

Calibration Techniques for Time-Interleaved SAR A/D Converters

Dusan Stepanovic



Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2012-225

<http://www.eecs.berkeley.edu/Pubs/TechRpts/2012/EECS-2012-225.html>

December 4, 2012

Copyright © 2012, by the author(s).
All rights reserved.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission.

Calibration Techniques for Time-Interleaved SAR A/D Converters

by

Dusan Vlastimir Stepanovic

A dissertation submitted in partial satisfaction of the
requirements for the degree of
Doctor of Philosophy

in

Electrical Engineering and Computer Sciences

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Borivoje Nikolic, Chair
Professor Paul Gray
Professor Paul Wright

Fall 2012

Calibration Techniques for Time-Interleaved SAR A/D Converters

Copyright 2012
by
Dusan Vlastimir Stepanovic

Abstract

Calibration Techniques for Time-Interleaved SAR A/D Converters

by

Dusan Vlastimir Stepanovic

Doctor of Philosophy in Electrical Engineering and Computer Sciences

University of California, Berkeley

Professor Borivoje Nikolic, Chair

Benefits of technology scaling and the flexibility of digital circuits favor the digital signal processing in many applications, placing additional burden to the analog-to-digital converters (ADCs). This has created a need for energy-efficient ADCs in the GHz sampling frequency and moderate effective resolution range. A dominantly digital nature of successive approximation register (SAR) ADCs makes them a good candidate for an energy-efficient and scalable design, but its sequential operation limits its applicability in the GHz sampling range. Time-interleaving can be used to extend the efficiency of the SAR ADCs to the higher frequencies if the mismatches between the interleaved ADC channels can be handled in an efficient manner.

New calibration techniques are proposed for time-interleaved SAR ADCs capable of correcting the gain, offset and timing mismatches, as well as the static nonlinearities of individual ADC channels stemming from the capacitor mismatches. The techniques are based on introducing two additional calibration channels that are identical to all other time-interleaved channels and the use of the least mean square algorithm (LMS). The calibration of the channel offset and gain mismatches, as well as the capacitor mismatches, is performed in the background using digital post-processing. The timing mismatches between channels are corrected using a mixed-signal feedback, where all calculations are performed in the digital domain, but the actual timing correction is done in the analog domain by fine-tuning the edges of the sampling clocks. These calibration techniques enable a design of time-interleaved converters that use minimum-sized capacitors and operate in the thermal-noise-limited regime for maximum energy and area efficiency.

The techniques are demonstrated on a time-interleaved converter that interleaves 24 channels designed in a 65 nm CMOS technology. The ADC uses the smallest capacitor value of only 50 aF, achieves 50.9 dB SNDR at $f_s = 2.8$ GHz with the effective-resolution bandwidth higher than the Nyquist frequency, while consuming only 44.6 mW of power.

Contents

List of Figures	iii
List of Tables	vi
1 Introduction	1
1.1 Motivation	1
1.2 Research Goal	2
1.3 Related Work	4
1.4 Thesis Organization	6
2 Error Sources in Time-Interleaved SAR A/D Converters	7
2.1 Basic SAR ADC Operation	7
2.2 Static Nonlinearities	10
2.3 Basics of Time-Interleaving	12
2.4 Offset Mismatch	13
2.5 Gain Mismatch	13
2.6 Timing Mismatch	15
2.7 Bandwidth Mismatch	15
2.8 Finite Sampling Aperture	20
3 Calibration of Static Nonlinearities in SAR A/D Converters	22
3.1 Overview of Techniques for Linearity Calibration in SAR ADCs	22
3.2 Direct and Reverse Switching	24
3.3 Trimming-Based Calibration	27
3.3.1 Single-Channel Single-Core SAR ADC Calibration	27
3.3.2 Single-Channel Dual-Core SAR ADC Calibration	28
3.3.3 Multi-Channel SAR ADC Calibration	29
3.4 Digital Background Calibration	30
3.4.1 Single-Channel Single-Core SAR ADC Calibration	31
3.4.2 Single-Channel Dual-Core SAR ADC Calibration	32
3.4.3 Multi-Channel SAR Calibration	33

3.5	Simulation Results	34
3.6	Algorithm Limitations	35
4	Calibration of Timing Mismatches	38
4.1	Overview of Timing Calibration Techniques	38
4.2	Basic Idea	39
4.3	Derivative Estimation	41
4.4	Convergence Analysis	43
4.5	Algorithm Modifications	45
4.6	Calibration Without Additional Channel	46
4.7	Simulation Results	47
4.8	Algorithm Limitations	50
5	Circuit Implementation	52
5.1	Single SAR Channel	52
5.1.1	Capacitive DAC	52
5.1.2	Top-Plate and Bottom-Plate Switches	55
5.1.3	Comparator	57
5.1.4	SAR Logic	60
5.1.5	SAR Layout Plan	62
5.2	Clock Generation	64
5.3	Calibration Logic	68
5.4	Full-Chip Integration	70
6	Measurement Results	72
6.1	Measurement Setup	72
6.2	Radix Measurements	74
6.3	Timing Mismatches	78
6.4	Bandwidth Mismatch	79
6.5	Single-Tone Measurements	80
6.6	Two-Tone Measurements	90
6.7	Performance Summary	93
6.8	Comparison to Prior Art	94
6.9	Design Limitations	97
7	Conclusion	99
7.1	Key Accomplishments	99
7.2	Future Work	100

List of Figures

1.1	Figure of merit of all ADCs with resolution between 6 and 10 bits and sampling frequency between 10MHz and 10GHz published at ISSCC and VLSI conferences from 1997 to 2012.	2
1.2	Energy per conversion vs. sampling frequency for a single-channel and a time-interleaved ADC.	3
2.1	Simplified schematic of SAR ADC.	8
2.2	Sampling phase of SAR ADC.	8
2.3	4-bit conversion example.	9
2.4	Constructing transfer functions of SAR ADC for a) radix 2.2 and b) radix 1.8.	11
2.5	Basic block diagram of a time-interleaved ADC.	13
2.6	Spectrum of a 4-time interleaved ADC with offset mismatches only.	14
2.7	Spectrum of a 4-time interleaved ADC with gain and/or timing mismatches.	14
2.8	SDR due to bandwidth mismatch vs. relative frequency.	17
2.9	SDR due to bandwidth mismatch vs. relative frequency after timing and gain calibration.	18
2.10	SDR at DC vs. relative calibration frequency after timing and gain calibration.	19
2.11	Relative cutoff frequency vs. relative calibration frequency after timing and gain calibration.	20
3.1	Direct and reverse switching in SAR ADC.	25
3.2	Transfer characteristics of direct and reverse switching in SAR ADC.	26
3.3	Typical ENOB learning curves for a) single-channel single-core, b) single-channel dual-core and c) eight time-interleaved SAR ADC calibration	36
3.4	FFT of a sinusoidal signal (a) before and (b) after digital calibration for eight time-interleaved SAR ADCs	37
4.1	RC circuit.	41
4.2	Two RC circuits with different bandwidths.	42
4.3	Practical realization of two RC circuits with different bandwidths.	42
4.4	Block diagram of the timing calibration.	44

4.5	$D(\omega)$ and enabling of the timing calibration.	45
4.6	Modification of the basic timing calibration algorithm.	46
4.7	Stability range in the first Nyquist zone.	48
4.8	Typical ENOB convergence.	49
4.9	Typical Δt_{dig} convergence.	50
4.10	ENOB vs. frequency with and without timing calibration.	51
5.1	Block diagram of a single SAR ADC channel	52
5.2	Capacitive DAC a) single-ended schematic and b) phases of operation.	53
5.3	Illustration of DAC layout.	54
5.4	Schematic of the top-plate bootstrapped switch with non-overlapping clock generator and timing diagram of clock phases.	56
5.5	Schematic of the bottom-plate switch.	57
5.6	Schematic of the StrongArm latch comparator.	58
5.7	Schematic of the SAR logic.	61
5.8	Schematic of the sar_cell block.	62
5.9	Schematic of the sw_drv block.	62
5.10	Layout plan of a SAR ADC channel.	63
5.11	Clock signals with timing diagrams.	64
5.12	Low-jitter bottom-plate sampling.	66
5.13	Effect of supply voltage change on the sampling-clock edges.	67
5.14	Jitter ENOB calculated at $f_{in} = 1.5GHz$ vs. RMS supply noise for single-ended and pseudo-differential sampling.	67
5.15	Implementation of clock tuning.	68
5.16	Chip layout.	71
6.1	Block diagram of the measurement setup.	73
6.2	Block diagram of the testing board.	73
6.3	Chip photograph.	74
6.4	Measured radices for 4 different chips.	75
6.5	Averaged radices across 24 different channels on a single die.	76
6.6	Standard deviation of radices across 24 different channels on a single die.	77
6.7	Measured timing mismatch for 4 different chips.	78
6.8	Measured timing mismatch for 3 different input frequencies.	79
6.9	Measured gain mismatch for 3 different input frequencies.	80
6.10	Spectrum before and after calibration for $f_{in} = 19.88 MHz$	81
6.11	Spectrum before and after calibration for $f_{in} = 1379.56 MHz$	82
6.12	Performance plots vs. input frequency ($f_s = 2.8 GHz$, $V_{DD} = 1.2 V$).	83
6.13	Harmonic distortion vs. input frequency.	84
6.14	SNDR vs. input frequency for different sampling frequencies.	85
6.15	Performance plots vs. input frequency ($f_s = 2.7 GHz$, $V_{DD} = 1.1 V$).	86

6.16 SNDR vs. input signal level.	87
6.17 SNR vs. input signal level.	88
6.18 SNDR vs. frequency for different number of output bits.	89
6.19 DFT with two input tones ($f_c = 999.88$ MHz, $\Delta f = 31.92$ MHz).	90
6.20 IM2 and IM3 vs. central frequency, $\Delta f = 5$ MHz.	91
6.21 IM2 and IM3 vs. Δf , $f_c = 999.88$ MHz.	92
6.22 Power consumption breakdown at $f_s = 2.8$ GHz, $V_{DD} = 1.2$ V.	94
6.23 Energy per conversion for all ADCs with $f_s > 1$ GHz published at ISSCC and VLSI conferences from 1997 to 2012.	95
6.24 Figure of merit of all ADCs with resolution between 6 and 10 bits and sampling frequency between 10MHz and 10GHz published at ISSCC and VLSI conferences from 1997 to 2012, including this work.	96

List of Tables

3.1	Operators \oplus and \ominus for different types of switching.	28
4.1	Simulation setup.	48
4.2	Error sources included in the simulation.	49
6.1	ADC performance summary	93

Chapter 1

Introduction

1.1 Motivation

Analog-to-digital converters (ADCs) have been one of the key electronic components in electronic devices that require interaction with the real world since the early days of digital signal processing systems. The three main performance metrics used to evaluate and categorize the ADCs are their speed, resolution and power. The research in the ADC area has been driven by new applications that constantly demand higher speeds and resolutions. In one group of applications, with a steady increase in performance of digital circuits, there is a trend of performing more signal processing in the digital domain, and the ADCs are being moved closer to the chip input. This means that more information about the analog signal needs to be captured, which translates into more stringent speed and resolution requirements for ADCs. An example of this kind of application is the direct-sampling receiver in TV tuners that requires a simultaneous sampling of more than 16 channels arbitrarily located in the 48-1002 MHz TV band instead of the integration of multiple single channel receivers. In other applications, the use of newly available resources dictate more demanding ADC requirements. A typical example is the WiGig (60 GHz) communication system where a wide frequency spectrum has enabled the possibility of high-rate GHz communications. With higher data rates comes the demand for higher sampling speeds. Higher resolutions are also needed if complex modulation schemes are to be used and/or if more channel equalization and filtering is to be done in the digital domain. Both aforementioned applications, the TV tuners and the WiGig systems, require the ADCs with the sample rate of around 2.5 GHz and the resolution of around 8 effective bits. At the time of writing this dissertation, the ADCs with these specifications are available, but their energy efficiency significantly lags behind that of the ADCs that sample in the 100 MHz sampling rate range. This is evident from the plot shown in Figure 1.1 [31]. The plot shows the standard figure of merit (FoM), often used as the measure of the converter energy efficiency, for all ADCs with resolution between 6 and 10 bits and sampling frequencies between 10 MHz and 10 GHz published at the ISSCC and

VLSI conferences from 1997 to 2012. The standard figure of merit is defined as

$$FoM = \frac{P}{2 * \min(f_s/2, ERBW) * 2^{ENOB}}, \quad (1.1)$$

where P is the power, f_s is the sampling frequency, $ERBW$ is the effective resolution bandwidth, and $ENOB$ is the effective number of bits. As it can be seen from the plot, the ADCs with sampling frequencies around 100 MHz can achieve figure of merit close to 10 fJ/conversion-step. On the other side, the best figure of merit of the ADCs with sampling frequency higher than 2 GHz exceeds 800 fJ/conversion-step.

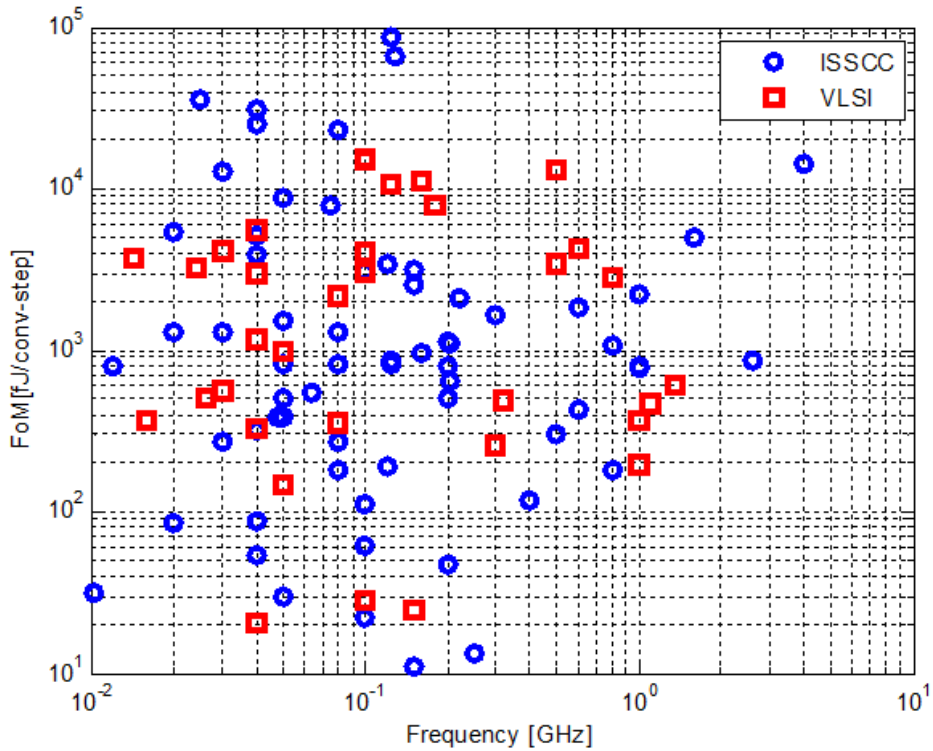


Figure 1.1: Figure of merit of all ADCs with resolution between 6 and 10 bits and sampling frequency between 10MHz and 10GHz published at ISSCC and VLSI conferences from 1997 to 2012.

1.2 Research Goal

The goal of this research is to develop techniques for improving the energy efficiency of the ADCs operating in 2-3 GHz sampling frequency range with resolution around 8 effective

bits. This problem can be solved by time-interleaving of many energy-efficient ADCs and employing simple calibration techniques to deal with the unwanted side effects of time-interleaving. The reasoning behind this approach is explained next.

The energy per conversion of an ADC, defined as the ratio of the power and the sampling frequency, typically increases with the sampling frequency as shown in Figure 1.2. An attractive way to move the efficiency towards higher frequencies is to interleave multiple ADCs, each operating in the energy-efficient regime. This way, by interleaving M channels, the effective sampling frequency increases by a factor of M , while the individual ADC channels still work efficiently. The equivalent time-interleaved ADC will always be less energy-efficient than its constituent interleaved channels because of the overhead associated with time-interleaving. This is also illustrated in Figure 1.2. The overhead includes the generation and distribution of high-quality multiple clock phases and the distribution of the input and reference signals to all the channels. Also, the channel mismatches like offset, gain and bandwidth mismatches, represent an obstacle in achieving the desired performance in time-interleaved ADCs. These mismatches need to be corrected, which further increases the overhead of time-interleaving.

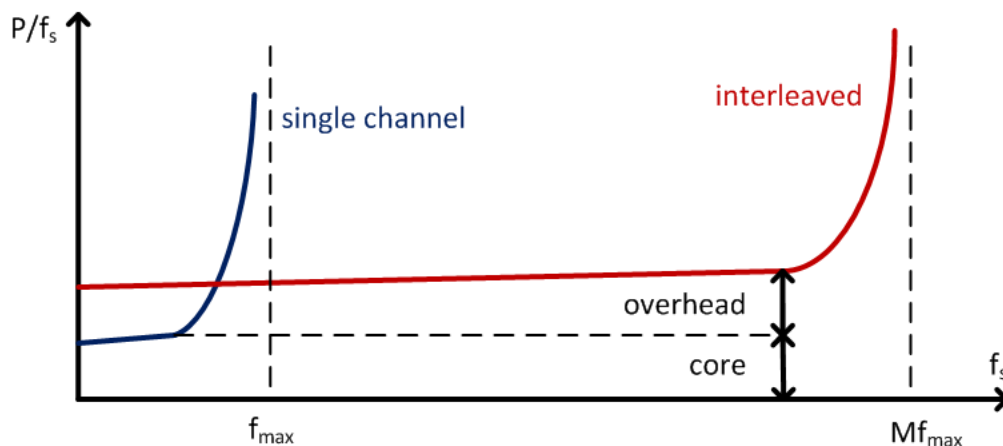


Figure 1.2: Energy per conversion vs. sampling frequency for a single-channel and a time-interleaved ADC.

The power optimization process roughly consists of two parts: minimizing the power of individual channels by choosing a proper architecture and design, and minimizing the overhead of the time-interleaving. Although there is no known exact answer to which architecture is the best choice for a given set of specifications, the empirical data show that SAR (successive approximation register) ADCs built in sub-100 nm CMOS technologies can achieve excellent power efficiency in moderate sampling frequencies (less than 200 MHz) and resolution (8-12 bits). The efficiency of the SAR ADCs stems from their mostly digital nature, which enables the power and area scaling of their digital part with the scaling of

technology. The analog power and area do not scale that well with the technology scaling and innovative techniques on the analog side are often needed to achieve the overall efficiency. It is important to note that the area of the ADC is even more important in the time-interleaved architecture because the overhead in distributing the sensitive analog signals common to all the channels is directly proportional to the physical size of the channels. The majority of modern SAR ADC implementations is based on switched capacitor circuits and includes a capacitive digital-to-analog converter (DAC) that is used to perform radix-based search. To get the maximum power and area savings these capacitors need to be minimized to the point where the resolution becomes limited by the thermal noise. For many applications this means that the smallest capacitor in the capacitive DAC needs to be much smaller than 1fF. Matching of the capacitors this small is limited by both random variations caused by process variability and systematic layout mismatches, and can easily limit the overall linearity of the converter. One of the central topics of this research is developing low-cost calibration techniques that will correct not only channel mismatches, but also the nonlinearities of all individual channels coming from the capacitor mismatches.

Another major problem of the time-interleaved architecture is the phase or timing mismatch of clocks in multiple channels. This problem can be solved by introducing a common front-end sampler, but this approach comes with a power and noise penalty in terms of buffering the sampled voltage and resampling it in the individual channels. It is more desirable to have a simple clock generation scheme that generates low-jitter multiple clock phases that can be fine-tuned using a calibration algorithm that introduces a very low overhead. Exploring this kind of approach is another central topic of this research.

1.3 Related Work

Time-interleaved converter arrays were first introduced by Black et al. in [5] with the intention of reducing the die size and relaxing the requirements on the fabrication process. More recently, the time-interleaved ADCs have been used to achieve extremely high sampling speeds that cannot be achieved by any other ADC architecture, or to improve the energy efficiency at the speeds that have traditionally been dominated by the flash and folding-interpolating architectures. Poulton et al. [34] interleaved 80 current-mode pipeline ADCs to get 20 GS/s speed for use in the sampling oscilloscopes. Abundant digital processing is used to calibrate channel and radix mismatches. In [14] 160 6-bit SAR ADCs were interleaved to obtain a 40 GS/s ADC for optical communications. FFT processing and calibration DACs were used to correct the offset, gain and timing mismatches. Interleaving of 8 flash ADC channels was used in [11] to achieve 5 bit resolution at 12 GS/s speed with the target application of digitally-equalized serial links. An additional channel consisting of a single comparator was introduced for the timing skew calibration. A background calibration algorithm maximizes the correlation between the calibration and time-interleaved channels, thus minimizing the timing errors. A potential of time-interleaved ADCs for higher resolutions at

GS/s speeds was demonstrated by Louwsma et al. in [27] at 1.35 GS/s and 7.7 effective bits. A careful layout and minimization of the clock path from the master clock to the sampling switches was used to achieve sufficient timing accuracy. Doris et al. [9] used interleaving of 64 SAR ADC channels to get more than 8 effective bits of resolution at 2.5 GS/s. Four track-and-hold circuits were interleaved to achieve low timing skew, and the sampled input signal was further multiplexed to the interleaved channel using a feedback-feedforward buffer interface. Current-steering DACs were used both as the main DAC and calibration DACs for offset and gain calibration. Large area in this solution led to a large interleaving power overhead, for a total power of 480 mW.

The first SAR ADC based on the capacitive charge redistribution was introduced by McCreary et al. in [29]. The capacitor mismatches were identified as a serious problem in the early days of charge redistribution ADCs and one of the first calibration techniques for the capacitor mismatches was presented by Lee et al. in [22]. The mismatch errors were measured after the power up and an auxiliary DAC was used to add the measured error during the normal operation. Kuttner [21] used a careful layout technique to achieve 10 bits linearity with unit capacitance of 1.5 fF. This technique requires a lot of effort at layout design level and may be hard to apply to even smaller capacitors. A foreground calibration with a known input signal and linear curve fitting was used in [8] to calculate weights of a non-binary series capacitive ladder. Liu et al. [24] proposed a background calibration based on the least-mean-square (LMS) algorithm, which uses an accurate algorithmic reference converter to calibrate capacitor mismatches in a time-interleaved SAR ADC. Another approach based on the LMS algorithm was presented in [26] where a small capacitor is added to the capacitive array to introduce a perturbation signal. Each signal sample is converted twice with different sign of the perturbation signal and the capacitor weights are adaptively calculated from the difference of the two conversion results. Split capacitor and C-2C arrays [42], [4] have been proposed to solve the problem of the smallest capacitor size. However, when designed to operate in thermal-noise-limited regime, these arrays need higher total capacitance and therefore larger area, and their linearity is dependent on bottom and top plate parasitic capacitances, which creates problems similar to the mismatch of small capacitors in radix-based arrays. In both cases some form of calibration or special layout techniques are needed to address the mismatch caused nonlinearities of SAR converters if minimum power and area are to be achieved.

Commercial solutions in the desired resolution and sampling frequency range are available, but they consume excessive amounts of power. The standalone ADC described in [38] uses folding and interpolating architecture and time-interleaving of two channels to realize the sampling speed of 3 GS/s with the effective resolution of 9 and 8 bits at DC and Nyquist frequency, respectively. This ADC uses a 1.9V supply and typically consumes 3.14 W of power. An advanced SiGe process is used to design the 8-bit, 2.2 GS/s ADC described in [28]. This converter achieves the effective resolution of 6.9 bits at the Nyquist frequency and consumes 6.8 W of power.

1.4 Thesis Organization

Chapter 2 begins with a description of the basic SAR ADC operation, and then progresses towards the effects of capacitor mismatches on the transfer function of a SAR ADC. The chapter ends with a discussion of errors caused by the channel mismatches in time-interleaved ADCs.

In Chapter 3 a set of techniques for calibration of capacitor mismatches in SAR ADCs based on the LMS algorithm is developed. The techniques can be applied to single-channel or parallel ADCs, and can be executed either in analog domain using electronic trimming of the capacitors, or in digital domain as a background post-processing.

Chapter 4 deals with the calibration of timing errors in time-interleaved ADCs. Two calibration techniques based on the LMS algorithm and a mixed-signal feedback for fine-tuning of the clock edges are presented, together with behavioral simulation results.

Chapter 5 shows the circuit-level implementation details of different ADC blocks such as the single SAR ADC channel, the clock generation circuitry and the calibration logic.

Chapter 6 describes the measurement setup and presents the measurement results of different performance metrics under varied conditions.

In Chapter 7 conclusions are drawn and potential topics for future improvements and research are suggested.

Chapter 2

Error Sources in Time-Interleaved SAR A/D Converters

This chapter discusses the error sources in time-interleaved SAR ADCs that come from either the capacitor mismatches or the channel mismatches in time-interleaved architecture. The former are common for both the single-channel and the parallel architecture, while the later are obviously only present in the time-interleaved architecture. These are the errors that represent a major obstacle to an energy-efficient design in the proposed architecture, and are calibrated using the techniques described in Chapter 3 and Chapter 4. An intuitive approach is used to explain the effects of the channel mismatches. More rigorous mathematical treatment can be found in [39]. Other error sources present in SAR ADCs, such as switch nonlinearities, charge injection, DAC settling etc., are dealt with by a careful design, as described in Chapter 5, and are not a topic of this discussion.

2.1 Basic SAR ADC Operation

A simplified schematic of a conventional N-bit SAR ADC is shown in Figure 2.1. Single-ended version is shown throughout this chapter for simplicity, although the whole analysis applies to a differential circuit as well. It consists of a binary comparator, SAR logic, switches, and a radix-weighted capacitor array $C_{0A}, C_0, C_1, \dots, C_{N-1}$. For a radix α ($1 < \alpha \leq 2$) the capacitor sizes are defined as

$$\begin{aligned} C_0 &= C_{0A} \\ C_i &= \alpha^i C_0 \quad , \quad i = \overline{1..N}. \end{aligned} \tag{2.1}$$

C_P is the total equivalent parasitic capacitance at the comparator input, and V_{OS} is the comparator input-referred offset.

Before the conversion process starts, the input signal is sampled onto all the capacitors, as shown in Figure 2.2. Next, in the following N bit-testing phases the switches connect

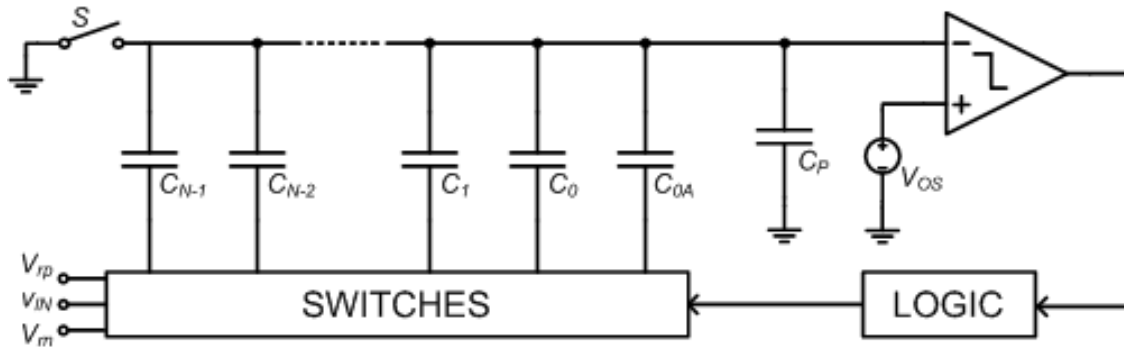


Figure 2.1: Simplified schematic of SAR ADC.

top plates of the capacitors to either positive reference (V_{rp}) or negative reference (V_{rn} , $V_{rn} = 0$ without a loss of generality) performing charge redistribution at the bottom plates of the capacitors and, in combination with SAR logic, effectively creating a series of reference voltages that input signal is compared to using the comparator. For example, in the first bit-testing phase, C_{N-1} is connected to V_{rp} and all other capacitors to V_{rn} . This way the input signal is compared to $\frac{C_{N-1}}{C_{act}}V_{rp}$, where C_{act} is the sum of all the capacitors in the array. If the input signal is larger than $\frac{C_{N-1}}{C_{act}}V_{rp}$, in the second phase it is compared to $\frac{C_{N-1}+C_{N-2}}{C_{act}}V_{rp}$ by connecting C_{N-2} to V_{rp} . If it is smaller, then it is compared to $\frac{C_{N-2}}{C_{act}}V_{rp}$ by connecting C_{N-2} to V_{rp} and C_{N-1} to V_{rn} . This process continues until all the bits are resolved. In order to clarify the conversion process, an example of a 4-bit conversion with $\alpha = 2$ and the input voltage of $V_{in} = \frac{19}{32}V_{rp}$ is presented next. Right after the sampling phase the capacitor C_3 is connected to V_{rp} and all other capacitors are connected to V_{rn} , as shown in Figure 2.3.a). After the DAC settling is completed, the voltage at the negative comparator input

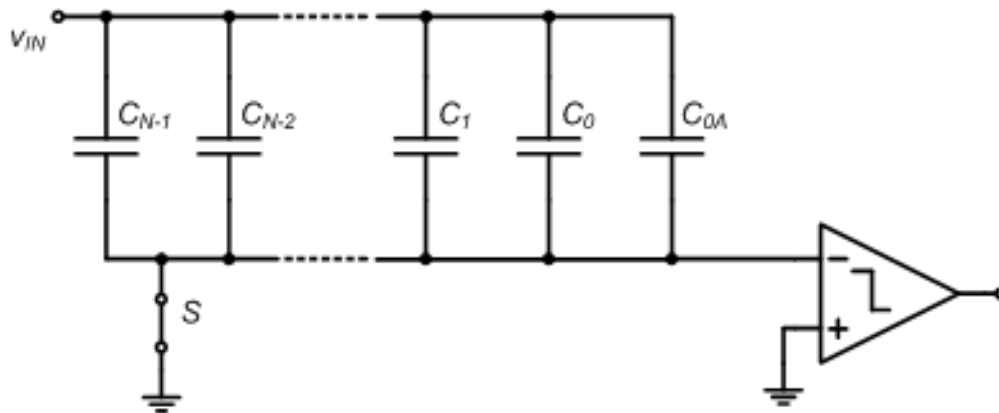


Figure 2.2: Sampling phase of SAR ADC.

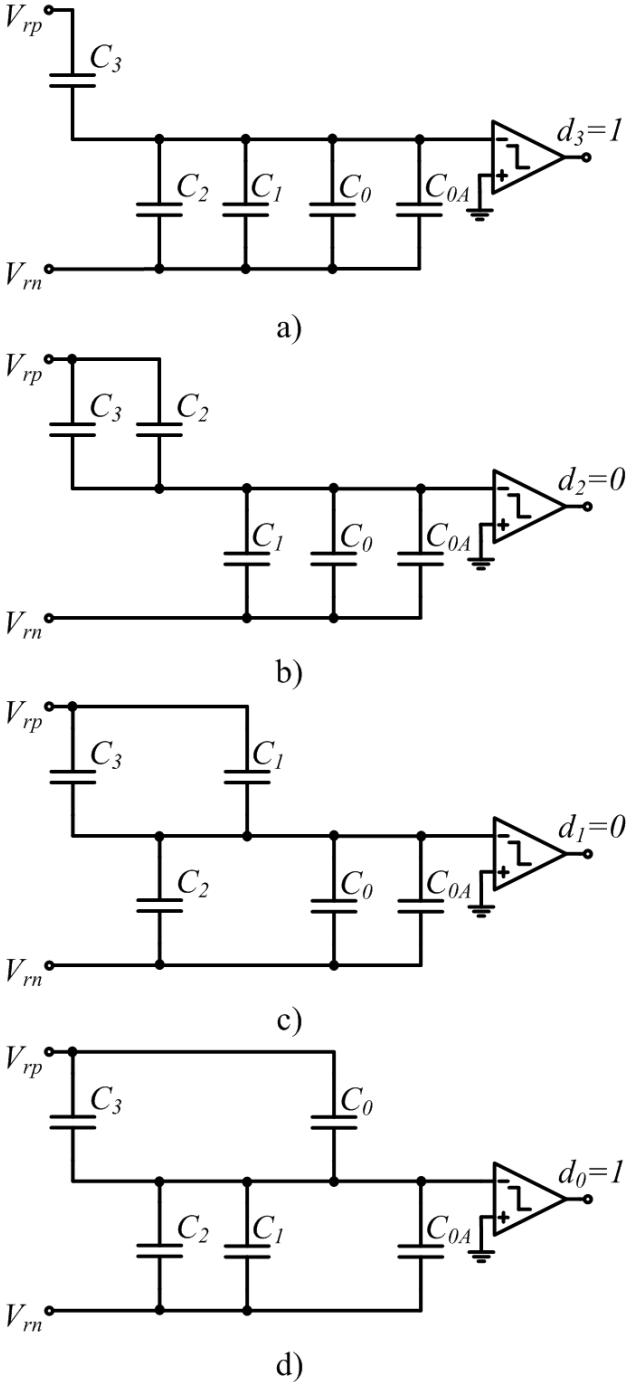


Figure 2.3: 4-bit conversion example.

is $\frac{C_{act}}{C_{tot}}(\frac{1}{2}V_{rp} - V_{in})$, where C_{tot} is the sum of all capacitors, including the comparator input parasitic capacitance. Therefore, the comparator compares V_{in} to $V_{r1} = \frac{1}{2}V_{rp}$. In our case, $V_{in} > \frac{1}{2}V_{rp}$, and the comparator output is equal to one. In the next phase C_3 stays connected to V_{rp} since the previous comparator decision was one, and C_2 is also connected to V_{rp} , as in Figure 2.3.b). This way the input signal is compared to $V_{r2} = \frac{3}{4}V_{rp}$. Since $V_{in} < V_{r2}$, the output of the comparator is equal to zero. Consequently, C_2 is connected to V_{rn} and C_1 is connected to V_{rp} , as shown in Figure 2.3.c) to effectively create a new reference level $V_{r3} = \frac{5}{8}V_{rp}$. V_{in} is smaller than V_{r3} causing the comparator to produce a zero. Finally, in the last step, C_1 is connected to V_{rn} and C_0 to V_{rp} , as shown in Figure 2.3.d). The last reference to which the input signal is compared to is equal to $V_{r4} = \frac{9}{16}V_{rp}$. $V_{in} > V_{r4}$, which produces a one at the output of the comparator for the final conversion output of 1001.

In the most convenient case, when $\alpha = 2$, the SAR ADC performs a binary search algorithm, and the saved outputs of the comparator represent the final conversion output. Otherwise, it performs a radix-based search, and the saved bits from the comparator output need to be weighted and summed up to get the final conversion output. Even though radix-2 SAR ADCs avoid the need for the digital summation logic, a reduced-radix architecture ($\alpha < 2$) is used in this work. The reasoning behind this decision is explained in the next section.

2.2 Static Nonlinearities

The transfer function or transfer characteristic of an ADC is a function that assigns a digital code to the analog value of the input signal. Ideally, in the case of a perfect radix-2 SAR ADC, the input signal range (from V_{rn} to V_{rp}) is divided into 2^N equal segments and each segment is assigned a unique digital code from 0 to $2^N - 1$ in ascending order, so that lower digital codes correspond to the smaller input analog voltage. In practical implementations a perfect radix two can never be achieved due to capacitor mismatches. Sufficiently small mismatches can be obtained if large capacitors are used, but that comes with power and area penalty. It is often beneficial to use minimum-sized capacitors dictated by the thermal noise requirements, and to allow for bigger mismatches if they can be somehow corrected. These mismatches will create deviations from the ideal transfer function. To examine these deviations, it is useful to study the transfer functions of the SAR ADCs that have a radix higher and smaller than two.

