Signals and Communication Technology

Paolo Carbone
Sayfe Kiaei
Fang Xu *Editors*

Design, Modeling and Testing of Data Converters

Springer

Signals and Communication Technology

For further volumes:
http://www.springer.com/series/4748

Paolo Carbone · Sayfe Kiaei · Fang Xu
Editors

Design, Modeling and Testing of Data Converters

Springer

Editors
Paolo Carbone
Department of Electronic and Information Engineering
University of Perugia
Perugia
Italy

Sayfe Kiaei
School of Electrical, Computer, and Energy Engineering
Arizona State University
Tempe, AZ
USA

Fang Xu
Polytope Solutions
Newton
USA

Additional material to this book can be downloaded from http://extras.springer.com/

ISSN 1860-4862 ISSN 1860-4870 (electronic)
ISBN 978-3-642-39654-0 ISBN 978-3-642-39655-7 (eBook)
DOI 10.1007/978-3-642-39655-7
Springer Heidelberg New York Dordrecht London

Library of Congress Control Number: 2013949180

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Contents

Reviewers List

Imran Ahmed	Kapik Integration, Canada
Markus Allén	Tampere University of Technology, Finland
Gregorio Andria	Politecnico di Bari, Italy
Soon-Jyh Chang	National Cheng Kung University, Taiwan
Jose M. de la Rosa	Centro Nacional de Microelectronica, Spain
Doug Garrity	Frescale, USA
Michael Epp	Cassidian, EADS Deutschland GmbH, Germany
Christian Eugène	Catholic University of Louvain, Belgium
Ayman Fayed	Iowa State University, USA
Stephan Henzler	Technische Universität München, Germany
Bengt Jonsson	ADMS Design AB, Sweden
Piero Malcovati	University of Pavia, Italy
Shanin Mehdizad Taleie	Qualcomm, USA
Antti Mäntyniemi	University of Oulu, Finland
Parastoo Nikaeen	Stanford University, USA
David Obrien	Teradyne Inc., USA
Rik Pintelon	Vrije Universiteit Brussel, Belgium
Pedro Ramos	Instituto Superior Técnico, Portugal
Francesco Rizzo	ABB, Switzerland
Gordon Roberts	McGill University, Canada
Pieter Rombouts	Ghent University, Belgium
Vishal Saxena	Boise StateUniversity, USA
Jesper Steensgaar	Linear Technology
Stephen Sunter	Mentor Graphics, Canada
Yannis Tsividis	Columbia University, USA
Mikko Valkama	Tampere University of Technology, Finland
Georg Vallant	Cassidian, EADS Deutschland GmbH, Germany
Wenhuan Yu	Silicon Labs, USA

Introduction

This book is the outcome of a scientific workshop on the subject of Data Converters, held in Orvieto (Italy) in 2011. After the two-day-long discussions, following presentations and talks, the organizers decided to challenge researchers and scientists, coming from many international laboratories and Universities, to write an original chapter about their main research topic. The results are published in this book, according to the same scheme that characterized the workshop: design, modeling, and testing are the three major subjects, covered by the authored contributions.

In the *Design* part, several chapters describe state-of-the-art knowledge in the area of pipeline analog-to-digital converters (ADC), of calibration of radiofrequency sigma-delta ADCs and in the domain of scalable ADCs. Two additional chapters contain a thorough overview of time-to-digital conversion and a description of an asynchronous conversion architecture capable to overcome the limitations induced by technological scaling.

In the *Modeling* part there are two chapters. One describes a converting architecture based on non-uniform quantization; the second is a detailed analysis of the methods applicable of minimizing the effects of distortion in ADCs by means of digital post-processing techniques.

In the *Testing* part, four chapters deal with the in-depth analysis of commonly used Data Converter testing procedures: amplitude and code-domain tests are described both from theoretical and practical points of view. The remaining two chapters comprise Digital-to-Analog Converter testing and statistical methods for the analysis of test data.

All chapters have been revised by knowledgeable and independent reviewers whose names are attached in this book. The editors greatly appreciate their hard work and recognize their achievements in improving presentations, readability, and usefulness of the enclosed material.

Part I
Design

Chapter 1
A Power-Optimized High-Speed and High-Resolution Pipeline ADC with a Parallel Sampling First Stage for Broadband Multi-Carrier Systems

Yu Lin, Athon Zanikopoulos, Kostas Doris, Hans Hegt and Arthur H. M. van Roermund

Abstract This chapter analyzes the statistical properties of multi-carrier signals and their impact on ADC design, reviews the general pipeline ADC architecture and conventional low power design techniques, and presents a parallel sampling technique for pipeline ADC to convert multi-carrier signals efficiently by exploiting the statistical properties of these signals. With the proposed parallel sampling technique, the input signal power of an ADC can be boosted without getting excessive clipping distortion and the ADC can have a higher resolution over the critical small amplitude region. Hence the overall signal to noise and clipping distortion ratio is improved. This technique allows reducing power dissipation and area in comparison to conventional solutions for converting multi-carrier signals. As an example, an 11b switched-capacitor pipeline ADC with the parallel sampling technique applied to its first stage is implemented in CMOS 65 nm technology. It achieves a full-scale input signal range of 2 V differentially with a 1.2 V supply voltage. Simulations show more than 5 dB improvement in signal-to-noise-and-clipping-distortion ratio (SNCDR) and around 8 dB improvement in dynamic range (DR) compared to a conventional 11b ADC for converting multi-carrier signals, simulations also show that is able to achieve a comparable SNCDR and noise power ratio (NPR) as a conventional 12b pipeline for converting multi-carrier signals with less than half the power and area.

Y. Lin (✉) · A. Zanikopoulos · H. Hegt · A. H. M. van Roermund
Mixed-signal Microelectronics Group, Eindhoven University of Technology, Eindhoven, The Netherlands
e-mail: L.Y.Lin@tue.nl

A. Zanikopoulos · K. Doris
NXP Semiconductors, Eindhoven, The Netherlands

P. Carbone et al. (eds.), *Design, Modeling and Testing of Data Converters*, Signals and Communication Technology, DOI: 10.1007/978-3-642-39655-7_1,

Keywords Analog-to-digital converters · Switched-capacitor pipeline ADC · Multi-carrier signals · Parallel sampling technique · Noise power ratio · Dynamic range

1.1 Introduction

The analog-to-digital converter (ADC) is one of the most commonly used building blocks of mixed signal circuits in both wired and wireless digital communication systems. The trend of achieving higher data throughput in these systems asks for more and more demanding specifications for ADCs in terms of sampling rate and conversion accuracy. The challenge here is to achieve a high sampling rate and high conversion accuracy at the same time with low power dissipation in the presence of component mismatch, nonlinearity, and thermal noise [1, 2]. Component mismatch and nonlinearity are not fundamental limitations, and can be therefore overcome in a power efficient way by digital calibration at the cost of additional design complexity and extra power for the calibration circuits [3–7]. Furthermore, these digital calibration circuits benefit from CMOS scaling. In contrast to mismatch and nonlinearity, thermal noise is a fundamental limitation. As the performance of general analog circuits relies on the relative contrast of the signal strength to that of thermal noise, measured by the signal-to-noise ratio (SNR), there is a strong tradeoff between power dissipation and SNR if thermal noise is the main limitation. This can be seen from (1.1), it is derived from a typical switched-capacitor circuit which is the basic building block of many ADCs. [8, 9]:

$$P \propto kT \cdot \frac{1}{\alpha^2} \cdot \frac{1}{V_{dd}} \cdot \frac{1}{\left(\frac{g_m}{I_d}\right)} \cdot SNR \cdot f_s \tag{1.1}$$

where kT is the thermal energy, V_{dd} is the supply voltage, α is the voltage efficiency factor which equals the root mean square (RMS) amplitude of the input signal over V_{dd}, g_m/I_D is related to the "gate overdrive" of the transistor that implements the transconductance, SNR is proportional to $(\alpha \cdot V_{dd})^2/(kT/C)$, and f_s is the sampling rate.

Considering constant α, V_{dd}, and g_m/I_D, it is clear from (1.1) that increasing the SNR by 6 dB requires a 4 times increase in power dissipation for a given sampling rate. Equation (1.1) also shows that power will tend to increase with the decrease of the supply voltage as the noise in the analog signals must be reduced proportionally to maintain the desired SNR. As a result, scaling of CMOS technology has negative impact on the power consumption of noise limited ADCs due to the decrease in supply voltage, although it offers a potential for faster operating speed of ADCs.

Improving the voltage efficiency (α) is then an effective way to improve the power efficiency for high-speed and high-resolution ADCs in advanced CMOS technology.

Enabling the ADC to process a large input signal range allows reducing the capacitor size that determines the thermal noise. The reduction of the capacitors brings benefits such as smaller area, lower power dissipation, higher bandwidth, and easier to drive. However, processing a large signal swing is normally constrained by the linearity of the input sampling stage, the amplifier's output stage, and further by the reference voltage. Therefore, new circuit techniques need to be introduced. In [10, 11] a range scaling technique is used in the first pipeline stage to decouple the choice of the stage's input and output signal swing, such that the signal range of the stage's input and output can be optimized separately for good power efficiency. In [12] thick oxide devices with a high supply voltage are utilized in the first pipeline stage where the signal swing and amplifier gain are limited. However, using thick oxide devices often reduces speed. In [13] both fast operation and large signal swing are achieved with a combination of thick and thin oxide devices at high and nominal supply voltages. In [14] an open-loop multiplexed buffer topology operating in feed forward-sampling and feedback-loop mode enables a large signal swing with good linearity. These techniques have demonstrated that enabling ADCs to process a large signal swing is the key to improve the power efficiency for high speed and high resolution ADCs in advanced CMOS technology. However, the maximum achievable input signal range in the previous works is still constrained by the input sampling stage which needs to process the whole signal range linearly.

Conventionally, signal statistics properties are not considered in guiding the design of ADCs. Given the complexity of today's applications, it is important to realize that opportunities do exist to optimize ADC for lower power or complexity by exploiting signal properties. For example, if the input signal has the sparse property in frequency or time domain, according to the "compressive sensing" theory, the signal can be sampled at a greatly reduced rate while still be able to be reconstructed with high fidelity, hence greatly reduce the complexity and power consumption of the ADC [15, 16]. In this chapter, we focus on the amplitude statistics property of the wideband multi-carrier signal, as many popular digital communication systems (e.g. xDSL, WLAN, WiMAX, DVB-T, MC-CDMA, LTE-Advanced) adopt multi-carrier transmission schemes and a high number of signal levels in the sub-carriers to increase data throughput. Also recent trends in cable and satellite receivers (e.g. DOCSIS 3.0, satellite digital TV receiver) ask for simultaneous reception of many channels [17]. The resulting signals at the input of the ADC have a high peak-to-average power ratio (PAR). In order to receive these signals properly, the input signal power needs to back off to avoid saturation of the ADC; hence the input signal range of the ADC is inefficiently exploited due to these special signal properties. In [17–19] the input signal power needs to back off by more than 12 dB for systems requiring a low bit error rate (BER). In that situation, the ADC needs to have two or more extra effective number of bits (ENOB) than what is normally required for receiving a narrowband signal in order to achieve a similar SNR for the multi-carrier signals.

In this chapter, we present a parallel sampling technique for ADCs to process a large input signal range with a low supply voltage. This technique relaxes the linearity constraints posed by the input sampling stage. ADCs with this technique

also allow exploiting the statistics of the multi-carrier signal to further improve the power efficiency. An 11b switched-capacitor pipeline ADC with this technique was implemented. It has a full-scale input signal range of 2 V differentially with a 1.2 V voltage supply in 65 nm CMOS technology and achieves a SNR similar to that of a conventional 12b pipeline ADC [20, 21].

The remaining part of the chapter is organized as follows. Section 1.2 analyzes the statistical properties of multi-carrier signals and their impact on ADC design. It also introduces the principle of the proposed parallel sampling technique and discusses its advantages. Section 1.3 reviews the general pipeline ADC architecture and conventional low power design techniques. Section 1.4 describes an 11b pipeline ADC with the parallel sampling technique applied to its first stage. Section 1.5 provides simulation results and, finally, Sect. 1.6 draws the conclusions of this chapter.

1.2 Signal Characterization and the Parallel Sampling Technique

1.2.1 Multi-Carrier Signal Characteristics

Systems adopting multi-carrier transmission schemes have many advantages, including high spectral efficiency and ability to cope with severe channel conditions. But one undesirable property of these schemes is that their transmitted signal in time domain is observed to have large 'peaks' when compared to their average power value. This is characterized by the peak-to-average ratio (PAR), which has a theoretical maximum value equal to $\sqrt{m} \cdot PAR_{subcarrier}$ where m is the number of subcarriers and $PAR_{subcarrier}$ is the peak-to-average power ratio of each subcarrier. This undesirable property requires an ADC to have a higher dynamic range in order to cope with the requirement of having a small probability of clipping events. Figure 1.1 shows a comparison of a single sine wave signal and a multi- carrier signal in time domain and their amplitudes statistical probabilities. It is clear that the characteristics of a multi-carrier signal are far different from that of a single sine wave signal with the same power. The multi-carrier signal is observed to have high peak amplitude much larger than that of the sine wave. Since the multi-carrier signal is a summation of a large number of narrowband signals with uncorrelated amplitude and phase, the resulting signal amplitude distribution is approaching a Gaussian distribution, according to the Central Limit Theorem. This means that small amplitudes have high probability of occurrence and the probability is exponentially decreased with increasing amplitude values.

Figure 1.2 shows a typical analog front-end block diagram which consists of a programmable gain amplifier (PGA) and an ADC with built in track and hold function. The PGA is used to adjust the incoming signal strength to optimally fit the

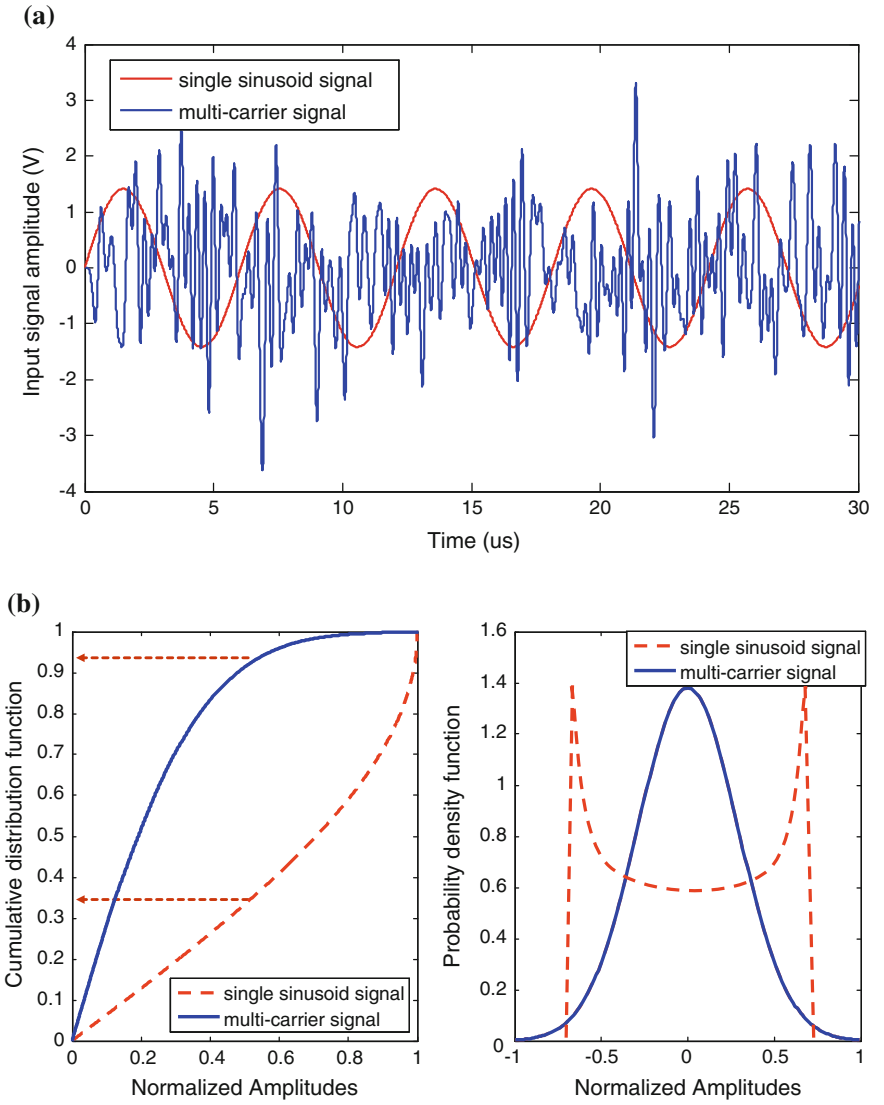

Fig. 1.1 **a** Multi-carrier signal versus single sinusoid signal in time domain, **b** Amplitude cumulative distribution function (CDF) and probability distribution (PDF)

input range of the ADC. In order to convert a multi-carrier signal with high PAR properly, both noise (including thermal, quantization noise) and distortion (circuit nonlinearity and clipping effects) have to be considered carefully. Too much gain (high input signal power) in the PGA will saturate the ADC and clip the signal, resulting in exponential growth of clipping distortion, while insufficient gain will

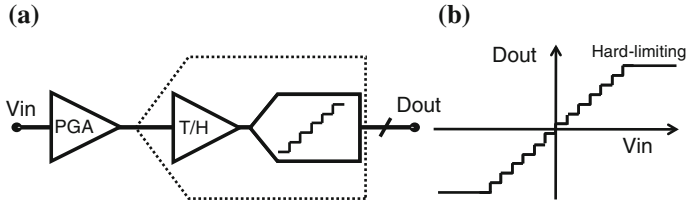

Fig. 1.2 **a** A typical analog front-end block diagram, **b** ADC input–output transfer curve

result in higher noise with respect to signal power, as the noise power[1] (such as thermal and quantization noise) is independent of signal power. Conventionally, a best compromise between noise and distortion can be found by backing off the input signal power from the ADC full scale power level by a large factor (e.g. 12 dB power back off for clipping probability less than 10^{-5}), in order to achieve an optimal signal-to-noise-and distortion ratio (SNDR) [18, 19]. This method leads to a very inefficient use of the ADC's dynamic range, and hence will result in low power efficiency, as it is shown in Fig. 1.1b that more than 90 % of the sampled signal amplitudes are below half of the maximum input voltage of the ADC.

1.2.2 Principle of the Parallel Sampling Architecture

A dual-channel version of the parallel sampling architecture is shown in Fig. 1.3 as an example, but the approach can be generalized easily to more channels. This ADC consists of two parallel sub-ADCs, each of them preceded by a range-scaling stage, and their outputs are combined by a signal reconstruction block. The front-end input signal is split into two signals which are just scaled versions of each other after the range scaling stages. The signal in the main path has the same strength as the front-end input signal, while the signal in the auxiliary path is an attenuated version of the front-end input signal. These two signals are sampled by the sub-ADCs at the same time. Depending on the input signal level, one of them will be chosen to reconstruct the signal in the digital domain.

The criterion is such that the signal in the main path is maximized to exploit the dynamic range of the sub-ADC efficiently; hence the ADC has more resolution over the small amplitudes that have relatively much higher probability of occurrence due to the multi-carrier signal statistical properties. Large amplitudes that saturate the sub-ADC in the main path will be replaced in the digital domain by the samples from the auxiliary path. This is possible as the sub-ADC in the auxiliary path quantizes an attenuated version of the ADC input signal which has lower

[1] The noise mentioned here only includes those sources which are independent of signal power, while other noise sources, e.g. jitter noise are considered not to be dominant in this discussion.

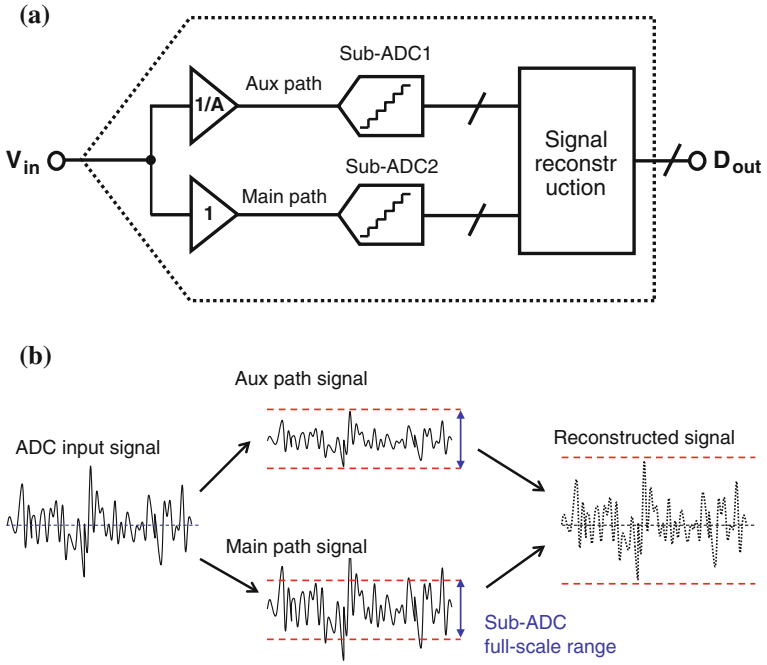

Fig. 1.3 **a** Block diagram of an ADC with the parallel sampling technique (a dual-channel version), **b** signal swing in different signal paths

probability of saturating the sub-ADC and a better linearity. During signal reconstruction, the auxiliary path provides coarsely quantized samples to replace the clipped or highly distorted large amplitude samples of the main path, hence avoiding excessive clipping noise and achieving good overall linearity. As shown in Fig. 1.3b, the front-end input signal full-scale range and the digital word-length of the reconstructed signal can be larger than that of each sub-ADC. The sampled signal in the main path has a better SNR than the auxiliary one due to a larger input signal swing, while the sampled auxiliary signal has a better signal-to-clipping-distortion ratio (SCDR). When the signal is reconstructed properly in the digital domain, the combination of the two sub-signals offers a better signal-to-noise-and-clipping-distortion ratio (SNCDR) compared to that of a single ADC.

1.2.3 Advantages of the Parallel Sampling Technique

The SNCDR of ADCs with and without the parallel sampling technique are expressed in the following equation:

$$SNCDR_{conventional\_ADC} = \frac{(\alpha \cdot V_{dd})^2}{F(RS_{ADC}) \cdot n + [1 - F(RS_{ADC})] \cdot D_{clip}} \tag{1.2}$$

$$SNCDR_{proposed\_ADC} = \frac{(A \cdot \alpha \cdot V_{dd})^2}{F(RS_{Sub.ADC1}) \cdot n_1 + F(RS_{Sub.ADC2}) \cdot n_2 + [1 - F(RS_{Sub.ADC1}) - F(RS_{Sub.ADC2})] \cdot D_{clip}} \tag{1.3}$$

where V_{dd} is the supply voltage, α is the voltage efficiency, A is the attenuation factor of the auxiliary signal, $F(x)$ is the distribution function of the input signal amplitudes, RS_{ADC} is the effective signal range processed by the ADC, n and D_{clip} are the noise and clipping distortion power, respectively.

For signals with known statistics, the optimal SNCDR of an ADC with and without the parallel sampling technique can be found from Eqs. (1.2) and (1.3). In order to make the analysis more clear, a comparison of the SNCDR of 11b ADCs with and without this technique is shown in Fig. 1.4. In this figure, the SNCDR versus input signal power is plotted. By properly choosing the attenuation factor A of the auxiliary signal path, the SNCDR of an ADC with the parallel sampling technique can be substantially improved compared to the one without this technique. This observation is also valid for ADCs with arbitrary number of bits. Using ADCs with lower resolution to achieve a desired system performance instead of higher resolution ADCs is much more power efficient, since generally power consumption of ADCs increases exponentially with resolution [1, 9].

In Fig. 1.4, it also shows that the ADC with the proposed technique at optimal SNCDR setting allows more than twice the input signal swing compared to a

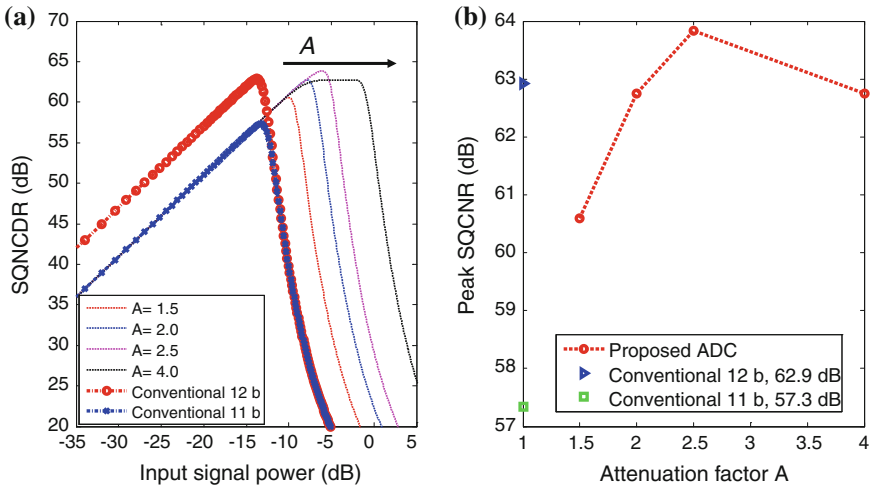

Fig. 1.4 Comparison of the SNCDR of the proposed ADC (with two 11b sub-ADCs) and the conventional ADCs for converting signals with Gaussian distributed amplitudes

conventional ADC (6 dB in power), which can be translated to a four times reduction of the sampling capacitors that determines the thermal noise floor (in case thermal noise is the dominant noise source) for getting a similar SNR as a conventional ADC. Although the ADC needs to process a signal at the ADC front-end with A times larger swing than normally can be applied, this large input signal swing does not need to pass through the sub-ADCs with this technique: the sub-ADC in the main path only looks at the part of the signal with small amplitude values, so the T/H and the quantizer do not need to maintain good linearity over the whole signal range. Compressing and clipping the large signal amplitude in the main path will not affect the final linearity of the output signal after reconstruction. While the range-scaling block in front of the auxiliary sub-ADC will translate this large input signal into the normal signal range of an ADC limited by linearity or supply voltage, a good linearity over the whole signal range can be achieved.

The proposed technique has similarity with conventional non-uniform methods (e.g. non-uniform Flash ADC [22] and floating-point ADCs [23]) in terms of improving the quantization resolution over small amplitudes without the need of a higher resolution ADC. One further benefit of this technique is that it allows enhancing the input signal range of an ADC without getting excessive clipping distortion. An enlarged input signal range and hence a higher voltage efficiency factor (α) is the key to lower power dissipation, as a desired SNR can be achieved with a much smaller total sampling capacitance. Although the ADC driver (e.g. PGA) may need to operate at a higher supply voltage to deliver a larger output signal swing, the reduction of the sampling capacitor of the ADC leads to better drivability and doesn't require extra power dissipation. This technique allows improving the power efficiency and reducing the silicon area of an ADC in comparison to conventional solutions, assuming equal system performance.

1.3 Review of Switched-Capacitors Pipeline ADCs

The pipeline ADC architecture is considered to offer optimum performance for medium-to-high resolution and medium-to-high sampling speed applications. It is very instructive to investigate the performance of the pipeline ADCs that have been published the last years. Such a study reveals the natural position of the pipeline ADC architecture with respect to other architectures, in terms of accuracy and speed. In [24], a survey which concerns all ADCs published in the ISSCC conference and VLSI symposium is presented. Using this survey and isolating the pipeline ADC, we can construct Fig. 1.5. It depicts the performance space of pipeline ADC, as it is expressed in ENOB (Effective Number of Bits) and sampling speed. Every point represents a pipeline ADC implementation.

The ellipse is added to include the majority of pipeline ADCs. It is clear that most of them lie in the region between 6 to 13b and 10 MS/s to 1 GS/s. Note that there are some pipeline ADC implementations targeting the high speed (>1 GS/s) and low resolution (<6b) range.

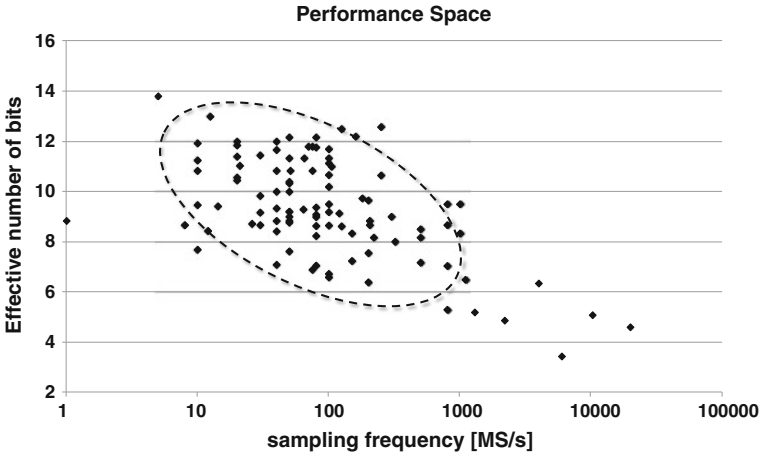

Fig. 1.5 Pipeline ADC's performance space

1.3.1 Pipeline ADC Architecture and Functionality

The general structure of a pipeline analog to digital converter architecture [25–27] is shown in Fig. 1.6. The ADC consists of a number of cascaded low resolution stages and a digital correction and encoding block. Each stage resolves a certain number of bits and generates a residue signal that is digitized by the succeeding stages.

A typical pipeline stage is shown in Fig. 1.7. It consists of the following sub-blocks:

- A sub-ADC, which typically consists of a set of comparators and quantizes the analog input of the stage.
- A sub-DAC that converts to analog the digital output of the sub-ADC.
- An analog subtraction block that calculates the quantization error.
- An amplifier that amplifies back to full range the subtraction's result, called the residue of the stage. The amplifier is often called residue amplifier.

It is common practice to implement the sub-DAC together with the subtraction block and the amplifier as a single block, called Multiplying-DAC (MDAC).

When the analog input voltage of a stage (V_{in}) is bounded by

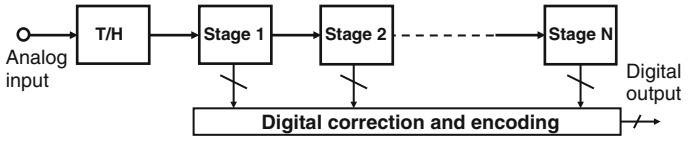

Fig. 1.6 General Pipelined ADC architecture

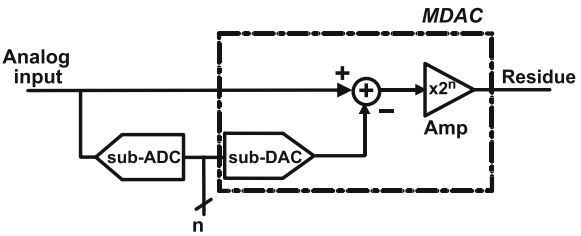

Fig. 1.7 Typical pipeline stage inner structure

$$-A_{\max} \leq V_{in} \leq A_{man} \tag{1.4}$$

the quantization error q is bounded by:

$$-\frac{A_{\max}}{2^n} \leq q \leq \frac{A_{\max}}{2^n} \tag{1.5}$$

where $[-A_{max}, A_{max}]$ is the input range of the ADC. The gain of the amplifier in each stage is chosen such to scale the quantization error back to a full scale input. Therefore, from (1.4) and (1.5) we conclude that the gain of the residue amplifier is equal to 2^n.

Digital logic is used to combine the separate digital outputs of each stage in a valid N-bit output code, where N is the resolution of the ADC.

Typically, in front of the first block of the pipeline a dedicated sample-and-hold stage is placed, sampling the analog input at the sample frequency f_s.

The main advantage of pipeline architecture is that due to stage pipelining, the maximum sampling frequency of the converter is determined only by the time period of a complete conversion cycle of a single stage. The propagation time through the cascade of pipeline stages results only in latency, meaning the time delay between an analog input and its digital representation. Depends on application, latency might causes a problem in case the ADC is used in the feedback path of a system. One of the critical design issues of a pipeline converter is the number of bits that is produced by each stage. Moreover, the number of bits can vary for each block in the pipeline. By making a correct distribution of the bits over the cells in the pipeline, the overall speed, accuracy, power consumption and chip area can be optimized.

1.3.2 Conventional Low Power Design Techniques for Pipeline ADC

Literature shows an abundance of power optimization techniques for pipeline ADCs and in this section we review two of the most common which are bit redundancy and stage scaling.

1.3.2.1 Bit Redundancy

A very popular correction technique that greatly eases the comparator's (sub-ADC) specifications is the introduction of code redundancy as described in [26]. By using this technique we can essentially use plain dynamic comparators, although they exhibit large offset.

The problem of a pipeline converter using a 1b resolution per stage and a gain-of-2 stage is that a deviation of one of the comparator levels results in exceeding the dynamic range of the next stage in the pipeline. Suppose that the level of the comparator in the sub-ADC (V_{ref}) is not exactly equal to 0. This can be due to static deviation of the reference voltage, or due to dynamic behavior of the comparator. In that case, it is possible that the stage's output voltage exceeds the allowed input range ($[-A_{max}, A_{max}]$) of the next stage, resulting in a large quantization error. The problem can be visualized in Fig. 1.8.

The solution is to use a gain stage with a gain less than 2 (radix < 2) or to maintain the same gain while increasing the number of bits in the sub-ADC and sub-DAC. The latter solution is more popular (since it eases the digital calculations) is demonstrated here. Whereas the gain remains equal to 2, the number of bits is increased from 1 to 1.5.

In this example the sub-ADC levels are placed at $\pm(1/3)A_{max}$ and the sub-DAC levels at $\pm(2/3)A_{max}$. Figure 1.9 shows an example of the behavior of a basic block with 1.5b per stage resolution.

We see that in the non-ideal case of Fig. 1.9 the V_{ref} is been moved and the output range changed. Nevertheless, as long as the consecutive stage is not saturated (output range <100 %), the error has absolutely no influence on the overall accuracy of the ADC and the correct digital word can be resolved.

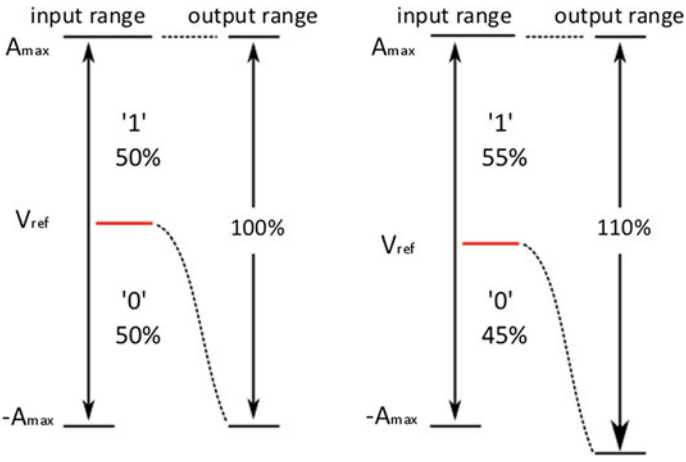

Fig. 1.8 Ideal behavior (*left*) and behavior in case of deviation of Vref (*right*) of a 1b per stage converter

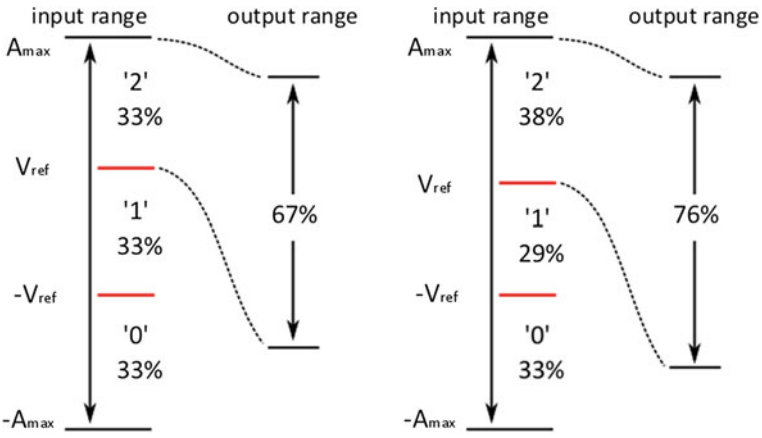

Fig. 1.9 Ideal behavior (*left*) and behavior in case of deviation of Vref (*right*) of a 1.5b per stage converter

The use of bit redundancy greatly reduces the accuracy requirements of the comparators used in the sub-ADC. Therefore they can be replaced by dynamic equivalents leading to large power savings.

1.3.2.2 Capacitor Scaling and Stage Resolution Optimization

Scaling down the sampling capacitors along the pipeline chain is enabled by the relaxed accuracy requirements (with respect to the thermal noise) due to the stages' amplification function. Therefore, proper scaling will decrease both the used chip area and the power consumption, without reducing overall speed or accuracy.

Figure 1.10 depicts a simplified model of closed-loop implementations of a pipeline stage and its loading [27].

According to ease the calculations and make the extraction of useful information feasible, the effective resolution (n) is assumed to be constant along the pipelined chain of the ADC. Although power savings do exist by allowing varying resolution from stage to stage, for the clarity of analysis we limit ourselves to

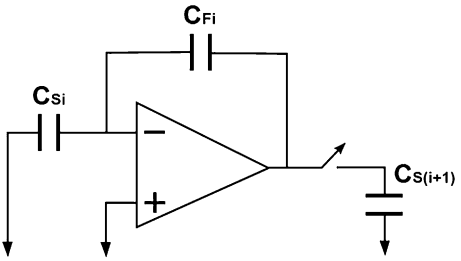

Fig. 1.10 Basic model for a closed-loop pipelined implementation

constant resolution per stage. The input referred thermal noise of a pipeline ADC is given by:

$$P_{therm} \propto k_B T \left[\frac{1}{C_{S_0}} + \frac{1}{2^{2n} C_{S_1}} + \frac{1}{2^{4n} C_{S_2}} + \frac{1}{2^{8n} C_{S_3}} + \cdots \right] \tag{1.6}$$

where k_B is Boltzmann's constant, T the temperature in K and C_{Si} the total sampling capacitance of the *i*th stage. It is clear that the input referred thermal noise and the noise distribution is dependent on two factors: (i) the size of the sampling capacitors (C_{Si}) and (ii) the effective resolution per stage (n). Therefore, finding the optimum scaling means finding the optimum values for those two parameters.

We can identify two extremes:

- *No scaling*, in which all the stages have the same size and contribute equally to the power consumption. In that case the thermal noise is dominated by the front-end of the pipelined chain.
- *Aggressive scaling*, in which all the stages contribute equally to the input referred thermal noise. In that case, the power is dominated by the front-end of the pipelined chain.

We can define the scaling factor [27] as the ratio of two consecutive sampling capacitances and equal to:

$$s = \frac{C_{Si}}{C_{S_{i+1}}} = 2^{nx} \tag{1.7}$$

where Cs_i is the sampling capacitor of the *i*th stage ($i \in [0, N/n - 1]$, with N the total ADC resolution), *n* is the effective number of bits per stage and *x* is the *taper factor*, $x \in [0, 2]$, a parameter that defines how "aggressive" or not is the applied scaling ($x = 0 \Rightarrow$ no scaling, $x = 2 \Rightarrow$ most aggressive scaling).

Assuming that the capacitance of the 0th stage is $C_{S0} = 2^n C_{unit}$, where C_{unit} is a unit capacitance, the sampling capacitance of the *i*th stage can then be expressed as:

$$C_{S_i} = \frac{2^n C_{unit}}{2^{inx}} \tag{1.8}$$

Given that the input-referred noise should be equal to or smaller than ½LSB leads to (1.9).

$$k_B T \left[\frac{1}{C_{S_0}} + \frac{1}{2^{2n} C_{S_1}} + \frac{1}{2^{4n} C_{S_2}} + \cdots \right] \leq \left(\frac{2FS}{2^N \sqrt{2}} \right)^2 \tag{1.9}$$

where ±FS is the voltage range.

By including in stage's load the sampling capacitor of the following stage ($C_{S(i+1)}$) and the feedback capacitor (C_{Fi}), the total pipelined power [27] is proportional to:

$$Total\ Pipelined\ ADC\ Power\ \sim \left[2^{n(1-x)} + 1 - \frac{1}{2^n}\right] \sum_i C_{S_k} \tag{1.10}$$

By combining the last two equations we can show that the total pipeline ADC power is proportional to:

$$Total\ Pipelined\ ADC\ Power\ \sim \left[2^{n(1-x)} + 1 - \frac{1}{2^n}\right] \left(\frac{1}{1-\frac{1}{2^{nx}}}\right) \left(\frac{1}{1-2^{n(x-2)}}\right) \tag{1.11}$$

The following figure shows how the total normalized power of the pipeline ADC changes as a function of the taper factor for different number of bits per stage.

Figure 1.11 reveals that the use of more bits per stage reduces the total power consumption (this model does not include sub-ADC loading or reduction of OTA's feedback factor for higher n values). Moreover, we see that the optimum taper factor moves to higher values, meaning that faster capacitor scaling is advantageous for higher number of bits per stage.

Concerning closed-loop pipeline ADC implementation, [2] enhances the analysis of [27] taking into account sampling speed considerations. The results also show that pipeline ADCs operating at low sampling speeds benefit from higher number of bits per stage. The reason is that in this way we minimize the total number of OTAs used in the pipeline chain. The extra loading due to the multi-bit sub-ADC and the reduction of OTAs' feedback factor can be tolerated. For high speed operation implementations with fewer bits per stage are advantageous.

Replacing the closed-loop MDAC implementation with an open-loop circuitry (as we mention later in this section) might lead to power advantages. A study on capacitor scaling in open-loop pipeline ADC realizations can be found in [28].

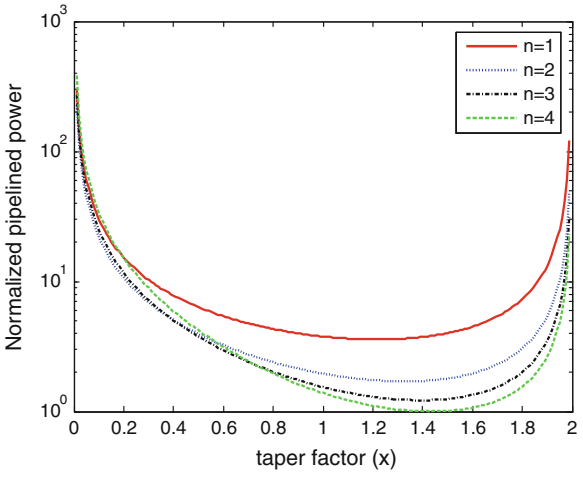

Fig. 1.11 Normalized power versus taper factor

1.3.2.3 Other Common Low Power Design Techniques for Pipeline ADC

Identifying the amplifier of the MDAC as the most power-demanding block of the pipeline ADC, several techniques have been developed aiming at the reduction of the total power consumed by them.

This reduction can be achieved in three ways: (i) by reducing the power consumption of each amplifier, (ii) by reducing the total count of amplifiers in the pipeline chain or (iii) by replacing the MDAC's amplifier with a less power hungry circuit.

The *Correlated Double Sampling* (CDS) technique [29] suppresses the finite gain effects of the amplifier and therefore we can use an OTA with lower gain (more power efficient). The basic idea is to sample the gain error in a preliminary charging phase using auxiliary capacitors. This technique requires more area (additional capacitors) and more complicated clocking scheme (preliminary phase), but achieves high precision.

An improvement to the CDS is the *Time-Shifted CDS* technique [30]. This technique uses different scheduling (timing) of the operations and implements CDS without the two aforementioned disadvantages.

A technique that reduces the number of OTAs in a closed-loop amplifier implementation is the *amplifier sharing* technique [31]. This technique is enabled by the operation mode of the pipeline ADC, which dictates that when one stage is in sampling mode the adjacent stage is in amplification mode. Therefore it is possible two pipelined stages to share an OTA, halving the total OTA count of the pipeline.

Omitting the front-end S&H amplifier (*S&H-less*) is another popular approach to reduce the number of OTAs [32]. The front-end S&H contributes significantly to the noise and linearity performance of pipeline ADCs, demanding high power S&H solutions. Employing this technique the first stage of the pipeline is not preceded by a dedicated S&H amplifier. Nevertheless the sampling operation is still needed and performed by the MDAC and sub-ADC of the first stage. For wideband (high-speed) operation this might lead to performance degradation due to bandwidth and timing mismatch between the two paths. For this reason digital correction techniques that measure and correct the mismatch have been developed [33].

Newer technologies offer less analog accuracy but more digital functionality. This trend justifies the solution of replacing the OTA of the MDAC with a less accurate circuit (*OTA-less*) and use digital means to amend the inaccuracies. An example of this line of thinking is the utilization of open-loop amplifiers with digital calibration [5] as a substitute to the OTAs. Furthermore, recent designs demonstrate that the MDAC amplifiers can be replaced by inherently power efficient dynamic circuits such as comparators [34], switched source followers [35], capacitive charge-pumps [36] and ring-oscillators [37].

1.4 The Proposed Pipeline ADC with a Parallel Sampling First Stage

The first stage of the proposed pipeline ADC is implemented with the parallel sampling technique as it is the most critical one in terms of speed and noise performance, hence consuming significant power. The requirements and power consumption are decreased exponentially for the succeeding stages with stage scaling as it is explained in the previous section. The first stage is a 2.5b stage when the main path is selected and 1.5b stage when the auxiliary is selected. The backend ADC is implemented with conventional pipeline stages applied with conventional low power technique (Bit Redundancy, Capacitor scaling and stage resolution optimization). The proposed ADC uses a SHA-less frontend to further reduce the total power consumption [32, 33].

Figure 1.12 shows the block diagram of the first stage of the pipeline ADC with the parallel sampling technique. There are three paths for the input signal which are the main signal path, the auxiliary signal path and the detection path. The main and auxiliary signal path each consists of a signal scaling block and a passive sampling network (T/H), and they are multiplexed by a channel selection block (MUX) to a subtraction block and then to the amplification block (AMP). The detection path consists of a Flash-ADC and a digital-to-analog converter (DAC).

Instead of maintaining the same input and output signal range in the first stage, a channel selection through a MUX is introduced to decouple the choice of the stage's input and output signal swing. This selection is input signal level dependent. The input signal range is maximized for the purpose of reducing the capacitor size while meeting the desired SNR, while the output signal range is optimized for the linearity consideration of the T/H and amplifier. The "scaling" of the signal range relaxes the linearity requirement of the T/H and yields a substantial reduction in the

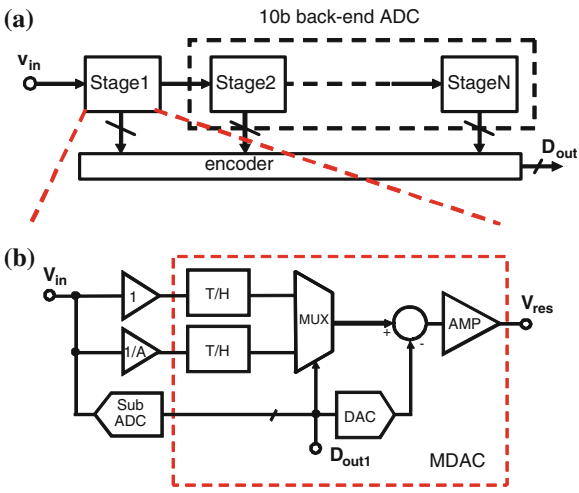

Fig. 1.12 **a** The architecture of the proposed pipeline ADC, **b** block diagram of the first pipelined stage with parallel sampling technique

power dissipated by the amplifier used for residue generation, due to the reduction of the output signal swing and loading capacitor size. With a 1.2 V supply voltage, the peak-to-peak differential input signal swing can be as high as 2 V. The first stage's residue amplifier and the back-end pipeline chain on the other hand require larger voltage headroom. The dynamic selection through the MUX therefore reduces the output signal swing with a factor of 2.5 to 0.8 Vppd. The voltage efficiency at the stage's input and output then is about 0.8 and 0.3 respectively.

Compared to the block diagram in Fig. 1.3, the back-end stages of this pipeline ADC are shared instead of using two ADCs in parallel to improve power efficiency. The amplifier in the first stage is also shared by the two paths, which further reduces the power dissipation significantly, since most of the power in a pipeline stage is dissipated by the amplifiers used for residue generation.

1.4.1 First Stage Implementation and Design Considerations

In Fig. 1.13, a schematic representation of the first stage and its timing diagram are shown. Although it is designed and implemented as a fully differential circuit, a single-ended version is shown for clarity. The signal scaling blocks at the left are designed simply with resistors. The resistive divider in the auxiliary path

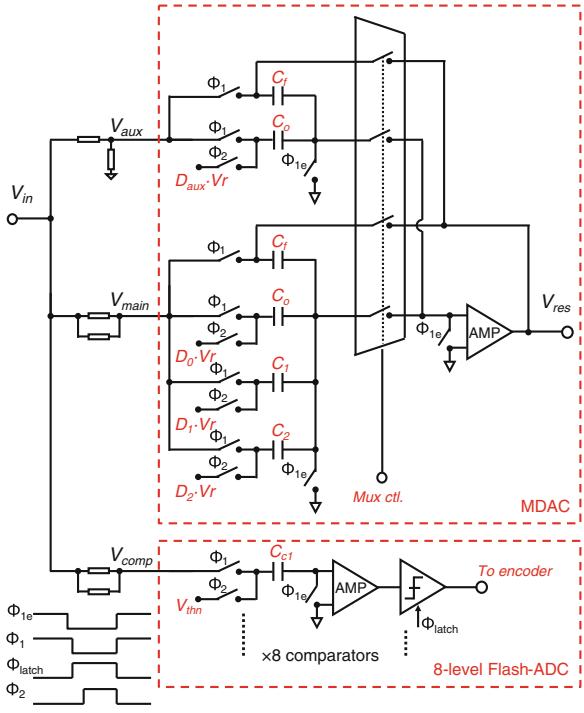

Fig. 1.13 Schematic and timing diagram of the first pipelined stage with the parallel sampling technique

attenuates the input signal swing by a factor of 2.5 (an attenuation factor chosen for achieving a close to optimal SNCDR from analytical simulation and taking into account implementation complexity), while the resistor circuit in the main signal path keeps the signal un-attenuated. The unit resistors are sized to have an intrinsic matching according to the design target, and together with the input common bias resistors, they provide a 50 Ohm input termination. The RC time constants in these two signal paths are designed to be the same to make sure that the bandwidth of the two signal paths matches well.

The resolution of the MDAC for the main signal path is chosen to be 2.5b for a good trade-off between power and speed [2], and with a signal gain of 4 to relax the requirements on the backend stages. The MDAC in the auxiliary path is chosen to be 1.5b [26] for simplicity, since the probability of utilizing the signal in this path is far lower than that of the main path. Both MDACs are switched-capacitor circuits employing "flip-around" charge redistribution which benefit from larger feedback factors compared to a "non-flip-around" architecture [38]. They implement the algorithm in (1.12) and its voltage transfer curve is shown in Fig. 1.14.

$$\begin{cases} V_{res} = \dfrac{C_f + C_0 + C_1 + C_2}{C_f} \cdot V_{in} - \dfrac{(\mathrm{D}_0 \cdot C_0 + \mathrm{D}_1 \cdot C_1 + \mathrm{D}_2 \cdot C_2)}{C_f} \cdot V_r \quad if \ -\dfrac{7}{8}V_r < V_{in} < \dfrac{7}{8}V_r \\ V_{res} = \dfrac{1}{A} \cdot \dfrac{C_f + C_0}{C_f} \cdot V_{in} - \dfrac{\mathrm{D}_{aux} \cdot C_0}{C_f} \cdot V_r \quad if \ v_{in} < -\dfrac{7}{8}V_r \text{ or } V_{in} > \dfrac{7}{8}V_r \end{cases} \tag{1.12}$$

where V_{in} and V_{res} are the input and output signal of the stage respectively, V_r is the reference voltage, $D_0 \sim D_2$ and D_{aux} are control bits generated by the encoder of the Flash-ADC in the detection path. The total sampling capacitor size ($C_f + C_0 + C_1 + C_2$ in the main path and $C_f + C_0$ in the aux path) is chosen to meet the overall SNR requirement with respect to the full-scale voltage of 2Vppd. This "flip-around" scheme achieves a closed loop gain of 4 with a feedback factor of ¼ in the main path and 2 and ½ in the auxiliary path respectively. Bootstrapped switches [8] and bottom-plate sampling are used to reduce distortion. The amplifier uses a single stage folded cascode with gain boosting configuration which is similar to the one in [39]. The simulated DC gain of the MDAC amplifier is about 70 dB and the gain-bandwidth higher than 1 GHz.

The flash-ADC in the detection path consists of eight comparators (each consists of a pre-amp and a regenerative latch) and a resistive reference ladder. Compared to that of a conventional 2.5b stage, two additional comparators are needed to identify if the input signal is smaller or larger than the allowable input range of the main channel and to decide which channel should be connected to the residue amplifier. The decisions of the stage, for the various input ranges, are listed in Table 1.1.

The MDAC operates with two-phase non-overlapping clocks denoted as Φ_1 and Φ_2, the sampling and the amplification phase, respectively. During Φ_1, the signal is tracked by the sampling capacitors in both signal paths and the sampling network

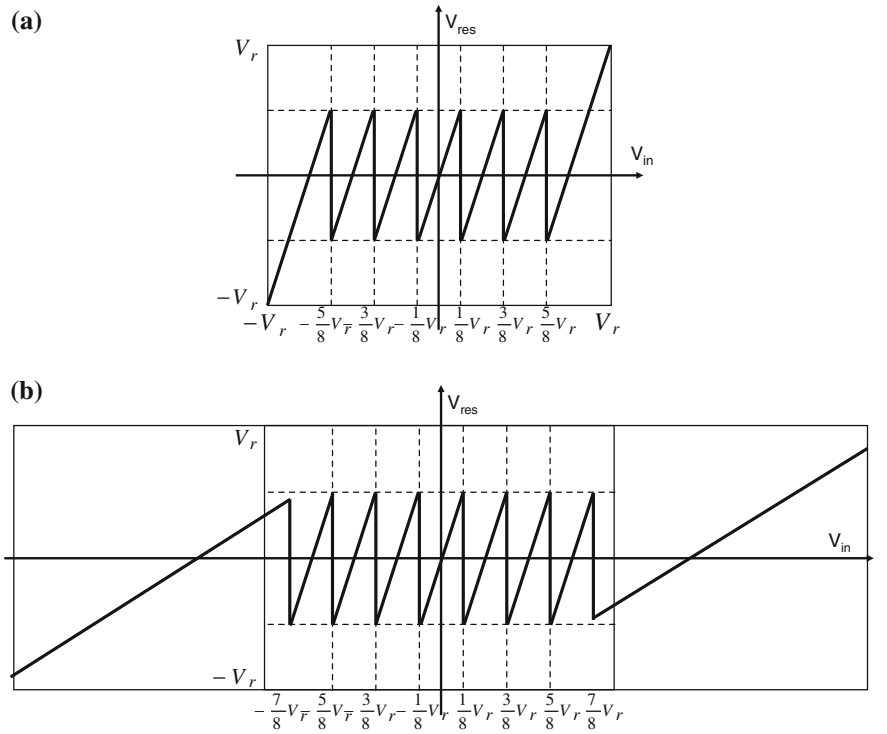

Fig. 1.14 First stage residue amplifier transfer curve: **a** conventional 2.5b stage, **b** proposed stage with enlarged input range

Table 1.1 Decision of the first pipeline ADC stage

Analog input	Stage decision			
	D_0	D_1	D_2	D_{aux}
$V_{in} \leq -7/8 \cdot Vr$				−1
$-7/8 \cdot Vr \leq V_{in} \leq -5/8 \cdot Vr$	−1	−1	−1	
$-5/8 \cdot Vr \leq V_{in} \leq 3/8 \cdot V$	−1	−1	0	
$-3/8 \cdot Vr \leq V_{in} \leq -1/8 \cdot Vr$	−1	0	0	
$-1/8 \cdot Vr \leq V_{in} \leq 1/8 \cdot Vr$	0	0	0	
$1/8 \cdot Vr \leq V_{in} \leq 3/8 \cdot Vr$	1	0	0	
$3/8 \cdot Vr \leq V_{in} \leq 5/8 \cdot Vr$	1	1	0	
$5/8 \cdot Vr \leq V_{in} \leq 7/8 \cdot Vr$	1	1	1	
$V_{in} \geq 7/8 \cdot Vr$				1

of the flash-ADC. The sampling actions are controlled by the same clock signal, and take place at the falling edge of Φ_{1e}; both V_{aux} and V_{main} are sampled onto the sampling capacitors simultaneously. Then, at the rising edge of Φ_{latch}, the sub-ADC detects the signal level. After the decision is made, proper reference voltages

($\pm V_r$ or V_{cm}) are chosen and connected to the sampling nodes of the capacitors for subtraction. At the same time, the feedback capacitor C_f of the main or auxiliary sampling network is selected and flipped around the amplifier through the MUX as a feedback capacitor for charge redistribution and produces a residue signal for the following stages.

In the proposed parallel sampling first stage, the overall performance could be affected by mismatches between two signal paths (e.g. gain, offset, bandwidth, and timing mismatches), as the input signal is processed by two parallel signal paths (main and auxiliary). This resembles the situation of the well-known time-interleaved ADCs [40]. However, they have very distinct differences. Two parallel paths of the proposed stage are sampled synchronously instead of in a time-interleaved fashion. The errors due to mismatches do not affect the reconstructed signal in a repetitive manner; these errors are only introduced when the samples from the main path need to be replaced by that of the auxiliary path. As the probability of signal amplitudes that are clipped in the main signal path is very small due to the multi-carrier signal properties, the number of samples from the auxiliary path that are used to replace the samples from the main path is very small compared to the total number of samples for the reconstructed signal. As a result, error power due to mismatch errors is far smaller than that of time-interleaved ADCs. Calibration techniques developed for time-interleaved ADCs can be applied to the proposed stage to minimize these errors. Furthermore, because there are only two signal paths in the proposed stage, the calibration complexity can be far lower compared to that of time-interleaved ADCs, which normally contain many parallel channels (e.g. 64 channels in [14]).

1.4.2 Power and Performance Comparison

In a high-speed and high-resolution switched-capacitor pipeline ADC, the SNR is normally dominated by thermal noise, as quantization noise can be reduced by adding extra stages to the last stage with a minimal impact on ADC power dissipation and sampling rate [2, 27]. Improving a conventional 11b pipeline ADC that is limited by thermal noise by one extra effective number of bit means about 4 times increase in power, assuming constant input signal range and the same sampling rate. The proposed 11b pipeline ADC with parallel sampling technique allows increasing the SNR without the need of a much larger sampling capacitor to reduce the noise power. By enlarging the input signal range 2.5 times, the SNR of the sampled signal in the main path is increased by more than 6 dB with little increase in power consumption, as power needed for the residue amplifier to drive the loading capacitor stays the same and the additional circuits (an additional passive T/H, a MUX and two more comparators in the sub-ADC for out of range detection) consume mostly dynamic power. As will be shown in the next section, the proposed 11b pipeline ADC has similar SNCDR compared to that of a conventional 12b pipeline ADC for converting multi-carrier signals. As a result, equal

system performance can be achieved with the proposed 11b pipeline ADC with less than half the power and area in ADC for multi-carrier systems which require a 12b conventional ADC.

1.5 Simulation Results

Conventionally, the dynamic performance of ADCs is characterized by a single sine wave test. This is not enough in proving the ADC's performance for multi-carrier systems. As was explained in Sect. 1.2, the statistical properties of multi-carrier signals are far different from that of a single full scale sine wave. In this paper, a multi-tone signal that has signal amplitude distribution approaching a Gaussian distribution was used to characterize the ADC performance which is shown in Fig. 1.15a, b. The performance of the proposed 11b pipeline ADC is verified by both behavioral simulation in Matlab and transistor level simulation in Cadence.

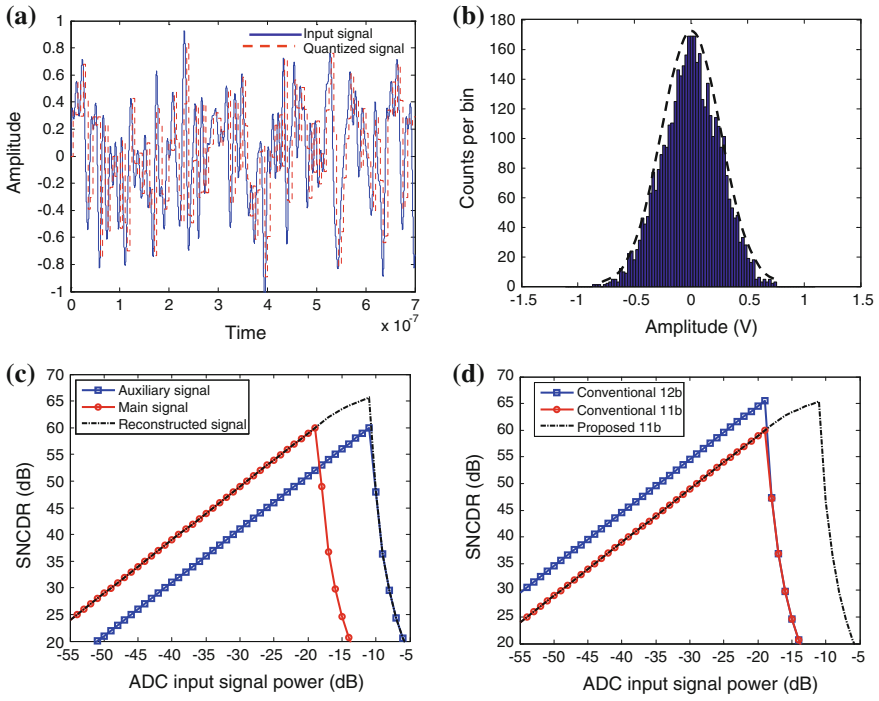

Fig. 1.15 The multi-tone input signal used for characterizing the ADC performance: **a** time domain signal, **b** amplitudes histogram, **c** SNCDR of the proposed 11b pipeline ADC (about 85 % of the input signals is processed by the main path at optimal SNCDR), **d** SNCDR of ADCs with and without parallel sampling technique

1.5.1 Behavioral Simulations

The behavioral model of the proposed ADC is implemented in MATLAB, and only thermal noise, quantization noise, and clipping distortion are considered in simulations. Figure 1.15c shows the plot of SNCDR of the output signal of main, auxiliary path, and the reconstructed signal with respect to input signal power. When the input signal power is low, the SNCDR of the reconstructed signal follows that of the output signal of main path as most of the samples are processed by the main path. With the increase of input signal power, large amplitudes that are clipped in the main path are replaced by their attenuated versions from the auxiliary path, but majority of the samples are still processed by the main path, hence the SNCDR of the recombined signal keeps increasing until the signal in the auxiliary path starts to clip excessively. The SNCDR of the 11b ADC with and without the parallel sampling technique are also compared and plotted in Fig. 1.15d, the proposed 11b pipeline ADC achieves a similar peak SNCDR as a conventional 12b ADC, and an improvement of about 5 dB in SNCDR and about 8 dB in dynamic range (DR) compared to an 11b ADC. Results of these simulations meet and support our analysis in the previous sections.

1.5.2 Transistor Level Simulations

The proposed pipeline ADC is implemented in TSMC 65 nm CMOS technology and operates at 1.2 V supply voltage and with a sampling rate of 200 MS/s. The linearity of the proposed ADC is verified by testing its Noise Power Ratio (NPR), which is recommended in [41] for wideband multi-carrier applications and it is similar with multi-tone power ratio (MTPR) testing. In these applications, the input signal contains a large number of narrow bandwidth signals, and it is desired that distortion should not interfere with the detection of weaker signals. In the testbench, a multi-tone signal (with 58 tones, PAR around 10 dB, and input signal amplitude 2Vppd) is generated which possesses an approximately uniform spectrum over the bandwidth of interest, except for a narrow band of frequencies intentionally "missing", as shown in Fig. 1.16a. The output waveforms are analyzed to determine how much power has leaked into the "missing" band. The NPR is then calculated from

$$NPR = 10\log 10\left(\frac{Psig.avg}{Pn.avg}\right) \tag{1.13}$$

where *Psig.avg* is the average power spectral density outside the missing frequency band and *Pn.avg* is the average power spectral density inside the missing band.

The simulated output spectrum of both the signal paths in the first stage is shown in Fig. 1.16b, c. With a 2Vppd input signal, the signal in the main path is

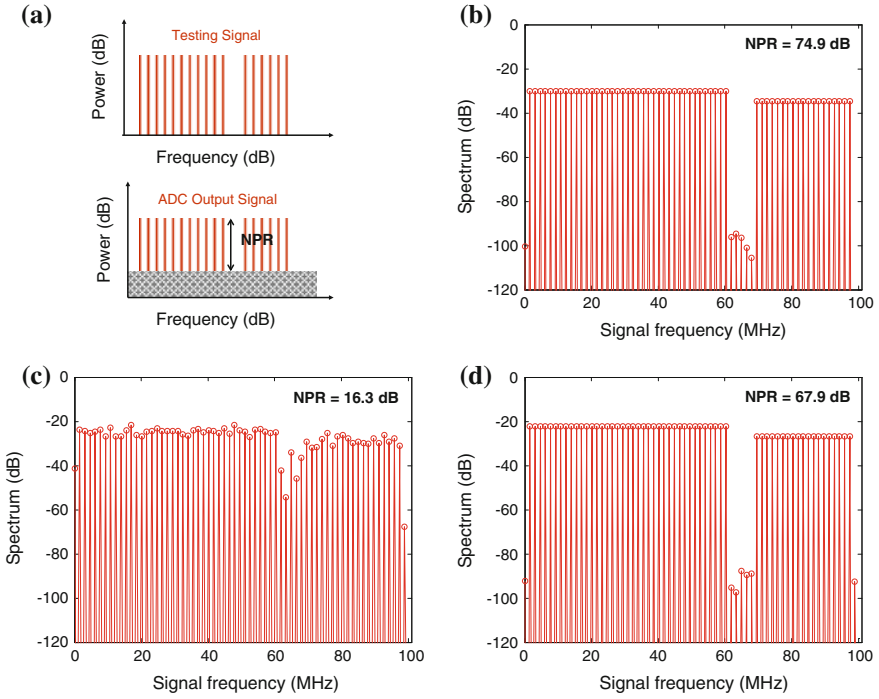

Fig. 1.16 **a** Principle of NPR testing, **b** Spectrum of the auxiliary path output signal, **c** Spectrum of the main path output signal, **d** spectrum of the reconstructed signal

highly distorted and results in only 16 dB NPR, while the NPR in the auxiliary channel is about 75 dB since the input signal of the auxiliary path is attenuated by a factor of 2.5 before it is sampled by the T/H. The reconstructed signal at the output of the first stage has a NPR of 68 dB (only linearity is included in this simulation), shown in Fig. 1.16d. Comparing it with the NPR of an ideal 12b ADC which is 62.7 dB [39], the distortion power of this ADC is below the noise floor of a 12b ADC by about 5 dB. The overall NPR of this ADC will be limited mainly by the thermal noise.

1.6 Conclusions

A parallel sampling technique for ADCs is proposed for converting multi-carrier signals efficiently by exploiting the statistical properties of these signals. It enables the ADC to have a large input signal range with a low supply voltage while relaxing the linearity requirements of the input sampling stage, hence reducing the power consumption and silicon area for achieving a desired SNR in comparison to conventional solutions. An 11b switched-capacitor pipeline ADC based on this

technique was implemented in 65 nm CMOS technology. It has a large input signal range of 2 V differentially with a 1.2 V supply voltage and achieves an SNCDR similar to that of a conventional 12b pipeline ADC with good linearity for converting multi-carrier signals but consuming less than half of its power and area.

References

1. Uyttenhove, K., Steyaert, M.S.J.: Speed-power-accuracy tradeoff in high-speed CMOS ADCs. IEEE Trans. Circuits Syst. II **49**(4), 280–287 (2002)
2. Chiu, Y.: Analysis and Design of Pipelined Analog-to-Digital Converters, Springer-Verlag, New York Incorporated (2006), ISBN: 0387270396
3. Frey, M., Loeliger, H.-A.: On the Static Resolution of Digitally Corrected Analog-to-Digital and Digital-to-Analog Converters With Low-Precision Components. IEEE Trans. Circuits Syst. I **54**(1), 229–237 (2007)
4. Moon, U., Song, B.S.: Background digital calibration techniques for pipelined ADCs. IEEE Trans. Circuits Syst. II **44**, 102 (1997)
5. Murmann, B., Boser, B.: A 12-bit 75-MS/s pipelined ADC using open-loop residue amplification. IEEE J. Solid-State Circuits **38**(12), 2040–2050 (2003)
6. Chiu, Y., Tsang, C.W., Nikolic, B., Gray, P.R.: Least mean square adaptive digital background calibration of pipelined analog-to-digital converters. IEEE Trans. Circuits Syst. I **51**, 38 (2004)
7. Panigada, A., Galton, I.: Digital background correction of harmonic distortion in pipelined ADCs. IEEE Trans. Circuits Syst. I **53**, 1885–1895 (2006)
8. Abo, M., Gray, P.R.: A 1.5-V, 10-bit, 14.3-MS/s CMOS pipeline analog-to-digital converter. IEEE J. Solid-State Circuits **34**, 599–606 (1999)
9. Murmann, B.: A/D converter trends: power dissipation, scaling and digitally assisted architectures. In: Proceedings IEEE Custom Integrated Circuits Conference (CICC) (2008)
10. Stoeble, O., Dias, V., Schwoerer, C.: An 80 MHz 10b pipeline ADC with dynamic range doubling and dynamic reference selection. IEEE international solid-state circuits conference (ISSCC) digest of technical papers, pp. 462−539, Feb 2004
11. Van de Vel, H., Buter, B., van der Ploeg, H., Vertregt, M., Geelen, G., Paulus, E.: A 1.2 V 250 mW 14b 100 MS/s digitally calibrated pipeline ADC in 90 nm CMOS. IEEE Symposium on VLSI Circuits, pp. 74−75, 18−20 June 2008
12. Choi, H.C., Kim, J.H., Yoo, S.M., Lee, K.J., Oh, T.H., Seo M.J., Kim, J.W.: A 15 mW 0.2 mm2 10b 50 MS/s ADC with wide input range. IEEE international solid-state circuits conference (ISSCC) digest of technical papers, pp. 226−227, Feb 2006
13. Chen, C.-Y., Wu, J.: A 12b 3 GS/s pipeline ADC with 500 mW and 0.4 mm2 in 40 nm digital CMOS. In: Symposium on VLSI Circuits (VLSIC), June 2011
14. Doris, K., Janssen, E., Nani, C., Zanikopoulos, A., Van der Weide, G.: A 480 mW 2.6 GS/s 10b 65 nm CMOS time-interleaved ADC with 48.5 dB SNDR up to Nyquist. In: IEEE international solid-state circuits conference (ISSCC) digest of technical papers, Feb 2011
15. Lampe, L., Witrisal, K.: Challenges and recent advances in IR-UWB system design. In: IEEE international Symposium on Circuits and Systems (ISCAS), 30 May 2010
16. Meng, J., Ahmadi-Shokouh, J., Li, H., Han, Z., Noghanian, S., Hossain, E.: Sampling rate reduction for 60 GHz UWB communication using compressive sensing. Asilomar conference on signals, systems and computers (2009)
17. Janssen, E., et al.: A direct sampling multi-channel receiver for DOCSIS 3.0 in 65 nm CMOS. In: Symposium on VLSI Circuits (VLSIC), pp. 292−293, 15−17 June 2011
18. Mestdagh, D.J.G., Spruyt, P.M.P., Biran, B.: Effect of amplitude clipping in DMT-ADSL transceivers. Electron. Lett. **29**(15), 1354–1355 (1993)

19. van Beek, P.: Multi-carrier single-DAC transmitter approach applied to digital cable television. Ph.D. thesis, TU Eindhoven (2011)
20. Lin, Y., Doris, K., Hegt, H., Roermund, A.: An 11b pipeline ADC with dual sampling technique for converting multi-carrier signals. In: IEEE international symposium on circuits and systems (ISCAS), May 2011
21. Lin, Y., Doris, K., Hegt, H., Roermund, A.: An 11b pipeline ADC with parallel-sampling technique for converting multi-carrier signals. IEEE Trans. Circuits Syst. I **99**, 1–9 (2012)
22. Lin, L.: Piecewise-linear, non-uniform ADC. US Patent US 6,498,577, 2002
23. Thompson, B.: Wooley, A 15-b pipeline CMOS floating-point A/D converter. IEEE J. Solid-State Circuits **36**(2), 299–303 (2001)
24. Murmann, B.: ADC performance survey 1997−2011, [Online]. Available: http://www.stanford.edu/∼murmann/adcsurvey.html
25. Lewis, S.H., Gray, P.R.: A pipelined 5-MS/s 9-bit analogto-digital converter. IEEE J. Solid-State Circuits **22**(6), 954−961 (1987)
26. Lewis, S.H., Fetterman, H.S., Gross Jr, G.F., Ramachandran, R., Viswanathan, T.R.: A 10-b 20-MS/s analog-to-digital converter. IEEE J. Solid-State Circuits **27**(3), 351–358 (1992)
27. Cline, D., Gray, P.: A power optimized 13-b MS/s pipelined analog-to-digital converter in 1.2 μm CMOS. IEEE J. Solid-State Circuits **31**(3), 294–303 (1996)
28. Zanikopoulos, A., Harpe, P., Hegt, H., van Roermund, A.: Power optimization for pipelined ADCs with open-loop residue amplifiers. In: IEEE ICECS 2006, Dec 2006
29. Nagaraj, K., Viswanathan, T.R., Singhal, K., Vlach, J.: Switched-capacitor circuits with reduced sensitivity to amplifier gain. IEEE Trans. Circuits Syst. I **34**(5), 571–574 (1987)
30. Li, Jipeng, Moo, Un-ku: A 1.8 V 67 mW 10b 100 MS/s pipelined ADC using time-shifted CDS technique. IEEE J. Solid-State Circuits **39**, 1468–1476 (2004)
31. Huang, Yen-Chuan, Lee, Tai-Cheng: A 10b 100 MS/s 4.5mW pipelined ADC with a time-sharing technique. IEEE Trans. Circuits Syst. I **58**(6), 1157–1166 (2011)
32. Mehr, I., Singer, L.: A 55-mW 10-bit 40-MS/s nyquist-rate CMOS ADC. IEEE J. Solid-State Circuits **35**, 318 (2000)
33. Huang, P., Hsien, S.-K., Lu, V., Wan, P., Lee, S.-C., Liu, W., Chen, B.-W., Lee, Y.-P., Chen, W.-T., Yang, T.-Y., Ma, G.-K., Chiu, Y.: SHA-less pipelined ADC with in situ background clock-skew calibration. IEEE J. Solid-State Circuits **46**, 1893–1903 (2011)
34. Brooks, L., Lee, H.S.: A 12b 50 MS/s fully differential zero-crossing-based ADC without CMFB, In: IEEE international solid-state circuits conference (ISSCC) digest of technical papers, Feb 2009
35. Hu, J., Doley, N., Murmann, B.: A 9.4b, 50 MS/s, 1.44 mW pipelined ADC using dynamic source follower residue amplification. In: IEEE international solid-state circuits conference (ISSCC) digest of technical papers, Feb 2009
36. Ahmed, I., Mulder, J., Johns, D.: A 50 MS/s 9.9 mW pipelined ADC with 58 dB SNDR in 0.18 μm CMOS using capacitive charge-pumps. IEEE J. Solid-State Circuits **35**, 318 (2009)
37. Hershberg, B., Weaver, S.,Sobue, K., Takeuchi, S., Hamashita, K.,Moon, U.: Ring amplifiers for switched-capacitor circuits. In: IEEE international solid-state circuits conference (ISSCC) digest of technical papers, Feb 2012
38. Song, B.-S., Tompsett, M.F., Lakshmikumar, K.R.: A 12-bit 1-MS/s capacitor error-averaging pipelined A/D converter. IEEE J. Solid-State Circuits **23**(6), 1324–1333 (1988)
39. Sumanen, L., Waltari, M., Halonen, K.A.I.: A 10-bit 200-MS/s CMOS parallel pipeline A/D converter. IEEE J. Solid-State Circuits **36**(7), 1048–1055 (2001)
40. Kurosawa, N., Kobayashi, H., Maruyama, K., Sugawara, H., Kobayashi, K.: Explicit analysis of channel mismatch effects in time-interleaved ADC systems. IEEE Trans. Circuits Syst. I **48**, 261–271 (2001)
41. IEEE standard for terminology and test methods for analog-to-digital converters. IEEE Std 1241–2010, pp. 1−139, 14 Jan 2011

Chapter 2
Design of Power, Dynamic Range, Bandwidth and Noise Scalable ADCs

B. Bakkaloglu, S. Kiaei, H. Kim and K. Chandrashekar

Abstract The proliferation of portable electronic devices with high data-rate wireless communication capabilities and the increasing emphasis on energy efficiency is continuously applying pressure on the performance and power consumption of ADCs and other mixed-signal systems. Power scalable designs enable an ADC core to be reusable under different input and sampling frequency conditions improving system efficiency. The power consumption of pipeline and ΣΔ ADCs scales approximately linearly with sampling rate and roughly quadruples for every additional bit resolved. Hence, increasing the performance requirements of an ADC in a system can significantly increase the power consumption to impractical levels especially in a battery powered environment. The approaches presented in this chapter focus on design techniques for power scalable and low power pipeline and bandwidth scalable continuous-time ΣΔ ADCs.

2.1 Introduction

The proliferation of portable consumer electronics and the increasing emphasis on energy efficiency continuously apply pressure on the power and performance of Analog-to-Digital Converters (ADC) and other mixed signal systems. Most electronic systems today rely heavily on digital processing to achieve higher integration and lower static power consumption. In most instances, it is desirable to move from

B. Bakkaloglu · S. Kiaei (✉)
School of Electrical, Computer and Energy Engineering, Arizona State University, Tempe, AZ 85287, USA
e-mail: Sayfe.Kiaei@asu.edu

H. Kim
Analog Design Engineer, Intel Corporation, Hillsboro, OR 97124, USA

K. Chandrashekar
Research Scientist, Intel Corporation, 2111 NE 25th Ave, Hillsboro, OR 97124, USA

P. Carbone et al. (eds.), *Design, Modeling and Testing of Data Converters*, Signals and Communication Technology, DOI: 10.1007/978-3-642-39655-7_2,

analog signal processing to digital signal processing as early as possible in the system's signal chain. Typical analog circuitry provides amplification of desired signals and filtering to improve the input signal dynamic range. The elimination of analog signal processing circuitry results in signals with poorer dynamic range, which carry less information, leading to lower data rates unless significant improvements are made in the resolution and sampling rate (F_S) of ADCs, which serve as the link between the analog and digital domains. Thus, the performance of ADCs is critical to applications such as Software Defined Radio (SDR), bio-medical sensors, and wideband wireless communication systems. These applications typically require pipeline ADCs, which have the capability to efficiently achieve medium to high resolution (8–14 b) at high sampling rates (20–200 MS/s).

The power consumption of pipeline and ΣΔ ADCs scales approximately linearly with sampling rate and roughly quadruples for every additional bit resolved.[1] Hence, increasing the performance requirements of an ADC in a system can significantly increase the power consumption to impractical levels especially in a battery powered environment. The research presented in this chapter focuses on design techniques for power and dynamic range scalable ADCs.

Power scalable designs enable an ADC core to be reusable under different input and sampling frequency conditions improving system efficiency. Power consumption of an ADC is typically optimized for a specified SNDR requirement at a given sampling rate and input frequency range. When the input frequency (f_{in}) range increases, increasing sampling rate to meet Nyquist conditions, the ADC analog core has to be re-designed for more stringent settling time requirements. Same approach is required for radio receivers where blocker profile of the receive channel changes. Scalable and reconfigurable designs aim to reuse the ADC core over a range of sampling and input frequencies by scaling power consumption.

The first part of this chapter presents a power scalable 12 b pipeline ADC that enables or disables OTAs connected in parallel to scale the settling response of Multiplying DAC (MDAC) and Sample/Hold (S/H) amplifiers in order to achieve constant SNDR performance over a range of sampling rates. The proposed technique facilitates optimal power consumption over the entire sampling rate range and reduces design complexity by maintaining constant DC bias conditions in the scaled analog blocks. The reduced design complexity allows for an earlier optimal design to be quickly reconfigured for changed specifications without requiring extensive re-design of the ADC analog core.

The second part of this chapter focuses on the development of an adaptive blocker-rejection wideband continuous-time sigma-delta ADC (CT ΣΔ ADC). An integrated blocker detector reconfigures the ADC architecture in real time to reject interference, which improves the selectivity and sensitivity of the receiver without increasing its dynamic-range requirements. To minimize power consumption, the ADC uses a built-in high-pass filter that performs blocker-level detection without

[1] Cho, T.: Low-power low-voltage analog-to-digital conversion techniques using pipelined architecures. PhD Thesis, University of California, Berkeley (1995)

utilizing any additional circuitry. The adaptive operation relaxes baseband channel-filtering requirements for a WiMAX receiver. The proposed ADC has been integrated in a 130 nm CMOS process occupying a silicon area of 1.5 × 0.9 mm^2. The CT ΣΔ ADC achieves 70 dB of DR, 65 dB of peak signal to noise-plus-distortion ratio (SNDR), and 68 dB of peak signal to noise ratio (SNR) over a 10 MHz signal bandwidth, consuming 18 mW from a 1.2 V supply. The ADC reconfigures the loop-filter topology within 50 μs without any transient impact on bit-error rate. In the blocker-suppression mode, the ADC can withstand 30 dBc blocker at the adjacent channel, achieving −22 dB error-vector magnitude with a 24 Mbps 16-QAM signal.

2.2 Power Scalable Pipeline ADC Design

Power consumption of a pipeline ADC is typically optimized for a specified SNDR requirement at a given sampling rate and input frequency range. When the input frequency (f_{in}) range increases, increasing sampling rate to meet Nyquist conditions, the ADC analog core has to be re-designed for more stringent settling time requirements. Scalable and reconfigurable designs aim to reuse the ADC core over a range of sampling and input frequencies by scaling power consumption. The relationship between the pipeline ADC power, sampling rate and resolution are discussed in the following sections and common design techniques for power scalable ADCs is presented.

2.2.1 MDAC Power and Performance

The architecture of a pipeline stage and its transfer function are shown in Fig. 2.1. For the MDAC to achieve N_i bit accuracy, the output voltage must settle with a gain error of less than $1/2^{Ni}$. The settling characteristic of the SC amplifier at sample instant k is given by

$$Vo[k] = \left(1 + \frac{C_S}{C_F}\right) \cdot \left(1 - e^{-t/n\tau}\right) \cdot \left(\frac{\beta}{\beta + 1/A_{OL}}\right) \tag{2.1}$$

where n is the number of time constants (τ) available for settling, β is the feedback factor of the amplifier, and A_{OL} is the open-loop DC gain of the OTA. The error in the first term from capacitor mismatch can be minimized by good layout practices. The errors in the second and third terms arise from finite settling time and finite open-loop DC gain of the OTA respectively. The error from finite open-loop DC gain can be minimized by designing for A_{OL} to be much greater than 2^{Ni}.

The large signal and small signal settling time of the amplifier can be related to the OTA characteristics as

$$t_{ls} = \frac{VFS}{SR} \tag{2.2}$$

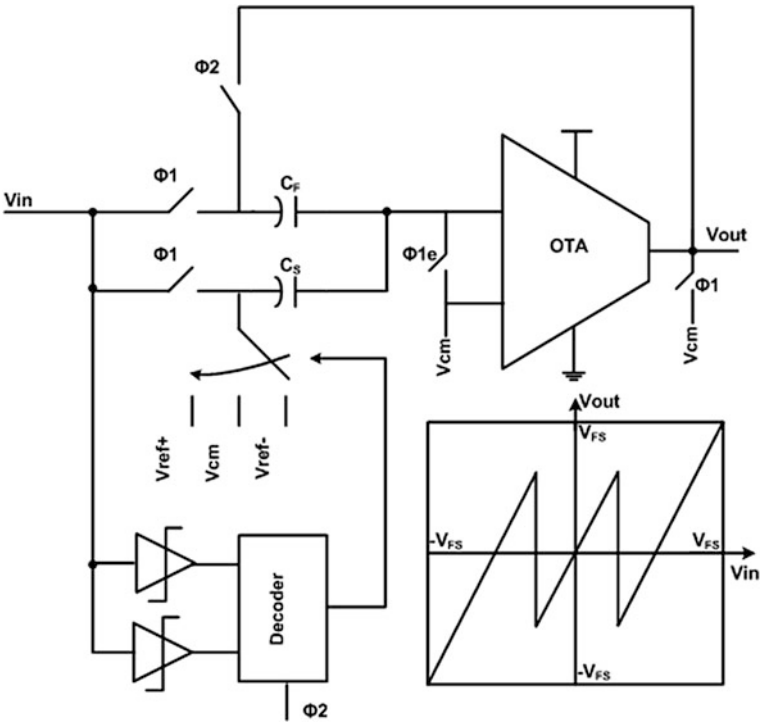

Fig. 2.1 Schematic of a typical 1.5 b/stage MDAC with transfer function

$$t_{ss} = n \cdot \tau = \frac{n}{\beta \cdot GBW} \tag{2.3}$$

where *SR* is the slew rate and *GBW* is the gain bandwidth product of the OTA. The value of n is determine by the resolution required of the stage as $n = N_i \bullet ln(2)$. This assumes a single pole frequency response for the OTA for simplicity. The worst case settling time (t_s) required to settle to N_i bit accuracy, which should be less than approximately one-half clock cycle ($\sim 1/2F_S$), is determined by the slew rate (*SR*) and gain-bandwidth product (*GBW*) of the OTA as[2]

$$t_s = \left(\frac{VFS}{SR}\right) + \left(\frac{n}{\beta \cdot GBW}\right) \leq \frac{1}{2 \cdot F_S} \tag{2.4}$$

For a single stage OTA, the *SR* and *GBW* are functions of the load capacitance (C_L), and the tail current (I_t) and transconductance (gm) of the OTA's input pair. Substituting these relationships in Eq. 2.4, the following relation between the sampling rate and OTA power is derived

[2] Lotfi, R., Taherzadeh–Sani M., Azizi, M.Y., Shoaei, O.: A low-power design methodology for high-resolution pipelined analog-to-digital converters. In: Proceedings of ISLPED, pp. 334–339 (2003)

$$\frac{1}{2 \cdot F_S} \approx \left(\frac{VFS}{I_t} + \frac{n}{\beta \cdot gm} \right) \cdot C_L \tag{2.5}$$

It can be seen from Eq. 2.5 that given a sampling rate and resolution requirement, the OTA current is a critical design parameter. The load capacitance is determined by thermal noise consideration and is typically fixed parameter during the design of the MDAC stage OTA. If sampling rate is increased, the OTA current must also be increased to maintain constant performance. This fact is utilized in the design and operation of power scalable pipeline ADCs.

The optimal OTA tail current ($I_{t,opt}$) of a pipeline MDAC stage for a given capacitive load, resolution and sampling rate can be shown to be

$$I_{t,opt} = \left(\frac{C_L}{2} \right) \cdot \left(VFS + \frac{n \cdot V_{dsat}}{2 \cdot \beta} \right) \cdot F_S \tag{2.6}$$

where V_{dsat} is the difference between the input pair transistor's V_{GS} and threshold voltage (V_{th}). This equation is derived assuming that a constant V_{dsat} is maintained. Equation (2.6) shows that the optimal power of the MDAC stage is approximately linearly related to the sampling rate. However, if an existing design is to be operated at a higher sampling rate and the power is increased to maintain performance, the relationship is no longer linear. While slew rate scales linearly with the tail current, the gain-bandwidth product of the OTA is proportional to the square-root of the current. Thus, the MDAC stage power does not scale linearly with sampling rate and for situations where small signal settling is much larger than large signal settling, the power is approximately proportional to the square of the sampling rate.

2.2.2 Bias Current Scaling and Switched-Opamp Scaling

Power scalability in pipeline ADCs is typically implemented by scaling the bias currents of the OTAs as shown in Fig. 2.2.[3,4,5,6] Scaling OTA bias currents results in large variations of the transistors' DC bias conditions. At the lower end of the sampling rate range the bias transistors are in moderate or weak inversion. In the sub-threshold region transistor mismatch increases thereby increasing the OTA

[3] Hernes, B. et al.: A 1.2 V 220MS/s 10b pipeline ADC implemented in 0.13/spl mu/m digital CMOS. In: IEEE International Solid-State Circuits Conference Digital Technology Papers, vol. 1 pp. 256–526 (2004)

[4] Andersen, T.N. et al.: A 97mW 110MS/s 12b pipeline ADC implemented in 0.18 mm digital CMOS. In: Proceedings of European Solid-State Circuits Conference, pp. 247–250 (2004)

[5] Gulati, K., Lee, H.-S.: A low-power reconfigurable analog-to-digital converter. IEEE J. Solid-State Circuits **36**(12), 2446–2455 (2001)

[6] Ahmed, I., Johns, D.A.: A 50-MS/s (35 mW) to 1-kS/s (15 uW) power scaleable 10-bit pipelined ADC using rapid power-on opamps and minimal bias current variation. IEEE J. Solid-State Circuits **40**(12), 2446–2455 (2005)

offset. The transistors are also more susceptible to transient disturbances due to the exponential dependence of the drain current on the gate voltage. Increasing the bias currents with sampling rate also increases the overdrive voltage (V_{dsat}) of the transistors. This reduces the maximum voltage swing at the OTA inputs and outputs, and adversely impacts the dynamic range of the ADC during low power supply operation. Also, in order to maintain a high OTA DC open-loop gain, transistors must be biased in saturation in order to maintain their high output resistance. Bias current scaling significantly increases the design complexity since the OTA operation must be verified not only over temperature and process variations, but also, over large bias current variations.

In Eq. (2.6) it was shown that the optimal OTA current required for a given accuracy and load capacitance scales linearly with sampling rate. However, this assumes that the V_{dsat} is maintained constant. Since a constant V_{dsat} cannot be maintained using bias current scaling, the scaled power consumption exceeds the optimal power required at a given sampling rate as shown in Fig. 2.3. Thus, bias current scaling requires that the OTA must be designed for the highest sampling rate with power scaled down for lower sampling rates. This leaves the design more susceptible to transistors operating in sub-threshold at lower sampling rates.

Powering off OTAs in the sampling phase, coupled with bias current scaling, provides appreciable power savings (see footnote 6). The OTAs are powered on only during the hold phase thereby halving the average power consumption. This technique is also utilized in switched-opamp ADCs.[7,8,9] However, a portion of the hold phase is required to power on the OTA, which reduces the time available for output voltage settling. The power-on interval occupies a significant portion of the hold phase at high sampling rates, requiring an increase in the OTA power to compensate for the loss of available settling time. The design challenges of a rapid power-on OTA limit this approach to low sampling rate applications.

The following chapter will present a power scalable pipeline ADC technique that enables or disables OTAs connected in parallel to scale the settling response of the MDAC and S/H amplifiers in order to achieve constant SNDR performance over a range of sampling rates. The proposed technique facilitates optimal power consumption over the entire sampling rate range and reduces design complexity by maintaining constant DC bias conditions in the scaled analog blocks. The reduced design complexity allows for an earlier optimal design to be quickly reconfigured for changed specifications without requiring extensive re-design of the ADC analog core.

---

[7] Kim, H-C., Jeong, D-K., Kim, W.: A partially switched-opamp technique for high-speed low-power pipelined analog-to-digital converters. IEEE Trans. Circuits Syst. I: Regul. Pap. **53**(4), 795–801 (2006)

[8] Waltari, M. Halonen, K.A.I.: 1-V 9-bit pipelined switched-opamp ADC. IEEE J. Solid-State Circuits **36**(1), 129–134 (2001)

[9] Wu, PY., Cheung, V.S-L., Luong, H.C.: A 1-V 100-MS/s 8-bit CMOS switched-opamp pipelined ADC using loading-free architecture. IEEE J. Solid-State Circuits **42**(4), 730–738 (2007)

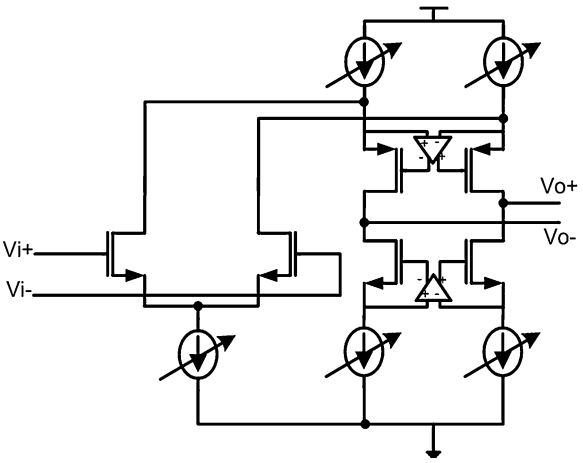

Fig. 2.2 Folded-cascode OTA with variable bias for scalable pipeline ADC

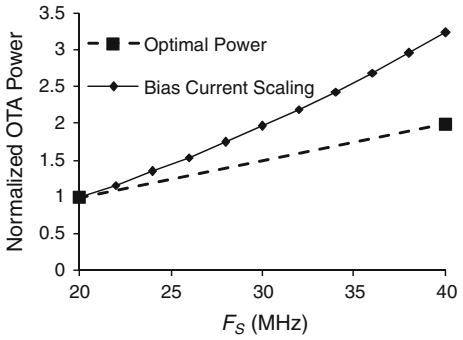

Fig. 2.3 Normalized OTA power with bias current scaling compared to optimal power over a range of sampling rates

2.3 Parallel OTA Scaling Approach

This section presents a design technique for scalable pipeline ADCs that enables or disables OTAs connected in parallel to scale the settling response of the MDAC and S/H amplifiers in order to achieve constant SNDR performance over a range of sampling rates. The proposed technique facilitates linear and optimal power consumption over the entire sampling rate range and reduces design complexity by maintaining constant DC bias conditions in the scaled analog blocks. The reduced design complexity allows for an earlier optimal design to be quickly reconfigured for changed specifications without requiring extensive re-design of the ADC analog core. In,[10] programmability in Gm-C filters was achieved by switching in

[10] Pavan, S., Tsividis, Y.P., Nagaraj, K.: Widely programmable High-frequency continuous-time filters in digital CMOS technology. IEEE J. Solid-State Circuits **35**(4), 503–511 (2000)

transconductances (Gm) in parallel. Since amplifiers in switched capacitor stages are essentially Gm stages driving capacitor loads, a similar switched transconductance technique can also be used in pipeline ADCs by enabling or disabling individual OTAs in parallel.

2.3.1 Description of Parallel OTA Scaling

In Eq. (2.4), it can be seen that the settling performance of switched capacitor amplifiers can be scaled by varying the OTA bias current. Increasing the OTA bias current increases the current available for slewing and increases the small signal settling performance by increasing the transconductance of the OTA. The improvement in settling performance is obtained at the cost of disturbing the DC bias conditions of OTA transistors. In the proposed parallel OTA scaling technique, scalable settling performance is obtained by enabling or disabling OTAs connected in parallel in the S/H and MDAC stages of the pipeline ADC. A scalable MDAC stage implemented using two OTAs connected in parallel is shown in Fig. 2.4. Since the OTAs only share the input and output nodes, the DC bias conditions of the internal nodes of each individual OTA are unperturbed.

2.3.2 Settling Analysis of Parallel OTA Scaling

The output current signal of the identical individual OTAs, each with transconductance gm_i, are summed at the shared output, scaling the effective transconductance (Gm_{eq}) of the stage as

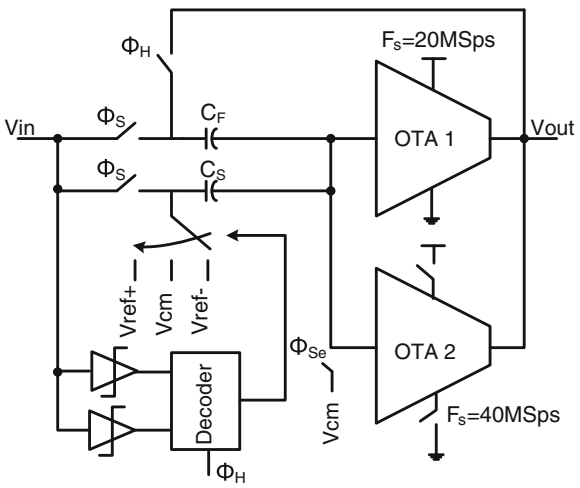

Fig. 2.4 Implementation of a scalable MDAC stage using the parallel OTA scaling technique

$$i_{out} = \sum_{i=1}^{k} i_{out,i} = \sum_{i=1}^{k} gm_i \cdot v_{in} \tag{2.7}$$

$$Gm_{eq} = \frac{i_{out}}{v_{in}} = \sum_{i=1}^{k} gm_i = k \cdot gm_i \tag{2.8}$$

Thus, Gm_{eq} is an integer multiple of gm_i and can be varied in discrete steps by enabling or disabling parallel OTAs. Since the GBW is proportional to the transconductance, the GBW of the equivalent OTA (GBW_{eq}) is now proportional to the number of parallel OTAs as

$$GBW_{eq} = \frac{Gm_{eq}}{C_L} = k \cdot \frac{gm_i}{C_L} = k \cdot GBW_i \tag{2.9}$$

where GBW_i is the gain bandwidth product of an individual OTA.

When the input voltage signal is stepped causing the OTAs to slew, each individual OTA contributes a current $I_{t,i}$ to slew the output voltage. Thus, the slew rate of the equivalent OTA (SR_{eq}) also increases linearly with the number of enabled parallel OTAs as

$$SR_{eq} = \frac{\sum_{i=1}^{k} I_{t,i}}{C_L} = k \cdot \frac{I_{t,i}}{C_L} = k \cdot SR_i \tag{2.10}$$

where SR_i is the slew rate of an individual OTA. Thus, the effect of enabling parallel OTAs on the settling time response of the switched capacitor amplifier is

$$t_s = \frac{1}{k} \cdot \left(\frac{VFS}{I_{t,i}} + \frac{N_i \cdot \ln(2)}{\beta \cdot gm} \right) \cdot C_L \leq \frac{1}{2 \cdot F_S} \tag{2.11}$$

The effect of parallel OTA scaling on the output settling response is illustrated in Fig. 2.5. From Eq. (2.11), it can be seen that the settling time response of the switched capacitor amplifier is now inversely proportional to the number of enabled parallel OTAs. Thus, the total power of the switched capacitor stage scales linearly with sampling rate. This linear relationship between power and sampling rate allows for optimal power consumption to be achieved over the entire sampling rate range.

The optimal OTA tail current ($I_{t,opt}$) of an MDAC stage at a fixed sampling rate, expressed by Eq. 3.6, assumes a fixed V_{dsat} for the input differential pair of the OTA, determined by matching and input voltage swing considerations. In the parallel OTA scaling technique, since the DC bias conditions of individual OTAs are unchanged, a constant V_{dsat} is maintained over the sampling rate range. Thus, if the individual OTAs are designed for optimal power consumption at one sampling rate, the scaled power consumption will track the optimal power consumption over the sampling rate range.

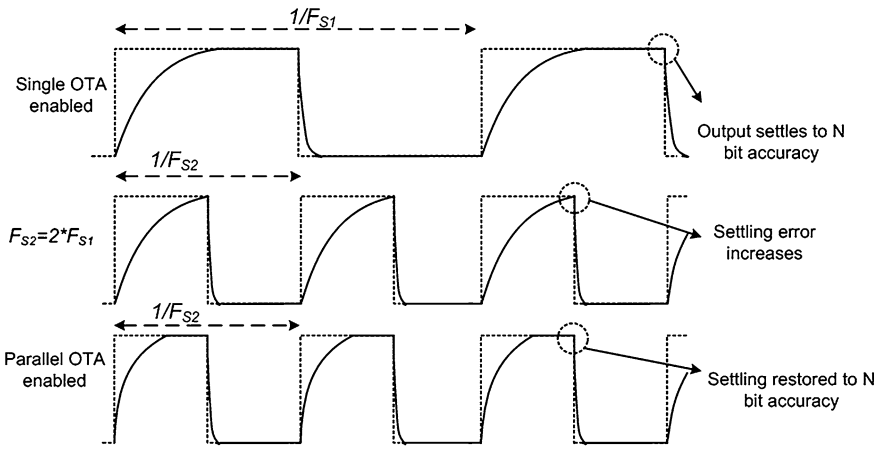

Fig. 2.5 Illustration of the effect of parallel OTA scaling on the output settling response

2.3.3 Parallel OTA Scaling Design Considerations

When the OTAs are connected in parallel, the effective output resistance is reduced. Since this decrease is accompanied by an increase in the effective transconductance, the DC open-loop gain of the OTAs in parallel is equal to open-loop gain of an individual OTA. This is also a result of the internal DC bias conditions remaining unchanged.

The previous analysis of the MDAC settling response assumes that the individual OTAs exhibit single-pole response. This is approximately true if the non-dominant pole of the OTA is located at a much higher frequency than the dominant pole. When two OTAs are connected in parallel the dominant pole frequency increases by a factor of 2. The non-dominant pole remains unchanged since the bias conditions of the internal nodes that are responsible for the non-dominant pole are unchanged. This results in a reduction of the phase margin. The frequency of the non-dominant pole limits the number of parallel OTAs that can be enabled before the phase margin is insufficient for stable operation.

The addition of OTAs in parallel also increases the total parasitic capacitance at the input (C_p). This results in a reduction of the feedback factor, which increases the settling time response of the MDAC. The load capacitance is also increased by the parasitic capacitance. The effect on the feedback factor and C_L will be minimized if the MDAC capacitors are much larger than the input parasitic capacitances of the OTA.

2.3.4 Scalable Pipeline ADC Implementation

The ADC core comprises a S/H stage, ten 1.5 b stages, and a 2 b flash ADC as shown in Fig. 2.1. The design process targeted a sampling frequency range between 20 and 40 MS/s. The reference voltages required for the analog to digital conversion is generated on-chip by a reference amplifier driving a resistor string. The ADC draws signal dependent current from the reference amplifier, which has finite settling time, leading to significant degradation of SNDR if the output impedance increases in the signal bandwidth. In order to minimize power consumption and maintain a low output resistance over the signal bandwidth, a reference amplifier with low DC output resistance and low bandwidth driving a large off-chip load capacitor is used. The large load capacitance ensures low output impedance at frequencies above the bandwidth of the reference amplifier.[11]

A clock buffer is used to buffer the off-chip clock and the non-overlapping clock signals are generated on-chip. In order to reduce the power consumption of the clock circuitry and layout complexity, 3 non-overlapping clock generators, each driving 4 stages (including S/H and Flash ADC), were used. This reduced the load capacitance driven by each clock generator and allowed for lower power dissipation in the clock circuitry. The RSD algorithm to correct for comparator offsets was implemented off-chip allowing for verification of individual stages during testing. Since comparator offsets as large as one-fourth the single-ended full-scale voltage are corrected for, dynamic comparators, without preamplifiers, are used in order to minimize power consumption.[12] The schematic of the dynamic comparator is shown in Fig. 2.6. Bootstrapped switches are used in the signal path of the S/H stage to minimize distortion caused by the dependence of the transistors' on-resistance on the gate-to-source and gate-to-drain voltages.

2.3.5 S/H and MDAC Amplifiers

The schematic of the scalable OTA used in the MDAC and S/H stages is shown in Fig. 2.7. The two individual OTAs are implemented in the folded cascode topology. Gain-boosting is used only for OTAs in the S/H and first 4 stages in order to achieve sufficient gain for 12 b resolution. The individual OTAs are each designed for operation at 20 MS/s and only one OTA is enabled in each stage for 20 MS/s operation.

---

[11] Maulik, P.C. et al.: A 16-Bit 250-kHz delta sigma-modulator and decimation filter. IEEE J. Solid-State Circuits **35**(4), 458–467 (2000)

[12] McCarroll, B.J., Sodini, C.G., Lee, H-S.: A high-speed CMOS comparator for use in an ADC. IEEE J. Solid-State Circuits **23**(1), 159–165 (1988)

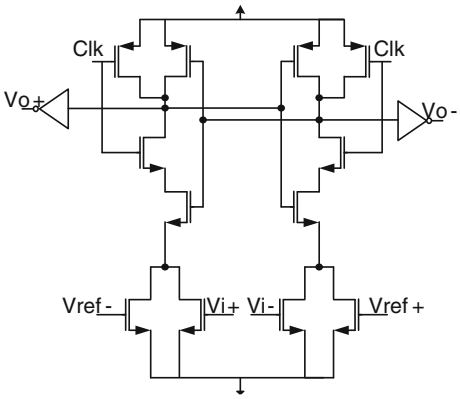

Fig. 2.6 Schematic of dynamic comparator

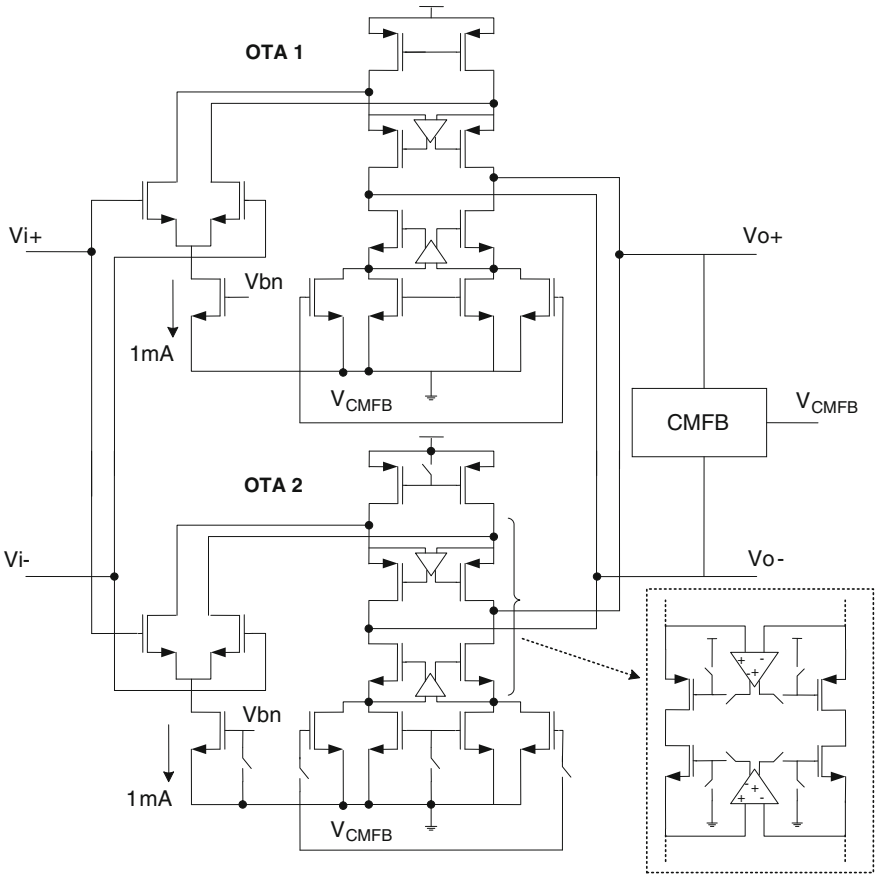

Fig. 2.7 Schematic of reconfigurable OTA in first MDAC stage which is implemented with two gain-boosted folded cascode OTAs for operation at 20 and 40 MS/s

The disabling switches, shown in Fig. 2.5, ensure that the unutilized OTA is turned off and exhibits a high output impedance so as not to affect the operation of the enabled OTA. A low or comparable output resistance in the disabled state will adversely impact the DC open-loop gain of the enabled OTA. Since the switches drive only transistor gates and do not pass any analog signals, low on-resistance is not required and minimum sized transistors are sufficient. A single switched capacitor common-mode feedback circuit is sufficient to maintain the output common-mode since the OTAs' outputs are shared.

When both parallel OTAs are enabled for operation at 40 MS/s, the GBW_{eq} and SR_{eq} are expected to be doubled. However, due to the non-dominant pole and the increase in the load capacitance from increased parasitic capacitance at the OTA input, a slightly lower than expected increase is achieved. The simulated open-loop frequency of the individual OTA in the first stage is compared to the response with two OTAs in parallel in Fig. 2.8. It can be seen that enabling an OTA in parallel increases GBW_{eq} by a factor of 1.97. The effect of reducing the dominant and non-dominant poles' separation is largely responsible for the <2 increase. The increased OTA input parasitic capacitance has a less significant effect since the sampling capacitors are comparatively larger. The simulated step response of an

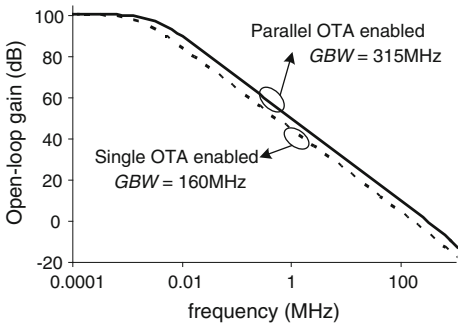

Fig. 2.8 Simulated scaling of open-loop frequency response of scalable OTA from first MDAC stage

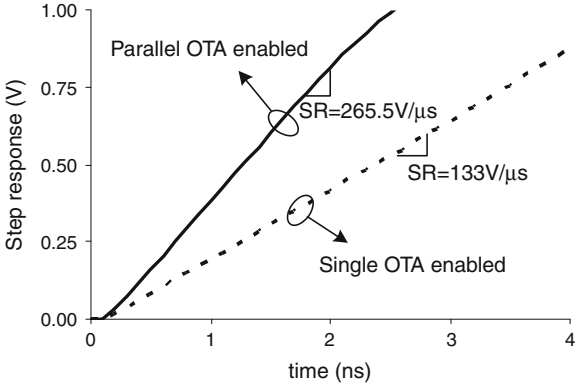

Fig. 2.9 Simulated scaling of step response of scalable OTA from first MDAC stage

individual OTA is compared to the response of two parallel OTAs in Fig. 2.9. The slew rate is increased by a factor of 1.996 when the parallel OTA is enabled. Adequate margin was incorporated in the final OTA design specifications to account for the effect of the non-dominant pole and process variation.

2.4 Characterization Results for the Power Scalable ADC

The proposed ADC was fabricated in a 1.8 V 0.18 μm CMOS process and occupies a die area of 1.9 mm^2. The die micrograph is shown in Fig. 2.10. For test purposes, two additional OTAs were added in parallel in the S/H and MDAC stages for a total of four OTAs in each stage. The differential full-scale voltage of the ADC is 1.2 V_{pp}. At $F_S = 20$ MS/s, only one OTA is enabled in each stage and the analog blocks consume 36 mW. When F_S is increased to 40 MS/s, a second parallel OTA is enabled in each stage and the measured analog power consumption increases to approximately 72 mW, which is twice the power consumed at 20 MS/s. Thus, the ADC achieves linear power scaling with respect to sampling rate. If a continuous range of power scaling is desired, bias current scaling can be used at each OTA over a ±10 MS/s range. Since the bias current scaling is utilized only over a limited frequency range, the resulting DC bias variation is minimized by the parallel OTA technique.

Figure 2.11 shows the measurement results for the SNR and SNDR of the ADC versus the input signal frequency at 20 and 40 MS/s sampling rates. For each F_S, it can be seen that the SNR and SNDR are fairly constant over the input frequency within the Nyquist range. The SNR for an input signal of 1 MHz sampled at 20 MS/s is 66.6 dB and the SNDR is 66.2 dB (ENOB = 10.7 b). At a sampling rate of 40 MS/s, the SNR for a 1 MHz input signal is 62.2 dB and SNDR is 62 dB (ENOB = 10 b). The reduction of SNR from 20 to 40 MS/s operation is due to the

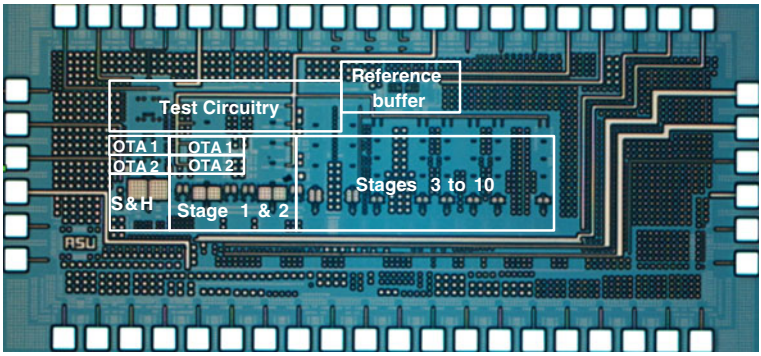

Fig. 2.10 Die micrograph of the power scalable ADC

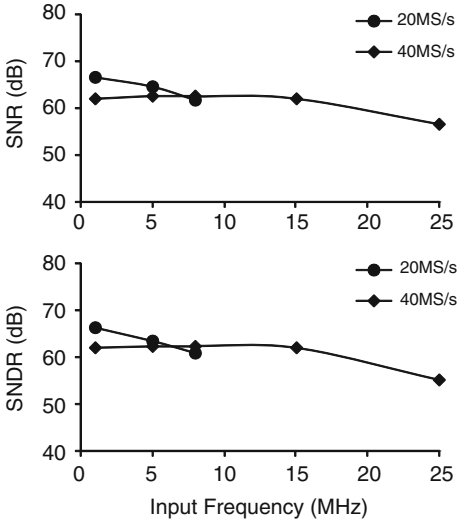

Fig. 2.11 Measured SNR and SNDR versus input frequency for $F_S = 20$ and 40 MS/s

increased supply and substrate noise from the on-chip digital logic and I/O circuitry coupling to the analog circuitry. The Figure of Merit (FOM), expressed in Eq. (2.12), at 20 MS/s is calculated to be 1.1 pJ and at 40 MS/s is calculated to be 1.75 pJ.

$$FOM = \frac{Power}{F_S \cdot 2^{ENOB}} \tag{2.12}$$

Since the power scales linearly with the sampling rate, the increase in the FOM at 40 MS/s operation can be attributed largely to the reduction in the effective resolution caused by increased switching noise from the digital logic and I/O circuitry. The measured power versus sampling rate is plotted in Fig. 2.12. The measured SNR and SNDR at 60 MS/s (3 parallel OTAs enabled) and 80 MS/s (4 parallel OTAs enabled) are plotted versus input frequency in Fig. 2.13. A SNDR of 57.4 dB (ENOB = 9.24 b) is obtained at 60 MS/s and the SNDR

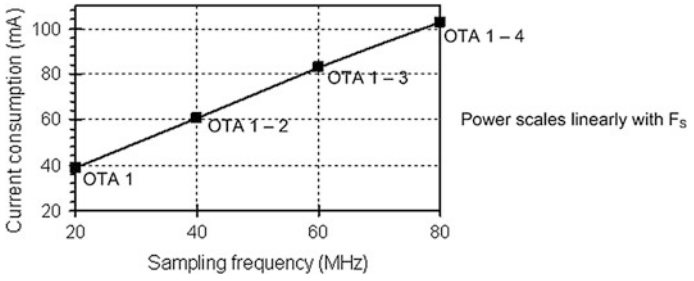

Fig. 2.12 Measured ADC power versus sampling rate

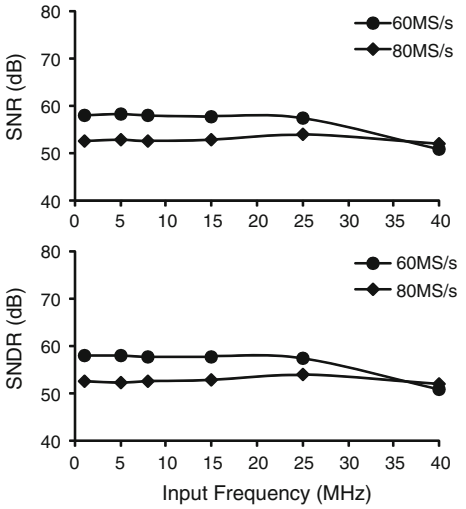

Fig. 2.13 Measured SNR and SNDR versus input frequency for $F_S = 20$ and 40 MS/s

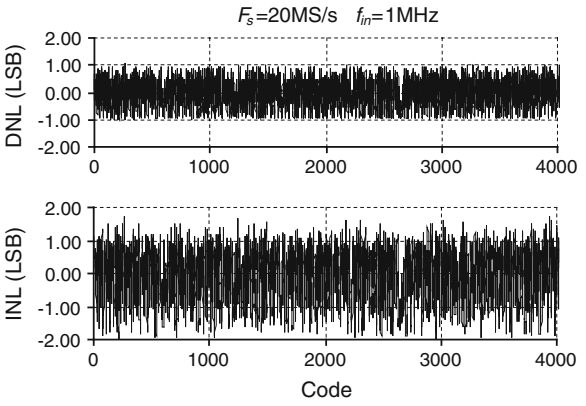

Fig. 2.14 Measured DNL and INL

Table 2.1 Measurement summary for the Scalable Pipeline ADC

	20 MS/s	40 MS/s
Technology	0.18 μm CMOS	
Power supply	1.8 V	
Resolution	12 b	
Full scale input	1.2 V_{pp}	
Area	1.9 mm^2	
Analog power	36 mW	72 mW
SNR (f_{in} = 1 MHz)	66.6 dB	62.2 dB
SNDR (f_{in} = 1 MHz)	66.2 dB	62 dB
ENOB (f_{in} = 1 MHz)	10.7 b	10 b
DNL, INL	±1 LSB, 1.7/−1.97 LSB	

further reduces to 52.2 dB (ENOB = 8.4 b) at 80 MS/s operation. The additions of the third and fourth parallel OTAs demonstrate the limitation on the number of possible parallel OTAs before degradation in settling performance. The individual OTAs were designed for operation between 20 MS/s and 40 MS/s. The location of the non-dominant pole was determined accordingly. Enabling more than two OTAs in parallel reduces the phase margin significantly affecting the settling performance. The DNL and INL, measured using the code density test for $F_S = 20$ MS/s and $f_{in} = 1$ MHz, are plotted in Fig. 2.14. The measurement results are summarized in Table 2.1.

2.5 Review of Recent ΣΔ ADCs with Adaptive Bandwidth and Interferer Filtering

In the second section of this chapter we will analyze techniques to adjust the loop filter bandwidth, dynamic range and out of band interferer rejection performance of continuous time ΣΔ ADCs. Highly digital direct-conversion receivers can reduce system complexity by removing analog automatic-gain control and DC-offset cancellation loops at the expense of increased DR requirements on the ADC. However, if the ADC DR specification is too high, the silicon area and power consumption of the receiver employing this approach will be larger than that of conventional direct conversion (DC) receivers.[13] In order to resolve this issue, several ADC design techniques have been proposed. In this subsection, these techniques are described briefly and their advantages and disadvantages are discussed.

2.5.1 Reconfigurable Discrete-Time Multi-Stage Noise Shaped ΣΔ ADC

During normal operation of the receiver, both the signal power and the interferer profile can change, and thus stringent channel-select filtering or ADC performance is not always necessary. A flexible and reconfigurable architecture targetted at the RF front-end and the ADC would allow for the optimization of power consumption and SNR performance of the receiver depending on operating conditions. Recently, a reconfigurable discrete-time (DT) multi-stage noise-shaped (MASH) ΣΔ ADC has been proposed for power-adaptive operation under blocker

[13] Garrity, D. et al.: A single analog-to-digital converter that converts two separate channels (I and Q) in a broadband radio receiver. IEEE J. Solid-State Circuits **43**(6), 1458–1469 (2008)

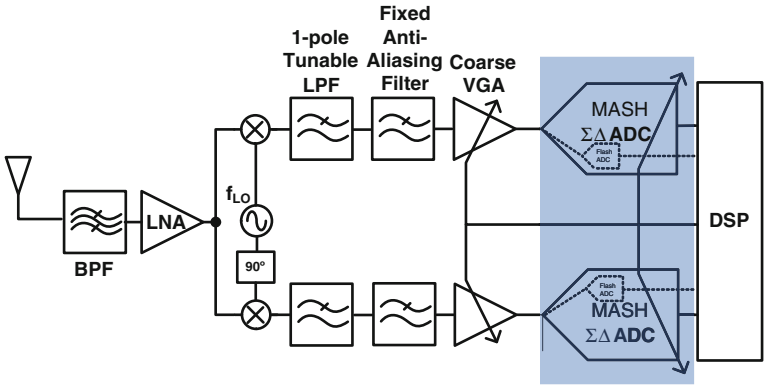

Fig. 2.15 Receiver architecture with a reconfigurable DT MASH ΣΔ ADC (see footnote 15)

condition.[14] Figure 2.15 shows a receiver architecture with the MASH ΣΔ ADC. It has a single-pole low-pass filter, fixed anti-aliasing filter, and coarse variable-gain amplifier in front of a MASH ΣΔ ADC. The system reconfigures the order of the ADC based on desired channel and interferer levels at the ADC input. The power-level estimation is performed by a 5-bit flash ADC at the modulator input and digital-signal processing (DSP). Latency in DSP processing may result in failure to meet the standard specifications or system instability when a high blocker is present at the ADC and a VGA control loop is required.

2.5.2 CT ΣΔ ADC with Increased Blocker Suppression

CT ΣΔ ADCs are widely used for mobile wireless systems because they can achieve high DR with low power consumption. In addition, thanks to their implicit anti-aliasing filtering and channel select-filtering performance by an STF, the requirements for analog baseband filtering or ADC DR can be relaxed.

Figure 2.16 shows commonly used CT ΣΔ ADC architectures. To increase immunity to interferers, a CT ΣΔ ADC with a chain of integrator with distributed feedback (CIFB) architecture can be used since its STF has a faster roll-off in out-of-channel frequencies in comparison to feed-forward loop architectures. Since each integrator output has a significant amount of input signal swing, lower integrator coefficients are necessary to avoid signal clipping and should be implemented with larger capacitance increasing silicon area. In addition, a reduced integrator coefficient in the first stage results in increased input-referred thermal noise and non-linearity caused by the following stages. Therefore, power

[14] Malla, P. et al.: A 28 mW spectrum-sensing reconfigurable 20 MHz 72 dB-SNR 70 dB-SNDR DT ADC for 802.11n/WiMAX receivers. In: IEEE ISSCC Digital Technical Papers, pp. 496–497 (2008)

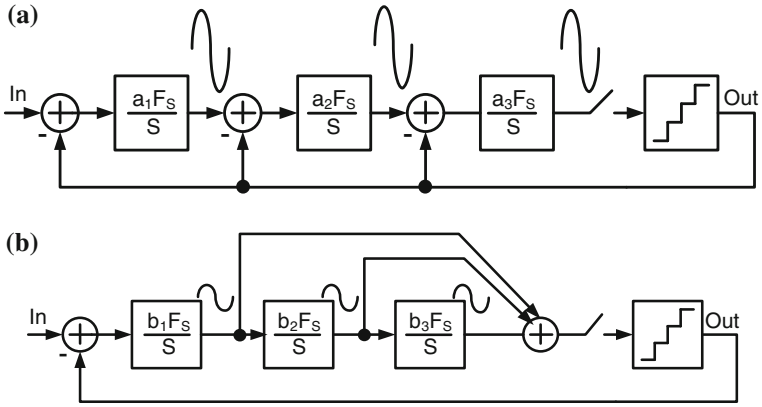

Fig. 2.16 **a** Chain of integrator with distributed feedback **b** chain of integrators with feedforward summation architecture

consumption of these stages should be increased to reduce their non-ideal contributions.

For low-power implementation, a chain of feed-forward summation (CIFF) topology can be used. Since each integrator output contains only a quantization-error signal, its output swing is relatively small compared to the feedback topology. Therefore, large integrator coefficients can be used and noise and distortion contribution of the second and following stage integrators caused by reduced bias current can be tolerated. However, since a CIFF architecture exhibits gain peaking in its STF at high frequencies, this architecture will be overloaded or unstable when high-power interferers are applied to the modulator input.

To maintain the fast roll-off while reducing power consumption and silicon area, a CT ΣΔ ADC with feed-forward topology can be modified as shown in Fig. 2.17.[15] If the two filters are complementary and they satisfy the following condition:

$$H_{LPF}(s) \cdot H_{HPF}(s) = 1,$$

the STF can be modified without changing the noise transfer function (NTF), which is given by

$$STF(s) = H_{LPF}(s)\frac{LF(s)}{1 + LF(s)}, \tag{2.13}$$

where $LF(s)$ is the loop filter of the modulator. These additional filters scarcely increase the total power consumption and area for narrow-band applications. However, if this ADC were used for wideband and high-speed applications such as

[15] Philips, K. et al.: A continuous-time ΣΔ ADC with increased immunity to interferers. IEEE J. Solid-State Circuits **39**(12), 2170–2178 (2004)

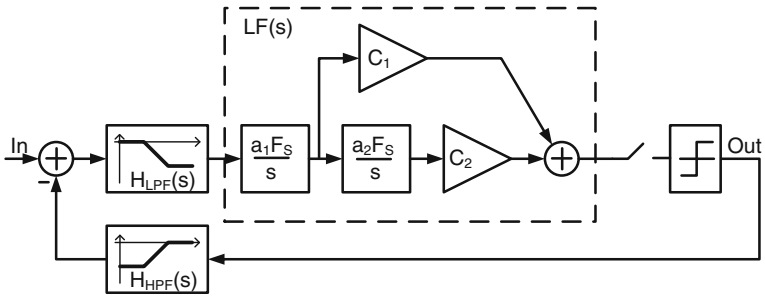

Fig. 2.17 ΣΔ ADC with increased interferer immunity (see footnote 15)

WiMAX, the required matching of the two filters would be very stringent. This requirement would increase the complexity of this architecture.

2.5.3 Direct Feedforward Compensation Technique in ΣΔ ADCs

Figure 2.18a shows a conventional single-loop ΣΔ ADC architecture. When the input signal *In* is stationary, the quantizer output *Out* is also stationary. Therefore, the error signal *E* entering the loop filter has small amplitude. If the input *In* changes abruptly, the quantizer output *Out* is still stationary during the excess loop delay. Consequently, the error signal *E* is fed to the loop filter and the modulator will be overloaded. This scenario frequently occurs in automobile tuner systems, which experience fading of received signals due to dynamically changing interferers. To avoid this problem, a direct feed-forward compensation technique in a ΣΔ DC has been proposed, as shown in Fig. 2.18b.[16] If a non-delayed feedback technique is employed, the quantizer output *Out* is not delayed with respect to the input *In*. Therefore, the error signal *E* can remain of small amplitude. However, this topology still requires channel-select filters to protect the ADC from overload when a high interferer is accompanied by the desired channel signal.

2.5.4 Comparing Advantages and Disadvantages of the Recent ΣΔ ADCs

In the previous subsection, recent ΣΔ ADC design techniques are briefly reviewed. The reconfigurable DT MASH ΣΔ ADC can optimize power consumption and SNR performance based on the desired channel signal and blocker-power levels.

[16] Yamamoto, T., Kasahara, M., Matsuura, T.: A 63 mW 112/94 dB DR IF bandpass ΣΔ modulator with direct feed-forward compensation and double sampling. IEEE J. Solid-State Circuits **43**(8), 1783–1794 (2008)

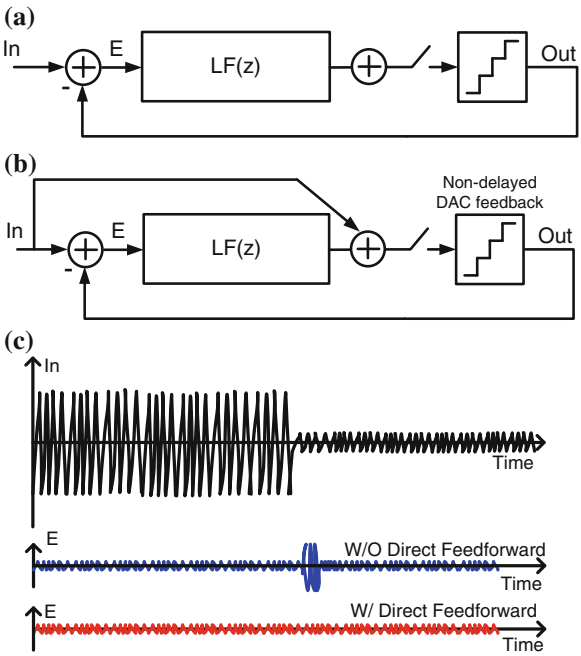

Fig. 2.18 **a** Conventional S ADC, **b** SD ADC with direct feedforward compensation, and **c** illustrations of transient waveforms with sudden input changing (see footnote 16)

The power-level estimation is performed by a 5-bit flash ADC at the modulator input with digital-signal processing, and thus latency in DSP processing can result in either failure to meet the standard specifications or system instability when a high blocker is present at the ADC. Moreover, this approach requires a VGA control loop as well as an anti-aliasing filter due to DT implementation.

The second approach, which employs addition filters in the feed-forward and feedback path, can achieve strong blocker-suppression strength with low power consumption. However, the required matching condition is very stringent, increasing the complexity of this architecture, if this ADC is used for wideband and high-speed applications. In addition, the filter in the feedback path would increase excess loop delay and thus reduce the stability of the modulator.

The last approach uses a direct feed-forward path to protect the ADC from overloading and instability when high blockers are applied or input signals change abruptly. However, this topology requires channel-select filters to protect the ADC from overload when a high interferer is accompanied by the desired channel signal.

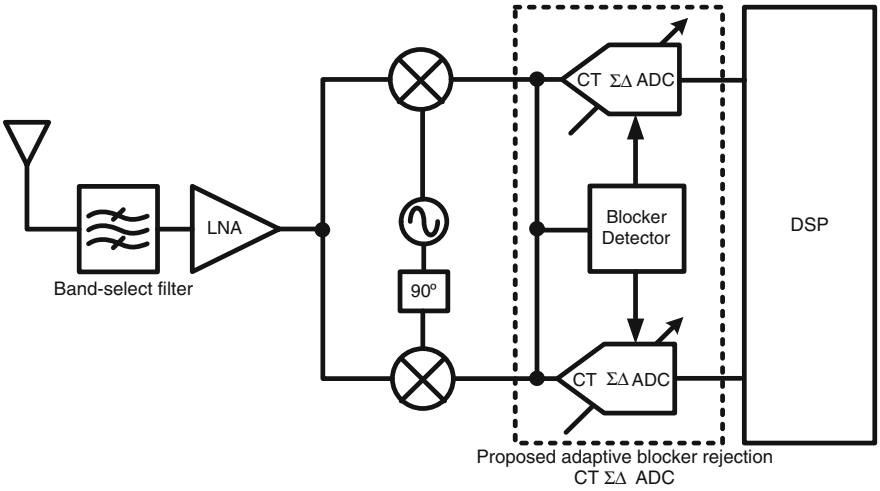

Fig. 2.19 Direct Conversion (DC) receiver architecture with the proposed adaptive blocker rejection CT ΣΔ ADC

2.6 Blocker-Adaptive ΣΔ ADC for Mobile WIMAX Applications

2.6.1 ADC Requirements

Figure 2.19 shows a DC receiver with the proposed SD ADC approach. By exploiting adaptive blocker rejection performance of the ADC, the receiver can improve system selectivity without an additional channel select filter and optimize ADC performance based on the blocker level. This section describes the procedure of defining DR and linearity requirements of the ADC and the same procedure presented in Chap. 1 is employed to define the specifications.

Before determining the ADC dynamic range (DR) requirement, the STF of the modulator should be defined because the DR specification strongly depends on the filtering performance of the STF. The reconfigurable modulator has two operation modes: One is normal mode and the other is blocker suppression mode. Details of the modes will be explained in the next section. The DR requirement is defined under condition of the worst case in which the desired input signal has the lowest power level and blockers have the strongest power level. In this condition, the modulator should be operated in the blocker suppression mode. If the STF has a second-order LPF characteristic with a corner frequency of 20 MHz, The STF can theoretically have 6 dB attenuation at the adjacent channel and 12 dB attenuation at the alternate channel frequency as illustrated in Fig. 2.20.

Figure 2.21 illustrates the link budget analysis to determine the ADC dynamic range requirements. With a 1.2 V maximum supply for 130 nm CMOS process,

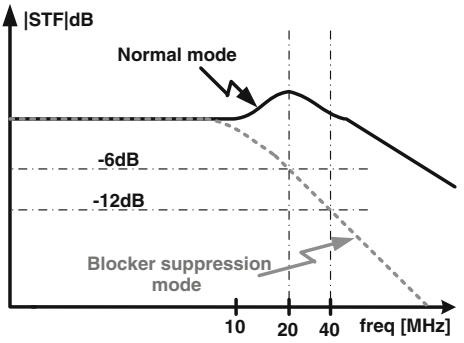

Fig. 2.20 Illustrated STFs of the normal and blocker suppression modes

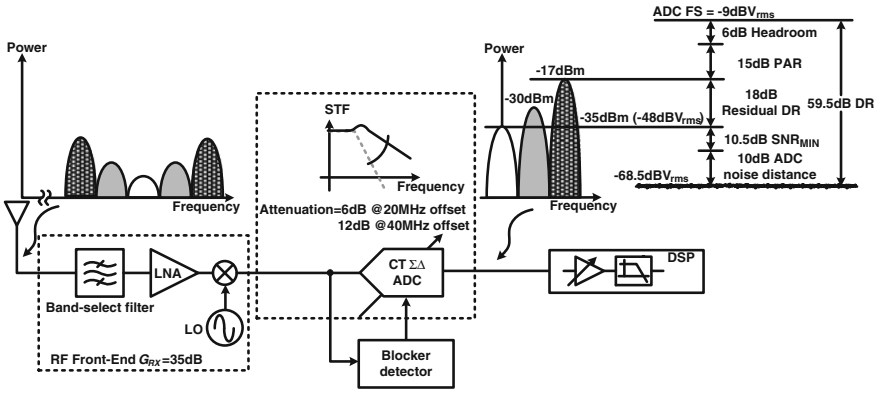

Fig. 2.21 ADC dynamic range requirement under blocking conditions

the full-scale input voltage (FS) of the ADC is set to 1 V_{pk-pk} differential. A 6 dB headroom margin is taken into account to cover DC offsets and transient signal variations. OFDM modulation used in the WiMAX standard has a typical PAR in the range of 12–17 dB depending on the number of sub-channels, and the ADC should account for this value to avoid signal compression or clipping. With a 35-dB RF front-end gain, G_{RX} provided by the band-select filter, LNA and mixer, the -70 dBm (-83 dBV_{rms}) desired channel at the antenna input can be amplified up to -48 dBV_{rms} which is 18 dB below the maximum signal level at the ADC input, defined as DR_{res}. To meet the bit error rate (BER) specifications, a minimum 10.5 dB SNR is required at the digital demodulator input. Moreover, a 10 dB additional noise margin is added to avoid ADC noise floor impacting the overall noise figure of the receiver. Therefore, the ADC noise floor is set at -68.5 dBV_{rms} while the required ADC DR is 59.5 dB.

