ sifts out $\mathbf{e}_{I(n)}$ when post-multiplied with $\mathbf{e}$. Hence, the matrix $\mathbf{S}_\mathbf{x}$ selects the appropriate correction term from the table $\mathbf{e}$ for each time n, and the correction for the entire sequence $\mathbf{x}$ can be written $\mathbf{S}_\mathbf{x}\,\mathbf{e}$.

In order to obtain a description for the sample vector $\mathbf{x}$ conforming with the selection matrix notation above, a column vector $\mathbf{r}$ of size 2^B is introduced. Since the employed bit mask is transparent, each index I is uniquely decodable to an ADC output level x_j. In fact, when backtracking the index building part of Figure 9.1,

it is evident that x_j is simply the b first bits in I. Thus, if the index I corresponds to the non-delayed sample $x(n)$ being equal to x_j, then let $\mathbf{r}_I = x_j$ so that the vector $\mathbf{x}$ can be written $\mathbf{x} = \mathbf{S}_\mathbf{x}\,\mathbf{r}$. An example is appropriate here. If $\mathbf{e}$ is a static table, i.e., $K = 0$, then $\mathbf{r}$ is just the vector of all possible ADC output levels, from x_0 to x_{2^b-1}, without repetitions. In the state-space case, when $K = 1$, then $\mathbf{r}$ is still composed of all ADC output levels. This time, each level is repeated 2^b times, so that for all indices I corresponding to the 'present sample' being equal to x_j it holds that $\mathbf{r}_I = x_j$.

Now, we can write the *corrected* ADC output $y(n)$ corresponding to the record $\mathbf{x}$ in a new vector $\mathbf{y} = [y(0)\ y(1)\ \ldots\ y(N-1)]^\mathrm{T}$ as

$$\mathbf{y} = \mathbf{x} + \mathbf{S}_\mathbf{x}\,\mathbf{e} = \mathbf{S}_\mathbf{x}\,\mathbf{r} + \mathbf{S}_\mathbf{x}\,\mathbf{e} = \mathbf{S}_\mathbf{x}\,(\mathbf{r} + \mathbf{e}). \tag{11.11}$$

The THD after correction is obtained by inserting (11.11) into (11.9), resulting in the expression

$$\mathrm{THD}_\mathbf{y} = \frac{1}{N}\sqrt{\mathbf{y}^*\,\mathbf{W}^*\,\mathbf{W}\,\mathbf{y}} = \frac{1}{N}\sqrt{(\mathbf{r}+\mathbf{e})^*\,\mathbf{S}_\mathbf{x}^*\,\mathbf{W}^*\,\mathbf{W}\,\mathbf{S}_\mathbf{x}\,(\mathbf{r}+\mathbf{e})}. \tag{11.12}$$

Having established a matrix expression for the THD after correction with a $(K+1)$-dimensional, B-bit indexed correction table $\mathbf{e}$, we are now interested in how the THD is affected when a *non-transparent* bit mask $\mathbf{q}$ is employed. If the bit mask maps $I \xrightarrow{\mathbf{q}} \tilde{I}$, then every occurrence of $\mathbf{e}_I$ in (11.12) should be replaced with $\tilde{\mathbf{e}}_{\tilde{I}}$ in order to get the resulting THD after correction with the table $\tilde{\mathbf{e}}$. Using the results of Section 10.3, the vector $\mathbf{e}$ in (11.12) should be replaced with the vector $\mathbf{f} = \mathbf{R}(\mathbf{q},\,\mathbf{a})\,\mathbf{e}$, with $\mathbf{R}(\mathbf{q},\,\mathbf{a})$ defined in (10.29), in order to evaluate the effect of reducing the address space. That is, the THD after correction with a table $\tilde{\mathbf{e}}$ – calibrated with the same set of calibration signals as $\mathbf{e}$ and with the same K, but this time with a specific bit mask $\mathbf{q}$ – is

$$\mathrm{THD}_{\tilde{\mathbf{y}}} = \frac{1}{N}\sqrt{(\mathbf{r} + \mathbf{R}(\mathbf{q},\,\mathbf{a})\,\mathbf{e})^*\,\mathbf{S}_\mathbf{x}^*\,\mathbf{W}^*\,\mathbf{W}\,\mathbf{S}_\mathbf{x}\,(\mathbf{r} + \mathbf{R}(\mathbf{q},\,\mathbf{a})\,\mathbf{e})}. \tag{11.13}$$

For example, we can evaluate the resulting THD after correction with a state-space ($K = 1$, $\mathbf{q} = \mathbf{1}$) table $\mathbf{e}$ versus that of a static table, simply by setting the appropriate bit mask $\mathbf{q}$ in (11.13), in this case $\mathbf{q} = [\mathbf{1}_b^\mathrm{T}\ \mathbf{0}_b^\mathrm{T}]^\mathrm{T}$, i.e., a vector of b ones and b zeros.

The function in (11.13) is the cost function to minimize. The expression within the square root of (11.13) is in fact a vector norm, and is thus nonnegative. Moreover, the square root function is monotonically increasing for nonnegative arguments. Minimizing $\mathrm{THD}_{\tilde{\mathbf{y}}}$ is therefore the same as minimizing the square $\mathrm{THD}_{\tilde{\mathbf{y}}}^2$, which is indeed a cost function of the form (11.1) (the normalization with N^2 does not change the minimization problem). The constraint is that a bit mask $\mathbf{q}$ of β ones and $B - \beta$ zeros is the only allowed solution, which was specified in (11.2).

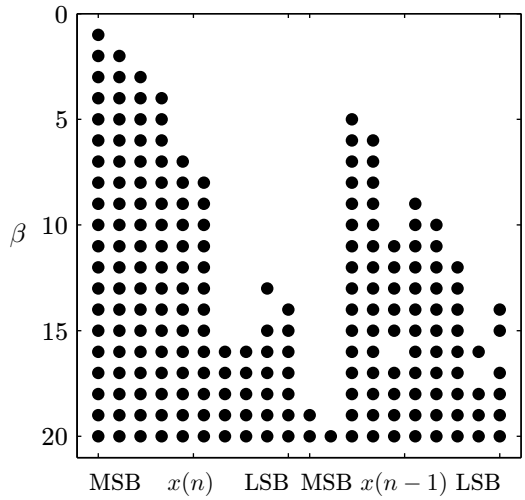


Figure 11.1: Exemplary optimization results for THD. Each row corresponds to a specific choice of β, and the dots indicates which positions in the bit mask $\mathbf{q}$ should be set to '1'. The results are obtained for $f_0 = 3.01$ MHz.

The optimization problem for minimizing the THD then becomes

$$\begin{cases} \min\limits_{\mathbf{q}} & (\mathbf{r} + \mathbf{R}(\mathbf{q},\, \mathbf{a})\, \mathbf{e})^* \, \mathbf{S}_{\mathbf{x}}^* \, \mathbf{W}^* \, \mathbf{W} \, \mathbf{S}_{\mathbf{x}} \, (\mathbf{r} + \mathbf{R}(\mathbf{q},\, \mathbf{a})\, \mathbf{e}) \\ \text{s.t.} & \sum\limits_i \mathbf{q}_i = \beta \quad \text{and} \quad \mathbf{q}_i \in \{0,\, 1\}, \quad i = 1,\, 2,\, \ldots,\, B. \end{cases} \tag{11.14}$$

The above expression is the optimization problem for finding the optimal β-bit bit mask such that the THD is minimized, given that the address bits must be taken from the samples $x(n)$ through $x(n - K)$.

Results

The optimization problem (11.14) has been solved and evaluated for an exemplary scenario. The same experimental ADC data that was used in Section 9.2 has been used here (see Appendix B.1). A state-space table is considered in this example. The table, denoted $\mathbf{e}$, is indexed using an index building structure with one delay element (i.e., $K = 1$) and a transparent bit mask (cf. Figure 9.1). Hence, the index I is $B = 20$ bits long and $\mathbf{e}$ consists of $M = 2^{20} = 1\,048\,576$ entries. The table is calibrated with a large number of different signals, all near full-scale sinewaves but each with a unique frequency. The vector $\mathbf{a}$ represents the number of times each entry in $\mathbf{e}$ was updated during the calibration (cf. Section 9.1).

Next, an optimization frequency f_0 is selected and a near full-scale sinewave record $\mathbf{x}$ of $N = 16\,384$ samples is taken; the frequency $f_0 = 3\,007\,273$ Hz is used

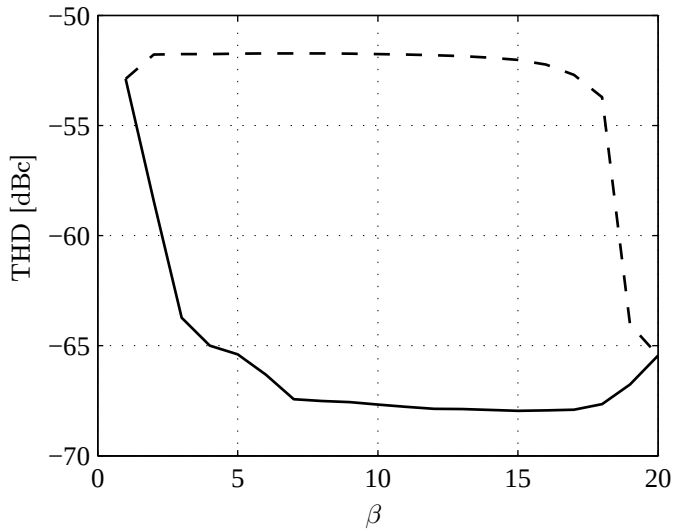


Figure 11.2: Resulting THD after correction, optimized for and evaluated at $f_0 = 3.01$ MHz, as a function of table index size β. The solid line shows the results using the best possible bit mask, while the dashed line shows the outcome of the worst choice.

in the results below. The matrices $\mathbf{S_x}$ and $\mathbf{W}$ are formed and the optimization problem (11.14) is solved – using an algorithm described in Chapter 12 – for all integers $\beta \in [1, 19]$ (the solutions for $\beta = 0$ and $\beta = 20$ are trivial, viz. $\mathbf{q}$ set to all zeros and all ones, respectively). Figure 11.1 illustrates the optimal bit masks for different choices of β: each row corresponds to a specific β and the dots indicates which of the positions in the bit mask $\mathbf{q}$ should be set to '1', or, which of the original 20 bits to use in a β-bit index $\tilde{I}$. For example, if a 10-bit index is desired, Figure 11.1 suggests that the 6 most significant bits from the present sample $x(n)$ should be selected, together with bits 8, 7, 5 and 4 from the previous sample $x(n-1)$, with 10 being the MSB. These 10 bits form the index $\tilde{I}$ used to address the table $\tilde{\mathbf{e}}$ of size $2^{10} = 1024$ entries. Note that Figure 11.1 illustrates the results for a specific ADC at a specific frequency f_0, and should not be taken to be optimal in general.

In Figure 11.2 the resulting THD after correction with a β-bit table is plotted. The solid line shows the results using the optimal choice of index bits for each β, as suggested in Figure 11.1. The THD is evaluated at the frequency f_0, i.e., the same frequency as the one for which the index was optimized. Somewhat surprisingly, the THD is not minimal at $\beta = 20$ bits, but rather at 15 bits. This phenomenon is most likely due to the fact that in our experiment the amount of calibration data is constant, so that a smaller table will have more calibration data *per table entry.* For example, a table indexed with 15 bits, thus having 2^{15} entries, will have 32 times

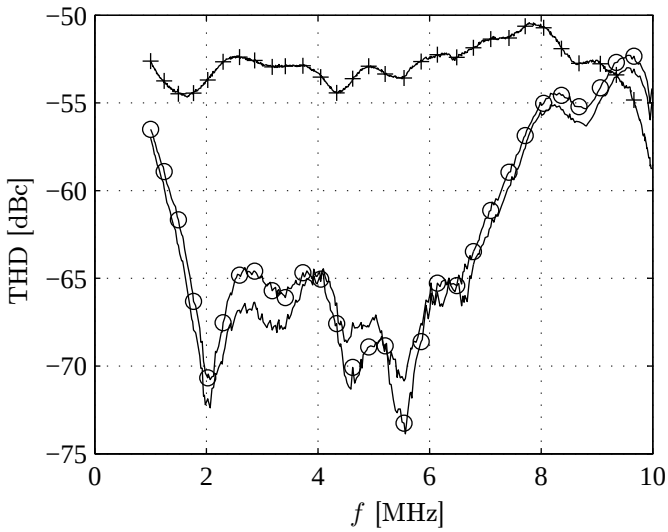


Figure 11.3: THD for the uncorrected ADC ('+'), THD after correction with a static 10-bit table ('○'), and THD after correction with a 10-bit table optimized for the frequency $f_0 = 3.01$ MHz (solid).

more calibration data per entry compared with a table of size 2^{20}. The conclusion is that given the fixed amount of calibration data, Figure 11.2 suggests that the performance in terms of THD at $f_0 = 3.01$ MHz improves when β is reduced from 20 to 15, but then deteriorates as β is further reduced.

In order to assess the importance of selecting the optimal bit mask, the resulting THD when selecting the *worst* bit mask has been calculated. The optimization was performed again, but this time maximizing instead of minimizing the cost function in (11.14). The dashed line in Figure 11.2 shows the resulting THD evaluated at the same frequency as the solid line. It is evident that the difference between choosing the best or worst bit-mask configuration implies a difference between 10 and 15 dB in THD after correction, in this experiment.

In Figure 11.3, the *evaluation* frequency is altered. The three curves show THD for the uncorrected ADC, THD after correction with a static 10-bit table, and THD after correction with a 10-bit table optimized for the frequency $f_0 = 3.01$ MHz. We see that near the optimization frequency the optimized table outperforms the static table, while this is not the case over the entire range. Thus, by clever selection of the index bits, we can gain a few dB in THD, *without increasing the table size.*

11.3 Maximizing SINAD

The signal-to-noise and distortion ratio (SINAD) is also defined in Section 1.4. Like in the case of THD above, the definition stipulates that the ADC under test should be exercised with a spectrally pure, large amplitude (near full-scale) sinewave, $s(t) = A\sin(2\pi f_0 t + \phi) + C$, with C and A chosen so that the signal is centered within and spans a major part of the ADC's analog input range. This time, however, the fundamental frequency f_0 does not have to line up with a DFT bin frequency. A record of N successive samples $x(0)$, $x(1)$, ..., $x(N-1)$ is taken from the ADC output. The basic requirement for the SINAD test method is that the input sinewave is pure enough so that noise input to the ADC is negligible. Then, the output from the ADC can be modeled as a sinewave plus a term containing all the distortion and noise in the output signal. The distortion and noise can be assumed to be a product of the quantization and non-idealities in the converter only, since the input was a pure sinewave. The SINAD is then the ratio of the rms signal to the rms distortion.

The sinewave part of the output signal $x(n)$ is found by calculating a sinewave least-squares fit, say $\check{s}(n)$, to the ADC data $x(n)$. Standardized methods for sinewave fitting are described in [Std1241, §4.1.4]. The SINAD is then defined as

$$\text{SINAD} = \frac{\text{RMS}_{\text{sig}}}{\text{RMS}_{\text{noise}}}, \tag{11.15}$$

where

$$\text{RMS}_{\text{sig}} = \frac{A}{\sqrt{2}} \tag{11.16}$$

and

$$\text{RMS}_{\text{noise}} = \left(\frac{1}{N} \sum_{n=0}^{N-1} \left(x(n) - \check{s}(n) \right)^2 \right)^{\frac{1}{2}}. \tag{11.17}$$

This is the SINAD for the *uncorrected* ADC at frequency f_0.