The first step in obtaining a transfer characteristic of a SAR ADC is to plot the transfer function of its capacitive DAC and construct a staircase-shaped curve, as shown in Figure 2.4 for the case of a 6-bit ADC with radix 2.2 (a) and 1.8 (b). Digital codes on the vertical axis are the inputs and the outputs are formed as the weighted sum of the bits in the binary representation of the input digital code. In the case of the radix-2.2 ADC, for any analog input signal there is only one output digital code. Therefore, the transfer function of the SAR ADC is identical to the curve constructed from the DAC transfer function. The situation

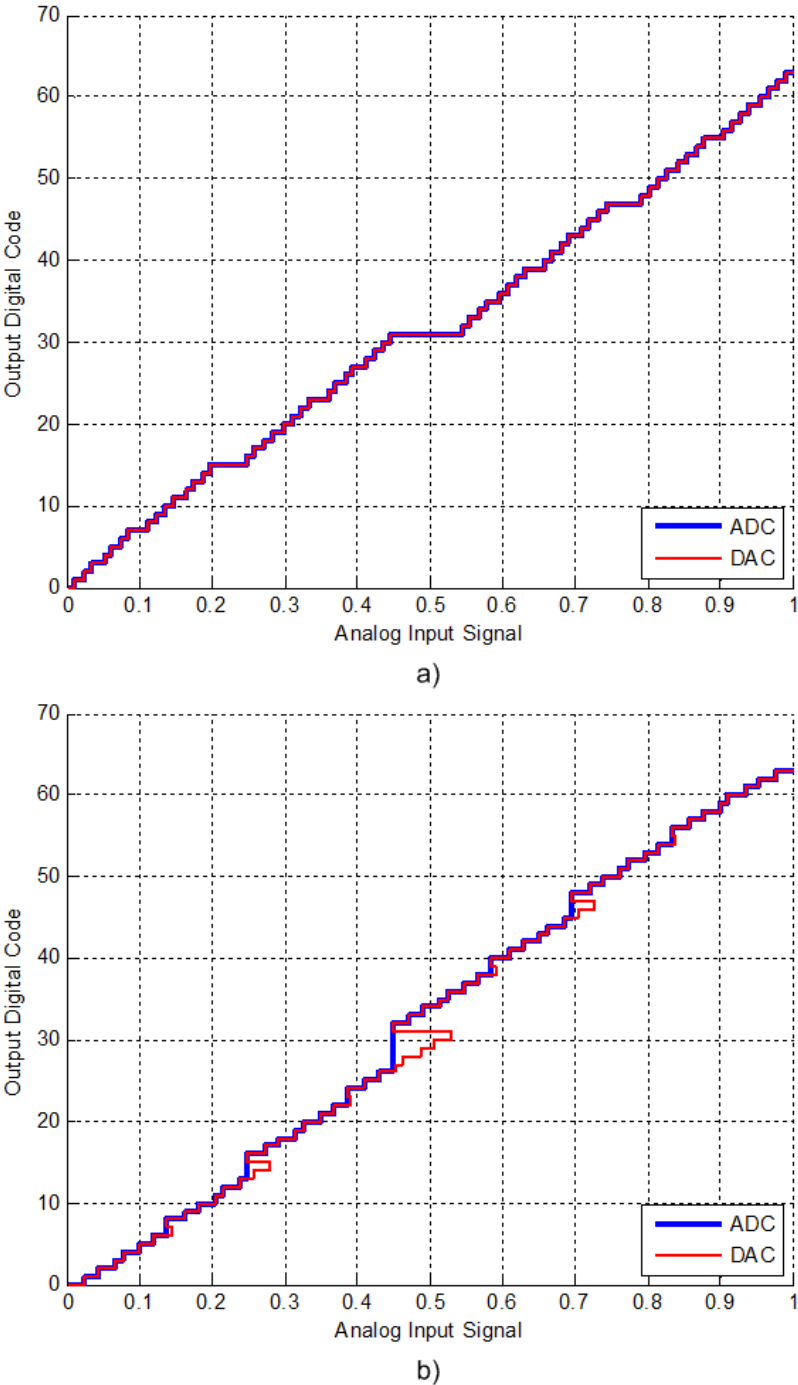


Figure 2.4: Constructing transfer functions of SAR ADC for a) radix 2.2 and b) radix 1.8.

is more complex in the case of radix 1.8. For some input signal (e.g. 0.5 in Figure 2.4) there are more than one output codes. Since the MSB bits are resolved first during the conversion process, the largest code will be produced as the output for a given input signal. The constructed SAR ADC transfer function is shown in Figure 2.4.b).

The deviations from the ideal transfer function that create nonlinearities manifest as large horizontal segments in the case of radix greater than two, and as large vertical segments in the case of radix less than two. The large horizontal and vertical segments are known as missing decision levels and missing output codes, respectively. The segmentation of the input signal range (horizontal axis) is completely determined by the sizes of the capacitors in the capacitive DAC. Therefore, the missing decision levels can be corrected only in analog domain by changing the values of the capacitors. The missing output codes can be corrected in digital domain by assigning different weight coefficient to the output bits. This is the reason why the reduced radix is needed if digital calibration is to be used. As shown in [25], in order to have a transfer function without missing decision levels, it is sufficient that the following condition is satisfied:

$$C_i < C_{0A} + \sum_{j=0}^{i-1} C_j, \text{ for } i = 1..N - 1. \quad (2.2)$$

2.3 Basics of Time-Interleaving

Time-interleaving of A/D converters is a technique used to achieve sampling speeds that would not be realizable with a single converter or would be prohibitively power-inefficient. A simple block diagram of a time-interleaved converter with an interleaving factor of M is shown in Figure 2.5. It consists of M converters (channels) that sample the input signal at the same frequency f_s/M , but with clocks that are equally phase-shifted. The phase of the i^{th} clock is $\frac{2\pi(i-1)}{M}$ rad. The outputs of the ADC channels are multiplexed to form a stream of the output data that correspond to the input signal sampled at the frequency f_s . If all ADC channels are identical, this time-interleaved ADC is equivalent to a single-channel ADC sampling at f_s . Unfortunately, in practical implementation the interleaved channels can have different offsets, gains and bandwidths, and the phases of the sampling clocks are not necessarily equidistant. The effects of these nonidealities on the spectrum of the output signal are analyzed in the following sections.

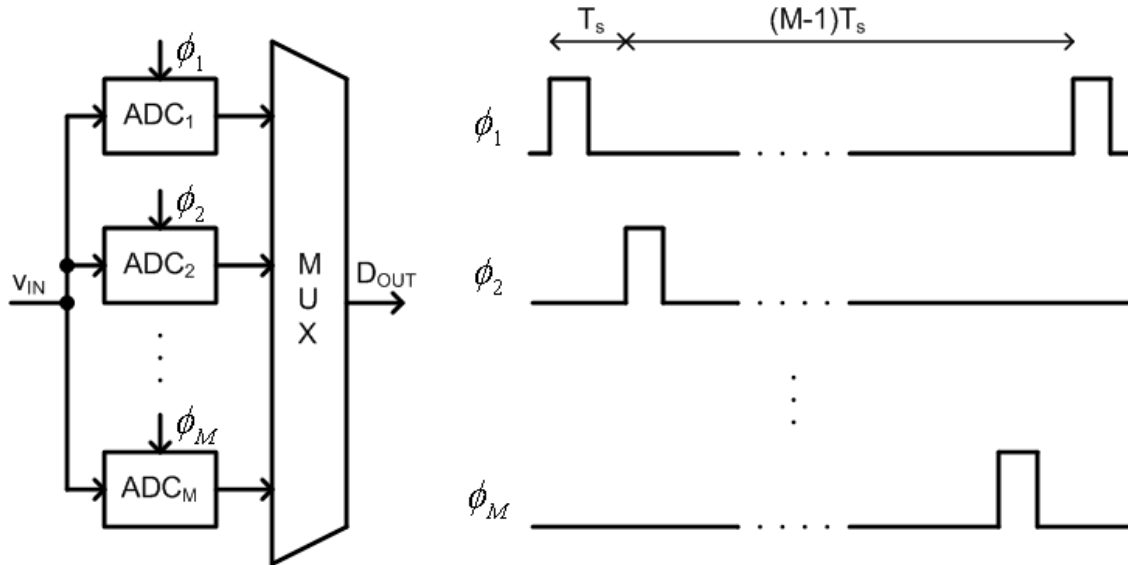


Figure 2.5: Basic block diagram of a time-interleaved ADC.

2.4 Offset Mismatch

An ADC offset is a random additive error typically coming from the comparator offset. In a single-channel ADC the offset error creates a DC tone that can be easily removed and is often ignored in many communication applications. The impact of the offset errors is much more detrimental in time-interleaved ADCs. If $o(i)$ is the offset of the i^{th} channel, then, for a given input signal $v_{in}(t)$, and assuming no other errors, the output signal can be written as:

$$D_{out}(n) = v_{in}(nT) + o((n-1) \bmod M + 1), \quad (2.3)$$

where mod is a modulo operation. $o((n-1) \bmod M + 1)$ is a periodic discrete signal with a period equal to M . This means that in addition to our desired signal $v_{in}(t)$, the spectrum of the output signal will have tones at frequencies that are multiples of $\frac{f_s}{M}$. The magnitude and relative strength of these tones depends on the amplitude and the shape of the introduced periodic error signal. An example of the output spectrum of a time-interleaved ADC with offset mismatches only is shown in Figure 2.6 for a 4-way time-interleaved ADC.

2.5 Gain Mismatch

Gain errors manifest itself as a change in the slope of the transfer function of an ADC. The gain error can come from a difference in reference voltages or from the sampling operation

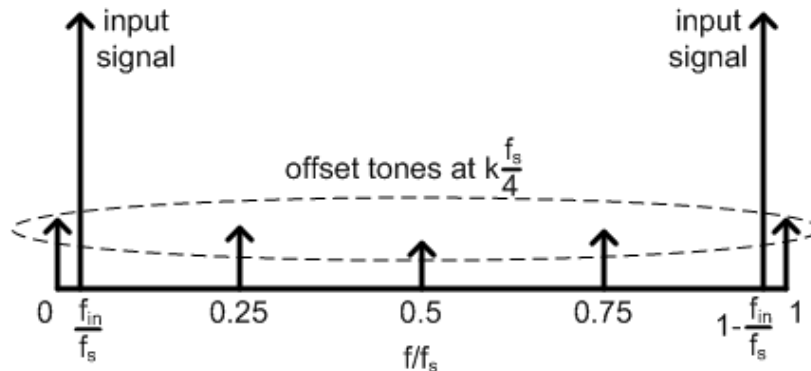


Figure 2.6: Spectrum of a 4-time interleaved ADC with offset mismatches only.

(e.g. charge injection). The gain of the i^{th} channel can be expressed as $1 + \Delta g_i$, where Δg_i is the gain error in the i^{th} channel. The composite output of the time-interleaved ADC can be written as:

$$D_{out}(n) = v_{in}(nT) + \Delta g((n-1) \bmod M + 1)v_{in}(nT). \quad (2.4)$$

$\Delta g((n-1) \bmod M + 1)$ is a periodic discrete signal with a period of M and can be represented in frequency domain by discrete tones at frequencies $\frac{kf_s}{M}$, $k = 0..M-1$. If the input signal is a sinusoid with the frequency f_{in} , the mixing effect of multiplying the input signal with the periodic signal $\Delta g((n-1) \bmod M + 1)$ will create tones at frequencies $\frac{kf_s}{M} \pm f_{in}$. An example of the output spectrum with gain mismatches in the case of a 4-way time-interleaved ADC is shown in Figure 2.7.

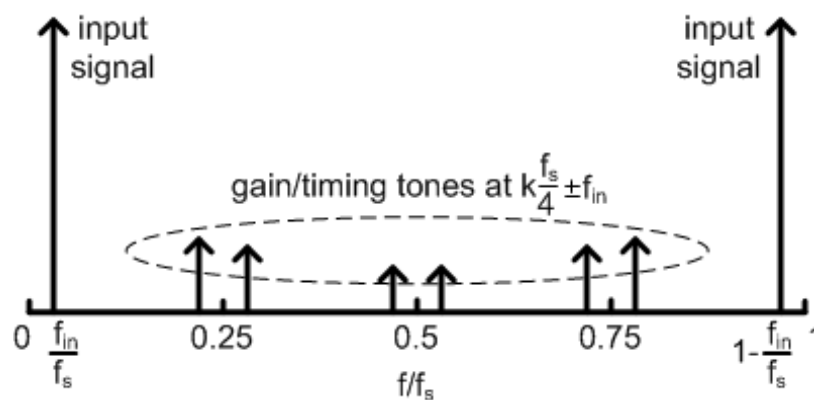


Figure 2.7: Spectrum of a 4-time interleaved ADC with gain and/or timing mismatches.

2.6 Timing Mismatch

The phase difference between the clocks of the neighboring channels should ideally be equal to $\frac{2\pi}{M}$. The phase (or, equivalently, timing) errors are unavoidable in a practical implementation due to finite propagation of the clock signal and variations in the clock buffers and sampling switches. At high input signal frequencies even small timing mismatches can create significant errors. If we denote the timing mismatch in the i^{th} channel by $\Delta t(i)$, then, for a sinusoidal input signal, the output signal can be expressed as:

$$D_{out}(n) = \cos(\omega nT + \omega \Delta t((n-1) \bmod M + 1)). \quad (2.5)$$

The input signal effectively becomes phase modulated by the periodic signal $\omega \Delta t((n-1) \bmod M + 1)$ with period M that has spectral components at $\frac{kf_s}{M}$. Therefore, the spectrum of the output signal will have the undesired spurs at $\frac{kf_s}{M} \pm f_{in}$, as shown in Figure 2.7. The location of the spurs is same as the location of the spurs that stem from the gain mismatches. Unlike in the case of the gain mismatch, the magnitude of the spurs depends on the input frequency. This gives a way of isolating the gain mismatches by performing a low-frequency testing, where the artifacts due to timing mismatches are not visible.

2.7 Bandwidth Mismatch

A finite bandwidth of the analog front-end produces different gain (attenuation) and phase shift at different frequencies. Having different bandwidth in different channels of a time-interleaved ADC will therefore create a frequency-dependent gain and timing mismatches. The dependence of the amount of the gain and timing mismatches on the nominal value of the bandwidth of the analog front-end can be easily analyzed in the case of a single-pole system. The nominal gain and phase are given by

$$A(\omega) = \frac{1}{\sqrt{1 + \left(\frac{\omega}{\omega_0}\right)^2}} \quad (2.6)$$

and

$$\varphi(\omega) = -\arctan \frac{\omega}{\omega_0}, \quad (2.7)$$

where ω and ω_0 are the input frequency and the bandwidth, respectively. The relative gain error due to the bandwidth mismatch can be calculated as:

$$\frac{\Delta A}{A}(\omega) = \frac{\left(\frac{\omega}{\omega_0}\right)^2}{1 + \left(\frac{\omega}{\omega_0}\right)^2} \frac{\Delta \omega_0}{\omega_0}. \quad (2.8)$$

The timing error due to the bandwidth mismatch is:

$$\Delta t(\omega) = \frac{1}{\omega_0} \frac{1}{1 + \left(\frac{\omega}{\omega_0}\right)^2} \frac{\Delta\omega_0}{\omega_0}. \quad (2.9)$$

The error signal can be expressed as:

$$e(t) = (A + \Delta A) v_{in}(t + \Delta t) - A v_{in}(t) \approx \Delta A v_{in}(t) + A \frac{\partial v_{in}(t)}{\partial t} \Delta t. \quad (2.10)$$

For a sinusoidal input signal the signal-to-distortions ratio (SDR) due to the bandwidth mismatch can be calculated as:

$$SDR_{BW} = \frac{1}{\left(\frac{\Delta A}{A}\right)^2 + \omega^2 \Delta t^2}. \quad (2.11)$$

By substituting the expressions for $\frac{\Delta A}{A}$ and Δt from (2.8) and (2.9), and assuming a large number of time-interleaved channels, the expression for SDR_{BW} on a dB scale becomes:

$$SDR_{BW}[dB] = -10 \log \frac{\left(\frac{\omega}{\omega_0}\right)^2}{1 + \left(\frac{\omega}{\omega_0}\right)^2} \sigma_{\frac{\Delta\omega_0}{\omega_0}}^2, \quad (2.12)$$

where $\sigma_{\frac{\Delta\omega_0}{\omega_0}}^2$ is the standard deviation of the relative bandwidth mismatch. The dependence of SDR_{BW} on the relative input frequency ω/ω_0 is plotted in Figure 2.8. As it can be seen, to maintain the SDR better than 60dB with the bandwidth mismatch of 1%, 2% and 4%, the bandwidth has to be approximately 10, 20 and 40 times larger than the largest input signal frequency, respectively. It can be very difficult to achieve this kind of bandwidth in ADCs that sample at multiple GHz frequencies.

Since the bandwidth mismatch creates frequency-dependent gain and timing errors, the timing and gain calibration cannot eliminate these errors at all frequencies. However, they can correct it completely at a given input frequency. This will effectively change the shape of the SDR_{BW} dependency on the relative input frequency.

After the gain and timing calibration at the input frequency ω_{cal} the new effective gain will be given by

$$\begin{aligned} A_{cal}(\omega) &= (A(\omega) + \Delta A(\omega)) \frac{A(\omega_{cal})}{A(\omega_{cal} + \Delta A(\omega_{cal}))} \\ &\approx A(\omega) + \Delta A(\omega) - A(\omega) \frac{\Delta A(\omega_{cal})}{A(\omega_{cal})}. \end{aligned} \quad (2.13)$$

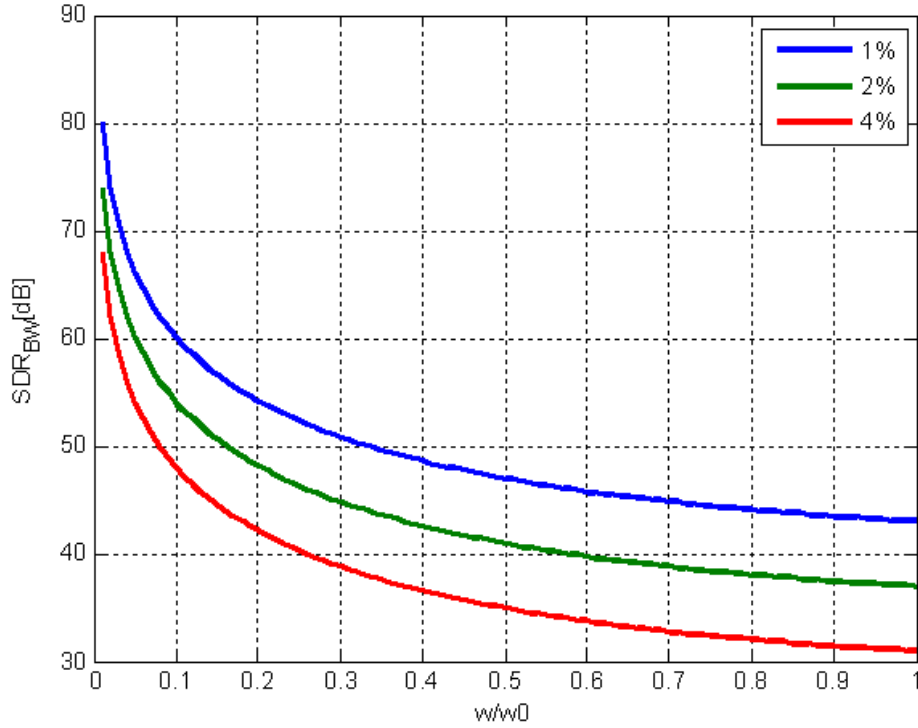


Figure 2.8: SDR due to bandwidth mismatch vs. relative frequency.

The new relative gain error is

$$\frac{\Delta A_{cal}(\omega)}{A} = \frac{A_{cal}(\omega) - A(\omega)}{A(\omega)} = \frac{\Delta A(\omega)}{A(\omega)} - \frac{\Delta A(\omega_{cal})}{A(\omega_{cal})} \quad (2.14)$$

or

$$\frac{\Delta A_{cal}(\omega)}{A} = \frac{\left(\frac{\omega}{\omega_0}\right)^2 - \left(\frac{\omega_{cal}}{\omega_0}\right)^2}{\left(1 + \left(\frac{\omega}{\omega_0}\right)^2\right) \left(1 + \left(\frac{\omega_{cal}}{\omega_0}\right)^2\right)} \frac{\Delta\omega_0}{\omega_0}. \quad (2.15)$$

The new timing error is

$$\Delta t_{cal}(\omega) = \Delta t(\omega) - \Delta t(\omega_{cal}) = \frac{1}{\omega_0} \frac{\left(\frac{\omega_{cal}}{\omega_0}\right)^2 - \left(\frac{\omega}{\omega_0}\right)^2}{\left(1 + \left(\frac{\omega}{\omega_0}\right)^2\right) \left(1 + \left(\frac{\omega_{cal}}{\omega_0}\right)^2\right)} \frac{\Delta\omega_0}{\omega_0}. \quad (2.16)$$

Finally, the effective SDR_{BW} after calibration at the frequency ω_{cal} can be calculated by substituting (2.15) and (2.16) into (2.11):

$$SDR_{BW,cal} = -10 \log \left[\frac{\left(\left(\frac{\omega}{\omega_0} \right)^2 - \left(\frac{\omega_{cal}}{\omega_0} \right)^2 \right)^2}{\left(1 + \left(\frac{\omega}{\omega_0} \right)^2 \right) \left(1 + \left(\frac{\omega_{cal}}{\omega_0} \right)^2 \right)^2 \sigma_{\frac{\Delta\omega_0}{\omega_0}}^2} \right]. \quad (2.17)$$

The new dependence of the SDR on the relative input frequency is shown in Figure 2.9 for the standard deviation of the bandwidth mismatch of 1%, 2% and 4%. The calibration frequency, ω_{cal} , is chosen such that the SDR at $\omega = 0$ is equal to 60 dB. It can be seen that the bandwidth requirements to achieve the SDR of 60 dB are significantly reduced. In the case of 1%, 2% and 4% bandwidth mismatch, the bandwidth needs to be approximately 2, 3 and 4.4 times larger than the highest input frequency, compared to 10, 20 and 40 times in the case with no gain and timing calibration.

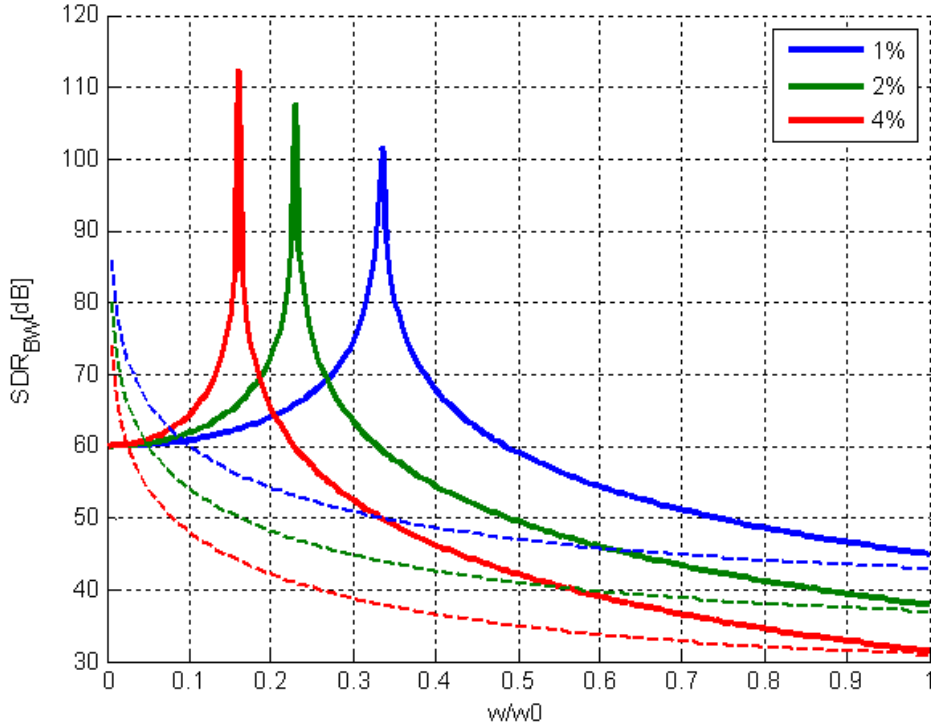


Figure 2.9: SDR due to bandwidth mismatch vs. relative frequency after timing and gain calibration.

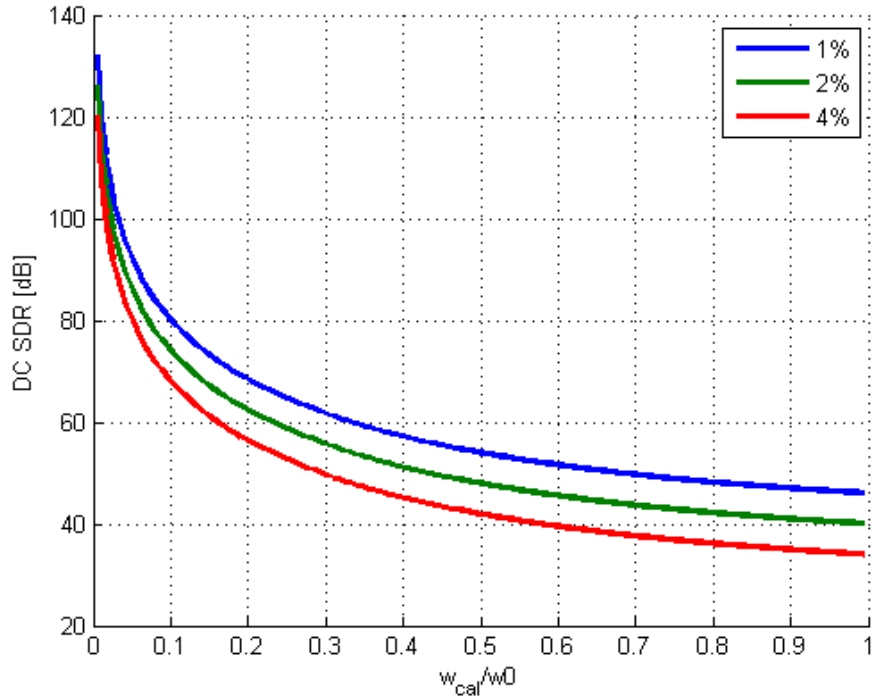


Figure 2.10: SDR at DC vs. relative calibration frequency after timing and gain calibration.

To achieve a high bandwidth, the value of the calibration frequency, ω_{cal} , should be the highest that still satisfies the SDR requirements at low frequencies. The value of the SDR at DC can be obtained by setting $\omega = 0$ in (2.17):

$$SDR_{BW,cal}(0) = -10 \log \left[\frac{\left(\frac{\omega_{cal}}{\omega_0}\right)^4}{\left(1 + \left(\frac{\omega_{cal}}{\omega_0}\right)^2\right)^2} \sigma_{\frac{\Delta\omega_0}{\omega_0}}^2 \right]. \quad (2.18)$$

This dependence of the SDR value at DC on the calibration frequency is shown in Figure 2.10 for the bandwidth mismatch of 1%, 2% and 4%. The new relative cut-off frequency can be defined as the frequency at which the SDR drops to its DC value, and it can be calculated by setting (2.17) to be equal to (2.18):

$$\frac{\omega_{cutoff}}{\omega_0} = \sqrt{\left(\frac{\omega_{cal}}{\omega_0}\right)^4 + 2 \left(\frac{\omega_{cal}}{\omega_0}\right)^2}. \quad (2.19)$$

The new relative cut-off frequency is independent of the bandwidth mismatch and its dependence on the calibration frequency is plotted in Figure 2.11.

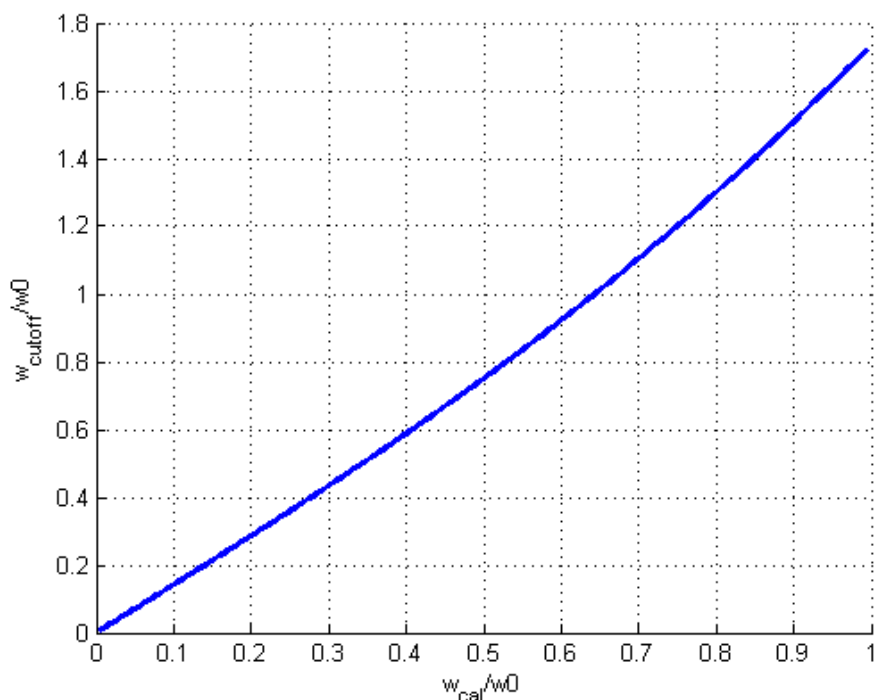


Figure 2.11: Relative cutoff frequency vs. relative calibration frequency after timing and gain calibration.

2.8 Finite Sampling Aperture

The bandwidth mismatch analysis from the previous section assumed that the transfer function of the analog front-end is same as the continuous transfer function of the underlying analog RC circuit $H(j\omega) = \frac{1}{1+j\omega/\omega_0}$. This is not entirely true since the analog front-end of an ADC is a sampled circuit, and it becomes noticeable when the timing constant of the RC circuit is comparable to the width of the sampling aperture.

The effect of the finite sampling aperture can be easily analyzed in the case of the simple RC sampling circuit. For a sinusoidal input signal the sampled voltage in the time domain

can be expressed as

$$\begin{aligned}
 v_{out}(t) = & \frac{1}{\sqrt{1 + \left(\frac{\omega}{\omega_0}\right)^2}} \sin\left(\omega t - \arctan \frac{\omega}{\omega_0}\right) + \\
 & + \left(v_{out}(t - MT) - \frac{1}{\sqrt{1 + \left(\frac{\omega}{\omega_0}\right)^2}} \sin\left(\omega(t - DT) - \arctan \frac{\omega}{\omega_0}\right) \right) e^{-\omega_0 DT},
 \end{aligned} \tag{2.20}$$

where D , T and M are the duty ratio of the sampling clock, the sampling period and the interleaving factor, respectively. The sampling aperture is equal to DT . The exponentially decaying term in (2.20) represents the initial condition on the sampling capacitor. The initial condition depends on the previous sampled signal value and the value of the input signal at the moment of closing of the sampling switch. If the sampling capacitor is reset after every cycle, then $v_{out}(t - MT) = 0$ in (2.20).

In frequency domain, without the capacitor reset, the equivalent transfer function can be derived as

$$\begin{aligned}
 H_{eq}(j\omega) = & H(j\omega) \frac{1 - e^{-\omega_0 DT} e^{-j\omega DT}}{1 - e^{-\omega_0 DT} e^{-j\omega MT}} \\
 \approx & H(j\omega) \left(1 - 2je^{-\omega_0 DT} \sin \frac{\omega(M - D)T}{2} e^{-\frac{j\omega(D+M)T}{2}} \right).
 \end{aligned} \tag{2.21}$$

With the capacitor reset, the equivalent transfer function becomes

$$H_{eq}(j\omega) = H(j\omega) (1 - e^{-\omega_0 DT} e^{-j\omega DT}). \tag{2.22}$$

The new equivalent transfer functions contain an additional term whose magnitude is proportional to $e^{-\omega_0 DT}$. For a target speed of 3 GHz and resolution of 8 effective bits the bandwidth higher than the sampling frequency can be easily achieved. This makes the $e^{-\omega_0 DT}$ term much smaller than one and the overall transfer function practically insensitive to small variations of the sampling aperture.

Chapter 3

Calibration of Static Nonlinearities in SAR A/D Converters

This chapter presents new techniques for calibration of capacitor mismatches in SAR ADCs. The core of the calibration techniques are two different types of switching during conversion - here referred to as direct and reverse switching. The presented calibration techniques can be used for calibration of conventional single-channel SAR ADCs, split-ADCs [30], as well as time-interleaved converters, and can be performed both in analog domain, by using electronic trimming of capacitors, and in digital domain, as an adaptive background postprocessing.

3.1 Overview of Techniques for Linearity Calibration in SAR ADCs

As discussed in Chapter 2, the capacitor mismatches in SAR ADCs create nonlinearities in the ADC transfer function. Many different calibration techniques have been developed in order to reverse the effect of mismatch-caused nonlinearities. Based on the point at which the nonlinearities are corrected, the techniques can be analog or digital.

One group of analog techniques corrects the mismatches by subtracting a signal equal to the error caused by the mismatches from the output of the capacitive DAC. A single calibration DAC combined with a digital logic can be used for this subtraction [23] or every capacitor in the main DAC can have its own calibration DAC [37]. The other group of analog techniques corrects the mismatches by modifying the effective value of the capacitors in the main DAC. This can be achieved by using small trimming capacitors that are connected in parallel with the main DAC capacitors [35]. The mismatches can be measured by using a known precise input signal [6] or by using a self-calibration technique [23]. In the self-calibration technique the difference between each capacitor in the array and the sum of all capacitors at the lower bit-positions is measured and later used for the mismatch correction.

All these techniques are performed in the foreground and require an interruption in the operation of the ADC.

The digital calibration techniques measure or infer the values of the capacitors, represent them as a set of digital coefficients, and then correct the nonlinearities in the digital domain by calculating weighted sums of those coefficients for each conversion. Inherently, the correction is performed in the background, but the process of obtaining the capacitor values can be performed both in the foreground and the background. An example of the foreground calibration can be found in [6] for a resistive DAC or in [8] for a capacitive DAC. The coefficients are obtained by using a known input signal, such as a precise DC voltage, a ramp signal or a sinusoidal signal. The background calibration techniques typically rely on converting the same input signal sample twice in two different ways, or injecting a dithering signal. One way of obtaining two different conversion results is presented by Liu et al. in [24]. An additional accurate reference channel, which samples and converts the input signal together with the ADC that is being calibrated, is introduced. A calibration based on the least-mean-square (LMS) algorithm minimizes the difference between the two conversion results, thus correcting the nonlinearities caused by the capacitor mismatches. This approach requires a design of an additional ADC, which doubles the design effort. In [26] a small capacitor is added to the capacitive array to introduce a perturbation signal. The two conversion results are obtained by converting the same sample twice with different sign of the perturbation signal. The capacitor weights are adaptively calculated from the difference of the two conversion results by using the LMS algorithm. Since every sample is converted twice by a single ADC, the conversion speed is reduced by a factor of two in this approach. Xu et al. [41] used the capacitors from the main DAC to inject a pseudo-random dithering signal and obtained the capacitor weights by correlating the output result with the dithering sequence. Since the capacitors used for dithering cannot be used for conversion, this approach also uses two conversions per sample to avoid estimation errors due to large code gaps. Also, a long pseudo-random sequence is needed for a small estimation error, which makes the calibration time prohibitively long for many applications.

The goal of this research is to develop a fast and low-overhead background calibration technique for time-interleaved converters. The calibration is based on a reference channel, as in [24], but the reference channel is an identical copy of the time-interleaved channels. This way the design of an additional slow and accurate ADC is avoided. Also, since the reference channel can convert the input signal at the same speed as the interleaved channel, the calibration time is significantly reduced compared to the solution in [24]. The developed technique naturally extends to the digital calibration of a single-channel ADCs, as well as the analog calibration of both single-channel and time-interleaved ADCs. All different versions of the calibration algorithm are presented in this chapter. The background necessary for understanding of these calibration algorithms is presented in the next section.

3.2 Direct and Reverse Switching

The basics of the SAR ADC operation have been described in Chapter 2 together with its main building blocks. While Chapter 2 describes only one way of performing the conversion of the input signal, there are at least two different ways. This section describes these two modes of conversion, called direct and reverse switching, that will later be used as a basis for development of the nonlinearity calibration techniques.

After the input signal is sampled onto all capacitors in the array, the first most significant bit (MSB) needs to be resolved. In direct switching, which is usually considered a standard way of performing conversion, the top plate of the MSB capacitor C_{N-1} is connected to the positive reference V_{rp} , while the top plates of all other capacitors are connected to the negative reference V_{rn} as shown in Figure 3.1.a. In the reverse switching C_{N-1} is connected to V_{rn} , while all other capacitors are connected to V_{rp} as shown in 3.1.d. After the input to the comparator has settled the comparison is triggered and the first bit is resolved. If the resolved bit is '1' in the next bit-testing phase C_{N-1} is connected to V_{rp} . If the resolved bit is '0' C_{N-1} is connected to V_{rn} . This is true for both direct and reverse switching. In the next bit-testing phase C_{N-2} is connected to V_{rp} in direct switching, and to V_{rn} in reverse switching as shown in 3.1.b and 3.1.e. After the second bit is resolved C_{N-2} is connected to V_{rp} if the second bit was '1' or to V_{rn} if the second bit was '0' in both direct and reverse switching. This process continues until all N bits are resolved as shown in 3.1.c and 3.1.f.

Assuming $V_{rn} = 0$ for simplicity, the relation between the input voltage and the output bits in the case of direct switching is given by

$$V_{in} = V_{rp} \sum_{i=0}^{N-1} d_{di} \frac{C_i}{C_{act}} - \frac{C_{tot}}{C_{act}} V_{OS} + \frac{C_{tot}}{C_{act}} V_{Qd} \quad , \quad (3.1)$$

where C_{act} is the sum of all capacitors in the array including C_{0A} , $C_{tot} = C_{act} + C_P$, d_{di} and V_{Qd} ($V_{Qd} \geq 0$) are the i^{th} output bit and the quantization error in direct switching conversion, respectively. In the case of reverse switching this relation becomes

$$\begin{aligned} V_{in} &= V_{rp} \left(1 - \sum_{i=0}^{N-1} d_{ri} \frac{C_i}{C_{act}} \right) - \frac{C_{tot}}{C_{act}} V_{OS} - \frac{C_{tot}}{C_{act}} V_{Qr} = \\ &= V_{rp} \left(\sum_{i=0}^{N-1} d_{ri} \frac{C_i}{C_{act}} + \frac{C_{0A}}{C_{act}} \right) - \frac{C_{tot}}{C_{act}} V_{OS} - \frac{C_{tot}}{C_{act}} V_{Qr} \quad , \end{aligned} \quad (3.2)$$

where d_{ri} and V_{Qr} ($V_{Qr} \geq 0$) are the i^{th} output bit and the quantization error in reverse switching conversion, respectively.