To account for gain variability of the RF chain (i.e., high-Q LC loads), the ADC must be able to cope with the worst case of a higher than expected RF gain. Since

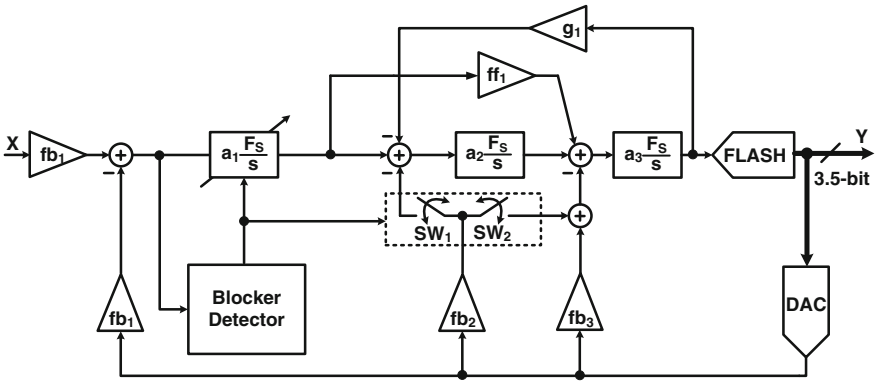

Fig. 2.22 Architecture of the proposed ΣΔ ADC

no filtering performance is also assumed for the RF front-end, a 10 dB additional margin is taken into account in the IIP3 requirement for the ADC. This yields the ADC IIP3 specification of 7.25 dBm.

2.6.2 Proposed ΣΔ ADC Architecture

Figure 22 shows the architecture of the proposed ΣΔ ADC, which consists of the reconfigurable loop filter and blocker detector. When blockers are weak or absent at the ADC input, the ADC operates in normal mode and switch SW_1 is open and SW_2 closed. In this mode the modulator shows a combination of feedforward and feedback stabilized loops.[17] By using the feedforward path ff_1, the first integrator's output swing can be reduced and thus its integrator coefficient can be increased. This results in reduction of overall power consumption since non-idealities of the second and third integrators due to reduced bias currents can be tolerated by the high coefficient of the first integrator. In addition, this architecture can achieve good anti-aliasing performance due to the feedback path. However, in the presence of strong blockers, the gain peaking in the STF at the adjacent channel frequency would produce a higher noise floor or lead to system instability.

In order to protect the ADC from overloading, the blocker detector reconfigures architecture by closing SW_1, opening SW_2 and reducing the first integrator coefficient a_1 by 50 %. The feedback path to the first integrator output removes the gain peaking, while the reduction of a_1 increases blocker suppression strength at the expense of reduced quantization noise shaping.

[17] Muñoz, F., Philips, K., Torralba, A.: A 4.7 mW 89.5 dB DR CT complex ADC with built-in LPF. In: IEEE ISSCC Digital Technical Papers, pp. 500–501 (2005)

2.6.3 Loop Filter and Feedback Path Design

In the normal operation mode, the loop filter $LF_{NOR}(s)$ and feedforward filter $FF_{NOR}(s)$ transfer functions are given by:

$$\begin{aligned} LF_{NOR}(s) &= \frac{a_3(fb_2+fb_3)F_S s^2 + a_1a_3fb_1ff_1F_S^2 s + a_1a_2a_3fb_1F_S^3}{s^3 + a_2a_3g_1F_S^2 s}, \\ FF_{NOR}(s) &= \frac{a_1a_3fb_1ff_1F_S^2 s + a_1a_2a_3fb_1F_S^3}{s^3 + a_2a_3g_1F_S^2 s}, \end{aligned} \tag{2.14}$$

where F_S is the sampling frequency. In the blocker suppression mode, the same transfer functions are calculated as

$$\begin{aligned} LF_{BLK}(s) &= \frac{a_3fb_3F_S s^2 + (0.5a_1a_3fb_1ff_1 + a_2a_3fb_2)F_S^2 s + 0.5a_1a_2a_3fb_1F_S^3}{s^3 + a_2a_3g_1F_S^2 s}, \\ FF_{BLK}(s) &= \frac{0.5\left(a_1a_3fb_1ff_1F_S^2 s + a_1a_2a_3fb_1F_S^3\right)}{s^3 + a_2a_3g_1F_S^2 s} \end{aligned} \tag{2.15}$$

The STF can be calculated by $FF(s)/1 + LF(s)$ and NTF can be calculated by $1/1 + LF(s)$. Figure 2.23 shows the simulated STF and NTF of the normal and blocker suppression modes. The signal bandwidth is set to 10 MHz and the sampling frequency is set to 250 MHz. The normal mode can achieve better SNDR performance with better noise shaping than the blocker suppression mode when blockers are weak, but it would show lower SNDR and poor stability under strong blocker conditions. On the other hand, the blocker suppression mode has an enhanced blocker suppression performance, with 8 dB attenuation at the adjacent

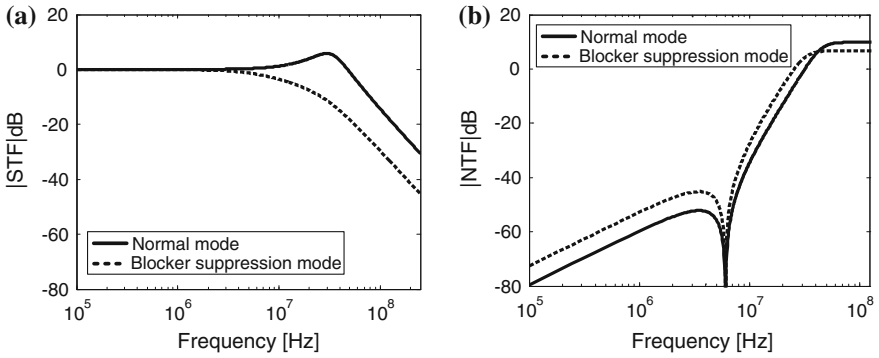

Fig. 2.23 ΣΔ ADC **a** STF and **b** NTF of the normal and blocker suppression mode

channel and 15 dB attenuation at the alternate channel. The drawbacks of this mode are reduced noise shaping and band-edge droop.

In the normal mode, increasing feedforward coefficient ff_1 can also reduce the gain peaking. However, it increases the −3 dB frequency of the STF and reduces the system stability because of increased high-frequency gain of the NTF. Thus an additional feedback path to the first integrator output is necessary.

2.6.4 Behavioral Simulations

Complex mixed-signal circuit design is often time-consuming and accompanies algorithmic iterative processes. Behavioral modeling and simulations can reduce

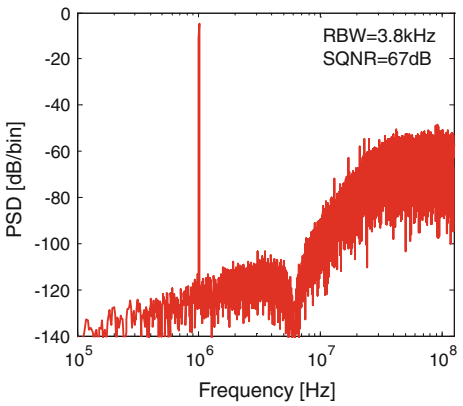

Fig. 2.24 Output PSD of the ADC with ideal blocks for an input signal 3 dB below full scale at 1 MHz

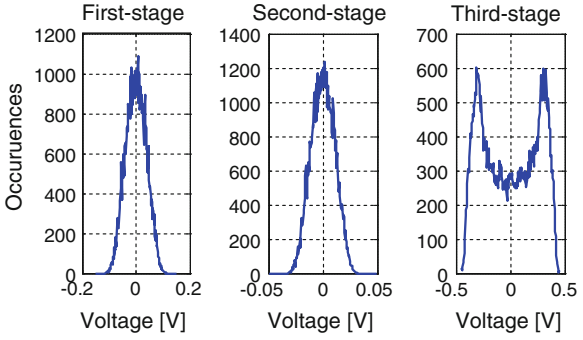

Fig. 2.25 Histograms of each integrator's output

such long design process and give initial specifications of circuit blocks for transistor level design.

Figure 2.24 shows the simulated output power spectrum density (PSD) of the ideal model and 67 dB SQNR is achieved. Figure 2.25 shows the simulated histograms of each integrator's output. From this simulation, a requirement of each integrator output swing can be estimated. The first stage integrator should have at least a 0.35 V_{pk-pk} output swing range to avoid signal clipping and a 0.1 V_{pk-pk} swing range is required for the second stage. Finally, the third stage must have a 1 V_{pk-pk} output swing range.

2.7 Behavioral Simulations with Non-idealities

2.7.1 Finite DC Gain and GBW in an OP-Amp

The finite DC gain and GBW of an op-amp creates an integrator coefficient's error and an additional pole which degrades performance of a modulator. To determine the minimum requirements for both parameters, iterative simulations with behavioral models can be performed.

The integrator transfer function with a finite DC gain and GBW can be shown as:

$$I_i(s) = \frac{k_i f_s}{s\left(1+\frac{1}{A(s)}\right)+\frac{1}{A(s)}\sum_{l=1}^{n} k_l f_s} = \frac{1}{1+\frac{s}{k_i f_s}} \cdot \frac{A(s)}{1+A(s)\beta(s)}, \tag{2.16}$$

where $\beta(s)$ is given by

$$\beta(s) = \frac{\frac{s}{k_i f_s}+\frac{1}{A(s)}\frac{\sum_{i} k_{l,l\neq i}}{k_i}}{1+\frac{s}{k_i f_s}} \tag{2.17}$$

Therefore the integrator with the finite gain and bandwidth can be modeled as shown in Fig. 2.26.

Figure 2.27 shows the simulated SQNR of the ADC with various open-loop DC gains and GBWs of the amplifier for each operation mode. With an ideal amplifier, the modulator has about 74 dB and 67 dB SQNR for the normal and blocker suppression mode, respectively. The SQNR starts decreasing, when the gain becomes lower than 40 dB and the GBW becomes lower than 1 GHz. Thus, the amplifier should have at least a 40 dB open-loop DC gain and 1 GHz GBW.

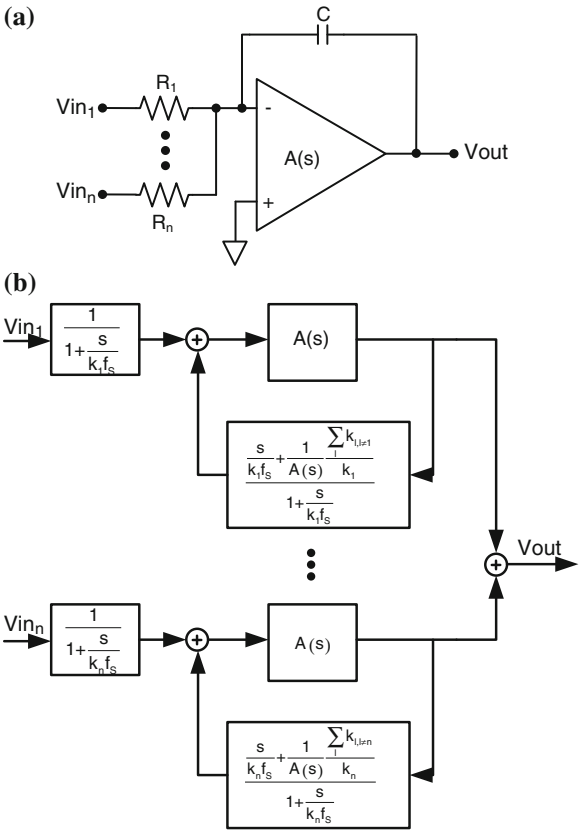

Fig. 2.26 **a** *n*-input active-RC integrator schematic and **b** its behavioral model

2.7.2 OP-Amp's Finite DC Gain for Blocker Detection

In an active-RC integrator with finite GBW OTAs, virtual ground at the OTA inputs degrades with the inverse of the amplifier AC response, providing a cost effective HPF performance to detect blockers.[18] Figure 2.28a shows a simple first-order CT ΣΔ ADC. With a linear quantizer model, the signal path shows LPF characteristic while a HPF characteristic results at the virtual ground node V_X due to a finite GBW of the OTA. The resulted HPF and STF equations are given by

$$HPF(s) = \frac{V_X(s)}{V_{IN}(s)} \approx \frac{\omega_1(s+\omega_c)}{s^2 + \omega_c A_{DC} s + \omega_1 \omega_c A_{DC}}$$

[18] Yoshizawa, A., Tsividis, Y.: A channel-select filter with agile blocker detection and adaptive power dissipation. IEEE J. Solid-State Circuits **42**(5), 1090–1099 (2007)

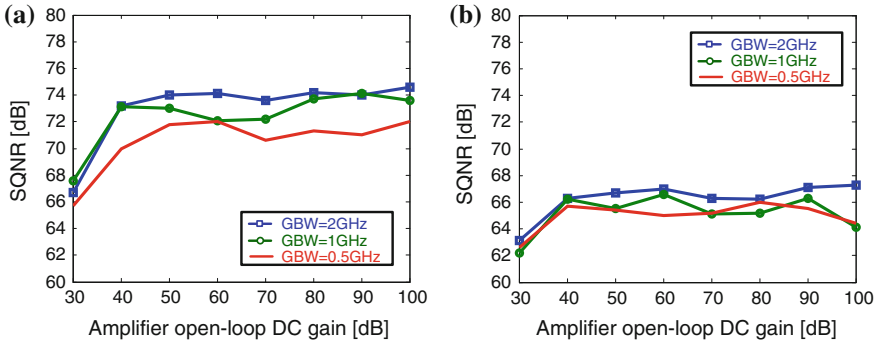

Fig. 2.27 Simulated SQNR of the proposed ADC with various open-loop DC gains and GBWs of the amplifiers for **a** the normal mode and **b** blocker suppression mode

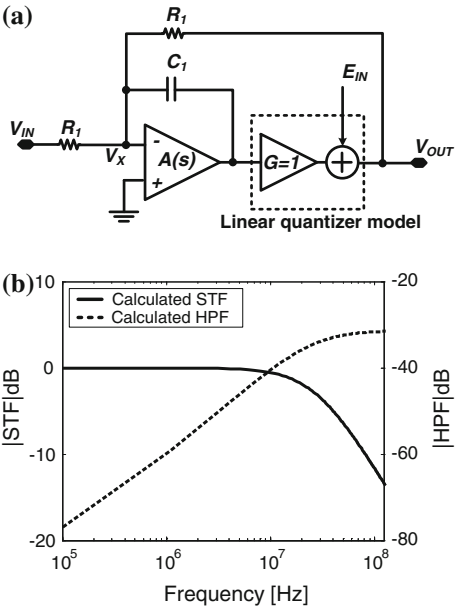

Fig. 2.28 **a** First-order CT ΣΔ ADC with a linear quantizer model **b** STF and HPF of the first-order CT ΣΔ ADC

$$STF(s) = \frac{V_{OUT}(s)}{V_{IN}(s)} \approx \frac{\omega_1 \omega_c A_{DC}}{s^2 + \omega_c A_{DC} s + \omega_1 \omega_c A_{DC}}$$

where A_{DC} and ω_c are the open-loop DC gain and -3 dB frequency of the op-amp, respectively and $\omega_1 = 1/R_1 C_1$ Fig. 2.28b shows the calculated STF and HPF when $A_{DC} = 75$ dB, ω_c=100 kHz, $R_1 = 1.5$ kΩ, and $C_1 = 4$ pF.

If the second and third integrators are assumed as ideal, the HPF characteristic of the proposed ΣΔ ADC in normal mode can be expressed as:

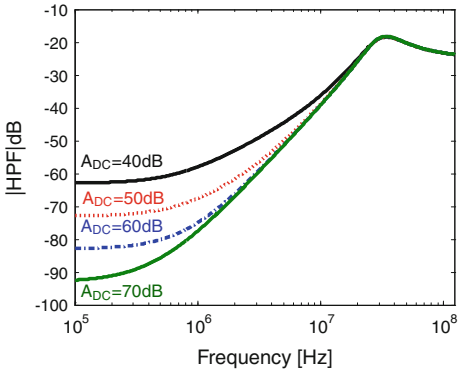

Fig. 2.29 Calculated HPF characteristic with various open-loop DC gains of the first op-amp

$$HPF(s) = \frac{1 + H_2(s)}{H_1(s)A_1(s) +_s C_1R_1[1 + A_1(S)][1 + H_2(s)] + 2H_2(s) + 2} \tag{2.18}$$

where $A_1(s)$ is the open-loop transfer function of the first op-amp. $H_1(s)$ is the transfer function from the first integrator output to the quantizer input in Fig. 2.22. $H_2(s)$ is the loop filter transfer function when the feedback path fb_1 is set to zero.

The transfer functions are given by:

$$\begin{aligned} H_1(s) &= \frac{a_3 ff_1 F_s S + a_2 a_3 F_s^2}{s^2 + a_2 a_3 g_1 F_s^2}, \\ H_2(s) &= \frac{a_3(fb_2 + fb_3)F_s^s}{s^2 + a_2 a_3 g_1 F_s^2} \end{aligned} \tag{2.19}$$

Figure 2.29 shows the HPF of the proposed ΣΔ ADC with various open-loop DC gains of the first stage op-amp when the amplifier has 1 GHz of GBW. The HPF has the corner frequency at 20 MHz and the gain peaking at adjacent channel frequency. For SQNR performance, 40 dB of DC gain is enough but in order to detect interferers at adjacent and alternate channel frequencies of WiMAX standard while suppressing the desired channel signals, additional DC gain is necessary. Thus, 60 dB of DC gain is set for the specification. This additional gain would also reduce the input referred thermal noise and non-linearity contributions from the second and third stage integrator and the quantizer.

2.7.3 Device Noise

In most of state-of-the-art ADCs, achievable DR is usually limited by device noise. Consequently, effort to reduce the device should be accompanied during design process with minimum power consumption. In CT ΔΣ ADCs, the input referred device noise primarily originates from the first stage integrator and feedback DAC. In order to achieve a 60 dB DR over a 10 MHz bandwidth with the 1 V_{pk-pk}

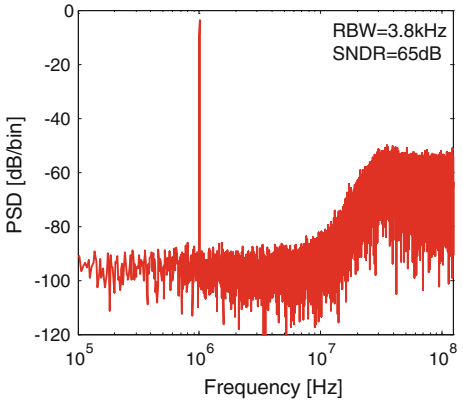

Fig. 2.30 Simulated output PSD for an input signal 3 dB below full scale at 1 MHz with 60 dB DC gain and 1 GHz GBW of the op-amps, and 56 nV/$\sqrt{\text{Hz}}$device input referred noise

differential FS range, the input referred noise density should be lower than 56 nV/$\sqrt{\text{Hz}}$. In reality the thermal noise will add to the quantization noise floor due to op-amp's finite DC gain and GBW.[19] Consequently, the ADC behavioral model must be simulated with equivalent device noise sources, to obtain a more accurate estimation of the DR. Figure 2.30 shows the simulated power spectrum density (PSD) of the modulator operating in the blocker suppression mode with 60 dB DC gain and 1 GHz GBW of the op-amps, and 56 nV/$\sqrt{\text{Hz}}$ device input referred noise. The modulator can achieve 65 dB SNDR which is 5 dB greater than the DR requirement.

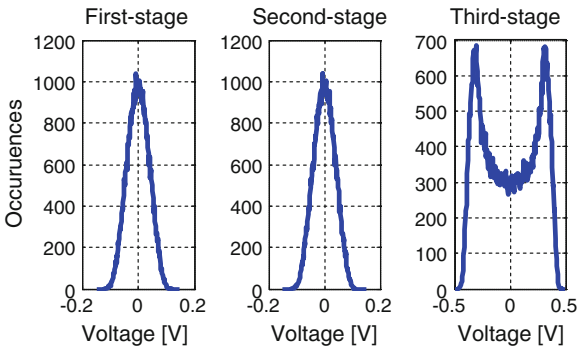

Fig. 2.31 Histograms of the integrators' output of the proposed modulator

[19] Park, M.: A fourth-order continuous-time ΣΔ ADC with VCO-based integrator and quantizer. Ph.D. dissertation, Massachusetts Institute Technology, Cambridge (2009)

2.7.4 Integrator Output Swing

In active-RC integrators, nonlinearity is usually determined by integrators' output swing range. Thus, loop filter architecture should be properly determined or integrators should have an enough output swing range. To specify the required output swing range, behavioral simulations can be performed. Figure 2.31 shows the histograms of the integrators' output of the proposed ΣΔ ADC. The first and second stage integrator should have at least 0.35 V_{pk-pk} output swing range to avoid signal clipping. The third stage must have 1 V_{pk-pk} output swing range. The requirement of the second and third stage can be reduced because nonlinearities caused by these stages would be reduced by the first stage integrator. Here, all signal swing range is differential.

2.7.5 DAC Mismatch

In a multi-bit ΣΔ modulator, a modulator's nonlinearity originates from mismatches in the first stage feedback DAC. To estimate system performance versus the DAC mismatch, the unit current of the multi-bit DAC i_{DAC}, can be modeled as:

$$i_{DAC} = i_{nom} + i_{error} \tag{2.20}$$

where i_{nom} is the nominal value of the unit current and i_{error} represents the mismatch between unit current elements, which is a Gaussian distributed random number. SNDR performance of the proposed modulator with DAC mismatch can be investigated by using the above model and Fig. 2.32 shows the behavioral simulation result with different standard deviations of the error when the modulator is operating in the blocker suppression mode. To meet the 60 dB DR requirement, the standard deviation should be lower than 0.4 %. This condition can be easily satisfied by employing a self-current calibration technique.[20]

2.7.6 Clock Jitter

Unlike DT ΣΔ modulators, CT counterparts are very sensitive to sampling clock uncertainties, called clock jitter. Clock jitter adds random noise to a loop filter of the modulator and increases noise floor. Additive error sequence due to the clock jitter can be expressed as[21]

---

[20] Groeneveld, D.W.J. et al.: A self-calibration technique for monolithic high resolution D/A converters. IEEE J. Solid-State Circuits **24**(6), 1517–1522 (1989)

[21] Hernandez, L. et al.: Modeling and optimization of low pass continuous-time sigma-delta modulators for clock jitter noise reduction. ISCAS, pp. 1072–1075 (2004)

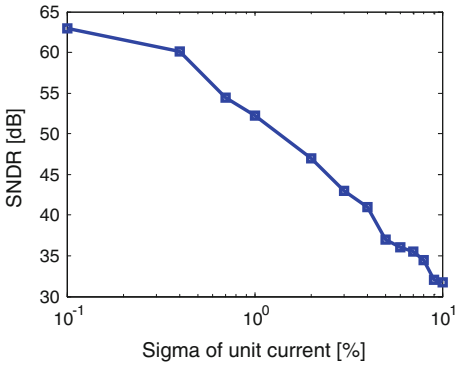

Fig. 2.32 SNDR versus DAC unit current mismatch

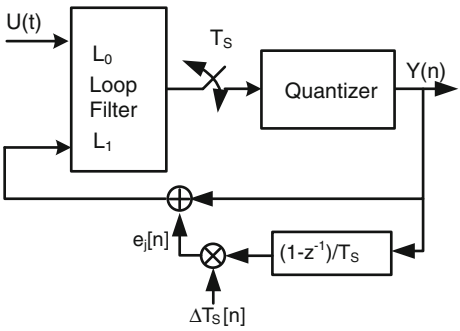

Fig. 2.33 CT ΣΔ ADC model with additive clock jitter

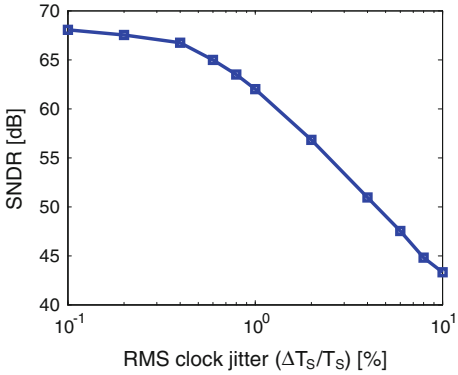

Fig. 2.34 SNDR versus RMS clock jitter

$$e_j[n] = (y[n] - y[n-1])\frac{\Delta T_s[n]}{T_s}, \tag{2.21}$$

where T_S is a clock period and ΔT_S represents clock uncertainty. From Eq. (2.21), a CT ΣΔ modulator with clock jitter can be modeled as shown in Fig. 2.33. The modulator is simulated with different amount of clock jitter and the result is shown

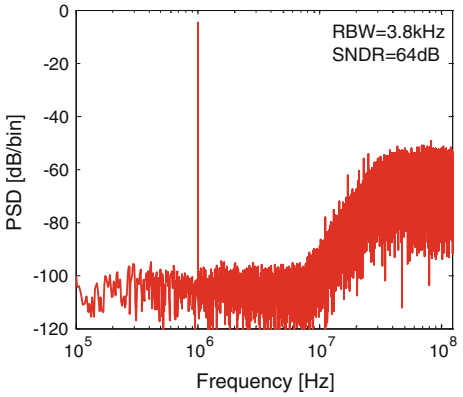

Fig. 2.35 Simulated output PSD for an input signal 3 dB below full scale at 1 MHz with 60 dB DC gain and 1 GHz GBW of the op-amps, 56 nV/$\sqrt{\text{Hz}}$ device input referred noise, and 16 ps RMS clock jitter

in Fig. 2.34. To achieve 66 dB DR, the RMS clock jitter should be less than 0.4 % respect to T_S (16 ps). The ADC is also simulated with non-idealities defined previously and the simulated PSD of the ADC output is shown in Fig. 2.35. With all non-idealities defined in this chapter, the modulator can achieve 64 dB DR.

2.7.7 Simulations with a 24 Mbps 16-QAM Signal

To investigate feasibility of the blocker-adaptive ΣΔ ADC for WiMAX applications, the modulator needs to be tested with a modulated input signal specified in the standard. Figure 2.36 shows the test setup for this behavioral simulation. A 24 Mbps 16-QAM signal with OFDM is generated by using MATLAB Communications toolbox and the desired channel and interferer powers are set as specified by the standard. Also the RF front-end gain, G_{RX}, is placed between the

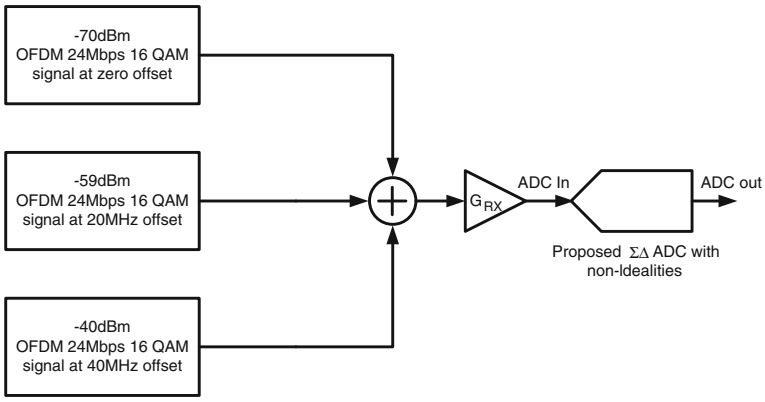

Fig. 2.36 Behavioral simulation test set-up with 24 Mbps 16-QAM WiMAX signals

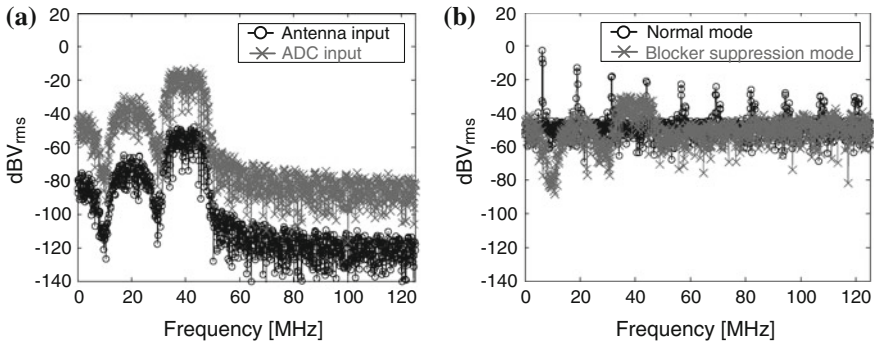

Fig. 2.37 Simulated PSDs of 16-QAM WiMAX signals at **a** the antenna and ADC inputs and **b** ADC output

Table 2.2 Summary of specifications of the proposed ADC

Process	130 nm digital CMOS process
VDD	1.2 V
FS	1 V_{pk-pk} differential
DR	60 dB
IIP3	7.25 dBm

ADC and the WiMAX signal generator. Furthermore, all the previously defined non-idealities are accounted for in the ADC model.

Figure 2.37 shows the PSDs at the antenna and ADC inputs and ADC output for the normal mode and blocker suppression mode. With 35 dB RF front-end gain, the ADC faces strong interferers which are -37 dBV_{rms} and -18 dBV_{rms} at the adjacent channel and alternate channel, respectively. In the normal mode, the 5 dB gain peaking in the adjacent and alternate channel leads to ADC instability. On the other hand, with 8 dB and 15 dB attenuation at those frequencies the ADC remains functional with good performance in the blocker suppression mode. As discussed above, to account for non-idealities of the proposed modulator such as finite DC gain and GBW of op-amps, DAC nonlinearities, integrators' output swing, clock jitter, and device noise, several behavioral modeling methods are introduced. Also, with iterative behavioral simulations, the circuit parameters are determined and they are summarized in Table 2.2. Corresponding ADC parameters are summarized in Table 2.3.

2.8 Implementation of the Blocker Adaptive ΣΔ ADC

Figure 2.38 shows the schematic of the proposed adaptive blocker rejection ΣΔ ADC. It can be divided into four blocks: the 3rd-order reconfigurable loop filter,

Table 2.3 Summary of specifications of the proposed ADC and circuit building blocks

Process	130 nm digital CMOS process
ADC specifications	
VDD	1.2 V
FS	1 V_{pk-pk} differential
DR	60 dB
IIP3	7.25 dBm
Circuits' specifications	
Op-amp DC gain	60 dB
Op-amp GBW	1 GHz
1st/2nd/3rd stage integrator output swing (differential)	0.35/0.35/1 V_{pk-pk}
Device input referred noise	56 nV/$\sqrt{\text{Hz}}$
DAC mismatch	0.4 % (standard deviation)
Clock Jitter	16 ps

quantizer, feedback path DACs, and blocker detector. For high linearity, active-RC integrators are used to implement to the loop filter. A 13-level flash ADC is used for quantizer to improve system stability and DR. For high-speed operation (250 MHz), current steering DACs are employed. Excess loop delay caused by the quantizer and DACs is compensated by the digital differentiator in the feedback path, to avoid an additional summing amplifier or a return-to-zero DAC, which would require extra power.[22] DAC2 is re-used for both operating modes, thus no extra die area is required to implement the adaptive architecture.

2.8.1 Loop Filter Design

The 3rd-order reconfigurable loop filter is designed with active-RC integrators due to their high linearity and well-defined common mode voltage. The reconfigurability can be performed by closing or opening switches SW1 s and SW2 s. The main drawback of the active-RC integrators is RC time constant variation which is up to ±40 %.[23,24] To compensate for these variations, manually controlled binary-weighted tunable capacitor arrays are employed as shown in Fig. 2.39. For the 1st and 2nd stage integrator, the main capacitor, C_{MAIN}, is set to 2.6 and a 0.2 pF of

[22] Mitteregger, G. et al.: A 20-mW 640-MHz CMOS continuous-time ADC with 20-MHz signal bandwidth, 80-dB dynamic range and 12-bit ENOB. IEEE J. Solid-State Circuits **41**(12), 2641–2649 (2006)

[23] Kappes, M.S., Jensen, H., Gloerstad, T.: A versatile 1.75 mW CMOS continuous-time delta-sigma ADC with 75 dB dynamic range for wireless applications. In: Proceedings European Solid State Circuits Conference pp. 279–282 (2002)

[24] Giandomenico, A.D. et al.: A 15 MHz bandwidth sigma-delta ADC with 11 bits of resolution in 0.13 m CMOS. In: Proceedings European Solid State Circuits Conference, pp. 233–236 (2003)

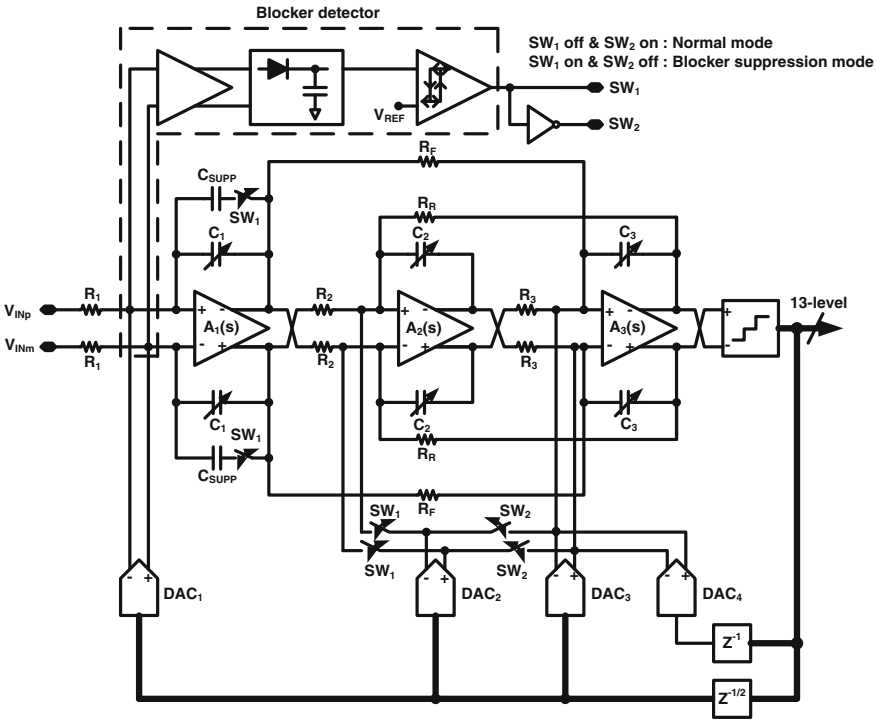

Fig. 2.38 Schematic of the proposed ΣΔ ADC

the least significant bit (LSB) capacitor, C_{LSB}, is employed. This configuration gives a tuning range of from 2.6 to 5.6 pF with 5 % accuracy.

Figure 2.39a shows the schematic of the OTA used for the first active-RC integrator which sets the performance of the overall modulator. Since the required differential output swing range is just 0.35 V_{pk-pk}, a power efficient telescopic cascode amplifier with gain boosting can be used.[25] The gain-boosting amplifiers are single-ended cascade common source amplifiers while the OTA's tail current source is biased in triode region to increase the output voltage swing.

Figure 2.41 shows the simulated open-loop frequency response of the first stage OTA. The OTA has 75 dB of open-loop DC gain and 1-GHz GBW with 4 pF load capacitance while consuming 3 mA quiescent current and achieves 70° phase margin. Although a low output impedance output stage is preferred to drive resistive loads, with a 40 kΩ load, the OTA gain is reduced by only 10 dB (red dashed line in Fig. 2.41), and thus the output stage was not adopted to save power.

[25] Christen, T., Burger, T., Huang, Q.: A 0.13 m CMOS EDGE/UMTS/WLAN tri-mode ADC with −92 dB THD. In: IEEE ISSCC Digital Technical Papers, pp. 240–241 (2007)

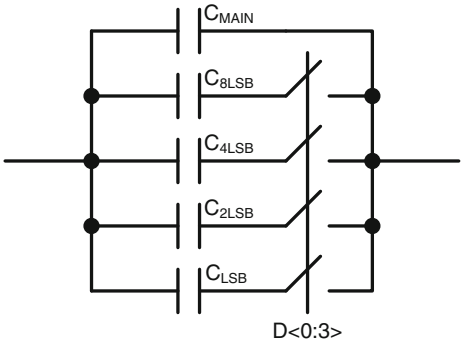

Fig. 2.39 Binary weighted tunable capacitor array

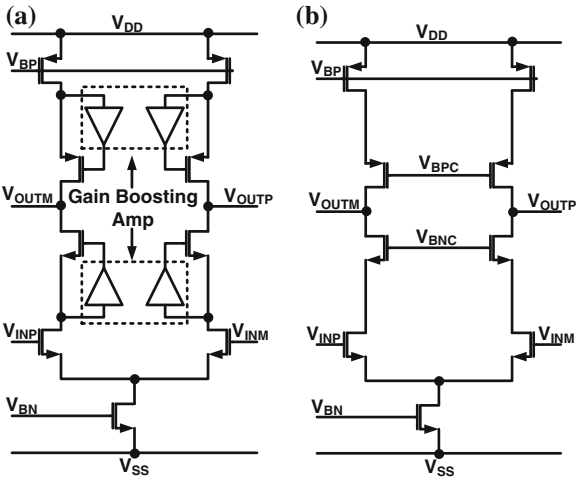

Fig. 2.40 Schematic of **a** the first stage and **b** following stages' OTA

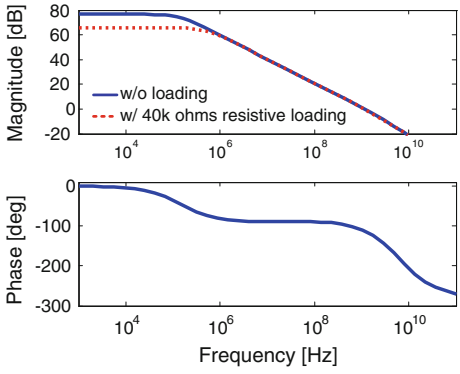

Fig. 2.41 Simulated open-loop frequency response of the first stage OTA

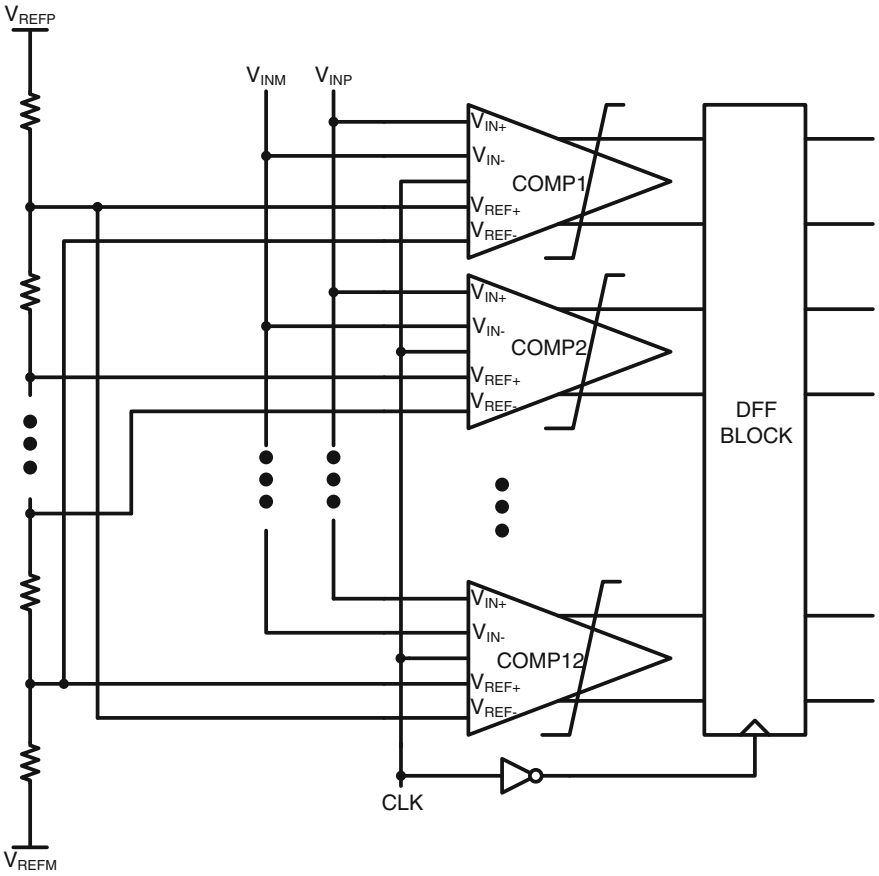

Fig. 2.42 Schematic of the quantizer

Once blockers, as specified, are detected, the ADC operates in the blocker suppression mode and the blockers would be suppressed. Thus, the feedback path would carry attenuated blockers, and the OTA input nodes of the first active-RC integrator can maintain a good virtual ground. Overall ADC loop-gain would help stabilize the first integrator virtual ground. Consequently, an additional linearity requirement is unnecessary for the input transistors in the OTA.

The schematic of the OTAs used in the remaining loop filter is shown in Fig. 2.40b. Because noise and distortion contributions of the second and third stages are reduced by the preceding gain stage, the gain boosting amplifiers are removed and the bias currents are scaled down by a factor of 2.

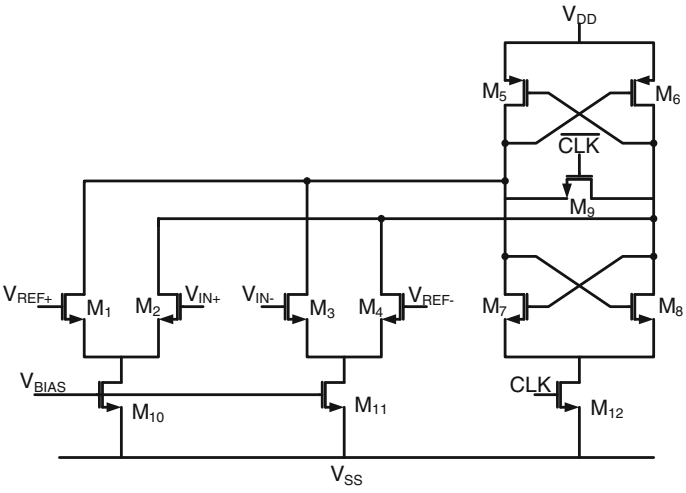

Fig. 2.43 Schematic of the comparator

2.8.2 Quantizer

For the Quantizer, the 13-level flash ADC is employed and its schematic is shown in Fig. 2.42. For reference generation, a resistor ladder is used and fully differential comparators are employed. The D flip-flop block is placed after the comparator to fix the large excess loop delay and this delay is compensated for by digital differentiator technique.

For comparators, a latched comparator with a preamplifier is designed and its schematic is shown in Fig. 2.43.[26] The preamplifier is added to reduce metastability and kick-back noise. Since the quantizer is the least critical block, the comparators are designed for low-power consumption.

Fifty Monte-Carlo simulations are performed to estimate input referred offset voltage of the comparator. Figure 2.44 shows the histograms of the input referred offset voltage. The mean value of the offset is 20 mV and the standard deviation is 10 mV. Since the LSB of the quantizer is about 83 mV, the offset voltage is well below a third of the LSB. Moreover, it is attenuated by the gain of the loop filter when it is referred to the ADC input and thus the ADC performance would not be degraded by the quantizer offset voltage.

[26] Razavi, B.: Principles of Data Conversion Systems Design. IEEE Press, Piscataway (1995)

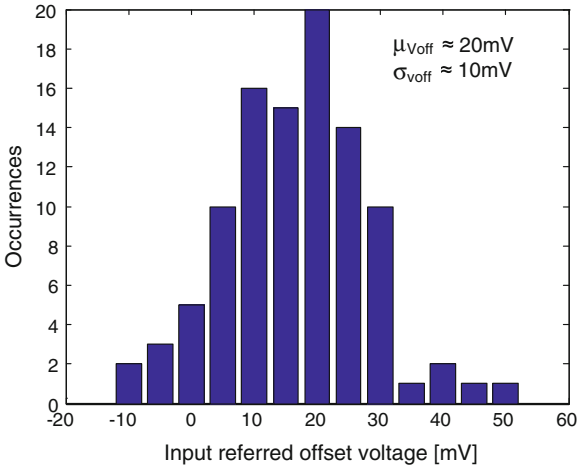

Fig. 2.44 Histograms of simulated input referred offset voltage obtained from 100 Monte-Carlo simulations

2.8.3 Feedback DAC

Figure 2.45a shows the schematic of a conventional current steering DAC unit cell. There are several critical design issues associated with feedback DAC design; nonlinear output impedance, glitch energy caused by clock feedthrough due to parasitic capacitor C_{gd} and voltage fluctuations at the source node of the switch transistors M_1 and M_2, resulting from charging and discharging the parasitic capacitor C_p. The first stage DAC should minimize these problems, since all of these non-idealities translate to the output without any noise shaping.

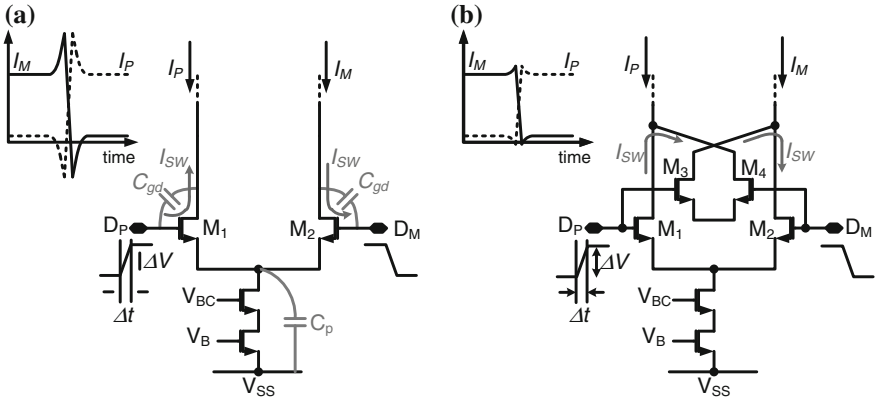

Fig. 2.45 Schematic of **a** conventional and **b** the proposed current steering DACs

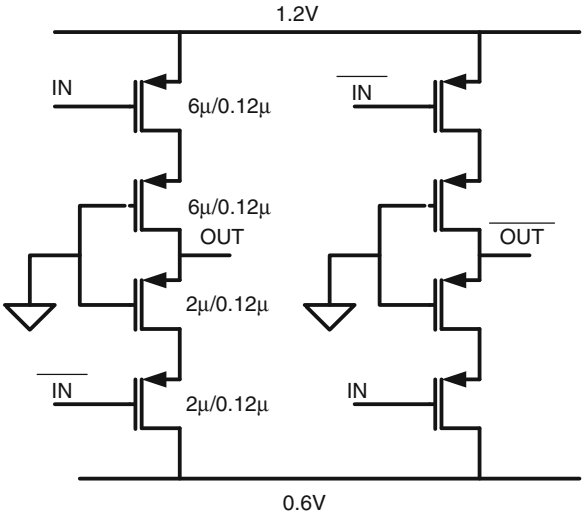

Fig. 2.46 Low-swing, high-cross over signal generation for current steering DACs

Figure 2.46 shows a reduced-swing and high-crossover DAC driver. It can reduce glitch energy by guaranteeing either one of the DAC switches closed and minimize clock feedthrough effects with reduced swing.[27] In addition, it can increase the output impedance by operating M_1 and M_2 in the saturation region, reducing nonlinearity of the output impedance.

The switching current I_{SW} generated by clock feedthrough is given by:

$$I_{SW} = \frac{dQ_{c_{gd}}}{dt} = C_{gd}\frac{dv}{dt}, \tag{5.1}$$

where dV and dt are the amplitude and transition time of the switch control signals, D_p and D_m. In high-speed applications, the clock feedthrough effects would be more prominent and degrade SNDR performance. To further reduce clock feedthrough effects, the current steering DAC unit cell employs two additional cross-coupled transistors, M_3 and M_4, as shown in Fig. 2.45b. If M_3 and M_4 have the same size of M_1 and M_2, the switching currents through the C_{gd} of M_3 and M_4 cancel the current injection of the main transistor pair, further minimizing spikes.

Figure 2.47 shows the block diagram configuration of the current steering DAC chain. It consists of a retiming flip-flop to synchronize with other DAC cells, a reduced-swing high-crossover driver stage, and the proposed current steering DAC cell. Figure 2.48 shows the transistor-level simulation of the unit current DAC cell with this configuration.

[27] Wu, T.-Y. et al.: A low glitch 10-bit 75-MHz CMOS video D/A converter. IEEE J. Solid-State Circuits **30**, 68–72 (1995)

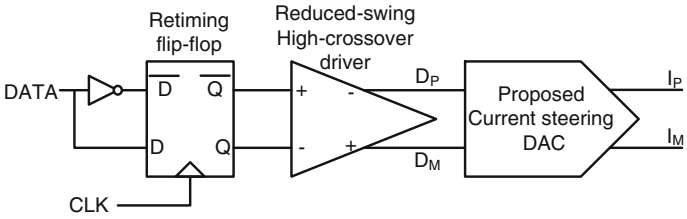

Fig. 2.47 Current steering DAC with the driver and retiming block

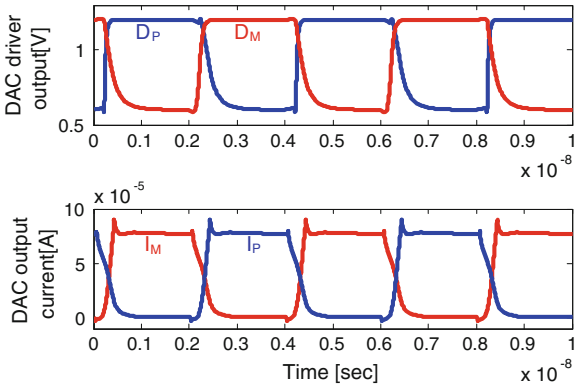

Fig. 2.48 Transistor-level simulation of the unit current DAC cell and the DAC driver

2.8.4 Self-Calibration Technique

Unit cell mismatches in multi-bit DACs limit the overall ADC linearity. In order to maintain system linearity, the first DAC should have at least 11-bit linearity even though a 3.5-bit DAC is used. To improve the linearity, data weighted averaging (DWA) or self-current calibration techniques can be used. [28,29] In this design, the self-current calibration technique is employed instead of DWA because it does not add further excess loop delay. Figure 2.49 shows the schematic of the self-current calibration technique where the two cross-coupled transistors are not shown for simplicity. 95–97 % of the reference current I_{REF} is assigned to the coarse current source M_1, I_{COARSE}. During the calibration phase, M_4 and M_5 are open and M_6 and M_7 are closed so that the fine current source M_2 can compensate the difference between I_{REF} and I_{COARSE}. M_8 and M_9 are added to reduce charge injection caused by M_7. The parasitic capacitance C_{gs} of M_2 holds its bias voltage until the next calibration phase, and this current cell can generate the calibrated current I_{REF}

[28] Li, Z., Fiez, T.S.: A 14 bit continuous-time delta-sigma A/D modulator with 2.5 MHz signal bandwidth. IEEE J. Solid-State Circuits **42**(9), 1873–1883 (2007)

[29] Geerts, Y., Steyaert, M., Sansen, W.: Design of Multi-Bit Delta-Sigma A/D Converters. Kluwer Academic, Norwell (2002)

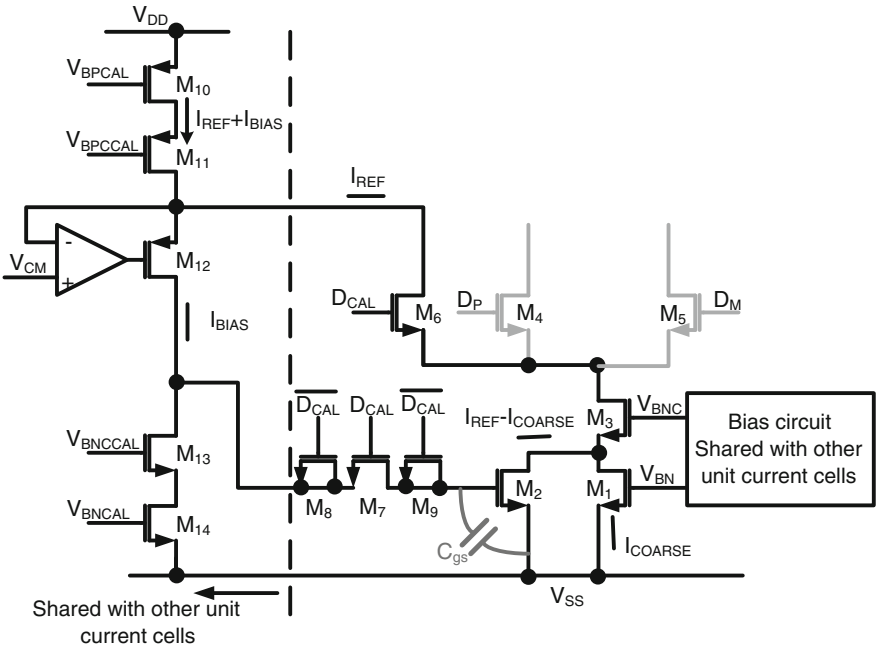

Fig. 2.49 Schematic of the self-current calibration technique

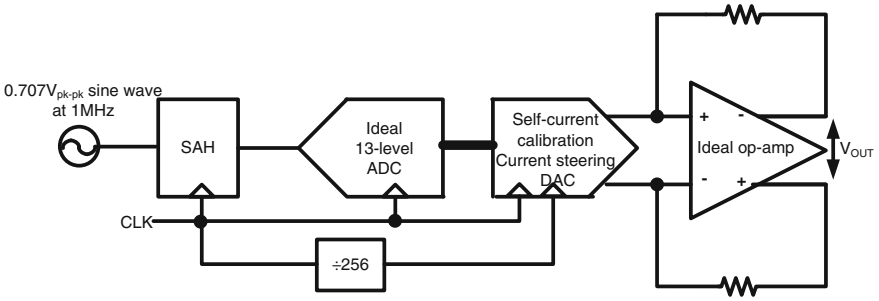

Fig. 2.50 Test setup for estimating performance of the self-current calibration DAC

during the normal operation phase. 1 μs is assigned to calibrate each unit current cell, and a total of 13 unit current cells are used for the first stage DAC. This rotational calibration is performed continuously at every 13 μs.

To estimate performance of the self-current calibration current steering DAC, the DAC which has 10 % of the unit current is driven by ideal ADC output and Fig. 2.50 shows the test setup. Figure 2.51 shows the simulated output PSD of the DAC when the self-current calibration is on and off. When the self-current calibration is off, the DAC has 30 dB of HD3 but when it is on, linearity of the DAC is improved significantly and the HD3 is not visible.

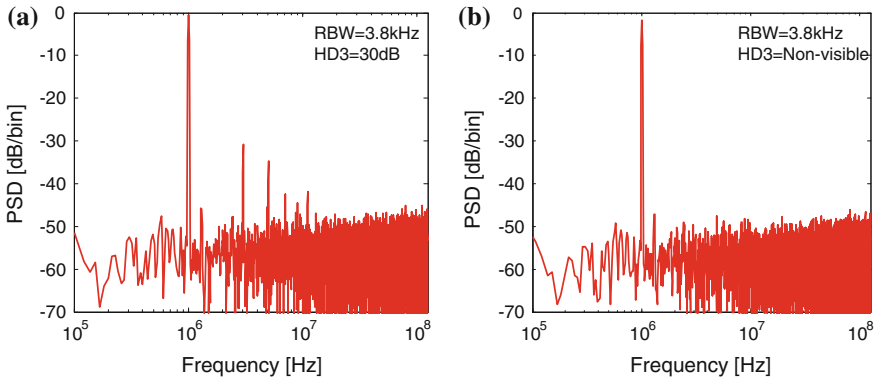

Fig. 2.51 Output PSD of the first stage DAC driven by the ideal 13-level flash ADC when self-current calibration is **a** on and **b** off

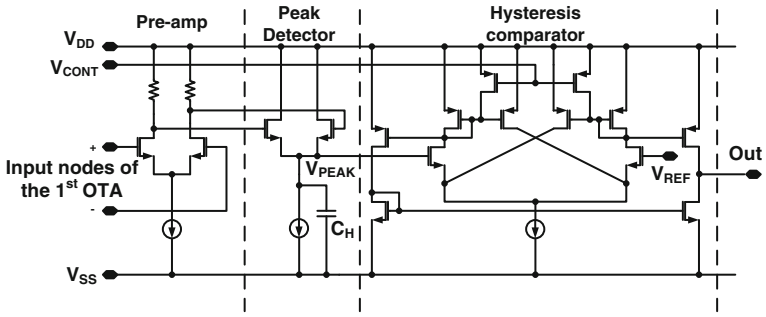

Fig. 2.52 Schematic of the blocker detector

Blocker Detector

A typical blocker detector employs a high-pass filter (HPF) followed by an amplitude detector. In this design, virtual ground at the first active-RC integrator's inputs is exploited to obtain a HPF characteristic.[18] Therefore power consumption and silicon area can be saved. To increase accuracy of blocker detection, low-noise pre-amplifier is added in front of the amplitude detector which is implemented by an NMOS-based rectifier. The detector output is compared with the hysteresis comparator and its output controls operation mode of the modulator.

Figure 2.52 shows the schematic of the blocker detector. It consists of a low-noise amplifier, peak detector and hysteretic comparator. A low-noise amplifier is added before the peak detector to increase detector accuracy. An NMOS rectifier and capacitor C_H achieves blocker peak detection. An important point to make is that quantization noise and tonal content at the quantizer output also undergoes the high-pass characteristic at the virtual ground nodes of the first integrator. Therefore, the hysteretic characteristic in level detection is required to avoid erroneous toggling of the comparator output due to quantization noise. The hysteresis level

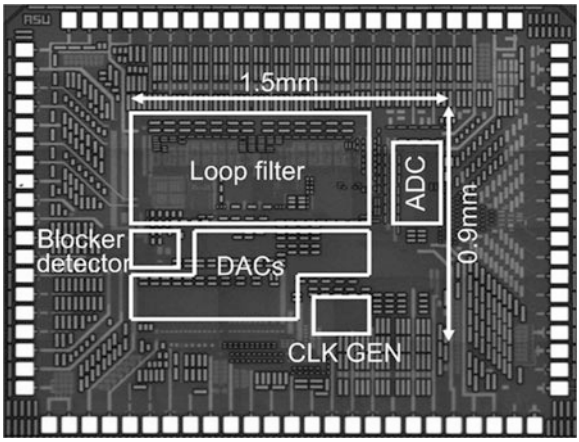

Fig. 2.53 Chip micrograph

and the accuracy of the blocker level detection are obtained through on transistor level simulations. When a −40 dBFS input signal is applied with a 20 dBc adjacent channel blocker, the ADC in the normal mode has lower SNDR than the blocker suppression mode. At this point, the peak detector generates V_{PEAK} of 700 mV, and this value is set as the threshold (V_{REF}) for changing between ADC modes. A ± 1 dB change in blocker power around its nominal value causes ±5 mV change in the peak detector output, setting the overall accuracy. A hysteresis level of 20 mV is determined to avoid false triggers due to quantization noise and tonal content. For testing purposes the hysteresis points are analog programmable through control voltage V_{CONT}.

A DC offset or low-frequency input signal could generate tonal content at high frequencies, and these tones can also cause erroneous toggling of the comparator output. From behavioral simulations, the worst case DC related tonal content is measured to be around −25 dBFS, which generates 600 mV V_{PEAK}. Since the hysteresis level is set at 700 mV, the worst case tonal content is safely below the trip threshold of the comparator.

2.8.5 Floor Plan and Layout

A floor plan and layout of the ADC must be carefully considered because performance of the ADC is strongly depends on routing of critical signal paths, noise coupling from digital sections to analog sections, and etc. Figure 2.16 shows the floor plan of the proposed ADC. The analog sections including the loop filter, the feedback DACs, and the blocker detector are separated from the digital sections consisting of the quantizer, the clock generator, the digital signal path for DAC control. For further isolation between the analog and digital sections, both sections are enclosed by double guard rings. Also power supply and ground for the analog sections are separated from ones for the digital sections to protect the sensitive

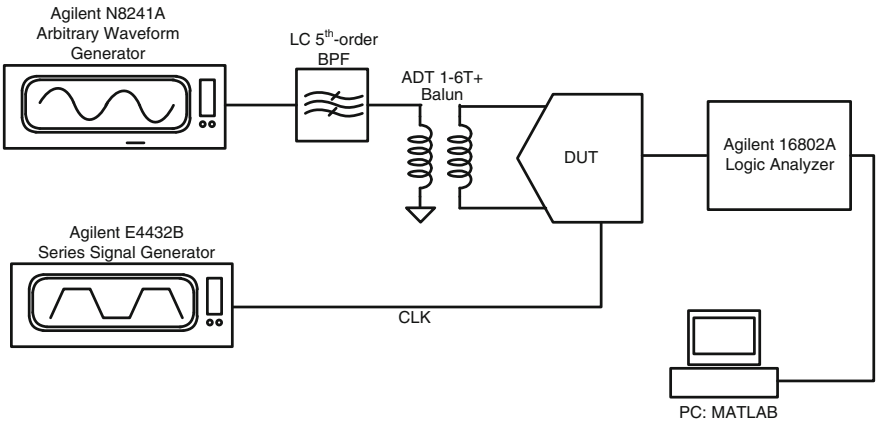

Fig. 2.54 Test setup for evaluation of the prototype ADC

analog building blocks from high speed switching current noise caused by the digital circuits. The proposed ΣΔ ADC is designed using 1.2 V 130 nm CMOS process which has 8 metal levels and MIM capacitors. The active occupies 1.5×0.9 mm^2 silicon area, as shown Fig. 2.53.

2.9 Measurement

2.9.1 Test Setup

Figure 2.54 shows the test setup used to evaluate the prototype ADC. A test input signal is generated by the arbitrary waveform generator (Agilent N8241A AWG)

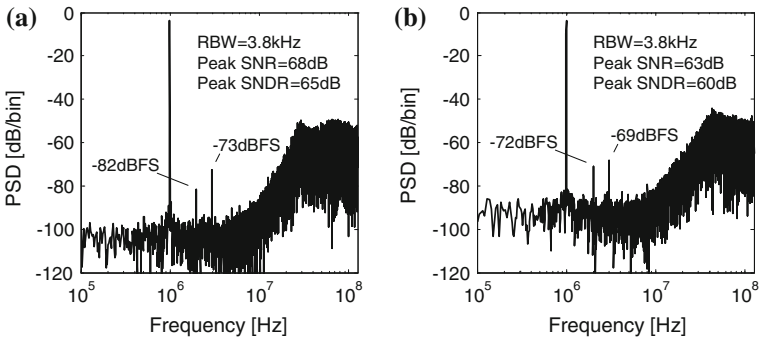

Fig. 2.55 Measured output PSD for an input signal 4 dB below full scale at 1 MHz for **a** the normal mode and **b** the blocker suppression mode

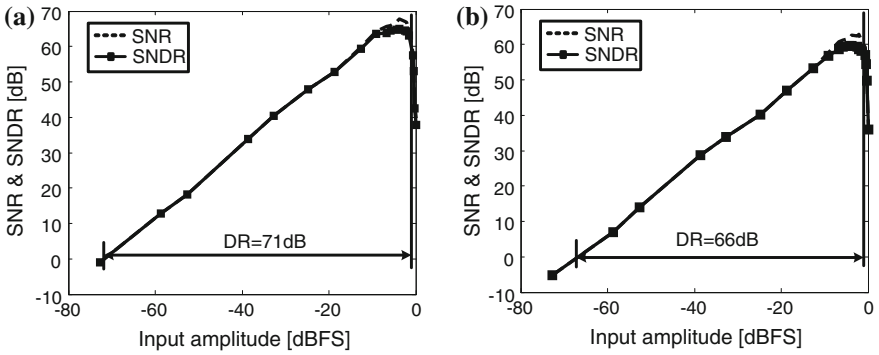

Fig. 2.56 Measured SNR and SNDR versus the normalized input signal amplitude for **a** the normal mode and **b** the blocker suppression mode

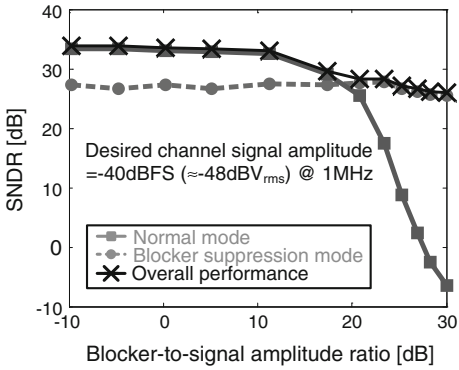

Fig. 2.57 Measured SNDR for the normal and blocker suppression mode at different blocker levels with −40 dBFS desired channel input signal

and it passes through the LC 5th-order BPF to suppress its harmonics. Then, the filter output signal is applied to the balun (ADT 1-6T+) for single to differential conversion. The differential input signal is fed to the prototype ADC, clocked by a 250 MHz pulse which is generated by clock generator (Agilent E4432B Series Signal Generator). The ADC's output is captured by the logic analyzer (Agilent 16802A Logic analyzer) and the captured data is downloaded to PC for post-processing.

2.9.2 Single-Tone Test

The proposed ΣΔ ADC is fabricated on a 1.2 V 130 nm CMOS technology, which features 8 metal levels and MIM capacitors. The die occupies a 1.5 × 0.9 mm^2 silicon area, as shown in Fig. 2.55.

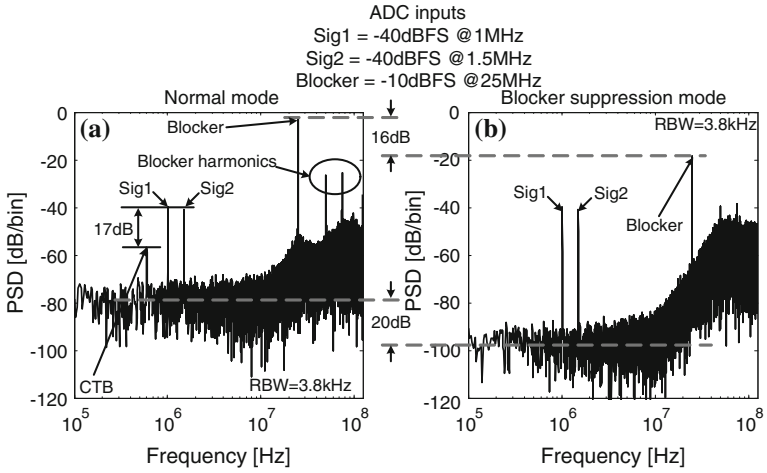

Fig. 2.58 Measured two-tone in-band tests with a 30 dBc extra adjacent channel blocker applied at the ADC input for **a** the normal mode and **b** the blocker suppression mode

Figure 2.56 shows the measured output PSD for an input signal −4 dB below full scale at 1 MHz for both operating modes. Figure 2.57 shows the measured SNR and SNDR versus the normalized input signal amplitude for both modes. Over a 10 MHz signal bandwidth, the ADC achieves 68 dB peak SNR, 65 dB peak SNDR, and 71 dB DR in the normal mode with 18 mW power consumption while the blocker suppression mode obtains 63 dB peak SNR, 60 dB peak SNDR, and 66 dB DR, consuming the same power. In Fig. 2.56, the raised noise around the band-edge smears the notch because the loop is close to instability as the second and third integrators get closer to saturation.

2.9.3 ADC Test under Blocking Conditions

2.9.3.1 Single-Tone Test with Blockers

Figure 2.58 shows SNDR performance at different blocker levels at 25 MHz offset. The amplitude of the desired channel signal is set to −40 dBFS (≈ -48 dBV$_{\text{rms}}$) at 1 MHz because it represents the minimum input signal level amplified by 35 dB RF front-end gain. When the blocker to signal amplitude ratio is less than 20 dB, the normal mode operation has better performance than the blocker suppression mode. However, the normal mode starts decreasing SNDR performance when the blocker is greater than 20 dBc while the blocker suppression mode keeps its SDNR performance around 26 dB. With reconfigurable operation between the two modes, at least 26 dB SNDR performance is guaranteed even with the 30 dBc blocker applied at the adjacent channel.

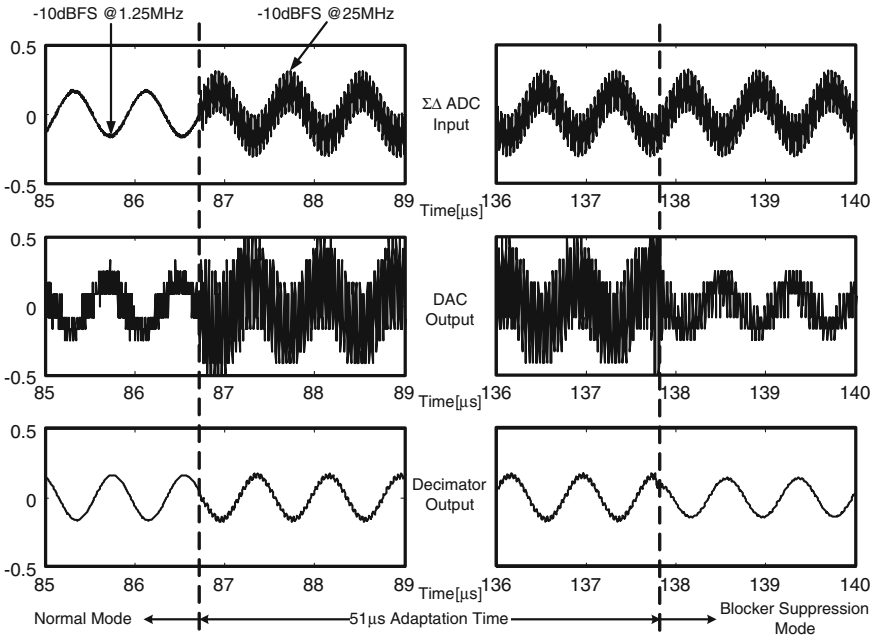

Fig. 2.59 ΣΔ ADC measured transient response with 25 MHz interferer applied to the system

2.10 Transient Measurement Under Blocking Condition

Figure 2.59 shows the measured output transient after a 25 MHz interferer appears at the ADC input during normal reception. The ADC can reconfigure to the blocker suppression mode in 51 μs. For the WiMAX standard, the symbol duration is 102.9 μs and the frame duration is 5 ms (48 symbols) for a 10 MHz bandwidth

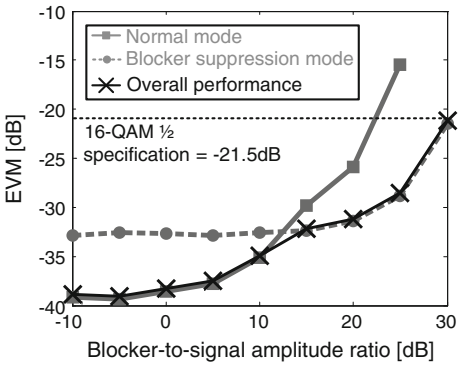

Fig. 2.60 Measured EVM of the normal and blocker suppression modes at different blocker levels

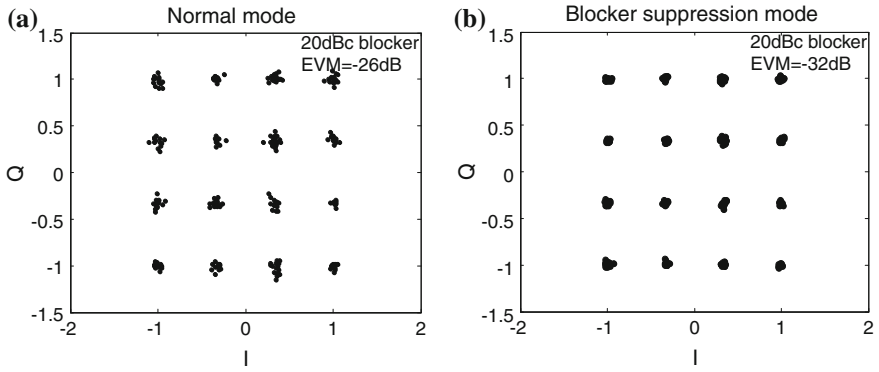

Fig. 2.61 Measured constellations for the normal and blocker suppression modes with 20 dB blocker to signal amplitude ratio

Table 2.4 Summary between the normal and blocker suppression modes

		Normal mode	Blocker suppression mode
DR (dB)		71	66
Composite triple beat (dBc)		17	N/A
SNDR (dB)	Weak blockers (<10 dBc)	34	27
	Strong blockers (30 dBc)	−6 (Unstable)	26
EVM (dB)	Weak blockers (<10 dBc)	−39	−33
	Strong blockers (30 dBc)	N/A	−22
Power consumption (mW)		18	
Integrators		7	
DACs		3	
Blocker detector		1	
Digital blocks		7	

modulation mode.[30] Since the reconfiguration time is 51 μs, only 1 % of a frame would be affected as a worst case. Furthermore, as long as the blocker does not appear during the preamble and frame-map-symbols, the error correction code will fix a corrupted data bit. Because the probability of interferers appearing specifically during the preamble period is about 2 %, the reconfiguration time would have minor effects in data reception. Moreover, because of the availability of interfame gap of the WiMAX standard, the ADC mode can be initialized with the blocker suppression mode in this gap, and then switched to the normal mode if the jammer is not present before the frame starts. This ensures that the preamble is received without an SNR impact.

[30] Mobile WiMAX: Part I: A technical overview and performance evaluation. WiMAX Forum (2006)

2.10.1 EVM Test with a 24 MBPS 16-QAM Signal

Figures 2.60 and 2.61 show the measured EVM performance and constellations in the presence of the adjacent channel interferer. Both of the desired channel signal and interferer are generated from 24 Mbps 16-QAM signal and the channel power is set to −40 dBFS. When the blocker-to-signal amplitude ratio is below 10 dB, the ADC achieves −39 dB EVM in the normal mode against −33 dB EVM in the blocker suppression mode. However, in the blocker suppression mode, the ADC can tolerate a 30 dBc blocker with −22 dB EVM, fulfilling the system requirements.

The performances of the two modes are summarized in Table 2.4. The overall performances are little bit lower than transistor-level full-chip simulation results due to imperfect ground plane and also noise coupling on the printed circuit board.

2.11 Conclusions

In this chapter, sampling rate, dynamic range, and bandwidth adaptation techniques for pipelined and continuous-time ΣΔM ADCs have been introduced. The first part of this chapter presents a power scalable 12 b pipeline ADC that enables or disables OTAs connected in parallel to scale the settling response of Multiplying DAC (MDAC) and Sample/Hold (S/H) amplifiers in order to achieve constant SNDR performance over a range of sampling rates. The proposed technique facilitates optimal power consumption over the entire sampling rate range and reduces design complexity by maintaining constant DC bias conditions in the scaled analog blocks. The reduced design complexity allows for an earlier optimal design to be quickly reconfigured for changed specifications without requiring extensive re-design of the ADC analog core. In the second section an adaptive-blocker-rejection CT ΣΔ ADC for mobile WiMAX receivers, implemented on a 130 nm digital CMOS process, is demonstrated. A key contribution of this research is that it develops a new ADC architecture that can adaptively suppress interferers based on their power level at the ADC input. This topology can reduce the design requirements of analog-baseband filters and VGAs without increasing ADC DR, a challenging task in state-of-the-art deep-submicron technologies. Another contribution is the design of an analog domain blocker-detection circuit with negligible power consumption and die area overhead. The proposed blocker detector can ensure agile estimation of interferers' level and control the reconfigurable loop filter without significant latency. This guarantees stable operation of the modulator even with sudden high blockers applied. To validate this theory, the prototype ADC is tested with various blocking conditions. First, the ADC achieves 65 dB SNDR and 71 dB DR over a 10 MHz signal bandwidth with 250 MHz sampling frequency and 18 mW power consumption. With the integrated blocker detector, the measurement results show that reconfigurable operation can be

obtained to optimize the system performance based on the blocker level at the ADC input. In the normal mode, the modulator is optimized for highest quantization noise suppression, while the blocker-suppression mode is designed to minimize the blocker interference. In the blocker-rejection mode, the ADC can achieve 8 dB and 15 dB blocker attenuation at the adjacent and alternate frequencies, which is almost equivalent to a third-order LPF. This argument proves that the enhanced blocker-rejection performance can improve system selectivity and stability without a base-band channel-select filter, simplifying the receiver's architecture. The ADC can tolerate a 30 dBc blocker at 25 MHz offset with a −40 dBFS desired channel signal, achieving −22 dB EVM. Further, the measured transient behavior shows that the modulator can change its operation mode within 51 s under blocking condition, which is much shorter than the frame duration of the WiMAX standard.

Chapter 3
Current and Emerging Trends in the Design of Digital-to-Analog Converters

Sidharth Balasubramanian, Vipul J. Patel and Waleed Khalil

3.1 Introduction

The advent of digital computing, coupled with the continued scaling of CMOS devices, has led to the rapid growth in theory and applications of digital signal processing (DSP). In order to leverage the available increase in speed, complexity, and integration, high-performance analog-to-digital converters (ADCs) and digital-to-analog converters (DACs) are needed to ensure the highest signal fidelity when moving between the analog and digital domains. Traditionally, high-speed ADCs have attracted more attention in the research and scientific community than their DAC counterparts, mainly due to the inherent shift towards receivers in the study and analysis of radio systems [1].

This chapter aims to describe some of the design challenges and emerging trends for high-speed and high-resolution digital-to-analog converters. Section 3.1 presents an overview of the digital-to-analog conversion process. Section 3.2 delves into DAC characterization by outlining different sources of error and metrics used to quantify the DAC performance. A summary of DAC topologies and circuit limitations in the context of current-steering (CS) DACs is provided in Sects. 3.3 and 3.4, respectively. Section 3.5 details four major considerations in the design space of CS DACs. Realizing that the segmented architecture is the de facto standard in high-resolution DACs, a supplemental approach to segmentation is presented in Sect. 3.6. An in-depth survey of current and emerging architectural trends in high-performance DACs is discussed in Sect. 3.7. Finally, Sect. 3.8 concludes with a summary and a brief discussion on future directions.

S. Balasubramanian · W. Khalil (✉)
Department of Electrical and Computer Engineering, The Ohio State University, Columbus, OH 43210, USA
e-mail: khalil@ece.osu.edu

V. J. Patel
Air Force Research Laboratory, Wright-Patterson AFB, Dayton, OH 45433, USA
e-mail: Vipul.Patel@us.af.mil

P. Carbone et al. (eds.), *Design, Modeling and Testing of Data Converters*, Signals and Communication Technology, DOI: 10.1007/978-3-642-39655-7_3,

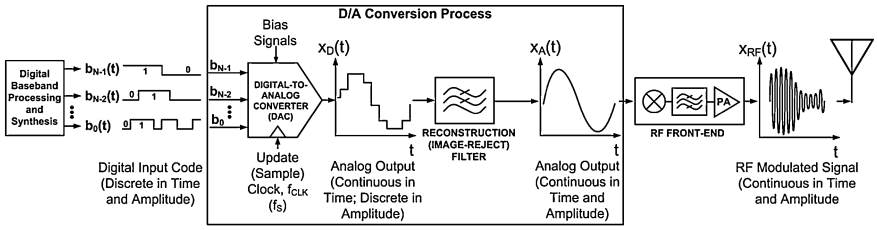

Fig. 3.1 Conventional transmitter block diagram

Figure 3.1 illustrates a conventional transmitter block diagram where the DAC is shown as the fundamental building block for waveform synthesis. Preceding the DAC, a digital baseband processor is used to generate N digital bits $(b_{N-1}(t), b_{N-2}(t), \ldots, b_0(t))$ that represent the waveform to be synthesized. The DAC and reconstruction filter then translate the digital input codes to an analog waveform, $x_A(t)$. Following the filter, $x_A(t)$ is upconverted to the desired RF band and amplified as represented by $x_{RF}(t)$.

The digital to analog translation process involves weighting and summing of voltages, charges, or currents derived from the input digital codes. A representative analog value, typically in the voltage domain, is then produced, where the full scale voltage (V_{FS}) is defined as the difference between the maximum and minimum voltage levels. Note that for current-steering DACs, the output current is converted to a voltage using a resistive load. The simplified representation of an N-bit DAC can be described as having N binary input bits defined by the following vector:

$$\hat{B} = \{b_{N-1}, b_{N-2}, b_{N-3}, \ldots, b_1, b_0\} \tag{3.1}$$

where $b_i \in \{0, 1\}$, b_{N-1} is the most significant bit (MSB), and b_0 is the least significant bit (LSB). The binary vector, $\hat{B}$, is converted to a corresponding decimal value, D:

$$D = \sum_{i=0}^{N-1} 2^i b_i \tag{3.2}$$

This weighted decimal value is then multiplied by a gain factor, such as V_{LSB} (where $V_{LSB} = V_{FS}/2^N$) for voltage-or charge-based DACs and I_{LSB} (where $I_{LSB} = I_{FS}/2^N$) for current-steering DACs, to yield the final analog voltage (or current):

$$V_{OUT}(D) = D \cdot V_{LSB} \tag{3.3}$$

$$I_{OUT}(D) = D \cdot I_{LSB} \tag{3.4}$$

The high-level architecture of a digital-to-analog converter can be conceptualized into two basic functions: (1) zero-order holding and (2) digital-to-analog translation, as illustrated in Fig. 3.2. Assuming an ideal sampling process, the

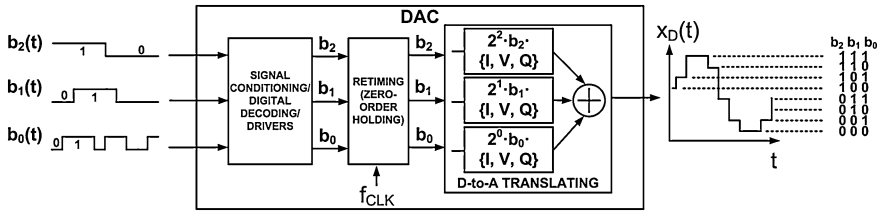

Fig. 3.2 Basic functions in a 3-bit digital-to-analog converter

DAC is fed with data in the form of impulse trains. However, finite switching time in real circuits makes it impossible to generate these impulses. Instead, the input signal is supplied as square-wave pulses, where a retiming register is often needed to ensure all digital input bits are aligned and synchronized to an update clock, f_{CLK} $(1/T_{CLK})$. This process of square waveform generation through holding the input for the duration of T_{CLK} is known as the zero-order hold (ZOH). The next step involves the actual translation from the digital to the analog domain. The converter circuit assigns analog (i.e. current, voltage, or charge) weights corresponding to the digital input code, and then sums them up to the final discrete (stair-step) output, $x_D(t)$. The reconstruction filter, also known as the image-reject filter, is typically an external component that smoothens the stair-step waveform by eliminating out-of-band frequencies. For signals generated at baseband, a low-pass filter is designed to eliminate frequencies greater than the Nyquist bandwidth (DC to $f_{CLK}/2$). However, in the case of intermediate signal generation beyond the first Nyquist zone, a bandpass filter is utilized.

3.2 Characterizing the DAC

In general, DACs are known to suffer from four inherent limitations: (1) quantization or truncation error, (2) image replicas, (3) nonlinear spurs, and (4) hold distortion. These limitations are illustrated in Fig. 3.3.

The finite resolution of the DAC results in inherent quantization noise that ultimately sets the minimum noise floor. Another inherent limitation in the DAC is attributed to its sampling nature. Assuming a DAC clocked at f_{CLK} is used to synthesize an output signal at f_0, image replicas are generated at $f_{CLK} \pm f_0$, $2f_{CLK} \pm f_0$, $3f_{CLK} \pm f_0$ and so on. Similar to an ADC, a DAC's output spectrum is divided into Nyquist zones defined at $nf_{CLK}/2$ where $n = 1, 2, 3, \ldots$, as illustrated in Fig. 3.3. When the generated analog signal approaches a Nyquist zone, the signal and its corresponding replica are close and comparable in magnitude. Hence, the replica acts as a strong interferer to the signal of interest. This requires a brickwall reconstruction filter and thus restricts the DAC signal generation to well below the Nyquist frequency. Another major limitation involves the nonlinear behavior in the DAC, which is well pronounced when the desired signal approaches the boundary of the first Nyquist zone. This results in high levels of harmonic and intermodulation

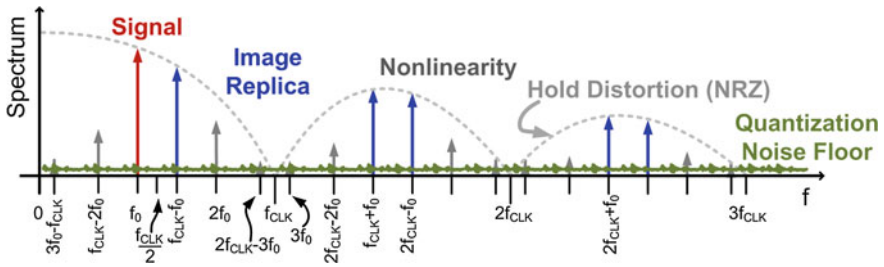

Fig. 3.3 Inherent DAC limitations

distortion products that can interfere with the desired band. Furthermore, the harmonic mixing amongst the DAC's update clock, the desired signal, and the distortion products, results in spurious components at $mf_{CLK} \pm nf_0$, for all integer m, n. Ideally, the signal and its image replicas are of equal magnitude at all points in the frequency spectrum. However, due to the imposed ZOH, the signal, image replicas, and distortion products are attenuated with a $\sin c$ response as described by (5), where D is the ON period during T_{CLK}. This effect is referred to as hold distortion. While there are several zero-order hold variations, the most often used is a non-return-to-zero (NRZ), where the DAC output is held for the entire duration of T_{CLK} (i.e. $D = 100$ %). However, the fast roll-off of the *sinc* response limits operation past the first Nyquist zone. Other hold techniques such as return-to-zero (RZ) can extend operation to multiple Nyquist zones. For instance, reducing the hold duty cycle to 50 % can extend the first *sinc* null to $2f_{CLK}$. Figure 3.4 illustrates both of these ZOH techniques in the frequency and time domains.

$$H(f) = \frac{\sin\left(\pi D \frac{f}{f_{CLK}}\right)}{\pi \frac{f}{f_{CLK}}} \exp\left(-j\pi D \frac{f}{f_{CLK}}\right) \tag{3.5}$$

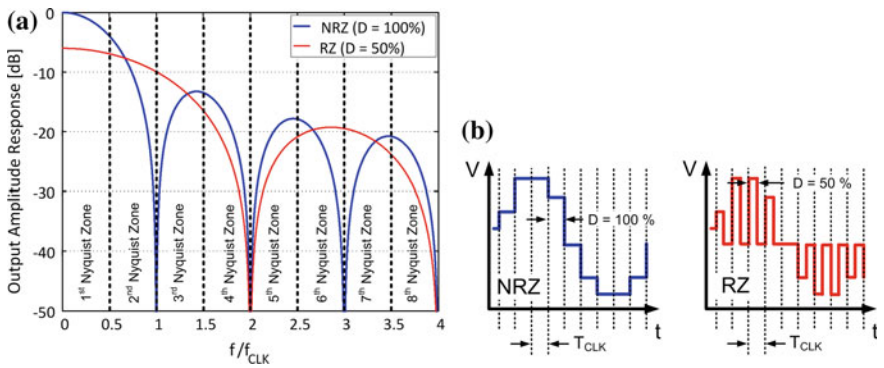

Fig. 3.4 **a** Common zero-order hold (ZOH) distortions segmented into Nyquist zones **b** NRZvsRZ Time domain ZOH for NRZ and RZ

3.2.1 Timing and Amplitude Errors

In addition to the aforementioned DAC limitations, there exist other numerous forms of DAC nonidealities which can be attributed to the physical circuit implementation. Specifically, these nonidealities can be lumped into two broad categories: timing-related errors and amplitude-related errors. While not seen as independent, each of the two categories can be further classified into static and dynamic errors. In basic terms, static refers to time-invariant errors that are induced by random or systematic mismatch effects. In contrast, dynamic refers to time-variant errors that can be attributed to code-dependent[1] transitions, jitter, glitches, and impedance variation. Figure 3.5 illustrates the four categories of error-static and dynamic timing as well as static and dynamic amplitude errors.

Examples of static timing errors Fig. 3.5a can be observed in delay mismatches amongst retiming flip-flops, clock skew between DAC circuit blocks, and delays attributed to switching pair mismatches. On the other hand, dynamic timing error

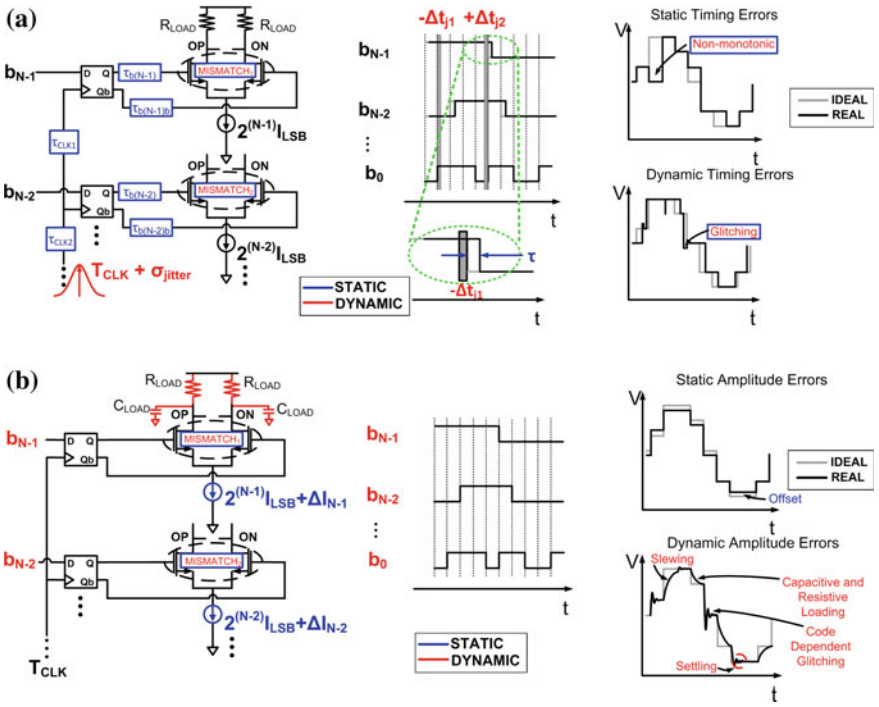

Fig. 3.5 **a** Static and dynamic timing errors **b** Static and dynamic amplitude errors

[1] Code-dependency is equivalently mentioned as signal dependency, where signal refers to the desired output waveform.

includes both random and deterministic clock jitter or phase noise. Similar to quantization noise, random jitter can increase the DAC's noise floor, while deterministic jitter is manifested as spurs in the DAC's output spectrum. Examples of amplitude-related errors are illustrated in Fig. 3.5b. The relative mismatch between the weighted current sources can induce static amplitude errors in the form of differential and integral nonlinearities as well as offset error. Whereas dynamic errors caused by large code transitions, parasitic loading, finite slewing, and finite settling times, further degrade the output amplitude accuracy. Timing-related errors can also be induced concurrently with amplitude-related errors resulting in varying settling times, slewing rates, and glitches.

3.2.2 Performance Metrics

In general, the performance of any given DAC can be quantified using a set of static and dynamic metrics. The choice of metrics depends on the desired application ranging from high precision systems such as potentiometers and medical instrumentation to waveform synthesis in high-speed communication systems. This section briefly introduces the commonly used metrics to characterize the DAC performance [2–5].

3.2.2.1 Static Metrics

Unlike the ADC's stair-step transfer function, the DAC's response is represented by discrete points, which maps a specific digital input code to a discrete analog value. The transfer function of the real DAC and its comparison to an ideal transfer function are used to evaluate the static (i.e. near DC) performance metrics. Generating the transfer function plot from a real DAC is a straightforward process by simply applying a digital input code and observing the output using a high-precision voltmeter. For simplicity, the transfer function of a real 3-bit DAC is overlaid on the ideal response as depicted in Fig. 3.6. The DAC static performance metrics, including offset, gain, monotonicity, differential nonlinearity, and integral nonlinearity errors can be extracted from the transfer function plots.

Offset and Gain Errors

A typical DAC transfer function is depicted in Fig. 3.6a. The DAC's offset and gain errors can be extracted from the transfer function using various methods. These include dividing the full scale voltage range by the number of quantization levels, using the end-points to generate a linear fit, or employing the best fit line. Owing to its simplicity, the end-point fit is the most preferred method to measure the DAC's offset [2]. The straightforward method to determine the DAC offset is

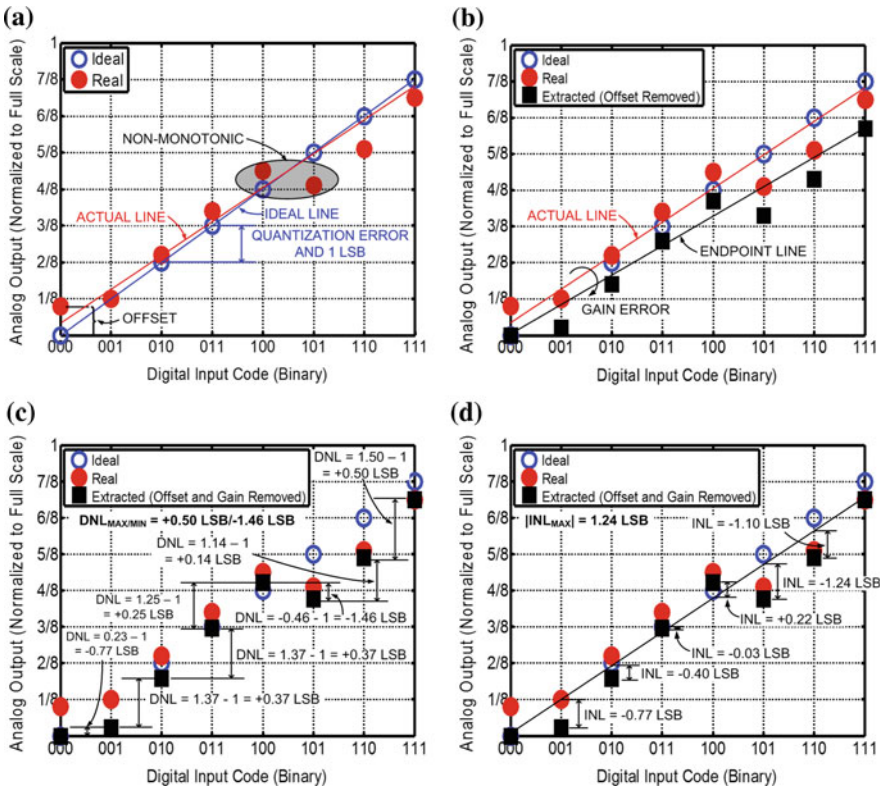

Fig. 3.6 3-bit DAC Transfer Function: **a** Offset error **b** Gain error **c** DNL **d** INL

by calculating the deviation between the real and ideal transfer functions when the binary input is all zeros. As illustrated in Fig. 3.6a, the y-intercept of the transfer function denotes the offset error. For target applications such as waveform synthesis, the DC offset can result in large carrier feedthrough, when upconverted in an RF transmitter.

After removal of the offset, the gain error is extracted from the deviation of the slope of the extracted transfer function versus the slope of the ideal transfer function ($y = x$), as depicted in Fig. 3.6b [5]. The gain error is seen as less critical, since it is often calibrated out by adjusting the input digital code. It is worth noting that both offset and gain errors need to be removed before extracting any further static metrics such as differential or integral nonlinearities.

Differential Nonlinearity (DNL)

The DNL error measures the step distance between the code i and the code $i - 1$ for the extracted DAC transfer function (after removal of offset and gain errors), as

illustrated in Fig. 3.6c. The difference is then compared to the ideal LSB value. The DNL for a given code i can be calculated as

$$DNL(i)[LSB] = (Code_i[LSB] - Code_{i-1}[LSB]) - 1[LSB] \tag{3.6}$$

For a given DAC, the minimum and maximum DNL values are typically reported to summarize its performance. The DAC is considered monotonic when the output signal is increasing (decreasing) as the digital input code is increasing (decreasing). Monotonicity is guaranteed if the minimum value of the DNL is greater than −1 LSB.

Integral Nonlinearity (INL)

The INL error can be quantified as the deviation of the extracted DAC transfer curve to the end-point line, as depicted in Fig. 3.6d. The INL for a code i can be calculated from the DNL as

$$INL(i)[LSB] = \sum_{k=0}^{i} DNL(k)[LSB] \tag{3.7}$$

In order to summarize the INL performance of a DAC, the absolute maximum INL is reported.

3.2.2.2 Dynamic Metrics

The DAC's dynamic performance can be inferred from its output spectrum. The most often used metrics to characterize the DAC's dynamic performance are SNR, SINAD, ENOB, SFDR, and IMD. It is worth mentioning that all of these metrics are typically measured within the desired Nyquist band of operation. A single-tone test is employed when measuring SNR, SINAD, ENOB, and SFDR, while a two-tone test is used to characterize IMD. A simulation of a 12-bit DAC with intrinsic nonlinearity is used to illustrate the extraction of these metrics in Fig. 3.7.

Signal-to-Noise Ratio (SNR)

Signal-to-noise ratio is defined as the ratio of the desired signal power (P_{Signal}) to the integrated noise power, excluding harmonics and DC offset. Typically, the specified noise power includes quantization noise (N_q), DNL error (N_{DNL}), thermal noise ($N_{Thermal}$), and random jitter (N_j).

$$SNR[dB] = 10\log_{10}\left(\frac{P_{Signal}}{N_q + N_{DNL} + N_{Thermal} + N_j}\right) \tag{3.8}$$

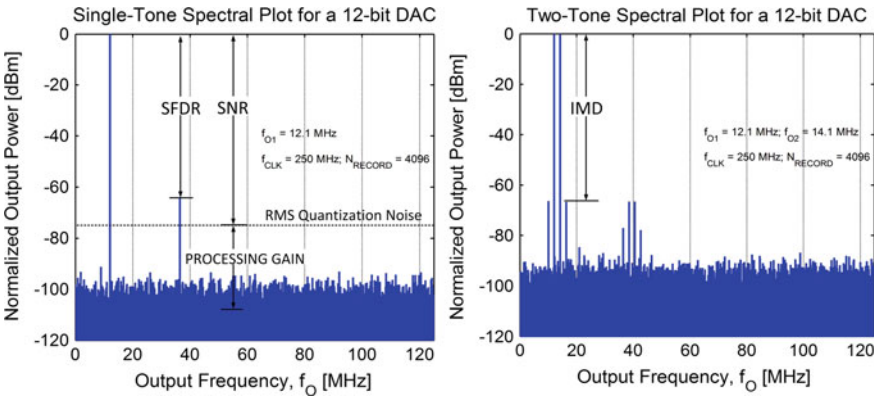

Fig. 3.7 Spectral plots for a 12-bit DAC

The SNR is generally specified over the entire Nyquist bandwidth. However, in some applications a narrow band filter is used following the DAC, and thus sets the integrated noise bandwidth well below its Nyquist. This process is otherwise known as oversampling and can effectively enhance the DAC resolution beyond its quantization limit.

Harmonic Distortion (HD*n*)

The *nth* order harmonic distortion is defined as the ratio between the power of the desired signal and the power of the *nth* harmonic, where $n = 1, 2, 3, \ldots$, and expressed as,

$$HDn[dBc] = 10\log_{10}\left(\frac{P_{Signal}}{n\text{th Harmonic Power}}\right) \tag{3.9}$$

When generating a single output tone at f_0, the *nth* harmonic component is observed at the $|nf_0 \pm kf_{CLK}|$ frequency where k is chosen to fold the harmonic term into the desired Nyquist zone.

Signal-to-Noise-and-Distortion Ratio (SINAD, SNDR)

Signal-to-noise-and-distortion ratio measures the ratio of the power of the desired signal to the power of the total noise, including harmonic distortion products ($P_{Distortion}$). The measurement does not include the DC component.

$$SINAD\,[dB] = 10\log_{10}\left(\frac{P_{Signal}}{N_q + N_{DNL} + N_{Thermal} + N_j + P_{Distortion}}\right) \tag{3.10}$$

Effective Number of Bits (ENOB)

ENOB is used to represent the effective resolution of the converter including all sources of noise and/or distortion. ENOB can be calculated from either SNR or SINAD and is represented as

$$ENOB[bits] = \frac{(SNR \text{ or } SINAD)[dB] - 1.76}{6.02} \tag{3.11}$$

Spurious-Free Dynamic Range (SFDR)

Spurious-free dynamic range measures the relative power of the desired signal to the power of the highest spur component generated within the targeted bandwidth.[2]

$$SFDR\,[dBc] = 10\log_{10}\left(\frac{P_{Signal}}{\text{Highest Spur Power}}\right) \tag{3.12}$$

This metric is considered the most critical in frequency synthesis since it determines the spectral purity of the output waveform with or without the presence of harmonics.

Intermodulation Distortion (IMD*n*)

In the presence of two or more input signals, inter-tone harmonic mixing can result in intermodulation distortion (IMD) products, located close to or further apart from the desired signals. IMD*n* measures the ratio of the power of the desired signals (P_{Signal}) to the power of the *nth*-order intermodulation product (P_{IMD_n}). The IMD products for two signals at f_{01} and f_{02} can span across $|(mf_{01} \pm nf_{02})|$ where $m, n = 1, 2, 3, \ldots$ Furthermore, the DAC sampling and aliasing processs can result in the folding of the IMD products to multiple Nyquist zones, which can be expressed as $|(mf_{01} \pm nf_{02}) \pm kf_{CLK}|$, where $m, n, k = 1, 2, 3, \ldots$

$$IMD_n = 10\log_{10}\left(\frac{P_{signal}}{P_{IMD_n}}\right) \tag{3.13}$$

In general, the third order intermodulation (IMD_3) is of most concern, as it generates the highest in-band distortion levels.

[2] Care should be taken in choosing appropriate FFT resolution bandwidth (or bin spacing) to set the minimum detectable power level.

3.2.3 INL Induced Distortion

Even though the previously described metrics can be divided into static and dynamic categories, their effects are not entirely independent. For instance, there is a clear link between the DAC static INL performance and its low frequency distortion behavior. A high INL error represents a large deviation of the DAC transfer curve from the straight line ($y = x$). This places an upper bound on SFDR at frequencies near DC [5, 6]. Thus, the INL can provide an estimate of the maximum SFDR before further degradation due to timing- and amplitude-related errors [2].

Continuing with the example of a 12-bit, 250 MS/s DAC, we estimate the prominent harmonic distortion order and its magnitude from the shape and maximum value of the INL, respectively. Expanding on the works of [5, 6], the INL is modeled using correlated second- and third-order polynomials. The second-order model is defined as $y = a \cdot x^2 + x - a$, while the third-order model is expressed as $y = a \cdot x^3 + (1 - a) \cdot x$. Here a is chosen such that the desired INL $_{MAX}$ is satisfied. Figure 3.8 illustrates the second- and third-order INL curves and their respective spectra with $\mathrm{INL}_{MAX} = 0.5$, 1, and 2 LSB. For each example of INL, the magnitudes of HD2 and HD3 are comparable. From the analyses of these two models, a heuristic approximation for SFDR as a function of the maximum INL can be expressed as

$$SFDR\,[dBc] \cong 20\log_{10}\left(\frac{2^N}{INL_{MAX}}\right) \tag{3.14}$$

3.3 DAC Implementation

The most straightforward implementation of the DAC involves an array of binary-weighted passive (capacitors and/or resistors) or active (current sources) components. Assuming a binary-weighted current-steering DAC with an LSB current of

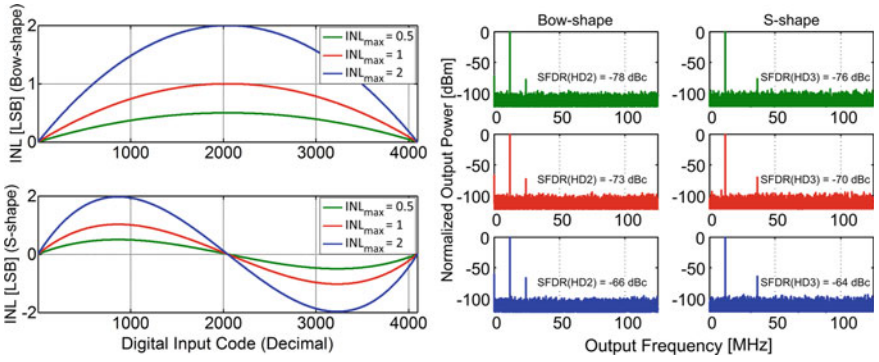

Fig. 3.8 Static INL shaping and the resulting effects on power spectral density

I_{LSB}, and denoting b_i[3] as the *ith* binary bit of a digital code, the output of the N-bit binary-weighted DAC can be expressed as

$$I_{OUT} = I_{LSB} \sum_{i=0}^{N-1} 2^i b_i \tag{3.15}$$

Alternatively, in a unary-weighted DAC, the current sources are all equal in magnitude; Thus, an N-bit DAC comprises $2^N - 1$ unary current sources. Denoting t_i[4] as the *ith* thermometer bit of a digital word, the effective analog current in response to the digital thermometer code is expressed as

$$I_{OUT} = I_{LSB} \sum_{i=0}^{2^N-2} t_i \tag{3.16}$$

Both the aforementioned architectures have their advantages and disadvantages. The binary architecture carries the benefit of using fewer control signals than the unary architecture. However, the accuracy requirements of the MSB cell versus LSB cell increases exponentially with the resolution. This results in the potential to exhibit code-transition glitches and loss of monotonic behavior. Such nonidealities can be mitigated by the unary architecture at the expense of increased chip area. A compromise between the two architectures is often made by segmenting the DAC, i.e. the MSBs and LSBs are represented by unary and binary structures, respectively. For an N-bit DAC segmented as $k : m$, such that the first k bits (MSBs) are realized as a unary structure, and the lower m bits are represented in binary, the effective analog output current is given by

$$I_{OUT} = I_{LSB} \sum_{i=0}^{m-1} 2^i b_i + 2^m I_{LSB} \sum_{i=0}^{2^k-2} t_i \tag{3.17}$$

For instance, a 12-bit DAC built using a 9-bit thermometer array and a 3-bit binary array is said to be 75 % segmented. The unary and binary architectures are two extreme cases of segmentation: a unary-weighted DAC is referred to as 100 % segmentation, while an all-binary DAC is 0 % segmented.

3.4 High-Speed DAC: Circuits and Limitations

A majority of high-speed DACs use the current-steering architecture, which offers faster switching and wider bandwidths compared to voltage- or charge-based DACs. This is primarily because the active devices are well known to switch faster in current

[3] b_i takes discrete values of 0 or 1 and referred in little-endian format.

[4] t_i takes discrete values of 0 or 1 and referred in little-endian format.

than in voltage mode. In addition, attempts to linearize the output buffer amplifier in voltage and/or charge mode DACs using feedback techniques, limit their speed of operation. This can be contrasted with using a simple load resistor in CS DACs.

An illustration of the current-steering architecture on a high-level abstraction is seen in Fig. 3.9. The DAC core comprises an array of binary and/or unary weighted current sources. The binary-switched current source array is scaled in units of $2^k \times I_{LSB}$, where $k = 0, 1, 2, \ldots, (N - m - 1)$ for an N-bit DAC segmented with m thermometer bits. The unary current source array, which is only used in thermometer or segmented DAC architectures, comprises 2^{m-1} current sources, each of magnitude $2^{N-m} \times I_{LSB}$. A corresponding array of current-commutating switch-pairs steer the direction of the current into one of the differential legs[5] of the DAC's output based on the input digital code. The switching pair cells can also be used to implement various hold operations at the output. Two load resistor cells, typically 50 Ω each, are used to convert the DAC's differential output current to a voltage signal (I–V). A binary-to-thermometer encoder maps the m most significant binary bits to a thermometer code that feeds the unary current source array.

In the context of high-speed DACs, it is important to examine the role and limitations of each component in the current-steering cell. A simple circuit schematic of a DAC current-steering cell is illustrated in Fig. 3.10. Transistors M_1 and M_2 constitute the current source and are normally biased from a current mirror reference cell. Transistor M_1 is the critical transistor that determines the magnitude of the cell current. However, the finite output impedance of M_1 affects the accuracy of the current source, thus forcing the need for the cascoding transistor, M_2. The source current is then steered to the positive or negative output leg in response to the input differential data signals, D and DB, by means of the commutating switch pair, ($M_{3,4}$). The size of the switch pair is scaled with the magnitude of the current to maintain the same source node voltages across all current cells. This is seen as critical to maintain an N-bit accuracy of the cascoded current source. If the output of the DAC is taken directly at the drain of the switching pair, there exists a data-to-output feedthrough resulting in high levels of switching glitches. In addition, the large aspect ratios and small channel lengths of the switching pair result in high load capacitance and low output impedance, respectively. Together, this limits the linearity performance of the DAC and hence cascode transistors ($M_{5,6}$) are added to isolate the output node from the gate of the switching pair.

3.5 DAC Design Space

In general, scaling of transistors, interconnect dimensions and power supply are quite favorable for digital designs. However, such trends are not entirely beneficial to analog and mixed-signal circuits. While the DAC in Fig. 3.9 exhibits a degree of repeatability that lends itself to an automated design methodology, the designer is

[5] Some books also use the term 'arm' as an equivalent to 'leg'.

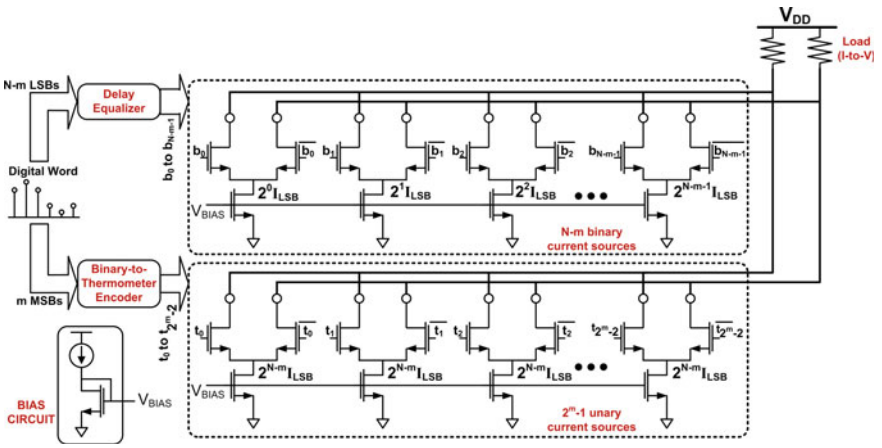

Fig. 3.9 A segmented current-steering DAC architecture

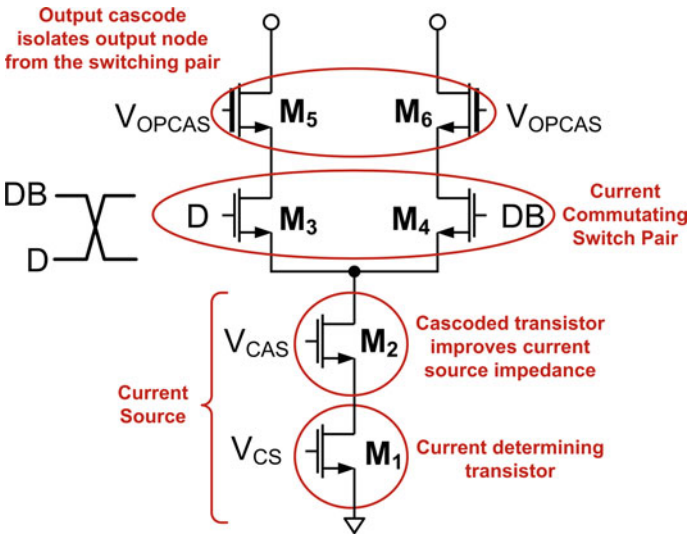

Fig. 3.10 A conventional current-steering cell

confronted with a complex design space and forced to resort to a custom design flow. To this end, a number of process-related limiting factors affect the performance of the DAC, resulting in failure to meet the target specifications. The design space of a DAC can be highlighted in terms of four major limitations: device noise, output impedance, signal swing and switching speed. A successful DAC design can only be achieved by carefully optimizing across this space to meet the desired specifications. The remainder of this section will address the DAC design space in detail.

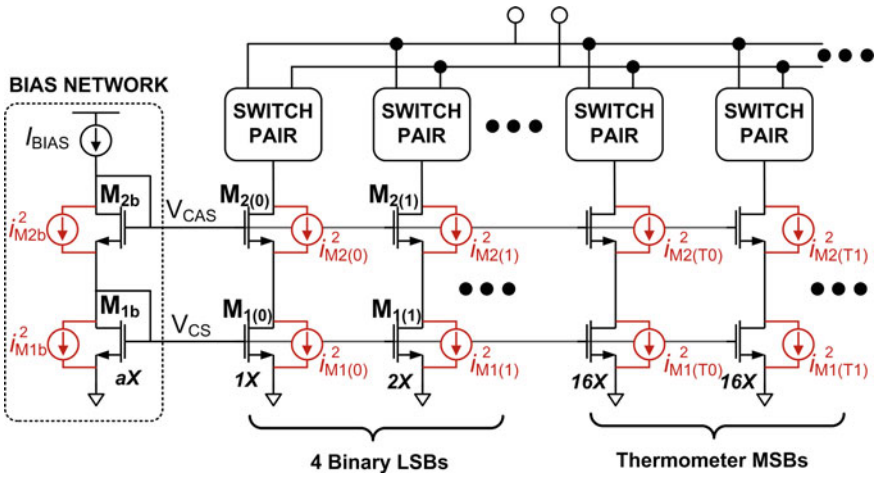

Fig. 3.11 Noise sources in a DAC

3.5.1 Device Noise

In addition to the intrinsic quantization noise, the DAC performance is also limited by the noise contribution from various circuit elements. The DAC's target resolution and desired bandwidth together set the maximum tolerable noise floor. In CS DACs, the current source array is the major contributor of noise. In addition to its own noise contribution, noise induced by the reference bias is further magnified by the current mirroring action, and thus can limit the overall noise performance. Figure 3.11 illustrates the noise sources in the DAC's circuit and its associated bias network. An $a:1$ current mirroring ratio is assumed between the bias network and the LSB cell. Accounting for the bias thermal noise contribution, the DAC's total output noise can be given by

$$i^2_{d_{DAC}} = 4kT\frac{\gamma}{\alpha}g_{m_{M1(0)}}2^N\left(1+\frac{2^N}{a}\right)\Delta f \tag{3.18}$$

where γ and α are process-defined noise parameters [7], and $g_{m_{M1(0)}}$ is the transconductance of the current source, $M_{1(0)}$.

Figure 3.12 illustrates the total output noise of a 12-bit DAC along with the isolated contribution of the bias network, as a function of a.[6] It is observed that the bias noise is the major contributor to the total output noise. In order to reduce the impact of the bias noise, a needs to be set much larger than 2^N. However, the power consumption specifications limit the maximum value of a that can be used; i.e. for a given LSB current and a bias mirroring ratio a, there is an expense of a times the LSB current in the bias cell.

[6] The flicker noise has been removed in the simulation.

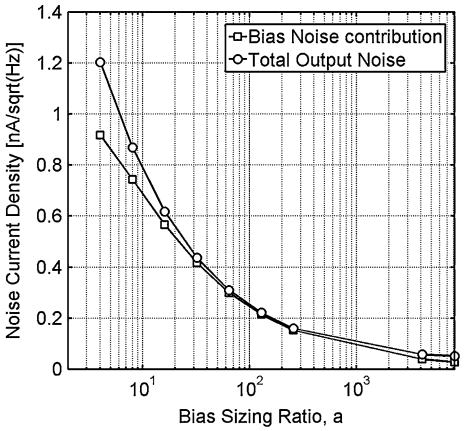

Fig. 3.12 Total output thermal noise contribution in a 12-bit DAC for various bias sizing ratios at 100 kHz

Assuming a full-scale output sinusoid current, the RMS signal power is given as

$$\text{Signal Power} = \frac{(2^N I_{LSB})^2}{2}. \tag{3.19}$$

Thus, the thermal SNR of the DAC over a bandwidth B is expressed as[7]

$$\text{Signal-to-Thermal Noise Ratio} \cong 10\log\left(\frac{aI_{LSB}(V_{GS}-V_T)\alpha}{16kT\gamma B}\right) \tag{3.20}$$

where $V_{GS} - V_T$ refers to the overdrive voltage of the current source transistor, $M_{1(0)}$. The SNR is seen to be independent of the resolution of the DAC, and can be only improved by increasing the bias mirroring ratio, a, or the LSB cell current, I_{LSB}. Let us consider a 12-bit DAC having an effective bandwidth of 100 MHz. The signal-to-quantization noise ratio is 74 dB. If the thermal noise floor is desired to be at least 10 dB below the quantization noise floor, and assuming $a = 1024$, $V_{GS} - V_T = 100$ mV and noise parameters, $\gamma = 2/3$, $\alpha = 1$, the minimum bound for LSB current is set to be 10.76 μA. The noise specification and bias sizing ratio together limit the minimum current (LSB cell) that may be used in the DAC. Along with the resolution and swing, it also sets the minimum achievable static current consumption of the DAC (including bias) in a given process technology.

3.5.2 Output Impedance

The static performance of the DAC is dependent on the intrinsic accuracy of the current sources.[8] Another element that can degrade the static performance is the

[7] The noise contribution of the DAC core is assumed negligible compared to the bias noise.

[8] This will be described later in detail in Sect. 3.6.

finite DC impedance ($Z_{CS}(0)$) of the current sources. In contrast, the DAC dynamic behavior is strongly dependent on the output impedance of the unit cell ($Z_{CELL}(s)$). In newer process technologies, channel length modulation effects are further exacerbated, significantly limiting the output impedance of a single transistor. Consider the current source in Fig. 3.13, comprised by transistors M_1 and M_2. Let us denote $Z_{DAC}(s)$ to be the effective parallel impedance looking into the DAC array. As the DAC cells are turned on, more cells are added in parallel, thus reducing $Z_{DAC}(s)$. Assuming an all-unary DAC, the total output impedance pertaining to the n^{th} code is given by

$$Z_{DAC}(s) = \frac{Z_{CELL}(s)}{n} \tag{3.21}$$

Accounting for the load impedance, R_L, the differential output voltage can be expressed as

$$V_{OUT} = I_{LSB} R_L \left\{ \frac{n}{1 + n\frac{R_L}{Z_{CELL}(s)}} - \frac{2^N - n - 1}{1 + (2^N - n)\frac{R_L}{Z_{CELL}(s)}} \right\} \tag{3.22}$$

It is observed that the total output impedance of the DAC changes as a function of the input code.

Such an effect is termed code-dependent impedance modulation, and is one of the fundamental factors limiting the distortion performance of a DAC. In the event of finite $|Z_{DAC}(0)|$, the static transfer characteristics become nonlinear. For instance,

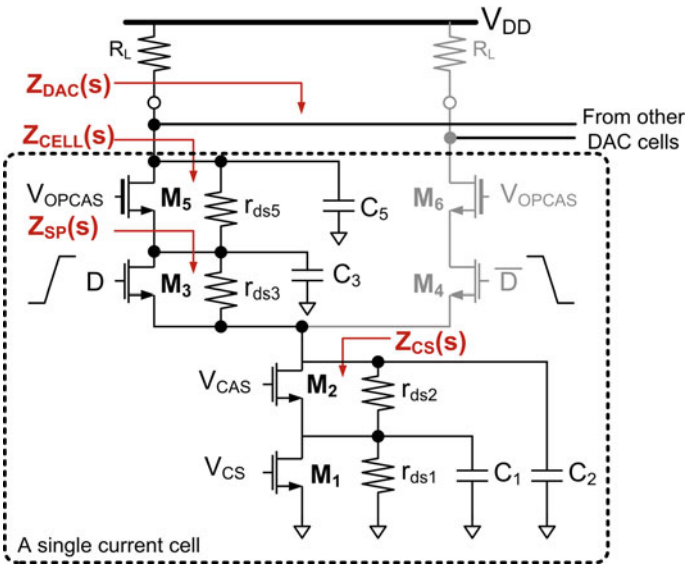

Fig. 3.13 Impedances in a current-steering cell

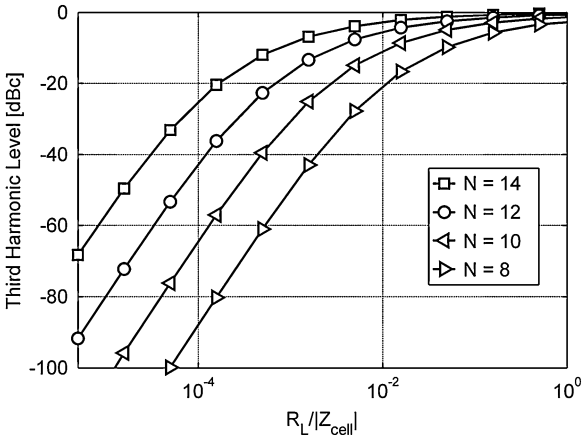

Fig. 3.14 HD_3 performance vs. $R_L/Z_{CELL}(0)$

a 12-bit DAC with an $R_L/|Z_{CELL}(0)|$ of 5×10^{-5} results in an HD_3 of about -53 dBc at low frequencies. This distortion level is significantly higher than the intrinsic performance of the DAC. Figure 3.14 illustrates the simulated HD_3 as a function of $R_L/|Z_{CELL}(0)|$ for various resolution. It is observed that the increase in resolution places more stringent requirement on the cell impedance.

The current cell in Fig. 3.13 is analyzed using a simple equivalent circuit model. In the absence of cascode transistors $M_{2,5}$, the DAC impedance is approximately $g_{m3}r_{ds3}r_{ds1}$. Since the switch pair is typically implemented using a minimum length device, it does not offer enough cascoding gain ($g_{m3}r_{ds3} \leq 1$). Thus, the DAC output impedance becomes a direct function of the current source impedance, r_{ds1}, and an R_L/r_{ds1} is not sufficiently small to attain high linearity. Cascoding is hence widely adopted to improve the impedance characteristics of the current source at the cost of voltage headroom. The cascoded current source impedance ($Z_{CS}(s)$) is analytically expressed as

$$Z_{CS}(s) = \frac{r_{ds1} + r_{ds2} + g_{m2}r_{ds1}r_{ds2} + sC_1r_{ds1}r_{ds2}}{1 + s(C_1 + C_2)r_{ds1} + sC_2r_{ds2} + sC_2g_{m2}r_{ds1}r_{ds2} + s^2C_1C_2r_{ds1}r_{ds2}} \quad (3.23)$$

Figure 3.15a illustrates the improvement in the low frequency impedance of the current source as a function of the ratio of the lengths of the transistors M_2 to M_1, for various M_1 lengths. While it is observed that a large channel length for M_1 improves the output impedance, increasing the length of M_2 further enhances the cascoding effect, resulting in output impedances in the order of tens of megohms. This significantly aids in achieving high static accuracy for the current sources. However, increasing transistor length, while maintaining its aspect ratio, increases the drain capacitances (C_1 and C_2), thus degrading the impedance at high frequencies (hundreds of megahertz). Figure 3.15b illustrates the impact of increasing channel lengths for M_1 and M_2 on the high frequency output impedance. Such opposing effects of increasing channel lengths call for an optimal choice to be

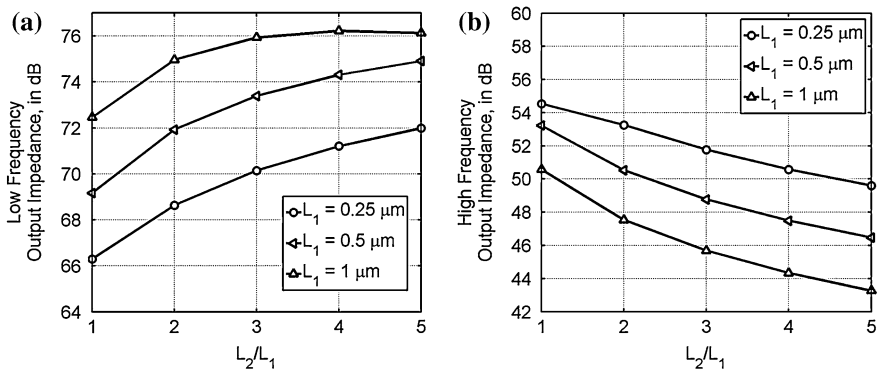

Fig. 3.15 **a** Low-frequency (DC) output impedance ($Z_{CS}(0)$) **b** High-frequency output impedance (Z_{CS} at 100 MHz)

considered when sizing the current sources, in order to strike a balance between the high and low frequency distortion limits.

On moving up the DAC cell, the impedance looking at the drain of the switch pair (when it is ON) is given by

$$Z_{SP}(s) = \frac{Z_{CS}(s) + r_{ds3} + g_{m3}r_{ds3}Z_{CS}(s)}{1 + sC_3(r_{ds3} + Z_{CS}(s)) + sC_3g_{m3}r_{ds3}Z_{CS}(s)} \tag{3.24}$$

In the case of high-resolution DACs (i.e. 10 bits or higher), Z_{SP} is not large enough to mitigate the effect of code-dependent impedance modulation. Furthermore, the large gate-drain capacitances of the switch pair ($M_{3,4}$) result in significant data feedthrough to the output. Cascoding transistors M_5 and M_6 are thus added to improve the isolation between the output node and the gates of the switching pair. Accounting for the cascode pair, the impedance of a single DAC cell (from Fig. 3.13) can be written as

$$Z_{CELL}(s) = \frac{Z_{SP}(s) + r_{ds5} + g_{m5}r_{ds5}Z_{SP}(s)}{1 + sC_5(r_{ds5} + Z_{SP}(s)) + sC_5g_{m5}r_{ds5}Z_{SP}(s)} \tag{3.25}$$

At low frequencies, the cell impedance reduces to

$$Z_{CELL}(0) \cong g_{m2}g_{m3}g_{m5}r_{ds1}r_{ds2}r_{ds3}r_{ds5} \tag{3.26}$$

As transistor feature size decreases, the channel length modulation parameter shows a dependence on the bias current and the overdrive voltage of the transistor [8]. Figure 3.16 illustrates the simulated impedance of the ON leg of a DAC cell, as a function of current in a 90 nm process technology. The use of cascoding in both the current source and the current cell, indeed improves the static linearity of the DAC by further increasing the output impedance $Z_{CELL}(0)$ enough to guarantee at least 12-bits of accuracy. The kink in impedance at low currents is attributed to the dependence of the switch pair channel length modulation parameter on the

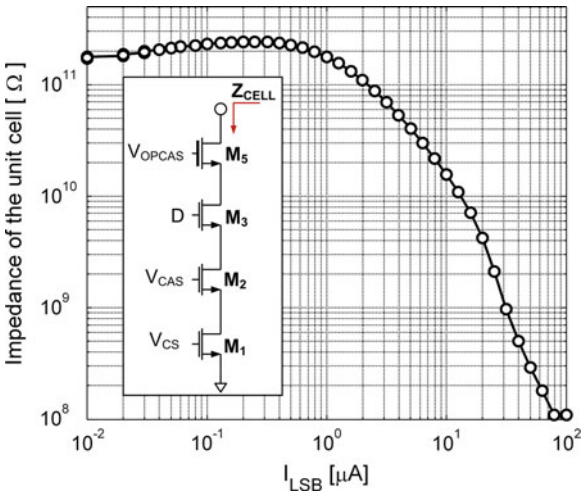

Fig. 3.16 DAC cell impedance as a function of current

overdrive voltage [8]. The distortion metrics in Fig. 3.14 can be used to determine the desired cell impedance. Figure 3.16 can then be used to estimate the maximum LSB current for the given process.

While the size of the current source array is made large enough to tolerate mismatches, it results in increased capacitance at the source node (V_S) of the switching pair. The manifestation of this capacitive effect is better understood in the temporal domain. As illustrated in Fig. 3.17a, when the data switch, transistors M_3 and M_4 shift from cut-off to saturation region or vice versa. However, this transition is not instantaneous; there exists a finite amount of time for which both switches are simultaneously on. During this period, the current source is immediately choked and the node V_S is discharged [9]. Once the switch pair's operating region transition is complete, the current source is forced to recharge the node, V_S, instead of delivering the desired current to the output node. The presence of a large capacitance at this node increases the recharge time constant. This effect is manifested as an output glitch that is proportional to the weight of the unit current source. In a DAC with multiple weighted cells connected together, the weighted glitches propagate to the output, creating a code-dependent glitch pulse.

Another form of dynamic distortion arises from the large aspect ratios of M_1 and M_2, thus resulting in large gate-drain capacitances and gate-source capacitances. These capacitances aid in the propagation of the switching behavior at node V_S to their bias nodes, as depicted in Fig. 3.17b. The bias fluctuation influences the magnitude of the i^{th} current source by a weighted coefficient α_i, such that

$$I_{OUT}(t) = I_{LSB}.\sum_{i=0}^{N-1} 2^i \alpha_i b_i(t) \tag{3.27}$$

It is seen that α_i is a function of several parameters such as the size of the switch pair, the impedance of the current source and the magnitude of the current. This

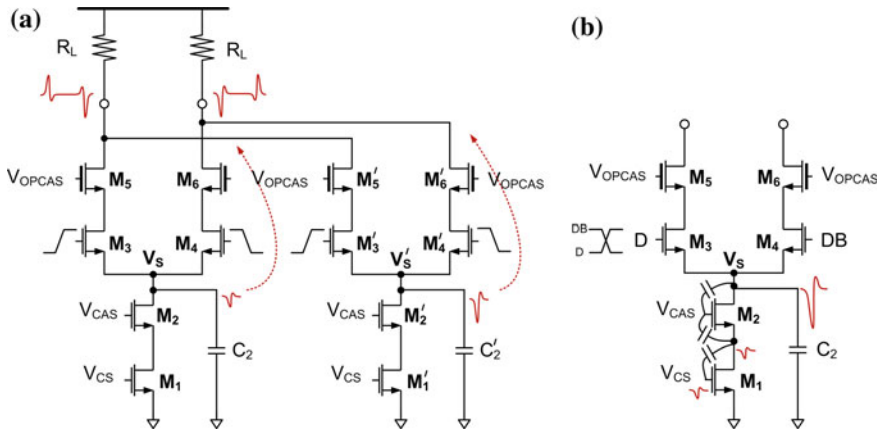

Fig. 3.17 **a** Glitch propagation from the source node **b** Modulation of the current source

distortion effect is mitigated by decreasing C_2 and increasing the gate capacitance of the bias nodes.[9] Care should also be taken to minimize the layout-induced coupling capacitance between the gates of M_1 and M_2.

3.5.3 Signal Swing

The noise-specified LSB current and resolution together determine the total current in a DAC. This eventually sets the output swing for a given load resistor, R_L. Thus, for a given supply voltage and transistor bias points, the maximum permissible signal swing is set to ensure all transistors are kept in saturation. For high-resolution DACs, the output swing can be allowed to increase further by raising the supply voltage. However, breakdown limits often mandate the use of thick-gate cascode devices ($M_{5,6}$) on top of the switch pair. Figure 3.18 illustrates the output swing as a function of the LSB current for different DAC resolutions. From the upper bound on swing limitations, the maximum permissible LSB current can be deduced.

Another major concern with large output swings is the linearity degradation due to the voltage-dependent drain capacitances of transistors, $M_{5,6}$. When the data switches, the output capacitance of the cell carrying no current is approximately given by

$$C_{OFF} = C_{gd5,6(OFF)}(V) + C_{db5,6}(V) \tag{3.28}$$

where C_{gd} is the gate-drain overlap capacitance of $M_{5,6}$, and C_{db} is the drain-bulk capacitance. In the ON state, the output capacitance of the cell is roughly given by

[9] Gate capacitances need to be relatively larger than the gate-drain capacitances of M_1 and M_2.

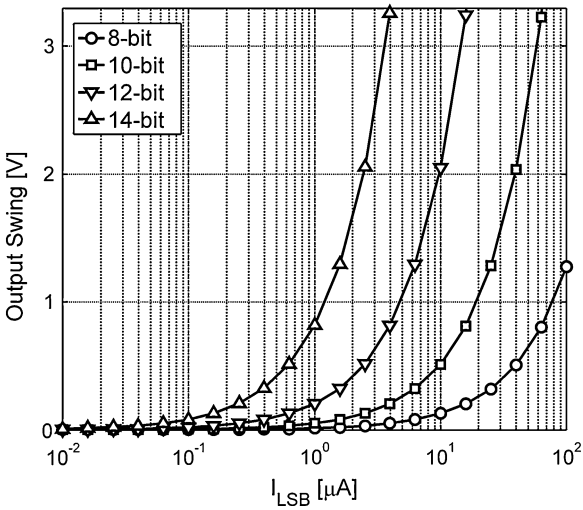

Fig. 3.18 Output swing as a function of the LSB current for various DAC resolution

$$C_{ON} = C_{gd5,6(ON)}(V) + C_{db5,6}(V) \tag{3.29}$$

This difference in capacitances in the ON and OFF states, along with the fact that the capacitance is voltage (output signal) dependent, results in code and amplitude dependent delays in addition to output load modulation. Together with the intrinsic transistor capacitances, interconnects can further increase the drain capacitance to the substrate. This eventually limits the maximum operation speed of the DAC cell. The parasitic capacitance at the output node can be minimized by layout techniques, while the mismatch in the ON and OFF impedances is alleviated by the use of leaker currents, as proposed in [10], and illustrated in Fig. 3.19b. It is shown that a leaker current of 1–2 % of the cell current is sufficient to maintain a fairly constant ON and OFF impedance [10]. While the use of leaker currents does not affect the differential output swing, it changes the single-ended swing by a constant DC value of $I_{LEAK} \times R_L$. The total sum of all leaker currents must be taken into consideration when estimating the lower bound of the single-ended swing, thus guaranteeing $M_{5,6}$ are maintained in saturation.

3.5.4 Switching Speed

The near-Nyquist performance of the DAC is highly dependent on the finite switching[10] and settling characteristics of the current cell [4]. This in turn is seen to be limited by the device transit or cut-off frequency (f_T) for a given process

[10] Switching refers to the action of turning a transistor from cut-off to saturation or vice versa.

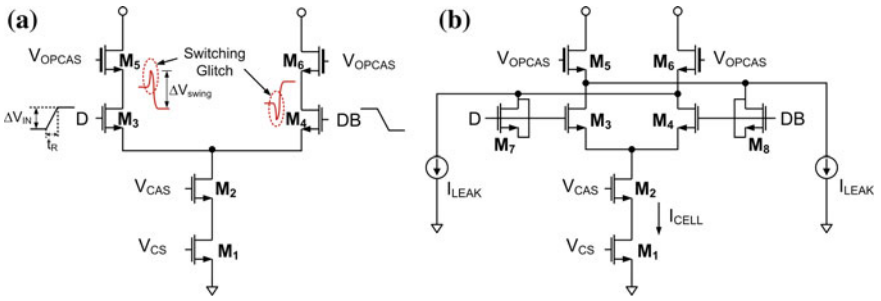

Fig. 3.19 **a** High-speed switching glitches in a DAC cell **b** A modified DAC cell with leakers and neutralization switches

technology. Figure 3.20 illustrates a hyperbolic increase in intrinsic device speeds with the reduction in gate lengths, as outlined by the International Technology Roadmap for Semiconductors (ITRS) [11]. In current-mode circuits, a general rule of thumb is to restrict the transistor switching speed to lesser than $f_T/20$ [12].