Let a correction system of the type described in Chapter 9 be applied to the ADC under test. Assume once again that a correction table $\mathbf{e}$ has been calibrated as described in Section 11.1, i.e., employing K delay elements and a transparent bit mask. The index building will then produce an index I for every sample time n. Denote the indices associated with the calibration samples with $\{I(n)\}_{n=0}^{N-1}$. Thus, when calculating the SINAD for the corrected ADC, $x(n)$ should be replaced by $y(n) = x(n) + \mathbf{e}_{I(n)}$ in (11.17), so that

$$\text{RMS}_{\text{noise},\, y} = \left(\frac{1}{N} \sum_{n=0}^{N-1} \left(x(n) + \mathbf{e}_{I(n)} - \check{s}(n) \right)^2 \right)^{\frac{1}{2}}. \tag{11.18}$$

The aim is now to find an alternative expression for the SINAD which is independent of the sample time n. Since $\mathrm{RMS}_{\mathrm{sig}}$ in (11.16) is already independent of n, $\mathrm{RMS}_{\mathrm{noise}}$ is the only one that must be rewritten. Therefore, assume that there exists a *reference table* $\mathbf{e}^0$, the same size as $\mathbf{e}$, but with entries such that we can write

$$\check{s}(n) \approx x(n) + \mathbf{e}^0_{I(n)}. \tag{11.19}$$

That is, the table $\mathbf{e}^0$ is such that when correcting the ADC output (obtained with $s(t)$ applied to the input) with $\mathbf{e}^0$, the result is approximately equal to a sinewave fit to the output signal. Inserting (11.19) in (11.18) results in

$$\begin{aligned} \mathrm{RMS}_{\mathrm{noise},\,y} &= \left(\frac{1}{N} \sum_{n=0}^{N-1} \left(x(n) + \mathbf{e}_{I(n)} - \left(x(n) + \mathbf{e}^0_{I(n)} \right) \right)^2 \right)^{\frac{1}{2}} \\ &= \left(\frac{1}{N} \sum_{n=0}^{N-1} \left(\mathbf{e}_{I(n)} - \mathbf{e}^0_{I(n)} \right)^2 \right)^{\frac{1}{2}}. \end{aligned} \tag{11.20}$$

Counting the occurrences of different indices I in $\{I(n)\}_{n=0}^{N-1}$ gives the distribution of the sampled evaluation signal over the tables $\mathbf{e}$ and $\mathbf{e}^0$. Thus, let $\mathbf{a}^0_I$ be the number of occurrences of the specific index I in $\{I(n)\}_{n=0}^{N-1}$, i.e., the number of times $\mathbf{e}_I$ and $\mathbf{e}^0_I$ is used in the sum (11.20). Note that the vector $\mathbf{a}^0$ is *not to be confused* with $\mathbf{a}$; the former is the distribution of the *evaluation samples* over the table, while the latter is the distribution of the *calibration samples*. Denote $\mathbf{A}_0 = \mathrm{diag}\{\mathbf{a}^0\}$, then (11.20) can be written as

$$\mathrm{RMS}_{\mathrm{noise},\,y} = \left(\frac{1}{N} \left(\mathbf{e} - \mathbf{e}^0 \right)^{\mathrm{T}} \mathbf{A}_0 \left(\mathbf{e} - \mathbf{e}^0 \right) \right)^{\frac{1}{2}}, \tag{11.21}$$

which, together with (11.15) and (11.16), is a matrix expression, independent of n, for the SINAD of the ADC corrected with a B-bit indexed table.

Once again we are interested in investigating the effects of reducing the index size from B bits to β bits. Using the same arguments that lead to (11.13) in the minimization of THD, the table $\mathbf{e}$ should be replaced with $\mathbf{f} = \mathbf{R}(\mathbf{q},\, \mathbf{a})\, \mathbf{e}$. As before, $\mathbf{q}$ is the considered bit mask and $\mathbf{a}$ is the distribution of the *calibration samples* over $\mathbf{e}$. Inserting this into (11.21), the root-mean-square noise can be written

$$\mathrm{RMS}_{\mathrm{noise},\,\tilde{y}} = \left(\frac{1}{N} \left(\mathbf{R}(\mathbf{q},\, \mathbf{a})\, \mathbf{e} - \mathbf{e}^0 \right)^{\mathrm{T}} \mathbf{A}_0 \left(\mathbf{R}(\mathbf{q},\, \mathbf{a})\, \mathbf{e} - \mathbf{e}^0 \right) \right)^{\frac{1}{2}}. \tag{11.22}$$

Thus, the RMS noise is expressed as the weighted RMS difference between tables $\mathbf{f}$ and $\mathbf{e}^0$. The weighting emphasizes each table entry according to the entry's significance to the SINAD, that is, how many times it is used in the sum in (11.20). For example, in the static case ($K = 0$) this results in the table entries corresponding to quantization levels near the sinewave endpoints $C \pm A$ being emphasized, since

the evaluation signal $s(t)$ spends most time in those regions. Furthermore, $\mathbf{a}^0$ can be extended to represent the distribution of *several* sinewave signals at different frequencies by simply adding the individual distribution of each frequency. The effect will then be that the mean SINAD over those frequencies is optimized.

Since the amplitude A in (11.16) is independent of the bit mask $\mathbf{q}$, maximizing the SINAD (11.15) is equivalent to *minimizing* the noise (11.22). Omitting the square root (which is monotonically increasing) and the normalization with N, we end up with a cost function of the form (11.1) again. The constraint (11.2) still applies, since we still want to find a bit mask that sifts out β bits. The following minimization problem can be posed:

$$\begin{cases} \min\limits_{\mathbf{q}} & (\mathbf{R}(\mathbf{q},\,\mathbf{a})\,\mathbf{e} - \mathbf{e}^0)^{\mathrm{T}}\,\mathbf{A}_0\,(\mathbf{R}(\mathbf{q},\,\mathbf{a})\,\mathbf{e} - \mathbf{e}^0) \\ \text{s.t.} & \sum\limits_i \mathbf{q}_i = \beta \quad \text{and} \quad \mathbf{q}_i \in \{0,\,1\}, \quad i = 1,\,2,\,\ldots,\,B. \end{cases} \tag{11.23}$$

That is, *minimize the weighted noise power with respect to the bit mask* $\mathbf{q}$ *consisting of* β *entries set to 1 and the remaining entries 0.*

The choice of reference table $\mathbf{e}^0$ naturally impacts on the optimization results. The reference table should be such that $x(n) + \mathbf{e}^0_{I(n)}$ matches a sinewave fit to the output *as closely as possible* and for all frequencies for which the performance shall be optimized. The restriction is that $\mathbf{e}^0$ must be the same size as $\mathbf{e}$, viz. 2^B entries. If the $\mathbf{e}^0$ is calibrated according to the method in Section 9.1, i.e., using Lloyd's optimal reconstruction levels, it minimizes the mean-squared error difference between the input and output during calibration, calculated over all calibration samples. Accordingly, the best reference table is that which is calibrated, using the methods of Section 9.1, for exactly those frequencies for which we want to optimize (11.23). Thus, if we calibrate the table $\mathbf{e}$ for our frequencies of interest, the best we can do is to let $\mathbf{e}^0 = \mathbf{e}$ and $\mathbf{a}^0 = \mathbf{a}$. The immediate interpretation of the cost function in (11.23) is then that the weighted difference between the original table $\mathbf{e}^0 = \mathbf{e}$ and the reduced size table $\tilde{\mathbf{e}}$ (represented by $\mathbf{R}(\mathbf{q},\,\mathbf{a})\,\mathbf{e}$) should be minimized.

Results

The optimization problem (11.23) has been solved and evaluated with experimental ADC data from an Analog Devices AD876, i.e., the same ADC as in Section 11.2. Two different sets of data are used which gives us two test cases:

Case A: The calibration and evaluation signals are sinusoids in the first Nyquist band, i.e., in $[0,\,f_s/2]$. The same data is used for the results presented in Section 11.2 (see Appendix B.1).

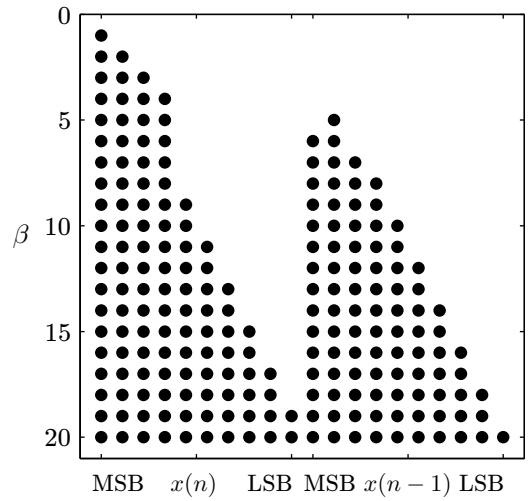
Case B: The calibration and evaluation signals are sinusoids in the fifth Nyquist band, i.e., in $[2f_s,\,5f_s/2]$ (see Appendix B.2).

In both cases, a state-space table $\mathbf{e}$ (i.e., a table addressed with $K = 1$ delay element and a transparent bit mask, $B = 20$) is calibrated using several large amplitude sinewaves at different frequencies distributed over the considered Nyquist band (first or fifth). The vector $\mathbf{a}$ is the corresponding distribution vector, i.e., the number of times each table entry was updated during calibration. The calibrated table $\mathbf{e}$ is also used for reference, i.e., we let $\mathbf{e}^0 = \mathbf{e}$. Furthermore, $\mathbf{a}^0$ is set equal to $\mathbf{a}$. This corresponds to evaluating the *average SINAD over all frequencies* for which the table $\mathbf{e}$ was calibrated, so that the SINAD is optimized over the same frequency range. The purpose of this is to achieve an optimization valid in a wide frequency range. The optimization problem (11.23) is solved using an algorithm that will be presented in Chapter 12 and for all integers $\beta \in [1, 19]$ (the solutions for $\beta = 0$ and $\beta = 20$ are trivial). The optimal bit masks for cases A and B are illustrated in Figure 11.4. The interpretation of the plots is that each row corresponds to a specific choice of β and the dots indicate which bits to use in a β-bit index $\tilde{I}$. We see that the results match closely for the two cases, but they are not identical. We also see that the results are not identical to those obtained while minimizing the THD.

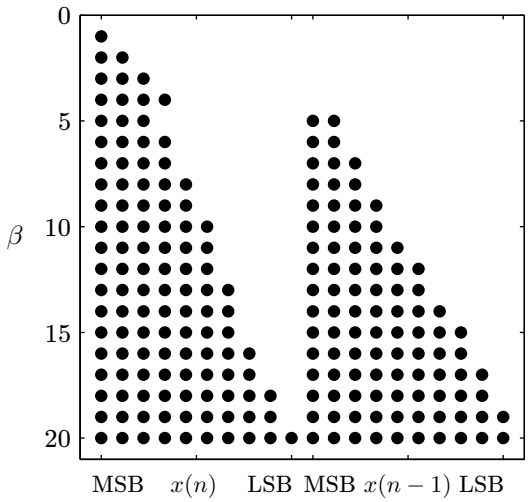
Next, the resulting SINAD after correction with a 2^β-size table $\tilde{\mathbf{e}}$ was evaluated. The SINAD was evaluated for different values of β and at the frequencies 3.01 MHz and 42.16 MHz in the cases A and B, respectively. For each value β, the index bits were selected according to the results of Figure 11.4. The solid lines in Figure 11.5 show that the performance only deteriorates slightly, or even improves, when β is reduced as low as 3. However, part of this is, as was pointed out in Section 11.2, due to the fact that in our experiment, the amount of calibration data is constant.

Just as in the case of THD in the previous section, the worst-case bit mask was tested, so that the importance of bit-mask selection could be assessed. The cost function in (11.23) was maximized instead of minimized. The dashed lines in Figure 11.5 shows the resulting SINAD evaluated at the same frequencies as the solid lines. The plots clearly show that selecting the appropriate bit mask improves the performance of the corrected ADC with up to 5 dB. The dash-dot lines are the SINAD of the ADC at the considered frequency without correction, included for reference.

Finally, Figure 11.6 shows the SINAD after correction with an optimized 10-bit table, compared with the SINAD after correction with a 10-bit static table ($K = 0$) and the uncorrected SINAD of the ADC. It is clear from the results that in case A, the optimized table performs significantly (approximately 2 dB) better than the static table in the higher frequencies. We also see that the optimized table does not deteriorate the ADC performance for frequencies close to 10 MHz, which the static table does. In the lower frequencies (except between 1 MHz and 1.5 MHz), on the other hand, the static table actually outperforms the 10-bit optimized table, but the difference is small (less than 1 dB). In case B, the optimized table outperforms the static table almost everywhere.

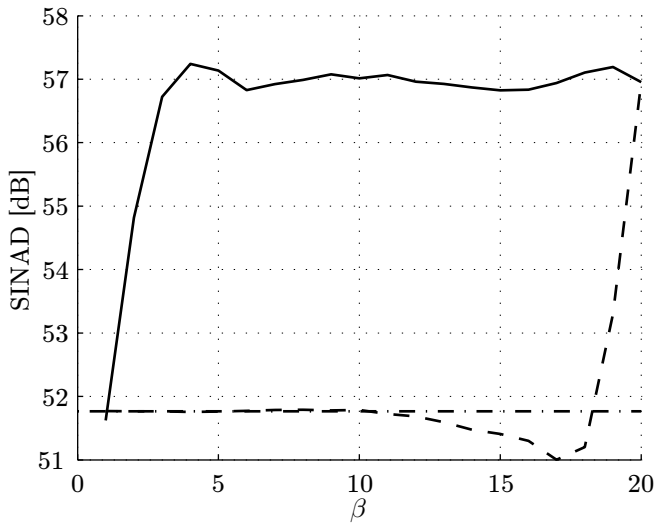


(a) Case A

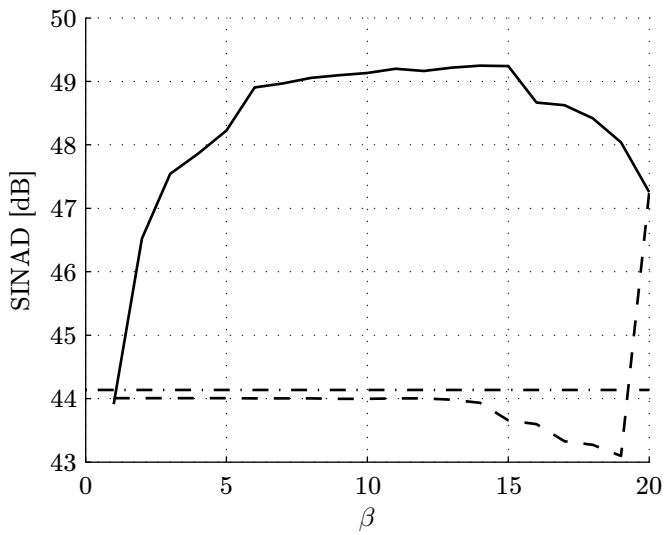


(b) Case B

Figure 11.4: Exemplary optimization results for SINAD. Each row corresponds to a specific choice of β, and the dots indicates which positions in the bit mask $\mathbf{q}$ should be set to '1'.

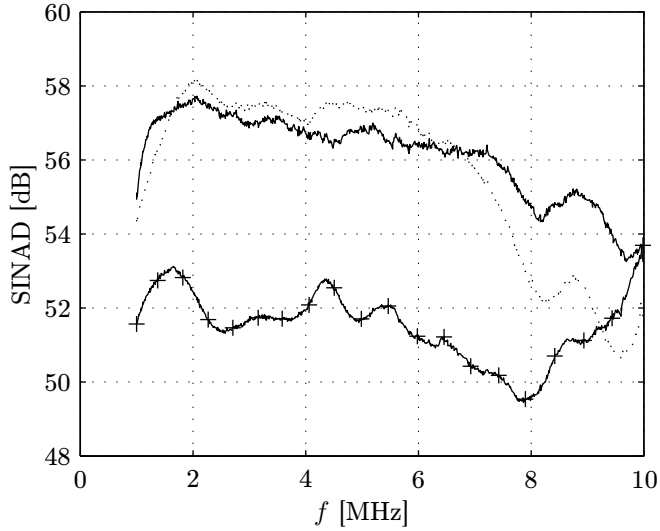


(a) Case A

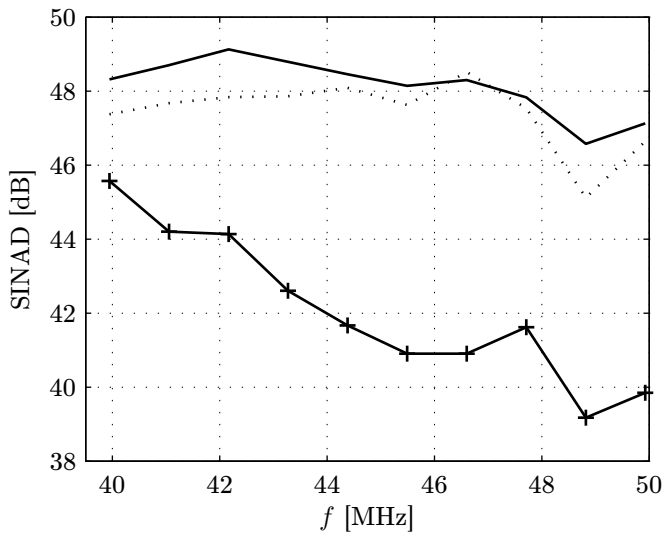


(b) Case B

Figure 11.5: Resulting SINAD after correction, evaluated at (a) $f = 3.01$ MHz and (b) $f = 42.16$ MHz, as a function of table index size β.



