University of Seville
Department of Electronics and Electromagnetism
Microelectronics Institute of Seville (IMSE-CNM-CSIC)

Digital Background Self-Calibration Technique for Compensating Transition Offsets in Reference-less Flash ADCs

Thesis submitted by

Cristina Aledo González

to get the degree of Doctor by the University of Seville

PhD supervisor:

Eduardo José Peralías Macías

Seville, July 2019

Universidad de Sevilla
Departamento de Electrónica y Electromagnetismo
Instituto de Microelectrónica de Sevilla (IMSE-CNM-CSIC)

Técnica de auto-calibración digital en modo background para compensar offsets de transiciones en un convertidor A/D tipo flash sin referencias

Memoria presentada por:

Cristina Aledo González

para optar al título de Doctor por la Univerdad de Sevilla

Director:

Eduardo José Peralías Macías

Tutor:

Diego Vázquez García de la Vega

Sevilla, Julio de 2019

To my family

Acknowledgements

This research work has been developed at the Microelectronics Institute of Seville (IMSE-CNM-CSIC) and University of Seville. It has been funded by Ministry of Economy and Competitiveness, under the Research Staff Training (FPI) program and the following projects: DANTE (Spanish Government project TEC2011-28302, cofinanced by Ministry of Science and Innovation) and n-PATETIC (Spanish Government project TEC2015-68448-R, cofinanced by Ministry of Economy and Competitiveness and the European FEDER program). During 2015, a four-month-stay has been accomplished at the Department of Electrical Engineering at Faculdade de Ciências e Tecnologia (FCT), Universidade Nova de Lisboa (UNL), Portugal.

I would like to thank my supervisor Dr. Eduardo Peralías for sharing with me his knowledge, for being a great instructor and a supportive mentor, and for his patient and dedication along theses thesis work. Also, I would like express my deep gratitude to Dra. Adoración Rueda for giving me the opportunity to work in a great group and for her infinite support and dedication during these years. Their advice and helpful guidance were crucial to carried out this work until the end.

I would like to express my special thanks to Professor João Goes for his advice on my PhD work during the months I spent at FCT. I wanted to thank as well to all my colleagues and friends of IMSE, especially to Joaquín Ceballos, Antonio Ragel, José Miguel Mora, Luis Carranza, Luis Camuñas, Rafaella Fiorelli, Juan Núñez, Manolo Suarez, Manu Jiménez, Iñaki Vázquez, Delia Velasco and Alberto Muñío, for their support and help, both technically and personally.

Finally, I would like to thank to my friends and family for their unconditional love and comprehension. Special thanks to my mother, Mireya, Manu and Enrique, for being a constant source of support throughout these years. This work would not have been possible without them.

Abstract

This Dissertation focusses on proving that background calibration using adaptive algorithms are low-cost, stable and effective methods for obtaining high accuracy in flash A/D converters. An integrated reference-less 3-bit flash ADC circuit has been successfully designed and taped out in UMC 180 nm CMOS technology in order to prove the efficiency of our proposed background calibration. References for ADC transitions have been virtually implemented built-in in the comparators dynamic-latch topology by a controlled mismatch added to each comparator input front-end. An external very simple DAC block (calibration bank) allows control the quantity of mismatch added in each comparator front-end and, therefore, compensate the offset of its effective transition with respect to the nominal value. In order to assist to the estimation of the offset of the prototype comparators, an auxiliary A/D converter with higher resolution and lower conversion speed than the flash ADC is used: a 6-bit capacitive-DAC SAR type. Special care in synchronization of analogue sampling instant in both ADCs has been taken into account.

In this thesis, a criterion to identify the optimum parameters of the flash ADC design with adaptive background calibration has been set. With this criterion, the best choice for dynamic latch architecture, calibration bank resolution and flash ADC resolution are selected.

The performance of the calibration algorithm have been tested, providing great programmability to the digital processor that implements the algorithm, allowing to choose the algorithm limits, accuracy and quantization errors in the arithmetic. Further, systematic controlled offset can be forced in the comparators of the flash ADC in order to have a more exhaustive test of calibration.

Index

1 INTRODUCTION

In this chapter an introduction of this work is given. Motivations for developing this thesis work are exposed in Section 1.1. Goals and applied proposals are summarized in Section 1.2. Finally, thesis organization is described in Section 1.3.

1.1 Motivation

Analog-to-digital converters (ADCs) are one of the main building blocks in current electronic systems, especially in communication, signal processing and biomedical fields. The evolution of ADCs has been changing with the requested specifications of the devices technology.

Among the different types of ADCs, flash converter is the simplest and fastest architecture, achieving a great bandwidth (tens of GHz) [1]. Flash converters have often low resolution (currently hardly achieve 6 bits) since as resolution increases, the number of comparators greatly rises, so does the power consumption [2]. Indeed, non-idealities in ADCs appear as higher speed is demanded.

Offset caused by element mismatch errors is an important issue in flash ADCs. These errors affect greatly to the flash performance, decreasing its linearity. For this reason, using calibration techniques to reduce the offset is crucial in this kind of converters. Thanks to calibration, errors in the ADC are corrected and, thus, it is possible to relax the architecture complexity design. This way, accuracy and speed increase and both power consumption and area decrease due to the simpler architecture.

These offset errors can be corrected either in the analogue or in the digital domain. Many analogue methods have been proposed, such as error storage or trimming of elements [3]-[18] or traditional auto-zeroing techniques [19]-[21]. There are also methods, such as averaging [6]-[10] [22]-[23] which can executed in the analogue as well as digital domain. These techniques are employed to cancel for the offset in comparators, but also they can be used in preamplifiers or even in the whole comparator.

Due to the additional hardware that requires analogue calibration and the quick evolution of digital technologies, the use of digital techniques to correct the resulting errors in ADCs is quite often employed. Digital methods based on adaptive processing algorithms greatly reduce the cost of the circuit without attenuating its performance, even for rather complex digital algorithms [24]. Most of these digital methods focus on reducing gain, offset and capacitor mismatch errors [25]-[35] or correcting the coder error [36]-[37]. Several digital methods could be used in the calibration of the offset in latch-based comparators, being DAC-based calibration [32] [38]-[40] and redundancy [41]-[55] of great importance.

In DAC-based calibration, the DAC creates an imbalance in comparator front-end, which is opposite to the actual mismatch, eliminating it. This forced imbalance could be created in several ways, for instance, by varying the current in the branches [47] [56]-[58] or by adding capacitance to them [59]-[61]. Capacitive imbalance could be created by a capacitive DAC [62]-[63], typically compose of a binary weighted capacitor array. The capacitive DAC acts as a SH and, hence, no SH circuit is required in the whole circuit [64].

Moreover, with capacitive imbalance using MOS calibration DAC [59]-[61][64]-[68], a voltage difference is applied to the latch inputs by modifying the DAC control bits. This way, a variation in the offset is done. Once the optimum bits are selected, the offset in the comparator is cancelled. This popular technique has been also employed in the IC demonstrator of this work.

Offset calibration could be accomplished either in foreground or background mode. The first method requires a dedicated period to perform the calibration and, hence, the normal operation of the converter is stopped to fulfill the error measurement and/or its correction [24]. Many foreground calibration techniques are found in the literature. Some of them are aimed to correct the linearity error of the subDAC [30] [69]-[71], others

intend to correct the amplifiers gains or comparators offsets [27] [30] [72]-[75], and many of them are employed in offset calibration in flash topologies [32] [39]-[40] [76].

In background calibration methods, measurement and correction tasks are accomplished simultaneously and continuously without ceasing the normal operation of the ADC. Unlike in foreground techniques, no break in the input signal path is needed. All parameters are estimated and corrected in a concurrent way, in contrast to foreground techniques where the calculation and correction of the parameters are done in a sequential manner [77]. Background techniques are much more effective than foreground architectures, however, area and power consumption are increased. Many different approaches in background calibration are considered in different ADCs architectures. Some are based on *skip and fill* [26] [78], others focus on the reduction of interstage gain errors in pipeline ADCs [79]-[81], and there are some techniques that use a noise input signal for calibration [37] [82]-[85]. Correlation-based background calibration techniques [25] [28] [84] [86]-[90] widely used, since they are faster than the previous methods. Some background calibration architectures require a reference ADC, synchronised with the main converter [9] [91]-[93]. This auxiliary ADC is employed to measure the parameters to be corrected and thus it must have low speed and high resolution to perform an efficient conversion of the analogue input. The estimated outputs of both ADCs are expected to be equal and errors can be measured from the difference between the outputs. Background calibration methods based on digital adaptive algorithms [25]-[26] [72] [81] [83] [94]-[104] exhibit quite reduced cost in the estimation of the errors and also show low power consumption.

1.2 Thesis proposals

Due to the great interest and popularity of background calibration techniques and because of the necessity of proving the efficiency and reliability of some of these methods in current circuits, this thesis has been focused on designing and proving an valuable background technique implemented in a flash ADC in CMOS technology. Since the research group in which I am working has been developed some works related with adaptive algorithms[103] [105]-[106], the first focus of this thesis was studying background calibration in depth, using the previously mentioned works as a reference for developing this dissertation. Therefore, one of the main goals of this work is to prove that background calibration using the adaptive algorithm proposed in [105] is a low-

cost, stable and effective method for obtaining high accuracy in flash and pipeline A/D converters. The algorithm in [105] is used to calibrate the comparator offsets in subADCs of pipeline ADCs and it performs two different tasks: first, the measurement of the offset in each comparator, and second, the generation of calibration code to move the transition of the comparator to its ideal value, thus cancelling its offset. Both jobs are fulfilled in the digital domain and simultaneously.

In this work, this calibration technique is employed for calibrating the comparators' offset in flash ADCs. An integrated circuit has been successfully design and taped out in UMC 180 nm CMOS technology in order to prove the efficiency of the proposed background calibration. To do this, a low-resolution reference-less flash ADC with simple structure and reduced area have been considered. Instead of using a resistive ladder, transitions are implemented built-in in the comparators, by a systemmatic mismatch added to each comparator input front-end.

The design of the comparator and its calibration circuit in a flash ADC is explained step-by-step, going deeply into the dynamic latch design and the different aspects of the comparator structure. In this work, a criterion to identify the optimum parameters of the flash design compatible with the calibration algorithm has been set. With this criterion, the best choice for dynamic latch architecture, calibration bank resolution and flash resolution are selected.

The algorithm in [105] is based on the use of an auxiliary converter with higher resolution (at least 2 more bits) than the converter under calibration to help calibrate its offset. In [105] the auxiliary ADC is the own ADC pipeline queue, but in general, if an auxiliary ADC is employed, it must be a low cost one and have reduced power consumption. It is placed in parallel to the ADC under calibration to avoid interrupting the normal working of the flash. In this work, the chosen auxiliary ADC is a SAR type with binary weighted capacitive DAC, synchronous clock and sequential SAR logic.

SAR output code and flash ADC output codes are compared in a digital processor that implements the adaptive algorithm. The processor output acts on the input comparator front-end, forcing a mismatch that corrects their offsets. In our design, a fully programmable algorithm more versatile than that proposed in [105] has been employed. This way, calibration could be improved by choosing the optimum parameters for the algorithm limits, accuracy, resolution or quantization errors. Further, it is possible to

inject 16 different offset values in the range of $\pm 2\ LSB$ in every comparator of flash ADC. This way, the limits of the ADC and calibration performance could be tested.

Synchronisation between both ADCs in the circuit is one of the most crucial issues in this work, since no dedicated sample and hold circuit is employed. When flash and SAR ADCs are working concurrently, both ADCs must capture the same input signal. In our design, a specific digital block has been designed to ensure the correct synchronisation between both ADCs clocks.

Using the presented calibration technique improves not only flash accuracy but also its speed, by relaxing the architecture complexity design. Also, both power consumption and area are lessened due to the simpler architecture. Moreover, calibration helps designing a more robust flash ADC, with a safety margin to assure that if there are PVT variations that affect the transitions, calibration process will cover enough range to correct the shifted transitions, bringing them back to their ideal locations. All of the above described have been validated experimentally.

1.3 Thesis Organization

Contents of this work are organized as follows:

- Chapter 1: An introduction of this work is given. Motivations and applied proposals are summarized.
- Chapter 2: A general outlook of ADCs is introduced, summarizing some of the most important properties of ADC performance. Also, the ADCs state-of-the-art is reviewed and some of the most relevant ADC architectures for this work are deeply addressed.
- Chapter 3: A sketch of calibration is presented. Foreground and background calibration methods as well as analogue and digital techniques are discussed. In addition, a summary of offset calibration in comparators is given and most important digital techniques for offset calibration are examined. Further, the applied solution to correct the offset in a flash ADC is addressed.
- Chapter 4: The design of a comparator and its calibration circuit in a flash ADC is explained step-by-step, going deeply into the dynamic latch design and the different aspects of the comparator structure. Offset measurement and calibration

DAC design are also exposed.

- Chapter 5: The design of the integrated demonstrator is proposed. A general description of chip architecture is given, analyzing in detail each of its blocks. Moreover, circuit operation is elucidated and designs of both ADCs and calibration block are detailed. Special attention is given to synchronisation between flash and SAR ADCs.
- Chapter 6: Experimental results of prototype are gathered. Additionally, chip architecture and PCB design are elucidated. Test setup and communication interface are both drawn. Finally, results for test chip are presented and analyzed.
- Chapter 7: Conclusions of the thesis work are collected.

2. ANALOGUE-TO-DIGITAL CONVERTERS OVERVIEW

Analog-to-digital converters (ADCs) are one of the main building blocks in current electronic systems. They perform the link between the analogue world and digital processes [24]. Nowadays, there is a great deal of different ADCs which cover a wide range of telecommunication applications. ADC performance is determined by parameters like resolution, conversion speed, consumption, input signal range, price,... [107].

In the transformation of the signal from analogue to digital domain, sampling and quantization are required. These two processes must be done properly to avoid losing signal information, which modifies the spectrum. Quantization as well as thermal and jitter errors are the most important problems in ADCs. Also process inaccuracies have influence over the appearance of errors in ADCs.

Due to the widely use of ADCs in modern systems, improving their speed and precision is crucial. A greater resolution will increase the precision at a cost of a greater circuit complexity [107]. There is a variety of ADCs architectures with different properties that make them more or less convenient for a particular application. To select the proper topology, their limitations must be studied. The most employed architectures in high-speed applications, i.e., with sampling rates greater than 5MSPS [108] are flash, interpolative, folding, pipeline and successive approximation [19].

In this chapter some of the most important specifications of the ADC performance and main errors in high-speed ADCs are summarized. Moreover, the state-of-the-art is discussed and some of the most relevant ADC architectures for this work are deeply explained.

2.1. ADCs Basics

Ideally, an N-bit ADC converts the continuous-time analogue input voltage x into a discrete N-bit digital output signal b with a maximum total number of codes of 2^N. Every analogue input value must be approached to one digital output level. This can be achieved by generating reference voltages (also called transition levels t_j) in the ADC and comparing the input voltage to these references, selecting the reference voltage which is closest to the input. The analogue input voltage takes infinite values from x_{min} to x_{max}, while the digital output can just have certain discrete values from 0 to 2^N-1. The digital signal can be represented in different codifications: binary, decimal, sign-magnitude, two's complement, Gray or one's complement. Binary code is the most used. Gray and thermometer code advantage is that only one bit changes from code to code.

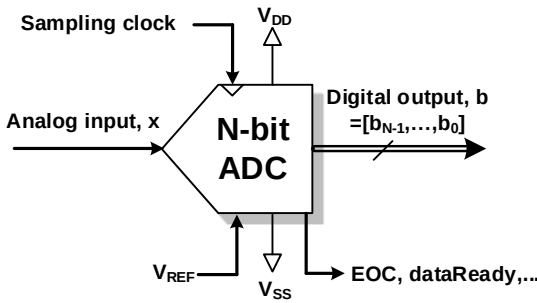

Figure 2.1. Basic N-bit ADC with external reference

An N-bit ADC symbol and its ideal input/output characteristic (transfer characteristic) are shown in Figure 2.1 and Figure 2.2. Analogue input signal x takes values in the range of $[-R, R]$, being the full range analogue range $FS = x_{max} - x_{min} = 2R$ [2]. The ADC translates the analogue value of x into a digital N-bit output $b = ADC(x)$, with $b \in [0, M]$. M is the total number of transition levels ($M = 2^N - 1$), corresponding the first

level to $t_1 = -R + q$ and the last one to $t_M = +R - q$, where quantization step q is the analogue value of the LBS and represents the minimum resolution in the ADC:

$$q = LSB = \frac{FS}{2^N} \tag{2–1}$$

where 2^N is the total number of quantization levels. Value t_k represents the analogue transition level at k-th digital code, that is, the analogue value where the output signal changes between code (k-1) and code k. The most significant bit (MSB) in digital output is b_{N-1} and least-significant bit (LSB) is b_0. Note that when analogue input takes large values, ADC output saturates [2].

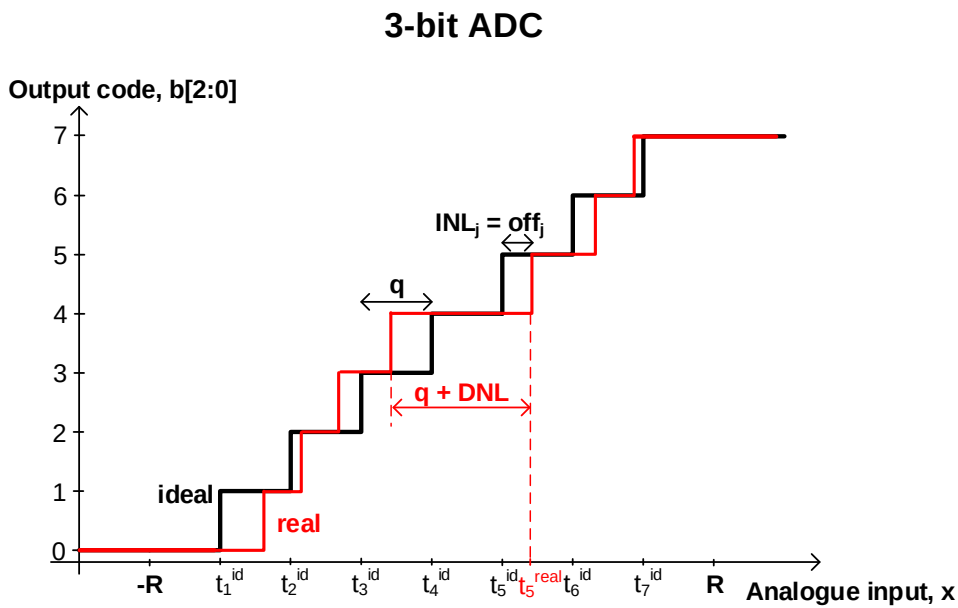

Figure 2.2. Ideal and real input/output characteristic for a 3-bit ADC

2.1.1. Static and dynamic specifications

The quantization error (ε_q) (see Figure 2.3) is the unavoidable error which appears due to discretization process. This rounding error introduces a quantization noise in the output signal, even in ideal ADCs [19]. As resolution increases, quantization error reduces and it can be considered a white noise. Assuming that into each code, quantization error has the same probability to be inside the range $\left[-\frac{q}{2}, \frac{q}{2}\right]$ [2], the

quantization noise error power is

$$P_{noise} = \frac{1}{q}\int_{-q/2}^{q/2} \varepsilon_q^2 d\varepsilon_q = \frac{q^2}{12} \qquad (2\text{–}2)$$

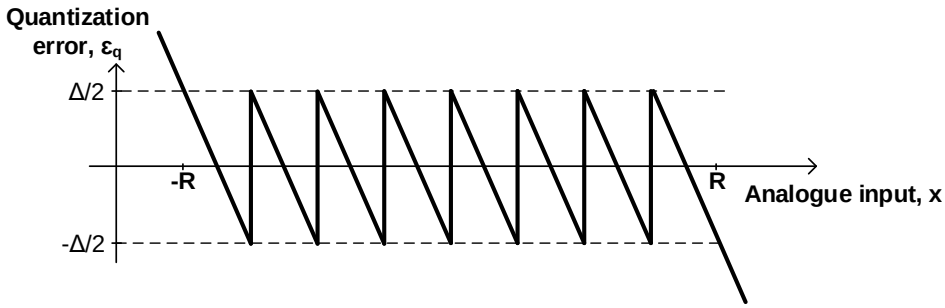

Figure 2.3. ADC ideal quantization error

Considering a sine input signal $x = R \cdot \sin(\omega_{in} t)$ with an amplitude of $R = \frac{FS}{2}$ and $\omega_{\text{in}} = 2\pi f_{in}$, the signal total power is:

$$P_{signal,max} = \frac{1}{T}\int_0^T \frac{FS^2}{4} \cdot \sin^2(2\pi f_{in} t) dt = \frac{FS^2}{8} = \frac{(q \cdot 2^N)^2}{8} \qquad (2\text{–}3)$$

being $T = \frac{1}{f_{in}}$. Hence, the effect of the quantization noise is valued by the signal-to-noise ratio (SNR) as the ratio between the maximum signal power and the ideal quantization error power:

$$SNR|_{dB} = \frac{P_{signal,max}}{P_{noise}} = (6.02 \cdot N + 1.76) dB \qquad (2\text{–}4)$$

This equation gives information about performance of the ADC depending on the number of bits, N. If ADC resolution increases in 1 bit, the SNR improves in 6.02dB while noise power is divided by 4. All the above calculations are valid for uniform distributed quantization levels. Usually, the amplitude of the input signal is not uniformly distributed around the full-scale range, being the quantization error different for each code.

Quantization noise is not the only limitation in ADCs. Other important errors in ADCs must be considered to correctly evaluate the performance of the ADC [2]. In the following, some of these relevant errors are discussed. To measure the performance in ADCs, some well-known specifications are considered. The most common static parameters measured in ADCs are offset and gain errors, differential non-linearity (DNL) and integral non-linearity (INL) (depicted in Figure 2.4 and Figure 2.5). Since elements in the ADC are not ideal, mismatch between components occurs and, hence, the transfer function varies. Any deviation from the ideal input/output characteristic will caused the static errors above mentioned. For instance, in an ideal ADC the difference between two consecutive transitions (t_j and t_{j+1}) is a constant:

$$q_j = t_{j+1} - t_j = q \tag{2–5}$$

In practice, the already discussed second order effects introduce deviations that modify the transition' location and offset appears. Offset of two consecutive transitions, $off_j = INL_j$ and $off_{j+1} = INL_{j+1}$, will modify the corresponding quantum width q_j this way:

$$q_j = \left(t_{j+1}^{id} - off_{j+1}\right) - \left(t_j^{id} - off_j\right) = q + DNL_j \tag{2–6}$$

Because of its relevance for this work, offset errors are profoundly studied in the following sections.

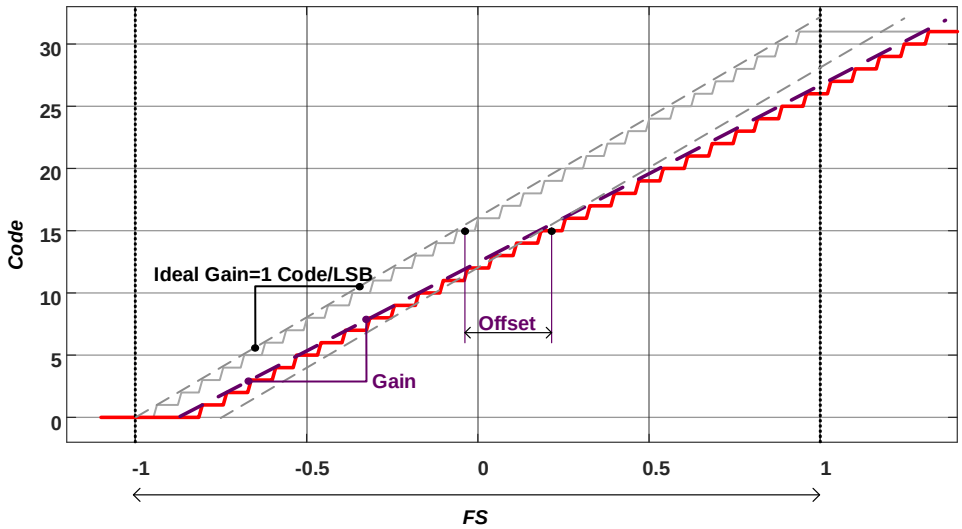

Figure 2.4. Example of offset and gain errors

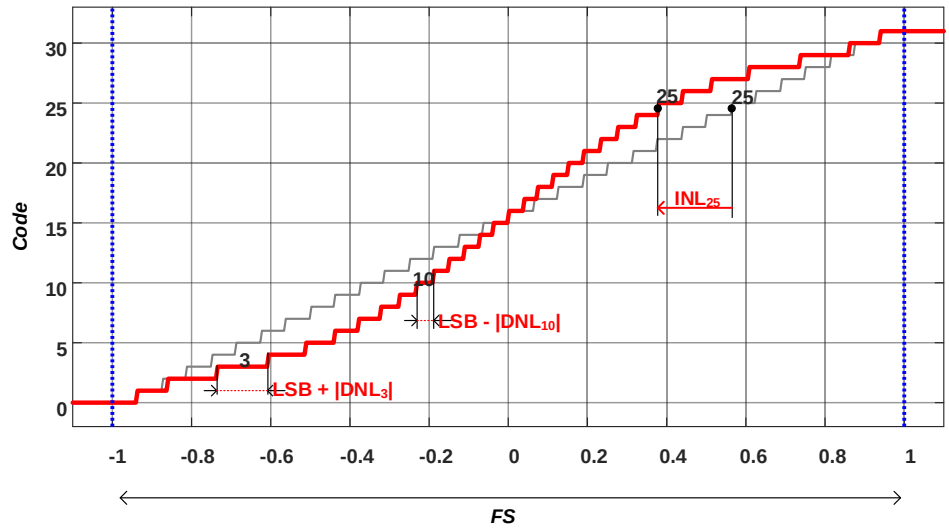

Figure 2.5. Example of INL and DNL errors

Considering the dynamic performance of A/D converters, there are several ways of characterizing the accuracy and noise in ADCs. Some of the most popular specifications are signal-to-noise ratio (SNR), signal-to-noise-distortion ratio (SNDR or SINAD),

effective number of bits (ENOB), total harmonic distortion (THD) and spurious-free dynamic range (SFDR). The SNR is related to the ENOB through equation (2-4), changing N by $ENOB$:

$$ENOB = \frac{SINAD|_{dB} - 1.76}{6.02} \quad (2\text{–}7)$$

Dynamic errors such as settling errors, non-linear slew, clock feedthrough or glitches [2] are due to the quick change of the input signal, especially as input frequency and amplitude increase. Based on an FFT analysis, these specifications will quantify the distortion and noise in the ADC and, hence, measuring them is essential to evaluate the ADC performance. Notice that since ADC samples the input signal, other sources of error related to sampling must be evaluated as they have a great impact in the SNR of the output signal. These kinds of errors are addresses in the following section.

2.2. Errors in ADCs

In real sampling ADCs, during the conversion of the analogue input into a digital level, a great number of noise errors could appear. Some of the dominant limits of high speed ADC are jitter and thermal noises. Also, comparator metastability is another source of error to be taken into consideration.

2.2.1. Sampling-time jitter

In ADCs the signal must be sampled. This sampling could be done by using a sample-and-hold circuit at the input of the ADC or without it. However, in the second case the performance of the ADC at high signal frequency could be rather poor since dynamic performance is restricted by the precision in the sampling instants [2].

Sampling is influenced by the uncertainty in the clock [24], which generates clock jitter at the sampling instants. Switch imperfections and delays generated by logic blocks contribute also to the alteration of the sampling instants. Jitter affects to dynamic performance of simple-and-hold in ADCs and, hence, to the value of the sampled signal. This variation in the sampling instants produced by clock jitter originates that sampling at time t is actually the sampling time $(t + \Delta t)$ [2]. For the sinusoidal input signal $x(t) =$

$R \cdot \sin(\omega_{in} t)$, the error $\Delta x(t)$ produced by clock jitter is

$$\Delta x(t) = R \cdot \omega_{in} \cdot \cos(\omega_{in} t) \cdot \Delta t \tag{2–8}$$

and its greatest effect is achieved when the slope of the input signal reaches its maximum, that is, when $t = 0$:

$$\Delta x = R \cdot \omega_{in} \cdot \Delta t \tag{2–9}$$

Sampling error increases for high input frequencies, since a small Δt will lead in a great error Δx. However, sampling error is not dependent on the sampling clock frequency.