While the initial charging time of the DAC's output node is a function of the transistor's switching characteristics, the finite settling behavior is a function of the load capacitance and the rise time of the input signal. Figure 3.19a illustrates the switching action in a current-steering cell, where the data signal at the gate switches between voltage levels separated by ΔV_{IN}, with a rise time t_R. Using the model described in [12], the delay of a current-mode switching cell can be expressed as

$$\text{Delay} = k_{RC}\frac{\Delta V_{swing}}{\Delta I_{swing}}(C_{load}) + t_R \min\left(\frac{V_{OD}}{\Delta V_{IN}}, 1\right) \tag{3.30}$$

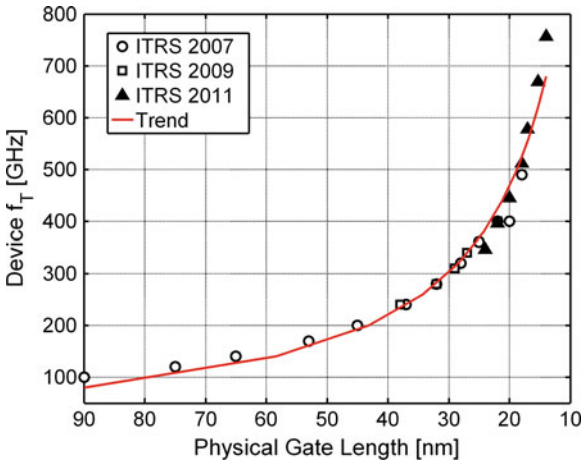

Fig. 3.20 Trend in device f_T as a function of feature length

where,

k_{RC}	$\cong \ln 2$
DV_{swing}	Single-ended voltage swing at drain of the switch pair
DI_{swing}	Single-ended current swing = Cell current
t_R	Rise time of the input at the gate of the switch pair
V_{OD}	Overdrive voltage of the differential switch pair = $V_{GS} - V_T$
DV_{IN}	Input voltage swing

All the above parameters are inter-dependent. For instance, a large value of ΔV_{IN} improves the switching characteristics. However, it also increases the swing at the drain of the switch pair (ΔV_{swing}), resulting in an increase in the overall cell delay. In addition, the maximum achievable ΔV_{IN} within a rise time t_R is limited by the process node. Further, the high-speed transition at the gate node increases the instantaneous voltage swing on the switch-pair drains, as well as those of the cascode transistors ($M_{5,6}$). In order to reduce these glitches, neutralization switches ($M_{7,8}$), are used to feed an equal and opposite glitch, as depicted in Figure 3.19b. Keeping the load capacitance fixed, the total LSB switch pair delay (sum of the switching and settling time to achieve 95 % of the final value of current) is simulated in a 90 nm CMOS process and the results are plotted with respect to the cell current in Fig. 3.21. The curves correspond to different widths for the switching pair, while the lengths are kept at minimum. From the settling time specifications, the minimum LSB current can be determined.

Hence, the DAC target speed and given process f_T together set the LSB current and the physical size of the switching pair. It is important to note that the current cells operating at lowest and highest current magnitudes have to switch simultaneously at the desired operation speed. In the event of mismatch in switching

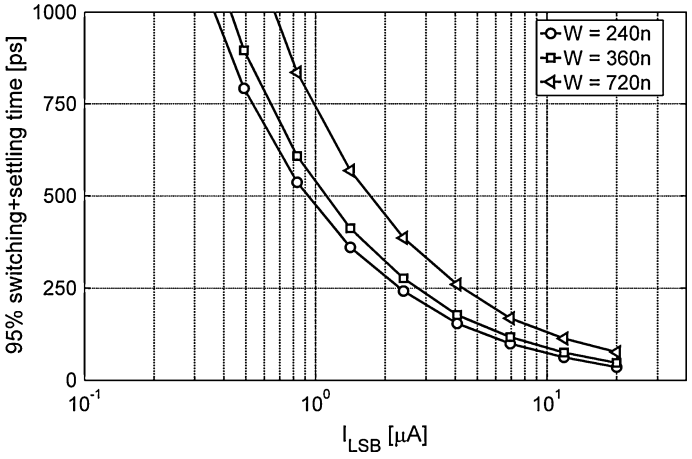

Fig. 3.21 Dependence of the transistor switching time on current

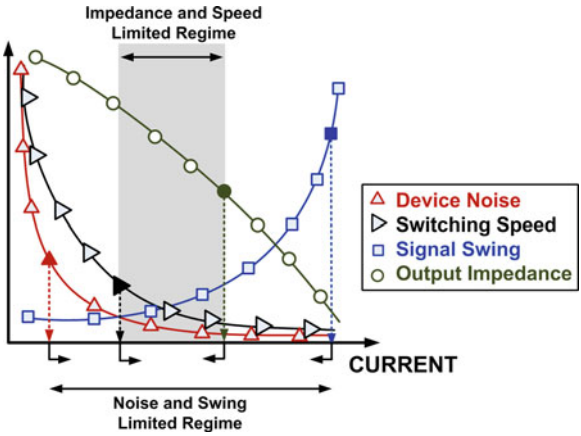

Fig. 3.22 The DAC design space: device noise, output impedance, signal swing and switching speed

speeds between the MSB and LSB cells, segmentation of the DAC is adopted as discussed in a later section.

Figure 3.22 illustrates the general trend for the four limiting parameters in the DAC design space as a function of the LSB current. The signal swing and output impedance together set the upper bound on LSB current, while the device noise and switching speed determine the smallest permissible current.

3.6 Segmenting the DAC

Although segmentation has been accepted as a mainstream option, it remains to be argued as to what is the optimal choice of the ratio of unary MSBs to binary LSBs. The limited accuracy of the MSB current sources can result in high levels of nonlinearity (i.e. large INL), which in turn degrades the dynamic performance of the DAC.

Based on the previous section, the DAC LSB current (I_{LSB}) is chosen to optimize across the design space. Subsequently, the transistor size and overdrive voltage need to be determined. An overdrive voltage of at least 100 mV is typically needed to ensure that the transistor is operating in strong inversion, and also to guarantee sufficient matching between the reference bias cell and the current mirrors. As discussed in Sect. 3.5.2, the lengths of the current source transistor and its cascoding device are set based on the output impedance requirement. As a result, the width of the transistor can be calculated. Another critical element that dictates the minimum size of the transistor is the mismatch accuracy between the MSB and LSB cell. One of the primary contributors to the variation between the two cells is the threshold voltage (V_T) mismatch between transistors [13, 14]. Let us denote the LSB and MSB currents for an N-bit DAC as

$$I_{LSB} = \frac{1}{2}\mu C_{OX}\frac{W}{L}V_{OD}^2 \tag{3.31}$$

$$I_{MSB} = \left(2^{N-1}\right)\frac{1}{2}\mu C_{OX}\frac{W}{L}\left(V_{OD} - \Delta V_T\right)^2 \tag{3.32}$$

where μ is the channel mobility, C_{OX} is the specific oxide capacitance, W (L) is the width (length) of the LSB current source transistor, V_{OD} is the overdrive voltage, and ΔV_T is the maximum mismatch error between MSB and LSB cell. In a binary cell array, the error in the MSB current source must be kept well below 0.5 LSB in order to maintain full accuracy. From (31) and (32), the maximum tolerable mismatch error can be expressed as

$$\Delta V_T \leq V_{OD}\left\{1 \pm \sqrt{1 + \frac{1}{2^N}}\right\} \tag{3.33}$$

This equation suggests that a lower V_{OD} implies a lower maximum tolerable V_T mismatch. On the other hand, if the V_T mismatch were to be fixed, we need a large V_{OD} to circumvent the mismatches. Figure 3.23a illustrates that the maximum tolerable V_T mismatch reduces as a function of the DAC resolution, further highlighting the issue of designing high-resolution DACs for a given mismatch constraint.

Reference [13] models the V_T mismatch as a Gaussian distribution with standard deviation, $\sigma_{V_T} = A_{V_{T0}}/\sqrt{WL}$, where $A_{V_{T0}}$ is a technology-dependent parameter. Therefore, in order to reduce the magnitude of σ_{V_T}, the transistor area can be increased, while maintaining a constant aspect ratio. Figure 3.23b illustrates the impact of increasing the area of the transistor on σ_{V_T} for a 90 nm CMOS process, assuming a V_{OD} of 100 mV. It is seen that even for a moderate resolution DAC, a large transistor area is required to minimize the impact of V_T mismatch. However, this results in reducing the high-frequency impedance of the current source, thus

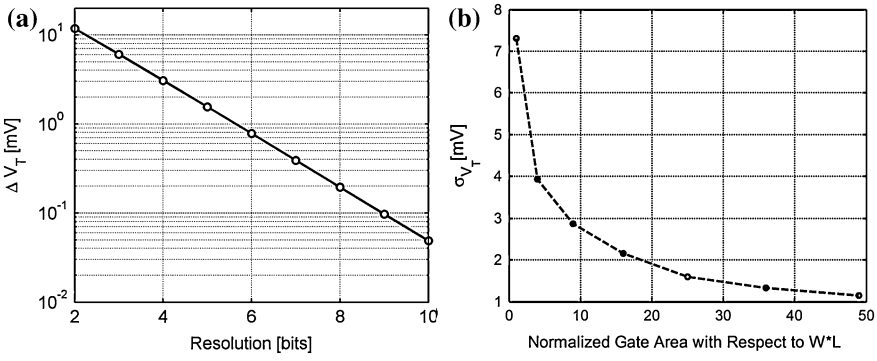

Fig. 3.23 **a** ΔV_T vs. Resolution **b** σ_{V_T} mismatch as a function of gate area

limiting the dynamic linearity of the DAC. The high-speed DAC designer is thus confronted with the challenge of meeting both static and dynamic linearity requirements. In the case of high-resolution DACs, the mismatch requirements dictate a transistor area that is prohibitive. In order to relax the transistor area requirements, a segmentation topology is adopted, such that the mismatch constraint is applied to the highest binary cell [14, 15].

Apart from mismatch-induced errors between the DAC current cells, fundamental circuit-level challenges, such as parasitic capacitance and finite output impedance, also influence the degree of segmentation. The degradation in transistor output impedance as channel lengths decrease, limits the maximum current that can flow through a transistor. As a result, a 100 % segmentation (unary DAC) is favored for low impedance processes. On the other hand, $2^N - 1$ current cells in a unary DAC increase the effective parasitic capacitance at the output node, thus limiting the speed of operation. In such cases, a 0 % segmentation (all-binary DAC) is preferred. In addition, a high degree of segmentation results in a significant increase in the DAC area (digital logic, analog current cells and interconnects) that makes timing compliance a challenge at the desired speed of operation. Thus, an optimal choice of segmentation needs to be made, with both process technology and circuit topology in mind.

3.6.1 The Segmentation Bound

As discussed in Sect. 3.5.2, a high current source impedance determines the accuracy of the current sources, while a high DAC cell impedance mitigates the effect of code-dependent impedance distortion. The extent by which a high Z_{CS} is required, is determined by the voltage fluctuation at the switch pair source node, V_S, relative to the LSB. It is noted that the resolution accuracy of the current sources needs to be maintained over the desired synthesis bandwidth of the DAC; i.e. both DC and AC impedances need to be sufficiently large. The LSB cell is designed to have the highest possible Z_{CS} over the synthesis bandwidth. As the current source is scaled in powers of two, Z_{CS} halves. The largest current cell that guarantees the desired output impedance across the synthesis bandwidth is where segmentation begins; all subsequent current cells are unary-weighted. This is the lower bound on segmentation (refers to the maximum number of binary cells that can be used) for the DAC. On the other hand, the desired DAC output bandwidth (computed from the load resistor and effective capacitance of all current cells) sets the upper bound on segmentation (or the maximum number of thermometer cells that can be used in the DAC).

Given the bounds on segmentation, the lower limit is more preferred as it implies fewer number of cells to be connected together, resulting in short routing lengths. This significantly aids in the reduction of the overall routing capacitance, thus decreasing clock skew mismatch and improving output bandwidth.

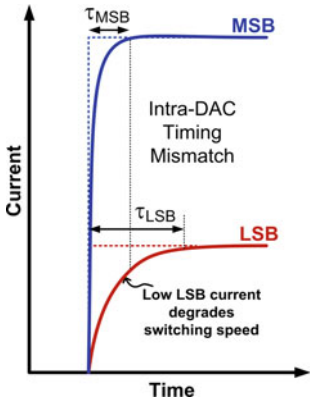

Fig. 3.24 Switching time mismatch in a DAC

Furthermore, a smaller effective area for the current sources helps decrease the ground line IR drop, which improves the cell matching. However, as discussed earlier, low segmentation designs demand high accuracy current sources. In addition, there exists a large ratio between the current magnitudes in the LSB and MSB cells in high-resolution DACs. As a result, the response times of the LSB cell (τ_{LSB}) and the MSB cell (τ_{MSB}) differ significantly, resulting in a mismatch in switching time instants, as depicted in Fig. 3.24. Such cell-by-cell timing mismatches result in the formation of output glitches that can limit the speed of operation, especially in gigahertz DACs. This creates a designer's paradox called the resolution-bandwidth trade-off. A high degree of segmentation helps alleviate this problem at the penalty of increased area and capacitance. A large chip area further results in being sensitive to process variations and IR drops that affect the matching between the current cells, both temporally and spatially. Consequently, the resolution-bandwidth trade-off comes into play.

3.7 Architectural Trends in High-performance DACs

Although BiCMOS technology is the predominant choice for high-speed operation, the issues of power, area and cost of integration have led to the wide adoption of CMOS-only processes. To this end, numerous circuit and architectural innovations have been proposed to improve the synthesis bandwidth and the linearity performance of DACs, enabling CMOS designs to compete with their BiCMOS counterparts [16]. In modern DACs, a high static linearity is obtained by using special layout techniques, trimming, calibration, dynamic element matching, etc. [15, 17, 18]. However, the dynamic performance of CS DACs have been known to fall rapidly with increase in signal frequency and clock rate. This section aims to introduce the reader to some of the well known techniques targeting high linearity and wideband signal synthesis.

3.7.1 Linearity Improvement

The dynamic performance of DACs is influenced by a number of factors such as MSB glitches, cell-to-cell timing skew, mismatch in settling time constants, low current source impedance and process gradients [14, 15]. Segmentation was one of the earliest approaches to enable high-speed DACs with good dynamic linearity [14]. It is seen as critical in minimizing the MSB glitch that occurs from delays in the switching cells. The DAC is split into segments that are thermometer or binary in nature. Each of these segments or sub-DACs can be implemented using either current dividers or resistor strings. The approach of combining current-steering with resistor strings was demonstrated in [19, 20]. In such architectures, great care needs to be taken to match time constants between the segments. Segmentation also increases the number of control signals to the DAC, resulting in timing skew between the cells. Such synchronization problems can occur from spatial variations or mismatch in the switch drivers. Spatial filtering using interleaving and inter-digitating help compensate for process gradients and thermal variations, thus improving intrinsic device matching accuracy [13]. Additionally, IR drops on the supply lines are reduced by the use of wide and thick metal lines, while power bus variations from the pins to the desired destination are addressed by binary tree structures [10]. However, these techniques often introduce large parasitic elements that limit the DAC's speed of operation.

The switching of currents also results in code-dependent glitches at the output. Deglitching or track-and-hold (T/H) circuits are typically employed to compensate for dynamic distortion, arising from switching and settling issues. A track-and-hold circuit holds the output constant while new data switches the DAC, and tracks only once the DAC output has settled. This enables the DAC glitches and settling errors to be mitigated. Nevertheless, the T/H circuit introduces its own errors such as pedestal, droop, clock feed-through and stepping errors [17]. When the input data switch, it must be guaranteed that the switch pair is not simultaneously off. Therefore, overlapping differential signals are used such that the transition point is optimal to achieve the best SFDR performance [21]. Ordinary switching also results in distortion from uneven pulse durations that is mitigated by the use of return-to-zero (RZ) switching. RZ implementation is realized by the use of an output switch, M_{RZ}, that shorts the output nodes together during the return phase, as illustrated in Fig. 3.25a. However at high frequencies, if RZ switching between two levels does not occur within the pulse duration, a memory effect of the input stream is manifested at the output in the form of code-dependent noise. Differential quad switching (DQS) Fig. 3.25b was proposed in [22] to mitigate the problems of RZ switching. In DQS, four logical signals obtained from the AND operation of data, clock and their inverted versions, result in a switching action at every clock edge; In other words, both high and low data signals are represented by rising and falling pulses, as illustrated in the timing diagrams in Fig. 3.25b. It is observed that the DQS scheme is equivalent to two RZ schemes operating in a staggered fashion. Owing to twice the frequency of switching than a single RZ, the code-dependent switching noise is pushed far out of the signal band. However, a major drawback

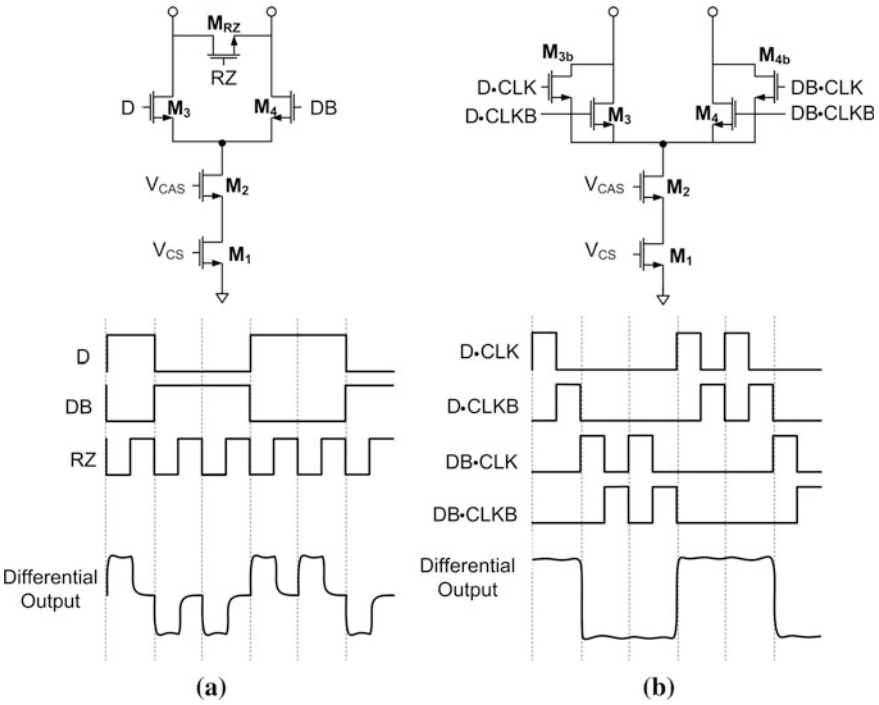

Fig. 3.25 DAC cell with **a** Return-to-zero (RZ) Switching **b** Differential-Quad Switching (DQS)

of the DQS approach is the increased dynamic power consumption due to the increased amounts of switching.

3.7.2 Towards RF Synthesis

One of the earliest and simplest techniques to synthesize high frequency signals beyond the Nyquist relied on band-pass filtering the inherent image replica components [23]. However, the availability of high selectivity filters limited the highest synthesizable frequency. In addition, the amplitude of the replicas in the higher Nyquist zones is reduced due to the *sinc* attenuation, translating to very low output power, or the need for a linear post-DAC amplification. RZ DACs [24] were used to push the *sinc* nulls to higher frequencies, thus improving the amplitude of the high-frequency spectral copies. The switch M_{RZ} in Fig. 3.25a is made large to have minimum ON resistance, so that the differential outputs are matched during the return phase. However, a large RZ switch loads the output node, thus degrading the bandwidth of operation. Furthermore, the rise-time requirements at the output node have to be met with respect to the effective ON

period of the DAC pulse, making the design of RZ DACs more cumbersome compared to their NRZ counterparts.

Another technique recently proposed to extend the DAC operation to near and beyond the Nyquist limit, is the use of partial-order hold (POH) [25]. A partial-order hold circuit, depicted in Fig. 3.26a, integrates the DAC output to realize a trapezoidal hold waveform. Such a technique has proven to result in image replica suppression over 40 dBc in the fourth Nyquist zone. However, this DAC architecture suffers from being sensitive to the POH period and demands tight control over the output signal rise time. In addition, both clock and POH jitter requirements are to be met, resulting in stringent specifications. Another means for high-frequency synthesis is the interpolation DAC, as depicted in Fig. 3.26b. It uses low sample-rate data that is fed into a digital interpolation filter and eventually fed to the DAC core. Although the speed of the digital interface is relaxed, the DAC still operates at the update rate. In the case of analog interpolation, the desired waveforms are created using microstepping methods [26] or using RZ DACs that perform an analog equivalent of zero padding.

Recent attempts have been made to combine the DAC and mixer functionality into a single topology to improve the overall system linearity and power consumption [27]. This DAC/mixer construct, referred to as the RF-DAC, is depicted in Fig. 3.27a. A mixer switch pair is placed on top of the DAC cells such that the local oscillator (LO) directly modulates the DAC output. The current-to-voltage-to-current conversion between the DAC and mixer, which introduces distortion, is completely removed. However, the up-conversion of the image-replicas and harmonic mixing of these replicas with the local oscillator, makes the post-RF-DAC filtering a daunting task. Reference [28] employs harmonic rejection mixers embedded into the DAC to suppress the harmonics caused by the mixer circuit, thus achieving greater than 70 dB of harmonic rejection. However, this technique greatly relies on the matching between the transistors that comprise the harmonic mixer.

Noise-shaping ($\Delta\Sigma$) techniques have also been employed in RF-DACs to improve the spectral quality of the DAC signal to obtain higher in-band SNRs, at the expense of large out-of-band noise [29, 30]. The delta-sigma modulator uses

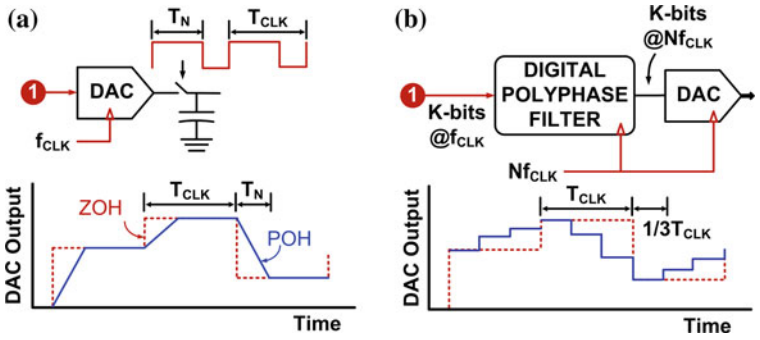

Fig. 3.26 DAC cell with **a** Partial-order hold DAC **b** Interpolation DAC

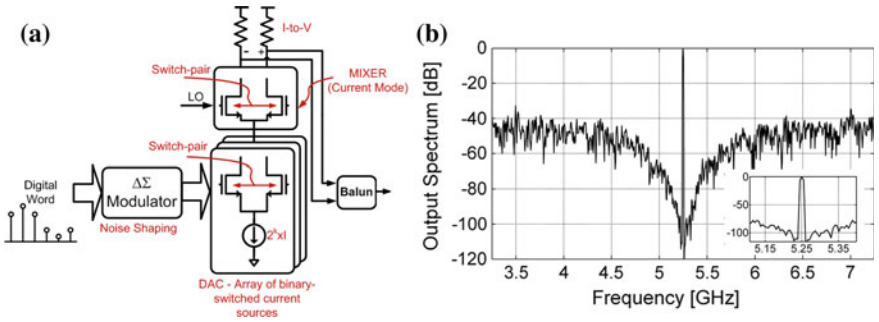

Fig. 3.27 **a** A simplified block diagram of a radio-frequency digital-to-analog converter **b** RF Output Spectrum using 2*nd*-order, 3-bit ΔΣ Modulator

lesser number of bits at the cost of increased rates of operation. However, the improvement in switching capabilities as processes emerge make this solution feasible to synthesize digital signals up to gigahertz frequencies. Figure 3.27b illustrates the output spectrum of a 5.25 GHz RF-DAC fed using a 2*nd*-order, 3-bit ΔΣ modulator. It is observed that the ΔΣ noise starts rising rapidly beyond the bandwidth of the modulator, eventually violating spurious emission requirements. The increased out-of-band ΔΣ noise and the spurious emission specifications together place stringent filtering requirements after the mixer, hence limiting the instantaneous bandwidth of operation to well below 100 MHz. Reference [29] integrates a high-Q passive LC bandpass filter to perform filtering of the out-of-band spurious and noise. However, the feasibility of this approach depends on the filtering requirements and the limited Q of on-chip passives. Alternatively, Reference [30] embeds a semi-digital FIR reconstruction filter in the digital-RF interface. The limitation of this approach lies in the need for large number of taps to obtain sufficient attenuation. Recently, a highly digital RF-DAC based transmitter exhibiting high linearity was proposed in Ref. [31]. The work demonstrated multi-band operation in 3G using a polar architecture, in which separate phase and amplitude paths were derived from the baseband digital signal. The phase signal modulates a digital controlled oscillator (DCO) and later acts as the LO signal, while a 14-bit amplitude signal is oversampled and then applied to the DAC current cells. A high dynamic range DAC with no noise-shaping was designed to relax the filtering requirements and obtain −160 dBc/Hz far-off noise specifications. However, there still exists the issue of image replicas in this structure, which limits the ability to further extend the bandwidth of operation.

The concept of a power DAC/mixer has recently evolved as an extension of the RF-DAC into the PA domain. In such a construct, as illustrated in Fig. 3.28, a high power transistor is used as the switching device and thus accomplishes both current commutating and current combining. However, the inherent trade-off between speed (f_T) and power capability (breakdown voltage) for semiconductor devices creates a

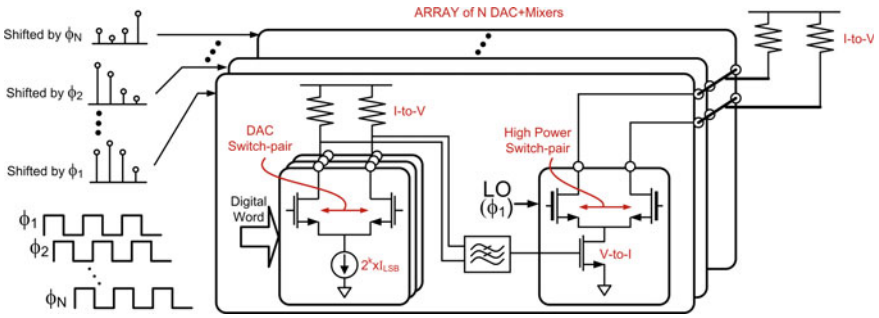

Fig. 3.28 Power Mixer Array

maximum achievable bandwidth that is power-limited. Parallel arrays of such DAC/mixers, proposed in [32, 33], was shown to cancel mixer nonlinearities by use of phase-shifted input signals and corresponding phase-shifted LO signals. Such a polyphase mixer has been shown to relax the mixer linearity requirements [34].

Another emerging architecture using time-interleaving topologies was proposed for high-frequency beyond-Nyquist synthesis [35, 36]. As illustrated in Fig. 3.29, a time-interleaved DAC comprises an array of DACs fed with interleaved signal samples and operating at interleaved instants of a clock period. It is also noted that the output of all DACs are connected together at all times; i.e. while one of the DACs is updating its output, the other DACs force their previously held values. This concept of hold and data interleaving was elaborated upon in Ref. [36] and proven to not hinder replica cancelation, while enabling beyond-Nyquist synthesis. The use of hold-interleaving was also shown to improve the replica suppression in the presence of gain and timing mismatches in [36].

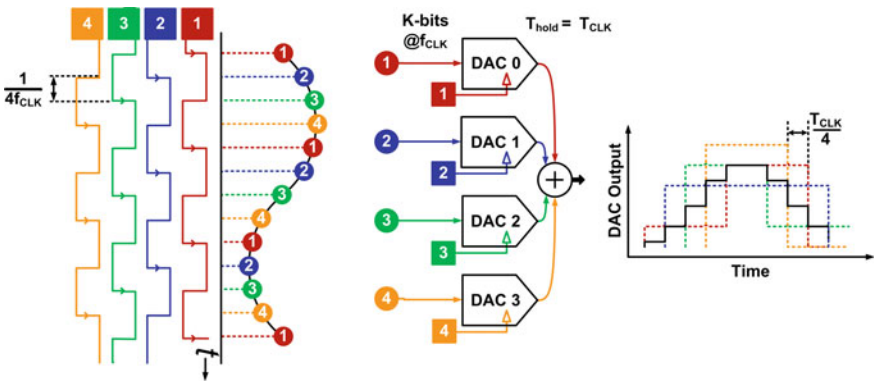

Fig. 3.29 DAC with data and hold interleaving [36]

3.8 Concluding Remarks

This chapter provided the reader with the holistic view of digital-to-analog conversion and associated challenges in CMOS process technologies. The DAC was first introduced as a system, and then abstracted to highlight some of the major limitations. A brief overview of performance metrics was then provided to quantify the DAC's static and dynamic performance. The complexity of the DAC design space, while perceived to be a simple array construct, forces a custom design flow. Attempts were made to simplify the DAC design space by breaking it into four parameters: device noise, output impedance, signal swing and switching speed. The reader was then introduced to the need for segmentation as a result of process variation and circuit limitations. Finally, a review of architectural and circuit techniques in the context of high-performance DACs to aid in circumventing the technological challenges, was presented. The need for higher resolution and higher speeds have kindled the interest in RF-DACs and interleaving, that are foreseen as promising for future ultra-wideband applications. The long-recognized fundamental limitations associated with process variations still remain a limiting factor in the implementation of high-performance DACs. However, the increasing gate densities in advanced CMOS processes allow for the realization of high-complexity mechanisms for self-calibration and self-compensation, which can effectively alleviate the various impairments suffered in the analog circuitry.

References

1. Balasubramanian, S., et al.: Ultimate transmission. IEEE Microwave Magazine. **13**(1), 64–82 (2012)
2. van den Bosch, A., Steyaert, M., Sansen, W.M.: Static and dynamic performance limitations for high speed D/A converters. Kluwer Academic Publishers, Dordrecht (2004)
3. Razavi, B.: Principles of data conversion system design. IEEE Press (1995)
4. Wikner, J.J.: Studies on CMOS digital-to-analog converters. Ph.D. Dissertation, Linkoping University, Linkoping (2001)
5. Maloberti, F.: Data converters. Springer, Dordrecht (2007)
6. Jenq, Y.C., Li, Q.: Differential non-linearity, integral non-linearity, and signal to noise ratio of an analog to digital converter. IMEKO International Measurement Confederation, Chicago (2002)
7. Tsividis, Y., McAndrew, C.: Operation and modeling of the MOS transistor. ISBN 978-0195170153, Oxford University Press, New York (2010)
8. Hiroki, A., Yamate, A., Yamada, M.: An analytical MOSFET model including gate voltage dependence of channel length modulation parameter for 20 nm CMOS. In: Proceedings of International Conference on Electrical and Computer Engineering. ICECE, Dec 2008, pp. 139–143
9. Chen, T., Gielen, G.G.E.: The analysis and improvement of a current-steering DACs dynamic SFDR-I: the cell-dependent delay differences. IEEE Trans. Circuits Syst. I, Regular Papers. **53**(1), (2006)

10. Lin, C.H., van der Goes, F., Westra, J., Mulder, J., Lin, Y., Arslan, E., Ayranci, E., Liu, X., Bult, X., abd, K.: A 12b 2.9GS/s DAC with IM3 $\ll -60$dBc beyond 1 GHz in 65 nm CMOS. IEEE J. Solid-State Circuits. **44**(12), 3285–3293 (2009)
11. Available online: www.itrs.net
12. Seckin, U., Ken Yang., C.: A Comprehensive delay model for CMOS CML circuits. IEEE Trans. Circuits Syst. I, Regular Papers. **55**(9), (2008)
13. Pelgrom, M.J.M., Duinmaijer, A.C.J., Welbers, A.P.G.: Matching properties of MOS transistors. IEEE J. Solid-State Circuits **24**(5), 1433–1439 (1989)
14. Lin, C.H., Bult, K.: A 10-bit, 500-MS/s CMOS DAC in 0.6 mm^2. IEEE J. Solid-State Circuits **33**(12), 1948–1958 (1998)
15. Bastos, J., Marques, A.M., Steyaert, M.S.J., Sansen, W.: A 12-bit intrinsic accuracy high speed CMOS DAC. IEEE J. Solid-State Circuits **33**(12), 1959–1969 (1998)
16. Balasubramanian, S., Khalil, W.: Architectural trends in GHz speed DACs. NORCHIP, Nov 2012
17. Bugeja, A.R., Song, B., Rakers, P.L., Gillig, S.F.: A 14-bit 100-MS/s CMOS DAC designed for spectral performance. IEEE J. Solid-State Circuits **34**(12), 1719–1732 (1999)
18. Oyama, B.,et al.: InP HBT/Si CMOS-based 13-bit 1.33Gsps digital-to-analog converter with $>$ 70 dB SFDR. IEEE Compound Semiconductor IC Symposium. Oct 2012
19. Jewett, B., Liu, J., Poulton, K.: A 1.2GS/s 15b DAC for precision signal generation. In: Proceedings of the IEEE International Solid-State Circuits Conference, Feb 2005 pp. 110–587
20. Halder, S., Gustat, H., Scheytt, C., Thiede, A.: A 20GS/s 8-Bit current steering DAC in 0.25 μm SiGe BiCMOS technology. In: European Microwave Integrated Circuit Conferenc. EuMIC, Oct 2008 pp. 147–150
21. Schofield, W., Mercer, D., Onge, L.S.: A 16 b 400 MS/s DAC with <-80 dBc IMD to 300 MHz and <-160 dBm/Hz noise power spectral density. In: Proceedings of the IEEE International Solid-State Circuits Conference, 2003 126–127
22. Park, S., Kim, G., Park, S., Kim, W.: A digital-to-Analog converter based on differential-quad switching. IEEE J. Solid-State Circuits **37**(10), 1335–1938 (2002)
23. Garceran, J.A.: Digital transmitter system and method. US Patent 6,944,238 (B2), Sep 2005
24. Choe, M-J.: Return-to-zero current switching digital-to-analog converter. US Patent 7,042,379 (B2), May 2006
25. Jha, A., Kinget, P.R.: Wideband signal synthesis using interleaved partial-order hold current-mode digital-to-analog converters. IEEE Trans. Circuits Syst. II, Exp. Briefs, **55**(11), 1109–1113 (2008)
26. Zhou, Y., Yuan, J.: An 8-bit 100-MHz CMOS linear interpolation DAC. IEEE J. Solid-State Circuits **38**(10), 1758–1761 (2003)
27. Luschas, S., Schreier, R., Lee, H.-S.: Radio frequency digital-to-analog converter. IEEE J. Solid-State Circuits. **39**(9), 1462–1467 (2004)
28. Maxim, A., et al.: A DDFS driven mixing-DAC with image and harmonic rejection capabilities. In: Proceedings of the IEEE International Solid-State Circuits Conference, Feb 2008 pp. 372–621
29. Jerng, A., Sodini, C.G.: A wideband $\Delta\Sigma$ digital-RF modulator for high data rate transmitters. IEEE J. Solid-State Circuits **42**(8), 1710–1722 (2007)
30. Taleie, S.M., Copani, T., Bakkaloglu, B., Kiaei, S.: A linear Σ-Δ digital IF to RF DAC transmitter with embedded mixer. IEEE Trans. Microw. Theory Tech. **56**(5), 1059–1068 (2008)
31. Boos, Z., et al.: A fully digital multimode polar transmitter employing 17b RF DAC in 3G mode. In: Proceedings of the IEEE International Solid-State Circuits Conference, Feb 2011 pp. 376–378
32. Shrestha, R., Eric, A., Klumperink, M., Mensink, E., Wienk, G.J.M., Nauta, B.: A polyphase multipath technique for software-defined radio transmitters. IEEE J. Solid-State Circuits. **41**(12), 2681–2692 (2006)

33. Kousai, S., Hajimiri, A.: An octave-range, watt-level, fully integrated CMOS switching power mixer array for linearization and back-off-efficiency improvement. IEEE J. Solid-State Circuits. **44**(12), (2009)
34. Xi, Y., et al.: Poly-harmonic modeling and predistortion linearization for software-defined radio upconverters. IEEE Trans. Microw. Theory Tech. **58**(8), 2125–2133 (2010)
35. Balasubramanian, S., Khalil, W.: Direct digital-to-RF digital-to-analogue converter using image replica and nonlinearity cancelling architecture. Electron. Lett. **46**(14), 1030–1032 (2010)
36. Balasubramanian, S., et al.: Systematic analysis of interleaved digital-to-analog converters. IEEE Trans. Circuits Syst. II, Exp. Briefs, **58**(12), 882–886 (2011)

Chapter 4
Digitally-Based Calibration Techniques for RF ΣΔ Modulators

Jose Silva-Martinez, Fabian Silva-Rivas, Cho-Ying Lu, John Mincey and Sebastian Hoyos

Abstract A calibration technique for Noise Transfer Function (NTF) optimization of Continuous-Time Sigma Delta (CT ΣΔ) modulators is presented. This technique employs a test tone applied at the input of the quantizer to evaluate the noise transfer function of the ΣΔ modulator using the capabilities of the Digital Signal Processing (DSP) platform usually available in mixed-mode systems. Once the modulator's output bit stream is captured, necessary information to generate the control signals to tune the ADC parameters for best Signal-to-Quantization Noise Ratio (SQNR) performance is extracted via an LMS software-based algorithm. This approach uses a simple test signature to measure both in-band and out-of-band loop behavior.

4.1 Introduction

With the increasing number of services and wireless standards in the last decade, the next generation of communication solutions must support fully-integrated systems on a chip (SoC) to advance towards the design of multi-standard CMOS devices. The emphasis is to perform the broadband signal processing to accommodate higher data throughput. A critical block in multi-standard transceivers is the ADC. Among the large variety of ADC architectures available [1–4],

J. Silva-Martinez (✉) · J. Mincey · S. Hoyos
Texas A & M University, Department of Electrical and Computer Engineering, College Station, TX, United States
e-mail: jsilva@ece.tamu.edu

F. Silva-Rivas
Broadcom Corporation, Austin, TX, United States

C.-Y. Lu
Intel Corporation, Hillsboro, OR, United States

P. Carbone et al. (eds.), *Design, Modeling and Testing of Data Converters*, Signals and Communication Technology, DOI: 10.1007/978-3-642-39655-7_4,

ΣΔ modulators are gaining significant momentum because this technique is compatible with the trend of the semiconductor industry towards scaled nanometric technologies: more digital and less analog. Employing a low-resolution ADC operating in closed loop moves the sample-and-hold operation after a continuous-time filter, simplifying the interface and reducing the aperture errors due to clock jitter. The migration to more advanced technologies is easier since continuous-time filtering can be efficiently realized even at the GHz range [1, 2, 5–38].

In ΣΔ architectures, the remaining analog blocks are the filter, a low-resolution ADC (quantizer) and the digital to analog converter (DAC). The ΣΔ architecture heals the limitations of a low-resolution ADC by closing the loop through a linear DAC and an analog filter; to achieve enough resolution the signal is oversampled leading to a high resolution ADC architecture. Since the system operates in closed loop, it is essential to ensure large enough loop gain in the frequency of interest while maintaining DAC linearity. A major issue in all closed loop architectures is loop stability; usually the higher the frequency of operation the more difficult it is to guarantee system stability due to the phase contribution of the unavoidable parasitic poles as well as delay in the digital blocks. This issue is especially critical in RF ΣΔ ADCs.

The requirements of large bandwidth, high operational frequency and resolution make the design of the ADC challenging. PVT variations affect the filter time-constants and may add significant excess phase that degrades the available phase margin. The typical 25 % variations in the location of filter's poles and zeros can not be tolerated and have to be corrected [1, 2, 6, 7, 13, 14, 20, 34, 37]. In order to retain the properties of the ΣΔ ADCs at high frequencies, the sampling frequency must be in the order of GHz leading to significant limitations due to clock jitter [1, 2, 11, 12, 15]. Also, the current mismatch issue in multi-bit DACs threatens the system's linearity and degrades the ADC's resolution [10, 14, 15, 23–25]. Although there are different reported solutions to either pseudo-randomize or shape the DAC mismatch, the reduced excess loop delay margin in the digital feedback path due to the rise of sampling frequency makes some of these solutions inadequate for wideband applications.

The software based loop calibration technique discussed in this chapter is intended for the optimization of the noise transfer function in both low-pass and band-pass ΣΔ modulators. The proposed approach measures the noise transfer function in the digital domain using an auxiliary and non-critical test signature composed by several tones; based on the response of the loop to the strategically selected tones, the loop parameters are sequentially adjusted until the noise transfer function presents its best possible performance. Although not covered, the main concepts can be exported for the calibration of all types of data converters as well as for linearity calibration.

4.2 Review of CT Sigma-Delta Modulators

Figure 4.1 shows the block diagram of a typical CT ΣΔ Modulator showing the embedded zero-order hold, the $Z^{-1/2}$ delay and a multi-bit quantizer-DAC. The filter's input referred noise is represented by n_f. Usually the combination of the sample and hold (S/H) and low-resolution ADC (Quantizer) add an extra half delay leading to a Z^{-1} loop delay as shown in the linearized block diagram of Fig. 4.1b. The S/H samples the input signal and holds it while the ADC converts it into digital format. It is modeled by an ideal sampling function that generates in-band alias (or image) signal n_i due to noise and out-of-band signals present at S/H input within the frequency band $f_s - BW < f < f_s$-BW with BW being the modulator's bandwidth. The effects of n_i should not be ignored since the tendency of the wireless architectures is towards the minimization of front-end filtering, and the rejection of high-frequency signals provided by the loop filter is very limited. The S/H model is complemented with the Sinc(x) function. The delay introduced by the S/H is accounted in the quantizer model which is complemented with and additional adder and the error signal Q_n due to the quantization function. Digital circuitry is usually necessary to couple the quantizer output to the DAC input; the delay added by this circuitry is adjusted to $Z^{-1/2}$ to end up with a Z^{-1} architecture. The signal n_d stands for the DAC output referred noise.

While considering this small signal linear model, the modulator digital output is a linear combination of all signals, given by

$$D_{out} = STF^*(V_{in} + n_f - n_d) + ITF^* n_i + NTF^* Q_n \tag{4.1}$$

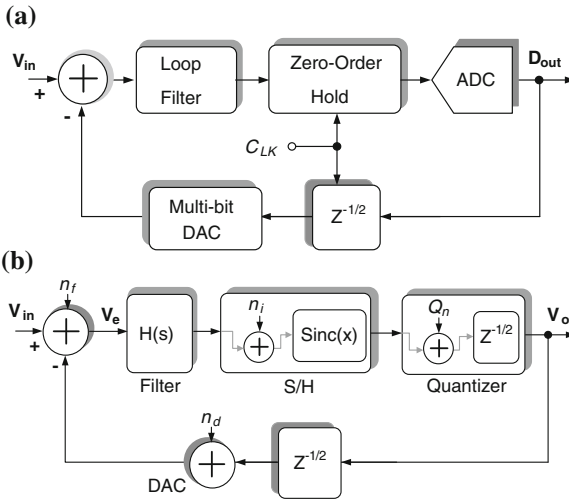

Fig. 4.1 Simplified schematic of a ΣΔ modulator: a Block diagram b Linearized model including the input referred filter noise n_f, in-band image signal n_i due to the S/H, ADC quantization noise Q_n and DAC output referred noise contribution n_d

where V_{in} is the analog input signal, Q_n is the quantization noise due to quantizer limited resolution, STF is the signal transfer unction, ITF is the image transfer function and NTF is the noise transfer function.

The loop gain LG can be obtained by breaking the loop at any point; in practical systems we have to account for the input impedance of the elements where the loop is opened. These issues are certainly minimized when breaking the loop in the digital section. Conventional circuit analysis techniques lead to the following result

$$LG(s) = H(s)\mathrm{Sinc}\left(\frac{\omega T_s}{2}\right)Z^{-1} = \left|H(s)\mathrm{Sinc}\left(\frac{\pi f}{f_s}\right)\right|\left\{e^{j\left(\phi_f - 2\pi\frac{f}{f_s}\right)}\right\} \tag{4.2}$$

where Ts $(= 1/f_s)$ is the clock period and $\phi_f$is the loop filter's phase response. Notice that the phase contribution of Z^{-1} is dominant at high frequencies; e.g. $-90°$ at $f = f_s/4$ and $-180°$ at $f = f_s/2$. To guarantee loop stability, the filter's phase contribution ϕ_f must be minimized at high frequencies, especially around the loop's unity gain frequency. It is advisable to maintain the phase of the loop at the unity gain frequency to no more than $-130°$; the following expression can be used for that purpose:

$$\mathrm{Phase}(LG(f_u)) = \phi_f(f_u) - 2\pi\left(\frac{f_u}{f_s}\right) \tag{4.3}$$

For high-order loops several medium frequency zeros must be added for that purpose. In fact, in most of the practical cases the loop filter presents equal number of zeros and poles, leading to a non-zero high frequency gain usually larger than unity; therefore, the loop's unity gain frequency is then determined by the product of the filter's high frequency gain and $\mathrm{Sinc}(\pi f/f_s)$. Although it is unclear at this stage of the discussion, it is advisable to recognize that the modulator's designed with large out-of-band gain are more prompt to present stability issues since the unity gain frequency moves to higher frequencies leading to excessive negative loop phase. Usually ϕ_f varies slowly at high frequencies, and then Phase(LG) maybe dominated by the term $-2\pi f/f_s$.

Although not accurate, in a first approximation the modulator's output can be considered as a linear combination of all the signals present in the loop. Employing Mason's rule, it can be shown that the modulator's output can be approximated as follows

$$V_0(s) = \frac{\left\{\left|H(s)\mathrm{Sinc}\left(\frac{\pi f}{f_s}\right)\right|\left\{e^{j\phi_f}\right\}\{V_{in} + n_d + n_f\} + \left|\mathrm{Sinc}\left(\frac{\pi f}{f_s}\right)\right|n_a + Q_n\right\}e^{j\left(-\pi\frac{f}{f_s}\right)}}{1 + \left|H(s)\mathrm{Sinc}\left(\frac{\pi f}{f_s}\right)\right|\left\{e^{j\left(\phi_f - 2\pi\frac{f}{f_s}\right)}\right\}} \tag{4.4}$$

The signal transfer function STF can be obtained by looking at the linear transfer function from V_{in} to V_0. Zeroing all other signals except V_{in} in eqn 4.4, it can be defined

$$\mathrm{STF}(s) = \frac{V_0(s)}{V_{in}(s)} = \frac{\left|H(s)\mathrm{Sinc}\left(\frac{\pi f}{f_s}\right)\right| e^{j\left(\phi_f - \pi\frac{f}{f_s}\right)}}{1 + \left|H(s)\mathrm{Sinc}\left(\frac{\pi f}{f_s}\right)\right| \left\{e^{j\left(\phi_f - 2\pi\frac{f}{f_s}\right)}\right\}} \tag{4.5}$$

Notice in (4.4) that STF applies for computing the output contribution of both filter and DAC noise components. The noise components n_i due to the sampling (blockers, plus thermal and quantization noise folded back to base-band) will appear at the modulator's output as

$$\mathrm{ITF}(s) = \frac{V_0(s)}{n_i(s)} = \frac{\left|\mathrm{Sinc}\left(\frac{\pi f}{f_s}\right)\right| e^{j\left(\pi\frac{f}{f_s}\right)}}{1 + \left|H(s)\mathrm{Sinc}\left(\frac{\pi f}{f_s}\right)\right| \left\{e^{j\left(\phi_f - 2\pi\frac{f}{f_s}\right)}\right\}}. \tag{4.6}$$

It is interesting to note that $n_i(s)$ is largely determined by the signal power of $V_i(s)H(s)$ around the clock frequency. The quantization noise appears at the modulator output shaped by the following transfer function:

$$\mathrm{NTF}(s) = \frac{V_0(s)}{Q_n(s)} = \frac{e^{-j\left(\pi\frac{f}{f_s}\right)}}{1 + \left|H(s)\mathrm{Sinc}\left(\frac{\pi f}{f_s}\right)\right| \left\{e^{j\left(\phi_f - 2\pi\frac{f}{f_s}\right)}\right\}}. \tag{4.7}$$

According to these results, it is advisable to increase $|H(s) * \mathrm{Sinc}(\pi f/f_s)|$ for in-band signals. In this case, STF approaches unity up to frequencies close to the unity gain frequency, while both ITF and NTF decrease drastically in the band of interest. Therefore, V_{in}, n_f and n_d will appear at the modulator output with equal magnitude because the loop provides almost no filtering up to $f = f_u$. On the other hand, the effect of in-band quantization noise and image noise are further attenuated by the factor $1 + \mathrm{LG}(s)$. The main benefit of the $\Sigma\Delta$ loop is better appreciated if we analyze the ratio of the modulator's output due to the desired input signal and output due to quantization noise; from (4.5) and (4.7) it follows that *for flat in-band STF and NTF* transfer functions

$$\mathrm{SQNR} = \frac{\int_{BW} (\mathrm{STF} * V_{in})^2 df}{\int_{BW} (\mathrm{NTF} * Q_n)^2 df} \cong \frac{|\mathrm{STF}|^2 \left(\frac{V_{in-pk}^2}{2}\right)}{|\mathrm{NTF}|^2 \left(\frac{\Delta V^2}{12}\right)\left(\frac{2BW}{f_s}\right)} \cong |H(s)|^2 * \mathrm{OSR} * \mathrm{SQNR}_Q \tag{4.8}$$

Notice in this result that $V_{in-pk}^2/(2DV^2/12)$ is the peak SQNR_Q of the low-resolution quantizer in open loop; ΔV is the quantization step and BW the filter's bandwidth. According to (4.8), the closed-loop SQNR improves by the in-band gain of the loop filter and oversampling ratio OSR. Hence, maintaining good

control on filter in-band gain and phase response is of primary importance. Of course, loop stability must be guaranteed to take advantage of these benefits.

4.3 Practical Loop Limitations

A major issue found in continuous-time networks is the lack of accuracy due to process-voltage-temperature (PVT) tolerances that may lead to over 25 % variations of the time constants [1, 2, 13–15, 17, 20]. To alleviate this problem, tuning techniques have been successfully used in continuous-time filters. However, the optimally tuned ADC loop requires correcting for filter's corner frequency deviations, excess loop delay and variations of DAC coefficients. A potential approach measures in the digital domain the notch performance of bandpass ADCs [26]; unfortunately this technique is affected by the power of the incoming out-of band information in on-line calibration schemes, but it is an interesting approach for off-line calibration. Optimization of individual building blocks and use of programmable delay lines for the optimization of the loop delay and reconfigurable filter-oscillator system for notch tuning were also reported in [20]. Matching the loop transfer function with a digital well controlled prototype is another interesting option [27, 28]. These approaches tune the ADC parameters at power up. Let's analyze these effects in more detail to understand better the complexity of this issue.

Process-Voltage-Temperature Variations. Assuming ideal components the loop regulation leads to an SQNR performance predicted by (4.8). However, RC products typically change by ± 25 % over PVT variations. Also, the finite in-band amplifier gain (<30 dB) as well as the parasitic poles present in high-gain multi-stage amplifiers affect both magnitude and phase of the filter's frequency response. If poles shift to lower frequencies, then the filter gain reduces at the edge of the pass-band reducing the in-band noise shaping that can be computed by (4.8). On the other hand, if the frequency of the poles and zeros shift to high-frequency, the loop unity gain frequency increases leading to reduced phase margin as depicted in Fig. 4.2.

PVT variations affect the filter response and thus affecting the modulator's SQNR. However, the most significant effect is on the location of the unity gain frequency f_u which has significant impact on the loop's phase margin. Usually the higher the unity gain frequency the lower the loop's phase margin due to the negative phase contribution of the delay element Z^{-1}. The negative phase contribution due to high-frequency parasitic poles (not considered in the previous analysis) is always present and should not be ignored, especially if those poles are not located far beyond the loop's unity gain frequency.

Notice that large out-of-band gain makes the loop more sensitive to the effects of the adjacent channels; in practical systems, the modulator must operate with weak in-band signals while the power of the adjacent (undesired) channel is very

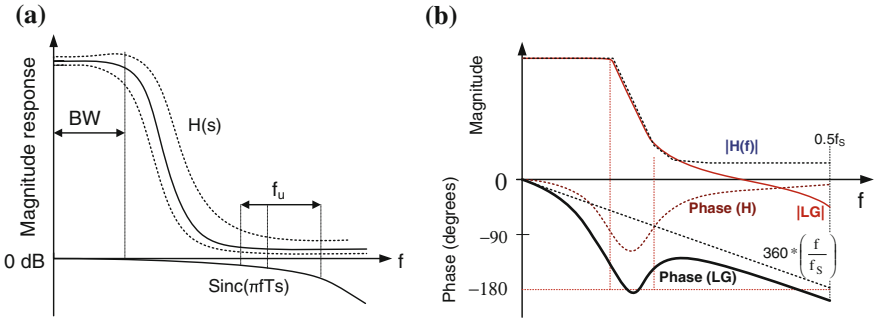

Fig. 4.2 a Effects of PVT variations on filter magnitude response b typical phase response. PVT variations usually have significant impact on both modulator's SQNR and loop stability

strong. A related issue is the potential saturation effects on the different amplifiers used for filter implementation. All internal nodes must be properly scaled to prevent saturation in some blocks. Saturation means a lack of response to a given input; hence that particular amplifier will operate as an open circuit while saturated, leading to drastic variations on filter response that result in large SQNR reductions and even loop instabilities.

Excess Loop Delay. The excess delay with respect to the ideal timing between the instantaneous quantizer sampling time and the time when a change in the filter output is generated will cause SQNR degradation and stability issues. The tolerance of the passive elements introduces phase errors that may increase the filter delay [2, 20–22]. The other major cause of excess loop delay is due to the inherent delays present in the digital circuits that drive the DAC leading to a time delay before it can output the current. This issue can be partially alleviated by employing early clocks to account for this timing error. In summary, the excess loop delay can be expressed as

$$\text{Excess Loop Delay} = \Delta T = \Delta T_{\text{filter}}(f) + \Delta T_{\text{Digital-DAC}}(f) \tag{4.9}$$

where ΔT_x stands for the excess delay associated with block x. The excess loop delay can be converted into phase error to get more insight on its effects on loop stability; the results are depicted in Fig. 4.3 as a function of $\Delta T/T_s$. The effects of the excess loop delay are more severe when the loop unity gain frequency is placed at higher frequencies; e.g. excess phase is in the range of 7.5° at $f = 0.1\, f_s$ if the excess loop delay is 20 % of T_s, and around 36° at $f = 0.5\, f_s$. The overall excess loop delay can be accounted for by adding it into the Z^{-1} block, leading to $Z^{-1-\Delta T/Ts}$. The loop phase (in degrees) can then be written as follows:

$$\text{Phase (LG)} = \phi_f - 360^0\left(1 + \frac{\Delta T}{T_s}\right)\left(\frac{f}{f_s}\right) \tag{4.10}$$

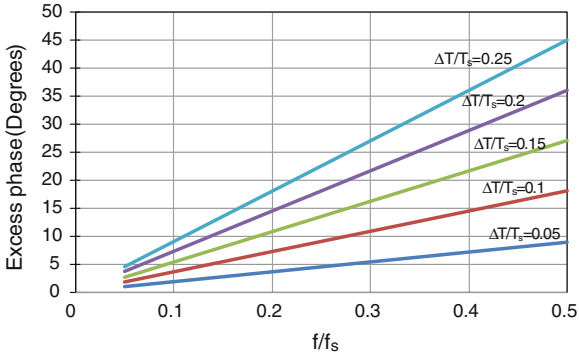

Fig. 4.3 Excess phase (in degrees) as function of frequency for several excess loop delay cases

The excess loop delay increases the phase contribution of the nominal delay element according to $(\Delta T/T_s)(360°)f/f_s$. The excess loop delay is plotted in Fig. 4.3 for various cases. This effect can also be visualized in Fig. 4.2 by increasing the slope of the phase contribution of the delay element by a factor of $1 + \Delta T/T_s$. Notice that architectures with high unity gain frequency that usually lead to better SQNR figures are particularly more sensitive to excess loop delays. To relax the demanding, high-frequency response of the amplifiers used for the loop filter's realization, the modulator's loop can be broken into two loops running in parallel such that low-frequency behavior is dominated by a regular filter while the high-frequency behavior minimizes the use of amplifiers. The result is that the operation of the modulator's loop around its unity gain frequency and beyond is determined by a fast path around the quantizer [1, 5–7, 11, 12].

According to Fig. 4.3, as a rule of thumb, excess loop delay should be limited to no more than 10 % of the clock period in order not to significantly degrade the SQNR. In this case, even if the loop unity gain is around $f_s/4$, the phase margin does not degrade by more than 9° which can be tolerated in most of the practical cases especially if the loop's ideal phase margin is over 60°. This is, however, very difficult to achieve when dealing with high-frequency modulators. For a clock frequency of 1 GHz, clock period is only 1000 ps, and to design the system such that the overall excess loop delay is under 100 ps is quite challenging, mainly due to the unavoidable PVT variations, lack of precision when predicting delays in buffers and flip-flops, as well as routing delays. *The global calibration strategy described in the following section takes into account all PVT variations, DAC coefficient accuracy and excess loop delay to effectively optimize the ADC's loop performance.* The proposed strategy is not able to correct DAC non-linearities which may require the use of randomization techniques such as dynamic element matching or calibration of individual elements [29–31].

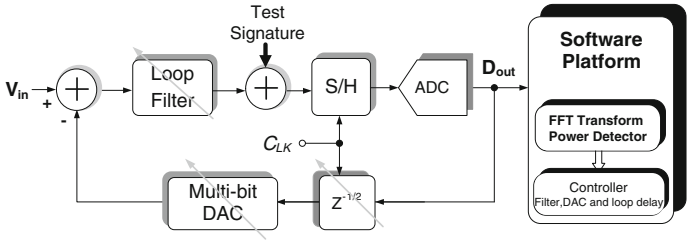

Fig. 4.4 Conceptual diagram for the digital-based loop calibration scheme

4.4 Digital-Based Loop Calibration Scheme

The loop calibration approach described in this section relies on a software-based platform instead of power hungry and inaccurate analog circuitry. The system level implementation of the digital based tuning scheme is shown in Fig. 4.4. During calibration, a test signature is applied at the input of the quantizer to emulate a systematic and testable signal that resembles the quantization noise. Since the test signature is applied after the high-gain filter, its noise is shaped by the loop transfer function and the auxiliary circuitry has very little effect on the dynamics of the loop. The magnitude of the test signature transfer function is given by:

$$\mathrm{TTF}(s) = \frac{V_0(s)}{V_{\mathrm{test}}(s)} = \frac{\left|\mathrm{Sinc}\left(\frac{\pi f}{f_s}\right)\right|\left\{e^{-j\left(\pi\frac{f}{f_s}\right)}\right\}}{1+\left|H(s)\mathrm{Sinc}\left(\frac{\pi f}{f_s}\right)\right|\left\{e^{j\left(\phi_f-2\pi\frac{f}{f_s}\right)}\right\}} = \left|\mathrm{Sinc}\left(\frac{\pi f}{f_s}\right)\right|\mathrm{NTF}(s) \tag{4.11}$$

The output bit stream then provides reliable information about the NTF modulated by the Sinc function which can be easily de-embedded in case accurate NTF measurements are necessary. The quantizer output digital bit stream is then processed by the digital signal processor, and the power of the test signature is then measured using the Fast Fourier Transform (FFT). The power of the test tones embedded in the test signature are used in an adaptive Least Mean Square (LMS) algorithm that controls the critical parameters such as the programmable delay element, filter corner frequency and DAC gain with the aim of minimizing the power of the measured signature and thus optimizing the loop parameters for the best possible NTF.

The LMS algorithm generates the digital control signals to tune the loop's parameters, starting with the tunable delay element to ensure loop stability. Once the controller recognizes that the loop is stable, the filter's time constants are then tuned to obtain the best possible shape for the NTF.

The algorithm for the digital tuning scheme is described by the following steps: (i) inject a test signature (several tones at the desired frequencies); the case of a low-pass modulator is depicted in Fig. 4.5; (ii) although the power of the signature

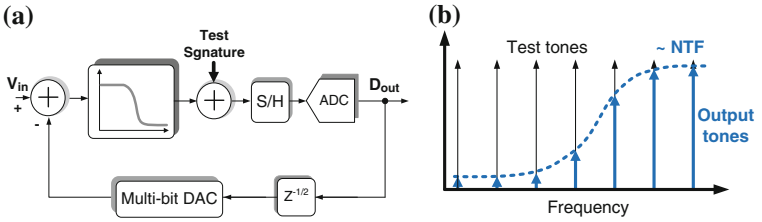

Fig. 4.5 a NTF characterization for a low-pass ΣΔ modulator b input and output test tones

is not quite relevant for the operation of the calibration algorithm it is desirable to maintain the total power 10 dB below the full-scale power to guarantee linear operation; (iii) detect the power of each tone at the modulator's output to construct the NTF; (iv) by means of an LMS algorithm, a digital control tuning bit stream is computed based on the difference between the stored and the new estimated power values of the detected test tones; (v) the parameters that control the loop delay are tuned first; (vi) iterate between (iii)–(v) until the shape in the NTF and the loop delay is optimized; (vii) the algorithm then tunes the DAC coefficients with the programmable delay element, again, until the power of the detected test tone is minimized thus maximizing SQNR. The requirements of the calibration tones are not rigorous, since the calibration algorithm will adjust the tuning knobs based on their relative power with respect to the previous iteration until the power of the in-band tones is minimized, thus maximizing the modulator's SQNR.

The detailed architecture is depicted in Fig. 4.6. Delayed versions of the master clock are generated by employing a set of delay elements. The output is selected from the available phases through a MUX controlled by the software based calibration algorithm. The loop response for each clock phase is measured until the calibration algorithm finds the one that results in maximum in-band attenuation. The filter poles are tuned by employing capacitor banks; it has been found that deviations in the filter's quality factor are not critical since this parameter is usually a function of the ratio of capacitors or ratio of resistors and therefore tolerant to PVT variations. It is advisable to locate the frequency of one of the calibration tones at the edge of the passband gain to ensure good NTF attenuation until the pass-band corner. Once the filter's time constants are calibrated, the DAC coefficients can then be calibrated. For that purpose, the current sources employed in the current steering DAC are adjusted through the calibration's LMS algorithm. With enough calibration time and digital resources, loop linearity could be calibrated too.

4.5 Bandpass ΣΔ Modulators

For RF and high-IF solutions, Continuous-Time Bandpass ΣΔ Modulators are frequently used because at intermediate frequencies the flicker noise is small compared to that of the quantization noise [1, 2, 16–20, 26, 27, 30, 32–38].

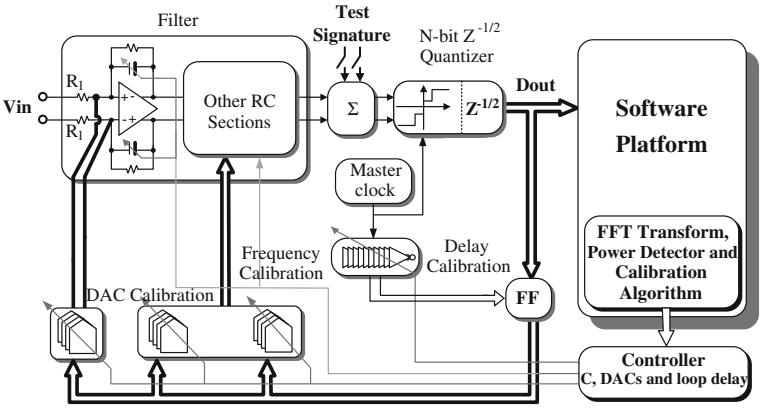

Fig. 4.6 Continuous-time sigma-delta modulator with frequency, excess loop delay and DAC calibration knobs

The system level implementation of the digital tuning scheme for a band-pass ΣΔ ADC is similar to the one shown in Fig. 4.6, but the filter realizes a bandpass transfer function. In addition to the in-band analog input signal, two out-of-band test tones equally spaced around the center frequency (f_0) are applied at the input of the quantizer to emulate a systematic and testable in-band quantization noise. Since the test tones are applied at the output of the loop, its noise is shaped by loop transfer function, and the auxiliary circuitry has little effect on the dynamics of the loop. The quantizer's output digital bit stream is then processed by the digital signal processor (DSP), and the power of the loop output that results of the test tone is then measured in the digital domain using the FFT. The estimated power of the test tone is used in an adaptive Least Mean Square (LMS) algorithm that controls several parameters with the aim of minimizing the power of the measured test tone and thus maximizing the rejection to quantization noise. The LMS algorithm generates the digital control signals to tune the loop's notch frequency by controlling a bank of capacitors used for the realization of the bandpass filter. Once the notch frequency of the NTF is set at the desired frequency, the DAC coefficients and excess loop delay are then adjusted with the same aim: power minimization of the test tone to reach the best possible SQNR.

Shown in Fig. 4.7a is the output spectrum of a 200 MHz Bandpass ΣΔ Modulator with 20 % deviation in its center frequency; SQNR is only 34 dB since the quantization noise is excessive at 200 MHz. The input signal is applied at the input of the ADC at 200 MHz, while two calibration tones are applied at the input of the quantizer. The frequencies of the out-of-band testing tones are 193 MHz and 207 MHz. Over 20 % variations on the loop parameters were intentionally introduced; this results in a notch frequency around 220 MHz. After several iterations using the aforementioned algorithm, the notch frequency is tuned to the desired value by just monitoring the power of the test tone set around 200 MHz as

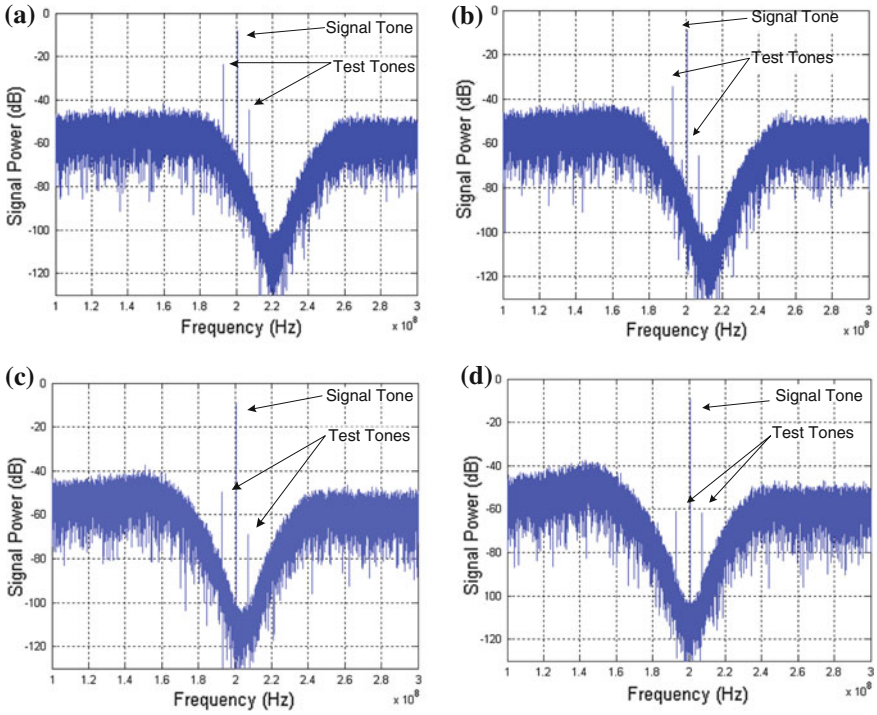

Fig. 4.7 Output spectrum of the 6th order ΣΔ modulator with digital calibration scheme vs. the number of iterations of the calibration scheme for 20 % PVT variations on ADC parameters

shown in Fig. 4.7b–c. The algorithm stops when the power of the test tones at the modulator's output are at its minimum value, e.g. −82 dB.

The LMS algorithm adjusted the bank of capacitors until the notch of the NTF was tuned at modulator's center frequency. Fig. 4.7d shows the ADC spectrum after calibration. The algorithm stops when the power of the tones at the quantizer input are equal and at its minimum value, e.g. −61 dB at the output while the power of the test tone applied to the quantizer input is −10 dB. Once the loop's notch frequency is tuned, there is room for additional (usually 3–9 dB) SQNR improvement by fine tuning the DAC coefficients and excess loop delay. At the end of the tuning process, the SQNR improves from 32 dB up to 82 dB. The calibration process takes over 20 iterations; each of the iterations consists of an FFT analysis of the output stream and measurement of the power of the testing tones.

Since the loop tuning approach relies on power estimation in software and on the well controlled frequency of the test tone, the algorithm is quite robust and ensures the optimization of the most critical parameter in the bandpass ADCs: the noise transfer function. Notice in (4.5) and (4.7) and Fig. 4.7 that the power of the tone applied at the input of the ADC at 200 MHz is almost insensitive to the tuning of the loop parameters suggesting that it is very difficult to calibrate the ADC loop

by injecting testing signals at the ADC's input. This result is expected since the closed loop gain is close to unity in the band where the filter's gain is large.

Case of Study: A 200 MHz Bandpass ΣΔ Modulator [29, 30]. A 6th-order bandpass continuous-time ΣΔ modulator achieving a peak signal to noise plus distortion ratio (SNDR) of 68.4 dB when measured in 10 MHz bandwidth was designed. With no specific location of the intermediate frequency in all wireless standards, a 200 MHz frequency was chosen to avoid the effects of flicker noise as well as to push the state of art for the ADC design in standard TSMC CMOS 0.18 μm technology. The 10 MHz bandwidth was selected to accommodate the bandwidth requirements for video applications.

System Considerations. In order to achieve the required signal-to-noise and distortion-ratio (SNDR), a detailed and practical planning on all different non-idealities such as quantization noise, jitter noise, thermal noise and building block non-linearities are needed.

Quantization noise. The signal-to-quantization noise ratio is normally overdesigned to ensure that quantization noise only contributes a small portion of the noise budget. By employing the fixed OSR = 40, a 4th-order architecture with a 2-bit quantizer and DACs will only give us 75 dB SQNR, which is slighter larger than the 70 dB target, and will make the other specifications of noise budget too difficult to be realized. As a result, the 6th-order architecture with 2-bit quantizer and DACs is chosen.

Jitter noise. Clock jitter effects are significant due to the high clock frequency used. To reduce the jitter noise contribution, non-return-to-zero (NRZ) DACs are employed. In this design, an off-chip clock is used; hence the clock jitter is fundamentally limited by the performance of the external clock generator. By employing fs/4 architecture with an NRZ DAC, the signal-to-jitter noise ratio can be estimated as [2, 14, 15]

$$\mathrm{SNR}_{\mathrm{jitter}} = 10^{*}\log_{10}\left(\frac{2^{*}\mathrm{OSR}}{\pi} * \frac{\mathrm{T_s}}{\sigma_{\mathrm{t}}}\right)^{2} \tag{4.12}$$

where σ_{t} is the standard deviation of the clock jitter. For $\sigma_{\mathrm{t}} = 0.4\,\mathrm{ps}$, the achievable $\mathrm{SNR}_{\mathrm{jitter}}$ is around 79 dB. Architectures with low sensitivity to clock jitter were reported in [8, 11, 14].

Thermal noise. Resistors and OPAMPs contribute to the overall thermal noise measured at the modulator's output. However, due to the large in-band gain in the filter's first stage, most of the thermal noise is contributed by the first biquadratic filter and the first DAC. To achieve the specifications, the signal-to-thermal noise-ratio has to be over 76 dB. Hence, the input referred noise of the system has to be less than 8 $\mathrm{nV/Hz^{1/2}}$ when a 10 MHz bandwidth is considered and the full scale range is 250 mV. For that purpose, the passives must be carefully computed, specifically the resistance assigned to the input resistor, $\mathrm{R_g}$ in Fig. 4.8a. Design details can be found in [31]. DAC output thermal noise is limited to 4 $\mathrm{nV/Hz^{1/2}}$.

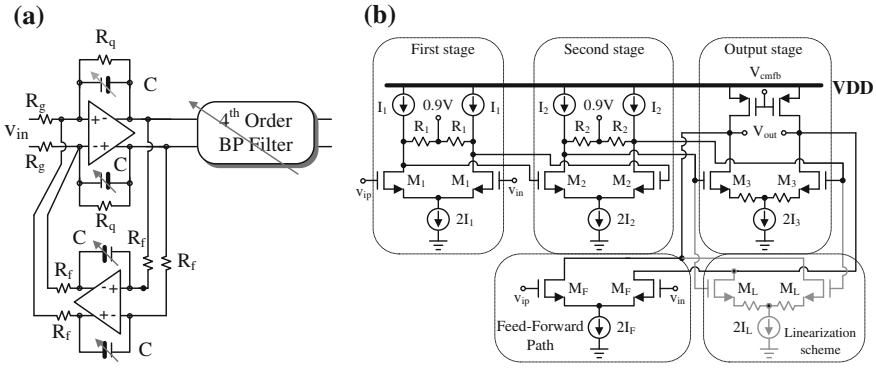

Fig. 4.8 6th order bandpass filter. a RC filter architecture b linearized OPAMP

Filter Linearity. After considering all noise contributions, the signal-to-distortion ratio (SDR) would be in the range of 76 dB. The linearity requirement for the first biquad is very demanding and required the design of a very linear OPAMP with large gain at 200 MHz. A 3-stage OPAMP architecture with gain of 20 dB at 200 MHz in the first 2-stages is used, see Fig. 4.8. The amplifier employs the feed-forward compensation that introduces high frequency zeros to provide enough phase margin [39]. The last stage processes the largest signal and is linearized employing light source degeneration as well as cross-coupled techniques that significantly suppress the 3rd order distortion [40]. Transistors M_L are smaller than M3, but with similar linear range. The first two-stages are resistive terminated and do not require any common-mode feedback circuitry. The P-MOS transistors of the last stage are used for controlling the common-mode level.

Injection of the calibration tones. As depicted in Fig. 4.9, in addition to the conventional quantizer input stage, another input stage is added to inject the test tones during the calibration stage. A source degenerated stage is added to combine the input tones and combine them with the signal generated by the loop filter to drive the flash quantizer. Linear range for this stage is $>200\,\mathrm{mV_{peak}}$.

Excess loop delay. The excess loop delay is compensated employing an array of 16 inverters that provide the same number of delayed clock options, see Fig. 4.9. The delay-line outputs are fed into a 4-bit router that selects one of them according to the 4-bit control signal generated by the LMS algorithm. The loop is tested for different clock phases until the one that results in the best possible performance is found.

DAC Linearity. DAC linearity is a critical issue [9, 10, 14]. Solutions such as noise-shaping dynamic element matching (DEM), tree-structure DEM, and the data weighted averaging technique were proposed in these [23], [24] and [25], respectively, to reduce the DAC linearity degradation from mismatch. In the described architecture, the DAC is implemented employing 5 current-sources

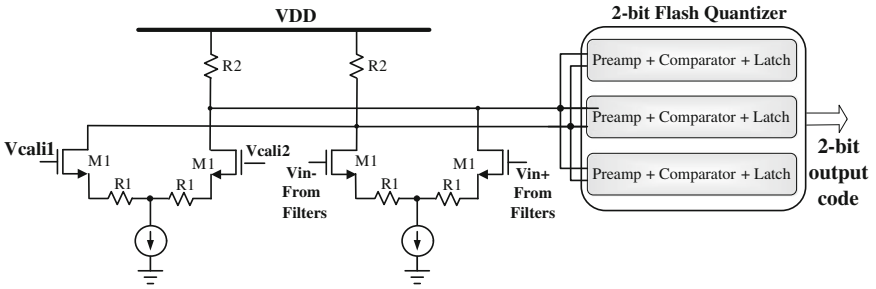

Fig. 4.9 2-bit quantizer showing the circuitry used for the test-tone injection

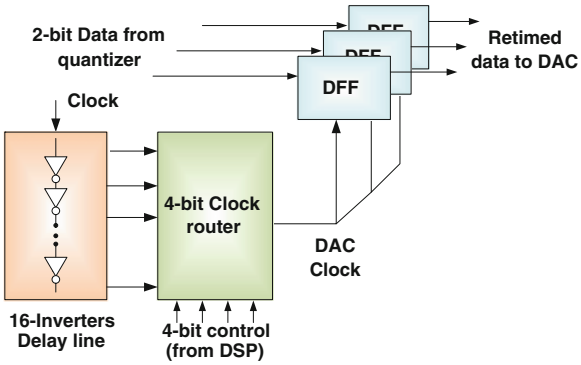

Fig. 4.10 Tunable DAC clock employing a delay line and a clock router driven by the control

instead of the 3 needed for a 2-bit DAC. These current sources are rotated by-1 every clock cycle such that the mismatch errors are partially randomized. A simple rotational pointer selects the 3-selected current sources to be modulated by the quantizer output Fig. 4.10.

Experimental Results. The BP ΣΔ modulator was fabricated in the TSMC 0.18 μm 1P6 M CMOS technology; Fig. 4.11 shows the chip microphotograph. The modulator occupies an active area of 2.48 mm^2. The total power consumption including clock buffers is 160 mW; static power consumption is 126 mW from a single supply voltage of 1.8 V. The 2-bit thermometer modulator output codes are captured by using an external oscilloscope synchronized at 800 Msample/sec and then post-processed using Matlab.

During modulator calibration, two extra signal tones are injected at the quantizer input at 220 MHz and 180 MHz; a 200 MHz tone was also injected at the modulator input. By detecting the power difference of these two tones at the output spectrum, the RC filter time constants are tuned; the modulator spectrum before and after calibration is shown in Fig. 4.12. The power difference of the tones (Fig. 4.12a) indicates that the loop filter's center frequency must be increased, which is done by reducing the value of the filter's capacitors. The algorithm

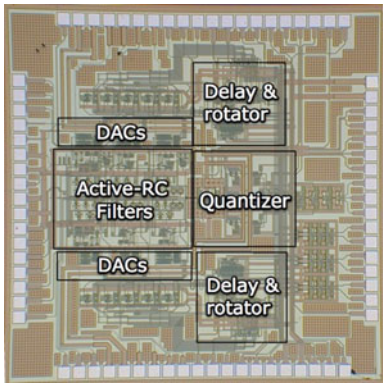

Fig. 4.11 Microphotograph of the chip

continues running until the power of the calibration tones are equalized, as depicted in Fig. 4.12b.

The measured RMS noise floor is around −100 dBr while noise resolution bandwidth is 20 kHz. It results in over 70 dB peak SNR when measured in 10 MHz bandwidth. The modulator's inter-modulation distortion is measured employing a two-tone test signal, 1.56 MHz apart from each other with a power of −8 dBr (measured with respect to the reference voltage of 250 mV). The measured third-order intermodulation distortion is −73.5 dB. The SNDR behavior with different input signal levels is illustrated in Fig. 4.13.

Table 4.1 provides a comparison of the proposed architecture with previously reported modulators and ADCs. In the last column, the topologies are compared using the classic figure of merit:

$$FoM = \frac{Power}{2^{ENOB} * (2^{*}BW)} \tag{4.13}$$

4.6 RF-Bandpass ΣΔ Modulator

The RF-to-Digital converter is the key block to implement the true software-radio as envisioned in [41]. Significant efforts have been devoted to reach this goal; some examples can be found in [1, 2, 32–38]. A high-performance receiver prototype for multi-standard communication systems based on a high-resolution bandpass Sigma-Delta Analog-to-Digital Converter (ΣΔ-ADC) is shown in Fig. 4.14 [37]. The receiver employs a linear broadband low-noise amplifier (LNA) and a programmable high-resolution bandpass ΣΔ-ADC to acquire a number of standards, and selects the desired channel by using a flexible frequency synthesizer. A 4th order continuous time LC bandpass sigma delta ADC is employed. The programmable ADC uses LC tanks, and its center frequency is tuned by controlling a bank of binary weighted capacitors. A couple of non-

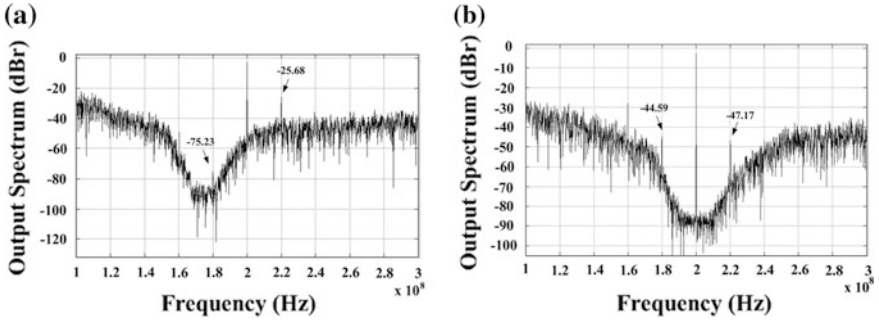

Fig. 4.12 Measured calibration process; a initial condition, and b after 20 iterations

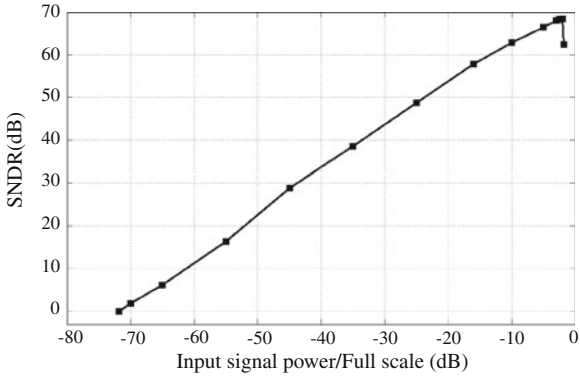

Fig. 4.13 SNDR versus input signal at 200 MHz

invasive programmable resonators placed in front of the ADC attenuate the power of the blockers that could alias into the signal band after sampling.

The architecture is tuned, stabilized and optimized with the aforementioned software-based calibration scheme. The aim of the calibration scheme is to optimize the noise-transfer function through a digital least mean square algorithm. The architecture operating in the range of 2–4 GHz has been designed and is currently under fabrication in the TSMC−130 nm CMOS technology. It is expected to achieve 10 bits resolution when measured in a signal bandwidth of 20 MHz. The architecture power consumption is under 500 mW.

The system level of the RF-ADC is illustrated in Fig. 4.15. Channel selection is achieved by varying the capacitors in the LC tank; however, only varying the capacitors will not achieve optimal system performance, so the feedback DACs and overall system loop delay are also tunable to achieve the best performance attainable from the system. This is a two-resonator system with multiple feedback paths to control the coefficients in the required transfer function.

As an example, Fig. 4.16 shows the output spectrum when the system is configured to convert the channel centered around 2 GHz where the system is

Table 4.1 Comparison with previously reported BP ΣΔ modulators

Reference	Technology	Fs	IF	BW	Peak SNDR	IM3	Power	Area mm^2	FoM (pJ/bit)
[37] BP	CMOS; 0.18 um	60 MHz	40 MHz	2.5 MHz	69 dB	–	150mW	–	13
[38] BP	CMOS; 0.35 um	240 MHz	60 MHz	1.25 MHz	52 dB	−51 dB	37mW	1.2	45.5
[31] BP^{Δ}	CMOS; 0.18 um	264 MHz	44 MHz	8.5 MHz	71 $dB^{\#}$	−72 dB	375mW*	2.5	7.6*
[33] BP	CMOS; 0.35 um	60 MHz	40 MHz	1 MHz	63 $dB^{\#}$	68 dB	16mW	0.44	6.69
[35] BP	SiGe; 0.25 um	3800 MHz	950 MHz	1 MHz	59 dB	−62 dB	75mW**	1.08	51.5**
[36] BP	SiGe; 0.13 um	40 GHz	2000 MHz	60 MHz	55 dB	–	1.6 W*	2.4	29*
This work	CMOS; 0.18 um	800 MHz	200 MHz	10 MHz	68.4 dB	−73.5 dB	160mW*	2.48	3.72*

$^{\Delta}$: This is an I/Q realization using an off-chip inductor; $^{\#}$: $SNDR = \frac{SignalPower}{Noise+IM3-component}$

*: doesn't include the power consumption of clock generator; **: only static power consumption

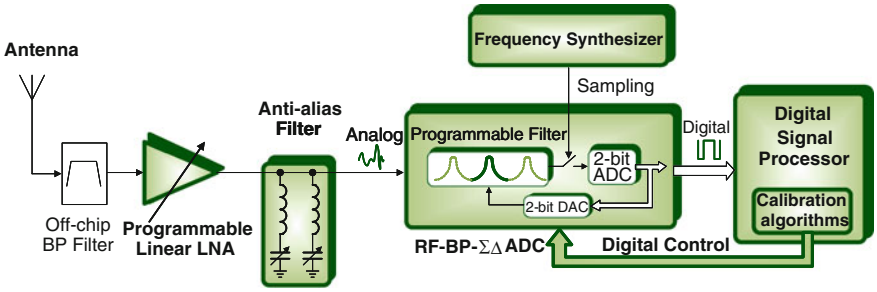

Fig. 4.14 RF-to-Digital converter employing an anti-alias filter and the BP-ΣΔ modulator

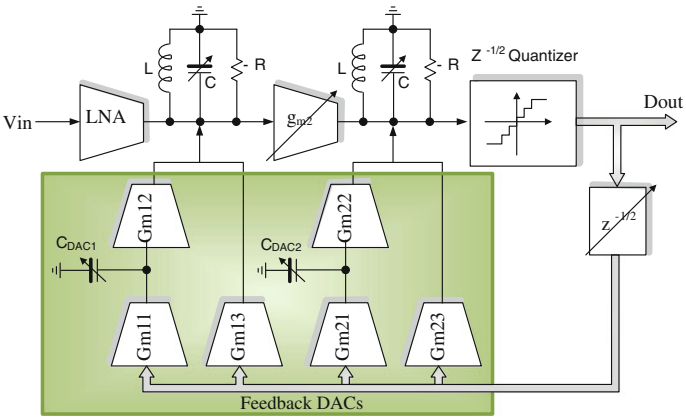

Fig. 4.15 Schematic for the 4th order RF BP-ΣΔ modulator

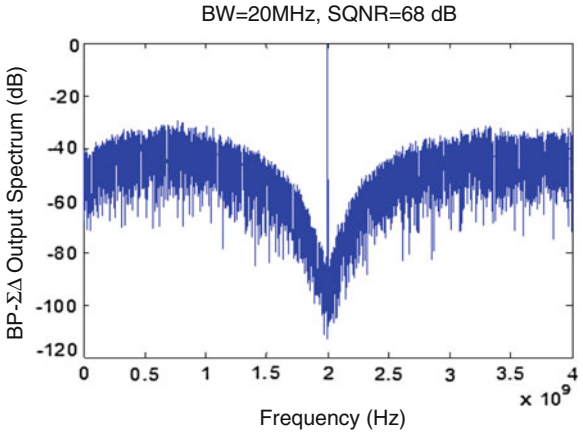

Fig. 4.16 Output Spectrum for $F_0 = 2$ GHz and Fs = 8 GHz

sampling at 8 GHz. For a 20 MHz bandwidth (1.99–2.01 GHz), 68 dB of SQNR is obtained from simulations. By properly adjusting the parameters in the loop as well as the sampling frequency, we can convert any frequency band in the 2–4 GHz range. It is expected to achieve SQNR values in the range of 60 dB due to inaccuracies in loop parameters as well as due to thermal and jitter induced noise components.

Ideally, the RF-ADC architecture could be used to convert any frequency band directly to digital; however, in a real application, transistor speed and parasitic capacitances will limit the tuning range and maximum frequency of the system. In this design, the limiting factor might be the capacitor bank. If a wide tuning range is desired, then the tradeoff is the tuning step-size will be very coarse, and certain channels would likely be skipped. In order for a continuous range of frequencies to be selected, the tuning steps need to be smaller. Doing this reduces the maximum overall tuning range due to the parasitic capacitors involved with the switches. For this system, centered at 2.75 GHz, a tuning range of ± 27 % can be achieved (2–3.5 GHz).

4.7 Conclusions

A review of the fundamental aspects of SD modulators and a general scheme including the effects of out-of-band signals was presented in Sect. 4.3. PVT variations on both magnitude and phase response are quantified. Excess loop delay is quantified too, and handy expressions are provided. As a conclusion of that section is the need for a global calibration scheme. A digitally aided loop calibration scheme that optimizes system performance is described in Sect. 4.4. The proposed scheme employs a test signature injected at the quantizer input to precisely measure and calibrate the NTF using an LMS algorithm. The proposed technique requires extensive digital computation since the power of the calibration tone must be extracted through an FFT, but this is not a major drawback since digital processing is well suited for current and future deep-submicron technologies wherein digital circuitry is becoming faster and cheaper.

The scheme is experimentally tested in a 200 MHz IF sixth-order continuous-time bandpass $\Sigma\Delta$ modulator with 800 MHz sampling clock implemented in 0.18 μm CMOS technology. The modulator achieves a peak SNDR of 68.4 dB in a 10 MHz bandwidth and a remarkable FoM of 3.72 pJ/bit which outperforms the previously reported architectures. The total power consumption is 160 mW from a single 1.8 V power supply.

Finally, the potential realization of RF-to-Digital Converters was discussed in Chap. 6. It is evident that the BP-$\Sigma\Delta$ modulators can operate properly at RF frequencies; however, it is not evident that they can be flexible enough to cover broadband applications. It is expected that faster, digital-intensive future technologies help in making more efficient these architectures. As a general

conclusion, excess loop delay, clock jitter and DAC linearity issues become even more relevant when designing faster broadband digitizers.

Acknowledgments The research on baseband modulators was partially sponsored by the Semiconductor Research Corporation under task number 1836.038. The research devoted to bandpass modulators was sponsored by NSF under Award Number 0824031. Authors would like to recognize the support of TSMC and Jazz Semiconductor for chip fabrication.

References

1. Schreier, R., Temes, G.: Understanding Delta-Sigma Data Converters. Wiley;IEE Press, Hobken, NJ (2005)
2. Cherry, J.A., Snelgrove, W.M.: Continuous time ΔΣ modulators for high speed A/D conversion—Theory, practice and fundamental performance limits, 1st edn. Kluwer Academic Publishers, Norwell, MA (2000)
3. van de Plassche, R.: Integrated Analog-to-Digital and Digital-to-Analog Converters. Kluwer Academic Publishers, Norwell, Massachusetts (2003)
4. Norsworthy, S.R., Schreier, R., Temes, G.C.: Delta-Sigma Data Converters: Theory, Design, and Simulation. Wiley-IEEE Press, New York, NY (1996)
5. Malla, P., et al.: A 28 mW spectrum-sensing reconfigurable 20 MHz 72 dB-SNR 70 dB-SNDR DT ΔΣ ADC for 802.11n/WiMAX receivers. In: IEEE ISSCC Digest of Technical Papers, pp. 496–497. San Francisco, CA (2008)
6. Breems, L.J., et al.: A 56 mW CT quadrature cascaded ΣΔ modulator with 77 dB DR in a near zero-IF 20 MHz band. In: IEEE ISSCC Digest of Technical Papers, pp. 238–239. San Francisco, CA (2007)
7. Mitteregger, G., et al.: A 20 mW 640 MHz CMOS continuous-time ΣΔ ADC with 20-MHz signal bandwidth, 80-dB dynamic range and 12-bit ENOB. IEEE J. Solid-State Circuits **41**(12), 2641–2649 (2006)
8. Straayer, M.Z., Perrott, M.H.: A 12-bit, 10-MHz Bandwidth, Continuous-Time ΣΔ ADC with a 5-bit, 950-MS/s VCO-based Quantizer. IEEE J. Solid-State Circuits **43**(4), 805–814 (2008)
9. Yang, W., et.al.: A 100mW 10 MHz-BW CT ΔΣ modulator with 87 dB DR and 91dBc IMD, In: IEEE ISSCC Digest of Technical Papers, pp. 498–631 (2008)
10. Fogleman, E., Galton, I.: A dynamic element matching technique for reduced-distortion multibit quantization in delta-sigma ADCs. IEEE Trans. Circuits Syst. II **48**(2), 158–170 (2001)
11. Colodro, F., Torralba, A.: New continuous-time multibit sigma-delta modulators with low sensitivity to clock jitter. IEEE Trans. Circuits and Systems I **56**(1), 74–83 (2009)
12. Dhanasekaran, V., Gambhir, M., Elsayed, M., Sanchez-Sinencio, E., Silva-Martinez, J., Mishra, C., Chen, L., Pankratz, E.: A 20 MHz BW 68 dB DR CT ΔΣ ADC based on a multi-bit time-domain quantizer and feedback element. In: IEEE ISSCC Digest of Technical. Papers, pp. 174–175 (2009)
13. Lu, C.-Y., Onabajo, M., Gadde, V., Chen, H.-P., Lo, Y.C., Periasamy, V., Silva-Martinez, J.: A 25 MHz bandwidth 5th-order continuous-time lowpass sigma-delta modulator with 69 dB dynamic range using time-domain quantization and feedback. IEEE J. Solid-State Circuits **45**, 1795–1808 (2010)
14. Løkken, I., Vinje, A., Hernes, B., Sæther, T.: Review and advances in delta-sigma DAC error estimation based on additive noise modeling. Analog Integr. Circ. Sig. Process. **62**(2), 179–192 (2010)
15. De la Rosa, J.: Sigma-Delta Modulators: Tutorial Overview, Design Guide, and State-of-the-Art Survey. IEEE Transactions on Circuits and Systems-I Regular Papers **58**, 1–21 (2011)

16. Gailus, P.H., Turney, W.J., Yester.Jr, F.R.: Method and arrangement for a sigma-delta converter for bandpass signals. US Patent 4857928, 15 Aug 1989
17. Engelen, van J.A.E.P, Plassche, van de R.J., Stikvoort, E., Venes, A.G.: A sixth-order continuous-time bandpass sigma-delta modulator for digital radio IF. IEEE J. Solid-State Circuits **34**(12), 1573–1764 (1999)
18. Maurino, R., Mole, P.: A 200 MHz IF 11-bit fourth-order bandpass ΣΔ ADC in SiGe. IEEE J. Solid-State Circuits **35,** 959–2640 (2000)
19. Bernardinis, G., Borghetti, F., Ferragina, V., Fornasari, A., Gatti, U., Malcovati, P., Maloberti, F.: A wide-band 280-MHz four-path time-interleaved bandpass sigma-delta modulator. IEEE Trans. Circuits and Systems **53**, 1423–1432 (2006)
20. Schreier, R., Nazmy, A., Shibata, H., Paterson, D., Rose, S., Mehr, I., Lu, Q.: A 375-mW quadrature bandpass ΔΣ ADC with 8.5-MHz BW and 90-dB DR at 44 MHz. IEEE J. Solid-State Circuits **41**, 2632–2640 (2006)
21. Frank, H., Langmann, U.: Excess loop delay effects in continuous-time quadrature bandpass sigma-delta modulators. In: Proceedings of the IEEE international symposium on circuits and systems, vol. 1, pp 1029–1032 (2003)
22. Cherry, J.A., Snelgrove, W.M.: Excess loop delay in continuous-time delta-sigma modulators. IEEE Trans on Circuits and Systems - II **46**, 376–389 (1999)
23. Yasuda, A., Tanimoto, H., Iida, T.: A third-order Δ-Σ modulator using second-order noise-shaping dynamic element matching. IEEE J. Solid-State Circuits **33**(12), 1879–1886 (1998)
24. Aghdam, E.N., Benabes, P.: Higher order dynamic element matching by shortened tree-structure in delta-sigma modulators. In: Proceedings of the European conference on circuit theory and design, Sept 2005, pp. 201–204
25. Radke, R.E., Eshraghi, A., Fiez, T.S.: A 14-bit current-mode ΣΔ DAC based upon rotated data weighted averaging. IEEE J. Solid-State Circuits **35**, 1074–1084 (2000)
26. Huang, L.H., Lee, E.K.F.: A 1.2 V direct background digital tuned continuous-time bandpass sigma-delta modulator. In: Proceedings of the 27th European Solid-State Circuits Conference ESSCIRC, pp. 526–529 (2001)
27. Rutten, R., Breems, L.J., Wetzker, G.: Digital Calibration of a Continuous-Time Cascaded ΣΔ Modulator based on Variance Derivative Estimation. In: Proceedings of the IEEE European solid-state circuits conference, Sept 2006, pp. 199–202
28. Shim, J.H., Park, I.-C., Kim, B.: A hybrid delta-sigma modulator with adaptive calibration. Proceedings of IEEE-ISCAS, pp. 1025–1028 (2003)
29. Silva-Rivas, F., Lu, C.-Y., Kode, P., Thandri, B.K., Silva-Martinez, J.: Digital Based Calibration Technique for Continuous-Time Bandpass Sigma-Delta Analog-to-Digital Converters. Analog Integr. Circ. Sig. Process **59**, 91–95 (2009)
30. Lu, C.-Y., Silva-Rivas, F., Kode, P., Silva-Martinez, J., Hoyos, S.: A 6^{th}-order 200 MHz IF Bandpass Sigma-Delta Modulator With over 68 dB SNDR in 10 MHz Bandwidth. IEEE J. Solid-State Circuits **45**(6), 1122–1136 (2010)
31. Silva-Martinez, J., Lu, C.-Y., Onabajo, M., Silva-Rivas, F., Hoyos, S.: Broadband high-resolution bandpass sigma-delta modulator with a software based calibration scheme. In: Iniewski, K. (ed.) Circuits for Emerging Technologies. CRC Press, Boca Raton (2011)
32. Thandri, B.K., Silva-Martinez, J.: A 63 dB 75 mW bandpass ΣΔ RF ADC at 950 MHz using 3.8 GHz clock in 0.25 μm SiGe BiCMOS technology. IEEE J. Solid-State Circuits **42**, 269–279 (2007)
33. Chavatzis, T., Gagnon, E., Repeta, M., Voinigescu, S.P.: A low noise 40 Gs/s continuous time bandpass ΣΔ ADC centered at 2 GHz for direct sampling receivers. IEEE J. Solid-State Circuits **42**, 1065–1075 (2007)
34. Ryckaert,J. et al.: A 2.4 GHz low-power sixth-order RF bandpass ΣΔ converter in CMOS. IEEE J. Solid-State Circuits **44**, 2873–2880 (2009)
35. Giannini, V., et al.: A 2 mm^2 0.1–5 GHz Software-Defined Radio Receiver in 45 nm Digital CMOS. IEEE J. Solid-State Circuits **44**, 3486–3498 (2009)

36. Beilleau, N., Aboushady, H., Montaudon, F., Cathelin, A.: A 1.3 V 26 mW 3.2 Gs/s under sampled LC bandpass ADC for a SDR ISM band receiver in 130 nm CMOS. In: Proceedings of IEEE RFIC, pp. 383–386 (2009)
37. Silva-Martinez, J., Hoyos, S., Mincey, J., Lo, Y.-C., Lu, C.-Y., Silva-Rivas, F.: Digitally Assisted RF-to-Digital Bandpass Converters for Broadband Communication Systems, Short Course, IEEE 2011 RF-IC Workshop New Architectures for Digitized Receivers. Baltimore Maryland, June (2011)
38. Shibata, H., et. al.: A DC-to-1 GHz tunable RF ΔΣ ADC achieving DR = 74 dB and BW = 150 MHz at f0 = 450 MHz using 550 mW. In: IEEE ISSCC Digest of Technical Papers, pp. 150–151 (2012)
39. Thandri, B.K., Silva-Martinez, J.: A robust feedforward compensation scheme for multi-stage operational transconductance amplifiers with no miller capacitors. IEEE J. Solid-State Circuits **38**, 237–243 (2003)
40. Lewinski, A., Silva-Martinez, J.: OTA linearity enhancement technique for high frequency applications with IM3 below -65 dB. IEEE Trans. Circuits Syst. II. Analog Digit Signal Processing **51**, 542–548 (2004)
41. Mitola III, J., Maguire Jr, G.Q.: Cognitive radio: Making software radios more personal. IEEE Pers. Commun. **6**, 13–18 (1999)

Chapter 5
Incremental and Extended-Range Data Converters

Gabor C. Temes

Abstract Incremental data converters (IDCs) are *Nyquist-rate* analog-to-digital data converters (ADCs) which use oversampling and noise shaping to convert a finite number of analog samples into a single digital word. Thus, they are a hybrid of Nyquist-rate and noise-shaping (ΔΣ) ADCs. This makes IDCs useful in applications where very high resolution (16–24 bits) is required, and simple structure and low power are also desirable. By combining the IDC with a serial ADC, such as a successive-approximation register (SAR) ADC, an *extended-range ADC* results which has increased accuracy.

5.1 High-Accuracy A/D Converters

Figure 5.1 illustrates the operating regions of various ADCs, as functions of the sampling rate and resolution. In high-accuracy applications (16 bits and over), the choice is between dual-slope, noise-shaping (ΔΣ) and incremental converters. Their relative advantages and disadvantages are listed below:

Dual-slope ADCs: these converters are based on integrating both the input voltage and the reference voltage in the same stage. When the integrated charges are equal, the ratio of the input and reference voltages can be obtained from the ratio of the integration time periods. Simple structure is possible and no pre- or post-filtering is needed, *but* an extremely large number of clock periods is required for conversion: the number is $M \sim 2^{N+1}$, where N is the resolution in bits. Thus M may be in the millions for high-accuracy applications.

ΔΣ ADCs: these converters use a combination of oversampling and quantization noise shaping to achieve high accuracy [1]. Fewer clock periods are needed

G. C. Temes (✉)
Oregon State University, 3091 Kelley Engineering Center, Corvallis, OR, USA
e-mail: temes@eecs.oregonstate.edu

P. Carbone et al. (eds.), *Design, Modeling and Testing of Data Converters*, Signals and Communication Technology, DOI: 10.1007/978-3-642-39655-7_5,

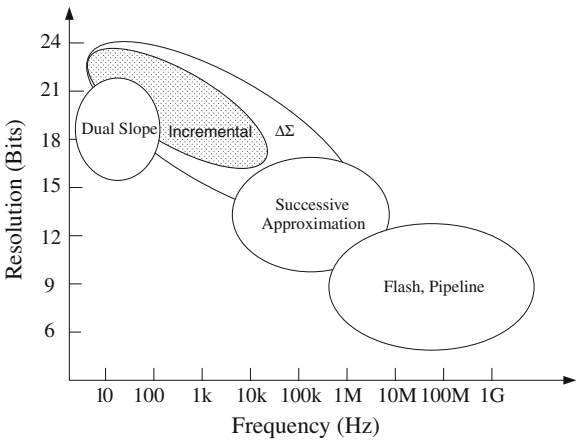

Fig. 5.1 The operational regions of various ADC structures

per output word than for dual-slope ADCs; now $M = OSR = f_s/f_N$, where f_s is the sampling frequency and f_N the Nyquist rate of the signal. Typically, the oversampling ratio *OSR* is between 64 and 256. The ΔΣ ADC needs more complicated structure than a dual-slope one, and it also needs a fairly elaborate digital post filter which introduces latency. It is also subject to instability, idle tones, mismatch effects, and other non-idealities [1].

Incremental ADCs (*IDCs*): like the ΔΣ ADCs, IDCs also need approximately $M = OSR$ clock periods to generate an output word. The IDC is structurally a ΔΣ ADC in which all memory elements are periodically reset, so their analog complexities are comparable. However, the digital filter needed by an IDC is much simpler, and it introduces only minimal latency. Also, thanks to the reset operations, IDCs can provide one-to-one mapping of the analog input samples into digital words. IDCs can also easily be multiplexed, and are less prone to instability and idle tones than their ΔΣ ADC counterparts. However, they provide a lower signal-to-quantization-noise ration (SQNR) for usual values of *OSR*. The SQNR can be boosted by *extended-range* operation, described later.

Typical applications for IDCs include instrumentation and measurement devices, where high accuracy and low power are important requirements; multichannel systems, where the ease of multiplexing is a key advantage; also, control systems, where the low latency is important for stability. Generally, the signal bandwidth in these applications is small, allowing high OSR and resolution.

5.2 Single-Stage Incremental ADCs

Figure 5.2 illustrates the block diagram of a single-stage ADC. The analog input $u(k)$ is converted during M clock periods by the ΔΣ ADC, and the digital output words are processed by the decimation filter DF. The last (Mth) output of the DF

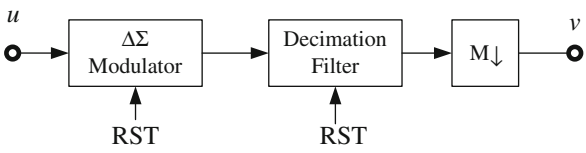

Fig. 5.2 The block diagram of a single-stage incremental ADC

gives the nth digital output word $v(n)$. After $v(n)$ is obtained, all storage elements (capacitors, registers) are reset to zero, and a new conversion cycle starts. The signals in the IDC are illustrated in Fig. 5.3. Note that there will be a gap between the conversion cycles. Often in single-channel operation this gap is only one clock period long, needed to allow the reset to take place. However, it is also possible to introduce a "sleep" period of several cycles between conversions to save power. In multi-channel ADCs, each channel has a (usually long) gap between its conversion operations while the other channels perform data conversion.

Figure 5.4a shows a first-order IDC [2] with an input voltage $u = v_{in}$. The negative feedback loop causes the output of the comparator to balance the effect of the input signal v_{in}. Every time the integrator output v crosses the 0 volt reference of the comparator, a negative pulse is fed into the integrator. Figure 5.4b illustrates the waveforms for a small positive constant input $v_{in} \ll V_{ref}$. After M steps, the output of the integrator is given by

$$v = Mv_{in} - NV_{ref} \tag{5.1}$$

where N is the number of zero crossings by v. From (5.1),

$$v_{in} = (N/M)V_{ref} + v/M \tag{5.2}$$

Since v is bounded ($|v| < V_{ref}$), for a sufficiently large number M of steps the input voltage v_{in} can be found accurately from the count N.

The operation of the circuit of Fig. 5.4a can be regarded as that of a dual-slope ADC, where the charging and discharging of the integrator are intermixed in time. Also, the circuitry is the same as that of a first-order single-bit ΔΣ ADC, operated for a limited number of clock periods. The structure of the first-order IDC is simple, and its operation stable and robust. However, the number of clock periods required to obtain a high resolution is high; $M \sim 2^N$. This is only half as many as for a dual slope converter since a two-phase clock is used, but it is still too high for most applications. Also, since the integrator is only used for a limited number of clock cycles, its effective dc gain is finite even for ideal operation. This finite gain introduces dead zones into the conversion characteristics, similar to those caused by finite op-amp gain in a ΔΣ ADC [1]. These may be eliminated by a dither signal, which changes the states in the circuit in a random fashion.

Improved operation and signal-to-quantization-noise ratio can be expected by increasing the order L of the loop filter, and/or the number of levels of the internal quantizer to a higher value than two. As an illustration, Fig. 5.5 shows the block diagram of a third-order IDC [3]. It is equivalent to a third-order ΔΣ ADC, with a

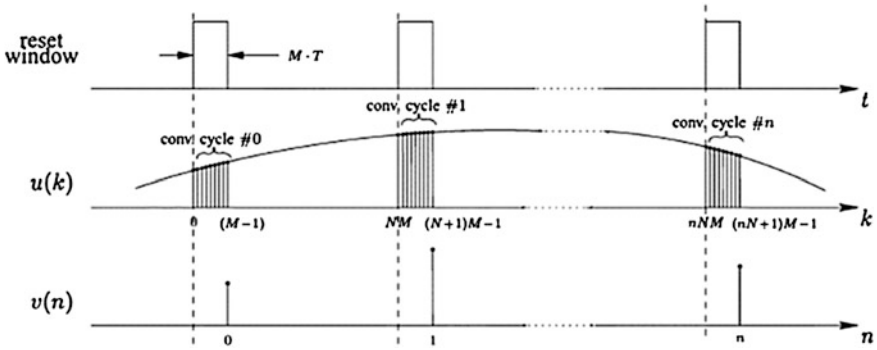

Fig. 5.3 Signal waveforms in an IDC

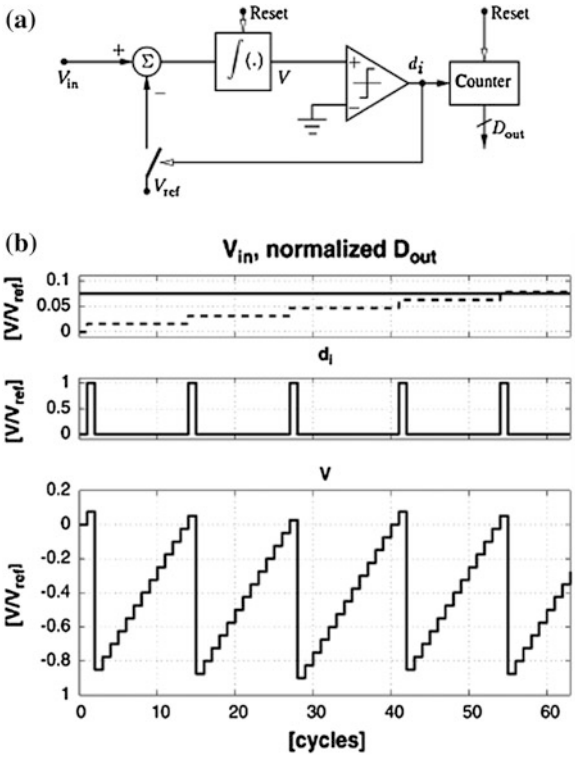

Fig. 5.4 **a** First-order incremental converter, **b** Signal waveform ($n_{bit} = 6$ bits, $V_{in} = 0.75V_{ref}$)

cascade-of-integrators feed-forward (CIFF) loop filter [1]. It also contains a feed-forward path from the input node to the quantizer. This insures that the loop filter only processes the quantization error $q(k)$, and not the input signal V_{in} [4]. Assuming a constant input V_{in}, detailed analysis in the time domain [3, 5] reveals that after M clock periods the output voltage of the last integrator is given by

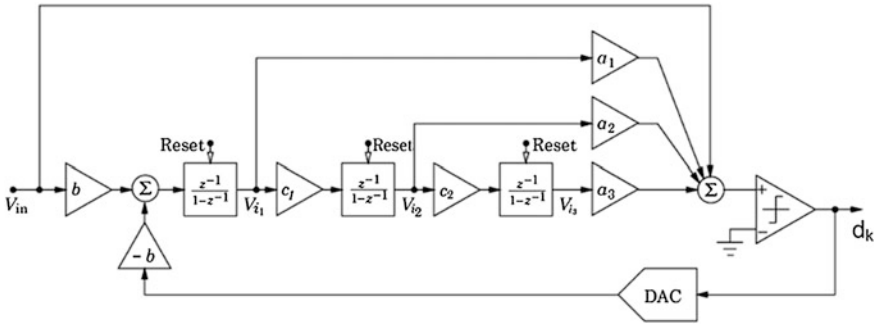

Fig. 5.5 Block diagram of a third-order ΔΣ loop

$$V_{i3}[M] \approx c_2 c_1 b \sum_{n=1}^{M} \sum_{l=0}^{n-1} \sum_{k=0}^{l-1} \left(V_{in}[k] - d_k \cdot V_{ref} \right) \tag{5.3}$$

Here, d_k is the quantizer's digital output in the kth clock period. For a single-bit quantizer, d_k may only be 0 or 1; for an N-bit one, it may be a fraction of the form $k/2^N$, $k = 1, 2, \ldots, 2^N - 1$. Assuming a constant V_{in}, (5.3) gives

$$\frac{M(M-1)(M-2)}{3!} \frac{V_{in}}{V_{ref}} - \sum_{n=1}^{M} \sum_{l=0}^{n-1} \sum_{k=0}^{l-1} d_k = \frac{1}{c_2 c_1 b} \frac{V_{i3}}{V_{ref}} \tag{5.4}$$

Thus, an estimate of $V_{in}[k]/V_{ref}$ can be found from

$$\frac{V_{in}}{V_{ref}} \approx \frac{3!}{M(M-1)(M-2)} \sum_{n=1}^{M} \sum_{l=0}^{n-1} \sum_{k=0}^{l-1} d_k \tag{5.5}$$

The error term can be bounded by assuming that for stable operation after M clock periods the output of the last integrator satisfies the condition $|v_3(M)| < V_{ref}$. Then, the LSB value corresponding to the estimation error is at most

$$V_{LSB} \approx \frac{3!}{M(M-1)(M-2)} \frac{1}{c_2 c_1 b} V_{ref} \tag{5.6}$$

Equation (5.6) can be used to obtain an initial estimate of the number of clock periods needed to obtain a desired accuracy. An efficient design process [5] uses the following steps:

1. The maximum permissible value of V_{in}/V_{ref} which prevents overloading of the quantizer is estimated. For a second-order ADC, this ratio may be around 0.9; while for a third-order one, it may be only 0.7.
2. The IDC loop is designed, and the scale factors b, c_1 and c_2 found using dynamic range optimization [1].

3. M is estimated from Eq. (5.6) for the specified conversion accuracy.

Note that the error estimate of (5.6) does not explicitly contain the resolution R of the internal quantizer. However, the product of the scale factors b, c_1 and c_2 increases linearly with R, and hence V_{LSB} decreases proportionately.

An alternative error bound may be obtained by analyzing the IDC in the z domain. Let $U(z) = V_{in}(z)$ denote the z-transform of the input signal, $Y(z) = D(z)$ that of the output signal, and $Q(z)$ of the quantization error. Using the customary linear approximation [1], and defining the *signal transfer function* $STF(z) = Y(z)/U(z)$ with $Q(z) = 0$, and the *noise transfer function* $NTF(z) = Y(z)/Q(z)$ with $U(z) = 0$, the output can be written as

$$Y(z) = D(z) = STF.V_{in}(z) + NTF.Q(z) \tag{5.7}$$

For the low-distortion configuration, $STF(z) = 1$. Using the maximally flat $NTF(z) = (1 - z^{-1})^L$, a possible transfer function for the decimation filter is $H(z) = 1/NTF(z) = (1 - z^{-1})^{-L}$. Then, the overall output in the z domain is given by

$$V(z) = H(z) \cdot Y(z) = (1 - z^{-1})^{-L} \cdot U(z) + Q(z) \tag{5.8}$$

In the time domain,

$$v[M] = \sum_{n=1}^{M} \sum_{l=0}^{n-1} \sum_{k=0}^{l} v_{in}[k] + q[M] \tag{5.9}$$

Thus, the error between $v(M)$ and the triple-integrated input is given by the last value of the quantization error. This makes the relation between the resolution of the internal quantization and the conversion error more explicit than in Eq. (5.6).

Note that the effects of constant delays were ignored for simplicity in the above analysis.

5.3 Decimation Filter Design for Single-Stage IDCs

The design of the analog loop for the IDC is similar to that for a ΔΣ ADC. However, the design of the decimation filter is quite different. Equations (5.5) and (5.9) indicate that a digital estimate of the input voltage v_{in} of the third-order IDC can be obtained simply as a triple sum of the digital output $y(k)$, scaled by $3!/(M - 2)(M - 1)M$. In general, for an Lth-order loop, L accumulators may be used to find the digital estimate, with a scale factor $SF = L!/[(M - L + 1)(M - L + 2) \ldots (M - 1)M]$. Alternatively, instead of using accumulators, it is also possible to implement the filter directly by the convolution of $y(k)$ with the impulse response $h(k)$ of the decimation filter. For $L = 1$, $h(k) = 1$ for all $k = 0, 1, \ldots, M - 1$. For $L = 2$, $h(k) = k + 1$; for $L = 3$, $h(k) = (k + 1)(k + 2)/2$, etc. The scale factor involving M also needs to be included. It is efficient to choose M as a

power of 2, and to use the approximations $1/(M-k) \cong (1/M)(1+k/M)$, $k = 1, 2, \ldots L-1$.The accumulator- or convolution-based decimation filter is simple to design and to implement, and it guarantees a specified worst-case error bound, but it is not exactly optimal.

References [6] and [7] contain sophisticated filter design processes based on information theory which achieve minimum quantization error by some performance index. However, a simpler design technique, which also allows the effect of thermal noise to be minimized, is also available [8]. It is described next, in terms of the M-element sequences associated with the operation of the IDC. The following notations are used:

$u(k) = (u_0, u_1, \ldots, u_{M-1})$ is the sequence of the M samples of the input signal of the loop;

$q(k)$ and $t(k)$ are similar representations of the quantization error and the input-referred noise of the loop, respectively. Also, $y(k)$ and $v(k)$ represent the digital outputs of the loop and the decimation filter, respectively.

$s(k) = (s_0, s_1, \ldots, s_{M-1})$ is the sequence of the impulse response of the signal path to the output of the loop. It is the inverse transform of the signal transfer function $STF(z)$ of the loop, windowed by the reset pulse.

$n(k) = (n_0, n_1, \ldots, n_{M-1})$ is the impulse response of the quantization noise transfer function, from the quantizer to the output of the loop. It is the inverse transform of the noise transfer function $NTF(z)$ of the loop, windowed by the reset pulse.

$h(k) = (h_0, h_1, \ldots, h_{M-1})$ is the impulse response of the decimation filter.

By Eqs. (5.7) and (5.8), these finite-length (M sample long) sequences satisfy

$$y(k) = s(k) * [u(k) + t(k)] + n(k) * q(k) \tag{5.10}$$

and

$$\begin{aligned} v(k) &= h(k) * y(k) \\ &= h(k) * s(k) * u(k) + h(k) * s(k) * t(k) + h(k) * n(k) * q(k) \end{aligned} \tag{5.11}$$

Note that the factors and the results of these convolutions are all M sample long sequences. Hence, each of the three terms on the RHS of (5.11) contain the sum of M terms.

The first term $h(k) * s(k) * u(k)$ gives the contribution of the input signal $u(k)$ to the overall output. It is a weighted sum of the input samples, with the weight factors given by the samples of $h(k) * s(k)$. To achieve a dc signal gain of 1 ($STF(1) = 1$), the sum of the M samples of $h(k) * s(k)$ must equal to 1. For the low-distortion loop, where $STF(z) = 1$, the weight factors are simply the samples of $h(k)$.

The second term $h(k) * s(k) * t(k)$ (or simply $h(k) * t(k)$ for low-distortion IDCs) gives the contribution of the input-referred thermal noise to the output. The weight factors are the same as for $u(k)$ in the first term.

Finally, the last term $h(k) * n(k) * q(k)$ represents the contribution of the quantization error to the output word $v(k)$.The weight factors are $h(k) * n(k)$.

Assuming that the signal-band gain of the decimation filter is 1, and that the reference voltages of the quantizer and DAC both have the value 1 volt, the maximum signal power in the output can be estimated from the largest peak-to-peak value V_{pp} of a sine-wave which does not overload the quantizer. This gives $P_s = V_{pp}^2/2$ as the maximum signal power.

To estimate the power of the thermal noise in the output, it will be assumed that $t(k)$ is due to the sampled switch noise at the input terminal, and hence it is a zero-mean noise with a mean-square voltage $\gamma kT/C_{in}$. Here, k is the Boltzmann constant, T is the temperature in K^0, and γ *is* a scale factor determined by the circuitry of the input branch [1]. The expression $h(k) * s(k) * t(k)$ indicates that the output noise is a weighted sum of the samples $t(k)$. Since these samples are nearly completely uncorrelated, their mean-square value is given by the sum of the mean-square values of the individual terms. Detailed analysis [8] shows that the mean-square value of the output thermal noise is

$$P_t = (\gamma kT/C_{in})\boldsymbol{h}^T\boldsymbol{S}^T\boldsymbol{S}\boldsymbol{h} \tag{5.12}$$

Here, $\boldsymbol{h}$ is an M-element column vector whose kth element is the sample $h(k)$, and $\boldsymbol{S}$ is an MxM lower triangular matrix generated from the samples of $s(k)$ [8].

The nth row of $\boldsymbol{S}$ is $[s(n-1)\ s(n-2)\ \ldots s(0)\ 0\ 0\ \ldots\ 0]$. For the low-distortion loop, $\boldsymbol{S}$ becomes the unit matrix, and hence

$$P_t = (\gamma kT/C_{in})|\boldsymbol{h}|^2 \tag{5.13}$$

The estimation of the power of the contribution of the quantization error in the output is similar to the one performed above for the noise. It will be assumed that the samples $q(k)$ behave as a zero-mean noise with uncorrelated samples, and have a mean-square value of $\Delta^2/12$, where Δ is the step size of the quantizer. (Note that this assumption rests on conditions which insure the randomness, and may necessitate the use of a dither signal in the loop.) Then, defining the $M \times M$ matrix $\boldsymbol{N}$ generated from the $n(k)$ samples the same way as $\boldsymbol{S}$ was generated from the $s(k)$, the power P_q of the output quantization noise can be expressed in the form

$$P_q = (\Delta^2/12)\boldsymbol{h}^T\boldsymbol{N}^T\boldsymbol{N}\boldsymbol{h} \tag{5.14}$$

The optimization of the digital filter transfer function $H(z)$ is based on minimizing the weighted sum of P_t and P_q subject to the prescribed dc gain $H(1)$ of the filter.

Specifying $H(1) = 1$, so that also $h(0) + h(1) + \ldots h(M-1) = 1$, and defining the matrix

$$\boldsymbol{O} = \frac{\gamma}{\boldsymbol{kT}}\boldsymbol{C_{in}}\boldsymbol{S}^T\boldsymbol{S} + \frac{\Delta^2}{12}\boldsymbol{N}^T\boldsymbol{N} \tag{5.15}$$

the task becomes finding $\boldsymbol{h}$ so as to achieve

$$\min_h \overline{v_n^2} = \boldsymbol{h}^T \cdot \boldsymbol{O} \cdot \boldsymbol{h} \tag{5.16}$$

subject to

$$\boldsymbol{e}.\boldsymbol{h} = 1 \tag{5.17}$$

where $\mathbf{e} = (1\ 1\ 1\ \ldots\ 1)$ is an M-element row vector.

It is interesting to analyze the extreme cases when only P_t or P_q is minimized. *In the former case*, the minimization of the thermal noise P_t with the overall dc signal gain equal to 1 gives $s(k)*h(k) = 1/M$ for $k = 0, 1, \ldots M - 1$. For the low-distortion case, all $h(k) = 1/M$. Thus, all tap weights of the optimized decimation filter are the same for thermal noise minimization if the low-distortion architecture is used.

In the latter case, the quantization noise power P_q given by (5.14) needs to be minimized, subject to constraint (5.17). This task can be performed analytically, using the Lagrange multiplication method [8]. The resulting optimum impulse response of the decimation filter is given by

$$\boldsymbol{h}* = \boldsymbol{Re}/\boldsymbol{e}^{\mathrm{T}}\boldsymbol{Re} \tag{5.18}$$

where $\boldsymbol{R} = [\boldsymbol{N}^{\boldsymbol{T}}\boldsymbol{N}]^{-\mathbf{1}}$ and $\boldsymbol{e}$ the unit-element vector defined above. Due to the structure of $\boldsymbol{N}$, the matrix $\boldsymbol{N}^{\boldsymbol{T}}\boldsymbol{N}$ cannot be singular, and thus $\boldsymbol{R}$ always exists. Alternatively, available software (e.g., the MATLAB function "quadprog") can be used to find $\boldsymbol{h}^*$. It can be predicted that for an Lth-order loop the first L elements of $\boldsymbol{h}^*$ will be zero or very small, because the last L output samples of the loop will contain quantization error samples $y(k)$ which will not be cancelled by subsequent samples. Hence, these must have small weight factors in the optimum solution. These weight factors are $s(k)*h(k)$ for $k = 0, 1, 2, \ldots, L - 1$. Since $s(0) = 1$ for the low-distortion structure, $h(k) \sim 0$ follows for $k = 0, 1, 2, \ldots, L - 1$.

For the general case, the process described for the minimization of P_q remains applicable, and $\boldsymbol{h}^*$ is still given by (5.18), but now $\boldsymbol{R} = [\boldsymbol{O}^{\boldsymbol{T}}\boldsymbol{O}]^{-\mathbf{1}}$ holds, where $\boldsymbol{O}$ is given by Eq. (5.15). The optimum impulse response $h(k)$ of the digital filter will now be a compromise between the ones applicable in the extreme cases discussed above. The situation is illustrated in Fig. 5.6 for the example discussed in [8]. The curves show the optimum $h(k)$ responses for minimizing the thermal noise power P_t (broken curve), P_q (dotted curve), and the total output noise (continuous curve). Note that the areas under the three curves are the same, but the individual properties differ, as discussed above.

The decimation filter DF performs the convolution of the loop output data $y(k)$ with the FIR impulse response $h(k)$, discussed above. It needs to be implemented in an economical way. Since the output $v(k)$ of the DF is down-sampled by M, only the last result of the convolution needs to be calculated. Thus the required result is

$$v(n) = h(k) * y(k)|_{k=n(M-1)} \tag{5.19}$$

The M required coefficients $h(k)$, $k = 0, 1, \ldots., M - 1$, can be stored, and a simple multiply-accumulate (MAC) stage may be used to carry out the calculation

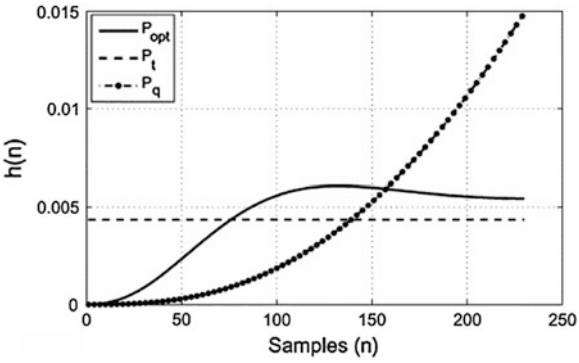

Fig. 5.6 Impulse response of the optimal decimation filter for a third-order IDC

of $v(n)$. Since usually the IDC quantizer has low resolution, y(k) may only be a simple rational number such as 0, ±1/2, ±¾, ±1, etc., making the MAC operations trivial.

In some applications, the decimation filter needs to suppress one or more interferers (e.g., line noise). In this case, the design may be based on the sinc function, which can provide transmission zeros at arbitrary frequencies [3].

5.4 Multi-Stage Incremental ADCs

Similarly to a ΔΣ ADC, the SQNR of the IDC may be improved by increasing the order L, the oversampling ratio M, and the resolution of the internal quantizer. However, for wide-band ADCs the OSR M may be limited to a low value by (say) the bandwidth of the amplifiers, or by the allowable power dissipation. For low oversampling ratios, the SQNR cannot be significantly improved by raising the order of the loop filter, and high SQNR can only be obtained by using impractically high quantizer resolution. The situation is illustrated in Fig. 5.7 for ΔΣ ADCs [9]. Note that the noise shaping achieved by the first-order ΔΣ ADC is very small, and to obtain low in-band noise (and thus high SNR) we need an eighth-order one. The situation is similar for IDCs.

The problems presented by the low OSR may be solved by utilizing the multi-stage ("MASH") architecture [1]. Here, the quantization error Q_1 of the first stage is cancelled by the output of the second stage, the error Q_2 the second stage by the output of the third stage, etc. The principle is illustrated for a two-stage ADC in Fig. 5.8. Here, the analog quantization error q_1 of the first stage is transmitted to a second stage, where it is converted into digital form. The digital outputs of the two stages are then combined using the error-cancelling filters H_1 and H_2. Linear analysis shows that if the digital filters satisfy the condition

$$H_1 \cdot NTF_1 = H_2 \cdot STF_2 \tag{5.20}$$

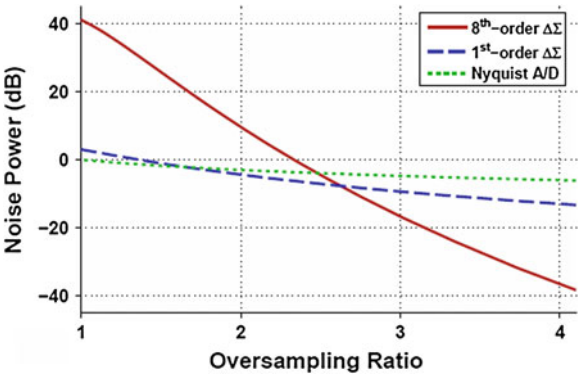

Fig. 5.7 SQNR vs OSR curves for various quantizer levels [9]. For OSR $= 3 \sim 4$, very high order is required to obtain a high SNR

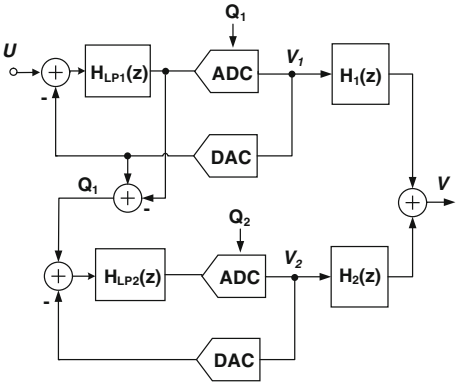

Fig. 5.8 Cascade (MASH) ΔΣ modulator

then Q_1 is cancelled in the resulting output signal. Choosing $H_1 = STF_2$ and $H_2 = NTF_1$, the output is given by

$$V = STF_1 \cdot STF_2 \cdot U - NTF_1 \cdot NTF_2 \cdot Q_2 \tag{5.21}$$

This shows that Q_1, the first-stage quantization error, was indeed cancelled in V, and that Q_2 is filtered by the *product* NTF_1. NTF_2 of the noise transfer functions of the individual stages. Thus, high-order noise shaping may be obtained while using low-order individual loops. In addition, if the first loop contains a multi-bit quantizer, then the error q_1 will be smaller than the full-scale voltage of the circuitry. Thus, q_1 may be amplified by a gain $A > 1$ before entering it into the second stage, and therefore an attenuation $1/A$ can also be applied to the second-stage output. In this case, the error term in V becomes $NTF_1.NTF_2.(1/A).Q_2$, improving the SQNR even more. The MASH scheme can extended to three or more ΔΣ stages as well.

While MASH was developed originally for ΔΣ DACs and ADCs, it is applicable to IDCs as well. Reference [10] describes a two-stage MASH IDC, where the second stage operates all the time, from the second clock period until period

$(M + 1)$. More IDC stages may also be cascaded; [9] describes an IDC containing 8 stages, and operating at a very low oversampling ratio.

A more economical MASH IDC can be obtained by recalling from Eq. (5.4) that the total conversion error of the first loop after digital filtering is simply the scaled last quantization error $q_1(M - 1)$ generated in the loop. In general, $q_1(M - 1)$ needs to be obtained by subtracting the input of the first-stage quantizer from its output. However, for the low-distortion structure with a maximally flat $NTF(z)$, $q_1(M - 1)$ can be found simply from the output $v_3(M - 1)$ of the last integrator in the first loop. Hence, an efficient MASH IDC can use a second stage which is inactive until the clock period $M - 1$, and then it converts and scales $v_3(M - 1)$ while the output of the first stages is processed by the decimation filter. This second stage will thus produce the N_{LSB} least significant bits of the overall output word. It may even be realized by a Nyquist-rate ADC (e.g., a successive-approximation ADC), and the operation may be fully pipelined, if $N_{LSB} < M - 1$. Multi-stage IDCs based on the principle described in this paragraph [11–15] are often called *extended-range* or *extended-counting* ADCs.

5.5 Examples of Incremental ADC Design

In this Section, three recently published IDCs will be described, to illustrate the use of these data converters.

(*a*) *A Single-Stage* 22*-bit IDC* [3]. The block diagram of this third-order IDC is shown in Fig. 5.5, the simplified circuit schematic in Fig. 5.9. The coefficients for various scaling conditions are given in [15]. A single-bit loop quantizer was used, to prevent nonlinear distortion due to mismatch in a multi-bit DAC. The chip used an enhanced chopper-stabilization method (fractal sequencing) to cancel the dc offset of the input stage at all integrator outputs. It used a dynamic element matching scheme to reduce the input amplitude by a factor 2/3.

The decimation filter was a $sinc^4$ programmable digital filter, with staggered zeros around the line frequency (50 and 60 Hz) and its harmonics.

The output noise and INL are shown for the input voltage range in Figs. 5.10 and 5.11. A summary of the performance is given in Table 5.1.

(*b*) *An Extended-Range Wide-Band IDC* [14]. Figure 5.12 shows the block diagram of an extended-range second-order IDC designed for biosensor arrays. Following the steps described in Sec. II, it can be shown that choosing the impulse response of the decimation filter as $h(k) = [2/M(M - 1)](k + 1)$ for $k = 0, 1, \ldots, M - 1$, the quantization error in the output of the decimation filter is

$$E_Q = s_{in} - s_{out} = \frac{2}{a_1 a_2 M(M-1)} w(M) \tag{5.22}$$

The second stage of the modulator was a successive-approximation ADC, with a dual-array capacitor DAC. It was used to convert the residue $w(k)$ into a 9-bit

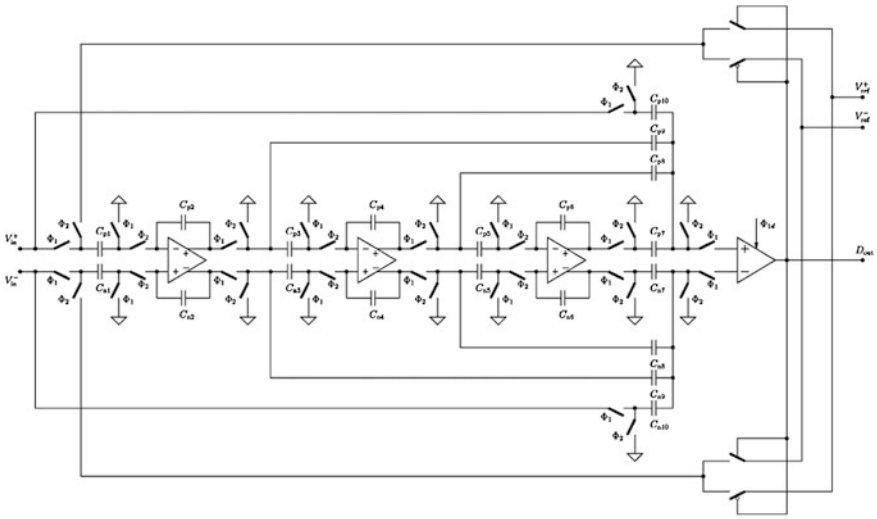

Fig. 5.9 Circuit diagram of a third-order IDC

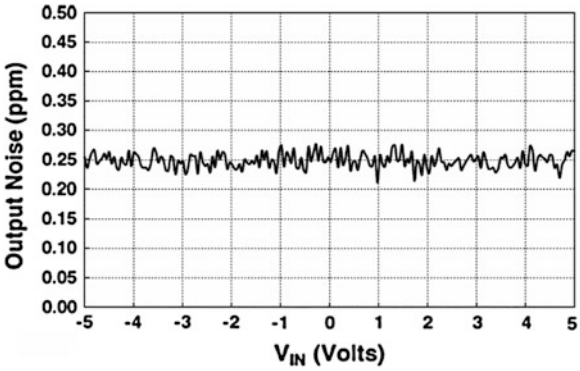

Fig. 5.10 Measured rms output noise versus input signal voltage for the third-order IDC (Ref [3].)

digital word. An OSR of 45 was chosen, which gave a theoretical SQNR of 98 dB. The measured performance is shown in Table 5.2.