(a) Case A



(b) Case B

Figure 11.6: SINAD for the uncorrected ADC ('+'), SINAD after correction with a static 10-bit table (dotted), and SINAD after correction with a 10-bit optimized table (solid).

11.A Simplifications of the Cost Function

Both the optimization problems above ended up in rather complicated cost functions, viz. (11.14) and (11.23). It has already been identified that both cost functions are of the general form (11.1). When considering the special case of the calibration samples being uniformly distributed over the table $\mathbf{e}$, some simplifications can be done. It is reasonable to assume that a uniform distribution over the table $\mathbf{e}$ is achieved by using independent identically distributed calibration samples with a uniform distribution over the full-scale range of the ADC. A uniform distribution corresponds to the vector $\mathbf{a}$ being $\mathbf{a} = \alpha\mathbf{1}$. In this case, the following lemma applies.

Lemma 14 (Cost function for uniform distribution)**.** *Let $V(\mathbf{q})$ be the cost function (11.1), i.e., $V(\mathbf{q}) = (\mathbf{R}(\mathbf{q},\, \mathbf{a})\,\mathbf{e} + \mathbf{c})^*\mathbf{B}(\mathbf{R}(\mathbf{q},\, \mathbf{a})\,\mathbf{e} + \mathbf{c})$ where $\mathbf{B}$ is a Hermitian M-by-M matrix and $\mathbf{c}$ and $\mathbf{e}$ are real column vectors of length M. Let $\mathbf{R}(\mathbf{q},\, \mathbf{a})$ be the reduction matrix (10.29). Assume that $\mathbf{a} = \alpha\mathbf{1}$ ($\alpha \neq 0$). Then*

$$V(\mathbf{q}) = \mathbf{g}(\mathbf{q})^{\mathrm{T}}\mathbf{F}\,\mathbf{g}(\mathbf{q}) + \mathbf{c}^{\mathrm{T}}\mathbf{B}\,\mathbf{c}, \tag{11.24}$$

where $\mathbf{g}(\mathbf{q})$ is defined in (10.19) and the matrix $\mathbf{F}$ is

$$\mathbf{F} \triangleq \frac{1}{M^2}\left((\mathbf{H}\,\mathbf{e}\,\mathbf{e}^{\mathrm{T}}\,\mathbf{H}) \circ (\mathbf{H}\,\mathbf{B}\,\mathbf{H})\right) + \frac{2}{M}\,\mathrm{diag}\{\mathrm{diag}\{\mathbf{H}_B\,\mathbf{B}\,\mathbf{c}\,\mathbf{e}^{\mathrm{T}}\,\mathbf{H}_B\}\}. \tag{11.25}$$

The proof is provided in Appendix 11.B. Since the second term in (11.24) is constant, optimizing $V(\mathbf{q})$ – minimizing or maximizing – is equivalent to optimizing $V'(\mathbf{q}) = \mathbf{g}^{\mathrm{T}}\mathbf{F}\,\mathbf{g}$. (The explicit dependence of $\mathbf{g}$ on $\mathbf{q}$ is omitted for brevity.) This is a quadratic function in $\mathbf{g}$. The constraints for optimization should be that $\mathbf{g}$ depends on $\mathbf{q}$ as in (10.19) in addition to the constraints on $\mathbf{q}$ already stated in (11.2).

It is in fact possible to rewrite the function $V'(\mathbf{q})$ as a *linear* function, although not linear in $\mathbf{q}$. From matrix analysis, e.g., [HJ91] or [Lüt96], we know that for matrices $\mathbf{X}$, $\mathbf{Y}$ and $\mathbf{Z}$ of suitable dimensions,

$$\mathrm{vec}\{\mathbf{X}\,\mathbf{Y}\,\mathbf{Z}\} = (\mathbf{Z}^{\mathrm{T}} \otimes \mathbf{X})\,\mathrm{vec}\{\mathbf{Y}\}, \tag{11.26}$$

where vec is the vectorization operator, stacking the columns of a matrix in one single column vector. The vectorization of a scalar is simply the scalar itself, so

$$V'(\mathbf{q}) = \mathrm{vec}\{V'(\mathbf{q})\} = \mathrm{vec}\{\mathbf{g}^{\mathrm{T}}\mathbf{F}\,\mathbf{g}\} = (\mathbf{g}^{\mathrm{T}} \otimes \mathbf{g}^{\mathrm{T}})\,\mathrm{vec}\{\mathbf{F}\} = \mathrm{vec}\{\mathbf{F}\}^{\mathrm{T}}(\mathbf{g} \otimes \mathbf{g}). \tag{11.27}$$

Defining

$$\boldsymbol{\xi} = \mathbf{g} \otimes \mathbf{g}, \tag{11.28}$$

we can write an equivalent optimization problem using the cost function

$$V'(\boldsymbol{\xi}) = \mathrm{vec}\{\mathbf{F}\}^{\mathrm{T}}\boldsymbol{\xi} \tag{11.29}$$

and the constraints

$$\begin{cases} \boldsymbol{\xi} = \begin{bmatrix} 1 \\ \mathbf{q}_B \end{bmatrix} \otimes \cdots \otimes \begin{bmatrix} 1 \\ \mathbf{q}_1 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ \mathbf{q}_B \end{bmatrix} \otimes \cdots \otimes \begin{bmatrix} 1 \\ \mathbf{q}_1 \end{bmatrix} \\ \sum_{i=1}^{B} \mathbf{q}_i = \mathbf{1}^{\mathrm{T}}\mathbf{q} = \beta \\ \mathbf{q}_i \in \{0,\, 1\} \quad i = 1,\, 2,\, \ldots,\, B. \end{cases} \tag{11.30}$$

The optimization problem has been transformed from the rather complicated form (11.1) in B binary variables to the linear form (11.29) in 2^{2B} binary variables. The transformation was made possible by constraining the distribution to be uniform, i.e., $\mathbf{a} = \alpha\mathbf{1}$. It is of course a significant simplification to reduce the problem to a linear program, while the increase in the number of variables can make the form (11.29) infeasible; in the examples of Sections 11.2 and 11.3, M is 2^{20} and the transformation to (11.29) would render a problem in $M^2 = 2^{40}$ variables.

Avoiding the Kronecker-type Constraint

Although the new cost function (11.29) is linear, the constraints (11.30) are still nonlinear because of the Kronecker-type constraint

$$\boldsymbol{\xi} = \begin{bmatrix} 1 \\ \mathbf{q}_B \end{bmatrix} \otimes \cdots \otimes \begin{bmatrix} 1 \\ \mathbf{q}_1 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ \mathbf{q}_B \end{bmatrix} \otimes \cdots \otimes \begin{bmatrix} 1 \\ \mathbf{q}_1 \end{bmatrix} \tag{11.31}$$

This constraint descends from the definition of $\mathbf{g}$ in (10.19),

$$\mathbf{g} = \begin{bmatrix} 1 \\ \mathbf{q}_B \end{bmatrix} \otimes \begin{bmatrix} 1 \\ \mathbf{q}_{B-1} \end{bmatrix} \otimes \cdots \otimes \begin{bmatrix} 1 \\ \mathbf{q}_1 \end{bmatrix}. \tag{11.32}$$

However, it is possible to reformulate this constraint in a linear matrix equality constraint, which will be shown in the sequel.

Let us start by defining a vector

$$\boldsymbol{\gamma}^{(i)\perp} = \underset{B}{\begin{bmatrix} 1 \\ 1 \end{bmatrix}} \otimes \cdots \otimes \underset{i+1}{\begin{bmatrix} 1 \\ 1 \end{bmatrix}} \otimes \underset{i}{\begin{bmatrix} 1 \\ 0 \end{bmatrix}} \otimes \underset{i-1}{\begin{bmatrix} 1 \\ 1 \end{bmatrix}} \otimes \cdots \otimes \underset{1}{\begin{bmatrix} 1 \\ 1 \end{bmatrix}} \tag{11.33}$$

which, by construction, has zeros where $\mathbf{q}_i$ contributes to $\mathbf{g}$, and ones elsewhere. That is, if position j in $\mathbf{g}$ depends on $\mathbf{q}_i$ through (11.32), then the j-th element $\boldsymbol{\gamma}_j^{(i)\perp}$ is 0, otherwise it is 1. Let $\boldsymbol{\gamma}^{(i)} = \mathbf{1} - \boldsymbol{\gamma}^{(i)\perp}$ which now, of course, has ones where $\mathbf{q}_i$ contributes to $\mathbf{g}$. This vector possesses some properties that will be useful.

Consider the scalar product $\boldsymbol{\gamma}^{(i)\mathrm{T}}\mathbf{g}$. Assume that $\mathbf{q}$ has exactly β ones and $B-\beta$ zeros. From the definitions of $\boldsymbol{\gamma}^{(i)}$ and $\mathbf{g}$ we have

$$\boldsymbol{\gamma}^{(i)\mathrm{T}}\mathbf{g} = (\mathbf{1} - \boldsymbol{\gamma}^{(i)\perp})^{\mathrm{T}}\mathbf{g} = \mathbf{1}^{\mathrm{T}}\mathbf{g} - \boldsymbol{\gamma}^{(i)\perp\mathrm{T}}\mathbf{g}. \tag{11.34}$$

The one-vector $\mathbf{1}$ of 2^B ones can be written as an iterated Kronecker product with B factors

$$\mathbf{1} = \begin{bmatrix}1\\1\end{bmatrix} \otimes \cdots \otimes \begin{bmatrix}1\\1\end{bmatrix}, \tag{11.35}$$

and, using the property of the Kronecker product that $\mathbf{A}\,\mathbf{C}\otimes\mathbf{B}\,\mathbf{D} = (\mathbf{A}\otimes\mathbf{B})(\mathbf{C}\otimes\mathbf{D})$, we can write

$$\begin{aligned}\mathbf{1}^{\mathrm{T}}\mathbf{g} &= \left(\begin{bmatrix}1\\1\end{bmatrix} \otimes \cdots \otimes \begin{bmatrix}1\\1\end{bmatrix}\right)^{\mathrm{T}} \left(\begin{bmatrix}1\\\mathbf{q}_B\end{bmatrix} \otimes \begin{bmatrix}1\\\mathbf{q}_{B-1}\end{bmatrix} \otimes \cdots \otimes \begin{bmatrix}1\\\mathbf{q}_1\end{bmatrix}\right)\\ &= \left(\begin{bmatrix}1 & 1\end{bmatrix}\begin{bmatrix}1\\\mathbf{q}_B\end{bmatrix}\right) \otimes \cdots \otimes \left(\begin{bmatrix}1 & 1\end{bmatrix}\begin{bmatrix}1\\\mathbf{q}_1\end{bmatrix}\right) = \prod_{k=1}^{B}(1+\mathbf{q}_k) = 2^{\beta}.\end{aligned} \tag{11.36}$$

The last equality comes from the fact that $\mathbf{q}$ has β elements equal to one, so that β factors in the product are equal to 2 and the remaining factors are unity. In a similar fashion we can calculate $\boldsymbol{\gamma}^{(i)\perp\mathrm{T}}\mathbf{g}$ as

$$\begin{aligned}\boldsymbol{\gamma}^{(i)\perp\mathrm{T}}\mathbf{g} &= \left(\underbrace{\begin{bmatrix}1\\1\end{bmatrix}}_{B} \otimes \cdots \otimes \underbrace{\begin{bmatrix}1\\1\end{bmatrix}}_{i+1} \otimes \underbrace{\begin{bmatrix}1\\0\end{bmatrix}}_{i} \otimes \underbrace{\begin{bmatrix}1\\1\end{bmatrix}}_{i-1} \otimes \cdots \otimes \underbrace{\begin{bmatrix}1\\1\end{bmatrix}}_{1}\right)^{\mathrm{T}} \times\\ &\qquad \left(\begin{bmatrix}1\\\mathbf{q}_B\end{bmatrix} \otimes \begin{bmatrix}1\\\mathbf{q}_{B-1}\end{bmatrix} \otimes \cdots \otimes \begin{bmatrix}1\\\mathbf{q}_1\end{bmatrix}\right)\\ &= \prod_{k\neq i}(1+\mathbf{q}_k).\end{aligned} \tag{11.37}$$

The outcome will be one out of two possible numbers, depending on wether the vector element $\mathbf{q}_i$ is one or zero. If $\mathbf{q}_i = 1$, the product in the last equality of (11.37) will be over $\beta - 1$ factors with value 2. Else, if $\mathbf{q}_i = 0$, (11.37) is a product over β factors with value 2. Hence, $\boldsymbol{\gamma}^{(i)\perp\mathrm{T}}\mathbf{g}$ is 2^{β} or $2^{\beta-1}$ when $\mathbf{q}_i$ is 0 or 1, respectively. To summarize the discussion, we conclude that

$$\boldsymbol{\gamma}^{(i)\mathrm{T}}\mathbf{g} = \begin{cases}2^{\beta} - 2^{\beta} = 0, & \mathbf{q}_i = 0\\ 2^{\beta} - 2^{\beta-1} = 2^{\beta-1}, & \mathbf{q}_i = 1,\end{cases} \tag{11.38}$$

or

$$\boldsymbol{\gamma}^{(i)\mathrm{T}}\mathbf{g} = 2^{\beta-1}\mathbf{q}_i \tag{11.39}$$

The relation (11.39) can be used to formulate an alternative to the Kronecker-type constraint on $\boldsymbol{\xi}$ in (11.30). Since $\boldsymbol{\xi} = \mathbf{g}\otimes\mathbf{g}$ we define the vector $\bar{\boldsymbol{\gamma}}^{(i)} = \boldsymbol{\gamma}^{(i)}\otimes\boldsymbol{\gamma}^{(i)}$ and observe that

$$\begin{aligned}\bar{\boldsymbol{\gamma}}^{(i)\mathrm{T}}\boldsymbol{\xi} &= (\boldsymbol{\gamma}^{(i)} \otimes \boldsymbol{\gamma}^{(i)})^{\mathrm{T}}(\mathbf{g}\otimes\mathbf{g}) = (\boldsymbol{\gamma}^{(i)\mathrm{T}}\mathbf{g}) \otimes (\boldsymbol{\gamma}^{(i)\mathrm{T}}\mathbf{g}) = (\boldsymbol{\gamma}^{(i)\mathrm{T}}\mathbf{g})^2\\ &= (2^{\beta-1}\mathbf{q}_i)^2 = /\mathbf{q}_i \in \{0,\,1\} \rightarrow \mathbf{q}_i^2 = \mathbf{q}_i/ = 2^{2\beta-2}\mathbf{q}_i,\end{aligned} \tag{11.40}$$

where the relation (11.39) was used. Stacking the vectors $\bar{\boldsymbol{\gamma}}^{(B)\mathrm{T}}$ through $\bar{\boldsymbol{\gamma}}^{(1)\mathrm{T}}$ in a matrix $\bar{\boldsymbol{\Gamma}}$, we can form the linear matrix equality constraint

$$\bar{\boldsymbol{\Gamma}}\,\boldsymbol{\xi} = \begin{bmatrix} \bar{\boldsymbol{\gamma}}^{(B)\mathrm{T}} \\ \bar{\boldsymbol{\gamma}}^{(B-1)\mathrm{T}} \\ \vdots \\ \bar{\boldsymbol{\gamma}}^{(1)\mathrm{T}} \end{bmatrix} \boldsymbol{\xi} = 2^{2\beta-2}\mathbf{q}. \tag{11.41}$$

Finally, this can be used to formulate a new set of constraints, replacing (11.30). The new constraints are

$$\begin{cases} \bar{\boldsymbol{\Gamma}}\,\boldsymbol{\xi} = 2^{2\beta-2}\mathbf{q} \\ \sum_{i=1}^{B} \mathbf{q}_i = \mathbf{1}^{\mathrm{T}}\mathbf{q} = \beta \\ \mathbf{q}_i \in \{0,\,1\} \quad i = 1,\,2,\,\ldots,\,B \\ \boldsymbol{\xi}_j \in \{0,\,1\} \quad j = 1,\,2,\,\ldots,\,2^B. \end{cases} \tag{11.42}$$

We have in this section shown that it is possible to reformulate the general optimization problem (11.1) with constraints (11.2) into a linear binary optimization problem (11.29) with linear constraints (11.42). The transformation of the cost function depended on the assumption that the calibration samples were uniformly distributed over the table $\mathbf{e}$, i.e., $\mathbf{a} = \alpha\mathbf{1}$.