The easiest way to understand differences between (3.1) and (3.2) is to plot transfer characteristics for the same converter, but with different types of switching. Figure 3.2.a and 3.2.b show these characteristics for the case of nominal radix-2 and radix-1.8 arrays with

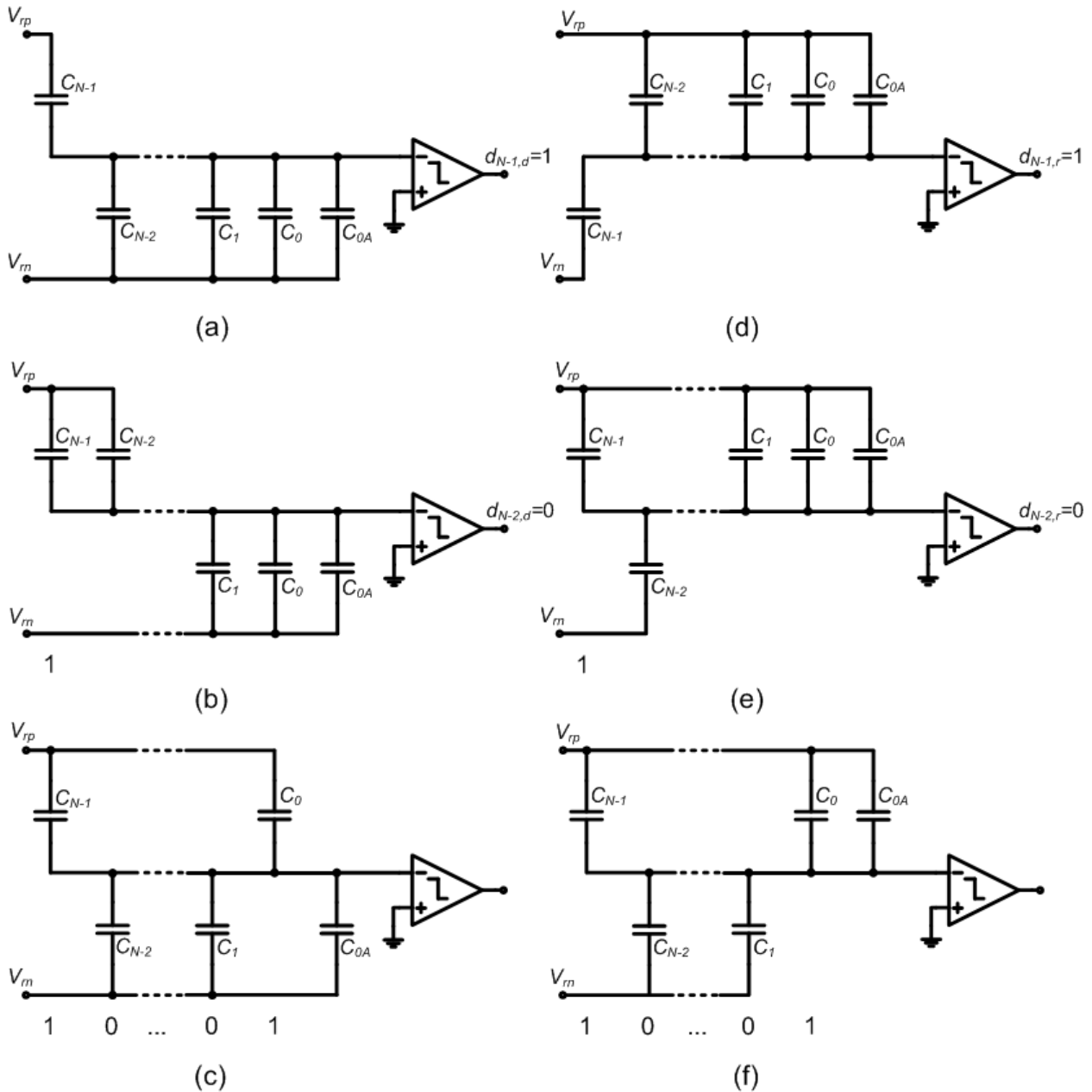
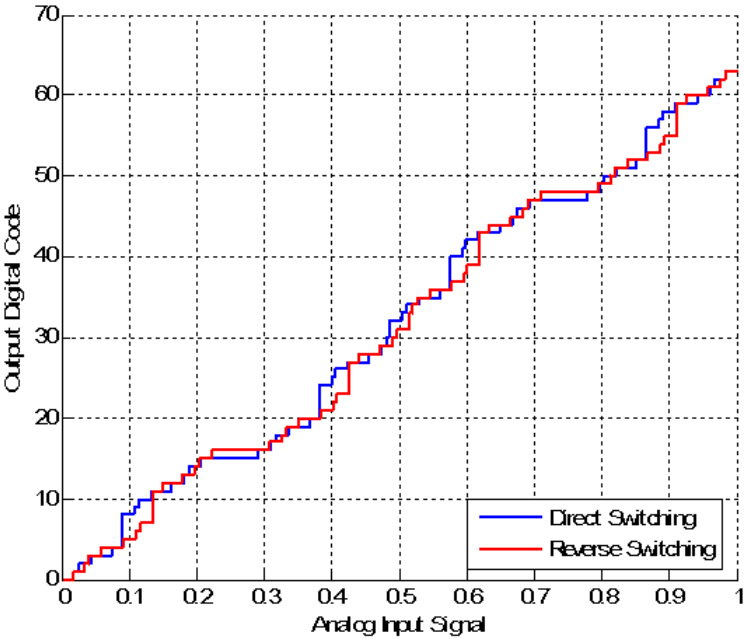
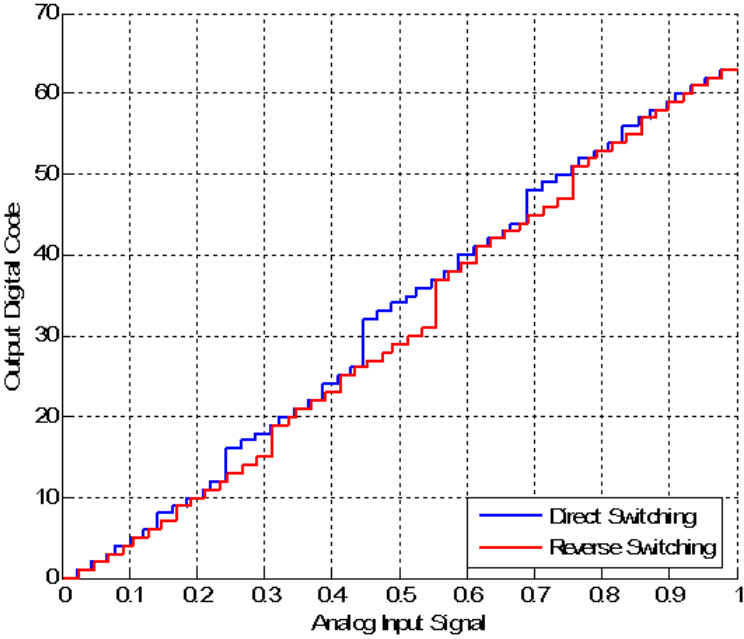


Figure 3.1: Direct and reverse switching in SAR ADC.

capacitor mismatches, respectively. By careful examination of the plots it can be seen that the transfer characteristics are 180° rotated around center with respect to each other. This effect can be explained by noticing that reverse switching conversion of the input signal V_{in} is equivalent to a direct switching conversion of the signal $V_{rp} - V_{in}$, but the output result is



(a)



(b)

Figure 3.2: Transfer characteristics of direct and reverse switching in SAR ADC.

coded in the complement one code, e.g. all bits are inverted. Due to inherent symmetries of the SAR ADC transfer characteristic, these two characteristics can overlap only in the case of perfect radix-2 array.

If the input signal is converted twice using direct and reverse switching, the difference between the two conversion results can be used as an input error signal to the algorithm that will minimize the error by forcing the transfer characteristics to look like the ideal radix-2 curve. This can be done either by adjusting the actual value of the capacitors in the DAC using very small capacitors and switches as in [35] (this technique is called electronic trimming or simply trimming), or by inferring the values of capacitors in digital domain. As explained in Chapter 2, a radix less than two is needed in the case of digital calibration.

3.3 Trimming-Based Calibration

In trimming-based algorithms the physical values of capacitors are adjusted to form the ideal radix-2 capacitive DAC, thus avoiding need for further digital processing of the output bits and the additional latency that would come from the digital processing logic. This could potentially lead to slightly lower power since the calibration logic can be completely turned off once the calibration has converged, but it comes with additional complexity in the analog domain and can be impractical for extremely small capacitors. Trimming-based calibration can be used in single-channel single-core, single-channel dual-core (split ADC), and time-interleaved ADCs. The following sections detail these three cases.

3.3.1 Single-Channel Single-Core SAR ADC Calibration

After the input signal is sampled in a SAR ADC its value is stored as a charge on the bottom plates of the DAC capacitors and the top plates of the parasitic capacitances at the input of the comparator (C_P). Assuming the leakage is negligible, the sampled input signal can be converted multiple times. In our case every input sample can be converted using direct and reverse switching, and the difference between conversion results can be used to tune values of the capacitors. The final output is obtained by averaging the outputs of two conversions. To derive an algorithm for capacitor tuning we observe that the goal of the calibration is to minimize the quantization error. From (3.1) and (3.2) the sum of quantization errors V_{Qd} and V_{Qr} can be derived as:

$$V_{Qr} + V_{Qd} = \frac{C_{act}}{C_{tot}} V_{rp} \sum_{i=0}^{N-1} (d_{ir} - d_{id}) \frac{C_i}{C_{act}} + \frac{C_{0A}}{C_{tot}} V_{rp} \quad . \quad (3.3)$$

Since both quantization errors are positive we need to minimize the expression on the right side of (3.3). This can be done adaptively by calculating the gradient of $V_{Qr} + V_{Qd}$ with respect to the vector of the capacitor values and moving in the opposite direction. Therefore, the calibration algorithm can be summarized in the following four steps:

1. Sample the input signal.
2. Perform direct and reverse switching conversions and obtain the digital bits $\{d_{ir}\}$ and $\{d_{id}\}$.
3. Update the values of the capacitors in the capacitive DAC:

$$C_i \leftarrow C_i + (d_{id} - d_{ir}) \Delta C \quad . \quad (3.4)$$

ΔC is the value of the small trimming capacitor, $i = 0..N - 1$.

4. Go to the step 1.

3.3.2 Single-Channel Dual-Core SAR ADC Calibration

Single-channel dual-core ADC is similar to the split-ADC concept described in [30]. The input signal is sampled into two SAR ADCs (two cores) simultaneously. The two cores convert the input signal and the final output is obtained by averaging the outputs of the two cores. Both cores can have different capacitor mismatches and different offsets, and we need to make sure that after the calibration their transfer characteristics are both equal and linear. SAR ADCs do not exhibit the gain mismatch problem as long as all channels use the same reference voltages [27]. The calibration technique previously described for single-channel single-core SAR ADC can be easily extended to the case of single-channel dual-core ADC. Instead of having each input sample converted twice with a single core now we have each input sample converted twice with two different cores. To avoid the situation where both converters are equal but nonlinear, it is sufficient to have one of the cores alternate randomly between direct and reverse switching . Also, the offset of the comparator has to be adjusted in the way that reduces the error. Many different techniques for offset adjustment [40], [32], [4] are available and can be used in this algorithm as well. The calibration algorithm can be summarized as follows:

1. Sample the input signal into two cores.
2. In the first core randomly choose the type of switching. The second core can always perform the same kind of switching or have it chosen randomly independent of the first

Table 3.1: Operators \oplus and \ominus for different types of switching.

SAR1	SAR2	\oplus	\ominus
Direct	Direct	+	-
Direct	Reverse	+	+
Reverse	Direct	-	-
Reverse	Reverse	-	+

core. Perform the conversions and obtain the digital bits $\{d_i^1\}$ and $\{d_i^2\}$, where the subscript index denotes the bit position and the superscript index is the core number.

3. Update the capacitor values according to the following equations:

$$\begin{aligned} C_i^1 &<= C_i^1 \oplus (d_i^1 - d_i^2) \Delta C \\ C_i^2 &<= C_i^2 \ominus (d_i^1 - d_i^2) \Delta C \quad . \end{aligned} \quad (3.5)$$

The operations \oplus and \ominus are dependent on the type of switching and are defined in Table 3.1.

4. Update the offset values according to the following equations:

$$\begin{aligned} V_{OS}^1 &<= V_{OS}^1 - \text{sgn}(D^1 - D^2) \Delta V_{OS} \\ V_{OS}^2 &<= V_{OS}^2 + \text{sgn}(D^1 - D^2) \Delta V_{OS} \quad , \end{aligned} \quad (3.6)$$

where D^i is the conversion output of the i^{th} channel.

5. Go to the step 1.

3.3.3 Multi-Channel SAR ADC Calibration

In a multi-channel time-interleaved architecture, M channels sample the input signal at a frequency of f_s/M and phase shift of $2\pi k/M$, where f_s is the aggregate sampling frequency and k is the channel number. The algorithm used for split-ADC can be easily generalized to this architecture if we introduce one additional ADC channel (reference channel). This ADC can be identical copy of all other channels and needs to sample input signal at the frequency $f_s/(M+1)$ and with phase that is aligned to other channels. This way the first sample of the additional channel is aligned with the first channel, the second sample is aligned with the second channel, and so on. After k^{th} sample of the reference channel the capacitor and offset values of the the channel number k and the reference channel itself are updated. The algorithm can be summarized as follows:

1. Initialize $i = 0$.
2. Sample the input signal into the reference channel and the channel number $k = i \bmod M + 1$ (mod is modulo operation).
3. Randomly choose the type of switching in the reference channel. The type of switching in the k^{th} channel can always be the same or randomly chosen independently of the reference channel. Perform the conversions and obtain the digital bits $\{d_i^0\}$ and $\{d_i^k\}$, where the subscript index denotes the bit position and the superscript index is the core number. The superscript 0 denotes the reference channel.

4. Update the capacitor values according to the following equations:

$$\begin{aligned} C_i^0 &\leq C_i^0 \pm_1 (d_i^0 - d_i^k) \Delta C \\ C_i^k &\leq C_i^k \pm_2 (d_i^0 - d_i^k) \Delta C \quad . \end{aligned} \quad (3.7)$$

5. Update the offset values according to the following equations:

$$\begin{aligned} V_{OS}^0 &\leq V_{OS}^0 - \text{sgn}(D^0 - D^k) \Delta V_{OS} \\ V_{OS}^k &\leq V_{OS}^k + \text{sgn}(D^0 - D^k) \Delta V_{OS} \quad . \end{aligned} \quad (3.8)$$

6. Update $i = i + 1$. Go to the step 2.

3.4 Digital Background Calibration

In many cases electronic trimming of capacitors can be impractical due to a multitude of reasons. Practical limits in the size of the trimming capacitors, which need to be multiple times smaller than the LSB capacitor, can limit the usability of the trimming calibration techniques in the low- to moderate-resolution ADCs. Also, estimation of the necessary trimming range in the design phase can be a difficult problem. Lastly, the wiring and layout overhead of adding many small capacitors and switches can be unacceptable in some cases.

In order to avoid all the limitations of the trimming calibration, the background digital calibration can be used instead. In digital calibration a digital weight coefficient C_{di} is assigned to every capacitor C_i in the capacitive DAC, and a digital offset V_{OSd} is assigned to every comparator offset V_{OS} . The final conversion output is calculated as a weighted sum of the output bits

$$D = \sum_{i=0}^{N-1} C_{di} d_i + V_{OSd} \quad . \quad (3.9)$$

The offset term is optional in a single-channel single-core configuration. The calibration task is to figure out the values of the digital coefficients and offsets. This is often done using adaptive algorithms such as the LMS algorithm.

It is important to note that the digital calibration linearizes the transfer characteristic of an ADC by assigning different output values to segments of input signal range created by physical values of the capacitors. Since we are not changing the capacitor values, this division of the input signal range into small segments cannot be changed. In radix-2 ADCs some of these segments can be much larger than the intended least significant bit (LSB) size creating big errors in the conversion result. Therefore these errors (known as missing decision levels in ADC literature) cannot be corrected with digital calibration. To ensure that our starting ADC transfer characteristic is free of missing decision levels, a radix less than two should be used in the SAR ADC [25]. Exact value of the radix depends on the expected level of random variation and systematic layout mismatches.

3.4.1 Single-Channel Single-Core SAR ADC Calibration

The linearization of a SAR ADC characteristic can be done by minimizing the difference between the conversion results in direct and reverse switching. This difference can be used as an error signal which is fed into an LMS algorithm that adaptively calculates the values of the weight coefficients C_{di} . The error signal is given by

$$e = \sum_{i=0}^{N-1} C_{di} (d_{di} - d_{ri}) \quad . \quad (3.10)$$

By applying the LMS algorithm, the update equations for digital coefficients can be written as

$$\begin{aligned} C_{di,j+1} &= C_{di,j} - \mu \frac{\partial (e^2)}{\partial (C_{di})} = \\ &= C_{di,j} - 2\mu e (d_{di} - d_{ri}) \quad . \end{aligned} \quad (3.11)$$

$C_{di,j}$ is the i^{th} digital weight coefficient at the j^{th} time step of adaptation, and μ is the LMS coefficient constant. If μ is a power of two, the multiplication by μ can be realized as a simple shift operation and does not require any dedicated hardware. The multiplications by d_{di} and d_{ri} can be realized as logic AND operation since d_{di} and d_{ri} are digital bits that can be either zero or one. Therefore, both (3.9) and (3.11) can be realized in hardware using only additions, subtractions and logic AND operations, which makes it suitable for a low-power hardware implementation.

Finally, the algorithm can be outlined as follows:

1. Initialize digital weight coefficients:

$$C_{di,0} = \alpha^i \quad , \quad i = \overline{0..N-1} \quad . \quad (3.12)$$

2. Sample the input signal.
3. Perform direct and reverse switching conversion and obtain the digital bits $\{d_{di}\}$ and $\{d_{ri}\}$.
4. Calculate the error signal using (3.10).
5. Update the digital coefficients using (3.11).
6. Go to the step 2.

3.4.2 Single-Channel Dual-Core SAR ADC Calibration

Same as in the case of the trimming calibration, the two cores in the single-channel dual-core architecture sample the input signal at the same time. At least one of the cores randomly chooses the type of switching. The conversion outputs are formed according to (3.9). The error signal that needs to be minimized is formed as the difference between the conversion outputs of the two cores, and it can be expressed as

$$e = \sum_{i=0}^{N-1} C_{di}^1 d_i^1 - \sum_{i=0}^{N-1} C_{di}^2 d_i^2 - V_{OSdeq} \quad (3.13)$$

C_{di}^k is the i^{th} digital weight coefficient in the k^{th} channel, d_i^k is the i^{th} digital bit in the k^{th} channel, and V_{OSdeq} is the equivalent offset voltage that is equal to the difference of the digital offsets in the two cores. The following update equations for digital weight coefficients can then be derived using the LMS algorithm:

$$\begin{aligned} C_{di,j+1}^1 &= C_{di,j}^1 - 2\mu e d_i^1 \\ C_{di,j+1}^2 &= C_{di,j}^2 + 2\mu e d_i^2 \end{aligned} \quad (3.14)$$

where $C_{di,j}^k$ is the value of the coefficient C_{di}^k at the j^{th} iteration of the LMS algorithm. For digital offset the update equation is

$$V_{OSdeq,j+1} = V_{OSdeq,j} + 2\mu e \quad (3.15)$$

where $V_{OSdeq,j+1}$ is the value of the offset coefficient V_{OSdeq} at the j^{th} iteration of the LMS algorithm.

The algorithm can be described by the following steps:

1. Initialize digital weight coefficients in both cores according to (3.12), and the digital offset to zero.
2. Sample the input signal into the two cores.
3. In the first core randomly choose the type of switching. The second core can always perform the same kind of switching or have it chosen randomly independent of the first core. Perform the conversions and obtain the digital bits $\{d_i^1\}$ and $\{d_i^2\}$
4. Calculate the error signal using (3.13).
5. Update the digital coefficients using (3.14).
6. Update the offset using (3.15).
7. Go to the step 2.

3.4.3 Multi-Channel SAR Calibration

The analog part and clock generation of a digitally calibrated multi-channel time-interleaved SAR ADC operate in the same way as in the case of the trimming calibration. Every channel has its own set of digital weight coefficients and a digital offset that are adaptively updated. Every sample of the reference channel corresponds to a sample in one of the time-interleaved channels (channel number k). The difference between the reference channel output and the corresponding (k^{th}) interleaved channel represents the error signal, which can be expressed as

$$e = \sum_{i=0}^{N-1} C_{di}^0 d_i^0 - \sum_{i=0}^{N-1} C_{di}^k d_i^k - V_{OSdeq}^k \quad , \quad (3.16)$$

where C_{di}^k is the i^{th} digital weight coefficient in the k^{th} channel, d_i^k is the i^{th} digital bit in the k^{th} channel, and V_{OSdeq}^k is the equivalent offset of the k^{th} channel relative to the reference channel. The reference channel is denoted by the superscript 0. By applying the LMS algorithm to (3.16), the following update equations for the weight coefficients and channel offsets can be derived:

$$\begin{aligned} C_{di,j+1}^0 &= C_{di,j}^0 - 2\mu e d_i^0 \\ C_{di,j+1}^k &= C_{di,j}^k + 2\mu e d_i^k \quad , \end{aligned} \quad (3.17)$$

$$V_{OSdeq,j+1}^k = V_{OSdeq,j}^k + 2\mu e \quad , \quad (3.18)$$

where $C_{di,j}^k$ and $V_{OSdeq,j+1}^k$ are the values of the coefficients C_{di}^k and V_{OSdeq}^k at the j^{th} iteration of the LMS algorithm, respectively.

The algorithm can be summarized as follows:

1. Initialize $i = 0$.
2. Sample the input signal into the reference channel and the channel number $k = i \bmod M + 1$ (mod is modulo operation).
3. Randomly choose the type of switching in the reference channel. The type of switching in the k^{th} channel can always be the same or randomly chosen independently of the reference channel. Perform the conversions and obtain the digital bits $\{d_i^0\}$ and $\{d_i^k\}$.
4. Calculate the error signal using (3.16).
5. Update the digital coefficients using (3.17).
6. Update the offset using (3.18).
7. Update $i = i + 1$. Go to the step 2.

3.5 Simulation Results

To verify the functionality and the effectiveness of the proposed algorithms, flexible behavioral simulations for both trimming and digital calibration have been implemented. To make simulations more realistic, the effects of random jitter and thermal noise, both from sampling switches and comparators, are included in the simulations. The plots in this section show simulation results for the case of SAR ADCs with quantization noise set to 11-bit level. This was achieved by 11 raw bits in the radix-2 ADCs for the trimming calibration, and by 12 raw bits in the radix-1.865 ADC for the digital calibration. The jitter and thermal noise are both set to 11-bit level. The thermal noise is equally split between the sampling switch and the comparator noise. The capacitor values were randomly generated using normal distribution with standard deviation inversely proportional to the square root of the nominal capacitor value. Comparator offsets were also normally distributed with standard deviation equal to 1% of the reference voltage. For multi-channel architecture the simulations are performed at the speed of the reference channel to reduce simulation time.

Figures 3.3.a), 3.3.b) and 3.3.c) show typical convergence of effective number of bits (ENOB) for the single-channel single-core, the single-channel dual-core, and the 8-way time-interleaved SAR ADC, respectively. The ENOB is defined as

$$ENOB = \frac{SNDR - 1.76}{6.02} \quad , \quad (3.19)$$

where SNDR is signal-to-noise-plus-distortion-ratio. The simulation results shown are for single tone test for both trimming and digital calibration, respectively. These plots are generated by taking a windowed FFT of length equal to 1024 samples. The window is shifted in steps of 128 samples. The calibration is turned on after the first 1024 samples, so the first point in the graph represents non-calibrated performance.

To get a fair comparison of the convergence speeds the LMS coefficient and the minimum trimming capacitor are set such that $2\mu = \frac{C_0}{\Delta C}$. This ensures that the finest relative step in which capacitor values or digital coefficients can change are the same. As expected, the convergence of the trimming calibration is significantly slower since the capacitor values are changed by at most one minimum step size at a time. The convergence speed can always be traded-off with the accuracy of the converged results. The convergence time in time-interleaved converters increases with the number of interleaved converters, which is expected since only one channel is updated at a time.

The final converged value of ENOB is approximately the same in the case of trimming and digital calibration. The ENOB in the multiple-channel ADC is lower than in the single channel ADCs by approximately 0.5 bits. This is because in the time-interleaved ADC the output is taken from a single time-interleaved channel, while in the single-channel ADCs the output is calculated as the average of two different conversions, which reduces the impact of all noise sources by 3 dB.

Figures 3.4.a) and 3.4.b) show the FFT plots in the time-interleaved case before and

after the trimming calibration. The FFT plots for trimming calibration look very similar. Improvements of about 40 dB in SFDR and more than 30 dB in SNDR can be observed. In a more detailed simulation the SFDR would probably be limited by the analog front-end and sampling distortions.

3.6 Algorithm Limitations

All described algorithms are based on the minimization of the error signal obtained as a difference between two conversion results of the same input signal. As it can be seen from Figure 3.2, for some values of the input signal, the error can be equal to zero, even though the direct and the reverse transfer characteristics are different. If the error for every sample of the input signal is equal to zero, the value of the capacitors or digital coefficients will not be updated.

In order to have the calibration algorithms that converge to the right values, the input signal has to be "busy", e.g. the input signal samples need to take values that are diverse enough to contain the information about all the capacitors in the array. The exact definition of "busy" signal is not known, but some necessary and a sufficient condition that the input signal needs to satisfy can be easily derived. First, it is obvious that the number of different input signal samples has to be larger than the number of unknown variables (digital coefficients representing the capacitor and offset values). Otherwise, the problem is equivalent to solving a system of equations that has less equations than unknowns. From (3.4), (3.5) and (3.8) it can be seen that, in the case of the trimming calibration, the updating of a given capacitor value occurs only if the bits at the corresponding location from the two conversions are different. The same holds for the digital calibration of a single-channel single-core SAR ADC, as evident from (3.11). In the case of digital calibration of a single-channel dual-core or multi-channel SAR ADCs, the updating of a given coefficient occurs only if the corresponding digital bit is equal to one. This gives a more restrictive necessary condition: the input signal samples have to contain the values that will produce different bits at location of the capacitor that needs to be calibrated, in the case of trimming calibration and digital calibration of a single-channel single-core SAR ADC, or produce output bits that are equal to one in the case of digital calibration of a single-channel dual-core and multi-channel SAR ADCs. For example, the input signal cannot take values only in the lower half of the input signal range, where the MSB bit is always equal to 0, providing no information about the MSB capacitor to the calibration algorithms.

A sufficient condition can be obtained if K subsequent error expressions in (3.3), (3.10), (3.13) and (3.16), where K is the number of unknown variables, are set to zero and treated as a system of equations. If the rank of the corresponding system is equal to K for every K subsequent equations, the system has a solution and the calibration will converge to the right values.

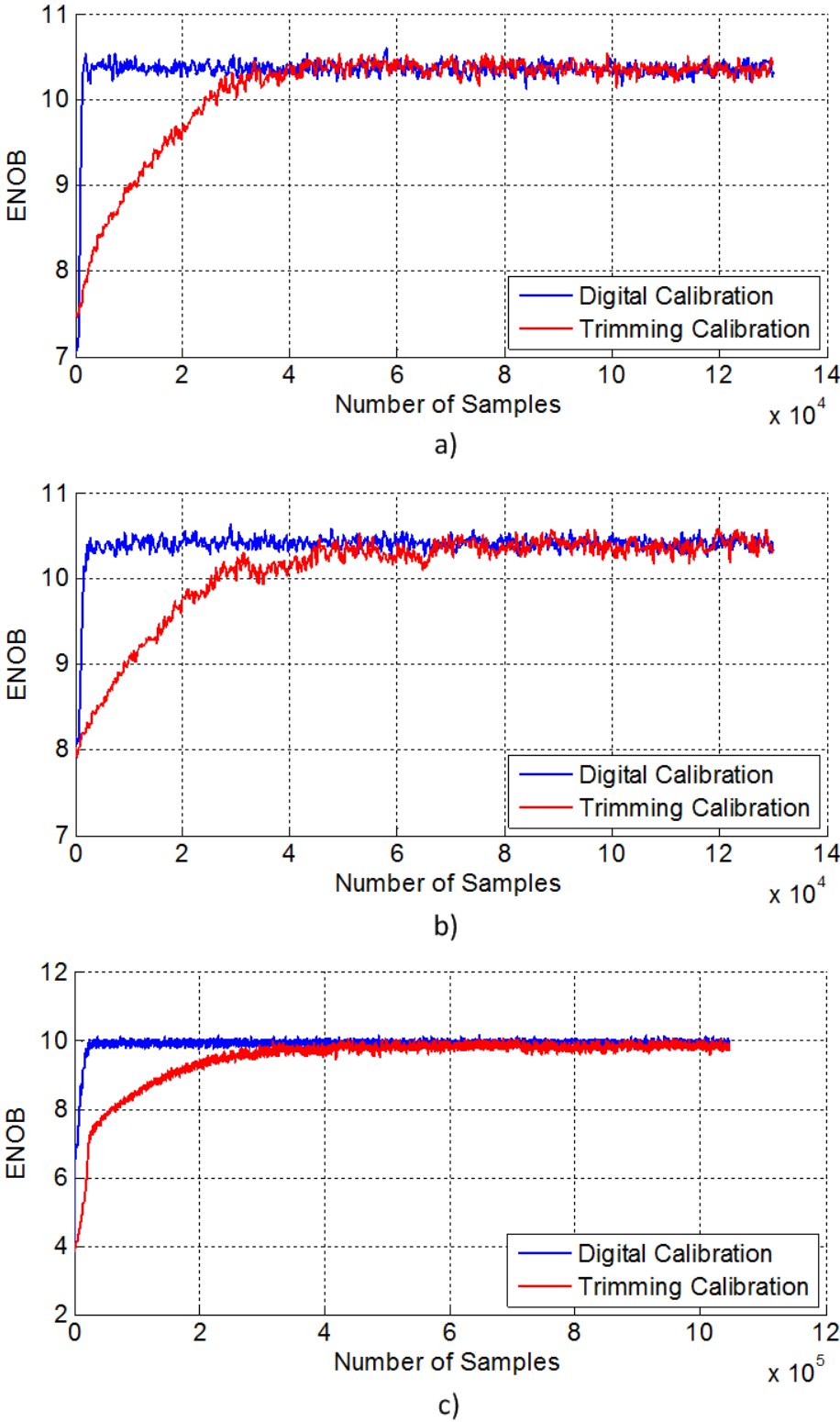
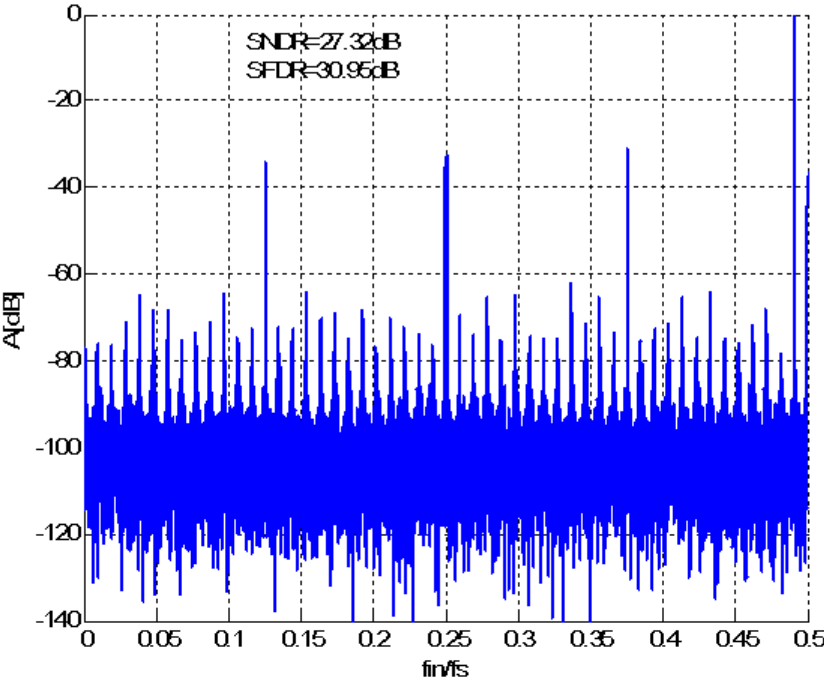
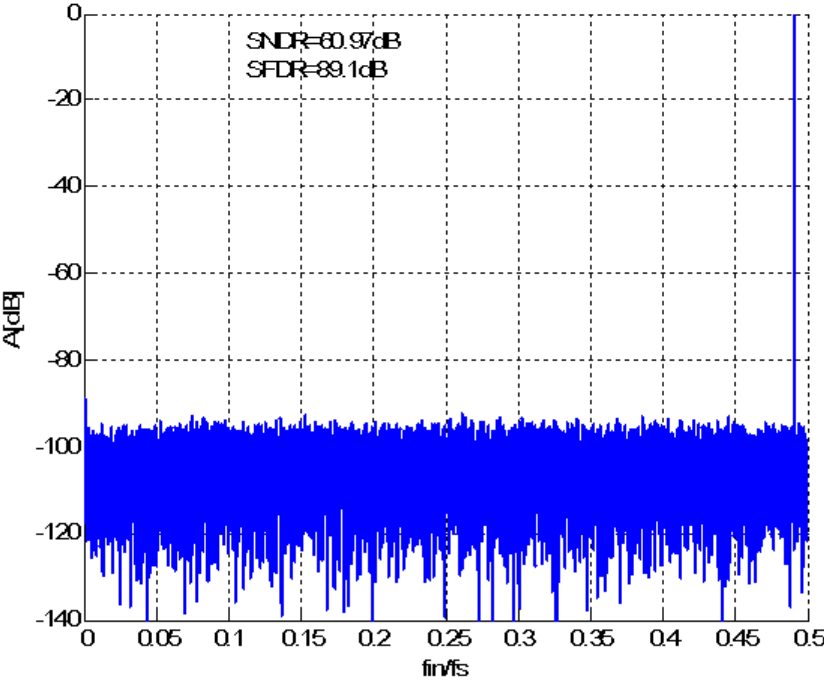


Figure 3.3: Typical ENOB learning curves for a) single-channel single-core, b) single-channel dual-core and c) eight time-interleaved SAR ADC calibration .



(a)



(b)

Figure 3.4: FFT of a sinusoidal signal (a) before and (b) after digital calibration for eight time-interleaved SAR ADCs

Chapter 4

Calibration of Timing Mismatches

In high-speed time-interleaved ADCs the sampling edges of the clocks in time-interleaved channels need to be set with precision of a fraction of a picosecond. With finite speed of clock signal propagation, systematic layout mismatches, delay variation of clock buffers, and threshold voltage variation of the sampling switches, it is almost impossible to achieve this level of accuracy only with a careful layout, and some form of timing calibration is needed. This chapter presents calibration techniques based on the LMS algorithm and using a mixed-signal feedback to fine-tune the clock edges in order to compensate for timing errors.

4.1 Overview of Timing Calibration Techniques

The problem of timing mismatches can be solved by introducing a common front-end sampler [10], [13], [15], [16], but this approach comes with a power and noise penalty in terms of buffering the sampled voltage and resampling it in the individual channels. It is more desirable to sample the input signal directly into the interleaved channels and correct the timing mismatches using a calibration with low power and area overhead. Every calibration entails two components: the estimation and the correction of timing mismatches.

The estimation of the timing mismatches can be performed in the foreground or in the background. In [34] and [14] a sinusoidal input signal and FFT processing was used to extract the timing mismatches in the foreground from the phase of the transformed output signal. These approaches require complex digital circuits not readily available in many systems. A ramp signal with a known slope can be used to measure the timing errors in the foreground [17]. The estimation of the timing errors using a ramp signal can be done in the background as well, if the ramp signal is added to the input signal and the output is filtered by a low-pass filter [19]. However, this requires a zero-mean input signal and it reduces the available input signal range. El-Chammas et al. [11] used an additional single-bit ADC channel for timing calibration. A background calibration algorithm maximizes the correlation between the calibration and time-interleaved channels, effectively forcing the zero-crossings in the

reference and the time-interleaved channels to occur at the same time, thus minimizing the timing errors. This approach is heavily reliant on the statistics of the input signal. Jamal et al. [18] proposed the chopping of the output signal and a FIR approximation of the Hilbert filter [33] to estimate the timing mismatches. For a 10-bit ADC, this approach needs a 21-tap FIR approximation of the Hilbert filter with 10-bit coefficients.