(*c*) *A 12-bit 7uW/Channel 1 kHz/Channel Incremental ADC for Biosensor Interface Circuits* [16]. Figure 5.13 shows the block diagram and switched-capacitor circuits of an incremental ADC embedded in a biosensor interface chip. It uses a noise-coupled multi-bit ΔΣ loop, integrated with a novel digital decimation filter operated in near-threshold. It was realized in the IBM 90 nm CMOS technology. The fabricated prototype device shows a measured 74 dB SNDR up to 2 kHz (1 kHz/channel signal bandwidth) with 1 V_{pp} differential input range. The total measured power of the modulator was 13.5 μW (7 μW/Channel). The performance summary is listed on Table 5.3.

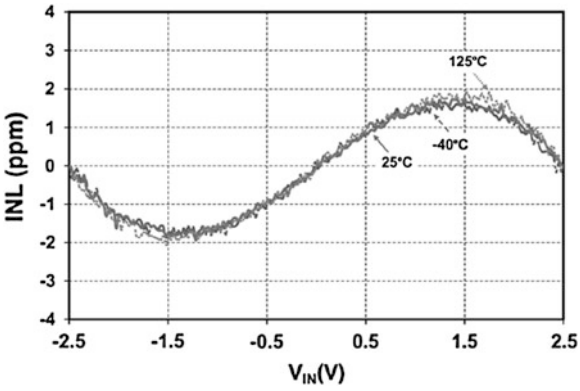

Fig. 5.11 Measured integral nonlinearity [3]

Table 5.1 Performance of the third-order IDC [3]

Parameter	Performance
Conversion time	
typ.	66.7 ms
DC offset	
typ.	2 μV (1.7 LSB)
max.	10 μV (8.4 LSB)
Gain error[a]	
typ.	2 ppm (8 LSB)
max.	10 ppm (40 LSB)
INL[a]	
$V_{ref} = 2.5$ V	Max C4 ppm (16 LSB)
$V_{ref} = 5$ V	Max 10 ppm (40 LSB)
Supply current	
Shutdown mode	Max. 1 μA
Op. mode $V_{DD} = 5$ V	typ. 120 μA
Op. mode $V_{DD} = 3$ V	typ. 100 μA
CMRR[b] @ 50/60 Hz	At least 135 dB
DC PSRR[b]	
$V_{DD} = 2.5$–6 V	At least 120 dB
Output noise[a]	
$V_{ref} = 5$ V	0.25 ppm (2.5 μV_{RMS}, or 1 LSB)
$V_{ref} = 2.5$ V	0.48 ppm (2.4 μV_{RMS}, or 2 LSB)
Oscillator frequency variation over V_{DD} and temperature range	±0.5 %

[a] ppm of $2V_{ref}$. 1 ppm = 4 LSB
[b] $V_{in}^{+} = V_{in}^{-} = V_{ref}/2$

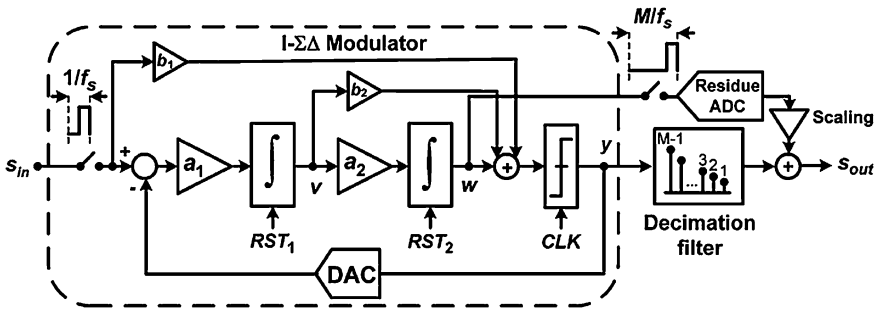

Fig. 5.12 Second-order IDC with extended-counting range [14]

Table 5.2 Performance summary of the extended-range IDC [14]

Technology	0.18 μm CMOS
Power Supply	1.8 V
Max sampling rate	45.2 MHz
Oversampling Ratio	45
Input Sampling Capacitor	7.1 pF
$SNR_{max}/SNDR_{max}$	89.1 dB/86.3 dB
SFDR	94.5
Dynamic Range	90.1 db
DNL/INL	<0.45 LSB/<1.0 LSB
Power (excluding output drivers)	33.4 mW (Analog) 4.7 mW (Digital)
FoM($P_{tot}/(2.BW.2^{ENOB})$	1.46 pJ/conv.
Active area	3.5 mm^2

Table 5.3 Performance summary of the biosensor IDC [16]

Process	IBM 90 nm
Sampling frequency	1 MHZ
Number of channels	2
OSR	256
Single bandwidth	977 Hz/channel
Over sampling ratio	256
Peak SNDR	73.5 dB
Maximum input range	IV_{pp} differential
Power consumption of the ΔΣ Modulator	13.5 μW Analog: 7.8 μW/Digital: 5.7 μW
Estimated power of digital filter	0.5 μW
Area	200 μW × 300 μW

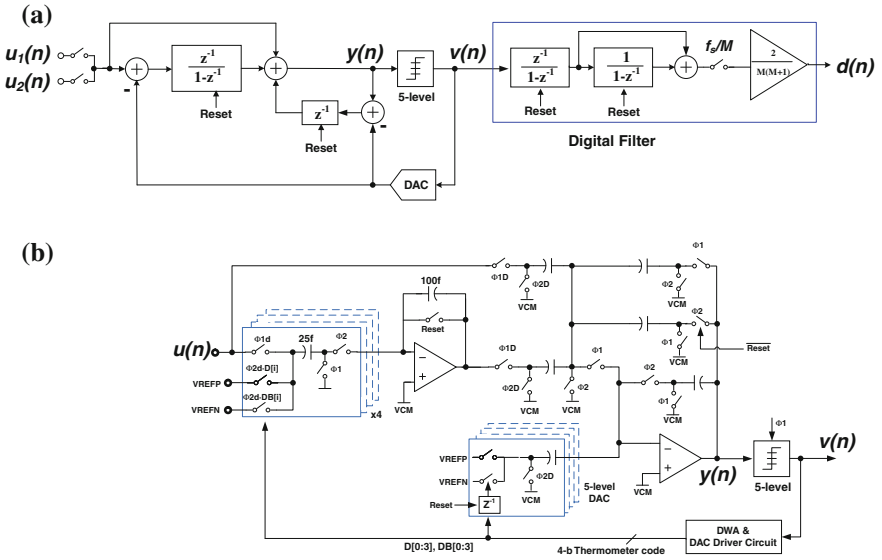

Fig. 5.13 Incremental ADC for biosensor interface circuit [16]

References

1. Schreier, R., Temes, G.C.: Understanding Delta-Sigma Data Converters. Wiley, New Jersey (2005)
2. Robert, J., Temes, G.C., Valencic, V., Dessoulavy, R., Deval, P.: A 16-bit low-voltage A/D Converter. IEEE J. Solid-State Circuits **22**, 157–163 (1987)
3. Quiquempoix, V., Deval, P., Barreto, A., Bellini, G., Collings, J., Markus, J., Silva, J., Temes, G.C.: A low-power 22-bit incremental ADC. IEEE J. Solid-State Circuits **41**, 1562–1571 (2006)
4. Silva, J., Moon, U.-K., Steensgaard, J., Temes, G.C.: A wideband low-distortion delta-sigma ADC topology. Electron. Lett. **37**(12), 737–738 (2001)
5. Markus, J., Silva, J., Temes, G.C.: Theory and applications of incremental ΔΣ converters. IEEE Trans. Circuits Syst. I **51**, 678–690 (2004)
6. Hein, S., Ibraham, K., Zakhor, A.: New properties of sigma-delta modulators with dc inputs. IEEE Trans. Commun. **40**(8), 1375–1387 (1992)
7. Kavusi, S., Kakavand, H., Gamal, A.E.: On incremental sigma-delta modulation with optimal filtering. IEEE Trans. Circuits Syst. I **53**, 1004–1015 (2006)
8. Steensgaard, J., Zhang, Z., Yu, W., Sarhegyi, A., Lucchese, L., Kim, D., Temes, G.C.: Noise-power optimization of incremental data converters. IEEE Trans. Circuits and Syst. I **55**, 1289–1296 (2008)
9. Caldwell, T. Delta-sigma modulators with low oversampling Ratios. PhD thesis, University of Toronto, Fig. 3.2
10. Robert, J., Deval, P.: A second-order high-resolution incremental A/D converter with offset and charge injection compensation. IEEE J. Solid-State Circuits **23**, 736–741 (1988)
11. Harjani, R. Lee, T.A.: FRC: a method for extending the resolution of Nyquist-rate converters using oversampling. IEEE TCAS-II **45**, 482–494 (1998)
12. De Maeyer, J., et al.: A double-sampling extended-counting ADC. IEEE J. Solid-State Circuits **39**, 411–418 (2004)

13. Rombouts, P. et al.: A 13.5-b 1.2-V micropower extended counting A/D converter. IEEE J. Solid-State Circuits **36**, 176–183 (2001)
14. Agah, A., Vleugels, K., Griffin, P., Ronaghi, M., Plummer, J., Wooley, B.: A high-resolution low-power incremental ADC with extended range for biosensor arrays. IEEE J. of Solid-State Circuits **45**, 1099–1110 (2010)
15. Markus, J.: Higher-order incremental delta-sigma analog-to-digital converters. PhD Thesis, Budapest University of Technology and Economics (1999). http://home.mit.bme.hu/~markus/pubs/markus_phd.pdf
16. Chen, Ch., Crop, J., Chae, J., Chiang, P., Temes, G.:A 12-bit 7 μW/channel 1 kHz/channel incremental ADC for biosensor interface circuits. In: Proceedings of IEEE International Symposium of Circuits and Systems, (2012)

Chapter 6
Event-Driven Successive Charge Redistribution Schemes for Clockless Analog-to-Digital Conversion

Dariusz Kościelnik and Marek Miśkowicz

Abstract The analog-to-digital conversion methods based on event-driven successive charge redistribution schemes are presented in the study. In the proposed schemes, the charge redistribution is forced by a self-timed mechanism that substitutes a clock in driving a converter operation. One of important implications is that the converter almost does not consume energy in breaks between conversion cycles that can be triggered irregularly in time.

6.1 Introduction

The ever growing demand for extending digital functionality on a single chip results in scaling the feature size of VLSI technology into nanoscale (<100 nm) to increase the integration density of semiconductor devices. A reduction of transistor dimensions enables faster operation of circuits on the one hand but forces decreasing the supply voltage of devices on the other due to a reduction of transistor gate oxide thickness. As a result of this, a design of analog and mixed signal systems has to cope with an ever increasingly challenging technological environment. This is particularly true for low supply voltages near 1 V or below.

6.1.1 ADC Design Challenges

In the context of analog-to-digital converters (ADCs), the technology scaling increases the maximum conversion rate but unfortunately decreases at the same time the signal-to-noise ratio (SNR). The latter is caused simply by a reduction of

D. Kościelnik · M. Miśkowicz (✉)
Department of Electronics, AGH University of Science and Technology, Kraków, Poland
e-mail: miskow@agh.edu.pl

D. Kościelnik
e-mail: koscieln@agh.edu.pl

P. Carbone et al. (eds.), *Design, Modeling and Testing of Data Converters*, Signals and Communication Technology, DOI: 10.1007/978-3-642-39655-7_6,

voltage increment corresponding to the least significant bit (LSB) in a signal amplitude quantization. This is currently the most serious problem of ADC design that will be even more critical in future [1].

To maintain a high SNR despite the low-voltage operation of classical ADCs, the power consumption needs to be increased [1]. However, the latter is in general unacceptable in portable equipment and in wireless sensor networking (WSNs) due to constraints on energy resources. Efficiency of power consumption becomes a primary criterion of designing ADCs for many applications. The representative examples are environmental monitoring and biomedicine. In particular, the ADCs for WSNs in biomedical applications (pulse-oximetry, ECG, PCG, EEG, blood pressure, etc.) need only modest precision (≤ 8 bit), and modest speed (≤ 40 kHz) but has to be very energy-efficient [2]. Summing up, the challenging problem of today's ADC design is a development of low voltage, low power and possibly high performance ADCs whose SNR does not decrease with supply voltage reduction.

6.1.2 Time Encoding and Asynchronous ADCs

One of important research directions emerged in the last decade is based on a postulate to encode the signals in the time domain [2–11]. Time encoding of signal values consists simply in converting amplitude information of an analog signal into time information. A discrete sample of the analog signal is then represented by a *time-difference variable* as the quantity of time between two events. Encoding signal values in time and their further processing in silicon-based information systems is a general postulate of Time Mode Signal Processing (TMSP) [6] declared independently by several authors with adopting various argumentation and terminology conventions [2–11].

In general, time encoding technique is not new. It has been used for decades for instance in multislope analog-to-digital converters, D-class amplifiers, spike-based sensors [11], data converters based on the pulse width modulation (PWM) applied in power electronics [33, 34], and in asynchronous Delta or Delta-Sigma modulations [35, 36]. In recent years, instead of an occasional use, time encoding is proposed to be an underlying principle in modern signal processing. The examples of a real-time asynchronous circuit for converting amplitude information of an analog signal into time information termed as the *time encoding machine* (TEM) are the asynchronous Sigma-Delta modulator or the frequency modulator [3]. On the other hand, the examples of biologically-inspired TEMs are spiking neurons with pulse trains outputs [3, 4]. Furthermore, the general TEM architecture for encoding space-time analog waveforms in video application as an extension of single-input–single-output (SISO) TEM that represents analog signals only in the time domain is introduced in [5]. Time encoding of signal values is a fully invertible process provided that a number of encodings per unit of time is high enough in relation to the signal bandwidth [3].

In the context of analog-to-digital conversion, the postulate of the TMSP led to introduce a new class of devices called *asynchronous analog-to-digital converters* (A-ADCs) [2–17, 38, 40, 65–67]. In the A-ADCs, the conventional signal sampling is substituted by time encoding of input signals and the quantization of their time representations [18, 19]. Moving the quantization process from the signal amplitude domain to the time domain allows to avoid the problems with decreasing accuracy of signal value quantization in the amplitude domain due to a reduction of operating voltage. The development of A-ADCs drives research on reconstruction of non-uniformly sampled signals [3–5, 14, 17, 37, 39–41] although early works addressed this problem have been known since the 1950s [42–44].

6.1.3 Principles of Successive Charge Redistribution ADCs

In the present study, we demonstrate the analog-to-digital conversion method in which the input signals originated in the voltage domain or encoded in a form of time-difference variables are first translated to the corresponding charge packet, and next processed in the charge domain by self-timed successive charge redistribution in the binary-scaled capacitor array. The analog-to-digital conversion with successive redistribution (SCR-ADC) is based on the *binary search algorithm* and belongs to *successive approximation* methods. A significant point is that the method applied to the SCR-ADC differs from the known techniques of charge redistribution used in classical DACs with the binary-weighted capacitors because the charge transfer is self-timed in the former whereas it is clock-driven in the latter. One of important implications is that the SCR-ADC almost does not consume energy between conversion cycles. Thus, the proposed converter belongs to the class of circuits of activity-dependent power dissipation [45]. The SCR-ADC can be regarded as one of propositions of introducing the event-driven architecture to analog-to-digital converters which is commonly applied by the use of time-triggered approaches.

The signals representing several physical magnitudes can be converted using the SCR-ADCs. We present architectures for the charge-to-digital, time-to-digital and voltage-to-digital conversion. In particular, the Successive Charge Redistribution Time-to-Digital Converters (SCR-TDCs) may be used for clockless time quantization in A-ADCs. In particular, together with the loss-free asynchronous analog signal recovery [3], the SCR-TDC provides possibility to establish the clockless asynchronous digital signal processing chain where digital data are produced irregularly in time according to temporal signal amplitude variations.

The SCR-ADC architecture is extremely simple, it can operate with low supply voltage and scales well with CMOS technology. The n-bit SCR-ADC is built of the binary-weighted capacitor array with $n + 1$ capacitors, two asynchronous comparators, one or two current sources, asynchronous state machine, and a group of controlled switches. Compared to the conventional successive-approximation ADCs, the SCR-ADC contains neither clock nor other time base. We discuss the

fundamental tradeoffs of SCR-ADCs design which consists in balance of conversion speed and accuracy.

The n-bit SCR-ADC operation cycle consists of three phases. In the first phase called *charge accumulation* corresponding to sampling operation, the analog signal value is translated to the corresponding charge packet. In the second phase, the charge packet is processed in the charge domain in a number of n steps by self-timed successive charge redistribution in the binary-scaled capacitor array. In each conversion step, the charge redistribution is realized by event-driven charge transfer between two capacitors (current *source capacitor* and current *destination capacitor*) to compare charge portions stored in both capacitors. A selection of current source and current destination capacitors in each conversion step and establishing the path between both capacitors to enable charge transfer is the main task of the conversion algorithm implemented in the asynchronous state machine. At the end of the charge redistribution, the output bits corresponding to these capacitors charged to a desired voltage are evaluated to 'one' whereas the other output bits are evaluated to 'zero'. When the conversion cycle is completed, all the capacitors are quickly discharged during the relaxation phase. We present two variants of the SCR-ADC conversion where respectively the charge accumulation and redistribution are proceed sequentially or concurrently. Furthermore, we provide analysis of the conversion time for both variants. Depending on the SCR-ADC variant, each conversion cycle consists just of a number of $n + 1$ or $n + 2$ state transitions in the circuit with no state transitions between conversion cycles which is a main reason of high energy efficiency of the converter.

The analog-to-digital conversion method and corresponding circuit architectures demonstrated in the present study are disclosed in recent patent applications WO 2011/152743 [20], WO 2011/152744 [21], WO 2011/152745 [22] and have been reported fragmentarily in [23–26].

6.2 Conversion Method and Circuit Architecture

6.2.1 Hydraulic Model

Let us assume that an analog input signal subjected to the conversion in the *Successive Charge Redistribution Analog-to-Digital Converter* (SCR-ADC) is charge or any physical magnitude (e.g., voltage, time, energy of particles in radiation detectors, etc.) transformed to electric charge. A translation of a signal value originated in the other domains to a portion of input charge may be referred to a sampling operation since charge is a discrete-time variable.

The concept of the analog-to-digital conversion with event-driven successive charge redistribution will be introduced on the basis of the model of discrete estimation of a liquid volume using a set of containers $Ct_4, \ldots, Ct_0$ of binary-scaled volumes (Fig. 6.1). In the translation of the electric circuitry to the

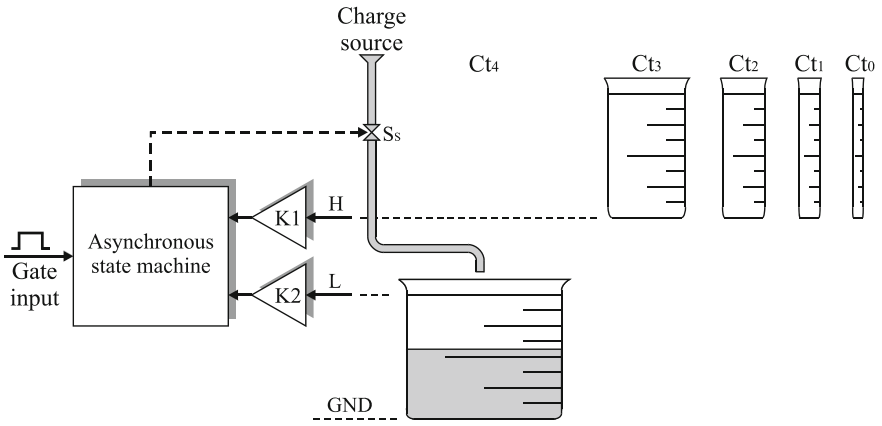

Fig. 6.1 Liquid accumulation

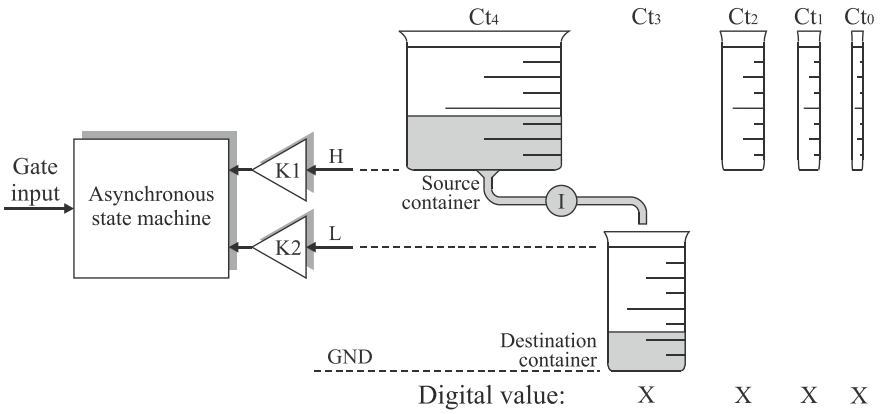

Fig. 6.2 The first step of liquid redistribution

hydraulic model, liquid is an equivalent of electric charge, containers represent capacitors, liquid levels refer to the voltages on the capacitors and the pump imitates the current source.

In the first step of the conversion model, the amount of liquid stored in the input container Ct_4 is transferred to the first auxiliary container Ct_3 whose volume is the half of the volume of the input container (Fig. 6.2). As soon as the destination container Ct_3 is filled to the mark, the liquid transfer is continued in the second step from the input container Ct_4 to the second auxiliary container Ct_2 whose volume is the one-fourth of the volume of the input container (Fig. 6.3). Let us assume that the rest of liquid stayed in the input container is too small to fill the container Ct_2 to the mark as seen in the exemplified scenario in Fig. 6.3. Then, this amount of liquid, as soon as moved to the container Ct_2, is further redistributed to

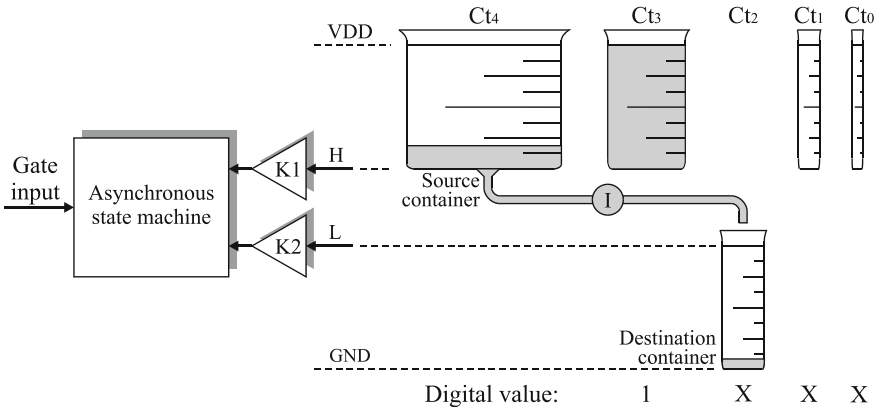

Fig. 6.3 The second step of liquid redistribution

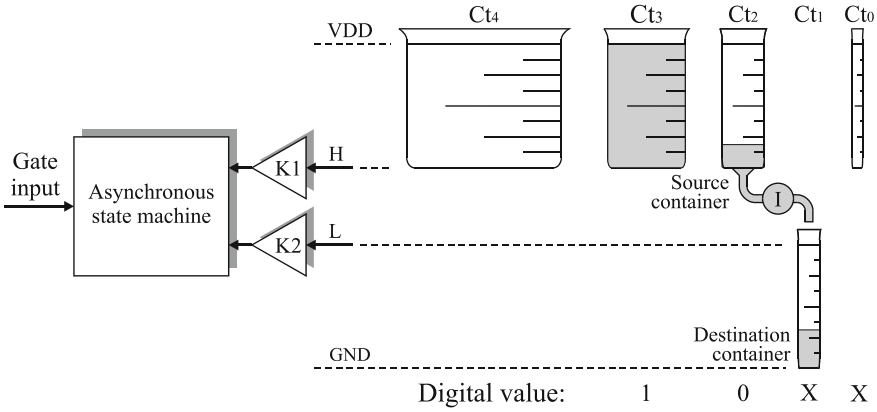

Fig. 6.4 The third step of liquid redistribution

the other containers in the order of decreasing volumes. In the third step, liquid is being transferred from the Ct_2 to the Ct_1. Since the amount of liquid stayed in the container Ct_2 is too small to fill the Ct_1, it is transferred in the fourth step from the Ct_1 to the Ct_0 (Fig. 6.4). At the end of the conversion cycle, the little amount of liquid, which is smaller than the volume of the smallest auxiliary container Ct_0, is located in the container Ct_1 (Fig. 6.5).

As follows from the provided description, the proposed conversion scheme belongs to the class of successive approximation algorithms based on the binary search principle.

The number of conversion steps is equal to the number of bits that define a resolution of the discrete estimate produced on the output. The state ‘one’ is attributed to these auxiliary containers that are filled to the mark and the state ‘zero’ accordingly to the other auxiliary containers. The most significant bit

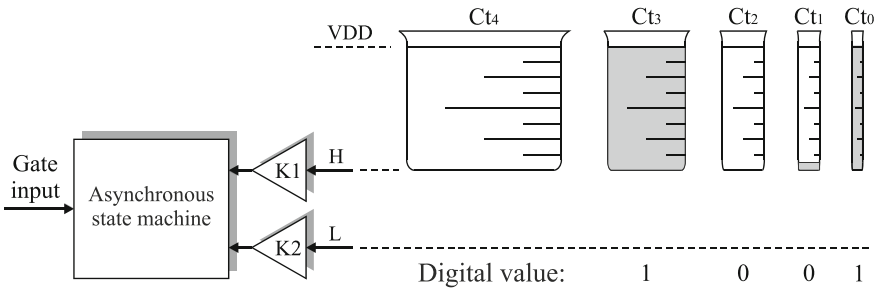

Fig. 6.5 States of redistribution system at the end of conversion cycle

(MSB) is assigned to the first auxiliary container, and the least significant bit (LSB) corresponds to the smallest auxiliary container. The comparators $K1$ and $K2$ detect respectively emptying the current source container and filling the current destination container to the mark. The former implies the evaluation of the state of the output bit corresponding to the destination capacitor to 'zero' and the latter causes the relevant bit to be set to 'one'. Thus, the digital output word '1001' is produced at the end of the conversion cycle example illustrated in Figs. 6.1, 6.2, 6.3, 6.4 and 6.5. The state machine is responsible for selecting the current source and current destination containers, and linking both containers by the pump.

In the proposed redistribution system, the liquid transfer between containers is enforced with the constant rate by the pump. In the other possible implementation variant, the liquid flow can be driven simply by gravity without the use of the pump because the destination container is placed always below the source container [68]. The analogy to the electric model implies that the charge flow can be effectively enforced also if a difference of potentials between the bottom plate of the source capacitor and the top plate of the destination capacitor is created. In the latter, the rate of liquid/charge transfer is no longer constant in time [68].

6.2.2 *Electric Diagram of SCR-ADC*

The architecture of the analog-to-digital converter with event-driven successive charge redistribution (SCR-ADC) is presented in Fig. 6.6 and the corresponding electric diagram respectively in Fig. 6.7. The block diagram of the SCR-ADC shown in Fig. 6.6 may be referred to classical charge redistribution converters [46] or even to delta modulation systems [47]. The converter has two inputs: the charge input for incoming charge packet delivery, and the gate input for providing the external signal to control a duration of the charge accumulation process. The main component of the SCR-ADC is the binary-weighted capacitor array that comprises a number of $n + 1$ cells where each cell contains a capacitor C_i and three controlled switches: $S_{Gi}, S_{Hi}, S_{Li}, i = 0, 1, \ldots, n$. In the ith cell, the capacitance C_i equals $2^i C_0$, where C_0 is the unit capacitance in the array (Fig. 6.7). The largest

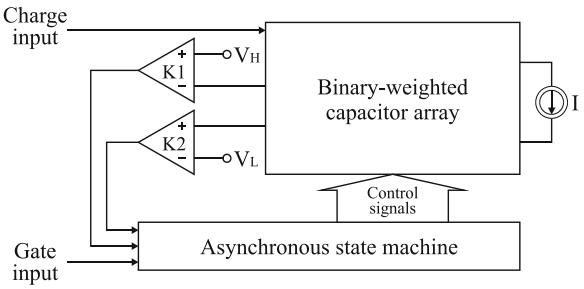

Fig. 6.6 Block diagram of event-driven successive redistribution analog-to-digital converter (SCR-ADC)

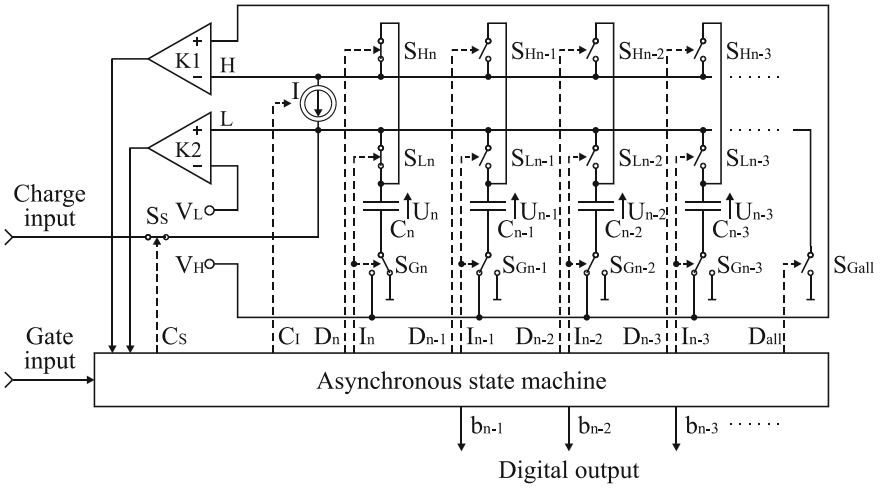

Fig. 6.7 The SCR-ADC circuit diagram. The controlled switch states correspond to the charge accumulation phase

capacitor C_n, called the input capacitor, is intended to store the input charge Q_{In} collected during the charge accumulation phase and represents an electric analogy to the input container. The working capacitors $C_{n-1}, C_{n-2}, \ldots, C_0$ as equivalents of auxiliary containers in the hydraulic model are used in the conversion process for redistribution of charge gathered previously in the input capacitor C_n. The role of switches controlled by the state machine is to select two capacitors involved currently in the charge transfer and to insert the current source I to the path linking both capacitors (Fig. 6.7). In particular, the switches enable connecting the bottom plates of the capacitors to the ground or keeping them on the desired potential V_H. The former refers to the capacitor that is a destination for charge being transferred and the latter is applied to the other capacitors since the converter architecture follows the hydraulic model also in terms of voltage arrangement (compare Figs. 6.2 to 6.7). The current source I that enforces charge redistribution acts in fact as a current stabilizer since there is a non-zero difference of potentials between the bottom plate of the source capacitor (rail H) and the top plate of the destination capacitor (rail L) during charge transfer ($V_H > V_L$, see Sect. 6.5).

6.2.3 SCR-ADC Operation Algorithm

The charge conversion cycle consists of an accumulation of incoming charge portion followed by a number of n conversion steps, and a relaxation phase in order to discharge both the input and the working capacitors. The input charge Q_{In} is accumulated in the input capacitor C_n during the time interval whose start and stop is defined by the external signal provided to the gate input.

The termination of charge accumulation begins the charge redistribution phase that consists of the number of n steps. In the first conversion step, the input charge Q_{In} is successively transferred from the input capacitor C_n to the capacitor C_{n-1}. The transfer of charge is forced by the current source I. Moving the charge results in growing the voltage U_{n-1} on the capacitor C_{n-1} and at the same time in falling the voltage U_n on the capacitor C_n. If U_{n-1} reaches the prespecified threshold V_L before U_n falls to zero, then the most significant bit b_{n-1} is set to 'one'. Next, the remaining charge stored in the input capacitor C_n is transferred in the second conversion step to the capacitor C_{n-2}. However, if the capacitor C_n is discharged before the voltage U_{n-1} reaches the threshold V_L during the first conversion step, then the most significant bit b_{n-1} is set to 'zero'. Furthermore, in the second conversion step, the capacitor C_{n-1} plays the role of the source of charge that is then transferred from the C_{n-1} to the C_{n-2}. The cycle is repeated for the subsequent steps. The states of bits of the output digital word are determined successively from the MSB to the LSB. During the ith conversion step, the states of the bits $b_{n-1}, \ldots, b_{n-i+2}, b_{n-i+1}$ are already fixed, thus each conversion step produces one bit more precise estimate of the input charge Q_{In}. The cycle is terminated as soon as the charge is stopped to be delivered to the capacitor C_0 of the lowest capacitance in the capacitor array. At the end of the conversion cycle, the small charge portion $Q_q \leq C_0 V_L$ representing the quantization error is stored in the capacitor $C_{n-\mathrm{k}}$ corresponding to the least significant bit b_{n-k} whose state has been evaluated to 'zero', or in the input capacitor C_n if all the bits in the output word have been evaluated to 'one'.

6.3 Analysis of Converter Operation

6.3.1 Conversion Principle

Similarly as presented in the hydraulic model, the conversion in the SCR-ADC consists of the sequence of charge transfers between the capacitor denoted as the *current source capacitor* and the capacitor that acts as the *current destination capacitor*. During the charge transfer in each conversion step, the comparison and conditional subtraction of two charge values are realized. Both operations are performed by monitoring the voltages on both capacitors involved in charge transfer and by detecting which event occurs first: crossings of a desired level V_L

by the voltage on the destination capacitor or zero-crossing of the voltage on the source capacitor.

In the first step, the input charge Q_{In} stored in the input capacitor is compared to $½Q$ where $Q = C_nV_L$ is the SCR-ADC input range (full scale) and V_L is the voltage level to which the capacitors are charged. At the end of the first conversion step, the one-bit quantization error $Q_q(1)$ is produced by reducing the input charge by $½Q$ provided that $Q_{In} \geq {}^1\!/_2Q$. At the same time, the most significant bit b_{n-1} is set to 'one'. On the other hand, if the input charge $Q_{In} < {}^1\!/_2Q$, the bit b_{n-1} is set to 'zero' and the input charge Q_{In} equals $Q_q(1)$ directly. Finally, it is evident that: $Q_q(1) = Q_{In} - {}^1\!/_2Qb_{n-1}$. Next, in the second step, the one-bit quantization error $Q_q(1)$ is compared to $¼Q$. At the end of the second conversion step, the 2-bit quantization error $Q_q(2)$ is produced through reducing $Q_q(1)$ by $¼\ Q$ if $Q_q(1) \geq {}^1\!/_4Q$. At the same time, the bit b_{n-2} is set to 'one'. If $Q_q(2) < {}^1\!/_2Q$, then the bit b_{n-2} is set to 'zero' and $Q_q(2)$ simply equals $Q_q(1)$. Thus: $Q_q(2) = Q_q(1) - b_{n-2}Q/4$.

Taking into account the recurrence:

$$Q_q(k) = Q_q(k-1) - b_{n-k}Q/2^k, k = 1, \ldots, n \tag{6.1}$$

it is clear that the input charge is the sum of the charge portions stored in the destination capacitors and the quantization error $Q_q(n)$:

$$Q_{In} = 2^{-1}Qb_{n-1} + 2^{-2}Qb_{n-2} + \cdots + 2^{-n}Qb_0 + Q_q(n) \tag{6.2}$$

On the other hand, the output digital word b_{n-1}, b_{n-2},..., b_0 forms the n-bit estimate of the input charge value Q_{In} with the accuracy equal to $Q_q(n)$.

6.3.2 Event-Driven Charge Transfer in SCR-ADC

The selections of the current source capacitor, current destination capacitor and establishing the path between both capacitors in each conversion step to enable charge transfer is the main task of the conversion algorithm implemented in the asynchronous state machine. These actions are carried out by the event-driven dynamic reconfiguration of states of controlled switches S_{Gi}, S_{Hi}, S_{Li} as a response to events occurring in the conversion process detected on the gate input or reported by both comparators $K1$ and $K2$. The events timing the SCR-ADC conversion cycle belong to four classes as follows (Fig. 6.8):

- detection of the start of the gate signal on the gate input triggering accumulation of charge provided to the charge input,
- detection of the stop of the gate signal on the gate input,
- detection of an instant when the voltage on the current destination capacitor reaches the desired threshold V_L which is reported by the comparator $K2$,

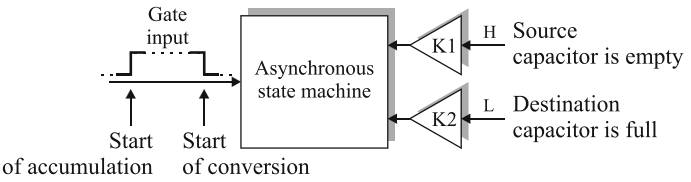

Fig. 6.8 Classes of events timing SCR-ADC conversion cycle

- detection of an instant when the current source capacitor is discharged which is reported by the comparator $K1$.

The asynchronous control logic is driven only by outputs of both comparators $K1$, $K2$ and by the start and the stop of the gate signal on the gate input. Thus, the converter is self-timed and no clock is needed to control its operation.

6.3.3 Binary Search Algorithm

As stated, the SCR-ADC adopts the binary search algorithm for estimating the input charge Q_{In}. Before starting the conversion, it is known by the assumption that the input charge does not exceed the conversion range $(0 \le Q_{In} \le Q)$ so the uncertainty interval for Q_{In} is simply $\Phi(0) = [0; Q]$.

After the first step, if b_{n-1} is evaluated to 'one', the lower bound of the uncertainty interval for Q_{In} is raised to $Q/2$ while the upper bound is kept the same $(Q/2 \le Q_{In} \le Q)$ so $\Phi(1) = [Q/2; Q]$. On the other hand, if b_{n-1} is evaluated to 'zero', the upper bound is reduced to $Q/2$ while the lower bound is kept the same, therefore $\Phi(1) = [0; Q/2]$. The size of the uncertainty interval at the end of the first step equals $\Delta Q_{In}(1) = Q/2$. In forthcoming steps, the conversion proceeds accordingly. Each step halves the size of uncertainty interval for the estimation of the input charge: $\Delta Q_{In}(k) = \Delta Q(k-1)/2$. Evaluating a state of a given bit to 'one' results in an increase of the lower bound. On the other hand, setting a bit state to 'zero' causes a decrease of the upper bound of the uncertainty interval.

Finally, after k conversion steps, the lower bound of the uncertainty interval for the input voltage value is defined by the sum of the binary-scaled components that correspond to these output bits $b_{n-1}, \ldots, b_{n-k}$ whose states have been fixed and evaluated to 'one'. On the other hand, the upper bound of the uncertainty interval equals the sum of the lower bound defined above and the weight 2^{-l} referred to the least significant bit b_l among the bits $b_{n-1}, \ldots, b_{n-k}$ whose states have been already set to 'zero' or the weight 2^{-n} if all the bits $b_{n-1}, \ldots, b_{n-k}$ have been evaluated to 'one':

$$\sum_{i=1}^{k} 2^{-i} b_{n-i} < \frac{Q_{In}}{Q} < \sum_{i=1}^{k} 2^{-i} b_{n-i} + 2^{-l} \tag{6.3}$$

where

$$l = \begin{cases} n; \text{ if } \forall l = n-k, n-k+1, \ldots, n-1 : b_l = 1 \\ \min\,(l; l = n-k, n-k+1, \ldots, n-1 : b_l = 0); \text{ otherwise} \end{cases}$$

It can be easily proved that the range of the uncertainty interval after the kth step defined as the difference between its upper and lower bound equals $V_L/2^k$ (see Table 6.1).

The uncertainty interval and its range defined by (6.3) is valid not only for the SCR-ADC but for any successive approximation converters including the classical synchronous Successive Approximation ADCs.

6.3.4 Source and Destination Capacitor Selection in SCR-ADC

In the present Section, the rules for selection of the source and destination capacitors by the state machine will be explicitly specified.

Let us denote by *Source_C*(i) the capacitor that acts as the source capacitor in the ith conversion step and accordingly by *Dest_C*(i) the relevant destination capacitor. In fact, during the charge accumulation preceding its further redistribution in the SCR-ADC, the input capacitor C_n operates as the destination capacitor for the input charge Q_{In}. In successive conversion steps, the role of the destination capacitor is taken over by the working capacitors $C_{n-1}, \ldots, C_0$ in the order of decreasing capacitances. In each conversion step, the capacitance of the source capacitor is higher than the capacitance of the destination capacitor.

The status of the first source capacitor is assigned to the input capacitor C_n. As follows from the description of the charge redistribution algorithm, the role of the source capacitor is delegated in the $(i + 1)$th conversion step to the capacitor C_{n-i} if the state of the bit b_{n-i} has been fixed and set to 'zero'. Otherwise, the index of the source capacitor is not changed in the next packet cycle. The conversion is completed after n steps when the capacitor C_0 stops to operate as the (last) destination capacitor.

In particular, if the input charge Q_{In} is large and close to the converter full scale (i.e. if the $n - 1$ most significant bits are set to 'one' in the output digital word), then the input capacitor C_n operates as the source capacitor during the whole conversion cycle. On the other hand, if $Q_{In} < C_1 V_L$ which means that Q_{In} is then lower than 2 * LSB (i.e., if the n-1 most significant bits are evaluated to 'zero' in the output digital word), the role of the source capacitor is switched successively from C_n to C_1 in each conversion step.

Table 6.1 Example of initial 6 steps of n-bit SCR-ADC conversion cycle

Stage	Conversion step index (i)	*Source_C*(i)	*Dest_C*(i)	Index of bit tested	Example of evaluated bit states	Active control signals	Lower bound of uncertainty interval ΔQ_{In} at the end of step	Upper bound of uncertainty interval ΔQ_{In} at the end of step
Relaxation	–	–	–	–	–	D_{all}, D_{n-1}, I_0,…, I_{n-1}	–	–
Accumulation	0	–	C_n	–	–	C_S, I_n, D_n	0	Q
Charge redistribution	1	C_n	C_{n-1}	n-1	1	C_I, D_n, I_{n-1}	$Q/2$	Q
	2	C_n	C_{n-2}	n-2	1	C_I, D_n, I_{n-2}	$3Q/4$	Q
	3	C_n	C_{n-3}	n-3	0	C_I, D_n, I_{n-3}	$3Q/4$	$7Q/8$
	4	C_{n-3}	C_{n-4}	n-4	1	C_I, D_{n-3}, I_{n-4}	$13Q/16$	$7Q/8$
	5	C_{n-3}	C_{n-5}	n-5	0	C_I, D_{n-3}, I_{n-5}	$13Q/16$	$27Q/32$
	6	C_{n-5}	C_{n-6}	n-6	…	C_I, D_{n-5}, I_{n-6}	…	…

6.3.4.1 Recursive Indexing of Source and Destination Capacitors

Generalizing, the index of $Dest\_C(i+1)$ is lower by one in relation to the index of $Dest\_C(i)$

$$\begin{aligned} Dest\_C(0) &= C_n \\ Dest\_C(i+1) &= Dest\_C(i) - 1 \end{aligned} \tag{6.4}$$

On the other hand, the index of the source capacitor in the next conversion step $Source\_C(i+1)$ is the same as the index of the source capacitor in the previous conversion step $Source\_C(i)$ if $b_{n-i} = 1$, or is the same as the index of the destination capacitor in the previous conversion step $Dest\_C(i)$ if $b_{n-i} = 0$:

$$Source\_C(1) = C_n$$

$$Source\_C(i+1) = \begin{cases} Source\_C(i); & \text{if } b_{n-i} = 1 \\ Dest\_C(i); & \text{if } b_{n-i} = 0. \end{cases} \tag{6.5}$$

6.3.4.2 Explicit Indexing of Source and Destination Capacitors

As follows from (6.4), the index of the destination capacitor $Dest\_C(i)$ is decremented from $n-1$ to zero in each conversion step and is independent of the states of output bits:

$$Dest\_C(i) = C_{n-i} \tag{6.6}$$

when $i = 0, 1, \ldots, n$ (the charge accumulation phase is considered as the 0th conversion step).

On the other hand, as follows from the formula (6.5) the index of the current source capacitor $Source\_C(i)$ equals the index of the least significant bit among the bits whose states have been fixed and set to 'zero', or equals the index of the input capacitor (n) if all the output bits have been evaluated to 'one' as follows:

$$Source\_C(i) = C_j, j \geq n - i + 1 \tag{6.7}$$

where

$$j = \begin{cases} n; \text{ if} \forall k = n - i + 1, n - i + 2, \ldots, n - 1 : b_k = 1 \\ \min\ (k; k = n - i + 1, n - i + 2, \ldots, n - 1 : b_k = 0); \quad \text{otherwise} \end{cases} \tag{6.8}$$

In particular, the source capacitor in the last conversion step $Source\_C(n)$ stores the small charge portion corresponding to the quantization error.

6.3.4.3 Selection of Source_C(i) and Dest_C(i) in Exemplified Cycle

In Table 6.1, the course of 6 initial steps of n-bit SCR-ADC conversion cycle is specified based on the exemplified states of output bits. In particular, note that according to the condition (6.5) $Source\_C(6) = Dest\_C(5) = C_{n-5}$ during step 6 since $b_{n-5} = 0$. Furthermore, according to the condition (6.5) $Source_C(5) = Source_C(4) = C_{n-3}$during the step 5 since $b_{n-4} = 1$. Note also that the integer $n - 3$ as the index of $Source\_C(5)$ and $Source\_C(4)$ is the least significant bit among the bits $b_{n-4}, \ldots, b_{n-1}$ whose states have been already known in step 5 and evaluated to 'zero' (see the condition (6.8)). The conversion is completed after the nth step when the role of a destination capacitor plays the capacitor C_0.

In Table 6.1, the control signals that are active in the relaxation, accumulation, and during SCR-ADC successive redistribution steps are also listed. These signals are referred to the reconfiguration of converter architecture presented in Fig. 6.7. The functionality of these signals is discussed in Sect. 6.5 in details. Furthermore, Table 6.1 contains the specification of lower and upper bounds of uncertainty interval ΔQ_{In} at the end of each conversion step (see formula (6.4)).

6.4 Event-Driven versus Classical Charge Redistribution ADCs

The binary search algorithm as a principle of successive approximation analog-to-digital conversion has been known at least since the sixteenth century [48] when it was applied to measuring unknown weights using the minimum number of weighing operations [49].

Early electronic devices adopting the binary search principle as ancestors of successive approximation ADCs were designed in Bell Telephone Laboratories for PCM systems in the 1940s. They were referred to as *sequential coders*, *feedback coders*, or *feedback subtractor coders* [48]. In particular, in 1947, the experimental PCM system using successive approximation algorithm based on subtracting binary weighted charges designated as the "coder" of "feedback subtraction type" was reported in [50]. This system was *de facto* the successive approximation ADC with charge as an intermediate variable and subtraction as the underlying operation used to successive reduction of the quantization error. The first commercial successive approximation analog-to-digital converter was introduced by Bernard M. Gordon in 1954 [51].

The classical design of early successive approximation ADCs was based on digital-to-analog converters with R-2R resistor ladder operating as a string of current dividers. In the middle of 1970s, McCreary and Gray developed the successive approximation ADC based on charge redistribution in the binary-weighted capacitor array [46]. Due to easier fabrication of capacitors, zero quiescent current and lower mismatch and tolerance, the capacitor array has successfully replaced

the R-2R resistor ladder in MOS technology [46]. As a result of this, charge substitutes current as the working medium in implementations of successive approximation converters.

Although the charge redistribution technique was known previously [50, 52], the advantage of scheme introduced by McCreary and Gray [46] consists in a simple architecture and a lack of high performance operational amplifier. This scheme of charge redistribution has been successfully implemented in commercial successive approximation ADCs for decades and is still used nowadays [53].

6.4.1 Architecture of Classical Successive Charge Redistribution ADC

The architecture of the classical successive charge redistribution ADC, which we denote as the C-ADC, consists of a voltage comparator *VC*, an array of binary-weighted capacitors $C_{n-1}, C_{n-2}, \ldots, C_0$ with an extra capacitor C_0 of weight corresponding to the least significant bit (LSB), and a set of switches controlled by the state machine connecting the capacitor plates to certain voltages (see Fig. 6.9 and compare Fig. 6.6). The conversion cycle is performed in three phases: the sample mode, the hold mode, and the redistribution mode in which the actual conversion is performed.

In the *sample mode* (Fig. 6.10), the top plates of capacitors are connected to the ground and the bottom plates respectively to the input voltage source which results in delivery of charge that creates the input voltage V_{IN} on the bottom plate of the capacitors.

In the *hold mode* (Fig. 6.11), the switch disconnects the top plates from and couples the bottom plates to the ground. Since the charge on the top plate is conserved, the potential of the capacitor top plate goes to $-V_{IN}$.

The actual conversion is performed by the *redistribution mode*. In the first step of n-bit conversion cycle, the bottom plate of the largest capacitor C_{n-1} is connected via the switch S_{n-1} to the reference voltage V_{REF}, which corresponds to the full-scale range of the converter (Fig. 6.12). The capacitor C_{n-1} forms the 1:1

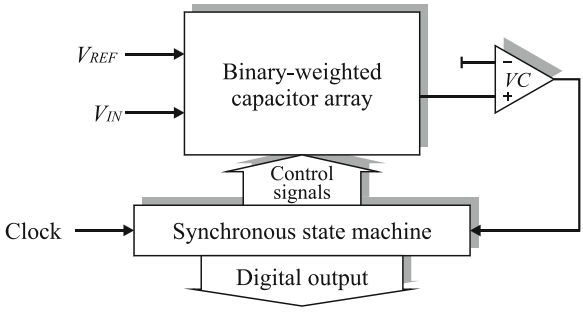

Fig. 6.9 Block diagram of classical successive redistribution analog-to-digital converter (C-ADC)

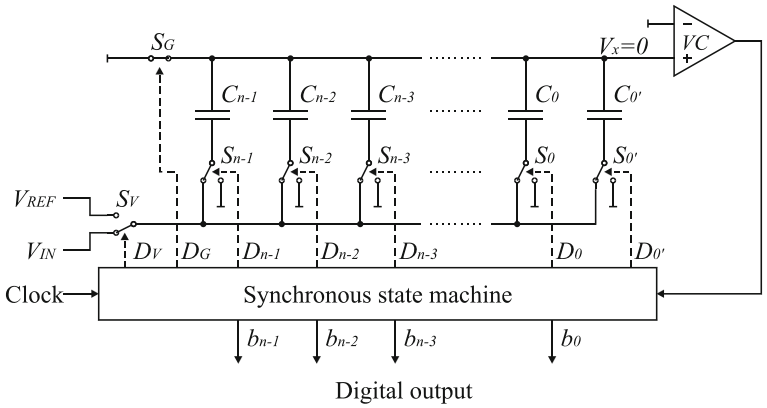

Fig. 6.10 The C-ADC circuit diagram. The controlled switch states correspond to the sample mode

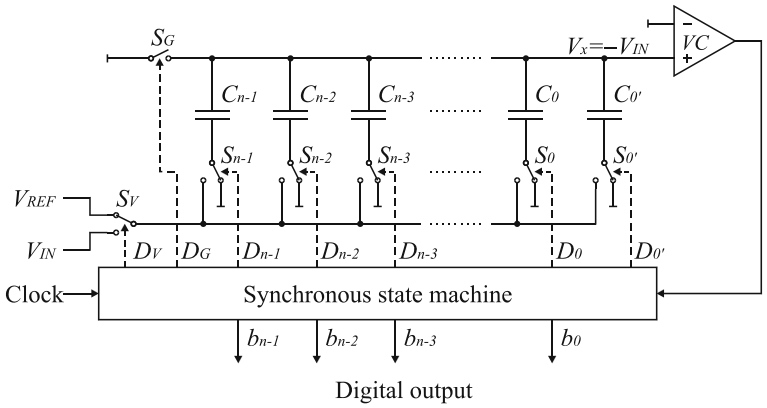

Fig. 6.11 The C-ADC circuit diagram. The controlled switch states correspond to the hold mode

capacitance divider with the remaining capacitors connected to the ground. The comparator input voltage becomes $V_x = -V_{IN} + V_{REF}/2$. If $V_{IN} > V_{REF}/2$, then $V_x < 0$, and the comparator output goes high providing the most significant bit b_{n-1} to 'one'. Furthermore, the bottom plate of the capacitor C_{n-1} is left to be connected to the reference voltage V_{REF}. On the other hand, if $V_{IN} < V_{REF}/2$, then $V_x > 0$, the bit b_{n-1} is evaluated to 'zero' and the bottom plate of the capacitor C_{n-1} is returned to the ground to discharge the capacitor C_{n-1}.

In the next step, the state of the bit b_{n-2} is tested by comparing V_{IN} to $V_{REF}/4$ or to $3V_{REF}/4$ through the different voltage dividers depending on the state of b_{n-1} (Fig. 6.13). It is performed by raising the bottom plate of the next largest capacitor (C_{n-2}) to V_{REF}, and checking the polarity of the resulting value of V_x. In this case,

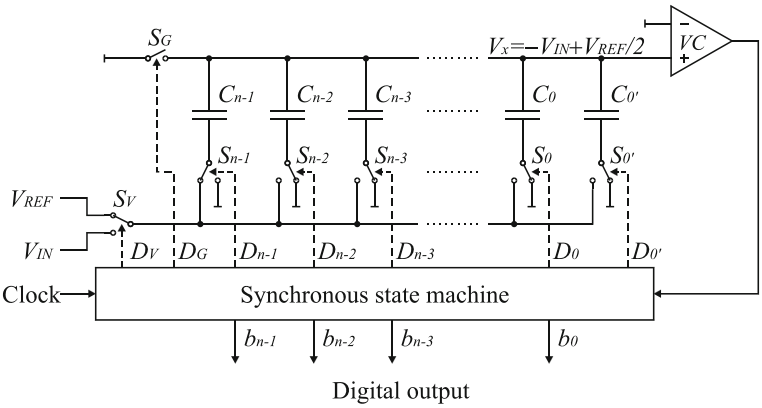

Fig. 6.12 The C-ADC circuit diagram. The controlled switch states correspond to the first step of redistribution mode

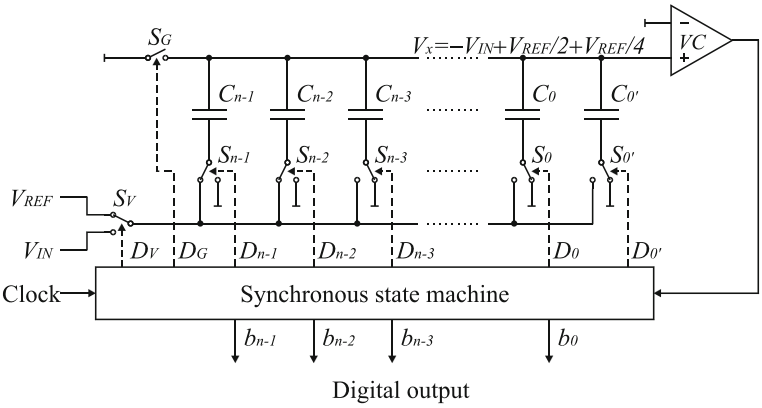

Fig. 6.13 The C-ADC circuit diagram. The controlled switch states correspond to the second step of redistribution mode assuming that the MSB has been evaluated to 'one'

however, the voltage division property of the array causes $V_{REF}/4$ to be added to V_x, thus $V_x = -V_{IN} + b_{n-1}V_{REF}/2 + V_{REF}/4$. If $V_{IN} > b_{n-1}V_{REF}/2 + V_{REF}/4$, then $V_x < 0$ and the bit b_{n-2} is set to 'one'. The bottom plate of the capacitor C_{n-2} is left to be connected to the reference voltage V_{REF}. Similarly, if $V_{IN} < b_{n-1}V_{REF}/2 + V_{REF}/4$, then $V_x > 0$, the bit b_{n-2} is evaluated to 'zero' and the bottom plate of the capacitor C_{n-2} is returned to the ground to discharge the capacitor C_{n-2}.

In forthcoming steps, the conversion proceeds accordingly until all the bits have been determined. After the final conversion step, the comparator input voltage equals: $V_x = -V_{IN} + b_{n-1}V_{REF}/2 + b_{n-2}V_{REF}/4 + \cdots + b_0V_{REF}/2^n$.

6.4.2 Energy Effectiveness of Classical Charge Redistribution Scheme

In an C-ADC, the power is mainly consumed by the capacitor array, the comparator, the reference buffers and by digital circuitry. The energy consumed for charging the capacitors (named also "capacitor switching") is one of the main sources of energy consumption in the converter that determines at the same time the lower bound on energy dissipation of the whole C-ADC [54]. The capacitor switching power consumption is directly proportional to the size of the unit capacitor C_0 in the array [55–57]. In practice, the smallest possible value for unit capacitor is determined by the following constraints: kT/C noise requirement, capacitor matching, design rules and the size of the parasitic capacitances [55]. The comprehensive analysis of the power supplied by the reference voltage source used for capacitor switching and on the linearity of the capacitive array is provided in [58].

The energy efficiency of charge redistribution according to the classical algorithm has been reviewed in [56]. As indicated in [56], the capacitor array is efficiently charged in these conversion steps when the tested bit is evaluated to 'one' and the relevant capacitor is left to be connected to the reference voltage to the end of the conversion cycle. On the other hand, the charge redistribution is highly inefficient during the steps when the tested bit is set to 'zero' and the relevant capacitor is coupled back to the ground throwing away charge that has been stored onto the array.

Furthermore, in [56], the enhancements to the classical algorithm are proposed. These enhancements increase energy efficiency of charge redistribution and are based on *charge recycling*. The idea of charge recovery is to transfer some of the charge from the capacitors falling to the ground to "precharge" the subsequent capacitors rising to the reference voltage, so that the additional energy required to fully charge the rising capacitors is reduced [59]. The energy reduction is directly proportional to the amount of charge being "reused" which depends on the value of sample being processed. The highest energy reduction refers to the cycles when all the bits are evaluated to 'one'. On the other hand, there is no energy savings if all the output bits are set to 'zero'.

For the most efficient charge recycling method proposed in [56] and based additionally on splitting the MSB capacitor into an extra binary scaled subcapacitors, the average reduction of energy needed to drive charge redistribution compared to the classical algorithm is 37 %. The implementation of the converter architecture with two capacitor arrays in the 65-nm CMOS technology is reported in [60]. As follows from comparative analysis, the relevant ADC has one of the best energy efficiencies among published works.

6.4.3 Successive Approximation ADCs Based on Charge Transfer Implemented in CCD Technology

The implementation of the charge redistribution schemes for signals produced by charge coupled devices (CCD) or other sources of charge domain signals

(i.e., infrared detectors) has been disclosed in several patent applications [61, 62] and presented in research papers (e.g., [63, 64]). In the CCD technology, the capacitor array is substituted by binary-weighted potential wells.

6.4.4 Comparison of C-ADC to SCR-ADC

The general similarity between classical (C-ADCs) and event-driven charge redistribution ADCs (SCR-ADCs) presented in Sects. 6.2 and 6.3 consists in that the conversion is realized by deployment of input charge in binary-scaled capacitors. Also the instrumentation used to implement both schemes is in general the same and comprises the capacitor array, comparator(s), reference source and a set of controlled switches. In both converters, the conversion proceeds in the charge domain based on successive reduction of the quantization error. The differences between both schemes consists in distinct methods of charge deployment. In the forthcoming subsections, the SCR-ADCs are compared to classical clock-based converters C-ADCs introduced by [46] and commonly used in analog-to-digital conversion for decades [53].

6.4.4.1 Time-Triggered versus Event-Triggered Architecture for ADCs

As stated before, the principal difference between the SCR-ADC and the C-ADC is that the charge redistribution is self-timed in the former whereas it is clock-driven in the latter. An important implication is that the SCR-ADC almost does not consume energy between conversion cycles that can be triggered on irregular demands. The SCR-ADC is one of propositions of introducing the event-driven architecture to analog-to-digital conversion which is still dominated by traditional time-triggered implementations. In general, the event-based ADC architectures are attractive for applications with constrained energy resources (e.g., in wireless sensor networks). The event-based ADCs are adopted to event-based sampling strategies [12, 19] but they are able to operate particularly with periodic sampling.

6.4.4.2 Reference Voltage Stability and Electromagnetic Interference

In the classical C-ADCs, the current used for switching the charge within the capacitor array is delivered from the reference voltage source V_{REF} (see the position of the switch S_V in Figs. 6.12 and 6.13 during charge redistribution). Therefore, the current intensity and its variability affect the reference voltage stability, and consequently, conversion accuracy. It is a significant point because rapid changes of the current that result in discontinuities of voltages on capacitors are enforced while the charge is switched in the capacitor array in each step. Fast

changes of the current absorbed from the reference voltage source cause not only fluctuations of the reference voltage but also transient electromagnetic interference.

Instead, in the event-driven SCR-ADCs, the reference voltage V_L connected to high impedance comparator input provides comfortable working conditions in terms of voltage stability. Moreover, the current formed by flow of charge transferred between the source and destination capacitors is stable and obtained from the voltage supply instead of the reference source (see the direction of the flow of charge in Fig. 6.16). The voltages on capacitors vary slowly except the relaxation phase. Due to slowly varying voltages, ensuring electromagnetic compatibility is easier in the event-driven SCR-ADCs than in classical C-ADCs because charge switching is no longer a source of electromagnetic interference. In the SCR-ADCs, the transient changes of voltage are driven only by comparators and control logic. The charge redistribution free of electromagnetic interference enables the use of the SCR-ADCs in solutions with low amplitude signals which refers particularly to biomedical applications.

6.4.4.3 Capacitor Leakage and Conversion Accuracy

In the classical SCR algorithm, the whole input charge is transferred among *all* the capacitors during *each* step of charge redistribution (see Figs. 6.11, 6.12, 6.13). Thus, the conversion accuracy is affected by leakage of all the capacitors. Instead, in the event-driven SCR-ADCs, the charge is exchanged in each step only between *two* capacitors (see Fig. 6.16). Therefore, the leakage of the capacitors corresponding to bits whose states have been already evaluated in previous steps does not influence conversion accuracy and the charge portion being processed in general decreases as the cycle proceeds.

6.4.4.4 Static versus Dynamic Conditions of Comparator Operation

There are different requirements for time response of the comparators in both converters. In the C-ADC, the comparator performs just the signum function in static conditions and the influence of a comparator delay can be eliminated by adequate selection of the clock frequency. Instead, in the SCR-ADC, both comparators perform voltage level-crossing detection in real time (see Fig. 6.19) and the comparator delay is one of primary factors that affects the conversion accuracy. Therefore, the speed of charge redistribution defined by the current source intensity I must be balanced and limited by comparator latency to keep the conversion accuracy higher than converter resolution. A precise detection of the level-crossing instant instead of evaluation of signum function in static conditions is a fundamental difference of the comparator function in the SCR-ADC compared to the C-ADC.

Due to a need of timely detection of level-crossings, the conversion time of the SCR-ADC is longer than for the C-ADC. As it will be shown further, the SCR-ADC conversion time can be reduced by starting the charge redistribution concurrently with its accumulation (see Sect. 6.7) which is impossible in the C-ADC where the sample mode always precedes the redistribution mode and cannot be started simultaneously.

6.5 Circuit Reconfiguration During Conversion Cycle

As stated, the SCR-ADC circuit operation consists in controlling the charge accumulation and its event-driven successive redistribution by a dynamic reconfiguration of states of controlled switches. The switches are reconfigured as a response to events occurring in the conversion process and reported by both comparators or detected on the input (see Sect. 6.3.2). Now we will discuss in details the process of reconfiguration of circuit topology managed by the asynchronous state machine during the conversion cycle.

During the charge accumulation, the charge input is coupled to the input capacitor C_n via a closure of the switch S_S by the control signal C_S. Next, the current source I is activated by the control signal C_I during the redistribution phase. The main task of the ASM is a selection of the source *Source_C*(i) and destination *Dest_C*(i) capacitors, and to establish the path between both capacitors to enable a charge transfer driven by the current source I. As follows from the analysis in Sect. 6.3.4, a selection of the *Source_C*(i) is a dynamic process whose course depends on the states of output bits already evaluated, and a selection of the *Dest_C*(i) is a static process repeated in the same order in every conversion cycle.

In the SCR-ADC configuration presented in Fig. 6.7, the input of the current source I is connected to the rail H, and its output to the rail L respectively. Thus, to enable the charge transfer, the *Source_C*(i) is connected between the rail H and the potential V_H, and the *Dest_C*(i) between the rail L and the ground. To simplify the switch control during the conversion process, the bottom plates of all the other capacitors, except of the *Dest_C*(i), are also kept on the potential V_H but their top plates, except the top plate of the *Source_C*(i), are disconnected both from rails L and H (compare the hydraulic model in Figs. 6.1, 6.2, 6.3, 6.4, 6.5). In this way, the current source I is put into the path established only between the *Source_C*(i) and the *Dest_C*(i).

Moving the charge results in growing the potential of the top plate of the *Dest_C*(i), and falling the potential of the top plate of the *Source_C*(i). If the former potential reaches $V_L < V_H$ before the latter falls to V_H, the comparator $K2$ signals to the ASM that the *Dest_C*(i) is charged to a desired voltage V_L. Then the ASM sets the output bit b_{n-i} corresponding to the *Dest_C*(i) to 'one', and changes the destination capacitor according to the condition (6.4) by appropriate switch reconfiguration [the source capacitor is kept the same according to the condition (6.5)]. On the other hand, if the top plate of the *Source_C*(i) falls to the V_H before

the voltage on the *Dest_C*(*i*) reaches a desired level V_L, the comparator *K*1 reports to the ASM that the *Source_C*(*i*) is discharged. Then, the ASM sets the appropriate output bit to 'zero', and reconfigures the states of switches to change both the source and the destination capacitors according to the conditions (6.5). The other tasks managed by the ASM are the controls of charge accumulation and of relaxation phases.

The proposed converter architecture requires to select the voltage V_H such that $V_{DD} - V_H \geq V_L$ to keep the bottom plates of the *Source_C*(*i*) (and the other capacitors except the *Dest_C*(*i*)) on the potential V_H during the *i*th conversion step. On the other hand, the condition $V_H > V_L$ is needed to enable charge transfer between *Source_C*(*i*) and *Dest_C*(*i*). Summing up, it is desirable to have the V_L possibly high in order to minimize the impact of comparator offsets on conversion accuracy but the voltage difference $U_I = V_H - V_L$ must be kept large enough to guarantee proper operation of the current source *I* used for charge transfer. Finally, for a given minimum voltage value U_I on the current source, the voltage V_L is limited as follows: $V_L \leq (V_{DD} - U_I)/2$. Thus, V_L is always lower than $U_I/2$.

6.5.1 Switch Control

Three controlled switches S_{Hi}, S_{Li}, S_{Gi} are assigned to the *i*th cell in the capacitor array in the proposed configuration of the SCR-ADC seen in Fig. 6.7. The group of the switches S_{L0},…, S_{Ln} are used in a selection of *Dest_C*(*i*) during the conversion process on the one hand, and to discharge all the capacitors simultaneously during the relaxation phase on the other. The group of the switches S_{G0}, …, S_{Gn} are used in order to hold the bottom plates of the capacitors C_0,…, C_n on the ground potential, or on the potential V_H. The former is needed in a selection of the *Dest_C*(*i*), and the latter for choosing the *Source_C*(*i*). The group of the controlled switches S_{H0},…, S_{Hn} are used in a selection of the *Source_C*(*i*) to couple it to the current source *I* input. The switch S_{Gall} is used to connect the rail *L* to the ground to discharge all the capacitors.

The switches are controlled by the signals I_i and D_i produced by the state machine. Both signals are active with a high logical level. The active state of the signal D_i closes the switch S_{Hi}. To simplify switch control, a pair of switches S_{Gi}, S_{Li}, assigned to a given capacitor C_i, is controlled by the same signal I_i, $i = 0,1,\ldots,n$. Thus, the active state of I_i closes the switch S_{Li} and also connects the switch S_{Gi} to the ground. Therefore, if the bottom plate of the capacitor C_i is grounded, its top plate is at the same time coupled to the rail *L*.

The switch S_{Hi} is controlled by the signal D_i. The control signals I_i and D_i are not both active during any step of charge redistribution. However, they are active simultaneously during charge accumulation (for $i = n$) as presented in Fig. 6.7. Thus, the active state of the control signal I_{n-i} selects respectively the *Dest_C*(*i*), and the active state of D_j indicates the *Source_C*(*i*). The cells that are not currently

involved in the charge redistribution are driven with inactive states of both control signals I_i and D_i. In Figs. 6.9, 6.10, 6.11, 6.12, the evolution of circuit configuration during the conversion cycle will be presented.

6.5.2 Switch Reconfiguration During Relaxation Phase

As stated, in the relaxation phase, the input capacitor and all the working capacitors are quickly discharged in order to make the SCR-ADC ready for the next conversion cycle. The states of the switches during the relaxation phase are shown in Fig. 6.14. All the signals I_i, $i = 0, 1,\ldots,n$ are active, thus the switches from the group S_{Gi} connect the bottom plates of capacitors to the ground, and the top plates to the rail L. Since the rail L is also grounded via the switch S_{Gall} that is then closed, the top plates of all the capacitors are also kept on the ground potential. The signals C_I and C_S are inactive to disconnect the current source I and also the charge input from the rest of converter circuitry (by opening the switch S_S).

By closing the switch S_{Hn} with an active state of the control signal D_n, the rail H is connected also to the ground in order to avoid the undesirable transitions of the comparator $K1$ output state due to possible appearance of electrostatic charge on the high resistance $K1$ input. Otherwise, the random transitions of the comparator $K1$ output may unnecessarily increase power consumption. The comparator $K1$ output is thus kept active but it is of no importance because the state machine ASM ignores both comparators output states in the relaxation phase.

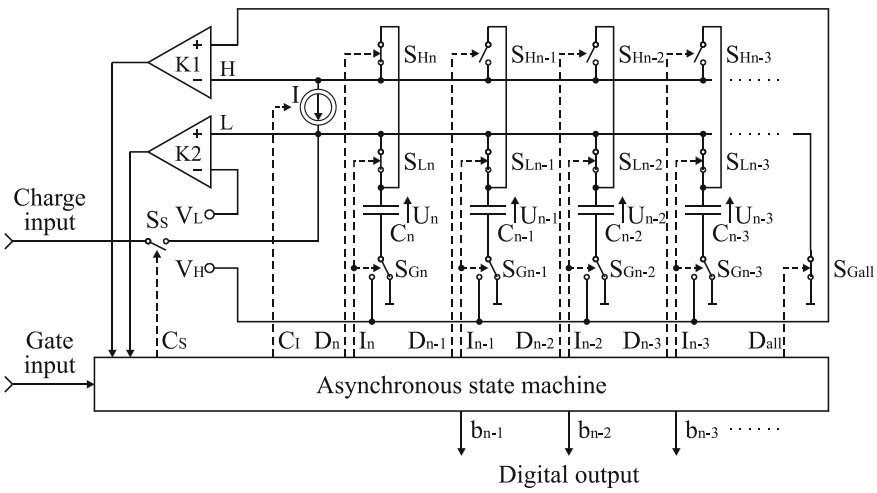

Fig. 6.14 The SCR-ADC circuit configuration during the relaxation phase

6.5.3 Switch Reconfiguration During Charge Accumulation

During the charge accumulation, the incoming charge packet is delivered to the input capacitor C_n through the charge input and the switch S_S activated by the ASM using the control signal C_S. The instants when the charge packet accumulation is started and terminated are defined by the external gate pulse provided to the gate input. As soon as the state machine detects the leading edge of the gate pulse on the gate input, it reconfigures the controlled switches to the states shown in Fig. 6.7 in Sect. 6.2.2. Since the input capacitor C_n operates as the destination capacitor, thus the control signals I_i except I_n are deactivated which causes at the same time moving bottom plates of all working capacitors $C_{n-1}\ldots,C_0$ to the potential V_H, and keeping the input capacitor C_n on ground. During the charge accumulation, the potential of the top plate of the input capacitor C_n coupled to the rail L (the switch S_{Ln} is closed) is non-decreasing in time and proportional to the charge packet portion already accumulated. The switch S_{Gall} is open.

The control signal D_n is active so the switch S_{Hn} is closed in order to keep the rail H on the well-defined potential similarly as during the relaxation phase. The situation when both signals D_n and I_n are active simultaneously occurs only during the charge accumulation process. The current source is disconnected from the rest of the circuit since the control signal C_I is inactive. The output states of both comparators are then ignored by the state machine.

6.5.4 Switch Reconfiguration During Conversion Phase

The trailing edge of the gate pulse on the gate input terminates the accumulation phase, and starts the conversion phase.

The subsequent steps of the conversion phase are self-timed by the outputs of the comparators $K1$, $K2$. The trailing edge of the gate pulse causes opening the switch S_S by deactivating the control signal C_S, terminating the charge accumulation in the input capacitor C_n, and connecting the current source I to the capacitor array in order to redistribute the charge stored in the input capacitor. The current source I is active during the whole conversion phase. The state machine selects a capacitor that serves as the $Source\_C(i)$ by activating the control signal D_j according to (6.5), and respectively a capacitor that serve as $Dest\_C(i)$ by activating the control signal D_{n-i} according to (6.4).

In Fig. 6.15, the states of the switches during the first conversion step ($Source\_C(1) = C_n, Dest\_C(1) = C_{n-1}$) are shown. The simplified version of circuit configuration presented in Fig. 6.15 is depicted in Fig. 6.16 (compare Fig. 6.16 to Fig. 6.2).

The input capacitor C_n as $Source\_C(1)$ is connected to the potential V_H by deactivating the control signal I_n. The signal D_n is kept active so the C_n is still coupled to the rail H as during the relaxation and charge accumulation when the

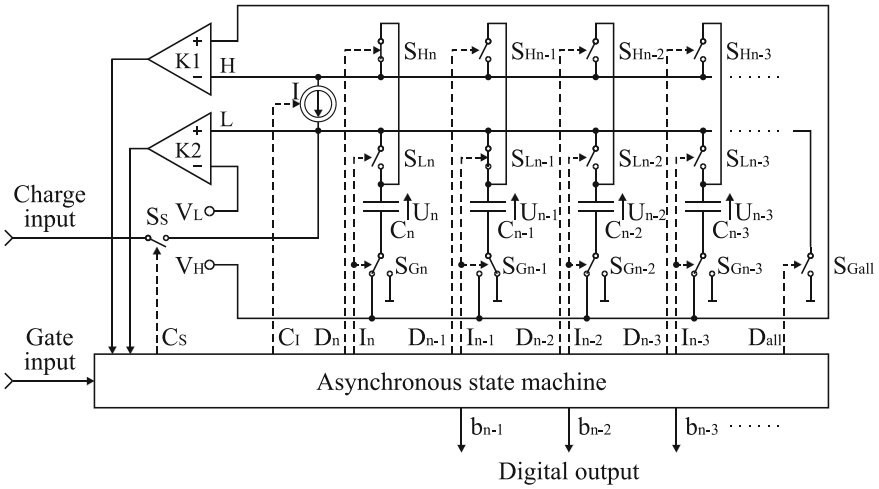

Fig. 6.15 The SCR-ADC circuit diagram. The switch states correspond to the first conversion step ($Source\_C(1) = C_n$, $Dest\_C(1) = C_{n-1}$)

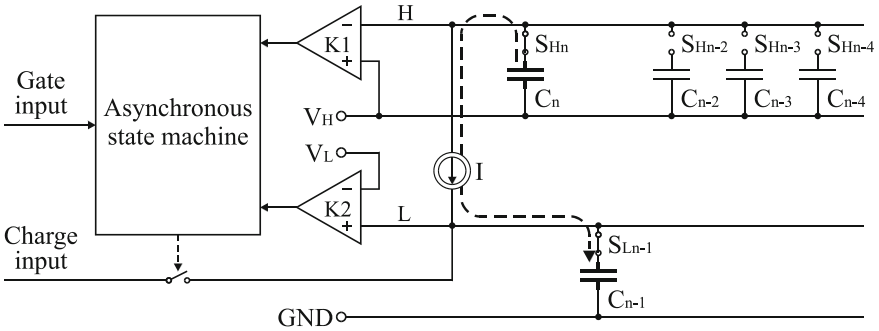

Fig. 6.16 Simplified version of SCR-ADC configuration shown in Fig. 6.15 referring to the first conversion step (compare to Fig. 6.2)

switch S_{Hn} has been selected to hold the rail H in a well-defined potential. The I_{n-1} is activated to enable operation C_{n-1} as $Dest\_C(1)$.

Since the selection of the $Dest\_C(i)$ is a static process, the hardware implementation of the selector of the current destination capacitor can be in the ASM realized by combinational logic, for example, by a ring counter cleared by the trailing edge of the pulse on ADC input, and driven by the output of the comparators $K1$ and $K2$. As the selection of the $Source\_C(i)$ is dynamic and depends on the state of bits in the digital output word evaluated in the previous conversion steps, the hardware implementation of the selector of the current source capacitor in the ASM must be based on sequential logic. Two cases have to be considered according to the conditions (6.5).

6.5.4.1 Evaluating Bit State to 'One'

If the voltage on the current destination capacitor $Dest\_C(i)$ reaches V_L resulting in the evaluation of the bit b_i to 'one', then the source capacitor will keep its role in the next conversion step according to (6.5). Thus, if $b_i = 1$, then the signal D_j is kept active in the $i + 1$ conversion step. The role of the next destination capacitor $Dest\_C(i + 1)$ is then delegated to the C_{n-i-1}, which causes the activation of the control signal I_{n-i-1} as shown in Fig. 6.17.

Figure 6.17 illustrates the reconfiguration of the switch states in relation to Fig. 6.15 soon after the bit b_{n-1} has been evaluated to 'one'. Thus, the switch states in Fig. 6.17 correspond to the second conversion step when $j = n$, $i = 2$, i.e., $Source\_C(2) = C_n, Dest\_C(2) = C_{n-2}$.

6.5.4.2 Evaluating Bit State to 'Zero'

If the voltage on the current source capacitor $Source\_C(i)$ reaches zero before the voltage on the current destination capacitor $Dest\_C(i)$ reaches V_L, then the $Dest\_C(i)$ will operate as the $Source\_C(i + 1)$ in the next conversion step according to (6.5).

The state machine changes the destination and the source capacitor in the next step by:

- deactivating the control signals D_j to stop the operation of C_j as the source capacitor,

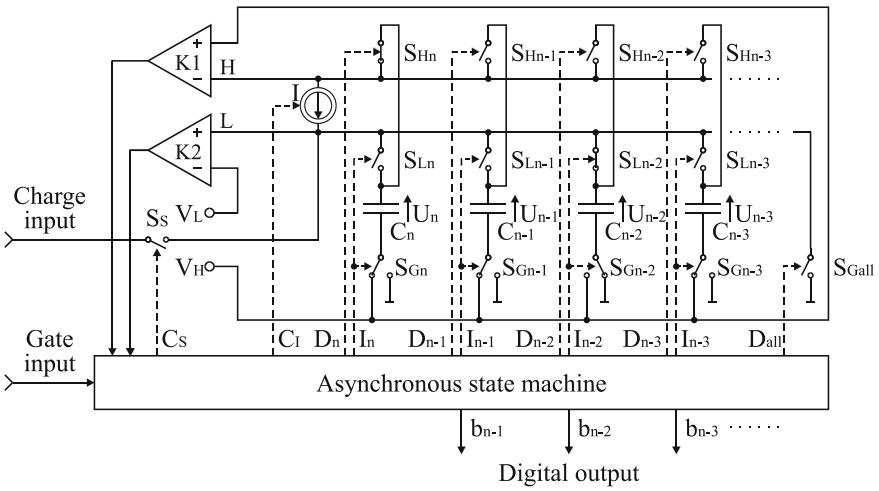

Fig. 6.17 The SCR-ADC circuit configuration during the second conversion step when $Source\_C(2) = C_n$, $Dest\_C(2) = C_{n-2}$ soon after the bit b_{n-1} has been evaluated to 'one'

- deactivating the control signals I_{n-i} and activating the control signal D_{n-i} to change the role of the previous destination capacitor to the current source capacitor,
- activating the control signal I_{n-i-1} to select the next destination capacitor.

Figure 6.18 presents the circuit configuration soon after the bit b_{n-2} has been evaluated to 'zero'. The switch states correspond to the third conversion step when $j = n-2$, $i = 3$, i.e., $Source\_C(3) = C_{n-2}, Dest\_C(3) = C_{n-3}$. In the relevant column of Table 6.1 in Sect. 6.3.4, the control signals that are active during the relaxation, charge accumulation, and also during particular conversion steps are listed.

The complete timing of signals during the relaxation phase, charge accumulation and during the conversion steps for the course of 4-bit conversion cycle in SCR-ADC is presented respectively in Fig. 6.19a. On the other hand, Fig. 6.19b shows the similar diagram for the DSCR-ADC (*Direct Successive Charge Redistribution* ADC) which is an enhanced version of the SCR-ADC that will be discussed in Sect. 6.6. The states of output bits (1001) correspond to the course of conversion cycle illustrated on the basis of the hydraulic model in Sect. 6.2.1 for the SCR-ADC. The timing includes the control signals C_I, C_S, I_4, I_3, I_2, I_1, D_4, D_3, D_2, D_1 produced by the asynchronous state machine for corresponding switches, the voltages $V(H)$ on the rail H, and $V(L)$ on the rail L, and the voltages on the capacitors U_4, U_3, U_2, U_1, and on the comparator $K1$, $K2$ outputs.

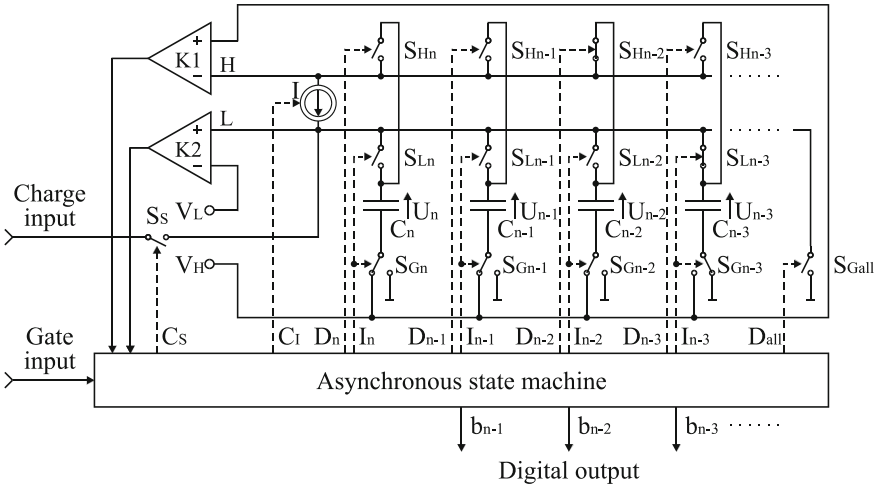

Fig. 6.18 The SCR-ADC circuit diagram. The switch states correspond to the third conversion step when $Source\_C(3) = C_{n-2}$, $Dest\_C(3) = C_{n-3}$ soon after the bit b_{n-2} has been evaluated to 'zero'

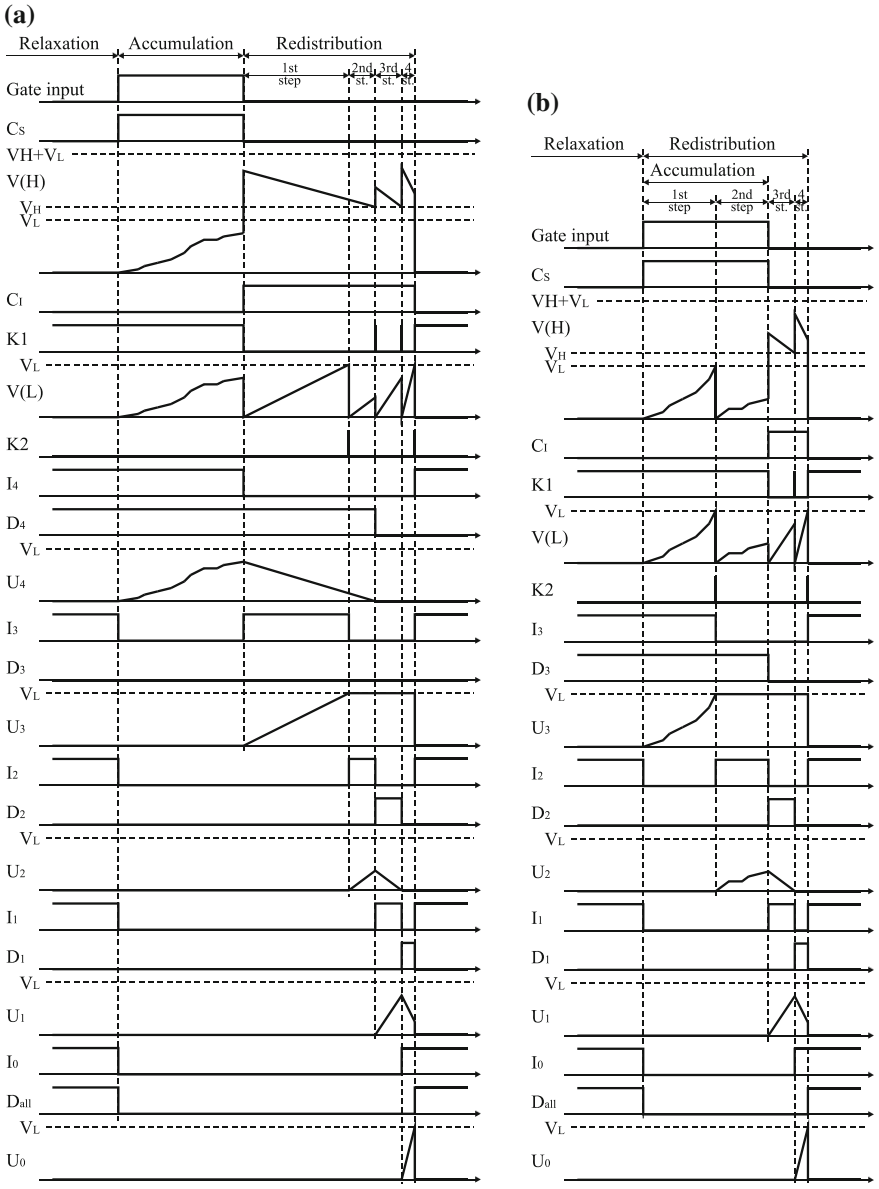

Fig. 6.19 Timing of signals during the relaxation, charge accumulation and during all the conversion steps for the course of 4-bit conversion cycle in SCR-ADC (on the *left*) (**a**) and DSCR-ADC (on the *right*) (**b**). The states of output bits (1001) correspond to the course of conversion cycle illustrated by the use of the hydraulic model in Sect. 6.2.1

6.5.5 Conversion Accuracy versus Implementation Non-idealities

The accuracy of the SCR-ADC is mainly determined by the accuracy of implementation of binary-scaled capacitances. The sum of the capacitances $C_{n-1} + \cdots + C_0$ defines the conversion range. On the other hand, the mismatch of the capacitance ratio determines the integral and differential nonlinearity of converter characteristics.

The worst-case integral (*INL*) and differential (*DNL*) nonlinearity for a number of n binary-scaled capacitors implemented with a tolerance $\Delta C/C$ are estimated as follows [27]:

$$INL = \pm 2^{n-1} \frac{\Delta C}{C} LSBs \tag{6.9}$$

$$DNL = \pm (2^n - 1) \frac{\Delta C}{C} LSBs \tag{6.10}$$

The relative tolerance of capacitance ratio in MOS technology can be as low as ± 0.1 % [27]. For the 8-bit SCR-ADCs, the worst-case integral and differential nonlinearity amounts to $INL \cong \pm 0.13 LSBs$ and $DNL \cong \pm 0.26 LSBs$ assuming that $\Delta C/C = \pm 0.1$ %.

The non-zero resistance of closed switches reduces intended voltages on the capacitors. As a result, the voltage on destination capacitors is less than V_L by the same increment if the resistance of each switch is fairly the same. This creates the gain error. On the other hand, the reduction of the voltage on the source capacitors creates dynamic errors. However, if the resistances of particular switches vary, the extra differential nonlinearity is introduced to converter characteristics. The voltages on the closed switches are proportional to the intensity of current source I used for charge redistribution.

The comparator delays cause a delivery of an extra charge portion to each destination capacitor. This charge value defined by the product of comparator delay and the intensity of current source I is independent of the destination capacitor capacitance. As a result, the comparator delays evoke additional integral nonlinearity in relation to the *INL* induced by capacitor mismatching. To keep the conversion errors created by non-zero resistance of switches and comparator delays bounded, the intensity of current source I must be limited.

6.6 Analysis of Conversion Time in SCR-ADCs

It is well-known that the analog-to-digital converters based on the successive approximation scheme are characterized by high resolution and accuracy obtained at the cost of relatively long conversion time. In the classical n-bit synchronous

successive approximation ADCs, the conversion consists of a number of n equal steps while each step takes one clock period. In the ADC with the event-driven successive charge redistribution, the conversion time is in general longer and the durations of particular steps are unequal. Usually the steps in the SCR-ADC became shorter as the conversion cycle proceeds (see Fig. 6.14). More precisely, the duration of steps decreases or remains constant with an increase of the step number. On the other hand, the durations of steps depend on the states of bits being tested. As will be demonstrated, a given conversion step terminated with evaluating the relevant bit to 'one' is longer than the same step completed with setting the bit to 'zero'. Therefore, the SCR-ADC conversion time is a non-linear function of the input charge value [26]. The evaluation of the conversion time is a key to estimate the energy consumption that is directly proportional to the conversion time.

6.6.1 Duration of Conversion Steps

The duration of a particular conversion step equals the time needed to transfer charge from the source to the destination capacitor within this step. Since the rate of charge redistribution is constant and equal to the intensity of the current source I enforcing charge flow, the duration T_k of the kth conversion step depends on the amount of charge Q_k being then moved:

$$T_k = \frac{Q_k}{I} \tag{6.11}$$

If the output bit b_{n-k} is evaluated to 'one', the charge value translocated during the kth conversion step from the source to the destination capacitor $Dest\_C(k) = C_{n-k}$ equals $C_{n-k}V_L = Q/2^k$. On the other hand, if b_{n-k} is evaluated to 'zero', the amount of charge transferred during the kth conversion step equals $Q_q(k-1)$ which is the $(k-1)$ bit quantization error (see Sect. 6.3.1). Therefore:

$$Q_k = \begin{cases} \frac{Q}{2^k}, & \text{if } b_{n-k} = 1 \\ Q_q(k-1), & \text{if } b_{n-k} = 0 \end{cases} \tag{6.12}$$

As follows from the discussion in Sect. 6.3.1:

$$Q_q(k-1) < Q_k/2^k \quad \text{if } b_{n-k} = 0 \tag{6.13}$$

therefore the duration T_k of the conversion step completed with the evaluation the bit b_{n-k} to 'one' is longer than in case if it is set to 'zero'.

6.6.1.1 Duration of Conversion Step with Evaluating Bit State to '1'

According to (6.11) and (6.12), the duration of the kth conversion step T_k provided that $b_{n-k} = 1$ equals:

$$T_k = T_{k\max} = \frac{Q}{2^k I} = \frac{T}{2^k} \quad \text{if}, b_{n-k} = 1 \tag{6.14}$$

where $Q = 2^n V_L C_0$ is the SCR-ADC input range (full scale) and $T = Q/I$.

As follows from (6.14), the maximum durations of the successive steps $T_{k\max}$ halves with a number of the conversion step k since the capacitances of subsequent destination capacitors are reduced twice, and the rate of charge redistribution defined by I is the same (see Fig. 6.19).

6.6.1.2 Duration of Conversion Step with Evaluating Bit State to '0'

The $(k-1)$ bit quantization error $Q(k-1)$ can be found as the difference of the input charge Q_{In} and the charge portions $Q/2, \ldots, Q/2^{k-1}$ disposed in the destination capacitors $C_{n-1}, \ldots, C_{n-k+1}$ in the previous $1, \ldots, k-1$conversion steps as follows:

$$Q(k-1) = Q_{In} - \sum\nolimits_{i=0}^{k-1} \frac{b_{n-i} Q}{2^i} \tag{6.15}$$

The duration T_k of the kth conversion step if $b_{n-k} = 0$ according to (6.11) is:

$$T_k = \frac{Q(k-1)}{I} = \frac{Q_{In}}{I} - \sum\nolimits_{i=0}^{k-1} b_{n-i} T_i < T_{k\max} \quad \text{if } b_{n-k} = 1 \tag{6.16}$$

where $\sum_{i=0}^{k-1} b_{n-i} T_i$ is the sum of durations of previous conversion steps from 0 to $k-1$ that have been terminated with evaluating the output bits $b_{n-1}, \ldots, b_{n-k+1}$ to 'one'. The initializing conditions $T_0 = 0$ and $b_n = 0$ have not any physical interpretation and are introduced to the formula (6.16) arbitrary in order to model the duration of the first conversion step ($k = 1$) accordingly.

6.6.2 Duration of Conversion Step in General Case

Summing up, the duration T_k of the kth SCR-ADC conversion step is modelled by the following formula:

$$T_k = \begin{cases} \frac{T}{2^k}; & \text{if} : Q_{In} \geq \frac{Q}{2^k} + \sum\limits_{i=0}^{k-1} b_{n-i} \frac{Q}{2^i} \\ \frac{Q_{In}}{I} - \sum\limits_{i=0}^{k-1} b_{n-i} T_i, & \text{if} : Q_{In} < \frac{Q}{2^k} + \sum\limits_{i=0}^{k-1} b_{n-i} \frac{Q}{2^i} \end{cases} \tag{6.17}$$

where $T_0 = 0$, $b_n = 0$ by the convention as stated before.

In particular, the duration of the first conversion step T_1 is:

- $T/2$ if $b_{n-1} = 1$ for $Q_{In} \geq Q/2$; or
- Q_{In}/Iif $b_{n-1} = 0$ for $Q_{In} < Q/2$.

The second conversion T_2 step lasts:

- $T/4$if $b_{n-2} = 1$ for $Q_{In} \geq Q/4 + b_{n1-}Q/2$; or
- $Q_{In}/I - T/2$ if $b_{n-2} = 0$ for $Q_{In} < Q/4 + b_{n-1}Q/2$.

Durations of subsequent conversion cycles can be estimated accordingly.

6.6.3 Conversion Time in SCR-ADC

By taking into account the formula (6.17), the SCR-ADC conversion time T_{C_SCR} versus the input charge Q_{In} is defined as the sum of the durations of the steps 1,…,n:

$$T_{C\_SCR}(Q_{In}) = \sum_{k=1}^{n} \left[\frac{b_{n-k}T}{2^k} + (1 - b_{n-k}) \cdot \left(\frac{Q_{In}}{I} - \sum_{i=1}^{k} \frac{b_{n-i}T}{2^i} \right) \right] \tag{6.18}$$

The plot of the normalized conversion time T_{C_SCR}/T versus the normalized input charge Q_{In}/Q for the 12-bit SCR-ADC according to (6.18) is shown in Fig. 6.20. As follows from (6.18) and Fig. 6.20, the relationship between T_{C_SCR}/T and Q_{In}/Q is highly non-linear. The maximum conversion time $T_{C\_SCR\max}$ appears when all the bits in the output digital word are evaluated to 'one' because the durations of all the conversion steps reach its maximum value defined by (6.14):

$$T_{C\_SCR\,\max} = \sum_{k=1}^{n} T_{k\,\max} = \sum_{k=1}^{n} \frac{T}{2^k} = T\frac{2^n - 1}{2^n} \cong T \tag{6.19}$$

This corresponds to a situation when the Q_{In} approaches the SCR-ADC input range Q. As follows from (6.19), the maximum conversion time $T_{C\_SCR\max}$ is

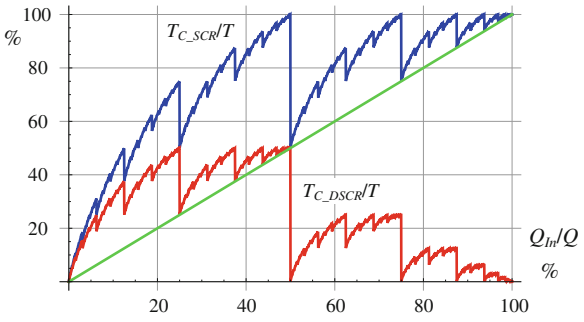

Fig. 6.20 The normalized conversion times T_{C_SCR}/T (*blue line*) and T_{C_DSCR}/T (*red line*) versus normalized input charge Q_{In}/Q according to (6.18) and (6.27) for 12-bit resolution. The *green line* represents Q_{In}/QI

almost independent of the SCR-ADC resolution (n) and with an increase of a number of bits n approaches asymptotically $T = Q/I$.

On the other hand, the conversion time T_{C_SCR} never falls below Q_{In}/QI which refers to the green line in Fig. 6.20. The ratio Q_{In}/I defines the minimum for T_{C_SCR} which is obtained when the charge redistribution process is as effective as possible. It occurs if the binary search algorithm finds directly the appropriate destination capacitors without transferring the charge to the intermediate (source) capacitors. Such situation happens if the role of the source capacitor is kept by the input capacitor through the whole conversion cycle. Then, the output word looks like a thermometer code, that is, it consists of 'one's at a certain number of the most significant bits, and 'zero's at the least significant bits (i.e., 11…100…0). For such output words, the conversion time T_{C_SCR} approaches Q_{In}/I neglecting an extra delay due to transfer of the charge portion less than charge unit $Q_n = V_L C_0$. Thus, the most effective charge redistribution happens when Q_{In} equals $Q/2$, $3Q/4$, $7Q/8, \ldots, (2^n - 1)Q/2^n$, respectively. Finally, the conversion time T_{C_SCR} has the following lower and upper bounds:

$$Q_{In}/I \leq T_{C\_SCR}(Q_{In}) \leq T. \tag{6.20}$$

6.6.4 Charge Redistribution Effectiveness

Taking into account a non-linear relationship of the T_{C_SCR} versus Q_{In}, the interesting issue is to answers the question how many times a charge unit is moved on average during the redistribution process. The measure Q_{In}/I represents the time needed to move the whole input charge from the input capacitor directly to destination capacitors without intermediate transfers. The efficiency of the charge redistribution process may be illustrated by examining the graphical relation between the plots of T_{C_SCR} (red curve) and Q_{In}/I (green line) in Fig. 6.20. If the plot of T_{C_SCR} is close to Q_{In}/I, then the conversion effectiveness is almost optimal. However, if T_{C_SCR} is much higher than Q_{In}/I, the efficiency of the charge redistribution process is lower. As seen in Fig. 6.20, the efficiency of the SCR-ADC conversion process is determined by the value of the input charge Q_{In}. The mean number of transfers of each charge unit included in the input charge portion during a redistribution process can be found by evaluating the ratio of the conversion time T_{C_SCR} to Q_{In}/I:

$$p_{C\_SCR} = \frac{IT_{C\_SCR}}{Q_{In}} \tag{6.21}$$

The plot of p_{C_SCR} versus input charge Q_{In}/Q for $n = 12$ is presented in Fig. 6.21. As illustrated, the mean number of transfers of each charge unit in a conversion cycle is both lower and upper bounded. The p_{C_SCR} reaches its maximum equal to n if the effectiveness of charge redistribution is low (i.e., the small

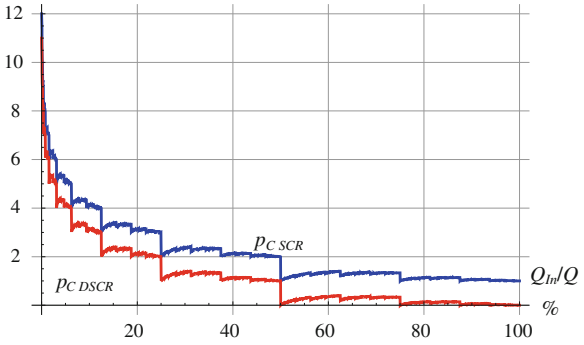

Fig. 6.21 Mean numbers of transfers of a charge unit per conversion cycle versus normalized input charge Q_{In}/Q for the SCR-ADC (p_{C_SCR}) and the DSCR-ADC (p_{C_DSCR}) if $n = 12$

charge portion less than charge unit Q_n is successively transferred through all the n capacitors $C_{n-1},\ldots,C_0$, and the role of source capacitor is changed in each conversion step). This happens when at the list the bits $b_{n-1},\ldots,b_1$ in the digital output word are evaluated to 'zero'.

On the other hand, the p_{C_SCR} reaches its minimum equal to one if the effectiveness of charge redistribution is high (i.e., the large charge portion is transferred directly to the destination capacitors $C_{n-1},\ldots,C_0$ while the input capacitor C_n acts as the source capacitor during the whole conversion cycle). This corresponds to the situation when the output word is of a thermometer code type (11…100…0).

Thus:

$$1 \leq p_{C\_SCR} \leq n \tag{6.22}$$

Summing up, the SCR-ADC conversion effectiveness is lower for small values of the input charge Q_{In} and higher for large Q_{In} since a small value of input charge is transferred even n times until it finds the final destination capacitor and large values of input charge may reach its destination without intermediate transfers (Fig. 6.21).

6.7 Enhancements of Basic Version of SCR-ADC

In the event-based SCR-ADC described above, the input charge accumulation phase and the charge-to-digital conversion are realized *sequentially* because the input charge Q_{In} is collected in the extra capacitor before starting the redistribution phase. The significant enhancement that can be introduced to the basic converter version is to start redistribution *concurrently* to input charge accumulation. The relevant converter version where Q_{In} is processed at the same time when it is collected in the capacitor array is termed the *Direct Successive Charge Redistribution ADC* (DSCR-ADC).

6.7.1 Principles of Direct Successive Charge Redistribution

Referring to the hydraulic model, the concept of the DSCR-ADC is based on a simple observation that liquid can be accumulated immediately in the auxiliary containers without the use of the input container (Fig. 6.1). In the electric analogy, the input charge is collected directly in the working capacitors in the order of decreasing capacitances and the input capacitor is removed from the converter architecture. In general, the redistribution algorithm is the same in the DSCR-ADC as in the SCR-ADC.

The primary benefit obtained in the DSCR-ADC is a reduction of the conversion time compared to the SCR-ADC. The time needed to split the input charge among the working capacitors is the same in the DSCR-ADC as in the basic version of the SCR-ADC. But the difference is that the charge deployment is started immediately at the beginning of its accumulation instead of at the end as it occurs in the SCR-ADC.

Usually the duration of the charge redistribution phase lasts longer than charge accumulation time T_a. After termination of the charge accumulation, the final part of the DSCR-ADC conversion cycle consists in redistributing the *residual charge* according to the classical algorithm used in the SCR-ADC. Figure 6.22 presents a comparison of timing of the SCR-ADC and DSCR-ADC conversion cycles. In particular, note that the conversion time T_{C_DSCR} of the DSCR-ADC is lower by T_a from the conversion time T_{C_SCR} of the SCR-ADC. Therefore, the use of the DSCR-ADC is especially attractive in the applications when the charge acquisition time T_a is long.

6.7.2 DSCR-ADC Architecture and Operation

The DSCR-ADC circuit diagram is shown in Fig. 6.23. The architecture of the DSCR-ADC is in general the same as the architecture of the classical *Successive Charge Redistribution Analog-to-Digital Converter* (SCR-ADC) presented in Fig. 6.6. The only difference concerns a lack of the input capacitor and the

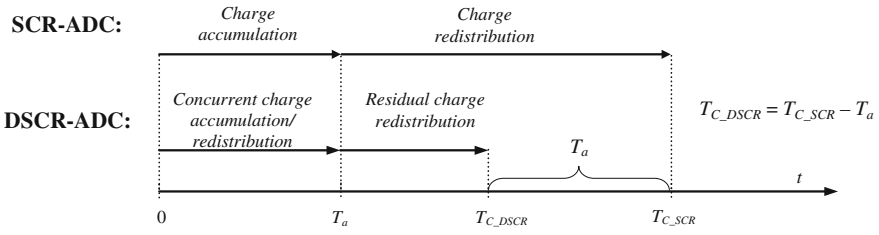

Fig. 6.22 Sequential versus concurrent charge accumulation and redistribution realized respectively in SCR-ADC and in DSCR-ADC

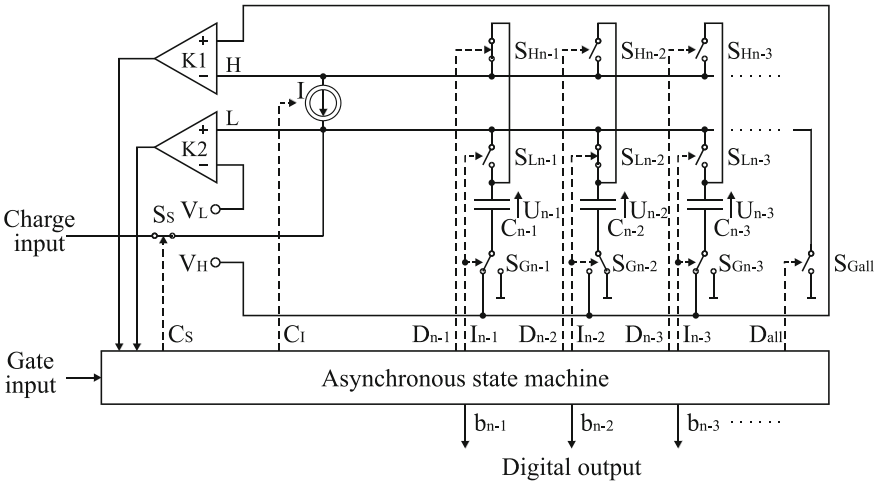

Fig. 6.23 Circuit diagram of the DSCR-ADC. The states of the controlled switches correspond to the second step of the accumulation/conversion phase when the input charge provided to the charge input is collected in the capacitor C_{n-2}

modification of the algorithm of converter operation implemented in the asynchronous state machine. Thus, the number of binary-scaled capacitors used in the n-bit DSCR-ADC equals n instead of $n-1$ as it appears in the SCR-ADC.

The DCSR-ADC operates as follows. Detecting the start of the external gate signal on the gate input by the asynchronous state machine triggers a collection of the incoming charge provided to the charge input in the capacitor C_{n-1} corresponding to the most significant bit (MSB). If the voltage U_{n-1} on the capacitor C_{n-1} reaches the prespecified threshold V_L before the stop of the gate signal is detected by the ASM, the most significant bit b_{n-1} is set to 'one'. Next, the input charge is collected in consecutive capacitors C_{n-2},C_{n-3},… sequentially in the order of decreasing capacitances. In particular, Fig. 6.23 shows the DSCR-ADC configuration during the time when the charge is collected in the capacitor C_{n-2}. If the voltages on these capacitors reach the desired value V_L, then the relevant output bits b_{n-2},b_{n-3},… are set to 'one' (Fig. 6.24).

Assume that the stop of the gate signal occurs in the kth step when charge is collected in the capacitor C_{n-k} where $k \in \{1, \ldots, n\}$. The detection of the stop of

Conversion step number:	1	2 …	*k*-1	*k*	*k*+1	*k*+2	...	*n*	
Bit index:	b_{n-1}	b_{n-2} …	b_{n-k+1}	b_{n-k}	b_{n-k-1}	b_{n-k-2}	…	b_0	
Bit state:	**1**	**1** …	**1**	**0**	***x***	***x***	…	***x***	$x=\{0,1\}$
	Concurrent charge accumulation and redistribution				*Residual charge redistribution*				

Fig. 6.24 Sequence of conversion phases in the DSCR-ADC

the gate signal terminates a delivery of charge to the capacitor array and starts to redistribute the charge portion stored in the latterly charged capacitor C_{n-k} (*residual charge*) into the set of binary-weighted capacitors C_{n-k-1},…,C_0 in the subsequent steps $k + 1$, $k + 2$,…, n.

The bits $b_{n-1}, b_{n-2}, \ldots, b_{n-k+1}$are evaluated to 'one' since all the capacitors $C_{n-1}, C_{n-2}, \ldots, C_{n-k+1}$ had been charged to the desired voltage V_L before the stop of the gate signal is detected. In general, the bit b_{n-k} is the most significant bit whose state is set to 'zero' (Fig. 6.24). Theoretically, the bit b_{n-k} can be set to 'one' in an ideal situation if the process of charging the capacitor C_{n-k} is stopped exactly when the stop of the gate signal is detected by the state machine. In such situation, the conversion phase is completed exactly after termination of the charge accumulation, and charge-to-digital conversion takes a non-redundant conversion time, that is, the digital word is produced immediately as soon as the charge accumulation is terminated ($T_{C\_DSCR} = 0$) (compare Fig. 6.20).

The index value $n - k$ of the capacitor C_{n-k} depends on the duration of the input charge accumulation T_a. The number k equals one for small values of Q_{In}, and respectively $k = n$ if Q_{In} is close to the converter range Q. In the first step of *residual charge redistribution*, which is at the same time the $(k + 1)$th conversion step, the capacitor C_{n-k} starts to operate as the first source capacitor whereas the role of the destination capacitor plays the C_{n-k-1}. Further conversion proceeds in the same way as in the basic version of the SCR-ADC during the charge redistribution. The residual charge redistribution is carried out by the current source I whose intensity determines the speed of the residual charge transfer process.

6.7.3 Event-Driven Charge Transfer in the DSCR-ADC

In the DSCR-ADC, four classes of events drive the conversion process as in the SCR-ADC (see Sect. 6.3.2). However, there is a certain difference in algorithm of the ASM in the DSCR-ADC compared to the SCR-ADC. The difference consists in the condition terminating the concurrent charge accumulation/redistribution. More specifically, the input charge is stopped to be collected in the capacitor C_{n-k} not due to a detection of the output signals from the comparators $K1$, $K2$ but because the stop of the gate signal occurs. This is the violation of the principle of the redistribution algorithm used in the SCR-ADC where the stop of charging any destination capacitor is triggered only by the active signal on one of the comparators $K1$, $K2$ outputs (Sect. 6.3.2). Thus, the algorithm of the DSCR-ADC is more complex compared to the algorithm of the SCR-ADC which results in higher complexity of its hardware implementation in the ASM.

6.7.4 Source and Destination Capacitor Selection in DSCR-ADC

In the DSCR-ADC, the destination capacitors *Dest_C*(*i*) are selected according to the formula (6.4) or (6.6) as in the SCR-ADC. The status of the source capacitor is assigned first time in the $k + 1$ step to the capacitor C_{n-k} that acted as the destination capacitor in the charge accumulation phase in the *k*th conversion step. The selection of the *Source_C*(*i*) in the conversion steps $k + 2$, $k + 3$,…, n proceeds according to the same algorithm as in the SCR-ADC and is stated by the formula (6.5), thus:

$$Source\_C(k+1) = C_{n-k} \text{ if } b_k = 0 \text{ and } b_{k+1} = b_{k+2} = \cdots = b_{n-1} = 1 \quad (6.23)$$

$$Source\_C(i+1) = \begin{cases} Source\_C(i); & \text{if}, b_i = 1 \\ Dest\_C(i); & \text{if}, b_i = 0 \end{cases}; i = k+1, k+2, \ldots, n \quad (6.24)$$

The status of the source capacitor *Source_C*(*i*) in the step *i*, $i \geq 2$ is assigned to the capacitor $C_j, j \leq n - i$ such that:

$$Source\_C(i) = C_j \quad (6.25)$$

where

$$j = \min\left(l; l = n - i + 1, \ldots, n - 1 : b_l = 0\right), \text{ for } i \geq 2 \quad (6.26)$$

As follows from the formulae (6.25)–(6.26), the capacitor C_j acts as the source capacitor *Source_C*(*i*) in *i*th conversion step where $i \geq 2$ provided that the bit b_j is the least significant bit among the bits $b_{n-i+1}, b_{n-i+2}, \ldots, b_{n-1}$ whose state has been evaluated to 'zero'. In particular, if $k \geq n - 2$, then no capacitor operates as the source capacitor in the DSCR-ADC conversion cycle because no residual charge redistribution phase occurs.

6.7.5 *Selection of* Source_C(i) *and* Dest_C(i) *in DSCR-ADC Exemplified Cycle*

In Table 6.2, the course of 6 initial steps of *n*-bit conversion cycle in the DSCR-ADC is specified for the same exemplified states of output bits as in Table 6.1. The concurrent charge accumulation/redistribution terminates in step 3 since b_{n-3} is the most significant bit whose state has been evaluated to 'zero'. Thus, the residual charge redistribution starts in step 4 (condition (6.6)). In later steps (5, 6,…), the *Source_*(*i*) and *Dest_*(*i*) are selected according to the conditions (6.4) and (6.5).

Table 6.2 Example of 6 initial steps of n-bit DSCR-ADC conversion cycle

Stage	Conversion step index (i)	*Source_C*(i)	*Dest_C*(i)	Index of bit tested	Example of evaluated bit states	Active control signals	Lower bound of uncertainty interval ΔQ_{In} at the end of step	Upper bound of uncertainty interval ΔQ_{In} at the end of step
Relaxation	–	–	–	–	–	D_{all}, D_{n-1}, I_0,…, I_{n-1}	–	–
Charge accumulation/ redistribution	1	–	C_{n-1}	n-1	1	C_S, D_{n-1}, I_{n-1}	Q/2	Q
	2	–	C_{n-2}	n-2	1	C_S, D_{n-1}, I_{n-2}	3Q/4	Q
	3	–	C_{n-3}	n-3	0	C_S, D_{n-1}, I_{n-3}	3Q/4	7Q/8
Residual charge redistribution	4	C_{n-3}	C_{n-4}	n-4	1	C_I, D_{n-3}, I_{n-4}	13Q/16	7Q/8
	5	C_{n-3}	C_{n-5}	n-5	0	C_I, D_{n-3}, I_{n-5}	13Q/16	27Q/32
	6	C_{n-5}	C_{n-6}	n-6	…	C_I, D_{n-5}, I_{n-6}	…	…

In Table 6.2, the control signals that are active in the relaxation, accumulation, and during successive conversion steps are also listed. These signals are referred to the DSCR-ADC converter architecture presented in Fig. 6.23 (see also Fig. 6.19b). The functionality of these signals is the same as for the SCR-ADC and has been discussed in Sect. 6.5.

6.7.6 Conversion Time in DSCR-ADC

As explained in Sect. 6.1, the T_{C_DSCR} is smaller from the $T_{C\ SCR}$ by the duration of the accumulation phase T_a. Therefore, the conversion time T_{C_DSCR} of the DCSR-ADC equals just the duration of residual charge redistribution phase (see Fig. 6.22). Taking into account the analysis of the $T_{C\ SCR}$ included in Sect. 6.3, the T_{C_DSCR} may be found as follows:

$$\begin{aligned} T_{C\_DSCR}(Q_{In}) &= T_{C\_SCR}(Q_{In}) - T_a \\ &= \sum_{k=1}^{n}\left[b_{n-k}\frac{T}{2^k} + (1-b_{n-k})\cdot\left(\frac{Q_{In}}{I} - \sum_{i=1}^{k} b_{n-i}\frac{T}{2^i}\right)\right] - T_a \end{aligned} \tag{6.27}$$

The plot of T_{C_DSCR}/T versus Q_{In}/Q for the 12-bit DSCR-ADC according to (6.27) is shown in Fig. 6.20 and marked with red line. The DSCR-ADC maximum conversion time equals nearly the half of the maximum SCR-ADC conversion time (compare (6.18)–(6.27)):

$$T_{C\_DSCR\,\max} \cong T_{C\_SCR\,\max}/2 = T\frac{2^n-1}{2^{n+1}} \cong \frac{T}{2} \tag{6.28}$$

The conversion time T_{C_DSCR} reaches its minimum value equal to zero if Q_{In} amounts to any of values from the set $\{Q/2,\ 3Q/4,\ 7Q/8,\ldots,(2^n-1)Q/2^n\}$. It happens when the output word is of a thermometer code type (11…100…0). Finally:

$$0 \le T_{C\_DSCR} \le T/2. \tag{6.29}$$

6.7.7 DSCR-ADC Charge Redistribution Effectiveness

The mean number of transfers of each charge unit included in the input charge portion can be for the DSCR-ADC found as follows (compare the formula (6.21)):

$$p_{C\_DSCR} = \frac{IT_{C\_DSCR}}{Q_{In}} \tag{6.30}$$

The plot of p_{C_DSCR} versus input charge Q_{In}/Q for $n = 12$ is presented in Fig. 6.20. It can be proved on the basis of (6.18) and (6.27) that:

$$p_{C\_DSCR} = p_{C\_SCR} - 1 \tag{6.31}$$

The p_{C_DSCR} is lower by one from p_{C_SCR} because the input charge is placed in the destination capacitors directly without a participation of the input capacitor. Thus, the mean number of transfers of each charge unit per conversion cycle is bounded as follows:

$$0 \leq p_{C\_DSCR} \leq n - 1 \tag{6.32}$$

The formula (6.32) shows that the conversion effectiveness is higher for DSCR-ADC than for SCR-ADC. In particular, p_{C_SCR} equals zero for selected values of the input charge corresponding to the output words of a thermometer code type because there is no residual charge redistribution within such conversion cycles and the input charge finds the final destination directly during the accumulation phase.

6.7.8 DSCR-ADC versus SCR-ADC

Due to several advantages, the DSCR-ADC is as the enhanced version of the SCR-ADC. First of all, the DSCR-ADC is characterized by significantly reduced conversion time and the energy consumption which is directly proportional to the conversion time. More specifically, the conversion time is shorter in the DSCR-ADC by the charge accumulation time T_a (Fig. 6.22). In particular, the DSRC-ADC conversion time and energy consumption is reduced at least twice for small values of Q_{In}, and at least four times for large Q_{In}, compared to the conversion time and energy consumption of the SCR-ADC. Moreover, the conversion time of the DCSR-ADC is very short for selected values of the Q_{In} and may even equal zero. In particular, it means that the energy needed to redistribute the charge for such Q_{In} values equals zero since the charge is deployed directly in the destination capacitors during the charge accumulation phase. By comparison, the zero-energy conversion cycles do not occur in classical C-ADCs. The conversion time jitter (i.e., the difference between the maximum and minimum conversion time) is reduced twice in the DSCR-ADC in relation to the SCR-ADC.

Second, in the DSCR-ADC, the capacitor array contains only a number of n sections, instead of $n + 1$ in the SCR-ADC. The die area is thus reduced twice since the input capacitor C_n equals approximately the sum of the other capacitances $C_{n-1}, \ldots, C_0$ in the capacitor array. The latter benefit is of significant importance since the die area required to implement the capacitor array occupies usually a large majority of the die area of the whole converter architecture. Finally, the number of state transitions per a single conversion in the DSCR-ADC equals $n + 1$ instead of $n + 2$ in the SCR-ADC, which slightly decreases power consumption.

6.8 ADCs with Event-Driven SCR for Signals Originated in Other Domains

The concept of the SCR-ADC and DSCR-ADC can be adapted to a conversion of the other physical magnitudes transformed previously to electric charge especially if the translation is linear. We present the frameworks for time-to-digital and voltage-to-digital conversion based on the event-driven successive charge redistribution although the appropriate schemes for the other physical magnitudes can be also proposed.

6.8.1 Time-to-Digital Converter with Successive Charge Redistribution (SCR-TDC)

The proposed time-to-charge translation is based on collecting a charge value Q_{In} delivered by the current source of constant intensity I_a during the discretized input time interval T_{In} (Fig. 6.25) [69]. Thus, the charge portion Q_{In} accumulated in the input capacitor C_n is proportional to T_{In} as follows:

$$Q_{In} = I_a T_{In} \tag{6.33}$$

The converter input range, that is, the maximum input time interval $T = \max(T_{In})$ converted by the SCR-TDC is determined by the duration of charging C_n to the desired voltage V_L which equals:

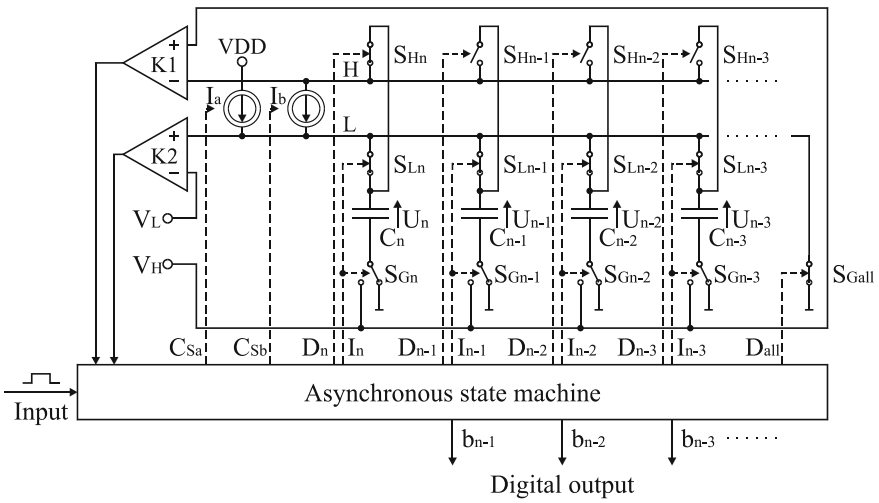

Fig. 6.25 Time-to-charge translation in SCR-TDC. The switch positions correspond to the relaxation phase

$$T = V_L C_n / I_a = 2^n V_L C_0 / I_a \tag{6.34}$$

The time-to-charge translation is an equivalent of a charge accumulation in the SCR-ADC. In the second phase of a conversion cycle in the SCR-TDC, the input charge Q_{In} is processed in the charge domain in a number of n steps by event-based successive charge redistribution in the binary-scaled capacitor array according to the algorithm specified extensively above. The current source used for charge redistribution can be the same as used for time-to-charge translation (I_a). However, in order to accelerate the conversion process, it is preferred to provide an extra current source (I_b) of higher intensity ($I_b \gg I_a$) dedicated only to charge redistribution. In the configuration with two current sources, the conversion time of the SCR-TDC is independent of the duration of input time interval T_{In} and much shorter than T_{In}. On the other hand, in the configuration with the single current source used both during the charge accumulation and the redistribution, the SCR-TDC conversion time depends on the T_{In} [25].

The analysis of the conversion time of the SCR-TDC is carried out in [26]. As proved in [30], the maximum conversion time $T_{C\_SCR\max}$ approaches asymptotically T/m where $m = I_b/I_a$ and T is the maximum input time interval defined by (6.34). In particular, for the SCR-TDC version with the single current source ($I_b = I_a$), the maximum conversion time $T_{C\_SCR\max}$ reaches the maximum input time interval T. The parameter m is tuned by the intensity of the current source I_b used for charge transfer and demonstrates how many times the maximum conversion time can be reduced in relation to the SCR-TDC input range. The selection of I_b is based on the same rules as the selection of I in the SCR-ADC.

The SCR-TDC can be implemented in the version with direct successive charge redistribution (DSCR-TDC) [25] that provides similar benefits as the DSCR-ADC especially in terms of reduction of the conversion time [26].

The concept of the SCR/DSCR-TDC may be referred to time-to-digital converters based on successive approximation (SA-TDC) carried out strictly in the time domain and exploited successfully in recent years [28–32]. Like SCR/DSCR-TDC, the SA-TDC algorithm is based on successive reduction of the quantization error and is characterized by a redundant conversion time. The first SA-TDC has been introduced in patent description [28]. The technique of SA-TDCs was further developed in [29–32].

As mentioned in Introduction, the SCR/DSCR-TDCs may be used for clockless time quantization in asynchronous ADCs for a new class signal processing chain based on time-encoded signals and irregular digital data records.

6.8.2 *Voltage-to-Digital Converter with Successive Charge Redistribution (SCR-VDC)*

Most signals originates in the voltage domain and most systems are still based on processing signals in the amplitude domain. To establish the link to classical

conversion schemes, the event-based successive charge redistribution algorithm can be adopted to digitize voltage signals.

We present two versions of voltage-to-charge mapping via voltage-mode and via current-mode voltage-to-charge translation accordingly. In both versions, the charge value Q_{In} proportional to the input voltage value V_{In} is collected in the input capacitor C_n:

$$Q_{In} = V_{In}C_n \tag{6.35}$$

In the voltage-mode SCR-VDC shown in Fig. 6.26a, the voltage-to-charge translation is realized by coupling the input capacitor C_n to the input voltage source V_{In} for the time long enough to charge the input capacitor C_n to the input voltage V_{In} with the accuracy higher than the converter resolution. The duration of creating a voltage on the input capacitor via the closure of the switch S_S is controlled by the signal C_S produced by the state machine and is determined a priori.

The process of charging the capacitor C_n is similar to sampling operation employed in classical sample-and-hold (S/H) circuits. However, the first difference is that the input capacitance C_n is usually higher than the capacitance used in S/H circuits because $C_n = 2^n C_0$ where C_0 is the unit capacitance in the capacitor array and the lower bound for C_0 is limited by implementability of the capacitance in silicon. The other difference consists in that, unlike the voltage on the S/H circuit output, the input charge Q_{In} is not conserved during the conversion process but is successively transferred to destination capacitors.

The voltage-mode SCR-VDC is not fully self-timed because it requires providing an extra signal to the gate input to control the voltage-to-charge translation. To avoid loading the input voltage source V_{In} with the current that charges the input capacitor C_n, the separator may be introduced between the voltage source and the voltage input. However, such solution requires to use an additional operational amplifier which costs energy and reduces conversion accuracy due to extra offset.

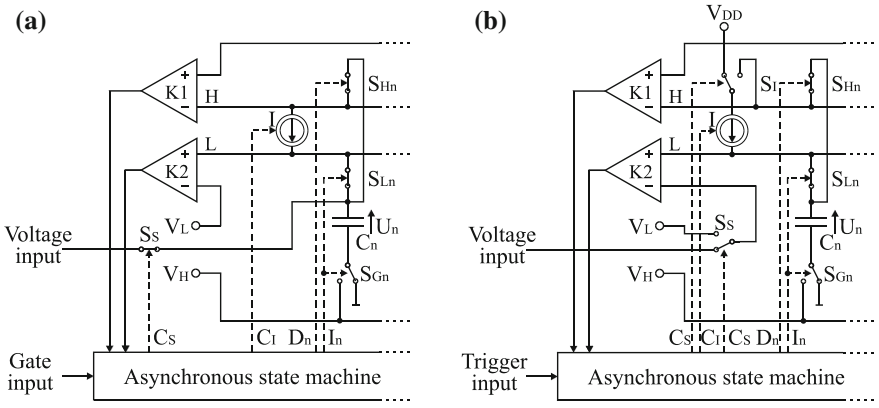

Fig. 6.26 Voltage-mode SCR-VDC (**a**) and current-mode SCR-VDC (**b**)

In the current-mode SCR-VDC presented in Fig. 6.26b, the voltage-to-charge translation is realized by charging the input capacitor C_n to the input voltage value V_{In} by the current source I. The crossing of the level V_{In} by the voltage U_n on the capacitor C_n is reported by the comparator $K2$. The current-mode SCR-VDC is thus fully self-timed and does not need neither any external control nor any external timing. During the current-mode voltage-to-charge translation, the input voltage source is not loaded with the current charging the input capacitor C_n. The disadvantage is however that the time of voltage-to-charge mapping $t_{In} = V_{In}C_n/I$ is variable and grows proportionally with the input voltage value V_{In}. The time t_{In} is usually longer than the time of S/H operation in the previous version of the SCR-ADC because the current source intensity I must be limited by the comparator delay to guarantee an accurate V_{In}-level-crossing detection. In order to guarantee the declared resolution, the inaccuracy of the voltage-to-charge translation must be lower than the *LSB*.

The SCR-VDC cannot be implemented in the version with direct successive charge redistribution (DSCR-VDC).

6.9 Conclusions

The present study deals with a concept of clockless analog-to-digital conversion based on event-driven successive charge redistribution. Due to moving the quantization process in the charge domain, it is predisposed to low voltage implementations. Furthermore, during idle intervals between subsequent conversion cycles, the circuit dissipates only static power. The proposed conversion method realized in the charge domain may be adapted both for voltage or time-encoded signals.