11.B Proof of Lemma 14

Let $V(\mathbf{q})$ be the cost function (11.1), i.e.,

$$V(\mathbf{q}) = (\mathbf{R}(\mathbf{q},\,\mathbf{a})\,\mathbf{e} + \mathbf{c})^{*}\mathbf{B}(\mathbf{R}(\mathbf{q},\,\mathbf{a})\,\mathbf{e} + \mathbf{c}).$$

When $\mathbf{a} = \alpha\mathbf{1}$ with $\alpha \neq 0$, we have from (10.29) that

$$\begin{aligned} \mathbf{R}(\mathbf{q},\,\alpha\mathbf{1}) &= \mathrm{diag}\{\mathbf{P}(\mathbf{q})\,\alpha\mathbf{1}\}^{\dagger}\,\mathbf{P}(\mathbf{q})\;\mathrm{diag}\{\alpha\mathbf{1}\} \\ &= \mathrm{diag}\{\frac{1}{M}\,\mathbf{H}_B\,\mathbf{G}\,\mathbf{H}_B\,\alpha\mathbf{1}\}^{\dagger}\,\frac{1}{M}\,\mathbf{H}_B\,\mathbf{G}\,\mathbf{H}_B\;\mathrm{diag}\{\alpha\mathbf{1}\} \\ &= \mathrm{diag}\{\mathbf{H}_B\,\mathbf{G}\,\mathbf{H}_B\,\mathbf{1}\}^{\dagger}\,\mathbf{H}_B\,\mathbf{G}\,\mathbf{H}_B\,\mathbf{I}. \end{aligned} \tag{11.43}$$

The matrix $\mathbf{G}$ is defined in (10.20). Now, since $\mathbf{1} = \mathbf{h}_0$, i.e., the first column of $\mathbf{H}_B$, we have that the first factor is

$$\begin{aligned} \mathrm{diag}\{\mathbf{H}_B\,\mathbf{G}\,\mathbf{H}_B\,\mathbf{H}_B\,\mathbf{u}_1\}^{\dagger} &= /\mathbf{H}_B\,\mathbf{H}_B = M\mathbf{I}/ = \mathrm{diag}\{M\mathbf{H}_B\,\mathbf{G}\,\mathbf{u}_1\}^{\dagger} \\ &= /\mathbf{G}\,\mathbf{u}_1 = \mathbf{u}_1/ = \mathrm{diag}\{M\mathbf{H}_B\,\mathbf{u}_1\}^{\dagger} = M^{-1}\,\mathrm{diag}\{\mathbf{1}\}^{\dagger} \\ &= M^{-1}\mathbf{I}^{\dagger} = M^{-1}\mathbf{I}. \end{aligned} \tag{11.44}$$

Thus, we have that

$$\mathbf{R}(\mathbf{q},\, \alpha\mathbf{1}) = \frac{1}{M}\, \mathbf{H}_B\, \mathbf{G}\, \mathbf{H}_B \tag{11.45}$$

which in fact is equal to $\mathbf{P}(\mathbf{q})$ in Lemma 12.

Expanding the quadratic form in $V(\mathbf{q})$ and inserting $\mathbf{a} = \alpha\mathbf{1}$ we have

$$\begin{aligned} V(\mathbf{q}) &= \left(\frac{1}{M}\, \mathbf{H}_B\, \mathbf{G}\, \mathbf{H}_B\, \mathbf{e} + \mathbf{c}\right)^* \mathbf{B} \left(\frac{1}{M}\, \mathbf{H}_B\, \mathbf{G}\, \mathbf{H}_B\, \mathbf{e} + \mathbf{c}\right) \\ &= \frac{1}{M^2}\, \mathbf{e}^*\, \mathbf{H}_B\, \mathbf{G}\, \mathbf{H}_B\, \mathbf{B}\, \mathbf{H}_B\, \mathbf{G}\, \mathbf{H}_B\, \mathbf{e} + \frac{1}{M}\, \mathbf{e}^*\, \mathbf{H}_B\, \mathbf{G}\, \mathbf{H}_B\, \mathbf{B}\, \mathbf{c} \\ &\quad + \frac{1}{M}\, \mathbf{c}^*\, \mathbf{H}_B\, \mathbf{G}\, \mathbf{H}_B\, \mathbf{B}\, \mathbf{e} + \mathbf{c}^*\, \mathbf{B}\, \mathbf{c} \\ &= /\mathbf{c} \text{ and } \mathbf{e} \text{ real} \Rightarrow\ \mathbf{c}^* = \mathbf{c}^{\mathrm{T}},\ \mathbf{e}^* = \mathbf{e}^{\mathrm{T}}/ \\ &= \frac{1}{M^2}\, \mathbf{e}^{\mathrm{T}}\, \mathbf{H}_B\, \mathbf{G}\, \mathbf{H}_B\, \mathbf{B}\, \mathbf{H}_B\, \mathbf{G}\, \mathbf{H}_B\, \mathbf{e} + \frac{2}{M}\, \mathbf{e}^{\mathrm{T}}\, \mathbf{H}_B\, \mathbf{G}\, \mathbf{H}_B\, \mathbf{B}\, \mathbf{c} + \mathbf{c}^{\mathrm{T}}\, \mathbf{B}\, \mathbf{c} \end{aligned} \tag{11.46}$$

The trace of a matrix product $\mathbf{X}\,\mathbf{Y}$ has the property that $\mathrm{Tr}\{\mathbf{X}\,\mathbf{Y}\} = \mathrm{Tr}\{\mathbf{Y}\,\mathbf{X}\}$. Also, the trace of a scalar a is $\mathrm{Tr}\{a\} = a$. We also know that for two vectors, $\mathbf{x}$ and $\mathbf{y}$, and two matrices, $\mathbf{X}$ and $\mathbf{Y}$, it holds that $\mathrm{Tr}\{\mathrm{diag}\{\mathbf{x}\}\,\mathbf{X}\,\mathrm{diag}\{\mathbf{y}\}\,\mathbf{Y}^{\mathrm{T}}\} = \mathbf{y}^{\mathrm{T}}(\mathbf{X} \circ \mathbf{Y})\mathbf{x}$, where $\circ$ is the Hadamard product[1] (see e.g., [HJ91, Lüt96]). Using these relations, the first term in (11.46) can be written as

$$\begin{aligned} \frac{1}{M^2}\, \mathbf{e}^{\mathrm{T}}\, \mathbf{H}_B\, \mathbf{G}\, \mathbf{H}_B\, \mathbf{B}\, \mathbf{H}_B\, \mathbf{G}\, \mathbf{H}_B\, \mathbf{e} &= \frac{1}{M^2}\ \mathrm{Tr}\{\mathbf{G}\, \mathbf{H}_B\, \mathbf{e}\, \mathbf{e}^{\mathrm{T}}\, \mathbf{H}\, \mathbf{G}\, \mathbf{H}\, \mathbf{B}\, \mathbf{H}\} \\ = \frac{1}{M^2}\, \mathbf{g}^{\mathrm{T}} \left((\mathbf{H}\, \mathbf{e}\, \mathbf{e}^{\mathrm{T}}\, \mathbf{H}) \circ (\mathbf{H}\, \mathbf{B}\, \mathbf{H})^{\mathrm{T}}\right) \mathbf{g} &= \frac{1}{M^2}\, \mathbf{g}^{\mathrm{T}} \left((\mathbf{H}\, \mathbf{e}\, \mathbf{e}^{\mathrm{T}}\, \mathbf{H}) \circ (\mathbf{H}\, \mathbf{B}\, \mathbf{H})\right) \mathbf{g}, \end{aligned} \tag{11.47}$$

and the second term becomes

$$\begin{aligned} \frac{2}{M}\, \mathbf{e}^{\mathrm{T}}\, \mathbf{H}_B\, \mathbf{G}\, \mathbf{H}_B\, \mathbf{B}\, \mathbf{c} &= \frac{2}{M}\ \mathrm{Tr}\{1\mathbf{e}^{\mathrm{T}}\, \mathbf{H}_B\, \mathbf{G}\, \mathbf{H}_B\, \mathbf{B}\, \mathbf{c}\} \\ &= 1\left((\mathbf{e}^{\mathrm{T}}\, \mathbf{H}_B) \circ (\mathbf{H}_B\, \mathbf{B}\, \mathbf{c})^{\mathrm{T}}\right) \mathbf{g} = \mathrm{diag}\{\mathbf{H}_B\, \mathbf{B}\, \mathbf{c}\, \mathbf{e}^{\mathrm{T}}\, \mathbf{H}_B\}^{\mathrm{T}}\, \mathbf{g}, \end{aligned} \tag{11.48}$$

where $\mathrm{diag}\{\mathbf{g}\} = \mathbf{G}$ (see (10.20)) has been used. Now the cost function $V(\mathbf{q})$ can be expressed as

$$\begin{aligned} V(\mathbf{q}) = \frac{1}{M^2}\, \mathbf{g}^{\mathrm{T}} \left((\mathbf{H}\, \mathbf{e}\, \mathbf{e}^{\mathrm{T}}\, \mathbf{H}) \circ (\mathbf{H}\, \mathbf{B}\, \mathbf{H})\right) \mathbf{g} \\ + \frac{2}{M}\ \mathrm{diag}\{\mathbf{H}_B\, \mathbf{B}\, \mathbf{c}\, \mathbf{e}^{\mathrm{T}}\, \mathbf{H}_B\}^{\mathrm{T}}\, \mathbf{g} + \mathbf{c}^{\mathrm{T}}\, \mathbf{B}\, \mathbf{c}. \end{aligned} \tag{11.49}$$

[1] The Hadamard product, also known as the Schur product, is the entrywise multiplication of the elements in two matrices of the same size.

Since $\mathbf{g}$ consists of only zeros and ones ($\mathbf{g} \in \mathbb{B}^M$), we can write the scalar product between a vector $\mathbf{x}$ and $\mathbf{g}$ as

$$\mathbf{x}^{\mathrm{T}}\mathbf{g} = \sum_{m=1}^{M} \mathbf{x}_m \, \mathbf{g}_m = \sum_{m=1}^{M} \mathbf{x}_m \, \mathbf{g}_m^2 = \mathbf{g}^{\mathrm{T}} \operatorname{diag}\{\mathbf{x}\} \, \mathbf{g}. \tag{11.50}$$

Inserting this into (11.49) we have

$$\begin{aligned} V(\mathbf{q}) &= \frac{1}{M^2} \, \mathbf{g}^{\mathrm{T}} \left((\mathbf{H}\,\mathbf{e}\,\mathbf{e}^{\mathrm{T}}\,\mathbf{H}) \circ (\mathbf{H}\,\mathbf{B}\,\mathbf{H}) \right) \mathbf{g} \\ &\quad + \mathbf{g}^{\mathrm{T}} \operatorname{diag}\{\operatorname{diag}\{\mathbf{H}_B \, \mathbf{B}\,\mathbf{c}\,\mathbf{e}^{\mathrm{T}}\,\mathbf{H}_B\}\}\,\mathbf{g} + \mathbf{c}^{\mathrm{T}}\,\mathbf{B}\,\mathbf{c} \\ &= \mathbf{g}^{\mathrm{T}} \Bigg(\frac{1}{M^2} \, \left((\mathbf{H}\,\mathbf{e}\,\mathbf{e}^{\mathrm{T}}\,\mathbf{H}) \circ (\mathbf{H}\,\mathbf{B}\,\mathbf{H}) \right) + \\ &\quad \frac{2}{M} \, \operatorname{diag}\{\operatorname{diag}\{\mathbf{H}_B \, \mathbf{B}\,\mathbf{c}\,\mathbf{e}^{\mathrm{T}}\,\mathbf{H}_B\}\} \Bigg) \mathbf{g} + \mathbf{c}^{\mathrm{T}}\,\mathbf{B}\,\mathbf{c}. \end{aligned} \tag{11.51}$$

Chapter 12

Suboptimal Solver

The optimization problems in the previous chapter – generalized in (11.1) – are difficult to solve optimally. They are in fact nonlinear binary problems. Under certain conditions it was shown in Appendix 11.A that the problem could be rewritten to a linear binary problem, but at the expense of a dramatic increase in the number of variables.

A heuristic algorithm for solving the optimization problem is proposed in this section. The method, based on successive deletion of bits, is a "greedy" type algorithm, which for each iteration deletes the bit that is locally best without considerations of global optimality. See for instance [NW88] for a thorough treatment of this class of algorithms.

First, recall the general cost function $V(\mathbf{q})$ in (11.1),

$$V(\mathbf{q}) = (\mathbf{R}(\mathbf{q},\, \mathbf{a})\, \mathbf{e} + \mathbf{c})^* \mathbf{B} (\mathbf{R}(\mathbf{q},\, \mathbf{a})\, \mathbf{e} + \mathbf{c}). \tag{12.1}$$

The proposed algorithm starts with a bit mask that selects all B bits in the index I – that is, we start with $\mathbf{q} = \mathbf{1}$. A bit mask that selects β bits is then found by deselecting one bit at a time, in an iterative fashion. In each iteration, the bit that upon removal results in the lowest cost function $V(\mathbf{q})$ is deselected. However, such an algorithm is prone to find local minima. Therefore, in each iteration the L different bit masks, resulting from deselecting L different bits, that yields the L lowest costs are used as starting points in the next iteration. The number L is a design variable. The algorithm is presented in Table 12.1. The optimal value $V(\mathbf{q}_{\text{opt}})$ is a non-increasing function of the design variable L, meaning that we are guaranteed that when increasing L, the solution will not get worse. However, it is not guaranteed that the algorithm will converge to a global minimum.

A rough estimate of the computational complexity of the algorithm is the number of times the cost function $V(\mathbf{q})$ must be computed. Let N_{alg} be the number of times the cost function must be evaluated in order to find a solution to the optimization problem using the algorithm in Table 12.1. This is obviously dependent on how many bits we want to use (β), how many bits we can choose from (B) and

Table 12.1: Algorithm used to solve the optimization problem.