Once the value of the timing mismatches is known, the correction can be performed in the analog or in the digital domain. All analog approaches perform the tuning of the edges of the sampling clock, which can be achieved by inserting a variable capacitive load in the clock path or by tuning of the clock buffers. Digital correction algorithms use digital interpolation filters to recover ideal samples from a non-uniformly-sampled input signal. A digital timing correction that uses adaptive fractional delay filters has been proposed in [18] for a 2-way interleaved ADC. For a 10-bit ADC, it needs a 29-tap FIR correction filter with 10-bit coefficients. In [19] and [17] an iterative algorithm based on Neville's method [7] was used for interpolation. $K * (K - 1)$ multiplications and $K * (K - 1)/2$ subtractions per sample are needed [17], where K is the number of samples used for interpolation. For a 10-bit ADC, $K = 11$ is used in [17] and the interpolated values were re-quantized to 10 bits. In [12] a noncausal IIR filter and a frequency-domain filtering are used for interpolation. Even in today's fine technologies, the power of the digital circuits necessary for these calibration algorithms would dominate the power consumption of the converter.

The goal of this work is to develop a simple background calibration algorithm with low complexity and relaxed requirements for the input signal statistics. The core of the algorithm are an estimation of the input signal derivative, which is used to aid the estimation of the timing mismatches, and a mixed-signal feedback, which is used to fine-tune the edges of the sampling clocks. The details of the proposed calibration algorithm are presented in the following sections.

4.2 Basic Idea

Timing mismatches can be corrected if sampling instances of all time-interleaved channels are aligned to the sampling instance of the same reference channel that is used for linearity correction. The reference channel samples the input signal at the rate of $f_s/(M + 1)$ while all other channels sample at f_s/M , so all time-interleaved channels are corrected in this manner. Let us assume that at time kT the k^{th} time-interleaved channel is supposed to sample the input signal together with the reference channel, but due to different factors mentioned above, there is a timing mismatch of Δt . Neglecting the quantization error and all other error sources the conversion results in the two channels can be written as

$$\begin{aligned} D_r &= v_{IN}(kT) \\ D_k &= v_{IN}(kT) + D\Delta t \quad , \end{aligned} \tag{4.1}$$

where D_r and D_k are conversion results from the reference and the k^{th} channel, respectively, and $D = \left. \frac{\partial v_{IN}}{\partial t} \right|_{t=kT}$ is the derivative of the input signal at the time kT . The error signal, defined as the difference between the reference and the time-interleaved channel outputs, can be approximated to the first order by

$$e = D_r - D_k = -D\Delta t \quad . \quad (4.2)$$

The goal of timing calibration is to estimate the value of the timing mismatch Δt . One way of estimating Δt is using an iterative algorithm, such as the LMS algorithm. Applying the LMS equation to (4.2) produces the update equation for the digital estimate Δt_{dig}^i of Δt in the i^{th} iteration:

$$\Delta t_{dig}^{i+1} = \Delta t_{dig}^i - \mu_{\Delta t} \frac{\partial (e^2)}{\partial (\Delta t)} = \Delta t_{dig}^i + 2\mu_{\Delta t} e D \quad , \quad (4.3)$$

where $\mu_{\Delta t}$ is the LMS coefficient in the timing calibration loop. Once the estimate of Δt is known, we can use it in different ways to obtain the correct value of the input signal at the nominal sampling time. The simplest way would be to just subtract the error given by (4.2) from the conversion result. This would require a knowledge of the precise value of the derivative D at every single sampling time instance. Derivatives can be computed in digital domain using digital differentiators. For a timing error of 5 ps and a sampling frequency of 3 GHz, the maximum error of approximately 5% can be expected for the input signal close to the Nyquist frequency. To reduce the error to less than a half LSB in a 10-bit ADC, the derivative should be known with a precision higher than 1%. As reported in [36], a 20-tap FIR filter with floating-precision coefficients produces a magnitude error better than 1.092%. To keep the quantization error in this 20-tap filter under 1%, the coefficients with at least 12-bit precision are needed. The power and area of the digital multipliers necessary for the realization of this digital differentiator could easily exceed the available budget in a low-power design. Precise estimation of derivatives in analog domain is not an easy problem. Another approach is to use fractional delay filters as in [18]. This approach also requires bulky digital filters that need to run at full speed all the time, and is not very suitable for low power specifications. Lastly, the estimate given by (4.3) can be used to fine tune the actual edge of the sampling clock in analog domain. Note that even though (4.3) includes the derivative D , its value does not need to be known precisely. As it will be seen later, it is good enough to have sufficiently accurate estimation of the derivative that will yield convergence Δt_{dig} possible. Once Δt_{dig} has converged to the right value, the error signal is equal to zero and the value of Δt_{dig} does not change anymore. This is the mixed-signal approach used in this work - the updates of Δt_{dig} are calculated using simple digital circuitry, but the correction is applied in the analog domain.

4.3 Derivative Estimation

The most important property that our derivative estimation circuit should have is simplicity in order not to increase the complexity of the design and the power consumption. The simplest possible circuit that produces an output that resembles the derivative of the input signal is an RC circuit shown in Figure 4.1. The transfer function of this circuit is given by

$$\frac{v_{OUT}(s)}{v_{IN}(s)} = \frac{sCR}{1 + sCR}. \quad (4.4)$$

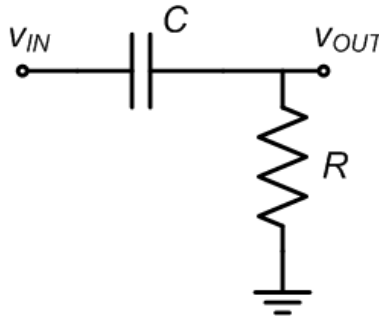


Figure 4.1: RC circuit.

Having a zero at zero in the transfer function looks as a promising approximation of the signal derivative. But, the information about the derivative needs to be available in digital format, and sampling the voltage across the resistor may not be the best option.

A similar transfer function can be obtained by using two RC circuits with different bandwidths as shown in Figure 4.2. The final output is calculated as difference of the two voltages across the capacitors C_1 and C_2 , and the equivalent transfer function is

$$\frac{v_{OUT1}(s) - v_{OUT2}(s)}{v_{IN}(s)} = \frac{s(C_2R_2 - C_1R_1)}{(1 + sC_1R_1)(1 + sC_2R_2)}. \quad (4.5)$$

This circuit is more suitable for our purposes because both voltages are taken from the capacitors, which makes it easier to incorporate it into a standard sample and hold circuit. A simple practical realization is shown in Figure 4.3. TI and REF denote a time-interleaved channel and the reference channel, respectively. One additional identical channel is added (called DIFF channel) to digitize the sampled voltage on the second RC circuit. The bandwidth of the second RC circuit is made different by adding a resistor in series with the sampling switch. Signals e and D are calculated in digital domain by subtracting corresponding conversion results from REF, TI, and DIFF channels, as shown in Figure 4.3. The block Δt represents a delay element that adjusts the edge of the sampling clock and is typically realized as a bank of small capacitors that are switched in or out of the clock signal path.

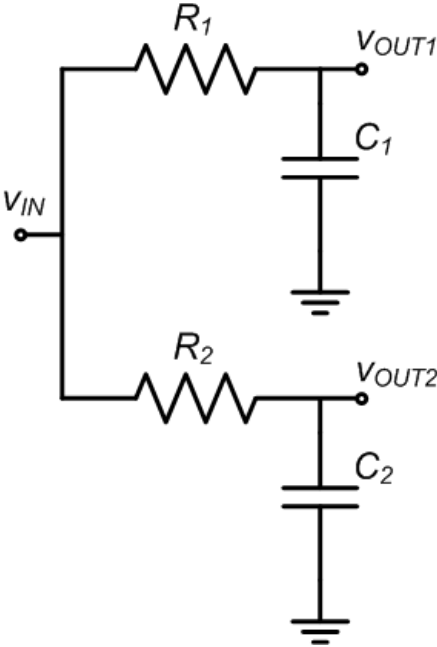


Figure 4.2: Two RC circuits with different bandwidths.

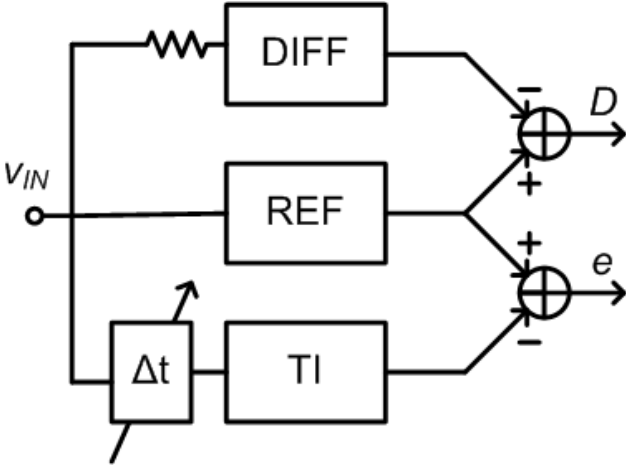


Figure 4.3: Practical realization of two RC circuits with different bandwidths.

In the digital domain the derivative can be estimated by simply taking a difference of the two consecutive input signal samples. The estimation error is small at low frequencies, but it gets worse with increasing the input frequency. From the analysis presented later in this chapter, the calibration using this method of derivative estimation can work only up to a frequency that is smaller than the Nyquist frequency. The estimation can be improved

by using a higher order digital differentiator. Even though the accuracy requirements for the estimate are greatly reduced when the errors are corrected in the analog domain, this approach still requires costly digital multipliers and is not further pursued in this work.

4.4 Convergence Analysis

The transfer function (4.5) is a good approximation of the derivative transfer function only at very low frequencies. At frequencies close to the Nyquist frequency the phase of (4.5) can be as low as 20 - 30 degrees. Still, this simple circuit can be used to generate D signal needed for the timing calibration. This section explains why even such poor approximation of the derivative satisfies our needs. It is not intention of this section to provide rigorous analysis, but rather to provides some common sense understanding of the strengths and weaknesses of the algorithm, and be the guide for modifications that may be necessary to make the algorithm applicable for other uses.

From (4.3) we can see that when $\Delta t_{dig} > \Delta t$ it is necessary that the average of the product eD be negative, and when $\Delta t_{dig} < \Delta t$ this average needs to be positive. In the phasor domain this can be expressed in terms of scalar product as

$$\begin{aligned} \langle \vec{e}, \vec{D} \rangle &< 0 \quad , \quad \Delta t_{dig} > \Delta t \\ \langle \vec{e}, \vec{D} \rangle &> 0 \quad , \quad \Delta t_{dig} < \Delta t \quad . \end{aligned} \quad (4.6)$$

If the condition (4.6) is satisfied, the convergence can be ensured by choosing a sufficiently low value of the LMS coefficient $\mu_{\Delta t}$.

A continuous-time equivalent block diagram of the timing calibration is shown in Figure 4.4. $H_t(s)$, $H_r(s)$, and $H_d(s)$ are the equivalent transfer functions of analog front-ends in TI, REF, and DIFF channels, respectively. V_{ost} , V_{osr} , and V_{osd} are analog offsets in the TI, REF, and DIFF channels, respectively. Since for the purpose of this analysis it is important to have error signal e that is zero-mean, the LMS loop for compensation of the offset in the TI channel is also included in the block diagram. V_{osdig} is the equivalent offset between the REF and TI channel in digital domain. Δt and Δt_d are timing mismatches in the TI and DIFF channel, respectively.

Using Mason's formula, the following transfer functions for $e(s)$ and $D(s)$ can be derived:

$$e(s) = \frac{1}{\mu_{os} f_s} \frac{s [H_r(s) - e^{s\Delta t} H_t(s)]}{1 + \frac{s}{\mu_{os} f_s}} V_{in}(s) + \frac{V_{osr} - V_{ost}}{\mu_{os} f_s} \frac{s}{1 + \frac{s}{\mu_{os} f_s}} \quad (4.7)$$

$$D(s) = [H_r(s) - e^{s\Delta t_d} H_d(s)] V_{in}(s) + V_{osr} - V_{osd} \quad . \quad (4.8)$$

The second term in (4.7) is offset term that is calibrated by the offset calibration loop and can be neglected when calculating the phase of $e(s)$. Assuming Δt and Δt_d are small and

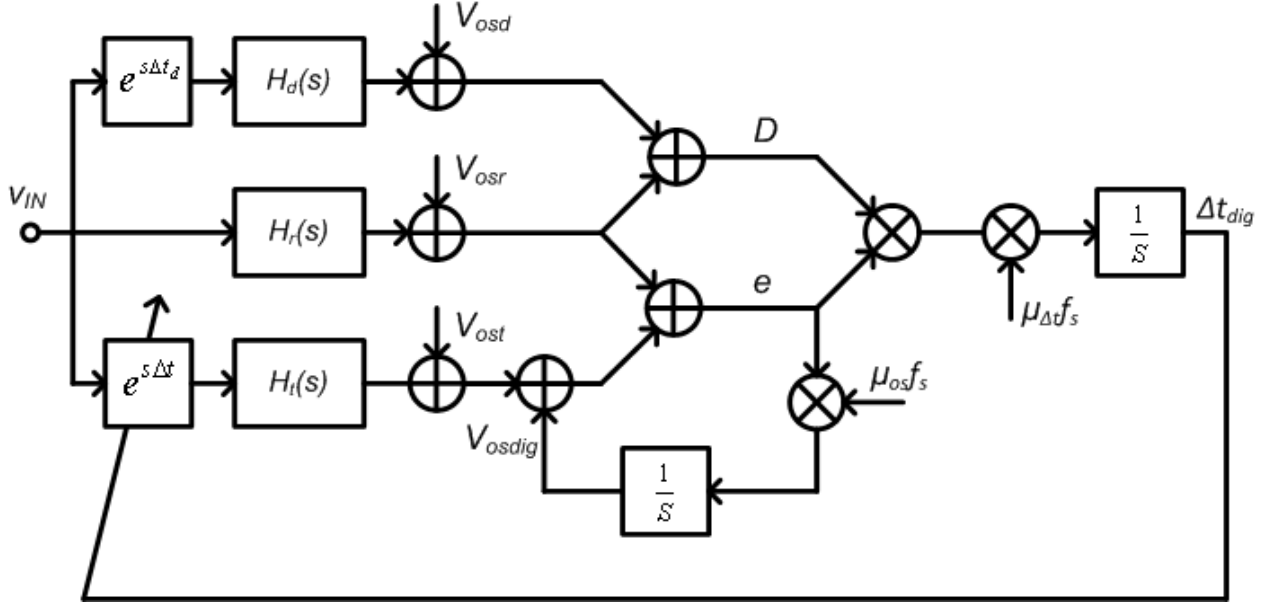


Figure 4.4: Block diagram of the timing calibration.

the analog front-ends can be approximated by the first order transfer function the phase of $e(s)$ and $D(s)$ relative to the phase of $V_{in}(s)$ can be calculated as

$$\begin{aligned} \angle \vec{e} &\approx -\arctan \frac{\omega}{\mu_{os}f_s} - \arctan \frac{\omega}{\omega_0} \quad , \quad \Delta t > \Delta t_{dig} \\ \angle \vec{e} &\approx 180^\circ - \arctan \frac{\omega}{\mu_{os}f_s} - \arctan \frac{\omega}{\omega_0} \quad , \quad \Delta t < \Delta t_{dig} \end{aligned} \quad (4.9)$$

$$\angle \vec{D} \approx 90^\circ - \arctan \frac{\omega}{\omega_0} - \arctan \frac{\omega}{\omega_{0d}} \quad , \quad (4.10)$$

where ω_0 and ω_{0d} are corner frequencies of $H_r(s)$ and $H_d(s)$, respectively. The phase difference between $e(s)$ and $D(s)$ can now be expressed as

$$\begin{aligned} \angle \vec{D} - \angle \vec{e} &\approx 90^\circ - \arctan \frac{\omega}{\omega_{0d}} - \arctan \frac{\omega}{\mu_{os}f_s} \quad , \quad \Delta t > \Delta t_{dig} \\ \angle \vec{D} - \angle \vec{e} &\approx -90^\circ - \arctan \frac{\omega}{\omega_{0d}} - \arctan \frac{\omega}{\mu_{os}f_s} \quad , \quad \Delta t < \Delta t_{dig} \end{aligned} \quad (4.11)$$

It can be seen that condition (4.6) is satisfied if

$$\omega_{0d} > \mu_{os}f_s, \quad (4.12)$$

which is easily achieved by selecting proper values of μ_{os} and ω_{0d} .

4.5 Algorithm Modifications

The analysis presented in the previous section has some shortcomings. Firstly, the uncertainties in the value of D at low frequencies, where amplitude of D is small and the sign of D can change due to secondary effects, such as residual offset and nonlinearities in the ADC transfer functions, can cause convergence problems. Secondly, we assumed that the analog front-ends are single-pole systems, which is never true in a real system. This can cause our analysis to be inaccurate at high frequencies when higher order poles start introducing significant phase shifts. Also, neglecting Δt and Δt_d can give misleading results at very high frequencies. Fortunately, at low frequencies the timing mismatches don't degrade the performance significantly, the timing calibration is not really necessary, and can therefore be turned off. At very high frequencies, far above the Nyquist zone for which the ADC is designed, the timing calibration can also be switched off. This can be achieved by observing the amplitude of D , which is small at both low and high frequencies. Whenever $|D|$ is smaller than some threshold value δ , the update of Δt_{dig} is not performed. This is graphically illustrated in Figure 4.5. The designer's job is to choose the right values of the corner frequencies in the analog front-ends and proper value of the threshold, so that stable operation is achieved in the desired signal bandwidth.

Finally, since the amplitude of D is varying a lot with frequency, so is the loop gain of the LMS loop. This may lead to non-smooth convergence of Δt_{dig} at high frequencies or calibration being switched off at moderate frequencies where it is still necessary. To solve

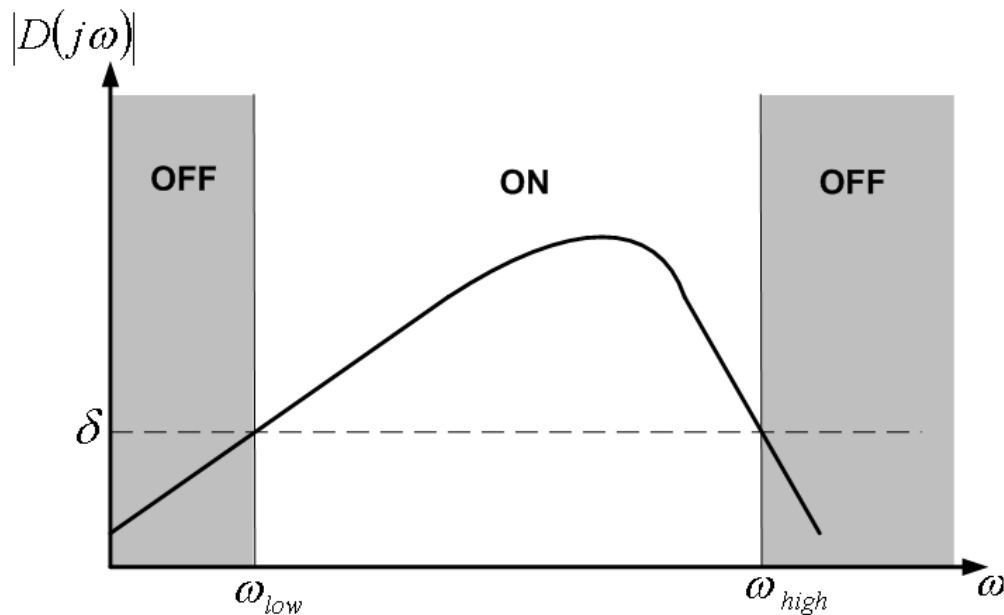


Figure 4.5: $D(\omega)$ and enabling of the timing calibration.

this problem we observe that all information needed for calibration of timing mismatches is contained in the phases of signals e and D . Hence, instead of e and D , we can use $\text{sgn}(e)$ and $\text{sgn}(D)$. Because of the problems described the sign function for D will also have a dead zone around zero. Finally, the modified LMS update equation becomes

$$\begin{aligned} \Delta t_{dig}^{i+1} &= \Delta t_{dig}^i + 2\mu_{\Delta t} \text{sgn}(e) \text{sgn}(D) \quad , \quad (|D| \geq \delta) \\ \Delta t_{dig}^{i+1} &= \Delta t_{dig}^i \quad , \quad (|D| < \delta) \end{aligned} \quad (4.13)$$

The proposed modifications are illustrated in Figure 4.6. The LMS integration is represented in z -domain. The final quantizer block is inserted to limit the resolution of the circuit that tunes the clock delay. These modifications not only improve the convergence of the algorithm, but also simplify hardware implementation by replacing costly multipliers with simple logic circuits.

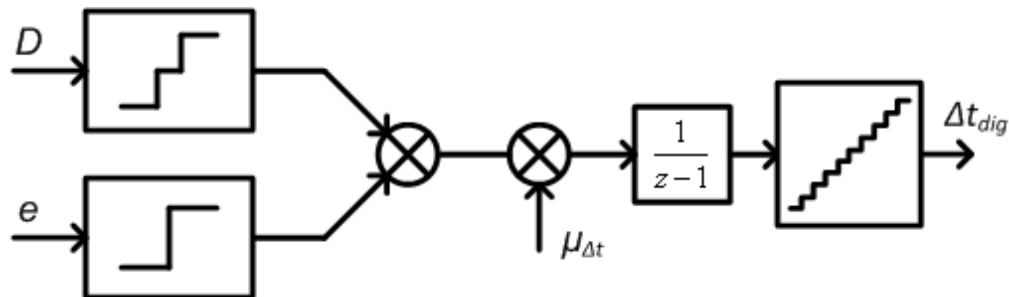


Figure 4.6: Modification of the basic timing calibration algorithm.

4.6 Calibration Without Additional Channel

Introducing an additional ADC channel for timing calibration purposes can sometimes present too large of an overhead, especially if the interleaving factor is relatively small. It would be convenient if there was a way to calibrate the timing mismatches using only data that are already available from the TI and REF channels. After all modifications to the algorithm, it can be noted that the DIFF channel does nothing else but produce a phase-shifted version of the input signal. Since a time-delayed version of the input signal by one sampling period T is readily available from the neighboring time-interleaved channel, one may consider using that output instead of introducing a completely new channel. Indeed this is a valid option and it works with some limitations.

To explore the limitations of this alternate algorithm we will do the same type of simplified analysis as in the case of timing calibration with DIFF channel. The equivalent block diagram is very similar to that of Figure 4.4, except the transfer function $H_d(s)$ is replaced by $H_t(s)$, and the delay element in the virtual additional channel is approximately equal to T (timing

mismatch is neglected). While the expression for $e(s)$ stays the same as in (4.7), $D(s)$ can now be calculated as

$$D(s) = H_r(s) [1 - e^{sT}] V_{in}(s) + V_{osr} - V_{osd} \quad , \quad (4.14)$$

and the phase of D is now given by

$$\angle \vec{D} \approx 90^\circ - \frac{\omega T}{2} - \arctan \frac{\omega}{\omega_0} \quad . \quad (4.15)$$

Finally, the phase difference between \vec{D} and \vec{e} is

$$\begin{aligned} \angle \vec{D} - \angle \vec{e} &\approx 90^\circ - \frac{\omega T}{2} + \arctan \frac{\omega}{\mu_{os} f_s} \quad , \quad \Delta t > \Delta t_{dig} \\ \angle \vec{D} - \angle \vec{e} &\approx -90^\circ - \frac{\omega T}{2} + \arctan \frac{\omega}{\mu_{os} f_s} \quad , \quad \Delta t < \Delta t_{dig}. \end{aligned} \quad (4.16)$$

In order to have satisfied (4.6) in the first Nyquist zone the following needs to be true:

$$\arctan \frac{\omega}{\mu_{os} f_s} > \frac{\omega T}{2} \quad . \quad (4.17)$$

Figure 4.7 shows a sketch of the left and right side of (4.17) versus frequency in the first Nyquist zone. It can be seen that (4.17) cannot be satisfied at the frequencies close to $f_s/2$. In order to increase the range of frequencies for which (4.17) is true, μ_{os} needs to be set smaller.

Assuming $\mu_{os} \ll 1$, in higher Nyquist zones, $\arctan \frac{\omega}{\mu_{os} f_s} \approx \pi/2$. Because the term $\omega T/2$ grows linearly with frequency, it changes the sign of $\langle \vec{e}, \vec{D} \rangle$ in every Nyquist zone. The calibration still can be performed, but the LMS update equation (4.13) should be modified in the following manner:

$$\begin{aligned} \Delta t_{dig}^{i+1} &= \Delta t_{dig}^i + 2\mu_{\Delta t} \operatorname{sgn}(e) \operatorname{sgn}(D) \quad , \quad (|D| \geq \delta) \text{ and odd Nyquist zone} \\ \Delta t_{dig}^{i+1} &= \Delta t_{dig}^i - 2\mu_{\Delta t} \operatorname{sgn}(e) \operatorname{sgn}(D) \quad , \quad (|D| \geq \delta) \text{ and even Nyquist zone} \\ \Delta t_{dig}^{i+1} &= \Delta t_{dig}^i \quad , \quad (|D| < \delta) \end{aligned} \quad (4.18)$$

4.7 Simulation Results

The calibration algorithm described above needs to coexist with the algorithm used to calibrate nonlinearities, offset, and gain mismatches. Together these two algorithms form multiple LMS loops that are extremely difficult to analyze. Practically, the only way to verify that the whole calibration system works properly is through the means of simulation.

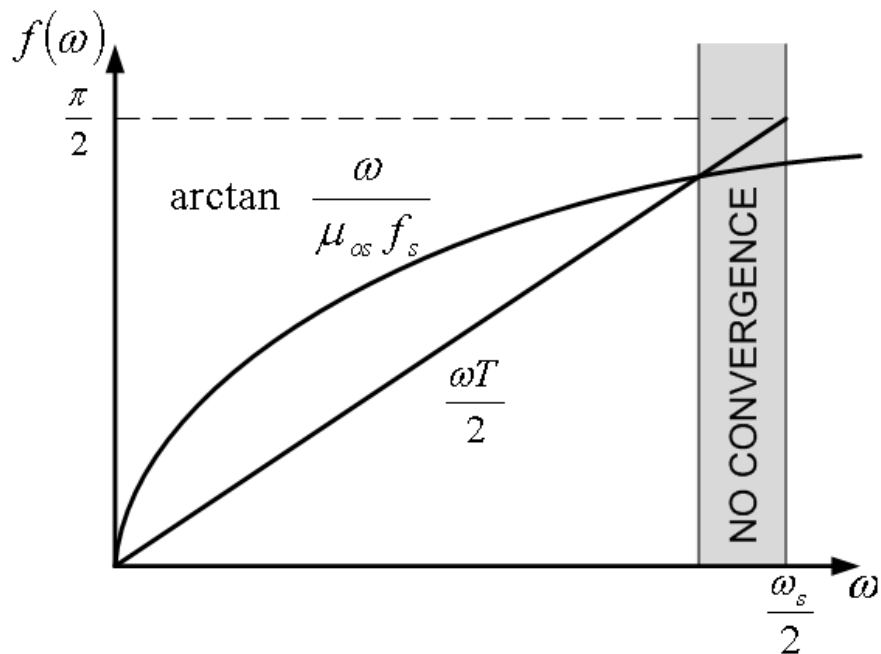


Figure 4.7: Stability range in the first Nyquist zone.

A flexible behavioral simulation framework has been implemented to verify the effectiveness of the algorithm. Although many different configurations have been simulated, the plots shown in this chapter are for a time-interleaved ADC with 24 interleaved channels, nominal radix of 1.85, and 11 raw bits. This corresponds to the configuration that is implemented in the chip prototype described later. Comparator offsets and capacitor values are generated randomly with normal distribution. Capacitor values include both random and systematic mismatches. A brief summary of the simulation setup is shown in Table 4.1. To make simulations more realistic different sources of non-idealities have been included. All non-idealities and their levels expressed in effective number of bits (ENOB) are shown in Table 4.2.

Figure 4.8 shows typical ENOB convergence versus the number of samples in the reference

Table 4.1: Simulation setup.

Parameter	Value
Interleaving factor	24
Number of raw bits	11
Radix	1.85
Offset σ	$1\%V_{ref}$
Random capacitor σ	$0.33 \times 10^{-9}/\sqrt{C}$
Systematic capacitor σ	$1.66 \times 10^{-9}/\sqrt{C}$

Table 4.2: Error sources included in the simulation.

Error Source	Value [ENOB]
Quantization noise	10
Thermal noise	9
Jitter	9
Phase skew	6
Bandwidth mismatch	7
AFE distortions	10

channel. The value of the ENOB is calculated from 2048 point windowed FFT, where the window is shifted in increments of 128. The convergence of Δt_{dig} is shown in Figure 4.9. This diagram looks very similar for the algorithm without additional channel if the input frequency is in the convergence range. Finally, Figure 4.10 shows ENOB vs. frequency with timing calibration switched on and off. The improvements of almost two bits can be observed at frequencies close to the Nyquist frequency. The residual roll-off of the ENOB at higher frequencies is due to the simulated jitter and the finite resolution of the timing correction.

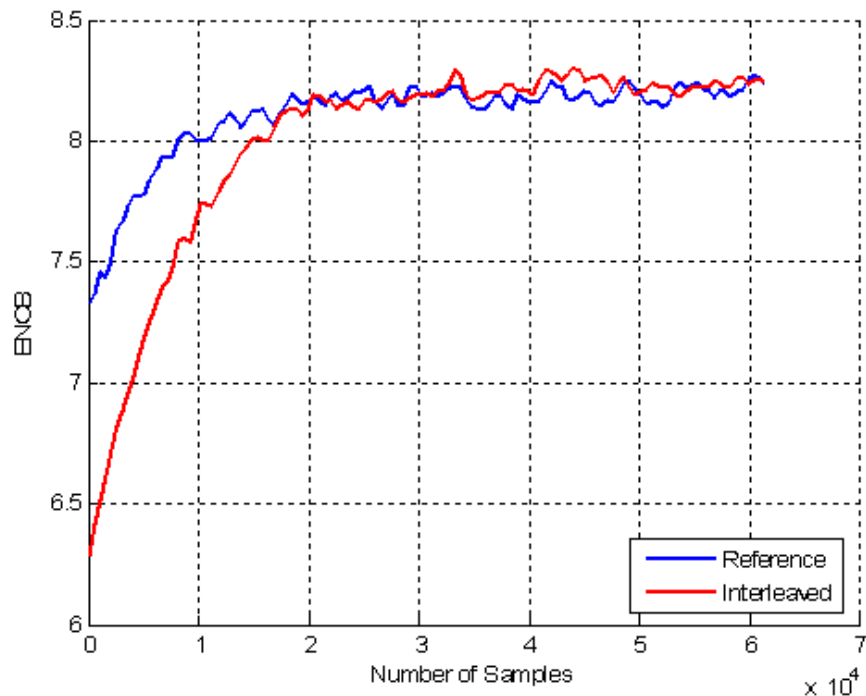
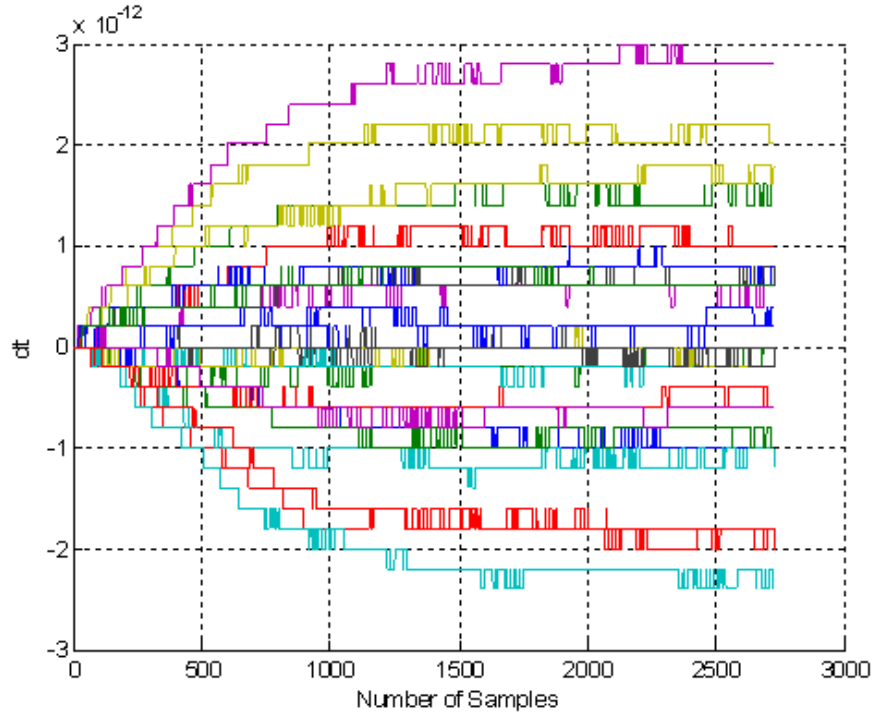


Figure 4.8: Typical ENOB convergence.

Figure 4.9: Typical Δt_{dig} convergence.

4.8 Algorithm Limitations

The timing errors will be calibrated only if the amplitude of D is larger than the introduced dead zone around zero. This poses a restriction on the amplitude and frequency of the input signal. For a given input signal frequency, the amplitude of the input signal has to be higher than a certain threshold value that is determined by the input frequency. Assuming a single-pole approximation of the analog front-ends, the amplitude, A , of the input signal at the frequency ω has to satisfy the following condition:

$$A > \delta \frac{\sqrt{\left(1 + \left(\frac{\omega}{\omega_0}\right)^2\right) \left(1 + \left(\frac{\omega}{\omega_{0d}}\right)^2\right)}}{\omega \left(\frac{1}{\omega_{0d}} - \frac{1}{\omega_0}\right)}, \quad (4.19)$$

where ω_0 and ω_{0d} are the corner frequencies of the analog front-end of the REF and the DIFF channel, respectively, and δ is the size of the dead-zone around zero introduced for the D signal.

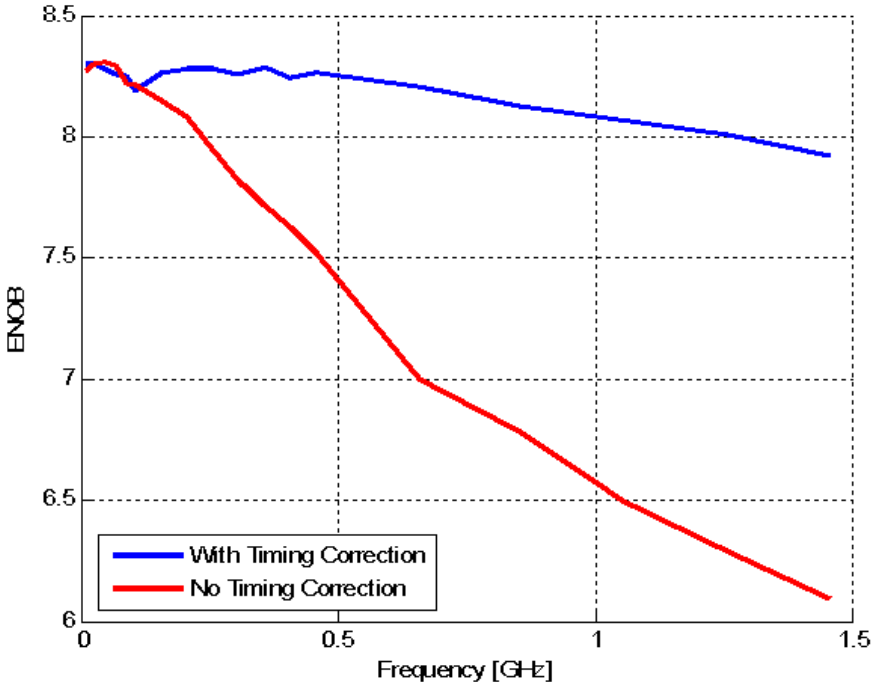


Figure 4.10: ENOB vs. frequency with and without timing calibration.

Chapter 5

Circuit Implementation

In this chapter circuit implementation details of the ADC chip are presented. The chapter starts with the design of a single SAR ADC channel and all its building blocks. Next, the clock generation and distribution circuitry is explained. Finally, the operation and synthesis of digital calibration logic and full-chip integration is discussed.

5.1 Single SAR Channel

Ensuring the desired performance of a single-channel SAR ADC entails careful design and verification of different constituent circuit blocks. The SAR ADC architecture used in this design is based on radix-weighted capacitive DAC. The main building blocks are shown in Figure 5.1. The capacitive DAC generates radix-weighted reference voltages during conversion. The input signal is connected to the top plates of the capacitors in the capacitive DAC through bootstrapped switches, and the sampling of the input voltages is performed using bottom-plate sampling. The comparator compares the input signal to the reference voltages generated by the capacitive DAC. Finally, the SAR logic controls the operation of the capacitive DAC based on the comparator decisions.