References

1. Matsuzawa, A.A.: Design challenges of analog-to-digital converters in nanoscale CMOS. IEICE Trans. Electron. **E90-C**, 779–785 (2007)
2. Yang, H.Y., Sarpeshkar, R.: A time-based energy-efficient analog-to-digital converter. IEEE J. Solid-State Circ. **40**(8), 1590–1601 (2005)
3. Lazar, A.A., Tóth, L.T.: Perfect recovery and sensitivity analysis of time encoded bandlimited signals. IEEE Trans. Circ. Syst. -I: Regul. Pap. **52**, 2060–2073 (2005)
4. Lazar, A.A., Simonyi, E.K., Toth, L.T.: Time encoding of bandlimited signals, an overview. In: Proceedings of the Conference on Telecom. Systems, Modeling and Analysis (2005)
5. Lazar, A.A., Pnevmatikakis, E.A.: Video time encoding machines. IEEE Trans. Neural Networks **22**, 461–473 (2011)
6. Taillefer, C.S., Roberts, G.W.: Delta–Sigma A/D conversion via time-mode signal processing. IEEE Trans. Circ. Syst.-I: Regul. Pap. **56**(9), 1908–1920 (2009)
7. Kościelnik, D., Miśkowicz, M.: Asynchronous sigma-delta analog-to-digital converter based on the charge pump integrator. Analog Integr. Circ. Sig. Process **55**, 223–238 (2008)

8. Daniels, J., Dehaene, W., Steyaert, M.S.J., Wiesbauer, A.: A/D conversion using asynchronous delta-sigma modulation and time-to-digital conversion. IEEE Trans. Circ. Syst.-II: Exp Briefs **57**(9), 2404–2412 (2010)
9. Hernandez, L., Prefasi, E.: Analog-to-digital conversion using noise shaping and time encoding. IEEE Trans. Circ. Syst.-I: Regul. Pap. **55**(7), 2026–2037 (2008)
10. Pekau, H., Yousif, A., Haslett, J.W.: A CMOS integrated linear voltage-to-pulse-delay-time converter for time based analog-to-digital converters. Proc. IEEE Int Symp. Circ. Syst. **2006**, 2373–2376 (2006)
11. Ravinuthula, V., Harris, J.G.: Time-based arithmetic using step functions. Proc. IEEE Int. Symp. Circ. Syst. ISCAS **2004**, 305–308 (2004)
12. Allier, E., Sicard, G., Fesquet, L., Renaudin, M.: A new class of asynchronous A/D converters based on time quantization. Proc. IEEE Int. Symp. Asynchronous Circ. Syst. ASYNC **2003**, 196–205 (2003)
13. Kozmin, K., Johansson, J., Delsing, J.: Level-crossing ADC performance evaluation toward ultrasound application. IEEE Trans. Circ. Syst. Part I: Regul. Pap. **56**, 1708–1719 (2009)
14. Guan, K.M., Kozat, S.S., Singer, A.C.: Adaptive reference levels in a level-crossing analog-to-digital converter, EURASIP J. Adv. Sig. Processing **2008**, 11 (Article ID 513706) (2008)
15. Kurchuk, M., Tsividis, Y.: Signal-dependent variable-resolution clockless A/D conversion with application to continuous-time digital signal processing. IEEE Trans. Circ. Syst. Part I: Regul. Pap. **57**, 982–991 (2010)
16. Trakimas, M., Sonkusale, S.R.: An adaptive resolution asynchronous ADC architecture for data compression in energy constrained sensing applications. IEEE Trans. Circ. Syst. Part I: Regul. Pap. **58**, 921–934 (2011)
17. Senay, S., Chaparro, L.F., Sun, M., Sclabassi, R.J.: Adaptive level-crossing sampling and reconstruction. Proc. of European Signal Processing Conf. EUSIPCO 1296–1300 (2010)
18. Tsividis, Y.: Event-driven data acquisition and digital signal processing: a tutorial. IEEE Trans. Circ. Syst II: Exp Briefs **57**, 577–582 (2010)
19. Miśkowicz, M.: Send-on-delta concept: an event-based data reporting strategy. Sensors **6**, 49–63 (2006)
20. Kościelnik, D., Miśkowicz, M.: Method and apparatus for conversion of portion of electric charge to digital word. PCT Patent Application WO 2011/152743, 2011
21. Kościelnik, D., Miśkowicz, M.: Method and apparatus for conversion of time interval to digital word. PCT Patent Application WO 2011/152744, 2011
22. Kościelnik, D., Miśkowicz, M.: Method and apparatus for conversion of voltage value to digital word. PCT Patent Application WO 2011/152745, 2011
23. Kościelnik, D., Miśkowicz, M.: A new method of charge-to-digital conversion. Proc. IEEE Int. Mixed-Signals, Sens. Syst. Test Workshop IMS3TW (2010)
24. Kościelnik, D., Miśkowicz, M.: A clockless time-to-digital converter. Proc. IEEE Convention Elect. Electron. Eng. Israel IEEEI **2010**, 516–519 (2010)
25. Kościelnik, D., Miśkowicz, M.: Time-to-digital converter with direct successive charge redistribution. In: Proceedings of IMEKO International Workshop on ADC Modelling, Testing and Data Converter Analysis and Design IWADC 2011, 2011
26. Kościelnik, D., Miśkowicz, M., Jabłeka, M.: Analysis of conversion time of time-to-digital converters with charge redistribution. Proceeding of IMEKO International Workshop on ADC Modelling, Testing and Data Converter Analysis and Design IWADC, 2011
27. Allen, P.E., Holberg, D.R.: CMOS analog circuit design, 2nd edn. Oxford University Press, Oxford (2002)
28. Maevsky, O.V., Edel, E.A.: Converter of time intervals to code. USSR Patent 1591183, Bulletin No. 33, 070990
29. Kinniment, D.J., Maevsky, O.V., Bystrov, A., Russell, G., Yakovlev, A.V.: On-chip structures for timing measurement and test. In: Proceeding of IEEE International Symposium Asynchronous Circuits and Systems ASYNC 2002, pp. 190–197 (2002)
30. Abas, M.A., Russell, G., Kinniment, D.J.: Built-in time measurement circuits—a comparative design study. IET Comput. Digital Tech. **1**(2), 87–97 (2007)

31. Mantyniemi, A., Rahkonen, T., Kostamovaara, J.: A CMOS time-to-digital converter (TDC) based on a cyclic time domain successive approximation interpolation method. IEEE J. Solid-State Circ. **44**(11), 3067–3078 (2009)
32. Al-Ahdab, S., Mantyniemi, A., Kostamovaara, J.: Cyclic time domain successive approximation time-to-digital converter (TDC) with sub-ps-level resolution. Proceedings of IEEE Instrumentation and Measurement Technology Conference I2MTC 2011, pp. 1–4 (2011)
33. Kazmierkowski, M.P., Malesani, L.: Current control techniques for three-phase voltage-source PWM converters: a survey. IEEE Trans. Ind. Electron. **45**(5), 691–703 (1998)
34. Malinowski, M., Jasinski, M., Kazmierkowski, M.P.: Simple direct power control of three-phase PWM rectifier using space-vector modulation (DPC-SVM). IEEE Trans. Ind. Electron. **51**(2), 447–454 (2004)
35. Inose, H., Aoki, T., Watanabe, K.: Asynchronous delta modulation system. Electron. Lett. **2**(3), 95–96 (1966)
36. Kikkert, C.J., Miller, D.J.: Asynchronous delta sigma modulation, Proc. IREE **36**, 83–88 (1975)
37. Guan, K., Singer, A.C.: A level-crossing sampling scheme for bursty signals. Proc. Int. Conf. Inform. Sci. Syst. **3**, 1357–1359 (2006)
38. Wang, T., Wang, D., Hurst, P.J., Levy, B.C., Lewis, S.H.: A level-crossing analog-to-digital converter with triangular dither. IEEE Trans. Circ. Syst. Part I: Regul. Pap. **56**(9), 2089–2099 (2009)
39. Guan, K.M., Singer, A.C.: Opportunistic sampling by level-crossing. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'07), vol. 3, pp. 1513–1516, Honolulu, 2007
40. Senay, S., Oh, J., Chaparro, L.F.: Regularized signal reconstruction for level-crossing sampling using Slepian functions. Sig. Process. **92**, 1157–1165 (2012)
41. Kafashan, M., Beygi, S., Marvasti, F.: Asynchronous analog-to-digital converter based on level-crossing sampling scheme. EURASIP J. Adv. Sig. Proc., **2011**, 109–117 (2011)
42. Yen, J.L.: On nonuniform sampling of bandwidth-limited signals. IRE Trans. Circ. Theory **CT-3**, 251–257 (1956)
43. Beutler, F.J.: Error-free recovery of signals from irregularly spaced samples. SIAM Rev. **8**, 328–335 (1966)
44. Mark, J., Todd, T.: A nonuniform sampling approach to data compression. IEEE Trans. Commun. **29**(1), 24–32 (1981)
45. Schell, B., Tsividis, Y.: A continuous-time ADC/DSP/DAC system with no clock and activity-dependent power dissipation. IEEE J. Solid-State Circuits **43**(11), 2472–2481 (2008)
46. McCreary, J.L., Gray, P.R.: All-MOS charge redistribution analog-to-digital conversion techniques I. IEEE J. Solid-State Circuits **10**(6), 371–379 (1975)
47. Steele, R.: Delta Modulation Systems. Wiley, New York (1975)
48. Data Converters History, in: *Analog-Digital Conversion*, ed. by W. Kester, Analog Devices Inc., USA, 2004
49. Rouse Ball, W.W., Coxeter, H.S.M.: Mathematical Recreations and Essays. Dover Publications, Thirteenth Edition (1987)
50. Goodall, W.M.: Telephony by Pulse Code Modulation. Bell Syst. Tech. J. **26**, 395–409 (1947)
51. Bernard M. Gordon and Robert P. Talambiras, Signal Conversion Apparatus. U.S. Patent 3,108,266
52. C.W. Barbour, Simplified PCM Analog-to-Digital Converters Using Capacity Charge Transfer. In: Proceedings National Telemetering Conference, pp. 4.1–4.11. Chicago, 1961
53. T. Kugelstadt, The Operation of the SAR-ADC Based on Charge Redistribution, *Texas Instruments Analog Applications Journal*, pp. 10–12, Feb. 2000
54. Scott, M.D., Boser, B.E., Pister, K.S.J.: An ultra low-energy ADC for smart dust. IEEE J. Solid-State Circuits **38**(7), 1123–1129 (2003)

55. Hong, H., Lee, G.: A 65fJ/conversion-step 0.9-V 200-kS/s rail-to-rail 8-bit successive approximation ADC. IEEE J. Solid-State Circuits **42**(10), 2161–2168 (2007)
56. B.P. Ginsburg and A.P. Chandrakasan, An energy-efficient charge recycling approach for a SAR converter with capacitive DAC. In: Proceedings of the IEEE ISCAS, pp. 184–187, 2005
57. R.Y.-K. Choi and C.-Y. Tsui, A low energy two-step successive approximation algorithm for ADC design. In: Proceedings of the IEEE ISQED, pp. 317–320, 2008
58. Saberi, M., Lotfi, R., Mafinezhad, K., Serdijn, W.A.: Analysis of Power Consumption and Linearity in Capacitive Digital-to-Analog Converters Used in Successive Approximation ADCs, pp. 1736–1748. IEEE Trans. Circuits Syst. I, Regular Papers (2011)
59. K.-Y. Khoo and A. Willson, Charge Recovery on a Databus. In: Proceedings of the International Symposium on Low Power Electrical and Design, pp. 185–189, 1995
60. Brian P. Ginsburg and Anantha P. Chandrakasan, 500-MS/s 5-bit ADC in 65-nm CMOS With Split Capacitor Array DAC, IEEE J. Solid-State Circ., **42**(4) (2007)
61. M.F. Tompsett, Semiconductor charge-coupled device analog-to-digital converter. U.S. Patent 4136335, 1979
62. Paul E. Green, Charge domain successive approximation analog to digital converter. Patent US 5010340
63. Kyung, C.M., Kim, C.K.: Pipeline analog-to-digital conversion with charge-coupled devices. IEEE J. Solid-State Circuits **15**, 255–257 (1980)
64. Kyung, C.M., Kim, C.K.: Charged-coupled analog-to-digital converter. IEEE J. Solid-State Circ. **16**(6), 621–626 (1981)
65. D. Kościelnik, M. Miśkowicz, Method and apparatus for analog-to-digital conversion using asynchronous Sigma-Delta modulation. U.S. Patent 7948413, 2011
66. Roza, E.: Analog-to-digital conversion via duty-cycle modulation. IEEE Trans. Circ. Syst. II **44**, 907–914 (1997)
67. Sayiner, N., Sorensen, H.N., Viswanathan, T.R.: A level-crossing sampling scheme for A/D conversion. IEEE Trans. Circ. Syst. II **43**, 335–339 (1996)
68. D. Kościelnik, M. Miśkowicz, Modeling event-driven successive charge redistribution in ADC with varying rate of charge transfer. In: Proceeding of IEEE Convention of Electrical and Electronics Engineers in Israel IEEEI 2012, 2012
69. Kościelnik, D., Miśkowicz, M.: Time-to-digital converters based on event-driven successive charge redistribution: a theoretical approach. Measurement **45**, 2511–2528 (2012)

Chapter 7
Time-to-Digital Converters

Ryszard Szplet

7.1 Introduction

Dynamic development in science and technology in the second half of the twentieth century caused, among others, an increase in the interest in methods and techniques for precise measurement of time interval that elapses between two physical events. The main objective of the meters which are used for such purposes is the creation of a numerical representation of the measured time interval with as high accuracy and precision as possible. Since the result of measurement is usually presented in digital form, this operation is called a *time-to-digital* (T/D) conversion, while measuring devices are universally called *time-to-digital converters* (TDCs). However, this term is quite general and in the metrological practice its meaning is often limited to devices with a narrow measurement range (<100 ns) while the devices offering a wider range, usually based on the interpolation methods, are called *time interval counters* (TICs). The name *time digitizer* can also be found in the literature to describe both TDCs and TICs.

The early devices for precise time interval measurement, developed in 70s and 80s of the last century, were based on analog conversion methods. Those devices offered very high resolution and precision reaching the level of single picoseconds. However, as they were built up of discrete components the devices were large and bulky, consumed a lot of power and were sensitive to operating conditions. Recently the digital methods have become dominant due to the ease of implementation in integrated circuits, shorter conversion times and higher immunity to external disturbances. In this chapter, the most representative methods and techniques used for the measurement of time interval with high resolution and precision are described. This includes time stretching, time-to-amplitude followed by amplitude-to-digital conversion, the counting method, direct time-to-digital

R. Szplet (✉)
Military University of Technology, Gen. S. Kaliskiego 2, Warsaw, Poland
e-mail: rszplet@wat.edu.pl

P. Carbone et al. (eds.), *Design, Modeling and Testing of Data Converters*, Signals and Communication Technology, DOI: 10.1007/978-3-642-39655-7_7,

conversions with a digital delay line in the single and Vernier versions, as well as single- and two-stage interpolation methods.

Valuable supplementary information on precise T/D conversion may be found in some previous review works [1–3]. Since the parameters of recent TDCs and methods used are strongly connected with microelectronic technology, its dynamic development allows for continuous progress in precise time interval metrology. Therefore, the most topical and detailed information is always contained in scientific databases (e.g., *Web of Science, Springer, Scopus, Elsevier*), but browsing through databases of patent offices may also be inspiring (e.g., www.epo.org, www.uspto.gov).

In general, the performance of TDCs are characterized by parameters typical for A/D converters [4]. However, on account of its specificity, definitions of the most important terms are quoted below. *Resolution* (denoted by q or LSB—least significant bit) is understood as the lowest value of the measured time interval that can be distinguished by a converter. It corresponds to the width of an ideal code bin at the transfer function of TDC (Fig. 7.1). In a real converter the widths of actual code bins differ from that of an ideal TDC. This causes problems with linearity of conversion which is described quantitatively by *differential nonlinearity* (DNL) and *integral nonlinaerity* (INL). The former one is a measure of evenness of code bins and its value for ith bin is given by: $\mathrm{DNL}_i = \mathrm{LSB}_i - \mathrm{LSB}$, where LSB and LSB_i are the widths of ideal and actual ith code bin, respectively. The latter one illustrates the degree of discrepancy between the ideal and actual transfer functions of a TDC (Fig. 7.1). For the ith code bin the value of integral nonlinearity is calculated using the formula $\mathrm{INL}_i = \Sigma\mathrm{DNL}_i$. The most informative and commonly used measure is the maximum value of INL. Precision or *single-shot precision* of a TDC (denoted by σ) is expressed in terms of the standard

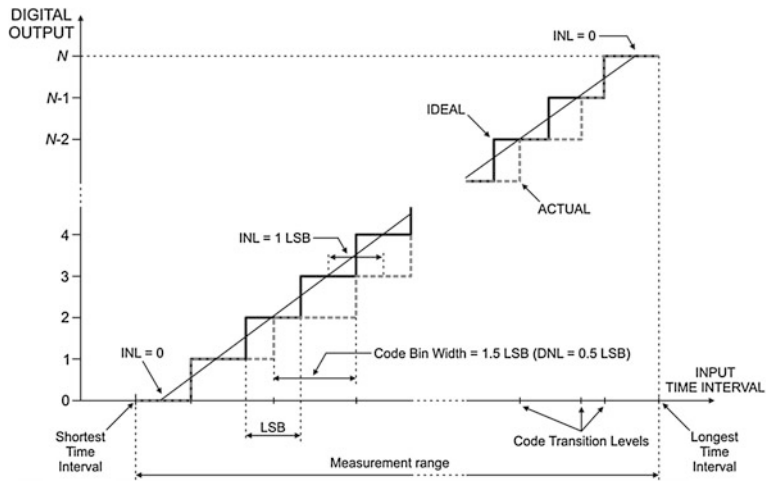

Fig. 7.1 TDC transfer function

deviation of the distribution of measurement readouts obtained as a result of constant time interval measurements performed in repeatable conditions. Usually the worst case value or the characteristic of the single-shot precision as a function of the time interval within a single clock period is provided for the narrow-range TDCs, and such characteristic determined over the whole measurement range is presented for interpolating TICs. *Measurement range* (MR) is understood as the difference between the longest and shortest time intervals measured by the converter (Fig. 7.1). *Measurement rate* determines the maximum frequency at which correct measurements can be performed. This frequency depends, among others, on the value of *dead time* needed for a converter to store or send the conversion data and to reset itself preparing for the next conversion process.

7.2 Counting Methods

The simplest, and one could even say classical, method of time interval measurement is the counting method. In this method pulses START and STOP that represent, respectively, the beginning and end of the measured time interval T are used to create an appropriate time gate (ENABLE in Fig. 7.2). This gate is then used as an enabling signal for binary counter counting periods T_C of a reference clock. The value of measured time interval is calculated from the simple formula $T = n\ T_C$, where n is a decimal equivalent to content of the counter after measurement. Since the beginning and end of the measured time gate may enable and disable the counter at any moment within the clock period, large measurement errors (ΔT_1 and ΔT_2) due to quantization are observed. The maximum value of quantization error may reach almost $2T_C$ [3].

The undeniable advantage of this method is its simplicity that makes the practical implementation relatively easy. In addition, the measurement range of such TDC may easily be extended by adding flip-flops to the counter. Each subsequent flip-flop doubles the range. As they are the main drawbacks that significantly limit its applicability in modern measurement devices the low resolution and precision must be mentioned. The resolution depends on the frequency of the

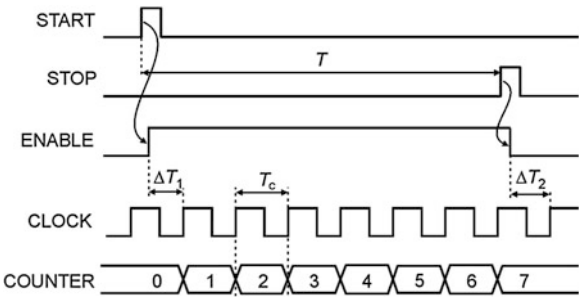

Fig. 7.2 Principle of counting method

reference clock and in a single measurement is equal to the period of the clock ($q = T_C$). Thus, the higher the frequency, the better the resolution. In modern integrated circuits the maximum value of the counter operating frequency is limited technologically and in the new CMOS technology (40 nm), for instance, the maximum applied frequency is 6.5 GHz resulting in a resolution of about 154 ps [5]. Apart from technological limitations of CMOS devices, a spectacular increase in frequency is questionable due to the proportional increase in power consumption.

If the measured time intervals are repeatable, the resolution of the counting method may be improved by *multiple measurements and averaging* results. The improvement in resolution is as the $\sqrt{N}$, where N is the number of averaged independent time intervals [6]. The fundamental assumption for this approach is a lack of synchronization between input pulses (START and STOP) and the reference clock. Such a solution is used in the counter HP5345A (*Hewlett-Packard*) which allowed a decrease in the value of resolution to 1 ps at the sample size of 4×10^6. Unfortunately, averaging is time consuming and, particularly if the measured time is long, the extended time of a measurement session may deteriorate precision due to changes in the measurement conditions.

Another way to increase the resolution of the counting method is to use a *multiphase clock* (MPC). In such an application, the MPC with k phases is the equivalent of a reference clock with k-times higher frequency and thus the resolution is improved proportionally ($q = 1/(kf_C)$). As periods of each MPC phase have to be counted by a separate counter, the resolution improvement is thus obtained at the expense of using $(k - 1)$ additional counters and a multibit adder. The measurement result is given by: $T = \sum n/(kf_C)$, where $\sum n$ is the decimal equivalent of total content of all counters after measurement. In integrated circuits an MPC may be created with the use of a ring oscillator [7] or a tapped delay line. Most often, however, the mentioned solutions are utilized as a part of *Phase Locked Loop* (PLL) [8–10] or *Delay Locked Loop* (DLL) [10–12] to provide better frequency and phase shift stability. The known applications of the counting method with an MPC include only TDCs implemented in FPGA (*Field Programmable Gate Array*) devices. They are equipped with built-in functional blocks dedicated for the management of clock signals (described in Sect. 7.4.2). The use of such blocks allowed to create the four phase clock with frequency of 250 MHz that enabled to reach a resolution of 1 ns [13]. The higher resolution of 78 ps was achieved in the approach where a 16-phase clock with both active edges and 400 MHz frequency was used [14]. A multiphase clock is more commonly used in the first stage of interpolation of TICs with multistage interpolation (see Sect. 7.4).

7.3 High-Resolution, Narrow-Range Time-to-Digital Converters

For measurements of time interval that need higher resolution and precision than provided by the counting methods (<200 ps), other methods in both analog and digital domains are used. The analog methods were introduced first and developed considerably with the coming of discrete electronic elements. However, these methods are based on a signal representation in the voltage domain that entails difficulties in implementation in contemporary, low-voltage nanometer CMOS technologies. Therefore they are now much less popular than they were to be. Instead, we can observe a continuos and rapid increase in the new methods and techniques for time digitization exclusively in digital domain. For obvious reason, the digital circuits cannot distinguish any information in a signal voltage level but they offer very high resolution in the time domain. This is the source of popularity of the conversion technique based on discrete delay lines that are used, in several variants, in most of modern integrated TDCs.

7.3.1 Analog TDC Architectures

Analog T/D converters involve double conversion: first time-to-amplitude conversion and then precision amplitude-to-digital conversion. The latter conversion can be performed directly with the use of a conventional voltage-to-digital converter or indirectly by means of a time stretcher followed by a digital counter.

7.3.1.1 Analog Time Stretcher

One of the oldest analog methods is *analog time stretching* (or *expansion*). In this method, a capacitor C is linearly charged with a constant current I_1 during the

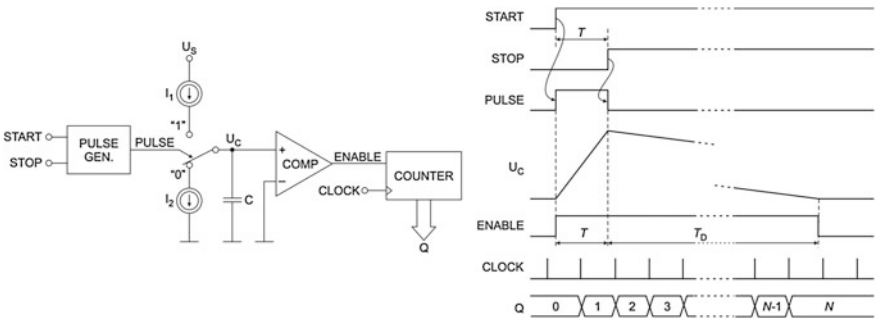

Fig. 7.3 Analog time stretching based TDC

measured time interval T and then is linearly discharged with a much smaller current I_2 (Fig. 7.3). Since the current I_2 is K times smaller than I_1 ($I_2 = I_1/K$), the discharging time of T_D is K times longer than the charging time ($T_D = K\,T$). The coefficient K is called a *stretching* (or *expansion*) *factor* while this dual-slope approach is also known as *time amplification*. The time interval, being the sum of charging and discharging times ($T + T_D$), is measured with the counting method whose typical resolution T_C, in this case, is improved by the stretching factor, i.e. $q = T_C/(K + 1)$. The value of the measured time interval is calculated as $T = N\,T_C/(K + 1)$, where N is the final content of the counter. In this case, as in most cases analyzed below, the quantization error is ignored for analysis simplification purposes.

The stretching factor may even assume values of thousands, hence the resolution of picoseconds may easily be obtained (e.g. $K = 1024$, $q = 19.5$ ps [15]). For further improvement in the resolution a *two-stage time stretching* was proposed [16], and applied in a TIC built of low-cost discrete components allowed to reach the high resolution of 1 ps. In a TIC integrated in BiCMOS technology and based on the single-stage time stretching the 30 ps resolution and precision were achieved [17]. Thanks to potentially very small values of achievable resolution the quantization error in this method may be significantly reduced. Thus the main sources of measurement uncertainty are the nonlinearity of transfer function and signals jitter. Efficient auto-calibration and error reduction techniques have been developed [16–18] leading to reducing the measurement uncertainty below 10 ps. An important disadvantage of the time stretching method is a relatively long conversion time, at least equal to TK, which eliminates the method from use in fast TDCs.

7.3.1.2 Time-to-Voltage Converter Followed by Analogue-to-Digital Converter

The long conversion time typical for the analog time stretching is significantly reduced in another analog method, i.e. *time-to-voltage conversion followed by analogue-to-digital conversion*. This method is also based on linear charging of a capacitor (C in Fig. 7.4) within the measured time interval T. However, in the second step of conversion the resulting voltage at the capacitor is temporarily held

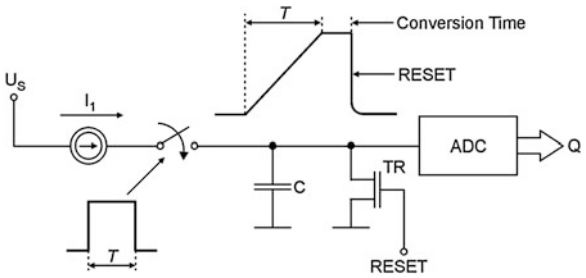

Fig. 7.4 TDC based on time-to-voltage conversion followed by analogue-to-digital conversion [3]

and digitized with the use of a conventional analog-to-digital converter (ADC). After that, the capacitor is discharged rapidly and thus the conversion time is virtually equal to the conversion time of the ADC used. The resolution q of the method depends on the assumed measurement range MR and the maximum number n of bits of the ADC in the following way: $q = MR/2^n$. So, to achieve better resolution in time the use of ADC with higher voltage resolution is needed.

Due to simplicity of the conversion principle and availability of integrated ADCs of high resolution and low power, this method is quite popular and is used in both unique scientific designs [19, 20] and commercial, series-produced TICs (SR620 *SRS*, 7186 *Phillips Scientific*, see Table 7.4). The highest achieved resolution and precision are at the level of single picoseconds and less, e.g. 0.1 and 1 ps, respectively [21]. Such high performance is obtained thanks to the use of Miller integrators that help to reduce the undesirable influence of parasitics on the nonlinearity of conversion. That was particularly important in this case because the device was built out of discrete components.

The performance of TDCs based on each of two analog methods described above is degraded by imperfections of all building blocks (pulse former, integrator, comparator or ADC), however, the precision is mainly limited by nonlinearity of the capacitor charging process. A detailed description of related problems may be found in [18–20].

7.3.2 Digital TDC Architectures

The important advantage of the digital methods over the analog ones and the main reason of their popularity is integrability in integrated circuits. The group of the most popular digital methods includes the method based on a single tapped delay line, Vernier delay line, Vernier principle with startable oscillators, and pulse shrinking.

7.3.2.1 Tapped Delay Line

The idea of the use of a *tapped delay line* for time digitizing is conceptually simple. In the basic version of a converter the tapped delay line may be created by serially connected non-inverting buffers with a unit propagation delay τ (Fig. 7.5). The flip-flops D detect and store the number m of delay buffers through which the pulse START has propagated until the STOP pulse appeared at the line input. The value of the measured time interval is then calculated as a product of the delay buffers number and the unit delay: $T = m\,\tau$. The measurement range MR of the converter is proportional to the length of the delay line: $MR = N\tau$.

The important merit of this method is that the conversion process is very fast and thus the conversion time is negligible due to a lack of any intermediate processing (direct time-to-digital conversion) and very short time needed by D

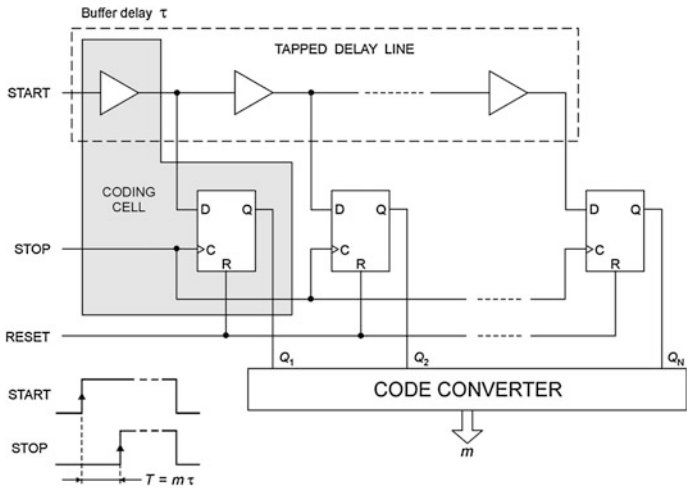

Fig. 7.5 TDC with a single tapped delay line

flip-flops to sample and store the state of the delay line (clock-to-output delay of a flip-flop). Also, the dead time is relatively short and equals the reset time of flip-flops. For these reasons such converters are also called *flash converters*.

As the resolution of the converter equals the buffer delay, then it strictly depends on the technology in which the converter was implemented. The first integrated TDCs based on this method were integrated in bipolar *Emitter Coupled Logic* (ECL) technology, which offers short intrinsic delays at the expense of large power consumption. There is also known design of TDC in GaAs technology, however, the most common are TDCs made in CMOS as *Application Specific Integrated Circuit* (ASIC) or FPGA devices. ASICs offer higher design flexibility resulting in the highest available resolution of delay line-based TDCs that reaches the level of about 10 ps [5, 9, 12]. In TDCs recently designed in CMOS FPGA devices, the multiplexers forming the fast carry chains are most often used as delay buffers [22, 23]. These multiplexers have the shortest propagation time among all logical elements available for this purpose on the FPGA chips. For example, in the programmable devices from the Spartan-6 family fabricated by *Xilinx*, the mean delay of such multiplexer is only about 20 ps. However, the delays of interconnections needed to create a delay line-based converter implemented in FPGA are strongly uneven, which causes the existence of ultra wide bins at the transfer characteristic. They limit the resolution of the converter and deteriorate its linearity. To effectively sub-divide those much wider bins, multiple coding during each conversion process in a single delay line structure is proposed [24]. The multiple coding is obtained by detecting in the delay line not a single, but several logic transitions (0-to-1 or 1-to-0) of a specially generated pulse train called the "*wave union*". Two types of the "wave union launchers", with the Finite Step Response (FSR) and the Infinite Step Response (ISR), were tested in several TDCs

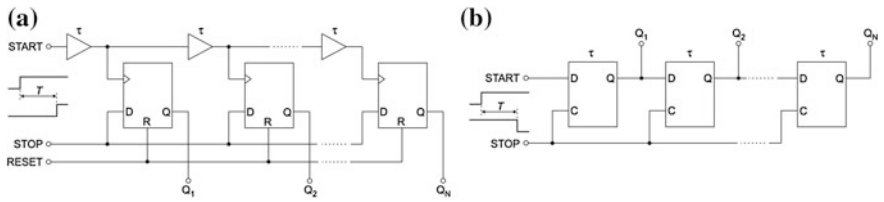

Fig. 7.6 Variants of TDC with a delay line created as a chain of buffers (**a**) and of latches (**b**)

implemented in the Cyclone II (*Altera*) device leading to an improvement in precision from 40 to 10 ps (rms value) [24]. A theoretical analysis of the wave union method and modified wave union launcher are presented in [25].

The delay line-based technique can be applied in different circuit configurations (Figs. 7.6). In the first variant (Fig. 7.6a) the pulses START and STOP are attached to the opposite inputs of flip-flops than in the version presented in Fig. 7.5. The tapped delay line is a source of the multiphase START signal that may be treated as a multiphase clock that consecutively samples the state of data inputs D of successive flip-flops. The state of the coding line is represented in the thermometric code and the measurement result is given by the number of the first flip-flop set in the line. The resolution is again equal to the delay time of a single buffer.

A simple way to double the resolution is the use of inverters instead of buffers [26, 27]. Such a solution needs two delay chains propagating the inverted and noninverted input signals and providing differential data to the fully symmetrical differential flip-flops, used as sampling elements. The resolution may further be improved by local passive interpolation based on the use of: (1) voltage dividers that create interleaved levels between signals in two inverter-based delay lines [28], or (2) an adjustable RC tapped delay line spanning the "length" of a single delay cell [11].

In the other version of such a converter (Fig. 7.6b) the delay line is formed as a chain of D latches. The START pulse propagates through the successive latches with increasing delay until the STOP pulse appears. The number of the first latch unset (remaining zeroed) is proportional to the measured time interval and its propagation time determines the converter's quantization step. This is the simplest variant of a converter with a tapped delay line which does not even require any reset signal. The drawback of this configuration is lower resolution (larger q value).

7.3.2.2 Vernier Delay Line

An increase in the resolution of a converter with a tapped delay line is possible by changing the technology to a newer one with a smaller feature size and consequently shorter propagation times or by making modifications in the structure of

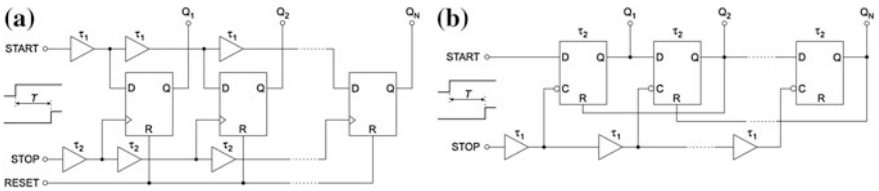

Fig. 7.7 Vernier delay line with two chains of buffers (**a**) and chains of buffers and latches (**b**)

the delay cell. The well-known modification consists of introducing a second tapped delay line for the STOP pulse (Fig. 7.7a). The cell delays in both delay lines are slightly different ($\tau_1 > \tau_2$) and their difference determines the resolution of the converter ($q = \tau_1 - \tau_2$). Such configuration of delay lines is known as the *Vernier delay line* (VDL) due to its similarity to the Vernier method described below in this chapter. The value of a measured time interval is calculated in the same manner as for a single tapped delay line (Fig. 7.5). The VDL in the form presented in Fig. 7.7a was used in TDC designed as ASIC for positron emission tomography (PET) imaging application and allowed to reach the resolution of 31 ps [29]. In another application, the VDL combined with dual PLL circuits provided an even higher resolution of 23 ps [8]. The solution based on VDL is also met in TDCs implemented in FPGA devices, where regular column architecture helps to create parallel delay lines of similar unit delays [22].

Figure 7.7b shows another version of such a converter in which the delay lines for the START and STOP signals are formed as chains of D latches and noninverting buffers, respectively. The quantization step is the difference of the time delay of the latch and the buffer. The digital coding of the measured time interval is realized by setting a single latch in the line. Thanks to the use of local feedback loops, the output data from the delay line is obtained directly in "1-out-of-N" code. The transparency feature of latches makes the separate reset signal obsolete. Such VDL was used in the first TDC integrated in an FPGA device (pASIC1, *QuickLogic*) where 200-ps resolution (LSB) and 10-ns measurement range were obtained [30, 31].

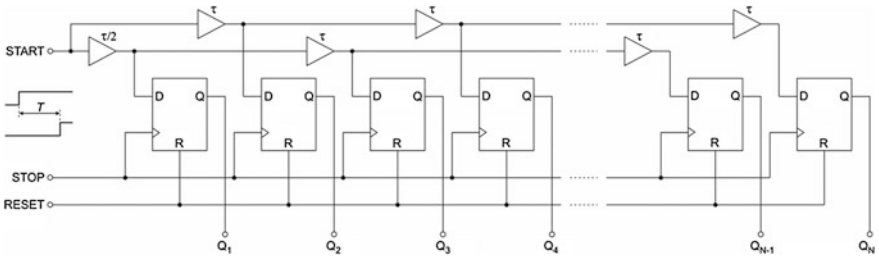

Fig. 7.8 Parallel tapped delay lines shifted to increase resolution

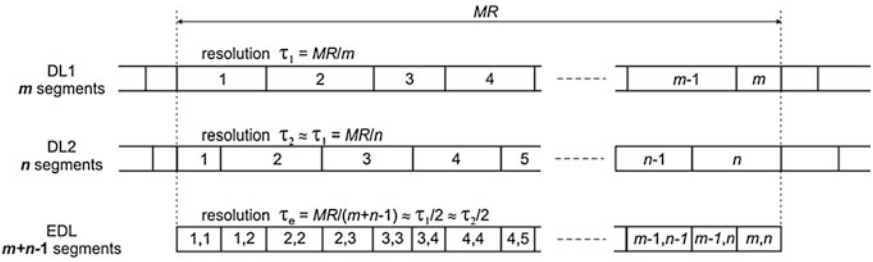

Fig. 7.9 Creation of the equivalent delay line [34]

7.3.2.3 Multiple Parallel Delay Lines

Another way to improve the TDC's resolution is the use of two identical tapped delay lines shifted one to another by half of a unit delay ($\tau/2$) instead of a single delay line (Fig. 7.8) [32].

In FPGA devices, where this approach was applied, a designer cannot introduce very short delays (tens of picoseconds) of precisely controlled value and this method may only be used if the expected resolution is at the level of hundreds of picoseconds [32]. This inconvenience is overcome by the use of two delay lines (DL1 and DL2 in Fig. 7.9) operating independently, whose transfer functions are used to create a single transfer function of a virtual equivalent delay line (EDL) [33, 34]. To create an EDL the transfer functions of both DLs have to be identified. It is usually performed with the aid of the statistical *code density test* (CDT) [18, 35], by precise identification of the quantization steps (m and n) of each DL within a measurement range MR. Then, the transfer function of EDL is calculated and stored either by the control software [33] or the hardware code processor [34].

The main advantages of this approach are: (1) the resolution $\tau_e \approx \tau_1/2 \approx \tau_2/2$ is improved approximately twice as much in comparison to τ_1 (τ_2) of a single TCL, (2) the time offset between the lines is not significant and does not have to be controlled in a design, and (3) the idea may further be expanded to create EDL with more than two DLs. Using the expanded version of the method with sixteen DLs in the interpolator of a TIC implemented in Spartan-6 device (*Xilinx*) [34], almost thirty eight times better resolution (1.14 ps) and twelve times better precision (6 ps) were achieved in comparison with a similar TIC but operating with a single tapped delay line in interpolator (45 and 70 ps respectively) [23].

7.3.2.4 Parallel Scaled Delay Line

A way of overcoming the technological limit in the unit delay of a tapped delay line is the use of the *parallel scaled delay line* (PSDL, Fig. 7.10a) [12, 36]. Delay elements forming this line are connected in parallel, while delay scaling is obtained by scaling the load capacitance of successive elements. An alternative way to provide the delay difference is to use scaled current starving transistors to

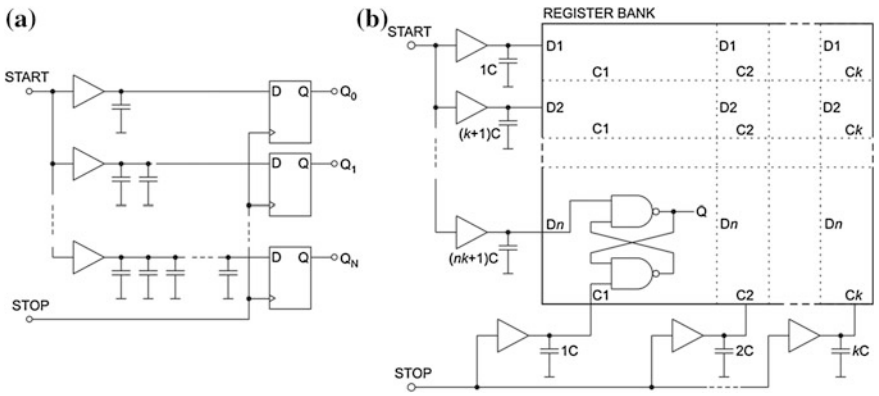

Fig. 7.10 Parallel scaled delay line (**a**) and scaled delay matrix (**b**)

control the pull-up and pull-down strength of the delay elements. The resolution of PSDL equals the difference between the propagation times of neighbor delay elements. The undeniable merit of such delay structures is that the differential nonlinearity does not accumulate as in cascaded structures. Moreover, making the load capacitances controllable by coupling successive capacitors on or off, the transfer characteristic of the line may be linearized in a significant degree. However, it complicates the approach due to the necessity of use of a large number (even hundreds) of small-value capacitors and related control circuits [37]. This method used in TDC designed in 40 nm CMOS technology for digital PLL allowed to reach 5.5 ps resolution and 4.4 ps single shot precision [5].

Two PSDLs with different numbers of delay elements (k and n) and delay scaling factors (1 and k) are used to create a high resolution matrix of input signals' phases (START and STOP) (Fig. 7.10b). Detecting the coincidence of resulting phases, the arbiter register bank, containing $k \times n$ registers, determines the original time relation between input signals. Such a solution, described in detail in [37], was recently used in the second interpolation stage of two-stage TIC where the high precision (8.1 ps) and wide measurement range (202 μs) were obtained [12].

7.3.2.5 Pulse Shrinking Delay Line

The inherent feature of the pulse transmission through a delay line is the pulse shrinking or stretching due to different propagation times of integrated delay elements (e.g. buffers) for opposite edges of the pulse ($t_{PLH} \neq t_{PHL}$). If information about the measured time interval is contained in the width of a pulse this feature may be useful for time digitizing. In the first implementation of the method in an integrated circuit, the *pulse shrinking* delay line was created as a cascade of simple buffers, each containing two serially connected inverters with a controlled

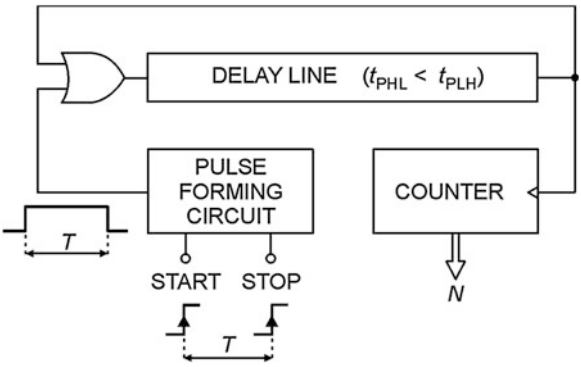

Fig. 7.11 TDC based on cyclic pulse shrinking

propagation time of the rising edge of a pulse [10]. As a result of thermal and supply noise a significant timing jitter was observed limiting the maximum obtained resolution to about 100 ps.

Due to technological spread and layout mismatches among the pulse-shrinking delay elements, the long line is usually characterized with the large linearity error of conversion. Closing the shorter delay line into a loop and using it cyclically is the proposed solution of the problem [38–40]. A conceptual block diagram of the converter based on the *cyclic pulse shrinking* is shown in Fig. 7.11. The measured time interval T is represented at the input of the pulse shrinking loop by the pulse width. Propagation times of the OR gate for both pulse edges have to be equal ($t_{PLH} = t_{PHL}$), while these times for the entire delay line should be different ($t_{PLH} > t_{PHL}$). Thus, after each cycle, the width time of a pulse circulating in the loop is reduced by the constant value equal to difference of $t_{PLH} - t_{PHL}$. The pulse circulates until it vanishes and the total number of cycles N is counted by the counter. The original width of the pulse is calculated as $T = Nq$, where the resolution is $q = (t_{PLH} - t_{PHL})$. The measurement range is roughly proportional to the number K of buffers in the delay line and the unit propagation time τ ($MR \approx K\tau$).

In ASICs, noninverting buffers or inverters are used as delay elements to create a pulse shrinking delay line. The pulse shrinking time per delay element is

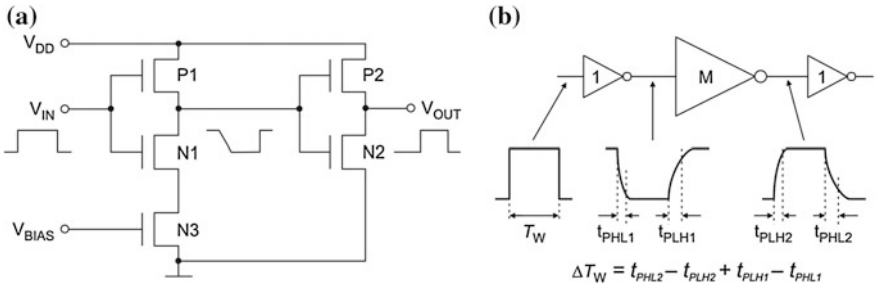

Fig. 7.12 Pulse shrinking delay element with adjusted bias voltage (**a**) and dimension-controlled pulse-shrinking delay element (**b**) [39]

typically varied by adjusting: the bias voltage of the current-starving series transistor (N3 in Fig. 7.12a) [38], the supply voltage of the delay element, or the dimension ratio of the adjacent gates (Fig. 7.12b) [39].

The delay line may include only one pulse shrinking delay element ($t_{PLH} > t_{PHL}$), while other "transparent" delay elements ($t_{PLH} = t_{PHL}$) are necessary to get a reasonably wide measurement range. Assuming, in the ideal case, that the pulse-shrinking time of the element is kept constant from cycle to cycle, the cyclic TDC has no nonlinearity problem. However, the important drawback of the TDC is the accumulation of jitter of a pulse circulating in the delay loop. Moreover, to achieve high resolution and precision, V_{BIAS} must be adjusted carefully during repeated calibrations. The inconvenience of frequent calibrations is remedied by the solution shown in Fig. 7.12b, which does not need any bias adjustment for pulse-shrinking control. The dimensions of the first and third inverters are the same (1). Only that of the second inverter is different (M > 1). The inhomogeneous dimension of the delay elements makes the input pulse undergo different rising and falling times at the interface boundaries between the inverters, and this mechanism is used to accurately control the pulse width.

The number of cycles in the loop may be reduced by increasing the pulse-shrinking time per cycle. In order not to deteriorate the resolution, a *multi-stage pulse shrinking* delay line may be used (Fig. 7.13) [39]. Each pulse shrinking delay element of the line consists of three inverters because the first or unit inverter of the next element is also shared with the previous one. Since each inverter couple reduces the circulating pulse by q, the pulse disappears after a certain number $m = Nk + n$ of inverter couples, where N is the content of counter, k is the total number of inverter couples in the line, and $n < k$ is the number of inverter couples

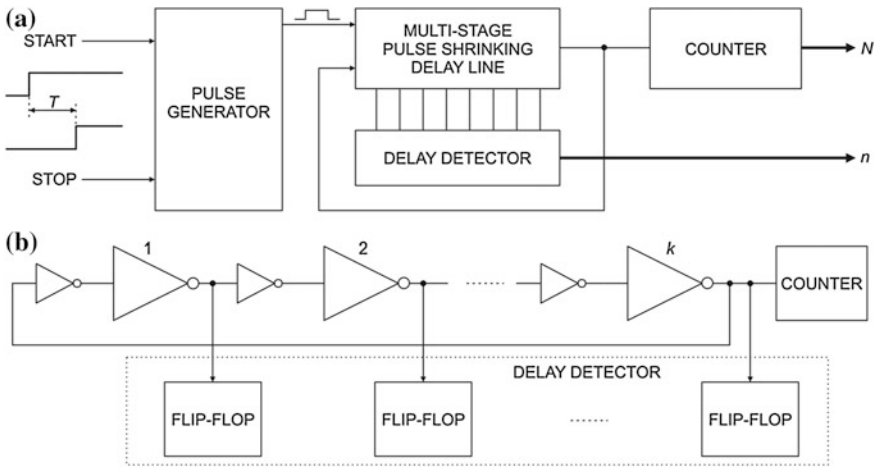

Fig. 7.13 Block diagram of the TDC (**a**) based on the multi-stage pulse shrinking delay line (**b**) [39]

in the line after which the circulating pulse has disappeared. The measured time interval is calculated as $T = mq$.

In FPGA devices, due to a lack of convenient means for adjusting the pulse-shrinking time per delay element, the other way of controlling the pulse shrinking per cycle is proposed [40]. The pulse shrinking is performed in a loop containing two complementary delay lines. The first line shrinks and the second one stretches the duration time of a circulating pulse, hence the length ratio of the lines determines the pulse shrinking capability of the converter. In addition, two forms of representation of a measured time interval (the pulse width and the time interval between two pulses) are utilized to improve the efficiency of resolution control. Moreover, to diminish the jitter of edges of a circulating pulse and consequently to increase the precision of the converter a two-stage conversion is introduced. The first stage having a low resolution shortens the measured pulse rapidly thus limiting the number of cycles, while the second one provides a final, high resolution within a narrow time interval range.

7.3.2.6 Vernier Oscillators (Digital Time Stretcher)

The next digital technique for time digitizing is the *Vernier method*, known also as *digital time stretching* (or *expansion*). This method consists of using two startable oscillators SO_ST and SO_SP triggered consecutively by active edges of the input pulses START and STOP (Fig. 7.14). After activation, oscillators produce signals of known and slightly different frequencies, $f_{ST} < f_{SP}$. Periods of signals are counted by respective counters until the coincidence of signal edges is detected and oscillators are disabled. The value of measured time interval is calculated as $T_M = N_{ST}\ T_{ST} - N_{SP}\ T_{SP}$, where N_{ST} and N_{SP} are the numbers of the counted periods $T_{ST} = 1/f_{ST}$ and $T_{SP} = 1/f_{SP}$, respectively. The resolution is equal to the difference between periods ($q = T_{ST} - T_{SP}$). As the frequencies of oscillators may be freely adjusted, this method creates a potential possibility to reach a theoretically unlimited high resolution.

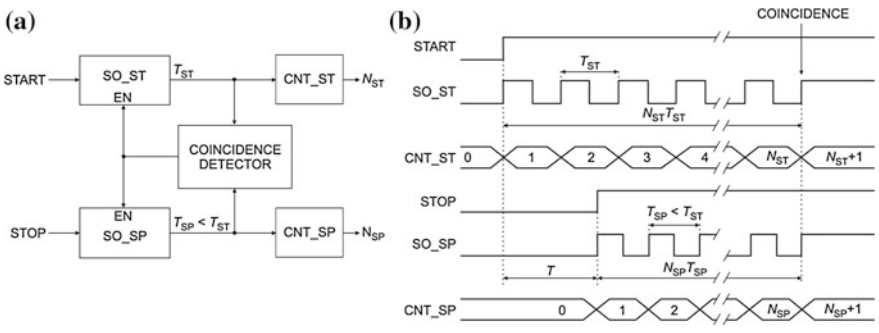

Fig. 7.14 TDC based on the Vernier method (**a**) and example waveforms (**b**)

If the required measurement range of the converter is narrower than the period of signal from the first oscillator, the final contents of both counters are equal to one another ($N_{ST} = N_{SP} = N$) after conversion. Therefore, the measurement result is calculated from the simpler formula $T_M = N(T_{ST} - T_{SP}) = Nq$ and the structure of the converter is simplified by removing one counter.

The conversion time, lasting from the appearance of a STOP pulse at the input of the converter to the detection of the coincidence, is proportional to the difference of the signals' phases when the second oscillator is triggered and inversely proportional to the quantization step. The maximum value of the conversion time T_{Cmax} is calculated as $T_{Cmax} = (T_{ST} \times T_{SP})/q$. For high resolution TDCs the conversion process may last relatively long (reaching μs), which is an important drawback of the method.

First converters based on the Vernier method with startable oscillators were built with the use of the small-scale-integration ECL integrated circuits, in the 1970s. To achieve high resolution, a careful electromagnetic shielding of the generators and a stable operating temperature after many hours of warming up were absolutely necessary. In modern converters the Vernier method in its original form (with oscillators) is rarely used. The TDC designed in 0.13 μm CMOS technology and described in [41] is one of the few. It is based on two startable ring oscillators in which oscillation frequencies are stabilized by related PLLs. Such an approach provides the resolution of 37.5 ps without any calibration or external bias adjustment. A novel Vernier ring structure obtained by connecting the outputs of a Vernier delay line to its inputs was used to achieve a high resolution of 8 ps [42]. Also two- [43] and three-dimensional [44] Vernier ring structures were developed to reduce the number of delay stages and power consumption, and to enlarge the measurement range without compromising the resolution.

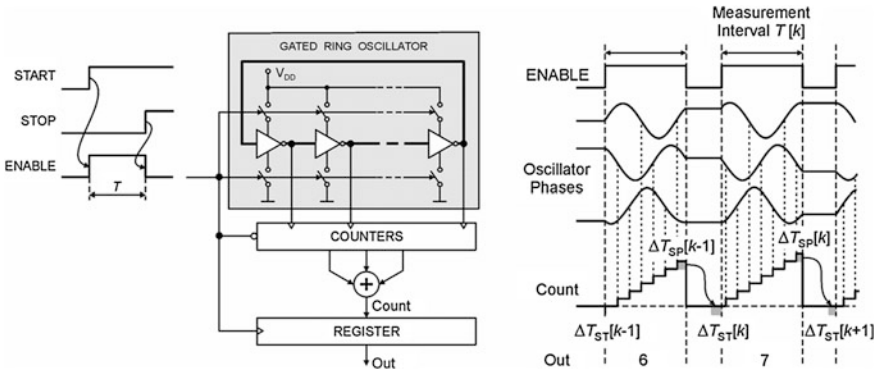

Fig. 7.15 Principle of TDC based on gated ring oscillator [46]

7.3.2.7 Gated Ring Oscillator

In the delay line-based TDCs the measured time interval is quantized with a quantization step equal to the propagation time τ of a single delay buffer (Fig. 7.5). After each measurement, as the START pulse propagates through the delay line despite the appearance of the STOP pulse, the information about the quantization error ΔT ($0 \leq \Delta T < \tau$) is irretrievably lost. Such information could be helpful to improve the resolution through noise shaping [2]. To memorize the value of quantization error the temporary state of delay line has to be "frozen" just after the sampling by the STOP pulse. This idea is applied in the TDC with a *gated ring oscillator* (GRO) [45, 46]. The principle of such a converter is shown in Fig. 7.15.

The ring oscillator is enabled only during the measured time interval T (ENABLE = 1) and disabled otherwise. All positive and negative transitions of all phases of GRO are counted by a set of counters. The total content of counters (OUT) is proportional to the value of T. After each measurement, an internal state of GRO (including transition states of some phases) is held and the starting point for the next measurement corresponds exactly to the stopping point of the previous one. In this way the quantization error $\Delta T_{SP}[k-1]$, which occurs at the end of a $k-1$ measurement, is transferred to the next measurement as an initial value $\Delta T_{ST}[k]$. The overall quantization error $\varepsilon[k]$ for the k measurement is the difference of two residual values $\varepsilon[k] = \Delta T_{SP}[k] - \Delta T_{ST}[k] = \Delta T_{SP}[k] - \Delta T_{SP}[k-1]$. Due to this difference operation on ΔT the quantization error is first-order noise shaped and the effective resolution is reduced well below an inverter delay [2, 45, 46].

Further improvement in the performance of the GRO-based TDC can be achieved by using a multipath technique. A *multipath gated ring oscillator* contains delay stages with a single output and several inputs (two inputs in the example in Fig. 7.16) [45]. A combination of output signals from previous stages

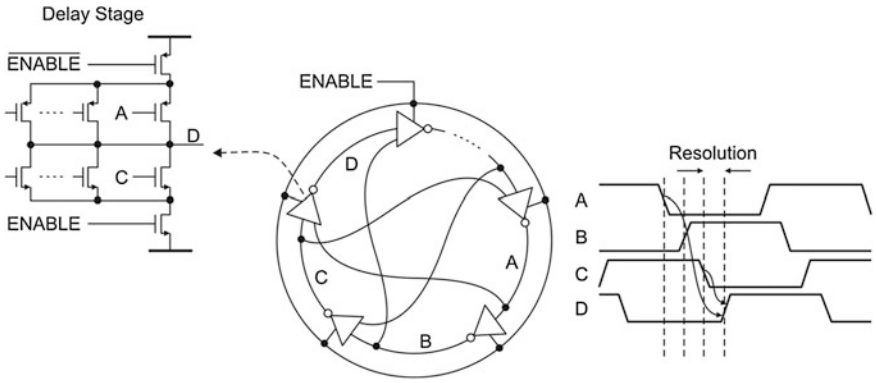

Fig. 7.16 Concept of the multipath gated ring oscillator [45]

(A and C) is used to speed up the transition time of a given stage (D). The implementation of this technique requires changing the structure of the basic inverter delay typically used in ring oscillators. Several means of accomplishing this are described in [46]. The use of the multipath technique in a TDC manufactured in 0.13 μm CMOS process allowed to reduce the delay per stage from 35 to 6 ps [45].

7.3.2.8 Successive Approximation TDC

The principle of successive approximation, well-known as an A/D conversion method, was also adopted for use in TDC [47]. The simplified block diagram of the TDC and conceptual timing diagram of the *cyclic time domain successive approximation* (CTDSA) are presented in Fig. 7.17.

Two pulses (P1 and P2) representing the beginning and end of the measured time interval circulate in separate loops, in which delays are successively adjusted to cause the coincidence of the pulses. According to the successive approximation algorithm, the delay in the loop of the earlier pulse is first increased by half the TDC measurement range (MR/2), and then the amount of the delay adjustment is halved in each following cycle (MR/4, MR/8, and so on). The time relation

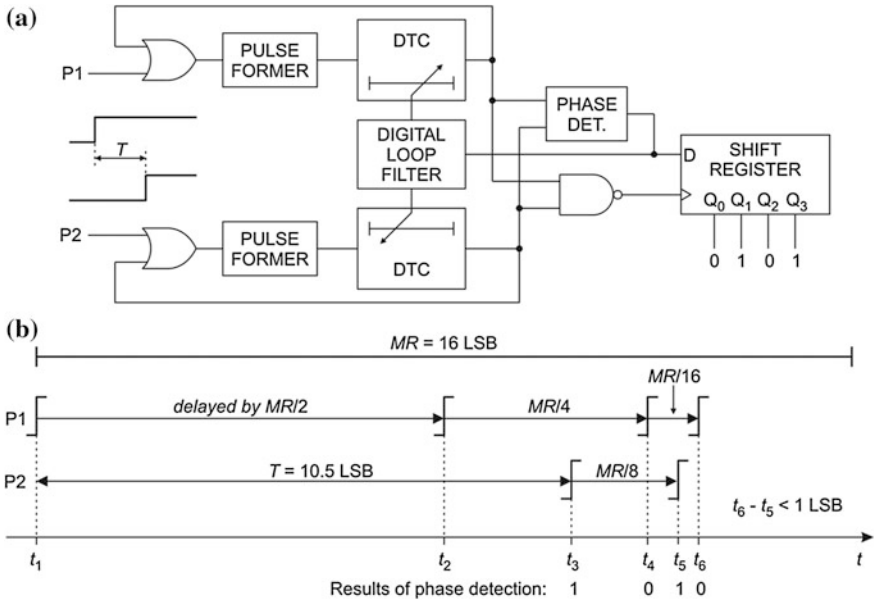

Fig. 7.17 Simplified block diagram of a 4-bit TDC based on CTDSA (**a**) and related timing diagram (**b**)

between pulses is verified by the phase detector after each cycle and the result of the verification is stored in the shift register. This is also the base for decision making of which loop delay should be increased. Delay adjustment is performed with the use of the appropriate digital-to-time converter (DTC) controlled by the digital loop filter. At the end of the conversion both pulses are accurately aligned within one LSB ($t_6 - t_5 < 1\text{LSB}$). The content of the shift register represents the conversion result in the binary format. Figure 7.17b presents the actions required in 4-bit TDC to convert an example time interval of 10.5 LSB [47].

7.3.2.9 Time Amplifier

Recently, the two-stage coarse-fine TDCs were developed. They, being followers of coarse-fine ADCs, improve the resolution by amplifying in the second stage the time residue existing at the output of the first stage. The time amplification principle is based on the use of a metastability effect in SR latches [48] or a dynamic change of delay of variable delay cells contained in two differential delay lines [49]. The proposed approaches led to high resolution of conversion that equals 1.25 and 9 ps, respectively. Both converters also offer high precision ($\sigma = 1.25$ and 2.37 ps), but the measurement range of the former one is narrow (0.64 ns) and cannot be easily extended due to the narrow range of metastable behavior of contemporary integrated latches.

7.4 Interpolation Method Based Time Interval Counters

A high resolution measurement of long time intervals requires at least a two stage approach, i.e. a coarse quantization of the long time interval and a fine quantization of the remainder. It is provided by the classic Nutt method [1, 3], which combines a simple counter method and a precise T/D conversion of short time intervals within a single clock period (ΔT_1 and ΔT_2 in Fig. 7.2). Since the calculation of values of the short time intervals correspond to the interpolation process, i.e. an estimation of a function value within the range of a discrete set of known data points (in that case defined by multiple of clock periods), such a method is known as the interpolation method. Instruments based on this method usually contain, among others, two narrow range TDCs and naturally are more complicated than each single TDC described in Sect. 7.3. This is another reason to differentiate them from the TDCs, and the name TICs is used for them in this chapter.

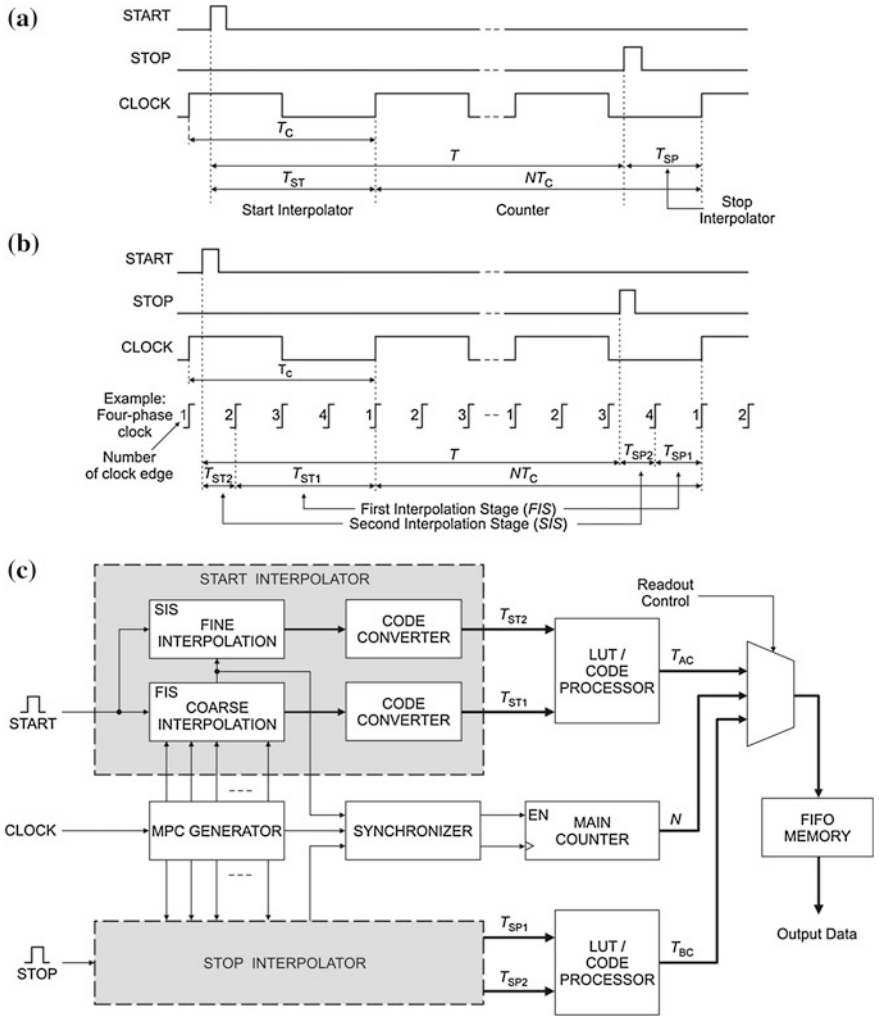

Fig. 7.18 Timing diagrams of the single-stage (**a**) and two-stage time interpolation (**b**), and the simplified block diagram of TIC with two-stage interpolation (c)

7.4.1 Single-Stage and Two-Stage Interpolation

According to the Nutt proposal the *single-stage interpolation* involves splitting the measured time interval into three parts, the first being the integer number of the reference clock periods, and two short intervals (at the initial and final part of the measured interval), each having the duration within one clock period (Fig. 7.18a). The first part (NT_C) is measured by the binary counter with a course resolution of T_C, while the remaining two parts (T_{ST} and T_{SP}) are measured with the aid of two

high resolution TDC's with fine quantization steps q_{ST} and $q_{SP} \approx q_{ST} \ll T_C$, respectively. Neglecting the conversion errors of the TDCs, the measured time interval is given by $T = N\,T_C + (T_{ST} - T_{SP})$.

The single-stage interpolation is often based on the tapped delay lines [10, 26]. The delay of a single cell of the line determines both the resolution of the digitizer and the length of the line at a given clock frequency. For example, at a 100 ps resolution and a 100 MHz clock the delay line should have at least 100 cells. It is almost impossible to obtain uniform cell delays in such a long line and this is the main cause of the linearity error of conversion. The line is also very sensitive to the temperature and supply voltage variations. Obviously, such a problem with the nonlinearity of conversion concerns also other methods that may be used. Therefore, the designers try to shorten the range of highly precise interpolation by increasing the clock frequency and/or using the multiphase clock (MPC). In the latter case a two-stage interpolation method is used [3, 12, 23].

In the *two-stage interpolation* method the value of the measured time interval T is evaluated through the estimation of values of five shorter time intervals (Fig. 7.18b). The time interval $N\,T_C$, similarly like in the single-stage interpolation, consists of an integer number N of such clock periods T_C, whose leading pulse edges appear between the leading edges of the START and STOP pulses. The number N is counted by the binary counter which is often called the *main counter* (Fig. 7.18c), in order to distinguish from other counters that have to be used in TICs based on other conversion methods. For example, the use of the analog time stretching for interpolation needs two additional counters [16]. The time interval between the START pulse and the nearest edge of the reference clock is simultaneously measured by two stages of interpolation in the START channel. In the first interpolation stage (FIS) a multiphase clock (MPC) is used (a four-phase clock in the example in Fig. 7.18b). After the START signal appears at the TIC input the nearest edge of the MPC is detected. Since the widths of the MPC time segments are known from calibration, the time T_{ST1} can be calculated accurately. In the second interpolation stage (SIS) the time interval T_{ST2} between the START pulse and the nearest edge of the MPC is precisely measured with the aid of a high resolution TDC with a narrow range. The time intervals T_{SP1} and T_{SP2}, related to the STOP pulse, are determined in a similar way by two respective stages of interpolation in the STOP channel. The value of the measured time interval T is obtained by combining the five terms: $T = N\,T_C + (T_{ST1} + T_{ST2}) - (T_{SP1} + T_{SP2})$.

Figure 7.18c shows the simplified block diagram of the TIC with two-stage interpolation. It contains the MPC generator, the main counter with synchronized enable input, two two-stage interpolators for the START and STOP inputs, two look-up tables (LUT) or code processors for calibration and correction of the interpolator transfer characteristics, and the FIFO memory for fast storage of the raw data from the measured samples (millions measurements per second [23]). The LUTs are primarily used for nonlinearity correction [12, 37, 56], while the more advanced hardware code processors can additionally calculate the corrected conversion result of the whole interpolator [23]. If the corrected output data (T_{AC}

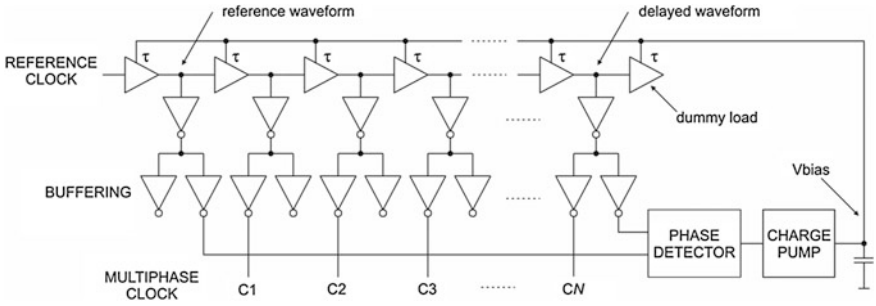

Fig. 7.19 Example circuit diagram of the DLL [37]

and T_{BC}) is normalized to T_C, the measurement result is calculated simply as $T = (N + T_{AC} - T_{BC})\ T_C$.

7.4.2 Multiphase Clock Generation

The most common way to generate the multiphase clock for the FIS of TICs implemented in ASICs is the use of a stabilized delay line configured as a delay-locked loop (DLL) (Fig. 7.19) [11, 12, 36, 37]. Such a circuit configuration makes the parameters of MPC virtually independent of process, temperature and voltage variations. The basic building block of a delay line is an adjustable delay buffer that typically consists of two identical current starved inverters (Fig. 7.12a). A DLL compares the phases of a clock signal from the line output and the reference clock applied to its input (next period in fact). The difference is detected, filtered, amplified, and used to adjust the buffer delay to meet the requirement $N\tau = 1/f_C$, where N is the number of buffers in the "active" part of the delay line, τ is a buffer delay, and f_C is a reference clock frequency. In this way, the N-phase clock of evenly delayed phases is generated. Simplified method of monitoring the total delay of delay line is described in [57].

A stabilized delay line can also be configured as a ring oscillator and utilized in a phase-locked loop (PLL) [41]. In both DLL and PLL loops an off-chip oscillator is needed as a source of reference signal. Such a solution has two important disadvantages, namely that the external reference cannot be integrated and typically consumes a lot of power. An interesting approach is the use of a ring oscillator whose frequency is stabilized indirectly by converting the frequency to voltage and locking the obtained value to a stable on-chip voltage reference [50]. Thus, the necessity of the off-chip reference is eliminated.

In contrast to ASICs, FPGA devices do not allow the integration of self-designed DLL due to logic-resource limitations and the determined supply voltage distribution net. To stabilize the internal delays of the FPGA chip, the supply

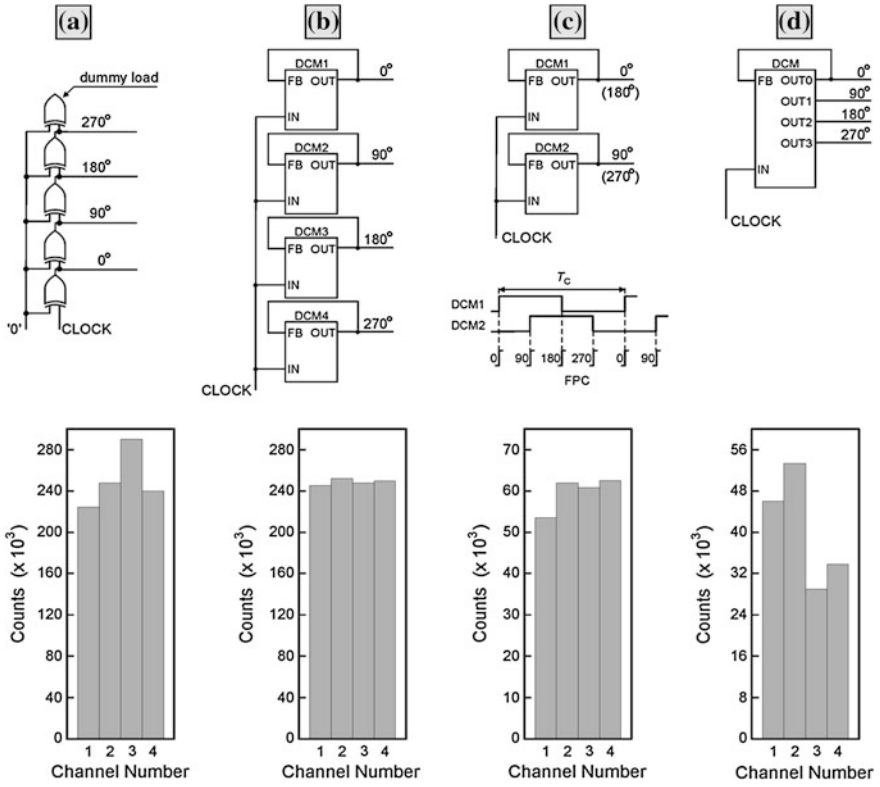

Fig. 7.20 Methods of FPC generation in FPGA device and related histograms

voltage of the whole chip has to be controlled. Hence, the DLL needs external components, at least a loop filter and a power amplifier [51].

The modern FPGAs contain specialized units for clock synthesis. For example, such units contained in programmable devices manufactured by *Xilinx* are called DCMs (Digital Clock Manager) and can operate as frequency synthesizers, digital phase shifters and DLLs. The units may then be used for generation of MPCs [22, 23]. Different approaches with regard to the controllability and complexity of a four-phase clock (FPC) generator are shown in Fig. 7.20. In the first approach, the tapped delay line formed as a chain of logic gates (XOR gates in Fig. 7.20a) is used, while in the next approaches a DCM feature that allows shifting the phase of an output signal by a fixed fraction of its period is utilized.

The histograms shown in Fig. 7.20, obtained as results of the CDT [18, 35], illustrate the phase uniformity of FPCs of 250 MHz frequency generated in FPGA device (Spartan-3, *Xilinx*). The second solution (Fig. 7.20b) provides the best phase uniformity (the maximum difference <2.4 %), at the expense of a relatively long trial and error process of phase adjustment.

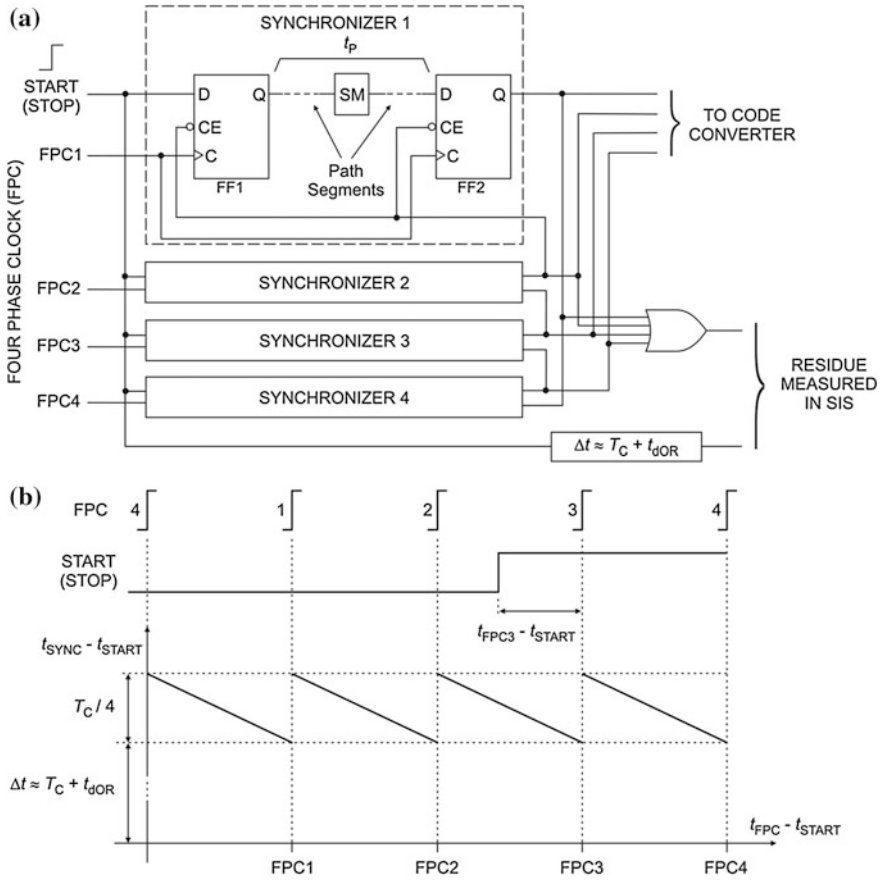

Fig. 7.21 Set of double synchronizers in FIS (**a**) and the output difference signal (**b**)

It may be pointed out that high phase uniformity of the MPC is an important feature which influences the linearity of a conversion. However, reaching the ideal uniformity is not absolutely necessary if an accurate enough calibration procedure is provided. The other parameters of the MPC, i.e. frequency and number of phases, have also to be carefully considered because they determine the power dissipation and the measurement range of SIS, respectively.

7.4.3 Synchronization in the First Interpolation Stage

The START (STOP) pulse appears at the input of FIS (Fig. 7.18c) randomly with regard to the MPC and due to the principle of FIS operation it has to be synchronized to the nearest edge of the clock. A single D flip-flop (FF) can be used as

the simplest synchronizer. However, if a change of the START (STOP) signal occurs within a narrow time window around the clock edge the *metastability effect* may be observed at the FF output. The frequency of its occurrence, among others, is proportional to frequencies of the clock and input signals. If both frequencies are high, the metastability effect appears frequently enough to lower the precision of TIC. To diminish the influence of the metastability effect on the behaviour of TICs, especially based on MPCs, more complex synchronizing circuits are applied. The designs are based either on a set of double synchronizers [23] or multi-bit registers [36, 37].

The *double synchronizer* is a two-bit shift register (the first synchronizer in Fig. 7.21a), which allows a decrease in the frequency of metastable states compared to a single DFF in the ratio of $e^{-t_R\tau}$, where τ is the metastability time constant and t_R is the resolution time or time needed for an FF to recover from a metastable state. The minimum value of the resolution time t_{Rmin} can be estimated indirectly, through the evaluation of the *mean time between failures* (MTBF) due to metastability [52]: $t_{Rmin} = \tau \ln(\text{MTBF}\ f_I\ f_C\ W)$, where W is the width of a metastability window or a narrow time window around the active edge of the clock signal, f_I and f_C are the frequencies of the input signal (START or STOP) and clock, respectively. The maximum operating frequency for which the required value of MTBF is still met can be calculated from the following equation: $f_{Cmax} = 1/(t_{CQ} + t_P + t_{SU} + t_R)$, where t_{CQ} is the nominal propagation time of FF1 from the clock (C) input to the Q output, t_{SU} is the setup time of FF2 and t_P is the interconnection delay. The last parameter (t_P) may be neglected in ASICs, however, in FPGAs the interconnection contains at least one programmable switch matrix (SM) and two path segments, whose total delay t_P is comparable with the

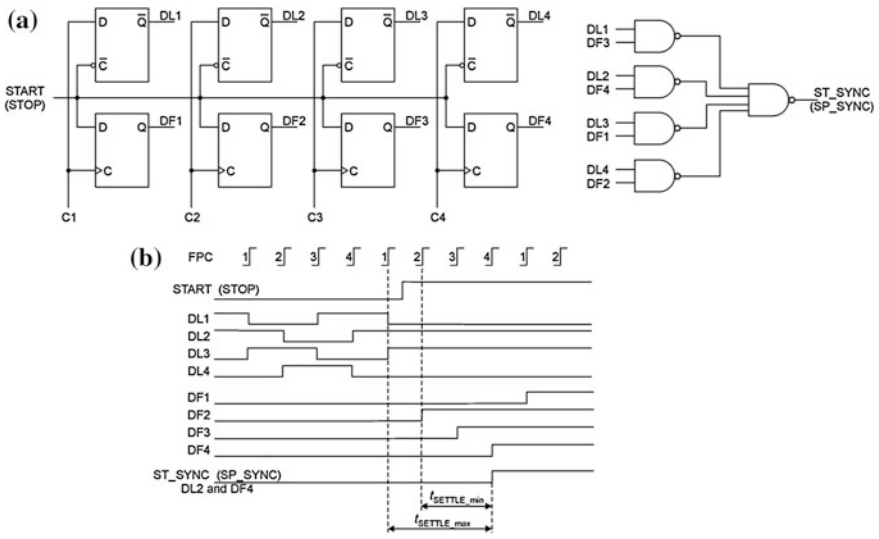

Fig. 7.22 Synchronizer with two registers (**a**) and related waveforms (**b**)

propagation time of a simple logic gate or FF. That is why it should be taken into consideration in precise timing analysis.

Precise synchronization in FIS requires the use of an individual double synchronizer per phase (four in FIS with a four-phase clock, Fig. 7.21a). The input signal is synchronized to the next edge of FPC and delayed by one clock period (double synchronization), and then transmitted via the OR gate to the SIS. The original input signal is also transmitted to the SIS, but to keep the primal time relation between this signal and the FPC, the input signal is delayed by T_C plus delay of the OR gate (t_{dOR}). In this way the set of synchronizers generates the output difference signal that has the cycle and the amplitude of the delay between phases of a MPC ($T_C/4$) (Fig. 7.21b).

Other synchronizing circuit for FIS is based on two registers and an AND-OR net [36]. The principle of this synchronizing method is explained in the example where the FPC is used (Fig. 7.22). When an input pulse (START or STOP) appears, the FPC state is stored in the first latch register. Then the data from the register is used to select the FPC edge far enough from the input pulse for reliable synchronization of the pulse in the second FF register. In the case of using FPC, the furthest possible clock edge is the 3rd edge after the arrival of the input pulse. It means that this synchronization principle provides almost half of the clock period for latches to settle their state ($¾\ T_C - t_{SU} \geq t_{SETTLE} > ½\ T_C - t_{SU}$, where t_{SU} is a setup time of FF). In the modified version of this synchronizer, the selection of the clock edge for safe synchronization in the second register is performed without the AND-OR gates [37].

7.4.4 Synchronization of the Main Counter

Since the frequency of the clock in modern TICs is relatively high (from tens of MHz up to hundreds of MHz) and both the beginning and end of the measured time interval are asynchronous with respect to the clock, the main counter must be carefully synchronized. Otherwise, the counting process would be affected by the metastability effect that would deteriorate the precision of the time interval

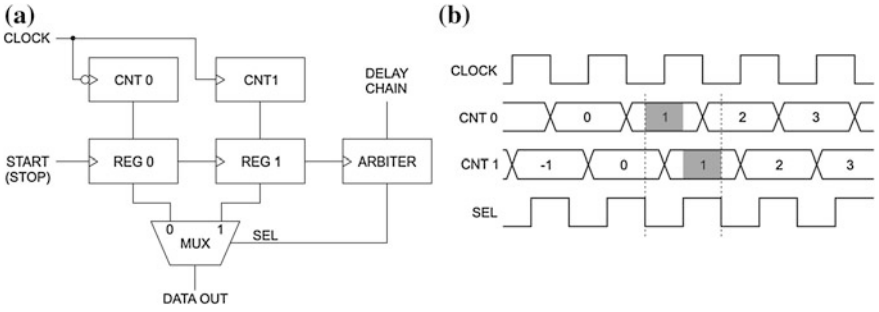

Fig. 7.23 Block (**a**) and timing (**b**) diagrams of a TIC with two main counters [53]

measurement. To achieve an undisturbed operation of a TIC, two solutions are commonly used: either two main counters fed with two opposite phases of the clock [39, 53] or a single counter with an additional synchronizer [23, 36, 37]. The former solution increases complexity and power consumption of a design and needs double the area on the chip for the counters, corresponding registers and an arbiter circuit (Fig. 7.23a).

Each of the main counters (CNT0 and CNT1) is incremented by the opposite edge of the clock. When an input signal (START or STOP) arrives the contents of both counters are stored in corresponding registers. The selection signal (SEL) created in the arbiter circuit is shifted with regard to the reference clock that ensures the chosen register is not a subject to violation of setup or hold times.

The latter solution providing errorless operation of a TIC, utilizes a single counter with an additional synchronizing circuit whose complexity depends on the interpolation method and the clock frequency used in a TIC. In the TICs with single-stage interpolation a simple synchronizer based on two D flip-flops (one per channel) and a single XOR gate may be used to control the counter. Flip-flops synchronize the input signals START and STOP to a clock, while XOR gate produces an enabling gate (START $\oplus$ STOP, both signals after synchronization). In the TICs with an MPC and two or more stages of interpolation the synchronizing circuit typically utilizes a single-edge or dual-edge double synchronization principle [23, 37].

In the *dual-edge double synchronizer* the input signal START (STOP) is synchronized to both rising and falling edges of the reference clock by two flip-flops (FF2 and FF3 in Fig. 7.24) [37]. So, if the proper time relations between signals at inputs of one of them are violated, the relations have to be appropriate at inputs of the second one, and no metastable behaviour will be observed at its output. The choice of the right synchronized signal is performed by the multiplexer and based on the state of the signal SEL from FIS. SEL carries information whether the input

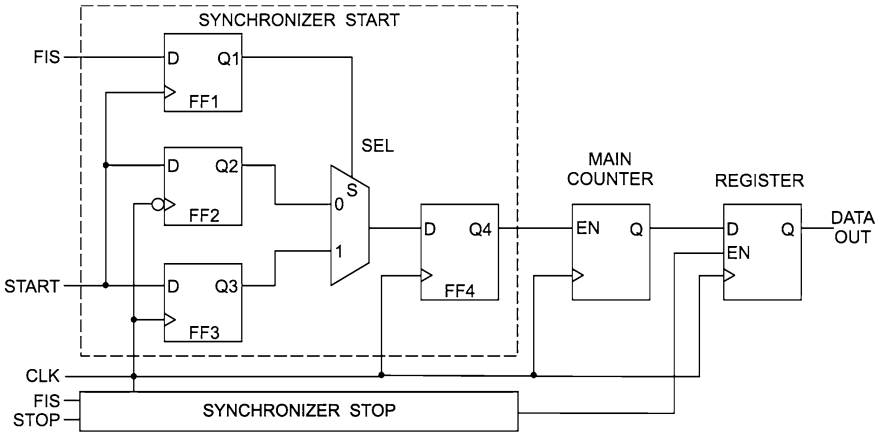

Fig. 7.24 Dual-edge double synchronizer of the main counter

signal appeared during the positive or negative half of the cycle of the reference clock period and whether it is safer to synchronize to the rising or falling edge of the clock. The selected signal is then synchronized again to the next edge of the clock (in FF4) and used to create an enabling signal for the main counter. The second synchronization is safe because it is performed at least half of the clock period after the first one.

For effective control of the counting process, two synchronizers are needed for the START and STOP signals. In the solution presented in Fig. 7.24 [37] the signal produced by the START synchronizer enables the main counter to start counting, while the signal from the STOP synchronizer enables the register for synchronous loading of the final state of the counter. If the main counter may be stopped after the first STOP signal has appeared, one can resign from the register using signals from synchronizers to create an unambiguous enabling gate for the main counter.

The maximum operating frequency of the dual-edge double synchronizer implemented in FPGAs is significantly limited due to longer delays of interconnections between flip-flops (particularly between FF2 and FF4). The frequency is almost doubled in the modified version of the synchronizer [23].

In FPGA devices, hard adjustable paths' delays cause that the synchronizers described above may not provide an undisturbed operation of the main counter, especially if a high frequency clock (hundreds of MHz) is utilized. This problem can be effectively solved with the use of a built-in functional unit such as, for example, DCM in *Xilinx's* devices. A simple solution involves the use of the DCM unit as a fixed phase shifter [23]. A more advanced approach is based on the *auto-tuned synchronization* principle which involves the DCM unit operating in the dynamic phase adjusting mode and correcting the needed phase shift automatically [54]. Ultimately, the active clock edge appears in the optimal position with regard to the enable signal, i.e. in the middle of the shortest enable gate, to avoid the metastable behaviour of the first FF in the main counter and to assure maximum operating frequency.

Issues of synchronization of the timing signals and operating blocks in TICs based on multilevel interpolation are also discussed in [55], where the problem of generation of the interpolation residue between the interpolators is presented in detail and over widening of the range of second interpolation stage is proposed to allow some intrinsic mismatch between the internal path delays.

7.5 Measurement Uncertainty and Calibration

Regardless of the measurement method used, the value obtained as a result of time interval measurement is usually not its real value but only a better or worse approximation of the value. Neglecting gross errors and blunders, the discrepancy is an effect of systematic and random errors.

The *systematic error* arises due to imperfections of meters or measurement methods and generally its value remains constant. In a TIC it is mainly caused by

the difference in delays of paths and circuits transmitting measured pulses in both channels (START and STOP) of the counter. Most differences in delays are roughly compensated during the TIC desingning process. The value of uncompensated delays has to be identified through a calibration procedure and taken into account, as a correction, during the calculation of a measurement result. The most common way of evaluating the value is the measurement of a well known time interval (e.g. $T = 0$).

The *random error* is a result of simultaneous influence of many uncorrelated disturbing factors, of which total impact changes from measurement to measurement, and manifests itself as fluctuations in the readings of a TIC while performing measurements of the same time interval repeatedly under the same conditions. The value of random error determines the *precision* of a counter. A quantitative measure of the precision is a *standard measurement uncertainty* that may be calculated as a root mean square (rms) of component uncertainties due to disturbing factors. For an interpolating TIC that measures time intervals longer than a single clock period the uncertainty is given by:

$$\sigma_{\text{rms}} = \sqrt{\sigma_Q^2 + \sigma_{\text{INLST}}^2 + \sigma_{\text{INLSP}}^2 + \sigma_{\text{CLK}}^2 + \sigma_{\text{TDC}}^2 + \sigma_{\text{ST}}^2 + \sigma_{\text{SP}}^2},$$

where σ_{Q} is a quantization error, σ_{INLST} and σ_{INLSP} are component uncertainties due to nonlinearities of interpolators in channels START and STOP respectively, σ_{CLK} is a timing jitter of a reference clock, σ_{ST} and σ_{SP} mean timing jitter of edges of measured signals START and STOP, and σ_{TDC} is a timing jitter of these signals inside a TIC [12]. All values of component uncertainties are expressed as rms values. An extensive discussion on measurement uncertainty and its sources in TDCs and TICs is presented in [3, 12, 18, 37, 47].

If relatively short time intervals are measured (order of nano- and microseconds), the main sources of random error are typically the quantization error and nonlinearity of interpolators.

The *quantization error*, called also quantization noise, is an immanent attribute of each quantization process and describes the difference between the actual analog value of the converted quantity and its quantized digital value. The value of the measured time interval T can be expressed as a function of resolution q of a converter: $T = q(Q + F)$, where Q is an integer part and F $(0 \leq F \leq 1)$ is a fractional part in terms of q. The measurement uncertainty due to quantization error is then given by standard deviation $\sigma = q\sqrt{(F(1 - F))}$ what graphically may be presented in the form of a semicircle with a radius of q [12]. The maximum value of standard deviation $\sigma_{\text{MAX}} = q/2$ is obtained for $F = 0.5$, while its rms value can be calculated as $\sigma_{\text{RMS}} = \text{q}/\sqrt{6} \approx 0.408\ q$ [57]. The quantization error can then be reduced by increasing the resolution of conversion and in the newest TICs its value is far below the picosecond level [21, 47].

In real TICs the widths of bins in interpolators are uneven, which causes *nonlinearity of conversion*. In the most popular delay line-based converters it is an effect of different unit delays due to inhomogeneity in the parameters of the silicon process, systematic interference of the essential signals with noise sources and

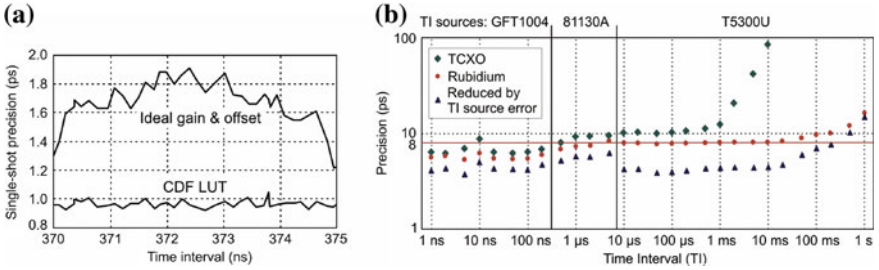

Fig. 7.25 Precision of a TIC within a single clock period (**a**) [21], and within a wide measurement range (**b**) [34]

finally variations in the layout design, or irregularity in the layout of FPGA devices. The integral nonlinearity reaches its maximum value in the middle of the delay line range, where its standard deviation is given by $\sigma_{\mathrm{INL}(n/2)} = (\sigma_1\sqrt{n})/2$, where σ_1 is the standard deviation of the unit delay and n is the number of delay elements in the chain [60]. To reduce the nonlinearity the shorter delay line should be used. Therefore, multistage interpolation and/or a high frequency clock are advised. Independently of it to achieve high measurement precision, the characteristic linearization or nonlinearity correction should be applied.

The linearization of a transfer function is realized by the delay adjustment of a single delay element (changing its bias voltage) or by the use of *dummy loads* attached to outputs of selected delay elements. These additional loads are implemented as spare inputs of logic gates (FPGA devices) [30] or correction capacitances of binary controllable values (ASIC chips) [5, 12, 37]. After the ways of "hardware" linearization are used up, the nonlinearity of transfer function may be identified and used for correction of a calculated result to improve the single-shot precision [19, 31, 37]. The LUTs are often utilized for these purposes [12, 37, 56]. They can also be used to calculate the cumulative distribution function (CDF) which maps the binary outputs of a converter to corresponding time intervals [21]. In this way, using the appropriately large sample size, the nonlinearity error may be virtually eliminated that results in a significant improvement in precision (Fig. 7.25a). Similar approach is used in [23, 34], where the hardware code processor executes the nonlinearity identification procedure, calculates and stores in the RAM memory the resulting transfer characteristic, and utilizes it for the calculation of the conversion result. Also neural networks can effectively be used for the nonlinearity correction [31].

Static parameters of a converter, such as the resolution and nonlinearities are evaluated during a *calibration procedure*. In general, at least two approaches are used, namely (1) two-points or average delay calibration and (2) statistical or bin-by-bin calibration [30, 58]. The second one, based on a large number of measurements of random time intervals within the converter measurement range (CDT, [18, 35]), is commonly utilized due to its simplicity and complete information that is obtained, especially concerning the converters nonlinearity. The

reliability of such calibration strongly depends on the number of measurements that should be large enough [35], and the distribution of the input stimuli that ideally should be uniform. A pseudorandom integrated stimuli generator is proposed in [59], while various issues regarding the calibration of TDCs and TICs are discussed in [16, 18, 30, 37, 56, 58, 59].

Table 7.1 Measurement performance of recent TDCs

Ref., year	Method/ techn. (nm)	LSB (ps)	σ (ps)	MR (ns)	DNL (LSB)	INL (LSB)	mr (MHz)	P (mW)	Size (mm^2)
[9], 2006	DL/130	12	–	0.5	1	1.15	–	2.5	0.4
[29], 2007	VDL/130	31	–	–	1.25	1.45	**500**	**1**	0.15
[41], 2007	VOS/350	37.5	40	50	0.2	1	0.168c	150	0.6 × 0.37b
[48], 2008	TDA/90	**1.25**	**1.25**	0.64	0.8	**3**	10	3	0.6b
[28], 2008	LPI/90	4.7	3.3	0.6	0.6	1.2	–		0.02
[42], 2010	VR/130	8	–	32	–	–	15	7.5	0.75 × 0.35
[5], 2010	PSDL/40	5.5	4.4	**90**	–	–	40	1.8	0.01
[33], 2010	DL/130 FPGA	10	24	6.66	2.5	5.5	–	–	–
[49], 2011	TDA/350	8.9	2.37	27	1.3	17	3.125	2.37	0.0264
[44], 2010	VR/130	16.5	8.08	5	–	–	15	4.5	0.16
[25], 2011	DL/90 FPGA	12	9	10.2	1	13.5	100	–	**–**
[27], 2011	DL/65	80	–	4	**0.01**	–	250	5.66	**0.0063**

PS pulse shrinking, *DL* delay line, *PSDL* parallel scaled delay lines, *VDL* vernier delay lines, *VOS* vernier oscillators, *VR* vernier ring, *LPI* local passive interpolation, *TDA* time difference amplification

Table 7.2 Measurement performance of selected TICs

Ref., year	Method/ techn. (nm)	LSB (ps)	σ (ps)	MR (μs)	INL (LSB)	CLK (MHz)	mr (kHz)	P (mW)	Size (mm^2)
[19], 1994	TAD/ discrete ECL	3	20	$\mathbf{2.2 \times 10^7}$	–	100	–	–	–
[17], 2000	ATS/800	32	30	2.5	0.16	100	156	350	3.5 × 3.4
[11], 2000	PDL/250	24.5	20	102.4	4	320	–	–	–
[12], 2006	PSDL/350	12.2	8.1	202	1.64	175	500	40	2.5 × 3
[50], 2009	MPC/180	61	46	0.082	–	**683**	–	**18**	**0.5 × 0.8**
[47], 2009	SA/350	1.2	3.2	327	–	100	$\mathbf{5 \times 10^3}$	33	2.1 × 2.06
[34], 2013	EDL/45 FPGA	1.14	6	1×10^7	–	500	$\mathbf{5 \times 10^3}$	750	13.44
[21], 2011	TAD/discrete ECL	**0.1**	**1**	Program.	**0.5**	200	–	5.8×10^3	–

TAD time-amplitude-digital conversion, *ATS analog time stretching*, *PDL* passive delay lines, *PSDL* parallel scaled delay lines, *MPC* multiphase clock, *SA* successive approximation, *EDL* equivalent delay line

As the most important measure of the random error, the single-shot precision is evaluated and typically the worst case value is presented. More informative, however, is the characteristic of variation of the precision as a function of the measured time interval. Since this variation repeats itself with the period of the reference clock [18], the time interval spanning the clock period is long enough (Fig. 7.25a). The exceptions from the rule are TICs with a very wide measurement range. For them, the characteristic of precision within the whole range is more useful because it reveals the influence of the reference clock instability (time interval $>$ 100 ms, Fig. 7.25b).

7.6 Comparison of Performance

The performance of several recent integrated realizations of TDCs and TICs are summarized in Tables 7.1 and 7.2, respectively. These two groups of time digitizers were separated to emphasize the significant difference in design complexity. The recent wide-range and high-precise TICs are very often based on multistage interpolation that requires dealing with additional problems, such as synchronization or clock jitter and instability. Surprisingly, despite the more complicated structure the most advanced TICs offer a precision comparably high to that of simpler TDCs. Further, the highest resolution (100 fs) and precision (1 ps) are achieved in the TIC based on the analog interpolation method (time-to-voltage-to-digital conversion) and built as a compound of FPGA technology and discrete components [21].

The other way of comparing the performance of the state of the art developments is a plot that allows presenting designs from various but isolated perspectives, e.g. LSB versus technology or area versus technology [49]. However, there is a lack of universal Figure of Merit (FoM) that allows efficiently and

Table 7. 3 Figures of merit of recent TDCs and TICs

Ref., year	ENOB	FoM ($1/J\mu m^2$)	FoM_P [1/nJ]
[41], 2007 TDC	8.49	1.82×10^3	4.04×10^1
[48], 2008 TDC	7.21	8.21×10^5	4.93×10^2
[5], 2010 TDC	12.52	1.31×10^{10}	1.31×10^5
[44], 2010 TDC	7.48	3.76×10^6	5.95×10^2
[17], 2000 TIC	14.55	9.01×10^2	1.07×10^1
[12], 2006 TIC	22.78	1.20×10^7	9.00×10^4
[47], 2009 TIC	24.81	1.03×10^9	4.47×10^6
[34], 2013 TIC	**38.81**	$\mathbf{2.39 \times 10^{11}}$	$\mathbf{3.21 \times 10^9}$

Fig. 7.26 Desktop and portable TICs

Table 7.4 Measurement performance of commercial TICs

Model	MR (μs)	LSB (ps)	σ (ps)	Td (ns)	mr (MHz)	P (W)
ACAM TDC-GPX (IC chip)	65	40	**10**	–	–	**0.15**
Agilent U1050A (CompactPCI)	20×10^6	5 ÷ 50	≤100	10	–	<25
LeCroy 2228A (CAMAC module)	0.1, 0.2, 0.5	50, 100, 250	–	10	10×10^{-6}	10.9
Ortec 9353 (PCI board)	6.7×10^6	100	100	–	~3.3	8.5
Phillips Scientific 7186 (CAMAC module)	0.1 ÷ 0.8	25 ÷ 200	–	–	~0.13	39
Blue Sky Electronics PicoTOF400 (PCI board)	842	402	–	**0**	–	4
SRS SR620 (desktop)	$\mathbf{10 \times 10^9}$	4	25	800	1.3×10^{-3}	70
VIGO T4100U (USB module)	1×10^9	**1.8**	**10**	200	**5**	2

unambiguously to compare various solutions. An interesting attempt at static characterization of time digitizers is proposed in [1], where two FoMs were defined. The first one allows the comparison of devices in terms of precision, speed, power dissipation and size, and it is calculated as follows: $\text{FoM} = (2^{\text{ENOB}} \times mr)/(P \times A)$, where mr is the measurement rate, P is the power dissipation, A is the device area, and ENOB is the Effective Number of Bits. The ENOB defined originally for A/D converters as a measure of the signal-to-noise and distortion ratio [5], after legitimate simplification based on the assumption that the rms value of noise is represented by the worst case σ value, may be expressed by the formula: $\text{ENOB} = \log_2 (MR/\sigma\sqrt{12})$.

The second figure of merit proposed in [1] is independent of the device area: $\text{FoM}_\text{P} = \text{FoM} \times A$. Both FOMs are used to compare measurement performance of several recently published TDCs and TICs (Table 7.3).

Commercially available TICs are usually designed and manufactured in form of desktop instruments (SR620, *SRS*, Fig. 7.26) or computer boards (9353, *Ortec*).

Recently, precise TICs are also made as small, portable modules, which may conveniently be controlled and supplied with a notebook or netbook (T3200U, *VIGO*, Fig. 7.26). The features of several commercial TICs are presented in Table 7.4.

Acknowledgment This work was supported by the Polish National Science Centre under contract no. DEC-2011/01/B/ST7/03278.

References

1. Napolitano, P., Moschitta, A., Carbone, P.: A survey on time interval measurement techniques and testing methods. In: Proceedings of the IEEE Instrumentation and Measurement Technology Conference (I2MTC), pp. 181–186 (2010)
2. Henzler, S.: Time-to-Digital Converters. Springer, Heidelberg (2010), 123 pp
3. Kalisz, J.: Review of methods for time interval measurements with picosecond resolution. Metrologia **41**, 35–51 (2004)
4. IEEE standard for terminology and test methods for analog-to-digital converters, IEEE Std, 2001
5. Borremans, K., Vengattarmane, J., Craninkx, A.: 6fJ/step, 5.5 ps time-to-digital converter for a digital PLL in 40 nm digital LP CMOS. In: IEEE Radio Frequency Integrated Circuits Symposium (RFIC), pp. 417–420 (2010)
6. Fundamentals of time interval measurements, Application Note 200-3, Hewlett-Packard, 1997
7. Nissinen, I., Mäntyniemi, A., Kostamovaara, J.: A CMOS time-to-digital converter based on a ring oscillator for a laser radar. In: Proceedings of the IEEE European Solid-State Circuits Conference (ESSCIRC'2003), Estoril, Portugal, pp. 469–472 (2003)
8. Arai, Y.: A high-resolution time digitizer utilizing dual PLL circuits. Proc. IEEE Nucl. Sci. Symp. Conf. Rec. **2**, 969–973 (2004)
9. Tonietto, R., Zuffetti, E., Castello, R., Bietti, I.: A 3 MHZ bandwidth low noise RF all digital PLL with 12 ps resolution time to digital converter. In Proceedings 32nd ESSCIRC, pp. 150–153 (2006)
10. Rahkonen, T., Kostamovaara, J.: The use of stabilized CMOS delay lines in the digitization of short time intervals. IEEE J. Solid-State Circ. **28**(8), 887–894 (1993)
11. Mota, M., Christiansen, J., Debieux, S., Ryjov, V., Moreira, P., Marchioro, A.: A flexible multi-channel high-resolution time-to-digital converter ASIC. Nucl. Sci. Symp. Conf. Rec. **2**, 9-155–9-159 (2000)
12. Jansson, J., Mäntyniemi, A., Kostamovaara, J.: CMOS time-to-digital converter with better than 10 ps single-shot precision. IEEE J. Solid State Circ. **41**(6), 1286–1296 (2006)
13. Spencer, D., Cole, J., Drigert, M., Aryaeinejad, R.: A high-resolution, multi-stop, time-to-digital converter for nuclear time-of-flight measurements. Nucl. Instrum. Methods Phys. Res. A **556**, 291–295 (2006)
14. Szplet, R., Gołaszewski, M.: Integrated time-to-digital converter based on counting method and multiphase clock. Meas. Autom. Monitor. **58**(8), 591–593 (2008)
15. Leskovar, B., Turko, B.: Optical timing receiver for the NASA spaceborne ranging system. Part II: High precision event-timing digitizer, Lawrence Berkeley Laboratory Report (1978)
16. Kalisz, J., Pawłowski, M., Pełka, R.: A method for autocalibration of the interpolation time interval digitiser with picosecond resolution. J. Phys. E: Sci. Instrum. **18**, 444–452 (1985)
17. Räisänen-Ruotsalainen, E., Rahkonen, T., Kostamovaara, J.: An integrated time-to-digital converter with 30-ps single-shot precision. IEEE J. Solid State Circ. **35**(10), 1507–1510 (2000)

18. Kalisz, J., Pawłowski, M., Pełka, R.: Error analysis and design of the Nutt time-interval digitiser with picosecond resolution. J. Phys. E: Sci. Instrum. **20**, 1330–1341 (1987)
19. Kalisz, J., Pełka, R., Poniecki, A.: Precision time counter for laser ranging to satellites. Rev. Sci. Instrum. **65**, 736–741 (1994)
20. Määttä, K., Kostamovaara, J.: High-precision time-to-digital converter for pulsed time-of-flight laser radar applications. IEEE Trans. Instrum. Meas. **47**, 521–536 (1998)
21. Keränen, P., Määttä, K., Kostamovaara, J.: Wide-range time-to-digital converter with 1-ps single-shot precision. IEEE Trans. Instrum. Meas. **60**(9), 3162–3172 (2011)
22. Aloisio, A., Branchini, P., Cicalese, R., Giordano, R., Izzo, V., Loffredo, S.: FPGA implementation of a high-resolution time-to-digital converter. In: IEEE Nuclear Science Symposium NSS '07, pp. 504–507 (2007)
23. Szplet, R., Kalisz, J., Jachna, Z.: A 45 ps time digitizer with two-phase clock and dual-edge two-stage interpolation in FPGA device. Meas. Sci. Technol. **20**, 025108, 11 (2009)
24. Wu, J., Shi, Z.: The 10-ps wave union TDC: improving FPGA TDC resolution beyond its cell delay. IEEE Nucl. Sci. Symp. Conf. Rec. 3440–3446 (2008)
25. Wang, J., Liu, S., Zhao, L., Hu, X., An, Q.: The 10-ps multitime measurement averaging TDC implemented in an FPGA. IEEE Trans. Nucl. Sci. **58**(4), 2011–2018 (2011)
26. Staszewski, R., Vemulapalli, S., Vallur, P., Wallberg, J., Balsara, P.: 1.3 V 20 ps time-to-digital converter for frequency synthesis in 90 nm CMOS. IEEE Trans. Circ. Syst-II: Express Briefs, pp. 1–5 (2005)
27. Elsayed, M., Dhanasekaran, V., Gambhir, M., Silva-Martinez, J., Sánchez-Sinencio, E.: A 0.8 ps DNL time-to-digital converter with 250 MHz event rate in 65 nm CMOS for time-mode_based $\sum\Delta$ modulator. IEEE J. Solid-State Circ. **46**(9), 2084–2098 (2011)
28. Henzler, S., Koeppe, S., Lorenz, D., Kamp, W., Kuenemund, R., Schmitt-Landsiedel, D.: A local passive time interpolation concept for variation-tolerant high-resolution time-to-digital conversion. IEEE J. Solid-State Circ. **43**(7), 1666–1676 (2008)
29. Yousif, A., Haslett, J.: A fine resolution TDC architecture for next generation PET imaging. IEEE Trans. Nucl. Sci. **54**(5), 1574–1582 (2007)
30. Kalisz, J., Szplet, R., Pasierbinski, J., Poniecki, A.: Field-programmable-gate-array-based time-to-digital converter with 200-ps resolution. IEEE Trans. Instrum. Meas. **46**, 51–55 (1997)
31. Pełka, R., Kalisz, J., Szplet, R.: Nonlinearity correction of the integrated time-to-digital converter with direct coding. IEEE Trans. Instrum. Meas. **46**(1), 449–453 (1997)
32. Zieliński, M., Chaberski, D., Kowalski, M., Frankowski, R., Grzelak, S.: High-resolution time-interval measuring system implemented in single FPGA device. Measurement **35**, 311–317 (2004)
33. Daigneault, M., David, J.: A novel 10 ps resolution TDC architecture implemented in a 130 nm process FPGA. In: 8th IEEE International NEWCAS Conference, pp. 281–284 (2010)
34. Szplet, R., Jachna, Z., Kwiatkowski, P., Rozyc, K.: A 2.9 ps equivalent resolution interpolating time counter based on multiple independent coding lines. Meas. Sci. Technol. **24**, 035904, 15 (2013)
35. Doenberg, J., Lee, H., Hodges, D.: Full-speed testing of A/D converters. IEEE J. Solid-State Circ. **19**(6), 820–827 (1984)
36. Mäntyniemi, A., Rahkonen, T., Kostamovaara, J.: A high resolution digital CMOS time-to-digital converter based on nested delay locked loops. In: Proceedings of the IEEE International Symposium Circuits and Systems ISCAS'99, vol. 2, pp. 537–540 (1999)
37. Mäntyniemi, A.: An integrated CMOS high precision time-to-digital converter based on stabilised three-stage delay line interpolation (2004) Acta Univ. Oul. C 210, Doctoral thesis. http://herkules.oulu.fi/isbn951427461X/
38. Chen, P., Liu, S., Wu, J.: High accurate cyclic CMOS time-to-digital conversion with extremely low power consumption. Electron. Lett. **33**(10), 858–860 (1997)

39. Liu, Y., Vollenbruch, U., Chen, Y., Wicpalek, C., Maurer, L., Mayer, T., Boos, Z., Weigel, R.: A 6 ps resolution pulse shrinking time-to-digital converter as phase detector in multi-mode transceiver. In: IEEE Radio and Wireless Symposium, pp. 163–166 (2008)
40. Szplet, R., Klepacki, K.: An FPGA-integrated time-to-digital converter based on two-stage pulse shrinking. IEEE Trans. Instrum. Meas. **59**(6), 1663–1670 (2010)
41. Chen, P., Chen, C., Zheng, J., Shen, Y.: A PVT insensitive vernier-based time-to-digital converter with extended input range and high accuracy. IEEE Trans. Nucl. Sci. **54**(2), 297 (2007)
42. Yu, J., Dai, F., Jaeger, R.: A 12-bit vernier ring time-to-digital converter in 0.13 μm CMOS technology. IEEE J. Solid-State Circ. **45**(4), 830–842 (2010)
43. Liscidini, A., Vercesi, L., Castello, R.: Time to digital converter based on a 2-dimensions Vernier architecture. In: IEEE Custom Integrated Circuits Conference, pp. 45–48 (2009)
44. Yu, J., Dai, F.: A 3-dimentional Vernier ring time-to-digital converter in 0.13 μm CMOS. In: IEEE Custom Integrated Circuits Conference, San Jose, USA, pp. 1–4 (2010)
45. Hsu, C., Straayer, M., Perrott, M.: A low-noise wide-BW 3.6-GHz digital ΔΣ fractional-N frequency synthesizer with a noise shaping time-to-digital converter and quantization noise cancellation. IEEE J. Solid-State Circ. **43**(12), 2776–2786 (2008)
46. Straayer, M., Perrott, M.: A multi-path gated ring oscillator TDC with first-order noise shaping. IEEE J. Solid-State Circ. **44**(4), 1089–1098 (2008)
47. Mantyniemi, A., Rahkonen, T., Kostamovaara, J.: A CMOS time-to-digital converter (TDC) based on a cyclic time domain successive approximation interpolation method. IEEE J. Solid-State Circ. **44**(11), 3067–3078 (2009)
48. Lee, M., Abidi, A.: A 9b, 1.25 ps resolution coarse-fine time-to-digital converter in 90 nm CMOS that amplifies a time residue. IEEE J. Solid-State Circ. **43**(4), 769–777 (2008)
49. Mandai, S., Charbon, E.: A 128-channel, 9 ps column-parallel two-stage TDC based on time difference amplification for time-resolved imaging, ESSCIRC, pp. 119–122 (2011)
50. Nissinen, I., Kostamovaara, J.: On-chip voltage reference-based time-to-digital converter for pulsed time-of-flight laser radar measurements. IEEE Trans. Instrum. Meas. **58**(6), 1938–1948 (2009)
51. Kalisz, J., Orżanowski, T., Szplet, R.: The delay-locked loop technique for temperature stabilization of internal delays of CMOS FPGA devices. Electron. Lett. **36**(14), 1184–1185 (2000)
52. Kinniment, D., Edwards, D.: Circuit technology in a large computer system. In: Proceedings of the Computers, Systems and Technology, pp. 441–450 (1972)
53. Ljuslin, C., Christiansen, J., Marchioro, A., Klingsheim, O.: An integrated 16-channel CMOS time to digital converter. IEEE Trans. Nucl. Sci. **41**, 1104–1108 (1994)
54. Szplet, R.: Auto-tuned counter synchronization in FPGA-based interpolation time digitizers. Electron. Lett. **45**(13), 2 pp (2009)
55. Jansson, J., Mäntyniemi, A., Kostamovaara, J.: Synchronization in a multilevel CMOS time-to-digital converter. IEEE Trans. Circ. Syst. **56**(8), 1622–1634 (2009)
56. Mäntyniemi, A., Rahkonen, T., Kostamovaara, J.: A nonlinearity-corrected CMOS time digitizer IC with 20 ps single-shot precision. Proc. IEEE Int. Symp. Circ. Syst. **1**, 513–516 (2002)
57. Kleinfelder, S., Majors, J., Blumer, K., Farr, W., Manor, B.: MTD132-a new subnanosecond multi-hit CMOS time-to-digital converter. IEEE Trans. Nucl. Sci. **38**(2), 97–101 (1991)
58. Wu, J.: Several key issues on implementing delay line based TDCs using FPGAs. FERMILAB-PUB-09-608-E, 6 pp
59. Amiri, A., Khouas, A., Boukadoum, M.: Pseudorandom stimuli generation for testing time-to-digital converters on an FPGA. IEEE Trans. Instrum. Meas. **58**(7), 2209–2215 (2009)
60. Toifl, T., Vari, R., Moreira, P., Marchioro, A.: 4-channel rad-hard delay generation ASIC with 1 ns timing resolution for LHC. IEEE Trans. Nucl. Sci. **46**(3), 139–143 (1999)

Part II
Modeling

Chapter 8
Look-Up Tables, Dithering and Volterra Series for ADC Improvements

Henrik Lundin and Peter Händel

Absrtact This chapter gives an introduction and an overview to correction methods for analog-to-digital converters (ADCs), including methods that are implemented using look-up tables, the method known as dithering, and methods that are based on a mathematical model of the ADC.

8.1 Introduction

This chapter gives an introduction and an overview to correction methods for analog-to-digital converters (ADCs). 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 sections, each covering a special form of ADC correction. The classification into different families of methods follows that of [5] to a large extent.

In Sect. 8.2, methods that are implemented using look-up tables are reviewed. Post-correction using look-up tables is a common way of diminishing ADC errors, and extensive research has been conducted within this field. As a natural consequence, Sect. 8.2 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 field.

Section 8.3 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. Section 8.3 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.

H. Lundin · P. Händel (✉)
Department of Signal Processing, KTH Royal Institute of Technology, Stockholm, Sweden
e-mail: ph@kth.se

P. Carbone et al. (eds.), *Design, Modeling and Testing of Data Converters*,
Signals and Communication Technology, DOI: 10.1007/978-3-642-39655-7_8,

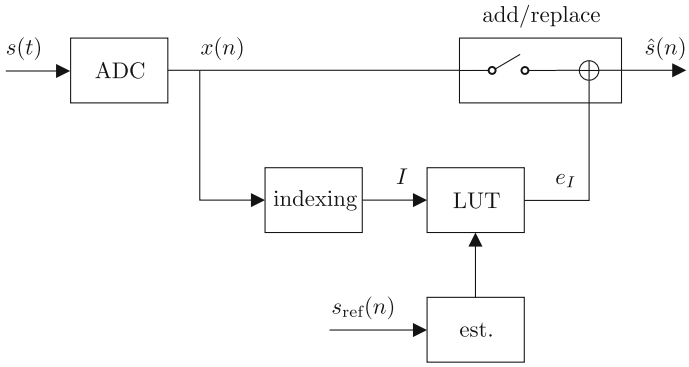

Fig. 8.1 A generic look-up table correction system

In Sect. 8.4 methods that are based on a mathematical model of the ADC are presented. In particular, the section is focused on inverting Volterra models.

8.2 Look-Up Table Based Methods

ADC post-correction using look-up tables (LUTs) is probably the most frequently proposed scheme 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 Fig. 8.1. 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 limits 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.

8.2.1 Classification of LUT Methods

Returning to Fig. 8.1, we will in this section classify various LUT methods depending on how they implement the various blocks of the figure. In particular, we will address the following parts:

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

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 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 Sect. 8.2.3 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.

8.2.1.1 Indexing Schemes

The indexing scheme is perhaps the most important 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 $[\,x(n-K_a) \quad x(n-K+1) \quad \ldots \quad x(n) \quad \ldots \quad x(n+K_b)\,]^T$ and produces a non-negative integer index I associated with sample index n. The indexing function is in most cases causal, so that $K_b = 0$ and $K \triangleq K_a \geq 0$. 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.

8.2.1.2 Static Indexing

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

$$x(n) \rightarrow I, \tag{8.1}$$

i.e., 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 [10].

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 [20, 24]. 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 dynamic error mechanism.

8.2.1.3 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

$$(x(n), x(n-1)) \Rightarrow I \tag{8.2}$$

This method is referred to as *state-space indexing*. 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, [23, 49].

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

$$(x(n), x(n-1), \ldots, x(n-K)) \Rightarrow I. \tag{8.3}$$

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 [51].

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 into the equation, but it is reasonable to say that for resolutions common in high-speed ADCs—some 8–14 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 [50], and in the context of phase-plane correction in [13, 39]). In [28, 30–32], 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 Fig. 8.2 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.

Selecting a beneficial bit mask within implementation constraints (e.g. memory size and number of delays) is a nontrivial bit allocation problem. In [28, 30, 32], an analysis of the allocation problem was made, and optimization methods for finding optimal bit masks were provided. It was also demonstrated that the bit mask could be made rather restrictive while still giving significant compensation for dynamic error effects.

8.2.1.4 Phase-Plane Indexing

As an alternative to state-space indexing, the phase-plane indexing, described in, for example, [6, 21, 36, 39], may be used; sometimes the 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 $\hat{s}'(n)$:

$$\left(x(n), \hat{s}'(n)\right) \Rightarrow I. \tag{8.4}$$

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 [19, 36, 50],

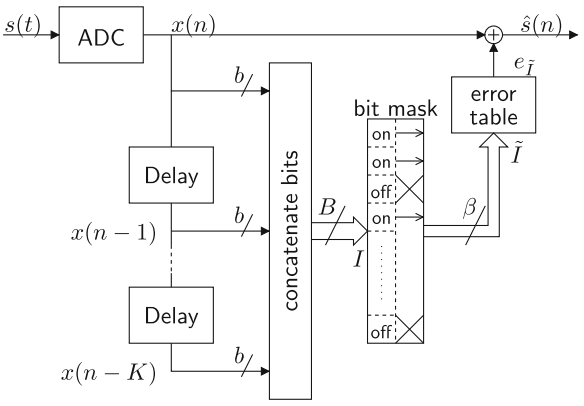

Fig. 8.2 Dynamic post-correction system outline. Since the ADC errors sought to mitigate are dependent on signal history, the correction is also history dependent through the use of K delay elements. In order to reduce the index size (and thereby the memory requirements), a subset of β samples are selected to address the table

or using an analog differentiator and a separate (possibly low resolution) ADC sampling the output of the filter [23]. Just as in the state-space case, the indexing can be generalized to higher order

$$\left(x(n), \hat{s}^{(1)}(n), \ldots, \hat{s}^{(K)}(n)\right) \Rightarrow I, \tag{8.5}$$

where $\hat{s}^{(k)}$ denotes the k-th derivative. Addressing with higher order derivatives has been reported in [13].

8.2.1.5 Alternative Indexing Schemes

The static, state-space and phase-plane schemes presented above are without doubt the most commonly proposed methods. However, alternative methods have occasionally been suggested.

One alternative extension to the static LUT is the frequency selective indexing proposed in [25, 32]. A frequency region estimator is applied, providing a frequency region estimate—from a finite set $\mathcal{A}_F$ of frequency regions—based on a number of successive output samples (typically around 10 samples are used). The estimator is very fast, providing a new estimate $\hat{F}(n) \in \mathcal{A}_F$ for every new sample, with the delay of a memory reference only. Thus, the frequency region estimate $\hat{F}(n)$ can be used in combination with the present sample $x(n)$ to build an index:

$$\left(x(n), \hat{F}(n)\right) \Rightarrow I. \tag{8.6}$$

The method is depicted in Fig. 8.3. With this scheme the correction is made frequency dependent, but the phase information is lost. An alternative frequency dependent correction without phase information is the approach based on INL modeling and end point correction [34, 35].

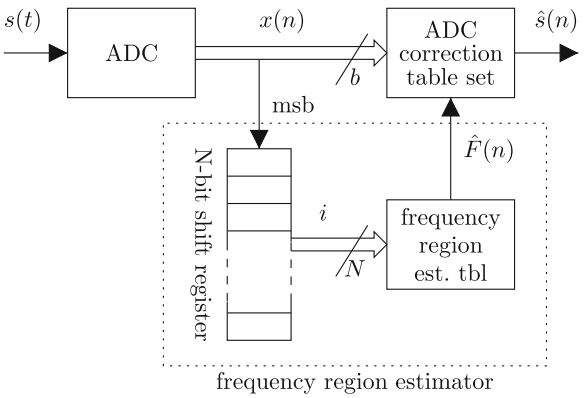

Fig. 8.3 Frequency selective correction system outline

8.2.2 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 would store $e_I + x$, where x is equal to the current output of the ADC.

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 Sect. 8.2.1.1 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 generalized method of Fig. 8.2 does not qualify, since if any of the bits in $x(n)$ are masked away, then different values of $x(n)$ that differ only in the masked bits may give the same index I.

8.2.2.1 Nominal Value

The goal with ADC post-correction is of course to produce an output that is better than before correction. A few approaches will be given here.

Midpoint Correction The midpoint correction strategy is based on the assumption that the ADC acts as a staircase quantizer. That is, the input range is divided into $M = 2^b$ disjoint sets, or quantization regions, $\{\mathcal{S}_j\}_{j=0}^{M-1}$, and if the input signal $s(nT_s)$ (where T_s is the sampling period and nT_s is the n-th sampling instant) falls within region $\mathcal{S}_i$ then the i-th output x_i is produced. The output value is frequently denoted reconstruction value. Nominally, 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. This is then the task of the

post-correction. Thus, in the case of midpoint correction and a static indexing scheme, the table value for an additive correction type system should be

$$e_i = \frac{T_i + T_{i+1}}{2} - x_i. \tag{8.7}$$

Related to mid-point correction is end-point correction [34].

Minimum Mean-Squared Error Correction The widely used squared-error distortion criterion can be applied to quantizer design and ADC post-correction. Optimal correction values for minimizing the mean-square error $E\left[(\hat{s}(n) - s(n))^2\right]$ have been derived in [26] ($E[\cdot]$ is the expected value operator taken with respect to $s(n)$, noting that $\hat{s}(n)$ is a function of $s(n)$). We will briefly review the key results and their implications.

In [26] a quantizer operating on a value s is considered; the quantization region model above is used. The value $s = s(n)$ is regarded to be drawn from a stochastic variable 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[2] $\{x_j\}$, in the mean-squared sense, are given by

$$x_{j,\,\text{opt}} = \arg\min_x \mathrm{E}\left[(x - s)^2 \middle| s \in \mathcal{S}_j\right] = \frac{\int_{s\in\mathcal{S}_j} s f_s(s)\,ds}{\int_{s\in\mathcal{S}_j} f_s(s)\,ds}, \tag{8.8}$$

i.e., the optimal reconstruction value for each region is the "center of mass" of the region. This choice of reconstruction value is evidently dependent not only on the characteristics of the ADC under test, but also on the test signal itself. Thus, we take into account what we know about the signal. 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, just as in the midpoint correction approach above. The midpoint approach is in fact consistent with Lloyd's approach (8.8) 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. [19, 21] 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 sine-waves 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 M first harmonics to the test frequency.

[2] In [26] the reconstruction values are denoted 'quanta'.

8.2.2.2 Precision of Correction Values

In a practical post-correction application the correction values are stored with fixed-point precision. However, most of the evaluations and experiments reported in the literature have been conducted with infinite (or floating-point) precision in the representation of the correction values stored in the LUT. One of few exceptions is [22], where experimental results indicated that the precision of the correction values strongly 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. Sect. 8.2.2.1 above) 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 8.4 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.

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 choice of correction word-length and number of bits to shift are both design parameters to be decided 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.

8.2.3 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

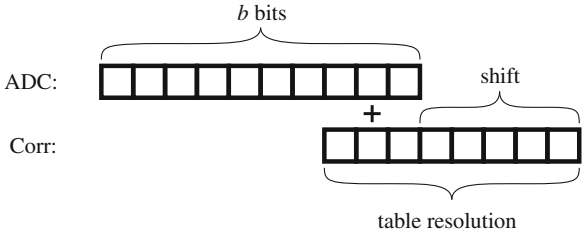

Fig. 8.4 Addition of the ADC output with a correction value. The bits of the table value are shifted in order to enhance the precision of the corrected ADC

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 [15], or a DAC feeding a digitally generated calibration signal to the ADC under test [21, 49]. 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 [14] methods for blind calibration, assuming only a smooth but otherwise unknown probability density function for the reference signal, are presented, while [20] proposes sine-wave reference signals in conjunction with optimal filtering techniques to extract an estimate of the reference signal. The latter method was further developed in [27, 31] 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 Lloyd's approach above, the following method is frequently used. Since neither the regions $\{\mathcal{S}_j\}$ nor the PDF $f_s(s)$ is known, 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, $s(n)$, $n = 0, 1, N-1$, are the calibration samples input to the ADC under test and $s_{\text{ref}}(n)$, $n = 0, 1, N-1$, 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 $s(n)$, $n = 0, 1, N-1$, 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 (8.8) with

$$\int_{s\in\mathcal{S}_j} s f_s(s)\,ds \approx \sum_{s\in\mathcal{C}_j} s\,\frac{1}{N} = \frac{1}{N}\sum_{s\in\mathcal{C}_j} s = \frac{N_j}{N}\overline{\mathcal{C}}_j \tag{8.9}$$

$$\int_{s\in\mathcal{S}_j} f_s(s)\,ds \approx \sum_{s\in\mathcal{C}_j} \frac{1}{N} = \frac{N_j}{N}, \tag{8.10}$$

$$x_{j,\text{opt}} \approx \overline{\mathcal{C}_j}, \tag{8.11}$$

where $\overline{\mathcal{C}}_j$ is the mean value of all samples in $\mathcal{C}_j$.

Recent research has shown that it is not always optimal to use the sample mean $\overline{\mathcal{C}}_j$. In [29] it was shown that under specific conditions, it is better to use an estimator based on ordered samples. In particular, it is shown that when the quantizer obeys the usual quantization region model above, when the calibration signal is symmetric within each quantization region and the reference signal is perturbed by a Gaussian noise, then it is better (in the mean-squared sense) to use $(\min(\mathcal{C}) + \max(\mathcal{C}))/2$ instead of $\overline{\mathcal{C}}_j$ when the standard deviation of the Gaussian perturbation is small compared to the quantization bin width. Note that this is an estimator based on ordered samples, and is therefore a nonlinear estimator.

Calibration methods that do not rely on any digital reference signal has also been proposed in the literature. In [15], 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 [11]. 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 sine-wave histogram method and Bayesian estimation.

8.3 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 8.5 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

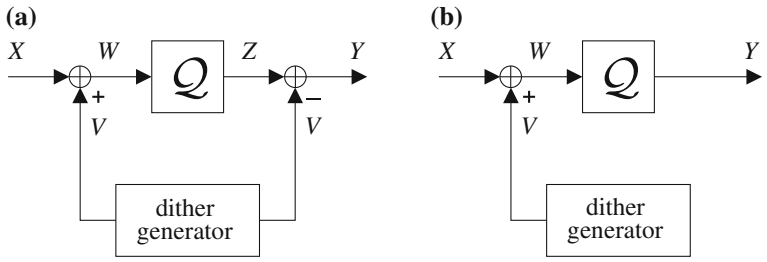

Fig. 8.5 Figures of subtractive and non-subtractive dither. **a** Subtractive, **b** Non-subtractive

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 (explained later) 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 can be found in [54], and briefly outlined in Sects. 8.3.1 and 8.3.2 explains how dithering can reduce the distortion by randomization. Finally, dithering in conjunction with low-pass post-processing is dealt with in Sect. 8.3.3.

8.3.1 Statistical Theory of Quantization and Dithering

Below, 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 resumé in [17], 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 is the theories for dithering. 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 Sects. 8.3.2 and 8.3.3.

Perhaps the oldest example of quantization is rounding off, first analyzed by Sheppard [43] 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 [38] 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

$$\sigma_Q^2 \approx \frac{\Delta^2}{12}, \tag{8.12}$$

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 (8.12)

can be applied to yield the classical "6-dB-per-bit" result for the signal-to-noise and distortion ratio (SINAD) [45]

$$\text{SINAD} = 10\log_{10}\frac{\text{signal power}}{\frac{V^2}{2^{2b}\,12}} \approx 6.02b + \text{constant}. \tag{8.13}$$

In response to the need for a simple 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. The model was popularized by Widrow [52, 53, 55], who also gave conditions for when it is valid.

The pioneering work on dithering was carried out by Roberts [40] in his work on image quantization. Roberts argued that adding noise to an image before quantization and subtracting it before reconstruction could mitigate the quantization effects on the image, seen as regular (i.e., signal dependent) patterns. The general theory of dithering was then further developed by Schuchman [41].

8.3.1.1 Quantization Theory

These results, many of them originally due to Widrow, gives the amplitude quantization counterpart to the widespread Nyquist's sampling theorem (note that sampling can be seen as time quantization). The following results and discussion applies to uniform, scalar, mid-tread quantization with a quantization step size Δ. Figure 8.6 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 8.7 shows the signal relations, including the quantization error $E \triangleq Y - X$. The Fourier transform of the input PDF, usually referred to as the characteristic function (CF), is

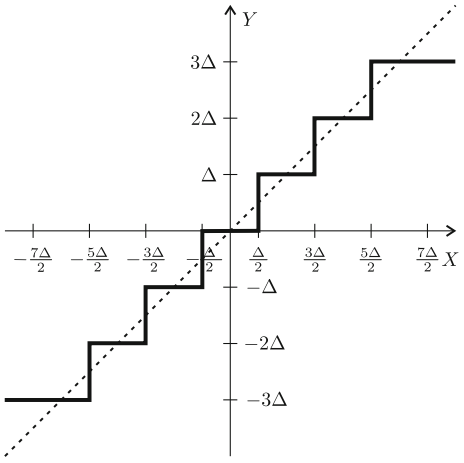

Fig. 8.6 The input–output transfer function of a uniform, scalar, mid-tread quantizer with quantization step size Δ

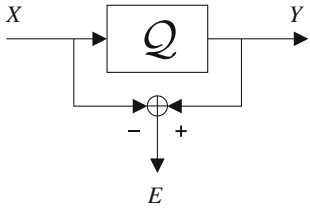

Fig. 8.7 The quantizer $\mathcal{Q}$ with input X, output Y and error E defined

$$\Phi_X(u) = \int_{-\infty}^{\infty} f_X(x) e^{jux} dx = \mathrm{E}\left[e^{jux}\right]. \tag{8.14}$$

Two quantization theorems based on the CF were provided by Widrow, and they are recapitulated here without proofs:

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

$$\Phi_X(u) = 0 \quad \text{for} \quad |u| > \frac{\pi}{\Delta}, \tag{8.15}$$

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*

$$\Phi_X(u) = 0 \quad \text{for} \quad |u| > \frac{2\pi}{\Delta} - \varepsilon, \tag{8.16}$$

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

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 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. 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 [55], and the PDF of the quantization error is exactly zero-mean uniform with variance $\Delta^2/12$ [52, 53]. Under the same conditions it can also be shown [52, 53] that the quantization error is uncorrelated with the input to the quantizer. Sripad and Snyder [44] gave a weaker sufficient and necessary condition for the quantization error.

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.

8.3.1.2 Dithering Theory

We have now established some 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 [56]. As before, we consider a uniform, scalar, mid-tread quantizer with quantization step size Δ (cf. Fig. 8.6). 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 is provided in the sequel.

The general framework for both cases is shown in Fig. 8.5. 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

$$E = Y - X = \begin{cases} \mathcal{Q}(X+V) - X & \text{non-subtractive dithering;} \\ \mathcal{Q}(X+V) - (X+V) & \text{subtractive dithering.} \end{cases} \tag{8.17}$$

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 [56]:

Theorem 3 (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*

$$\Phi_V\left(\frac{2\pi k}{\Delta}\right) = 0 \quad \textit{for all integers } k \neq 0. \tag{8.18}$$

Moreover, the total error E is statistically independent of the system input X if and only if the same condition (8.17) 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

$$\Phi_{V_1 V_2}\left(\frac{2\pi k_1}{\Delta}, \frac{2\pi k_2}{\Delta}\right) = 0 \quad \textit{for all } (k_1, k_2) \in \mathbb{Z}_0^2. \tag{8.19}$$

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., $\mathbb{Z}_0^n = \mathbb{Z}^n \backslash \mathbf{0}$. The first part of the theorem (uniform distribution) was proven by Schuchman [41], and the condition (8.18) 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. [44]. Note that Schuchman's condition is satisfied if V is a s.v. with uniform density in $[-\Delta/2, \Delta/2]$, and (8.19) 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 3, 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. [56], 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 4 *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 tells 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 [33, 56]:

Theorem 5 *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*

$$G_V^{(m)}\left(\frac{2\pi k}{\Delta}\right) = 0 \quad \textit{for all integers } k \neq 0. \tag{8.20}$$

Further, when (8.20) is fulfilled, the m-th moment is

$$\mathrm{E}[E^m] = (-j)^m G_V^{(m)}(0). \tag{8.21}$$

Here, the function $G_V(u)$ is

$$G_V(u) \triangleq \operatorname{sinc}\left(\frac{\Delta}{2\pi}u\right)\Phi_V(u), \tag{8.22}$$

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

$$\operatorname{sinc}(x) \triangleq \frac{\sin(\pi x)}{\pi x}. \tag{8.23}$$

Since $\mathrm{E}[V^m] = (-j)^m \Phi_V^m(0)$, we can use (8.21) to express the moments of E in the moments of V; for instance,

$$\mathrm{E}[E] = \mathrm{E}[V] \tag{8.24}$$

and

$$\mathrm{E}\left[E^2\right] = \mathrm{E}\left[V^2\right] + \Delta^2/12 \tag{8.25}$$

when (8.20) holds. An immediate result of (8.25) 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, however, that comes at the expense of making the noise power dependent of the input signal.

It can also be shown that when condition (8.20) in Theorem 5 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 (8.20) is provided:

Corollary 1 *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*

$$\Phi_V^{(r)}\left(\frac{2\pi k}{\Delta}\right) = 0 \quad \textit{for all integers } k \neq 0 \textit{ and all } r = 0, 1, \ldots, m-1. \tag{8.26}$$

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

Theorem 6 *The joint moment* $\mathrm{E}\left[E_1^{m_1} E_2^{m_1}\right]$ *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*

$$G_{V_1 V_2}^{(m_1, m_2)}\left(\frac{2\pi k_1}{\Delta}, \frac{2\pi k_2}{\Delta}\right) = 0 \quad \textit{for all } (k_1, k_2) \in \mathbb{Z}_0^2. \tag{8.27}$$

Here, the function $G_{V_1V_2}(u_1, u_2)$ is defined as

$$G_{V_1V_2}(u_1, u_2) = \operatorname{sinc}\left(\frac{\Delta}{2\pi}u_1\right)\operatorname{sinc}\left(\frac{\Delta}{2\pi}u_2\right)\Phi_{V_1V_2}(u_1, u_2) \tag{8.28}$$

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 [56]. Finally, an important implication of Theorem 6 is pointed out:

Corollary 2 *Any i.i.d. non-subtractive dither signal that satisfies* (8.26) *for* $m = \max(m_1, m_2)$ *will provide*

$$\mathrm{E}\left[E_1^{m_1} \cdot E_2^{m_2}\right] = \mathrm{E}\left[E_1^{m_1}\right] \cdot \mathrm{E}\left[E_2^{m_2}\right] \tag{8.29}$$

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

When this hold true, the moments $\mathrm{E}\left[E_1^{m_1}\right]$ and $\mathrm{E}\left[E_2^{m_2}\right]$ are given by (8.21). From (8.24) we have that when Corollary 2 holds and the dither signal is zero-mean, the total error signal is uncorrelated in time, i.e., $\mathrm{E}[E_1E_2] = 0$.

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.

8.3.1.3 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 3. Hence, a *subtractively* dithered system using such a rectangular noise totally obeys the PQN model.

In the *non-subtractive* case, we see that Theorem 5 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 *noise modulation*. From Corollary 2 we conclude that since (8.26) 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 5 tells us that we need a dither distributed such that (8.20) is fulfilled for $m = 1, 2$.

One such distribution is the symmetric triangular distribution ranging from $-\Delta$ to Δ, i.e.,

$$f_V(v) = \begin{cases} \frac{\Delta - |v|}{\Delta^2} & |v| < \Delta \\ 0 & \text{otherwise.} \end{cases} \tag{8.30}$$

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 5 for both $m = 1$ and $m = 2$ and also Corollary 1 for $m = 2$. Thus, $\mathrm{E}[E] = 0$ from (8.24) and $\mathrm{var}(E) = \mathrm{E}[E^2] = \Delta^2/4$ from (8.25). Again, if the individual dither samples are i.i.d. the error sequence is white.

8.3.2 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 Sect. 8.3.1 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 Fig. 8.8, 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 spectrum of the output signal. However, if we add a dithering signal v to the input x prior to quantization, we might shift the input to the quantizer 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 result is that the deterministic nature of the distortion is broken.

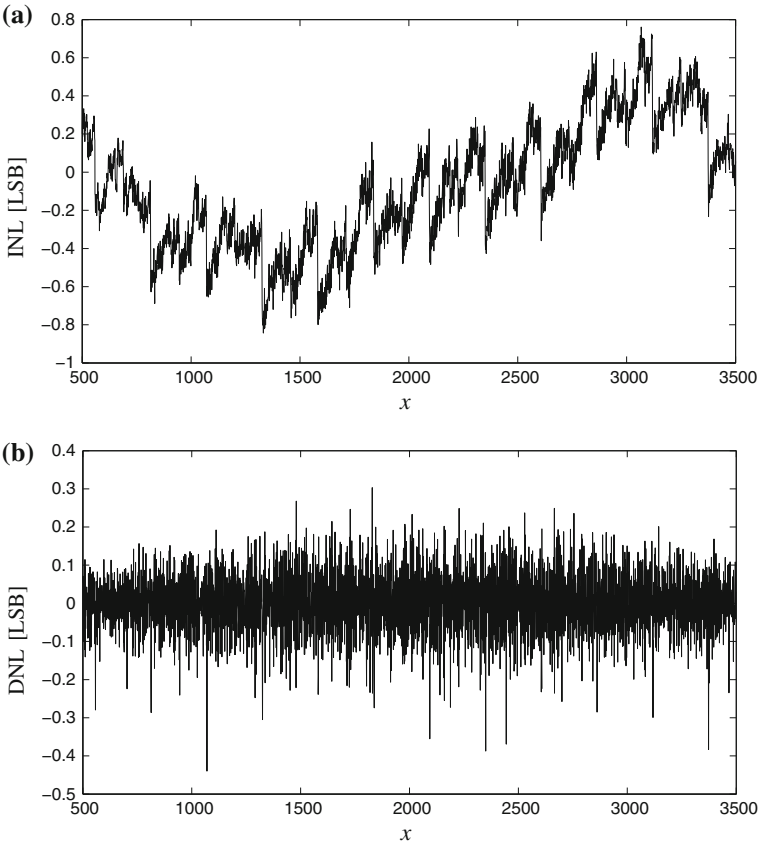

Fig. 8.8 INL and DNL from an Analog Devices AD9430. **a** INL, **b** DNL

An early contribution in this direction was given by De Lotto and Paglia in [12], 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 [4], 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.