1. Let $\mathbf{q}^{(\ell)}$, $\ell = 1, 2, \ldots, L$ be the $(B-1)$-bit bit masks that yields the L lowest costs $V(\mathbf{q}^{(\ell)})$ (found by trying all B possible combinations).
2. Set $r := B - 2$.
3. If $r < \beta$, go to 8.
4. Let $\Omega := \emptyset$, i.e., the empty set.
5. For all $\ell = 1, 2, \ldots, L$: Append to Ω all possible vectors $\mathbf{p}$ formed by changing one '1' into a '0' in $\mathbf{q}^{(\ell)}$.
6. Let $\mathbf{q}^{(\ell)}$, $\ell = 1, 2, \ldots, L$ be the L (unique) vectors $\mathbf{p} \in \Omega$ that results in the L lowest values for $V(\mathbf{p})$.
7. Let $r := r - 1$ and goto 3.
8. Let $\mathbf{q}_{\text{opt}} = \arg\min_{\ell=[1,\,L]} V(\mathbf{q}^{(\ell)})$.

how many solutions that are kept in each iteration (L). In fact, it is easy to see that

$$N_{\text{alg}} \leq B + \sum_{r=\beta}^{B-2} L\,(r+1) + L = \frac{L}{2}\,(B^2 - B - \beta^2 - \beta) + L + B. \tag{12.2}$$

The inequality comes from the fact that in each iteration, the number of *unique* vectors $\mathbf{p} \in \Omega$ is no more than $L\,(r+1)$, but possibly less. This can be compared with the number of times N_{es} the cost function $V(\mathbf{q})$ must be evaluated if an *exhaustive search* is performed. When selecting β out of B bits there are $\binom{B}{\beta}$ possible combinations, which must all be evaluated. Hence,

$$N_{\text{es}} = \binom{B}{\beta} = \frac{B!}{\beta!\,(B-\beta)!}. \tag{12.3}$$

In Figure 12.1 the complexity of the proposed algorithm and the complexity of an exhaustive search are plotted for $B = 20$, $L = 6$ and different values of β (these are the parameter values used in the exemplary results presented in Chapter 11). We see that the complexity is reduced with as much as two orders of magnitude when using the above algorithm. For low values of β, on the other hand, the figure suggests that an exhaustive search is more effective. However, for $\beta < B/2$ it is beneficial to 'reverse' the algorithm, so that it starts with an empty bit mask (only

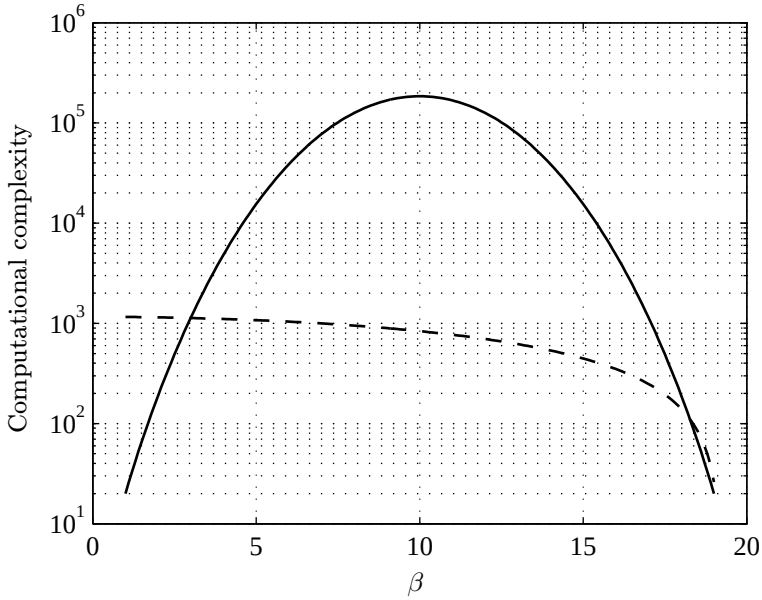


Figure 12.1: Comparison of the computational complexity (number of evaluations of $V(\mathbf{q})$) of the proposed algorithm (dashed) and of an exhaustive search (solid).

zeros) and then *adds* one bit in each iteration. The complexity of such an algorithm is the reverse of the dashed line in Figure 12.1, so that

$$N_{\text{alg, rev}} \leq \frac{L}{2}\left(B^2 - B - (B-\beta)^2 - (B-\beta)\right) + L + B. \tag{12.4}$$

Chapter 13

Post-correction Using Frequency Selective Tables: An Alternative Approach for Narrow-Band Signals

So far, we have tried to incorporate dynamic behaviour into the correction of ADCs by employing a state-space indexing of the correction table. When narrow-band signals are considered, one might think of the error of the ADC to be *frequency dependent*. A correction scheme which is explicitly dependent on the instantaneous signal frequency is a desirable countermeasure.

The correction scheme presented in this chapter utilizes a *frequency selective correction table*. This is accomplished by extending the usual one-dimensional correction table of classical look-up table correction (cf. Section 3.1) to a two-dimensional table, using both the present ADC output $x(n) = x_j$ and the present frequency region estimate $\hat{F}(n) = F_\ell \in \{F_1, \ldots, F_L\}$ for addressing. The correction system is depicted in Figure 13.1. The frequency region estimate is updated for each sample $x(n)$. This method can also be interpreted as selecting a specific one-dimensional correction table for each frequency region estimate F_ℓ. Thus, the corrected output $\hat{s}(n)$ is the table entry $\hat{s}_{j,\ell}$ associated with x_j and F_ℓ.

13.1 Frequency Region Estimator

A traditional way of constructing frequency estimators is by optimizing some criterion related to the frequency. One of the most commonly used methods is the method of maximum likelihood, or approximate variants thereof. In common for most frequency estimation methods is that the output frequency estimate is a continuous variable. Here, on the other hand, we consider the problem of finding the most probable *region* to hold the unknown frequency, out of a finite (small) set of regions.

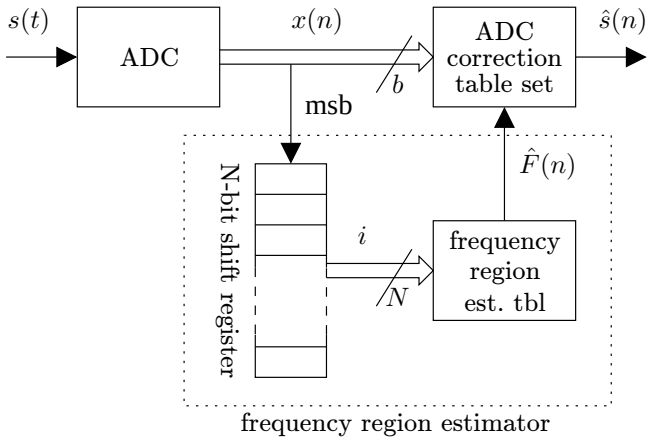


Figure 13.1: Correction system outline.

It has been illustrated [ASH00, And05] that there exists a high-performance frequency estimator of low complexity employing only 1-bit of the input signal. The use of 1-bit data also has the advantage that the estimator does not depend on the power of the input signal, i.e., no gain control is needed. Here, we are not limited to use 1-bit data but the resulting structure with a table look-up procedure is tractable since it meets the demand of a fast estimator of low complexity.

The frequency estimator input $z(n)$ is related to the most significant bit (MSB) of $x(n)$,

$$z(n) = \text{sign}(s(n)), \tag{13.1}$$

where

$$\text{sign}(x) \triangleq \begin{cases} 1 & x \geq 0; \\ -1 & x < 0. \end{cases} \tag{13.2}$$

(We can assume, without loss of generality, that the full-scale input range of the ADC is symmetric around zero.) By collecting N successive binary samples at each time instant n we can uniquely map one input sequence onto an integer $i \in \{0, \ldots, 2^N - 1\}$. The index i is then used as a pointer to an entry in a *frequency region estimation table*, see Figure 13.1. Finally, the i-th table entry contains a region estimate $\hat{F}(n) \in \{F_1, \ldots, F_L\}$, indicating that the instantaneous signal frequency is within the ℓ-th frequency region. The frequency regions F_ℓ are determined by the region center frequencies $\{f_\ell\}_{\ell=1}^{L}$ through the definition

$$F_\ell = \{f \in [0, 1/2) : |f - f_\ell| \leq |f - f_p|,\ p = 1, \ldots, L\} \tag{13.3}$$

where $\ell = 1, \ldots, L$. The center frequencies $\{f_\ell\}$ can be chosen arbitrarily, but the size and location of the regions $\{F_\ell\}$ is of course dependent on the actual choice.

As a frequency region estimate we choose the region that maximizes the probability of including the unknown frequency f_0 given the N 1-bit samples $z(n)$ through $z(n-N+1)$, that is

$$\hat{F}(n) = \arg\max_{\forall F_\ell} \Pr\{f_0 \in F_\ell | z(n), \ldots, z(n-N+1)\}, \tag{13.4}$$

where $\Pr\{\cdot\}$ is the probability function. Since $z(n)$ through $z(n-N+1)$ are binary variables (± 1) there is a finite number of possible combinations, viz. 2^N combinations. This observation makes it possible to precalculate an estimate $\hat{F} = F_\ell$ for each combination and store these in a table. A straightforward way to obtain the table is to use a training approach [ASH00, And05].

13.2 Correction Table

During ADC operation with correction engaged, the ADC output sample, $x(n)$, is mapped through the ADC correction table to a compensated output value $\hat{s}(n)$. The correction is determined by the present ADC output together with the current frequency region estimate, as depicted in Figure 13.1. Thus, the correction becomes[1]

$$\begin{gathered} s(t) \rightarrow (x_j,\, F_\ell) \rightarrow \hat{s}_{j,\,\ell} = \hat{s}(n) \\ \hat{s}_{j,\,\ell} \in \{\hat{s}_{i,\,p}\}_{(i,\,p)=(0,\,1)}^{(M-1,\,L)}\,. \end{gathered} \tag{13.5}$$

With this structure, the compensation is made dynamic, with table indexing depending on the frequency contents of the signal.

The table entries $\hat{s}_{j,\,\ell}$ should be selected such that the resulting conversion $s(t) \rightarrow \big(x_j,\, F_\ell\big) \rightarrow \hat{s}_{j,\,\ell} = \hat{s}(n)$ is "better" than without correction. The employed design criterion is again to minimize the mean squared error, $\mathrm{E}[(\hat{s}(n)-s(n))^2]$. Since the selection of $\hat{s}(n) = \hat{s}_{j,\,\ell}$ depends on the ADC output $x(n)$ and the frequency region estimate $\hat{F}(n)$, the criterion becomes

$$\hat{s}_{j,\,\ell} = \arg\min_{\hat{s}} \mathrm{E}\big[(\hat{s}-s(n))^2 \,\big|\, x(n) = x_j,\, \hat{F}(n) = F_\ell\big] \tag{13.6}$$

Quite analogously to the discussion in Section 9.1, $\hat{s}_{j,\,\ell}$ should be estimated as the mean value of all input samples, $s(n) = s(t)\big|_{t=n\,T_s}$, that were quantized into the value x_j while the frequency region estimate was equal to F_ℓ. Obviously, an estimate of the calibration signal input to the ADC under test is needed, and several methods have been proposed earlier; some of these have already been dealt with in Section 3.4.

[1] The correction table is here described as a *replacement table* such that the ADC output $x(n)$ is replaced with the new value $\hat{s}(n)$. It can, however, just as well be described as a table of *correction terms* added to the ADC output, so that $\hat{s}(n) = x(n) + \varepsilon(n)$ (cf. Figure 3.4(a) and Figure 3.4(b)).

13.3 Performance

Once again, the experimental ADC data described in Appendix B was used to evaluate the proposed method. The ADC correction table was calibrated using sinusoid calibration signals at several different frequencies. The calibration signal estimate $\hat{s}(n)$ was obtained using the optimal filtering method proposed in [HSP00].

Spurious-free dynamic range (SFDR) and signal-to-noise and distortion ratio (SINAD) are used to evaluate the method, and the results are presented in Figures 13.2 and 13.3, respectively. The results for static correction ($K = 0$) and for the uncompensated ADC are also presented in the figures.

The frequency-selective correction was evaluated for two test cases: the first case involving 8 frequency regions ($L = 8$) and the second case having 16 regions ($L = 16$). In both cases N was set to 16 and the L region center frequencies were distributed uniformly over the Nyquist range, resulting in uniform frequency regions. The results indicate that the frequency-selective correction method is superior to the frequency-static method in general, but also that increasing the number of frequency ranges L from 8 to 16 does not give any significant improvement. We see from the plots that the SFDR is improved with between zero and 7 dB, while the SINAD in general is improved with less than 1 dB, both compared with the results obtained using static correction. Furthermore, it is interesting to see that the static correction yields *deterioration* of the ADC performance at frequencies near the Nyquist rate, while for the frequency selective correction methods this is not the case.

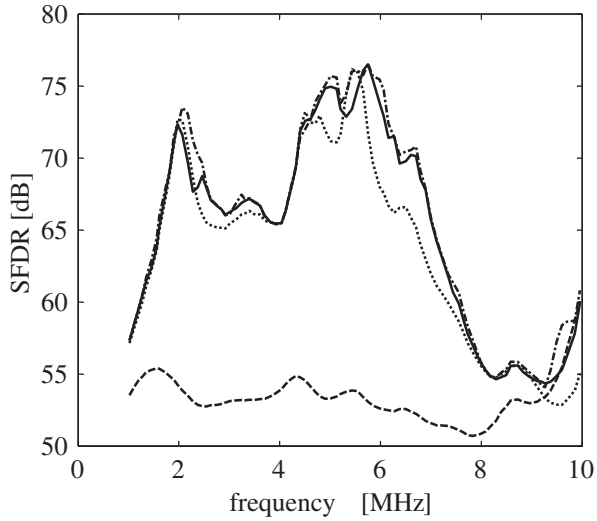


Figure 13.2: SFDR for frequency-selective correction with $L = 8$ (solid) and $L = 16$ (dash-dot) compared with static correction (dotted) and uncompensated ADC (dashed).

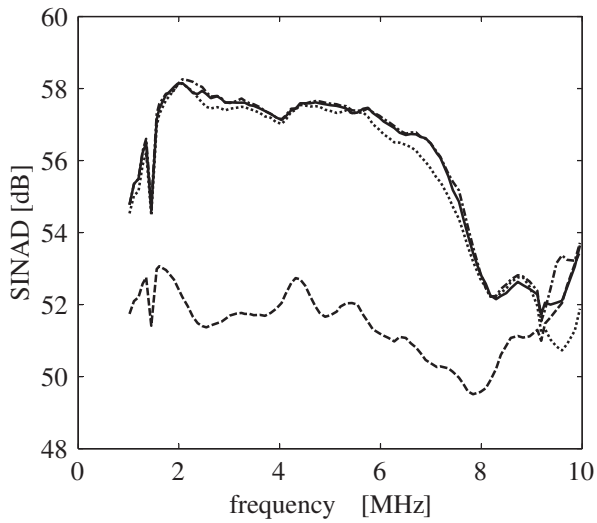


Figure 13.3: SINAD for frequency-selective correction with $L = 8$ (solid) and $L = 16$ (dash-dot) compared with static correction (dotted) and uncompensated ADC (dashed).

Chapter 14

Conclusions

14.1 Summary

In this thesis the topic of post-correction of ADCs has been discussed. The work has been focused on correction methods based on look-up tables.

In the first part of the thesis, an overview to the state of the art on ADC error correction methods was given. The part touched upon look-up table (LUT) methods, small- and large-scale dithering and model inversion methods. The techniques were in most cases only briefly described, but several references to the literature were given.

The second part of the thesis was dedicated to novel contributions to the art of LUT post-correction, with emphasis on methods with close relationship to estimation and information theory. Chapter 6 reviewed the most commonly used distortion criteria, and the implications in terms of quantizer design and post-correction were given. Following that, the problem of estimating an ADC transfer function was revised in Chapter 7. The setting of the problem was that an ADC was calibrated using a calibration signal applied to the input. The exact calibration signal was unknown, but a perturbed version of the calibration signal was provided as a reference signal. The problem of interest was to estimate the optimal reconstruction levels of the quantizer from the reference measurements. Using a statistical quantization model, the problem was rewritten as a classical constant-in-noise estimation problem. Two plausible noise assumptions were suggested, and estimators were found for both cases. The first scenario resulted in a Gaussian-noise problem, easily solved using standard methods. The second scenario, arising from the classical staircase quantizer model, resulted in a non-linear estimator based on order statistics. Optimality for the non-linear estimator could not be proven, but it was conjectured that it is an asymptotically efficient estimator. This was concluded by comparing the estimator performance with the Cramér–Rao bound and with numerically obtained maximum-likelihood estimates. Numerical simulations were performed and reported to support the theories.

The theoretical performance of an ADC after correction was derived in Chapter 8. The effects of limited resolution in the correction terms were investigated. Furthermore, the outcome of a post-correction applied to an ADC with DNL errors was explored. The results were combined to expressions for the theoretical SINAD and ENOB for an ADC with post-correction as a function of four parameters. These were the number of bits in the ADC, the variance of the intrinsic DNL, the resolution of the correction terms and the ADC input noise variance. Simulation results were provided to confirm the theories. Also, evaluations using experimental ADC data supported the theories, but pointed out important issues regarding the random noise.

Finally, the third part introduced a generalized approach for ADC correction based on multidimensional tables. The correction method used several samples in sequence from the ADC to form the address, or index, to the table. The rationale for introducing an index dependent not only on the 'present sample,' but also on the signal history, was that ADC errors in general exhibit a dynamic behavior. That is, the errors that the correction scheme is to mitigate are dependent on the dynamics of the signal, e.g., signal history. However, the table size grows exponentially with the number of samples incorporated into the index, which has been pointed out before. Therefore, the novel generalized scheme also comprises *bit-masking* in order to reduce the address space, and thereby the table size. The bit-masking operation selects a subset of the available bits. Through the use of a bit mask, the memory requirements are reduced while the correction remains dynamic, at least to some extent.