Figure 5.1: Block diagram of a single SAR ADC channel

5.1.1 Capacitive DAC

In a SAR ADC the input signal is sampled onto multiple DAC capacitors of different sizes that share one of their plates, and the sampled input signal is proportional to the charge

sampled on the shared plates. The power of the sampled noise charge on the shared plates is equal to the sum of the powers of the noise charges on all DAC capacitors. Therefore, the total sampled thermal noise (kT/C noise) is determined by the sum of all capacitors in the DAC, (C_{tot}). In our design we set the total thermal noise level to 9 effective bits (56 dB) and allocate half of it to the sampling kT/C noise. For a rail-to-rail input signal with amplitude of 2.4 V peak-to-peak differential and total kT/C signal-to-noise ratio (SNR) of 59 dB, $C_{tot} = 50$ fF. The quantization noise is set to 10-bit level during the design phase, which means that the smallest capacitance in the DAC should approximately be $C_{tot}/1000 = 50$ aF. Based on the mismatch model of the capacitors, the radix of 1.85 is chosen to provide enough redundancy for digital calibration. By combining thermal and quantization noise requirement with the redundancy level, the final structure of the capacitive DAC is obtained. It consists of 12 capacitors: $C_{0A} = C_0 = 50$ aF, $C_i = 1.85C_{i-1}, i = 1..10$. The schematic of the DAC is shown in Figure 5.2.a). The single-ended version is shown for simplicity, although the real

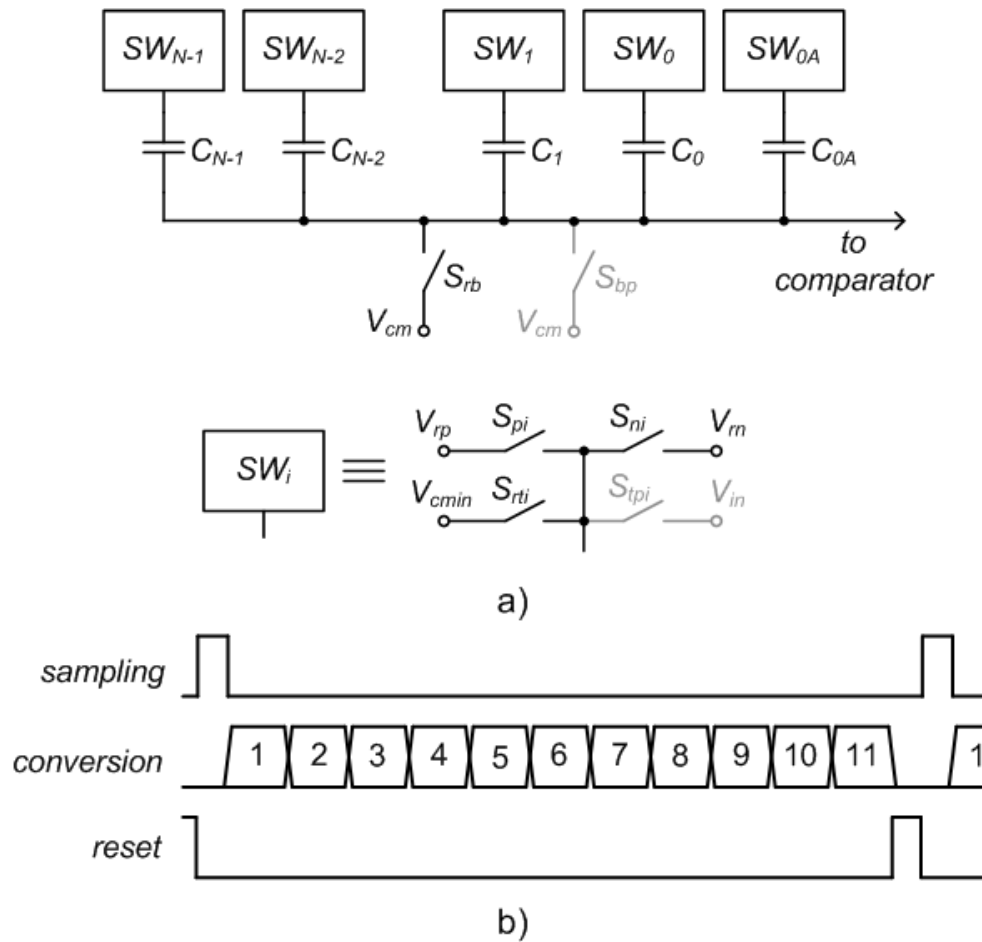


Figure 5.2: Capacitive DAC a) single-ended schematic and b) phases of operation.

implementation is fully differential. The timing diagram of the phases of DAC operation are shown in Figure 5.2.b). During the sampling phase, the switches S_{tpi} and S_{bp} charge the DAC capacitors to the input voltage. After the sampling, the switches S_{ni} and S_{pi} connect the top plates of the capacitors to one of the reference voltages V_{rp} and V_{rn} during the conversion process, as described in Chapter 3. Finally, the capacitors are reset to the common-mode reference voltages using the switches S_{rti} and S_{rb} during the reset phase.

The DAC capacitors are realized as parallel-plate capacitors between two regular metal layers. To avoid further confusion, terms bottom plate and top plate will denote the plates that, during sampling phase, are connected to a DC common-mode level and the input signal, respectively. Explicit number of the metal layer will be used when referring to the physical implementation of the capacitor. Metal-4 and metal-5 layers are used in this design to provide low parasitics to substrate, and to leave metal-6 and metal-7 for routing of references in thick metal and isolation of sensitive nodes between neighboring channels. Metal-4 plates of the capacitors are connected to the DAC switches, while metal-5 plates are all connected to the comparator input and sampling switches. There are two main reason for this choice of top and bottom plates. First and most important, choosing metal-4 plates as top plates makes it possible to minimize the parasitic capacitance between bottom plates and wires that connect the capacitors to the DAC switches by hiding these wires under the metal 4 plates. Otherwise, these parasitic capacitances could easily alter the ratios of the capacitors in the DAC. The second reason is to minimize the parasitic capacitance at the comparator input in order to minimize the equivalent input-referred comparator thermal noise. Capacitors with calculated values are simply instantiated from the process design kit and placed in the layout. No special matching techniques are used. An illustration of the DAC layout is shown in Figure 5.3.

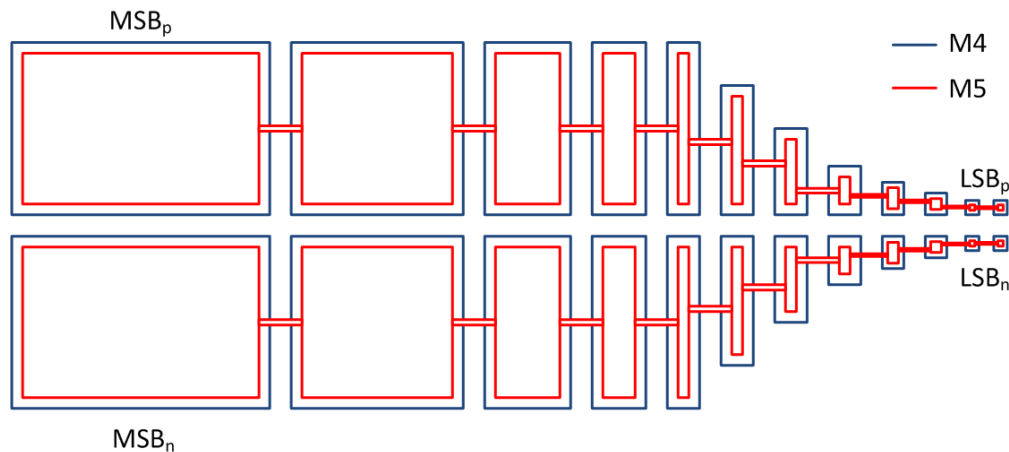


Figure 5.3: Illustration of DAC layout.

The switches S_{pi} are implemented as PMOS switches and connect the top plates of the

capacitor to the positive reference $V_{rp} = V_{DD}$, while S_{ni} are realized as NMOS transistors and connect the top plates to the negative reference $V_{rn} = 0$. All switches have minimum channel length and their width is proportional to the size of the corresponding capacitors with correction for capacitor parasitics that also need to be driven. The sizing of the switches is determined by the settling requirement of the DAC. The DAC needs to settle to a fraction of an LSB (this fraction determines the ADC's DNL performance) in the most critical bit-testing phase. The first bit-testing phase is not critical since the voltage change at the input of the comparator is given by (5.1). This means that when the input voltage is close to the threshold of the first bit ($\frac{C_{N-1}}{C_{tot}}V_r$) the voltage step at the comparator input ΔV_1 is close to zero.

$$\Delta V_1 = \frac{C_{N-1}}{C_{tot}}(V_r - V_{in}) - \frac{C_{tot} - C_{N-1}}{C_{tot}}V_{in} = \frac{C_{N-1}}{C_{tot}}V_r - V_{in} \quad (5.1)$$

Since the voltage steps at the comparator input during the remaining bit-testing phases are directly proportional to the sizes of the capacitors at the corresponding bit-location, the most critical phase is the second one.

The reset switches S_{rti} and S_{rb} are simple NMOS switches. S_{rti} switches reset the top plates to the input signal common-mode reference V_{cm} , while the S_{rb} switches reset the bottom plates to the comparator input common-mode voltage V_{cm} . These switches do not need to reset the capacitor voltages to zero, but only to a voltage low enough that the memory effect is negligible at the end of the next tracking phase. S_{rb} provides a leakage path for the charge sampled at the bottom plates, so a transistor type with sufficiently low leakage needs to be used.

The switches S_{tpi} and S_{bp} , shown in gray, are the top- and bottom-plate sampling switches, respectively, and they are not a part of the capacitive DAC. More details about these switches are presented in the next subsection.

5.1.2 Top-Plate and Bottom-Plate Switches

Switches in A/D converters perform the sampling operation and a proper design is needed to ensure input signal integrity. Different nonidealities of realistic switches contribute to the degradation of the sampling accuracy. The most important are nonlinear signal-dependent resistance and capacitance of the switches and signal-dependent charge injection. To preserve linearity with a rail-to-rail input signal swing, two techniques have been employed: bootstrapping of the top-plate switches and bottom-plate sampling.

The schematic of the top-plate switch together with the non-overlapping clock generator is shown in Figure 5.4. A single-ended version is shown for simplicity. The switch is a modified version of the bootstrapped switch proposed in [1]. ϕ is one of the twenty four phases obtained by division of the main clock. The non-overlapping clock generator is used to separate the precharging phase from the tracking phase in a robust way. The transistors M_1 and M_2 , together with the capacitors C_1 and C_2 , form a charge pump that is producing a boosted clock signal with low and high level of V_{DD} and $2V_{DD}$ ideally (neglecting charge

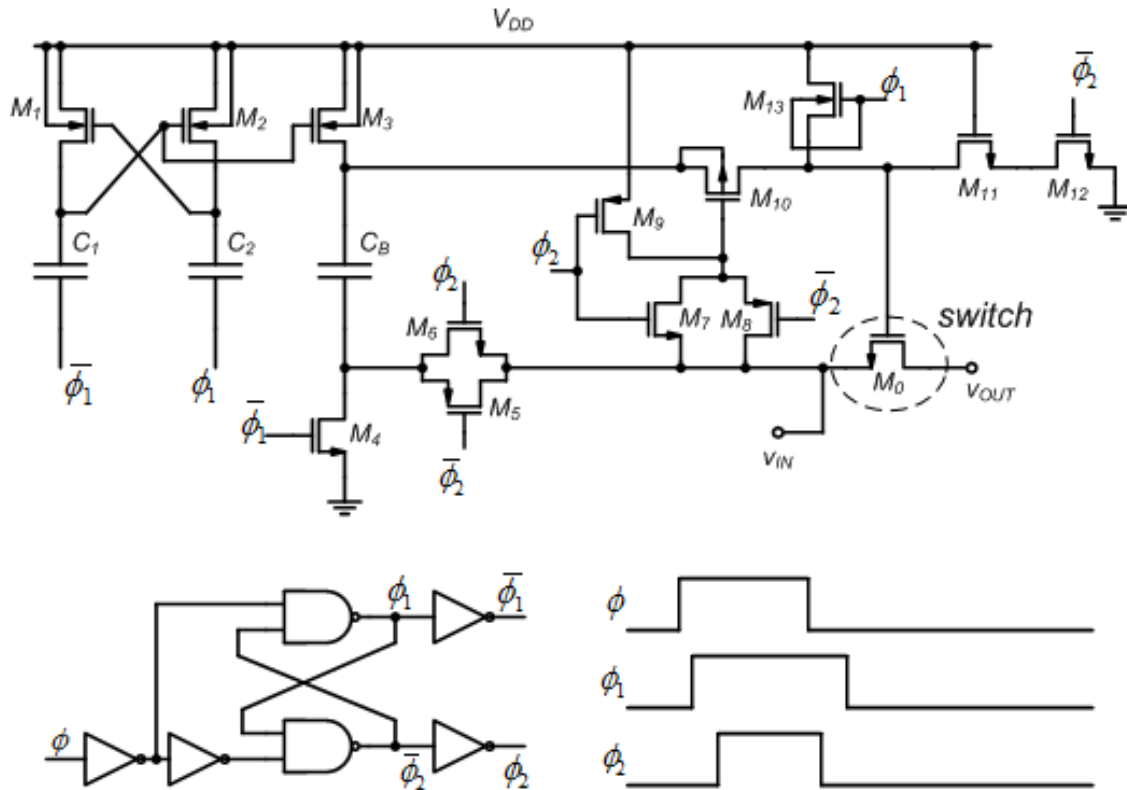


Figure 5.4: Schematic of the top-plate bootstrapped switch with non-overlapping clock generator and timing diagram of clock phases.

sharing). This boosted clock is driving the gate of M_3 , which is precharging the boosting capacitor C_B , together with M_4 , when ϕ_1 is low. Tying the body contacts of M_1 , M_2 and M_3 to V_{DD} lowers the threshold of these devices, making the use of minimum-sized transistors possible, and reduces the required capacitances of C_1 and C_2 . It also improves reliability of the circuit since no two terminals of M_1 , M_2 and M_3 experience voltage difference larger than V_{DD} . The transistors M_5 - M_8 and M_{10} connect the C_B between the input signal node and the gate of M_0 , which is the actual bootstrapped switch. The sizing of M_5 , M_6 and M_{10} is critical for achieving sufficient bandwidth in the bootstrapping circuit, and, consequently, for the linearity of the switch at high input frequencies. C_B has to be large enough to minimize the effect of charge sharing between the C_B and the all the parasitic capacitances connected to the gate of M_0 . The M_0 , although drawn as a single device, actually represents $N + 1$ switches with all sources and gates connected together, and drains connected to the top plates of the capacitors in the capacitive DAC. During the precharge phase, M_9 and M_{12} turn M_{10} and M_0 off, respectively. M_{11} is a cascoded device that shields M_{12} from breakdown and also reduces the leakage through M_{12} during the tracking phase. M_{13} precharges the

gate of M_0 to approximately $V_{DD} - V_{th}$ right before the tracking phase. This reduces the effect of charge sharing and the required capacitance of C_B . It also keeps transient voltage between the source and the drain of M_{10} under V_{DD} .

The schematic of the bottom plate switch and a timing diagram of its clock relative to the clock of the top-plate switch are shown in Figure 5.5. The switch is realized as a CMOS switch with a PMOS/NMOS width ratio that produces a flat transconductance around the common mode level $V_{cm} \approx V_{DD}/2$. The importance of the flat transconductance will be discussed in more detail in the section on clock generation. Dummy NMOS devices M_{d1} and M_{d2} are added to further cancel charge injected from the PMOS transistor, as well as to provide symmetrical load for the clock drivers. The bottom-plate switch should be implemented with fast switches to reduce the switch size and clock driver power, but also the leakage current of the bottom-plate switch has to be small enough not to change the value of the sampled signal during the entire conversion cycle. This trade-off dictates the choice of the transistor type for the bottom-plate switch.

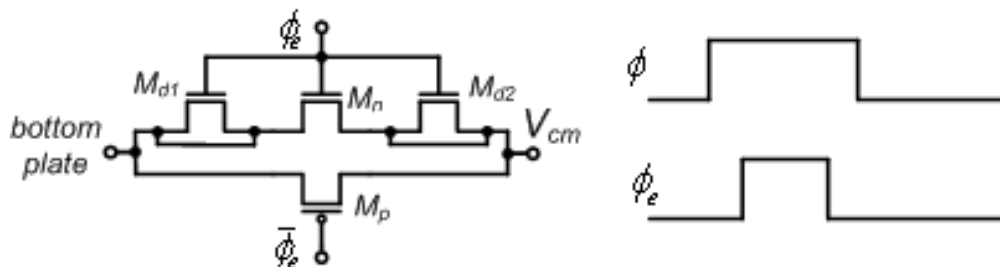


Figure 5.5: Schematic of the bottom-plate switch.

5.1.3 Comparator

The comparator effectively compares the input signal to the radix-weighted set of reference voltages generated by the capacitive DAC, and represents the central part of a SAR ADC. The comparator used in this design is a StrongArm latch-based comparator [20] shown in Figure 5.6. The StrongArm latch was chosen as a fully dynamic, simple and power-efficient solution. The operation of the comparator is as follows. First, when the clock signal is low, the tail transistor M_{clk} is turned off, and the transistors M_{r1} to M_{r6} reset internal nodes of the comparator to V_{DD} . After the clock signal goes high, the M_{clk} turns on and its current is split between M_1 and M_2 discharging the capacitance at the source nodes of M_3 and M_4 . When the source voltages of M_3 and M_4 reach $V_{DD} - V_{th}$, M_3 and M_4 turn on and start discharging the capacitance at the gates of M_5 and M_6 . Finally, when M_5 and M_6 turn on, the regenerative action of the positive feedback in the cross-coupled inverter pair formed by M_3 , M_4 , M_5 , and M_6 forces one of the outputs of the comparator to go to 0 and the other to V_{DD} , depending on the polarity of the input signal. Either *op* or *on* output is used

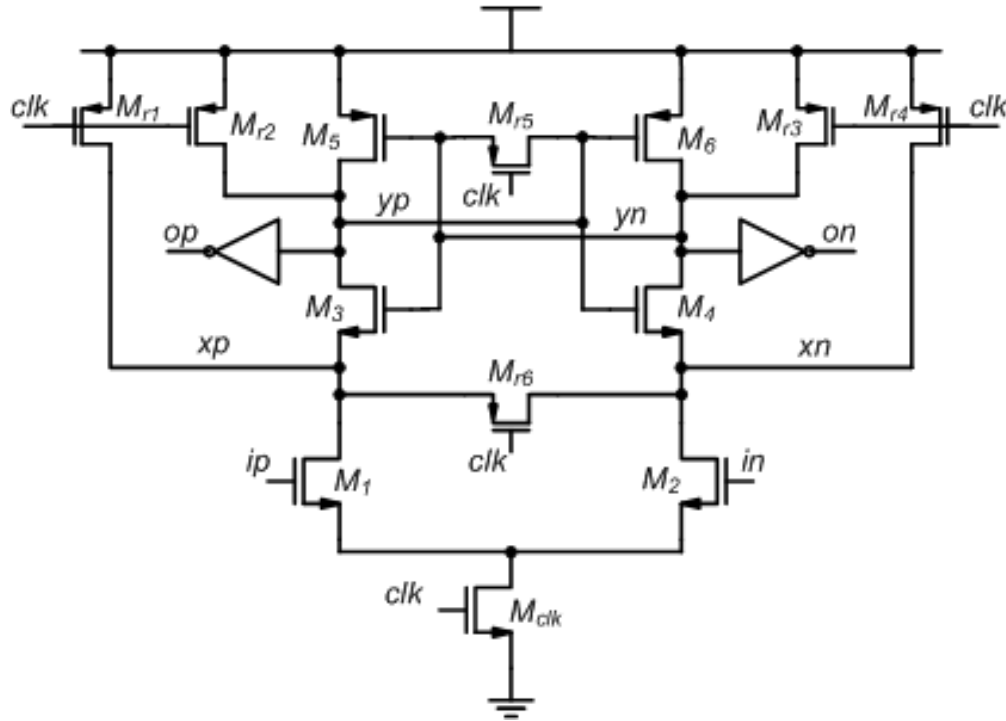


Figure 5.6: Schematic of the StrongArm latch comparator.

depending on whether the direct or reverse switching is selected. The comparator has to be carefully designed in order to satisfy different requirements, such as thermal noise, speed, metastability, offset, leakage, hysteresis, and kickback noise.

A detailed noise analysis of the latch-based comparator can be found in [32]. Based on this analysis, a set of intuitive guidelines for the low-noise design is outlined here. The initial noise voltage sampled by the reset switches is inversely proportional to the capacitances at the internal nodes. Therefore, increasing the width of all transistors at the same time will reduce the input-referred noise. Of course, this comes with a power penalty. During the first phase, when the source nodes of M_3 and M_4 are being discharged, the input stage of the comparator behaves as a g_m - C integrator. The signal power at the output of a simple g_m - C integrator is proportional to $g_m^2 T^2$, where g_m is the transconductance and T is the time of integration. The noise voltage power is proportional to $g_m T$ [32]. This means that the signal-to-noise ratio is proportional to $g_m T$. The noise performance can therefore be improved in two ways. First, it is desirable that the transconductance of the input transistors M_1 and M_2 be large. This can be achieved by upsizing the input devices until they operate at the onset of weak inversion. Second, the integration time should be increased. This can be achieved by reducing the current of the M_{clk} devices, either by reducing the width of M_{clk} or by reducing the input common mode of the comparator. The integration time is limited by the speed requirement of the comparator.

The speed of a regenerative comparator depends on the value of the input voltage. The comparator makes the decision faster if the input voltage is larger. Therefore, it is important to design the comparator so it can resolve the input voltage that is much smaller than one LSB for the highest intended sampling frequency. Still, for very small inputs, the comparator will not be able to make a decision, and the cross-coupled inverter pair in the comparator core will be in a metastable region. This means that the voltages at the nodes yp and yn will be close to the mid-rail. A special care needs to be taken so that, even with this voltage close to the mid-rail, the output nodes op and on have a valid logic zero level. This can be achieved by increasing the value of the metastable voltage at the nodes yp and yn or by decreasing the threshold of the inverters that produce the outputs op and on . The metastable voltage at yp and yn can be increased by increasing the driving strength of PMOS transistors in the comparator core relative to the NMOS transistors, either by increasing their size or by choosing a device type with a lower threshold voltages. The threshold of the output inverter can be made smaller by increasing the driving strength of its NMOS transistor relative to the PMOS transistor. The output of the comparator is stored in a SR latch, which is a part of the SAR logic explained in the next section. The latch is a regenerative circuit with positive feedback and can also enter a metastable state. A similar technique of skewing inverter thresholds should be applied to block the propagation of the invalid logic level from the latch to the DAC switches. These techniques prevent excessive power consumption and reference disturbance due to a short circuit in the logic gates or in the DAC switches. They do not solve the problem of the conversion error due to metastability that can occur if the SR latch takes too long to resolve its output. The probability of the metastable error should be reduced to an acceptable level for a given application by reducing the time constants in the comparator and the latch relative to the comparator strobing period.

The calibration algorithm described in the Chapter 3 calibrates comparator offsets of all channels. The price paid for digital calibration of offsets is that different ADC channels clip at different values of the input signal. This effectively reduces the available input signal range by $V_{OS,max} - V_{OS,min}$, where $V_{OS,max}$ and $V_{OS,min}$ are the maximum and the minimum offset value among all interleaved channels. Monte Carlo simulations with global process variations and local device mismatches are used to determine that the standard deviation of the comparator offsets is approximately 12 mV after noise requirements were satisfied. Since the analog front-end of the ADC was designed for rail-to-rail operation, the reduction of the dynamic range due to offsets is smaller than 4% with 99.8% probability, and no coarse calibration of offsets in analog domain was needed.

A comparator decision does not depend only on the value of the signal at its inputs. It also depends on the initial voltages at the internal nodes of the comparator, which depend on the previous decision of the comparator. This memory effect is known as the comparator hysteresis, and it can be eliminated if the internal nodes of the comparator are always reset to the same voltage before the next comparison is started. Transistors $M_{r1} - M_{r5}$ reset the internal nodes of the comparator to V_{DD} . Since resetting these nodes exactly to V_{DD} requires relatively large reset transistors, the transistors M_{r5} and M_{r6} are introduced to balance both

sides of the comparator during the reset.

During comparison phase the internal nodes of the comparator can couple to the input nodes causing disturbance at the input voltage. This effect is known as the comparator kickback. This disturbance can propagate to other sensitive circuits, such as other comparators. In our case, all channels use only one comparator, so the only problem is interference between the comparators in different channels. This interference would mostly occur through the common reference voltages. At precision levels of interest, this problem can be easily eliminated by using large decoupling capacitors on the reference voltages and low-resistance interconnect for the distribution of the reference voltage. It should be also noted that having smaller capacitors in the capacitive DAC reduces the propagation of the kickback noise from one channel to the others.

5.1.4 SAR Logic

The SAR logic accepts the outputs of the comparator and produces the control signals for the switches in the capacitive DAC in order to perform a radix-based search algorithm. The schematic of the SAR logic is shown in Figure 5.7. It consists of SR latches with accompanying logic, a shift register, SAR unit cells, and switch drivers. The input signals to the SAR logic are *reset*, *start*, *clk*, *cmp*, and *rs/ds*. *reset* is the global synchronized reset signal and is active low. *start* is a buffered version of the bottom-plate sampling clock, also active low. *clk* is the clock signal for SAR logic derived by dividing the master clock by two. A delayed version of this clock is also used to clock the comparator. *cmp* is the output of the comparator, and *rs/ds* is a control signal that selects either reverse or direct switching mode. The output signals of the SAR logic are $D_{10} - D_0$, *clkout*, nmn_i , nmp_i , pmn_i , pmp_i , $i = 0A..10$. $D_{10} - D_0$ are the raw output conversion bits. *clkout* is the output clock signal used in the calibration logic. nmn_i , nmp_i , pmn_i , pmp_i , $i = 0A..10$ are the signals that drive the switches in the capacitive DAC.

The SR latches produce the *cmpen* signal, which, when low, disables the comparator clock and turns off all the switches from the capacitive DAC, thus enabling the sampling of the input signal. When *reset* is low, the *cmpen* is also low. After the *reset* is released, and after the first negative pulse of the *start* signal, the *cmpen* can be set high by the t_0 , which marks the beginning of the first bit-testing phase during the conversion process. The 12-bit shift register is asynchronously initialized to '100...0' by the negative pulse of the *start* signal before every conversion. After the negative pulse of the *start* signal, *clk* goes high, producing $t_0 = 1$. t_0 initializes the state of all *sar_cell* elements to $D_{10}D_9...D_1D_0 = 10...00$. The signals $D_{10}D_9...D_1D_0D_{0A}$ drive the switches of the capacitive DAC through the *sw_drv* drivers. The capacitive DAC settles while $clk = 1$. After the falling edge of *clk*, the 1 in the shift register is shifted to the next position, and the comparator is triggered. On the rising edge of *clk*, the next state (D_9) is set to 1, and the previous state (D_{10}) is reset to zero if the comparator output is equal to 1. This process continues until all bits are resolved. The *clkout* signal resets D_0 if $cmp = 1$, and its falling edge is used to write the state of

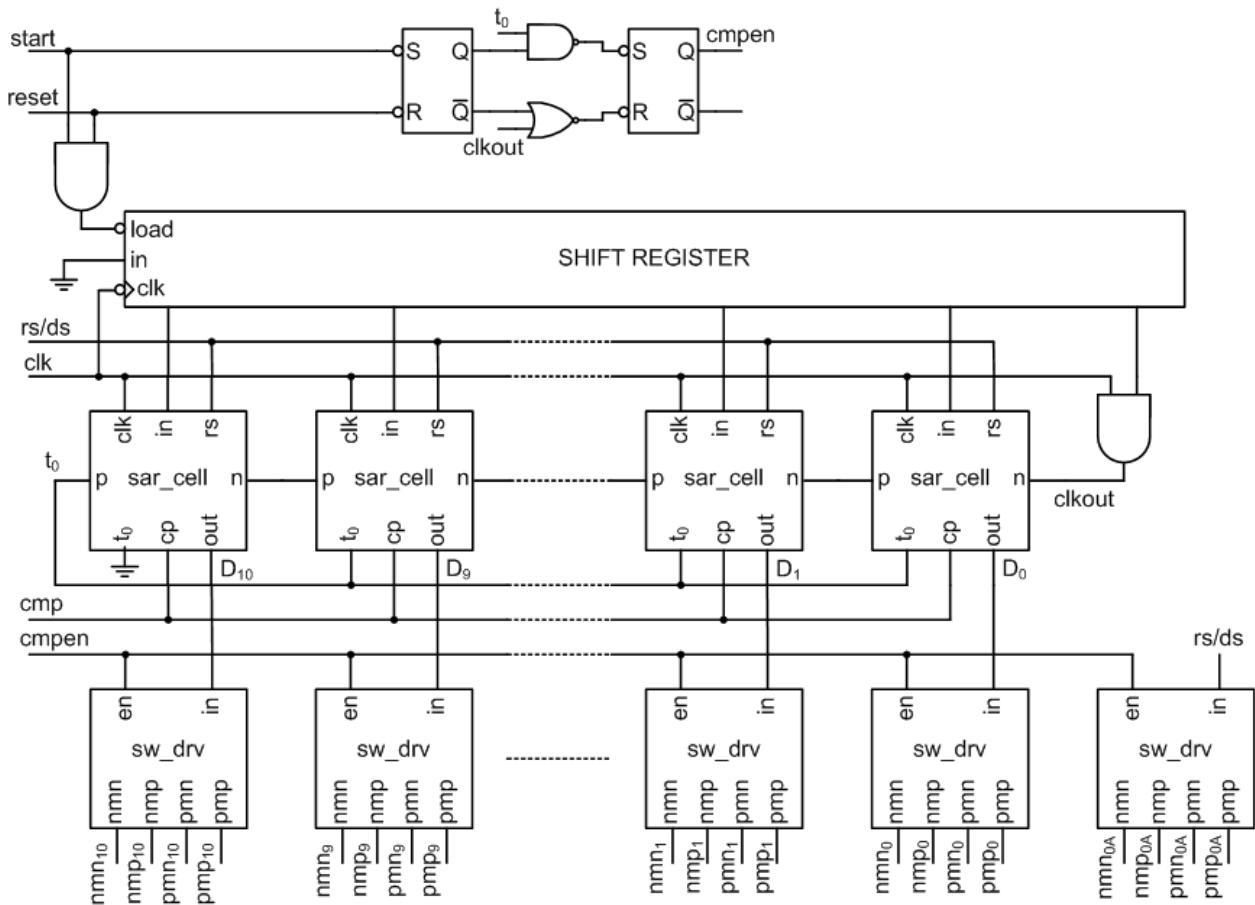


Figure 5.7: Schematic of the SAR logic.

the *sar_cell* blocks, which represent the final conversion output, to a register. *clkout* also resets the *cmpen* signal, its rising edge is used to clock the calibration logic that follows the interleaved channels, and a buffered version of *clkout* is used to drive the reset switches in the capacitive DAC.

The schematic of the *sar_cell* block is shown in Figure 5.8. The SR latch is set when both *in* and *clk* are high, and reset either when *t₀* is high or both *cmp* and *n* are high. The state of the *sar_cell* is equal to the state of the SR latch, if direct switching is being performed (*rs* = 0), or to the complementary value of the SR latch’s state, if reverse switching is selected (*rs* = 1).

The schematic of the *sar_drv* block is shown in Figure 5.9. It is a simple combinational logic that produces the appropriate logic values necessary to drive the DAC switches. A special care needs to be taken during the design to ensure that no low-resistance path is created between V_{rp} and V_{rn} during the transient switching.

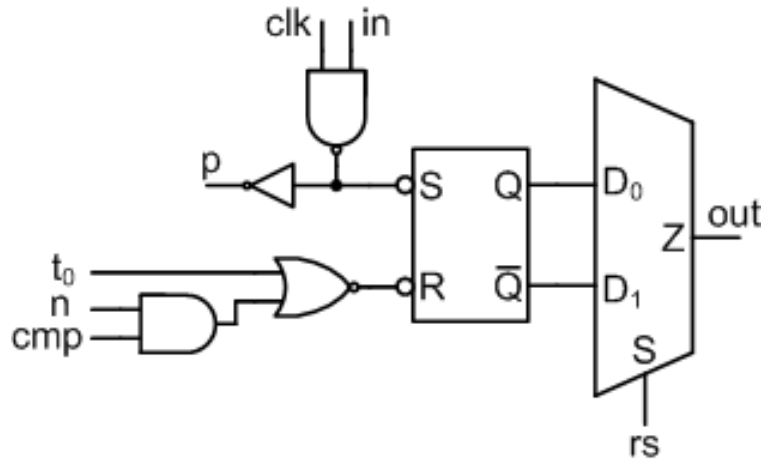


Figure 5.8: Schematic of the sar_cell block.

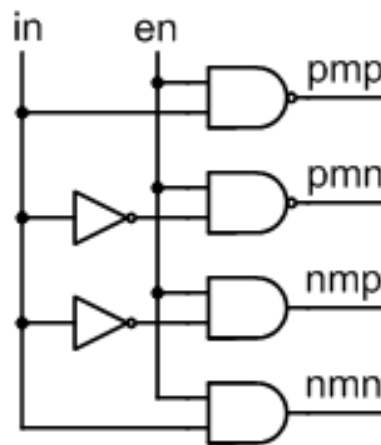


Figure 5.9: Schematic of the sw_drv block.

5.1.5 SAR Layout Plan

The layout of a SAR ADC that is used as a channel in a time-interleaved ADC can differ significantly from the layout of a single-channel SAR ADC. The minimization of wiring parasitics in a single-channel ADC often results in an approximately square shape of the layout. In high-performance time-interleaved ADCs the distribution of the common sensitive analog signals and supply lines gets a higher priority compared to the local wiring parasitics. To minimize the length of the wires used for these analog signals, the height of the ADC channel needs to be as small as possible. This results in a layout with a large aspect ratio. A good empirical rule is that the layout of the resulting time-interleaved ADC should have approximately a square shape.

The layout plan used in this design is shown in Figure 5.10. The aspect ratio used is roughly 6:1. The layout is divided into four parts: top-plate switch, comparator and bottom-plate switch, DAC and SAR logic. The DAC occupies the largest area and is placed in the middle of the layout, between the SAR logic, on the right, and the comparator, the bottom-plate switch and the top-plate switch, on the left. This arrangement requires relatively long wires to connect the DAC to the SAR logic and the top-plate switch, but it is desirable since it minimizes the height of the ADC channel ($30\ \mu\text{m}$ in this design).

The arrows indicate the location and the routing direction of the common signals and supplies. The analog input signal and the sampling clock are routed outside of the ADC channels in parallel vertical metal-7 wires and connections to the channels are made from the left side. This minimizes the systematic timing skew between different channels if the delay on the clock and the input signal lines is approximately equal. In order to minimize the coupling between the input and the clock, the distance between the clock and the input signal wires is made much larger than the distance from the metal 7 to the substrate, and twisting of the wires is used for the clock signal. Additionally, the ground and the clock supply lines are routed in multiple metal layers between the clock and the input wires (not shown here). Multiple reference lines (one of the references is ground) are routed above the DAC. This provides the low impedance on the references and it also provides additional isolation of the sensitive comparator inputs. Analog supplies and the common-mode bias voltages are routed above the analog circuits, while the digital supply is placed above the SAR logic, same as the SAR logic clock and the reset signal.

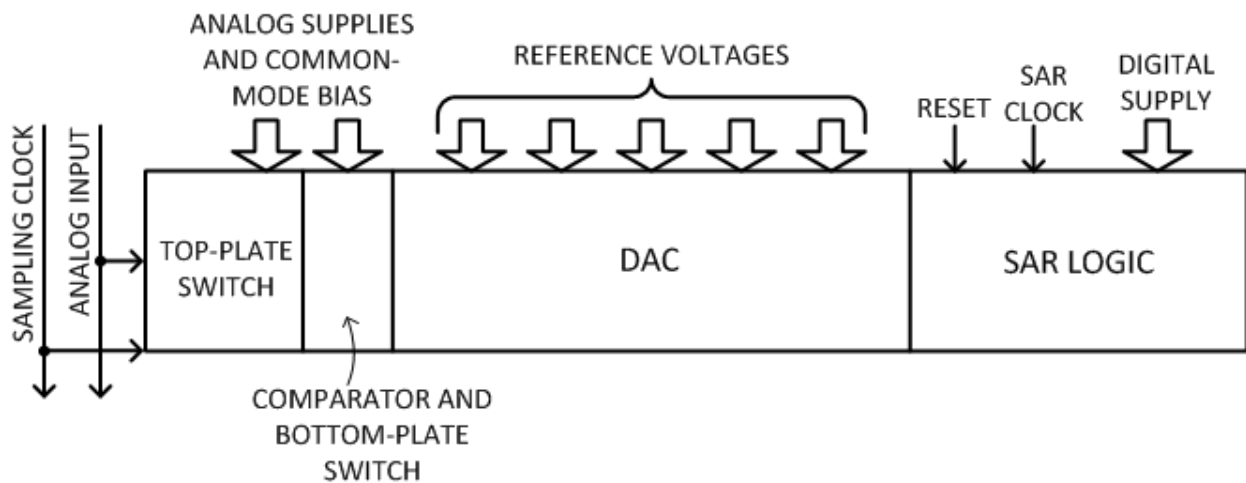


Figure 5.10: Layout plan of a SAR ADC channel.