8.3.3 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. [2, 3, 8, 9]. 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 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 [9], 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 [9] indicate that an increase of up to 3 effective bits can be obtained using dithering and averaging.

In [3], 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

$$\mathrm{ENOB}_{\,\mathrm{U\ determ}} = b + \log_2 N \tag{8.31}$$

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

$$\mathrm{ENOB}_{\,\mathrm{U\ stoch}} = b + \frac{1}{2}\log_2 \frac{N}{1+k^2}, \tag{8.32}$$

when the dither is stochastic and uniform in $[-k\Delta/2, k\Delta/2]$. The number of samples averaged is N. Experimental results in [3] indicates that the effective number of bits for a practical 12-bit converter can be increased up to 16 bits using non-subtractive stochastic uniform dithering in $[-\Delta, \Delta]$ and $N = 16384$, and even further for subtractive dither.

8.4 Model Inversion Methods

ADC nonlinearity correction through model inversion has been proposed in the literature on several occasions (cf. [5]). 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, i.e. operating on the output samples, in order to reduce or even cancel the unwanted distortion. Figure 8.9 shows the general concept of ADC correction with inverse models.

The vast majority of the references proposing a model inversion method are based on the Volterra model, e.g. [16], but the use of other models have been reported, e.g. orthogonal polynomials and Wiener models in [51] and Chebyshev polynomials in [1]. In this chapter we will concentrate on the Volterra model.

8.4.1 Volterra Model

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

$$y(n) = \sum_{k=0}^{\infty} h_k x(n-k), \tag{8.33}$$

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

$$y(n) = H_0 + H_1(x(n)) + \cdots + H_M(x(n)) \tag{8.34}$$

where

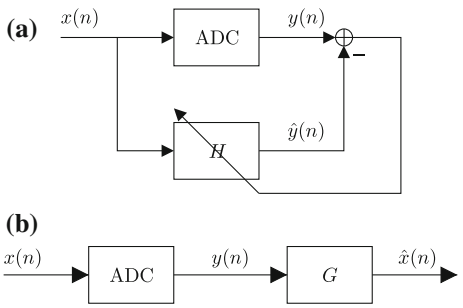

Fig. 8.9 ADC correction using inverse system models. In (**a**) a model H for the ADC is found by identification. In (**b**) the inverse $G = H^{-1}$ of the identified model is used in sequence with the ADC, applying post-distortion to reduce the nonlinearities of the converter

$$H_m(x(n)) = \sum_{k_1=0}^{\infty} \sum_{k_2=k_1}^{\infty} \cdots \sum_{k_m=k_{m-1}}^{\infty} h_m(k_1, k_2, \ldots, k_m) \times x(n-k_1)x(n-k_2)\cdots x(n-k_m). \tag{8.35}$$

The Volterra kernels h_m are assumed to be symmetric, hence the indexing in (8.35) where subsequent summation indices start at the index of the preceding sum. Finally, the kernels are truncated to finite length sums

$$H_m(x(n)) = \sum_{k_1=0}^{K_1} \sum_{k_2=k_1}^{K_2} \cdots \sum_{k_m=k_{m-1}}^{K_m} h_m(k_1, k_2, \ldots, k_m) \times x((n-k_1)x(n-k_2)\cdots x(n-k_m). \tag{8.36}$$

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 [46], 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 [47] and make a comparison between a look-up table correction and a Volterra based correction [48]. Gao and Sun [18] 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. Experimental three tone characterization of Volterra kernels of a pipelined ADC was performed in [7].

8.4.2 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 [42, 51]

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 are a Volterra model denoted F, then*

$$F(x(n)) = x(n) + \sum_{m=p+1}^{\infty} F_m(x(n)). \tag{8.37}$$

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 it has been shown in [42] 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 8.9 shows this approach.
2. The inverse system G is identified directly as in Fig. 8.10, 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 [42], 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.

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 [47]. 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 [47]) while a look-up table has (theoretically) infinite nonlinearity order at a very low computational cost.

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 [37], where the Volterra model was compared to an

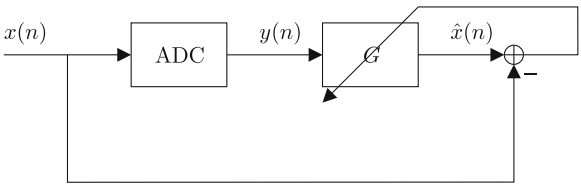

Fig. 8.10 Direct identification of the inverse system from experiments

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.

References

1. Adamo, Francesco, Attivissimo, Filippo, Giaquinto, Nicola, Savino, Mario: FFT test of A/D converters to determine the integral nonlinearity. IEEE Trans. Instrum. Meas. **51**(5), 1050–1054 (2002)
2. Aumala, Olli, Holub, Jan: Dithering design for measurement of slowly varying signals. Measurement **23**(4), 271–276 (1998)
3. Aumala, O., Holub, J.: Practical aspects of dithered quantizers. In: Holub, J., Smid, R. (eds.) *Dithering in Measurement: Theory and Applications, Proceedings of 1st International Online Workshop*. CTU FEE Department of Measurement, Prague (TUT Measurement and Information Technology, Tampere (March 1998))
4. Overcoming converter nonlinearities with dither. Application Note AN–410, Analog Devices, Norwood, MA. Brad Brannon. 1995
5. Balestrieri, E., Daponte, P., Rapuano, S.: A state of the art on ADC error compensation methods. In: *Proceedings IEEE Instrumentation and Measurement Technology Conference*, vol. 1, pp. 711–716. Como (2004)
6. Bergman, David I.: Dynamic error correction of a digitizer for time domain metrology. IEEE Trans. Instrum. Meas. **53**(5), 1384–1390 (2004)
7. Bjorsell, N., Suchanek, P., Handel, P., Ronnow, D.: Measuring Volterra kernels of analog-to-digital converters using a stepped three-tone scan. *IEEE Trans. Instrum. Meas.* **57**(4), 666–671 (2008)
8. Carbone, Paolo: Quantitative criteria for the design of dither-based quantizing systems. IEEE Trans. Instrum. Meas. **46**(3), 656–659 (1997)
9. Carbone, Paolo, Petri, Dario: Performance of stochastic and deterministic dithered quantizers. IEEE Trans. Instrum. Meas. **49**(2), 337–340 (2000)
10. Dent, A.C., Cowan, C.F.N.: Linearization of analog-to-digital converters. *IEEE Trans. Circuits Systems* **37**(6), 729–737 (1990)
11. Daponte, P., Holcer, R., Horniak, L., Michaeli, L., Rapuano, S.: Using an interpolation method for noise shaping in A/D converters. In: *Proceedings of the 7th European Workshop on ADC Modelling and Testing*, Prague, 147–150 June 2002. IMEKO
12. Ivo De Lotto, Paglia, G.E.: Dithering improves A/D converter linearity. *IEEE Trans. Instrum. Meas.* **35**(2), 170–177 (1986)
13. Deyst, J.P., Vytal, J.J., Blasche, P.R., Siebert, W.M.: Wideband distortion compensation for bipolar flash analog-to-digital converters. In: *Proceedings of the 9th IEEE Instrumentation and Measurement Technology Conference*, pp. 290–294, Metro New York (1992)
14. Elbornsson, J.: 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
15. Eduri, U., Maloberti, F.: Online calibration of a Nyquist-rate analog-to-digital converter using output code-density histograms. *IEEE Trans. Circuits Systems—Part I: Fundam. Theory Appl.* **51**(1), 15–24 (2004)
16. Goodman, J., Miller, B., Herman, M., Vai, M., Monticciolo, P.: Extending the dynamic range of RF receivers using nonlinear equalization. In: *2009 International WD&D Conference*, pp. 224–228 (2009)
17. Gray, Robert M., Neuhoff, David L.: Quantization. IEEE Trans. Inf. Theory **44**(6), 2325–2383 (1998)

18. Gao, X.M., Sun, S.: Modeling the harmonic distortion of analog-to-digital converter using Volterra series. In: *Proceedings IEEE Instrumentation and Measurement Technology Conference, IMTC/1994*, vol. 2, 911–913 May 1994
19. Hummels, D.M., Irons, F.H., Cook, R., Papantonopoulos, I.: Characterization of ADCs using a non-iterative procedure. In: *Proceedings IEEE International Symposium on Circuits and Systems*, vol. 2, pp. 5–8. London (1994)
20. Händel, Peter, Skoglund, Mikael, Pettersson, Mikael: A calibration scheme for imperfect quantizers. IEEE Trans. Instrum. Meas. **49**, 1063–1068 (2000)
21. Hummels, Don: Performance improvement of all-digital wide-bandwidth receivers by linearization of ADCs and DACs. Measurement **31**(1), 35–45 (2002)
22. Irons, F.H., Chaiken, A.I.: Analog-to-digital converter compensation precision effects. In: *29th Midwest Symposium on Symposium on Circuits and Systems*, pp. 849–852. Lincoln (1986)
23. Irons, F.H., Hummels, D.M., Kennedy, S.P.: Improved compensation for analog-to-digital converters. IEEE Trans. Circuits Systems **38**(8), 958–961 (1991)
24. Irons, F.H.: Dynamic characterization and compensation of analog to digital converters. In: *Proceedings of IEEE International Symposium on Circuits and Systems*, vol. 3, pp. 1273–1277 (1986)
25. Lundin, H., Andersson, T., Skoglund, M., Händel, P.: Analog-to-digital converter error correction using frequency selective tables. In: *RadioVetenskap och Kommunikation (RVK)*, pp. 487–490. Stockholm (2002)
26. Lloyd, S.P.: Least squares quantization in PCM. *IEEE Trans. Inf. Theor.* **28**(2), 129–137 (1982)
27. Lundin, H., Skoglund, M., Händel, P.: On external calibration of analog-to-digital converters. In: *IEEE Workshop on Statistical Signal Processing*, pp. 377–380. Singapore (2001)
28. Lundin, Henrik, Skoglund, Mikael, Händel, Peter: A criterion for optimizing bit-reduced post-correction of AD converters. IEEE Trans. Instrum. Meas. **53**(4), 1159–1166 (2004)
29. Lundin, H., Skoglund, M.,Händel, P.: On the estimation of quantizer reconstruction levels. In: *Proceedings IEEE Instrumentation And Measurement Technology Conference*, vol. 1, pp. 144–149. Ottawa (2005)
30. Lundin, Henrik, Skoglund, Mikael, Händel, Peter: Optimal index-bit allocation for dynamic post-correction of analog-to-digital converters. IEEE Trans. Signal Process. **53**(2), 660–671 (2005)
31. Lundin, H.: Dynamic compensation of analogue-to-digital converters. Master's thesis, Royal Institute of Technology (KTH), Department of Signals, Sensors & Systems, Signal Processing, December 2000. IR-SB-EX-0023
32. Lundin, H.: Post-correction of analog-to-digital converters. Technical report, Royal Institute of Technology (KTH), May 2003. Licentiate Thesis
33. Lipshitz, Stanley P., Wannamaker, Robert A., Vanderkooy, John: Quantization and dither: a theoretical survey. J. Audio Eng. Soc **40**(5), 355–375 (1992)
34. Medawar, S., Handel, P., Bjorsell, N., Jansson, M.: Input-dependent integral nonlinearity modeling for pipelined analog digital converters. *IEEE Trans. Instrum. Meas.* **59**(10), 2609 – 2620 (2010)
35. Medawar, S., Handel, P., Bjorsell, N., Jansson, M.: Postcorrection of pipelined analog digital converters based on input-dependent integral nonlinearity modeling. *IEEE Trans. Instrum. Meas.* **60**(10), 3342 –3350 (2011)
36. Moulin, D.: Real-time equalization of A/D converter nonlinearities. In: *Proceedings of the IEEE International Symposium on Circuits and Systems*, vol. 1, pp. 262–267. IEEE, Portland (1989)
37. Mikulik, Pavol, Šaliga, Ján: Volterra filtering for integrating ADC error correction, based on an a priori error model. IEEE Trans. Instrum. Meas. **51**(4), 870–875 (2002)
38. Oliver, B.M., Pierce, J., Shannon, C.E.: The philosophy of PCM. Proc. IRE **36**, 1324–1331 (1948)

39. Rebold, T.A., Irons, F.H.: A phase-plane approach to the compensation of high-speed analog-to-digital converters. In: *Proceedings of IEEE International Symposium on Circuits and Systems*, vol. 2, pp. 455–458 (1987)
40. Roberts, L.G.: Picture coding using pseudo-random noise. *IRE Trans. Inf. Theory* 8, 145–154 (1962)
41. Schuchman, Leonard: Dither signals and their effect on quantization noise. IEEE Trans. Commun. Technol. **12**(4), 162–165 (1964)
42. Schetzen, M.: Theory of pth-order inverses of nonlinear systems. *IEEE Trans. Circ. Syst.* 23(5), 285–291 (1976)
43. Sheppard, W.F.: 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)
44. Anekal B. Sripad and Donald L. Snyder. A necessary and sufficient condition for quantization errors to be uniform and white. *IEEE Trans. Acoust. Speech Signal Process.* **25**(5), 442–448 (1977)
45. IEEE. *IEEE Standard for Terminology and Test Methods for Analog-to-Digital Converters*. IEEE Std. 1241. 2000
46. Tsimbinos, J., Lever, K.V.: 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*, 379–383 June 1993
47. Tsimbinos, J., Lever, K.V.: Computational complexity of Volterra based nonlinear compensators. Electron. Lett. **32**(9), 852–854 (1996)
48. Tsimbinos, J., Lever, K.V.: 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*, vol. 2, pp. 861–864 August 1996
49. Tsimbinos, J., Lever, K.V.: Improved error-table compensation of A/D converters. IEEE Proc.—Circ. Devices Syst. **144**(6), 343–349 (1997)
50. Tsimbinos, J.,Marwood, W., Beaumont-Smith, A., Lin, C.C.: Results of A/D converter compensation with a VLSI chip. In: *Proceedings of Information, Decision and Control, IDC 2002*, 289–294 Feb 2002
51. Tsimbinos, J.: *Identification and Compensation of Nonlinear Distortion*. PhD thesis, School of Electronic Engineering, University of South Australia (1995)
52. Widrow, Bernard: A study of rough amplitude quantization by means of Nyquist sampling theory. IRE Trans. Circuit Theory **3**(4), 266–276 (1956)
53. Widrow, B.: Statistical analysis of amplitude-quantizer sampled-data systems. *Trans. AIEE Part II: Appl. Ind.* **79**(52), 555–568 (1961)
54. Widrow, B., Kollár, I.: Quantization Noise: Roundoff Error in Digital Computation, Signal Processing, Control, and Communications. Cambridge University Press, Cambridge, UK (2008)
55. Widrow, Bernard, Kollár, István, Liu, Ming-Chang: Statistical theory of quantization. IEEE Trans. Instrum. Meas. **45**(2), 353–361 (1996)
56. Wannamaker, R.A., Lipshitz, S.P., Vanderkooy, J.,Wright. J.N.: A theory of nonsubtractive dither. *IEEE Trans. Sign. Proces.* **48**(2), 499–516 (2000)

Chapter 9
A/D Conversion with Non-uniform Differential Quantization

Dušan Agrež

Abstract The result of measurement has an uncertain value-band depending on the dynamic behavior of the measured object and the measurement instrumentation. Digitalization as the main part of measurement consists of prefiltering, sampling, windowing, and quantization. In addition to this, the dynamics of the analogue-to-digital (A/D) conversion are also important. Here a trade-off between the number of references for generating the reference levels and the number of steps of the conversion is presented. On the one hand, the pure parallel techniques have considerable redundancy of the used reference levels for estimation of the current measurement value and its change; on the other hand, the successive-approximation techniques are relatively slower and they have no possibility of adaptation to the signal until the end of the conversion [1]. Serial-parallel A/D techniques have similar structures as differential pulse-code converters. Numerical value is attained with a successive approximation of the difference between the reference and the measured quantity, g_r and g respectively. Estimation of the difference between two levels of signal Δg is possible with at least two sampling pulses. The dynamic error is proportional to that difference. In the differential multi-step A/D techniques, the change of signal Δg in the time between two sampling points Δt is tracked by a suitably large step y of the reference quantity, $y \propto \Delta g = g'(t)\Delta t$. Considering the presumption of dynamic error, a larger uncertainty can be added to a larger step. It is reasonable to use a finite set of possible step representatives $(y_i, i = 1, 2, \ldots)$. They have to be selected in such a way that their quantization intervals Δ_i ($\propto$ uncertainty intervals) have a minimum overlap. At the same time there should be no empty space between the quantization intervals. For effectiveness of differential tracking, the non-uniform quantization must fulfill three conditions: partitions into halves, increasing quantization uncertainty with difference, and low overlapping of the quantization intervals. The best trade-off between the number of decision levels and the settling time is with the pure exponential quantization rule: $y_i = a^{i-1}\Delta_0 = 2^{i-1}\Delta_0$, where $\Delta_0 = R/2^b$ is the

D. Agrež (✉)
Faculty of Electrical Engineering, University of Ljubljana, Ljubljana, Slovenia
e-mail: dusan.agrez@fe.uni-lj.si

P. Carbone et al. (eds.), *Design, Modeling and Testing of Data Converters*, Signals and Communication Technology, DOI: 10.1007/978-3-642-39655-7_9,

smallest quantization interval ($i = 1, 2, \ldots, b$, R—full scale of the measurement range, b—a number of bits). The fastest response is achievable with base 2. The principle that the larger the distance between two levels Δg is, the larger also the dynamic error of estimation is, shows the limitation of the information flow. In comparison with uniform quantization, the information H of the first step does not increase with an increase in the number of bits, b. It is limited to the constant value $H(a = 2)_{b\to\infty} = 3$. The quantity of information decreases when approaching the last step of conversion. The limited band of the measurement channel or more precisely the finite impulse response of the sampling device, which is not infinitely short, is the cause of the finite information capacity of the measurement channel. The number of bits required for the conversion decreases towards the end of conversion, and the sharp stop of the uniform constant information flow is smoothed. The differential tracking b-bit A/D conversion gives better results than the classical A/D conversion with the successive approximation procedure, due to b-times more available sampling points, and the adaptive property of the A/D procedure that every previous approximation step to signal becomes the center of observation with an exponential increase in resolution in the new step. Taking into consideration only the quantization noise contribution, the adaptive A/D conversion provides better results if the sampling ratio $s = f(\text{sampling})/f(\text{signal})$ is high enough. Considering together the systematic and the random errors in the signal parameters estimation, shows the advantage of the adaptive A/D conversion at lower values of the sampling ratio, s [2].

Keywords Measurement dynamics · Differential tracking · Exponential quantization · Settling time · Limited information flow · Quantization noise

9.1 Introduction

Estimation of the measured quantity in a technical system has the discrete nature in the amplitude and time dimensions. In the majority of different structures of A/D converters [1, 3, 4], a numerical value of the measured quantity is attained by reducing the error E between the auxiliary reference quantity g_r and the real value of the measured quantity g (Fig. 9.1: $E_g = g_r - g$).

The residual error is reduced through the estimation procedure in steps (Fig. 9.2: Δt_k is time of one step) and it finally attains the basic resolution $\Delta_0 = R_g/2^b$ of b-bit conversion in the input dynamic range R_g. Here a trade-off between the number of references for generating the reference levels x_i and the number of steps of conversion is presented. The fastest A/D converters use the pure one-step parallel (or flash) method or a few-step sub ranging method [3, 5]. The flash converter in which the input analog signal is presented to a bank of $2^b - 1$ reference levels and belonging comparators, has the advantage that no front-end sample-and-hold circuit is required. However, the exponential increase in the

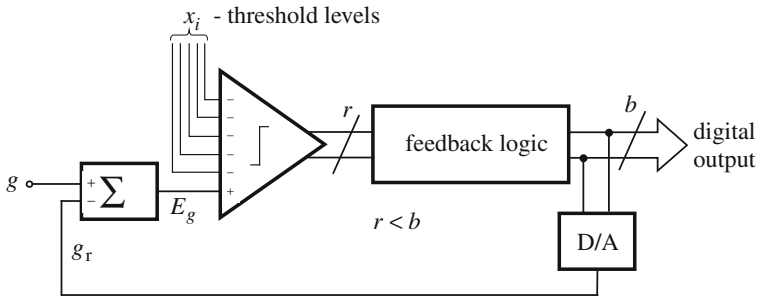

Fig. 9.1 Differential tracking converter

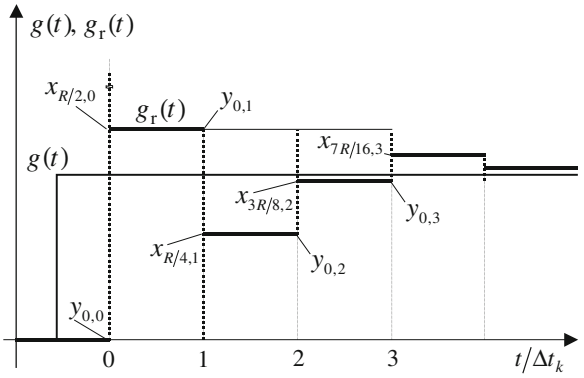

Fig. 9.2 Step response of the successive approximation procedure of the error reduction between signal g and reference quantity g_r

number of the reference levels (as a function of b) limits the realization of this approach as the value of b increases (practical realizations are up to 9–10 bits). A slower class of digitizer in common use is the feedback A/D converter (Fig. 9.1). Here the analog input signal is compared with the output of a D/A converter. The comparator output is then used to control the feedback logic in such a way as to equalize the D/A output and the input signals. Perhaps the simplest method of this class is the tracking type with fixed uniform quantization, in which the feedback logic comprises a digital up/down counter clocked at a constant rate, and whose count direction is controlled by the comparator output [3]. This technique is quite effective for slowly varying input signals whose rate of change can be more than matched by the counting rate of the feedback logic. However, shortcomings arise for rapidly changing input signals; for example, a full-scale step input would take $2^b - 1$ clock intervals to track. The tracking A/D converter requires no sample-and-hold stage. The speed limitation of the tracking A/D converter (slope overload) is partially overcome by the successive approximation type. Successive approximation A/D converters require a sample-and-hold stage. Here each bit is compared in turn commencing with the most significant bit (Fig. 9.2: $x_{*,k}$ are threshold levels of the examination intervals, which become the

origin representatives $y_{0,k}$ in the next successive steps k). The total conversion time is now fixed to b clock intervals Δt_k.

Optimal results for the A/D conversion with regard to the time of conversion, resolution, and used references are obtained with multi-step parallel techniques [5]. As they estimate the residual error with more threshold levels then the pure successive approximation procedure with one threshold level in every step (Fig. 9.2), they are faster, and at the same time they need a lower number of references for threshold levels in comparison with the pure one-step parallel flash method, due to the higher number of steps in the estimation procedure.

Serial-parallel A/D techniques have similar structures as differential pulse-code converters. Numerical value is attained with a successive approximation of the difference between the reference and the measured quantity. Estimation of the difference between two levels is possible with at least two sampling pulses, and if the amplitude difference is larger, then the estimation error is also larger due to the pulse leakage, and this will be shown in the following analysis.

Before quantization the band-limited signal $g(t)$ with the maximal frequency f_m has to be sampled by a sampling function where we have to consider the finite sampling frequency $1/t_s = f_s < \infty$ [6, 7] and the finite-time sampling pulses $h(t) \neq \delta(t)$ [8, 9]. In this acquisition process one can theoretically obtain only a finite number of samples by multiplication of the signal with the finite-time measurement interval $\Theta(t) < \infty$ [10–12] (Fig. 9.3a). These operations perform filtering on the signal as can be better seen in the frequency domain (Fig. 9.3b).

Considering real circumstances in the sampling process shows us the problem of the time resolution of sampling pulses. They carry the information of the amplitude of the measured signal. Distinguishing pulses in the time space is the basis for distinguishing the amplitude values of the measured signal. If they overlap too much, then the amplitude values are not resolved. Distinguishing them is possible by the relative shortening of the sampling pulses to the sampling interval t_s so that the leakage tails of the pulses decline sufficiently. However, there

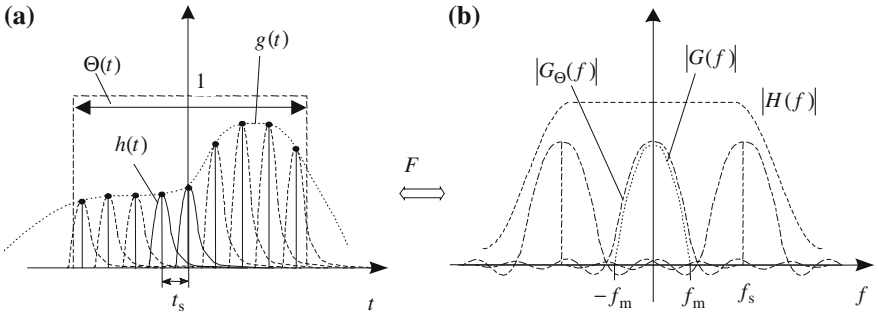

Fig. 9.3 Signal sampling process waveforms in the time domain (**a**) and the frequency domain (**b**) where $|G(f)|$ is the Fourier spectrum amplitude part of the signal before sampling and windowing, $|G_\Theta(f)|$ is the Fourier spectrum amplitude part of the windowed signal, and $|H(f)|$ is frequency spectrum of the sampling pulses

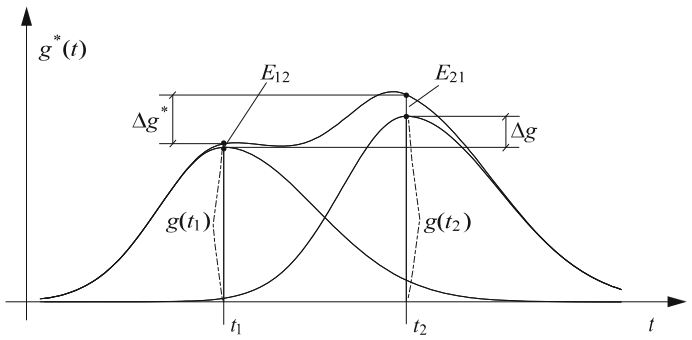

Fig. 9.4 Dynamic error of the amplitude difference estimation

will always remain dynamic measurement errors caused by the finite disposal energy in the real sampling.

During the successive approach towards the real value of the measured signal, the distance between the current state and the searched value is estimated in every step. This is made possible with at least two sampling pulses. We can distinguish two pulses when both of them decrease to less than a half of their original peak values at the middle time point between the time origins of pulses (Fig. 9.4). When two peaks in the common sampling signal are perceived, the pulses are distinguished, but the amplitude of the pulses $g^*(t)$ at their sampling origins t_1 and t_2 are not equal to the real values $g(t_i)$ of the measured signal. They are generally distorted by the leakage parts, E_{12} and E_{21}.

$$\begin{aligned} g^*(t_1) &= g(t_1) + E_{12} = g(t_1) + k_{12}g(t_2), \\ g^*(t_2) &= g(t_2) + E_{21} = g(t_2) + k_{21}g(t_1) \end{aligned} \tag{9.1}$$

With two dynamically distorted pulses as carriers of the information of the sampled signal, it is possible to estimate the distorted amplitude difference Δg^*. The dynamic error of estimating the difference of the two amplitudes $\Delta g = g(t_2) - g(t_1)$ is proportional to that difference with the addition of some constant:

$$\begin{aligned} E_{\Delta g} &= \Delta g^* - \Delta g = g^*(t_2) - g^*(t_1) - g(t_2) + g(t_1) = k_{21}g(t_1) - k_{12}g(t_2) \\ &= -k_{12}\Delta g + g(t_1)(k_{21} - k_{12}) \end{aligned} \tag{9.2a}$$

or generally:

$$E_{\Delta g} = k_a \Delta g + k_b \tag{9.2b}$$

If the sampling pulses have some symmetry in the pulse shape $k_{12} \approx k_{21}$, the error is proportional to the amplitude difference $E_{\Delta g} \approx k_a \Delta g$.

Considering the presumption of dynamic error, a larger uncertainty $\Delta_y/2$ can be added to a larger step and the change of signal is tracked by a suitably large step of the reference quantity, $y_i \propto g'(t)\Delta t$. Using the dynamic error property of the amplitude difference estimation in the quantization, one should increase the

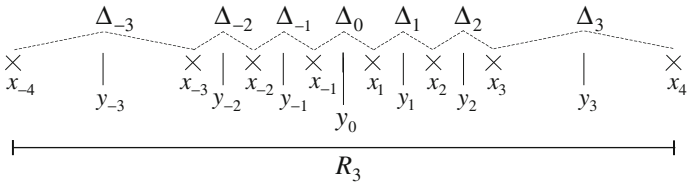

Fig. 9.5 Representatives $y_i = 2^{i-1}\Delta_0$ and threshold levels x_i of the non-uniform exponential quantization in the measurement range R_3 for base $a = 2$ and $i = 1, 2, 3$

quantization uncertainty of the increased values of the representatives y_i with the distance from the origin of the examination y_0 (Fig. 9.5). The representatives y_i and the associated threshold levels x_i can be arranged in a non-uniform exponential rule with the increased quantization resolution at the origin of the examination of the residual error [2, 4, 13] (Fig. 9.5).

The purpose of this chapter is to present a means of processing the error signal, i.e., the difference between the reference value of the feedback loop in the digitizer, and the input signal with the non-uniform differential quantization which yields the following performance features:

(a) the conversion rate for an arbitrary input step is faster than that achieved with successive approximation;
(b) data is valid during all clock intervals with different quantization error;
(c) a sample-and-hold stage is not required; and
(d) the quantization noise is comparable to the uniform one if the sampling ratio is high enough.

9.2 Non-uniform Quantization

Non-uniform quantization in contrast to the uniform one introduces the problems of positioning and spacing the representatives, and the threshold decision levels of the quantization intervals in the measurement range of the A/D converter. Their arrangement can depend on the expected probability density function (*pdf*) of the input signal, as with well-known 'μ-law' and '*A*-law' compression in telecommunications systems [3, 14], or on the logical structure of the presentations as in the floating-point quantization in terms of binary numbers [14]. Because there are always dynamic errors in the sampling process, it is reasonable to append the larger quantization intervals to the estimations of the larger amplitude difference. These intervals should contain the real value represented by the corresponding representatives. The optimum spacing of the levels of representatives in the quantizing intervals is achieved when they are centers of their intervals, or better centers of the probability density function of the measured signal appearing in their intervals [4, 15].

With the differential A/D techniques, the change of signal is tracked by a suitably large step $y \propto g'(t)\Delta t$ of the reference quantity. It is reasonable to use a finite set of possible step representatives $\{y_i, i = 1, 2, \ldots\}$. They have to be selected in such a way that their quantization intervals ($\propto$ uncertainty intervals) Δ_i have a minimum overlap. At the same time there should be no empty space between the quantization intervals.

Using the above conclusions, it is possible to specify the levels of representatives y_i and the decision levels x_i, which fix the quantization intervals $\Delta_i = x_{i+1} - x_i$. For an effective approach to the real measured value, the non-uniform quantization must fulfill three conditions:

(a) the representative must lie in the middle of the quantization interval [16, 17], especially if it becomes the origin during the next step of approaching;

$$y_i = \frac{x_{i+1} + x_i}{2} \tag{9.3}$$

(b) increasing the distance between the representative y_i and the current starting point y_0, causes also increasing of the quantization uncertainty $\pm\Delta_i/2$ for that representative ($y_{i+1} - y_0 \geq y_i - y_0 \Rightarrow \Delta_{i+1}/2 \geq \Delta_i/2$) according to (9.2a) and (9.2b);

$$y_{i+1} - x_{i+1} = \frac{\Delta_{i+1}}{2} \geq \frac{\Delta_i}{2} = x_{i+1} - y_i \Rightarrow y_{i+1} \geq 2x_{i+1} - y_i \tag{9.4}$$

(c) quantization intervals Δ_i should cross each other as low as possible. At the same time they must not leave empty spaces on the quantization scale [18]. The distance between the representative y_i of the quantization interval and the first decision level x_i of that interval must not be larger than the distance between the starting point y_0 and x_i. In the opposite case, the successive approximation steps of the A/D conversion alter their values for the difference y_i, if the searched value lies between $y_0 + y_i - x_i$ and $y_0 + x_i$.

$$|y_i - x_i| \leq |x_i - y_0| \tag{9.5}$$

The first condition defines that the best position of the representative is the middle of the quantization interval, if the representative serves as a decision level in the next step of approximation. If this condition is fulfilled, then the relation between the representatives and the decision levels is fixed. The quantization characteristic depends on the pattern of representatives.

The minimum step of the representative's arrangement can be derived from the first ($y_i = (x_{i+1} + x_i)/2 \Rightarrow x_{i+1} = 2y_i - x_i$) and the second condition ($y_{i+1} \geq 2x_{i+1} - y_i \Rightarrow (y_{i+1} + y_i)/2 \geq x_{i+1}$).

$$\begin{aligned} &\frac{y_{i+1} + y_i}{2} \geq x_{i+1} = 2y_i - x_i \quad \text{and} \quad \frac{y_i + y_{i-1}}{2} \geq x_i \\ &\frac{y_{i+1} + y_i}{2} \geq 2y_i - x_i \geq 2y_i - \frac{y_i + y_{i-1}}{2} \end{aligned} \tag{9.6}$$

After rearranging, it can be written as follows:

$$y_{i+1} \geq 2y_i - y_{i-1} \text{ or } y_{i+2} \geq 2y_{i+1} - y_i \tag{9.7}$$

The maximum step spacing between y_{i+1} and y_i can be obtained from the conditions ($y_i - x_i \leq x_i - y_0; y_0 = 0 \Rightarrow y_i \leq 2x_i$) and $x_{i+1} = 2y_i - x_i$ by induction:

$$\frac{y_{i+1}}{2} \leq x_{i+1} \leq 2y_i - \frac{y_i}{2} \Rightarrow y_{i+1} \leq 3y_i \tag{9.8}$$

Independently of the pattern function, the successive representatives of one step must be situated within the following borders:

$$3y_i \geq y_{i+1} \geq 2y_i - y_{i-1} \tag{9.9}$$

Considering equality in (9.7) the minimum step is defined. The first two representatives are determined by $y_0 = 0$ as origin, and $y_1 = 1\Delta_0$ as the smallest step of quantization. Other higher representatives can be obtained as follows:

$$y_2 = 2y_1 = 2\Delta_0; \quad y_3 = 2y_2 - y_1 = 3\Delta_0; \quad y_4 = 2y_3 - y_2 = 4\Delta_0; \ldots \tag{9.10}$$

With the minimum spacing of y_{i+1} and y_i, the representatives are uniformly spaced and the A/D conversion is a pure parallel one.

9.2.1 Exponential Distribution of Representatives

The selection of the distribution function of representatives depends on the boundary conditions (9.3), (9.4), and (9.5). The exponential distribution function has several advantages, such as: simplicity of the mathematical implementation, and exponential emphasis on the surroundings of origin [19].

$$y_i = a^{i-1}\Delta_0 \quad i = 1, 2, \ldots, b \tag{9.11}$$

There must be symmetry around the origin y_0 to attain effectiveness of quantization—minimum distortion and the fastest response of conversion. Index i of representatives obtains a negative sign, but their distance to y_0 remains the same $|y_i - y_0| = |y_{-i} - y_0|$ (Fig. 9.6).

Independently of the base a, the nearest representative from the origin with index $i = 1$ is for the smallest quantization interval Δ_0 apart. The lower bound of the base can be easily obtained from (9.7) with $y_1 = 1\Delta_0$ and $y_0 = 0$.

$$a^1\Delta_0 = y_2 \geq 2y_1 = 2 \cdot \Delta_0 \Rightarrow a \geq 2 \tag{9.12}$$

The upper bound from (9.8) is 3.

$$3 \geq \frac{y_i}{y_{i-1}} = \frac{a^{i-1}\Delta_0}{a^{i-2}\Delta_0} = a \tag{9.13}$$

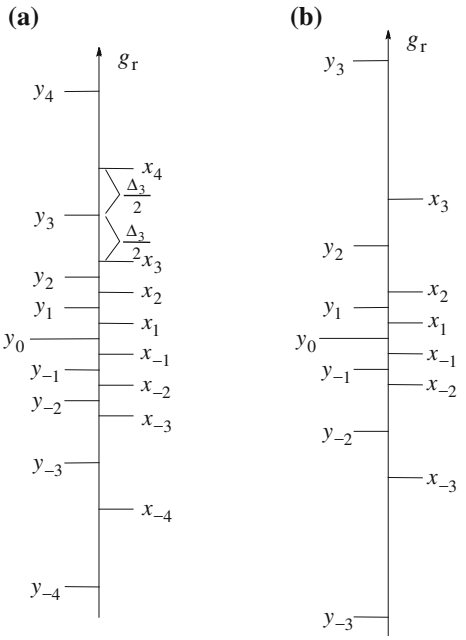

Fig. 9.6 Examples of the bias pattern of the representative levels; **a** $a = 2$ and **b** $a = 3$

The minimum value of the base for the pure exponential quantization is 2 (9.12). If we want to fulfill the condition under b and also for $a < 2$, it is possible to uniformly distribute representatives from the origin, to the representative where the associated quantization interval becomes larger than Δ_0 (9.10). From this point onwards the quantization interval Δ_i should increase with proportion to the exponent on a.

The first threshold-decision level x_1 must lie in the middle of y_1 and y_0. This assures that the smallest quantization interval Δ_0 is that around the origin.

$$y_1 - x_1 = \frac{\Delta_1}{2} \geq \frac{\Delta_0}{2} = x_1 - y_0 = x_1 \Rightarrow \frac{y_1}{2} \geq x_1 \quad \text{and} \quad x_1 = \frac{\Delta_0}{2} \tag{9.14}$$

Other decision levels can be obtained by induction if the generic function $x_{i+1} = 2y_i - x_i$ (9.3), x_1, and y_1 are known.

$$x_2 = 2y_1 - x_1 = \left(2a^0 - \frac{1}{2}\right)\Delta_0$$

$$\vdots$$

$$\frac{x_i}{\Delta_0} = (-1)^i 2\left[(-a)^{i-2} + \cdots + (-a)^2 + (-a)^1 + (-a)^0\right] - (-1)^i \frac{1}{2}$$

In the square brackets there is a sum of the exponential series $1 + a + a^2 + \cdots + a^n = (1 - a^{n+1})/(1 - a)$ and the expression for x_i can be written as:

$$x_i = (-1)^i \left[2\frac{1-(-a)^{i-1}}{1-(-a)} - \frac{1}{2} \right] \Delta_0 \tag{9.15}$$

The distance of y_i from y_0 increases the uncertainty of the representative $\Delta_i/2$ or the quantization interval Δ_i. The quantization interval is fixed with the difference of adjacent thresholds.

$$\begin{aligned} \frac{\Delta_i}{\Delta_0} &= \frac{(x_{i+1} - x_i)}{\Delta_0} \\ &= (-1)^{i+1} \left[2\frac{1-(-a)^i}{1+a} - \frac{1}{2} \right] - (-1)^i \left[2\frac{1-(-a)^{i-1}}{1+a} - \frac{1}{2} \right] = \frac{2a^{i-1}(a-1)}{1+a} + (-1)^{i+1}\frac{3-a}{1+a} \end{aligned} \tag{9.16}$$

9.3 Settling Time

Emphasizing the dynamics of response, the base that enables the shortest settling time of the A/D conversion to the signal step function at the input can be investigated (Fig. 9.7). The speed of the approximation procedure for different structures of A/D converters is measured by the number of steps required to achieve the basic resolution around the searched value. The number of steps of the A/D conversion with an exponential differential quantization depends on the value of the signal step. The larger the signal change is, the larger the number of approximation steps needed to within $\pm\Delta_0/2$.

Many A/D structures have to fulfil the condition that the uncertainty of the previous step must be smaller than current step plus the uncertainty of the current step. At the beginning of the conversion process the uncertainty is the approximation of the signal step. Being $y_i(k)$ the step k representative with index i, $\Delta_i(k)/2$ it's uncertainty (9.18), $y_j(k-1)$ the previous step representative with index j, and $\Delta_j(k-1)/2$ it's uncertainty, this leads to the following condition:

$$y_i(k) + \frac{\Delta_i(k)}{2} \geq \frac{\Delta_j(k-1)}{2} \tag{9.17}$$

Figure 9.7 gives a practical representation of the condition fulfilment. The representative y_i denotes all values $g(t)$ in the interval Δ_i $(y_i - \Delta_i/2 < (g - y_0) \leq y_i + \Delta_i/2)$.

The difference between representatives' indexes $l = j - i$ $(j > i;$ $i = 0, 1, \ldots, j-1;\ j = 1, 2, \ldots, b)$ in the successive steps $(j(k-1) \rightarrow i(k))$ represents the reduction of the approximation representative y_i and the reduction of the examination range Δ_i in the procedure of approaching to the measured value. Figure 9.7 shows the reduction of the approximation representative from y_7 in step $k-1$ to representative y_5 in the next k step with $l = 7 - 5 = 2$ in the case of signal

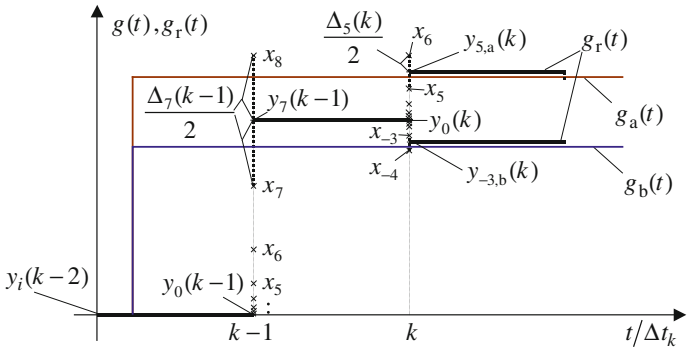

Fig. 9.7 Approximation procedures of the non-uniform differential quantization to the signals with two different steps ($g_a(t)$ *red* and $g_b(t)$ *blue*)

$g_a(t)$ and the reduction to representative $|y_{-3}| = y_3$ in the k step with $l = 7 - 3 = 4$ in the case of signal $g_b(t)$. A larger value of l ($l = 1, 2, \ldots$) means a larger reduction of the quantization uncertainty between the steps and a faster response. Condition (9.17) can be rearranged $\Delta_i(k) + 2y_i(k) \geq \Delta_j(k-1)$ and expressed by the terms of y_i (9.11) and Δ_i (9.16).

$$\frac{2a^{i-1}(a-1)}{a+1} + (-1)^{i+1}\frac{3-a}{1+a} + 2a^{i-1} \geq \frac{2a^{j-1}(a-1)}{a+1} + (-1)^{j+1}\frac{3-a}{1+a} \tag{9.18}$$

The solution for a^l ($l = j - i$) can be written as:

$$a^l \leq \frac{2a}{a-1} - (-1)^i \frac{3-a}{2(a-1)} a^{-i+1}\left(1 - (-1)^l\right) \tag{9.19}$$

The difference l depends on the base a and index i. To resolve equation (9.19) the value of l should be defined and the solution for a is found:

$$l = 1: \quad a \leq \frac{2a}{a-1} - (-1)^i \frac{3-a}{a-1} a^{-i+1} \Rightarrow 1 < a \leq 3 \tag{9.20}$$

If the base a is between 1 and 3 (9.20), the index i is reduced during the successive steps for at least $l = 1$.

$$l = 2: \quad a^2 \leq \frac{2a}{a-1} \Rightarrow 1 < a \leq 2 \tag{9.21}$$

In this case index i is reduced at least for 2. For the pure exponential distribution the value for base $a = 2$ is the only solution. Examination intervals of the successive steps are certainly reduced with the index change of $l = 2$ ($\Delta_{i+2}(k-1) \rightarrow \Delta_i(k)$) (Fig. 9.7).

$$l = 3: \quad a^3 \leq \frac{2a}{a-1} - (-1)^i \frac{3-a}{a-1} a^{-i+1} \Rightarrow 1 < a < 1.66 \tag{9.22}$$

$$l = 4: \quad a^4 \leq \frac{2a}{a-1} \Rightarrow 1 < a \leq 1.5437 \tag{9.23}$$

The results of the last two expressions (9.22) and (9.23), show that the index reduction l is larger with a decrease of the base and the response is faster, but the number of thresholds or the required references increases exponentially and in the limit $a = 1$ we get the pure parallel AD conversion with $r = 2^b - 1$ references for decision levels and the settling time of one step $k = 1$. Optimal results for the A/D conversion with regard to the used references and the time of conversion using criterion $r(\text{number of references}) \times k(\text{number of steps}) \rightarrow \min$, are obtained with through multi-step serial-parallel techniques [1, 3]. The best trade-off between the number of decision levels and the settling time is achieved with the pure exponential quantization rule $r \cdot k_{\max} \approx b^2$, where the number of decision levels can be approximated by $r \approx 2b$ due to b references on both sides of the origin and the maximal number of steps can be halved $k_{\max} \approx b/2$ if there is a reduction of representative index by 2 in the successive steps. The fastest response or the shortest settling time in the error band of the smallest quantization interval is achievable with the base $a = 2$ if we vary the base between 2 and 3 since only this base gives reduction of the representative index by 2. The exponential distribution function of representatives y_i with base $a = 2$ has several advantages: the fastest step response, and as mentioned, the simplicity of the mathematical implementation. As steps y_i are exponentially interspaced with base $a = 2$ (9.11), it is easy to logically implement one step by increment or decrement the value in the register of the feedback logic by 1 at the suitable weighted bin $(000_{\downarrow y_i} 1_{\uparrow y_{i-1}} 110\ldots)$ [19].

The problem of the worst case settling time is similar to the problem of the range width—the signal step, which can be examined at the fixed number of approximation steps k. For a base varying between $2 < a \leq 3$ the maximal possible examinations widths (Fig. 9.8) are:

$$\begin{aligned}
&k_{\max}: \Delta_0/2\\
&k_{\max} - 1: y_1 + \Delta_1/2\\
&\ldots;\\
&k_{\max} - i: \left[a^{i-1} + \left(a^{i-1}(a-1)\right)/(a+1) + (-1)^{i+1}(3-a)/(2(1+a))\right] \cdot \Delta_0 \text{— from (9.11)}\\
&\qquad\qquad \textit{and (9.16)}\cdot
\end{aligned} \tag{9.24}$$

For the base $a = 2$, two ways of expansion of the examination interval are possible, because there are two possibilities in reduction within the smallest quantization uncertainty $\pm\Delta_0/2$ in the last step ($k_{\max-1} \rightarrow k_{\max}$): y_1 with the preceding steps $\left(2^{2k+1} + (-1)^{2k}\right)\Big/6 \cdot \Delta_0 \cdots \rightarrow y_1 + \Delta_1/2 \rightarrow \Delta_0/2$ and y_2 with the successive steps $\left(2^{2k+2} + (-1)^{2k+1}\right)\Big/6 \cdot \Delta_0 \cdots \rightarrow y_2 + \Delta_2/2 \rightarrow \Delta_0/2$.

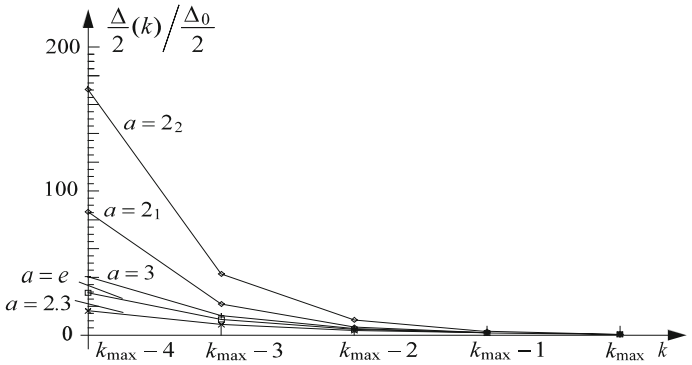

Fig. 9.8 The worst case settling to $\pm\Delta_0/2$ during the last four steps ($a = 2_1,\ 2_2,\ 2.3,\ e,\ 3$)

9.3.1 Probability Distribution of Steps

In the worst case of approaching the searched value, the index is reduced only for 2 ($a = 2$) or for 1 ($2 < a \le 3$). There are also positions of the searched signal to which the approximation steps y_i approach within the borders of $\pm\Delta_0/2$ and the approximation procedure is finished in one step. The question is: what is the probability of the appearance of the approaching path by a given number of steps k ($k_0 = 0, k_1 = 1, \ldots, k_{\max}$)? The probability density function of the measured signal $pdf(g) = f(g)$ is assumed to be constant in the measurement range as in many A/D conversion techniques. The increase of index i increases the uncertainty interval of the first step—the range of examination ($d_i = y_i + \Delta_i/2$, $i = 0,\ 1,\ 2, \ldots,\ b$). In these intervals the probability density functions are constant $f_i(g) = 1/d_i$ ($0 \le g \le d_i$). For the base $a = 2$ with the shortest worst case settling time (Fig. 9.8), the following relations are valid ($d_i = \left(2^{i+2} + (-1)^{i+1}\right) \Delta_0 \Big/ 6$):

$i = 0$: $d_0 = y_0 + \Delta_0/2 = \Delta_0/2$. In this trivial case the signal is positioned within $\Delta_0/2$ and the number of steps is 0.

$i = 1$ (Fig. 9.9a): $d_1 = y_1 + \Delta_1/2 = 3 \cdot \Delta_0/2$

$$\begin{aligned} \bar{k} &= p_0 k_0 + p_1 k_1 = \int_{y_0}^{x_1} f_1(g)dg \cdot k_0 + \int_{x_1}^{x_2} f_1(g)dg \cdot k_1 \\ &= f_1(g)\left(\frac{\Delta_0}{2} \cdot k_0 + 2\frac{\Delta_0}{2} \cdot k_1\right) = \frac{\Delta_0}{2d_1}(k_0 + 2k_1) = \frac{1}{3}(k_0 + 2k_1) = 0.666 \end{aligned} \tag{9.25}$$

The probability p_1 of path with one step is $2/3$ and without step p_0 is $1/3$ (Fig. 9.9a). The examination is finished with the uncertainty of $\pm\Delta_0/2$ and the average number of steps is 0.66.

$i = 2$ (Fig. 9.9b): $d_2 = y_2 + \Delta_2/2 = 5 \cdot \Delta_0/2$

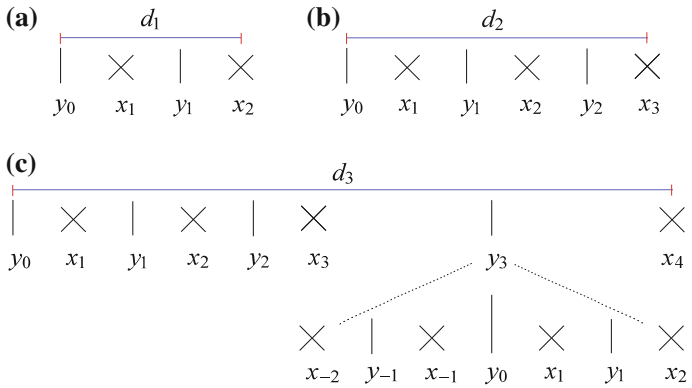

Fig. 9.9 Ranges of examination for the first three values of index i

$$\bar{k} = \int_{y_0}^{x_1} f_2(g)dg \cdot k_0 + \int_{x_1}^{x_2} f_2(g)dg \cdot k_1 + \int_{x_2}^{x_3} f_2(g)dg \cdot k_1 = \frac{1}{5}(k_0 + 4k_1) = 0.8 \tag{9.26}$$

For the probability of the path with one step ($p_1 = 4/5$), two equal length quantization intervals $\Delta_1 = x_2 - x_1$ and $\Delta_2 = x_3 - x_2$ participate in the expression (9.26) (Fig. 9.9b).

$i = 3$ (Fig. 9.9c): $d_3 = y_3 + \Delta_3/2 = 11 \cdot \Delta_0/2$

$$\bar{k} = \int_{y_0}^{x_1} f_3(g)dg \cdot k_0 + \int_{x_1}^{x_3} f_3(g)dg \cdot k_1 + 2\left(\int_{y_0}^{x_1} f_3(g)dg \cdot k_0 + \int_{x_1}^{x_2} f_3(g)dg \cdot k_1\right) \cdot k_1$$

$$= \frac{\Delta_0}{2d_3}(k_0 + 4k_1 + 2(k_0 + 2k_1)k_1) = \frac{1}{11}(k_0 + 4k_1 + 2k_{1+0} + 4k_{1+1})$$

$$\bar{k} = \frac{1}{11}(k_0 + 6k_1 + 4k_2) = \frac{1}{11}(1 \cdot 0 + 6 \cdot 1 + 4 \cdot 2) = 1.2727 \tag{9.27}$$

If y_3 is the approximation to the signal in the first step then one more step is possible to achieve $\pm\Delta_0/2$ (Fig. 9.9c). The term $k_{1+0} = k_1$ denotes the path with one step to all values in the interval $y_3 - \Delta_0/2 < g \le y_3 + \Delta_0/2$. The term $k_{1+1} = k_2$ is used for the path with two steps ($\rightarrow y_3 \rightarrow y_{+1,-1}$). The double probability at y_3 is obtained, because the approximation y_3 in the first step becomes the origin y_0 in the next step with symmetrically arranged thresholds x_i and x_{-i} around it.

$$i = 4 : d_4 = y_4 + \Delta_4/2$$

$$\bar{k} = \frac{\Delta_0}{2d_4}(k_0 + 6k_1 + 4k_2 + 2(k_0 + 4k_1) \cdot k_1) = \frac{1}{21}(k_0 + 8k_1 + 12k_2) = 1.5238 \tag{9.28}$$

With an increasing value of i, the average number of steps $\bar{k}$ can be written by the induction and the way of determining the path probability of the k steps ($k_{1+2} = k_3, \ldots$) should be considered.

$$i = 5: \quad \bar{k} = \frac{\Delta_0}{2d_5}\left(k_0 + 8k_1 + 12k_2 + 2\left(\overbrace{k_0 + 6k_1 + 4k_2}^{\text{from}(9.27)}\right) \cdot k_1\right)$$
$$= \frac{1}{43}(k_0 + 10k_1 + 24k_2 + 8k_3) = 1.9070 \tag{9.29}$$

$$i = 10: \quad \bar{k} = \frac{1}{1365}(k_0 + 20k_1 + 144k_2 + 448k_3 + 560k_4 + 192k_5) = 3.5546 \tag{9.30}$$

$$i = 11: \quad \bar{k} = \frac{1}{2731}(k_0 + 22k_1 + 180k_2 + 672k_3 + 1120k_4 + 672k_5 + 64k_6) = 3.8894 \tag{9.31}$$

$$i = 12: \quad \bar{k} = \frac{1}{5461}(k_0 + 24k_1 + 220k_2 + 960k_3 + 2016k_4 + 1792k_5 + 448k_6) = 4.2219 \tag{9.32}$$

An approximation of the arbitrary value with the quantization uncertainty of $\pm\Delta_0/2$ has the probability distribution of steps as is shown in Fig. 9.10. The average number of steps is limited to two thirds of the maximal value of steps ($i = 12 \rightarrow 2/3 \cdot k_{\max} = 2/3 \cdot 6 = 4$; $i = 10 \rightarrow 2/3 \cdot k_{\max} = 2/3 \cdot 5 = 3.33$; … etc.).

The maximum number of conversion steps $k_{\max}$ in the worst case of the non-uniform differential A/D conversion is half of that with the successive-approximation method ($k_{sa} = i = b$) and the average number is one third (Fig. 9.11 and Eqs. (9.26)–(9.32). The settling time is shortened three times on average in comparison to the successive-approximation method where we have b steps for b-bit A/D conversion.

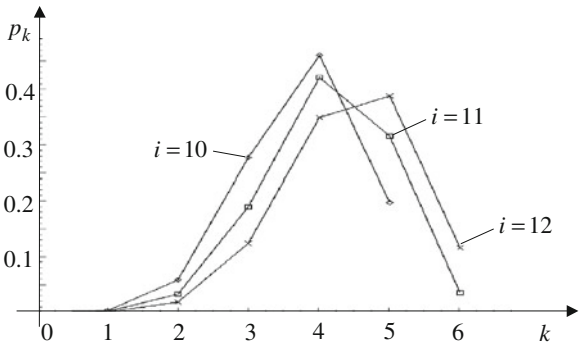

Fig. 9.10 Probability distributions of the k steps at $i = 10,\ 11,\ 12$ (normalized values from (9.30), (9.31), and (9.32))

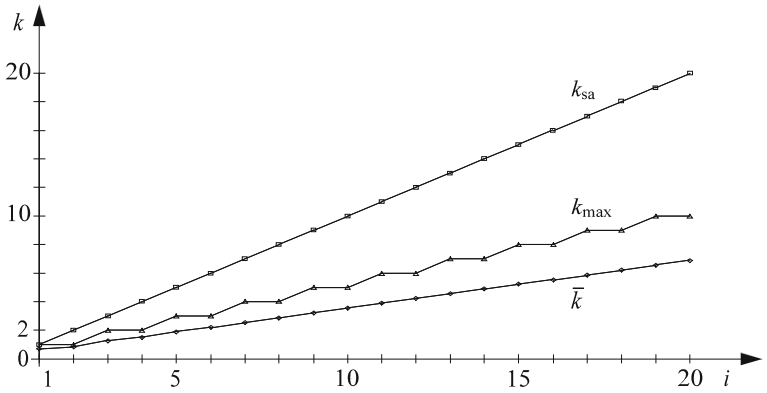

Fig. 9.11 Number of steps of A/D conversions: $k_{sa} = i$ is a number of steps of the successive-approximation A/D conversion; k_{max} and $\bar{k}$ are the maximal and the average number of steps of the non-uniform differential A/D conversion

9.3.2 Dynamics of One Step

The adaptive property of the A/D procedure that every previous approximation step becomes the center of observation with the exponential increasing resolution in the new step reduces the time of conversion (Fig. 9.11). The difference between the approximation and the real value is measured in every step, and it is not necessary to begin testing the most significant bits if the new value at the input of the converter does not differ greatly from the previous one. The value of the previous approximation $g_r(t_{k-1})$ holds the latch register in the logic of the feedback path (Figs. 9.1 and 9.7) until the estimation by the non-uniform quantization $\langle E_G \rangle = Q(E) = y_i$ of difference between the new value of the measured signal and the previous approximation is available after $\Delta t_k = t_k - t_{k-1}$.

$$g_r(t_k) = g_r(t_{k-1}) + y_i(t_k) \tag{9.33}$$

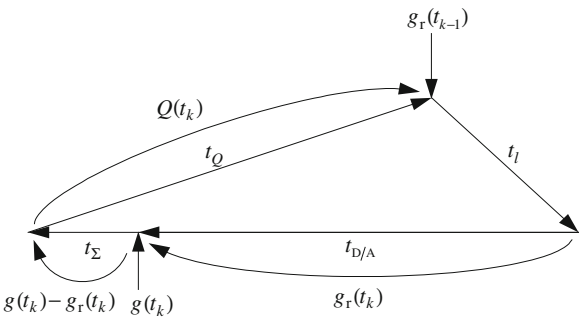

Fig. 9.12 Loop timing diagram of one approximation step

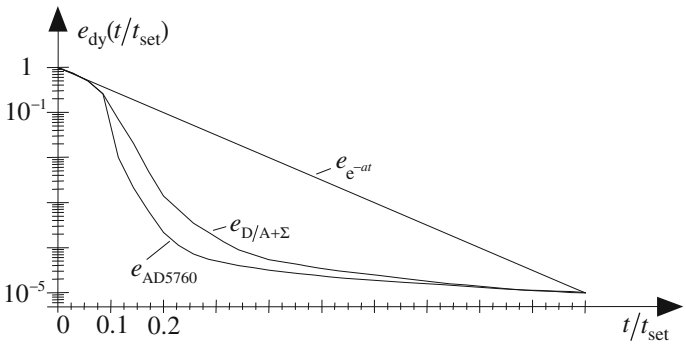

Fig. 9.13 Typical dynamic error e_{dy} in full scale step response (t_{set} settling time) of D/A converter and adder

In one step we have three distinct timing intervals: $t_{D/A}$—D/A converter settling time and t_Σ—adder response time; t_Q—quantizer response time and t_l—time to accumulate and latch (Fig. 9.12).

The most time consumption is $t_{D/A}$ with added t_Σ [20, 21] (Fig. 9.13).

9.4 Information Contents

A finite impulse response of the sampler, which has been considered in the procedure of the non-uniform quantization, also causes the finite information rate of the measurement channel. Quantization of a certain measurement interval gives a finite set of output levels or representatives N, and has a great accent on the rate-distortion theory [22]. At the initial state of the measurement approximation procedure, the uncertainty of the site of the searched signal is determined by the measurement range $R = g_{max} - g_{min}$. In our case, the representative of the measurement range lies in the middle and the uncertainty interval is the half of range. At the end of the measurement procedure, the uncertainty interval is reduced to the uncertainty of the resolution interval—the smallest quantization interval. Some information is gained in the Hartley sense $I = \log_2(N) = \text{lb}\, N$ [22–24] if one representative among a certain number of alternatives N is determined.

The uniform quantization equalizes the probabilities of individual representatives, assuming the constant probability density function of the searched signal in the measurement range $f(g) = 1/R$. With the non-uniform quantization in the A/D conversion procedure, the probabilities p_j of representatives are no longer equal, if $f(g)$ is constant. It is better to use the Shannon probabilistic definition of uncertainty—the Shannon entropy [14, 25].

$$H = -\sum_j p_j \mathrm{lb} p_j \tag{9.34}$$

The measurement range R, calibrated with the resolution interval, is the whole interval around the origin y_0 within the borders $\pm(y_i + \Delta_i/2)$ (Fig. 9.5).

$$R_i = 2d_i = 2y_i + \Delta_i = \left(a^{i-1}\frac{4a}{a+1} + (-1)^{i+1}\frac{3-a}{a+1}\right)\Delta_0 \tag{9.35}$$

If the absolute value of the range remains unchanged, then the smallest quantization interval relatively decreases by the increasing of i. The probability of the representatives occurrence is proportional to the width of the associated quantization interval.

$$p_j = \frac{\Delta_j}{R_i} \quad j = -i, \ldots, -1, 0, 1, \ldots, i \tag{9.36}$$

Now the entropy for the range R_i can be written:

$$H_i = -\left(p_0 \mathrm{lb} p_0 + 2\sum_{j=1}^{i} p_j \mathrm{lb} p_j\right). \tag{9.37}$$

The entropy depends on the value of index i, or in other words on the number of bits $b = i$ of the A/D conversion, and can be expressed by the quantization intervals as follows:

$$\begin{aligned} H_i &= -\left(\frac{\Delta_0}{R_i}\mathrm{lb}\left(\frac{\Delta_0}{R_i}\right) + 2\sum_{j=1}^{i}\frac{\Delta_j}{R_i}\mathrm{lb}\left(\frac{\Delta_j}{R_i}\right)\right) \\ H_i &= -\frac{\Delta_0}{R_i}\left(2\sum_{j=1}^{i}\frac{\Delta_j}{\Delta_0}\mathrm{lb}\left(\frac{\Delta_j}{\Delta_0}\right) - \mathrm{lb}\left(\frac{R_i}{\Delta_0}\right)\left(1 + 2\sum_{j=1}^{i}\frac{\Delta_j}{\Delta_0}\right)\right) \end{aligned} \tag{9.38}$$

In (9.38) the range and the quantization interval are normalized by the minimal quantization interval.

From (9.38), (9.35), and (9.16) it can be deduced that the increase of i, or the number of representatives, does not proportionally increase the entropy as is the case with the uniform quantization $H_{\max} = \mathrm{lb}\,(R_i/\Delta_0)$. It is limited to the constant value $(H(a=2) \underset{i\to\infty}{=} 3.0000\,\text{bits}, \quad H(a=2.3) \underset{i\to\infty}{=} 2.7475\,\text{bits}, \quad H(a=e) \underset{i\to\infty}{=} 2.5013\,\text{bits}, H(a=3) \underset{i\to\infty}{=} 2.3774\,\text{bits})$. The non-uniform exponential quantization with its limited information rate, shows one of the possible ways how a measurement system with a limited capacity of channel ($c = H/\text{step} = \text{const.}$) can be adapted to the increased information rate at the input ($i \to \infty$).

9.4.1 Expected Rate of Information in Successive Steps

The expected information of the first step is determined by the probabilities of possible representatives, which could become the origins in the next step of examination. The information content is expressed as the weighted arithmetic mean of uncertainties $-\operatorname{lb} p_j$, which is exactly the same as the Shannon entropy. Because the quantization intervals have no overlap, and the probability density function of the searched signal is assumed to be constant in the measurement range, the probabilities are proportional to the widths of quantization intervals.

During the next steps, the branching property of the Shannon entropy is used. In the second step, the entropy depends upon the probabilities of the paths, which have remained for examination to the final approximation within $\pm\Delta_0/2$, and upon the entropy that is obtained during the first steps of continuation on these paths.

As an illustration, the example of $i = 3\,,\ a = 2$ (Figs. 9.5 and 9.9c) is analyzed. According to (9.38), the entropy of the first step is ${}_1H = H_i = H_3 = 2.5949$ bit. In the second step, the intervals around representatives y_{-3} and y_3 $(\Delta_{-3}\,, \Delta_3 > \Delta_0)$ have remained for examination. They have equal probabilities $p_3 = \Delta_3/R_3$. The information that is obtained in one of these intervals is equal $H_i = H_1$ (9.38). The entropy for the second step is ${}_2H = 2p_3H_1 = 2 \cdot \Delta_3/R_3 \cdot H_1 = 2 \cdot 3/11 \cdot 1.5850 = 0.8645$ bit. The sum of entropies of both steps gives us the maximal information for the measurement range $R_3 = 11\Delta_0$ with the resolution of Δ_0.

$$H(i = 3) = {}_1H + {}_2H = 3.4594\,\text{bit} = \operatorname{lb}(11) \tag{9.39}$$

Similar considerations are also valid for larger values of i. The number of steps $k_{\max}$ is increased for 1 at odd values of the index ($k_{\max}(i = 1) = 1$, $k_{\max}(i = 3) = 2$, $k_{\max}(i = 5) = 3$,…).

The entropy of the first step of approximation is expressed by (9.38).

1. step:

$${}_1H = H_i \tag{9.40}$$

For the second step, the reflection from the example can be generalized:
2. step:

$${}_2H = 2(p_3H_1 + p_4H_2 + p_5H_3 + \cdots + p_iH_{i-2}) = 2\frac{1}{R_i}\sum_{j=3}^{i}\Delta_j H_{j-2} \tag{9.41}$$

During the remaining steps of the A/D conversion with the exponential quantization and base $a = 2$ the index i is reduced by 2 (in Appendix (A1)).
The entropy for the third step can be generally obtained with induction as in the example in Appendix.

3. step:

$$ {}_3H = 4\frac{1}{R_i}\sum_{j=3}^{i-2}(i-j-1)\Delta_j H_{j-2} \tag{9.42} $$

For the remaining higher steps, the entropy can be written as follows:
k. step:

$$ {}_kH = \frac{2^{k-1}}{R_i}\cdot\sum_{j=3}^{i-2(k-2)}\left[\frac{1}{(k-2)!}\prod_{m=0}^{k-3}(i-2k+5-j+m)\right]\Delta_j H_{j-2} \tag{9.43} $$

The entropy curve of the kth step in dependence of i monotonously increases from zero to three (Fig. 9.14). In (9.43) the expressions for $\Delta_{j=i}$ (9.16), R_i (9.35), and $H_{j=i}$ (9.38) are used.

If we now collect the entropy's contribution of several steps at the same index i (Fig. 9.14, vertical dotted lines), it appears that the information flow is not constant as in the uniform parallel-serial A/D conversions. The number of bits decreases towards the end of conversion and the sharp stop of the constant information flow is smoothed (Fig. 9.15).

At the beginning of the conversion procedure, the information rate per step is a little below 3. During the last two steps it decreases. Depending upon how much information is neglected, the last steps may be omitted. However, the sum of entropy contributions of all steps is equal to the maximal information contents of the measurement range calibrated by $\Delta_0 (H_{\max} = \mathrm{lb} R_i/\Delta_0)$.

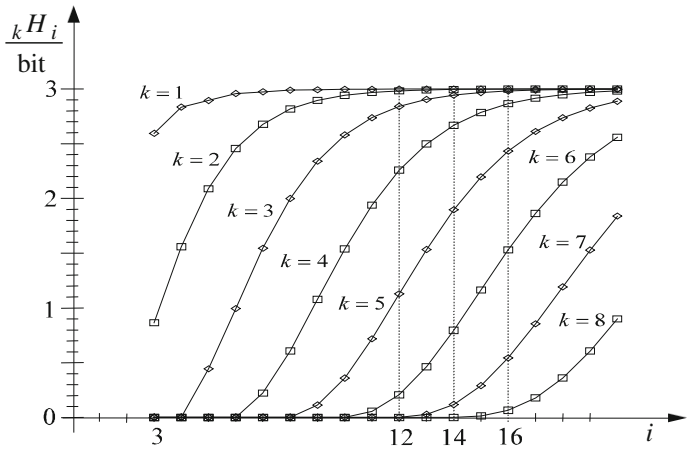

Fig. 9.14 Increasing of entropies in steps ($k = 1, \ldots, 8$)

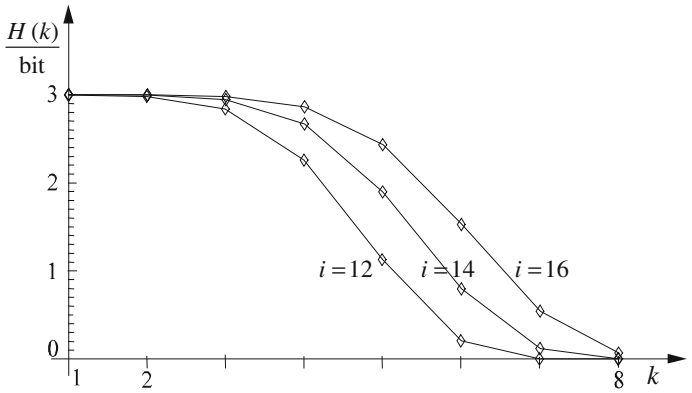

Fig. 9.15 Information rate per step during the conversion procedure ($i = 12,\ 14,\ 16$)

9.5 Quantization Noise

It is possible to use the adaptive A/D converter in two different ways: with and without the sample/hold device. In the first case, the converter proceeds with a non-linear approach of reducing uncertainty to the constant measured value (Fig. 9.7). Each conversion is finished with the quantization uncertainty of the smallest quantizing interval $\pm\Delta_0/2$. The adaptive A/D converter also works without sample/hold device (Fig. 9.16). Part of the sample/hold function takes over the latch register in the logic of the feedback path (Fig. 9.1). Tracking of the signal is achieved by the non-uniform estimation of the error difference $\langle E_G \rangle$ and by increasing the value of the reference quantity by a suitable step y_i. Estimation of the signal at time instant t_k has uncertainty of the quantization interval Δ_i, which

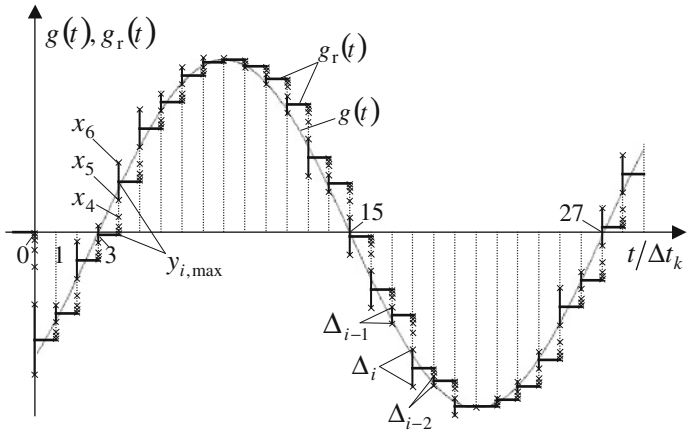

Fig. 9.16 Estimation and tracking of the sine shape signal with 24 samples in the period

belongs to y_i. Typically, they are the largest at the zero level crossings of the AC signals. The quality of tracking the signal, which can be evaluated by the quantization noise, depends mainly on the number of samples in one period $s = T/\Delta t_k$. The lower value of the sampling ratio s causes the larger signal change in one time increment Δt_k and consequently the larger non-uniform increment y_i with associated quantization uncertainty Δ_i (Fig. 9.16).

9.5.1 Tracking of the Square Signal

Tracking of the square signal is like using A/D converter with the sample/hold device (Fig. 9.17) if the sampling ratio is high enough.

Larger steps in tracking and belonging uncertainties $\Delta_i/\sqrt{12}$ also increases quantization noise [26] in comparison with the lowest value of the quantization standard uncertainty $\sigma_0 = \Delta_0/\sqrt{12}$ (Fig. 9.18). The relative increase of the quantization uncertainty was tested by the square signal with an amplitude of half of the maximal possible first step with belonging uncertainty $d_b = y_b + \Delta_b/2 = (1 + 0.333)R/2 = 1.333R/2$, $A = d_b/2 = 0.6661 \cdot R/2$ for the longest settling response. In this worst case scenario the tracking algorithm uses the maximal possible number of steps to attain the minimal quantization interval $k_{\max} = b/2$. Ten cycles of the signal were examined for $b = 10$ and $b = 12$ bit A/D converters, and the number of sampling points in one signal period was changed between $6 \le s \le 200$.

After settling time ($k_{\max} = b/2$) around the amplitude within the interval of $\pm\Delta_0/2$ all remaining samples of the half period are quantized by this minimal resolution, and the collective quantization uncertainty relatively decreases with an increase of the sampling ratio s. In Fig. 9.18, it can be seen that with a 12-bit AD converter the standard deviation is 600 times larger than σ_0 at $s = 2k_{\max} = 12$, with a 10-bit AD converter the ratio is about 100 at $s = 10$.

It can be seen that the quantization standard deviation of tracking the square signal differs from the expected standard deviation using the probability distributions of the approximation steps by about two times (Fig. 9.18: curves a and b

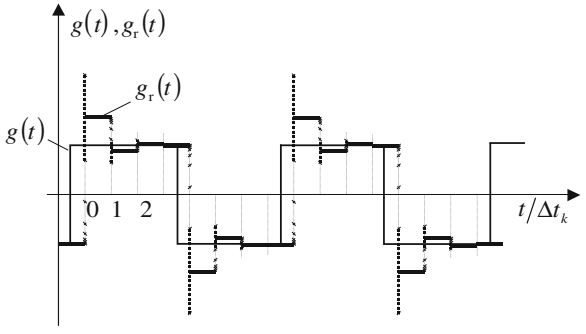

Fig. 9.17 Tracking of the square shape signal $s = 8$

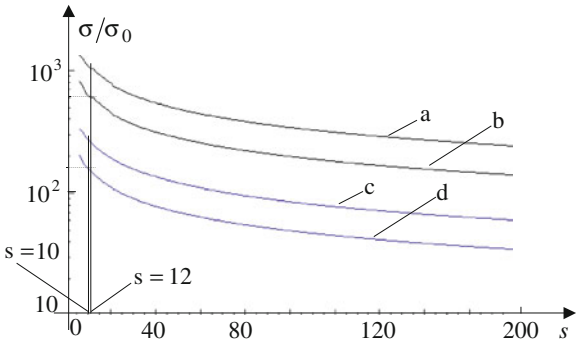

Fig. 9.18 Ratios of the standard deviations to $\sigma_0 = \Delta_0/\sqrt{12}$ in relation to the sampling ratio s in tracking the square signal: *a* with 12-bit A/D converter; *b* approximation using the probability distributions of the approximation steps of 12-bit A/D converter; *c* with 10-bit A/D converter; *d* approximation using the probability distributions of the approximation steps of 10-bit A/D converter

for 12-bit A/DC and c and d for 10-bit A/DC), due to the small number of samples in the particular quantization intervals and errors on the borders as we have used the worst case of the signal position. The ratios of the standard deviations are smaller with a 10-bit AD converter due to the relatively larger minimum quantization interval ($\Delta_0(b = 10) > \Delta_0(b = 12)$ if we fix the range R.

9.5.2 Tracking and Quantization of the Noisy Signal

From the previous section, it can be concluded that the adaptive tracking A/D converter searches for the largest values of the signal *pdf*. By adding noise to the DC signal the search behavior and robustness of the adaptive AD converter can be investigated. For this purpose, two shapes of white noise with rectangular and approximated Gaussian distributions were added to the DC signal with value $g_{DC} = -0.125 \cdot R/2$. For this value of signal the tracking algorithm needs only one settling step and all other steps depend only on the added noise ($N = 4000$). Both shapes of the noise have the same standard deviation σ_s and different amplitudes ($A_{noise}(\text{rect}) = \sqrt{3} \cdot \sigma_s$, $A_{noise}(\text{Gauss}) \approx 3.4 \cdot \sigma_s$). The noise standard deviation σ_s was changed in relation to the quantization standard uncertainty $\sigma_0 = \Delta_0/\sqrt{12} = 0.141 \cdot 10^{-3} \cdot R/2$ of the basic resolution $\Delta_0 = 0.488 \cdot 10^{-3} \cdot R/2$ for 12-bit AD conversion (x-axes in Fig. 9.19).

In Fig. 9.19a, a small difference of tracking the noise with different shapes can be seen. After the noise standard deviation crosses the value of minimal quantization standard uncertainty σ_0, the standard deviations of the A/D output become constant relative to σ_s ($\sigma_t/\sigma_s(\text{rect}) \approx 1.13$, $\sigma_t/\sigma_s(\text{Gauss}) \approx 1$). The difference is due to the wider shape of the rectangular noise *pdf*.

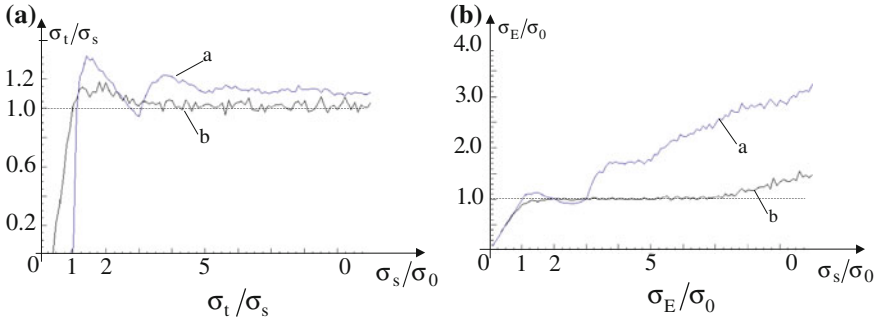

Fig. 9.19 Ratios of the standard deviations of the tracking values σ_t to the standard deviation of the input noise σ_s (**a**) and the standard deviations of the output errors ($\sigma_E = \sigma_{g_r - g}$) to the minimal quantization standard uncertainty σ_0 (**b**): *a* the rectangular noise distribution, *b* the Gaussian noise distribution

The quantization standard deviations of the errors $E = g_r - g$ increase with an increase of the input noise standard deviation (Fig. 9.19b). If the noise *pdf* shape is Gaussian then the output errors have almost the same values as with the uniform quantization with Δ_0 between $1 < \sigma_s/\sigma_0 < 7$. Both conclusions show the robustness of the non-uniform exponential quantization tracking procedure. Probabilities of the steps in tracking the noisy signals depend on the noise *pdf* shape (Fig. 9.20) and with the rectangular shape, larger steps contribute in the tracking (Fig. 9.20: curve a).

9.5.3 Tracking of the Sine Shape Signal

Increasing the sampling ratio s in the case of tracking the sine shape signal decreases the largest step $y_{i_{max}}$ (Fig. 9.16), but that is not the case in tracking the square signal (Fig. 9.17). The reduction of the largest step with index i_{max} was tested by the sine-shape signal with amplitude $A = R/2$ for the worst case

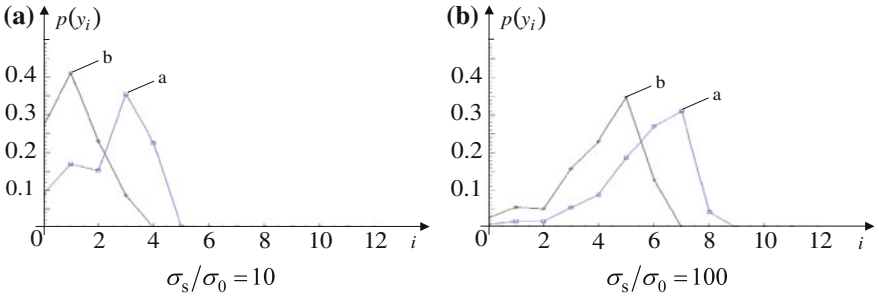

Fig. 9.20 Probabilities of the steps in tracking the noisy signals: *a* the rectangular noise distribution, *b* the Gaussian noise distribution

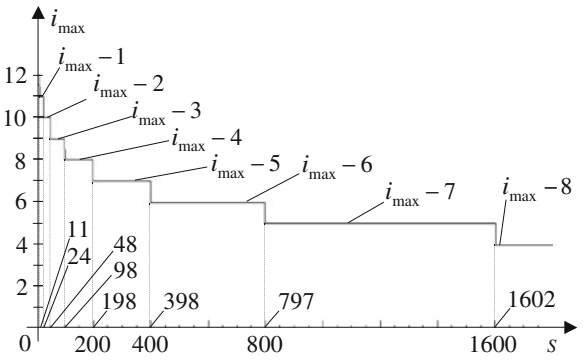

Fig. 9.21 Reduction of the index of the maximal steps in relation to the sampling ratio s of the adaptive ADC for tracking the sine shape signal

(Fig. 9.21). As the number of the A/D converter bits was $b = 12$ the first six settling steps of the tracking were removed and following that, four cycles of the signal were examined. To find the maximal possible step $y_{i_{max}}$ at every sampling ratio s the phase of the sine signal was changed $0° \leq \varphi \leq 90°$ with resolution $\Delta\varphi = 1°$. The first reduction of the largest step $y_{i_{max}} = R/2$ to $y_{i_{max}-1} = R/4$ is at $s = 11_{i_{max}-1}$ samples in the period. The next reductions are at $s = 24_{i_{max}-2}$, $48_{i_{max}-3}$, $98_{i_{max}-4}$, $198_{i_{max}-5}$, $398_{i_{max}-6}$, $797_{i_{max}-7}$, $1602_{i_{max}-8}$, …. The reduction of the largest steps is independent of the number of bits b.

The probability distributions of the steps show (Fig. 9.22) that effectively the largest six steps make the greatest contribution in tracking.

The largest six steps in tracking and belonging uncertainties $\Delta_i/\sqrt{12}$ increases quantization noise in comparison to the lowest value $\sigma_0 = \Delta_0/\sqrt{12}$(Fig. 9.23b). With an increase of the sampling ratio s in tracking, it is possible to decrease very quickly the quantization uncertainty σ_q(Fig. 9.23, testing conditions are the same as for Fig. 9.21). It can be seen that the quantization standard deviation of tracking the signal differs slightly from the expected standard deviation using the probability distributions of the approximation steps at the lower values of s (Fig. 9.23: curves a and b), due to the small number of samples in the particular quantization

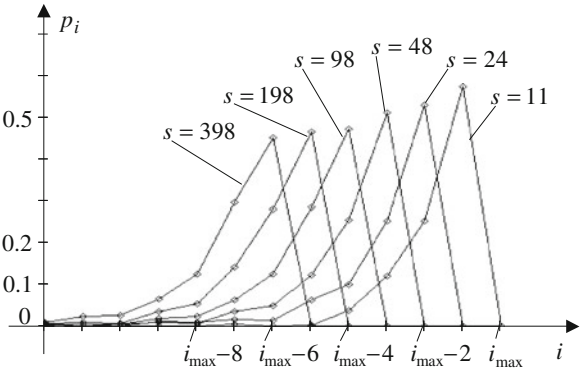

Fig. 9.22 Probability distributions of the approximation steps in relation to the characteristic sampling ratios s of the adaptive A/D converter

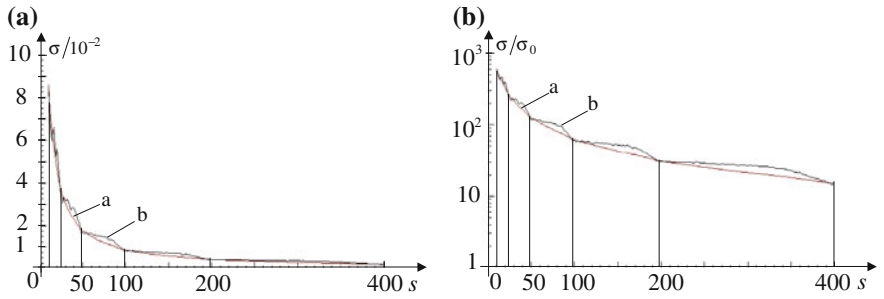

Fig. 9.23 Standard deviations (**a**) and ratios of the standard deviations to $\sigma_0 = \Delta_0/\sqrt{12}$ (**b**) in relation to the sampling ratio s of the 12-bit adaptive A/D converter: *a* approximation using the probability distributions of the approximation steps like in Fig. 9.21; *b* results of tracking the signal

intervals, and accordingly the expected rectangular error distributions are not achieved.

The increased quantization noise at lower sampling ratios s is at the price of increasing b-times the available sampling points in comparison with the classic A/D converter, with the successive approximation procedure having to perform all b steps to the final uncertainty $\Delta_0/\sqrt{12}$ of the smallest quantization interval Δ_0. As we have b-times more sampling points using the adaptive A/D converter, the noise contribution in the estimation of the sine signal parameter (like amplitude, frequency, and phase) is reduced by $\sqrt{b}$ [26] (Fig. 9.24). Taking into consideration only the quantization noise contribution in parameter estimations, the adaptive A/D conversion performs better than the successive approximation A/D conversion, if

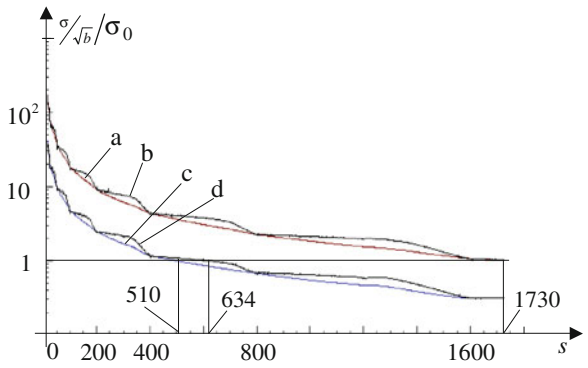

Fig. 9.24 Ratios of standard deviations of the adaptive A/D conversion to $\sigma_0 = \Delta_0/\sqrt{12}$ in relation to the sampling ratio s: *a* approximation using the probability distributions of the approximation steps with 12-bit A/D converter; *b* results of tracking the signal with 12-bit A/D converter; *c* approximation using the probability distributions of the approximation steps with 10-bit A/D converter; *d* results of tracking the signal with 10-bit A/D converter

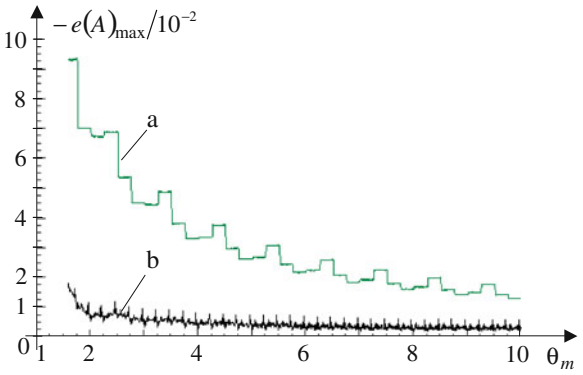

Fig. 9.25 Maximal errors of the amplitude estimations with 12-bit A/D converters in relation to the relative frequency: *a* the successive approximation A/D converter with $s_0 = 4$ samples per period; *b* the adaptive A/D converter with $s = b \cdot s_0 = 48$ samples per period

the sampling ratio s is high enough (Fig. 9.24: $s > 634$ for 10-bit A/D converter and $s > 1730$ for 12-bit A/D converter).

Considering together contributions of both—the systematic and the random errors—in the signal parameters estimations, shows the advantage of the adaptive A/D conversion even at lower values of the sampling ratio s. To demonstrate, we checked the error of the amplitude estimation $e(A) = A/A^* - 1$ (A^* is the true value of the amplitude) for one sine component m with a double scan varying both the relative frequency $\theta_m = f_m \cdot T_M = i_m + \delta_m$ (the number of cycles in the measurement interval T_M) and the phase of the signal, because the long-range leakages are frequency and phase dependent (Fig. 9.25: $A = 1$, $1.6 \le \theta_m \le 10$, $\Delta\theta = 0.01$ and $-90^\circ \le \varphi \le 90^\circ$, $\Delta\varphi = 3^\circ$, $b = 12$). In estimations, the three-point interpolated DFT algorithms and the Hann window were used [27].

$$\delta_m = 2\frac{|G(i_m+1)| - |G(i_m-1)|}{|G(i_m-1)| + 2|G(i_m)| + |G(i_m+1)|} \tag{9.44}$$

$$A = \frac{\pi\delta_m}{\sin(\pi\delta_m)}\frac{\left(1-\delta_m^2\right)\left(4-\delta_m^2\right)}{3} \cdot \left[|G(i_m-1)| + 2|G(i_m)| + |G(i_m+1)|\right] \tag{9.45}$$

The maximum values of errors (from 60 iterations) at the given relative frequency show the benefit of the adaptive A/D conversion, even at lower values of the sampling ratio, if the number of the measured signal cycles is small.

9.6 Conclusion

In this chapter, the possibility of an adaptive A/D conversion with the differential tracking procedure of the signal by non-uniform exponential quantization is presented. The adaptive property of the method that every previous approximation step to signal becomes the center of observation with an exponential increase in resolution at the new step, gives a possibility of reducing the time of conversion.

The best trade-off between the number of decision levels and the settling time is achieved by the pure exponential quantization rule. The fastest response is achievable with base 2. The adaptive method is equivalent to the optimal parallel-serial method at worst by a maximum possible number of steps $k_{\max} \approx b/2$. On average, the number of steps is smaller than one third.

The principle that the larger the distance is, the larger the dynamic error of estimation is, shows the limitation of the information flow. In comparison with uniform quantization, the information of the first step does not increase with increasing the number of bits b. It is limited to the constant value $H(a = 2)_{i\to\infty} = 3$. The quantity of information decreases on approach to the last step of conversion. The finite impulse response of the sampling device, which is not infinitely short, is the cause of the finite information capacity of the measurement channel.

By analyses, it has been shown that the non-uniform exponential quantization b-bit A/D conversion gives better results in tracking the signal and its parameters estimations than the classical A/D conversion with the successive approximation procedure, due to b-times more available sampling points. The non-uniform exponential quantization procedure shows robustness in tracking the noisy signals with standard deviations a few times higher than the quantization standard uncertainty of the basic resolution. Taking into consideration only the quantization noise contribution, the adaptive A/D conversion provides better results if the sampling ratio s is high enough. Considering together both the systematic and the random errors in the signal parameter estimations, shows the advantage of the adaptive A/D conversion even at lower values of the sampling ratio s.

Appendix A

After the second step of the A/D conversion with the exponential quantization and base $a = 2$ the index i is reduced by 2 (9.21). The representatives with the index above $i = 5$ enable the following paths and their entropies:

$$\begin{aligned} &y_5 \to 2p_3H_1 \\ &y_6 \to 2(p_3H_1 + p_4H_2) \\ &\vdots \\ &y_i \to 2(p_3H_1 + p_4H_2 + \cdots + p_{i-2}H_{i-4}) \end{aligned} \tag{A1}$$

In an example with parameters $i = 8,\ a = 2 \quad \Rightarrow \quad k_{\max} = 4$, the following information flow can be evaluated.

$$
\begin{aligned}
&{}_1H = H_8 = 2.9897 \text{ bit}\\
&{}_2H = 2\big(p_3H_1 + p_4H_2 + \underbrace{p_5H_3}_{y_5 \text{ in step 2}} + \underbrace{p_6H_4}_{y_6 \text{ in step 2}} + \underbrace{p_7H_5}_{y_7 \text{ in step 2}} + \underbrace{p_8H_6}_{y_8 \text{ in step 2}}\big) = 2.8176 \text{ bit}\\
&{}_3H = 2\Big(\overbrace{2p_3H_1}^{y_5 \text{ in step 2}} + \overbrace{2\big(p_3H_1 + p_4H_2\big)}^{y_6 \text{ in step 2}} + \overbrace{2\big(p_3H_1 + p_4H_2 + p_5H_3\big)}^{y_7 \text{ in step 2}} + \overbrace{2\big(p_3H_1 + p_4H_2 + p_5H_3 + p_6H_4\big)}^{y_8 \text{ in step 2}}\Big)\\
&{}_3H = 4\big(4p_3H_1 + 3p_4H_2 + \underbrace{2p_5H_3}_{y_5 \text{ in step 3}} + \underbrace{p_6H_4}_{y_6 \text{ in step 3}}\big) = 1.9994 \text{ bit}\\
&{}_4H = 4\Big(\overbrace{2\big(2p_3H_1\big)}^{y_5 \text{ in step 3}} + \overbrace{2\big(p_3H_1 + p_4H_2\big)}^{y_6 \text{ in step 3}}\Big) = 8\big(3p_3H_1 + p_4H_2\big) = 0.6070 \text{ bit}\\
&\sum_{k=1}^{4} {}_kH = 8.4136 \text{ bit} = \text{lb}\left(\frac{R_8}{\Delta_0}\right) = \text{lb}\,(341)
\end{aligned}
\tag{A2}
$$

References

1. Bayati, A.: Reducing differential non-linearity in A/D converters. Siemens Forsch.-u. Entwickl., Berlin **3**(6), 348–352 (1974)
2. Agrež, D.: Quantization noise of the non-uniform exponential tracking A/D conversion. In: Proceedings of the 16th IMEKO TC4 IWADC Workshop and IEEE 2011 ADC Forum, Orvieto, Italy, 6 p (2011)
3. Gray, R.M., Neuhoff, D.L.: Quantization. IEEE Trans. Inf. Theory **44**(6), 2325–2383 (1998)
4. Gersho, A.: Principles of quantization. IEEE Trans. Circuits Syst. CAS **25**(7), 427–436 (1978)
5. van de Plassche, R.: CMOS Integrated Analog-to-Digital and Digital-to-Analog Converters. Kluwer, Dordrecht (2010)
6. Siebert, W.Mc.C.: Circuits, Signals and Systems. The MIT Press, Cambridge/McGraw-Hill, New York (1986)
7. Jerri, A.J.: The Shannon sampling theorem—its various extensions and applications: a tutorial review. Proc. IEEE **65**(11), 1565–1596 (1977)
8. Papoulis, A.: Error analysis in sampling theory. Proc. IEEE **54**(7), 947–955 (1966)
9. Landau, H.J.: Sampling, data transmission and the Nyquist rate. Proc. IEEE **55**(10), 1701–1706 (1967)
10. Woschni, E.G.: Minimizing aliasing errors of sensors with digital output. J. Phys. E: Sci. Instrum. **20**(2), 119–124 (1987)
11. Harris, F.J.: On the use of windows for harmonic analysis with the discrete Fourier transform. Proc. IEEE **66**(1), 51–83 (1978)
12. Herold, H., Woschni. E.G.: Fehler bei der Abtastung von Signalen endlicher Dauer. msr(Berlin) **28**(11), 485–509 (1985)
13. Bacrania, K.: A 12-bit successive-approximation-type ADC with digital error correction. IEEE J. Solid-State Circuits SC-**21**(6), 1016–1025 (1986)
14. Shannon, C.E.: A mathematical theory of communication. The Bell Syst. Tech. J. **XXVII**(3), 379–423 (1948)
15. Wagdy, M.F., Ng, W.M.: Validity of uniform quantization error model for sinusoidal signals without and with dither. IEEE Trans. Instrum. Meas. **38**(3), 718–722 (1989)
16. Max, J.: Quantizing for minimum distortion. IRE Trans. Inform. Theory IT-**6**, 7–12 (1960)

17. Panter, P.F., Dite, W.: Quantizing distortion in pulse-count modulation with nonuniform spacing of levels. Proc. IRE **39**, 44–48 (1951)
18. Belmont, M.R.: Nonuniform sampling specifically for finite-length data. IEEE Proc. **F140**(1), 55–62 (1993)
19. Bruggemann, H.: Ultrafast feedback A/D conversion made possible by a nonuniform error quantizer. IEEE J. Solid-State Circuits SC **18**(1), 99–105 (1983)
20. Analog Devices (2011) AD5760, Ultra Stable 16-Bit ± 0.5 LSB INL Voltage Output DAC – Data Sheet. Analog Devices, Inc
21. Fu, G., Mantooth, H.A., Di, J.: A 12-bit CMOS current steering D/A converter with a fully differential voltage output. In: Proceedings of 12th International Symposium on Quality Electronic Design, pp. 398–404 (2011)
22. Klir, G.J., Folger, T.A.: Fuzzy sets, uncertainty and information. State university of New York, Binghamton, Prentice-Hall, Inc (1988)
23. Woschni, E.G.: Informationstechnik - Signal, System, Information, 4th edn. VEB Verlag Technik, Berlin (1990)
24. Woschni, E.G.: Zur Bedeutung der A-priori-Information speziell in der Messtechnik. msr(Berlin) **32**(6), 271–274 (1989)
25. Verdu, S.: Fifty years of Shannon theory. IEEE Trans. Inf. Theory **44**(6), 2057–2078 (1998)
26. Widrow, B., Kollar, I.: Quantization Noise. Cambridge University Press, Cambridge (2008)
27. Agrež, D.: Weighted multi-point interpolated DFT to improve amplitude estimation of multi-frequency signal. IEEE Trans. Instrum. Meas. **51**, 287–292 (2002)

Part III
Testing

Chapter 10
Dynamic Testing of Analog-to-Digital Converters by Means of the Sine-Fitting Algorithms

Dario Petri, Daniel Belega and Dominique Dallet

Abstract The chapter is dedicated to dynamic testing of Analog-to-Digital Converters (ADCs) by means of both time- and frequency-domain sine-fitting algorithms (SFAs). At first the sine-fitting procedure used for the estimation of SIgnal-to-Noise And Distortion ratio (*SINAD*) and Effective Number Of Bits (*ENOB*) parameters is described. In the following the expressions for the bias and the standard deviation of the *ENOB* estimator provided by a SFA are derived. Then, the SFAs based on the Interpolated Discrete Fourier Transform (IpDFT) method, the Energy-Based (EB) method, and the well known three- and four-parameter SFAs are separately analyzed. For each algorithm, the basic theoretical background and the operational detail are given. Moreover, the accuracy of all the presented algorithms are compared by means of both theoretical and simulation results. Some aspects concerning the influence of the harmonics, time jitter, and time base distortions on the dynamic performance of an ADC are also discussed. Besides, some Multi-Harmonics Sine-Fitting Algorithms (MHSFAs) are briefly described. Finally, the accuracy of the *ENOB* estimates provided by the considered SFAs and MHSFAs are compared through real-world data.

D. Petri
Department of Industrial Engineering, University of Trento, Via Sommarive, No. 5, 38100 Trento, Italy
e-mail: dario.petri@unitn.it

D. Belega (✉)
Faculty of Electronics and Telecommunications, "Politehnica" University of Timişoara, Bv. V. Pârvan, No. 2 300223 Timisoara, Romania
e-mail: daniel.belega@etc.upt.ro

D. Dallet
IMS Laboratory, University of Bordeaux-IPB ENSEIRB MATMECA 351 Cours de la Libération, Bâtiment A31 33405 Talence Cedex, France
e-mail: dominique.dallet@ims-bordeaux.fr

P. Carbone et al. (eds.), *Design, Modeling and Testing of Data Converters*, Signals and Communication Technology, DOI: 10.1007/978-3-642-39655-7_10,

10.1 Introduction

The overall dynamic performance of an Analog-to-Digital Converter (ADC) is evaluated by means of different parameters. Two of the parameters mostly used are the SIgnal-to-Noise And Distortion ratio (*SINAD*) and the Effective Number Of Bits (*ENOB*) [1, 2]. The first one is defined as the ratio between the rms value of the adopted test sine-wave and the rms value of the overall ADC output noise. Conversely, the *ENOB* represents the number of bits of an ideal ADC with a quantization error rms value equal to the rms value of the overall output noise of the ADC under test. This last parameter is used very often since it provides an easy to understand figure of an ADC dynamic performance [1]. The Sine-Fitting Algorithms (SFAs) are a very powerful tool for estimating these parameters. They operate either in the time-domain or in the frequency-domain in order to determine the best sine fit to the output signal of an ADC fed with a pure sine-wave. Then, the parameters of interest are estimated by evaluating the power of the residual signal. To this aim the current Standards for ADC dynamic testing—the IEEE Standard 1241 [1] and the European Project DYNAD [2]—suggest the use of time-domain SFAs, which are based on the application of the least squares approach. They are named three-parameter sine-fitting algorithm (3PSFA) and four-parameter sine-fitting algorithm (4PSFA), according to the number of the sine-wave parameters to be estimated. Due to the least squares approach the above algorithms are robust with respect to non-coherent sampling, and provide accurate estimates. Besides, they are simple to implement.

Among different frequency-domain SFAs, those based on the Interpolated Discrete Fourier Transform (IpDFT) method or the Energy-Based (EB) method provide accurate *ENOB* and *SINAD* estimates [3, 4]. In the following, they are called SFA-IpDFT and SFA-EB, respectively. The SFA-IpDFT and the SFA-EB exhibit a smaller computational complexity than the 3PSFA and 4PSFA, and are very simple to understand and to apply.

This chapter is focused on the dynamic testing of ADCs based on either time-domain and frequency-domain SFAs. At first, the statistical description of the *ENOB* estimator returned by a SFA is provided. Then, the main features of the 3PSFA, 4PSFA, SFA-IpDFT, and SFA-EB are presented. Moreover, the accuracies of these algorithms as expressed by the expected sum-squared fitting error are compared through theoretical and simulations results. The influences of harmonics, time jitter, and time base distortions on the dynamic performance of an ADC are also briefly discussed. Also, some Multi-Harmonics Sine-Fitting Algorithms (MHSFAs) are briefly described. Further, the accuracies of the considered SFAs and MHSFAs are compared by means of experimental results. Finally, some conclusions are presented.

10.2 Estimation of the *SINAD* and *ENOB* Parameters by Sine-Fitting Algorithms

Let us consider an *N*-bit ADC with full-scale range *FSR*, fed with a pure sine-wave. Ideally, to test all the ADC output codes, the sine-wave amplitude should be equal to *FSR*/2, while the sine-wave offset should be zero for bipolar ADCs and *FSR*/2 for unipolar ADCs. The signal obtained at the ADC output can be expressed as follows:

$$y(n) = s(n) + r(n) = A\sin\left(2\pi\frac{f_{in}}{f_s}n + \varphi\right) + B + r(n), \quad n = 0,\ 1,\ 2, \ldots, M-1 \tag{10.1}$$

where A, f_{in}, φ, and B are, respectively, the amplitude, the frequency, the phase, and the offset of the output sine-wave $s(\cdot)$, M is the number of samples acquired with sampling frequency f_s, whereas $r(\cdot)$ is the ADC output noise, which represents the overall error introduced during conversion and includes the effects of random noise, fixed pattern errors, nonlinearities (e.g. harmonic or spurious components), aperture uncertainty, etc. The frequency f_{in} is usually chosen smaller than $f_s/2$ to satisfy the Nyquist theorem. The ratio between f_{in} and f_s can be expressed as:

$$\frac{f_{in}}{f_s} = \frac{v}{M} = \frac{J+\delta}{M}, \tag{10.2}$$

where J and δ ($-0.5 \le \delta < 0.5$) are respectively the integer and the fractional parts of the number of recorded sine-wave cycles v. It is worth noticing that v represents also the sine-wave normalized frequency expressed in bins and it is usually evaluated by estimating J and δ separately. It is well-known that $\delta = 0$ corresponds to the so called coherent sampling. However, non-coherent sampling (that is $\delta \neq 0$) often occurs in practice due to lack of synchronization between sine-wave and sampling frequencies.

Usually the value of J can be determined exactly by means of (10.2) when enough accurate estimates for f_{in} and f_s are available, or by using a maximum search routine applied to the DFT samples of the ADC output spectrum. Thus, from (10.2) it follows that the estimation uncertainties of v and δ coincide.

All the SFAs estimate the *SINAD* and *ENOB* parameters through the following steps [1, 2]:

1. Acquire M consecutive samples of the ADC output signal $y(n)$, $n = 0, 1, 2, \ldots, M-1$.
2. Determine the best sine fitting of the ADC output sequence $y(\cdot)$:

$$\hat{s}(n) = \hat{A}\sin\left(2\pi\hat{v}\frac{n}{M} + \hat{\varphi}\right) + \hat{B}, \quad n = 0,\ 1, \ldots,\ M-1 \tag{10.3}$$

where $\hat{A}$, $\hat{\nu}$, $\hat{\varphi}$, and $\hat{B}$ are respectively the amplitude, normalized frequency, phase, and offset of the best fitting sine-wave. In particular, to achieve $\hat{\nu}$, only an estimate of δ, $\hat{\delta}$, is needed since $\hat{\nu} = J + \hat{\delta}$.

The above parameter estimates can be determined by means of time-domain or frequency-domain methods [1–4].

3. Evaluate the residual signal:

$$\hat{r}(n) = y(n) - \hat{s}(n), \quad n = 0, 1, \ldots, M-1 \tag{10.4}$$

4. Evaluate the residual rms value:

$$\hat{r}_{rms} = \sqrt{\frac{1}{M}\sum_{n=0}^{M-1} \hat{r}^2(n)}. \tag{10.5}$$

5. Determine the *SINAD* and *ENOB* parameters as:

$$\hat{SINAD} = 10\log_{10}\left(\frac{\hat{A}^2}{2\,\hat{r}_{rms}^2}\right) \quad (dB) \tag{10.6}$$

$$\hat{ENOB} = N - \frac{1}{2}\log_2\left(\frac{\hat{r}_{rms}^2}{\sigma_q^2}\right) \quad (bits) \tag{10.7}$$

where σ_q^2 is the variance of the quantization error of an ideal quantizer, which is usually assumed uniformly distributed [1], that is

$$\sigma_q^2 = \frac{Q^2}{12} = \frac{1}{12}\left(\frac{FSR}{2^N}\right)^2, \tag{10.8}$$

where Q is the theoretical code bin width of the ADC under test.

It is worth noticing that when the input sequence can be described as a zero-mean, uniform and stationary random process, the quantization noise can be modeled as additive white noise, uniformly distributed over the range $(-Q/2, Q/2)$, and uncorrelated with the input [5, 6]. Conversely, when sine-wave inputs are applied, the corresponding quantization noise variance can be accurately expressed as [7]:

$$\sigma_{q,\,\sin}^2 \cong \frac{Q^2}{12} + \frac{1.34 \cdot Q^2}{\pi^3 2^{N/2}}. \tag{10.9}$$

This implies that (10.8) provides very good accuracy even when sine-waves are quantized by means of small resolution ADCs. For instance, when $N = 6$ bits, the relative error introduced by (10.8) is about 6.1 %, a value that can be accepted in many engineering applications. This is the reason why the current Standards for

ADC testing recommend the evaluation of the ideal quantization variance σ_q^2 by using (10.8), which can be evaluated very easily.

It is well known that a linear relationship exists between the *ENOB* and *SINAD* (in dB) [1]:

$$SINAD\ (dB) = 6.02ENOB + 1.76. \tag{10.10}$$

In the following we limit our analysis to the *ENOB* parameter only without any loss of generality.

10.3 Statistical Performance of the *ENOB* Estimator Provided by a Sine-Fitting Algorithm

The *ENOB* of an ADC is defined as [1, 2]:

$$ENOB = N - \frac{1}{2}\log_2\left(\frac{\sigma_r^2}{\sigma_q^2}\right) \quad (bits), \tag{10.11}$$

where σ_r^2 is the variance of the output noise $r(\cdot)$ and $\sigma_q^2$is defined by (10.8).

Using (10.7) and (10.11) we obtain:

$$\hat{ENOB} = ENOB - \frac{1}{2}\log_2\left(\frac{r_{rms}^2}{\sigma_r^2}\right) \quad (bits) \tag{10.12}$$

Thus, the bias and the standard deviation of the *ENOB* estimator can be expressed as:

$$bias\left[\hat{ENOB}\right] = E\left[\hat{ENOB}\right] - ENOB = -\frac{1}{2}E\left[\log_2\left(\hat{r}_{rms}^2\right)\right] + \frac{1}{2}\log_2\left(\sigma_r^2\right), \tag{10.13}$$

and

$$std\left[\hat{ENOB}\right] = \frac{1}{2}std\left[\log_2\left(\hat{r}_{rms}^2\right)\right]. \tag{10.14}$$

By modeling the residual rms value $\hat{r}_{rms}^2$ as a random variable and linearizing the $\log_2(\cdot)$ function, we have [8]:

$$E\left[\log_2\left(\hat{r}_{rms}^2\right)\right] \cong \log_2\left(E\left[\hat{r}_{rms}^2\right]\right) - \frac{1}{2\ln(2)}\left(\frac{\mathrm{var}\left[\hat{r}_{rms}^2\right]}{E^2\left[\hat{r}_{rms}^2\right]}\right), \tag{10.15}$$

and

$$std\left[\log_2\left(\hat{r}_{rms}^2\right)\right] \cong \frac{1}{\ln(2)}\frac{std\left[\hat{r}_{rms}^2\right]}{E\left[\hat{r}_{rms}^2\right]}, \tag{10.16}$$

where $E\left[\hat{r}_{rms}^2\right]$, $std\left[\hat{r}_{rms}^2\right]$, $\mathrm{var}\left[\hat{r}_{rms}^2\right]$ are respectively the expectation, the standard deviation, and the variance of the random variable $\hat{r}_{rms}^2$.

By replacing (10.15), expression (10.13) becomes:

$$bias\left[\hat{ENOB}\right] \cong \frac{1}{2}\log_2\left(\frac{\sigma_r^2}{E\left[\hat{r}_{rms}^2\right]}\right) + \frac{1}{4\ln(2)}\frac{\mathrm{var}\left[\hat{r}_{rms}^2\right]}{E^2\left[\hat{r}_{rms}^2\right]}, \tag{10.17}$$

while using (10.16), expression (10.14) provides:

$$std\left[\hat{ENOB}\right] \cong \frac{1}{2\ln(2)}\frac{std\left[\hat{r}_{rms}^2\right]}{E\left[\hat{r}_{rms}^2\right]}. \tag{10.18}$$

It should be noted that (10.17) and (10.18) hold regardless the overall ADC output noise characteristics.

In the following we assume that the residual signal $\hat{r}(\cdot)$ can be modelled as a zero-mean white noise. This occurs in practice when the input sine-wave frequency is well below the ADC sampling rate. In this case, from (10.5), it follows:

$$E\left[\hat{r}_{rms}^2\right] = E\left[\hat{r}^2\right] = \sigma_{\hat{r}}^2, \tag{10.19}$$

and:

$$\mathrm{var}\left[\hat{r}_{rms}^2\right] = E\left[\hat{r}_{rms}^4\right] - E^2\left[\hat{r}_{rms}^2\right] = E\left[\hat{r}_{rms}^4\right] - \sigma_{\hat{r}}^4, \tag{10.20}$$

where $\sigma_{\hat{r}}^2$ represents the variance of the residual signal.

Using (10.5) and (10.19) we have:

$$E\left[\hat{r}_{rms}^4\right] = \frac{1}{M}E\left[\hat{r}^4\right] + \frac{M-1}{M}\sigma_{\hat{r}}^4. \tag{10.21}$$

By replacing (10.21) in (10.20) we achieve:

$$\mathrm{var}\left[\hat{r}_{rms}^2\right] = \frac{1}{M}\left(E\left[\hat{r}^4\right] - \sigma_{\hat{r}}^4\right). \tag{10.22}$$

Finally, using (10.19) and (10.22), the bias and the standard deviation of the *ENOB* estimator result:

$$bias\left[\hat{ENOB}\right] \cong \frac{1}{2}\log_2\left(\frac{\sigma_r^2}{\sigma_{\hat{r}}^2}\right) + \frac{1}{4\ln(2)\,M}\left(\frac{E[\hat{r}^4]}{\sigma_{\hat{r}}^4} - 1\right), \tag{10.23}$$

and:

$$std\left[\hat{ENOB}\right] \cong \frac{1}{2\ln(2)\sqrt{M}}\sqrt{\frac{E[\hat{r}^4]}{\sigma_{\hat{r}}^4} - 1}. \tag{10.24}$$

Since $\sigma_{\hat{r}}^2$ converges to σ_r^2 as M increases [9], (10.23), and (10.24) show that both $bias\left[\hat{ENOB}\right]$ and $std\left[\hat{ENOB}\right]$ tend to zero. Hence we can conclude that the *ENOB* estimators provided by SFAs are statistically consistent.

In particular, when the residual signal $\hat{r}(\cdot)$ is a Gaussian noise, we have [10]:

$$E\left[\hat{r}^4\right] \cong 3\sigma_{\hat{r}}^4. \tag{10.25}$$

and the above expressions becomes:

$$bias\left[\hat{ENOB}\right] \cong \frac{1}{2}\log_2\left(\frac{\sigma_r^2}{\sigma_{\hat{r}}^2}\right) + \frac{0.72}{M}, \tag{10.26}$$

$$std\left[\hat{ENOB}\right] \cong \frac{1.02}{\sqrt{M}}. \tag{10.27}$$

Simulation results with white noise affecting the ADC output showed that, independently of the ADC resolution, the estimator bias (10.26) is always between $1/M$ and $4/M$. Thus, (10.27) shows that the ratio between the *ENOB* estimator bias and the standard deviation is proportional to $1/\sqrt{M}$. As a consequence, for values of M commonly adopted in practice (e.g. $M \geq 512$) the estimator bias due to wide-band noise is negligible as compared to the related standard deviation.

10.4 Interpolated DFT and Energy-based Methods

The aim of this Section is to provide the essential information needed to correctly apply the IpDFT and EB methods to sine-fitting algorithms.

10.4.1 Interpolated DFT method

In the IpDFT method, the acquired sequence $y(\cdot)$ is firstly multiplied by a suitable window $w(\cdot)$ [11–15]. Usually a H-term cosine window ($H \geq 2$) is employed, which is defined as:

$$w(n) = \sum_{h=0}^{H-1} (-1)^h a_h \cos\left(2\pi h \frac{n}{M}\right), \quad n = 0, 1, \ldots, M-1 \tag{10.28}$$

where a_h, $h = 0, \ldots, H-1$ are the window coefficients.

Then the Discrete-Time Fourier Transform (DTFT), $Y_w(\cdot)$, of the windowed signal $y_w(n) = y(n) \cdot w(n)$ is evaluated. From (10.1) we obtain:

$$Y_w(\lambda) = \frac{A}{2j}\left[W(\lambda - \nu)e^{j\varphi} - W(\lambda + \nu)e^{-j\varphi}\right] + BW(\lambda) + R_w(\lambda), \quad \lambda \in [0,\ M) \tag{10.29}$$

where $W(\cdot)$ is the DTFT of the adopted window $w(\cdot)$ and $R_w(\cdot)$ is the DTFT of the sequence $r_w(n) = r(n) \cdot w(n)$, $n = 0, 1, 2, \ldots, M-1$.

The window transform $W(\cdot)$ can be expressed as [12]:

$$W(\lambda) = \sin(\pi\lambda)e^{-j\pi\frac{M-1}{M}\lambda}W_0(\lambda), \quad \lambda \in [0,\ M) \tag{10.30}$$

where

$$W_0(\lambda) = \sum_{h=0}^{H-1}(-1)^h 0.5a_h\left[\frac{e^{-j\frac{\pi}{M}h}}{\sin\frac{\pi}{M}(\lambda-h)} + \frac{e^{j\frac{\pi}{M}h}}{\sin\frac{\pi}{M}(\lambda+h)}\right], \quad \lambda \in [0, M). \tag{10.31}$$

For $|\lambda| \ll M$ and high values of M, we have [14]:

$$W(\lambda) \cong \frac{M\lambda\sin(\pi\lambda)}{\pi}e^{-j\pi\frac{M-1}{M}\lambda}\sum_{h=0}^{H-1}\frac{(-1)^h a_h}{\lambda^2 - h^2}. \tag{10.32}$$

It is worth noticing that the second term within the square brackets in (10.29) represents the sine-wave image component.

The integer part J of the number of acquired sine-wave cycles can be determined by using a maximum search routine applied to the DFT samples $|Y_w(\lambda)|$, $\lambda = 0, 1,\ldots, M/2-1$. Moreover, the fractional part δ can be estimated by evaluating the ratio α of the two maximum DFT spectrum samples:

$$\alpha = \frac{|Y_w(J+i)|}{|Y_w(J-1+i)|}, \tag{10.33}$$

where $i = 0$ if $|Y_w(J-1)| > |Y_w(J+1)|$ and $i = 1$ if $|Y_w(J-1)| < |Y_w(J+1)|$.

In practice the number of acquired sine-wave cycles and the number of samples are usually quite high (e.g. $v \geq 15$ and $M \geq 512$). Thus, the effect of both the spectral interference from the image component and the output noise $R_w(\cdot)$ on the spectrum samples close to the peak (that is for $\lambda \cong v$) are negligible [13, 15], and (10.29) becomes:

$$Y_w(\lambda) \cong \frac{A}{2j}W(\lambda - v)e^{j\varphi}, \quad if\ \lambda \cong v \tag{10.34}$$

By using (10.34), expression (10.33) provides:

$$\alpha \cong \frac{|W(i-\delta)|}{|W(-1+i-\delta)|}. \tag{10.35}$$

Thus, the knowledge of the ratio (10.33) allows us to estimate the fractional frequency δ by simply inverting (10.35) [12, 15]:

$$\hat{\delta}_{ip} = g(\alpha), \tag{10.36}$$

The function $g(\cdot)$ is often approximated in least squares sense by a polynomial. However, if a Maximum Sidelobe Decay (MSD) window (known also as the class I Rife-Vincent windows) is used, analytical expressions for the function $g(\cdot)$ can be easily achieved starting from (10.32) and (10.35). The H-term MSD window ($H \geq 2$) has the most rapidly spectrum sidelobe decaying rate among all the cosine

windows of a given number of terms H. Its coefficients are given by the following expressions [16]:

$$a_0 = \frac{C_{2H-2}^{H-1}}{2^{2H-2}}, \quad a_h = \frac{C_{2H-2}^{H-h-1}}{2^{2H-3}}, \quad h = 1, 2, \ldots, H-1 \tag{10.37}$$

in which $C_m^p = m!/((m-p)!\,p!)$.

In fact, for the H-term MSD window, the DTFT (10.32) becomes [14]:

$$W(\lambda) = \frac{M \sin(\pi\lambda)}{2^{2H-2}\pi\lambda} e^{-j\pi\frac{M-1}{M}\lambda} \frac{(2H-2)!}{\prod_{h=1}^{H-1}\left(h^2-\lambda^2\right)}. \tag{10.38}$$

By using (10.35) and (10.38), (10.36) reduces to the following very simple formula:

$$\hat{\delta}_{ip} = \frac{(H-1+i)\alpha - H + i}{\alpha+1}. \tag{10.39}$$

From (10.34) and (10.38) the amplitude A can then be estimated as:

$$\hat{A}_{ip} = \frac{2^{2H-1}\pi\hat{\delta}_{ip}|Y_w(J)|}{M\sin(\pi\hat{\delta}_{ip})(2H-2)!} \prod_{h=1}^{H-1}\left(h^2-\hat{\delta}_{ip}^2\right). \tag{10.40}$$

Also (10.34) and (10.30) show that the sine-wave phase can be estimated as:

$$\hat{\varphi}_{ip} = \arg[Y_w(J)] - \pi\hat{\delta}_{ip} + \pi\frac{\hat{\delta}_{ip}}{M} - \frac{\pi}{2}sign(\hat{\delta}_{ip}) - \arg\left[W_0\left(-\hat{\delta}_{ip}\right)\right], \tag{10.41}$$

where $sign(\cdot)$ is the sign function.

The estimators $\hat{v}_{ip} = J + \hat{\delta}_{ip}$, $\hat{A}_{ip}$, and $\hat{\varphi}_{ip}$ represent the parameters of the sine fit returned by the IpDFT method based on the H-term MSD window.

Moreover, the acquired signal offset can be estimated as [3]:

$$\hat{B}_{ip} = \frac{Y_w(0)}{M \cdot NPSG}, \tag{10.42}$$

where $NPSG$ is the window normalized peak signal gain [17], which for a H-term MSD window is given by $NPSG = a_0 = C_{2H-2}^{H-1}/2^{2H-2}$ [14].

The estimators $\hat{A}_{ip}$ and $\hat{\varphi}_{ip}$ depend on $\hat{\delta}_{ip}$. Hence, the related estimator error $\Delta_{\delta_{ip}} = \hat{\delta}_{ip} - \delta$ is of main interest. This error is mainly due to the spectral interference from the image component of the ADC output signal. Indeed, the effect of this component is usually quite stronger than harmonics or spurious tones. In particular, $\Delta_{\delta_{ip}}$ depends on the sine-wave phase and exhibits a sine-wave like behavior of amplitude [15]:

$$\left|\Delta_{\delta_{ip}}\right|_{\max} \cong \frac{2|\delta|(J+\delta)(H-|\delta|)}{(2J+\delta)(2J+\delta+(-1)^{i+1}H)} \frac{\prod_{h=1}^{H-1}\left(h^2-\delta^2\right)}{\prod_{h=1}^{H-1}\left[(2J+\delta)^2-h^2\right]}. \tag{10.43}$$

Thus, its mean square value is equal to $\left|\Delta_{\delta_{ip}}\right|_{\max}^2/2$.

Conversely, the frequency estimator standard deviation due to wide-band noise is [3, 14]:

$$\begin{aligned}\sigma_{\hat{\delta}_{ip}} = \sigma_{\hat{y}_{ip}} &\cong \frac{H-|\delta|}{2H-1}\cdot\frac{1}{A\cdot SL(\delta)}\cdot\sqrt{\frac{2ENBW}{M}}\cdot\sqrt{\frac{2(4H-3)\left(\delta^2-|\delta|\right)+2H^2-1}{2H-1}}\cdot\sigma_r \\ &= \frac{2(H-|\delta|)}{(2H-1)!}\cdot\frac{\pi\delta}{\sin(\pi\delta)}\cdot\frac{\prod_{h=1}^{H-1}\left(h^2-\delta^2\right)}{A}\cdot\sqrt{\frac{C_{4H-4}^{2H-2}}{2M}}\cdot\sqrt{\frac{2(4H-3)\left(\delta^2-|\delta|\right)+2H^2-1}{2H-1}}\cdot\sigma_r,\end{aligned} \tag{10.44}$$

where *ENBW* is the window equivalent noise bandwidth [17], expressed as [14]:

$$ENBW \triangleq \frac{M\sum_{n=0}^{M-1}w^2(n)}{\left(\sum_{n=0}^{M-1}w(n)\right)^2} = 1+0.5\sum_{h=1}^{H-1}\left(\frac{a_h}{a_0}\right)^2 = \frac{C_{4H-4}^{2H-2}}{\left(C_{2H-2}^{H-1}\right)^2}, \tag{10.45}$$

and $SL(\cdot)$ is the window Scalloping Loss [17], defined as [14]:

$$SL(\delta) \triangleq \frac{|W(\delta)|}{|W(0)|} = \frac{|W(\delta)|}{M\,a_0} = \frac{\sin(\pi\delta)}{\pi\delta}\cdot\frac{[(H-1)!]^2}{\prod_{h=1}^{H-1}\left(h^2-\delta^2\right)}. \tag{10.46}$$

Expressions (10.43) and (10.44) provide the combined standard deviation of the IpDFT frequency estimator as [18]:

$$\sigma_c = \sqrt{\frac{\left|\Delta_{\delta_{ip}}\right|_{\max}^2}{2}+\sigma_{\hat{\delta}_{ip}}^2}, \tag{10.47}$$

which maximum, $\sigma_{c\ max}$, is reached for $\delta = -0.5$ [14].

It is worth noticing that the values of $\sigma_{\hat{\delta}_{ip}}$ provided by (10.44) decrease as H decreases, whereas (10.43) shows that $\left|\Delta_{\delta_{ip}}\right|_{\max}$ decreases as H increases. Thus, there exists a value of H that ensures a minimum value for $\sigma_{c\ max}$, i.e. an optimal number of terms H_{opt}. In order to determine H_{opt}, let us define ${}_H(\sigma_{c\max})_{\sigma_r=\sigma_q}$, as the value of $\sigma_{c\ \max}$ achieved using the H-term MSD window when $\sigma_r = \sigma_q$. Then, since $\sigma_r \geq \sigma_q$, the following proposition holds [3]:

Proposition *If* $_H(\sigma_{c\max})_{\sigma_r=\sigma_q} <_{H+1}(\sigma_{c\max})_{\sigma_r=\sigma_q}$, *then* $_H(\sigma_{c\max})_{\sigma_r} <_{H+1}(\sigma_{c\max})_{\sigma_r}$ *for any* $\sigma_r \geq \sigma_q$.

This implies that the optimal MSD window to be used in the IpDFT method can be selected by means of the following two-step procedure:

- if $_2(\sigma_{c\max})_{\sigma_r=\sigma_q} <_3(\sigma_{c\max})_{\sigma_r=\sigma_q}$, then $H_{opt} = 2$, that is the optimal MSD window is the two-term or Hann window.
- if $_2(\sigma_{c\max})_{\sigma_r=\sigma_q} >_3(\sigma_{c\max})_{\sigma_r=\sigma_q}$, then we continue the comparison by increasing the value of H until $_{H^*}(\sigma_{c\max})_{\sigma_r=\sigma_q} <_{H^*+1}(\sigma_{c\max})_{\sigma_r=\sigma_q}$, $H = 3, 4,\ldots$ The first value of H satisfying the previous relationship provides the optimal MSD window, that is $H_{opt} = H^*$.

  For values of J used in practice (e.g. $J \geq 15$) we usually obtain $H_{opt} = 2$ or 3.

10.4.2 Energy-based Method

Another frequency-domain method providing fast and accurate estimates of the best sine-wave parameters is the so-called Energy-Based (EB) method [8, 19, 20].

The estimator for the fractional part δ of the normalized sine-wave frequency is given by [19]:

$$\hat{\delta}_{eb} = \frac{\sum_{k=-H}^{H} k|Y_w(J+k)|^2}{\sum_{k=-H}^{H} |Y_w(J+k)|^2}. \tag{10.48}$$

It is worth noticing that the frequency estimator (10.48) exhibits quite low accuracy when two-term cosine windows are used. Thus, only windows with higher number of terms are adopted in the EB method [20].

Moreover, the sine-wave amplitude and phase can be estimated either directly from selected spectral samples [19] or indirectly by applying the same expressions of the IpDFT method and using the frequency estimate returned by (10.48) [4]. In the following, only the indirect procedure is analysed. Thus we have:

$$\hat{A}_{eb} = \frac{2|Y_w(J)|}{\left|W\left(-\hat{\delta}_{eb}\right)\right|}, \tag{10.49}$$

and:

$$\hat{\varphi}_{eb} = \arg[Y_w(J)] - \pi\hat{\delta}_{eb} + \pi\frac{\hat{\delta}_{eb}}{M} - \frac{\pi}{2}sign(\hat{\delta}_{eb}) - \arg\left[W_0\left(-\hat{\delta}_{eb}\right)\right], \tag{10.50}$$

where $W(\cdot)$ and $W_0(\cdot)$ are defined in (10.32) and (10.31), respectively.

The offset B is finally estimated by means of the same expression used by the IpDFT method, that is:

$$\hat{B}_{eb} = \frac{Y_w(0)}{M \cdot NPSG}. \tag{10.51}$$

The accuracy of the estimator (10.48) is mostly affected by the algorithm error, the spectral interference due to the image component, and wide-band noise [20]. The best accuracy is usually achieved by adopting three- or four-term cosine windows [4]. Among the most commonly used three-term windows, optimal results are achieved with the rapid sidelobe decay 18 dB/octave (rsd18) window [20], for which $a_0 = 0.40897$, $a_1 = 0.5$, and $a_2 = 0.09103$ [21]. Differently, the four-term cosine window providing the most accurate results is the rapid sidelobe decay 30 dB/octave (rsd30) window [20], characterized by $a_0 = 0.338946$, $a_1 = 0.481973$, $a_2 = 0.161054$, and $a_3 = 0.018027$ [21].

10.5 Three- and Four-Parameter Algorithms

The three-parameter and the four-parameter algorithms determine the best sine fit of the analyzed data by using the least squares approach. Specifically, the best sine-wave parameters are determined by minimizing the following sum of squared residuals [1, 2, 9, 22]:

$$\gamma = \sum_{n=0}^{M-1} \hat{r}^2(n) = \sum_{n=0}^{M-1} [y(n) - \hat{s}(n)]^2. \tag{10.52}$$

Both algorithms are briefly described in the following subsections.

10.5.1 Three-Parameter Algorithm

In the three-parameter algorithm, the signal normalized frequency $\hat{v}$ is assumed to be known and the amplitude, phase, and offset providing the best sine fit are to be determined. Since the signal frequency is known, the cost function γ can be minimized by using a simple linear least squares procedure, which provides a closed form analytical expression for all the unknown parameters. As a consequence, the computational effort required by the algorithm is very low.

The sine fit $\hat{s}(\cdot)$ expressed by (10.3) can be rewritten as:

$$\hat{s}(n) = \hat{A}_s \cos(\hat{\omega}\, n) + \hat{A}_c \sin(\hat{\omega}\, n) + \hat{B}, \quad n = 0, 1, \ldots, M-1 \tag{10.53}$$

where $\hat{\omega} = 2\pi\hat{v}/M$ is known, while $\hat{A}_s = \hat{A}\sin(\hat{\varphi})$, $\hat{A}_c = \hat{A}\cos(\hat{\varphi})$, and $\hat{B}$ are unknown. For the sake of notation, we represent the unknown parameters as the vector $x = \left[\hat{A}_s\ \hat{A}_c\ \hat{B}\right]^T$.

The three-parameter algorithm is based on the following four steps procedure:

1. Acquire M consecutive samples of the ADC output signal $y(n)$, $n = 0, 1, 2,\ldots, M-1$, and construct the vector:

$$y = [y(0)\ y(1)\ldots y(M-1)]^T \tag{10.54}$$

2. Use the known frequency $\hat{\omega}$ to construct the $M \times 3$ matrix D with the following entries:

$$\begin{cases} D_{k1} = \cos((k-1)\hat{\omega}) \\ D_{k2} = \sin((k-1)\hat{\omega}) \\ D_{k3} = 1 \end{cases} \tag{10.55}$$

where $k = 1, 2,\ldots, M$.

3. Determine the solution of the set of equations $\boldsymbol{y} = \boldsymbol{Dx}$ that minimize the cost function (10.52):

$$\hat{x} = \left(D^T D\right)^{-1} D^T y. \tag{10.56}$$

4. Determine the unknown sine-wave parameters as follows:

$$\hat{A} = \sqrt{\hat{A}_s^2 + \hat{A}_c^2}, \tag{10.57}$$

$$\hat{\varphi} = \arctan\left(\frac{\hat{A}_s}{\hat{A}_c}\right), \tag{10.58}$$

and the offset is equal to $\hat{B}$.

The uncertainty of the estimates (10.57) and (10.58) depends on the accuracy of the sine-wave frequency $\hat{v}$ [1, 2, 9, 22]. It can be shown that accurate *ENOB* estimates can be achieved if the normalized frequency error $\Delta_v = \hat{v} - v$ satisfies the following constraint [23]:

$$|\Delta_v| < \frac{\sqrt{2}}{\pi 2^{N+2}}. \tag{10.59}$$

This constraint can be satisfied when the sine-wave frequency is estimated by means of the IpDFT method based on the MSD window, i.e. by using (10.39), and the adopted window is selected according to the criterion presented in Sect. 10.4.1. Also, the EB method can be used to achieve accurate estimates of the sine-wave frequency.

10.5.2 Four-Parameter Algorithm

In the four-parameter algorithm, all the sine-wave parameters are assumed to be unknown. Thus, the minimization of the cost function (10.52) requires the solution of a nonlinear problem, which must be performed by means of an iterative process. It follows that the required computational effort is quite high and the convergence to the optimal solution is not always ensured.

The best sine fit $\hat{s}(\cdot)$ expressed by (10.3) can be rewritten as:

$$\hat{s}(n) = \hat{A}_s \cos(\hat{\omega}\, n) + \hat{A}_c \sin(\hat{\omega}\, n) + \hat{B}, \quad n = 0,\, 1, \ldots,\, M-1 \tag{10.60}$$

where $\hat{\omega} = 2\pi\hat{\nu}/M$, $\hat{A}_s = \hat{A}\sin(\hat{\varphi})$, $\hat{A}_c = \hat{A}\cos(\hat{\varphi})$, and $\hat{B}$ are unknown.

A first-order Taylor series expansion around the estimate $\hat{\omega}_i$ achieved at the i-th iteration gives:

$$\cos(\hat{\omega}n) \cong \cos(\hat{\omega}_i n) - n\sin(\hat{\omega}_i n)\Delta\omega_i, \tag{10.61}$$

and:

$$\sin(\hat{\omega}n) \cong \sin(\hat{\omega}_i n) + n\cos(\hat{\omega}_i n)\Delta\omega_i, \tag{10.62}$$

where $\Delta_{\omega_i} = \hat{\omega} - \hat{\omega}_i$.

Using (10.60), (10.61), and (10.62), the best sine fit $\hat{s}(\cdot)$ can be estimated as:

$$\hat{s}(n) \cong \hat{A}_s \cos(\hat{\omega}_i\, n) + \hat{A}_c \sin(\hat{\omega}_i\, n) + \hat{B} + \left[-n\hat{A}_s \sin(\hat{\omega}_i\, n) + n\hat{A}_c \cos(\omega_i\, n)\right]\Delta_{\omega_i}, \quad n = 0,\, 1, \ldots,\, M-1 \tag{10.63}$$

in which $\hat{A}_s = \hat{A}\sin(\hat{\varphi})$ and $\hat{A}_c = \hat{A}\cos(\hat{\varphi})$.