The experimental results showed that a correction scheme using multidimensional indexing can outperform a static correction, in terms of SFDR, SINAD and THD. It was also shown that significant performance improvements can be achieved even without increasing the actual size of the look-up table, simply by letting some of the index bits come from delayed samples.

When introducing the bit-masking concept, the question of which bits to select in the table index arises. One of the contributions of this work is an *analysis framework* for analyzing the effect of different bit allocations. It has been shown that a correction table indexed with all bits from the present sample and a specific number (K) of delayed samples, can be used to 'mimic' any correction scheme with a more *restrictive* bit mask, i.e., a table indexed with any subset of the bits used to index the original table. The relationship between the original table and the new table was derived, and was shown to be linear. The derivation of the analysis framework was based on the Hadamard transform.

The analysis framework was then applied in an optimization problem. The problem consisted of allocating a specific number (β) of index bits. Two different figures of merit were used, viz. THD and SINAD. In both cases, the optimization problem was the following: which β bits, selected from all bits of the present sample and K previous samples, should be used in order to maximize the performance of the corrected ADC in terms of THD or SINAD? Both problems could be explicitly posed using the previously derived analysis framework.

Experimental ADC data was used to illustrate the optimization problems. The results revealed two important facts. First, the optimal bit allocation depends on which figure of merit – THD or SINAD – is applied. Second, the number of bits to allocate, β can be significantly reduced without crippling the correction scheme.

Finally, an alternative approach was proposed for correcting ADCs when the input signal is assumed to be narrow-banded. Instead of using a multidimensional table indexed with present and delayed samples, the *frequency selective* correction method utilized a frequency estimator. The correction table used was two-dimensional with the present sample indexing one dimension and an instantaneous frequency estimate indexing the other dimension. Thus, the correction scheme selected different static correction tables depending on which frequency the sampled signal was, or more precise, which frequency *region* it was estimated to reside in. The results showed that a frequency selective table was superior to a static table, especially in the highest frequencies.

14.2 Future Work

The art of analog-to-digital conversion is constantly evolving. Irons said [Iro] that 'if you start measuring a device for certain performance and specify the procedures and errors, designers will "design out" the errors in subsequent designs.' The implication of this is twofold. First, some of the theories and methods for ADC characterization and correction presented here and in the contemporary literature might become obsolete with future ADC designs. Second, we do not know what methods will be needed for future ADCs. The methods presented in this thesis are generic in the sense that they are applicable to any converter architecture. It is, however, likely that an application where post-correction can be considered today, will not need any kind of correction in the future, thanks to the development of new and improved converters. Meanwhile, new frontiers are opened with higher frequencies and bandwidths, and these are the areas where post-correction is likely to find its place.

Although solving a few problems, the work presented in this thesis points out a number of new problems and tasks to confront. A few of them are posed here.

Account for Aperture Uncertainty

The experimental results presented in Section 8.7 indicated that knowing the variance of the thermal noise is not quite sufficient for accurately predicting the SINAD after correction. As was pointed out in the subsequent discussion, it is likely that aperture uncertainty, or jitter, is responsible for the increase in appreciated noise variance. It should therefore be pleasing if the jitter could be explicitly included into the theory for predicting the performance – in particular into (8.39) and (8.40).

Theoretical Limits for Dynamic Errors and Correction

The derivations in Chapter 8 were all based on an assumption that the quantizer (or ADC) to correct had a purely static behavior, in the sense that the same output signal was produced for a given input regardless of signal dynamics, such as frequency and sample history. This is obviously not true in general for a practical ADC. An extension of the theories is necessary.

One such extension is to parameterize the dynamic dependence of the DNL. A suggestion could be to model the DNL with a multidimensional stochastic variable, in analogy with the multidimensional LUT in Part III. The benefits of this would then be that the prediction of the performance after correction would take dynamics into account. In particular, the effects of not taking care of dynamically dependent errors – for instance by assigning too few dimensions in the correction scheme – could be predicted. The weakness of this extension is of course the difficulty to asses the multidimensional statistics of the DNL for a practical ADC.

Optimality for the Order-Statistics Estimator.

In Chapter 7 it was indicated that the estimator $\hat{g}_{\text{os}}$ based on order statistics could be the maximum-likelihood estimator in the case of a uniform-and-Gaussian mixture noise. This was, however, not analytically shown. It would of course be satisfying – at least from a scientific point of view – to be able to prove this conjecture.

Roadmap to an Order Statistics Calibration System

In Chapter 7 it was pointed out that the theories provided for an ADC transfer function characterization based on order statistics estimators are not ready for practical use yet. Section 7.7 points out the key problems that must be solved in order to make such a characterization scheme feasible—these problems are of course points of interest for future research.

Calibration Signals

The search for sufficiently exciting signals for calibration and testing, that are feasible to use, must go on. With more applications that demand high bandwidth, meaning that the signal of interest occupies a significant part of a Nyquist band, it becomes increasingly insufficient to use narrow-band calibration signals, such as sinewaves. Sinewaves do, on the other hand, have the advantage of being a well-defined waveform. This enables us to use signal processing techniques (filtering or waveform fitting) to obtain a reference signal from the recorded output of the ADC—the calibration can be considered as "blind." The feasibility of this approach has been demonstrated several times, and also in this thesis. Sinewaves are also relatively easy to produce with high fidelity.

Migrating to wide-band calibration signals is not a trivial task. Preserving the ability to use a blind calibration scheme requires two things: The calibration signal must have some kind of structure (such as the sinewave has), and an estimator that takes advantage of this structure must be devisable. In Section 7.7 the possibility of using multisines as calibration signals was alluded to. This signal possesses sufficient structure, and estimators that benefit from this structure are readily available. It is, however, not trivial to generate a sufficiently pure multisine in practice – intermodulation products created in the signal generator seems to be the main hurdle at the moment. Other possible calibration signals must therefore also be investigated.

Appendix A

ADC Test Beds

This appendix describes the test beds used to acquire the experimental ADC data used in this thesis. Two test beds have been used, and they are described in brief.

A.1 KTH Test Bed

This first test bed, assembled at KTH, Stockholm, is intended for measuring a 10-bit, 20 MSPS converter. The test setup follows closely the recommendations in IEEE standard 1241 [Std1241, §4.1.1.1], and the block schematics are provided in Figure A.1.

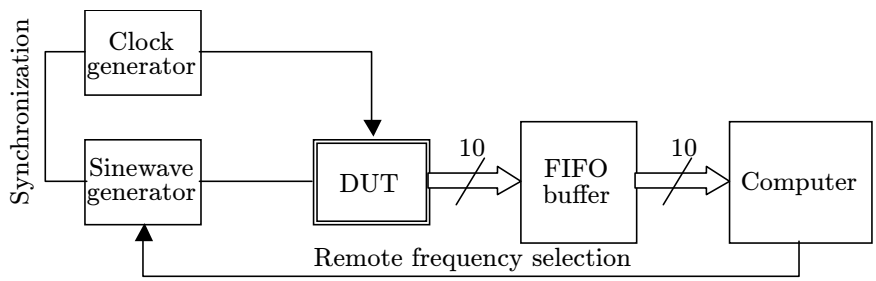


Figure A.1: Test bed block schematics. The frequency generators are synchronized to enable coherent sampling without drifting. The first-in first-out (FIFO) buffer is used as a temporary storage between the high-speed ADC and the computer.

The different parts of the test bed are briefly described in the sequel.

DUT (Device under test) The ADC under test is an Analog Devices AD876 evaluation board. The AD876 is a 10-bit pipelined flash converter operating at a maximum sampling rate of 20 MSPS. The evaluation board provides easy access to input, clock and output signals. See [AD876] for details.

Signal generators The input sinewave signal and the clock signal are generated by two Marconi Instruments MI 2024 tone generators. The input signal is directly connected to the evaluation board (appropriate DC bias is added on the evaluation board). The clock signal, on the other hand, must be a 0–5 V square wave, so the tone generator is connected to a pulse shaping circuit (custom built), converting the sinewave to a square wave of the same frequency. The two generators are synchronized so that they both utilize one and the same reference oscillator. This is to ensure that the clock and the test signal are phase-locked.

FIFO-buffer Since the computer, which is used to finally store the sampled data, is unable to read the data coming from the ADC at the sampling rate, an intermediate memory is used to store one batch of samples, viz. 16 384 samples. The samples are stored in a first-in first-out (FIFO) type memory. When the memory is full, the sampling is stopped and the data is retrieved to the computer at a lower rate.

Computer A standard PC equipped with a digital input/output board is used to read the sample sequences from the FIFO board and store them to a file. The computer is also used to control the test bed hardware.

A.2 HiG Test Bed

The experiments reported in Chapter 8 have been conducted using data from a 12-bit, 210 MSPS converter. The test bed used is at the University of Gävle (HiG), and is described in [BH05]. The main parts of the test bed are:

Vector Signal Generator The signals, both clock and test signals, are generated from a Rohde & Schwarz SMU200A.

Amplifier A custom-designed ultra low noise amplifier for driving the input signal to the ADC under test. The amplifier has a gain of 14 dB in the range 20–300 MHz.

Filters Several different SAW filters can be applied for signal conditioning.

Frame Grabber The custom-built frame grabber stores the samples coming from the ADC under test in real time, at sample rates up to 300 MSPS. The word length is 16 bits and 2000 kSample can be stored. The data is transferred to a standard PC via LAN.

The data used in Chapter 8 was acquired from an Analog Devices AD9430. See Appendix B.3 and [AD9430] for details.

Appendix B

Experimental ADC Data

All the data used to illustrate and evaluate the methods proposed in this work has been acquired using the test beds described in Appendix A. The data was in most cases collected using *coherent sampling*, as described in [Std1241, §4.1.4.5 and §4.1.5.1] and also in Section 1.4. In order to facilitate subsequent Fourier analysis of the data, the test signal frequency should also line up with a discrete Fourier transform bin frequency. Thus, the input signal frequency f_0 must satisfy two criteria:

1. The frequency must coincide exactly with a DFT bin, i.e.,
$$f_0 = \frac{m}{N} f_s, \quad m = 0, 1, \ldots, \frac{N}{2} - 1. \tag{B.1}$$

2. The frequency must satisfy
$$f_0 = \frac{J}{N} f_s, \tag{B.2}$$
where the integer J and the record length N are *relatively prime*, i.e., have no common factors. Thus, $\mathrm{GCF}(m, N) = 1$.

We immediately see from the two above equations that this implies that m and N must be relatively prime. When N is a power of 2, which is the case of the data used here, any odd integer m will satisfy the above condition 2. Additionally, the standard stipulates that the number of samples N in a batch record should be at least $\pi 2^b$, where b is the number of bits in the ADC under test, in order to ensure a sample of every code bin.

B.1 Data Set A: AD876 in First Nyquist Band

The first set of ADC data was acquired from an Analog Devices AD876 using the test bed in Section A.1. The converter performance, according to the manufacturer supplied data sheet [AD876], is given in Table B.1.

Table B.1: Performance parameters for Analog Devices AD876.

Resolution	10 bits
Maximum sampling frequency	20 MHz
Full power bandwidth	150 MHz
SINAD @ 3.58 MHz	56 dB
ENOB @ 3.58 MHz	9.0 bits
THD @ 3.58 MHz	–62 dBc
SFDR @ 3.58 MHz	65 dB

Table B.2: Characteristics of data set A.

Sampling frequency	$f_s = 19\ 972\ 096$ Hz
Input frequency	$1\ 000\ 799 \leq f_0 \leq 9\ 984\ 829$ MHz
Record size	$N = 16\ 384$ samples
Sinewave amplitude	-0.45 dBFS, or approx. 95% of full-scale

Table B.3: Characteristics of data set B.

Sampling frequency	$f_s = 19\ 972\ 096$ Hz
Input frequency	$39\ 945\ 411 \leq f_0 \leq 49\ 929\ 021$ MHz
Record size	$N = 16\ 384$ samples
Sinewave amplitude	-0.45 dBFS, or approx. 95% of full-scale

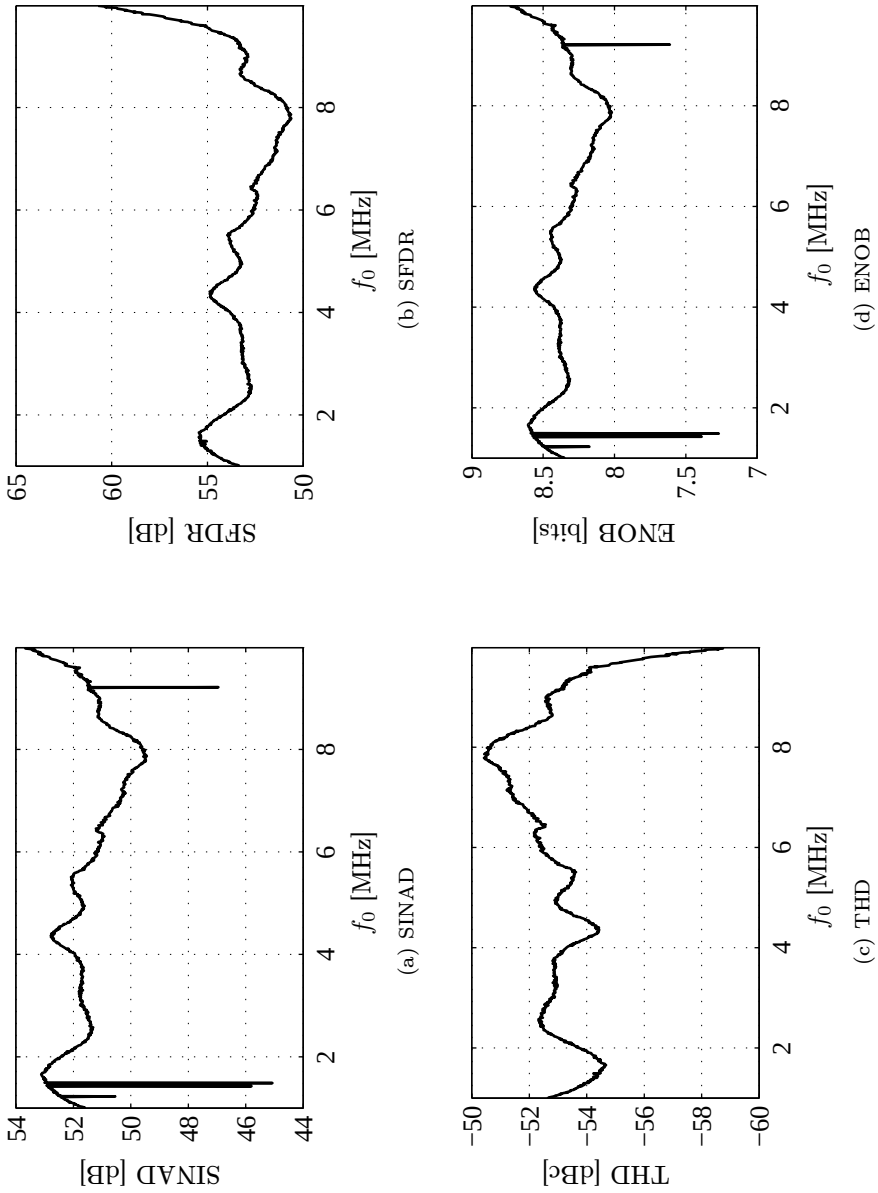
887 sequences, of 16 384 samples each, with distinct frequencies were recorded. All frequencies were selected to fulfill the requirements for coherent sampling as above. The data has the characteristics listed in Table B.2.