The power of the jitter error is

$$P_{jitter} = \langle \Delta x(t)^2 \rangle = \langle (R \cdot \omega_{in} \cdot \cos(\omega_{in} t))^2 \rangle = \frac{1}{2} \cdot R^2 \cdot {\omega_{in}}^2 \cdot \langle \Delta t^2 \rangle \tag{2–10}$$

The power of the input sine signal is

$$P_{signal} = \langle x(t)^2 \rangle = \frac{1}{2} \cdot R^2 \tag{2–11}$$

and the SNR is given by

$$SNR = \frac{P_{signal}}{P_{jitter}} = \frac{1}{{\omega_{in}}^2 \cdot \langle \Delta t^2 \rangle} \tag{2–12}$$

which can be expressed in dB as

$$SNR|_{dB} = -20 \cdot \log({\omega_{in}}^2 \cdot \langle \Delta t^2 \rangle) \tag{2–13}$$

Due to sampling jitter, the SNR is degraded. For instance, achieving a SNR of 66 dB at input frequency $f_{in} = 100MHz$ requires $\langle \Delta t^2 \rangle = 80ps$ clock jitter [24].

2.2.2. kT/C noise

Being one of the main limitations in SH circuits, thermal noise is an unavoidable error that appears in sampling switch due to the thermal excitation of charge carriers [109]. In

a typical SH circuit during sampling, switch is turned off and the capacitor holds the input signal plus the thermal noise.

The operation of a sampling circuit can be modelled by a simple RC circuit [24] and the noise power stored on the capacitor C in the band-base when switch turns off is

$$P_{noise,C} = \frac{kT}{C} \qquad (2\text{–}14)$$

This noise will limit the signal performance, leading to a reduction of SNR. In order to increase the SNR, the sampling capacitance has to be greater and, hence, power consumption rises. If the capacitance increases by k, the noise voltage decreases by $\sqrt{k}$ [24]. In low resolution ADCs, thermal noise is a minor issue. However, in high resolutions, a capacitor of tens pF is necessary to cope with the thermal noise [2].

2.2.3. Comparator metastability

Metastability is a fundamental problem in a comparator output and it is associated to comparison. This source of error deteriorates the performance of ADCs at high sampling frequencies. Metastability appears when input signal is small and the comparator cannot make a decision in the assigned time, being stacked in an undefined situation where the output signal is not in a valid logic level ('1' or '0'). Usually comparators are composed of a preamplifier and a regenerative latch. The use of a preamplifier will reduce the chance of having metastability [2] [24], since the probability of having this kind of error is inversely proportional to circuit gain [2]. Metastability errors are of great importance in flash ADCs, where the number of comparators increases exponentially with the number of bits, and mitigating metastability errors in flash is crucial. This problem will be discussed in Section 4.1.

2.3. State-of-the-Art

ADCs evolution has been changing with the requested specifications of the devices technology. Technology scaling in digital applications has been beneficial for ADC performance. CMOs technology is the most convenient for cost-efficient implementation of high-performance data converters, filters and radio frequency transceivers [77]. ADCs are demanding in communication, signal processing and biomedical fields. Their design is nowadays a challenge since they require great conversion efficiency, low power consumption and high bandwidth. Several architectures for ADCs have been proposed, being each of them more convenient to a certain application, depending on its performance parameters. Factors like resolution, sampling frequency, power consumption or area are considered when comparing the performance of different ADCs.

Figure 2.6 illustrates the ENOB versus the conversion rate, f_s. It contains all publications reported at the ISSCC and VLSI Symposium from 1997 to 2016, gather by Murmann in [110].

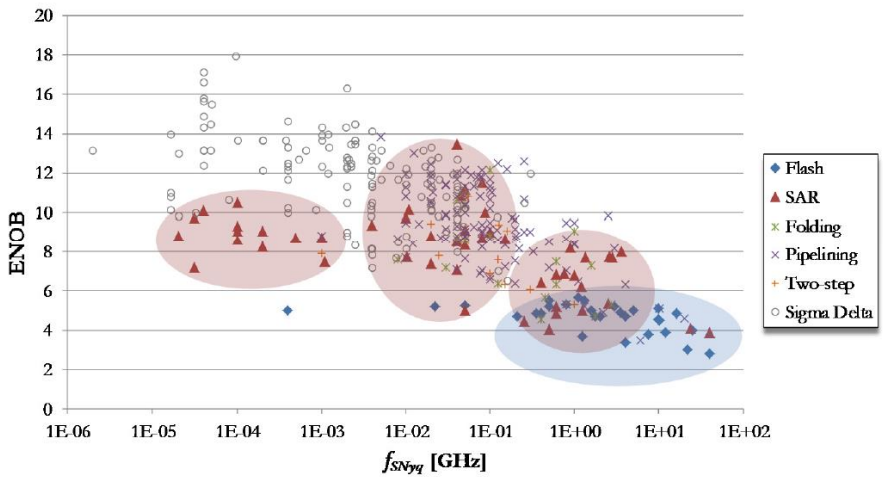

Figure 2.17 – ENOB versus sample frequency for published ADCs.

Figure 2.6. ENOB versus conversion rate (f_s) [111]

Flash converter is the fastest architecture [1] and achieves a great bandwidth (tens of GHz). It has limited resolution (up to 8 bits) because as its resolution increases, the

number of comparators greatly increases, so does the power consumption. The main limitation is mismatch errors, which reduce the flash linearity. A conventional 35GS/s, 4-bit flash ADC (ENOB of 3.7 bits) without calibration is presented in [112] for technology 180nm SiGe BiCMOS. Work [113] presents an excellent power efficient 20GS/s 6-bit time-interleaved ADC with 8 flash-type subADCs, each of them operating at 2.5GS/s with 6 bits resolution. Time-interleaved is implemented in a 32 nm CMOS SOI process and achieves a 5.49 bits of ENOB. Folding, interpolating and averaging schemes are popular among flash ADCs, as well as calibration techniques, particularly calibration methods based on DAC. Several offset correction architectures [40] [114] has been proposed achieving a great efficiency in flash. Work [115] proposed a 5-bit folding flash (4.5 ENOB) in 90nm digital CMOS with a DAC for digital calibration that greatly reduces the power consumption and a rather small trade-off between power consumption, sampling frequency and ENOB.

Pipeline ADC is employed in applications that need medium to high resolution (10-16 bits) with great bandwidth. The major drawback is the non-idealities of each stage, creating distortion. In [116] a 4GS/s 13b pipelined ADC with SNDR of 56dB implemented in 16 nm CMOS is presented. Quite challenging specifications are achieved when using nonlinearity calibration, such as in [117].

SAR ADC is used in medium resolution (8-12 bits) applications that employ medium to large bandwidths (<100MHz). This is the most interesting topology in terms of power efficiency and has a reduced size when compare with flash and pipeline ADCs.

Sigma-delta converter is employed in very high resolution (16 bits or greater) applications that require small to medium bandwidth. It provides a high SNR, however DAC nonlinearities limits the performance of this type of ADC. Work [118] proposes a 12-bit ENOB wide bandwidth sigma-delta ADC operating between 20 and 40 MS/s in 130nm CMOS technology.

Figure 2.7 [111] illustrates the Walden Figure-of-Merit (FoM) as a function of the conversion rate, f_s, with publications in [110]. Walden FoM is defined as the power consumption divided by the conversion rate and 2^{ENOB}. Works after 2010 are pointed out in red colour and time-interleaved designs are indicated by solid fill mark.

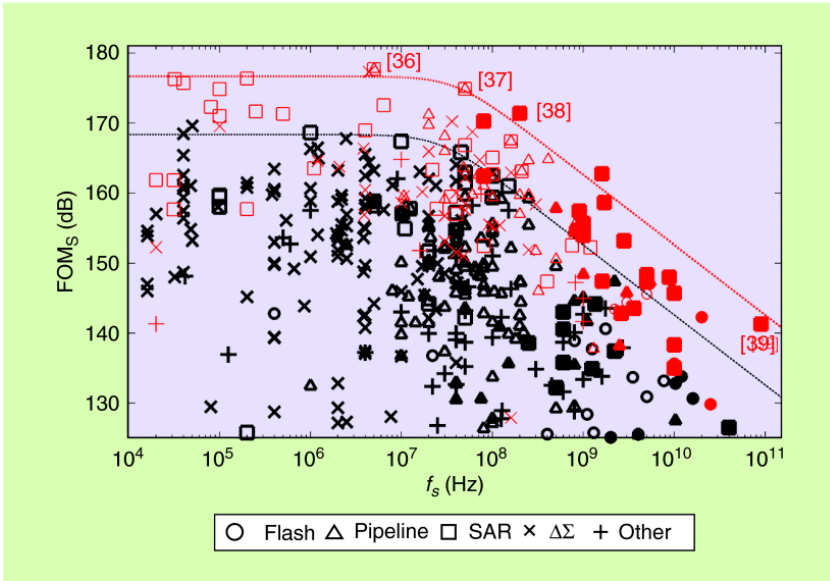

Figure 2.7. Walden FoM versus conversión rate (f_s) [111]

Nowadays, some of the ADCs performances are overlapped by different architectures. Although flash has been the fastest ADC, in recent years SAR topology has achieved ultrahigh speed, being this topology rather demanding. Such a high speed has been reached by using time-interleaving [111] [119]. For instance, a 2.6 GS/s 10b SAR interleaved ADC implemented in 65nm CMOS and with SNR up to 49dB is presented in [120] and in a recent paper [121] a 90GS/s 8-bit interleaved SAR in 32nm digital SOI CMOS is proposed, with a SNDR. Above 36dB SNDR up to 6.1GHz. Some other improved ideas include combination of SAR and pipelining as in [122] or pipelined flash-SAR ADC [123], where better resolutions than flash are achieved at high speed. The prototype in [122] is a 12-bit 50MS/s pipeline ADC that includes a SAR subADC in its first stage. The proposed circuit has been implemented in 65nm and 90nm CMOS, achieving an ENOB of 10.6 bits and Walden figure-of-merit of 52fJ/conversion-step. Paper [123] shows a 14-bit 200MHz power-efficient pipelined flash SAR ADC with no dedicated sample-and-hold circuit in the first stage, which has been designed using a 65nm CMOS technology.

2.4. Relevant ADC architectures for this work

To select the most convenient ADC for each application, basic parameters and limitations of each ADC topology must be well investigated. In the following sub-sections a study of the most relevant ADC architectures for this work is carried out.

2.4.1. Flash ADC

Flash ADC (see Figure 2.8) is the simplest and fastest topology, since it is formed of a series of comparators in parallel that allows achieving a high conversion speed [19]. There is one comparator for each reference level (transition t_j in the figure). Each comparator senses the input signal to its corresponding reference level [2], giving a '1' if the input signal is larger than the reference voltage, otherwise the comparator output is '0'. Therefore, the output pattern corresponds to a thermometric representation. An N-bit flash consists of $M = 2^N - 1$ comparators and an encoder as depicted in Figure. The $2^N - 1$ thermometric output is not so useful and, hence, an encoder is employed to transform the thermometric code into a more convenient codification, such an N-bit binary output [24] [124].

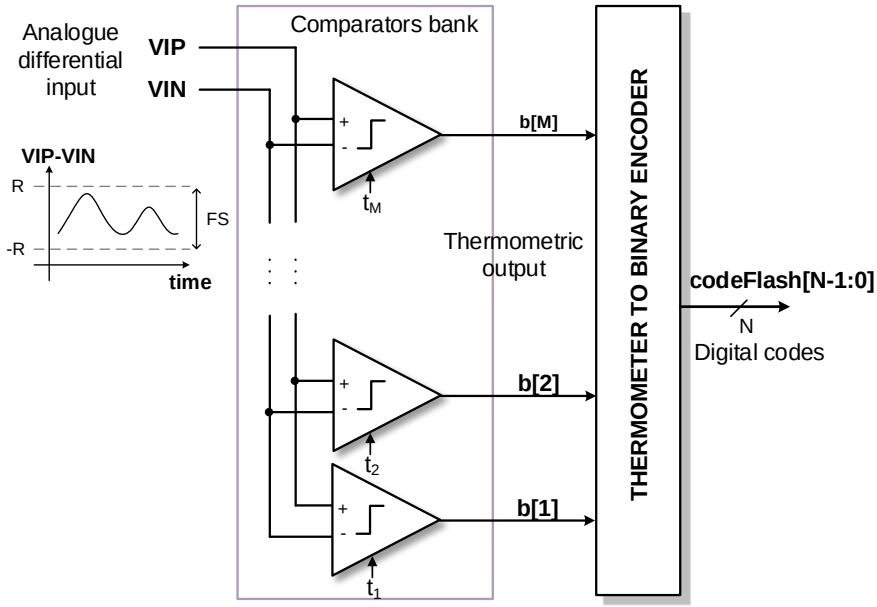

Figure 2.8. An N-bit flash ADC

As input signal is applied to all comparators at the same time, flash output is only delayed by one comparator delay and by the small delay introduced by the encoder gates [124]. However, the number of comparators increases exponentially with the number of bits, leading to large power consumption and area (and therefore cost) for resolutions above 8 bits [2]. The use of traditional techniques such as interpolation and folding [115] [125]-[128] has increased the resolution in flash while reducing the number of preamplifiers (interpolation) or comparators (folding) and also decreasing the power consumption [124]. Even for flash resolutions of 4-6 bits, to achieve good power efficiency, small transistors are used, requiring hence offset calibration in flash [38].

In addition to this limitation in resolution, another drawback due to the great number of comparators in this architecture is kickback noise at the analogue input. The effects of kickback noise from latch to the inputs when latch goes from reset mode to decision mode creates a great amount of noise at the comparators' inputs, producing an error that is proportional to the square of the number of comparators [19]. Using preamplifiers attenuates kickback noise as well as comparator offsets. Also, the capacitance at input node increases with the number of comparators. Moreover, metastable errors (already discussed in Section 2.2.3) due to the lack of SH can create sparkle errors (unwanted

zeros) in the thermometer code which can be mitigated by using additional circuitry such as a fault tolerant encoder [124]. Sparkle and offset problems will be discussed in Chapter 4.

2.4.2. Pipeline ADC

A pipeline ADC employs several stages, each of them resolving consecutively a part of the whole ADC resolution. A block diagram of a pipeline ADC with k stages is illustrated in Figure 2.9. A front-end SH circuit is followed by k stages, being STG_1 the most significant stage and STG_k the least significant one. Each stage is form of a low resolution ADC (subADC), which converts the analogue input y_j into a digital code c_j, and a multiplying DAC (MDAC) to substract the subADC output from the input, obtaining a quantization error that is later amplified to obtain the residue (y_{j+1}). The residue amplification allows using the full-scale range in next stage input. In just one single clock period, the first stage samples the input and gives the MSB, while the second stage generates the residue of the previous sampled input. The first stage generates N_1 bits, the second stage N_2 bits, and so on. The pipeline total resolution is the addition of the bits in each stage ($N = N_1 + N_2 + \cdots + N_k$) [24] and is generated by logic circuitry. Due to this pipeline operation in which each sample must propagate through the complete ADC structure, the output is generated with a latency time which is directly proportional to the number of stages.

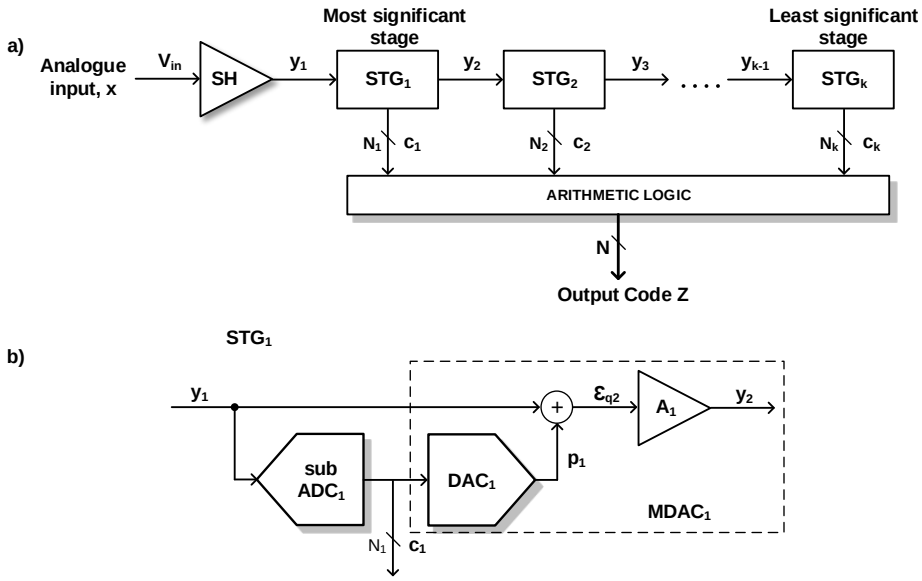

Figure 2.9. a) Pipeline ADC architecture with k stages, b) Architecture of stage 1

The accuracy and, hence, design of first stages is more demanding than that of last stage. The DAC accuracy is relaxed after few stages, since input referred residue in stage k decreases by 2^k [24]. Also, the amplified residue could have greater or lower amplitude than the full-scale range due to non-idealities in ADCs, DACs and amplifiers. This will be a problem if the next stage ADC is not able to convert the residue outside the full-scale range. These ADC accuracy errors could be overcome by employing techniques such as reducing the amplifiers' gain or adding redundancy levels in ADCs [24], and then compensating them with digital correction techniques.

2.4.3. SAR ADC

Successive Approximation Register (SAR) ADC has become very attractive due to its outstanding power efficiency at moderate conversion speed. SAR ADC is based on successive approximations algorithm and consists of a comparator, a control logic register and a DAC (see in Figure 2.10).

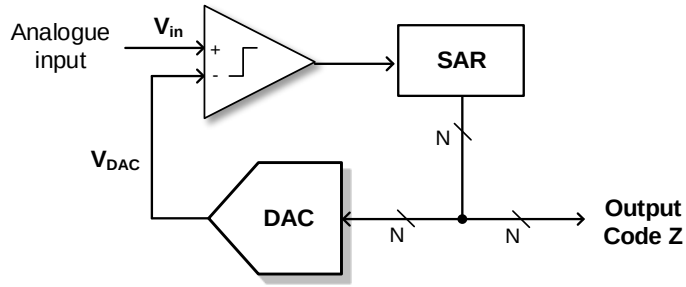

Figure 2.10. Typical SAR ADC circuit

Conversion starts by forcing all the bits in the register to '0', except for the most-significant bit (MSB) which is set to '1'. The DAC converts its digital input into an analogue value V_{DAC} and the comparator compares it to the analogue input V_{in}. If analogue input is greater than the DAC output, the MSB of the register is maintained at '1', otherwise it is set to '0'. The next most-significant bit is then set to '1' and the following are set to '0'. The process is repeated for all bits in DAC. Then conversion is finished and the register output corresponds to the analogue input. Figure 2.11 shows an example of DAC output in a 4-bit SAR ADC. Notice that in first step the comparator compares V_{in} with $V_{DAC} = \frac{V_{ref}}{2}$, being V_{ref} the SAR reference voltage and the dynamic full-scale range $FS = [0, V_{ref}]$. In second step, the comparison of V_{in} to $V_{DAC} = \frac{V_{ref}}{4}$ or to $V_{DAC} = \frac{3V_{ref}}{4}$ is carried out, and so on. As conversion progresses, DAC output is approached to a stable value, which will correspond to the measured analogue input. The number of iterations is determined by the number of bits. In the example, a 4-bit SAR employs four iterations to complete the conversion. Hence, the speed of SAR converters is reduced in high resolutions [2].

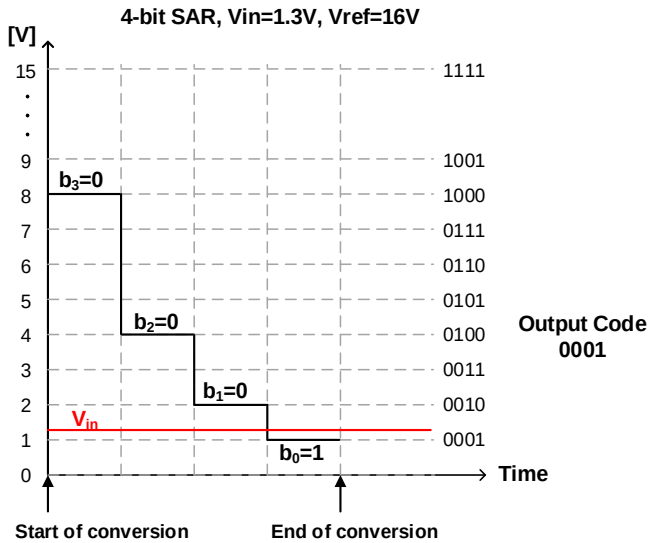

Figure 2.11. Example of a DAC operation in a 4-bit SAR ADC

SAR architecture has become very popular in low-power medium-resolution, moderate-speed applications. In this kind of converters, DAC determines the conversion speed of the converter and also consumes most of the overall power dissipation of the SAR [129]. Moreover, the DAC determines the accuracy and linearity. Charge redistribution configurations in switched capacitor DAC are widely used since their accuracy and linearity are determined by the accuracy of the capacitor plate area, capacitance and mismatch emerged from fabrication processes [124]. The power consumption of the circuit is determined only by the comparator and the dynamic charging and discharging of the capacitor array [24]. In [129], the power consumed by switching the capacitors of the DAC and its linearity behaviour in three different topologies is compared: conventional binary-weighted capacitive array [130]-[132], binary-weighted capacitive array with attenuation capacitor [133]-[134] and split binary-weighted capacitive array DAC [135]-[137].

The serial topology of the SAR ADC limits its conversion speed. At high speeds, its power consumption increases. There are several methods for increasing speed in SAR ADCs. One of the most popular strategies is is the using time-interleaving [120]. In [138] authors use a nonbinary 2-bit/cycle structure, which increases the speed and robustness while relaxing the DAC settling requirements. Asynchronous dynamic logic [139]-[141]

has been also employed to reduce the conversion time, decreasing the complexity of the digital logic and, hence, the power consumption. More improvements have been included in [142], where a fast 1-bit/cycle SAR ADC containing only one comparator and asynchronous clock with domino logic is proposed.

3. OFFSET COMPARATOR CALIBRATION IN ADCS

Non-idealities in ADCs appear as higher speed is demanded. Errors due to inaccuracy and element mismatches affect greatly to the ADC performance. For this reason, some sort of correction is needed to compensate for these errors. When the non-idealities are measured, they could be corrected afterwards using extra analogue or digital architecture. Error compensation could be carried out online or offline, requiring the second method a period of time for calibration [24]. These aspects will be addressed in Section 3.1.

Moreover, error correction is used to improve the precision of a certain block of the ADC or the whole circuit [19]. An overview of offset calibration in comparators is presented in Section 3.2 and most important digital techniques for offset calibration are discussed in Section 3.3, such redundancy or DAC-based calibration. Finally, in Section 3.4 the applied solution to correct the offset in a flash ADC is explained. This solution consists on a digital background calibration based on the adaptive algorithm presented in paper [105]. With this algorithm is, first, measured the offset in the comparator and, then, a calibration code is generated to shift the transition of the comparator to its ideal value and, hence, cancel its offset. Both tasks are explained in this section. Moreover, two different ways of generating the digital input are considered and some possible applications of this calibration method are addressed.

3.1. Calibration Overview

Traditionally, analogue techniques have been employed to compensate for static and dynamic errors in ADCs, specially mismatch, non-linearity and gain deviations. Many analogue methods have been proposed, such as error storage or trimming of elements. The quick evolution of digital technologies has incited the use of digital techniques to correct the resulting errors in ADCs. These digital methods, based on adaptive processing algorithms, greatly reduce the cost of the circuit without attenuating its performance, even for rather complex digital algorithms [24].

Thanks to calibration, errors in ADC are corrected and, thus, it is possible to relax the architecture complexity design. This way, accuracy and speed increases and both power dissipation and area decrease due to its simpler architecture.

The process of error calibration in ADCs is accomplished in two phases. First phase is called *Error Measurement* and it comprises the error estimation regarding to the ideal ADC. Then the second phase, named *Error Correction*, becomes involved to cancel the errors. Both steps could be done either in foreground or background mode. Furthermore, calibration could be fulfilled in the analog or digital domain. Next sections will introduce these terms, giving some examples to a better understanding.

3.1.1. Foreground vs. Background Calibration

Two types of calibration could be carried out depending on whether or not the conversion is interrupted to measure and correct the errors. First group, *Foreground Calibration*, needs a dedicated period of time to perform the calibration. This approach (see Figure 3.1) is also called offline calibration, since the normal operation of the converter is stopped to perform the error measurement or its correction. Both error measurement and correction are not simultaneously executed. During the measurement phase, the ADC input is disconnected and a proper input is then applied to the ADC. Sometimes this input is generated by an accurate DAC placed on-chip, calling this process self-calibration, since the input for measurement cycle is produced by the circuit itself [24].

Most frequently, calibration cycle is carried out at power-on or during inactivity periods of the ADC [24]. Thus, any unexpected environmental change (such as power supply or

temperature) or aging effects will influence the measurements, shifting their values, and, hence, they will be wrong. Then, a new calibration cycle that interrupts the conversion is required to measure the errors properly. Hence, having periodic calibration cycles will be required in ADCs that work for long time [24].

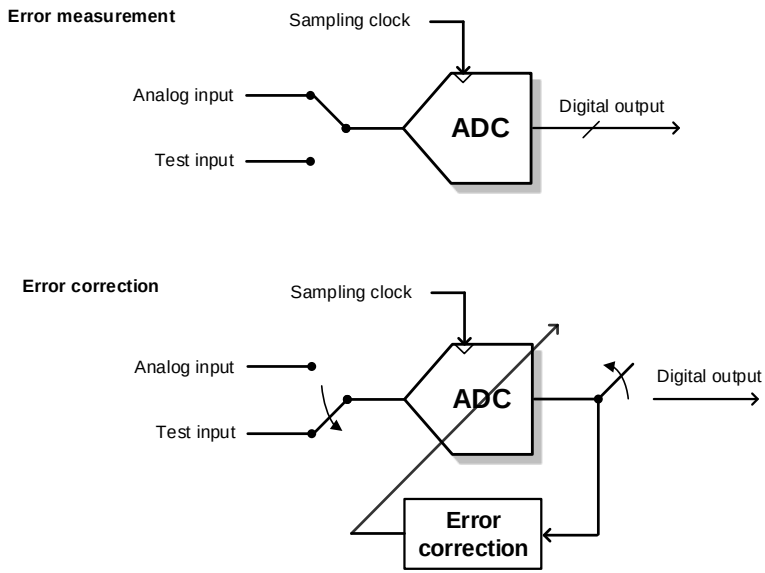

Figure 3.1. Foreground Calibration

There are many foreground calibration techniques applied to Pipeline ADCs, either to correct the linearity error of the DAC [30] [69]-[71] or the amplifiers gains and comparators offsets [27] [30] [72]-[75]. In addition, foreground methods are employed in offset calibration in flash topologies [32] [39]-[40] [76], folding structures [128] or SAR ADCs [143]-[144].

In applications in which the ADC cannot be disconnected or where the parameters have great variations, a second approach has been considered, which is faster than the foreground method. *Background Calibration*, also called online calibration, works simultaneously with the ADC as shown in Figure 3.2. Error measurement and calibration are accomplished simultaneously and continuously without ceasing the normal operation of the ADC. Unlike in foreground techniques, no break in the input signal path exists. All parameters are estimated and corrected in a concurrent way, in contrast to online techniques where the calculation and correction of the parameters are done in a sequential manner [77].

Many different approaches in background calibration are considered. One possible calibration technique is *skip and fill* [26] [78] and consists of skipping occasionally an input sample for measuring the errors and employing nonlinear interpolation to fill the skipped sample. Another calibration technique involves applying correlation-based calibration to correct for the errors in the ADC. In [79]-[81] the interstage gain errors are reduced in pipeline ADCs by means of correlation-based techniques. Papers [37] [82][85] use a noise input signal for calibration, avoiding the use of redundant analogue hardware. However, in some of these architectures the dynamic range of the ADC is decreased because the calibration signal employs some of the input full scale range. In addition, correlation-based calibration with split-channel ADC is widely used, which is faster than the previous methods. In this technique, the original single-channel ADC is split into two identical parallel ADC channels whose devices' sizes are halved of those of the original ADC. The same analogue input is quantized by both ADC channels and the correlation between their digital outputs is then extracted. The split-channel calibration technique has been applied to cyclic [86]-[87] as well as pipeline ADCs [88]. The latter makes use of random choppers to improve robustness of calibration scheme and a radix-based gain correction of the pipeline stages with slight signal-path modification. This technique modifies the structure of the ADC and the correlation-based algorithm needs a large number of input samples, increasing the convergence speed of the ADC. Also, correlation-based digital calibration with random choppers has been utilized for cancelling the offset, gain and timing mismatches between two channels [25] and more than two channels [89] in time-interleaved ADCs. Also, [90] employs random choppers to calibrate the offset of the comparators in a flash ADC, but conversion frequency is limited due to the slow chopping operation. Further, [28] applies correlation-based calibration to radix extraction in pipeline or cyclic ADCs to correct the errors resulting from capacitor mismatches and amplifier gain. A correlation-based algorithm 1000 times faster than the previous papers is presented in [84] applied to a pipeline or cyclic ADC without the need of any additional analogue block and able to correct the interstage gain and capacitors mismatches in MDAC.