In the i-th iteration the unknown parameters $\hat{A}_s$, $\hat{A}_c$, and $\hat{B}$ in (10.63) are substituted by their estimates achieved in the previous iteration $\hat{A}_{s_{i-1}}$, $\hat{A}_{c_{i-1}}$, and $\hat{B}_{i-1}$.

Representing the unknown parameters as the vector $x_i = \left[\hat{A}_{si}\ \hat{A}_{ci}\ B_i\ \Delta_{\omega i}\right]^T$, the four-parameter algorithm can be described by the following eight steps procedure:

1. Acquire M consecutive samples of the ADC output signal $y(n)$, $n = 0, 1, 2,\ldots, M-1$, and construct the vector:

$$y = [y(0)\ y(1)\ldots y(M-1)]^T. \tag{10.64}$$

2. Set iteration index $i = 0$. Provide the initial estimates $\hat{A}_{s_0}$, $\hat{A}_{c_0}$, and $\hat{\omega}_0$ for the unknown parameters.
3. Start the next iteration, i.e. $i := i + 1$.
4. Construct the $M \times 4$ matrix $\boldsymbol{D}_i$ with the following entries:

$$\begin{cases} (D_i)_{k1} = \cos((k-1)\hat{\omega}_{i-1}) \\ (D_i)_{k2} = \sin((k-1)\hat{\omega}_{i-1}) \\ (D_i)_{k3} = 1 \\ (D_i)_{k4} = -\hat{A}_{s_{i-1}}(k-1)\sin((k-1)\hat{\omega}_{i-1}) + \hat{A}_{c_{i-1}}(k-1)\cos((k-1)\hat{\omega}_{i-1}) \end{cases} \tag{10.65}$$

where $k = 1, 2, \ldots, M$.

5. Determine the solution $\boldsymbol{x_i}$ of the set of equations $\boldsymbol{y} = \boldsymbol{D_i}\,\boldsymbol{x_i}$ that minimize the cost function (10.52) at the iteration i:

$$\hat{x}_i = \left(D_i^T D_i\right)^{-1} D_i^T y. \tag{10.66}$$

6. Determine the parameters $\hat{A}_{s_i}$, $\hat{A}_{c_i}$, $\hat{B}_i$, and Δ_{ω_i} related to the *i-th* iteration.
7. Update the frequency value:

$$\hat{\omega}_i = \hat{\omega}_{i-1} + \Delta_{\omega i}. \tag{10.67}$$

8. Repeat the steps (3)–(7) until the relative distance between two successive iterations $\left|\frac{\Delta_{\omega_i}}{\omega_i}\right|$ is smaller than a suitable threshold. When that constraint is fulfilled, the iterations are stopped and the sine-wave parameters provided by the last iteration are returned, that are:

$$\hat{A} = \sqrt{\hat{A}_{si}^2 + \hat{A}_{ci}^2}, \tag{10.68}$$

$$\hat{\varphi} = \arctan\left(\frac{\hat{A}_{s_i}}{\hat{A}_{c_i}}\right), \tag{10.69}$$

and:

$$\hat{B} = \hat{B}_i. \tag{10.70}$$

The performance of the four-parameter algorithm strongly depends on the accuracy of the initial estimates [1, 2, 9, 22]. They should be properly chosen to avoid convergence to local minima. Very good performance is achieved when the initial frequency estimate $\hat{\omega}_0$ is determined by means of the IpDFT method based on the rectangular window and $\hat{A}_{s_0}$ and $\hat{A}_{c_0}$ are achieved by the three-parameter algorithm [24] or when all the initial estimates are determined by means of the IpDFT method [3]. In this latter case they are given by [25] (see expressions derived in the Sect. 10.4.1 for $H = 1$):

$$\hat{\omega}_0 = 2\pi\left(\frac{J+\hat{\delta}_0}{M}\right), \tag{10.71}$$

$$\hat{A}_{s_0} = \frac{2\pi\hat{\delta}_0|Y(J)|}{M\sin\left(\pi\hat{\delta}_0\right)}\sin(\hat{\varphi}_0), \tag{10.72}$$

$$\hat{A}_{c_0} = \frac{2\pi\hat{\delta}_0|Y(J)|}{M\sin\left(\pi\hat{\delta}_0\right)}\cos(\hat{\varphi}_0), \tag{10.73}$$

in which $\hat{\delta}_0 = \frac{i\alpha-1+i}{\alpha+1}$, $\alpha = \frac{|Y(J+i)|}{|Y(J-1+i)|}$, and $\hat{\varphi}_0 = \arg[Y(J)] - \pi\hat{\delta}_0 + \pi\frac{\hat{\delta}_0}{M} + \frac{\pi}{2}$, where $Y(\cdot)$ is the DTFT of the ADC output signal (10.1).

Simulations show that for all actual ADCs, the error $|\Delta ENOB|$ achieved by using the above algorithm is smaller than 0.1 bits when $J \geq 10$ and $M \geq 1024$.

10.6 Sine-Fitting Algorithms Accuracy Comparison

In this Section, the accuracy of the SFAs based on the IpDFT and EB methods are compared with those achieved by the 3PSFA and 4PSFA through both theoretical and simulation results. The analysis is performed assuming that the overall ADC output noise can be modelled as a zero mean white Gaussian noise.

The fitting error is defined as:

$$e(n) = \hat{s}(n) - s(n), \quad n = 0, 1, \ldots, M-1 \tag{10.74}$$

From (10.1), (10.4), and (10.74) we obtain:

$$\hat{r}(n) = r(n) - e(n), \quad n = 0, 1, \ldots, M-1 \tag{10.75}$$

The accuracies of the aforementioned algorithms will be evaluated by means of the sum-squared of fitting and residual errors. The former error is defined as:

$$\varepsilon = \frac{1}{M}\sum_{n=0}^{M-1} e^2(n). \tag{10.76}$$

The above expression can be accurately estimated as [3]:

$$\varepsilon \cong \Delta_B^2 + \frac{\Delta_A^2}{2} + \frac{2\pi^2 A^2 \Delta_v^2}{3} + \frac{A^2\Delta_\varphi^2}{2} + A^2\pi\Delta_v\Delta_\varphi, \tag{10.77}$$

where $\Delta_A = \hat{A} - A$, $\Delta_v = \hat{v} - v = \hat{\delta} - \delta$, $\Delta_\varphi = \hat{\varphi} - \varphi$, $\Delta_B = \hat{B} - B$ are the parameter estimation errors, which can be modeled as random variables [3].

For values of M often used in ADCs testing (e.g. $M \geq 512$) the estimators provided by the considered SFAs are almost unbiased [3, 9]. Thus, the expected sum-squared fitting error results:

$$E[\varepsilon] \cong \sigma_{\hat{B}}^2 + \frac{\sigma_{\hat{A}}^2}{2} + \frac{2\pi^2 A^2 \sigma_{\hat{v}}^2}{3} + \frac{A^2 \sigma_{\hat{\varphi}}^2}{2} + \pi A^2 E\left[\Delta_v \Delta_\varphi\right], \tag{10.78}$$

where $\sigma_{\hat{A}}^2$, $\sigma_{\hat{v}}^2$, $\sigma_{\hat{\varphi}}^2$, and $\sigma_{\hat{B}}^2$ represent the variances of the estimators provided by the considered SFAs.

The relationship between the sine-wave phases at the beginning (that is for $n = 0$) and at the center of the observation interval, respectively φ and φ_s, is given by:

$$\varphi = \varphi_s - \pi v. \tag{10.79}$$

Since the φ_s and v estimators are asymptotically uncorrelated [13, 26], (10.79) provides:

$$E\left[\Delta_v \Delta_\varphi\right] \cong -\pi\, E\left[\Delta_v^2\right] = -\pi\, \sigma_{\hat{v}}^2, \tag{10.80}$$

and:

$$\sigma_{\hat{\varphi}}^2 = \sigma_{\hat{\varphi}_s}^2 + \pi^2 \sigma_{\hat{v}}^2. \tag{10.81}$$

Using (10.80) and (10.81), (10.78) becomes:

$$E[\varepsilon] \cong \sigma_{\hat{B}}^2 + \frac{\sigma_{\hat{A}}^2}{2} + \frac{\pi^2 A^2 \sigma_{\hat{v}}^2}{6} + \frac{A^2 \sigma_{\hat{\phi}_s}^2}{2}. \tag{10.82}$$

In the following, the expression (10.82) will be evaluated for each of the considered algorithms.

- Sine-fitting based on the IpDTF or EB methods

When the IpDFT or the EB methods are employed, we have [3, 14]:

$$\sigma_{\hat{B}}^2 \cong ENBW \frac{\sigma_r^2}{M}, \tag{10.83}$$

$$\sigma_{\hat{A}}^2 \cong \frac{2ENBW}{SL^2(\delta)} \frac{\sigma_r^2}{M}, \tag{10.84}$$

and:

$$\sigma_{\hat{\varphi}_s}^2 \cong \frac{2ENBW}{M A^2 SL^2(\delta)} \sigma_r^2, \tag{10.85}$$

where the window parameters *ENBW* and $SL(\delta)$ are defined by the first equality in (10.45) and (10.46), respectively.

Using (10.83–10.85), expression (10.82) becomes:

$$E[\varepsilon] \cong ENBW\left(1+\frac{2}{SL^2(\delta)}\right)\frac{\sigma_r^2}{M}+\frac{\pi^2A^2\sigma_{\hat{v}}^2}{6}. \tag{10.86}$$

Thus, the expected sum-squared fitting error that occurs when using the IpDFT method can be expressed as:

$$E\left[\varepsilon_{ip}\right] \cong ENBW\left(1+\frac{2}{SL^2(\delta)}\right)\frac{\sigma_r^2}{M}+\frac{\pi^2A^2\sigma_{\hat{v}_{ip}}^2}{6}, \tag{10.87}$$

where $\sigma_{\hat{v}_{ip}}^2$ is given by (10.44).

It is worth noticing that the minimum and the maximum values of $\sigma_{\hat{v}_{ip}}^2$ are reached for δ equal to -0.5 and 0, respectively [14]. Conversely, $SL(\delta)$ assumes its maximum value (equal to 1) for $\delta = 0$, but it remains very close to it for $-0.5 \le \delta < 0.5$. Hence, the minimum and the maximum values of the $E[\varepsilon_{ip}]$ are reached for δ equal to -0.5 and 0, respectively.

Similarly, the expected sum-squared fitting error when using the EB method can be expressed as:

$$E[\varepsilon_{eb}] \cong ENBW\left(1+\frac{2}{SL^2(\delta)}\right)\frac{\sigma_r^2}{M}+\frac{\pi^2A^2\sigma_{\hat{v}_{eb}}^2}{6}, \tag{10.88}$$

in which the variance $\sigma_{\hat{v}_{eb}}^2$ is given by [20]:

$$\sigma_{\hat{v}_{eb}}^2 \cong \frac{16}{M\cdot NNPG}\left[2\sum_{i=1}^{H-2}\sum_{j=i+1}^{H-1} ijb_ib_j\,r(i,j)+\sum_{i=-H+1}^{-1}\sum_{j=1}^{H-1} ijb_{|i|}b_{|j|}\,r(i,j)+\sum_{i=1}^{H-1} i^2b_i^2\right]\frac{\sigma_r^2}{A^2}. \tag{10.89}$$

In (10.89) *NNPG* is the window noise normalized power gain [17], given by:

$$NNPG \triangleq \frac{\sum_{n=0}^{M-1} w^2(n)}{M} = a_0^2+0.5\sum_{h=1}^{H-1} a_h^2, \tag{10.90}$$

$$b_i = \begin{cases} a_0, & for \quad i=0 \\ 0.5a_i, & for \quad i=1,2,\ldots,H-1 \\ 0, & for \quad i\ge H \end{cases} \tag{10.91}$$

and $r(i,j)$ is the correlation coefficient between the spectral samples $|Y_w(J+i)|$ and $|Y_w(J+j)|$ [27]:

$$r(i,j) = r(u) = \frac{\sum_{k=-H+1}^{H-1-u} b_{|k|} b_{|k+u|}}{NNPG}, \tag{10.92}$$

in which $u = |i - j|$.

Since $\sigma^2_{\hat{v}_{eb}}$ does not depend on δ, the minimum and the maximum values of $E[\varepsilon_{eb}]$ are reached for δ equal to 0 and -0.5, respectively.

- 3PSF and 4PSF algorithms

As already mentioned, in practice both the number of acquired samples M and the number of acquired sine-wave cycles v are quite high (e.g. $M \geq 512$ and $v \geq 15$). Thus, the 3PSFA and 4PSFA estimator variances almost attain their related single-tone Cramér-Rao lower bounds (CRLBs), that is [26]:

$$\sigma^2_{\hat{B}} \cong \frac{\sigma^2_r}{M}, \tag{10.93}$$

$$\sigma^2_{\hat{A}} \cong \frac{2\sigma^2_r}{M}, \tag{10.94}$$

$$\sigma^2_{\hat{\varphi}_s} \cong \frac{2\sigma^2_r}{A^2 M}, \tag{10.95}$$

$$\sigma^2_{\hat{v}} \cong \frac{6\sigma^2_r}{\pi^2 A^2 M}. \tag{10.96}$$

Using (10.93–10.95), expression (10.82) provides the expected sum-squared fitting error for the 3PSFA as:

$$E\left[\varepsilon_{3p}\right] \cong \frac{3\sigma^2_r}{M} + \frac{\pi^2 A^2 \sigma^2_{\hat{v}}}{6}. \tag{10.97}$$

When v is estimated by the IpDFT method, in (10.97) we have $\sigma^2_{\hat{v}} = \sigma^2_{\hat{v}_{ip}}$, where $\sigma^2_{\hat{v}_{ip}}$ is given by (10.44). Also, the minimum and the maximum values of the $E[\varepsilon_{3p}]$ are reached for δ equal to -0.5 and 0, respectively.

Conversely, using (10.93–10.96), expression (10.82) shows that the expected sum-squared fitting error for the 4PSFA is:

$$E\left[\varepsilon_{4p}\right] \cong \frac{4\sigma^2_r}{M}. \tag{10.98}$$

By comparing the expressions (10.87), (10.88), (10.97), and (10.98), it follows that: $E[\varepsilon_{4p}] < E[\varepsilon_{3p}] < E[\varepsilon_{ip}] < E[\varepsilon_{eb}]$. Thus, the time-domain SFAs provide more accurate sine-wave fitting than the frequency-domain SFAs. In particular, the best accuracy is provided by the 4PSFA, while the worst accuracy is achieved by the SFA-EB.

However, the above expressions show that the expected sum-squared fitting error is always proportional to σ_r^2/M. Hence, for values of M used in practice, it is negligible with respect to the noise variance σ_r^2. This implies that for any SFA, the expectation of the residual mean square value results very close to the noise variance:

$$E\left[\hat{r}_{rms}^2\right] \cong \sigma_r^2. \tag{10.99}$$

In order to confirm the above expressions, both theoretical and simulation results for the ratio $E[\varepsilon]/\left(\sigma_r^2/M\right)$ are reported in Fig. 10.1 as a function of δ for all the considered SFAs. The input sine-wave was characterized by the following parameters: $A = 5$, $\varphi = \pi/3$ rad, and $B = 0.02$. It was corrupted by additive Gaussian noise with zero mean and variance σ_r^2 chosen in such a way that the Signal-to-Noise Ratio (*SNR*) is equal to 60 dB. The integer part J was set to 37. The fractional part δ was varied in the range $[-0.5, 0.5)$ with a step of 1/20. For each value of δ, 10,000 runs of $M = 512$ samples each were performed. In the 3PSFA and the SFA-IpDFT, the Hann window was used, while the SFA-EB is based on the rsd18 window. The initial values used in the four-parameter algorithm were estimated through the IpDFT method based on the rectangular window, while the iterations were stopped when the relative distance between the frequency values estimated in two consecutive iterations was smaller than 10^{-6}.

As we can see, the agreement between theoretical and simulation results is very good.

In Fig. 10.2 both theoretical and simulation results of the ratio $E\left[\hat{r}_{rms}^2\right]/\sigma_r^2$ are reported as a function of δ. The adopted parameters were exactly the same as in the previous figure. As we can see, the ratio $E\left[\hat{r}_{rms}^2\right]/\sigma_r^2$ is always very close to 1. Indeed, the contribution of the fitting error to the residual signal is negligible, as suggested by the theoretical analysis.

All the considered SFAs were implemented in MATLAB running on a portable computer with a CPU clock rate of 2 GHz. Choosing $M = 512$, the average

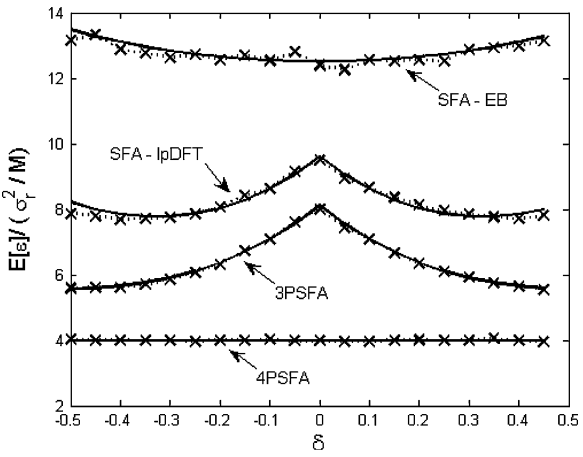

Fig. 10.1 Ratio $E[\varepsilon]/\left(\sigma_r^2/M\right)$ provided by theoretical (*continuous line*) and simulation (*crosses*) results versus. δ. The number of acquired samples was $M = 512$

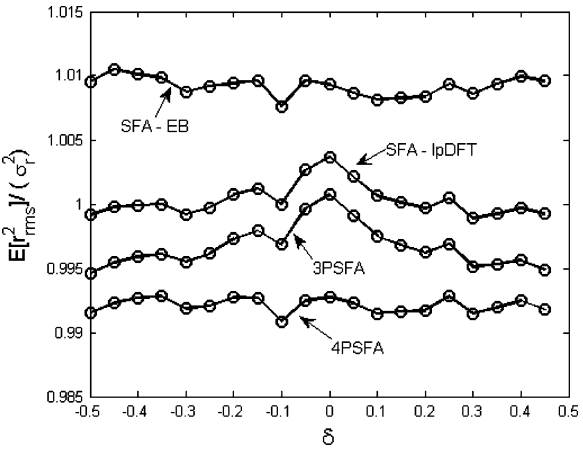

Fig. 10.2 Ratio $E\left[\hat{r}_{rms}^2\right]/\sigma_r^2$ provided by simulation results (*circles*) versus. δ. The number of acquired samples was $M = 512$

computational time required to determine a single residual mean square value $\hat{r}_{rms}^2$ for a given value of δ was equal to 0.47, 0.57, 0.74, and 1.28 ms, for the SFA-IpDFT, SFA-EB, 3PSFA, and 4PSFA, respectively. Thus, the SFA-IpDFT and the 4PSFA exhibit the smaller and the highest processing burden, respectively. The same conclusion holds regardless the number of acquired samples.

Thus, the SFA-IpDFT is a good choice when dealing with *ENOB* estimation.

10.7 Influence of Harmonics, Time Jitter, and Time Base Distortions

The detrimental effects of harmonics, time jitter, and time base distortions on an ADC performance increases with the input signal frequency. Thus, these effects must be taken into account when estimating the ADC dynamic parameters at high frequencies. In the following, some important issues concerning the estimation of harmonics, time jitter, and time base distortions are discussed.

10.7.1 Harmonics Estimation

When non-coherent sampling occurs, the 3PSFA and the 4PSFA provide biased estimates when harmonics affect the ADC output signal [28]. To improve the accuracy of the related procedures the Multiharmonics Sine-Fitting Algorithms (MHSFAs) have been proposed in the scientific literature [29–31]. They provide accurate estimates for the parameters of both the fundamental component and its harmonics. Hence, by using these algorithms, not only the *ENOB* and the *SINAD* parameters can be estimated, but also the Total Harmonic Distortion ratio (*THD*), and the Signal to Non-Harmonic Ratio (*SNHR*).

Two of the most accurate MHSFAs are proposed in [29, 30]. In the following we denote them as MHSFA1 and MHSFA2. Moreover, the latter one is denoted as MHSFA2a or MHSFA2b depending on whether the signal frequency is assumed to be known or unknown, similarly to the 3PSFA and the 4PSFA, respectively. Compared with the MHSFA1, the MHSFA2a and MHSFA2b have a higher robustness to inaccuracies in the signal frequency initial value, are faster, easier to implement [32], and can be applied even when the number of acquired samples is high, as needed when testing high resolution ADCs.

In the MHSFA2a and MHSFA2b the related best sine fit $\hat{s}(\cdot)$ is modelled as [30]:

$$\begin{aligned}\hat{s}(n) &= \hat{B} + \sum_{k=1}^{K} \hat{A}_k \sin\left(2\pi k\hat{v}\frac{n}{M} + \hat{\varphi}_k\right) \\ &= \hat{B} + \sum_{k=1}^{K} \left[\hat{A}_{sk}\cos(kn\hat{\omega}) + \hat{A}_{ck}\sin(kn\hat{\omega})\right], \quad n = 0, 1, \ldots, M-1\end{aligned} \tag{10.100}$$

where K represents the number of harmonics, $\hat{A}_k$ and $\hat{\varphi}_k$ are the amplitude and the phase of the k-th harmonic, $\hat{B}$ is the offset, $\hat{\omega} = {}^{2\pi\hat{v}}/_{M}$, $\hat{A}_{sk} = \hat{A}_k \sin(\hat{\varphi}_k)$, and $\hat{A}_{ck} = \hat{A}_k \cos(\hat{\varphi}_k)$.

When the frequency $\hat{\omega}$ is known the procedure to determine the unknown parameter vector $x = \left[\hat{A}_{s1}\ \hat{A}_{c1}\ \ldots \hat{A}_{sK}\ \hat{A}_{cK}\ \hat{B}\right]^T$ is an extension of the three-parameter algorithm. The entries of the related matrix $\boldsymbol{D}$, of size $M \times (2K + 1)$, are given by:

$$\begin{cases} D_{p(2q-1)} = \cos((p-1)q\hat{\omega}) \\ D_{p(2q)} = \sin((p-1)q\hat{\omega}) \\ D_{p(2K+1)} = 1 \end{cases} \tag{10.101}$$

where $p = 1, 2, \ldots, M$ and $q = 1, .., K$. The unknown parameters of the k-th signal spectral line ($k = 1, 2, 3, \ldots, K$) are derived as follows:

$$\hat{A}_k = \sqrt{\hat{A}_{sk}^2 + \hat{A}_{ck}^2} \tag{10.102}$$

$$\hat{\varphi}_k = \arctan\left(\frac{\hat{A}_{sk}}{\hat{A}_{ck}}\right), \tag{10.103}$$

and the offset is equal to $\hat{B}$.

It is worth noticing that the frequency $\hat{\omega}$ can be determined by means of the IpDFT method based on a MSD window chosen according to the criteria presented in Sect. 10.4.1 [31, 32].

Opposite, when the frequency $\hat{\omega}$ is considered unknown, a non-linear optimization procedure based on the four-parameter algorithm must be adopted and the unknown parameter vector employed in the i-th algorithm iteration is

$x_i = \left[\hat{A}_{s1_i}\ \hat{A}_{cl_i}\ \ldots \hat{A}_{sK_i}\ \hat{A}_{cK_i}\ \hat{B}_i\ \Delta_{\omega_i}\right]^T$, where $\Delta_{\omega_i} = \hat{\omega} - \hat{\omega}_i$. The entries of the matrix $\boldsymbol{D_i}$, of size $M \times (2\ K + 2)$, are given by:

$$\begin{cases} (D_i)_{p(2q-1)} = \cos((p-1)q\hat{\omega}_{i-1}) \\ (D_i)_{p(2q)} = \sin((p-1)q\hat{\omega}_{i-1}) \\ (D_i)_{p(2K+1)} = 1 \\ (D_i)_{p(2K+2)} = \sum_{k=1}^{K} \left[-\hat{A}_{sk_{i-1}}(p-1)k\sin((p-1)k\hat{\omega}_{i-1}) + \hat{A}_{ck_{i-1}}(p-1)k\cos((p-1)k\hat{\omega}_{i-1})\right] \end{cases} \tag{10.104}$$

where $p = 1, 2, \ldots, M$ and $q = 1, .., K$. The unknown parameters of the k-th signal spectral line ($k = 1, 2, 3, \ldots, K$) obtained by the *i-th* iteration ($i \geq 1$) are given by:

$$\hat{A}_k = \sqrt{\hat{A}_{sk_i}^2 + \hat{A}_{ck_i}^2} \tag{10.105}$$

$$\hat{\varphi}_k = \arctan\left(\frac{\hat{A}_{sk_i}}{\hat{A}_{ck_i}}\right), \tag{10.106}$$

and:

$$\hat{B} = \hat{B}_i. \tag{10.107}$$

The above procedure is repeated until the relative distance $\left|\frac{\Delta_{\omega_i}}{\omega_i}\right|$ is smaller than a suitable small threshold [31].

Of course, the initial value for the frequency $\hat{\omega}$ can be estimated by means of the IpDFT method based on a MSD window chosen according to the criteria presented in Sect. 10.4.1, and the initial estimates for the remaining signal parameters can be determined by using the MHSFA2a [31, 32].

Moreover, using the values returned by the previous expressions, the following parameters can be easily estimated [1, 33]:

$$THD\ (dB) = 20\log_{10}\left(\frac{\sqrt{\sum_{k=2}^{K}\hat{A}_k^2}}{\hat{A}_1}\right), \tag{10.108}$$

$$SNHR\ (dB) = 20\log_{10}\left(\frac{\hat{A}_1}{\sqrt{2\hat{r}_{rms}^2 - \sum_{k=2}^{K}\hat{A}_k^2}}\right), \tag{10.109}$$

where the residual error $\hat{r}_{rms}^2$ is given by (10.5).

The processing times required by the MHSFA1, the MHSFA2a, and the MHSFA2b were also analyzed. To this aim they were implemented in MATLAB and the same portable computer used for the SFAs analyzed in the previous section was employed to test a 12-bit ADC. The ADC output signal was synthesized by adding a sine-wave, its second, third, and fourth harmonics, and wide-band noise, resulting from the superposition of both quantization noise and Gaussian noise. The harmonics power and the noise variance were chosen in such a way that $ENOB = 10.5$ bits were obtained. The number of analyzed samples was set to $M = 512$ and the number of analyzed sine-wave cycles was $v = 72.5$. The initial values of the parameters used by the algorithms were estimated through the IpDFT method based on the Hann window. The MHSFA1 and the MHSFA2b were stopped when the relative distance between the frequency values estimated in two consecutive iterations was smaller than 10^{-6}. The average computational time required to determine a single residual mean square value $\hat{r}^2_{rms}$ and the harmonic parameters was equal to 53.6, 5.1, and 14.5 ms, for the MHSFA1, MHSFA2a, and MHSFA2b, respectively. Thus, the MHSFA1 exhibits the highest processing burden. Also, all the considered MHSFAs require a much higher computational burden than the SFA, as expected.

10.7.2 Time Jitter and Time Base Distortion Estimations

Time base errors exhibit both a random component (time jitter) and a systematic component (time base distortions). In an ADC dynamic three different jitter components contribute to the overall time jitter: two or them are due to the test setup (input signal jitter and sampling clock jitter), while the third one is related to the ADC under test (sampling circuit jitter) [34]. The time jitter spreads the input sine-wave energy over a range of frequencies and it acts as a low-pass filter on the average spectra [35, 36]. Conversely, the time base distortions yield to new spectral components in the ADC output signal spectrum [33, 37]. In particular, the *ENOB* and *SINAD* parameters decrease as the time jitter standard deviation or the time base distortions increase [1].

The standard deviation of the time jitter can be estimated by [38]:

$$\hat{\sigma}_t = \frac{\sqrt{\hat{r}^2_{rms_2} - \hat{r}^2_{rms_1}}}{\sqrt{2}\pi A\sqrt{f_2^2 - f_1^2}}, \tag{10.110}$$

where $\hat{r}^2_{rms_1}$ and $\hat{r}^2_{rms_2}$ are the rms of the residual errors (see (10.5)) achieved at the frequencies f_1 and f_2 (with $f_2 > f_1$), respectively.

Assuming a coherent sampling and null harmonics the estimator (10.110) is consistent [38]. However, since at high frequencies both harmonics and time base distortions are significant and (10.110) includes their effects in practice it provides only an upper bound to the time jitter standard deviation [33]. Several other methods have been proposed for the estimation of the time jitter standard deviation [36, 39, 40]. Moreover, a sine fit procedure able to take into account all the

considered phenomena, that are harmonics, time base distortions, and time jitter has been proposed in [41]. This is a complex procedure which combines the MHSFA proposed in [29] and the time base distortions measurement technique presented in [42].

10.8 Applications of the Sine-Fitting Algorithms in ADC Testing

The accuracies of the *ENOB* estimates achieved by means of all the considered sine-fitting algorithms have been compared using real-world data. The three- and four-parameter algorithms presented above, the IpDFT and EB methods, and the multi-parameter algorithms proposed in [29, 30], were implemented in MATLAB. They are named as follows: *sf3p.m* (three-parameter algorithm), *sf4p.m* (four-parameter algorithm), *ipdft1.m* (IpDFT method based on the rectangular window), *ipdft2.m* (IpDFT method based on the Hann window), *ebm.m* (EB method based on the rsd18 window), *sfmh1.m* (multi-harmonics sine-fitting algorithm proposed in [29]), *sfmh2a.m* (multi-harmonics sine-fitting algorithm proposed in [30] with known frequency), and *sfmh2b.m* (multi-multiharmonics sine-fitting algorithm proposed in [30] with unknown frequency). The related code is freely available at the book support website.

The *SINAD* and *ENOB* estimates were achieved by implementing the relationships (10.6) and (10.7). For example, if we wish to achieve these estimates for a 12-bit ADC with $FSR = 10$ V, by means of the 3PSFA, we can use the following very simple program implemented in MATLAB, in which the *adc.dat* file contains the values of the acquired samples:

```
load adc.dat;
y=adc;

M=max(size(y));
FSR=10;
N=12;
Q=FSR/2^N;
sq=Q/sqrt(12);

param=sf3p(y);
offset=param(1);
lambda=param(2);
Amp=param(3);
phase=param(4);

err=y-Amp*sin(2*pi*lambda*(0:M-1)/M+phase)-offset;
rms=sqrt(sum(err.^2)/M);
SINAD=10*log10(Amp^2/(2*rms^2));
ENOB=N-((log10(rms/sq))/log10(2));

disp(SINAD);
disp(ENOB);
```

The considered SFAs were applied to data acquired using a 12-bit data acquisition board NI-6023E, developed by the National Instruments. This board allows a maximum sampling rate of 200 kHz and contains a 12-bit bipolar successive approximation ADC. The *FSR* and the sampling frequency were set to 10 V and 100 kHz, respectively.

The mean and the standard deviation of the *ENOB* estimates achieved by all the considered SFAs are reported in Table 10.1 for different values of the fractional frequency δ. Conversely, the statistical parameters achieved when using the considered MHSFAs are reported in Table 10.2.

A signal generator Agilent 33220A was employed. The signal amplitude was set to 5 V while the frequency was varied in the range [14.70, 14.79] kHz with a step of 10 Hz. The achieved value of J was equal to 151. According to the criteria presented in Sect. 10.4.1, the Hann window was used for the 3PSFA and the SFA-IpDFT, while the rsd18 window was adopted for the SFA-EB. The parameters used in the 4PSFA were the same as in Fig. 10.1. The initial values for the parameters used in the considered MHSFAs were estimated through the IpDFT method based on the Hann window. The MHSFA1 and the MHSFA2b were stopped when the relative distance between the frequency values returned in two consecutive iterations was smaller than 10^{-6}. The number of harmonics adopted in the considered MHSFAs was set to 5, since higher order harmonics were negligible.

For each value of δ, 1000 runs of $M = 1024$ samples each were performed. The values of the fractional frequency δ reported in Tables 10.1 and 10.2 represent the mean value returned by the IpDFT method.

Coherently with the theoretical results, both Tables 10.1 and 10.2 show that the standard deviations are negligible with respect to the mean values, which are always very close to each other regardless the value of δ. This implies that harmonics did not significantly affect the accuracy of the *ENOB* estimates provided by

Table 10.1 *ENOB* mean and standard deviation achieved at different values of δ by all the considered SFAs

δ	SFA-IpDFT		SFA-EB		3PSFA		4PSFA	
	Mean (bits)	Standard deviation (bits)	Mean (bits)	Standard deviation (bits)	Mean (bits)	Standard deviation (bits)	Mean (bits)	Standard deviation (bits)
−0.471	10.685	0.042	10.683	0.043	10.686	0.042	10.687	0.042
−0.368	10.687	0.045	10.685	0.045	10.689	0.045	10.689	0.045
−0.266	10.690	0.054	10.688	0.055	10.691	0.054	10.691	0.054
−0.163	10.684	0.043	10.683	0.043	10.685	0.043	10.686	0.043
−0.061	10.686	0.040	10.684	0.040	10.687	0.040	10.687	0.040
0.041	10.682	0.034	10.681	0.034	10.683	0.034	10.684	0.034
0.144	10.680	0.030	10.678	0.030	10.681	0.029	10.682	0.030
0.246	10.681	0.028	10.679	0.029	10.682	0.028	10.683	0.028
0.349	10.683	0.029	10.681	0.029	10.684	0.029	10.684	0.029
0.451	10.679	0.029	10.677	0.029	10.680	0.029	10.681	0.029

Table 10.2 *ENOB* mean and standard deviation achieved at different values of δ by all the considered MHSFAs

δ	MHSFA1		MHSFA2a		MHSFA2b	
	Mean (bits)	Standard deviation (bits)	Mean (bits)	Standard deviation (bits)	Mean (bits)	Standard deviation (bits)
−0.471	10.685	0.042	10.686	0.042	10.685	0.042
−0.368	10.687	0.045	10.689	0.045	10.687	0.045
−0.266	10.689	0.054	10.691	0.054	10.689	0.054
−0.163	10.684	0.043	10.685	0.043	10.684	0.043
−0.061	10.685	0.040	10.687	0.040	10.685	0.040
0.041	10.680	0.035	10.683	0.034	10.680	0.035
0.144	10.680	0.030	10.681	0.029	10.680	0.030
0.246	10.681	0.028	10.682	0.028	10.681	0.028
0.349	10.683	0.029	10.684	0.029	10.683	0.029
0.451	10.679	0.029	10.680	0.029	10.679	0.029

the 3PSFA and the 4PSFA. Moreover, it is worth noticing that the maximum relative difference between the experimental standard deviations and the values returned by (10.18) when $E\left[\hat{r}_{rms}^2\right]$ and $std\left[\hat{r}_{rms}^2\right]$ are determined from the acquired data, is less than 0.39 % for all the considered algorithms.

The mean and the standard deviation values of the *THD* and *SNHR* parameters estimated by means of (10.108) and (10.109) are reported in Table 10.3 for all the considered MHSFAs. In particular, for any given value of δ, all algorithms returned the same *THD* value, so only one column is reported for this parameter.

Table 10.3 shows that the standard deviations of the estimated *THD* and *SNHR* parameters are negligible with respect to the mean values, which are always close

Table 10.3 *THD* and *SNHR* mean and standard deviation achieved at different values of δ by all the considered MHSFAs

δ	MHSFAs		MHSFA1		MHSFA2a		MHSFA2b	
	THD		SNHR		SNHR		SNHR	
	Mean (dB)	Standard deviation (dB)	Mean (dB)	Standard deviation (dB)	Mean (dB)	Standard deviation (dB)	Mean (dB)	Standard deviation (dB)
−0.471	−71.31	0.28	67.61	0.34	67.62	0.34	67.61	0.34
−0.368	−71.31	0.27	67.63	0.37	67.64	0.37	67.63	0.37
−0.266	−71.29	0.27	67.65	0.45	67.67	0.45	67.65	0.45
−0.163	−71.28	0.27	67.61	0.35	67.62	0.35	67.61	0.35
−0.061	−71.29	0.26	67.62	0.33	67.63	0.33	67.62	0.33
0.041	−71.28	0.31	67.58	0.26	67.60	0.26	67.58	0.26
0.144	−71.27	0.30	67.58	0.21	67.59	0.21	67.58	0.21
0.246	−71.24	0.29	67.60	0.21	67.61	0.21	67.60	0.21
0.349	−71.28	0.28	67.60	0.21	67.61	0.21	67.60	0.21
0.451	−71.25	0.29	67.59	0.21	67.59	0.21	67.59	0.21

Table 10.4 *ENOB* mean and standard deviation achieved at different frequencies by all the considered SFAs

F_{in} (kHz)	SFA-IpDFT		SFA-EB		3PSFA		4PSFA	
	Mean (bits)	Standard deviation (bits)	Mean (bits)	Standard deviation (bits)	Mean (bits)	Standard deviation (bits)	Mean (bits)	Standard deviation (bits)
3.1	10.837	0.031	10.836	0.031	10.837	0.031	10.838	0.031
7.9	10.790	0.031	10.789	0.031	10.791	0.031	10.792	0.031
12.3	10.699	0.033	10.698	0.033	10.699	0.033	10.700	0.033
17.5	10.544	0.030	10.543	0.030	10.544	0.030	10.545	0.030
22.1	10.391	0.021	10.390	0.021	10.391	0.021	10.391	0.021
26.7	10.264	0.021	10.263	0.021	10.264	0.021	10.265	0.021

Table 10.5 *ENOB* mean and standard deviation achieved at different frequencies by all the considered MHSFAs

f_{in} (kHz)	MHSFA1		MHSFA2a		MHSFA2b	
	Mean (bits)	Standard deviation (bits)	Mean (bits)	Standard deviation (bits)	Mean (bits)	Standard deviation (bits)
3.1	10.836	0.031	10.837	0.031	10.836	0.031
7.9	10.789	0.031	10.791	0.031	10.789	0.031
12.3	10.698	0.033	10.699	0.033	10.698	0.033
17.5	10.543	0.030	10.544	0.030	10.543	0.030
22.1	10.390	0.021	10.391	0.021	10.390	0.021
26.7	10.264	0.021	10.264	0.021	10.264	0.021

to each other. In particular, it is worth observing that the MHSFA1 and MHSFA2b provided the same results, as happened for the estimated *ENOBs*.

The mean and the standard deviation values of the *ENOB* estimates returned by the considered SFAs and MHSFAs are reported in Tables 10.4 and 10.5 for different values of the input sine-wave frequency. For each value of δ, 1000 runs of $M = 1024$ samples each were performed.

As expected from the theoretical results, the standard deviation is much smaller than the mean value and, at each considered frequency, all the achieved results are very close to each other. This implies that the effect of harmonics on the *ENOB* estimates provided by the 3PSFA and the 4PSFA is negligible. The maximum relative difference between the experimental standard deviation and the value returned by (10.18) is less than 0.40 % for the SFAs and less than 0.38 % for the MHSFAs.

The mean and the standard deviation values of the *THD* and *SNHR* estimates achieved by the considered MHSFAs are reported in Table 10.6. As above, for each given frequency all algorithms returned the same *THD* value, so only one column is reported for this parameter.

Again, the *THD* and the *SNHR* estimates returned by all the considered MHSFAs are very close to each other.

Table 10.6 *THD* and *SNHR* mean and standard deviation achieved at different frequencies by all the considered MHSFAs

f_{in} (kHz)	MHSFAs		MHSFA1		MHSFA2a		MHSFA2b	
	THD		SNHR		SNHR		SNHR	
	Mean (dB)	Standard deviation (dB)	Mean (dB)	Standard deviation (dB)	Mean (dB)	Standard deviation (dB)	Mean (dB)	Standard deviation (dB)
3.1	−83.55	1.51	67.09	0.19	67.10	0.19	67.10	0.19
7.9	−76.73	0.61	67.15	0.20	67.16	0.20	67.15	0.20
12.3	−72.97	0.61	67.16	0.19	67.17	0.19	67.16	0.19
17.5	−69.60	0.30	67.17	0.21	67.18	0.21	67.17	0.21
22.1	−67.44	0.16	67.16	0.19	67.17	0.19	67.16	0.19
26.7	−65.98	0.17	67.17	0.20	67.17	0.20	67.17	0.20

10.9 Conclusion

This Chapter has been focused on the estimation of the *ENOB* parameter of an ADC by means of both time- and frequency-domain SFAs. The analyzed frequency-domain SFAs are based on the IpDFT method (SFA-IpDFT) or the EB method (SFA-EB), while the three-parameter algorithm (3PSFA) and the four-parameter algorithm (4PSFA) have been considered when operating in the time-domain. In Sect. 10.3 the expressions for the bias and the standard deviation of the *ENOB* estimator provided by a SFA, have been derived. These expressions hold regardless the overall ADC output noise characteristics. In particular, when the overall ADC output noise is zero mean and white, as often occurs in practice when the input sine-wave frequency is well below the ADC sampling rate, it has been shown that the *ENOB* estimators provided by SFAs are statistically consistent. Also, the bias is negligible as compared to the standard deviations when the noise is Gaussian. In Sects. 10.4 and 10.5 the information needed to understand and easily implement the IpDFT method, the EB method, and the three- and four-parameter algorithms has been provided. In Sect. 10.6, it has been shown that the time-domain SFAs provide better sine-wave fitting accuracies when the overall ADC output noise is zero mean, white, and normally distributed. In particular, the best fitting accuracy is provided by the 4PSFA, followed by the 3PSFA, the SFA-IpDFT, and the SFA-EB respectively. However the sum-squared fitting error related to any considered SFA is negligible with respect to the ADC output noise variance. This implies that the *ENOB* estimates provided by all the considered SFAs exhibit almost the same accuracy. Ranking the algorithms with respect to the processing time, the SFA-IpDFT is the faster, followed by the SFA-EB, the 3PSFA, and the 4PSFA, respectively. Thus, the SFA-IpDFT is a good choice when dealing with *ENOB* estimation. In Sect. 10.7 the main issues concerning the effect of harmonics, time jitter, and time base distortions on the estimation accuracy are discussed. Since these effects become significant when estimating the ADC dynamic parameters at high frequencies, some procedures for their estimation have

been briefly introduced. In particular, it should be noticed, that even though the MHSFAs require a higher processing times than the SFAs, they can provide further important ADC dynamic parameters, like *THD* and *SNHR*. Finally, some experimental results have been presented in order to compare the different SFAs and MHSFAs accuracies. As expected from the theoretical analysis, the *ENOB* estimates provided by the 4PSFA, the 3PSFA, the SFA-IpDFT, and the SFA-EB exhibit very close mean values and standard deviations. Moreover, the experimental results validated the theoretical expression for the *ENOB* estimates standard deviation The MATLAB programs implementing the methods used by the considered SFAs and MHSFAs are freely available at the book support website.

References

1. IEEE Std. 1241, Standard for Terminology and Test Methods for Analog-to-Digital Converters. IEEE, New York December (2000)
2. European Project DYNAD. Methods and Draft Standards for the Dynamic Characterization and Testing of Analog-to-Digital Converters
3. Belega, D., Dallet, D., Petri, D.: A high-accuracy procedure for effective number of bits estimation in analog-to-digital converters. IEEE Trans. Instrum. Meas. **60**(5), 1522–1532 (2011)
4. Belega, D., Petri, D., Dallet, D.: Sine-fitting by the energy-based method in the dynamic testing of ADCs. In: Proceedings of 6th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, vol. 1, pp. 33–38. Prague, Czech Republic 15–17 Sept 2011
5. Kollár, I.: Bias of mean value and mean square value measurements based on quantized data. IEEE Trans. Instrum. Meas. **43**(5), 733–739 (1994)
6. Widrow, B., Kollár, I., Liu, M.–C.: Statistical theory of quantization. IEEE Trans. Instrum. Meas. **45**(2), 353–361 (1996)
7. Hejn, K., Morling, R.C.S.: A semifixed frequency method for evaluating the effective resolution of A/D converters. IEEE Trans. Instrum. Meas. **41**(2), 212–217 (1992)
8. Petri, D.: Frequency-domain testing of waveform digitizers. IEEE Trans. Instrum. Meas. **51**(3), 445–453 (2002)
9. Andersson, T., Händel, P.: IEEE standard 1057, Cramér-Rao bound and the parsimony principle. IEEE Trans. Instrum. Meas. **55**(1), 44–53 (2006)
10. Papoulis, A.: Probability, Random Variables, and Stochastic Processes. McGraw Hill, New York (1989)
11. Rife, D.C., Vincent, G.A.: Use of the discrete Fourier transform in the measurement of frequencies and levels of tones. Bell Syst. Tech. J. **49**, 197–228 (1970)
12. Offelli, C., Petri, D.: Interpolation techniques for real-time multifrequency waveforms analysis. IEEE Trans. Instrum. Meas. **39**(1), 106–111 (1990)
13. Offelli, C., Petri, D.: The influence of windowing on the accuracy of multifrequency signal parameter estimation. IEEE Trans. Instrum. Meas. **41**(2), 256–261 (1992)
14. Belega, D., Dallet, D.: Multifrequency signal analysis by interpolated DFT method with maximum sidelobe decay windows. Measurement **42**(3), 420–426 (2009)
15. Belega, D., Dallet, D., Petri, D.: Statistical description of the sine-wave frequency estimator provided by the interpolated DFT method. Measurement **45**(1), 109–117 (2012)
16. Belega, D.: "The maximum sidelobe decay windows", L'Académie Roumaine, Revue Roumaine des Sciences Techniques. Série Electrotechnique et Energétique **50**(3), 349–356 (2005)

17. Harris, F.J.: On the use of windows for harmonic analysis with the discrete Fourier transform. Proc. IEEE **66**(1), 51–83 (1978)
18. Belega, D., Dallet, D., Petri, D.: Accuracy of sine wave frequency estimation by multipoint interpolated DFT approach. IEEE Trans. Instrum. Meas. **59**(11), 2808–2815 (2010)
19. Offelli, C., Petri, D.: A frequency-domain procedure for accurate real-time signal parameter measurement. IEEE Trans. Instrum. Meas. **39**(2), 363–368 (1990)
20. Belega, D., Dallet, D., Petri, D.: Accuracy of the normalized frequency estimation of a discrete-time sine-wave by the energy-based method. IEEE Trans. Instrum. Meas. **61**(1), 111–121 (2012)
21. Nuttall, A.H.: Some windows with very good sidelobe behavior. IEEE Trans. Acoust. Speech Signal Process. **ASSP-29**(1), 84–91 (1981)
22. Händel, P.: Properties of the IEEE-STD-1057 four-parameter sine wave fit algorithm. IEEE Trans. Instrum. Meas. **49**(6), 1189–1193 (2000)
23. Belega, D., Dallet, D.: Dynamic testing of A/D converters by means of the three-parameter sine-fit algorithm. Measurement **40**(1), 1–7 (2007)
24. Bilau, T.Z., Megyeri, T., Sárhegyi, A., Márkus, J., Kollár, I.: Four-parameter fitting of sine-wave testing result: iteration and convergence. Comput. Stand. Interfaces **26**(1), 51–56 (2004)
25. Jain, V.K., Collins, W.L., Davis, D.C.: High-accuracy analog measurements via interpolated FFT. IEEE Trans. Instrum. Meas. **IM-28**(2), 113–122 (1979)
26. Kay, S.M.: Fundamentals of Statistical Signal Processing: Estimation Theory, vol. 1. Prentice-Hall, Upper Saddle River, New Jersey (1993)
27. Novotný, M., Slepička, D., Sedláček, M.: Uncertainty analysis of the RMS value and phase in frequency domain by noncoherent sampling. IEEE Trans. Instrum. Meas. **56**(3), 983–989 (2007)
28. Deyst, J.P., Souders, T.M., Solomon, O.M.: Bouds on least-squares four-parameter sine-fit errors due to the harmonic distortion and noise. IEEE Trans. Instrum. Meas. **44**(3), 637–642 (1995)
29. Pintelon, R., Schoukens, J.: An improved sine-wave fitting procedure for characterizing data acquisition channels. IEEE Trans. Instrum. Meas. **45**(2), 588–593 (1996)
30. Ramos, P.M., da Silva, M.F., Martins, R.C., Cruz Serra, A.M.: Simulation and experimental results of multiharmonic least-squares fitting algorithms applied to periodic signals. IEEE Trans. Instrum. Meas. **55**(2), 646–651 (2006)
31. Ramos, P.M., Cruz Serra, A.: Least squares multiharmonic fitting: convergence improvements. IEEE Trans. Instrum. Meas. **56**(4), 1412–1418 (2007)
32. Dallet, D., Petri, D., Belega, D.: ADCs dynamic testing by multiharmonic sine fitting algorithms. In: Proceedings of the International Workshop on ADC Modelling, Testing and Data Converter Analysis and Design and IEEE 2011 ADC Forum, Orvieto, Italy, June 30 to July 1, 2011
33. IEEE Standard for Digitizing Waveform Recorders, IEEE Std. 1057–2007, 2007
34. Shinagawa, M., Akazawa, Y., Wakimoto, T.: Jitter analysis of high-speed sampling systems. IEEE J. Solid-State Circuits **25**(1), 220–224 (1990)
35. Sounders, T.M., Flach, D.R., Hagwood, C., Yang, G.L.: The effects of jitter in sampling systems. IEEE Trans. Instrum. Meas. **39**(1), 80–85 (1990)
36. Verbeyst, F., Roland, Y., Schoukens, J., Pintelon, R.: System indentification approach applied to jitter estimation. In: Proceedings of the IEEE Conference Instrumentation and Measurement Technology Conference, pp. 1752–1757, Sorrento, Italy, Apr 24–27 2006
37. Jenq, Y.C.: Digital spectra of nonuniformly sampled signals: A robust sampling time offset estimation for ultra high-speed waveform digitizers using interleaving. IEEE Trans. Instrum. Meas. **39**(1), 71–75 (1990)
38. Shariat-Panahi, S., Alegria, F.A.C., Mànuel, A., Serra, A.M.C.: IEEE 1057 jitter test of waveform recorders. IEEE Trans. Instrum. Meas. **58**(7), 2234–2244 (2009)
39. Chiorboli, G., Fontanili, M., Morandi, C.: A new method for estimating the aperture uncertainty of A/D converters. IEEE Trans. Instrum. Meas. **47**(1), 61–64 (1998)

40. Janik, J.-M., Bloyet, D., Guyot, B.: Measurement of timing jitter contributions in a dynamic test setup for A/D converters. IEEE Trans. Instrum. Meas. **50**(3), 786–791 (2001)
41. Schoukens, J., Pintelon, R., Vandersteen, G.: A sinewave fitting proceure for characterizing data acquisition channels in the presence of time base distortion and time jitter. IEEE Trans. Instrum. Meas. **46**(4), 1005–1011 (1997)
42. Verspecht, J.: Accurate spectral estimation based on measurements with a distorted-time based digitizer. IEEE Trans. Instrum. Meas. **43**(2), 210–215 (1994)

Chapter 11
Histogram-Based Techniques for ADC Testing

Antonio Moschitta, David Macii, Francisco Corrêa Alegria and Paolo Carbone

11.1 Introduction

Over the past 20 years, researchers both in academia and in industry have carried out a large amount of research on the topic of Data Converter Testing. The motivations are both economical and technical. In fact, both the high percentage of testing costs over total production costs and the large increase in the market demand for both Analog-To-Digital (ADC) and digital-to-analog converters have justified investigations and development of new and more efficient testing methods. On the technical side, the problem of data converter testing is challenging for several reasons, such as the nonlinear behavior of the quantizer inside an ADC and the ever-increasing sampling rate. Test engineers have also been challenged both in modeling the test process and in actual implementation of testing procedures. Both sides of this issue are important: modeling brings about new insights in the solution of practical testing problems and, conversely, experimental results help validate and improve models accuracy and usability.

The aim of this chapter is that of presenting in an ordered way, the state-of-the-art in the properties of one of the most important ADC testing techniques, that is the

A. Moschitta (✉) · P. Carbone
Department of Electronic and Information Engineering, University of Perugia,
Via G. Duranti 93, 06100 Perugia, Italy
e-mail: antonio.moschitta@unipg.it

P. Carbone
e-mail: paolo.carbone@unipg.it

D. Macii
Department of Industrial Engineering, University of Trento,
Via Sommarive 14, 38100 Trento, Italy
e-mail: david.macii@unitn.it

F. C. Alegria
Instituto Superior Técnico/Universidade de Lisboa and Instituto de
Telecomunicações, Av. Rovisco Pais 1, 1049-001 Lisbon, Portugal
e-mail: falegria@lx.it.pt

P. Carbone et al. (eds.), *Design, Modeling and Testing of Data Converters*,
Signals and Communication Technology, DOI: 10.1007/978-3-642-39655-7_11,

histogram or code density test. The intended audience is supposed to be made of both electrical engineers interested in the subject and experts seeking a reference and a reasoned comparison between the features and applicability of various histogram-based techniques published in the scientific and technical literature. Starting from the basic definitions of most important terms, the reader is exposed to the details regarding the test design, including configuration and expressions of estimators with the related properties. All relevant literature is considered, including several international technical standards on this subject, owing to the long-lasting experience of some of the authors of this chapter in the group of experts drafting the standards contents.

11.2 A General Overview of Histogram-Based Methods

As known, all ADC test methods apply a supposedly known signal to the input and acquire a given number of samples (i.e., the signal in the digital domain). Any Analog-to-Digital Converter testing method compares the captured samples from the output to a theoretical model of the signal used to stimulate the input. Some methods, such as sine-wave fitting or Fourier analysis, compare directly the signal itself; others, like the histogram method, compare the distribution characteristics. While the Static Test Method uses the simplest input signal imaginable (a DC voltage), the Histogram Test Method uses an alternate signal, which is often a sinewave. In particular, a mathematical model of input is compared in the digital domain to the output signal samples, through the comparison of *code density functions*. A code density function gives the percentage of occurrence for which an analog signal is in the range of a given ADC output code, that is, how long its value lies between the two transition voltages ($T[k]$ and $T[k + 1]$) corresponding to code k. This function is a representation of the analog signal in the digital domain defined by the ADC. The code density function is thus compared with the actual relative number of samples obtained with each possible output code—the Histogram [1, 2]. If the analog input signal (i.e. the stimulus) is a stationary and ergodic process, the histogram built using the ADC output samples asymptotically estimates the probability $P(T[k], T[k + 1])$ that the input signal takes values in the interval with endpoints $T[k]$ and $T[k + 1]$. Thus, if the stimulus probability model is known a priori, it can be inverted to estimate the transition levels. Therefore, the experimentally collected histogram could be used to estimate each probability $P(T[k], T[k + 1])$. In practice, histogram tests are carried out more easily by evaluating experimentally the cumulative histogram, that is the fraction of time that the stimulus is below a given transition level $T[k]$. This estimation technique would be perfect if were it not for the finite number of samples that can be acquired, the imperfections of the actual input signal, and other non-idealities associated to the behaviour of an ADC, such as jitter and voltage noise.

The estimation of the actual ADC transfer function allows the estimation of several interrelated parameters, namely, transition levels, code bin widths, gain error, offset, integral nonlinearity (*INL*) and differential nonlinearity (*DNL*) [45]. The knowledge of these parameters is important not only to compare the quality of different ADCs, but also to determine the values of an unknown input signal. In addition, various ADC defects, like the existence of missing codes could be discovered as well.

The histogram-based testing methods are recommended (although with some differences) in the main standard documents dealing with data acquisition devices, most notably the standards IEEE 1241 and IEC 60748-4-3 (which are specifically focused on ADCs) [3, 4], and the dual norms IEEE 1057 and IEC 62008 concerning with digitizers and data acquisition systems [5, 6].

In principle, any kind of deterministic or stochastic signal with a known distribution function can be used for histogram testing. The mentioned standard documents suggest using just simple deterministic signals, such as sinewaves (IEEE 1241, IEC 60748-4-3 and IEEE 1057), linear ramps (IEEE 1241, IEC 60748-4-3 and IEEE 1057), full-scale triangle waveforms (IEEE 1057) or high-precision increasing DC values with or without ramp vernier (IEEE 1057 and IEC 62008). The main reason underlying the use of deterministic waveforms is that the parameters of such stimuli can be defined and controlled much better than those of more complex signals. Furthermore, the theoretical output code density functions associated with such waveforms are generally quite simple and easier to compare with the corresponding experimental histogram-based patterns. For instance, when a triangular or sawtooth waveform with proper amplitude is used as input signal, the code density function is constant. However, the histogram test accuracy in this case depends on ramp linearity, which in turn is influenced by the nonlinearities affecting the Digital-to-Analog (DAC) converter inside the adopted arbitrary waveform generator commonly used nowadays to produce test signals. Moreover, the high-frequency components of the test sawtooth or triangle wave can be unwillingly filtered by the limited bandwidth of the experimental setup. For all the reasons above, the full-range ramp-based methods are recommended just to determine either the static or low-frequency characteristics of an ADC.

An alternative approach based on a highly accurate DC source combined with a small triangle waveform is presented in [7] and it is also included in the standards IEEE 1057 and IEC 62008. In this case, instead of estimating all the transition levels at the same time using a full-range sawtooth or a triangle waveform, a small triangular waveform superimposed to a progressively increasing DC signal is used to excite small groups of adjacent transition levels in quasi static conditions. The main advantage of this approach is that the linearity and accuracy requirements of the ramps can be relaxed. Also, the number of DC source output changes is much lower than in the classic step-by-step static test. Similarly, the amount of samples that must be collected to reach a given target uncertainty is much smaller than using a full-range waveform. As a result, the test duration is typically shorter than

in the other cases, especially when high-resolution ADCs are considered. Unfortunately, this method can never be used in dynamic conditions. On the contrary, the sinewave histogram test (SHT) is preferable in dynamic conditions, although only a specific operating frequency can be explored at a time. In practice, SHT is the most common histogram test because the sinewaves can easily be generated, characterized and adjusted in order to optimize the test performance. In addition, all test parameters (e.g., amplitude and frequency of the input waveform, as well as number of data records) can be set properly to minimize the effect of noise and other uncertainty contributions affecting measurement results.

Deterministic test waveforms are apparently better also to implement *built-in self-test* (BIST) techniques [8–11]. Generally speaking, any BIST strategy relies on the integration of some additional testing-oriented circuitry on the same chip as the device to be tested (e.g., low-costs standard waveform generators in this case), thus reducing the complexity of the automated testing equipment and the related costs. In mixed-signal integrated circuits, such costs are particularly high. Typically, from 15 % to 20 % of the ADC unit selling price is indeed due to testing procedures [12]. Unfortunately, producing high-accuracy signals having adequate spectral purity, linearity and frequency stability by means of low-cost on-chip circuitry is a very challenging task [13], especially when the resolution of the converter grows [14]. In order to relax the tight low distortion requirements of the input test waveforms, some researchers proposed using either nonstandard deterministic signals (e.g. exponential ramps) or stochastic signals, particularly Gaussian noise [15], step Gaussian noise [16], or truncated Gaussian noise [17]. Suitable exponential test signals with a given shape and a given accuracy can easily be generated by discharging a high-precision capacitor across a known resistance [18]. Therefore, the main advantage of this approach is the easiness in generating the wanted test waveform.

Stochastic test signals instead are advantageous because a noise generator is completely described by the distribution of the output signal. In addition, it requires simpler circuitry than high-accuracy ramp and sinewave generators and it makes the ADC testing intrinsically robust to the influence of noise. If the noise is normally distributed, any additional independent Gaussian noise source introduced by the experimental setup just increases the total variance of the test signal, with no significant effects on the resulting output code histogram, except for a possible gain error [19]. Moreover, since noise is intrinsically a wideband signal, it is more suitable than narrowband deterministic periodic waveforms to evaluate the performance of an ADC over a given frequency band. In fact, the transition levels change as a function of frequency (e.g. due to various parasitic capacitances). Accordingly, also the *INL* and *DNL* patterns may change significantly as a function of the frequency content in the signal to be converted. For all the reasons above, the noise generators are particularly interesting for BIST strategies [20], especially when high-resolution ADCs or ultra high-speed ADCs are considered. The main limitation of the Gaussian histogram test (GHT) is that for a given amount of

Table 11.1 Qualitative comparison of different histogram tests [22]

	Easiness of signal generation	Speed of the histogram test	Histogram test accuracy
Sinewave	Medium	Medium	High
Full scale triangular wave/ramp	Low	Medium	High (low frequency only)
Ramp Vernier	Medium	Low	High (quasi-static only)
Exponential curve	High	High	Medium
Gaussian noise	High	High	Medium
Step-Gaussian noise	High	Low	Medium
Truncated Gaussian noise	High	High	Medium/low

collected samples, the transition level estimation accuracy is about 10 times smaller than the accuracy achievable with a standard sinewave histogram test at different frequencies over a given band [21]. Moreover, no perfect Gaussian distributions can be built in practice. As a consequence, the actual noise variance is never exactly the same as the assumed one. Moreover, the amplitude of the input signal may occasionally exceed the full-scale range of the ADC, thus unnecessarily prolonging the time required to collect a sufficiently large number of output code samples. A simple, but effective idea to tackle these problems is to make the peak-to-peak noise amplitude just large enough as the full-scale ADC range. This can be done in various ways. In [16] the authors suggest that a noise-based test signal with a quasi-uniform distribution over the ADC input range can be generated by adding multiple, slightly shifted low-variance Gaussian noise sources exhibiting the same stochastic behaviour. Alternatively, a truncated Gaussian noise generator is presented in [17]. This generator is able to excite all and only the ADC transition levels (including the lowest and the highest ones) with a reasonably high probability. Consequently, the histogram test duration is shorter than the GHT, because a reasonably large number of samples in each code bin can be reached in a shorter time. In fact, the Cramer-Rao lower bound (CRLB) to the variance of the transition level estimators is higher in the case when the noise is truncated [17].

In Table 11.1 the various types of considered histogram tests are compared qualitatively, in terms of easiness of signal generation, testing speed and accuracy [22].

Note that the nonstandard input stimuli are easier to generate and lead to faster results at the expense of some loss in accuracy. However, the SHT provides the best trade-off between testing speed and accuracy in dynamic conditions. For this reason, as well as for the widespread use of sinewaves in practical scenarios, the rest of this chapter will be mostly focused on SHT description and optimization. In Sect. 11.3 the theoretical fundamentals of the SHT are explained. In Sect. 11.4 the main criteria used for choosing the test parameters are presented. Section 11.5 focuses on some processing techniques for reducing test duration. Finally, some examples of test results are reported in Sect. 11.6.

11.3 Histogram Test Fundamentals

11.3.1 General Definitions

The Histogram Test Method is used to estimate the transfer function of an ADC. This transfer function relates each value of an input signal (usually a voltage) to the output code obtained as a result of the analog-to-digital conversion process. As mentioned in Sect. 11.2, its knowledge is essential to determine the unknown interval of input voltages leading to a given output code.

The transfer function can be described using various inter-related parameters. At first, the transition voltages are usually estimated from the codes of the test samples. There are 2^N-1 transition voltages for a N-bit ADC (with 2^N output codes) which are typically represented by $T[k]$ (with $k = 1,\ldots,2^N-1$). In practice, due to noise, there is not just one single input voltage value for which the output code is always $k-1$ and an infinitesimally higher voltage value for which the output code is always k. Therefore, $T[k]$ represents the signal value at which the ADC generates an output code equal to or smaller than $k-1$ 50 % of the time and an output code larger than or equal to k the rest of the time [23].

Although the transition voltages are useful for determining the input signal from the output code, they are not useful to compare different ADCs. In this case a parameter called *integral nonlinearity* (*INL*) is generally used. The underlying rationale is that a better ADC has a transfer function closer to the ideal one. A good performance figure is then given by the difference between actual and nominal transition thresholds $T_{nom}[k]$. There are, however, two additional considerations to be made. First of all, possible differences in gain or offsets do not affect linearity. Thus, the actual transition voltages can be "corrected" to eliminate their effect in the computation of $INL[k]$. In particular, the gain and the offset, are computed so as to make the actual value of the first and last transition voltages (after "correction") equal to the nominal ones. As a consequence, $INL[1]$ and $INL[2^N-1]$ are 0 by definition. Actually, this is just one definition of *INL* (called *terminal based*). According to another quite common definition the values of gain and offset are computed in such a way that the mean squared difference between "corrected" and nominal transition voltages is made as small as possible.

The second consideration is that, in order to compare ADCs having different input ranges and number of bits, the difference in transition voltages should be preferably expressed as a function of the ideal code bin width Q. In fact, Q is the (constant) difference between any pair of consecutive ideal transition voltages. Note that there are some differences among manufacturers and among researchers about how the transfer function, the ADC input range, the value of the transition voltages and the value of the ideal code bin width are defined [24].

In general, the integral non-linearity is defined as [3]

$$INL[k] = \frac{G \cdot T[k] + O - T_{nom}[k]}{Q} \qquad k = 1, \ldots, 2^N - 1 \tag{11.1}$$

and expressed in units of LSB (least significant bit). An ideal ADC has, by definition, a unity gain, a null offset and all values of $INL[k]$ equal to 0.

Note that the INL depends on the specific transition level considered. When a single parameter is desirable to express ADC integral linearity, the maximum absolute value of the INL pattern is typically used, i.e. $\overline{INL} = \max_{k=1,\ldots,2^N-1} |INL[k]|$.

Similarly, the differential non-linearity (DNL) is defined as

$$DNL[k] = \frac{G \cdot W[k] - Q}{Q} \qquad k = 1, \ldots, 2^N - 2 \tag{11.2}$$

where $W[k] = T[k+1] - T[k]$ is the actual width of the kth code bin. Note that $W[k] = Q$ in an ideal ADC. This equivalently means that $DNL[k] = 0$ in an ideal ADC. In the particular case, which is sometimes encountered in practice, of an ADC that never outputs a given code (a missing code) the corresponding code width is 0 and the DNL of that code is equal -1. In practice, however, a missing code is conventionally reported when $\mathrm{DNL}[k] \leq -0.9$ [3].

Again, if a single parameter has to be used to express ADC differential linearity, the maximum absolute value of the DNL pattern is commonly adopted, i.e. $\overline{DNL} = \max_{k=1,\ldots,2^N-1} |DNL[k]|$. Alternatively, the root mean square (rms) value is employed.

Assume that S samples of a known periodic waveform are collected by the ADC under test. At the ADC output, the codes produced by the converter and associated with the same level k, for $k = 0,\ldots,2^N-1$, can be counted and used to build a histogram. In the following, we will refer to $H[k]$ as the total number of samples associated with code bin k and to $H_c[k] = \sum_{j=0}^{k} H[j]$ as the cumulative number of samples ranging from 0 to k. Depending on the properties of the input signal, different relationships between $H_c[k-1]$ and $T[k]$ exist. Accordingly, the values of $T[k]$ as well as the INL and DNL patterns can be estimated from the bins of the cumulative histogram $H_c[k]$, as it will be shown in the next section for the case of a sinusoidal test signal.

Histogram methods may return inaccurate results if the device under test exhibits a significant non-monotonic behavior. In such cases, the test result may apparently look even when some output codes are swapped. For this reason, the Signal-to-Noise and Distortion Ratio (SINAD) of the converter should be measured as well to confirm that the ADC characteristic exhibits a monotonic behaviour [3].

11.3.2 The Sinewave Histogram Test

In a SHT the input waveform is a high-purity sinewave defined as

$$v(t) = A \cdot \cos(\omega t + \varphi) + C \tag{11.3}$$

where ω, φ, A and C represent the angular frequency, the initial phase, the waveform amplitude and the offset (i.e. DC level), respectively. As explained in Sect. 11.2, the SHT is the most widely used histogram-based test and it is recommended in all standard documents dealing with analog-to-digital converter and digitizer testing [3, 4]. If the instantaneous phase $\theta(t) = \omega t + \varphi$ of the sinewave is assumed to be uniformly distributed in the interval $[-\pi, \pi)$, then the probability density function associated with the amplitude of the input waveform to be converted is [25]

$$f_v(v) = \begin{cases} \frac{1}{\pi\sqrt{A^2-(v-C)^2}}, & |v-C| < A \\ 0, & elsewhere \end{cases}. \tag{11.4}$$

Therefore, the probability that a given input signal value lies between V_a and V_b, with $|V_a - C| \leq A$ and $|V_b - C| \leq A$, is

$$P(V_a, V_b) = \frac{1}{\pi}\left[\cos^{-1}\left(\frac{V_b - C}{A}\right) - \cos^{-1}\left(\frac{V_a - C}{A}\right)\right]. \tag{11.5}$$

This signal in the analog domain may be converted to the digital domain by considering an ADC with transition voltages $T[k]$. Considering that the stimulus signal amplitude is large enough to excite the whole ADC input range (i.e., $A \geq FS$, where FS is the ADC full-scale range), one can set $V_b = T[k]$ and $V_a = (-A + C)$. In this case, (11.5) can simply be rewritten as

$$P_k = P(-A, T[k]) = \frac{1}{\pi}\cos^{-1}\left(\frac{T[k] - C}{A}\right) - 1. \tag{11.6}$$

Therefore, the transition voltage can be retrieved from (11.6), and results from

$$T[k] = C + A \cdot \cos(\pi \cdot P_k + \pi) = C - A \cdot \cos(\pi \cdot P_k). \tag{11.7}$$

A common estimator of P_k is the ratio between the number of elements of the bin $k-1$ of the output code cumulative histogram and the overall number of collected samples S, i.e. $\hat{P}_k = H_c[k-1]/S$. Accordingly, the kth transition level can be estimated from

$$\hat{T}[k] = C - A \cdot \cos\left(\pi \cdot \frac{H_c[k-1]}{S}\right) \quad \text{for } k = 1, 2, \ldots, \left(2^N - 1\right) \tag{11.8}$$

and the code bin widths from

$$\hat{W}[k] = \hat{T}[k+1] - \hat{T}[k] \qquad \text{for } k = 1, 2, \ldots, \left(2^N - 2\right). \tag{11.9}$$

Accordingly, the *INL* and *DNL* patterns can be estimated by replacing (11.8) and (11.9) into (11.1) and (11.2), respectively.

Note that, if the values of A and C are known, the SHT assures that the transition levels are estimated with a comparable accuracy. If instead the amplitude and the offset of the sinewave are unknown, A and C can be obtained from (11.8) and from two independent estimates of any two of the transition levels [23].

11.4 Configuration and Execution of the Sinewave Histogram Test

11.4.1 Test Description and Achievable Performance

The SHT allows to trade estimator accuracy for test duration, and can tolerate additive noise as long as a proper amount of overdrive is used [23]. The sinusoidal stimulus has also been recently proposed for ADC static non-linearity estimation as an alternative to ramp signals, since this kind of signals can be considered approximately linear when its argument is close to zero [26]. Consistently with the statements of Sect. 11.3, and assuming that the stimulus (11.1) is coherently sampled and affected by Additive White Gaussian Noise (AWGN) $w(\cdot)$ with standard deviation σ_w, the test is performed by coherently sampling and acquiring a record of M samples from stimulus (11.1), thus obtaining the sequence

$$\begin{gathered} v[n] = A \cdot \cos(\omega n T_s + \varphi) + C + w[n], \;\; n = 0, \ldots, M-1 \\ T_s = 1/f_s, \quad \omega = 2\pi f_i, \quad \frac{f_i}{f_s} = \frac{J}{M}, \end{gathered} \tag{11.10}$$

where $T_s = 1/f_s$ is the sampling period, f_i is the stimulus frequency, and J is an integer number, such that J and the record length M are co-prime integers. Then, after evaluating the cumulative histograms, the transition levels can be estimated using (11.8). Notice that the SHT approach provides meaningful results only if no hysteresis is present. Techniques to detect hysteresis are described in various standards [3–5]. For instance, the SHT may be carried out twice, using the two subsets of samples acquired respectively in the rising edges and in the falling edges of the stimulus [27]. Hysteresis, if present, would lead to differences in the estimated transition levels. As known, the Cramér-Rao Lower Bound (CRLB) provides the minimum achievable estimator variance for a parameter of interest. The SHT is indeed asymptotically efficient and almost unbiased in practical testing conditions. In fact, the CRLB of the T_k estimator variance has been modeled in [28] for 1-bit ADCs and in [29] for the general case of an N-bit memory-less converter. Let us define $\overline{T}_k = T_k - C$. By assuming that $A > \left|\overline{T}_k\right|$ and that an

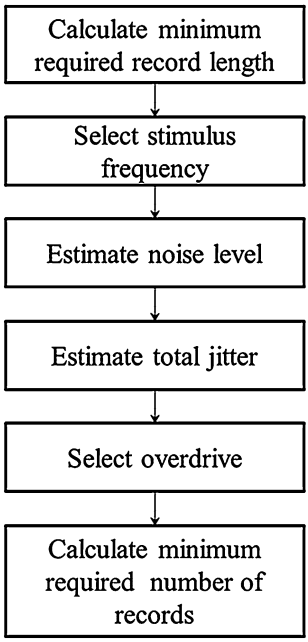

Fig. 11.1 Flowchart of SHT configuration procedure

overall amount of M samples is collected in a single- or in multi-record acquisitions, the estimator variance for the transition voltages is approximately given by [30]

$$\sigma_k^2 \cong \alpha_k(1-\alpha_k)\frac{\pi^2}{M^2}\left(A^2-\overline{T}_k^2\right)+\frac{1.78}{M}\sigma_w\sqrt{A^2-\overline{T}_k^2} \tag{11.11}$$

where $\Delta\phi_M = \frac{2\pi}{M}$, $\psi_k = \arccos\left(-\frac{\overline{T}_k}{A}\right)$, $\alpha_k = \left\langle\frac{2\psi_k}{\Delta\phi_M}\right\rangle$, and $\langle\cdot\rangle$ is the fractional part operator. Note that, for large values of M, the first term in (11.11), modeling the effect of phase variability between different records can be neglected. As a result, (11.11) can be approximately rewritten as

$$\sigma_k^2 \cong \frac{1.78}{M}\sigma_w\sqrt{A^2-\overline{T}_k^2}. \tag{11.12}$$

Moreover, under the same conditions, the estimator bias is expressed as [30]

$$m_k = E\left[\widehat{\overline{T}}_k - T_k\right] = \frac{\sigma_w^2}{2}\frac{\overline{T}_k}{A^2-\overline{T}_k^2}, \tag{11.13}$$

where the "hat" symbol denotes an estimated quantity.

In general, the design of an SHT is based on the flowchart shown in Fig. 11.1. The individual steps are described in the following.

11.4.2 SHT Configuration Criteria and Procedures

The test configuration should guarantee that the stimulus excites each ADC bin with enough samples to estimate the corresponding transition levels with a suitable accuracy.

In addition, the probability density function (*pdf*) of the collected samples should not significantly differ from the theoretical arcsine one (11.4), postulated in [3]. In fact, a stimulus with a *pdf* different from (11.4) may result in biased transition level estimators. Notice that bin excitation is guaranteed by selecting both a suitably large record length M and a suitable ratio between sampling frequency and sinewave frequency. On the other hand, the *pdf* of arcsine distribution is ensured by overdriving the ADC, that is by applying a sinewave with a larger amplitude than the ADC Full Scale Range. In this way, the noise contribution to the stimulus *pdf* becomes negligible [23].

The degrees of freedom in configuring the SHT are sinewave amplitude, offset, and frequency, coupled with the record length M. Furthermore, input noise power, sinewave phase noise, and sampling jitter should be preliminarily estimated, because such parameters are required to properly select the stimulus amplitude and the test duration.

11.4.2.1 Noise Estimation

Consistently with IEEE Standard 1241, noise can be defined as "any deviation between the output signal (converted to input units) and the input signal except deviations caused by linear time invariant system response (gain and phase shift), a dc level shift, or total harmonic distortion".

The noise level estimation is needed, since it is necessary both to design the required overdrive level (see Sect. 11.4.2.3) and to assess the INL and DNL estimation uncertainty (see Sect. 11.4.2.4). Various estimation procedures are available. In the following, both the main standardized approaches and some approaches proposed in the literature are mentioned. Notice that the effectiveness of the described procedures may be assessed by comparing each estimator variance against the Cramér-Rao Lower Bound (CRLB) on the noise power [31]. In particular, for a bipolar, memory-less ADCs with quantization levels uniformly distributed in $[-FS, FS]$ and affected by AWGN, the CRLB when estimating the noise power is given by [31]

$$CRLB_{\sigma_w^2}\left(\frac{\sigma_w}{Q}\right) \cong \frac{2FS^4}{9M2^{4N}}\left(12\left(\frac{\sigma_w}{Q}\right)^2+1\right)^2, \tag{11.14}$$

where Q is the ADC converter quantizer step, N is the ADC resolution, and M is the record length.

IEEE standard 1241 presents three different methods for noise estimation. The first two approaches are differential, and entail feeding the ADC twice with the same stimulus (a constant signal in the first approach and the rising front of a triangular signal in the second, which is preferred for low noise ADCs). In particular, in the case of a constant stimulus, the noise is estimated with

$$\delta = \hat{\sigma}_w^2 = \frac{1}{2M}\sum_{n=0}^{M-1}(x_1[n]-x_2[n])^2, \tag{11.15}$$

where $\delta = \hat{\sigma}_w^2$ is the estimator of the noise power σ_w^2, while $x_1[\cdot]$ and $x_2[\cdot]$ are the two collected ADC output records of length M. Such an approach is suitable when noise variations are at least comparable with the quantization step. For lower noise levels, the estimator (11.15) is biased, depending on the position of the selected DC level within a code bin [3].

In the case of low-level noise, the standard IEEE 1241 suggests an alternative estimation methodology. In particular, the constant input signal is replaced by a period of a triangular wave, spanning at least 10 code bins and triggered by the beginning of the rising ramp. Accordingly, the noise is estimated as

$$\hat{\sigma}_w^2 = \frac{1}{\sqrt{\left(\frac{2}{\delta}\right)^2+\left(\frac{Q}{0.886\delta}\right)^4}} \tag{11.16}$$

where δ is given by (11.15). Notice that the standard deviation of such noise level estimator is given by [3]:

$$\sigma_\delta = \frac{\sigma_w}{\sqrt{M-1}}. \tag{11.17}$$

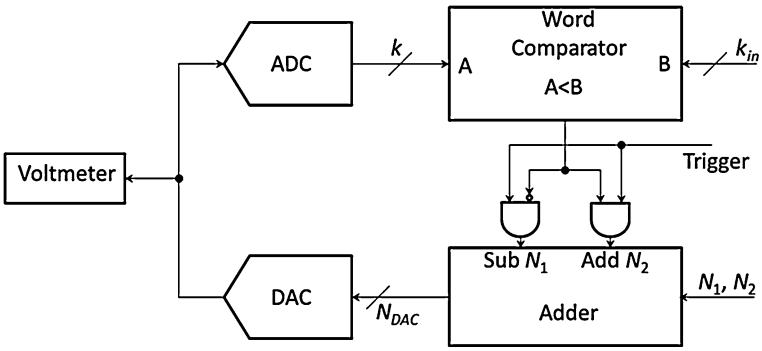

Fig. 11.2 Servo-loop based setup for noise estimation

This means that the uncertainty associated with noise power estimation decreases when the number of collected samples becomes larger.

The third approach proposed in [3] relies on a feedback loop strategy, and it is based on the very same servo-loop architecture proposed for locating ADC transition levels. This approach, sketched in Fig. 11.2, requires feeding the ADC with a DAC or an accurate analog source. In fact, this is driven so as to track a desired ADC output word, by subsequently changing the DAC/analog source by a given step. By using two different steps for the servo-loop, the ADC input settles to two different mean values that can be measured using a voltmeter. The noise standard deviation may then be inferred from the difference between the two voltage values. This procedure may provide accurate results, but it requires an accurate voltmeter and a longer test duration. Therefore, it is not advisable for slow converters.

In addition to the procedures reported in IEEE Std. 1241 and 1057, it is worth noticing that other approaches have been developed. For instance, the IEC 60748-4-3 proposes three approaches. Two of them are similar to those reported in the IEEE standards, whereas the third one is a frequency-based method, relying on the DFT computed on the samples collected when the ADC is stimulated with a sinewave. In this case, the DFT bins corresponding to the sinewave frequency and to its harmonics are discarded while all the others are used to estimate the noise power. Such a methodology is very similar to those described in clauses 8.2.1, 8.2.2 and 8.2.3 of IEEE Std. 1057, although they are aimed at SNR estimation, rather than at noise power estimation.

Another method based on Maximum Likelihood Estimation is described in [31]. Such an approach relies on the difference sequence

$$d[n] = x_1[n] - x_2[n], \; n = 0, \ldots, M-1, \tag{11.18}$$

and requires evaluating the likelihood $P(d[\cdot], \sigma_w)$, namely the probability of observing the collected sequence $d[\cdot]$ as a function of the noise standard deviation σ_w. By maximizing $P(d[\cdot], \sigma_w)$ with respect to σ_w, an estimator of σ_w is obtained. This estimator is characterized by the same statistical efficiency of the IEEE 1241 algorithms, as it trades unbiasedness for computational complexity.

Finally, a method considering a sinewave as a slow drift in the mean value of a wideband Gaussian noise has been recently proposed in [32]. This approach is based on the analysis of the output code oscillations. In particular, at first the mean value and the standard deviation of the ADC output codes are estimated around a suitable transition level. The estimation results are then weighted by the sine-wave slope corresponding to the transition level itself, thus obtaining an estimator of the noise standard deviation [32]. A similar analysis has been carried out in [33] and in [34], where a ramp signal is considered as stimulus and a Maximum Likelihood estimator is used.

11.4.2.2 Jitter and Phase Noise Estimation

Timing fluctuations may lead to non-uniform sampling intervals. As a result, the sample values can be affected by the product between timing error and signal slope at the sampling instant, thus influencing SNR, SINAD, INL, and DNL [3, 5, 27]. Similar effects are associated to phase noise. However, phase noise is a stimulus feature rather than a performance limit of the device under test (DUT). Both stimulus phase noise and ADC sampling time fluctuations are important, since their joint contribution, together with any other significant timing uncertainty source (such as, for example, the jitter introduced by active circuitry and buffers), contribute to the total jitter. Such jitter should be properly measured in order to configure some of the SHT parameters [4, 27]. In particular, the standard deviation σ_{φ_i} of the total jitter expressed in radians, is given by [27]

$$\sigma_{\varphi_i} = \sqrt{\sigma_{sig}^2 + \sigma_{clk}^2 \frac{f_i^2}{f_s^2} + \sigma_T^2 (2\pi f_i)^2}, \tag{11.19}$$

where f_i and f_s are the stimulus and sampling frequencies respectively, σ_{sig} is the standard deviation of the input signal phase noise σ_{clk} is the standard deviation of the ADC clock and σ_T is the ADC aperture jitter, expressed in seconds. Since in [3–5] the total jitter is referred to the sampling period and expressed in radians, (11.19) may be rewritten as

$$\sigma_{\varphi} = \sqrt{\sigma_{sig}^2 \frac{f_s^2}{f_i^2} + \sigma_{clk}^2 + \sigma_T^2 (2\pi f_s)^2}. \tag{11.20}$$

Various procedures exist to estimate the mentioned contributions, listed mainly in [3, 5, 27]. For instance, if the ADC clock can be directly accessed, its jitter σ_{clk} may be directly estimated using a counter. Similarly, the stimulus phase noise may be estimated using a wideband oscilloscope, since modern instrumentation usually provides automated tools for such a measurement.

Phase noise estimation procedures in the frequency domain also exist. For instance, a procedure is provided in [5, Sect. B.3], where the phase noise, expressed as time jitter, is derived as a function of the phase noise spectrum. Regarding the aperture uncertainty, also referred to as timing jitter in [4, 5], the most recent approach is described in the IEEE Standard 1241 [3]. This approach is based on an external and stable clock that feeds both the ADC analog input and the ADC clock input. An adjustable delay is used to control the synchronization between the two signal replicas, as shown in Fig. 11.3a.

Notice that, in general, the ADC Analog Input (AI) path and the ADC Clock Input (CI) path experience different delays. Such a difference may be compensated using the variable delay term mentioned above. In this way, the ADC can actually sample its clock in the midpoint of its rising edge. For instance, in Fig. 11.3b the clock, with unitary period and a duty cycle of 50 %, experiences delays $\Delta_{AI} = 0.37$ s and $\Delta_{CI} = 0.15$ s when propagating along the AI and CI paths,

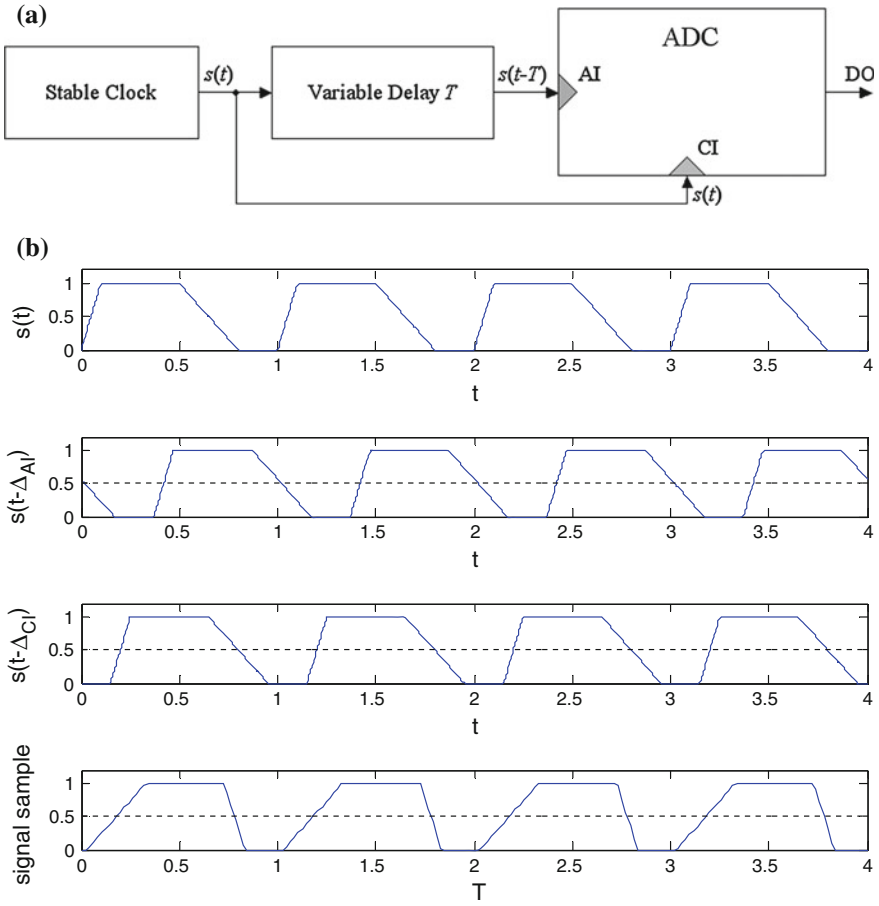

Fig. 11.3 **a** Test setup for assessing the ADC aperture jitter. AI is the ADC analog input, CI is the ADC clock input, DO is the ADC digital output. **b** From *top* to *bottom*, clock signal, clock delayed by the AI path, clock delayed by the CI path, and collected samples as a function of the variable delay T

respectively. In particular, the topmost sub-plot in Fig. 11.3b shows the original clock signal, while the two underlying sub-plots show the two delayed replicas. The bottom-most sub-plot, obtained by assuming that the ADC collects a sample when also the rising edge of the CI signal crosses its midpoint, shows the actually collected sample as a function of the variable delay. Under such conditions, the overall output noise power σ_A^2 may be estimated according to methods like those described in Sect. 11.4.2.1. By disconnecting the input signal and by measuring again the output noise power σ_B^2, the output noise contribution introduced by aperture uncertainty may be estimated as $\sigma_A^2 - \sigma_B^2$. Then, the aperture uncertainty σ_T may be obtained with

$$\sigma_T = \frac{\sqrt{\sigma_A^2 - \sigma_B^2}}{S_{eff}} \tag{11.21}$$

where the effective slope S_{eff} relates amplitude variations to temporal variations. The effective slope is obtained by setting the variable delay equal to two distinct values T_1 and T_2. If m_1 and m_2 are the respective average ADC outputs the effective slope finally results from

$$S_{eff} = \frac{|m_2 - m_1|}{|T_2 - T_1|}. \tag{11.22}$$

Further methods for jitter estimation are listed in [5], specifically in Sects. 12.1.1, 12.1.2 and 12.3.3, with the last one being similar to that discussed in [3].

11.4.2.3 Amplitude and Offset Selection

The selected stimulus amplitude A and offset C should guarantee two conditions. As a first requirement the offset C should be set equal to the midpoint of the ADC Full-Scale range. Furthermore, the sinewave amplitude should be set large enough to overdrive the ADC, in order to reduce the effect of noise on the signal *pdf*, which is maximum when the input sinewave reaches its peaks [23]. This can be seen in Fig. 11.4a, b. In particular, Fig. 11.4a shows the *pdf* of a zero-mean noiseless sinewave with peak amplitude $A = 1$ and random initial phase (black line), and the *pdf* of a noisy sinewave with $A = 1$, affected by AWGN with SNR = 20 dB (blue line). Two dotted vertical lines highlight the Full-Scale range $[-FS, FS]$ of a bipolar uniform ADC, with $FS = 1$. It can be observed that noise significantly alters the signal distribution, especially when the sinewave reaches its peaks. This in turn tends to bias the estimates of the ADC outermost transition

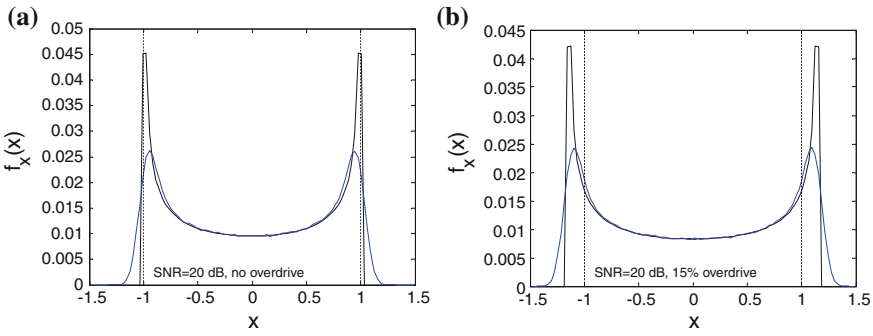

Fig. 11.4 **a** Probability density function of a noisy sinewave, in absence of overdrive. *Vertical lines* mark the normalized FSR of the ADC to be tested. **b** Probability density function of a noisy sinewave, in presence of a 15 % overdrive. *Vertical lines* mark the normalized FSR of the ADC to be tested

levels, when the SHT is carried out. Conversely, Fig. 11.4b shows the same results, obtained by overdriving the sinewave amplitude by 15 % with respect to the ADC Full-Scale. As expected, noise effects are negligible in the ADC dynamic range, thus improving the accuracy of the following histogram test.

It should be noticed that overdriving the ADC is not effective in reducing the influence of phase noise. In fact, it is shown in [35] that phase noise increases the transition level estimator variance, but it does not introduce any estimator bias.

A procedure for amplitude selection is reported in the following. According to [3], the minimum amplitude required to excite all the ADC levels may be identified iteratively, by choosing an initial amplitude and by progressively increasing this value, until the collected record systematically includes samples from the ADC outermost bins. Notice that, if the ADC input range is non-symmetrical around 0, a DC offset may be applied so as to center the stimulus within the ADC input range [5]. Various standards and standard drafts address the problem of determining the required overdrive level V_O, as a function of the parameter to be estimated (INL or DNL), noise level, and target accuracy.

In particular, in the latest editions of IEC 60748-4-3, IEEE 1057 and IEEE 1241 the amount of overdrive is defined as follows, i.e.

$$V_O \geq \max\left\{3\sigma_w,\ \sigma_w\sqrt{1.43\frac{3}{8B}}\right\} \tag{11.23}$$

for DNL measurements, and

$$V_O \geq \max\left\{2\sigma_w,\ \frac{\sigma_w^2 2^N}{FS \cdot B}\right\} \tag{11.24}$$

for INL measurements. In (11.23) and (11.24) σ_w is the rms of the input-referred noise, B is the target expanded uncertainty expressed as a fraction of the nominal code bin width and FS is the input full-scale range of the ADC [46]. When choosing the sinewave amplitude and offset values, the uncertainty due to the input signal generator should be also taken into account. In particular, the input sinewave amplitude should be at least larger than the sum of the worst-case uncertainty values affecting amplitude (e_A) and offset (e_C) of the generated stimulus, in accordance with the instrument specifications.

One final consideration about the measurement uncertainty affecting the SHT, is related to the presence of possible gain and offset errors in the ADC under test. This implies that the actual transition voltages may be different (in absolute value) from those of the corresponding ideal ADC. An estimation of the actual values will be available after the Histogram Test is carried out. However, the lowest possible magnitude of the gain (G_{low}) and the highest possible absolute value of the offset (O_{high}) have to be estimated if they are not known a priori.

In short, the amplitude of the sinusoidal stimulus should meet the following condition to minimize testing uncertainty, i.e.

$$A \geq \frac{1}{G_{low}} \left(\frac{T_{ideal}[2^N - 1] - T_{ideal}[1]}{2} + \max\{V_{O_{INL}}, V_{O_{DNL}}\} + e_A + e_C + O_{high} \right). \tag{11.25}$$

This value should be halved in the case of unipolar ADCs. The offset C instead should be set equal to 0 for a bipolar ADC or equal to the sinewave amplitude for a unipolar ADC.

11.4.2.4 Record Length Selection

The main advantage of the SHT is that the variance of the transition level estimators tends to the Cramér-Rao lower bound even in presence of noise [23, 30]. Unfortunately, when the resolution of the converter grows, also the amount of data required to reduce the estimator accuracy below one Least Significant Bit (LSB) tends to increase exponentially. In fact, the relationship between accuracy and record size depends on the estimator variance described by (11.11) and (11.12), for large values of M. Thus, since $Q = FS/(2^N - 1)$, the number of samples for SHT testing tends to increase exponentially as a function of the ADC resolution. This in turn means that the testing time and the related costs also increase exponentially. Various constraints exist in choosing the record length M. On one hand, the upper bound to M is limited by the maximum memory depth available on the chosen acquisition system considered. On the other, the lower bound to M is related to the need of exciting all the ADC codes. An additional bound is described by the following expression, i.e.

$$M \leq \frac{1}{2J v_\rho}, \tag{11.26}$$

which relates the maximum record length to the tolerance v_ρ associated with the frequency ratio f_i/f_s, as described in Sect. 11.4.2.5 [3]. Jitter and phase noise effects, when present, can be compensated by further increasing the number of collected samples through multi-record acquisitions. In particular, if R records of M samples each are collected, the SHT can be performed over $M{\cdot}R$ samples. Moreover, if (11.26) holds, R can be chosen so as to assure the target uncertainty B with a desired confidence level by using the following expression [3, 23, 27], i.e.

$$R \leq D \left[\frac{2^{N-1} K_u}{B} \right] \frac{c\pi}{M} \left\{ 1.13 \left[\frac{\sigma^*}{FS} + \frac{c\sigma_\varphi}{2} \right] + 0.25 \cdot \frac{c\pi}{M} \right\} \tag{11.27}$$

where:

- $D = 1$ for INL measurements and $D = 2$ for DNL measurements;
- $c = 1 + (V_o/V_{FS})$;
- FS is the full-scale range of the ADC under test;
- V_o is the input overdrive;

- σ^* is the total rms value of the random noise (including both additive noise and jitter) for INL measurements; while $\sigma^* = \min(\sigma_w, Q/2.26)$for DNL measurements;
- σ_φ is the rms value of the random phase fluctuations (expressed in radians) affecting the input stimulus;
- B is the target expanded uncertainty, expressed as a fraction of the code bin width;
- K_u can be set equal to $Z_{0,u/2}$ to assure that an individual transition level or code bin width does not exceed B with a specified confidence level v (with $u = 1 - v$). Setting $K_u = Z_{N,u/2}$ corresponds to the "worst case" confidence level, which is associated to the event that none of the transition levels (for INL measurements) or code bin widths (for DNL measurements) is estimated with expanded uncertainty larger than B.

In particular, the percentiles of the normal pdf are given by:

$$Z_{N,u/2} = \sqrt{2} erfc^{-1}\left(1 - (1-u)^{2^{-N}}\right). \tag{11.28}$$

Notice that for a given number of records R, (11.27) can be rearranged to determine the data record size, i.e. [4]

$$M = a\left(4.5\sigma_T \pm \sqrt{(4.5\sigma_T)^2 + \frac{c\pi}{a}}\right) \tag{11.29}$$

where $\sigma_T = \frac{\sigma^*}{V_{FS}} + \frac{c\sigma_\varphi}{2}$ and $a = \frac{c\pi D\left(\frac{2^{N-1}K_u}{B}\right)^2}{R}$.

Observe also that, if the effect of jitter and phase noise is negligible, by inverting (11.12) for the chosen value of B, the target uncertainty can be reached if the following condition is met, i.e.

$$M \cdot R \geq \frac{1.78\, A\, \sigma_w\, K_u^2}{Q^2 \cdot B^2}. \tag{11.30}$$

Expression (11.30) will be used in Sect. 11.5, where solutions to reduce the SHT test duration are explored and discussed.

11.4.2.5 Frequency Selection

The SHT performance models have been derived under the assumption that coherent sampling is used. In addition, the coherent sampling hypothesis can also be used to derive sufficient conditions to maximize the probability that each code bin is excited in the presence of noise. Coherent sampling is ensured by setting

$$\rho = \frac{f_i}{f_s} = \frac{J}{M}, \tag{11.31}$$

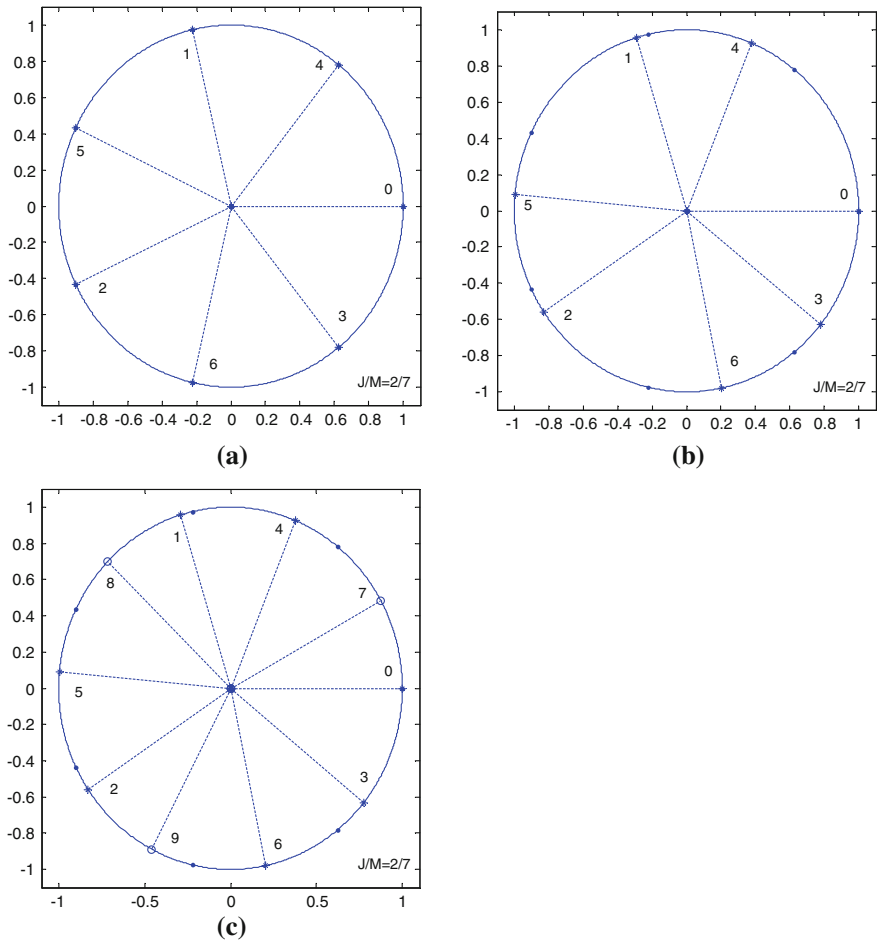

Fig. 11.5 Phase gaps $\Delta\phi$ for $M = 7$ collected samples, with $f_i/f_s = J/M = 2/7$, when sampling is coherent (**a**), and when the ratio slightly differs from 2/7, before (**b**) and after (**c**) collecting 3 more samples

where M and J are reciprocally prime integers as explained in Sect. 11.4.1. The coherence condition ensures that the separation $\Delta\phi$ between consecutive phases of the collected samples is constant and equal to $\Delta\phi_M = 2\Pi/M$, as shown in Fig. 11.5a. Notice that in practice the ratio ρ cannot be completely controlled, due to inaccuracies and drifts affecting both input frequency and ADC sampling frequency. Moreover, variations in ρ induce variations in $\Delta\phi$, resulting in a non-uniform phase separation between the acquired samples. As a result, some $\Delta\phi$ values are smaller than $\Delta\phi_M$. Others instead are larger than $\Delta\phi_M$. Due to this phenomenon (shown in Fig. 11.5a, b), the chosen input stimulus could fail to excite all the ADC codes [36–38]. In order to cope with this problem, some results

have been derived in the literature to provide an upper bound to the drift ratio v_ρ, defined as

$$v_\rho = \frac{\Delta\rho}{\rho}, \tag{11.32}$$

which guarantees the validity of the SHT design procedure. For instance, in [3, 5] the condition

$$v_\rho \leq \frac{1}{2JM} \tag{11.33}$$

has been derived under the assumption that the $\Delta\phi$ variations corresponding to a given v_ρ are within ±25 % [3, 5].

An alternative solution to design an effective test in presence of limited frequency accuracy is to collect a data record longer than M samples. This would result in a smaller maximum phase separation. In fact, in general the ratio between f_s and f_i is usually a real irrational number. Consequently, if the record length is suitably increased, the phase intervals are also progressively filled and partitioned into smaller ones, as shown in Fig. 11.5c, until when none of the remaining phase intervals exceeds $\Delta\phi_M$. In particular, if the drift ratio (11.32) is due to a frequency offset Δf caused by the stimulus frequency f_i only, the following condition holds

$$M^* \cong \Gamma\left(1 + M^2 \frac{|\Delta f|}{f_s}\right), \tag{11.34}$$

where

$$\Gamma = \begin{cases} M_L, & \Delta f > 0 \\ M_R = M - M_L, & \Delta f < 0 \end{cases}, \tag{11.35}$$

with M_L and M_R being the denominators of the two fractions J_L/M_L and J_R/M_R of the Farey Series of order M, adjacent to J/M such that $J_L/M_L < J/M < J_R/M_R$. This expression shows how to guarantee that the maximum phase interval is smaller than the nominal value $\Delta\phi_M$ [36, 38].

11.4.2.6 Comments on the Stimulus Spectral Purity

Spectral purity is an important requirement, since the accuracy of some SHT results strongly relies on the assumption that the stimulus is a pure sinewave. For instance, limited spectral purity may affect the estimation of the ADC Effective Number of Bits (ENOB) [3, 5]. According to [3], typical spectral impurities are harmonics, spurs, wideband noise, and both amplitude and phase modulations in the converted stimulus [3, 5]. Spectral purity may be assessed using a spectrum analyzer, and improved by filtering the signal generator stimulus [27]. Spectral purity requirements for SHT have been defined in [27]. In particular, if E_{INL} and

E_{DNL} are the maximum admitted systematic errors in estimating INL and DNL respectively, the following conditions hold, i.e.

$$\frac{1}{Q}\sum_{i=2}^{h} A_i \cong \frac{2^{N-1}}{A_1}\frac{1}{Q}\sum_{i=2}^{h} A_i \leq E_{INL}, \tag{11.36}$$

and

$$\sqrt{\frac{2}{QA_1}}\left(\sqrt{1+\frac{V_O}{Q}}-\sqrt{\frac{V_O}{Q}}\right)\sum_{i=2}^{h} iA_i \leq E_{DNL}, \tag{11.37}$$

Such conditions have been derived in [27, Sect. 4.2.2], where A_i is the amplitude of the ith harmonic, and $A_1 = A$ is the amplitude of the fundamental component.

11.5 Techniques for Reducing Histogram Test Duration

As explained in Sect. 11.4, the SHT testing time and the related costs increase exponentially with the ADC resolution. In order to tackle this problem and to speed up ADC characterization, some researchers proposed to combine the basic histogram test with other techniques or general error models [39]. Some of these solutions are shortly described in the following sections.

11.5.1 Combined Spectral and Histogram-Based Test

The nonlinear error pattern of an ADC can often be described by a behavioral model consisting of low and high code frequency components (LCF and HCF) [40]. According to this model, the INL value associated with the kth output code is given by

$$INL(k) = {}^{LCF}INL(k) + {}^{HCF}INL(k) = {}^{LCF}INL(k) + \sum_{j=0}^{k} DNL(j). \tag{11.38}$$

In practice, the LCF term is due to significant systematic deviations of the ADC transfer curve from the corresponding nominal values and can be described by a polynomial, i.e.

$${}^{LCF}INL(k) = G_0 + G_1 k + G_2 k^2 + \cdots + G_L k^L, \tag{11.39}$$

with L depending on the wanted accuracy (typically $L < 6$). The HCF component instead is due to the superimposition of several less relevant circuit-related factors

(e.g., component mismatching, glitches due to imperfect simultaneous switching and other input-related systematic singularities of the ADC transfer curve [41]) and can be described by the accumulation of the individual DNL terms, whose low code frequency content is indeed negligible in most cases [40]. The estimation of the LCF and HCF components in (11.38) is performed in 2 steps. First, the sinewave applied to the ADC input is replaced into (11.39). Let $\mathbf{G} = [G_0,G_1,\ldots,G_L]^{\mathrm{T}}$ be the (unknown) vector of the model parameters and $\mathbf{Y} = [Y_0,Y_1,\ldots,Y_L]^{\mathrm{T}}$ be the vector containing the magnitude of the harmonics arising from nonlinear distortion. If the elements of $\mathbf{Y}$ are measured in the frequency or in the time domains (e.g., using a variant of the four-parameter best fitting algorithm [40]), the coefficients of $\mathbf{G}$ can easily be obtained from

$$G = P^{-1} \cdot \hat{Y} \tag{11.40}$$

where $\hat{Y}$ is the vector of the estimated harmonics, and $\mathbf{P}$ is a matrix describing the analytical relationship between the polynomial coefficients and the spectral components observed at the output of the ADC according to model (11.39). In particular, if L is an even number, $\mathbf{P}$ can be rewritten as

$$P = \begin{bmatrix} \binom{0}{0} & 0 & \frac{1}{2}\binom{2}{1} & \cdots & \frac{1}{2}\binom{L}{\frac{L}{2}} \\ 0 & \binom{1}{1} & 0 & \cdots & 0 \\ 0 & 0 & \binom{2}{2} & \cdots & \binom{L}{\frac{L}{2}+1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \binom{L}{L} \end{bmatrix}. \tag{11.41}$$

Once the coefficients of the LCF term are estimated, the HCF component of the INL pattern can be obtained by adding the individual DNL values resulting from a histogram test over a reduced number of samples.

In [40] the authors suggest that using a small triangular waveform superimposed to a progressively growing DC signal is preferable to relax histogram test requirements. However, in practice any kind of histogram test (including the SHT) can be used. Of course, in this case the minimum number of samples that needs to be collected to meet given uncertainty boundaries depends not only on the histogram test per se, but also on the performance of the method used to estimate the LCF component only.

11.5.2 Test Duration Reduction Through Filtering

The procedure described in this section trades a reduction in the histogram test duration for an increment in complexity due to the additional use of the spectral approach used to determine the INL LCF component. However, if the INL pattern exhibits mostly a low frequency spectral content (as it often occurs in practice [42]), then the HCF component can be neglected and the INL can be estimated just by low-pass filtering the transition level sequence obtained with a SHT over a reduced number of samples. In order to remove the estimation noise at no cost in terms of accuracy, the following general conditions must hold, i.e.

- The equivalent noise bandwidth (ENBW) of the chosen filter in the code frequency domain should be large enough not to filter the LCF component significantly;
- The magnitude response of the filter should be as flat as possible in the passband (ideally equal to 1);
- The filter should have a zero phase response to avoid any phase distortion;
- The correlation coefficients between groups of adjacent transition level estimators have to be preliminarily estimated.

Let ε_k, for $k = 1,\ldots, 2^N-1$, be the random variable modeling the estimation error associated with the kth transition level. The correlation coefficient between the kth and the $(k + i)$th estimators is defined as

$$r_{k+i,k} = \frac{E\{\varepsilon_{k+i} \cdot \varepsilon_k\}}{\sigma_{k+i}\sigma_k} \quad i = 1-k,\ldots,2^N-1-k,\ \ k = 1,\ldots,2^N-1, \tag{11.42}$$

where $E\{\cdot\}$ is the expectation operator, while σ_k and σ_{k+i} are the standard deviations associated with the estimators of transition levels k and $k + i$, respectively. In a standard SHT the amount of correlation depends on the amplitude of the AWGN superimposed to the test stimulus. In particular, if the amplitude of the AWGN is smaller than the ideal code bin width Q, the estimation errors tend to be uncorrelated and identically distributed. On the contrary, when the noise amplitude grows, the amount of correlation also increases [23]. In this case the correlation coefficients decrease as function of the distance i between any pair of transition levels, regardless of both the output code value (i.e., $r_{k+i,k} = r_i$) and the resolution of the ADC. In [43] it is shown that any pair of transition level estimators that are distant more than S bins from one another, with S given by

$$S = \left\lfloor 2 \cdot \frac{\sigma_w}{Q} \right\rfloor \ll 2^N \tag{11.43}$$

can be assumed to be uncorrelated. Note that in (11.43) the operator $\lfloor\cdot\rfloor$ rounds the argument to the nearest smaller integer value. If we assume that

1. $r_i \approx 0$ for $i > S$;
2. the adopted filter has a finite impulse response;

3. the coefficients of the filter impulse response are almost constant in the range $[k - S, k]$ for any value of $k = 1,\ldots 2^N - 1$;
4. the ADC input is suitably overdriven by the sinewave stimulus and
5. the number of acquired record $R = 1$,

then the bias of the kth filtered transition level estimator is negligible and the corresponding variance based on (11.30) is modified as follows [43]

$$\sigma_{k_f}^2 \cong \frac{1.78\sigma_w}{M \cdot Q^2} \sqrt{A^2 - \overline{T}_k^2} \cdot ENBW \cdot \left(1 + 2 \cdot \sum_{i=1}^{S} r_i\right) k = 1, \ldots, 2^N - 1, \quad (11.44)$$

where the subscript f in (11.43) stands for "filtered", $M_f < M$ is the reduced number of test samples used for the SHT, and $ENBW \leq 1$ is the two-side normalized equivalent noise bandwidth of the chosen unit-gain filter. In essence, expression (11.44) implies that in the best case (i.e., when the peak-to-peak amplitude of the superimposed AWGN noise is smaller than $Q/2$), adjacent estimators can be assumed to be uncorrelated and the accuracy in estimating the LCF component is comparable with a standard SHT over $M = M_f/ENBW$ samples. Conversely, if the noise amplitude is larger than $Q/2$ (e.g., simply because the resolution of an ADC with a given full-scale range grows), the correlation coefficients in (11.42) are no longer negligible. Accordingly, the estimation uncertainty as well as the test duration are reduced by a factor lower than $ENBW$.

Two types of elementary filters are proposed in the literature to implement the *fast sinewave histogram test* based on the principle above (FHST). In [43] the authors describe a procedure relying on Discrete Fourier Transform (DFT) filtering. Such a procedure is summarized by the flow chart shown in Fig. 11.6. A similar approach based on a centered (i.e., non-causal) moving average filter over P samples is reported in [44]. In this case, $ENBW = 1/P$. Both solutions are very friendly because the relationship between the definition of the low-pass filter and the $ENBW$ value is extremely simple. Also, both filters exhibit intrinsically a zero-phase response. This instead is not possible when more sophisticated (e.g. infinite input response, IIR) filters are chosen.

11.6 Illustrative Examples

11.6.1 Simplest Possible Example

We will begin our illustration of how the SHT can estimate the transfer function of an ADC in a very simple and ideal situation. All the examples in this section are based on simulations. The first stage of the SHT consists in applying a sinewave to the input and to acquire a given number of samples. Let us consider a 3-bit ADC (i.e. $N = 3$) with an ideal bipolar transfer function with "no true zero," a 5 V full-scale voltage and a sampling rate of $f_s = 200$ kHz. Assume that both internal noise

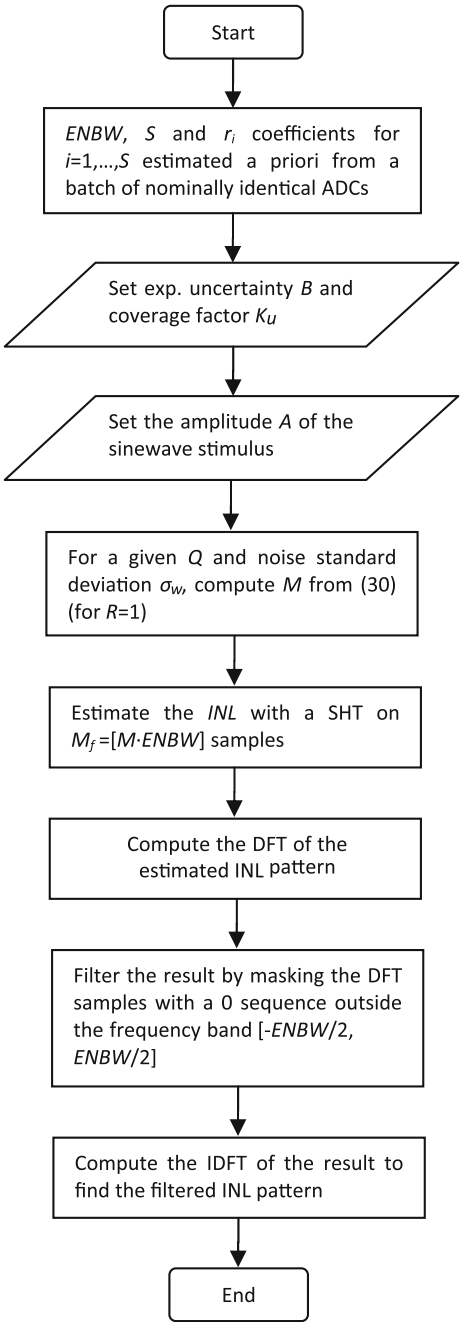

Fig. 11.6 Flowchart of a fast sinewave histogram test (FSHT) based on INL filtering

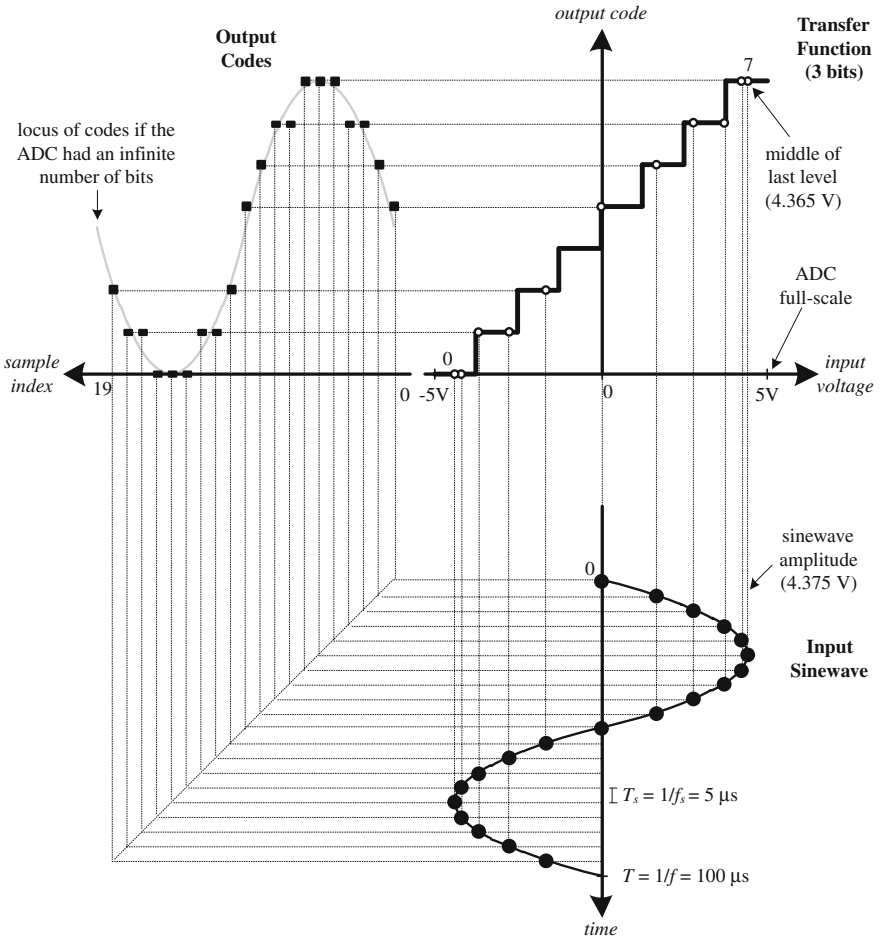

Fig. 11.7 Illustration of the quantization of a sinewave

and timing errors are negligible and that a sinewave with amplitude $A = 4.375$ V, no offset (i.e. $C = 0$ V) and frequency $f = 10$ kHz is used as input stimulus. Since the ratio between stimulus signal frequency and sampling frequency is 1/20, 20 samples cover exactly one full period of the sinewave.

Figure 11.7 illustrates this step—it shows the input signal, the ADC transfer function and the output codes. The bottom right quadrant of the figure shows $M = 20$ input sinewave samples spaced by 5 µs over one full period (i.e. 100 µs). The solid circles show the values of the input voltage at each sampling instant.

The dotted vertical lines that emerge from those circles reach the 3-bit ADC transfer function depicted in the upper right quadrant of the figure. The output codes resulting from the quantization of each sample (open circles) cover the whole range from 0 (000_2) to 7 (111_2). Note that the amplitude of the sinewave

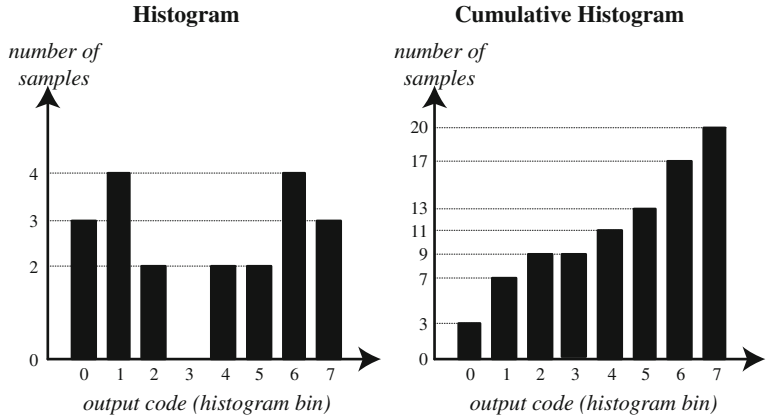

Fig. 11.8 Histogram (*left*) and cumulative histogram (*right*) of the collected sample codes

(4.375 V) corresponds to the middle of the last quantization level. In the example under consideration all quantization levels have the same width W equal to the ideal code bin width $Q = 1.25$ V ($2 \times V_{FS}/2^N = 2 \times 5/2^3$). The transition voltages in a bipolar ADC with a transfer function with no true zero are also the ideal ones, that is, $T[k] = -V_{FS} + k \times Q$ with k ranging from 1 to 7. This ideal transfer function has a unit gain and no offset error. Moreover, all the values of $INL[k]$ and $DNL[k]$ are equal to 0.

The resulting output codes of all the 20 samples can be seen in the upper left quadrant of the figure (filled squares). As expected many samples have the same output code. Just for clarity one can see in grey the locus of the output codes if the ADC had an infinite number of bits. Note that the solid squares are not on top of that grey line because the ADC used in this example has just 3 bits.

The next stage of SHT consists in building the histogram (and the cumulative histogram) of the output codes associated with the acquired samples. The result is shown in Fig. 11.8 (left) together with the cumulative histogram (right) built by counting the number of samples with a code equal to *or lower* than each of the 8 possible output codes. Note that the cumulative histogram is going to be used in the next stage of the SHT for the estimation of the ADC transition voltages. Observe that there are no samples with code 3. Furthermore, the number of counts of the histograms slightly differ if the phase of the first sample changes.

Afterwards, by using (11.6), with $C = 0$, $A = 4.375$ V, $M = 20$ and $H_c[k]$ as depicted in Fig. 11.8 (right) the threshold levels are obtained. In general, the estimated values are different from the actual ones even when there is no noise or uncertainty affecting the sampling instant. This is a consequence of equating the values of the code density function to the relative number of samples obtained for each output code. If an infinite number of samples were acquired for testing, the estimated transition voltages would be exactly the same as the actual ones. Consider, for example, transition level 6. The probability that the input signal is lower than $T[6]$ (i.e. 2.5 V), for a uniformly distributed initial phase between 0 and 2π, is

Table 11.2 SHT results in the case of a 3-bit ADC with an ideal transfer function

k	$T[k]$ (V)	$H_c[k]$	$\hat{T}[k]$ (V)	$\hat{W}[k]$ (V)	$INL[k]$ (LSB)	$DNL[k]$ (*LSB*)
0	–	3	–	–	–	–
1	−3.75	7	−3.898	1.912	0	0.471
2	−2.5	9	−1.986	1.302	−0.471	0.002
3	−1.25	9	−0.684	0	−0.473	−1
4	0	11	−0.684	1.369	0.526	0.053
5	1.25	13	0.684	1.302	0.473	0.002
6	2.5	17	1.986	1.912	0.471	0.471
7	3.75	20	3.898	–	0	–

The columns, from left to right, contain (1) the index (k), (2) the actual transition voltage, (3) the number of counts of the cumulative histogram H_c, (4) the estimated transition voltages, (5) the estimated code bin widths, (6) the estimated terminal based INL and (7) the estimated terminal based DNL

69.4 % in accordance with (11.6). However, in our example, 13 out of 20 samples have an output code lower than 6 (see Table 11.2). This means that 65 % of the collected samples has a code lower than 6, which is clearly smaller than 69.4 %. This equivalently means that $\hat{T}[6] = 1.986$ V.

The rest of the parameters of the ADC transfer function are computed from the estimated transition voltages. The code bin widths, for example, are computed by subtracting the values of two consecutive transition voltages. The result can be seen in the last column of Table 11.2. Note that the first (0) and the last (7) quantization levels have no width, since they extend to infinity.

In the following, the ADC gain and offset errors are estimated using a "terminal-based" definition. According to this definition gain and offsets are the values that need to be multiplied and added, respectively, to all transition voltages so as to make the first and the last transition levels (i.e. $T[1]$ and $T[7]$) equal to the corresponding ideal values. In this case the ideal values are −3.75 V and 3.75 V, respectively. The estimated values, however, are different as seen in Table 11.2 (−3.898 and 3.898 V respectively). The estimated gain and offset are, as a consequence, 0.962 and 0 respectively ($\pm 3.898 \times 0.962 + 0 = \pm 3.75$). The null offset error is expected, since in the current example the actual and estimated transition voltages are symmetrical (i.e. the bipolar transfer function has *no true zero*). Note once again that in spite of the fact that the actual simulated ADC behaves exactly like an ideal one and no noise or other uncertainty contributions are considered, the estimated gain is different from the actual one (i.e. equal to 1).

Finally, the *INL* and *DNL* can be computed from (11.1) and (11.2), in Sect. 11.2. The results are shown in the two rightmost columns of Table 11.2. Since the values of gain and offset error are "terminal-based", the obtained *INL* and *DNL* are also "terminal-based". Notice the relatively high values of *INL*. This is because only 20 samples are acquired. In fact, the higher the number of collected samples the more accurate the estimation becomes.

Notice also that the *DNL* value is equal to −1 for quantization level 3 ($DNL[3] = -1$). This generally means the ADC has a "missing code" that is, no

sample can be produced with this output code. In terms of the transfer function, this means that the width of the quantization level is 0 (i.e., $W[3] = 0$) or equivalently that the transition voltages $T[3]$ and $T[4]$ have the same value. Of course this is an erroneous conclusion in the case of the ADC under consideration and is simply due to the low number of samples acquired. It just happened that none of the 20 samples acquired ended up in the range of the quantization level 3 (between -1.25 and 0). If more samples were acquired then, eventually, some samples would have an output code of 3.

11.6.2 Example Using a Real-World Scenario

We will consider now the more realistic case of an 8-bit ADC clocked at 1 MHz in the ± 5 V range, with $Q = 39$ mV. The transfer function is characterized by a gain $G = 0.95$, an offset $O = 40$ mV and an INL pattern exhibiting an inverted parabola shape with a maximum value $INL_{max} = 3$ LSB (Fig. 11.9, left). The made-up analytical description of the transition voltages created for this example is

$$T[k] = \frac{1}{G}\left(T_{ideal}[k] + \frac{4Q \times INL_{\max}}{2^N - 2}\left(\frac{1}{2^N - 2}k^2 - k\right)\right) - O. \tag{11.45}$$

The corresponding DNL has a linear dependence on its index (Fig. 11.9, right). In fact, the DNL can be seen has the derivative of the INL. Output codes in the low end of the range have negative DNL, that is, their width is smaller than the ideal value, whereas the output codes in the high end have a width larger than the ideal value (positive DNL). The maximum value of the actual DNL is 0.047 LSB (DNL_{max}). Typical ADCs do not have such a "well-behaved" transfer function. However, the regular shape of this INL pattern leads to a better illustration.

The SHT requires setting the values of stimulus signal amplitude, offset and frequency as well as the number of samples to acquire. As seen in Sect. 11.4, additive noise, phase noise, timing jitter and frequency error affect the estimates and should be preliminarily determined. In the example presented here 60 mV due to the internal ADC noise are referred to its input and 80 mV are added to the input stimulus. On the whole, the standard deviation σ_w of the total additive noise is

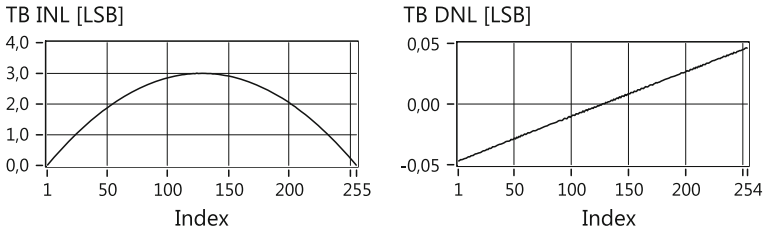

Fig. 11.9 Actual terminal based INL (*left*) and a DNL (*right*) of the 8-bit simulated ADC

100 mV. Phase noise with $\sigma_\varphi = 0.01\pi$ was assumed to affect the generated signal. However, no ADC sampling jitter (i.e. $\sigma_t = 0$) was taken into account.

Besides such sources of noise, other uncertainty sources were considered to affect the signal produced by the generator. In particular, a maximum error of 20 mV for both amplitude (e_A) and offset settings (e_C) and a maximum relative frequency error $\varepsilon_f = 50$ ppm were included in the simulations.

Again, the minimum gain and the maximum offset of the ADC must be preliminarily estimated in order to choose a stimulus signal that does not fail to excite all the output codes. In the presented case study, the minimum ADC gain and the maximum offset are assumed to be $G_{low} = 0.9$ and $O_{high} = 50$ mV respectively. The actual values for the simulated ADC (i.e. 0.95 and 40 mV, respectively) are in accordance with the chosen limits.

In this section a bipolar ADC is considered. Therefore, using expression (11.25) from Sect. 11.4, we have that the amplitude setting of the stimulus generator results from

$$A_{setting} \geq \frac{1}{0.9}\left(V_{FS} - Q + \max\left\{\max\left\{2 \times 0.1, \frac{0.1^2 \times 2^8}{5 \times 0.1}\right\}, \max\left\{3 \times 0.1, 0.1\sqrt{1.43\frac{3}{8 \times 0.1}}\right\}\right\} + 0.02 + 0.02 + 0.05\right) \tag{11.46}$$

where in (11.46) $B = 0.5$ LSB and the offset setting is 0. From (11.46) it follows that the signal amplitude should be larger than 6.75 V. The next step in setting up the SHT is to determine the number of records and the number of samples per record. The dynamic performance of an ADC is dependent on the frequency of the input signal. The Histogram Test is usually carried at different stimulus frequencies. Here, we suppose to characterize the ADC around $f = 10$ kHz. The exact testing frequency and the number of samples to acquire per record have to be determined considering two conditions. First of all, the ratio between the stimulus frequency and sampling frequency has to be equal to the ratio of two integer numbers that have no common factors (as per Eq. (11.31) in Sect. 11.4). These integer numbers represent the number of signal cycles J in one record of samples and the number of samples per record M, respectively. The second condition is that the product of those two integers has to be lower than a given limit determined by the frequency accuracy of the stimulus signal and of the sampling clock as described by (11.33) in Sect. 11.4. Since the sampling clock is assumed not to be affected by frequency fluctuations, while the stimulus has a relative frequency offset up to 50 ppm, $J \times M \leq 10^4$.

Since the stimulus frequency has to be around 10 kHz and the sampling frequency is 1 MHz the frequency ratio ρ has to be about 0.01. As a consequence J and M can be set equal to 9 and 1024, respectively ($\rho = 0.0088$). In fact, all numbers which are a power of 2 never have common factors with odd numbers. With such values the stimulus signal frequency is 8.789 kHz, which is close to 10 kHz.

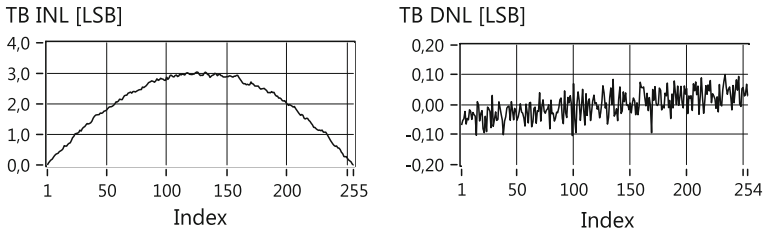

Fig. 11.10 Estimated terminal based INL (*left*) and a DNL (*right*) of the 8-bit simulated ADC

Having defined the number of samples per record, the next step is to determine the minimum number of records to achieve the given target uncertainty (i.e. $B = 0.5$ LSB in this example). In particular, it follows from (11.27) that

$$R_{INL} \geq \left[\frac{2^{8-1} \times 3.719}{0.5}\right]^2 \frac{1.716\pi}{1024}\left\{1.13\left[\frac{0.1}{5} + \frac{1.716 \times 0.01\pi}{2}\right] + 0.25\left[\frac{1.716\pi}{1024}\right]\right\} = 259.3,$$

and

$$R_{DNL} \geq 2\left[\frac{2^{8-1} \times 3.719}{0.5}\right]^2 \frac{1.716\pi}{1024}\left\{1.13\left[\frac{0.017}{5} + \frac{1.716 \times 0.01\pi}{2}\right] + 0.25\left[\frac{1.716\pi}{1024}\right]\right\} = 340.3$$

for INL and DNL estimation, respectively. Variable c, depends on the amount of overdrive and it is defined as $c = 1 + 2(Vo/V_{FS})$ with $V_o = A_{setting} - (V_{FS} - Q)$. Since we want to estimate INL and DNL at the same time (in the same test) the larger of these two values is used, i.e. $R = 341$. The total number of samples to acquire is thus 349,184 (341×1024).

The results obtained for INL and DNL are represented in Fig. 11.10 with an estimated gain of 0.94975 and offset of 40.784 mV. By comparing Fig. 11.10 with Fig. 11.9 we can see that the estimated values of INL and DNL are close to the actual ones. Moreover, the uncertainty associated with the INL estimates is smaller than expected, because the chosen number of records was set to meet the accuracy requirements for DNL estimation.

In fact, the maximum estimated INL and DNL values are 3.1 LSB and 0.103 LSB, respectively. Table 11.3 sums up symbols and values of the parameters used in this simulation.

11.6.3 Examples with Incorrect Test Set-Up

Imagine that the sinewave amplitude used ($A_{setting}$) does not include the overdrive needed to avoid threshold estimation biasing. This may happen, for instance, if the previous simulation is repeated using $A_{setting} = 5$ V.

Table 11.3 Simulation setting for a SHT applied to an 8-bit ADC with a non-ideal transfer function

Analog-to-digital converter				Stimulus generator		
Parameter name	*Symbol*	*Actual*	*Estimated*	*Parameter name*	*Symbol*	*Value*
Number of bits	N	8	–	Amplitude	$A_{setting}$	6.75 V
Full-scale	V_{FS}	5 V	–	Offset	$C_{setting}$	0
Sampling frequency	f_s	1 MHz	–	Frequency	$f_{setting}$	8.789 kHz
Gain	G	0.95	0.94975	Amplitude error	e_A	20 mV
Offset	O	40 mV	40.784 mV	Offset error	e_C	20 mV
Prior gain	G_{low}	0.9	–	Frequency error	ε_f	50 ppm
Prior offset	O_{high}	50 mV	–	Additive noise	σ_{GEN}	80 mV
Maximum INL	INL_{max}	3 LSB	3.047 LSB	Phase noise	σ_φ	0.01π rad
Maximum DNL	DNL_{max}	0.047 LSB	0.103 LSB	Histogram test		
Additive Noise	σ_{ADC}	60 mV	–	*Parameter name*	*Symbol*	*Value*
Jitter	σ_t	0	–	Number of samples	M	1024
Clock frequency error	ε_{fs}	0	–	Number or records	R	341

The corresponding results can be seen in Fig. 11.11. It is noticeable that the DNL values at the edges of the transfer function are missed. This has an even more severe consequence in the estimation of the terminal-based INL.

In another simulation only 1 record of 1024 samples (instead of 341) is collected. As a result, the uncertainty affecting the estimated INL and DNL values is much larger, as shown in Fig. 11.12.

A further simulation was carried out to show the importance of choosing the proper value for stimulus signal frequency. In the original simulation a frequency of 8.789 kHz was used and 341 records of 1024 samples were collected. If the stimulus signal frequency were exactly 10 kHz and just one record of 349,184 samples were acquired, the testing results would be completely different, as shown in Fig. 11.13. This is due to the fact the phases of the stimulus signal are not uniformly distributed. Indeed, there are some input voltages that are sampled much more often than expected, and others that are sampled rarely. Consequently, the samples used to build the histogram are not distributed according to the model described in Sect. 11.3.