The SINAD, SFDR, THD and ENOB of the experimental data vary over the frequency range according to Figure B.1

B.2 Data Set B: AD876 in Fifth Nyquist Band

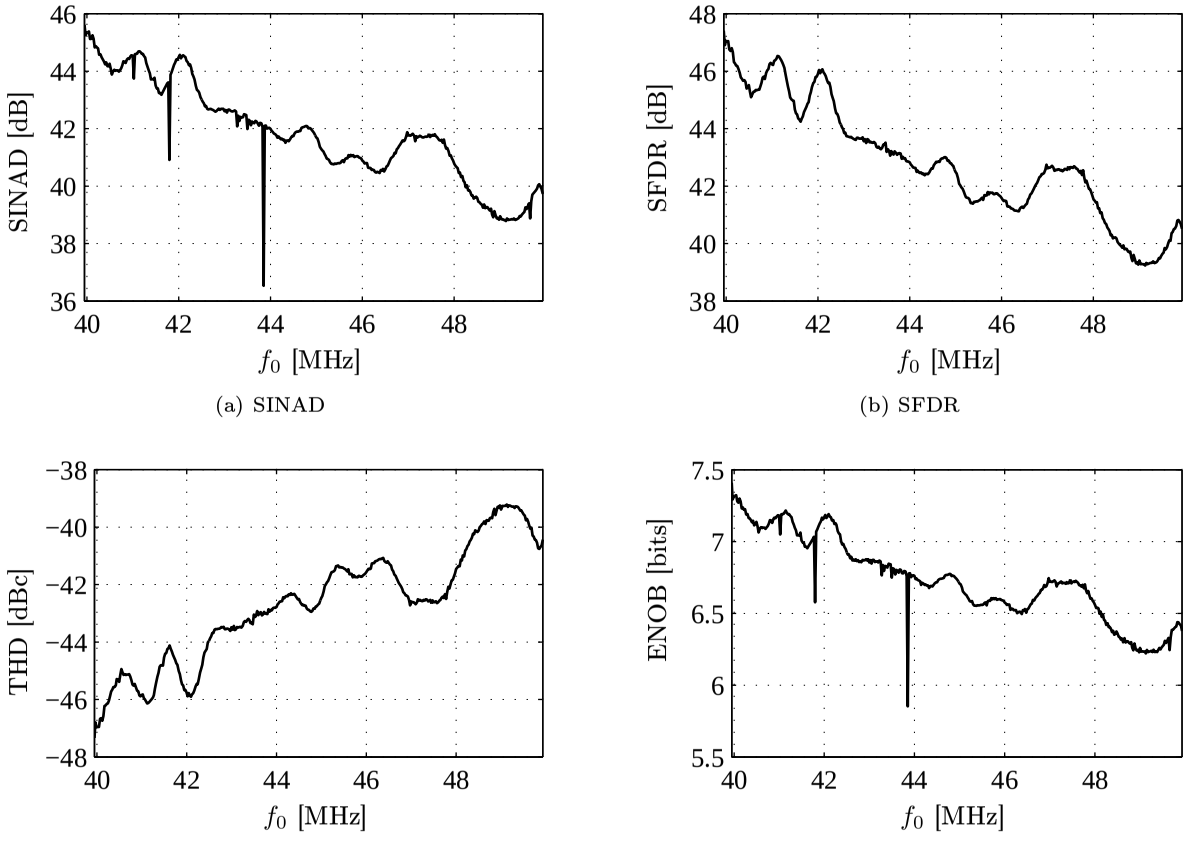
The second set of ADC data was acquired from the same AD876 and test bed as for Set A. 400 sequences, of 16 384 samples each, with distinct frequencies were recorded. All frequencies were selected to fulfill the requirements for coherent sampling as above. The data has the characteristics listed in Table B.3.

The SINAD, SFDR, THD and ENOB of the experimental data vary over the frequency range according to Figure B.2



(a) SINAD (b) SFDR (c) THD (d) ENOB

Figure B.1: The characteristics of the experimental data in Set A, without post-correction.



(a) SINAD

(b) SFDR

(c) THD

(d) ENOB

Figure B.2: The characteristics of the experimental data in Set B, without post-correction. The data is under-sampled, i.e., $f_0 > f_s/2$.

B.3 Data from AD9430

A series of measurements from an Analog Devices AD9430 was used to test the theories of Chapter 8. The basic characteristics of the ADC, as stated in the manufacturer data sheet [AD9430], is given in Table B.4. The data was acquired using the test bed described in Appendix A.2. In order to try the theory of Chapter 8, two parameters of the ADC is needed, viz. the random noise variance σ_n^2 and the variance of the DNL, σ_D^2. These were acquired using standardized tests, and are described in the following sections.

The converter was clocked using a clock frequency of 209 993 728 Hz during all measurements.

Random Noise Estimate from Triangular Wave Measurement

The random noise of the ADC was estimated as described in [Std1241, §4.5.3.2], using small amplitude triangular waves. Two records of small scale triangular waves was taken. The frequency of the triangle wave was approximately 20 kHz, and the amplitude was adjusted so that the signal covered about 10 LSBs. Each record consisted of 2 048 000 samples. The two records were lined up so that they had the same initial phase, and the random input noise σ_n^2 was estimated as

$$\mathrm{MSE} = \frac{1}{2N}\sum_{k=1}^{N}(y_a(k) - y_b(k))^2 \tag{B.3}$$

$$\widehat{\sigma_n^2} = \left(\left(\frac{\mathrm{MSE}}{2}\right)^{-2} + (0.886\,\mathrm{MSE})^{-4}\right)^{-\frac{1}{2}}, \tag{B.4}$$

where y_a and y_b are the two recorded sequences after aligning them, and N is the number of samples in each sequence. (N will probably be less than 2 048 000, since the start of one sequence and the end of the other will be truncated during the aligning operation.)

The resulting estimate was $\widehat{\sigma_n^2} = 0.5374$ LSB^2.

Table B.4: Performance parameters for Analog Devices AD9430.

Resolution	12 bits
Maximum sampling frequency	210 MHz
Full power bandwidth	700 MHz
SINAD @ 60 MHz	64.5 dB
ENOB @ 60 MHz	10.5 bits
SFDR @ 60 MHz	76 dB

DNL Estimate from Sinewave Histogram Test

The DNL of the converter was estimated using the sinewave histogram test (SHT) in [Std1241, §4.1.6.3]. The following parameters were used:

Desired tolerance	$B = 0.05$ LSB
Overdrive	$V_{\text{od}} = 4$ LSB
Random relative phase error	$\sigma_\phi = 0$

These parameters inserted in equation (69) of [Std1241] resulted in that at least 3 records of 2 048 000 samples each should be used to estimate the DNL.

Four 2 048 000-sample records of a sinewave were recorded with a signal frequency of 60 124 547 Hz and sample frequency as above. The resulting estimated DNL is shown in Figure B.3. The variance of the DNL was estimated to $\widehat{\sigma_D^2} = 0.004206$ LSB^2.

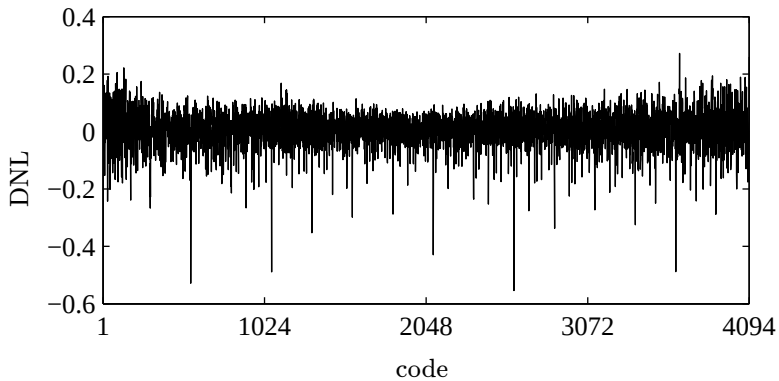


Figure B.3: The DNL estimated using sinewave histogram test.

Sinewave Measurements

Two sets of sinewave measurements were recorded for the purpose of look-up table calibration and evaluation. The sinewave had a frequency of 60 124 547 Hz and sample frequency as above, so that the conditions for coherent sampling were fulfilled for a record length of 65 536 samples. The amplitude was set to –0.5 dBFS. Each set of measurements consists of 31 sequences of 65 536 samples each. (In fact, the 31 sequences were recorded in one long measurement of 2 031 616 samples, which was split up into 31 sequences. Two such long sequences were recorded independently.) The power spectral density of the data is shown in Figure B.4.

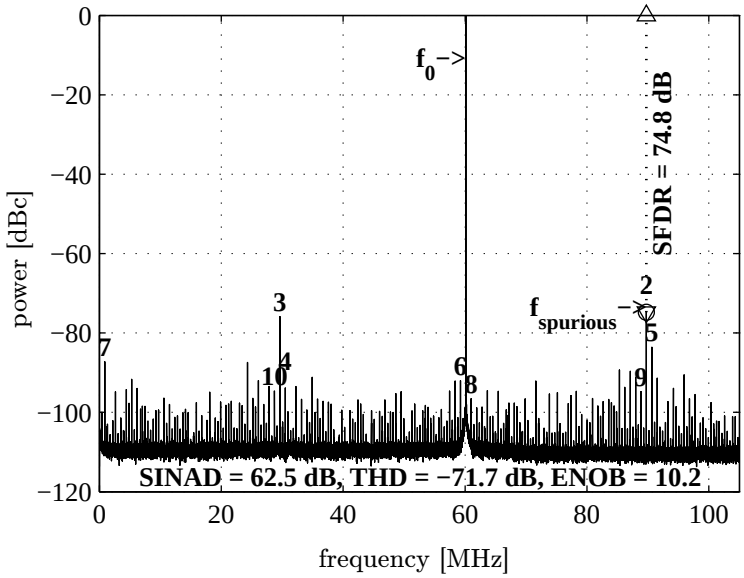


Figure B.4: Power spectral density for the AD9430.

Random Noise Estimate from Sinewave Measurement

Since the sinewave measurements above are made in one sequence of 2 031 616 samples, but with the sampling and signal frequencies set to guarantee only 65 536 unique phases, the input signal will repeat itself exactly 31 times, disregarding from random noise. Therefore, when comparing two of the 31 sequences, the only difference should be the non-deterministic errors and noise effects. Thus, we can use the formulas of (B.3) and let y_a and y_b be two sequences of 65 536 samples each, originating from the same long measurement. Using this technique, the random noise was estimated to $\widehat{\sigma_n^2} = 0.8092$, which is higher than what was obtained using the triangular method above. The reason for this is probably that random jitter will affect the high-frequency large-scale sinewave much more than it will affect the triangular wave with its low slew-rate (cf. (1.3) on page 9).

Note that this should not be interpreted as the triangular method failing in estimating the correct noise level. The rationale for using a small-scale low-frequency input signal in that test is that it should capture the random input noise (thermal noise), and not the aperture jitter. Specific methods for estimating the jitter are specified in [Std1241].

Bibliography

[AAGS02] F. Adamo, F. Attivissimo, N. Giaquinto, and M. Savino. FFT test of A/D converters to determine the integral nonlinearity. *IEEE Transactions on Instrumentation and Measurement*, 51(5):1050–1054, October 2002.

[AD876] Analog Devices, www.analog.com. *AD876: 10-Bit, 20 MSPS 160 mW CMOS A/D Converter Data Sheet*. December 1997. Rev. B. Available from: `http:www.analog.com`.

[AD9430] Analog Devices, www.analog.com. *AD9430: 12-Bit, 170/210 MSPS 3.3 V A/D Converter Data Sheet*. August 2005. Rev. D. Available from: `http:www.analog.com`.

[AD9446] Analog Devices, www.analog.com. *AD9446: 16-Bit, 80/100 MSPS, A/D Converter Data Sheet*. August 2005. Rev. PrF. Available from: `http:www.analog.com`.

[AH98a] O. Aumala and J. Holub. Dithering design for measurement of slowly varying signals. *Measurement*, 23(4):271–276, June 1998.

[AH98b] O. Aumala and J. Holub. Practical aspects of dithered quantizers. In J. Holub and R. Smid, editors, *Dithering in Measurement: Theory and Applications, Proc. 1st International On-line Workshop*, Prague, March 1998. CTU FEE Dept. of Measurement, Prague, Czech Republic, TUT Measurement and Information Technology, Tampere, Finland. Available from: `http://measure.feld.cvut.cz/dithering98`.

[AH03] T. Andersson and P. Händel. Multiple-tone estimation by IEEE standard 1057 and the expectation-maximization algorithm. In *IEEE Conference on Instrumentation and Measurement*, Vail, CO, May 2003.

[AH04] T. Andersson and P. Händel. Toward a standardized multi-sinewave fit algorithm. In *9th European Workshop on ADC Modelling and Testing*, volume 1, pages 337–342, Athens, Greece, September 2004.

[AK97] J. Astola and P. Kuosmanen. *Fundamentals of Nonlinear Digital Filtering*. CRC Press LLC, Boca Raton, FL, 1997.

[AN410] Overcoming converter nonlinearities with dither. Application Note AN–410, Analog Devices, Norwood, MA. Brad Brannon. 1995. Available from: `http:www.analog.com`.

[And05] T. Andersson. *Parameter Estimation and Waveform Fitting for Narrowband Signals*. PhD thesis, Royal Institute of Technology (KTH), June 2005. SB-0540.

[ASH00] T. Andersson, M. Skoglund, and P. Händel. Frequency estimation by 1-bit quantization and table look-up processing. In *Proceedings European Signal Processing Conference*, pages 1807–1810, Tampere, Finland, September 2000.

[BDR05] E. Balestrieri, P. Daponte, and S. Rapuano. A state of the art on ADC error compensation methods. *IEEE Transactions on Instrumentation and Measurement*, 54(4):1388–1394, August 2005.

[Ber04] D. I. Bergman. Dynamic error correction of a digitizer for time domain metrology. *IEEE Transactions on Instrumentation and Measurement*, 53(5):1384–1390, October 2004.

[Bey98] J. Beyerer. Is it useful to know a nuisance parameter? *Signal Processing*, 68(1):107–111, July 1998.

[BH04] N. Björsell and P. Händel. Benefits with truncated Gaussian noise in ADC histogram tests. In *Proceedings of the 9th Workshop on ADC Modelling and Testing*, volume 2, pages 787–792, Athens, Greece, September 2004. IMEKO.

[BH05] N. Björsell and P. Händel. High dynamic range test-bed for characterization of analog-to-digital converters. In *10th Workshop on ADC modelling and testing, Gdynia and Jurata, Poland.*, September 2005.

[BNPP00] M. Bertocco, C. Narduzzi, P. Paglierani, and D. Petri. A noise model for digitized data. *IEEE Transactions on Instrumentation and Measurement*, 49(1):83–86, February 2000.

[Car97] P. Carbone. Quantitative criteria for the design of dither-based quantizing systems. *IEEE Transactions on Instrumentation and Measurement*, 46(3):656–659, June 1997.

[CNP02] P. Carbone, E. Nunzi, and D. Petri. Statistical efficiency of the ADC sinewave histogram test. *IEEE Transactions on Instrumentation and Measurement*, 51(4):849–852, 2002.

[CP00] P. Carbone and D. Petri. Performance of stochastic and deterministic dithered quantizers. *IEEE Transactions on Instrumentation and Measurement*, 49(2):337–340, April 2000.

[DC90] A. C. Dent and C. F. N. Cowan. Linearization of analog-to-digital converters. *IEEE Transactions on Circuits and Systems*, 37(6):729–737, June 1990.

[DHH+02] P. Daponte, R. Holcer, L. Horniak, L. Michaeli, and S. Rapuano. Using an interpolation method for noise shaping in A/D converters. In *Proceedings of the 7th European Workshop on ADC Modelling and Testing*, pages 147–150, Prague, Czech Republic, June 2002. IMEKO.

[DLP86] I. De Lotto and G. E. Paglia. Dithering improves A/D converter linearity. *IEEE Transactions on Instrumentation and Measurement*, IM-35(2):170–177, June 1986.

[DVBS92] J. P. Deyst, J. J. Vytal, P. R. Blasche, and W. M. Siebert. Wideband distortion compensation for bipolar flash analog-to-digital converters. In *Proceedings of the 9th IEEE Instrumentation and Measurement Technology Conference*, pages 290–294, Metro New York, NJ, USA, 1992.

[DVLR05a] L. De Vito, H. Lundin, and S. Rapuano. Bayesian calibration of a look-up table for ADC error correction. *IEEE Transactions on Instrumentation and Measurements*, June 2005. Submitted.

[DVLR05b] L. De Vito, H. Lundin, and S. Rapuano. A Bayesian filtering-approach for calibrating a look-up table used for ADC error correction. In *Proceedings IEEE Instrumentation And Measurement Technology Conference*, volume 1, pages 293–297, Ottawa, Canada, May 2005.

[Elb01] J. Elbornsson. Equalization of distortion in A/D converters. Technical Report Licentiate Thesis no. 883, Department of Electrical Engineering, Linköping University, SE-581 83 Linköping, Sweden, April 2001.

[EM04] U. Eduri and F. Maloberti. Online calibration of a Nyquist-rate analog-to-digital converter using output code-density histograms. *IEEE Transactions on Circuits and Systems—Part I: Fundamental Theory and Applications*, 51(1):15–24, January 2004.