5.2 Clock Generation

Providing multiple phases of the clock to different channels is one of the most challenging problems in a time-interleaved ADC architecture. An alternative solution of having a single front-end sampler that operates from one phase of a full-speed master clock is possible[10], [13], [15], [16], but it requires resampling and buffering, which increases the noise and the power. Sampling the input signal directly in different time-interleaved channels can lead to significant power savings if sufficiently accurate multiple phases of the clock can be generated in a power-efficient way, and if the input signal can be delivered to the multiple channels through an analog front-end network that has sufficiently low bandwidth mismatch. This section describes a simple clock generation scheme that produces multi-phase clocks with low timing skew and low jitter.

A simplified schematic of the sampling network of the time-interleaved ADC is shown in Figure 5.11, together with timing diagrams of the different clocks needed for the circuit operation. C_S , although shown as a single sampling capacitor, represents the whole bank of

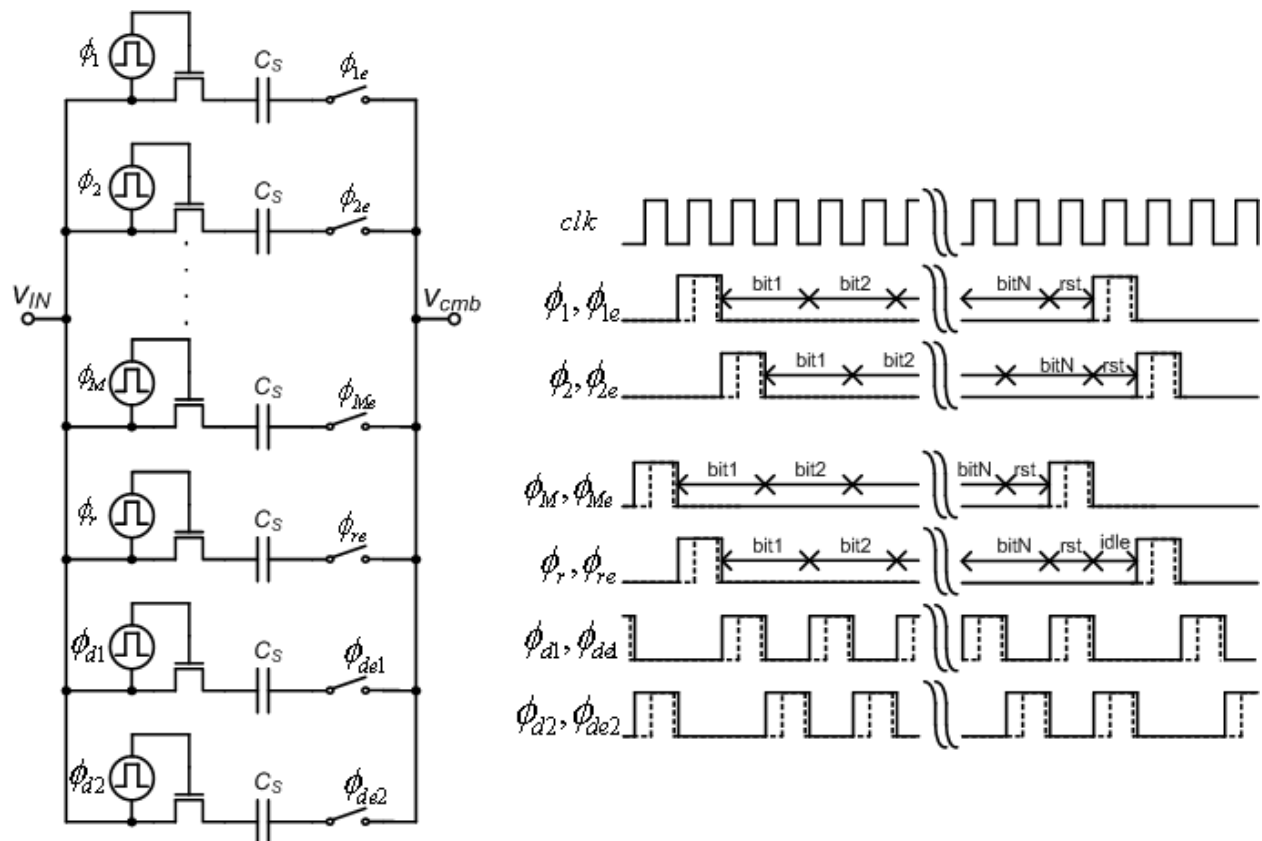


Figure 5.11: Clock signals with timing diagrams.

capacitors from the capacitive DAC. In addition to the $M(= 24)$ time-interleaved channels and the reference channel, two dummy channels sample the input signal while the reference channel is performing the conversion. This way the same impedance is present at the input of the ADC for every sample. Also, the second reference channel for timing calibration with its set of two dummy channels is implemented, but not shown in the Figure 5.11 since its clocks, while being physically separate signals, have the same timing diagrams as the ones for the first reference channel. Every channel needs two clocks, one for the top-plate switch, and the other for the bottom-plate switch. The falling edge of the bottom-plate clock needs to occur before the top-plate switch closes in order to get the benefits of the bottom-plate sampling. This way, the bottom-plate switch always has the same terminal voltages and "sees" the same impedance while sampling the input signal. Consequently, the charge injected by the bottom-plate switch is independent of the input signal and it contributes only to a constant offset. Also, since the action of sampling is performed by the bottom-plate switches, the precision requirements for the bottom-plate clocks are much more stringent.

The top plate clocks ϕ_1, \dots, ϕ_M have the same frequency of f_s/M , but different phases equal to $k\frac{2*\pi}{M}, k = 0..M - 1$. They are generated using a simple circular shift register. The shift register is clocked by a buffered version of the master clock. After the global reset is released, and after the first rising edge of the master clock, ϕ_1 is set to one and ϕ_2 to ϕ_M are set to zero. The one in the circular shift register is shifted every rising edge of the master clock, so only one of the $\phi_1.. \phi_M$ is active at a time, which creates the desired multi-phase clocks. ϕ_r, ϕ_{d1} , and ϕ_{d2} are generated by flip-flops that are clocked by the same clock as the circular shift register to achieve the same delay from the master clock to multi-phase clocks. The inputs of these flip-flops are controlled by a digital counter with modulo $M + 1$ and a simple digital logic. The frequency of the reference clock is $f_s/(M + 1)$ and its first pulse after the reset coincides with the first pulse of ϕ_1 .

The circuit that generates the bottom-plate clocks is shown in Figure 5.12. The input clock is a sinusoidal differential signal and it is AC-coupled to the chip using the big capacitors C_B . The first stage buffer is common for all channels, thus minimizing the timing skew between channels, and it is realized as a simple inverter with low-threshold transistors and a large bias resistor $R_{BIAS} \approx 80 \text{ k}\Omega$ connected between the input and the output of the buffer. The output of the first stage buffer is distributed to all channels using thick low-resistance metal-7 wires. In each channel, the pseudo-differential clock is gated by two simple CMOS switches. These switches are controlled by the same clock signals used to drive the top-plate switches. The second (and last) stage of buffering is also realized as simple inverters with low-threshold devices. The bottom-plate switches are CMOS switches and are driven pseudo-differentially. A simplified version of the bottom-plate switches without dummies is shown in Figure 5.12. The big resistors $R_{BIG} \approx 45 \text{ k}\Omega$ at the input of the second-stage buffers keep the bottom-plate switches closed during the conversion process. The variable capacitors C_{D2T} are used for fine-tuning of the edges of the sampling clocks, and their value is controlled by the timing calibration algorithm.

An interesting property of this simple two-stage buffering scheme is that it can effectively

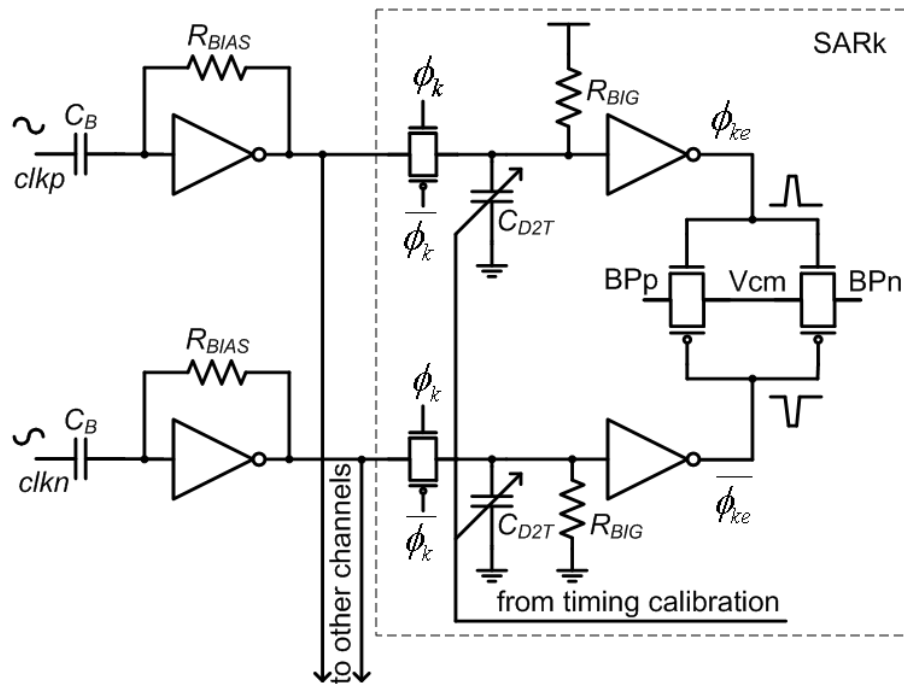


Figure 5.12: Low-jitter bottom-plate sampling.

reduce the jitter coming from the supply noise, if buffers are designed with fast low-threshold devices, and if used with appropriate switches. The effect of the supply change on the edges of the sampling clock is shown in Figure 5.13. The dotted and dashed-dotted lines show the edges when the supply voltage is higher and lower than nominal, respectively. The rising and the falling edges of the clocks shift in opposite directions in a way that makes the voltage difference between the clocks less dependent on the supply voltage. This may be counterintuitive to someone who is used to think of inverters as of digital circuits whose delay is inversely proportional to the supply voltage, but it can be easily understood if the two-stage buffers are treated as fast analog amplifiers. If the transconductance of the CMOS sampling switch is flat around $V_{cm} \approx V_{DD}/2$ then the transconductance change in time, and, consequently, the effective sampling instance, will depend on the voltage difference between ϕ_{ke} and $\bar{\phi}_{ke}$, rather than the individual voltages of ϕ_{ke} and $\bar{\phi}_{ke}$. This is exactly the effect achieved by the shifting of edges in the opposite direction, as shown in Figure 5.13. Effectively, this reduces the jitter caused by the supply noise. Circuit simulations of jitter expressed as the ENOB calculated at $f_{in} = 1.5$ GHz versus the RMS value of the supply noise in the case of single-ended and pseudo-differential sampling are shown in Figure 5.14. It can be seen that for a desired ENOB of 9 bits the RMS supply noise of higher than 15 mV can be tolerated in the pseudo-differential case versus only around 1.5 mV in the single-ended case.

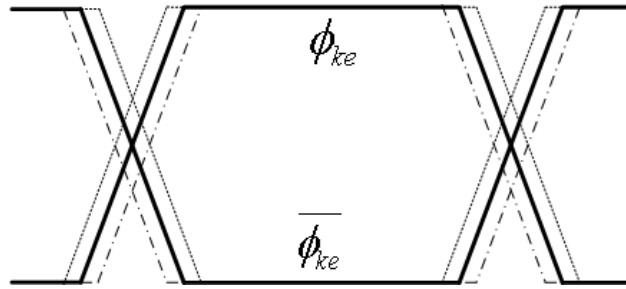


Figure 5.13: Effect of supply voltage change on the sampling-clock edges.

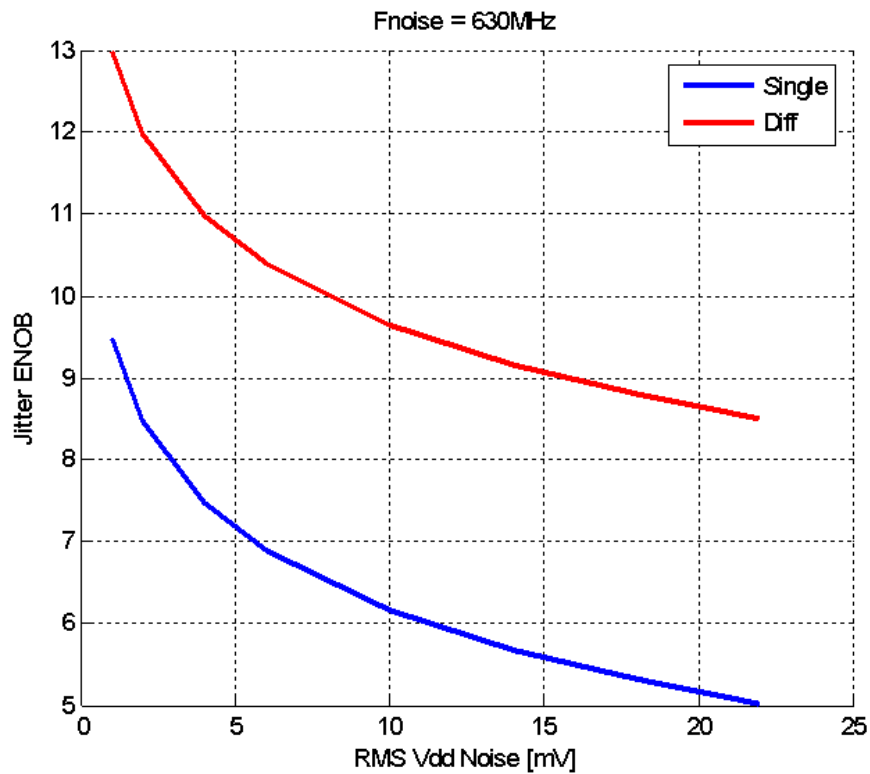


Figure 5.14: Jitter ENOB calculated at $f_{in} = 1.5GHz$ vs. RMS supply noise for single-ended and pseudo-differential sampling.

The implementation of the clock tuning is shown in Figure 5.15. The variable capacitors C_{D2T} are implemented as a bank of 31 small MOS capacitors of approximately 5 fF that can be switched in or out of the clock path, enabling a 5-bit control of the capacitor value. This simple circuit acts as a digital-to-time converter. To ensure monotonicity, the switching of the MOS capacitors is controlled by thermometer-coded outputs from the timing calibration

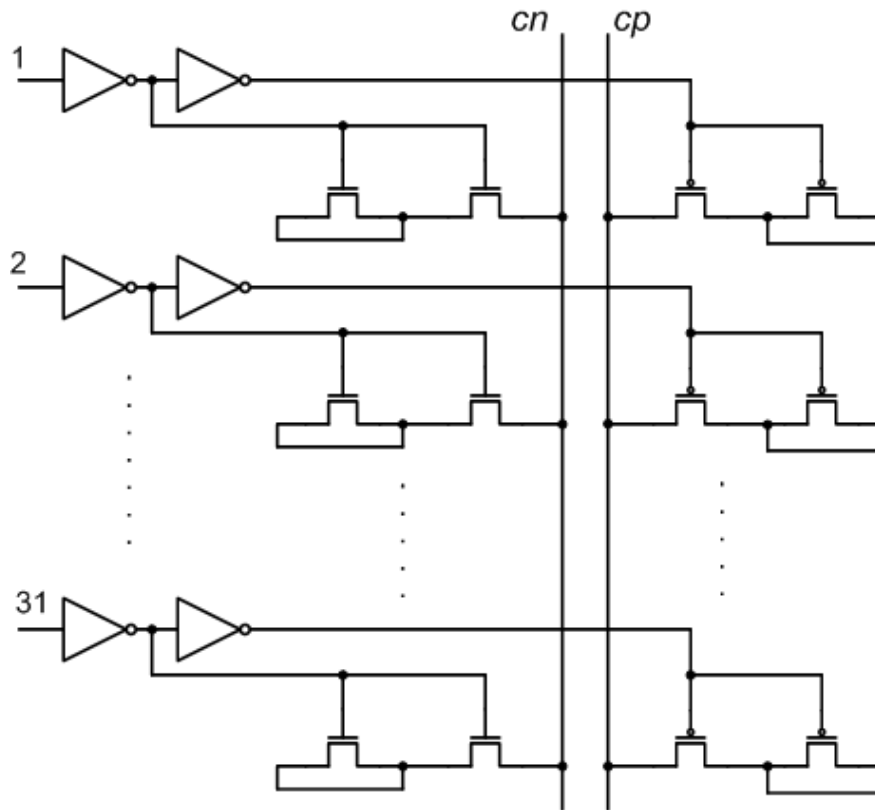


Figure 5.15: Implementation of clock tuning.

algorithm. Since the values of the control word can only change by plus or minus one (by algorithm design), the thermometer coding is implemented by a 31-bit shift register. The bits 1-15 are initialized to 0 and the bits 16-31 to 1 to set the initial state at the middle of the tuning range, so that both positive and negative timing errors can be corrected. The increment or decrement of the control word value is performed by two signals, *up* and *down*, that come from the digital calibration logic. The tuning resolution is $\Delta t = 300$ fs and the tuning range is approximately ± 4.8 ps.

5.3 Calibration Logic

Calibration logic consists of two parts: the first one calculates the weighted sum of digital raw output bits from all ADC channels, and the second one implements the LMS algorithm and iteratively calculates the value of digital weight coefficients. The first one needs to be running all the time, while the second one can be shut down after the calibration converges, can be run periodically to track the environment changes, or can be run continuously if it can

be guaranteed that the input signal is going to be 'busy'. The inputs to the calibration logic are the raw output bits from all ADC channels, M multi-phase clocks at f_s/M frequency generated by the time interleaved ADC channels, and a clock at $f_s/(M + 1)$ frequency from the reference channel. The outputs of the calibration logic are final conversion results. The calibration logic can be bypassed in order to capture the raw outputs for testing purposes.

The summation logic performs the following operation:

$$D^j = \sum_{i=0}^{N-1} C_{di}^j d_i^j + V_{OSdi}^j, \quad (5.2)$$

where C_{di}^j , d_i^j , V_{OSdi}^j , and D^j are the i^{th} digital weight coefficient, raw digital bit, offset coefficient, and final conversion result, respectively, all in the j^{th} channel. The multiplications from (5.2) can be realized as simple AND logic operation since d_i^j are single bits that can be either zero or one. All $N + 1$ coefficients in (5.2) are prone to quantization errors. Assuming that all the coefficients are represented with the same accuracy, the final result can have $N + 1$ times larger quantization error in the worst case. To minimize the quantization error in the final result, the coefficients are represented with higher accuracy (4 binary places), and the final result is truncated to have only one binary place, for a total of 11 bits. The length of the coefficients is optimized to minimize the area and the power of the digital logic. The MSB and LSB coefficients are represented using 14 and 8 bits, respectively.

The logic that implements the LMS algorithm performs the following computations:

$$C_{di,j+1}^k = C_{di,j}^k + 2\mu e d_i^k \quad (5.3)$$

$$V_{OSdeq,j+1}^k = V_{OSdeq,j}^k + 2\mu e \quad (5.4)$$

$$\begin{aligned} \Delta t_{dig}^{i+1} &= \Delta t_{dig}^i + 2\mu_{\Delta t} \operatorname{sgn}(e) \operatorname{sgn}(D) \quad , \quad (|D| \geq \delta) \\ \Delta t_{dig}^{i+1} &= \Delta t_{dig}^i \quad , \quad (|D| < \delta) \end{aligned} \quad (5.5)$$

as described in Chapter 4. In addition to (5.3), (5.4), (5.5), arithmetic overflow checks need to be performed, and, if an overflow is detected, the maximum or minimum allowed value for a given coefficient is chosen.

The calibration logic has been described in the Verilog hardware description language. Synthesis from a register transfer level description and place-and-route have been performed to obtain the final physical design with the area of $0.23 \text{ mm} \times 0.76 \text{ mm}$ and an estimated power of approximately 10 mW. The circuit's functionality has been verified using mixed-signal simulations with models of the analog part of the chip. The timing has been verified using static timing analysis on the circuit with extracted parasitics.

5.4 Full-Chip Integration

Integrating all different blocks previously described on a single die, while preserving the integrity of sensitive signals, can be a challenging task, and careful planning and layout are necessary. The layout of the whole chip is shown in Figure 5.16. The analog part of the chip is on the right-hand side. The odd- and even-numbered channels are physically separated and have separate reference voltages. This is to minimize the cross-talk between even and odd channels that have 180° shifted operation of their SAR logic, and, consequently, their DAC settling and comparator active time are out of phase. Large decoupling capacitors of approximately 0.8 nF are used for both sides of the reference voltages. The input and clock signals are fed to the chip in a fully differential fashion from the opposite sides of the chip (top and bottom side in Figure 5.16), and are routed in thick metal in the same direction alongside the time-interleaved channels to minimize the phase mismatch coming from the finite speed of the signal propagation. Separate supply voltages are used for clock generation and distribution circuits, comparator and top-plate switches, and SAR logic. The digital calibration logic is inside the block named 'filter'. To facilitate testing and to minimize the number of pins, a large memory block is placed on the chip, and it is used to capture the conversion outputs from all interleaved and reference channels. The memory content is slowly read out using a scan-chain circuits. The scan chain is also used to set some control signals and to read the values of internal registers.

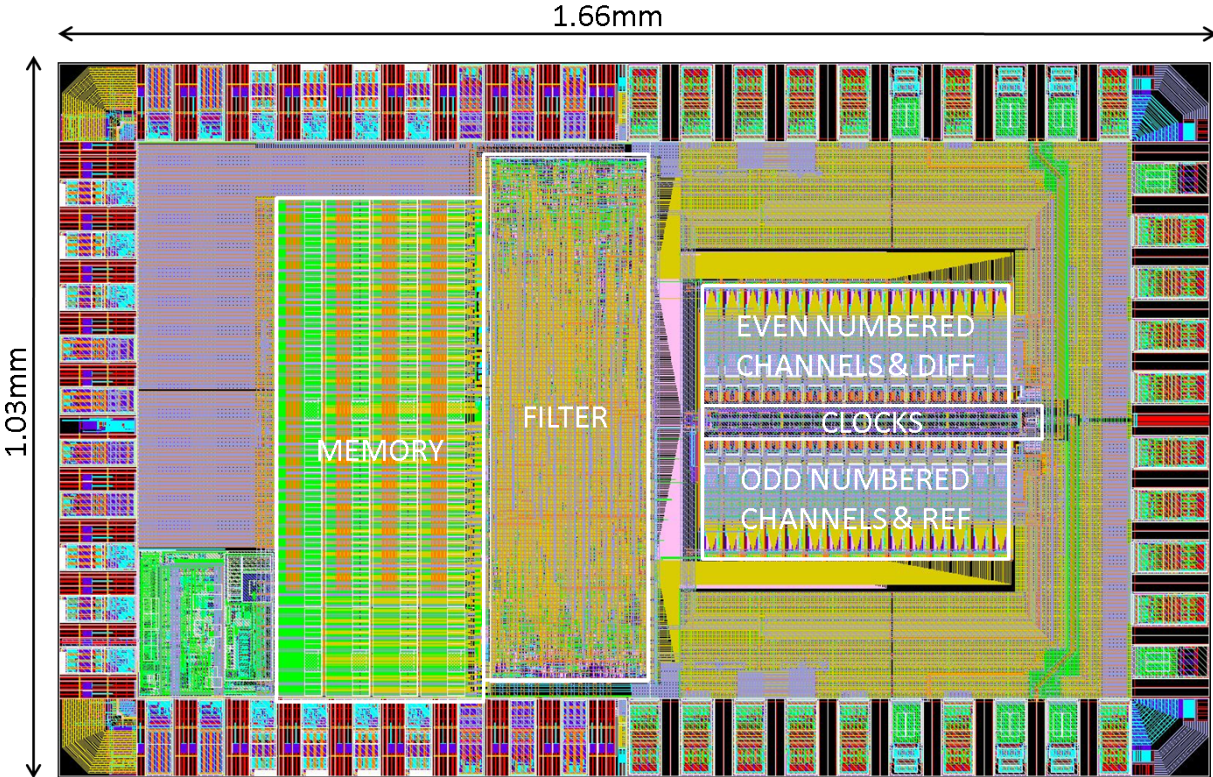


Figure 5.16: Chip layout.

Chapter 6

Measurement Results

This chapter explains the measurement setup and shows the measurement results of the ADC prototype. First, the values of the radices in the capacitive DAC and timing mismatches have been measured in order to confirm that they are inside the correction range of the calibration algorithm. Then, a single-tone test has been performed for different input and sampling frequencies, input signal amplitudes, and supply voltages. Finally, the output for two-tone input signal has been measured to evaluate the intermodulation distortions.

6.1 Measurement Setup

The challenge of testing a sensitive analog circuit, such as a high-performance ADC, is in separating the non-idealities coming from the laboratory equipment and the environment from the nonidealities of the sensitive circuit itself. A well-designed measurement setup will keep the equipment and environment nonidealities below the precision level of the device that is being tested. A block diagram of the measurement setup used to obtain the results presented in this chapter is shown in Figure 6.1. Low-jitter signal generators need to be used as both the input and the clock signal. Since the harmonic distortion performance of a typical signal generator is on the order of 30 to 50 dB below the fundamental [2], [3], a bandpass filter is used to filter out the undesired harmonics and tones from the input signal. Bandpass filtering is also applied to the clock signal to help further reduce the wideband phase noise. A standard laboratory voltage supply source is used to provide a single 3.5 V supply to the testing board. The data and control signals are communicated between the testing board and a computer running the Matlab software.

A block diagram of the test board is shown in Figure 6.2. The power section consists of linear voltage regulators and decoupling capacitors that provide clean voltage supplies to different parts of the chip. For improved flexibility, five voltage regulators with adjustable output voltages are used to generate supplies for the clock generation circuits, comparators and bootstrapped switches, references, digital circuits, and the pad ring. The input and

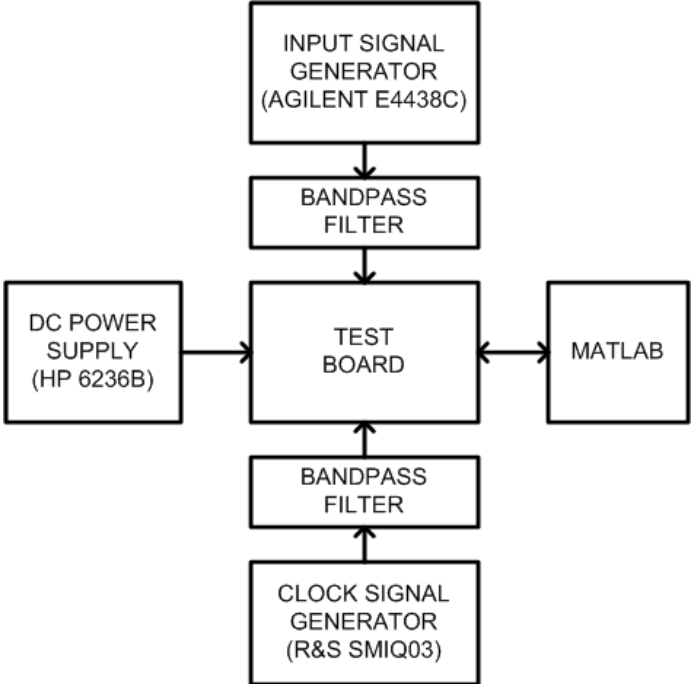


Figure 6.1: Block diagram of the measurement setup.

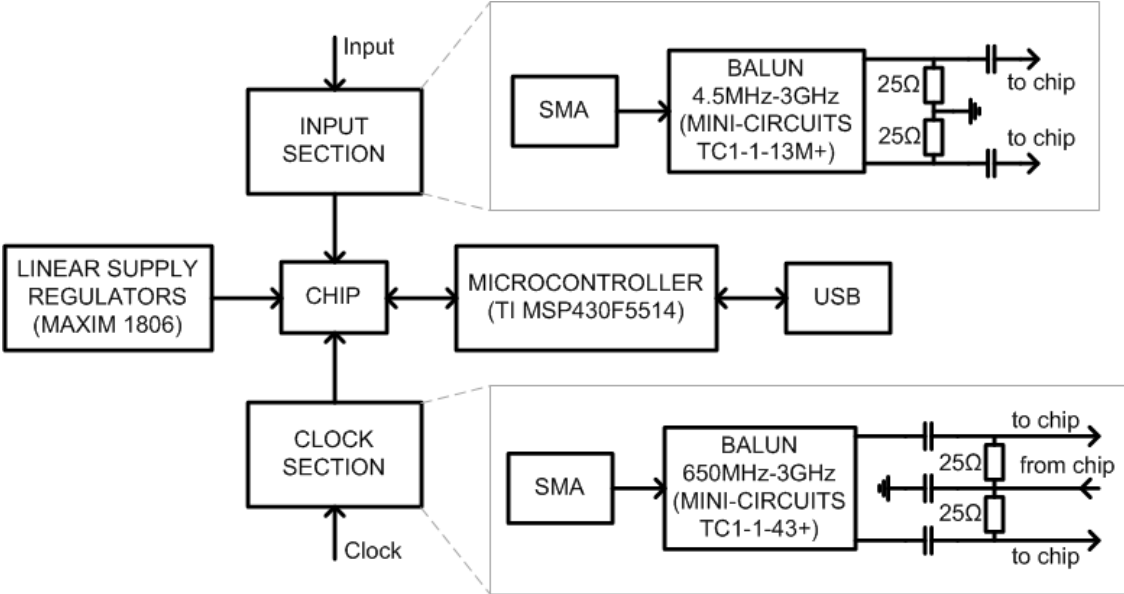


Figure 6.2: Block diagram of the testing board.

clock sections consist of an SMA connector, a wideband transmission-line-based balun with 1:1 ratio, termination resistors, and AC coupling capacitors. A microcontroller has been put on board to facilitate communication between the chip and the computer for data post-processing. The microcontroller communicates with the chip through a scan chain, and with the computer via a USB serial interface. The chip has been assembled directly on the board using chip-on-board assembly technique. A photo of the chip is shown in Figure 6.3.

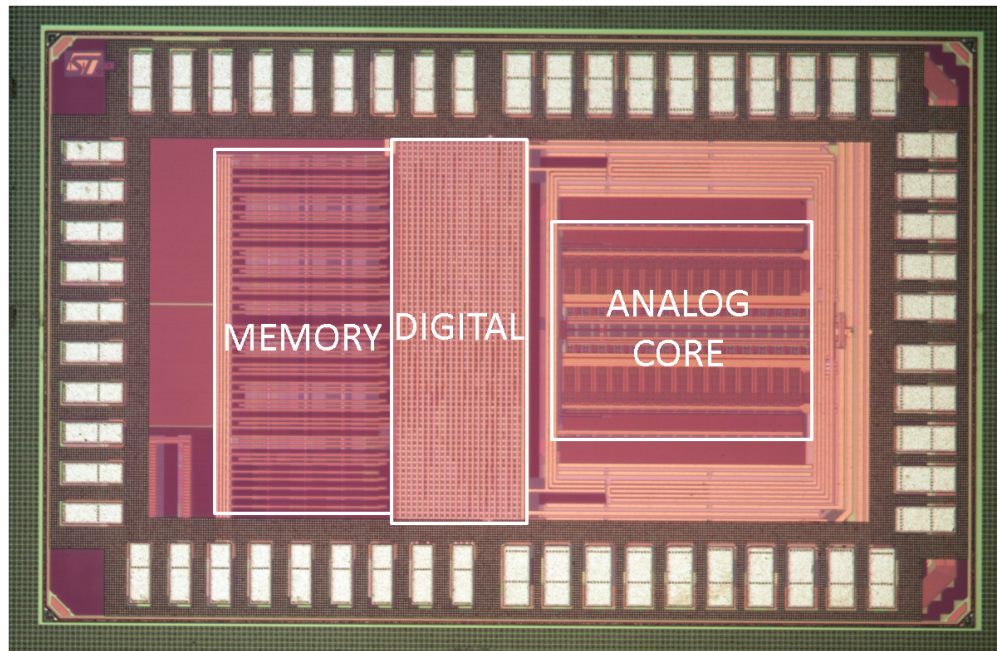


Figure 6.3: Chip photograph.

6.2 Radix Measurements

The digital calibration of nonlinearities caused by the capacitor matching is possible only if capacitor ratios satisfy certain conditions. This is one of the major concerns during the design phase, since the capacitor models are not very reliable at sub-fF capacitor values. The radices can be measured by optimizing the SNDR at the output of one SAR ADC channel when a sinusoidal input is applied with respect to the digital weight coefficients. The measurement procedure is as follows. A filtered sinusoidal signal is applied to the input of the ADC and the raw output bits from one of the channels are obtained. Low input and sampling frequencies are used to minimize the effects of dynamic imperfections. The final conversion outputs are formed by multiplying the raw output bits by their corresponding weight coefficients and summing all the products, where the weight coefficients are the unknowns that we want to measure. Out of all possible coefficient values, the best SNDR is

obtained for the coefficients that correspond to the values of the capacitors in the analog domain. By optimizing the SNDR with respect to the weight coefficients, the value of the capacitors, up to a scaling factor, can be inferred. The radices in the reference channel of four different chips are shown in Figure 6.4. The radices slightly higher than two at lower bit-positions will create a small DNL at the LSB level, which is set to approximately 10 bits by the capacitor sizing in the design phase. This is not a significant factor for our performance goals. Using the same method, the radices in all time-interleaved channels on one of the chips have been obtained. The mean values and standard deviation for all bit positions are shown in Figure 6.5 and Figure 6.6.

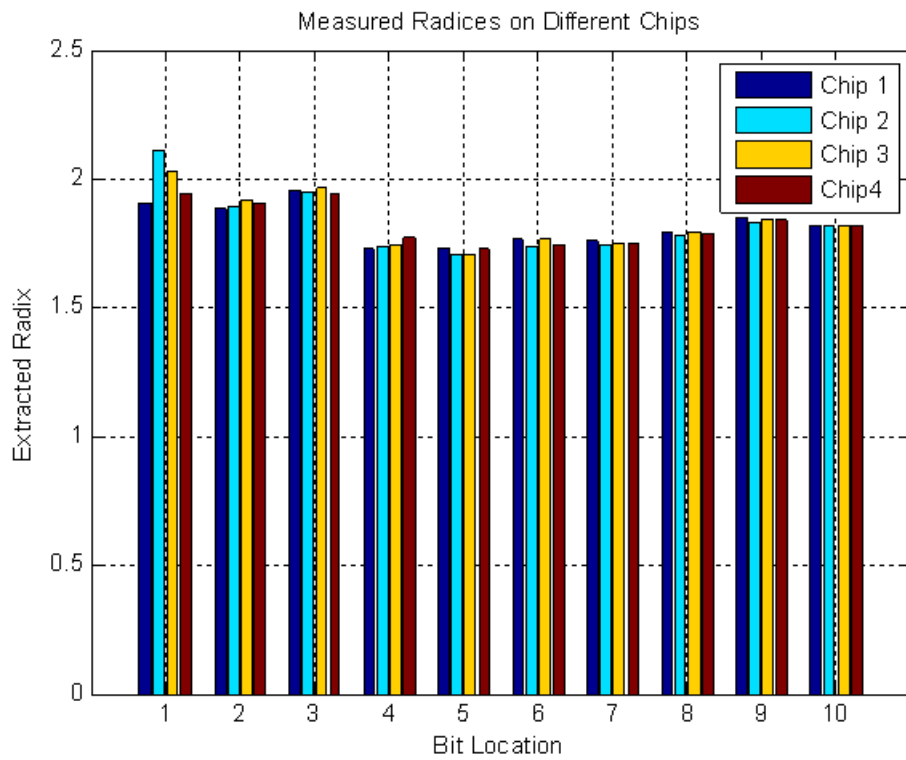


Figure 6.4: Measured radices for 4 different chips.

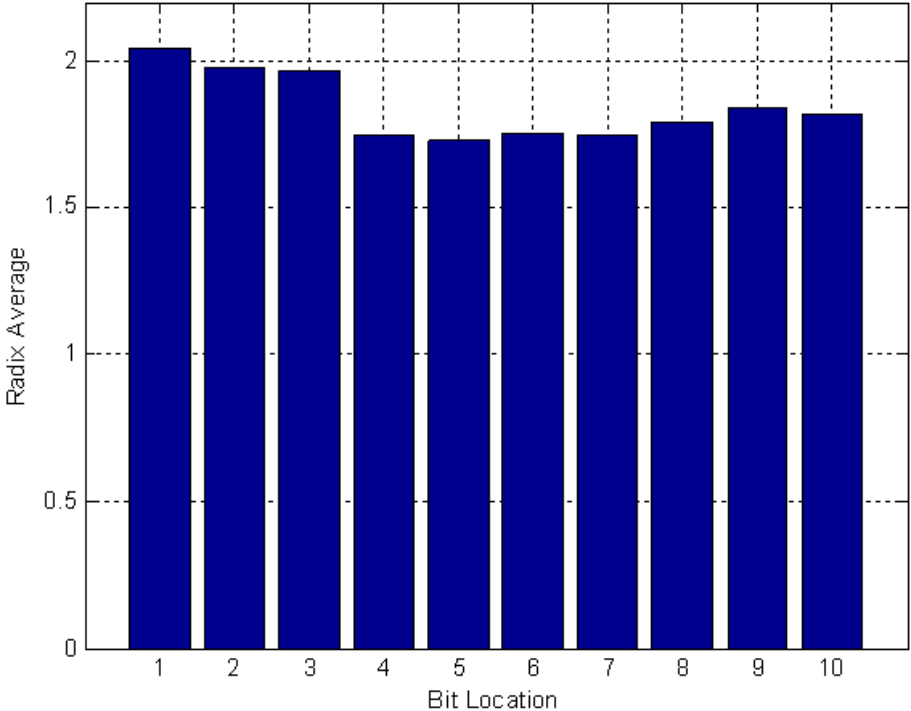


Figure 6.5: Averaged radices across 24 different channels on a single die.