Moreover, some calibration schemes require a reference ADC, synchronised with the main converter. This auxiliary ADC is employed to measure the parameters to be corrected and thus it must have low speed and high resolution to perform an efficient conversion of the analogue input. The estimated outputs of both ADCs are expected to

be equal and, based on them, errors can be measured. Several examples could be found in the literature. For instance, in [9] a digital background calibration to calibrate preamplifier and comparator offsets of a folding ADC is published. Calibration employs two-channel ADCs that convert the same input and averaging of the two channel outputs to obtain the whole circuit output. The difference between the two channel outputs is employed in the digital calibration. In contrast with [10] [28] [90], the ADC achieves multi-GHz with this background calibration with no need for extra analogue circuit. Further, in [91] a background calibration technique compares the pipeline ADC output to that of a cyclic ADC and improve the linearity of a fast and inaccurate pipeline ADC. Also, [92] and [93] calibrate the interstage gain in pipeline ADCs by using a slow and high-resolution reference sigma-delta ADC. Since errors are measured by the circuit on-chip, this type of background calibration is self-calibrated. Background techniques are much more effective than foreground architectures, however, area and power consumption are increased.

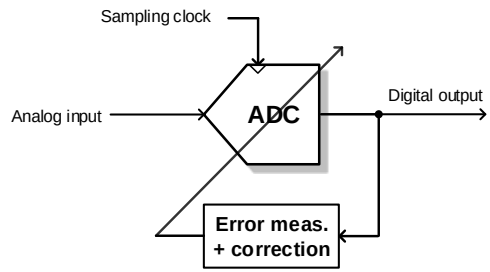

Figure 3.2. Background Calibration

Furthermore, background methods include digital adaptive algorithms [25]-[26] [72] [81] [83] [94]-[104] with quite reduced cost to obtain the estimation of the errors and low power consumption. In this thesis a background self-calibration technique based on this kind of adaptive algorithms is applied to a flash ADC to correct the offset errors in comparators.

3.1.2. Analogue vs. Digital Calibration

ADC errors can be corrected either in the analogue or in the digital domain. *Analogue calibration* fulfills the error measurement in the analogue domain, regardless of whether the error correction is done in the analogue or in the digital domain. On the contrary,

digital calibration performs the error estimation in the digital domain, based on a digital algorithm that measures the error to subsequently compensate for it. This error compensation could be done either in the analogue or in the digital domain

A *mixed-signal calibration* approach uses a digital processor to detect the parameters and then adjusts certain analogue circuits to suppress the effects of these parameters. If error measurement and correction are both done in the digital domain, this type of calibration is called *pure digital calibration*.

Many examples of analogue calibration techniques have been published. In [3]-[4] trim-capacitors are employed to calibrate the DAC mismatches and interstage gain errors in a pipeline ADC. Capacitor-averaging technique has been employed in [5]-[11] to correct the offset in comparators. In [12] a non-restoring algorithm that changes the polarity of the reference depending on the previous bit decision is used to correct errors sources in switched-capacitor in pipeline ADC in the analogue domain without the need of trimming. In [13] a SAR ADC with an analogue background calibration that detects the sign of the errors and modify the reference voltage to correct the offset error of the comparator and the DAC mismatch error. Also, in [14]-[16] the reference voltage is periodically modified in a cyclic and a pipeline ADC, respectively. [17] presents a background analogue technique to calibrate nonidealities in pipeline and SAR ADCs without the need for complex signal processing, extra data converter or extra clock. In addition, paper [18] employs a reference DAC to generate the voltages required to adjust the thresholds of the comparators and suppress the offset errors in the ADC.

Nevertheless, analogue calibration requires additional analogue hardware, hence, exhibiting greater complexity than digital techniques, since the latter employs just a digital algorithm to detect the error. Therefore, digital methods have lower cost and power consumption. They have become very popular since they benefit from fast speed due to the simplicity when measuring the errors, in contrast to the penalty in speed that analogue calibration suffers. Nevertheless, either analogue or digital calibration techniques could be employed in a single block, such as preamplifier, dynamic latch or comparator, or even in the entire ADC.

Digital calibration acts either on the hardware of the ADC or on the output code to correct the errors. For instance, many papers focus on reducing gain, offset and capacitor mismatch errors by means of digital calibration in time-interleaved [25]-[26], pipeline

[27]-[31], flash [32]-[33], algorithmic [28] [34] or SAR [35] ADCs. Others are aimed to calibrate in the digital domain the coder error of a flash [36] or in a pipeline [37] ADC. In Section 3.2 more examples of digital calibration employed to correct the offset in comparators are given.

3.2. Offset Calibration in Comparators

In ADCs, an optimal design of comparators is rather important since comparators are the main blocks in ADCs. ADC properties such as resolution, power dissipation and speed are influenced by the comparator design. The major concern in comparators is the offset. Caused by mismatch devices, offset can distort the output signal of the ADC, hence deteriorating its accuracy.

Typically, comparators for low-power high-speed applications are designed considering small transistor sizing to diminish the power dissipation, thus reducing parasitic capacitances and increasing the comparator speed. However, small elements in the dynamic latch introduce mismatches between the two branches of the latch which originate an offset in the comparator. Therefore, the use of calibration techniques to reduce the offset is essential when using small elements. Calibration acts on the comparator itself to correct the offset by moving its threshold back to its ideal value. Figure 3.3 shows the result of calibration acting on a comparator which presents an offset. Ideally, comparator threshold voltage is t_{ideal}. The exiting offset in the comparator has varied the location of the transition to the new position t_{real}. Thanks to the application of calibration, the imbalance is compensated, moving the transition location back to its ideal value t_{ideal} and, hence, being now the offset value close to zero.

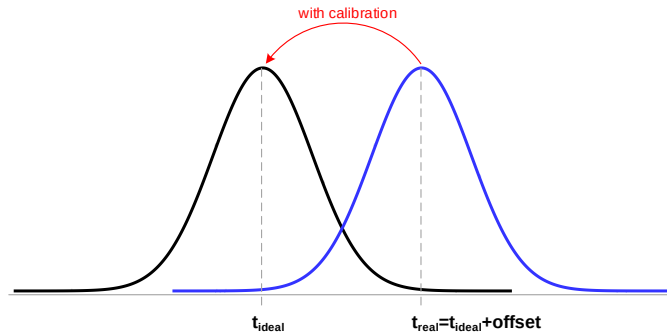

Figure 3.3. Offset calibration application

Offset calibration could be done either in the analogue or in the digital domain. However, digital calibration is more often employed due to its reduced circuitry, cost and power consumption.

Traditionally, auto-zeroing techniques have been used to cancel for the offset in comparators. During the first phase, the input is disconnected and the offset is measured and stored on capacitors. In second phase, the input is connected and the offset is corrected by subtracting it from the signal [19]. This technique is employed either in dynamic latches or in preamplifiers [19]-[21].

A method that could be executed in the analogue as well as digital domain is averaging. As well as auto-zeroing, this technique is utilised in preamplifiers, dynamic latches or even in the whole comparator. Averaging is a popular method for reducing offset in preamplifiers in flash ADCs. This technique uses resistors to connect the amplifier outputs of contiguous comparators. These resistors average the offsets and reduce them. Work [22] presents a flash ADC with two-stage resistor offset averaging to reduce preamplifiers offsets without the need for calibration techniques. Paper [23] proposes a flash whose preamplifier uses a resistor averaging/interpolating network to reduce the random offsets caused by the small device mismatch. Alternatively, capacitive averaging technique could be employed, substituting resistors by capacitors. This technique is effectively used in [6]-[8]. In addition, paper [145] proposes a combination of resistors and capacitors to perform the offset averaging in a flash ADC and fulfill a greater offset reduction than previous averaging methods.

Averaging is often employed in combination with other techniques. For instance, paper [9] proposed a folding ADC with averaging of two comparator offsets and

compensation of device variation by calibration techniques. It employs a two-channel background calibration technique to improve the INL and DNL. The two ADCs convert the same input and their outputs are averaged to give the circuit output. The difference of the ADCs outputs is used in the digital calibration algorithm to reconfigure the reference connection between reference ladder and preamplifiers and correct the offset of the comparators. With this technique the offset of contiguous comparators in each ADC are averaged and reduced. Similar two-channel architecture with averaging is proposed for the algorithmic ADC in [10]. However, digital background calibration uses an adaptive process based on the least mean square algorithm.

In contrast to the offset correction in the analogue domain, different digital methods could be employed, being redundancy and digital-to-analog converter (DAC)-based methods the most employed. While redundancy is aimed only in correcting the errors in the ADC output with no need for measuring [77], DAC-based calibration corrects the offset by first measuring it and then compensating for it. These two techniques are widely explained in Section 3.3.

3.3. Digital Techniques for Offset Calibration in Comparators

As already explained, increasing the transistor sizes in comparator reduces the offset. However, it is convenient to work with small area transistors to reduce power and comparator area. To compensate for the existing mismatch between the two branches of the dynamic latch that appears when small devices are employed in the design, some additional elements are added to the circuit that control the imbalance in order to cancel the offset voltage in the comparator. Several digital methods could be used, being DAC-based calibration and redundancy of great importance. First method modifies the topology of the comparator to vary its threshold to the ideal value, while the second technique adds extra circuits to the comparator.

3.3.1. DAC-based calibration

Digital calibration is typically performed by a DAC, as in Figure 3.4. A DAC-based calibration circuit (also called trim-DAC in [38]) is utilized to compensate for the existing mismatch between the two branches of the dynamic latch. The DAC applies an

imbalance which is opposite to the actual mismatch, eliminating it. The forced imbalance could be created in several ways, for instance, by varying the current in the branches[47] [56]-[58] or by adding capacitance [59]-[61] to them. Once the proper value for the forced imbalance is generated, the offset in the circuit will be suppressed. DAC is formed of a binary weighted array whose elements are current sources, capacitors or resistors. To control the forced imbalance, each of the elements in the DAC turns ON or OFF with switches that are controlled by the digital input control bits $ctrlp$ and $ctrln$, both generated by an algorithm [24]. The full scale range covered by the DAC must be enough to correct the offset [77]. Extra hardware to control the bits is needed. In foreground calibration, DAC control bits could be determined at start-up, while in background the bits are obtained ceaselessly. An advantage of DAC-based background calibration is that the extra digital circuitry required for controlling the DAC control bits works at very low speed during the normal operation of the ADC [77].

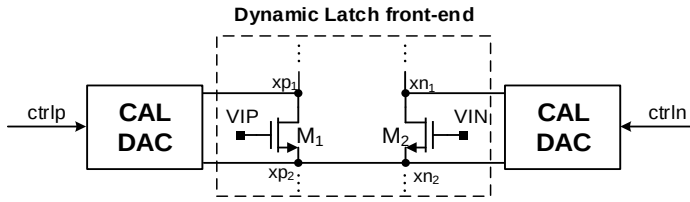

Figure 3.4. Calibration DAC in latch front-end

A dynamic latch using a current source DAC (IDAC) for offset calibration is depicted in Figure 3.5. This simple architecture is similar to those employed in [47] and [57]. IDAC is composed of a bank of binary weighted current sources and switches controlled by the DAC input bits. The DAC is placed in parallel to the input transistors of the latch and injects current through the two comparator branches at the nodes xp_1 and xn_1 to reduce the offset. Work in [58] uses an additional pair of transistors NMOS connected in parallel to the differential input pair of the dynamic latch as well as a charge pump to create a current imbalance to compensate for the offset in high-speed ADCs.

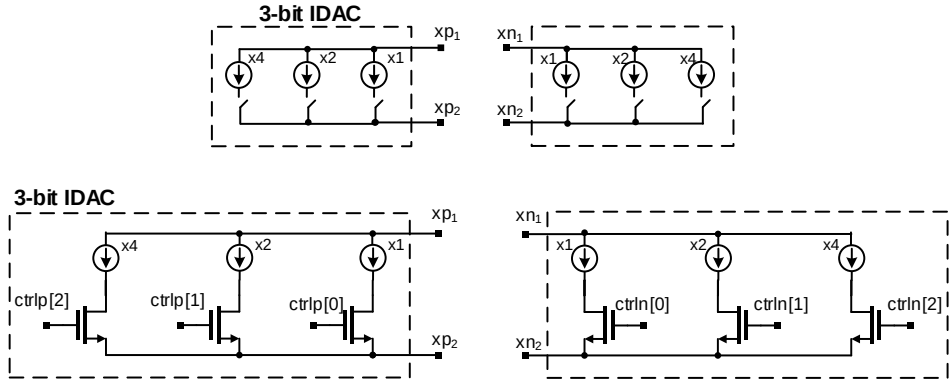

Figure 3.5. Dynamic latch using IDAC

Moreover, a reference ladder could be used to force a current imbalance between the branches of the comparator. Two examples of offset cancelation in flash ADCs are found in [39] and [62]. The proper reference voltage to cancel de offset in the comparator is chosen with switches connected to the resistor array.

In paper [40] [63] a DAC formed of MOS transistor array (as in Figure 3.6) is placed in parallel to the input transistor of the latch in a time-interleaved flash ADC. DAC is utilized to inject currents through the branches of the latch to calibrate offset in a time-interleaved flash. DAC digital control bits ($ctrlp$ and $ctrln$) will increase or decrease to reduce the offset. Also, work [32] applies this method to a flash to cancel the offset.

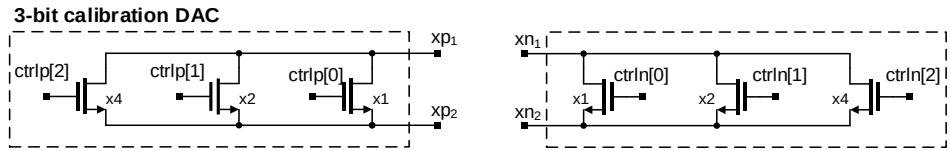

Figure 3.6. MOS calibration DAC (resistor DAC, RDAC)

Furthermore, capacitive imbalance could be created by using a capacitive DAC (CDAC), typically compose of a binary weighted capacitor array (see Figure 3.7). The previous architectures consume only static power while CDACs dissipate dynamic power, being the latter less power consuming. Another advantage of CDAC is that they act as a SH, so no SH circuit is required in calibration circuits [64].

A standard CMOS flash ADC with CDAC calibration is presented in [146]. First, the offset must be stored in the MSB capacitor. Then, each comparator stores the offset on

the capacitor while the input voltage is stored in the capacitor bottom-plate, eliminating the offset. Also, [147] and [148] employ a CDAC for fine tuning of the offset in a flash. Further, capacitive imbalance using MOS varactors array is applied to address the offset problems in latches in SAR [60] [65] and flash ADCs [61]. As well as the previous MOS transistor DAC, MOS varactor array is connected to the internal nodes of the latch and the digital control bits control the gate voltage of the varactors. By modifying the DAC control bits, a voltage difference is applied to the latch inputs and the offset shifts. Once the optimum bits are selected, the offset in the comparator is cancelled. MOS varactor DAC has been used in [59] [61] [64] [66]-[68].

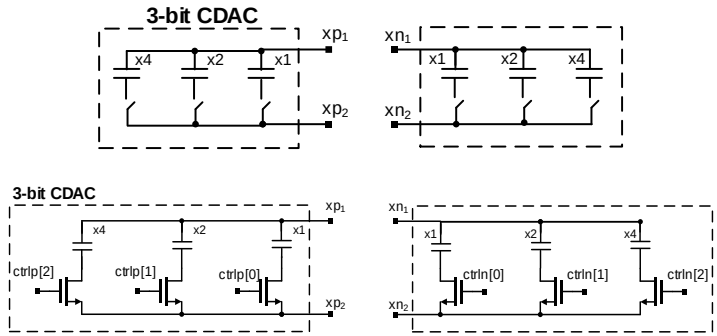

Figure 3.7. Capacitive calibration DAC (CDAC)

3.3.2. Redundancy

Redundancy consists of including additional comparators in the ADC and activating the proper comparator that reduces the offset of the circuit. This technique does not require error measurement. Since hardware is designed to absorb the errors without the need of measuring them previously, redundancy is a pure digital calibration method.

Consider a N-bit flash ADC with $(2^N - 1)$ comparators. The same flash with redundancy has $M \cdot (2^N - 1)$ comparators, with M comparators for each transition. The M comparators have the same topology but different offset. With redundancy, the optimal comparator that provides minimum offset will be selected. The rest of the comparators can be switched off without consuming power, thus being only $(2^N - 1)$ comparators activated, the same number as in a conventional flash ADC [41]. With the use of redundancy, comparators can be designed without caring of the offset. Notice

that as the number of redundant comparators *M* increases, the probably of selecting a better comparator raises.

Work [41] applies redundancy to the flash ADC and, with the use of a finite state machine, the comparator which gives the minimum offset is searched for. Figure 3.8 a) shows the ideal thresholds voltages of the comparators and their actual values, which differ from the ideal values due to offset. In second figure, comparators' positions have changed due to the offset. The selected comparators are highlighted in blue colour. For instance, comparator 2a is chosen for code 2 and comparator 3b is selected for code 3. However, comparator 2b, due to its large offset, has been reassigned [42] to code 4. After reassignment, it is unknown which comparator corresponds to each code and comparators output code is no longer thermometric as in conventional flash. Comparators not selected generate a zero output, without affecting to encoder output and a Wallace tree encoder is used to generate the output in order to disengage comparators with codes. Redundancy with the use of reassigning comparators achieves fast comparators and low power consumption even when the offsets are large.

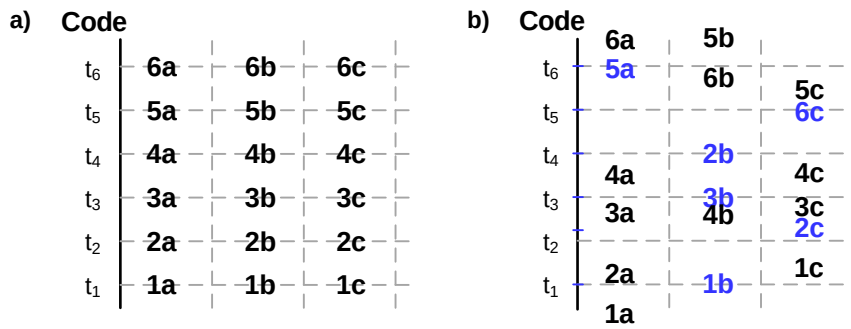

Figure 3.8. a) Ideal threshold voltages of comparators. b) Example of real threshold voltages where reassigned comparators are selected (in blue colour)

Comparator redundancy has been also applied to a flash ADC in [43] and a folding flash ADC in [44]. In [45] an extension of redundancy to cancel the offset in a flash ADC is presented. A statistical element selection methodology based on combinatorial redundancy allows measuring the offset when a certain subset of redundant element is selected, in contrast to conventional redundancy, where only one of the redundant elements is selected. This is a finer calibration than traditional redundancy.

Usually, redundancy is used with DAC-based calibration. For instance, paper [47] proposes a combination between redundancy and current DAC-based calibration to correct the comparator offsets in a flash ADC. This combination achieves small measured offsets and a better resolution and DAC range than using just a DAC to calibrate.

To sum up, redundancy increases the area as extra comparators are needed. However, the devices in comparators can be designed as small as possible, without caring how great the offset is, since using redundancy a great comparator performance is reached. Once the circuit is calibrated, the matching between devices is improved and offset is cancel. Power consumption does not increase with redundancy since the non-selected comparators are at power down.

In SAR ADCs, redundancy is applied to the SAR algorithm, introducing redundancy to the conversion steps. Consider an N-bit SAR ADC with one comparator. The algorithm requires more than N steps to resolve the N-bit output, that is, the algorithm number of steps is redundant. Redundancy in non-binary search algorithm is deeply analysed in [46], proving its benefits in speed when compared to binary search or conventional non-binary algorithm. In order to digitally correct the static nonlinearities in a SAR ADC, work [48] employs a sub-radix-2 redundant architecture where redundancy algorithm sizes the capacitors. Non-binary redundant search has been also used in [49] and in [50] redundancy is incorporated in a tri-level switching algorithm. Further examples of redundancy in SAR ADCs could be found in [51]-[52].

In addition, pipeline ADCs could employ redundancy. Redundancy in pipeline ADC consists of having in each stage a resolution of N-bit and a single bit redundancy for a comparator offset correction algorithm. With this single bit redundancy, the accuracy of each stage could be relaxed. A single bit redundancy has been used in [53]-[54]. Also, the redundancy algorithms employed in SAR and Pipeline ADCs are analysed in [55], comparing their results to that of half-bit redundancy.

3.4. Applied Solution: Adaptive Digital Background Calibration

One of the main goals of this thesis is to prove that background calibration using the adaptive algorithm proposed in paper [105] is a low-cost, stable and effective method

for obtaining high accuracy in flash and pipeline converters. The proposed algorithm in [105] is used to calibrate the comparator offset in pipeline ADCs and it performs two different tasks. First task, the offset adaptive estimation (OAE), is the measurement of the offset in the comparator based on the relationship between the integral non linearity (INL) and the offset in comparators, as explained in [103]. Second task is the calibration code generation (CCG) and consists on the generation of calibration code to move the transition of the comparator to its ideal value, thus cancelling its offset. Both jobs are fulfilled in the digital domain and simultaneously. This is a low-cost technique and allows relax the comparator design requirements, being suitable for low-power high-speed applications. The algorithm is based on the use of an auxiliary converter with higher resolution than the main converter to help calibrate the offset of the main ADC.

3.4.1. Offset Adaptive Estimation (OAE)

In this subsection, some terms and notation are defined. Assume an N-bit flash with M comparators. Its analogue input signal is $x \in [-R, R]$ and the output code is $k = ADC(x)$, where $k \in [0, M]$. Considering the relationship $x \in [t_k, t_{k+1}]$ between the analogue input signal x and the transition voltages t_k, where k is its output code, it can be stated that

$$X \in [T_k, T_{k+1}] \tag{3–1}$$

where X is the N_{cal}-bit digital representation of x ($N_{cal} \gg N$) and T_i the N_{cal}-bit digital representation of transition t_i. Both digital representations have greater precision than the ADC output k. The k-th offset is the difference between the current transition and its ideal value:

$$off_k = t_k - t_k^{id} \tag{3–2}$$

Then, its N_{cal}-bit digital representation of the k-th offset is given by:

$$OFF_k = T_k - T_k^{id} \tag{3–3}$$

with T_k^{id} the digital representation of the ideal transition t_k^{id}.

Hence, equation (3-1) can be written as

$$X \in [T_k^{id} + OFF_k, T_{k+1}^{id} + OFF_{k+1}] \tag{3–4}$$

A simplified schema of the adaptive offset estimation is shown in Figure 3.9

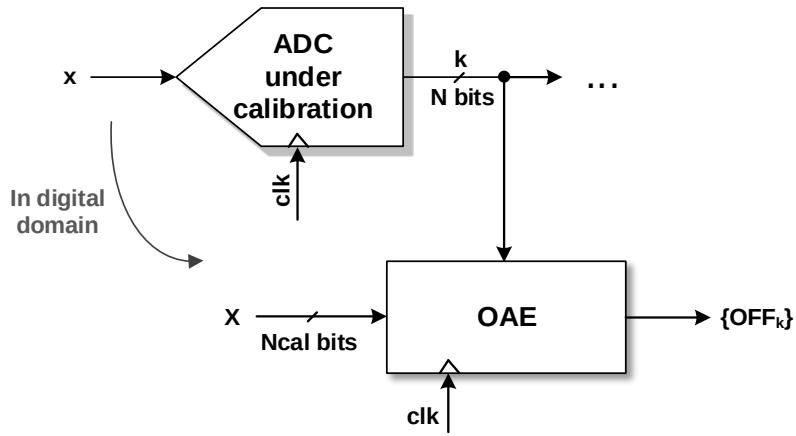

Figure 3.9. Simplified schema of adaptive offset estimation

The offset adaptive estimation (OAE) algorithm verifies continuously if X is within the above interval. If this is not satisfied, the digital representations of the transitions are updated, so does the offset. When $k = ADC(x)$:

$$\begin{gathered} If\ X > T_{k+1} \rightarrow OFF_{k+1}\ increases \\ If\ X < T_k\ \rightarrow\ OFF_k\ decreases \\ otherwise, OFF_k\ and\ OFF_{k+1}\ do\ not\ change \end{gathered} \tag{3–5}$$

Notice that for first ($k = 0$) and end ($k = M$) codes, transitions T_0 and T_{M+1} do not exist and only one of the comparisons is considered in (3-5).

Figure 3.10 shows an example of how the OAE algorithm works. Consider an input x and code k. If X is below k-th transition T_k, the offset OFF_k corresponding to transition T_k is decreased by the OAE algorithm, in order to move X into the interval $[T_k, T_{k+1}]$. In this case, the offset OFF_{k+1} of (k+1)-th transition (T_{k+1}) does not change. If X is greater than T_{k+1}, the offset OFF_k does not change and offset OFF_{k+1} is increased. Otherwise,

that is, when $X \in [T_k, T_{k+1}]$, offsets OFF_k and OFF_{k+1} do not change.

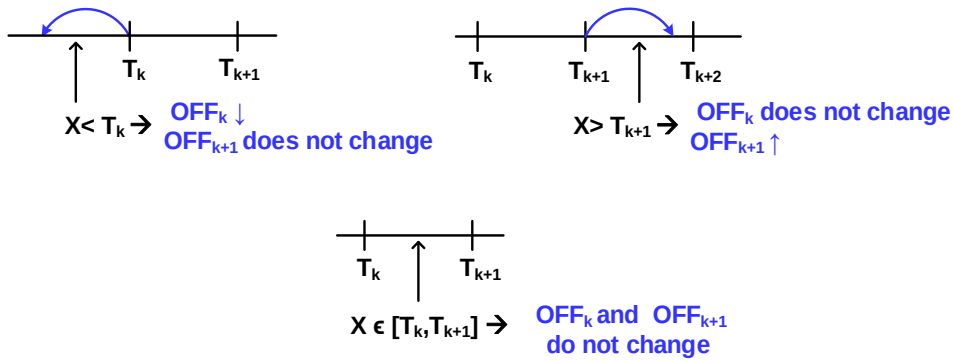

Figure 3.10. Example of OAE task

3.4.2. Calibration Code Generation (CCG)

Every time OAE is updated, the calibration code generation (CCG) algorithm comes into play. When the estimated offset OFF_k is modified, CCG generates a signed calibration code $C_k \in [-|C_{min}|, C_{max}]$ that is applied to the comparator k in order to move t_k closer to its ideal value t_k^{id}. The value of C_k depends on the current value of the offset OFF_k, measured with OAE.

In the analogue domain, the transition after calibrating $t_k^{after\ CAL}$ can be expressed in terms of the transition before calibrating $t_k^{bef\ CAL}$, the correction step δ and the calibration code C_k:

$$t_k^{after\ CAL} = t_k^{bef\ CAL} - \delta \cdot C_k \tag{3–6}$$

The correction step δ could be represented as:

$$\delta = \frac{off_{max}}{C_{max}} \tag{3–7}$$

where off_{max} is the maximum offset that calibration codes can correct. An example of a 2-bit calibration code is depicted in Figure 3.11.

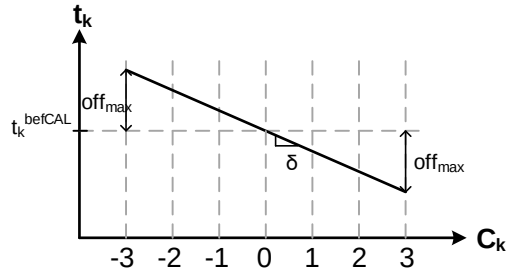

Figure 3.11. Example of 2-bit calibration code

The CCG is performed in the digital domain as well as OAE. The CCG block forces the offset to remain within a small interval:

$$OFF_k \in [-\Delta, \Delta] \tag{3–8}$$

$\Delta = floor\left(\frac{\delta}{q_{cal}}\right)$ is the digital N_{cal}-bit representation of the correction step δ and $q_{cal} = \frac{2R}{2^{N_{cal}}}$ is the LSB of the input estimation X. The algorithm checks if the offset is within the mentioned interval. If this is accomplished, offset has achieved a low enough value and calibration is complete. If the offset is greater than Δ, the calibration code increases, otherwise its value is reduced as shown:

$$\begin{aligned} If\ OFF_k > \Delta \rightarrow C_k = C_k + 1 \\ If\ OFF_k < -\Delta \rightarrow C_k = C_k - 1 \end{aligned} \tag{3–9}$$

3.4.3. Input estimation for calibration purpose

In [103] two ways of establishing the use of the digital representation X are proposed. First method includes the use of a high-resolution DAC with input X as in Figure 3.12, however it works in foreground mode and calibration requires stopping the circuit operation. In contrast, a high-accurate low-speed auxiliary ADC (see Figure 3.13) has been considered to work in a background mode without interrupting the normal operation of the main ADC. Both ADCs have the same input. Although both auxiliary ADC and DAC need to be one or two bits more accurate than the main ADC, the auxiliary ADC topology has been chosen for this work because of its power dissipation (less than 5% of the whole circuit) [103], since it works at a very low operation frequency.