[GDG04] Z. Guo, M. D'Amore, and A. Gutierrez. A 2-bit 20 Gsps InP HBT A/D converter for optical communications. In *IEEE Compound Semiconductor Integrated Circuit Symposium*, pages 93–96. IEEE, October 2004.

[GG92] A. Gersho and R. M. Gray. *Vector Quantization and Signal Compression.* Kluwer Academic Publishers, Boston, MA, 1992.

[GN98] R. M. Gray and D. L. Neuhoff. Quantization. *IEEE Transactions on Information Theory*, 44(6):2325–2383, October 1998.

[Gra90] R. M. Gray. *Source Coding Theory.* Kluwer Academic Publishers, Boston, MA, 1990.

[GS94] X. M. Gao and S. Sun. Modeling the harmonic distortion of analog-to-digital converter using Volterra series. In *Proceedings IEEE Instrumentation And Measurement Technology Conference, IMTC/1994*, volume 2, pages 911–913, May 1994.

[GST96a] N. Giaquinto, M. Savino, and A. Trotta. Detection, digital correction and global effect of A/D converters nonlinearities. In P. Daponte and L. Michaeli, editors, *International Workshop on ADC Modelling*, pages 122–127, Slovak Republic, May 1996. IMEKO.

[GST96b] N. Giaquinto, M. Savino, and A. Trotta. Testing and optimizing ADC performance: A probabilistic approach. *IEEE Transactions on Instrumentation and Measurement*, 45(2):621–626, April 1996.

[GVL96] G. H. Golub and C. F. Van Loan. *Matrix Computations.* Johns Hopkins University Press, Baltimore, 3rd edition, 1996.

[Hän00] P. Händel. Properties of the IEEE-STD-1057 four parameter sine wave fit algorithm. *IEEE Transactions on Instrumentation and Measurement*, 49:1189–1193, December 2000.

[HICP94] D. M. Hummels, F. H. Irons, R. Cook, and I. Papantonopoulos. Characterization of ADCs using a non-iterative procedure. In *Proceedings IEEE International Symposium on Circuits and Systems*, volume 2, pages 5–8, London, May 1994.

[HJ91] R. A. Horn and C. R. Johnson. *Topics in Matrix Analysis.* Cambridge University Press, Cambridge, 1991.

[HSP00] P. Händel, M. Skoglund, and M. Pettersson. A calibration scheme for imperfect quantizers. *IEEE Transactions on Instrumentation and Measurement*, 49:1063–1068, October 2000.

[Hum02] D. Hummels. Performance improvement of all-digital wide-bandwidth receivers by linearization of ADCs and DACs. *Measurement*, 31(1):35–45, January 2002.

[IC86] F. H. Irons and A. I. Chaiken. Analog-to-digital converter compensation precision effects. In *29th Midwest Symposium on Symposium on Circuits and Systems*, pages 849–852, Lincoln, Nebraska, August 1986.

[IHK91] F. H. Irons, D. M. Hummels, and S. P. Kennedy. Improved compensation for analog-to-digital converters. *IEEE Transactions on Circuits and Systems*, 38(8):958–961, August 1991.

[Iro] F. H. Irons. Memoirs: Career high-lights [online, cited 4 November 2005]. Available from: `http://www.eece.maine.edu/~irons/memoirs.html`.

[Iro86] F. H. Irons. Dynamic characterization and compensation of analog to digital converters. In *Proceedings of IEEE International Symposium on Circuits and Systems*, volume 3, pages 1273–1277, 1986.

[Jes01] P. G. A. Jespers. *Integrated Converters.* Oxford University Press, Oxford, 2001.

[Jon00] B. E. Jonsson. *Switched-Current Signal Processing and A/D Conversion Circuits.* Kluwer Academic, Boston, MA, USA, 2000.

[Kay93] S. M. Kay. *Fundamentals of Statistical Signal Processing: Estimation Theory.* Prentice-Hall, Englewood Cliffs, NJ, 1993.

[KW69] A. J. Kurtenbach and P. A. Wintz. Quantizing for noisy channels. *IEEE Transactions on Communication Technology*, COM-17(2):291–302, April 1969.

[LASH02] H. Lundin, T. Andersson, M. Skoglund, and P. Händel. Analog-to-digital converter error correction using frequency selective tables. In *RadioVetenskap och Kommunikation (RVK)*, pages 487–490, Stockholm, Sweden, June 2002.

[LHS04] H. Lundin, P. Händel, and M. Skoglund. Adaptively calibrating analog-to-digital conversion with correction table indexing. U.S. Patent 6,690,311 B2, February 2004.

[LHS06] H. Lundin, P. Händel, and M. Skoglund. Accurate prediction of analog-to-digital converter performance after post-correction. In *Proceedings IMEKO XVIII World Congress*, Rio de Janeiro, Brazil, September 2006. IMEKO. Submitted.

[LHSP02] H. Lundin, P. Händel, M. Skoglund, and M. Pettersson. Adaptively calibrating analog-to-digital conversion. U.S. Patent 6,445,317, September 2002.

[Lju99] L. Ljung. *System Identification: Theory for the User.* Prentice Hall, Upper Saddle River, NJ, 2nd edition, 1999.

[Llo52] E. H. Lloyd. Least-squares estimation of location and scale parameters using order statistics. *Biometrika*, 39:88–95, 1952.

[Llo82] S. P. Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, IT-28(2):129–137, March 1982.

[LSH01] H. Lundin, M. Skoglund, and P. Händel. On external calibration of analog-to-digital converters. In *IEEE Workshop on Statistical Signal Processing*, pages 377–380, Singapore, August 2001.

[LSH02] H. Lundin, M. Skoglund, and P. Händel. A framework for external dynamic compensation of AD converters. In *Proceedings 7th European Workshop on ADC Modelling and Testing*, pages 135–138, Prague, Czech Republic, June 2002.

[LSH03a] H. Lundin, M. Skoglund, and P. Händel. Minimal total harmonic distortion post-correction of ADCs. In *International Workshop on ADC Modelling and Testing*, pages 113–116, Perugia, Italy, September 2003.

[LSH03b] H. Lundin, M. Skoglund, and P. Händel. Optimizing dynamic post-correction of AD converters. In *Proceedings IEEE Instrumentation And Measurement Technology Conference*, pages 1206–1211, Vail, Colorado, USA, May 2003.

[LSH04] H. Lundin, M. Skoglund, and P. Händel. A criterion for optimizing bit-reduced post-correction of AD converters. *IEEE Transactions on Instrumentation and Measurements*, 53(4):1159–1166, August 2004.

[LSH05a] H. Lundin, M. Skoglund, and P. Händel. ADC post-correction using limited resolution correction values. In *Proceedings IMEKO 10th Workshop on ADC Modelling and Testing*, volume 2, pages 567–572, Gdynia/Jurata, Poland, September 2005.

[LSH05b] H. Lundin, M. Skoglund, and P. Händel. On the estimation of quantizer reconstruction levels. In *Proceedings IEEE Instrumentation And Measurement Technology Conference*, volume 1, pages 144–149, Ottawa, Canada, May 2005.

[LSH05c] H. Lundin, M. Skoglund, and P. Händel. On the estimation of quantizer reconstruction levels. *IEEE Transactions on Instrumentation and Measurements*, July 2005. Submitted.

[LSH05d] H. Lundin, M. Skoglund, and P. Händel. Optimal index-bit allocation for dynamic post-correction of analog-to-digital converters. *IEEE Transactions on Signal Processing*, 53(2):660–671, February 2005.

[LSH05e] H. Lundin, M. Skoglund, and P. Händel. Signal processing results on dynamic ADC post-correction. In *GigaHertz, Proceedings of*, Uppsala, Sweden, November 2005.

[LSZ+04] H. Lundin, P. Svedman, X. Zhang, M. Skoglund, P. Händel, and P. Zetterberg. ADC imperfections in multiple antenna wireless systems—an experimental study. In *9th European Workshop on ADC*

Modelling and Testing, pages 808–813, Athens, Greece, September 2004.

[Lun00] H. Lundin. Dynamic compensation of analogue-to-digital converters. Master's thesis, Royal Institute of Technology (KTH), Dept. of Signals, Sensors & Systems, Signal Processing, December 2000. IR-SB-EX-0023. Available from: `http://www.ee.kth.se`.

[Lun05] H. Lundin. An introduction to ADC error correction. Technical report, Royal Institute of Technology (KTH), May 2005. Course literature for the 5th Summer School on Data Acquisition Systems, Italy, 2005. Available from: `http://www.ee.kth.se`.

[Lüt96] H. Lütkepohl. *Handbook of Matrices*. John Wiley & Sons, 1996.

[LWV92] S. P. Lipshitz, R. A. Wannamaker, and J. Vanderkooy. Quantization and dither: A theoretical survey. *J. Audio Eng. Soc.*, 40(5):355–375, May 1992.

[MCP03] A. Moschitta, P. Carbone, and D. Petri. Statistical performance of Gaussian ADC histogram test. In *Proceedings of the 8th International Workshop on ADC Modelling and Testing*, pages 213–217, Perugia, Italy, September 2003. IMEKO.

[Mou89] D. Moulin. Real-time equalization of A/D converter nonlinearities. In *Proceedings of the IEEE International Symposium on Circuits and Systems*, volume 1, pages 262–267, Portland, OR, USA, 1989. IEEE.

[MŠ02] P. Mikulik and J. Šaliga. Volterra filtering for integrating ADC error correction, based on an a priori error model. *IEEE Transactions on Instrumentation and Measurement*, 51(4):870–875, August 2002.

[N+04] H. Nosaka et al. A 24-Gsps 3-bit Nyquist ADC using InP HBTs for electronic dispersion compensation. In *IEEE MTT-S International Microwave Symposium Digest*, volume 1, pages 101–104. IEEE, June 2004.

[NST97] S. R. Norsworthy, R. Schreier, and G. C. Temes, editors. *Delta-Sigma Data Converters*. IEEE Press, New Jersey, USA, 1997.

[NW88] G. L. Nemhauser and L. A. Wolsey. *Integer and Combinatorial Optimization*. Wiley-Interscience series in discrete mathematics and optimization. Wiley, New York, 1988.

[OPS48] B. M. Oliver, J. Pierce, and C. E. Shannon. The philosophy of PCM. *Proc. IRE*, 36:1324–1331, 1948.

[OWN97] A. V. Oppenheim, A. S. Willsky, and S. H. Nawab. *Signals and Systems*. Prentice-Hall, 2nd edition, 1997.

[PM96] J. G. Proakis and D. G. Manolakis. *Digital Signal Processing — Principles, Algorithms, and Applications*. Prentice-Hall, 3rd edition, 1996.

[RI87] T. A. Rebold and F. H. Irons. A phase-plane approach to the compensation of high-speed analog-to-digital converters. In *Proceedings of IEEE International Symposium on Circuits and Systems*, volume 2, pages 455–458, 1987.

[Rob62] L. G. Roberts. Picture coding using pseudo-random noise. *IRE Trans. Information Theory*, IT-8:145–154, February 1962.

[Sch64] L. Schuchman. Dither signals and their effect on quantization noise. *IEEE Transactions on Communication Technology*, 12(4):162–165, December 1964.

[Sch76] M. Schetzen. Theory of pth-order inverses of nonlinear systems. *IEEE Transactions on Circuits and Systems*, CAS-23(5):285–291, May 1976.

[SD98] J. Schoukens and T. Dobrowiecki. Design of broadband excitation signals with a user imposed power spectrum and amplitude distribution. In *Instrumentation and Measurement Technology Conference. IMTC/98.*, volume 2, pages 1002–1005, St. Paul, USA, May 1998. IEEE.

[She98] W. F. Sheppard. On the calculation of the most probable values of frequency constants for data arranged according to equidistant divisions of scale. *Proc. London Math. Soc.*, 24(2):353–380, 1898.

[SNM99] A. K. Salkintzis, H. Nie, and P. T. Mathiopoulos. ADC and DSP challenges in the development of software radio base stations. *IEEE Personal Communications Magazine*, 6(4):47–55, August 1999.

[SO96] P. Stoica and B. Ottersten. The evil of superefficiency. *Signal Processing*, 55(1):133–136, November 1996.

[SØ05] R. Skartlien and L. Øyehaug. Quantization error and resolution in ensemble averaged data with noise. *IEEE Transactions on Instrumentation and Measurement*, 54(3):1303–1312, June 2005.

[SS77] A. B. Sripad and D. L. Snyder. A necessary and sufficient condition for quantization errors to be uniform and white. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-25(5):442–448, October 1977.

[Std1057] IEEE. *IEEE Standard for Digitizing Waveform recorders.* IEEE Std. 1057. 1994.

[Std1241] IEEE. *IEEE Standard for Terminology and Test Methods for Analog-to-Digital Converters.* IEEE Std. 1241. 2000.

[TL93] J. Tsimbinos and K. V. Lever. Applications of higher-order statistics to modelling, identification and cancellation of nonlinear distortion in high-speed samplers and analogue-to-digital converters using the Volterra and Wiener models. In *Proceedings of the IEEE Signal Processing Workshop on Higher-Order Statistics*, pages 379–383, June 1993.

[TL96a] J. Tsimbinos and K. V. Lever. Computational complexity of Volterra based nonlinear compensators. *Electronics Letters*, 32(9):852–854, April 1996.

[TL96b] J. Tsimbinos and K. V. Lever. Error table and Volterra compensation of A/D converter nonlinearities – a comparison. In *Proceedings of the Fourth International Symposium on Signal Processing and its Applications*, volume 2, pages 861–864, August 1996.

[TL97] J. Tsimbinos and K. V. Lever. Improved error-table compensation of A/D converters. *IEE Proceedings - Circuits, Devices and Systems*, 144(6):343–349, December 1997.

[TMBSL02] J. Tsimbinos, W. Marwood, A. Beaumont-Smith, and C. C. Lin. Results of A/D converter compensation with a VLSI chip. In *Proceedings of Information, Decision and Control, IDC 2002*, pages 289–294, February 2002.

[Tsi95] J. Tsimbinos. *Identification and Compensation of Nonlinear Distortion.* PhD thesis, School of Electronic Engineering, University of South Australia, February 1995.

[vdP94] R. J. van de Plassche. *Integrated Analog-to-Digital and Digital-to-Analog Converters.* Kluwer Academic Publishers, 1994.

[Wal99] R. H. Walden. Analog-to-digital converter servey and analysis. *IEEE Journal on Selected Areas in Communication*, 17(4):539–550, April 1999.

[Wep95] J. A. Wepman. Analog-to-digital converters and their applications in radio receivers. *IEEE Communications Magazine*, 33(5):39–45, May 1995.

[Wid56] B. Widrow. A study of rough amplitude quantization by means of Nyquist sampling theory. *IRE Trans. Circuit Theory*, 3(4):266–276, December 1956.

[Wid61] B. Widrow. Statistical analysis of amplitude-quantizer sampled-data systems. *Trans. AIEE, Part II: Applications and Industry*, 79(52):555–568, January 1961.

[WKL96] B. Widrow, I. Kollár, and M.-C. Liu. Statistical theory of quantization. *IEEE Transactions on Instrumentation and Measurement*, 45(2):353–361, April 1996.

[WKN75] C. E. Woodward, K. H. Konkle, and M. L. Naiman. A monolithic voltage-comparator array for A/D converters. *IEEE Journal of Solid-State Circuits*, SC-10(6):392–399, 1975.

[WLVW00] R. A. Wannamaker, S. P. Lipshitz, J. Vanderkooy, and J. N. Wright. A theory of nonsubtractive dither. *IEEE Transactions on Signal Processing*, 48(2):499–516, February 2000.

[YH97] R. K. Yarlagadda and J. E. Hershey. *Hadamard Matrix Analysis and Synthesis.* Kluwer Academic Publishers, Boston, MA, 1997.

[ZJL+04] P. Zetterberg, J. Jaldén, H. Lundin, D. Samuelsson, P. Svedman, and X. Zhang. Implementation of SM and rxtxIR on a DSP-based wireless MIMO test-bed. In *The European DSP Education and Research Symposium EDERS*, November 2004.

[ZSLZ05] X. Zhang, P. Svedman, H. Lundin, and P. Zetterberg. Implementation of a smart antenna multiuser algorithm on a DSP-based wireless MIMO test-bed. In *Proceedings IEEE International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*, September 2005.