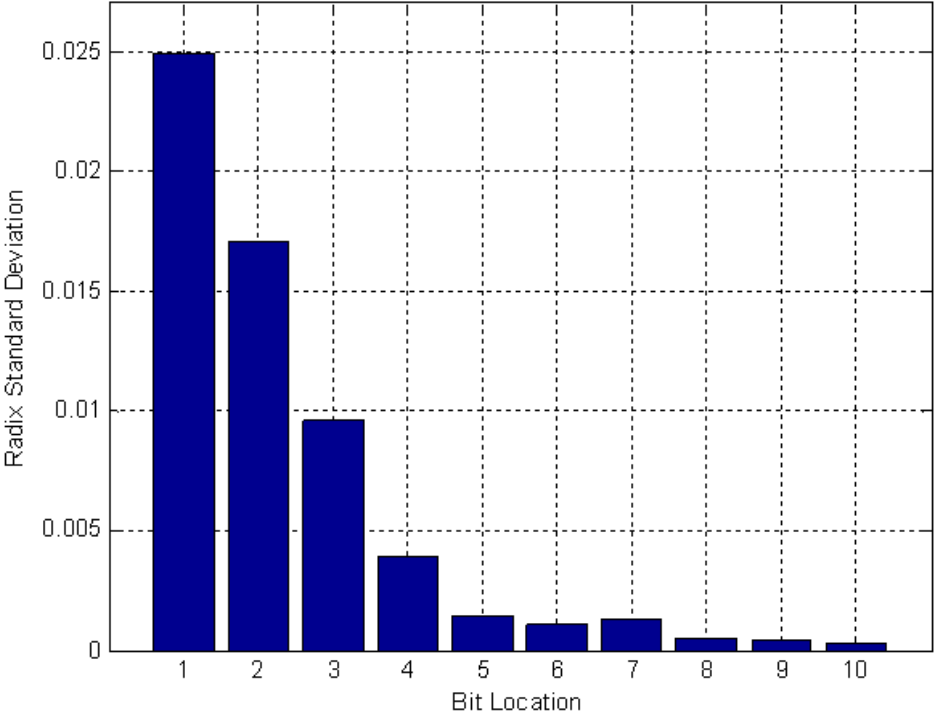


Figure 6.6: Standard deviation of radices across 24 different channels on a single die.

6.3 Timing Mismatches

Another big concern during the design phase is the tuning range of the clock edges. The tuning range has to be relatively small because of the finite resolution of digital-to-time converter and required tuning step, as well as to preserve the falling and rising time of the sampling clock edges. In our design, the tuning range is ± 4.8 ps with a tuning step of 300 fs. Measured converged values of timing mismatches in four different chips are shown in Figure 6.7. The maximum timing mismatch is $\Delta t_{max} = 2.1$ ps, which is more than two times smaller than the available tuning range. The standard deviation of the timing mismatches is $\sigma_{\Delta t} = 0.69$ ps. At the input frequency of 1.4 GHz, this translates into an SDR of 44 dB, which means that even without timing calibration the performance of seven effective bits would have been possible.

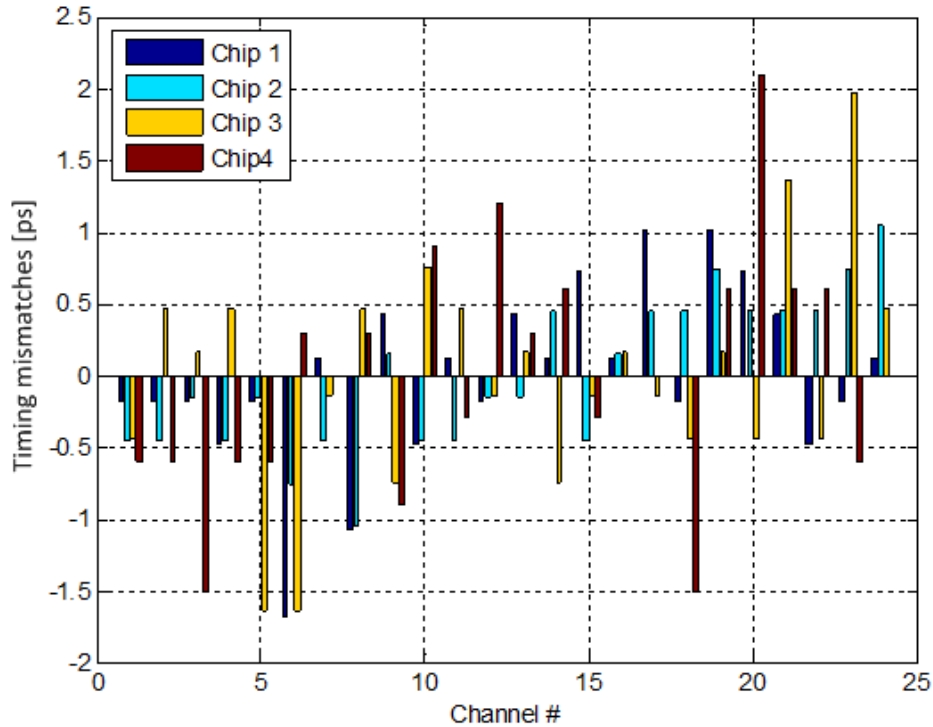


Figure 6.7: Measured timing mismatch for 4 different chips.

6.4 Bandwidth Mismatch

Bandwidth mismatch between time-interleaved channels is equivalent to frequency-dependent gain and timing mismatches. This kind of error is not possible to identify in a single-tone test with continuously running calibration, but can be a big problem in a real-world application. Therefore, it is important to ensure by design that the bandwidth mismatch is sufficiently low. One way to measure the level of bandwidth mismatch is to compare converged values of timing and gain errors at different input frequencies. This has been done with the input frequencies of 450, 900 and 1450 MHz, and the results are shown in Figure 6.8 and Figure 6.9, respectively. Since the parasitic capacitances of the comparator, top-plate and bottom-plate switches are significant compared to the sampling capacitance, the analog front-end does not behave as a single-pole system. The incomplete reset coupled with a narrow sampling pulses further increases the order of the equivalent analog front-end circuit. This makes it difficult to determine the exact value of the bandwidth mismatch, but the difference in the timing and gain errors of the same channels at different frequencies implies that the bandwidth mismatch is sufficiently low to enable more than 60 dB of resolution, which is significantly higher than required by our specifications.

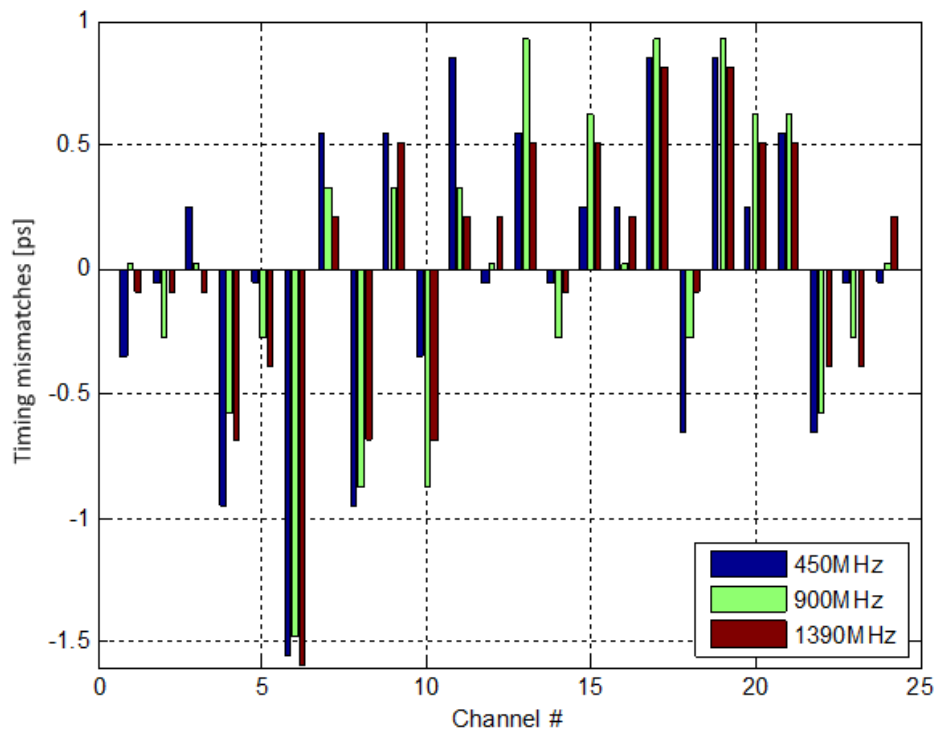


Figure 6.8: Measured timing mismatch for 3 different input frequencies.

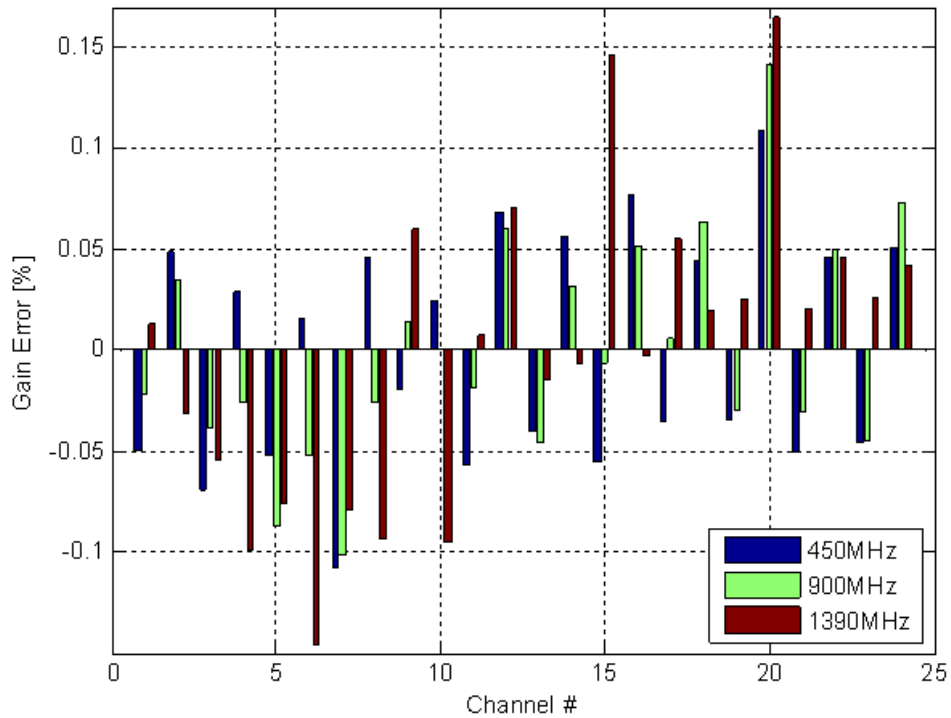


Figure 6.9: Measured gain mismatch for 3 different input frequencies.

6.5 Single-Tone Measurements

Single-tone test consists of applying a sinusoidal analog signal at the input of the ADC and observing its digital output. Discrete Fourier transform (DFT) is usually used to analyze the output signal in the frequency domain. The input and sampling clock frequencies should be chosen so that the maximum number of different samples is obtained. In other words, the transfer characteristic of the ADC should be exercised as much as possible. Also, it is desirable that an integer number of periods of the input signal is sampled, so that the fundamental and all harmonics of the output signal fall into a single bin. If this condition is not met, the output signal needs to be multiplied by a window function before calculating the DFT.

All measurements presented in this chapter were performed with a single set of calibration coefficients. The calibration was done at one input frequency (around 900 MHz) and the sampling frequency of 2.8 GHz. This way, the bandwidth mismatch effect is included in the measurements, since the frequency-dependent gain and timing mismatches are not calibrated at each frequency separately. The amplitude and the frequency of the input signal were

chosen to ensure that the input signal is "busy". The amplitude of 2.2 V was used, which covers almost the full input range, with a sufficient margin to avoid clipping due to different channel offsets. The exact value of the input frequency was chosen such that it provides 10000 different input signal samples from the input signal range. After allowing sufficient time for the calibration convergence, the values of the coefficients were frozen and used in all subsequent measurements.

Examples of the output spectrum after a single-tone test are shown in Figure 6.10 for the input signal frequency of 19.88 MHz, and in Figure 6.11 for the input frequency of 1379.56 MHz. Two DFT plots are shown for each frequency - one before and one after the calibration. The calibration improves both signal-to-noise-plus-distortions ratio (SNDR) and spurious-free dynamic ratio (SFDR) by more than 19 dB.

To verify performance of the ADC across different input frequencies, the input signal frequency has been swept from 5 MHz to 3 GHz. The range of the input frequencies was limited by the bandwidth of the balun that converts the single-ended input signal from the signal generator to a differential signal. The sampling frequency was 2.8 GHz and the supply voltage was 1.2 V. The performance plots are shown in Figure 6.12. The plot shows signal-to-

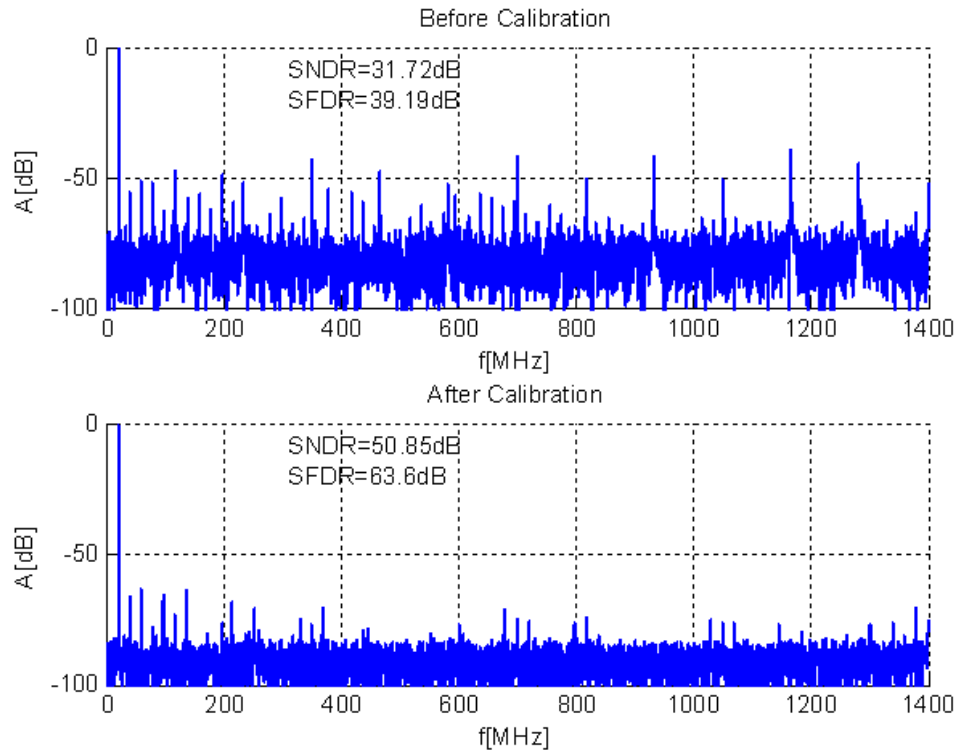


Figure 6.10: Spectrum before and after calibration for $f_{in} = 19.88$ MHz.

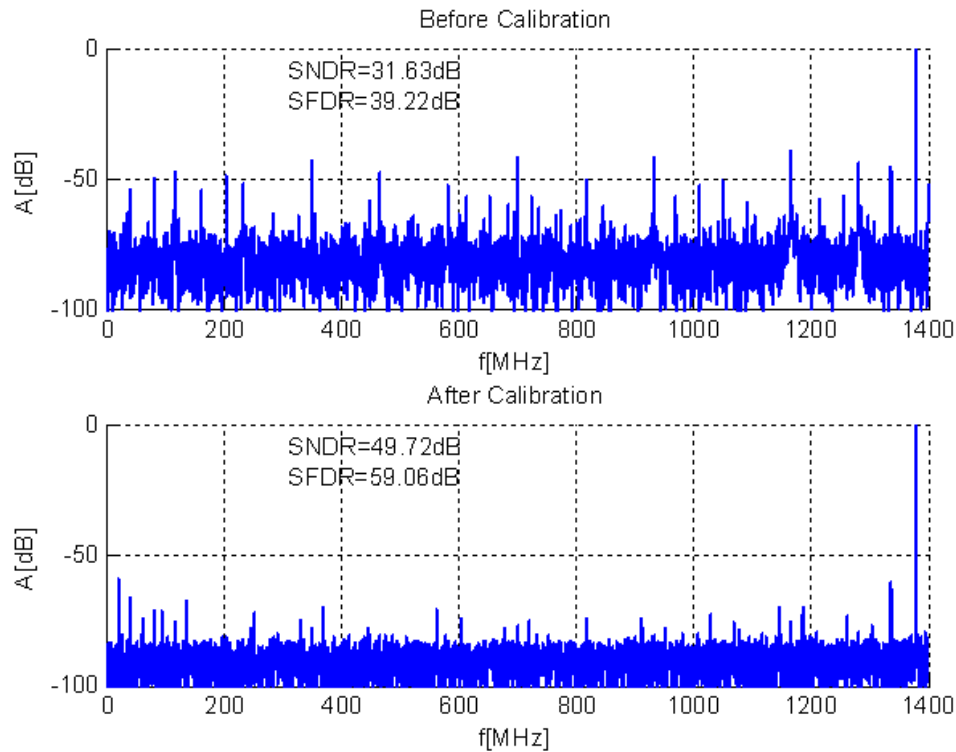


Figure 6.11: Spectrum before and after calibration for $f_{in} = 1379.56$ MHz.

noise-plus-distortions ratio (SNDR), signal-to-noise ratio (SNR), spurious-free dynamic ratio (SFDR), total harmonic distortions (THD), signal-to-timing-and-gain tones ratio (T&G), and signal-to-offset tones ratio (OFFSET). T&G is calculated as the ratio of the input signal power and the sum of powers of all tones produced by the timing and gain mismatches. OFFSET is defined as the ratio of the input signal power and the sum of powers of all tones produced by the channel offset mismatches. SNR curve was obtained by nulling first thirteen harmonics in software post-processing and all interleaved offset, timing and gain tones, and treating the remaining spectral content as noise. A visual inspection has been performed to ensure that no other visible tones were present. The ADC achieves the SNDR of 50.9 dB at low input frequencies, and maintains the SNDR higher than 48.2 dB up to the Nyquist frequency. The SFDR stays above 55 dB up to 2 GHz. The 3 dB effective resolution bandwidth is 1.5 GHz. The THD is limited mainly by the third harmonic, except at low frequencies where the second harmonic dominates due to a high phase and amplitude imbalance in the input balun. Different harmonic distortions (HD2 to HD5) are shown in Figure 6.13. The SNDR is limited by thermal noise at low input frequencies, as intended. At higher frequencies the jitter noise starts to dominate. The rms jitter value of 320 fs explains

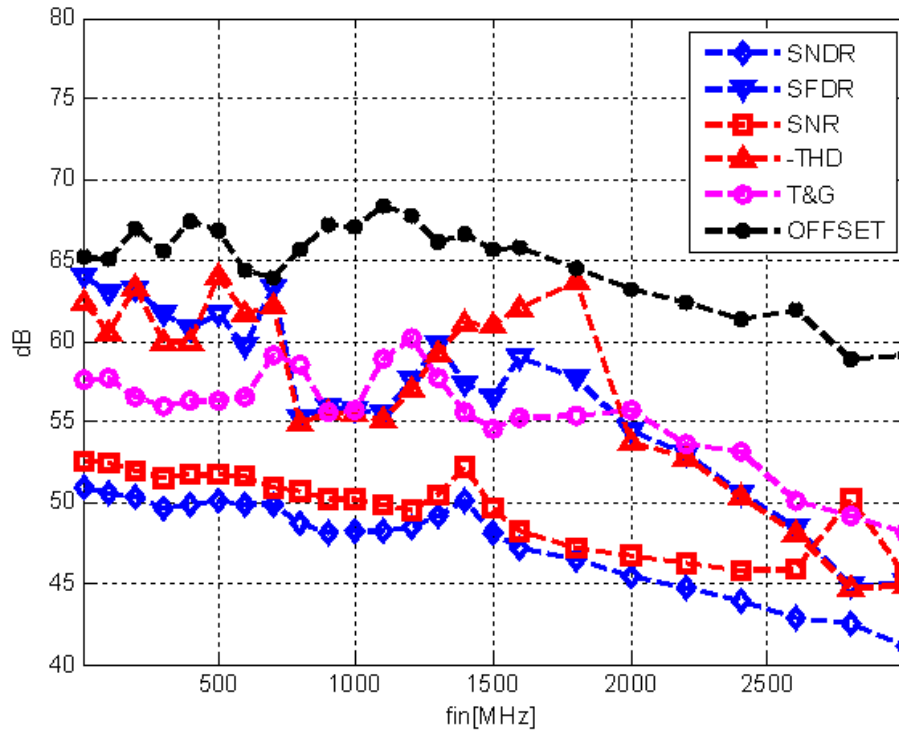


Figure 6.12: Performance plots vs. input frequency ($f_s = 2.8$ GHz, $V_{DD} = 1.2$ V).

the high frequency behavior at most frequencies, except around $kf_s/2$, where the rms jitter value is lowered to 110 fs. This can be explained by the layout of the ADC. Even- and odd-numbered ADC channels are placed on the separate sides of the chip, effectively creating two interleaved ADCs that sample at the frequency of $f_s/2$. When the input signal frequency is close to $kf_s/2$, the sampled input signal in both halves of the ADC is equivalent to a low-frequency signal. Signal-dependent switching in the SAR logic changes slowly from sample to sample, and only small number of output buffers, which drive the long wires connecting the ADC channels to the calibration logic, switches every cycle. This lowers the coupling of the digital gates to the clock buffers through substrate and common ground connections, which, in turn, lowers the jitter on the sampling clocks.

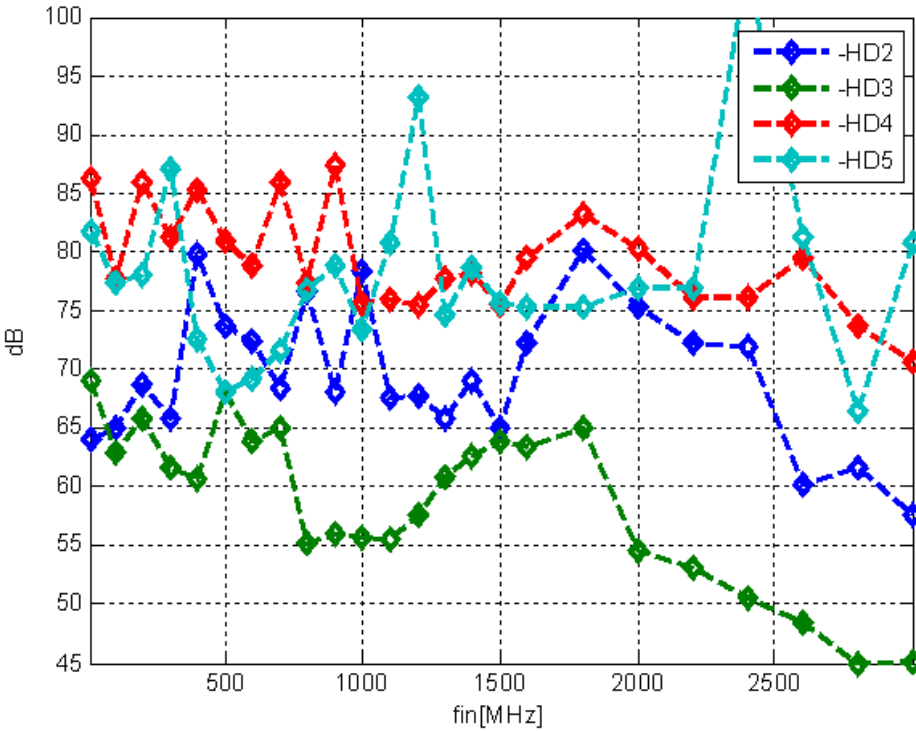


Figure 6.13: Harmonic distortion vs. input frequency.

The ADC has been tested with different sampling frequencies. The SNDR plots vs. the input frequency for the sampling frequencies of 1 GHz, 2GHz, 2.8GHz, and 3GHz are shown in Figure 6.14. The SNDR at low input signal frequencies is slightly higher for lower sampling frequencies, but it drops faster with increase of the input frequency. This is due to more charge leakage in the bootstrapping circuit of the top-plate switches.

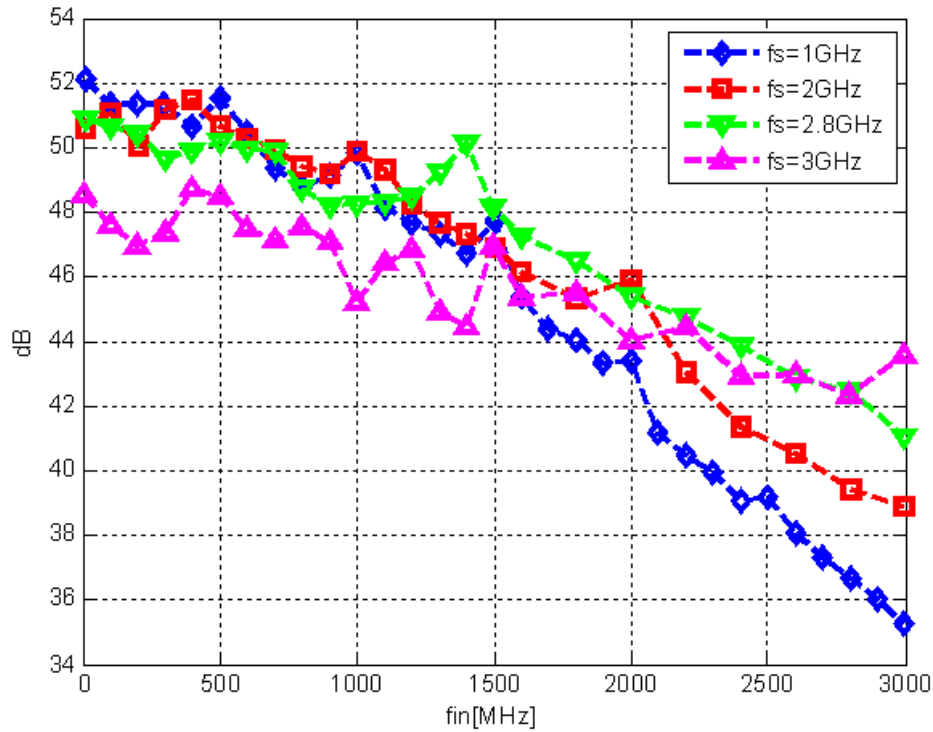


Figure 6.14: SNDR vs. input frequency for different sampling frequencies.

All performance plots for the supply voltage of 1.1 V are shown in Figure 6.15. The performance start to degrade at slightly lower sampling frequencies, so the measurements with $f_s = 2.7$ GHz are shown.

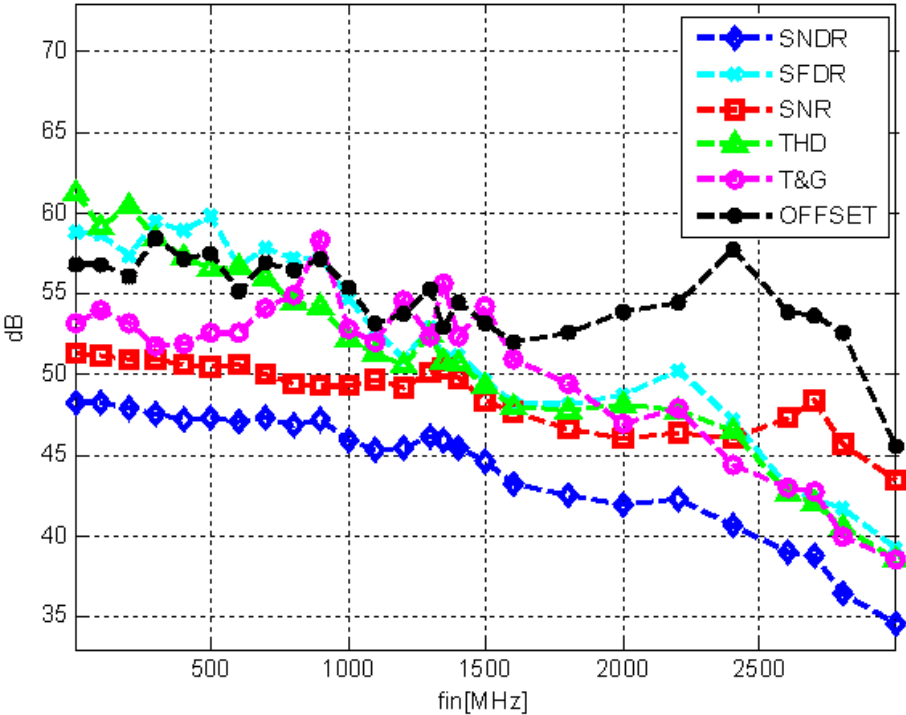


Figure 6.15: Performance plots vs. input frequency ($f_s = 2.7$ GHz, $V_{DD} = 1.1$ V).

The SNDR vs. the input signal amplitude is shown in Figure 6.16 for three different input frequencies. The SNDR levels off at approximately 3/4 of the full scale because the distortion from the top-plate switches starts to increase at that level. When some of the channels begin to saturate, the SNDR falls off sharply. A similar plot of SNR vs. the input signal amplitude is shown in Figure 6.17. The theoretical curves with combined thermal and quantization noise of 52.8 dB and jitter of 320 fs (110 fs for $f_{in} = 1.4$ GHz) are plotted in dotted lines. These curves prove that the high frequency noise indeed behaves as jitter.

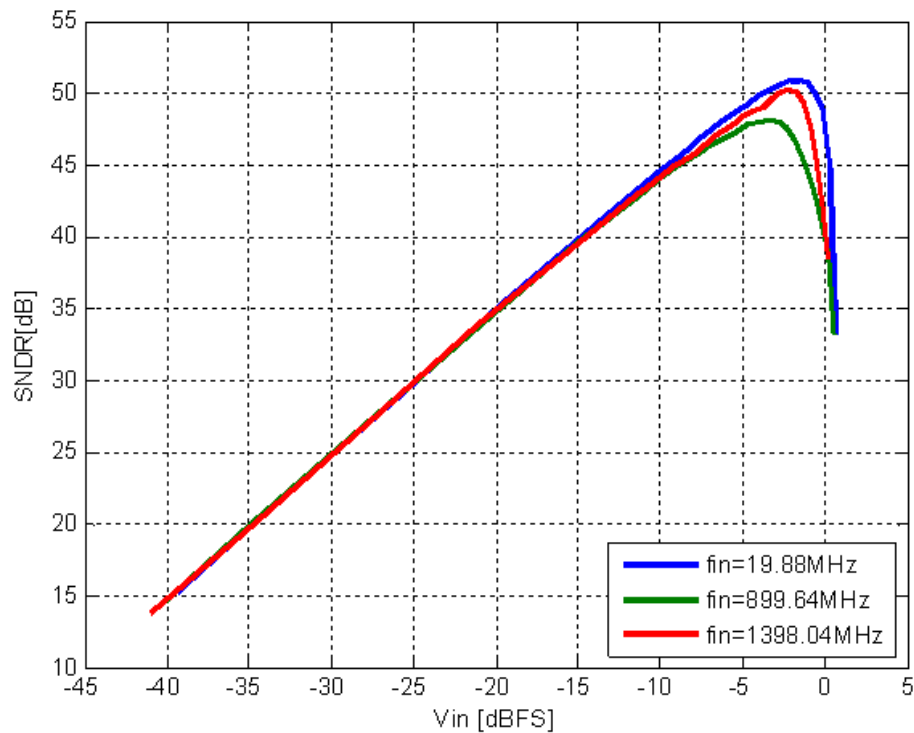


Figure 6.16: SNDR vs. input signal level.

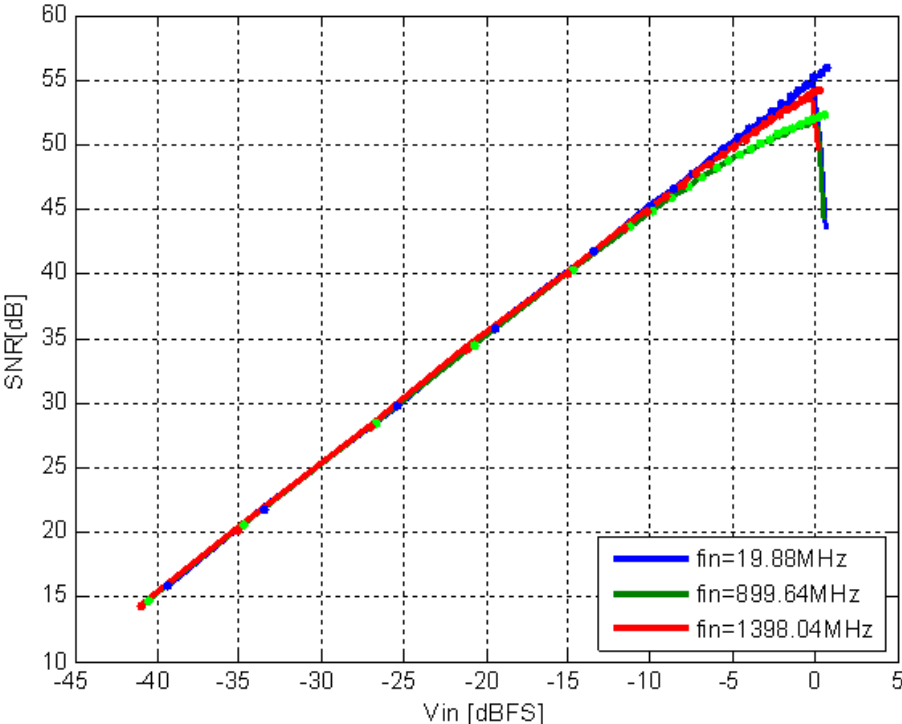


Figure 6.17: SNR vs. input signal level.

The final conversion results at the output of the digital correction circuit are formed as 14-bit words to minimize the accumulation of the quantization noise, and then the outputs can be rounded to the desired number of bits. All results shown so far are with the outputs rounded to 11 bits.. Lower number of bits can be desired in order to reduce the complexity and power of the digital circuits that follow. The SNDR performance with the output results rounded to 11, 10, 9, and 8 bits are shown in Figure 6.18. As it can be seen, the performance loss of less than 0.5 dB is observed when the output resolution is reduced from 11 to 10 bits.

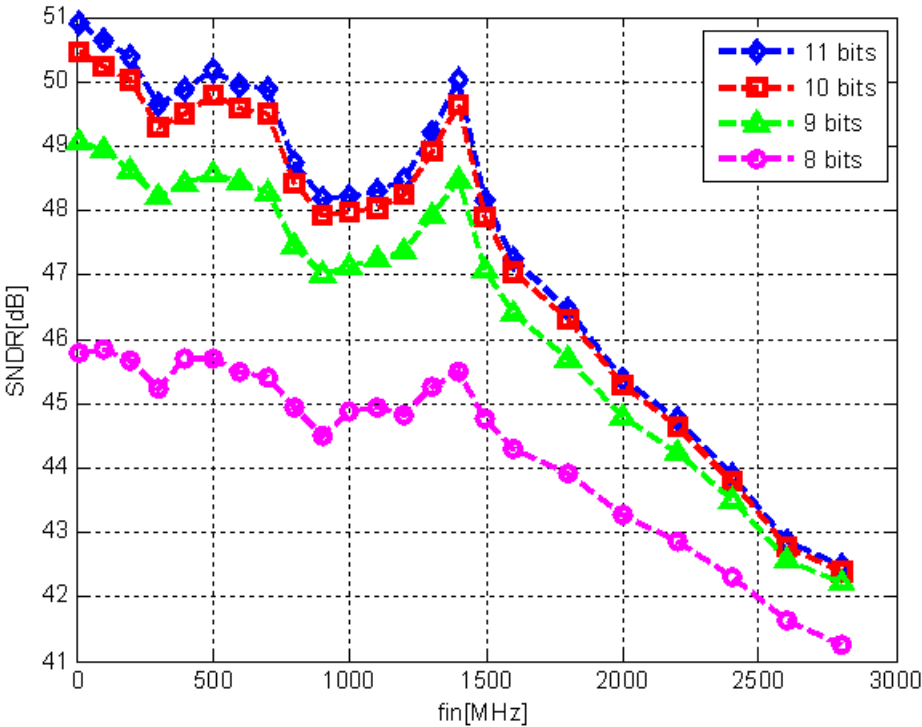


Figure 6.18: SNDR vs. frequency for different number of output bits.