However, in background calibration synchronisation of the auxiliary ADC with the main ADC is no easy task and it will be explained in Section 5.3.

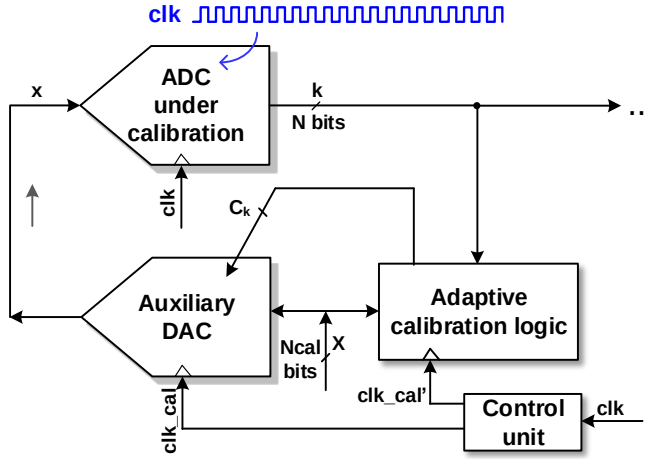

Figure 3.12. Analogue input x generation with an auxiliary DAC in foreground calibration

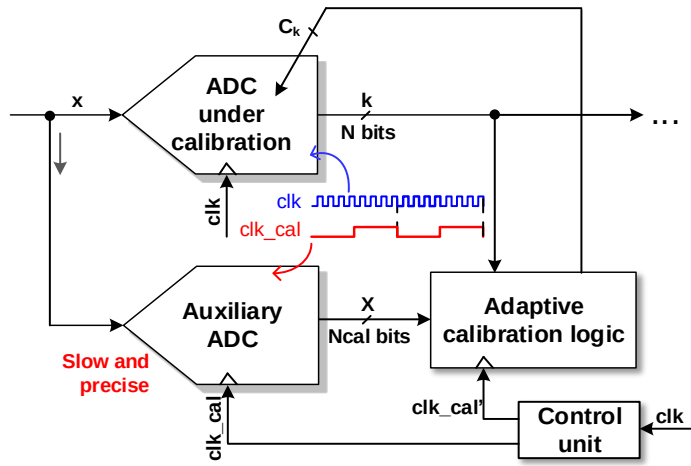

Figure 3.13. X generation with an auxiliary ADC in background calibration

3.4.4. Possible applications

The adaptive background calibration could be applied to different ADCs, such as flash, pipeline or folded types. The only necessary condition to properly use this method is that the auxiliary ADC must be slower and more accurate than the main ADC. The algorithm always fulfills the equation (3-5). However, there are several ways of applying this algorithm to an ADC.

Consider a flash ADC with a resistor ladder and each comparator composed of a preamplifier, a stand-alone dynamic latch (SADL), a RS latch and buffers. First application of this adaptive calibration is straightforward and consists of trimming the value of the resistors in the reference ladder, as shown in Figure 3.14. Comparators reference voltages are generated externally by the resistor string. Then, the variation of their resistances implies the shifting of the comparators' transitions. Assuming that trim-resistors are composed of resistors and switches, calibration code C_k could be applied to a set of switches, activating and deactivating them, hence, changing the values of the resistors in the reference ladder. Nevertheless, this method is rather costly, since a great number of resistors is required.

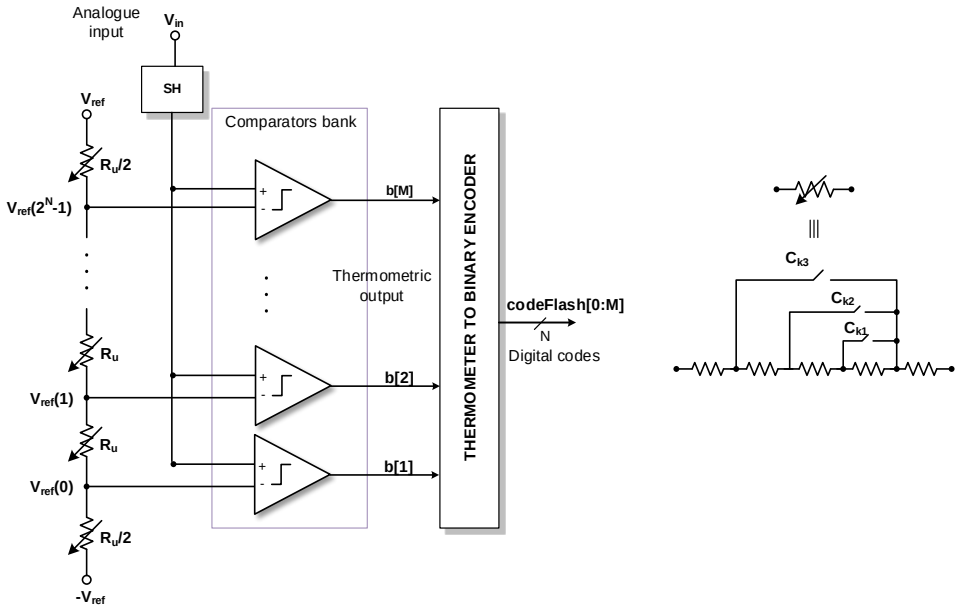

Figure 3.14. Adaptive calibration applied to resistors in a flash ADC

Another possibility is to apply the calibration code C_k to the preamplifier to move the transitions of the comparators, as exhibit Figure 3.15 . For instance, a current source could be placed in parallel to one of the input transistor of the preamplifier. With a switch controlled by C_k, the current source is set to on or off, allowing a greater current through the branch and, hence, creating an offset between both branches.

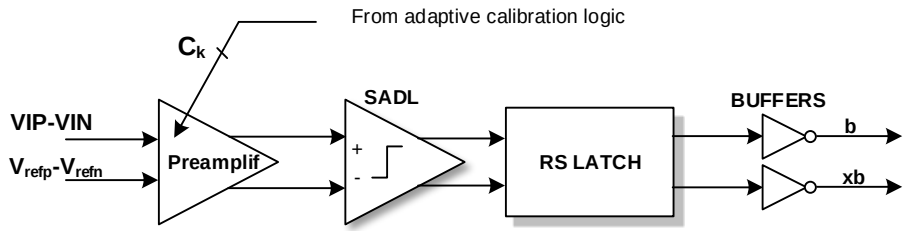

Figure 3.15. Adaptive calibration applied to preamplifier in a flash ADC

Sometimes, a switched capacitor circuit is placed before the preamplifier. The calibration code C_k could be applied to the SC circuit to change the capacitors capacitance. Since the SC calculates the difference between the transition and the reference voltage, varying

the capacitors of the SC will move the transition as expected.

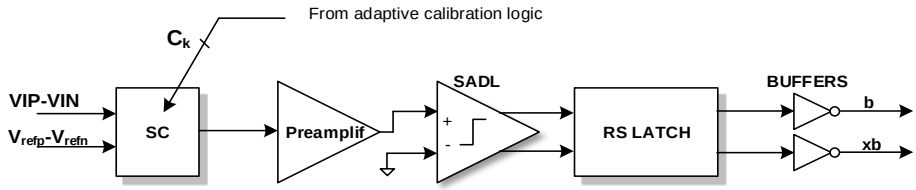

Figure 3.16. Adaptive calibration applied to SC in a flash ADC

This adaptive calibration could be also applied directly to the SADL as in Figure 3.17. The threshold of SADL is modified by applying the calibration code C_k. The SADL includes a DAC in parallel to its MOS transistors, as explained in Section 3.3.1, and C_k sets on and off the devices in the DAC to generate an imbalance between the branches of the SADL, thus creating an offset that moves the transition of the comparator (see Figure 3.17). This is the proposed calibration of paper [105].

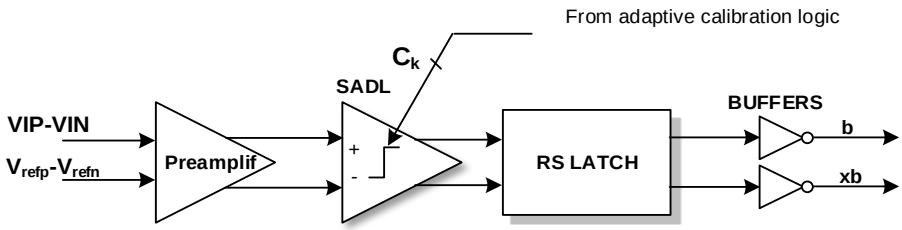

Figure 3.17. Adaptive calibration applied to a SADL with built-in transitions in a flash ADC

The cost of some of the solutions presented above will be analysed in next chapter and parameters of the adaptive algorithm will be studied in Chapter 5.

4. DESIGN OF A COMPARATOR AND ITS CALIBRATION CIRCUIT IN A FLASH ADC

Among the different types of ADCs, flash ADC is the most sensitive to offset [77]. As already described in Section 2.4.1, offset is the main issue in flash ADCs. Therefore, using calibration techniques to reduce the offset is essential in this kind of converters.

This chapter explains step-by-step a flash ADC design which does not require an explicit voltage reference, such as a resistor ladder. Instead, the reference is implemented built-in in the comparator using the transistor sizes. Further, no elements that contribute to DC power dissipation are employed (i.e., reference ladder or preamplifier).

In this chapter the design of a comparator and its calibration circuit are explained. These considered aspects to design the flash ADC are discussed in Section 4.1. Section 4.2 presents the comparator structure. Section 0 goes deep into the dynamic latch design and, finally, offset measurement and calibration DAC design are exposed in Sections 4.4 and 4.5, respectively.

4.1. Flash ADC considerations

In this section, some of aspects to take into account when designing a flash ADC are described. Transitions as a function of resolution and the different ways of generating the transitions of each comparator are explained. Also, dynamic latches performance is examined. Further, offset problems are output code errors are considered.

4.1.1. Comparator reference voltage generation

Due to the current high-speed trend, CMOS dynamic latched comparators are very attractive for many applications, such as high-speed ADCs. Different architectures for comparators have been presented in last years. Most of the ADCs use external references to generate the reference voltages, while others generate the references internally.

The most common comparator has both external input and reference voltages. Switched-capacitor (SC) differential comparator [149] includes a SC circuit before the preamplifier (as in Figure 4.1.a) which gives the difference between the analogue inputs (VIP, VIN) and the reference voltages (V_{refp}, V_{refn}). Comparator positive input is connected to ($V_{in} - V_{ref}$), generated by the SC circuit, and negative input to ground. That is, every comparator in flash will be the same since they all compare the input ($V_{in} - V_{ref}$) to the transition $t_{id} = 0V$, having V_{ref} a different value for each comparator. It is usually implemented with a preamplifier. This is the best solution when resolution is high, since transitions are rather close. Its main advantage is the small introduced capacitive load, providing high common-mode rejection. In [150] a preamplifier embedded into the SC circuit is proposed. The preamplifier gain is obtained by changing the values of the capacitances of the SC circuit, providing no static power dissipation.

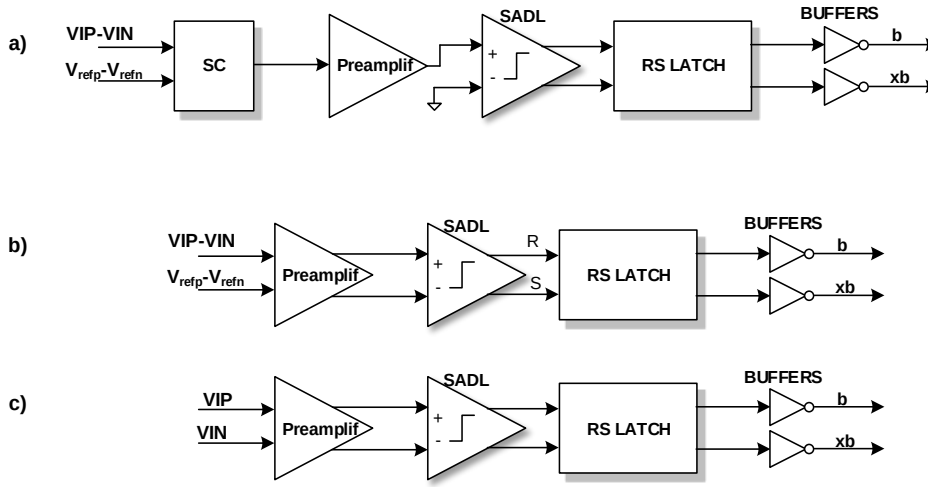

Figure 4.1. Types of comparators: a) SC comparator with external references, b) comparator with external references, c) comparator with built-in transitions.

Most used comparator connects its positive input to the differential input signal ($VIP - VIN$) and its negative input to the differential reference voltage ($V_{refp} - V_{refn}$), as depicted in Figure 4.1.b. Notice that no sample-and-hold (SH) is required to sample the input signal. The comparators themselves are in charge of the sampling, achieving much higher speed and faster flash than using a SH circuit [19].

The simplest way of generating the reference voltages externally is by placing a resistor string connected between the positive and the negative references R and $-R$ [24], as shown in Figure 4.2. The resistor ladder divides the reference R into 2^N reference voltages, all of them equally spaced, and comparators will compare these reference levels to the input voltage [19].

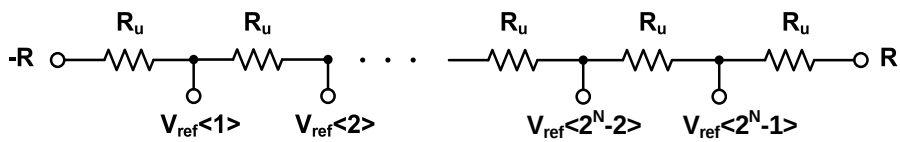

Figure 4.2. Reference ladder

Variations in resistors values will lead to incorrect reference voltages. Mismatches between resistors in layout will cause errors in reference voltages. Also, environmental variations due to temperature gradient along the resistive ladder will result in erroneous

reference levels. As flash ADCs in CMOS process are limited to 8 bit resolution, these variations are not a major concern.

Generating reference voltages internally allows eliminate the resistor array used in conventional ADCs. A comparator with built-in reference voltage generation [147] [151] is the simplest topology, since ADC chip has no external connection to the reference [124]. Analogue input voltages (VIP, VIN) are directly connected to the comparator differential inputs as exhibits Figure 4.1.c. Transition is generated built-in in the dynamic latch by introducing an intentional imbalance in the input differential pair, shifting the threshold point of the comparator to a certain voltage [147]. This imbalance is achieved by varying the widths of the input transistors until obtaining the optimum reference voltage. In a flash, each comparator needs a different reference voltage and, hence, each dynamic latch has different input transistor sizes to generate each reference voltage. Embedded transition generation avoids the use of external circuit to generate the references, such as resistor string, reducing thus drastically the power consumption in the circuit. For all these reasons, generating built-in transitions is the solution considered in this work.

4.1.2. Transitions in flash ADCs

Consider an N-bit flash with a full-scale range of $FS = 2R$. The maximum number of comparators in flash is $M = 2^N - 1$, that is, $2^N - 1$ transitions are needed. The quantum step is $q = LSB = \frac{2R}{2^N}$ and the ideal threshold transitions are given by:

$$\{t_{id}\} = \{-R + q, -R + 2q, \dots, R - q\} \tag{4–1}$$

The output characteristic of an ideal 3-bit flash with $FS = 2R = 2V_{pp}$ is shown in Figure 4.3. The flash has $M = 7$ transitions, quantum step $q = \frac{2V}{8} = 250mV$ and the following ideal transitions:

$$\{t_{id1}, t_{id2}, \dots, t_{id7}\} = \{-750, -500, -250, 0, 250, 500, 750\}mV \tag{4–2}$$

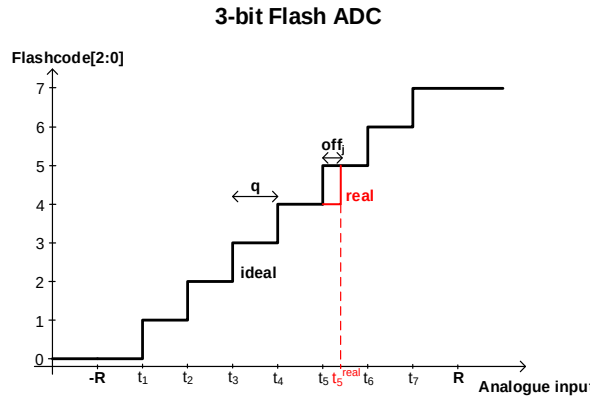

Figure 4.3. Ideal 3-bit flash ADC output characteristic

Table 4.1 gathers some parameters for flash ADCs with different resolutions.

Table 4.1. Parameters of flash ADCs with $FS = 2V$

N	M	$q\ (mV)$	Minimum transition, $t_{min}\ (mV)$	Maximum transition, $t_{max}(mV)$
2	3	500	-500	500
3	7	250	-750	750
4	15	125	-875	875
5	31	62.5	-937.5	937.5
6	63	31.25	-968.75	968.75
7	127	15.625	984.375	984.375
8	255	7.8125	-992.1875	992.1875

The threshold transition t_j of a comparator will vary from its ideal value when any offset in the comparator exists, since offset is applied to the comparator differential input. In an ideal ADC the difference between two consecutive transitions (t_j^{id} and t_{j+1}^{id}) is a constant:

$$q_j = t_{j+1}^{id} - t_j^{id} = q \qquad (4\text{–}3)$$

In practice, already discussed second order effects introduce deviations that modify the transition' location and offset appears. The offset of two consecutive transitions will modify the corresponding quantum q_j this way:

$$q_j = off_{j+1} - off_j + q \tag{4–4}$$

The maximum achievable offset in comparators to ensure no missing codes or monotonicity is [24]:

$$off_{max} = \frac{q}{2} \tag{4–5}$$

For a 3-bit flash ADC with $FS = 2V$:

$$off_{max} = 125mV \tag{4–6}$$

If the number of bits N increases, the number of comparators and, hence, the transitions rise. Since the full-scale range has not been modified, there is a greater number of transitions in the same FS and transitions are closer.

Furthermore, if flash reference R decreases, the total number of transitions must be now in a smaller FS range, locating the transitions closer to each other. The likelihood of inverting two transitions is much greater, even if comparators have small offset. In this work, the proposed calibration method combined with the use of a Wallace Tree encoder is able to correct inversion between transitions.

4.1.3. Dynamic latched comparator

Comparators are one of the basic blocks in ADCs since every ADC employs at least one of them in its topology. Hence, it is really important to understand how comparators work, since they have a great influence in the ADC dynamic performance.

The architecture of a typical high-speed differential latched-comparator is illustrated in Figure 4.4. A dynamic latch is employed to reduce the required conversion time [152]. The dynamic latch is form of a cross coupled inverters in a positive feedback configuration which are capable of amplify a small differential input voltage into a full-scale digital output in a short time [153]. Also, a preamplifier is used to achieve a higher

voltage difference at the input of the dynamic latch, minimizing the conversion time.

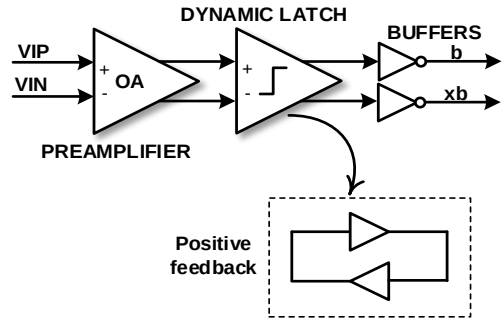

Figure 4.4. Typical high-speed dynamic latched comparator

Figure 4.5 shows the evolution of the output dynamic latch. Two modes of operation are considered. During first phase (reset or precharge mode), the preamplifier amplifies the differential input voltage while the latch is turned off. During the second phase (decision or comparison or evaluation mode), the preamplifier is disabled, the latch is turned on and the positive feedback makes a decision (the output goes high or goes low) based on the comparator inputs generated by the preamplifier. The clock signal determines the sampling instant without requiring the use of a SH circuit.

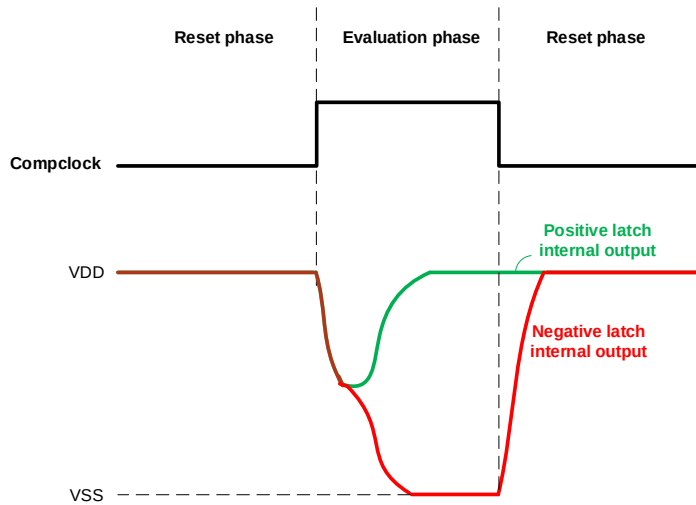

Figure 4.5. Output evolution of a dynamic latched comparator

This dynamic latched comparator topology must ensure that no memory is carried from

one decision cycle to the next one. If this happens, hysteresis will appear. To decrease hysteresis, usually latch internal output nodes are reset to power supply during reset phase. This also accelerates the decision phase [153].

As explained in Section 2.2.3, this kind of latches can suffer from metastability when differential input signal is small. Employing a preamplifier with enough gain will decrease metastability errors. The quick change of the comparator from reset to decision mode creates a great amount of noise at the comparators' input. Using a preamplifier before the comparators is a common way of suppressing the kickback noise injected to the reference ladder as well as the comparator offset due to mismatches between its elements. This kickback limits the accuracy of the comparator and without the preamplifier this kickback will originate large glitches in the input voltages [153].

Furthermore, the major drawback of dynamic latches is their great sensitivity to mismatch [152]. Any mismatch between transistors in dynamic latch will result in large offsets, degrading the linearity output signal and, hence, limiting the accuracy of the comparator. Hence, it is essential to design the comparator properly to reduce this offset. Using large devices will decrease mismatch between elements at a cost of increasing the decision time as well as the power consumption. Amplifying the small differential input with a preamplifier will reduce the input-referred offset associated to the dynamic latch by its gain.

In the design of the flash ADC no preamplifier is used, since it increases the static power dissipation of comparators. To overcome kickback problems without the use of a preamplifier, transistors' dimensions are decreased to reduce input currents. Although offset voltage will increase, offset calibration techniques (discussed in Chapter 3) are employed to supress the offset in the circuit.

4.1.4. Sparkles in thermometer codification

Considering a high-speed input signal, some comparators in the flash could measure a reference voltage that is higher or lower than the actual one, giving at their thermometric outputs an erroneous code. The appearance of an incorrect '0' at the comparator output is called sparkle and it occurs close to the transition point. Sparkle errors (also called 'bubble' errors) are generated because of the uncertainty in the sampling instants, since no SH is used. Sparkles can also occur if the offset in comparator is large enough. Taken

into account that comparators are usually designed to have minimum offset, sparkle errors are mainly created by timing errors in the comparators of the flash [19]. As resolution and speed in flash ADC increase, more bubble errors appear. Table 4.2 shows two examples of incorrect codes in thermometric codification. Flash output with sparkles is an incorrect representation of the sampled input and the SNR of the sampled signal is deteriorated with sparkles. Decoding this thermometric code is not an easy task since decoders make great mistakes in presence of sparkles [19].

Table 4.2. Sparkles (in red colour) in 7-bit thermometric code

	Thermometer code (MSB…LSB)
No sparkles	0011111
1 sparkle	0101111
2 sparkles	1001111

There are many approaches proposed to cancel bubble errors in flash. The use of a Gray encoder between thermometer and binary codes reduces the probability of having bubble errors, since Gray code does not suffer from degradation when more sparkles appear [19]. Another strategy is employing bubble error correction (BEC) circuit in typical encoders to suppress these errors, but extra circuitry increases the power consumption and area of the circuit. For instance, ROM-based encoder (based on Gray or one-hot codes) [154]-[156] is the simplest encoder, but it has slow speed and high power consumption. A more power efficient circuit [157] is the fat-tree encoder [158], but it has a much more difficult layout and is more time consuming than the former. Both require 3-input AND gate to correct single bubble errors.

A topology with fewer elements and regular structure that reaches an easier layout design than the previous architectures is MUX-based encoder [159]. It is made only by multiplexers and its main disadvantage is the huge fan-out that increases power consumption and delay. It employs 2-input OR gate to supress the bubble error [155]. Unfortunately, if there are second order bubbles in thermometric output (as shown in table above), none of the encoders above will be able to correct the bubbles and the flash output will be incorrect [160].

In paper [161] the authors present a simple ROM architecture with slight modifications which is able to correct up to second order bubble errors and also eliminates the Gray encoder ROM. In [154] a ROM encoder capable also of correcting second order bubble errors is proposed. Paper [160] proposes a circuit to detect all bubble errors.

To overcome the errors caused by sparkles without the need of a BEC circuit, Wallace Tree encoder [162]-[163] is employed. It can suppress all bubble errors, since it counts the number of '1's in the thermometric code, independent of their positions. The process is similar to reordering the codes, as in an ideal thermometric code. Table 4.3 gathers examples of bubble errors in thermometric codifications bt_i and the binary output representation. A 3-bit Wallace tree architecture is shown in Figure 4.6. This encoder has a hierarchical tree topology and, hence, increasing one bit in flash resolution will introduce one stage extra, duplicating the number of elements of the encoder. This will lead to disadvantages like increase in power and area and also relatively low speed due to the tree structure.

A review and other improvements to the above topologies could be found in [155]-[157] [159] [164].

Table 4.3. Wallace Tree performance

Number of sparkles	7-bit thermometer code (MSB…LSB)	3-bit binary code (MSB…LSB)	Correct code?
No sparkles	0011111	101	Yes
1 sparkle	0001111	100	No
1 sparkle	0010111	100	No
1 sparkle	0111111	110	No
2 sparkles	0101111	101	Yes
2 sparkles	1001111	101	Yes

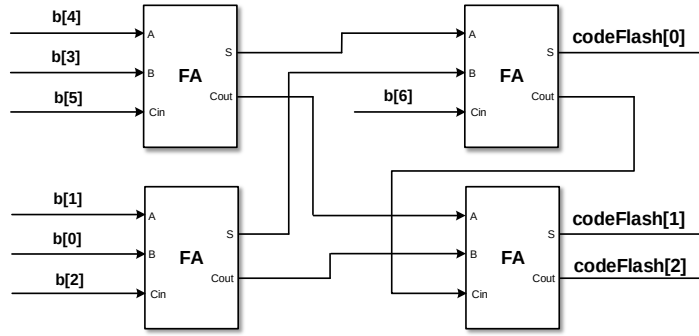

Figure 4.6. 3-bit Wallace Tree encoder

4.2. Comparator design

One of the aims in this thesis is to design a low-resolution and high-speed self-calibrating flash ADC with reduced area. To achieve this, a comparator based on a dynamic latch (DL) with built-it transition is designed (similar as the one shown in Figure 4.1c). Dynamic latch with differential structure offers a high level of input common-mode rejection. Further, no elements that dissipate DC power, such as preamplifiers or resistor ladder are used. This way, a faster response and lower power dissipation are achieved. Since a DL with no preamplifier and no resistor ladder is highly sensitive to environmental condition and mismatches, offset calibration in comparators is recommended.

This comparator is going to be part of a flash ADC with a FS of $2V$. Its topology is depicted in Figure 4.7. The basic block of the comparator is the dynamic latch. The DL is followed by a RS latch and buffers. The RS latch has a typical CMOS design.

The offset of the DL is going to be calibrated by a calibration DAC (Cal DAC in the picture). Since an offset calibration technique is applied to the flash ADC, comparator design is quite relaxed and dynamic latches with minimum size devices are used. Also thanks to calibration, no preamplifier is required since there is no need for a precise signal.