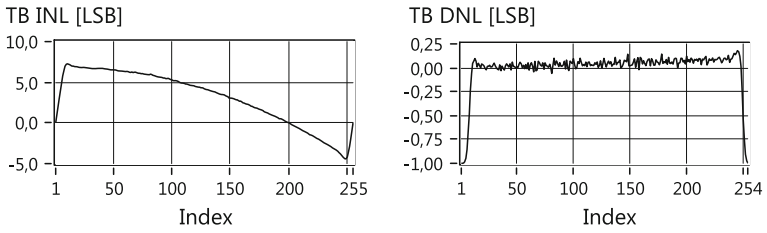

Fig. 11.11 Estimated terminal-based INL (*left*) and a DNL (*right*) of the 8-bit simulated ADC when an insufficient stimulus signal amplitude is used (5 V instead of 6.75 V)

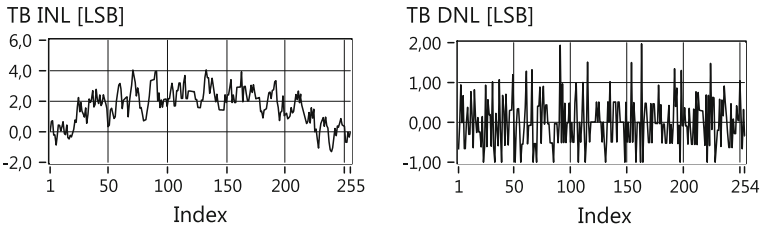

Fig. 11.12 Estimated terminal-based INL (*left*) and a DNL (*right*) of the 8-bit simulated ADC when only 1 record of samples is acquired (instead of 341)

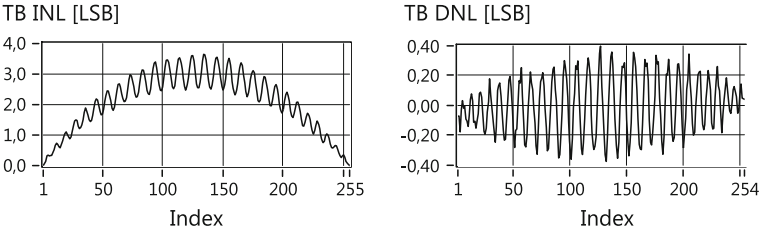

Fig. 11.13 Estimated terminal based INL (*left*) and a DNL (*right*) of the 8-bit simulated ADC when a stimulus signal of 10 kHz is used and all the 349184 samples are acquired consecutively (1 record)

11.7 Conclusion

The histogram-based techniques are recommended in the main standard documents for ADC testing as they assure excellent performance in estimating the ADC transition voltages as well as the INL and DNL patterns, especially at low frequencies. While the theory underlying the histogram-based techniques is straightforward in principle, the practical design of this kind of tests requires a careful analysis of various (nonideal) issues and a proper selection of several parameters. The main issues and parameters affecting histogram-based tests include the following, i.e. type, frequency and spectral purity of the input stimulus, additive wideband noise power, phase noise power and jitter of both input stimulus and sampling signal, input overdrive amplitude with respect to the ADC full-scale range, number of data records and number of samples per record. In this chapter, after a general description of the histogram-based techniques, the selection criteria of the various testing parameters listed above are orderly presented and explained in the case of the Sine-wave Histogram Test (SHT), which is indeed the most widely used approach. In order to provide the reader with a clear overview of the testing procedure some simple, but realistic examples of SHT testing results are reported and commented. In addition, some solutions to reduce the SHT duration are shortly described.

References

1. Doernberg, J., Lee, H.-S., Hodges, D.A.: Full-speed testing of A/D converters. IEEE J. Solid-State Circ. **19**(6), 820–827 (1984)
2. Hagelauer, R., Oehler, F., Rohmer, G., Sauerer, J., Seitzer, D., Schmitt, R., Winkler, D.: Investigations and measurements of the dynamic performance of high-speed ADCs. IEEE Trans. Instrum. Meas. **41**(6), 829–833 (1992)
3. IEEE Standard for Terminology and Test Methods for Analog-to-Digital Converters: IEEE standard 1241–2011, 2011
4. IEC 60748-4-3: Semiconductor devices—integrated circuits—part 4–3: interface integrated circuits—dynamic criteria for analogue-digital converters (ADC), 1st edn, 2006–2008
5. IEEE Standard for Digitizing Waveform Recorders: IEEE Standard 1057–2007 (Revision of IEEE 1057–1994), 2008
6. IEC 62008: Performance characteristics and calibration methods for digital data acquisition systems and relevant software, 1st edn, pp. 28–29, 2005–2007
7. Alegria, F., Arpaia, P., Cruz Serra, A.M., Daponte, P.: Performance analysis of an ADC histogram test using small triangular waves. IEEE Trans. Instrum. Meas. **51**(4), 723–729 (2002)
8. Renovell, M., Azaïs, F., Bernard, S., Bertrand, Y.: Hardware resource minimization for histogram-based ADC BIST. In: Proceedings of IEEE VLSI Test Symposium, pp. 247–252, May 2000
9. Provost, B., Sanchez-Sinencio, E.: On-chip ramp generators for mixed-signal BIST and ADC self-test. IEEE J. Solid-State Circ. **38**(2), 263–273 (2003)
10. Korhonen, E., Hakkinen, J., Kostamovaara, J.: A robust algorithm to identify the test stimulus in histogram-based A/D converter testing. IEEE Trans. Instrum. Meas. **56**(6), 2369–2374 (2007)
11. Wei, J., Agrawal, V.D.: A DSP-based ramp test for on-chip high-resolution ADC. In: Proceedings of IEEE Southeastern Symposium on System Theory (SSST), pp. 203–207, March 2011
12. Linnenbrink, T.E., Tilden, S.J., Miller, M.T.: ADC testing with IEEE Standard 1241–2000. In: Proceedings of IEEE Instrumentation and Measurement Technology Conference (IMTC), Budapest, Hungary, pp. 1986–1991, May 2001
13. Macii, D., Pianegiani, F., Carbone, P., Petri, D.: A stability criterion for high-accuracy delta-sigma digital resonators. IEEE Trans. Instrum. Meas. **55**(2), 577–593 (2006)
14. Xing, H., Jiang, H., Chen, D., Geiger, R.L.: High-resolution ADC linearity testing using a fully digital-compatible BIST strategy. IEEE Trans. Instrum. Meas. **58**(8), 2697–2705 (2009)
15. Martins, R.C., da Cruz Serra, A.M.: Automated ADC characterization using the histogram test stimulated by Gaussian noise. IEEE Trans. Instrum. Meas. **48**(2), 471–474 (1999)
16. Holub, J., Vedral, J.: Stochastic testing of ADC step-Gauss method. Comput. Stand. Interface **26**(3), 251–257 (2004)
17. Björsell, N., Händel, P.: Truncated Gaussian noise in ADC histogram tests. Elsevier Meas. **40**(1), 36–42 (2007)
18. Holcer, R., Michaeli, L., Šaliga, J.: DNL ADC testing by the exponential shaped voltage. IEEE Trans. Instrum. Meas. **52**(3), 946–949 (2003)
19. Gamad, R.S., Mishra, D.K.: Gain error offset error and ENOB estimation of an A/D converter using histogram technique. Elsevier Meas. **42**(4), 570–576 (2009)
20. Flores, MdGC, Negreiros, M., Carro, L., Susin, A.A.: INL and DNL estimation based on noise for ADC test. IEEE Trans. Instrum. Meas. **53**(5), 1391–1395 (2004)

21. Moschitta, A., Carbone, P., Petri, D.: Statistical performance of Gaussian ADC histogram test. In: Proceedings of 8th International Workshop on ADC Modelling and Testing (IWADC), Perugia, Italy, pp. 213–217, Sept 2003
22. Corrado, M., Rapuano, S., Šaliga, J.: An overview of different signal sources for histogram based testing of ADCs. Elsevier Meas. **43**(7), 878–886 (2010)
23. Blair, J.: Histogram measurement of ADC nonlinearities using sinewaves. IEEE Trans. Instrum. Meas. **43**(3), 373–383 (1994)
24. Corrêa Alegria, F., Cruz Serra, A.: ADC transfer curve types—a review. Comput. Stand. Interfaces, Elsevier **28**(5), 553–559 (2006)
25. Papoulis, A.: Probability, Random Variables and Stochastic Processes, 3rd edn. McGraw-Hill, Singapore (1991)
26. Vora, S.C., Satish, L.: ADC static nonlinearity estimation using linearity property of sinewave. IEEE Trans. Instrum. Meas. **60**(4), 1283–1290 (2011)
27. Dynad Draft v3.4: Dynamic Testing of Analog-to-Digital Converters Using Sinewaves—DYNAD
28. Carbone, P., Nunzi, E., Petri, D.: Statistical efficiency of the ADC sinewave histogram test. IEEE Trans. Instrum. Meas. **51**(4), 849–852 (2002)
29. Corrêa Alegria, F.A., Moschitta, A., Carbone, P., Serra, A.C., Petri, D.: Effective ADC linearity testing using sinewaves. IEEE Trans. Circ. Syst. I Regul. Papers **52**(7), 1267–1275 (2005)
30. Carbone, P., Petri, D.: Noise sensitivity of the ADC histogram test. IEEE Trans. Instrum. Meas. **47**(4), 849–852 (1998)
31. Moschitta, A., Carbone, P.: Noise parameter estimation from quantized data. IEEE Trans. Instrum. Meas. **56**(3), 736–742 (2007)
32. Fodor, B., Kollar, I.: ADC testing with verification. IEEE Trans. Instrum. Meas. **57**(12), 2762–2768 (2008)
33. Gendai, Y.: The maximum-likelihood noise magnitude estimation in ADC linearity measurements. IEEE Trans. Instrum. Meas. **59**(7), 1746–1754 (2010)
34. Sarhegyi, A., Balogh, L., Kollar, I.: An efficient approximation for maximum, likelihood estimation of ADC parameters. In: Proceedings of IEEE International Instrumentation and Measurement Technology Conference (I2MTC), pp. 2656–2661, Graz, Austria, May 2012
35. Alegria, F.C., Serra, A.C.: The hystogram test of ADCs with sinusoidal stimulus is unbiased by phase noise. IEEE Trans. Instrum. Meas. **58**(11), 3847–3854 (2009)
36. Blair, J.J.: Selecting test frequencies for sinewave tests of ADCs. IEEE Trans. Instrum. Meas. **54**(1), 73–78 (2005)
37. Carbone, P., Chiorboli, G.: ADC sinewave histogram testing with quasi-coherent sampling. IEEE Trans. Instrum. Meas. **50**(4), 949–953 (2001)
38. Moschitta, A., Carbone, P.: An automated procedure for selecting frequency and record length when testing data converters. In: Proceedings of IEEE International Instrumentation and Measurement Technology Conference (I2MTC), Singapore, pp. 1711–1715, May 2009
39. Wegener, C., Kennedy, M.P.: Linear model-based testing of ADC nonlinearities. IEEE Trans. Circ. Syst. I **51**(1), 213–217 (2004)
40. Serra, A.C., da Silva, M.F., Ramos, P.M., Martins, R.C., Michaeli, L., Šaliga, J.: Combined spectral and histogram analysis for fast ADC testing. IEEE Trans. Instrum. Meas. **54**(4), 1617–1623 (2005)
41. Arpaia, P., Daponte, P., Michaeli, L.: The influence of the architecture on ADC modeling. IEEE Trans. Instrum. Meas. **48**(5), 956–967 (1999)
42. Attivissimo, F., Giaquinto, N., Kale, I.: INL reconstruction of A/D converters via parametric spectral estimation. IEEE Trans. Instrum. Meas. **53**(4), 940–946 (2004)
43. Stefani, F., Macii, D., Moschitta, A., Carbone, P., Petri, D.: A simple and time-effective procedure for ADC INL estimation. IEEE Trans. Instrum. Meas. **55**(4), 1383–1389 (2006)

44. Stefani, F., Moschitta, A., Macii, D., Carbone, P., Petri, D.: Fast estimation of A/D converter nonlinearities. Meas. Elsevier Sci. **39**(3), 232–237 (2006)
45. Linnenbrink, T.E., Blair, J., Rapuano, S., Daponte, P., Balestrieri, E., De Vito, L., Max, S., Tilden, S.J.: Addition to ADC testing. IEEE Instrum. Meas. Mag. **9**(5), 46 (2006)
46. ISO/IEC Guide 98-3:2008: Guide to the Expression of Uncertainty in Measurement, Geneve, Switzerland

Chapter 12
DAC Standardization and Advanced Testing Methods

Eulalia Balestrieri, Domenico Luca Carnì, Pasquale Daponte, Luca De Vito, Domenico Grimaldi and Sergio Rapuano

12.1 Introduction

Nowadays, due to the wide availability and low cost of digital processors, most of the electronic equipment interacting with analogue quantities process the incoming data in digital format and translate the results into the analogue format by means of Digital-to-Analogue Converters (DACs). As a consequence, the quality of the results provided by such equipment depends on the characteristics of the DAC.

To quantitatively assess the non-ideal behaviour of DACs in different operating conditions it is necessary to rely on a suitable set of parameters and test methods. Several proposals can be found in scientific and technical literature to provide design and test engineers with the minimal toolset allowing a reliable quantitative assessment of the performances of an actual DAC. Recently, with the aim of organizing them better, the Waveform Measurement and Analysis Technical Committee (TC-10) of the IEEE Instrumentation and Measurement Society has delivered an IEEE Standard on "Definitions and test methods for DAC converters".

Sections 12.1 and 12.2 provide respectively an overview of the main existing DAC standards and a brief discussion about the need for a standard consistent terminology. Then, some of the most used figures of merit definition issues are dealt with and new proposals suggested as a guidance to the design and test engineers as well as researchers starting their involvement in the DAC field. Finally, Sect. 12.3 focuses on test methods proposed in the literature, which are used to overcome the problems arising from the increasing resolution and speed of the new generations of DACs.

E. Balestrieri · P. Daponte · L. De Vito · S. Rapuano
Department of Engineering, University of Sannio, Benevento BN, Italy

D. L. Carnì (✉) · D. Grimaldi
Department of Informatics, Modeling, Electronics and System Engineering, University of Calabria, Rende CS, Italy
e-mail: dlcarni@deis.unical.it

P. Carbone et al. (eds.), *Design, Modeling and Testing of Data Converters*, Signals and Communication Technology, DOI: 10.1007/978-3-642-39655-7_12,

12.2 DAC Standardization

Until 2012 the main existing DAC standards were: (i) IEC 60748–4, which included only DAC static specifications and test methods [1]; (ii) IEEE Std. 746 which addressed the testing of Analog-to-Digital (ADC) and Digital-to-Analog converters, used for PCM television video signal processing [2], (iii) JEDEC Std. 99, addendum number 1, which dealt with the terms and definitions used to describe ADC and DAC converters and did not include test methods [3], and (vi) EBU Technical Information I15–1998 [4] which reported ADC and DAC performance parameters for testing in conformity with ITU-R Recommendations BT.601 and BT.656.

The standardization of DAC terminology and test methods has been characterized, for years, by the lack of a comprehensive approach focusing specifically on terms, definitions and test methods for a wide range of applications. In order to fill this lack the Waveform Measurement and Analysis Technical Committee (TC-10) of the IEEE Instrumentation and Measurement Society has recently published a new standard providing common terminology and test methods for the testing and evaluation of DACs [5].

12.3 Proposals for DAC Standard Consistent Terminology

Although DACs perform the same function, in practice different converters may behave quite differently. Moreover, DACs made by different manufacturers are often not comparable, due to the different ways of specifying and testing parameters, making the user fail in selecting the best suited device for his needs.

Standards can help users to avoid misinterpretation of the real device performance and the dependence on a single manufacturer allowing a broader choice among cheaper products. Users can also have increased confidence in the quality and reliability of manufacturers who use standards. DAC metrology standardization leads to benefits also to manufacturers in terms of compatible products and services, reduction in development costs, easy conformity assessment, mass production, economy of scale as well as facilitation of satisfying user requirements and access to new markets. However, developing a DAC metrology standard means not only providing methods to be applied in testing but also a consistent terminology, essential so that written specifications can be interpreted properly and misunderstandings about common terms can be avoided. Moreover, the choice of the term to be used for the particular specification to be tested is as important as its unambiguous definition, prerequisite for DAC test method consistency and interoperability.

In this Section DAC terminology taken from existing standards, scientific literature and manufacturers' documentation has been collected in order to highlight similarities, ambiguities and voids in the parameter definitions. Each proposed definition has been joined with its measurement unit taking as reference the

International System of Units (SI). The main parameters presented in the following have been classified according to the test method (static or dynamic) and the measurement domain (time or frequency).

12.3.1 Static DAC Parameters

The target of the DAC static characterization is its transfer characteristic. The differences among the actual and the nominal output values are used to estimate the static parameters. Next subsections analyze one by one the most used DAC static parameters: gain, offset, INL and DNL.

12.3.1.1 Gain and Offset

The manner in which the DAC actual characteristic is matched to the ideal characteristic must be clearly specified to define gain and offset unambiguously [6]. The general transfer curve of the DAC can be represented by the following equation:

$$y = mx + b \tag{12.1}$$

where m is the DAC gain and b is the DAC offset. There are two main methods for determining gain and offset. The former is the *endpoint* method which sets gain based on the minimum scale point and full-scale (FS) point. Offset is determined from the intercept of the line (Fig. 12.1a). The latter is the *best-fit* line. The m and b (gain and offset) parameters are set based on the minimum mean squared error from line to sample (Fig. 12.1b) [7].

The IEC 60748–4 defines the gain as "*the slope of the straight line of the transfer diagram or of a specified part of it expressed as the quotient of a change in analogue output quantity, by the change in digital input quantity, stated as number of steps, producing it*". From the reported definition, the ideal straight line is traced between *"the specified points for the most positive (least-negative) and*

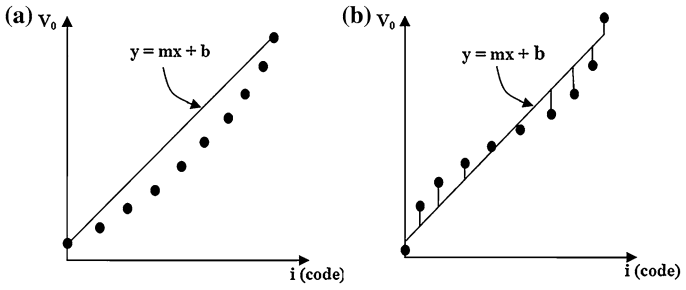

Fig. 12.1 **a** Endpoint method. **b** Best-fit method [7]

the most negative (least positive) nominal step values respectively". As a consequence, the endpoint method seems to be adopted for the gain value determination. The best-fit method is not considered. Moreover, the slope of the straight line representing the entire transfer characteristic is different from the slope values achieved considering only a part of the transfer diagram, so the definition is ambiguous. According to the IEC 60748–4 the terms gain error and offset error should be used only *"for converters that have arrangement for an external adjustment of offset and gain errors"* while the terms zero-scale error and full-scale error should be used in all other cases. The IEC Standard defines offset error and gain error for adjustable DAC introducing a gain point and an offset point. The first one is "*a point in the transfer diagram corresponding to the step value of the step for which the offset error is specified, and by reference to which the offset adjustment must be performed*". The gain point is "*a point in the transfer diagram corresponding to the step value of the step for which the gain error is specified, and in reference to which the gain adjustment is performed*". The gain error is defined as "*the difference between the actual step value and the nominal step value in the transfer diagram at the specified gain point, after the offset error has been adjusted to zero*". The offset error is defined as "*the difference between the actual step value and the nominal step value at the offset point*".

These definitions seem to consider the possibility to determine offset and gain errors referring to any point of the transfer diagram. An offset error should affect all codes by the same amount, but its value strictly depends on the chosen offset point because "*the difference between the actual and the nominal step value*" can include linearity errors. So, clearly defining an offset point is essential to achieve an unambiguous offset error definition. In the same way, gain error has different values for different *gain points* (smaller values nearest the first codes). So a univocal gain error definition is necessary too.

Dealing with ADCs, the IEEE Std. 1241 [6] reports two different definitions for gain and offset: *independently based* and *terminal based*. In the first instance gain and offset are "*the values by which the input values are multiplied and then to which the input values are added, respectively, to minimize the mean squared deviation from the output values*". In the second instance gain and offset are "*the values by which the input values are multiplied and then to which the input values are added, respectively, to cause the deviations from the output values to be zero at the terminal points, that is, at the first and last codes*".

IEEE Std. 1241 *independently based* and *terminal based* gain and offset are perfectly equivalent to the *best-fit* and the *endpoint* method respectively. The IEEE 1658 gain and offset definitions follow the IEEE 1241 approach with suitable modifications for the DAC case, stating that *"Gain and offset are the values by which the ideal output values are multiplied and then to which the ideal output values are added, respectively, to cause the output values to satisfy specified conditions. Two conditions are commonly used: "independently based" where the sum of the squared deviations from the ideal output values is minimized, and "terminal-based" where the ideal outputs are matched at the first and the last code"*.

Reference [8] states that in the case of bipolar DAC it is common to specify the bipolar zero error (or zero error). This parameter is measured by applying the midscale code to the DAC and measuring its output. If no gain error exists (i.e., slope error), the bipolar zero error is the same as the offset error.

Several DAC manufacturers relate offset error to the digital input zero and define gain error as the difference between the output voltage (or current) with full-scale input code and the ideal voltage (or current) that should exist with a full-scale input code, specifying that the offset has to be already removed [9]. However, definitions strongly dependent on the input code can't be used for all DACs (they do not work for complementary coding DAC). The definition for gain error [9] considers the error at the last code without taking into account the other ones.

From what is quoted above, some conclusions can be drawn. Gain and offset definitions are strictly related to each other and to the definition of the transfer characteristic. These relations should lead to joint parameter definitions. Moreover, the definitions have to be independent from the input code used and valid for unipolar and bipolar DACs. The IEEE 1658 offset and gain definitions seem to satisfy these requirements but the definitions should be clarified by joining them with the DAC transfer function equations. Therefore, these could be used for the corresponding DAC parameters with additional comments. The common use of the bipolar zero error in the case of bipolar DACs has to be taken into account, paying attention to the considered ideal value. Many DACs are commonly used in application requiring a single power supply, in this case the ideal value corresponding to the midscale DAC code should be nonzero [10].

Offset can be expressed in least significant bit (LSB), ampere, volt, %FSR (full-scale range) and %FS. The gain error can be both the difference between the real and the ideal gain of the DAC (dimensionless) and also defined as a percent:

$$gain\_error = \left(\frac{G_{actual}}{G_{ideal}} - 1\right) \cdot 100\,\% \tag{12.2}$$

12.3.1.2 Differential and Integral NonLinearity

The Differential NonLinearity (DNL) and the Integral NonLinearity (INL) represent DAC nonlinearity errors. These parameters should be evaluated once gain and offset errors have been cancelled by trimming (if possible), or compensated for by mathematical operations, so that they can be distinguished from linear errors [11]. DNL is computed by the difference between the analogue output values corresponding to two successive input codes relative to one LSB.

IEC 60748–4 defines DNL as "*the difference between the actual step height and the ideal value (1 LSB)*", specifying also that a DNL greater then 1 LSB can lead to non monotonicity of a DAC.

IEEE 1658, again coherently with IEEE 1241, defines DNL as "the difference, after correcting for static gain, between the outputs corresponding to two

consecutive input codes minus the difference between two consecutive ideal output values (Q), divided by Q. When given as one number without specification, it is the absolute maximum differential nonlinearity over the entire range".

The DNL formula as expressed in [5, 12] requires the static gain correction only, as it is independent from the offset error. From the previous definitions it's clear that DNL value depends also on the definition of the average LSB size, too. To highlight this matter [11] introduces three types of DNL calculation methods: *best-fit*, *endpoint* and *absolute*. *Best-fit DNL* method uses the *best-fit* line's slope to calculate the average LSB size; *endpoint DNL* method calculates DNL by dividing the full-scale range by the number of transitions; *absolute DNL* method uses the ideal LSB size derived from the ideal maximum and minimum full-scale values. The order of methods from most relaxed to most demanding is *best-fit*, *endpoint* and *absolute* [11]. The three methods result in nearly identical results (Fig. 12.2), as long as the DAC doesn't exhibit grotesque gain or linearity errors so the choice of method is actually not so important for DNL as for INL calculation.

Most datasheets define DNL as the difference between the measured output voltage or current difference between two adjacent codes without specifying if gain and offset correction has been carried out. This is important information despite the small difference seen in Fig. 12.2, because non-linearity errors should be evaluated once gain and offset errors have been evaluated. The definition reported in IEEE Std. 1658 could be adopted by adding that *"DNL is expressed in LSB"* to take into account the DNL dependence from the LSB definition.

While DNL is a measurement of the uniformity in the voltage or current DAC step from one code to the next, INL is a measure of accumulated errors in the step sizes. INL is obtained by comparing the actual DAC characteristic and a reference DAC line, so the value of this parameter is strongly dependent on the line chosen as reference.

IEC 60748 proposes two different INL definitions valid for linear and adjustable DAC. The *endpoint* linearity error is "*the difference between the actual and the*

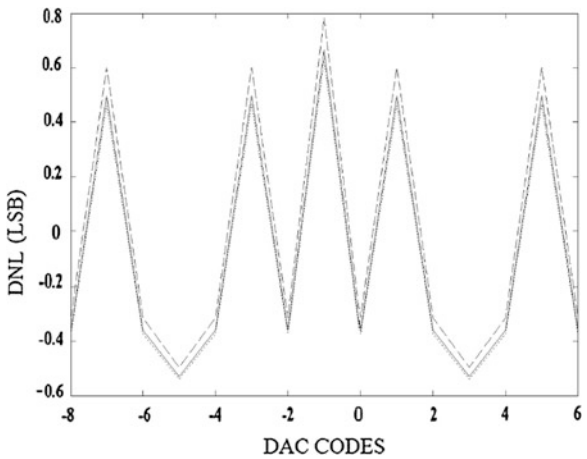

Fig. 12.2 Best-fit DNL (*dotted line*), endpoint DNL (*solid line*), and absolute DNL (*dashed line*) of a 4-bit two's complement DAC [9]

nominal step value, after offset and gain errors have been adjusted to zero". The *best-straight-line* linearity error is "*the difference between the actual and nominal step value, after offset and gain error have been adjusted to minimize the extreme value of this difference (either positive or negative)*".

IEEE 1241 defines INL as "*the difference between the ideal and measured code transition levels after correcting for static gain and offset. Integral nonlinearity is usually expressed as a percentage of full-scale or in units of LSBs. It will be independently based or terminal-based depending on how static gain and offset are defined. When the integral nonlinearity is given as one number without code specification, it is the maximum absolute value integral nonlinearity of the entire range*".

IEEE 1658 INL definition is the same as 1241, but the parameter formula has been modified according to the DAC case.

It is worth noting that the INL value depends not only on the considered reference line. In fact, since it is calculated after gain and offset error correction, the definition of gain and offset parameters could lead to different INL values. Manufacturer's datasheets adopt both INL definitions, although the *endpoint* is the most adopted, often without specifying if the gain and offset errors have been deleted or not [9]. Sometimes understanding the adopted INL definition isn't possible because it is not reported or it is referred to a generic "best straight line" [9].

As above quoted, gain, offset and INL definitions are strictly related and need to be clearly defined. A good definition for INL, is that proposed by IEEE 1658, because it specifies that INL must be calculated after correcting gain and offset and that its value is dependent on how these two parameters have been defined. This definition also agrees with the gain and offset definition previously proposed in this subsection.

12.3.2 Dynamic DAC Parameters

DAC dynamic characterization tracks the response of the component to steps, impulses or sinusoidal stimuli in the time or frequency domain. Several dynamic parameters can be measured by digitizing the DAC output responses. Next subsections analyze a couple of the most used DAC dynamic parameters starting from the time domain with the settling time and then moving to the frequency domain with Spurious Free Dynamic Range (SFDR).

12.3.2.1 Settling Time

In DACs the settling time gives information about the time required by the converter to produce a steady-state output value after a change in the input code.

DAC settling time has four distinct components (Fig. 12.3).

The *delay* (or *dead*) *time* is usually very small relative the total settling time interval and during this interval there is no output change. During *slew time*, the output amplifier moves at its highest possible speed towards the final value. The *recovery time*, describes when the DAC is recovering from its fast slew and may

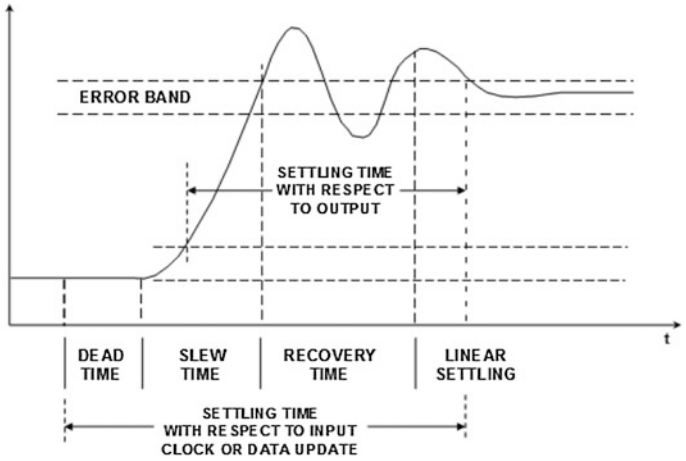

Fig. 12.3 DAC settling time components [8]

overshoot, and the *linear settling time*, describes when the DAC output approaches its final value in an exponential or near-exponential manner [8].

Several different settling time definitions can be found, mostly depending on the chosen starting and ending instants that can include or not the delay time or only a part of the slew time, or the considered error band amplitude. The most traditional definition is "*the amount of time required for the output to settle with the specified error band measured with respect to the 50 % point of either the data strobe to the DAC (if it has a parallel register driving the DAC switches) or the time when the input data to the switches changes (if there is no internal register)*" [8]. Another definition considers the settling time with respect to the instant when the output leaves the initial error band excluding the dead time from the measurement (Fig. 12.3) [8].

IEC Standard 60748–4 reports three definitions: *digital*, *reference, and steadystate ramp settling time* taking into account also the reference voltage variations in multiplying DACs. The former is included in the digital characteristics of a linear or multiplying DAC considering as starting instant a "*change at the digital input, the reference voltage being constant*". The latter two are, instead, included in the reference signal characteristics of a multiplying DAC, referring to a "*change of the reference voltage, the digital input being constant*".

The same three definitions quoted above are reported in the JEDEC Standard No.99 A.01, together with the definition of *analog settling time* as "*the time interval between the instant when the analog output passes a specified value and the instant when the analog output enters for the last time a specified error band about its final value"*.

The *final value* is not really determined in many cases. Long tails due to thermal or dielectric absorption effects can led to long term settling phenomena.

The more recently released IEEE Std. 181 [13] states that the word time refers exclusively to an instant and not an interval. So the term settling time, although widely used, is deprecated because it is ambiguous and confusing. So, IEEE Std. 181 defines the transition settling duration also considering as a deprecated term also the word "mesial" which is replaced with the 50 % reference level.

All the definitions discussed until now do not give information about which code has to be chosen to produce the initial and final output levels. The suggested approach consists in determining and measuring the codes representing the worst-case transitions. Reference [8] states that typical prescribed changes for the settling time measurements are "*full-scale, 1 MSB and 1 LSB at a major carry*".

Concerning the error band, it could be expressed as a percentage "*of the full-scale range, of the final voltage or of a fixed voltage*" [10] or in terms of an LSB [8]. Clearly for a specified DAC the greater the error band, the shorter the settling time.

Concerning the manufacturers' documentation several different settling time definitions are provided, too. Also in this case an agreement concerning the starting instant from which to compute the settling time is missing. Often information about the error band amplitude or the DAC transition considered is provided, but the final value determination and reference voltage variations in multiplying DACs are not taken into account.

From the presented state of art of settling time definitions some conclusions can be drawn. Starting from the two parameter definitions with respect to the input and to the output, in the former case the settling time measurement is clearly more complete since it includes all its components (dead time, slew time, recovery time and linear settling). However, this kind of definition involves considering two different signals (at the input and at the output of the DAC) requiring a more complex test bench, (an instrument with at least two channels is needed). On the other hand, defining the settling time with respect to the output, involves the analysis of only the DAC output signal and consequently a more immediate measurement. However, since it is often difficult to identify the transition starting instant, setting an output amplitude threshold is necessary. The 50 % point stated in [12] and [13] as a triggering threshold is fairly unambiguous, and is not affected by noise, such as clock or trigger transients, which can mislead the observer into believing that the transition has started. Another problem is the determination of the final value to be considered in the settling time computation. Because of the presence of noise, and specifically 1/f noise, the final value is not really determined in many cases. IEEE Std. 1241 suggests the 1 s time interval as a good compromise as after this time the transition can be measured statically with instruments like digital voltmeters. When waiting for 1 s is not allowable, the short settling time definition [12] has to be considered. The problem concerning the ambiguity of the word time both to refer to an instant and an interval, could be addressed with the simple addition of the word "interval" after settling time preserving the traditional parameter name taking into account its currently wide use.

All these considerations led to the following proposed definitions. In the case of settling time with respect to the input two different definitions are suggested:

The settling time interval is the time interval measured from the instant when the digital input changes and the instant at which the step response enters and subsequently remains within a specified error band around the final value. The final value (unless otherwise specified) is defined to occur 1 s after the beginning of the step. Unless otherwise specified the worst-case transition has to be considered.

The short-term settling time interval differs from the settling time interval for the final value determination, being *defined to occur at a specified time less than 1 s after the beginning of the step* [14].

In the case of settling time, with respect to the output, two definitions are proposed:

The output settling time interval is the time at which the step response enters and subsequently remains within a specified error band around the final value, measured from 50 % point of the response. The final value (unless otherwise specified) is defined to occur 1 s after the beginning of the step.

The output short-term settling time interval has the same definition of the output settling time except for the final value determination, since, in this case, it *is defined to occur at a specified time less than 1 s after the beginning of the step* [14].

These definitions require the specification of the error band, the time at which the final value is defined to occur and the transitions considered as the worst case. Four is the minimum number of definitions to take into account the settling time measurements including all its components (*settling time interval)* and or not (*output settling time interval)* as well as applications for which it is not allowable waiting for 1 s to determine the final value (*short-term settling time interval* and *short-term output settling time interval*) [14].

In case of multiplying DACs, two other definitions are needed: the *reference settling time interval* and the *settling time to steady-state ramp* as suggested by IEC Std. 607478. The former is "*the time interval between the time when a specified step change of the reference voltage occurs and the instant when the analog output enters and subsequently remains within a specified error band around the final value*". The latter is "*the time interval between the instant a ramp in the reference voltage starts and the instant when the analog output value and subsequently remains within a specified error band about the final ramp in the output*"[14].

IEEE 1658 has partially adopted these definitions by adding the word "interval" to the parameter name and considering only the settling time and short-term settling time definitions with respect to the output, as follows:

the settling time interval: the time at which the step response enters and subsequently remains within a specified error band around the final value, measured from the 50 % reference level instant of the response. The final value is defined to occur 1 s after the beginning of the step unless otherwise specified.

the short-term settling time interval that is equivalent to the settling time interval *except for the final value determination, since it is defined to occur at a specified time less than or equal to one second after the beginning of the step.*

12.3.2.2 Spurious Free Dynamic Range

Spurious Free Dynamic Range (SFDR) is the usable DAC dynamic range before spurious noise interferes or distorts the fundamental signal [15]. It is the difference between the fundamental and the highest spur power over a frequency band of interest. There are different definitions for SFDR, mainly depending on the exclusion of harmonics and on the definition of a frequency window around the fundamental in the computation of this parameter (Fig. 12.4) [16].

Considering or not the harmonically related distortion components as spurious is subject to debate. A spur is defined in [10] as "*any non signal component that is confined to a single frequency*", and can be caused by "*harmonic and intermodulation distortion, clock feedthrough, Sigma-Delta converter self-tones, stray oscillations, or any of dozens of other undesirable processes*". The worst spur, considered in the SFDR computation is "*the largest spectral component excluding the input signal and DC*" [17], "*the highest peak of any of the harmonic or intermodulation distortion products*" [18], "*possibly harmonic component*" [11] or "*usually but not necessarily always a harmonic of the fundamental*" [8]., Hendriks [19] states that "*the spur does not have to be harmonically related to the fundamental*".

Reference [20] explains the exclusion of harmonics in the SFDR computation stating that "*since harmonic distortion typically exceeds noise in the DAC spectrum, little information about the characteristics of the noise floor are obtained*".

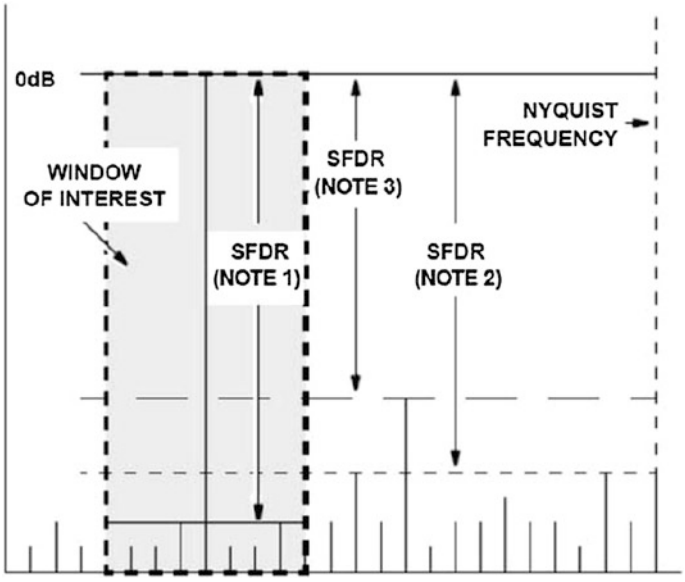

Fig. 12.4 SFDR computation methods [20]. *NOTE 1*: SFDR defined in a narrow band. *NOTE 2*: SFDR to Nyquist without harmonics. *NOTE 3*: SFDR to Nyquist including harmonics

But the inclusion of harmonics gives important information for example in high frequency telecommunications DACs, because "*an offending spur at an odd harmonic is likely related to amplitude distortion and at even harmonics is likely related to phase distortion*" [7]. Moreover, both spectral spurious and harmonics restrict the dynamic range. The frequency band of interest over which the SFDR is specified does not always coincide with the full Nyquist band. Many definitions do not set the frequency band to be considered for the SFDR only mentioning "*a specified bandwidth*" [15]. The reason for this practice, reported in [19], is that "*SFDR is sometimes specified over a narrow bandwidth which typically excludes the worst spur falling within the Nyquist zone*", assuming the user "*will operate over a narrow frequency band and will filter out any larger out of band spurs*", as for example in the case of frequency synthesizers employing direct digital synthesis, for maintaining low phase noise [19].

Reference [15] takes into account the designers' point of view stating that "*by picking an arbitrary window size, the 2nd or 3rd harmonic are often not included in the measurement. Because many systems designers intend to use a narrow band pass filter around the fundamental signal, they are more interested in the spectral performance within a band that the filter will pass. However, having full knowledge of a DAC spectral performance is essential to the selection of an appropriate band pass filter to remove the harmonics*". Therefore, the spurs and spectral performance up to Nyquist frequency is also important for designers.

SFDR over a narrow bandwidth takes into account the high frequency spurs generated by glitch impulses that fold back in band, reside close to the fundamental, cannot be filtered and dominate the noise within that range of frequencies. However, "*unless the user filters the output signal in a similar fashion to that being used to test the converter, the effect of the remainder of the noise floor on the application is unknown*" [20].

In any case, different SFDR values can arise depending on the method used for its measurement. SFDR is generally a function of the amplitude and the frequency of the input sine wave and the sample frequency. For signal amplitudes near full-scale, one of the first input harmonics usually determines the highest spectral spur. As the signal falls several dB below full-scale, other spurs which are not harmonics of the input signal can become dominant [8].

SFDR is usually measured with respect to the carrier frequency amplitude in dBc, (degrading as signal level is lowered) or with respect to the DAC full-scale range, in dBFS, (always relating back to converter full-scale).

Taking into account all these considerations, the proposed definition for SFDR is the following: "*For a pure sinewave input of specified amplitude and frequency, SFDR is the ratio of the amplitude of the DAC output averaged spectral component at the input frequency to the amplitude of the largest unwanted spectral component observed over a specified frequency band. SFDR is expressed in dBc or in dBFS*".

This definition requires the specification of amplitude and frequency of the input sine wave as well as of the frequency band considered. Reporting in the definition unwanted spectral component allows the choice of considering the

largest harmonic spectral component or not in the SFDR without having to define what the term "*spurious*" is intended to mean. Harmonics can be excluded by a suitable choice of frequency band over which the SFDR is computed [16].

The IEEE 1658 has partially adopted the proposed definition stating that SFDR "*specifies the available signal range as the difference in magnitude between the amplitude of the fundamental and the amplitude of the largest spurious component in the frequency band of interest*", and adding a note stating that "*some industry spec sheets include harmonics along with spurious components in their specification of SFDR. It is important to specify for any SFDR spec whether or not the harmonics are included*".

12.4 Advanced Test Methods for Assessing DAC Static Parameters

There are many methods to carry out the DAC static characterization. The manufacturers generally perform production testing on DACs using specialized automatic test equipment [8] that feeds the DAC with all possible digital input codes and relies on accurate digital voltmeters for measuring all the possible DAC output values.

As a result, the test execution time is an exponential function of the DAC number of bits. If the DAC has resolution N, and the time T_M is necessary to evaluate the analog output corresponding to each DAC input code, the time to obtain the whole static characteristic is equal to $2^N T_M$.

For high resolution DACs the test time becomes significant, leading to an economic interest in its reduction.

The test time reduction could be achieved by using waveform digitizers instead of high accuracy voltmeters. However this approach leads to an additional problem when testing high resolution and high linearity DACs: the digitizer should have higher resolution and linearity than the Device Under Test (DUT). There are significant technology problems in realizing ADCs with such characteristics.

To overcome these drawbacks, some advanced test strategies have been proposed in the literature. Each strategy has been implemented in several procedures to execute the DAC test. The next subsections are devoted to describe such test strategies and the corresponding experimental procedures.

The advanced test strategies can be divided in two main groups. The former includes the strategies to reduce the execution time of the DAC tests. The latter includes the strategies to reduce the resolution and linearity requirements of the digitizer.

Strategies belonging to the former group focus on the evaluation of the analog output corresponding only to suitable subsets of the 2^N possible input codes [21–27] instead of looking at all of them. These subsets are usually referred to as test vectors

and they are defined considering that, in the different DAC architectures, each elementary part has a different influence on the actual output voltages. In this way, only the input codes that have a significant effect on the DAC characteristic nonideality have to be used.

In order to find the most efficient test vector it is necessary to define appropriate models able to describe the influence of each part of the DAC.

Strategies belonging to the latter group focus on:

- analogue comparison of the DAC output signal to reference voltages [28–31]. The results of the comparison are analysed to reconstruct the DAC output signal or to evaluate the difference between the real and theoretical trend.
- inference of the nonlinearity from a pulse counter [32–34]. The DAC output signal is not directly acquired, but it is included in an appropriate structure that permits evaluating the DAC output values or the nonideality.
- amplification of the DAC output by variable gain amplifier [35]. Different DAC input codes are translated to the same amplitude level and compared to a single reference value.

Based on each strategy, suitable test procedures have been developed.

12.4.1 Procedures Based on Strategies to Reduce the Execution Time

Several papers [21–27] have been presented aiming to determine the optimal test vectors, however, the solutions depend on specific DAC schemes.

Many DACs are designed by using an architecture in which a series of binary weighted resistors or capacitors is used to convert the bits of the converter code to binary weighted currents or voltages. These currents or voltages are summed together to produce the DAC output. For instance, a binary weighted DAC output can be obtained as a sum of binary weighted voltage or current values, $W_0, W_1, \cdots W_n$, multiplied by the individual bits of the DAC input code, $D = [B_0, B_1, \cdots B_n]$. In this case the DAC ideal output is equal to:

$$V_{id}(D) = B_0{}^*W_0 + B_1{}^*W_1 + \cdots + B_n{}^*W_n + DC\,Base \tag{12.3}$$

where $W_1 = 2{}^*W_0$, $W_2 = 2{}^*W_1 \ldots W_n = 2{}^*W_{n-1}$ and *DC Base* is the DAC output value corresponding to $D = 0$. If this theoretical model of the DAC is accurate enough, the DACs transfer characteristic could be obtained by measuring the W_i. This test is called "major carrier". However, the W_i levels are widely different in magnitude and are difficult to be realized and measured accurately [21].

In order to overcome the technology problem, the manufacturers usually design high resolution DACs with segmented architectures including binary weighted parts for the least significant bits (LSB) and equally weighted parts for the most significant bits (MSB).

In [21] a modified segmented method is proposed. The MSB part is measured by all-code testing while the LSB part is performed by the major carrier method. This method requires only 36 measurements for testing a 10-bit current-steering DAC.

The paper [36] describes a test procedure for DAC with segmented current-steering architecture modified to switch each current source individually. The current-steering DAC is based on the sum of the currents from reference binary-weighted current sources according to the digital inputs. The segmented architecture combines binary-weighted and unary-weighted sources.

The procedure uses a loopback composed by a gain control and offset control blocks. The gain control block is a resistor network that scales the DAC output signal. The offset control block adds dc offset to the DAC output signal. The procedure utilizes the output voltage measure produced by each current source. The measures are provided first by selecting a scale-up of the DAC output by using a known factor and iteratively connecting one of the current source of the DAC architecture corresponding to the less significant bits. For the DAC current sources corresponding to the unary-weighted sources the scale factor is changed to obtain that the full-scale of the digitizer is equal to the full-scale of the DAC. For these sources a measure is obtained for each of the most significant bits. Finally, the DAC transfer characteristic is calculated as combination of the binary-weighted and unary-weighted sources measurements after the scale compensation. Experimental validation confirms the possibility to test a 10-bits DAC with a 10-bits ADC.

In [22] a different approach has been proposed for the DAC testing. Figure 12.5 shows the corresponding model scheme. It consists of M different sub-DAC sections of N_m bits each, so that the overall converter resolution N is equal to the sum of all the section resolutions N_m. The input word D is divided in independent M sub-words $D_m = (d_1 \ldots d_{Nm})$, with $m = 1, \ldots, M$. Each sub-word feeds a different sub-DAC so that each bit d_i drives a dedicated output circuit element. In the first B sub-DACs the circuit elements are binary-weighted sources. The other M-B sub-DACs include equally weighted sources after a thermometer decoding. The bits $(s_{B+1} \ldots s_2{}^N{}_{B+1})$ control the summation of the equally weighted sources.

Defining W_m as the binary-weight that expresses the significance level of the mth DAC section as:

$$W_m = \prod_{k=0}^{m-1} 2^{N_k} \tag{12.4}$$

The ideal transfer characteristic is:

$$V_{id}(D) = V_{ref}\left\{\sum_{m=1}^{B}\left[W_m \sum_{i=1}^{N_m}\left(\frac{d_{mi}}{2^i}\right)\right] + \sum_{m=B+1}^{M}\left[W_m \sum_{j=1}^{2^{N_m}-1}\left(\frac{s_{mj}}{2^j}\right)\right]\right\} \tag{12.5}$$

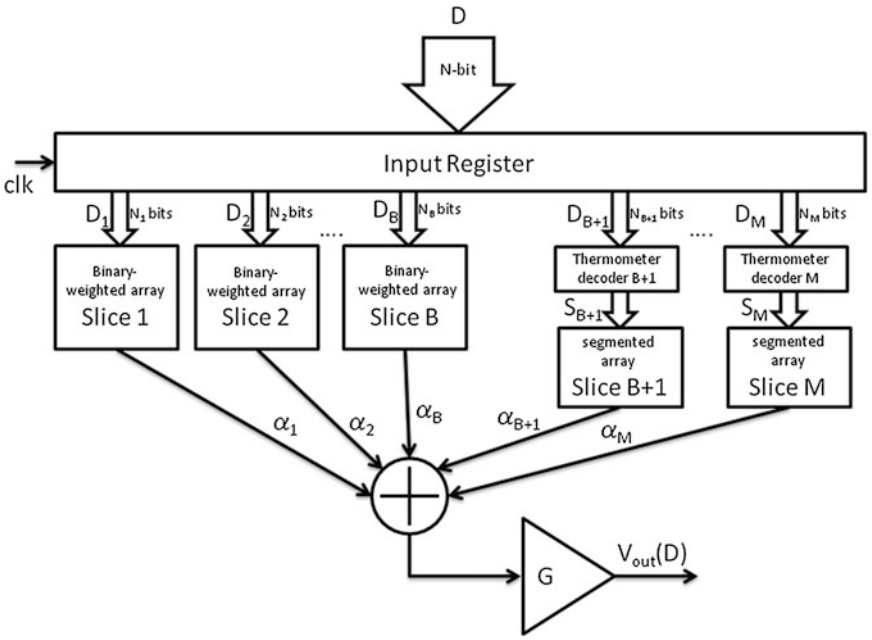

Fig. 12.5 Block diagram of the DAC model used in the test method [22]

This expression doesn't take into consideration the influence of the nonidealities in the device that cause offset O, gain errors G and non-linearity. So the DAC output is:

$$V_{out}(D) = GV_{id}(D) + O + INL(D) \tag{12.6}$$

By using this model, it is possible to establish a test vector for the DAC under test. The procedure starts with the estimation of the offset and gain errors of the DAC under test. To obtain these results the DAC output voltage corresponding to the code *0* and 2^N–1 are measured and used in the following relations:

$$O = V_{out}(0), \tag{12.7}$$

$$G = \frac{V_{out}(2^N - 1) - O}{V_{ref}(1 - 2^{-N}) - 1}, \tag{12.8}$$

where V_{ref} is the reference voltage coming from the power supply module. The test vectors are determined by taking into consideration the digital word D with only one sub-word D_m different from zero.

By means of (12.6) the *INL* contribution is evaluated for each code from the measured DAC output voltage $V_{out}(D)$ after the offset and gain error compensation [22].

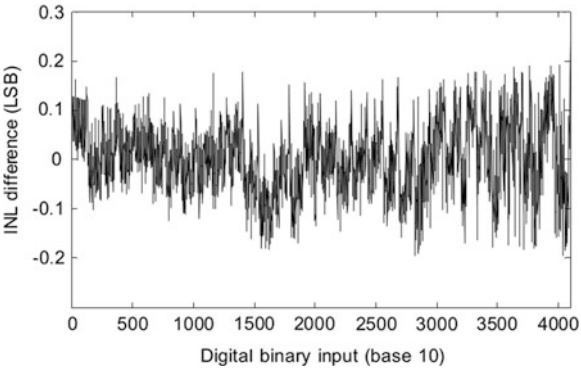

Fig. 12.6 Difference between the measured INL and that obtained with the model proposed in [22] by using 55 test vectors

In order to experimentally validate the proposed method the *INL* of the Analog Devices AD9762 12-bit DAC is measured by using the all-code method. Figure 12.6 shows the difference between the all-code *INL* and that obtained with the model proposed in [22] by using 55 test vectors. From Fig. 12.6 it can be seen that the reconstruction error is lower than 0.2 LSB.

Other procedures have been proposed to minimize the number of input codes aimed at testing both specific device families and basic DAC parametric tests using the Linear Error Mechanism Model Algorithm (LEMMA) test point selection strategy is proposed in [26, 27]. In [27] the authors show that this leads, in the particular case of the AD5320 DAC, to a cut in the number of test points from 124 to 66.

12.4.2 Procedures Based on Strategies to Improve the Test Resolution

12.4.2.1 Analogue Comparison of the DAC Output Values with Reference Voltages

In order to remove strict specifications about resolution and linearity of the digitizer it has been proposed to compare the DAC outputs with a reference signal [28].

The reference signal must be similar to a theoretical sine wave whose frequency matches the DAC-generated one. In this way, it is possible to acquire the DAC output signal with a resolution greater than the digitizer amplitude resolution and to reconstruct the DAC transfer characteristic.

The reconstruction method proposed in [28] uses a reference signal obtained by suitably filtering the DAC output.

A high performance differential amplifier is used to implement the difference between the two signals. The residue has amplitude lower than the DAC absolute

maximum error, however it can be amplified and acquired by means of a low-resolution digitizer to obtain the desired enhancement of the measurement system resolution. In fact, this last is acquired using a digitizer with full-scale voltage corresponding to a few DAC LSBs, in order to provide the needed resolution. In order to reconstruct the DAC output, the DAC output signal is simultaneously acquired with the differential amplifier output signal. The obtained samples are used to estimate the frequency, amplitude and phase of the filtered DAC output. The reconstructed DAC output is equal to the sum of the theoretical signal and the differential amplifier output.

The attenuation and the phase displacement introduced by the filter generate a residual sine wave in the difference signal with small amplitude, and frequency equal to the DAC output frequency. To overcome this problem a modified version of the test scheme is used, Fig. 12.7, where the gain mismatch is compensated by means of a potentiometer reducing the amplitude of the DAC output to the same value as the filtered version. The phase displacement is negligible if finely tuned at the DAC signal frequency.

The experimental tests on the method have been carried out by testing a 12-bit DAC, with LSB equal to 1.95 mV, by means of an 8-bit ADC with 10 mV of full-scale. The results presented in Fig. 12.8 highlight how the DAC error signal exhibits amplitude within the range –1.3–0.8 mV that is comparable with the DAC LSB. This result is remarkably near the specifications of the DAC manufacturer.

In [29, 30] the problem of testing the DAC by using high-resolution acquisition is translated into high frequency–low-resolution acquisition. With this aim, an acquisition procedure is proposed based on the detection of the Zero Crossing Time Sequence (ZCTS) within the difference signal obtained by subtracting a reference signal from the DAC output one.

Figure 12.9 shows the block scheme of the test system. The DAC under test is forced to generate the sinusoidal signal $v_k(t)$, with amplitude V_{DAC} and frequency f_{DAC}. A sinusoidal signal $v_{st}(t)$, with amplitude V and frequency f, provided using a reference generator with higher resolution and linearity than the DUT is subtracted from $v_k(t)$. The reference signal is band-pass filtered to ensure adequate spectral purity compared to the DAC under test.

The difference signal $v_r(t) = v_k(t) - v_{st}(t)$ is oversampled by means of low resolution ADC#1 and stored in the PC.

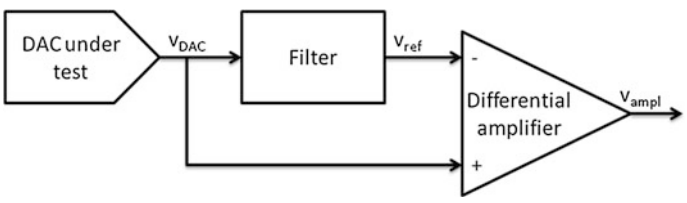

Fig. 12.7 Block scheme of the procedure for the DAC static characterization based on reference signal and high performance differential amplifier [28]

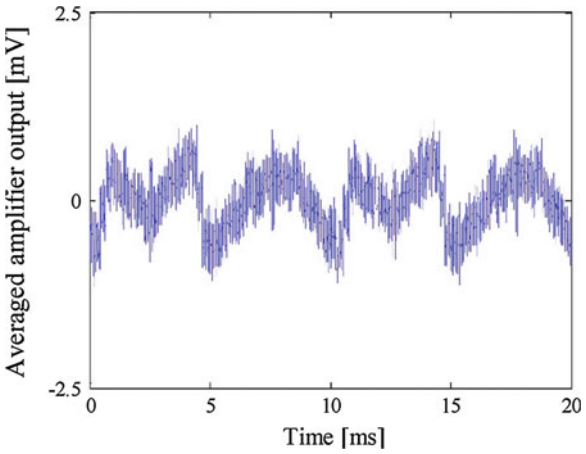

Fig. 12.8 Trend of the differential amplifier output in the case of 12-bit DAC and 8-bit ADC [28]

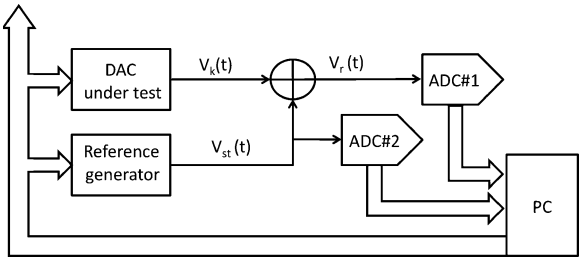

Fig. 12.9 Block scheme of the acquisition system of DAC output signal [29]

The DAC output voltage is obtained by using the ZCTS $t_1,\ldots,t_n$ detected on the difference signal. In each of the n elements of the ZCTS the value of the reference signal is inferred and, consequently, n corresponding values of the DAC output signal are determined.

The obtained ZCTS is non-uniformly distributed in the time domain, therefore, the reconstructed signal $v_k(t_m)$ is characterised by non-uniform sampling. By applying the Discrete Fourier Transform to the non-uniformly sampled signal, the resulting spectrum shows spectral lines not included into the original signal. This spectrum does not allow determining the dynamic characteristics of the effective input signal. To overcome such problem the reconstruction algorithm of the uniformly sampled spectrum from a non-uniformly sampled one, presented in [30], has been adopted.

ADC#2 digitizes the reference signal in order to evaluate amplitude, frequency and phase of $v_{st}(t)$, by the sine fit algorithm [12], and to reconstruct the output signal of DUT.

Experimental results presented in [29] confirm that a 12-bit DAC can be characterized by means of a 6-bit ADC and a 14-bit reference signal generator. The same DAC has been tested by a direct acquisition of the DAC output signal by

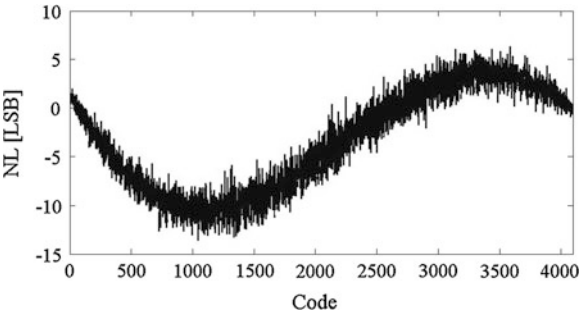

Fig. 12.10 Nonlinearity affecting the DAC transfer characteristic estimated by the procedure proposed in [29]

means of a high resolution digitizer and by applying the method proposed in [29] and the two estimated nonlinearities (Fig. 12.10) showed the same shape.

Paper [37] proposes the use of a wobbling signal to test high resolution DAC with a low resolution, high speed digitizer. The DAC under test generates a periodic ramp signal and adds this last to the output of the wobbling generator. The wobbling voltage is constant during each period of the DAC-generated ramp as shown in Fig. 12.11.

The output amplitude of the wobbling generator is scaled to cover 3 LSBs of the digitizer. The digitizer acquires several periods of this composite signal each one characterized by a different wobbling level. The digitizer output codes are then used to obtain a matrix where the rows mark the DAC output values and the columns mark the wobbling generator input code.

By using this matrix, the method first calculates the digitizer code bin widths for each code j as the maximum difference of the wobbling levels corresponding code j in the rows of the matrix. The results for all the rows for the same code *j* are averaged to reduce the noise effects. Then, the transition levels T_j of the digitizer are calculated as the cumulative sum of code bins width evaluated. Finally, the DAC output voltages corresponding to each digital input code *k* are calculated as the mean value of the difference between the digitizer transfer level and the wobbling level, corresponding to *k,* that causes a code transition of the digitizer.

Experimental tests confirm that a 10-bits DAC can be characterized by using a 6-bit ADC and 9-bit wobbling generator as described in [37]. The 10-bits DAC has been emulated by using a 16-bit DAC to generate 10-bit output values affected by a sinusoidal shape *INL*. Such pseudo-DAC has been tested by using the proposed

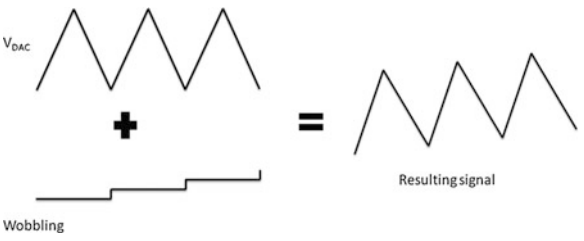

Fig. 12.11 Trend of the output voltage of the DAC, wobbling and resulting signals [37]

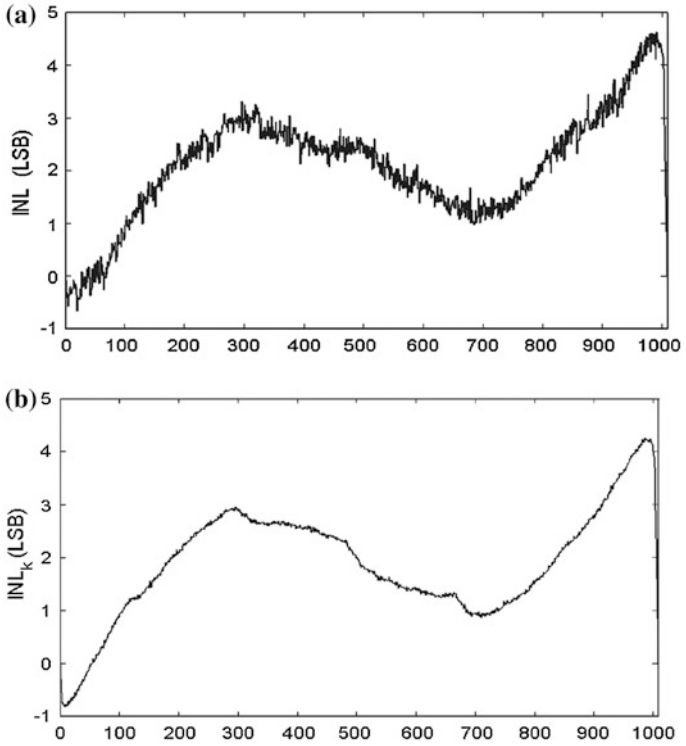

Fig. 12.12 Trend of measured INL using: **a** the method in [37] and **b** a high resolution digitizer [37]

method and a 14-bit resolution digitizer. The resulting INLs are reported in Figs. 12.12. As it can be seen that the two INLs are very close.

12.4.2.2 Inferring the Nonlinearity from a Pulse Counter

In [31], the time taken by a linear ramp to cross two consecutive output levels is considered as measure of the difference between two corresponding DAC outputs level.

In [32] the DAC under test is included in the feedback path of a ΣΔ modulator, as shown in Fig. 12.13. The input of the DAC is switched between two codes with same value but opposite signs C_j and C_k. The one bit digital output of the modulator feeds an up/down counter. For an ideal DAC with $C_k = -C_j$, the counter output goes to zero. For a non-ideal DAC the non ideality affecting the DAC static characteristic affects the counter results. In fact, using *N(j,k)* the mean value of the counter for each pair of codes C_j and C_k, it is possible to evaluate the input offset O and the INL associated to a code j (INL_j):

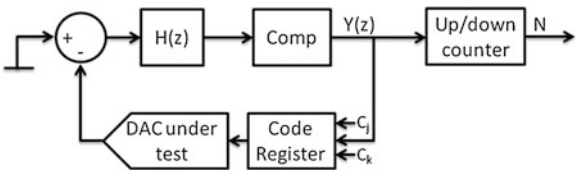

Fig. 12.13 Block diagram of the ΣΔ modulator DAC test scheme [32]

$$O = j \times N(j,k) \times LSB \tag{12.9}$$

$$INL_j = \left(j - k\frac{N(j,k)+1}{N(j,k)-1} - i\right)LSB \tag{12.10}$$

In [33] the DAC output voltage is used to control a Voltage Controlled Oscillator (VCO) obtaining the DAC output errors in terms of the frequency shift. However, this method requires that the used VCO has a linearity better than the DAC under test in the entire output range.

In [34], a modification of such scheme, which reduces the linearity requirements of the VCO is proposed. The tests are based on estimating the voltage step corresponding to adjacent codes. This is done using an on-chip offset-compensated sample-and-subtract module and a VCO. An up-down counter is used to measure the frequency. The reference clock is used to set the counting window.

In [38] the DAC output voltages, corresponding to each input code, are converted to time intervals. From the difference between the ideal and actual values of the time intervals the DAC static parameters can be estimated. Figure 12.14 shows the block diagram of the test setup. The main blocks are: a test pattern code generator to impose the input code of the DAC, a ramp generator that integrates the DAC output, a threshold detector that indicates when the ramp from the integrator is within the range defined by two reference voltages, a tick counter that evaluates the time interval between the beginning of the ramp signal and the threshold exceeding, and a control module that manages the test procedure.

Initially, the pattern generator sets the DAC input code, the ramp generator output is set to zero and the counter is reset. Afterwards, the ramp generator integrates the DAC constant output. When the threshold detector indicates that the ramp is equal to the low reference voltage V_{OL} the counting is started and stopped when the ramp is higher than the high reference voltage V_{OH}. The estimated time interval ΔT is inversely proportional to the ramp slope, which in turn is proportional to the DAC output. Therefore the DAC output is:

$$V_{DAC}(D) = \frac{V_{OH} - V_{OL}}{-k_{RC}\Delta T} \tag{12.11}$$

where k_{RC} is the integrator circuit constant.

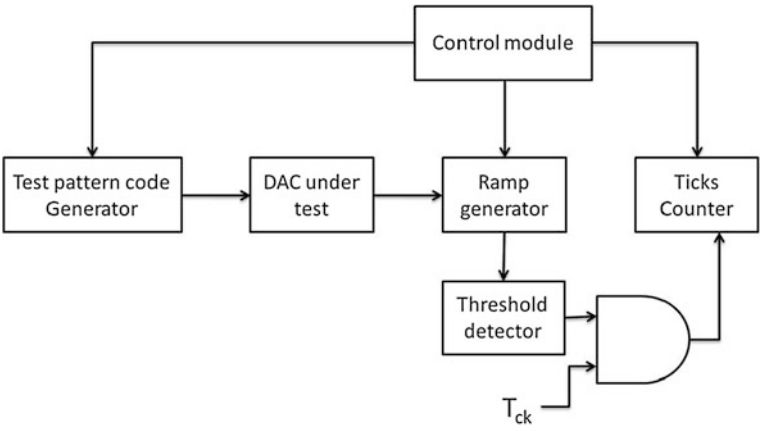

Fig. 12.14 Block diagram of the test setup [38]

12.4.2.3 Procedure Based on the Comparison with a Few Reference Voltages

The paper [35] proposes a procedure for the static characterization of the DACs based on a variable gain amplifier with low resolution for a Built-In Self-Test. The procedure starts by dividing all the possible input codes of the DAC into a predetermined number of segments. The DAC output voltages corresponding to different codes in the same segment are amplified to reach a specified number of reference voltages. In this way, it is not necessary to use many reference voltages, because the same reference voltage can be used to test all codes in each segment. The procedure responses are analyzed by comparing the actual outputs of the amplifier and the comparator reference voltages. By knowing gain, reference values and comparator outputs, it is possible to determine a range where the DAC output is included. By setting this range equal to $\pm 1/2$LSB, the DAC transfer characteristic is determined with accuracy equal to $\pm 1/2$LSB.

12.5 Conclusions

The considerable development that has characterized DACs in these last years has given a wider and deeper choice to the user. However such development has produced burdensome problems due to the lack of a unique approach to standard terminology and the difficulty of testing high resolution DACs.

With the aim of providing a basic knowledge to quantitatively assess the non-ideal behaviour of DACs in different operating conditions a suitable set of parameter definitions and test methods have been provided in the chapter. In particular, the most used figures of merit and a discussion about their meaning have been given, along with new proposals concerning their definitions.

Some advanced test methods proposed in the literature that overcome the problems arising from the increasing resolution and speed of the new generations of DACs, have been summarized.

First the problem of test time reduction has been dealt with by presenting procedures that exploit the characteristics of the current-steering and segmented current-steering architectures to reduce the number of acquisitions needed to reconstruct the DAC transfer characteristic.

Then, the problem of testing high resolution and linearity DACs with similar or lower resolution digitizer has been taken in account. Several proposals, coming from the literature, to overcome such problem have been presented based on different strategies. Several papers from literature present encouraging experimental results that show how, by accepting a result variability below 0.5 LSB, it is possible to test high resolution DACs also by means of low resolution ADCs.

References

1. IEC 60748–4: Semiconductor Devices—Integrated Circuits—Part 4: Interface integrated circuits—Sec. 2: Blank detail specification for linear analogue-to-digital converters, 2nd edn. (1997)
2. IEEE Standard 746: IEEE Standard for performance measurements of A/D and D/A converters for PCM television video circuits (1984)
3. JEDEC Standard 99, A.01: Terms, definitions, and letter symbols for microelectronic devices (2000)
4. EBU Technical Information I15: Testing for conformity with ITU-R recommendations BT.601 and BT.656 (1998)
5. IEEE Std. 1658: IEEE Standard for terminology and test methods for digital-to-analog converter devices (2012)
6. Tewksbury, S.K., Meyer, F.C., Rollenhagen, D.C., Schoenwetter, H.K., Souders, T.M.: Terminology related to the performance of S/H, A/D and D/A circuits. IEEE Trans. Circ. Syst. CAS-**25**(7), 419–426 (1978)
7. Jasper, B., Practical telecom DAC testing. http://www.testedgeinc.com
8. Kester, W.: The Data Conversion Handbook Analog Device Inc. Elsevier, Burlington (2005)
9. Balestrieri, E., Rapuano, S.: DAC consistent terminology: static parameter definitions. Measurement **40**(5), 500–508 (2007)
10. Burns, M., Roberts, G.W.: An Introduction to Mixed-Signal IC Test and Measurement. Oxford University Press, Oxford (2001)
11. Atmel, Data Converter Terminology, AN. http://www.atmel.com
12. IEEE Std. 1241 (2010) IEEE Standard for terminology an test methods for analog-to-digital converters
13. IEEE Std. 181 (2011) IEEE Std on transitions, pulses, and related waveforms
14. Balestrieri, E.: DAC time-domain specifications toward standardization. IEEE Trans. Instrum. Meas. **57**(7), 1290–1297 (2008)
15. Intersil, Measuring spurious free dynamic range in a D/A converter, Technical Brief. http://www.intersil.com
16. Balestrieri, E., Rapuano, S.: Defining DAC performance in the frequency domain. Measurement **40**(5), 463–472 (2007)
17. Baker, M.: Demystifying Mixed Signal Test Methods. Newnes, Amsterdam (2003)
18. Tweed, D.: Digital Processing in an Analog World, Circuit Cellar INK (1998)

19. Hendriks, P.: Specifying communication DACs. IEEE Spectr. **34**(7) (1997)
20. Intersil, Understanding the HI5721 D/A converter spectral specifications, AN, http://www.intersil.com
21. Ting, H., Chang, S., Huan, S.: A Design of Linearity Built-in Self-Test for Current-Steering DAC. J. Electron. Test. **27**(1), 85–94 (2011)
22. Macii, D.: A novel approach for testing and improving the static accuracy of high performance digital-to-analog converters. In: Proceedings of 8th International Workshop on ADC Modelling and Testing, pp. 197–200 2003
23. Fasang, P.: An optimal method for testing digital to analog converters. In: Proceedings of the 10th IEEE International ASIC Conference and Exhibit, Portland, pp. 42–46 1997
24. Vargha, B., Schoukens, J., Rolain, Y.: Using reduced-order models in D/A converter testing. In: Proceedings of IEEE IMTC, pp. 701–706 2002
25. Souders, T., Stenbakken G.: A comprehensive approach for modeling and testing analog and mixed-signal devices. In: Proceedings of IEEE ITC, pp. 169–176 1999
26. Wrixon, A., Kennedy, M.: A rigorous exposition of the LEMMA method for analog and mixed-signal testing. IEEE Trans. Instrum. Meas. **48**(5), 978–985 (1999)
27. Wegener, C., Muller, B., Straube, B., Kennedy, M.: Inter-weaving functional and parametric tests for the example of a digital to analog converter. In: Proceedings of the 12th ITG/GI/GMM-Workshop, Grassau/Chiemsee, Germany, 2000
28. Baccigalupi, A., D'Arco, M., Liccardo, A., Vadursi, M.: Test equipment for DAC's performance assessment: design and characterization. IEEE Trans. Instrum. Meas. **59**(5), 1027–1034 (2010)
29. Baccigalupi, A., Carnì, D., Grimaldi, D., Liccardo, A.: Characterization of arbitrary waveform generator by low resolution and oversampling signal acquisition. Measurement **45**, 2495–2507 (2012)
30. Carnì, D., Grimaldi, D.: Characterization of high resolution DAC by DFT and sine fitting. In: Proceedings of Instrumentation and Measurements Technology Conference, Singapore, pp. 1244–1249 2009
31. Huang, J., Ong, C., Cheng, K.: A BIST scheme for on-chip ADC and DAC testing. In: Proceedings of Design Automation and Test in Europe Conference and Exhibition, Paris, pp. 216–220 2000
32. Hassan, I., Arabi, K., Kaminska, B.: Testing digital to analog converters based on oscillation-test strategy using sigma–delta modulation. Proc. ICCD **98**, 40–46 (1998)
33. Chang, S., Lee, C., Chen, J.: BIST scheme for DAC testing. Electron. Lett. **38**(15), 776–777 (2002)
34. Sunil Rafeeque, K., Vasudevan, V.: A built-in-self-test scheme for digital to analog converters. In: Proceedings of VLSID, pp. 1027–1032 Jan 2004
35. Wen, Y., Lee, K.: BIST structure for DAC testing. Electron. Lett. **34**(12), 1173–1174 (1998)
36. Huang, X.L., Huang, J.L.: ADC/DAC loopback linearity testing by DAC output offsetting and scaling. IEEE Trans. Very Large Scale Integr. (VLSI) Syst. **19**(10), 1765–1774 (2011)
37. Le J., Haggag, H., Geiger, R.L., Degang, C.: Testing of precision DAC using low-resolution ADC with wobbling. IEEE Trans. Instrum. Meas. **57**(5), 940–946 (2008).
38. Hariharan, K., Gouthamraj, S., Subramaniam, B., Venkatesh Babu, S.R., Abhaikumar, V.: A novel method for testing digital to analog converter in static range. Am. J. Appl. Sci. **7**(8), 1157–1163 (2010)

Chapter 13
Uncertainty Analysis of Data Converters Testing Parameters

Andrea Zanobini, Lorenzo Ciani and Marcantonio Catelani

13.1 Summary

There has been a rapid increase in the speed and accuracy of data conversion systems, whose characteristics play a fundamental role in the performance of digital instruments as well as in the quality of measurement systems. Consequently, the test equipment required for checking is becoming increasingly expensive. Characterization and testing activities represent, in fact, a major factor of cost in integrated circuit (IC) manufacturing, circuits may cover nearly 50 % of the whole production budget. Apart from being a design challenge, testing of high-performance data converters has become a an important issue for the engineers. For this reason, in order to provide a guide for technicians in designing test methods and systems for Analog-to-Digital Converters (ADCs) and Digitizing Waveform Recorders (DWRs), two technical standards IEEE Std 1241 [1] and IEEE Std 1057 [2] have been developed throughout the years by the IEEE Instrumentation and Measurement Society TC-10 "Waveform generation, measurement and analysis." In addition, a Standard concerning Digital-to-Analog Converters (DACs) terminology and test methods is currently under development.

Of course, the removal of the ambiguities existing in data converters terminology and testing methods cannot provide useful compatibility information without specifying the parameter measurement uncertainty. The measurement uncertainty is a quantitative indication of its quality, allowing the comparison of results coming from different sources or from reference values given in specifications or standards. Therefore, a customer is able to compare correctly and choose the converter suitable to particular needs only if the uncertainty about the parameters of interest has been specified in the manufacturer's datasheet. Unfortunately, at the moment, the evaluation of the measurement uncertainty is not yet included in any data converters standard.

A. Zanobini (✉) · L. Ciani · M. Catelani
Department of Information Engineering, University of Florence, Florence, Italy
e-mail: andrea.zanobini@unifi.it

P. Carbone et al. (eds.), *Design, Modeling and Testing of Data Converters*, Signals and Communication Technology, DOI: 10.1007/978-3-642-39655-7_13,

In this chapter the uncertainty analysis of data converters testing parameters will be introduced and practical case studies, carried out on commercial generation and acquisition equipment, will be described.

For the test parameters taken into consideration, the uncertainty assessment and the corresponding confidence level is evaluated according to different statistical methodologies proposed by the "Guide to the expression of uncertainty in measurement" (GUM) [3]. Other different test methodologies are proposed in literature. In [4], for instance, a methodology to test high-speed A/D converters using low-frequency resources is discussed. However, often, the uncertainty assessment of the converters test parameters does not well explained in the approaches propose in literature. The aim of the proposed technique in this chapter is to obtain a suitable confidence interval estimation by using only a few measures; considering a minimization of the acquisition record length, the approach could be more attractive from the industrial point of view. To this end all the distributions that allow to solve statistical inference problems will be considered in the follows.

When a sample is taken from a discrete and finite distribution it is necessary:

1. to extract all the possible n-random measurements from a finite population of dimension N;
2. to compute the desired statistic related to the testing parameters under examination;
3. to achieve a table containing all the different values assumed from this statistic and the corresponding frequencies. The achieving of a sampling distribution is difficult with huge population and sometimes impossible if the population is infinite.

In general the main characteristics of a sampling distribution that allow to evaluate the measurement uncertainty are the mean, the variance and the shape.

13.2 Uncertainty Evaluation in the Measure of Data Converters Testing Parameters

The objective of a measurement, such as the testing of data converters, is to determine the value of the measurand, that is the specific quantity subject to measurement. In general, no measurement or test is perfect and the imperfections give rise to errors in the result. Consequently, the result of a measurement, as testing parameters, is only an approximation to the value of the measurand and is complete only when accompanied by a statement of the uncertainty of the approximation [5, 6]. The International Vocabulary of Basic and General Terms in Metrology defines uncertainty as "a parameter associated with the result of a measurement, that characterizes the dispersion of the values that could reasonably be attributed to the measurand" [7].

Obviously, it is important to take into consideration that testing parameters are strictly linked to their corresponding definitions. Such relations are connected to some physical states like temperature variation, reliability, electromagnetic coupling and other important phenomena. With these relations it is possible to elevate the quality level of the whole measurement process. It is important to reduce the uncertainty in the information and to enhance the detail of the measurand. The measure of the parameters above mentioned cannot be characterized by just one value.

First of all it is important to point out that the measurement process, say the measurement *activity*, cannot generate a rational number as result; in fact, some information connected to its validity and to the quality of the whole measurement process are necessary. Both these numbers should be referred to their measurement unit.

Now it should be clear that a measurement process should be able to transform an unknown measurand in a measurement result, with all the information necessary to state its quality level. To this aim, measurement instruments or sensors, samples and often also an acquisition data system with software would be necessary.

In this process the statistical inference plays a fundamental role but some questions have to be taken into account in the parameters estimation: expected value, variance and standard deviation by means of sample statistics. It is possible to obtain different punctual estimation for the same parameter. For example, to assess the expected value of a population it is possible to adopt also the sample median. However in the selection of the estimators it is fundamental to verify the properties of such estimators. These are the correctness, or unbiased, and the efficiency. An estimator is said to be unbiased if its bias, that is the difference between the expected value of the estimator and the true value of the statistical parameter being estimated, is equal to zero. For example the expected value of the mean of the sample distribution is an unbiased estimation of the population mean. It is important to remember that the unbiased estimation is not the only one; in fact, the sample median is also an unbiased estimation of the population mean.

For this reason it is necessary to emphasize another important property, called efficiency, to decide what is the better parameter estimator.

If two or more statistical parameters are both unbiased estimators, the estimator for which the variance of the sample distribution is lower is called a more efficient estimator.

It is possible to demonstrate that, between all the parameters that estimate the sample population, the sample mean is the more efficient. Also the sample variance is an unbiased and efficient estimator of the population variance.

In addition to the properties of the estimators other aspects, which are not taken into consideration here, are also important such as measurement procedures, calibrations with reference samples and international standards link the conformity of the process to requirements.

Concerning the uncertainty it is possible to identify two main contributes: the first is originated by random effects *(GUM—type A)*; the other one depends on systematical effects *(GUM—type B)* [3, 8]. In particular:

- The first contribute is essentially related to the stochastic variations of the influence quantities which will give different observation obtained in *repeatability* conditions. This means that the same person can collect these observations, in an independent way, with the same procedure, the same instrument, in the same conditions of use and in a restricted time interval.
- The second contribute is essentially related to those effects that are identical every time the measure is repeated and it depends, for example, on instrument accuracy and resolution, reference samples, and others.

If these two contributes are present, it is often possible to correct them and, eventually, to delete many of these effects.

The measurement, thus the data converters testing parameters, can be treated like a *random variable, M,* distributed within the measurement interval. This random variable determines the measurement process and so the quality of the test results.

It is important to introduce the *confidence level* to attribute to every single event, associated to M in $S = \{m_{\min} \leq M \leq m_{\max}\}$, the space domain of the results.

Obviously the maximum confidence level, equal to one, can be assigned when M belongs to S; vice versa the minimum confidence level, equal to zero, is assumed when the values of M do not belong to the space S of all the possible measurement results. Consequently, it is logical to assign a real positive number, in the range from zero to one, to the confidence level defined in a subinterval $\{ m_a \leq M \leq m_b \}$ of *S;* this number is named *probability*.

To each M is associated a probability distribution, or a function of random events which represents the probability that a measure belongs to one of the possible subintervals in which it is possible to divide the space S of all the possible measurement results. The probability distribution of M is all that is known about the measurement interval. Then, if a connection is established with the classic probability definition, the one that, according to *Laplace,* is referred to the relative frequency, it is possible to write

$$P\{m_a \leq M \leq m_b\} = \lim_{N \to \infty} \frac{N\{m_a \leq M \leq m_b\}}{N} \tag{13.1}$$

where $N\{m_a \leq M \leq m_b\}$ is the number of the events $\{m_a \leq M \leq m_b\}$, with respect to the total number N of the events, with $N \to \infty$.

Generally M is a continuous random variable, so it is necessary to define the *Probability Density Function* (PDF) $f_M(m)$ expressed as

$$f_M(m) = \lim_{\delta m \to 0} \frac{P\{m \leq M \leq m + \delta m\}}{\delta m} \tag{13.2}$$

where δm is greater than zero. It is also possible to consider the following equation

$$P\{M \in S\} = P\{m_{\min} \le M \le m_{\max}\} = \sum_{i=1}^{N} P\{m_i \le M \le m_i + \delta m\} = 1 \quad (13.3)$$

So, for the probability density function as well, Eq. (13.3) means that the area under the curve is equal to one as follows

$$\int_{m_{\min}}^{m_{\max}} f_M(m)\, dm = P\{m_{\min} \le M \le m_{\max}\} = 1 \quad (13.4)$$

It is often useful to introduce the Cumulative Distribution Function (CDF) of M, defined by

$$F_M(m) = P\{M \le m\} = \int_{m_{\min}}^{m} f_M(m)\, dm \quad (13.5)$$

The PDF is the derivative (when it exists) of $F_M(m)$, and

$$f_M(m) = \frac{dF_M(m)}{dm} \quad (13.6)$$

The more immediate and important consequence of the Eq. (13.5) is

$$P\{m_a \le M \le m_b\} = F_M(m_b) - F_M(m_a) = \int_{m_a}^{m_b} f_M(m)\, dm \quad (13.7)$$

The CDF, as is showed in Fig. 13.1, is always greater than zero, monotone non-decreasing and characterized by the properties

$$F_M(+\infty) = 1, \quad F_M(-\infty) = 0 \quad (13.8)$$

A fundamental parameter of a distribution is the *mean value* or *expected value* $E\{M\}$ of the random variable M. It could be considered as a reference value for the measure, with the same dimensions, defined by the following formula

$$E\{M\} = \int_{m_a}^{m_b} m\, f_M(m)\, dm \quad (13.9)$$

If there are only random effects a good estimate of such a value could be the arithmetical mean of the n measurement observations, m_i

$$E\{M\} = \overline{m} = \sum_{i=1}^{n} \frac{m_i}{n} \quad (13.10)$$

Denoting $g(M)$ as a function of the random variable M, it is possible to deduce

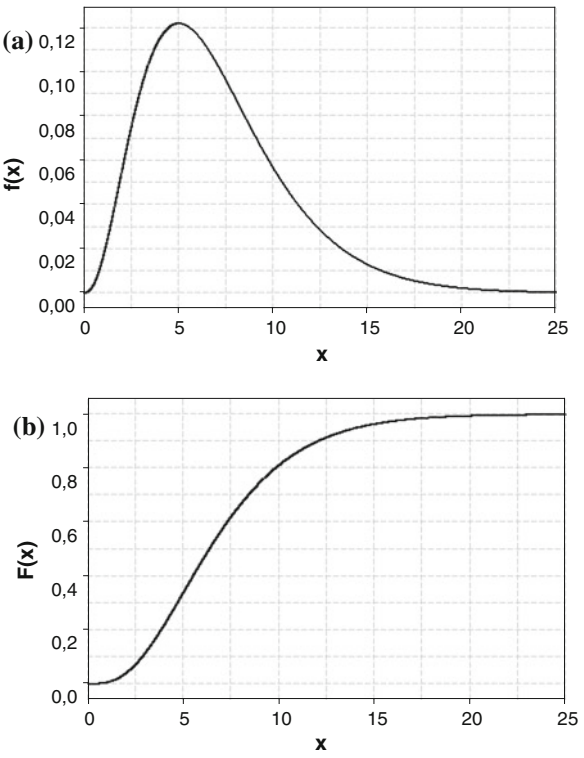

Fig. 13.1 Example of probability density function (**a**), and its cumulative distribution function (**b**)

$$E\{g(M)\} = \int_{m_a}^{m_b} g(m)\, f_M(m)\, dm \tag{13.11}$$

It is also easy to demonstrate the following expressions

$$E\{aM + b\} = a\, E\{M\} + b \qquad \text{with } a,\ b \text{ constants} \tag{13.12}$$

$$E\{M - E\{M\}\} = 0 \tag{13.13}$$

$$E\left\{\sum_{i=1}^{n} a_i M_i\right\} = \sum_{i=1}^{n} a_i E\{M_i\} \tag{13.14}$$

Another important parameter of a distribution is the variance of the random variable M defined as

$$Var\{M\} = E\left\{[M - E\{M\}]^2\right\} = E\{M^2\} - E^2\{M\}, \tag{13.15}$$

Equation (13.15) is general and it can be applied to both discrete and continuous random variables. In the case of continuous random variable it is possible to write the following expression

$$Var\{M\} = \int_{m_a}^{m_b} [m - E\{M\}]^2 f_M(m)\, dm = \int_{m_a}^{m_b} m^2 f_M(m)\, dm - [E\{M\}]^2 \quad (13.16)$$

$Var\{M\}$ represents a synthesis of the distribution spread around the mean value $E\{M\}$. If there are only random effects a good estimate of the variance of m_i observations in n successive repetitions is given by the empirical (or sample) variance, that is defined as

$$Var\{M\} = \sum_{i=1}^{n} \frac{(m_i - \overline{m})^2}{n-1} \quad (13.17)$$

where *(n−1)* are the so called *degrees of freedom*.

The estimation of variance, always greater than zero, corresponding with the square of the dimension of the measurement value. For this reason it is possible to take its square root: the *standard deviation* or *mean square root*. Both the *standard uncertainty* u_M, expressed as $u_M = \sqrt{Var\{M\}}$ and the *relative standard uncertainty* u'_M are strictly connected to this last concept and defined as follows

$$u'_M = \frac{\sqrt{Var\{M\}}}{|E\{M\}|} = \frac{u_M}{\overline{m}} \quad (13.18)$$

This is evident because the standard deviation, like the variance, is the best variability index of the measure around its expected value.

Other two important properties, easy to demonstrate, could be expressed by the following equations

$$Var\{\mathrm{a}\} = 0 \quad (13.19)$$

$$Var\{aM + b\} = a^2\, Var\{M\} \quad (13.20)$$

If a measure M is equal to the sum on n independent measures, the variance of M is equal to the sum of the single variances

$$Var\left\{\sum_{i=1}^{n} a_i M_i\right\} = \sum_{i=1}^{n} a_i^2 Var\{M_i\} \quad (13.21)$$

For the case of dependent measures see [9] and [10] where covariance is introduced.

If, instead, a measure is given by the product of two independent measures, it can be demonstrated that the variance of such a product is

$$Var\{M_1M_2\} = Var\{M_1\}Var\{M_2\} + E^2\{M_1\}Var\{M_2\} + E^2\{M_2\}Var\{M_1\} \tag{13.22}$$

In some applications the *k-order* moments, with $k = 1,2,\ldots$, are of a certain interest, and they are defined by the following expression

$$\mu_k = E\{M^k\} = \int_{-\infty}^{+\infty} m^k f_M(m)\, dm \tag{13.23}$$

where, obviously, $\mu_o = 1$ and $\mu_1 = E\{M\}$.

The *k-order* central moments can also be defined as follows

$$m_k = E\left\{(M - \mu_1)^k\right\} = \int_{-\infty}^{+\infty} (m - \mu_1)^k f_M(m)\, dm \tag{13.24}$$

In this case too, it is easy to deduce that $m_o = 1$, $m_1 = 0$, $m_2 = Var\{M\}$.

It is of a particular interest the so called *Mean Square Error* (MSE) that could be expressed as

$$m_a^2 = E\left\{(M - a)^2\right\} = m_2 + (\mu_1 - a)^2 = Var\{M\} + (E\{M\} - a)^2 \tag{13.25}$$

The difference $(\mu_1 - a) = (E\{M\} - a)$, where a is a value different from the mean, in the GUM [3] is denoted as *bias*.

In the follows some important continuous distributions that can be considered on the uncertainty evaluation of the data converters testing parameters are reported.

13.2.1 Uniform Distribution

In the case in which the probability density function is constant in a given interval, the distribution is denoted as *uniform,* or *rectangular*. Such distribution is considered in a situation where no information on the measurement value, thus on the testing data, and therefore on the data converters testing parameters is given in its definition interval. In this case

$$f_M(m) = \begin{cases} \frac{1}{m_{\max} - m_{\min}} & when \;\; m_{\min} \le M \le m_{\max} \\ 0 & otherwise \end{cases} \tag{13.26}$$

$$P\{m_a \le M \le m_b\} = F_M(m_b) - F_M(m_a) = \frac{m_b - m_a}{m_{\max} - m_{\min}} \tag{13.27}$$

where F(·) denotes the cumulative distribution function (Fig. 13.2).

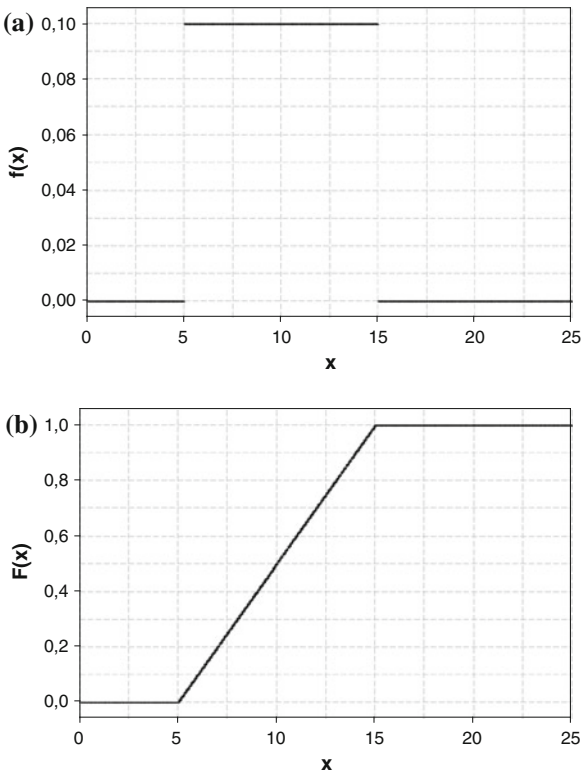

Fig. 13.2 Probability density function (PDF) and Cumulative distribution function (CDF) for a rectangular distribution, defined in the interval [5, 15] and zero outside

From Eqs. (13.9) and (13.16) the expected value and the variance of the rectangular distribution can be respectively obtained as

$$E\{M\} = \frac{m_{\min} + m_{\max}}{2}, \qquad Var\{M\} = \frac{(m_{\max} - m_{\min})^2}{12} \tag{13.28}$$

13.2.2 Gaussian or Normal Distribution

The probability density of the normal random variable M (measurement—testing data) is expressed by

$$f_M(m) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(m-\mu)^2}{2\sigma^2}} \qquad -\infty < x < +\infty \tag{13.29}$$

Note that $f_M(m)$ contains two statistical parameters: μ is the expected value and σ denotes the standard deviation. Such distribution is denoted as *Gaussian or Normal distribution* with plot shown in Fig. 13.3 and notation M $\sim$ N (μ, σ^2).

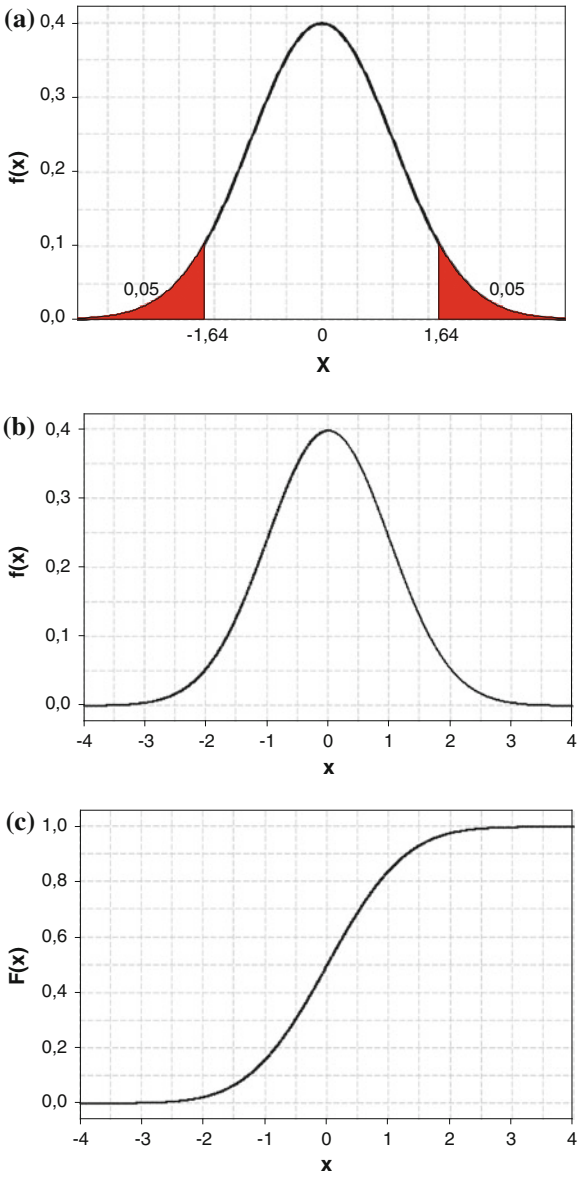

Fig. 13.3 Probability density function (PDF) for **a** Gaussian distribution $M \sim N(\mu, \sigma^2)$; **b** Probability density function (PDF); **c** Cumulative distribution function for standardized Gaussian $Z \sim N(0, 1)$

The graph is symmetrical with respect to μ, with higher concentration of measurement values near μ. From Eq. (13.29) it also appears that the density is the same for the two symmetrical points of μ; the points on the graph at which correspond $\mu - \sigma$ and $\mu + \sigma$ in the abscissa, represent the points of inflexion of the curve.

Table 13.1 Different probability values, in percent, in function of the coverage factor k

$P\{\mu - k\sigma \le M \le \mu + k\sigma\}$	k
68.27	1
90	1.645
95	1.960
95.45	2
99	2.576
99.73	3

For different values of the mean μ and same σ, the plot changes in the position with respect to the abscissa. Instead, with different σ values and same μ, the graph changes in shape maintaining the symmetrical condition; in other words, the higher σ is, the higher is the dispersion of measures close to μ, and vice versa.

The distribution function of Eq. (13.29) is

$$\begin{aligned} P\{m_a \le M \le m_b\} &= F_M(m_b) - F_M(m_a) \\ &= \frac{1}{\sigma\sqrt{2\pi}} \int_{m_a}^{m_b} e^{-\frac{(m-\mu)^2}{2\sigma^2}}\, dm = \frac{1}{\sqrt{2\pi}} \int_{m_a-\mu}^{m_b-\mu} e^{-\frac{m^2}{2}}\, dm \\ &= Z\left(\frac{m_b - \mu}{\sigma}\right) - Z\left(\frac{m_a - \mu}{\sigma}\right) \end{aligned} \tag{13.30}$$

where $Z = (M - \mu)/\sigma$ denotes the standardized random variable.

Considering k the *coverage factor*, in Table 13.1 the probability, in percent, that M is in the closed interval $\mu \pm k\sigma$ is shown in function of k.

As application in metrology, it is possible to centre the uncertainty interval in the measurement value $\overline{m} \neq E\{M\} = \mu$, where the random variable M is normally distributed; in addition, the interval $\{\mu \pm k\sigma\}$ is assumed as measurement result. Recalling Eq. (13.25) the quantity $\{\ \overline{m} - \mu\ \}$ is denoted as *bias*.

Equation (13.30) represents the confidence level associated to uncertainty interval; in the case of Gaussian distribution it can be evaluated as

$$P\{|M - \overline{m}| \le 1.96\ \sigma\} = Z\left(\frac{\overline{m} - \mu}{\sigma} + 1.96\right) - Z\left(\frac{\overline{m} - \mu}{\sigma} - 1.96\right) \tag{13.31}$$

where $k = 1.96$, corresponding to 95 % probability (see Table 13.1), is assumed as example.

13.2.3 Symmetrical Trapezoidal and Triangular Distributions

The probability density functions of a symmetrical trapezoidal and triangular distributions are shown in Fig. 13.4.

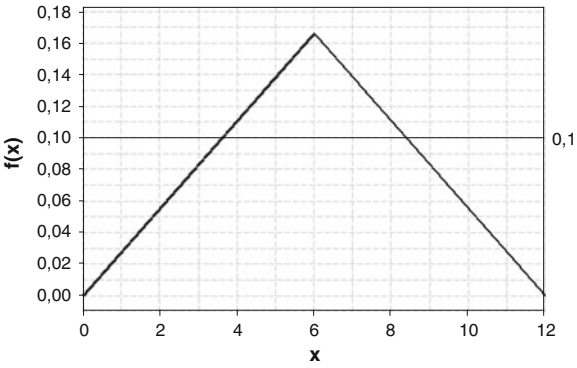

Fig. 13.4 The probability density function of a symmetrical trapezoidal and triangular distributions with parameters a = 1, b = 11 and h = 0.1

Where $h = \frac{2}{(1+\beta)(b-a)}$, a represents the lower limit of the M variable, b its upper limit, $(b-a)$ the M range; β is a particular value in the interval [0, 1] evaluated as ratio between the lowest and highest basic of the trapezium.

The probability density above introduced is equivalent to considering the convolution of densities for two independent variables R_1 and R_2, with uniform distribution in the intervals $[a_1,\ b_1]$ and $[a_2,\ b_2]$, respectively, and so with PDFs: $f_{R_1} = \frac{1}{b_1-a_1}$ and $f_{R_2} = \frac{1}{b_2-a_2}$ on the basis of the relation $M = R_1 + R_2$.

Parameters R_1 and R_2 are correlated to M parameters by means of the following properties

$$a = a_1 + a_2; b = b_1 + b_2; \beta = \frac{|(b_2 - a_2) - (b_1 - a_1)|}{(b - a)}$$

Considering $M = R_1 + R_2$, it is possible to calculate the expected value and the variance of the trapezoidal distribution as

$$E\{M\} = \frac{a+b}{2} \tag{13.32}$$

$$Var\{M\} = \frac{(b_1 - a_1)^2}{12} + \frac{(b_2 - a_2)^2}{12} = \left(1 + \beta^2\right)\frac{(b-a)^2}{24} \tag{13.33}$$

Assuming $\beta = 0$ and then $(b_1 - a_1) = (b_2 - a_2)$, the trapezoidal distribution corresponds to triangular shape (see Fig. 13.4) with $E\{M\} = \frac{a+b}{2}$, $\sqrt{Var\{M\}} = \frac{(b-a)}{2\sqrt{6}}$. With $\beta = 1$, instead, an uniform distribution can be obtained.

From the foregoing it is clear that the transition from a rectangular distribution to a trapezoidal and, then, to a triangular distribution, implies an increase that focuses on a greater probability of values near the mean value. The uncertainty is greatest in the case of rectangular distribution and gradually decreases to the normal distribution. In Table 13.2 the values of the standard deviation are summarized for different types of distributions.

Table 13.2 Continuous distributions: there is a decrease in uncertainties

Distribution	Standard deviation
Uniform in [a, b]	$\frac{(b-a)}{2\sqrt{3}}$
Trapezoidal in [a, b]	$\sqrt{(1+\beta^2)}\,\frac{(b-a)}{2\sqrt{6}}$
Triangular in [a, b]	$\frac{(b-a)}{2\sqrt{6}}$
Normal with probability 99.73 % in [a, b]	$\frac{(b-a)}{6}$

13.2.4 Student's t-Distribution

Considering n independent successive observations $[o_1, \cdots, o_n]$ of the same measurand (data converters testing parameters) at the output of the same measurement process in the same conditions of repeatability, random effects inside the measurement process are present.

Assuming each observation as a random variable normally distributed with same expected value m_o and uncertainty u_o so that $O_i = N(m_o, u_o)\ \ \forall i = 1, \ldots, n$, the arithmetic mean $\overline{O} = \sum_{i=1}^{n} \frac{o_i}{n} = N\left(m_o, \frac{u_0^2}{\sqrt{n}}\right)$ is, as known, a random variable normally distributed with the same expected value $\mu = m_o$ and variance $\sigma^2/n = u_0^2/n$.

Considering the arithmetic mean in standardized (normalized) form, so that $\frac{\overline{O}-m_o}{\frac{u_o}{\sqrt{n}}} = N(0,1)$, an estimator $S(\overline{O})$ of $\frac{u_o}{\sqrt{n}}$ can be defined as

$$S(\overline{O}) = \sqrt{\sum_{i=1}^{n} \frac{(o_i - \overline{O})^2}{[n(n-1)]}} \tag{13.34}$$

As a consequence it can be proved that the quantity

$$T_{n-1} = T_\upsilon = \frac{\overline{O} - m_o}{S(\overline{O})} \tag{13.35}$$

is distributed as a Student variable with $\upsilon = (n-1)$ degrees of freedom and probability density function [9]

$$f_v(t) = \frac{1}{\sqrt{\pi v}} \frac{\Gamma[(v+1)/2]}{\Gamma(v/2)} \frac{1}{(1+t^2/v)^{(v+1)/2}} \quad -\infty < t < +\infty; \quad v = 1, 2, \ldots \tag{13.36}$$

as shown in Fig. 13.5. In Eq. 13.36 $\Gamma(\alpha) = \int_0^{+\infty} e^{-x}\, x^{\alpha-1}\, dx$ represents the generic Gamma function.

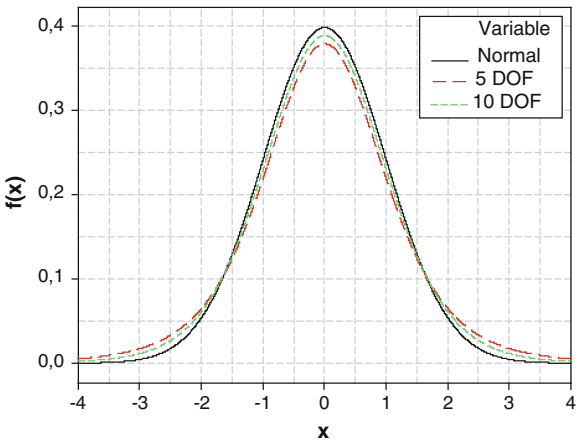

Fig. 13.5 Probability density function for a student's *t*-variable with 5, 10 and infinity degrees of freedom (normal distribution)

In the case of $v = 1$, Eq. (13.36) represents the probability density function of the *Cauchy* [10] distribution for which neither the expected value nor the variance is defined. It can be demonstrated

$$\lim_{v\to\infty} f_v(t) = \frac{1}{\sqrt{2\pi}} e^{-t^2/2}; \quad \lim_{v\to\infty} Var\{T_v\} = \lim_{v\to\infty} \frac{v}{v-2} = 1$$

From the previous formula it can be observed that in the case of infinite degrees of freedom the Student variable tends to the standardized variable. The probability, or the confidence level p, and the corresponding uncertainty interval $\left[-t_p, +t_p\right]$, centered around zero for the Student's t-variable with v degrees of freedom, are evaluated by Eq. (13.35) as follows

$$P\left\{-t_p \le T_v \le t_p\right\} = P\left\{m_o - t_p S\left(\overline{O}\right) \le \overline{O} \le m_o + t_p S\left(\overline{O}\right)\right\} = \int_{-t_p}^{t_p} f_v(t)\, dt = p$$

The values of t_p computed for different degrees of freedom are summarized in tables that can be found in literature [11] or by means of statistical software.

13.2.5 Chi Square Distribution (χ^2)

In the case of v normal variables expressed in reduced (or standardized) form $N_1(0,1), \ldots, N_v(0,1)$, such variables can be assumed as mutually independent and independent from M [9], the sum of squares $\chi^2_v = \sum_{i=1}^{n} N_i^2(0,1)$ is a random variable distributed as a Chi square (χ^2) with v degrees of freedom [12].

The probability density, asymmetrical as shown in Fig. 13.6, is represented by [9]

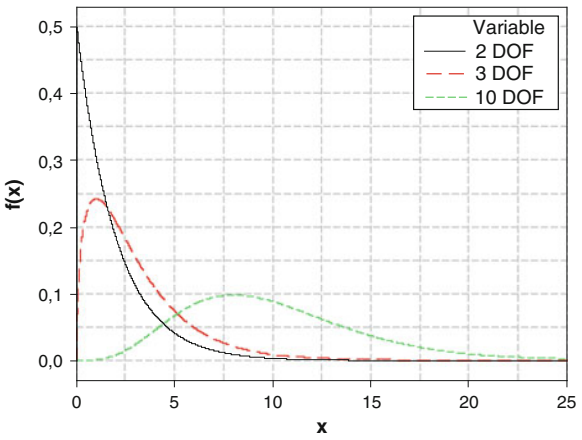

Fig. 13.6 Probability density function (PDF) for a *Chi square distribution with* 2, 3 and 10 degrees of freedom

$$f_v(m) = \frac{m^{(v/2)-1}}{2^{(v/2)}\Gamma(v/2)} e^{-(m/2)} \quad 0 \le m < +\infty; \quad v = 1, 2, \ldots \tag{13.37}$$

where $\Gamma(\alpha) = \int_0^{+\infty} e^{-x}\, x^{\alpha-1}\, dx$ denotes again the generic Gamma function.

For this distribution it is possible to evaluate the expected value and variance as υ and 2υ, respectively.

From Eq. (13.37), if $\upsilon = 2$, the *Rayleigh distribution* can be obtained and the *Maxwell* distribution with $v = 3$ [13].

Recalling Eq. (13.34) for the Chi Square

$$S(\overline{O}) = \sqrt{\sum_{i=1}^{n} \frac{(o_i - \overline{O})^2}{[n(n-1)]}} = \frac{u_o}{\sqrt{n}} \sqrt{\frac{\chi^2_{n-1}}{n-1}} \tag{13.38}$$

being

$$\chi^2_{n-1} = \sum_{i=1}^{n} N_i^2(0,1) = \sum_{i=1}^{n} \frac{(o_i - \overline{O})^2}{u_o^2} = \frac{(n-1)S^2(\overline{O})}{\sigma^2} \tag{13.39}$$

a Chi square with n−1 degrees of freedom, considering N_i *(0,1)* mutually independent and independent with respect to $\overline{O}$ that is N_o*(0,1)*. Consequently, Eq. (13.35) can be rewritten as

$$T_{n-1} = \frac{\overline{O} - m_o}{S(\overline{O})} = \frac{(\overline{O} - m_o)/(u_o/\sqrt{n})}{\sqrt{\sum_{i=1}^{n} \frac{(o_i - \overline{O})^2}{u_o^2}}} = \frac{N_o(0,1)}{\sqrt{\chi^2_{n-1}/(n-1)}} \tag{13.40}$$

where the relationship with the *Student* t-variable with *(n−1)* degrees of freedom appears clear.

In addition, from Eq. (13.38) and considering a confidence level p = (1−α), a confidence interval for the variance of the population can be estimated and, consequently, the standard deviation σ.

$$P\left\{\chi^2_{1-\frac{\alpha}{2}} \le \chi^2_{n-1} \le \chi^2_{\frac{\alpha}{2}}\right\} = P\left\{\frac{(n-1)S^2(\overline{O})}{\chi^2_{\frac{\alpha}{2}}} \le \sigma^2 \le \frac{(n-1)S^2(\overline{O})}{\chi^2_{1-\frac{\alpha}{2}}}\right\} = \int_{\chi^2_{1-\frac{\alpha}{2}}}^{\chi^2_{\frac{\alpha}{2}}} f_v(m)\, dt$$
$$= p$$

13.2.6 Confidence Interval of Measurement Results

The data converters testing parameters, denoted as the measure *M*, can be defined, as introduced in the paragraph 2, by the following expression

$$P\{|M - E\{M\}| \le k\, u_M\} = P\{E\{M\} - k\, u_M \le M \le E\{M\} + k\, u_M\} = p \quad (13.41)$$

This is the probability that the measure *M* is comprised within an interval that is a function of its expected value and of the standard uncertainty u_M multiplied by the *coverage factor k*. Obviously the *confidence level p* should be greater than possible, better if near to one. The interval

$$E\{M\} - k\, u_M \le M \le E\{M\} + k\, u_M \quad (13.42)$$

is called *confidence interval*; it represents the interval in which the probability to find a great number of possible values of *M*, within this interval, is near to one.

When the probability density function of *M* is known, it is possible to evaluate the confidence level *p* by the following expression

$$p = \int_{E\{M\}-k\, u_M}^{E\{M\}+k\, u_M} f_M(m)\, dm \quad (13.43)$$

It is now possible to deduce the measurement result as the **uncertainty interval**, connected to the measurand with confidence level equal to *p*.

From Eq. (13.7), it is possible to write

$$P\{m_\alpha \le M \le m_{p+\alpha}\} = \int_{m_\alpha}^{m_{p+\alpha}} f_M(m)\, dm = F(m_{p+\alpha}) - F(m_\alpha) = p \quad (13.44)$$

where α is a value, comprised between zero and one, defined by $0 \le p \le p + \alpha \le 1$. Finally

$$F(m_\alpha) = P\{M \le m_\alpha\} = \alpha \quad (13.45)$$

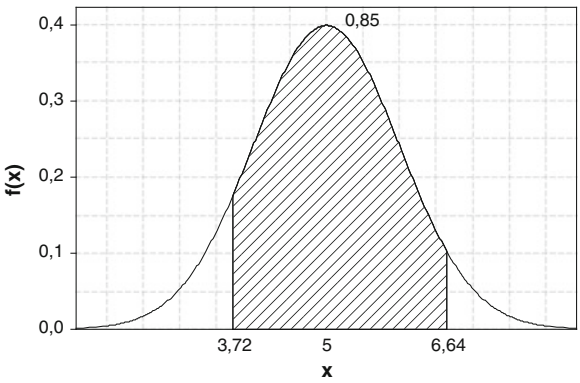

Fig. 13.7 PDF of a normal distribution with expected value equal to 5 and standard deviation equal to 1. The three regions under the *curve*, from the *left*, correspond respectively to $\alpha = 0.1$; $p = 0.85$ and $(1-p-\alpha) = 0.05$

The following figure represents an example of this interesting situation.

The choice of $\alpha = (1-p)/2$ generates, a **symmetric uncertainty interval**, with p as the uncertainty level, in the sense that the three sectors showed in Fig. 13.7 seem identical.

If the probability distribution of M is symmetrical, in the sense that its PDF is also symmetrical, this means that the distribution is symmetrically centered around the expected value of M, and the borders of the narrower interval, with confidence level equal to p, are equidistant.

In this case, introducing again the standard uncertainty of M, $u_M = \sqrt{Var\{M\}}$, the uncertainty interval can be written as $E\{M\} \pm ku_M$ where k is the coverage factor opportunely chosen to realize that $E\{M\} - ku_M$ and $E\{M\} + ku_M$ are respectively the so called $[(1-p)/2]$-*quantile* and $[(1+p)/2]$-*quantile* of the cumulative distribution function $F(m)$.

In case of asymmetrical distribution, all things change but, conceptually, the problem remains the same and in this case the value of α in (13.44) should be different by $(1-p)/2$ in order to have the minimum amplitude of $\left(m_{p+\alpha} - m_\alpha\right)$ and the uncertainty interval is the narrowest interval with a given confidence level p.

13.2.7 The Bootstrap Method in the Uncertainty Assessment

In the case of testing activities voted, for instance, to the fault diagnosis or destructive testing a limited number of data is often available. In this case the bootstrapping is a statistical method for estimating properties of an estimator from an approximating distribution, such as the empirical distribution of the observed values [14–19]. The method can be used for deriving robust estimates of standard errors and confidence intervals of a population parameter like mean, median, proportion, odds ratio, correlation coefficient or regression coefficient. It can be implemented by resampling the original measurement set several times. Each new

sample is obtained by random sampling with replacement from the original measurement set.

Bootstrapping may also be used for constructing hypothesis tests and it can be applied when the population is affected by outliers, too. The basic idea behind bootstrapping consists in drawing inferences from the observed sample rather than making potentially unrealistic assumptions about the unknown population the sample has been taken from. Therefore, the observed sample is processed as if it were the whole population.

In fact, the bootstrap method is often used as a robust alternative to inference based on parametric assumptions when those assumptions are in doubt and when parametric inference is impossible or requires very complicated formulas for the calculation of standard errors.

Moreover, the bootstrap doesn't require a big sample size: approximately 5–10 samples are enough [14, 15]. This is an important property, from the manufactures' point of view, considering the increase in testing time and equipment cost due to the exponential growth in DAC internal complexity.

A technical standard including easy test procedures that can be executed in the industrial environment, within a reasonable amount of time, would be more easily adopted by manufacturers.

As written above, the basic idea of the bootstrap method is to generate a large number of independent bootstrap samples by random resampling at the original observed values, $W_1, \ldots, W_n$. The bootstrap samples $W_1^*, \ldots, W_n^*$, are defined as being a random sample of size n drawn randomly with replacement from the original observed values, with elements taken zero, once or multiple times. Provided that n is big enough, for example about 1000, the 95 % confidence interval of the mean estimator can be obtained using the 0.025 percentile, $\overline{X}^*_{0.025}$, and 0.975 percentile, $\overline{X}^*_{0.975}$, of the bootstrap sample arithmetic means as endpoints.

13.3 Case Studies

13.3.1 Word Error Rate Measurement and Uncertainty Analysis in A/D Converters

A first example of the application of the proposed method based on the Student's t distribution can be carried out by using the test setup shown in Fig. 13.8, [20]. The Tektronix Arbitrary Waveform Generator AWG420 has been used to provide a sinusoidal signal to the Tektronix oscilloscope TDS 7704B. The records acquired by the oscilloscope have been then processed by a PC to compute the Word Error Rate (WER). Since the WER is small (usually measured in parts per million or parts per billion), a lot of samples must be collected to test for it [1, 2]. Before starting the test, a qualified error level (QEL) has to be chosen. This should be, according to [1, 2], the smallest value that excludes all other sources of error from

Fig. 13.8 WER measurement setup

this test. Particular attention should be paid to excluding the tails of the noise distribution as a source of word errors [21]. Therefore, the WER test has been carried out several times changing the considered QEL to find the value complying with the IEEE Std. 1241 and 1057 requirements.

The WER tests have been carried out in two phases in order to compare the results achieved by applying the proposed approach with those obtained by applying the Annex C of the IEEE Std. 1057. In the first phase six successive records of 10 M samples each have been acquired, spaced out at 1 min intervals. During the second phase, three sets of sixty records of 1 M samples each have been acquired. The time delay between two successive records has been set to 1 min. The first two acquisition sets (Set#1 and Set#2) were spaced 6 h from each other, while the third one (Set#3) has been taken 24 h after the first one.

The test sine wave, having an amplitude of 2 V_{PP} and a frequency of 100 kHz, has been acquired with a sampling frequency of 10 GS/s. Before starting each acquisition phase a warm up time of 1 h has elapsed.

The Student's t distribution [22–24] has been applied to the WER values obtained by means of the test procedure described above, the results have then been compared with the ones obtained by applying the method described in [2].

In Table 13.3 the experimental WER measurements obtained during the first phase are shown, considering two QEL, equal to 8 and 9 LSB, respectively.

In order to use the Student's t distribution for a more accurate WER measurement, the population distribution should be hypothesized as approximately normal. When sample size is big enough, it isn't necessary to analyze the nature of the population, because the central limit theorem guarantees that the expected value of the sample will be approximately distributed as a Gaussian. But when, as

Table 13.3 Experimental WER measurements

WER 8 LSB	Samples 8 LSB (M)	Wrong words 8 LSB	WER 9 LSB	Samples 9 LSB (M)	Wrong words 9 LSB
7.7×10^{-6}	10	77	2×10^{-7}	10	2
7.6×10^{-6}	10	76	4×10^{-7}	10	4
6.4×10^{-6}	10	64	3×10^{-7}	10	3
8.6×10^{-6}	10	86	2×10^{-7}	10	2
7.1×10^{-6}	10	71	8×10^{-7}	10	8
6.1×10^{-6}	10	61	4×10^{-7}	10	4

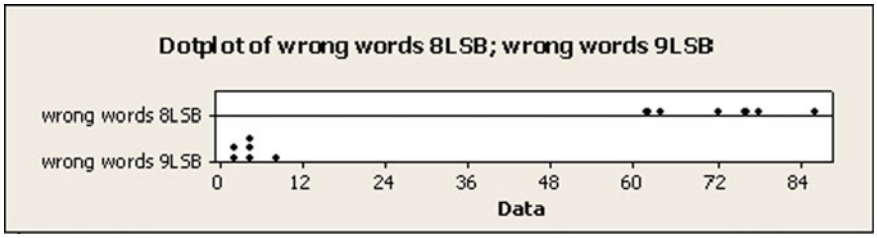

Fig. 13.9 Measurement dot plot trend

in this case, the sample size is small, less than approximately 30–35 samples, an assessment of the sample is required. A possible way is to build a box plot of the sample. If this representation does not reveal any significant asymmetries or outliers, it is reasonable to use a Student's *t* distribution [9].

Figures 13.9 and 13.10 show the dot plot and the box plot trend of the measured values. The dot plots report in a bidimensional graph the number of times a given WER has been observed (y axis) for each WER value (x axis), as can clearly be seen by comparing the values in Table 13.3 and Fig. 13.9.

Dot-plot is used to assess the distribution of continuous data. This plots each observation as a dot along a number line (x-axis). When values are close or the same, the dots are stacked. Dot plots are particularly useful for assessing distributions when there is a relatively small amount of data.

Boxplot is a graphical summary of the distribution of a sample that shows its shape, central tendency, and variability. Box plots can help to understand the distribution: the smallest observation (sample minimum), lower quartile (Q1), median (Q2), upper quartile (Q3), and largest observation (sample maximum). Box

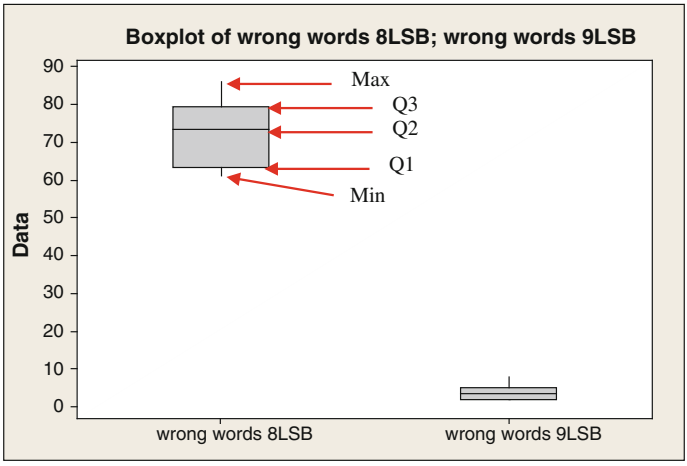

Fig. 13.10 Measurement box plot trend

Table 13.4 Student's t distribution statistical analysis

Variable	Samples	Mean	StDev	Minimum	Q1	Median	Q3	Maximum
Wrong words 8LSB	6	72.5	9.2	61.0	63.2	73.5	79.2	86.0
Wrong words 9LSB	6	3.8	2.2	2.0	2.0	3.5	5.0	8.0

plots is also useful for comparing several distributions; it can also indicate which measurements, if any, might be considered outliers. This is possible considering the spacing between the different parts of the box that indicates the degree of dispersion (spread) and skewness in the data, and identify outliers.

Figures 13.9 and 13.10 represent respectively the dot plot and the box plot trends of the data summarized in Table 13.3. Both the figures allow to see that the trends do not show a significant drift from normality: the plots are not strongly asymmetric and contain no outliers. Under these conditions the use of Student's *t* distribution has been considered appropriate and the statistical analysis results based on such distribution are shown in Table 13.4.

For a confidence level of 95 and 99 % with five degrees of freedom, the corresponding uncertainty intervals, determined from Eq. (13.42), have been reported in Table 13.5.

The results obtained by applying the proposed method, appropriately rounded, can be compared with the calculation of worst-case error rate as indicated by Annex C of the IEEE Std. 1057.

Only when the sample of WER observations is extracted from a normal population with known standard deviation, the quantiles of normal distribution can be used and, therefore, uncertainty intervals can be obtained using the method specified in Annex C.

From the results shown in Table 13.6, instead, it can be seen that by using the Student's *t* distribution the upper limit of the confidence interval is always greater that that obtained by following the method reported in Annex C, in proportion to the QEL, as in 8 LSB and 9 LSB cases. In both cases, in fact, using a *t* distribution with five degrees of freedom is enough to obtain a confidence interval wider than the one obtained using normal distribution, as suggested by the standard.

Table 13.5 Uncertainty intervals versus confidence levels

Measurement data					Confidence level (%)	Uncertainty interval
Variable	N	Mean	StDev	SE mean	95	(62.87–82.14)
Wrong words 8LSB	6	72.50	9.18	3.75		
					99	(57.39–87.61)
Variable	N	Mean	StDev	SE mean	95	(1.50–6.17)
Wrong words 9LSB	6	3.83	2.23	0.91		
					99	(0.17–7.50)

Table 13.6 Comparison between the upper limits of confidence intervals obtained by applying the proposed approach and the IEEE Std 1057-Annex C

	Student's *t*	Annex C
8 LSB (95 %)	82	78
8 LSB (99 %)	88	81
9 LSB (95 %)	6	5
9 LSB (99 %)	8	6

Only the significant digits have been used

13.3.2 Uncertainty Assessment of DAC Time Response Parameters

Modern digital storage scopes (DSOs) and digital phosphor scopes (DPOs) are widely diffused and provide an excellent instrument to perform rise and fall time measurements [21, 25].

The method described in the Sect. 13.7 has been applied in the measurements of rise time and fall time of square waveforms produced by different equipment including DAC output sections, i.e. arbitrary waveform generators and data acquisition boards [26]. DACs represent an essential component that provide a link between the digital and analog sections of a mixed-signal system. These components are crucial in fields such as optical networking, high-end medical imaging, cellular and smart phones, and digital video camcorders, requiring wide frequency band and high resolution. Therefore, there is an increasing interest in these devices, both from the scientific and industrial communities, resulting in a wider and deeper choice to the potential users.

A given customer can correctly compare and choose the converter suitable to his/her particular needs only if the uncertainty about the parameters of interest has been specified in the manufacturer's datasheet. Unfortunately, currently the evaluation of the measurement uncertainty is not yet included in any DAC standard [27].

The measures have been carried out by automating the preset measurement functions of two digital oscilloscopes. The number of generating and digitizing equipment has been chosen to highlight the potential influence of instrument-related biases on the correct estimation of the confidence levels.

On each generator-digitizer couple, ten groups of ten rise time and fall time measures have been taken, thus achieving 100 values for each parameter transferred on a PC where the calculations have been carried out. The aim of the whole validation phase is to compare the uncertainty analysis results obtained by the proposed bootstrap method, for which only ten samples are required, with the Student's *t* method, based on 100 samples, in order to prove the validity of the proposed approach.

The test phase has been carried out by using a data acquisition board as generator. In particular, the parameter values of the DAC embedded in the acquisition board NI PCI MIO 16E1, installed in and controlled by a PC, have been measured by means of the oscilloscope Tektronix TDS 5104 and an oscilloscope LeCroy

Table 13.7 DAC time response parameters: statistical analysis

DAC parameter	Bootstrap method			Student's t-method		
	Mean	Standard deviation	Confidence interval @ 95 %	Mean	Standard deviation	Confidence interval @ 95 %
S1 rise time [ns]	715	2	711–719	715	6	714–716
S1 fall time [ns]	825	3	820–829	825	8	823–826
S2 rise time [ns]	1.38	0.01	1.37–1.39	1.38	0.02	1.38–1.39
S2 fall time [ns]	1.44	0.01	1.43–1.45	1.44	0.02	1.43–1.44

SDA 6000 by changing the characteristics of the generated signals. A square wave having an amplitude of 10 V and a frequency of 1 kHz, has been first generated by the DAC of the acquisition board, using a LabVIEW virtual instrument, at an update rate of 100 kHz acquired by means of the Tektronix oscilloscope (signal S1). Then, the rise and fall time have been measured by the oscilloscope at the sampling frequency of 5 GSa/s. The second signal, generated with the same data acquisition board, a square wave with an amplitude of 150 mV, a frequency of 1 kHz, and an update rate of 1 MHz, has been acquired by the LeCroy oscilloscope at the sampling frequency of 20 GSa/s (signal S2).

The uncertainty intervals, with a confidence level of 95 %, for the rise and the fall time measurement are shown in Table 13.7 and compared with the results achieved by applying the traditional estimation method relying on Student's *t* distribution.

By analyzing the first results, it can be seen that both methods provide similar uncertainty intervals. The main differences between the measures corresponding to S1 and S2 are due to the very different slew rates set for the tests. The comparison between the confidence intervals for the same figure of merit on the same signal shows the effectiveness of the bootstrap method and its advantages: (i) no assumptions about the samples distribution, (ii) a reduced number of measurements required, (iii) easier and faster acquisition sessions than the inference-based methods, and (iv) therefore, more suitable for industrial applications.

The second experimental session has been carried out by changing the generation equipment while keeping the same digitizer. Both the second test phase configurations use the oscilloscope LeCroy SDA 6000 to measure 10 values of both rise time and fall time by setting a sampling frequency of 20 GS/s.

In the first test setup the DAC is embedded in an Agilent 33220A arbitrary waveform generator. Both the rise time and fall time have been measured, by means of the oscilloscope, on a square wave having an amplitude of 1 V and a frequency of 20 MHz (Signal S3).

In the second test setup the DAC is embedded in a Tektronix AWG 420 arbitrary waveform generator. The square wave signal used for the test has been generated with an amplitude of 1 V and a frequency of 10 MHz (Signal S4).

Table 13.8 DAC time response parameters from second measurement campaign: statistical analysis

DAC parameter	Bootstrap method			Student's t-method		
	Mean	Standard deviation	Confidence interval @ 95 %	Mean	Standard deviation	Confidence interval @ 95 %
S3 rise time [ns]	9.25	0.11	9.17–9.33	9.25	0.13	9.22–9.27
S3 fall time [ns]	9.24	0.09	9.17–9.31	9.24	0.11	9.22–9.26
S4 rise time [ns]	2.20	0.01	2.18–2.22	2.20	0.03	2.20–2.21
S4 fall time [ns]	2.02	0.01	2.01–2.04	2.02	0.03	2.02–2.03

In this second measurement campaign too, ten groups of ten rise time and fall time measures have been carried out, thus achieving 100 values for each parameter. The achieved confidence intervals, with a confidence level of 95 %, for the rise and fall time measurement are shown in Table 13.8 along with the results achieved by applying the traditional estimation method.

By analyzing the results shown above, it can be seen yet again, that the uncertainty intervals are almost the same in both of the two methods. In this way a further confirmation of the effectiveness of the bootstrap method has been achieved.

13.4 Conclusions

The aim of this chapter is to describe the uncertainty analysis of data converters testing parameters by means of some theoretical recalls and practical case studies, carried out on commercial generation and acquisition equipment.

The measurement uncertainty is an important parameter that gives a quantitative indication of the measurement quality, allowing the comparison of results coming from different sources or from reference values given in specifications or standards. Unfortunately, at the moment, the evaluation of the measurement uncertainty is not yet included in any data converters standard.

In particular, two case studies have been proposed: a new approach to assess the Word Error Rate (WER) parameter in digitizing waveform recorders based on Student's t distribution and a new methodology to evaluate the measurement uncertainty of rise time and fall time in DACs by using the bootstrap technique.

The first proposed test method has been experimentally verified by means of WER measurements on actual waveform recorders, where the test results have been compared to those achieved by implementing the IEEE Std. 1057 method. The validity of the proposed approach has been proven by applying it two different hardware platforms and by comparing the achieved results with those provided by

means of the IEEE method. In both cases, even changing the instrument under test, the signal generator, the signal frequency, the sampling frequency, the room temperature control, the warm-up time and the test length, the proposed method shown that the correct confidence intervals to be used in the uncertainty specifications are wider that those suggested in the Annex C of the IEEE Std. 1057.

The approach used to evaluate DAC time domain parameters allows to define confidence intervals from short acquisition records without needing hypotheses about the measurand distribution. Moreover, other advantages consist in a reduced number of required measurements, easier and faster than the inference-based methods and, therefore, more interesting from an industrial point of view. In order to validate the proposed approach, several measurement set-ups have been used, with different instrumentation and configurations.

From the assessment of the experimental results, it has been possible to prove the effectiveness of the proposed bootstrap technique, compared a traditional uncertainty estimation method such as the Student's t method. Therefore, the proposed method can give a contribution, in terms of uncertainty assessment, to the IEEE Std. 1658 concerning terminology and test methods for DACs.

References

1. IEEE Std. 1241: IEEE Standard for terminology an test methods for analog-to-digital converters. IEEE, Piscataway (2010)
2. IEEE Std. 1057: IEEE Standard for digitizing waveform recorders. IEEE, Piscataway (2007)
3. BIPM, IEC, IFCC, ISO, IUPAC, IUPAP, OIML, JCGM 100: Guide to the expression of uncertainty in measurement (2008)
4. Goyal, S., Chatterjee, A., Purtell, M.: A low-cost test methodology for dynamic specification testing of high-speed data converters. J. Electron. Test. Theory Appl. **23**(1), 95–106 (2007)
5. Linnenbrink, T., Blair, J., Rapuano, S., Daponte, P., Balestrieri, E., De Vito, L., Max, S., Tilden, S.: ADC testing. IEEE Instrum. Meas. Mag. **9**(2), 37–47 (2006)
6. Rapuano, S., Daponte, P., Balestrieri, E., De Vito, L., Tilden, S.J., Max, S., Blair, J.: ADC parameters and characteristics. IEEE Instrum. Meas. Mag. **8**, 44–54 (2005)
7. JCGM 200:2008: International vocabulary of metrology—Basic and general concepts and associated terms (VIM). JCGM, Paris (2007)
8. Joint Committee for Guides in Metrology, JCGM 101: Evaluation of measurement data—Supplement 1 to the guide to the expression of uncertainty in measurement—Propagation of distributions using a Monte Carlo method. Bureau International des Poids et Mesures (2008)
9. Navidi, W.: Statistics for engineers and scientists. McGraw Hill, NY (2007)
10. Papoulis, Athanasios: Probabilityrandom variables and stochastic processes. McGraw Hill, India (2002)
11. Ross, S.M.: A first course in probability, 7th edn. Sheldon Ross, Pearson (2010)
12. Catelani, M., Ciani, L., Zanobini, A.: Uncertainty interval evaluation using the chi square and fisher distributions in the measurement process. Metrol. Meas. Syst. **17**(2), 195–204 (2010)
13. Montgomery, D.C.: Introduction to statistical quality control, 6th edn. Wiley (2009)
14. Efron, B., Tibshirani, R.J.: An introduction to the bootstrap, Monographs on statistics and applied probability, vol 57. Chapman and Hall, London (1993)
15. DiCiccio, T.J., Efron, B.: Bootstrap Confidence Intervals. Stat. Sci. **11**(3), 189–228 (1996)

16. Davison, A.C., Hinkley, D.V.: Bootstrap methods and their applications. Cambridge Univ. Press, Cambridge (1997)
17. Wehrens, Ron, Putter, Hein, Buydens, Lutgarde M.C.: The bootstrap: a tutorial. Chemometr. Intell. Lab. Syst. **54**(1), 1 (2000)
18. Zoubir, A.M., Iskander, D.R.: Bootstrap methods and applications. IEEE Signal Process. Mag. **24**(4), 10–19 (2007)
19. Farooqui, Sami A., Doiron, Ted, Sahay, Chittaranjan: Uncertainty analysis of cylindricity measurements using bootstrap method. Measurement **42**(4), 524–531 (2009)
20. Balestrieri, E., Catelani, M., Ciani, L., Rapuano, S., Zanobini, A.: The student's t distribution to measure the word error rate in analog-to-digital converters. Measurement **45**(2), 148–154 (2012)
21. Doebelin, Ernest O.: Measurement systems: application and design, 5th edn. McGraw-Hill, Boston (2003)
22. Catelani, M., Zanobini, A., Ciani, L.: Introduction to the t and Chi square distribution for a more accurate evaluation of the measure of the Word Error Rate in analog-to-digital converters. Metrol. Meas. Syst. **15**, 483–488 (2008)
23. Balestrieri, E., Catelani, M., Ciani, L., Rapuano, S., Zanobini, A.: New statistical approach to word error rate measurement in analog-to-digital converters. In: Proc. of 17th symposium IMEKO TC 4 on instrumentation for the ICT Era and 15th international workshop on ADC modelling and testing, Kosice, 8–10 Sept. 2010
24. Balestrieri, E., Catelani, M., Ciani, L., Rapuano, S., Zanobini, A.: Efficient estimation of the word error rate of analog-to-digital converters. In: Proc. of 2011 IMEKO IWADC & IEEE ADC Forum, Orvieto (PG) (2011)
25. Kester, Walt: Analog Devices Inc., Evaluating high speed DAC performance, MT-013 Tutorial, Rev. A, 2008. Available online <http://www.analog.com>
26. Balestrieri, E., Catelani, M., Ciani, L., Rapuano, S., Zanobini, A.: Uncertainty evaluation of DAC time response parameters. In: Proc. of 2011 IMEKO IWADC & IEEE ADC forum, Orvieto (PG) (2011)
27. Balestrieri, E., Daponte, P., Rapuano, S.: Recent developments on DAC modelling, testing and standardization. Measurement **39**(3), 258–266 (2006)