6.6 Two-Tone Measurements

In realistic conditions, the ADCs almost never sample a single-tone input signal. A two-tone test is typically used to characterize the nonlinear effects called intermodulation distortions in ADCs. The intermodulation distortions are represented by undesired tones that appear in the output signal at the frequencies that are equal to the linear combination of the input frequencies. Particularly important are the second order intermodulation distortions that, for input signal consisting of two tones at the frequencies f_1 and f_2 , appear at the frequencies $f_1 - f_2$ and $f_1 + f_2$, and the third order intermodulation distortions, that appear at $2f_1 - f_2$, $2f_2 - f_1$, $2f_1 + f_2$, and $2f_2 + f_1$. An example of the output signal spectrum with two input tones at the input is shown in Figure 6.19.

Two types of two-tone tests have been performed. First, the central frequency f_c , defined as $f_c = \frac{f_1+f_2}{2}$ is swept, while the frequency gap $\Delta f = f_2 - f_1$ is kept constant. After that, the central frequency was kept constant, while the frequency gap was swept. The powers of the two tones at the input of the ADC was the same, and the phase was randomly selected. The amplitudes are adjusted so that the input peak-to-peak signal value is equal to the peak-to-

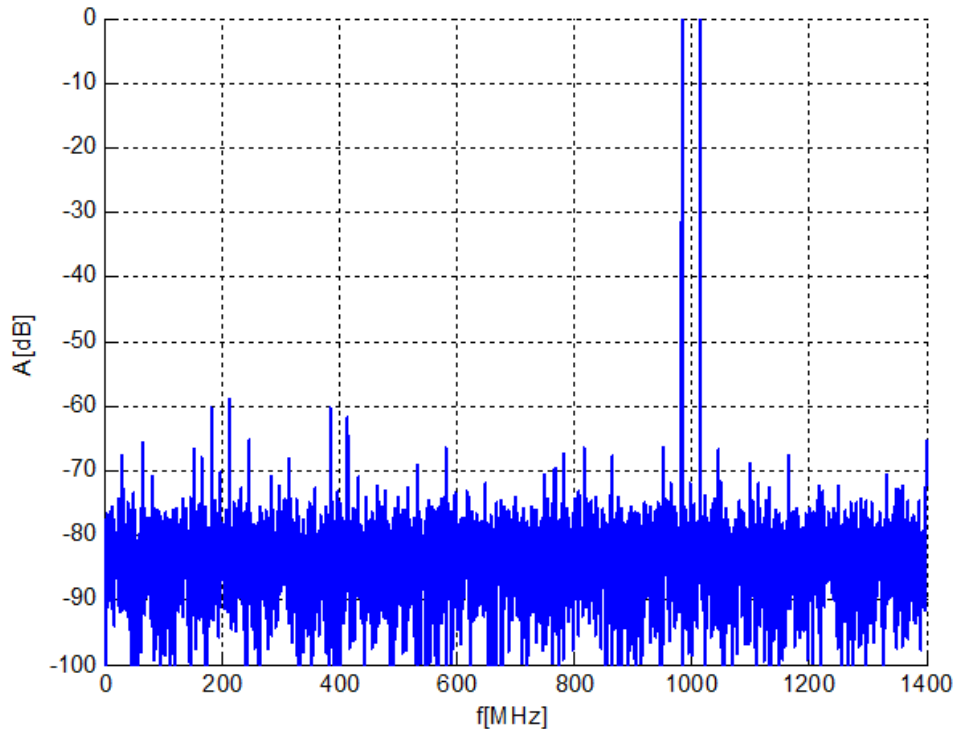


Figure 6.19: DFT with two input tones ($f_c = 999.88$ MHz, $\Delta f = 31.92$ MHz).

peak value of the input signal in the single-tone test. This is justified by the nonlinearities of the input analog front-end stemming mostly from the nonlinear bootstrap switches and nonlinear charge injection, both deteriorating with the increase of the input signal range.

The IM2 and IM3 dependences on the central frequency with $\Delta f = 5$ MHz are shown in Figure 6.20. The dependence on the frequency gap with the central frequency $f_c = 800$ MHz is shown in Figure 6.21.

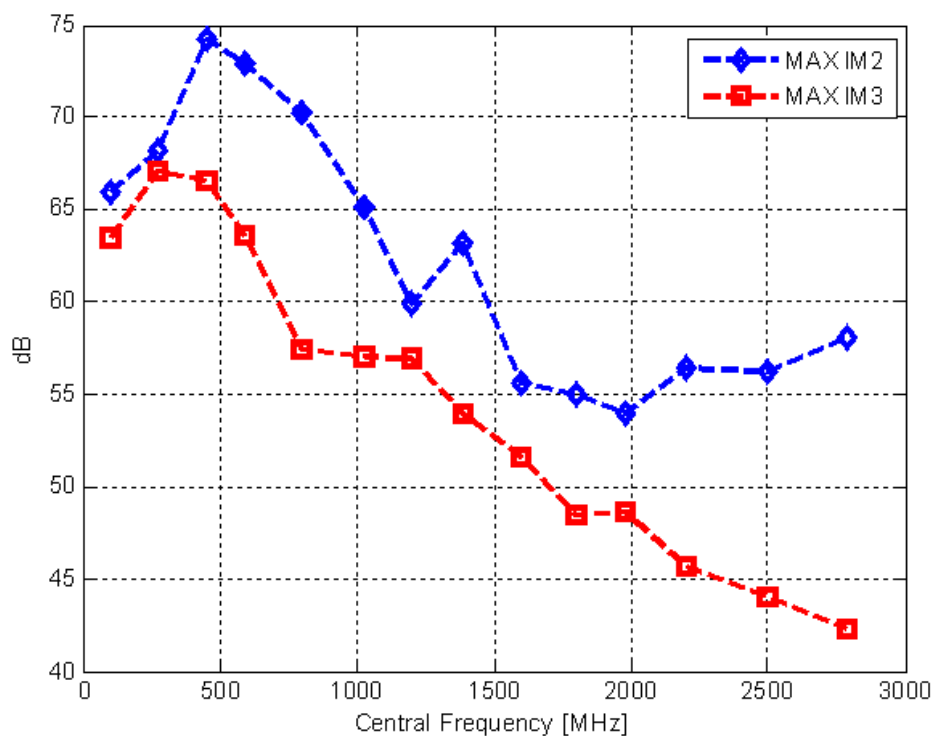


Figure 6.20: IM2 and IM3 vs. central frequency, $\Delta f = 5$ MHz.

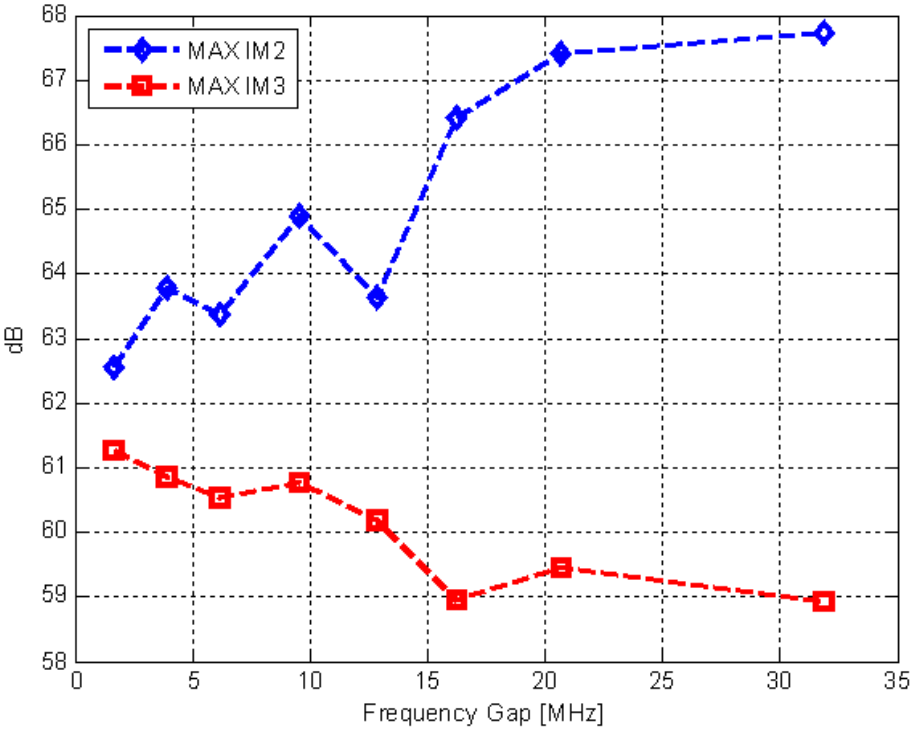


Figure 6.21: IM2 and IM3 vs. Δf , $f_c = 999.88$ MHz.

6.7 Performance Summary

With the sampling frequency $f_s = 2.8$ GHz this ADC achieves the SNDR of 50.9 dB at low input frequencies and retains the SNDR higher than 48.2 dB across the entire first Nyquist zone. The SFDR stays above 55 dB up to $f_{in} = 2$ GHz. The 3 dB effective resolution bandwidth is 1.5 GHz, and the 3 dB power bandwidth is higher than 3 GHz. The performance summary is shown in Table 6.1. The total power consumed by the ADC at $f_s = 2.8$ GHz and $V_{DD} = 1.2$ V with ongoing calibration is 44.6 mW. The power consumption breakdown is shown in Figure 6.22. Most of the power is consumed by the SAR logic (43%) and digital calibration logic (23%). The rest of the power is divided between clock generation and distribution circuits (19%), comparators and switches (12%), and the power drawn from the references (3%). The figure of merit (FoM), defined as

$$FoM = \frac{P}{2 * \min(f_s/2, ERBW) * 2^{ENOB}} \quad (6.1)$$

is 56 fJ/conversion-step calculated with low-frequency ENOB, or 78 fJ/conversion-step, if calculated with the minimum ENOB in the first Nyquist zone. The energy per conversion $E_c = P/f_s$ for this ADC is 16 pJ.

Table 6.1: ADC performance summary

Sampling rate	2.8 GS/s
Resolution	11b (10b with < 0.5 dB SNDR loss)
Peak SNDR	50.9 dB
SNDR	> 48 dB (up to 1.5 GHz)
SFDR	> 55 dB (up to 2 GHz)
ERBW	1.5 GHz
Input bandwidth	> 3 GHz
Input	1.8 V _{p-p,diff}
Chip area	1.7 mm ²
Analog core area	0.18 mm ²
Technology	ST 65 nm
Supply voltage	1.2 V
Power	44.6 mW

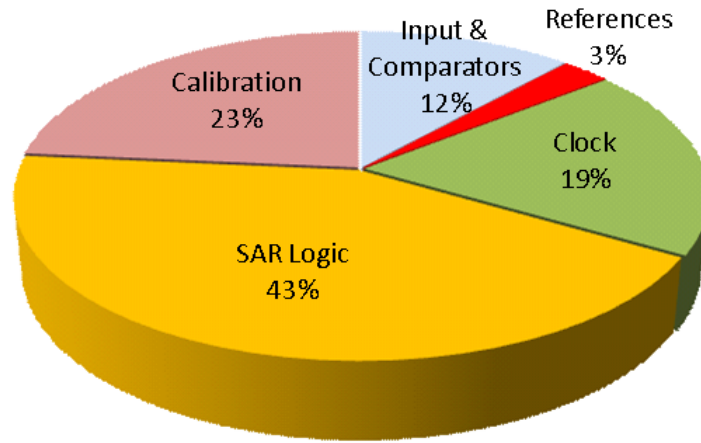


Figure 6.22: Power consumption breakdown at $f_s = 2.8$ GHz, $V_{DD} = 1.2$ V.

6.8 Comparison to Prior Art

A comparison of energy per conversion of this ADC to the prior-art ADCs with sampling frequency higher than 1 GHz published at the ISSCC and VLSI conferences from 1997 to 2012 is shown in Figure 6.23. As it can be seen, the ADCs from this group of designs with similar effective resolution have an order of magnitude higher energy per conversion, and the ones that achieve the same level of energy efficiency have at least 8 dB lower resolution. Another way to express the energy efficiency of an ADC is its standard figure of merit (6.1). The figure of merit plot from the introductory chapter of this dissertation is repeated here in Figure 6.24, but with the results from this work included. It includes the figure of merit of all ADCs with resolution between 6 and 10 bits and the sampling frequency between 10 MHz and 10 GHz published at ISSCC and VLSI conferences from 1997 to 2012. Again, an order of magnitude improvement in the figure of merit can be observed compared to the ADCs of similar speed in this group of designs. The figure of merit lower than 100 fJ/conversion-step was previously achieved only by the ADCs with sampling speed lower than 300 MS/s.

It may be argued that it is only fair to compare the ADCs with a similar level of performance since the design considerations differ greatly among different corners of the sampling frequency/resolution space. To address this concern, a more detailed comparison to the work presented in [9] is given. This ADC is designed in a 65 nm CMOS technology, uses a time-interleaved SAR architecture, samples at 2.6 GS/s and achieves the SNDR higher than 48.5 dB in the first Nyquist zone. First, the specifications in which this design outperforms our design are discussed. It has slightly better low-frequency performance (52.8 dB SNDR vs.

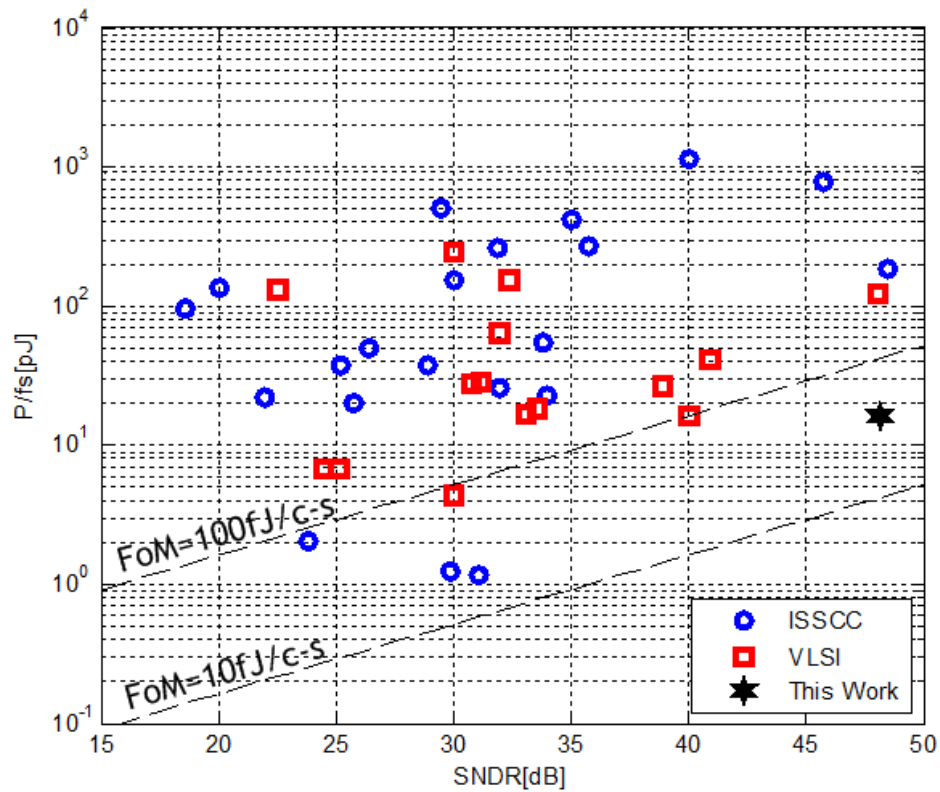


Figure 6.23: Energy per conversion for all ADCs with $f_s > 1$ GHz published at ISSCC and VLSI conferences from 1997 to 2012.

50.9 dB in our design), but it also has a faster roll-off due to the lack of timing calibration. Better jitter performance (< 110 fs vs. 320 fs) is achieved using current-mode logic (CML) buffers, which are powered from a higher-than-nominal supply of 1.3 V and consume a DC current. The THD performance of this ADC in the first Nyquist zone is about 3 dB better than in our design. One of the main techniques for achieving good linearity was the use of the feedforward-feedback interface, which requires input buffers that also consume a DC current and are powered from a 1.6 V supply. However, the main difference between this ADC and our work is the energy and area efficiency. The energy per conversion and the standard figure of merit of our design are approximately 11.6 and 10.9 times lower, respectively. The area of our design, including the analog core, the digital calibration logic and decoupling capacitors, is approximately 6.9 times lower than the one reported in [9] (it is not clear whether this number includes the area of decoupling capacitors). The large improvement in the power and area in our design comes mostly from the use of the minimum-size capacitors in the DAC, which leads to a compact design. This, in turn, enables easy distribution of the input

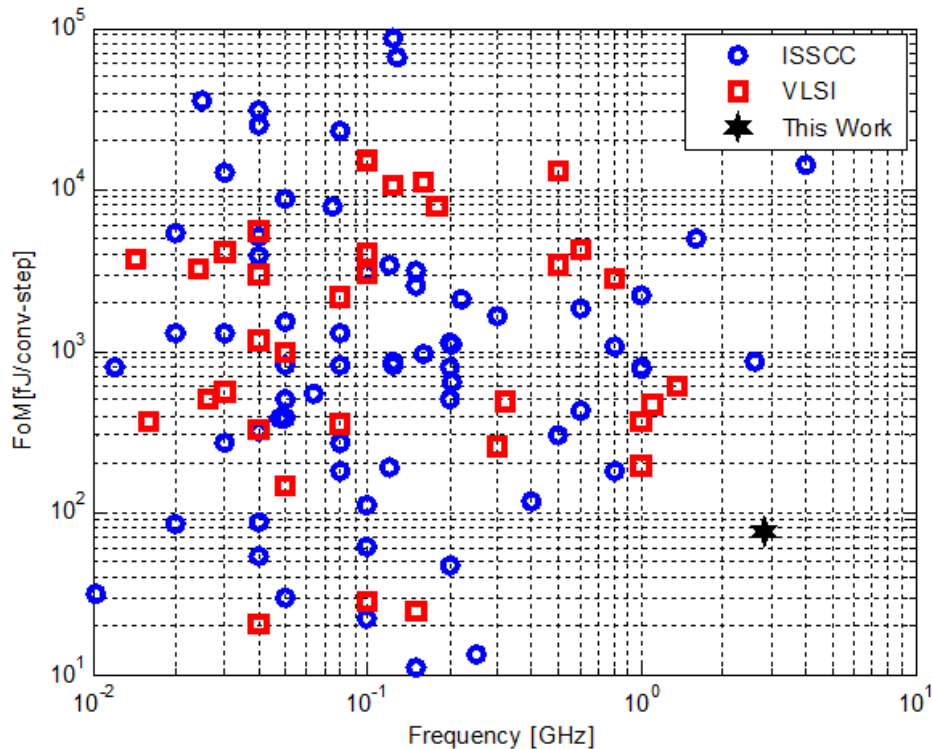


Figure 6.24: Figure of merit of all ADCs with resolution between 6 and 10 bits and sampling frequency between 10MHz and 10GHz published at ISSCC and VLSI conferences from 1997 to 2012, including this work.

signal without a need for power-hungry buffers, a simple low-power clock generation and distribution scheme, and the distribution of common reference voltages without reference buffers. The use of the minimum-size capacitors is supported by a low-overhead calibration of capacitor mismatches. The timing calibration helps maintaining a flat response over a wide range of input frequencies.

Since the calibration algorithms are one of the main contributions of this work, a brief comparison to other state-of-the-art background calibration algorithms is also given. The main feature of the calibration algorithms presented in this dissertation is their low overhead. The power consumed by the calibration circuits is around 23%. If the calibration coefficients are frozen, the power overhead is estimated to be between 10 and 15%. The area of the calibration circuits represents approximately 25% of the whole ADC area (including the decoupling capacitance). In the analog domain, the calibration requires two additional channels, one for linearity and the other for timing calibration. With 24 time-interleaved channels, this amounts to approximately 8% of the analog power and area. These two

channels can be turned off after the calibration algorithm converges.

First, the linearity calibration algorithms are compared. The algorithms presented in [24], [26], [41] and [30] all have high overhead, each in different way. Although the calibration circuits in [24] are of a similar complexity as in our design, the overhead is in terms of the design effort. This algorithm requires a linear reference ADC that needs to be designed separately from the time-interleaved channels. The algorithms in [26] and [41] require two conversions per sample, therefore creating an overhead in terms of the conversion time. The algorithm in [30] is formulated for algorithmic ADCs, but it could be extended to SAR ADCs as well. The algorithm is based on an iterative matrix inversion and it has a complexity of $o(K^2)$ (number of arithmetic operations is proportional to K^2), where K is the number of unknown calibration coefficients, compared to $o(K)$ complexity of our algorithm. Since the number of coefficients in a SAR ADC is much higher than in an algorithmic ADC, the overhead of this calibration would be much higher in SAR ADCs.

The calibration speed of the algorithm was not measured on the chip prototype due to a limited size of the memory buffer used for testing and limited control available in the digital on-chip hardware. From the behavioral simulations and the formulation of the algorithm, it can be concluded that it can achieve much faster convergence than previously published algorithms. The speed advantage compared to [24] comes from the speed of the reference channel, which is the same as the speed of other time-interleaved channels in our design, and much slower in [24]. The algorithm presented in [30], if applied to SAR ADCs, would update the calibration coefficients every K samples, whereas our algorithm performs the update every sample. The algorithm in [41] is inherently slow, as it requires long correlation sequences to achieve the desired accuracy.

The overhead of the timing calibration algorithm is even smaller than the overhead of the linearity calibration algorithm. This is a big advantage over the digital calibration algorithms presented in [18], [19], [17] and [12], as detailed in Chapter 3. To the best of the author's knowledge, no silicon implementation of these algorithms is available at the time of writing this dissertation. The algorithm presented in [11] has similar complexity as our algorithm, but it has very stringent requirements for the input signal during the calibration since it is based on aligning the zero-crossings of the input signal in the reference and time-interleaved channels. Our algorithm relaxes the input signal requirements, as explained in Chapter 3.

6.9 Design Limitations

The key limitation of this design stems from the limitations of the calibration algorithms, as described in Chapters 3 and 4. The input signal has to be "busy" and of certain amplitude and frequency content during calibration. This is not a problem in the systems with a foreground calibration or in many communication systems, where the calibration signal can be included in the preamble. This accounts for most commonly encountered practical applications. The requirements for the input signal make it hard to integrate this design

only into the systems that require uninterrupted operation and can have an arbitrary input signal.

The jitter was a limiting factor in achieving better SNR performance at higher frequency. It is believed that the jitter comes from the interference of the digital SAR logic and the output buffers with the clock buffers. More careful isolation would need to be applied to reduce jitter if this design should be used in subsampling applications or for further interleaving.

Chapter 7

Conclusion

This dissertation has explored the possibility of designing energy-efficient high-speed moderate-resolution ADCs by the means of massive parallelism and digital correction of analog impairments. The energy efficiency has been achieved by both optimizing the efficiency of the interleaved channels and by minimizing the overhead of interleaving. The efficiency in the individual channels has been achieved by using the minimum capacitor size in the capacitive DAC of only 50 aF. This miniaturization is also a key to minimizing the overhead of interleaving since small ADC channels ease the task of distribution of input, clock and reference signals. Additionally, the overhead is minimized by using simple background calibration techniques that correct the offset, gain and timing mismatches, as well as the capacitor mismatches in the interleaved channels. The concepts have been demonstrated on an ADC prototype that operates at 2.8 GHz sampling rate with effective resolution of 8 bits, while consuming less than 45 mW of power. Around two thirds of the power is consumed in the digital domain, which makes this approach even more suitable for implementation in finer technologies.

7.1 Key Accomplishments

The key accomplishments of this research are:

- Showed that it is possible to use capacitors as small as 50 aF to design radix-weighted capacitive DACs for use in SAR ADCs. This capacitor size is an order of magnitude smaller than previously reported and it enables a compact and energy-efficient design of the SAR ADCs.
- Analyzed error sources in single-channel and time-interleaved SAR ADCs and showed that, for a given level of bandwidth mismatch, the bandwidth requirements can be greatly relaxed in the systems that have timing and gain calibration.

- Developed a background calibration technique for correction of capacitor mismatches in SAR ADCs. The technique uses an additional reference channel identical to the time-interleaved channels and two modes of conversion in order to make the transfer characteristics of all channels equal and linear. The calibration is based on the LMS algorithm and it is executed in the background. The power overhead of the calibration is reasonably small compared to the overall power of the ADC, thus proving its practicality. The complexity and speed compare favorably to previously published background calibration techniques.
- Developed a background timing calibration technique for parallel ADCs. The technique uses an additional reference channel with intentionally mismatched bandwidth of the sampling network in order to estimate the derivative of the input signal. The derivative information is forwarded to the LMS algorithm that estimates the timing mismatches between the channels and tunes the edges of the sampling clocks using a mixed-signal feedback. The technique can be applied to any time-interleaved ADC architecture. The power overhead of this calibration is very small and it reduces the statistical requirements of the input signal compared to previously published timing calibration techniques.
- Improved the energy and area efficiency over the current state-of-the-art ADC design with comparable performance by an order of magnitude.

7.2 Future Work

We suggest several directions for further research and improvement of our work:

- The capacitors used in our design were implemented as parallel plate capacitors between two regular metal layers. Much smaller design, and consequently lower power, can be achieved if more compact capacitor structure, such as multi-layer finger capacitors, is used.
- Interleaving two or four current designs to further increase the sampling rate could yield a very energy-efficient ADC design in 10 GHz sampling frequency range.
- The calibration techniques developed in this work require a "busy" input signal in order to work properly. It would be desirable to define the term "busy" in a more mathematically strict way and to develop the algorithm that would detect the "busy" input signal condition. This would make the calibration algorithm truly background and completely transparent to the end user.
- In a design with even higher speed and resolution requirement the bandwidth mismatch can become a serious issue. In Chapter 2 we showed that timing and gain calibration

with a sinusoidal input signal can reduce the bandwidth requirements for a given bandwidth mismatch. Further reduction in the bandwidth requirements would be possible if the calibration was performed with a signal that has a wider spectral content. Finding the optimal input signal would be an interesting topic of research.

- Finally, the calibration can be extended to include bandwidth correction by fine-tuning the bandwidths of the input analog front-ends. This would be particularly beneficial if even higher speeds need to be achieved with more parallelism.

Bibliography

- [1] A. Abo and P. Gray, “A 1.5-V, 10-bit, 14.3-MS/s CMOS pipeline analog-to-digital converter,” *IEEE J. Solid-State Circuits*, vol. 34, no. 5, pp. 599–606, 1999.
- [2] Agilent Technologies, Inc. (2012). 8 hints for making better measurements using RF signal generators, [Online]. Available: <http://cp.literature.agilent.com/litweb/pdf/5988-5677EN.pdf>.
- [3] —, (2012). Agilent E4438C ESG vector signal generator, [Online]. Available: <http://cp.literature.agilent.com/litweb/pdf/5988-4039EN.pdf>.
- [4] E. Alpman, H. Lakdawala, L. Carley, and K. Soumyanath, “A 1.1V 50mW 2.5GS/s 7b time-interleaved C-2C SAR ADC in 45nm LP digital CMOS,” in *IEEE ISSCC Dig. Tech. Papers*, 2009, pp. 76–77.
- [5] W. Black and D. Hodges, “Time interleaved converter arrays,” *IEEE J. Solid-State Circuits*, vol. 15, no. 6, pp. 1022–1029, 1980.
- [6] Z. Boyacigiller, B. Weir, and P. Bradshaw, “An error-correcting 14b/20 μ s CMOS A/D converter,” in *IEEE ISSCC Dig. Tech. Papers*, 1981, pp. 62–63.
- [7] R. Burden and J. Faires, *Numerical analysis*, ser. Prindle, Weber & Schmidt Series in Mathematics. PWS-Kent Publishing Company, 1989, ISBN: 9780534915858.
- [8] S.-W. Chen and R. Brodersen, “A 6b 600MS/s 5.3mW asynchronous ADC in 0.13 μ m CMOS,” in *IEEE J. Solid-State Circuits*, 2006, pp. 2350–2359.
- [9] K. Doris, E. Janssen, C. Nani, A. Zanicopoulos, and G. van der Weide, “A 480 mW 2.6 GS/s 10b time-interleaved ADC with 48.5 dB SNDR up to Nyquist in 65 nm CMOS,” *IEEE J. Solid-State Circuits*, vol. 46, no. 12, pp. 2821–2833, 2011.
- [10] K. Dyer, D. Fu, S. Lewis, and P. Hurst, “An analog background calibration technique for time-interleaved analog-to-digital converters,” *IEEE J. Solid-State Circuits*, vol. 33, no. 12, pp. 1912–1919, 1998.
- [11] M. El-Chammas and B. Murmann, “A 12-GS/s 81-mW 5-bit time-interleaved flash ADC with background timing skew calibration,” *IEEE J. Solid-State Circuits*, vol. 46, no. 4, pp. 838–847, 2011.

- [12] J. Elbornsson, F. Gustafsson, and J.-E. Eklund, "Blind adaptive equalization of mismatch errors in a time-interleaved A/D converter system," *IEEE Trans. Circuits Syst. I: Regular Papers*, vol. 51, no. 1, pp. 151–158, 2004.
- [13] D. Fu, K. Dyer, S. Lewis, and P. Hurst, "A digital background calibration technique for time-interleaved analog-to-digital converters," *IEEE J. Solid-State Circuits*, vol. 33, no. 12, pp. 1904–1911, 1998.
- [14] Y. Greshishchev, J. Aguirre, M. Besson, R. Gibbins, C. Falt, P. Flemke, N. Ben-Hamida, D. Pollex, P. Schvan, and S.-C. Wang, "A 40GS/s 6b ADC in 65nm CMOS," in *IEEE ISSCC Dig. Tech. Papers*, 2010, pp. 390–391.
- [15] S. Gupta, M. Inerfield, and J. Wang, "A 1-GS/s 11-bit ADC with 55-dB SNDR, 250-mW power realized by a high bandwidth scalable time-interleaved architecture," *IEEE J. Solid-State Circuits*, vol. 41, no. 12, pp. 2650–2657, 2006.
- [16] C.-C. Hsu, F.-C. Huang, C.-Y. Shih, C.-C. Huang, Y.-H. Lin, C.-C. Lee, and B. Razavi, "An 11b 800MS/s time-interleaved ADC with digital background calibration," in *IEEE ISSCC Dig. Tech. Papers*, 2007, pp. 464–615.
- [17] E. Iroaga, B. Murmann, and L. Nathawad, "A background correction technique for timing errors in time-interleaved analog-to-digital converters," in *IEEE ISCAS Dig. Tech. Papers*, 2005, pp. 5557–5560.
- [18] S. Jamal, D. Fu, M. Singh, P. Hurst, and S. Lewis, "Calibration of sample-time error in a two-channel time-interleaved analog-to-digital converter," *IEEE Trans. Circuits Syst. I: Regular Papers*, vol. 51, no. 1, pp. 130–139, 2004.
- [19] H. Jin and E. Lee, "A digital-background calibration technique for minimizing timing-error effects in time-interleaved ADCs," *IEEE Trans. Circuits Syst. II: Analog and Digital Signal Processing*, vol. 47, no. 7, pp. 603–613, 2000.
- [20] T. Kobayashi, K. Nogami, T. Shirotori, and Y. Fujimoto, "A current-controlled latch sense amplifier and a static power-saving input buffer for low-power architecture," *IEEE J. Solid-State Circuits*, vol. 28, no. 4, pp. 523–527, 1993.
- [21] F. Kuttner, "A 1.2V 10b 20MSample/s non-binary successive approximation ADC in 0.13 μm CMOS," in *IEEE ISSCC Dig. Tech. Papers*, 2002, pp. 176–177.
- [22] H.-S. Lee, D. Hodges, and P. Gray, "A self-calibrating 15 bit CMOS A/D converter," *IEEE J. Solid-State Circuits*, vol. 19, no. 6, pp. 813–819, 1984.
- [23] H.-S. Lee and D. Hodges, "Self-calibration technique for A/D converters," *IEEE Trans. Circuits Syst.*, vol. 30, no. 3, pp. 188–190, 1983.
- [24] W. Liu, Y. Chang, S.-K. Hsien, B.-W. Chen, Y.-P. Lee, W.-T. Chen, T.-Y. Yang, G.-K. Ma, and Y. Chiu, "A 600MS/s 30mW 0.13 μm CMOS ADC array achieving over 60dB SFDR with adaptive digital equalization," in *IEEE ISSCC Dig. Tech. Papers*, 2009, pp. 82–83.

- [25] W. Liu and Y. Chiu, "An equalization-based adaptive digital background calibration technique for successive approximation analog-to-digital converters," in *Int. Conf. on ASIC*, 2007, pp. 289–292.
- [26] W. Liu, P. Huang, and Y. Chiu, "A 12b 22.5/45MS/s 3.0mW 0.059mm² CMOS SAR ADC achieving over 90dB SFDR," in *IEEE ISSCC Dig. Tech. Papers*, 2010, pp. 380–381.
- [27] S. Louwsma, A. van Tuijl, M. Vertregt, and B. Nauta, "A 1.35 GS/s, 10 b, 175 mW time-interleaved AD converter in 0.13 μm CMOS," *IEEE J. Solid-State Circuits*, vol. 43, no. 4, pp. 778–786, 2008.
- [28] Maxim Integrated. (2008). 8-Bit, 2.2Gsps ADC with track/hold amplifier and 1:4 demultiplexed LVDS outputs, [Online]. Available: www.maxim-ic.com/datasheet/index.mvp/id/5391.
- [29] J. McCreary and P. Gray, "All-MOS charge redistribution analog-to-digital conversion techniques I," *IEEE J. Solid-State Circuits*, vol. 10, no. 6, pp. 371–379, 1975.
- [30] J. McNeill, M. Coln, D. Brown, and B. Larivee, "Digital background-calibration algorithm for split ADC architecture," *IEEE Trans. Circuits Syst. I: Regular Papers*, vol. 56, no. 2, pp. 294–306, 2009.
- [31] B. Murmann. (2012). ADC performance survey 1997-2012, [Online]. Available: <http://www.stanford.edu/~murmann/adcsurvey.html>.
- [32] P. Nuzzo, F. De Bernardinis, P. Terreni, and G. Van der Plas, "Noise analysis of regenerative comparators for reconfigurable ADC architectures," *IEEE Trans. Circuits Syst. I: Regular Papers*, vol. 55, no. 6, pp. 1441–1454, 2008.
- [33] A. V. Oppenheim and R. W. Schaffer, *Digital signal processing*. Prentice-Hall, 1975.
- [34] K. Poulton, R. Neff, B. Setterberg, B. Wuppermann, T. Kopley, R. Jewett, J. Pernillo, C. Tan, and A. Montijo, "A 20 GS/s 8 b ADC with a 1 MB memory in 0.18 μm CMOS," in *IEEE ISSCC Dig. Tech. Papers*, 2003, pp. 318–319.
- [35] A. Shikata, R. Sekimoto, T. Kuroda, and H. Ishikuro, "A 0.5V 1.1MS/sec 6.3fJ/conversion-step SAR-ADC with tri-level comparator in 40nm CMOS," in *IEEE Symp. VLSI Circuits Dig. Tech. Papers*, 2011, pp. 262–263.
- [36] V. Sondur, V. Sondur, and A. Narasimha, "Design of digital differentiator to optimize relative error," *International Journal of Electrical and Electronics Engineering*, vol. 2, no. 4, pp. 240–245, 2008.
- [37] K. Tan, S. Kiriaki, M. de Wit, J. Fattaruso, F.-Y. Tsay, W. Matthews, and R. Hester, "A 5 V, 16 b 10 μs differential CMOS ADC," in *IEEE ISSCC Dig. Tech. Papers*, 1990, pp. 166–167.

- [38] Texas Instruments. (2011). ADC10D1000/1500 low power, 10-bit, dual 1.0/1.5 GSPS or single 2.0/3.0 GSPS ADC, [Online]. Available: <http://www.ti.com/product/adc10d1500>.
- [39] C. Vogel, "The impact of combined channel mismatch effects in time-interleaved ADCs," *IEEE Trans. on Instrumentation and Measurement*, vol. 54, no. 1, pp. 415–427, 2005.
- [40] K. L. J. Wong, "Comparison of digital offset compensation in comparators," Master's thesis, University of California, Los Angeles, 2002.
- [41] R. Xu, B. Liu, and J. Yuan, "Digitally calibrated 768-kS/s 10-b minimum-size SAR ADC array with dithering," *IEEE J. Solid-State Circuits*, vol. 47, no. 9, pp. 2129–2140, 2012.
- [42] M. Yoshioka, K. Ishikawa, T. Takayama, and S. Tsukamoto, "A 10b 50MS/s 820 μ W SAR ADC with on-chip digital calibration," in *IEEE ISSCC Dig. Tech. Papers*, 2010, pp. 384–385.