In the following section a study on the design of a dynamic latch focus on finding the best figure-of-merit (FoM) based on settling time and power consumption is accomplished. In addition to this, several DL topologies are introduced and compared

evaluating their $FoMs$. In Section 4.4 the static offset measurement procedure is explained. In Section 4.5 the calibrating DAC is designed for each DL architecture, measuring their offsets and calibration ranges. The DL and bank which fulfills the best performance is selected for the flash ADC design

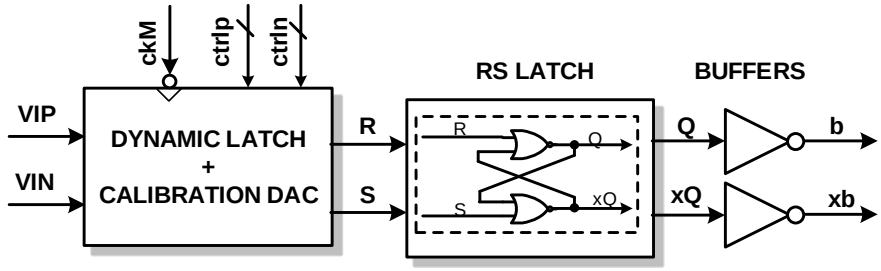

Figure 4.7. Comparator structure

4.3. Dynamic latch design

In this sub-section, a study on the design of a fast and simple DL topology based on a method to find the best trade-off between power consumption and settling time is fulfilled. Several DL architectures are considered and analyzed. Finally, the topologies with the best performance are selected.

4.3.1. Dynamic latch operation

Figure 4.8 exhibits the transient evolution of the outputs of a dynamic latched comparator. The decision point (also called quiescent point, Q) of the regeneration latch is the point where the differential outputs are equal and it is expected to be close to the transition. Some considered time parameters are indicated in the picture. Quiescent time (t_q) represents the spent time from the instant where the latch clock CLK is set to high until the latch makes a decision (at decision point Q). The settling time (t_{set}) is the time invested in doing the decision. It is measured from the point where the latch turns to high until the point t_{90} in which the dynamic latch is said that reaches the stationary

state. This stable situation is measured at a 10% of the final stationary value VDD supply. Notice that during reset phase, the dynamic latch has already decided and no power is consumed. However, during the evaluation phase (also called comparison phase), the dynamic latch is evaluating the inputs and, hence, consumes.

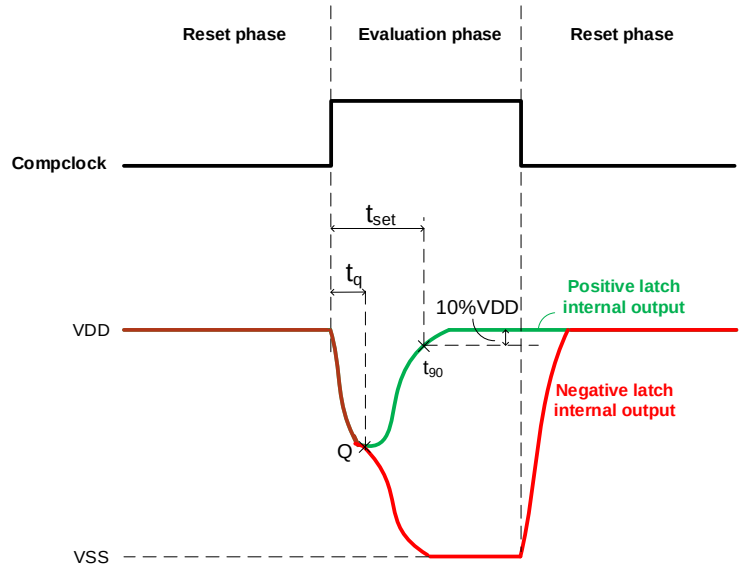

Figure 4.8. Output evolution of a dynamic latched comparator

4.3.2. Different dynamic latches topologies

Among all different DL topologies employed during last years, most of them use external threshold voltages and include SC circuits to subtract the reference voltage from the input voltage [7] [165]-[174], several employ a resistor ladder [172] [175]-[178] to generate the thresholds and a few have built-in threshold voltages. Dynamic latches with external references (V_{refp},V_{refn}) as depicted in Figure 4.1b add extra input transistors for reference voltages [179]-[181] and, since our design aims to generate transitions internally, this type of DL has not been considered in this study. In this section, seven different SADL architectures compatible with comparators with built-in transition are presented below and their working are explained.

Consider the conventional DL with current source (DL-CS) [166] [172]-[173] [182]-[183] depicted in Figure 4.9. Transistors $M1 - M2$ compose the differential input pair, the

regenerative latch is formed of two inverters ($M3-M5$ and $M4-M6$), $M7$ and $M8$ are switches to reset the nodes and $M9$ is a current source, as shown in the picture.

During reset phase (CLK set to low), NMOS switch is off and PMOS reset transistors $M7$, $M8$ are conducting. As a result, internal output nodes are reset to VDD. When CLK changes to high value, comparison phase starts: NMOS differential input transistors pair are perfectly matched and in linear region, as well as NMOS switch, and PMOS reset switches are open. Outputs start discharging from their previous value VDD (charged at reset phase) and in case of having $VIP > VIN$, the positive output VOP discharges faster than negative output VON and regeneration latch starts, leading to a situation where VOP achieves VSS value and VON reaches VDD. In the opposite situation, where $VIP < VIN$, VOP is finally set to VDD and VON to VSS. This internal nodes reset helps eliminate the probability of having memory effects in the latch and, hence, an offset that affects the threshold value.

The threshold voltage of the comparator is zero ($t = 0V$) if transistors of both branches ($M3-M5$ and $M4-M6$) are perfectly matched. This kind of latch achieves high input impedance and rail-to-rail output swing with no static power consumption.

dynamic latch is evaluating the inputs and, hence, consumes.

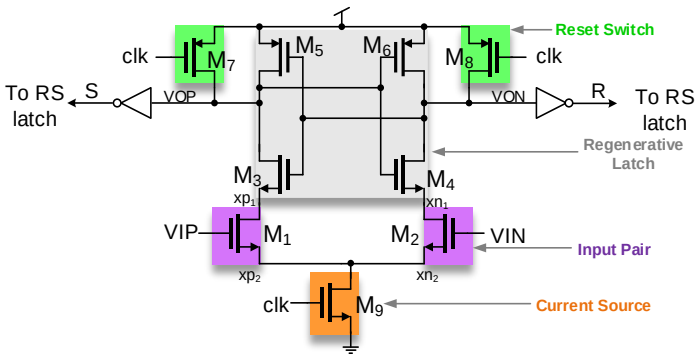

Figure 4.9. Conventional dynamic latch with current source (DL-CS)

Some common variations of conventional latch DL-CS are illustrated in the following figures. A PMOS-input DL with current source (PDL-CS) [165] is depicted in Figure 4.10. It has a similar topology than DS-CS, but input transistors are PMOS instead of employing a NMOS pair. Figure 4.11 shows a DL with current source and internal nodes resets (DL-CS-INR) [167] [169]-[170] [175] [184]-[185]. These internal nodes resets are

accomplished by adding two extra PMOS switches that reset the internal drain nodes of the input transistors in order to avoid memory problems. Further, the DL with no current source (DL-NoCS) [7] presented in Figure 4.12 has the same structure as DL-CS but it eliminates the NMOS current source. Also, in Figure 4.13 is exhibited a DL with no current source which incorporates switches to open the regenerative latch (DL-NoCS-LS) [178] during reset phase, eliminating the existing path between VDD and VSS. In addition, a DL with no current source and switches to open the regenerative latch and regenerative latch NMOS in parallel to NMOS input (DL-NoCS-LS2) [105] is depicted in Figure 4.14. It has two NMOS switches to suppress the connection between VDD and VSS. This type of structure makes the latch faster than the previous one. Finally, a DL with no current source and internal nodes reset switches (DL-NoCS-INR) [170] is shown in Figure 4.15 and includes two NMOS switches to disconnect the supplies during reset phase.

Most of the current DL topologies are based on double-tail latched comparators. Conventional double-tail dynamic latches are found in [182]-[184]. In this kind of topology offset and noise are reduced, since the input transistors of the tail part are the only transistors that contribute to noise, but their sizes are greater and the number of transistors is increased compared to the conventional DL. Improved double-tail latches are presented in [183] [185]-[186]. Although the outstanding performance achieved by double-tail latches, the goal of the analysis in this chapter is to design a simple comparator to be included in a flash ADC with small area and for this reason, double-tail type latches are not considered.

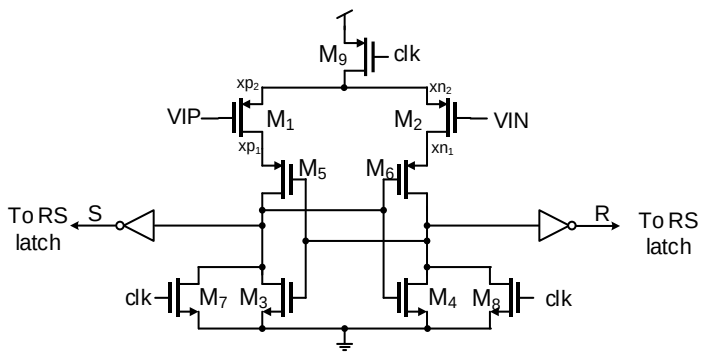

Figure 4.10. PDL-SC

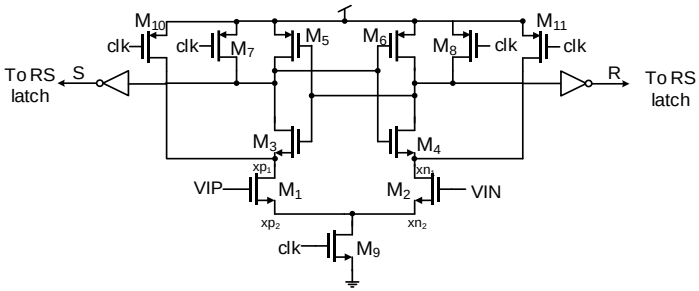

Figure 4.11. DL-CS-INR

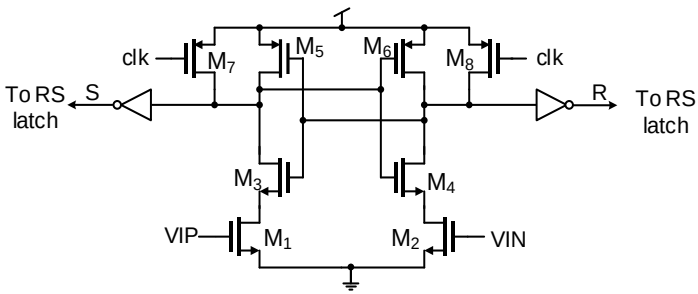

Figure 4.12. DL-NoCS

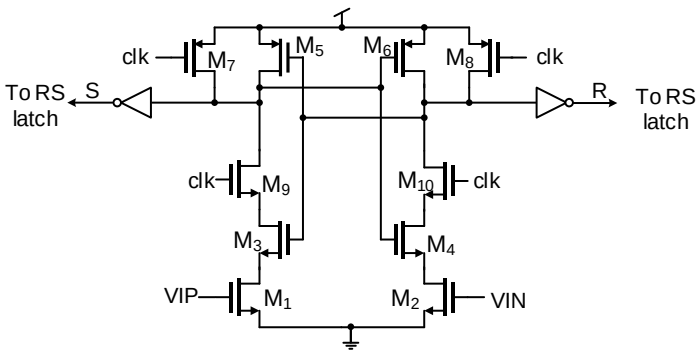

Figure 4.13. DL-NoCS-LS

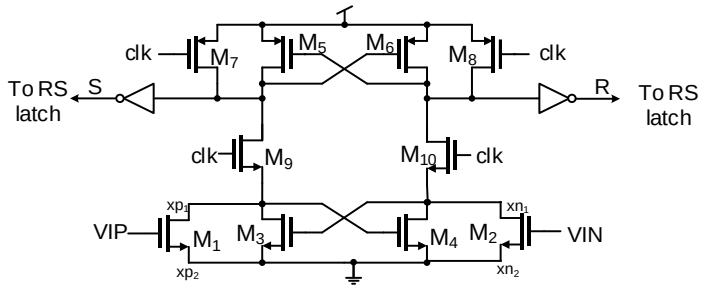

Figure 4.14. DL-NoCS-LS2

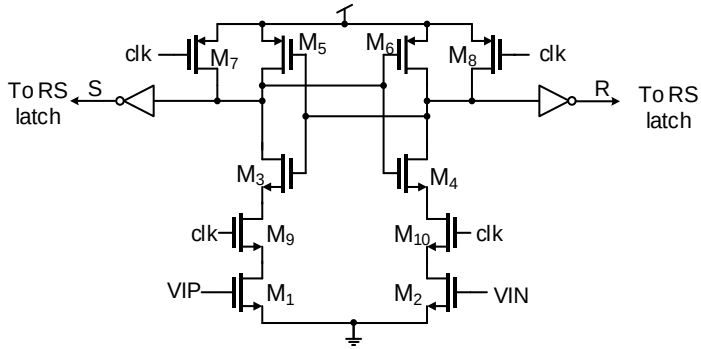

Figure 4.15. DL-NoCS-INR

4.3.3. Comparison study

In this sub-section the performance of the topologies above presented is evaluated. To achieve this, the power dissipation and settling time are measured in open-loop and the architecture with the best trade-off between both parameters is selected as the one that presents the best performance. All measurements are done with transistor-level simulations in a 180nm technology using Cadence Virtuoso tools with supply voltage of 1.8V.

Open-loop simulation setup consists of a DC analysis where the clock signal has a frequency of 100MHz, a pulse width of 2.5ns and a common-mode input voltage of 0.9V (midpoint between power rails). All DL architectures are designed for a threshold voltage of $t = 0V$, which represents the symmetrical transition in a flash ADC. In a

previous DC analysis the sizes of the transistors of every DL topology have been selected to minimize both the settling time and power consumption when the differential input voltage is closed to the transition (that is, for an input voltage of $1mV$). Transistors width (W) and length (L) design parameters for conventional latch DL-CS are shown in Table 4.4. Two different cases are studied for the input and regenerative latch transistors in the circuit: (1) minimum length ($L = L_{min} = 180nm$) and (2) non-minimum length ($L > L_{min}$), as shown in the table. Reset switches are considered with minimum length and the output buffers are designed to have minimum power dissipation, with same sizes in all topologies.

Table 4.4.Transistors W, L values for conventional latch DL-CS

	M1, M2	M3-M6	M7-M8	M9
Non-minimum length	$W = 1um$ $L = 250nm$	$WN = 1um$ $WP = 1.5 \cdot WN$ $LN = LP = 250nm$	$W = 1um$ $L = 180nm$	$W = 1.3um$ $L = 180nm$
Minimum length	$W = 2um$ $L = 180nm$	$WN = 1um$ $WP = 1.5 \cdot WN$ $LN = LP = 180nm$	$W = 1um$ $L = 180nm$	$W = 1.3um$ $L = 180nm$

To evaluate the different candidates, settling time and power consumption are measured and depicted in the following pictures. Results are measured for two different scenarios: differential input voltage closed to the transition (input voltage of $1mV$ as explained before) and far from it (for example, for an input voltage of $10mV$), both with common-mode input of $0.9V$. Results for the seven different topologies are shown in the following graphics. Figure 4.16 and Figure 4.17 depict the power dissipation and settling time for each of the DL architectures with non-minimum transistor length ($L = 250nm$). Likewise, Figure 4.18 and Figure 4.19 consider minimum transistor length ($L = 180nm$). Blue colour is used to represent the obtained values for an input closed to the $0V$ transition and in green are shown the results for an input far from the symmetrical transition.

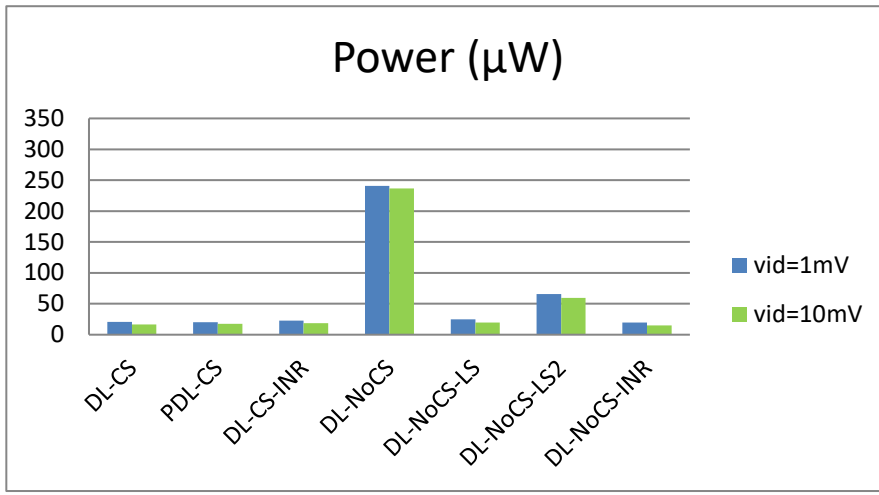

Figure 4.16. Power consumption comparison for DL with transistor length $L = 250nm$

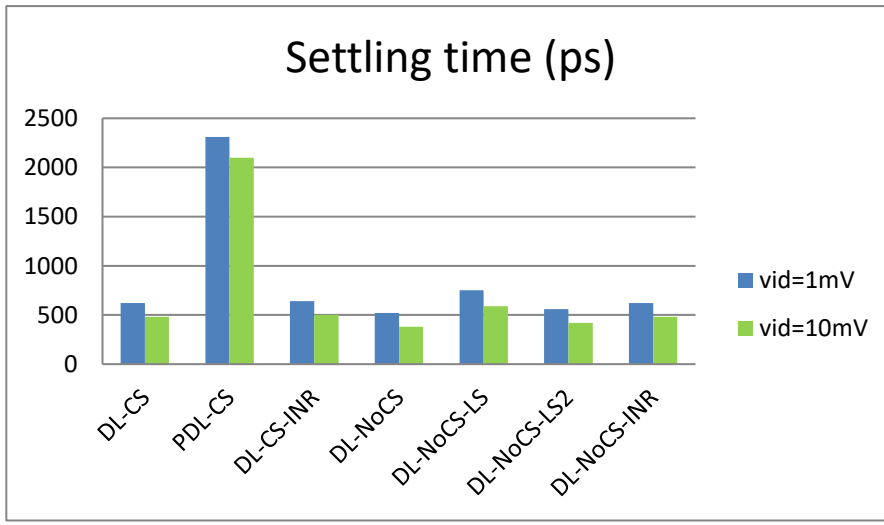

Figure 4.17. Settling time comparison for DL with transistor length $L = 250nm$

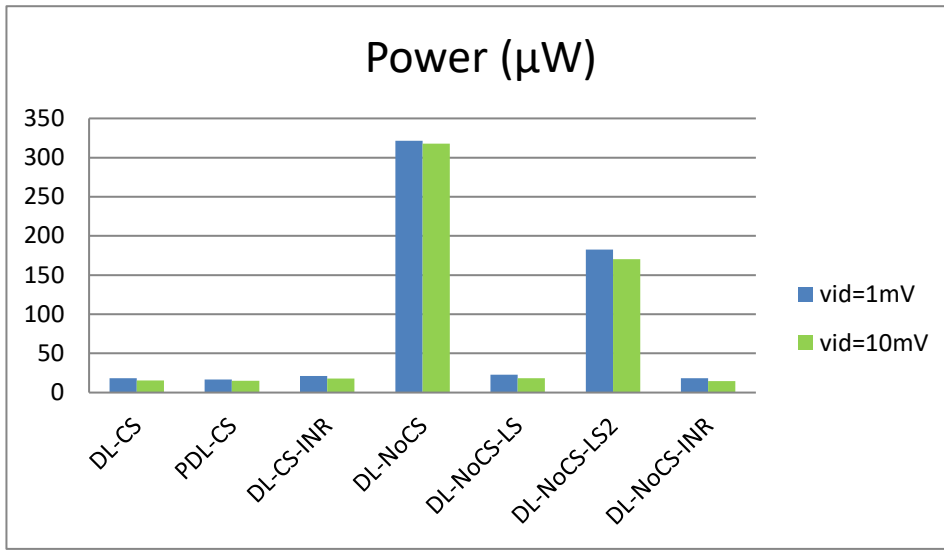

Figure 4.18. Power consumption comparison for DL with transistor length $L = 180nm$

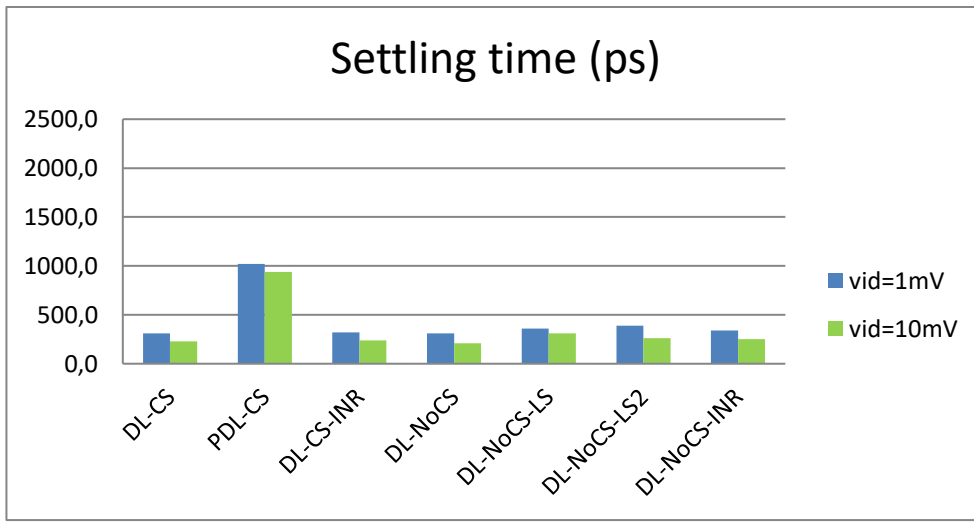

Figure 4.19. Settling time comparison for DL with transistor length $L = 180nm$

Notice that consumption and settling time decrease as input signal moves away from the $0V$ transition. At quiescent point, the dynamic latch takes longer to decide between the two levels. In addition, it is remarkable that DL-NoCS consumes by far much more power than the rest. Since its power dissipation exceeds $200\mu W$, DL-NoCS will not

employed in our design. On the other hand, it can be observed that the time the design PDL-CS takes in settling is much higher than that of the others DL topologies. This architecture is discarded as our goal is to design a high-speed flash ADC.

It can be concluded that the use of PMOS input pair in PDL-CS increases drastically the settling time of the latch. Furthermore, similar results as in conventional DL-CS are obtained for DL-CS-INR. In addition, supressing the current source from the conventional DL-CS raises the power consumption as shown for DL-NoCS. However, similar power dissipation as for DL-CS is achieved by reseting the internal nodes or opening the latch as shown for DL-NoCS-LS, DL-NoCS-LS2 and DL-NoCS-INR.

Since our major concern while designing the flash ADC are the time restriction and area, settling time results are summarized in Figure 4.20 and Table 4.5. Non-minimum transistor length (in blue) and minimum transistor length (in green) are compared for an input voltage of $1mV$.

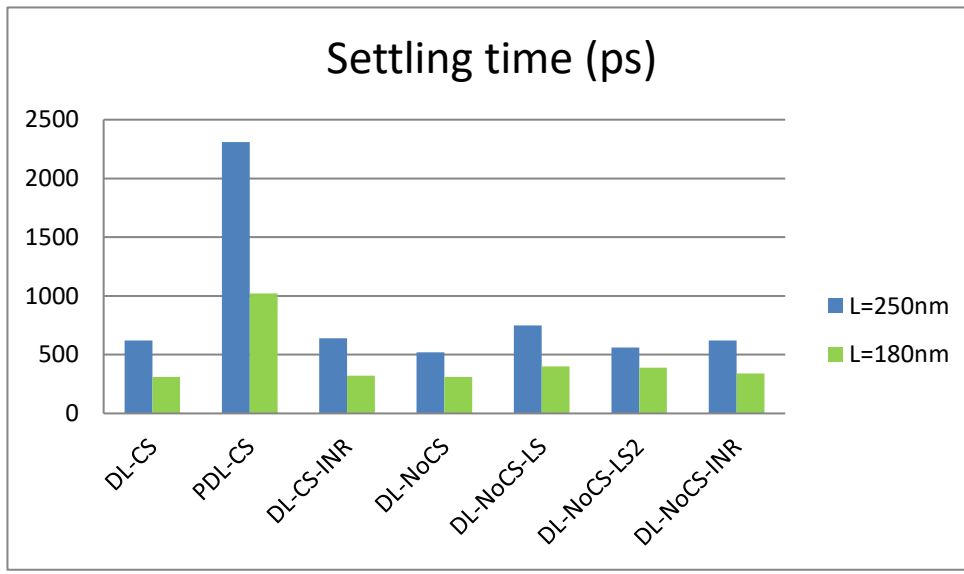

Figure 4.20. Settling time comparison when $vid = 1mV$

Table 4.5. Settling time for each DL topology

DL topology	Settling time, t_{set} (ps), for $L = 250nm$	Settling time, t_{set} (ps), for $L = 180nm$
DL-CS	620	310
PDL-CS	2310	1020
DL-CS-INR	640	320
DL-NoCS	520	310
DL-NoCS-LS	750	400
DL-NoCS-LS2	620	340
DL-NoCS-INR	560	390

Once the two circuits are eliminated from our study, the optimum circuit performance could be chosen from the rest of topologies. To do this, a figure-of-merit is defined as:

$$FoM = t_{set}/Power \tag{4–7}$$

This FoM represents the efficiency of the dynamic latch. In this FoM, settling time is more important than power consumption since the designed comparator is aimed to be used in a high-speed flash ADC. The more the comparator in the transition consumes, the less its settling time will be. Figure 4.21 depicts the new FoM for each topology when differential input is close to the $0V$ transition. It is observed that a slightly better FoM than that of DL-CS is achieved for DL-CS-INR at a cost of reseting the intermediate nodes. In addition, the suppression of the current source in DL-NoCS-LS and DL-NoCS-INR gives similar FoM results as latches with current source (DL-CS and DL-CS-INR). However, with its different topology, DL-NoCS-LS2 reaches the best performance in terms of FoM, with a much lower FoM than the rest of the DLs.

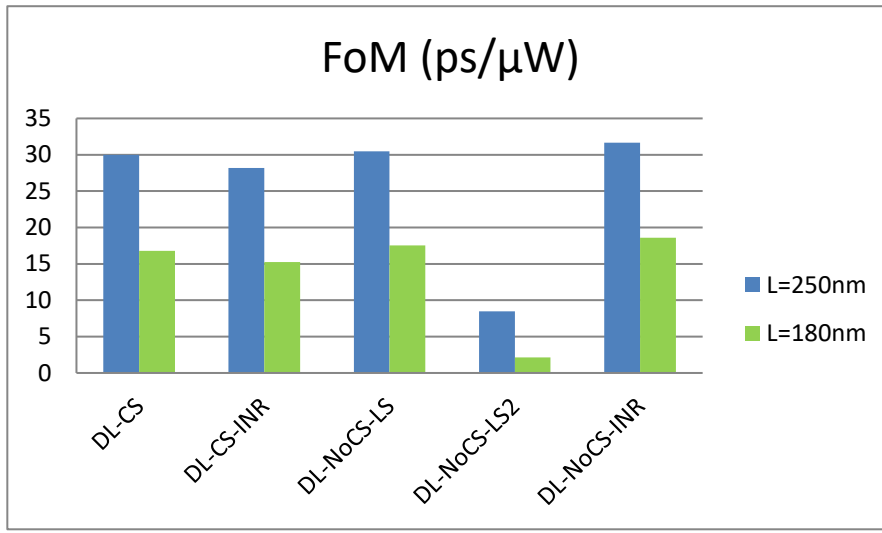

Figure 4.21. FoM comparison when $vid = 1mV$

Taking into consideration all of the above presented, circuit DL-NoCS-LS2 seems to be a good candidate due to its settling time – power trade off. In the following sections, this architecture as well as topologies DL-CS, DL-CS-INR, DL-NoCS-LS and DL-NoCS-INR are more deep studied in order to obtain the optimum design with its bank calibration. PDL-CS and DL-NoCS topologies are rejected due to their great settling time and consumption, respectively.

4.3.4. Search of the maximum resolution for the flash ADC

In the design of the flash ADC, the dynamic latch comparators are designed with built-in transitions by creating an imbalance in the front-end of each dynamic latch. Will any of the topologies described in last section be compatible with any resolution of the flash with built-in transitions? A particular DL structure with specific transistors sizes could limit the number of bits of the flash ADC: the designed latch could not be able to reach the desired extremal transition, that is, its structure limits the resolution of the flash ADC.

Extremal transitions as a function of the resolution N of the ADC are given in Table 4.1. Considering these values, in this section the maximum number of bits (N_{max}) of the flash ADC that can be reached with each of the five selected topologies is found. This

procedure is done not only for each of the DL architectures but also for both considered transistor sizes (transistors with non-minimum and minimum length).

All DL topologies are designed for a symmetrical transition $t = 0V$. In order to implement the extremal transition, an artificial mismatch is introduced in the latch by modifying the width of the input transistor pair ($M1 - M2$), thus creating an imbalance between both branches of the latch. Once the extremal transition as well as the rest of the transitions are found, the number of bits for the design of the flash ADC is determined. For instance, conventional latch DL-CS with input pair sizes of $W/L = 1\mu m/250nm$ is valid to design a flash ADC up to 4 bits of resolution. Positive extremal transition for a 4-bit design is $875mV$ and it is achieved by unbalancing the input pair sizes, one transistor (M_1) with $(\frac{W}{L}) = 0.1\mu m/250nm$ and the other (M_2) with $(\frac{W}{L}) = 1.9\mu m/250nm$. Similarly, negative extremal transition ($-875mV$) is obtained inverting the sizes of transistors M_1 y M_2. Higher flash ADC resolution is not reached since the extremal transitions cannot be implemented by just varying the input pair transistors widths.

The maximum number of bits (N_{max}) in the design of a flash ADC that can be achieved with the different architectures and transistor lengths are gathered in Table 4.6.

Table 4.6. Maximum reachable flash ADC resolution for each DL topology

DL topology	$N_{max}\ (bits)$ for $L = 250nm$	$N_{max}\ (bits)$ for $L = 180nm$
DL-CS	4	3
DL-CS-INR	4	3
DL-NoCS-LS	4	5
DL-NoCS-LS2	5	4
DL-NoCS-INR	4	4

With each of the considered DL topologies all of the transitions for an N-bit flash ADC could be implemented in a built-in way. Architectures DL-NoCS-LS and DL-NoCS-LS2 are the only two that achieve a 5-bit flash ADC resolution, that is, they are able to implement transitions closer to the ADC reference limit (i.e. closed to $R = 1V$) than the rest of the DL topologies. A 4-bit flash ADC could be designed with the rest of

topologies. Furthermore, topologies DL-CS and DL-CS-INR with transistors length of $180nm$ are limited to 3-bit flash ADCs.

4.4. Static offset estimation

Two different methods for measuring the static offset of the comparator are employed. First method is the traditional ServoLoop technique that searches for the transition by applying a known input signal with low frequency. This input signal has ramp form and covers the flash ADC FS range ($[-1V, 1V]$). As exhibits Figure 4.22, a ServoLoop algorithm generates the analogue input x which is applied to the comparator and, with a negative closed-loop configuration, the digital output b is observed. Finally, the output will be stable and the static parameters could be measured from it. Both the ServoLoop block and the comparator are latched with same clock clk. From now on, transitions with this method are measured for a pulse clock signal frequency of $150MHz$ and low input frequency of $1MHz$. Figure 4.23 shows the ServoLoop method operation. For instance, suppose the built-in transition t_{ref} in the comparator. Input signal starts from a small value ($-R$) and keep increasing its value until sensing a point where the value is greater than transition t_{ref}. At this point, the input ramp changes its slope and starts to decrease. Input decreases its value until sensing a point with a higher value than the built-in transition, where the ramp starts to increase again. This way, the ramp will finally reach the proper transition. Static offset is evaluated as the difference between the final measured transition (t) and the built-in one value (t_{ref}):

$$off = t - t_{ref} \tag{4–8}$$

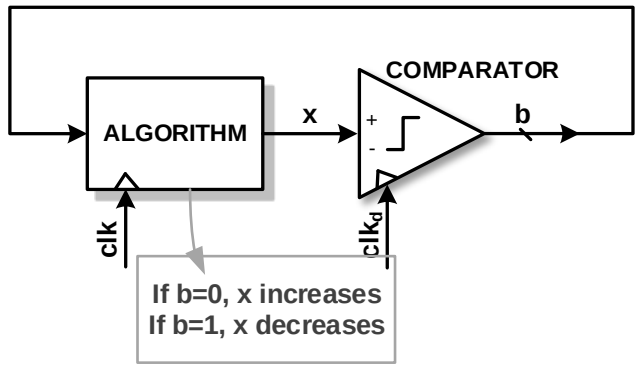

Figure 4.22. Offset measurement configuration

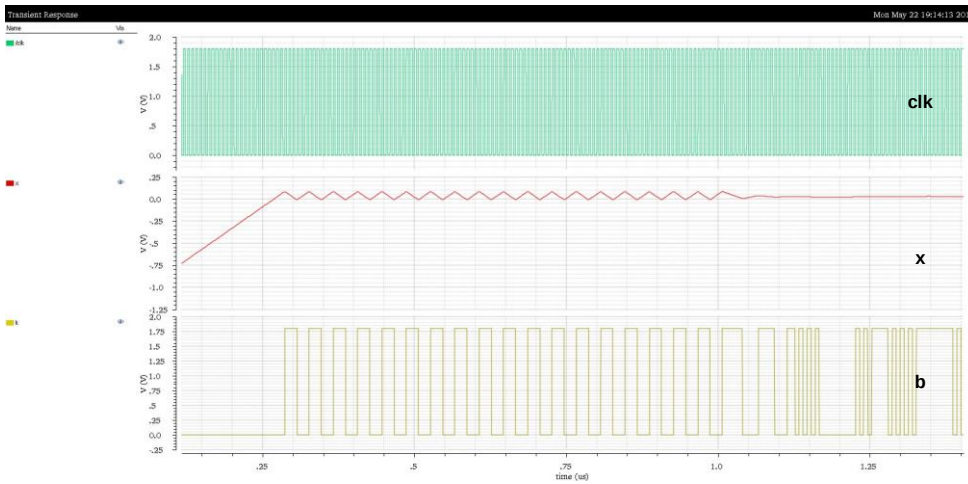

Figure 4.23. ServoLoop signals and operation

A much faster method to measure the static offset in comparators is that published in [187]. It consists of a successive-approximation (SAR) algorithm and a closed-loop configuration similar to the depicted in Figure 4.22. Its operation is shown in Figure 4.24. At the rise edge of the clock clk_d, the comparator compares the analogue input x from the SAR block with the built-in transition and gives a digital output b. When clock clk goes high, the SAR algorithm evaluates the value of the digital output. If $b = 0$, the algorithm will increased the analogue signal x in a certain value. Otherwise, x will be decreased by the same amount. After few clock cycles, the static offset could be directly obtained from the analogue input. With this SAR algorithm the real transition will be

approaching to the ideal transition value, being x after several clock cycles the value of the difference between both transitions, that is, the offset of the circuit. Its fast convergence to the transition is the reason to use this method instead of ServoLoop technique, especially in MonteCarlo simulations, where mismatch is introduced and the initial point has to be changed in ServoLoop method, which increases its convergence time.

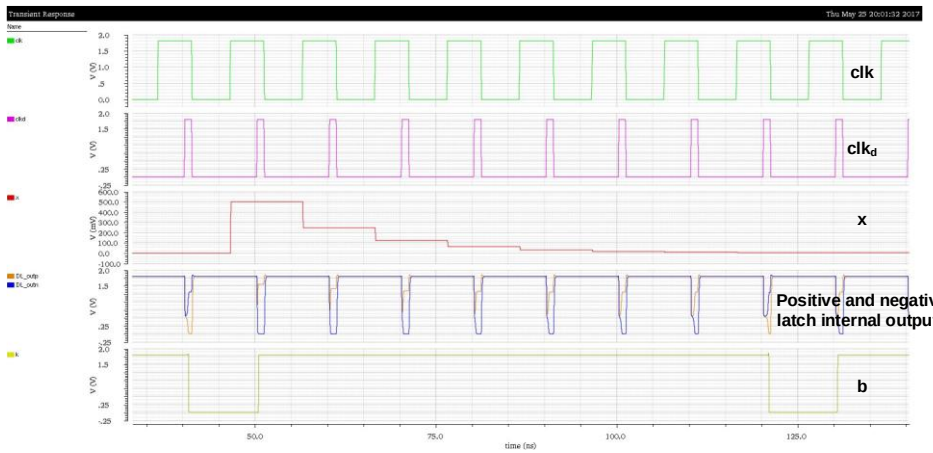

Figure 4.24. SAR-based method signals and operation

In this work, SAR-based method is employed to evaluate the offset and ServoLoop technique is used to check that measurements with SAR algorithm are correct.

4.5. Calibration bank design

As explained in Section 3.3.1, to correct the offset of the dynamic latches, a calibration DAC (see Figure 3.4) could be employed. In this section, a switchable bank of MOS resistors (SBMOS) as that shown in Figure 3.6 is used as a calibration DAC. Different bank sizes for each DL topology are evaluated, selecting the optimal bank for each DL architecture.

Furthermore, offset of the DL topology with its calibration DAC is measured with SAR-based closed-loop method and examined with ServoLoop method, both described in Section 4.4. Also, offset correctable range covered by the calibration DAC is calculated and MonteCarlo offset results for each DL topology are given.

4.5.1. Selected calibration bank: Switchable bank of MOS resistors (SBMOS)

Among the different calibration DAC structures presented in Section 3.3.1, the resistor switchable MOS transistor bank, SBMOS (in Figure 3.6) is the DAC chosen for the calibration purpose. Figure 4.25 depicts the positive branch of a SBMOS DAC with 3 bits of resolution (r). The r-bit calibration DAC is controlled by a signed r-bit control bus $C = \{-2^r + 1, \dots, 0, 1, \dots, 2^r - 1\}$ or, equivalently, $ctrlp[(r-1):0], ctrln[(r-1):0]$, being $ctrlp$ and $ctrln$ the controls for SBMOS in positive and negative branches, respectively. Since the considered bank has resolution $r = 3$, possible control codes range from $C_{min} = -2^r + 1 = -7$ to $C_{max} = 2^r - 1 = 7$. Transistors in DAC are binary weighted as depicted in the figure because of its easy language to program the calibration algorithm and since non-binary weighted transistors do not cover all possible control words. Transistor in SBMOS bank have been assumed to have minimum width ($W = 240nm$) in order to reduce the area and increase the speed of the comparator. Also, minimum width in calibration DAC increases the mismatch in comparator, however, this is not a concern since DL offset is going to be calibrated. Different DACs are designed depending on the transistors length value ($L = luc$).

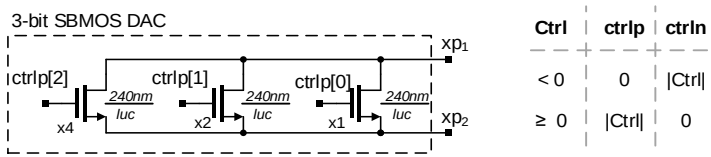

Figure 4.25. Positive branch of a 3-bit SBMOS DAC

An N-bit flash ADC has $(2^N - 1)$ transitions. Same DAC topology is used for each of the $(2^N - 1)$ comparators, but different sizes of DAC could be employed in each comparator. Although one different DAC could be used to cover the offset range of each comparator, an only DAC has been employed to cover all the N comparators for simplicity.

4.5.2. Considered assumptions in the design

Before designing the calibration SBMOS bank, some considerations have to be taken into account. Figure 4.26a depicts the probability density distribution of an effective transition of a comparator due to fabrication process variations. Nominal transition is

t_{nom} and the effective transition for a possible implementation in a comparator is t_{act}. In Figure 4.26b the locations of the output of a r-bit SBMOS bank are shown and Figure 4.26c exhibits the probability density distribution of the transition of a comparator after calibration with the SBMOS bank.

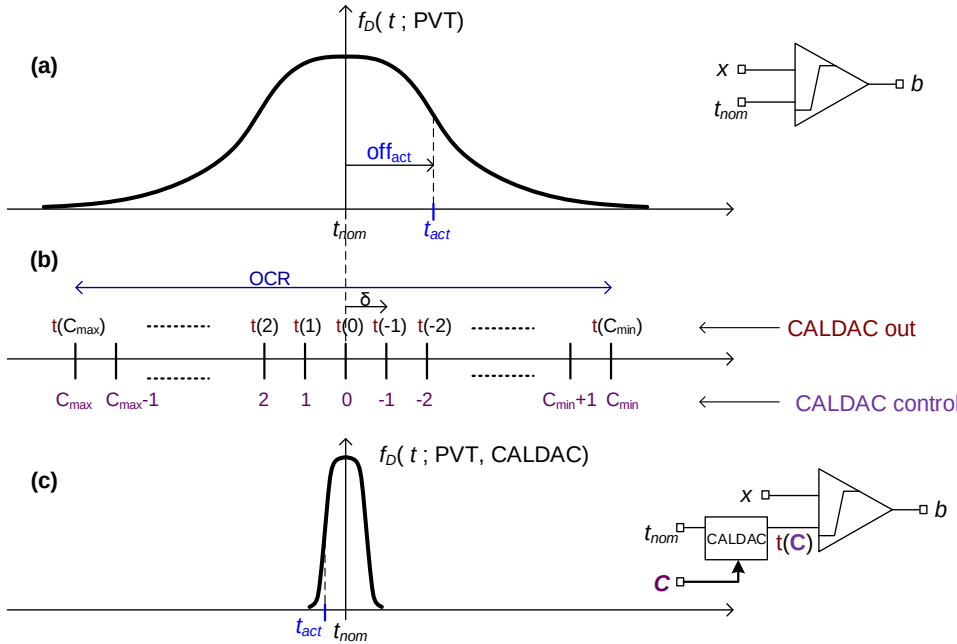

Figure 4.26. a) Probability density distribution of transition t_{nom}. b) Control codes and output of an r-bit SBMOS bank. c) Probability density distribution of transition of the calibrated comparator.

From equation (3-6), the transition of a comparator with its SBMOS bank should implement an approximately linear response with the calibration code input:

$$t(C) \approx t_{nom} - \delta \cdot C \qquad (4\text{–}9)$$

being $C \in [-|C_{min}|, C_{max}]$. For instance, in order to compensate the effective transition t_{act} (depicted in Figure 4.26a) in a real implementation, the calibration bank should have an input of $C = -2$ or $C = -3$ for the best correction.

In our design, the correctable offset range (OCR) is the complete range reachable by the transitions with the extremal control codes and it is defined as:

$$OCR = t(C_{min}) - t(C_{max}) \tag{4–10}$$

The OCR of the SBMOS bank should be great enough to cover the effective range of the density distribution of the effective transition, depicted in Figure 4.26a. The maximum correctable offset, $offC$, is calculated as the half of the OCR value:

$$offC = \frac{OCR}{2} = \frac{t(C_{min}) - t(C_{max})}{2} \tag{4–11}$$

In general, for a flash ADC, the PVT distributions of the effective transitions of the comparators could be overlapped as shown in Figure 4.27. The graphic shows 6 different nominal transitions $\{t_{k,nom}\}$ and their effective transitions $\{t_{k,act}\}$ in a possible implementation in a real flash ADC.

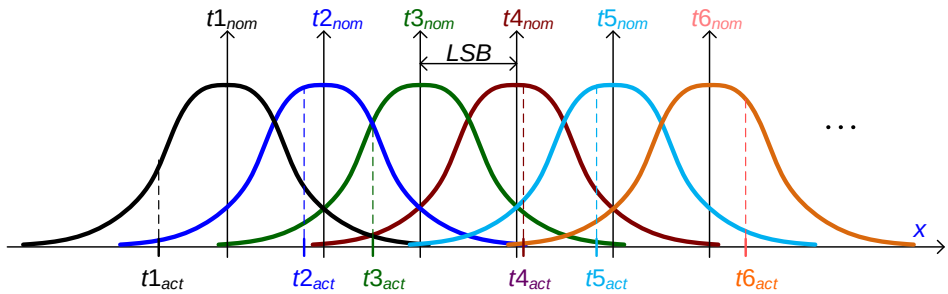

Figure 4.27. Probability density distribution of effective transitions of comparators in a flash ADC

Figure 4.29 exhibits the relation between the offset and control codes distributions of the SBMOS banks of two adjacent comparators with transitions $t_{k,nom}$ and $t_{k+1,nom}$.

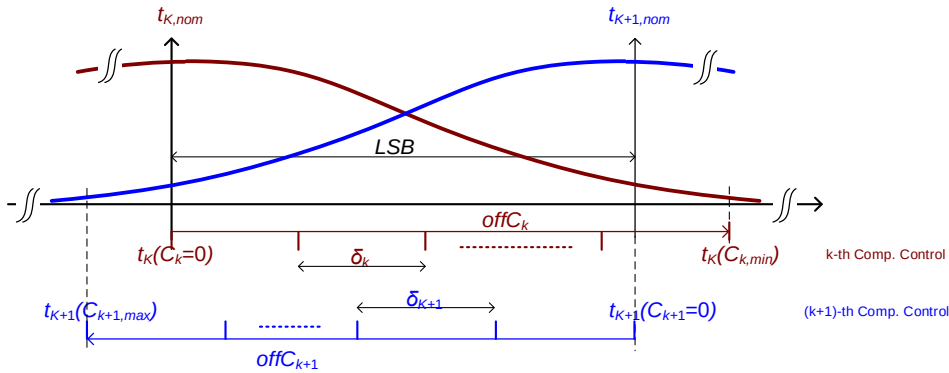

Figure 4.28. Offset and control codes distributions for two contiguous comparators

From now on, we will consider only flash ADC designs with more than 50% of overlapped in the distributions, that is, the minimum of the maximum correctable offset $offC_k$ of each SBMOS bank should be of $1LSB$ of the flash, or, equivalently, the OCR should be $2LSB$ at the least. This is the first assumption (A.1):

$$A.1: \quad offC_k \geq LSB_{flash}\,, \forall k \;\Rightarrow\; OCR_k \geq 2LSB_{flash} \tag{4–12}$$

This way, great range of non-lineality is corrected, even if inversions between contiguous transitions occur (bubble errors).

In the considered CMOS technology (180nm) for the circuit implementation, typical offset distributions for a dynamic latch comparator could exhibit a range of around 300mV. Assuming a full scale of 2 Volts in the flash ADC ($FS = 2V$), the ADC resolution should be at least of $N = 3$bits in order to have an overlap between adjacent distributions. Therefore, from now on, the considered flash designs to implement the background calibration will have at least 3 bits of resolution.

On the other hand, to achieve the enough precision in the correction of the offset, we assume that SBMOS will have a correction step, δ_k, of $LSB/2$ of the flash at the most. This is the second assumption (A.2):

$$A.2: \quad \delta_k \leq \frac{LSB_{flash}}{2}\,, \forall k \tag{4–13}$$

Considering assumptions A.1 and A.2, the resolution of the control word should be of at least 2 bits:

$$r \geq 2\ , C = \{\dots, -2, -1, 0, 1, 2, \dots\} \tag{4–14}$$

4.5.3. Bank design for symmetric transition

Once the DL and bank topologies are selected, the flash could be sized considering the two assumptions explained above. All the comparison study in Section 4.3.3 has been done for symmetrical transition $t = 0V$ since the DL candidates have been optimized for this transition. In this sub-section, one bank is designed for each of the considered DL topologies. Also, the performance of the selected DL topologies and the SBMOS bank architecture above proposed are compared. To reach this goal, the static offset is measured using the closed-loop method and later checked with ServoLoop technique, both explained in Section 4.4.

Consider a specific DL topology with transistor length L and the 3-bit SBMOS calibration DAC depicted in Figure 4.25. For sake of efficiency, the resolution of the bank is limited to 3 bits in order to avoid great calibration logic. 2-bit SBMOS bank is not considered since the covered range by the bank could be not enough to calibrate the transitions offset.

With a fixed width value of $240nm$, the length luc of the SBMOS bank transistors is adjusted until obtaining the assumed values in the section before. The minimum luc possible is chosen as the correct length for designing the bank in order to reduce the area in the comparator. Table 4.7 gathers the bank correction step (δ), maximum correctable offset ($offC$) and the correctable offset range covered by the bank (OCR) for each DL architecture (with transistor length of $L = 250nm$ as well as $L = 180nm$) and its 3-bit calibration bank SBMOS. Results show that with some of the DL architectures, it is not able to find a suitable luc value for the calibration bank. This is the case of DL-NoCS-LS and DL-NoCS-LS woth $L = 250nm$, and DL-NOCS-INR, regardless of L. In these topologies, DL and bank transistors should be modified at the same time to find a proper design.

Table 4.7. Results for symmetrical transition $t = 0V$ for each DL topology and its 3-bit SBMOS calibration DAC with LSB=250mV and FS=2V

DL topology	$L\ (nm)$	$luc\ (\mu m)$	δ $(mV/word)$	$offC$ (mV)	$OCR(mV)$
DL-CS	250	7	18.95	143.28	286.56
	180	6	25.93	138.98	277.96
DL-CS-INR	250	4.5	16.96	164.38	328.76
	180	3.5	13.93	148.08	296.16
DL-NoCS-LS	250	-			
	180	1	25.57	151	302.01
DL-NoCS-LS2	250	-			
	180	1	25.57	151	302
DL-NoCS-INR	250	-			
	180	-			

Figure 4.29 shows the nominal transition (t_{nom}) distribution of the Monte Carlo analysis for the symmetrical built-in transition $t_{ref} = 0V$ for topology DL-CS with $L = 250nm$ and its corresponding 3-bit SBMOS bank at digital control word $C = 0$. The MonteCarlo transient analysis has been performed for mismatch and process variations with 100 iterations. Since the considered built-in transition is the symmetrical $t_{ref} = 0V$, the current offset of the circuit corresponds to the measured transition ($off = t_{nom} - t_{ref} = \overline{t_{nom}} - t_{ref} = \overline{t_{nom}}$). The simulation shows that the nominal offset voltage of this structure is $-488.3\mu V$ and the maximum variation of the transition is $22.95mV$, value given for three sigma variation.

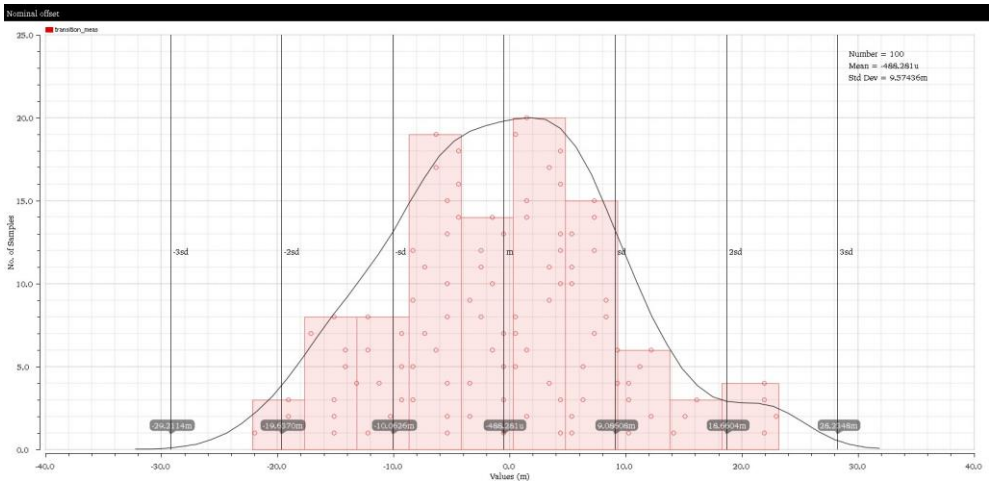

Figure 4.29. Nominal offset MonteCarlo distribution for mismatch and process variations for architecture DL-CS with 3-bit SBMOS calibration DAC for code $C = 0$

Same procedure can be accomplished for every DL candidate and their corresponding banks. Mismatch and process variation results are gathered in Table 4.8. Table gathers the length L of the DL transistors, the transition mean $\overline{t_{nom}}$, standard deviation σ_{tnom} and maximum offset off_{MC}^{max} which is obtained from:

$$off_{MC}^{max} = \max_{MC}\{t_{nom}\} \qquad (4\text{–}15)$$

In this chapter, $t_{ref} = 0V$. The design should cover not only the maximum offset off_{MC}^{max}, but also a safety margin to assure that if there is any PVT variations that affect the transitions, the bank will cover enough range to correct them. Thus, if this maximum offset is greater than half the correctable range ($OCR/2$), the bank will not be suitable for calibration the offset in the circuit and a new design of the bank or a DL different topology will be required. Otherwise, the bank will fit the specifications and design will be finished.

Table 4.8. Nominal transition measurement results with MonteCarlo mismatch and process analysis for every DL topology with 3-bit SBMOS bank and null control code

DL topology	$L\ (nm)$	Transition mean, $\overline{t_{nom}}\ (V)$	Transition standard deviation, $\sigma_{tnom}\ (V)$	Maximum offset, $off_{MC}^{max}(V)$
DL-CS	250	-488.3u	9.526m	22.95m
	180	-449.2u	7.991m	19.04m
DL-CS-INR	250	869.1u	12.25m	-28.81m
	180	908.2u	10.36m	24.9m
DL-NoCS-LS	250	-		
	180	-7.793m	54.65m	134.3m
DL-NoCS-LS2	250	-		
	180	4.199m	25.88m	83.5m
DL-NoCS-INR	250	-		
	180	-		

In the table above, some cases are left blank, as in Table 4.7, due to their incompatibility with the 3-bit SMOB bank. Observe that DL-CS and DL-CS-INR have a relatively low maximum offset, even considering mismatch and process variation. The rest of structures (DL-NoCS-LS and DL-NoCS-LS2) presents higher maximum offset (134.3mV and 83.5mV, respectively). However, such great offsets obtained in MonteCarlo simulations are not a concern since calibration banks for these two latches are able to correct an offset up to 300mV, as shown in Table 4.7.

Finally, in order to prove that calibration works, good candidates are DL-NoCS-LS and DL-NoCS-LS2 both with transistors of $L = 180nm$, since they exhibit a great offset and their correction will be rather significant.

4.5.4. Results for extremal transition

Same bank is employed in each of the comparators of the flash ADC, that is, no variation in bank transistors sizes are done from one comparator to another. Thus, it is important to know how the bank works for the extremal transition and if the different DL

architectures fail at extremal transition. Results for extremal transitions are shown in Table 4.9. The table gathers the DL topology with its transistors length L, maximum flash ADC resolution N for which the considered structure is valid (according to the parameters shown in Table 4.1), selected transistors length luc of the bank, calibration step δ, maximum correctable offset $offC$ and correctable range OCR.

For instance, DL-NoCS-LS2 with transistor lengths of $180nm$ could be employed to design a flash ADC up to 4 bits, being $875mV$ the extremal transition. For symmetrical transition $t = 0V$, a bank with $luc \geq 1\mu m$ fulfills the assumptions A.1 and A.2 (equations (4-12) and (4-13)). Therefore, $luc = 1\mu m$ is chosen to design the bank, since its area is smaller. This bank covers a range of $OCR = 791.6mV$.

Table 4.9. Results for extremal transition @FS=2V

DL topology	$L\ (nm)$	N $(bits)$	Extremal transition (mV)	luc (μm)	δ $(mV/word)$	$offC$ (mV)	Able to implement?
DL-CS	250	4	875	7	106.9	395.8	No. A.2 not fulfilled.
	180	3	750	6	49.3	292.7	Yes
DL-CS-INR	250	4	875	4.5	137.4	431	No. A.2 not fulfilled.
	180	3	750	3.5	40.8	379.8	Yes
DL-NoCS-LS	250	4	875	-			
	180	5	937.5	1	92.3	973.8	No. A.2 not fulfilled.
DL-NoCS-LS2	250	5	937.5	-			
	180	4	875	1	62.1	277.4	Yes
DL-NoCS-INR	250	4	875	-			
	180	4	875	-			

Sometimes the extremal transition could not be implemented not only because of the bank topology and transistors sizes, but also due to selected topology of the DL. The table above shows that all the structures could implement their extremal transitions. Topologies DL-NoCS-LS, DL-NoCS-LS2 and DL-NoCS-INR with $L = 250nm$ are not a

good choice for this kind of bank structure since it is not possible to find transistor sizes that satisfy the assumptions of Section 4.5.2, even modifying bank transistors widths. These circuits presents a step δ greater than $LSB_{flash}/2$ and to avoid this, a finer control digital bank (that is, $r > 2$) could be used to reduce the step. The use of a higher bank resolution could be better for the calibration of the offset but also will be less visual. In addition, another type of bank, such as those described in Section 3.3.1, could be studied for these DL architectures.

Moreover, DL-CS needs a calibration bank whose transistors lengths are greater than that of the rest of the DL topologies. DL-NoCS-LS and DL-NoCS-LS2 have both $luc = 1\mu m$ when DL transistors have length $L = 180nm$, however, the former do not accomplish both assumptions A.1 and A.2 at the same time. Finally, DL-NoCS-LS2 with $L = 180nm$ is the circuit with a lower δ and, hence, lower OCR. Therefore, this circuit will be better for correcting the transitions when those are close to each other.

5. INTEGRATED DEMONSTRATOR

In this thesis the design and tape out in UMC's 180nm technology of the prototype called CALFLASH has been fulfilled. The aim of this design is to prove the obtaining of high accuracy in a low resolution, high-frequency flash ADC with no references through background calibration. To achieve this, the idea of [105] has been implemented. In this paper a low-cost digital technique for background calibration of comparator offsets in flash ADCs is presented. This type of calibration allows relax comparator design requirements and comparator optimization for low-power high-speed applications. Comparator offset errors above half the least-significant bit (LSB) margin are admissible. Since the one of the goals of this thesis is to prove the proper functioning of the background self-calibration technique, a low-resolution reference-less 3-bit flash ADC with a simple structure and reduced area has been considered. The implemented flash ADC has 3-bit resolution and its transitions are implemented by a forcing mismatch in each comparator input front-end.

In order to calibrate the offset of the low-resolution flash ADC, a slower and more accurate auxiliary ADC is used, as depicted in Figure 3.13. The chosen auxiliary ADC is a 6-bit SAR type.

In the following subsections, chip architecture is elucidated and each of its blocks is explained in detail.

5.1. ASIC prototype architecture

A simplified block diagram of the prototype called CALFLASH is depicted in Figure 5.1. Every sub-circuit and the most significant signals are exhibited. The low-resolution ADC under calibration is a 3-bit flash ADC (in blue colour) and a slower 6-bit SAR ADC (in red colour) is chosen as auxiliary ADC to calibrate the offset of the main flash Same analogue input voltage is employed in both flash and SAR ADCs, as shown in Figure 3.13. The other essential block in the circuit is the calibration block COCL, which accomplishes the digital adaptive algorithm for offset calibration of the flash ADC.

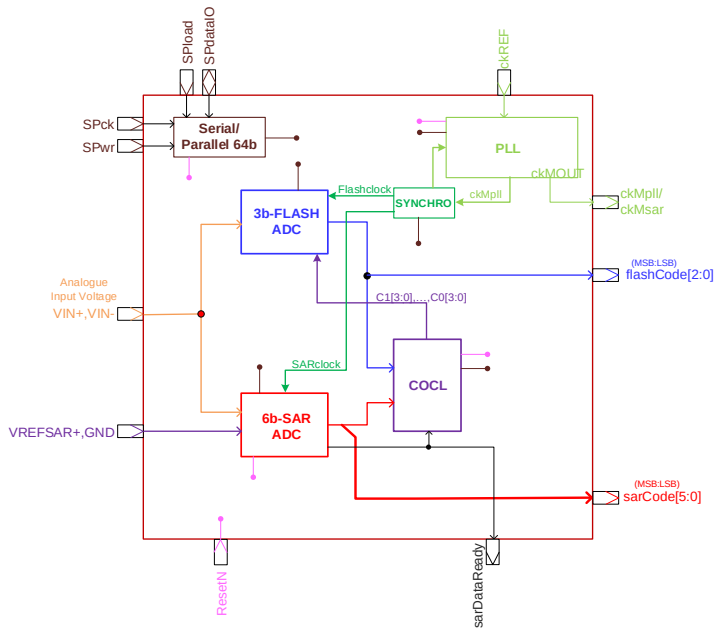

Figure 5.1. Simplified block and signalling diagram of the chip

Further, there are also a number of auxiliary blocks in the circuit:

- A phase locked loop (PLL) generates the desired master clock frequency ($fckMpll$). Since the reference clock ($ckREF$) of the circuit is single-ended, its practical frequency is limited to a relative low value and the internal blocks could work with high frequency thanks to the use of the PLL.
- A synchronisation block (SYNCHRO) which helps synchronising both ADCs sampling clocks. This block generates two synchronised clocks with the master

clock ($ckMpll$), one for the flash ADC ($Flashclock$) and the second clock for the SAR ADC ($SARclock$). This way, both ADCs capture the same input voltage. See chronogram in Figure 5.3 for a better understanding of the synchronisation of both ADCs clocks.

- A 64-bit serial/parallel interface (SP64b) to inject programmability in all blocks. It is used for configuring and transmitting the data from/to the circuit. This block transforms a serial digital input data into a parallel bus. The different configurations of the circuit are introduced in the circuit via series. Also, with this block every calibration control code could be read. SP64b has 64 internal output signals and 28 internal input signals (control codes). Since it has many internal signals (64+28), its routing is complicated. The 28 input signals are the external control codes for the 7 comparators (7x4=28) and these COCL calibration codes can be read in series on demand. All signals without a specific pad are outputted through this block. SP64b is in charge of writing/reading configuration signals, like parameters to configure the clock system (PLL and SYNCHRO blocks), to program the COCL block, to configure flash and SAR ADCs and to inject offset in flash ADC. To do this, SP64b block is controlled by the digital serial-parallel signals $SPwr$ (write), $SPdataIO$ (data input/output), $SPload$ (load), $SPck$ (clock). With this block, both the number of required pins and the package size are decreased. In case the number of bits of the flash ADC increases, the Ctrl code number of bits will highly rise and, hence, more routing from COCL block to flash ADC will be required. This way, the use of a serial-to-parallel circuit is highly recommended.

Designs of the above major functional blocks are detailed in the following sections.

5.2. Circuit operation

A simplified ADCs clock signals scheme is shown in Figure 5.2 and the most important signals and operation of the circuit CALFLASH are depicted in Figure 5.3. In a normal calibration mode, from the master clock ($ckMpll$) the SYNCHRO block generates a flash clock ($Flashclock$) which is a delayed version of the master clock ($ckMpll$) and a flash code is obtained in every flash clock interval. In addition, from the master clock ($ckMpll$), a SAR clock signal ($SARclock$) is generated as a divided version of clock

Flashclock. Every 8 SAR clock cycles, one SAR code is outputted. However, every flash clock cycle, flash outputs its code. By implementation, minimum delay between both ADCs clock falling edges has been searched. As shown in Figure 5.3, among all of the possible samples, only a few of them are employed in calibration.

SYNCHRO block purpose is to coordinate the flash and SAR ADCs sampling instants and synchronise both output codes that reach the COCL. COCL block is triggered by a divided clock from SAR ADC (*Dataready* signal), and flash and SAR codes are captured respectively in falling and rising edges of *Dataready*.

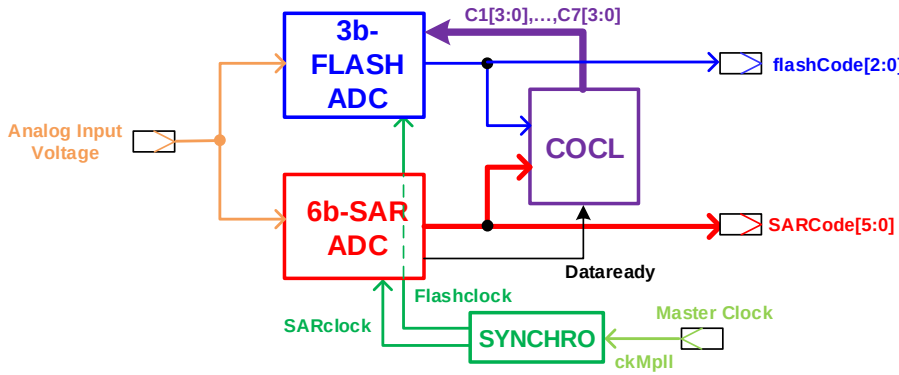

Figure 5.2. Simplified block diagram of CALFLASH

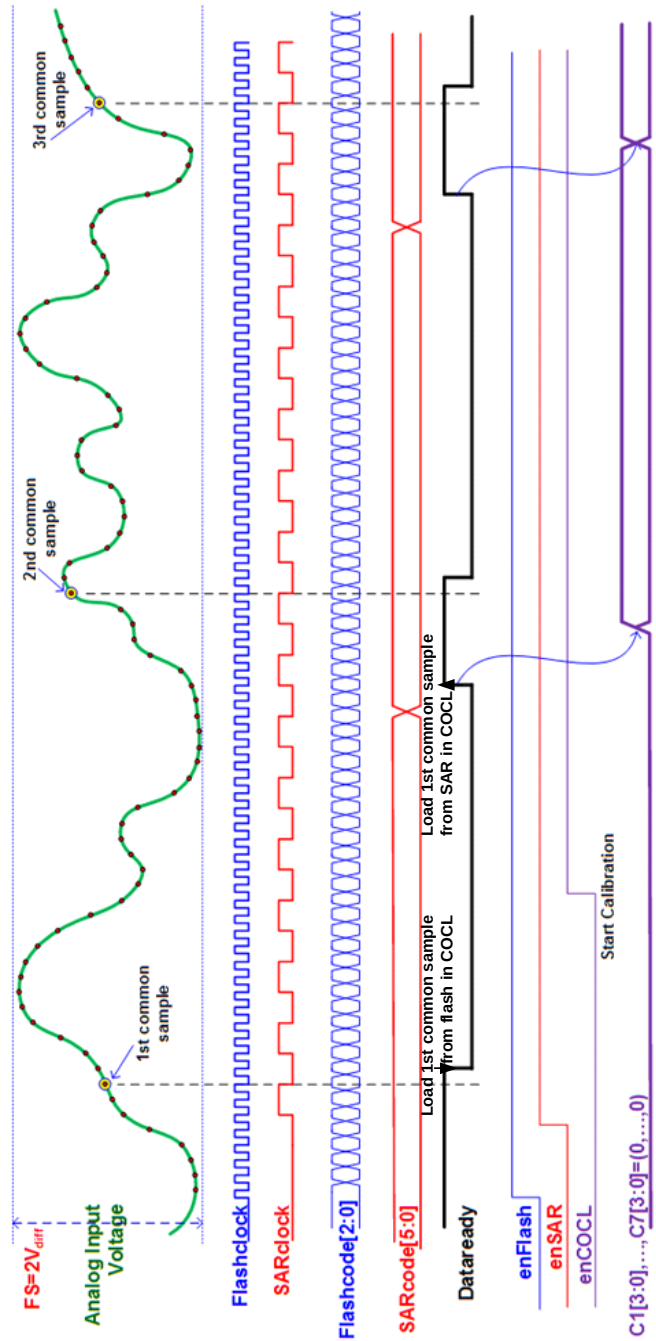

Figure 5.3. Chronogram of CALFLASH

5.3. Flash and SAR ADCs Synchronisation

Synchronisation between both ADCs in the circuit is one of the most crucial issues in this work. When flash and SAR ADCs are working concurrently, both ADCs must capture the same clock signal. Sampling must be fast and precise. Without a good synchronisation between flash and SAR ADCs clocks, different analogue input values will be captured by the two ADCs and a correct calibration will not be possible.

Since one of the aims of this thesis is to prove the proper performance of the calibration and as, in order to simplify, no common SH is employed, flash and SAR sampling frequencies must be well synchronised. To do this, some delays are introduced in the clock signals, as explained in the following sub-section.

5.3.1. Clock system

The clock system of the circuit is based on a phase-locked loop circuit (PLL) and a synchronisation block (SYNCHRO). Clock signals and the block diagram of generation of the signals are depicted in Figure 5.4.

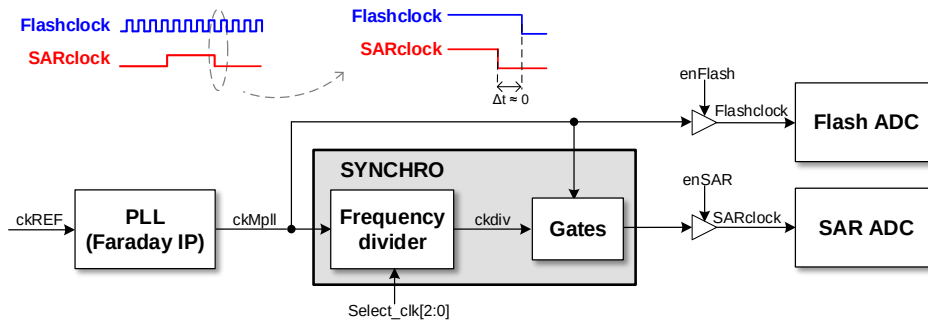

Figure 5.4. Clock signals schema and block diagram of their generation

The employed PLL block is an IP from Faraday Technology. Although taking up vast area of the chip, the differential LBDS clock receiver provided by the technology exhibit greater area that the PLL, and this is the reason for choosing the PLL. Furthermore, the chosen PLL has a great versatility and, thanks to its programmability, it is able to work at different frequencies. Its clock input $ckREF$ goes from $5MHz$ to $300MHz$, giving an

output $ckMpll$ that reaches values in the range of $[20{,}300]MHz$.

When flash and SAR ADCs are working concurrently, both ADCs must capture the same clock signal. SYNCHRO block is used to synchronise the clocks of both ADCs, $Flashclock$ and $SARclock$, with the master clock $ckMpll$. As explained in Section 5.1, the former ADC clock is a delayed version of $ckMpll$ and the latter is a divided version of $Flashclock$. By implementation, minimum delay between falling edges of both ADCs clocks has been search when designing SYNCHRO block. This semicustom design has been done using Faraday gates. It is composed of a Frequency divider and some gates which allow synchronise both ADCs clocks at their falling edges. The frequency divider is employed to divide the SAR ADC clock by a factor $freqdiv = \left\{\frac{1}{2}, \frac{1}{4}, \dots, \frac{1}{32}\right\}$, depending on the value of the signal $Select\_clk[2:0]$.

5.3.2. Sampling precision

In calibration mode, flash and SAR ADCs are working together and both ADCs must capture the same clock signal. Hence, sampling must be fast and precise. In this subsection, the precision of the sampling signal is to be estimated. In other words, a theoretical calculation has been carried out to evaluate how adequate the synchronisation between $ckMflash$ and $ckMsar$ is.

Consider a sinusoidal input signal

$$V_{in} = Acos(\omega t + \varphi) \tag{5–1}$$

Speed is given by the slope of the sinusoidal input V_{in}, that is:

$$\frac{dV_{in}}{dt} = -A\omega sin(\omega t + \varphi) \tag{5–2}$$

Maximum speed is

$$\left|\frac{dV_{in}}{dt}\right|_{max} = +A\omega \tag{5–3}$$

And it occurs when signal crosses 0V, changing its sign.

From last expression, it can be witten that

$$\Delta t_{max} \approx \frac{\Delta V_{in,max}}{A\omega} \tag{5–4}$$

which represents the maximum delay that should exist between flash and SAR ADCs clocks to have a maximum difference (ΔVin_{max}) between ADC samples. Figure 5.5 shows this case when a delay between sampling clocks of two ADCs ($clk1$ and $clk2$) exists.

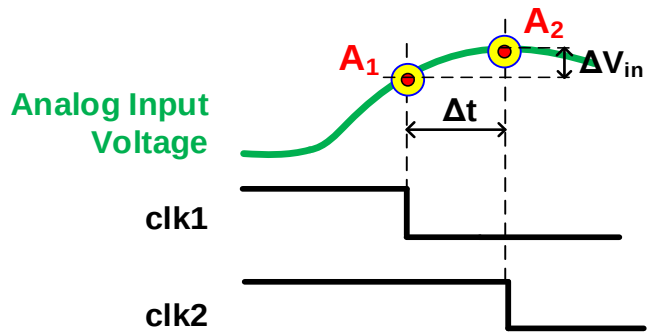

Figure 5.5. Delay between sampling clocks of two ADCs

In order to have a great synchronisation between both ADCs clocks, sampling errors between both ADCs (ΔV_{in}) should not be greater than 1 LSB of the flash ADC:

$$\Delta V_{in} \leq LSB_{flash} \tag{5–5}$$

Imposing a maximum error of $\Delta V_{in,max} = \frac{LSB_{flash}}{4}$ and with $LSB_{flash} = \frac{FS}{2^N}$, for the considered N-bit flash ADC, the maximum delay between clocks could be calculated from equation (5-4):

$$\Delta t_{max} = \frac{\Delta V_{in,max}}{A\omega} = \frac{\frac{FS}{4 \cdot 2^N}}{\frac{FS}{2} \cdot \omega} = \frac{1}{2^{N+1}\,\omega} \tag{5–6}$$

where a signal at full-scale is used: $A = FS/2$. For a maximum error of $\frac{LSB_{flash}}{4}$, the maximum delays between clock signals in a 3-bit ADC are shown in Table 5.1.

Table 5.1. Delays between clock signals @$\Delta V_{in,max} = LSB_{flash/4}$

Input frequency (f_{in}) (MHz)	Δt_{max} (ps)
20	497.4
100	99.47
150	66.3

Comparing to results obtained at schematic level simulations with the considered implementation, maximum delay between both ADCs clocks is $40ps$ at $f_{ckMpll} = 300MHz$. At layout level, simulations give a maximum delay below $100ps$, which is valid to use the calibrating algorithm with an input frequency up to 100MHz.

5.4. 3-bit flash ADC design

A 3-bit flash ADC with reference $R = 1V$ and quantum step of $q = 250mV$ has been designed. It achieves a maximum working frequency of $250MHz$ in simulation. Instead of using a resistive ladder, transitions are implemented by hardware, with a forcing mismatch in each comparator input frond-end. This means that for a 3-bit flash ADC, seven different comparators with built-in transitions are designed, each comparator for a different transition voltage: $\{-750, -500, -250, 0, 250, 500, 750\}mV$. The 3-bit flash ADC is depicted in Figure 5.6. Each differential comparator senses the differential input signal to its built-in transition voltage, outputting a bit of a thermometric code $\{b[7], \dots, b[1]\}$.

In order to correct the non-ideality of each comparator, a 3-bit calibration bank controlled by a 4-bit signed control signal $C[3:0]$ has been introduced in each comparator to force a mismatch and compensate for its offset. Also, an external signed control signal C^0 coming from the SP64b block is injected in each comparator in order to introduce systematic imbalances, as explained later in Section 5.4.3. With the considered calibration bank, the achieved correction step is $\delta \approx 30mV/word$ in simulations. Further, maximum correctable offset is $offC = 240mV$.

Each comparator has an input capacitance of about $50fF$, which is constant for each comparator. And the total input capacitance of the flash is seven times the comparator capacitance.

The comparator outputs are connected to a thermometer-to-binary encoder which produces a 3-bit binary output. Due to its minimum bubble error, the selected encoder is a 3-bit Wallace tree as that depicted in Figure 4.6. Its four full-adder cells are implemented with full-adder Faraday gates. Wallace tree topology has been chosen thanks to its simple, straight-forward and fast architecture, with no need to add a bubble error correction circuit, which would increase the ADC area.

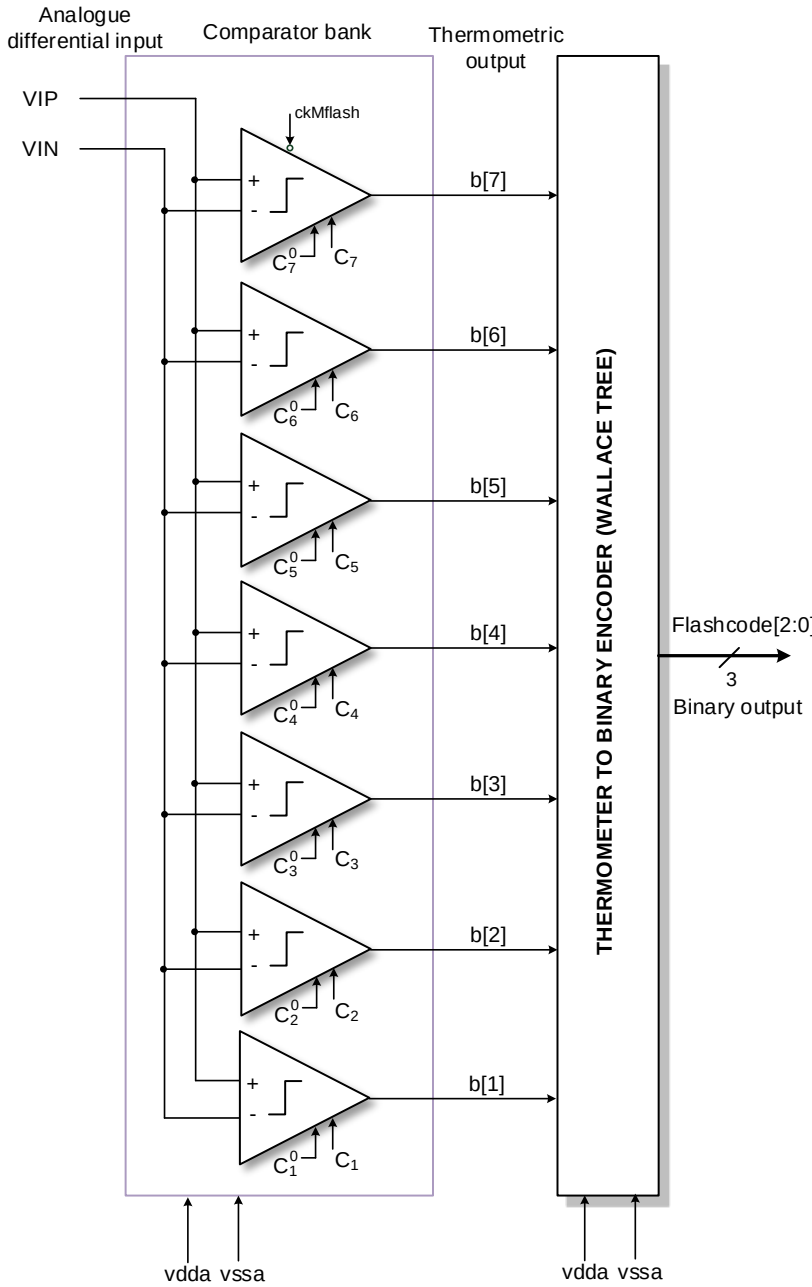

Figure 5.6. 3-bit flash ADC

5.4.1. Comparator design

Comparators exhibit differential structure. Typical topology of comparators with built-in transitions is illustrated in Figure 4.1c. Usually, a preamplifier is used in front of the SADL to improve the precision. However, in the design of the 3-bit flash ADC no preamplifier has been employed. Instead of gaining precision with excellent comparator hardware, the goal of the flash design is having some offset which will be later corrected through the implemented self-calibration technique. Therefore, the final comparator structure is shown in Figure 5.7. It is composed of a chain of DL, a RS latch and buffers. RS latch has been full-custom implemented with a typical structure with two NOR CMOS gates. DL design is explained in Section 0 and Control-Word Adder is illustrated in Section 5.4.3.

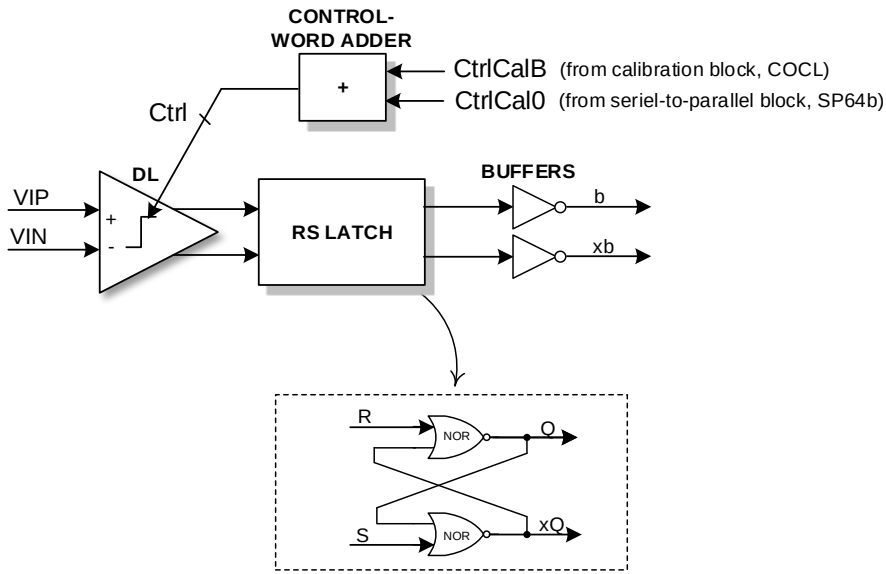

Figure 5.7. Implemented comparator architecture

5.4.2. Dynamic latch design

From DL comparison study carried out in Section 4.3.3, it has been deduced that DL-NoCS-LS2 (depicted in Figure 4.14) is the optimal architecture among the considered structure for a 3-bit flash ADC. This topology is rather sensitive to environmental conditions and mismatches, but exhibits really good advantages in terms of power consumption and speed response. Also, a (3+sign)-bit calibration bank has been added to force an intentional mismatch and, hence, correct the comparators offsets. The 3-bit SBMOS calibration DAC with transistor length of $luc = 1um$ (see Figure 4.25) happens to exhibit the best performance and satisfy the hypotheses considered in Section 4.5.2.

Having taken all the above into consideration, an optimised version of the latch has been designed for the prototype and depicted in Figure 5.8. Its sizing is gathered in Table 5.2. The DL has been implemented with minimum transistors sizes. Since the main idea of this work is to prove the self-calibration technique of the flash ADC, the DL has been implemented to be in the worst case mismatch conditions, that is, transistors must have minimum widths and lengths. In addition to this, a system to force the mismatch in the circuit has been implemented.

Note that in Figure 5.8 the applied control bus is called $ctrlp[2:0]$ and $ctrln[2:0]$. In Section 5.4.3, relationship between control bus $Ctrl$ and $ctrlp$ and $ctrln$ is detailed.

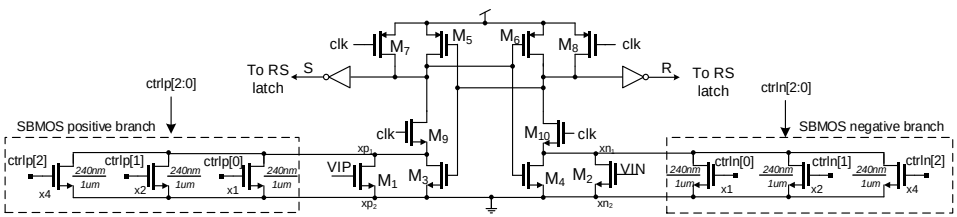

Figure 5.8 Dynamic latch with calibration bank

Table 5.2. Transistors W, L values for implemented DL in each comparator

M1, M2	M3-M6	M7-M8	M9-M10
$W_1 = m_1 w_u$	$WN = 1um$	$W = 1um$	$W = 1um$
$W_2 = m_2 w_u$	$WP = 3 \cdot WN$	$L = 180nm$	$L = 180nm$
$w_u = 700nm$	$LN = LP = 180nm$		
$m_1, m_2 \in [1,9]$			
$m_1 + m_2 = 10$			
$L = 250nm$			

Each of the transitions has been implemented with an artificial mismatch introduced by the DL MOS input sizing (M_1, M_2). Comparators differ only in MOS input widths (W_1, W_2) since the same calibration bank is employed for all comparators. Table 5.3 gathers the non-inverter input widths (W_1) and inverter input widths (W_2) for each of the comparators. Parameters W_1, W_2 are a function of a unitary width $w_u = 700nm$. The layout of the full-custom comparator 3 (corresponding to transition $-250mV$) is shown in Figure 5.9. Input transistors and calibration bank layout is shown in Figure 5.10. At the bottom left corner, 6 unitary width (w_u) transistors plus half width ($w_u/2$) transistor correspond to $6.5 \cdot w_u = 6 \cdot w_u + 0.5 \cdot w_u$. In the right, 3 w_u transistors and half width transistors corresponding to $3.5 \cdot w_u$.

Table 5.3. M1-M2 transistors widths $W_{1,2}$ for each comparator

Comparator	Ideal transition (mV)	W_1	W_2
1	-750	$9 \cdot w_u$	w_u
2	-500	$8 \cdot w_u$	$2 \cdot w_u$
3	-250	$6.5 \cdot w_u$	$3.5 \cdot w_u$
4	0	$5 \cdot w_u$	$5 \cdot w_u$
5	250	$3.5 \cdot w_u$	$6.5 \cdot w_u$
6	500	$2 \cdot w_u$	$8 \cdot w_u$
7	750	w_u	$9 \cdot w_u$

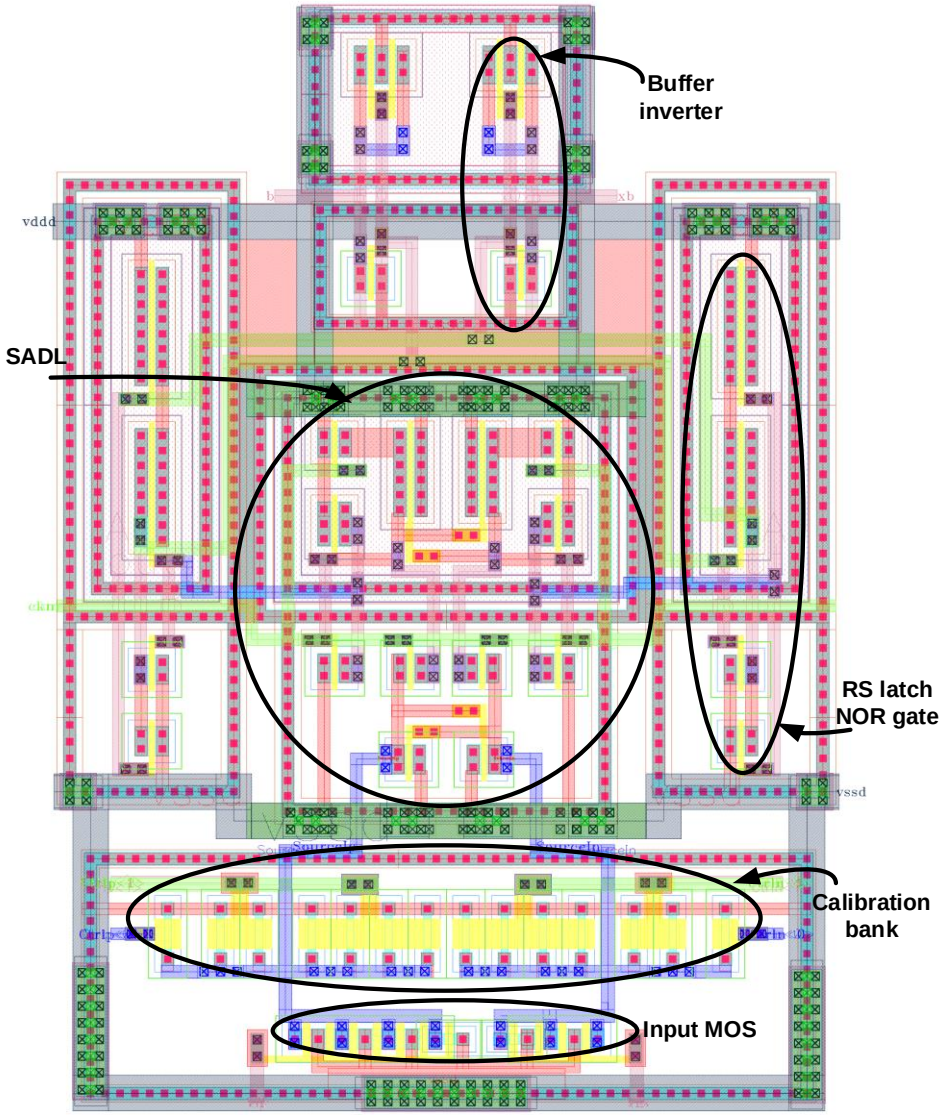

Figure 5.9. Layout of comparator for 3th transition (37µm high x 26µm wide)

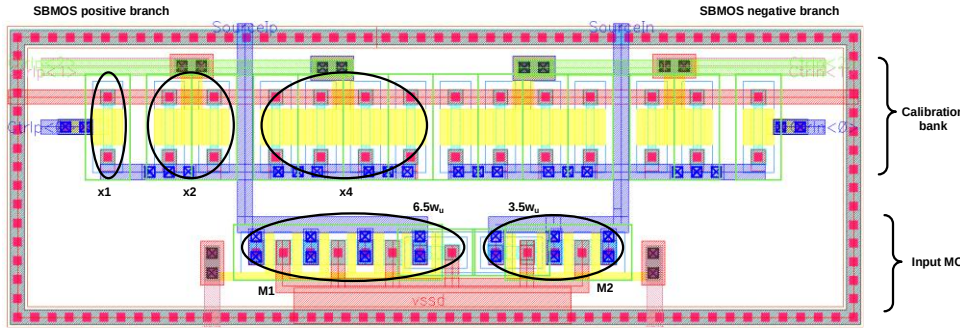

Figure 5.10. Layout details of input MOS and calibration bank

5.4.3. Control-Word Adder

Although comparators layout has been symmetric designed, tape out process adds additional mismatch. Also, no preamplifier has been used to reduce mismatch and parasitics have a great influence in this kind of dynamic latches. However, the mismatch due to the design is not enough to examine the limits of the flash design.

In order to generate an additional mismatch in the comparators and also to test the calibration limits, an external signed control signal $CtrlCal0[3:0]$ through the serial-to-parallel block has been applied to the calibration bank. This way, an external control code could be forced in the circuit, that is, an offset is injected in the circuit and the calibration block must come into play and correct the applied offset. To do this, each of the comparators adds an additional block called control-word adder, as depicted in Figure 5.11. Such a method of injecting offset in the circuit does not require extra transistors, since the same transistor bank used for calibration is now employed for offset injection.

When $CtrlCal0 = 0$ and no artificial mismatch is applied to the circuit. Then, the control-word adder is in charge of converting the control codes $CtrlCalB[3:0]$ coming from COCL block into $ctrlp[2:0]$ and $ctrln[2:0]$, which can be applied directly to the calibration bank. When an external control code $CtrlCal0$ different from 0 is applied to the circuit, an offset appears in the flash ADC and $CtrlCalB$ has to cancel it. The control-word adder block calculates the sum (named $Ctrl$ in figure) of $CtrlCal0$ and $CtrlCalB$ and obtains the $ctrlp$ and $ctrln$ signals. Table 5.4 gathers how control-word adder evaluates this $Ctrl$ signal.

Further, with control-word adder block it is possible to inject, through the SP64b block, 16 different offset values in the range of ±2 LSB in every comparator of the flash ADC, without the need of extra circuitry.

The control-word adder exhibits a vast layout as shown in Figure 5.12. The comparator has a small analogue area compared to digital control-word adder block. This block has been employed due to test reasons. However, in a definite integration, there is no need to include the control-word adder and the whole circuit area will be reduced.

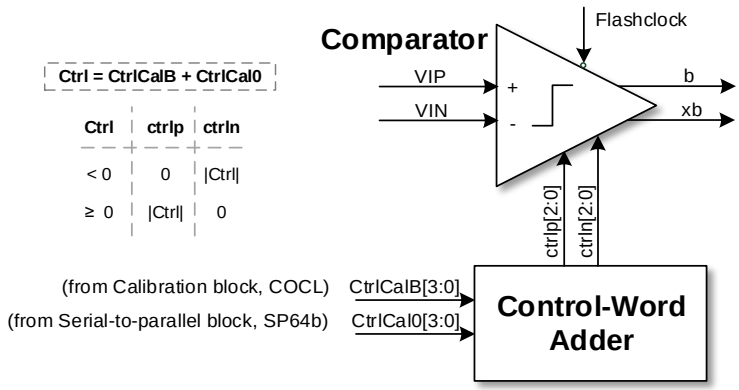

Figure 5.11. Comparator with control-word adder block

Table 5.4. Control-word adder operation

$Ctrl$	$ctrlp$	$ctrln$
< 0	0	$\|Ctrl\|$
≥ 0	$\|Ctrl\|$	0

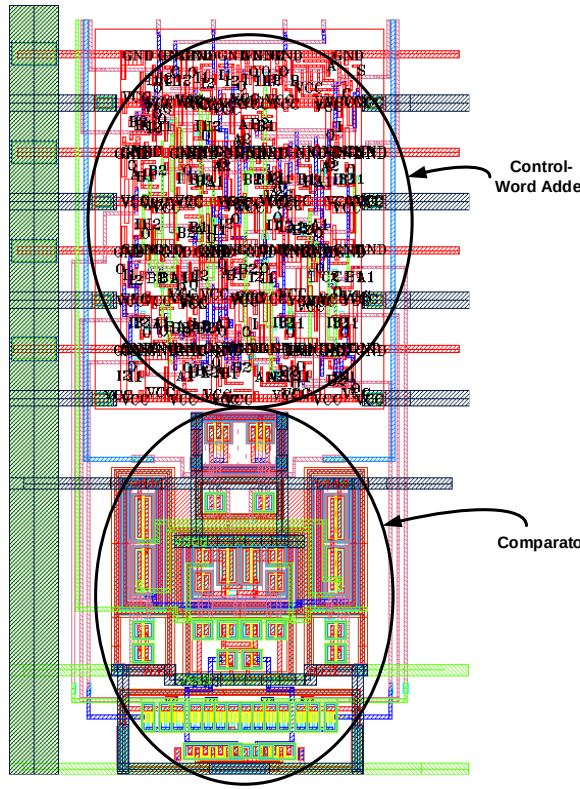

Figure 5.12. Comparator and its control-word adder layout (73μm high x 26μm wide)

5.4.4. Layout and simulations

Figure 5.13 shows the layout of the 3-bit flash ADC. Note that the analogue part (at the bottom of layout) is separated from the digital part (at the top). Comparators are not randomly located to get random mismatch. On the contrary, they have been ordered from comparator 1 (in the left) to comparator 7 (in the right). The whole flash area occupies 120μm high x 340μm wide and Wallace tree encoder takes up a small space compared to other blocks. Also, analogue comparator fills a small area in comparison to the digital control-word adder block (which forces an extremal offset in the circuit). However, since control-word adder has test purposes, once the test of the circuit is done, this block could be supressed from a forthcoming design. Since $CtrlCal0$ bits are placed in parallel, this bus occupies a great area in the layout. In addition, decoupling capacitors are situated around the comparators. They have their rings overlapped, sharing their

top and bottom plates. Also, guard rings have been used. All of the above presented greatly improves the layout. However, its layout area could be decreased in a future design, once calibration is tested.

First, post-layout simulations of the stand-alone comparators (including the control-word adder block) in open-loop are carried out. With flash virtual reference of $1V$, consider an input frequency of $1MHz$, clock frequency of $150MHz$ and parameters $CtrlCal0 = 3$ and $CtrlCalB = -3$, that is, $Ctrl = 0$. Results are shown in Table 5.5. The table gathers the following information: the number of comparator (M) and its corresponding nominal transition (that is, ideal transition t_{id}), the measured transition with schematic level simulation t_{sch}, the measured transition with no R/C parasitic extraction simulation $t_{noR/C}$, the measured transition with R+C+CC parasitic extraction simulation t_{R+C+CC} and error ε evaluated as the difference between measured transitions with R+C+CC and with no R/C parasitic extraction are shown in the table. Error ε represents how good the parasitic extracted simulation is. Note that maximum error is $11.5mV$, which is a very low offset value.

Table 5.5.Stand-alone comparators open-loop simulation results @$150MHz$

M	$t_{id}\ (mV)$	$t_{sch}\ (mV)$	$t_{noR/C}\ (mV)$	$t_{R+C+CC}\ (mV)$	$\varepsilon\ (mV)$
1	-750	-734.0	-743.6	-740.5	3.1
2	-500	-511.5	-517.0	-514.5	2.5
3	-250	-259.0	-255.0	-244.7	10.3
4	0	-0.7	0.2	-4.4	4.2
5	250	261.5	223.4	215.0	-8.4
6	500	511.0	518.7	512.5	-6.2
7	750	735.6	745.0	733.5	-11.5

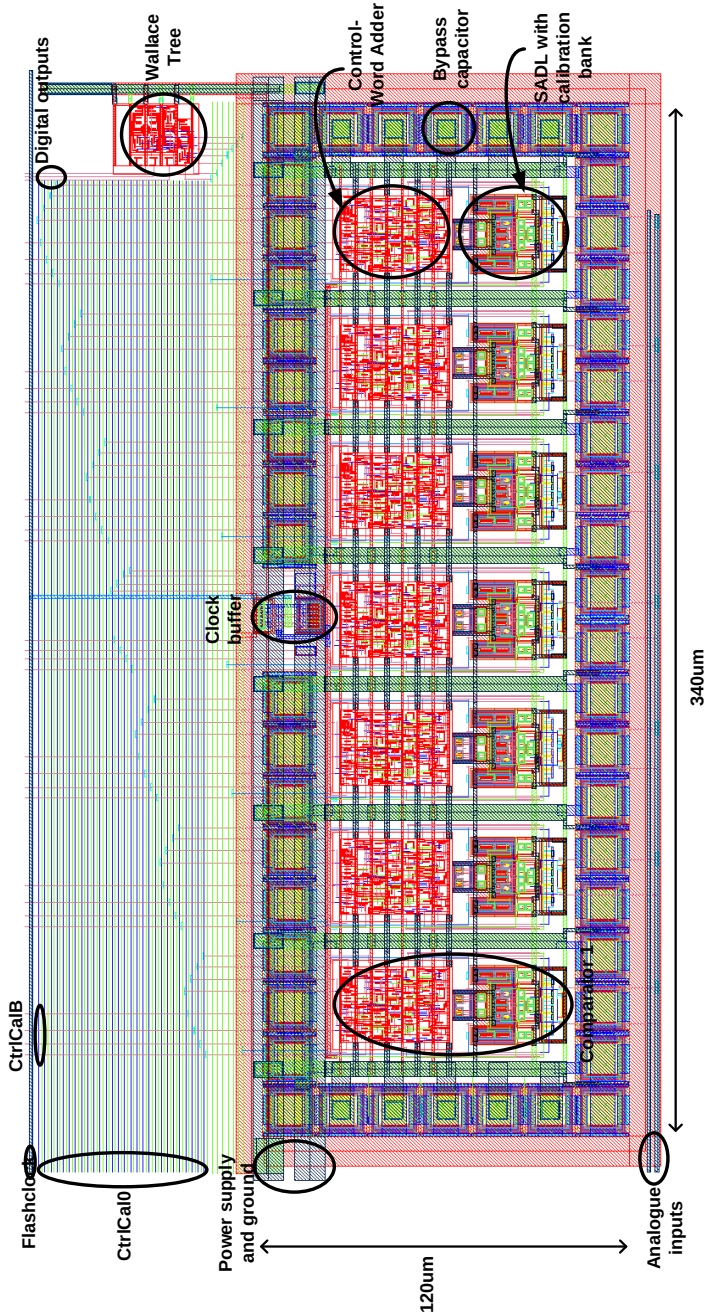

Figure 5.13. 3-bit flash ADC layout ($120\mu m$ *high x* $340\mu m$ *wide)*

Tuning curves represent the measured transition versus the control code. Each of the curves is due to a different intentional mismatch. The way to introduce this unbalance is by varying the width of the input transistors, as explained in Section 0. Figure 5.14 depicts the measured transition versus the applied external control code $CtrlCal0$. This way, the correction step and maximum correctable offset could be both evaluated with equations described in Section 4.5.2. By averaging, the correction step is $31.31mV/word$ and maximum correctable offset is $239.2mV$.

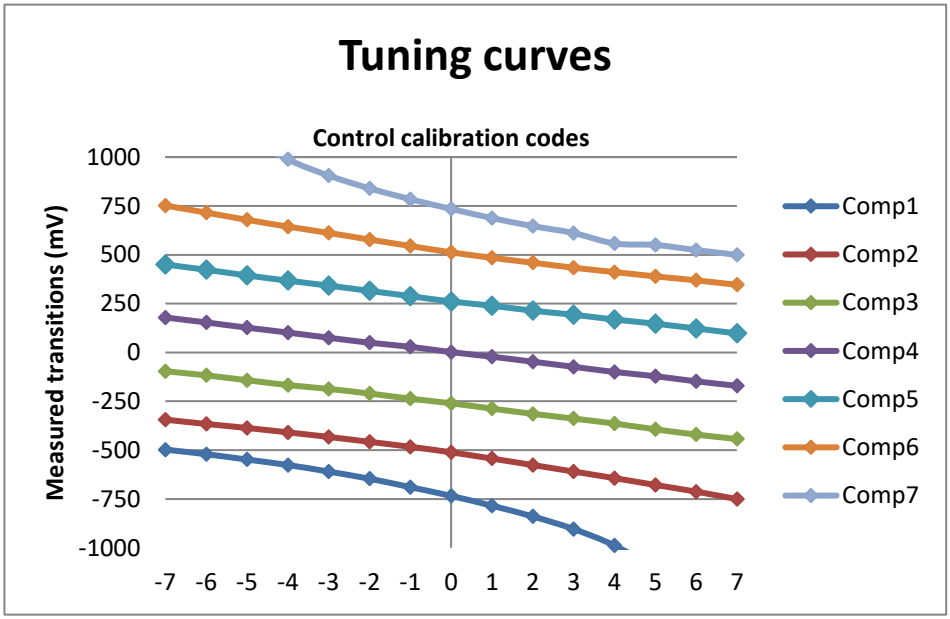

Figure 5.14. Tuning curves

Furthermore, post-layout simulations of the complete flash are done in open-loop configuration. Results for R+C+CC parasitic extracted flash at sampling frequency of 150MHz are gathered in Table 5.6. The number of comparator M and its nominal transition t_{id} are shown in the table. Also, an external control code $CtrlCal0$ has been applied and the transition forced with this external control has been measured ($t_{CtrlCal0}$). In the complete circuit, the flash ADC will react to the applied external control and COCL block will give a control code $CtrlCalB$ that counteracts the effect of the external code. In this simulation, an opposite $CtrlCalB$ value to the external code $CtrlCal0$ has been applied and the effective transition t_{eff} has been measured. This way, both control

codes are tested. The effective transition is the final transition after summing both control codes applied, which should be coincident to the nominal transition value. This way, both control codes are tested and control-word adder block shows its proper operation. It can be concluded that this calibration method is robust enough to calibrate a quite wide voltage range and also the effect in the limit codes has been measured.

Table 5.6.Post-layout open-loop simulation results of complete flash ADC @$150MHz$

M	$t_{id}\ (mV)$	$CtrlCal0$	$t_{CtrlCal0}\ (mV)$	$CtrlCalB$	$t_{eff}\ (mV)$
1	-750	-1	-689.6	1	-734.0
2	-500	2	-576.3	-2	-509.3
3	-250	-3	-186.6	3	-250.4
4	0	-2	49.9	2	3.8
5	250	4	168.5	-4	266.6
6	500	-2	577.8	2	516.9
7	750	-3	905.5	3	733.0

Results show that, for every comparator, the effective transition is rather close to the nominal transition, which proves that every comparator has been designed conveniently for its own transition and so flash ADC is working correctly. In addition, simulations up to $400MHz$ have been tested, obtaining for all of them a quite good ENOB of approximately 3bits.

5.5. 6-bit SAR ADC design

In the design, SAR ADC should act as an ideal quantizer, since it is employed to help calibrate flash ADC. Hence, SAR must have a resolution at least 2 bits greater than that of the flash. The selected SAR is a 6-bit resolution one and it should be able to convert the 64 transitions with no input saturation. SAR is based on Tripathi and Murmann's work [142]. The authors present an asynchronous version of an 8-bit SAR ADC, reaching a frequency of $450MS/s$ and in a 65nm CMOS technology. However, the designed SAR ADC in this thesis has a lower resolution and employs a synchronous clock to reduce the complexity in the design. To convert single ended clocks into complementary clocks,

SAR uses two clock diplexers to control a switch capacitor DAC from a sequential SAR logic. This simple topology employs mirrored arrays and differential switches as analogue gates. Bits are obtained in a serial manner while switches from array are controlled. Finally, SAR ADC architecture is shown in Figure 5.15.

SAR logic block generates the output code in a serial manner. These bits are also employed in controlling the switches of the DAC array. Capacitive DAC is explained in Section 5.5.1. Two clock diplexers are required to convert single ended clocks (SAR clock clk and EN) into differential clocks, which symmetrically act on the switches of the DAC array. Also, same comparator as in Flash for transition $t_j = 0V$, with preamplifier and without SBMOS control. It has been detailed in Section 5.5.3.

The ADC has a conversion reference voltage of $R = \pm 1V$ $(FS = 2V)$. However, the external implemented reference V_{ref} has to be selected properly in order to spread transitions within the range $[-R, R] = [-1V, 1V]$.

SAR enable signal ENs (from SAR logic) goes high every 8 cycles of SAR clock clks (corresponding to $SARclock$). For instance, if clock frequency is $f_{SARclock} = 50MHz$, the sampling frequency will be $f_{sampSAR} = \frac{f_{SARclock}}{8} = 6.25MHz$.

At $clks$ rising edge, both comparator inputs are compared and if differential input $(inp\_comp - inn\_comp)$ is positive, comparator output out_comp will be high and SAR logic will give SAR code output MSB.

Most important blocks are explained in detail in the following subsections.

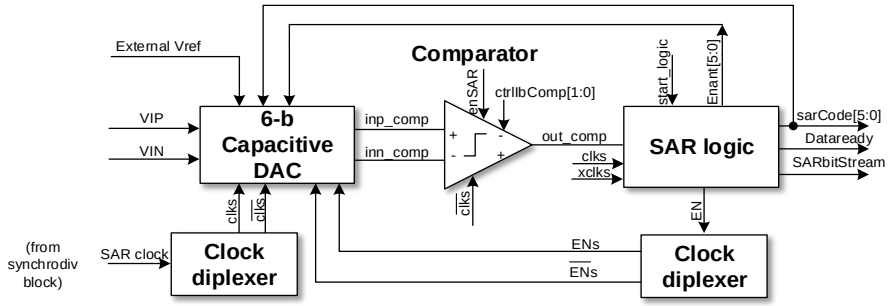

Figure 5.15. SAR ADC architecture

5.5.1. Binary weighted DAC

The employed DAC has a fully differential switch capacitor topology, as depicted in Figure 5.16. It is form of switches, a capacitor array and some NOR gates and inverters.

A detailed picture of the DAC capacitor architecture is shown in Figure 5.17. It consists of a 6-bit binary weighted capacitive DAC, where the input voltage is injected through capacitors top plates. One bit is resolved in each clock cycle. Figure shows capacitors connections for value of SAR code MSB.

The capacitor array has common centroid architecture to cancel mismatch effects. Only 6 different capacitors are needed in each branch (the unitary capacitor C_0, $2C_0$, $4C_0$, $8C_0$, $16C_0$ and $32C_0$). However, the number of capacitors are doubled to have common centroid signals (hence, increasing DAC area) and randomise the mismatch errors that appear when changing between V_{ref} and ground. This way, each of the capacitors is split into two devices, one connected to V_{ref} and the other to ground, as shown in the picture. This division is applied to every capacitance in the DAC, except for the LSB, which has minimum size. Therefore, due to its small capacitance, the LSB capacitor connection changes do not affect to the input common-mode and, hence, there is no need for LSB capacitor to be divided into two different capacitors and it could be tied straight to V_{ref}. The LSB capacitor does not affect the SAR conversion, it only influences the layout parasitics instead.

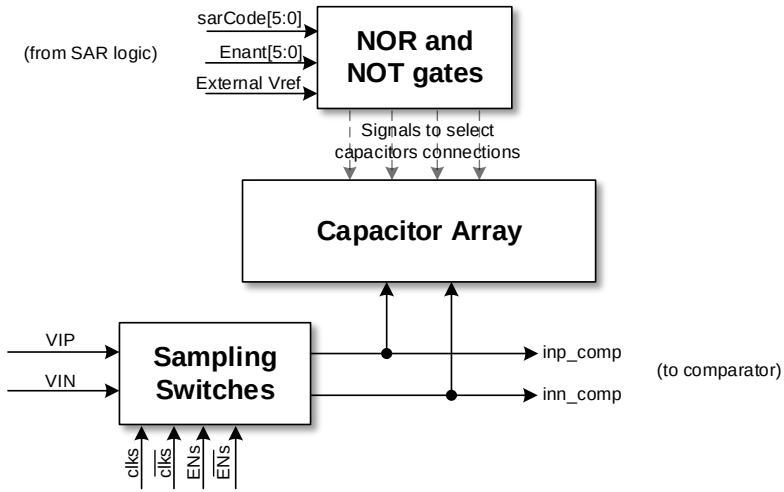

Figure 5.16. Implemented DAC

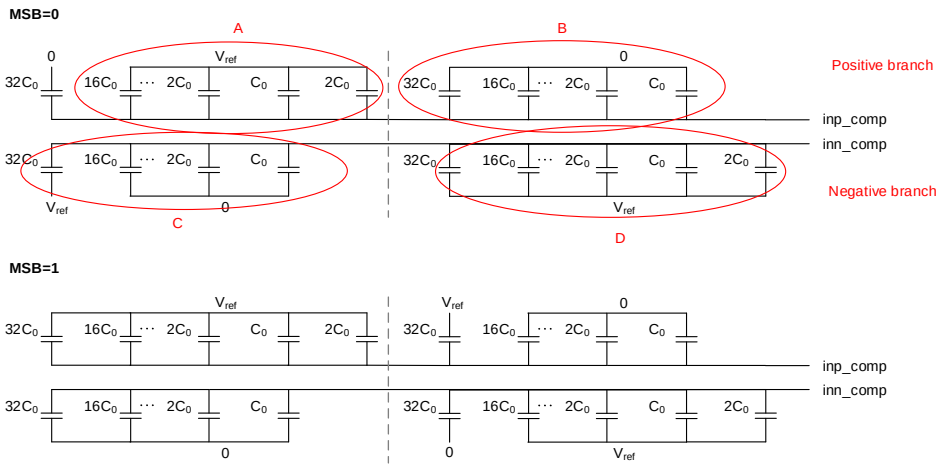

Figure 5.17. DAC capacitor array

Figure 5.18 exhibits the DAC layout, where common centroid structure is visible.

The DAC reference voltage V_{ref} is generated externally. With a value of $\sim 1.2V$, this voltage V_{ref} is applied to obtain a SAR effective reference of $R = 1V$ $(FS = 2V)$, as depicted in Figure 5.17. Considered unitary capacitance is $C_0 = 5.04fF$ and, hence, SAR input capacitance is $C_{in} = (2^r + 1)C_0 = (2^6 + 1)C_o = 327.6fF$.

In contrast to the usual switches, logic gates are employed and they are supplied by the reference voltage V_{ref}.The use of logic gates instead of switches improves the immunity of the circuit and reduces the complexity of the architecture. SAR code bits reach these logic gates, whose supply is V_{ref}.

On the other hand, logic gates exhibit peaks when changing from low to high and vice versa. However, these errors are negligible since resolution of SAR ADC (6 bits) is low and work frequency is slow, being dynamic not critic in SAR, as there is enough time for the signal to settle.

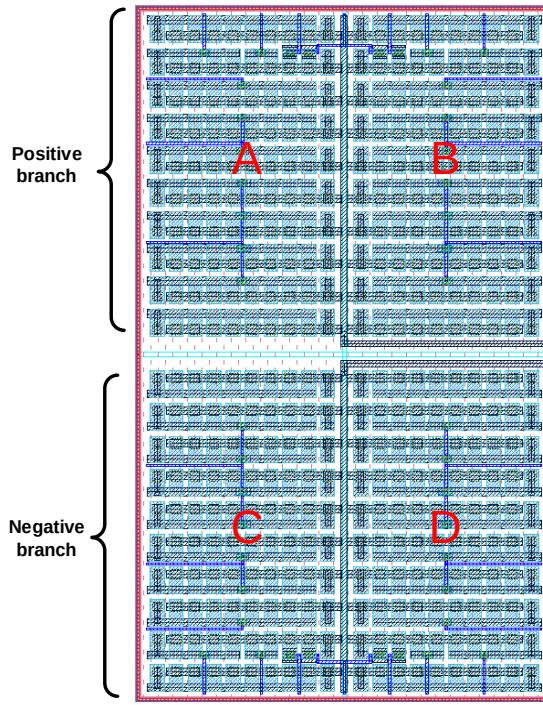

Figure 5.18. DAC layout (144µm high x 84µm wide)

5.5.2. SAR sampling switches

As explained in Section 5.3.1, flash and SAR must capture the same input voltage. To achieve this, two CMOS switches are used, which include dummy switches to cope with injection charge in switches opening. The CMOS switches have a differential schema with four switches, as depicted in Figure 5.19. The first two switches are employed to adapt SAR clock $clks$ (corresponding to $SARclock$) and the other two for enable signal ENs (from SAR logic). Clock $\overline{clks}$ is synchronised with the comparator and signal ENs is synchronised with $clks$ to capture the correct input voltage. Switches have been optimized to do this job.

Notice that ENs and $\overline{ENs}$ signals are generated by SAR logic block, which is clocked by signal clks. Enable signal ENs is related to SAR clock $clks$ by a factor of 8, that is, ENs activates every 8 cycles of signal $clks$, as shown in Figure 5.21.

Also, capacitor array immunity is achieved thanks to the second pair of switches. This can be seen in Figure 5.19. It is observed that sampling is produce by $clks$ while ENs signal is used to immunize the capacitor array since $clks$ is still working and ENs goes low to avoid the influence of $clks$ in the DAC capacitor array. Switches layout is shown in Figure 5.20. The area of this block has a height of $50\mu m$ and a width of $22\mu m$.

Figure 5.21 exhibits the generation of SAR sampling signals. Figure 5.22 explains how SAR sampling and bits generation are accomplished. First graphic (A) shows the signal tracking. Gates are connected to virtual ground since capacitors connected to V_{ref} have a total voltage of $V_{ref} - V_{ref} = 0V$ and capacitors connected to ground have also $0V$. Hence, total capacitance of the array is connected to ground and comparator is off.

Second graphic (B) represents the sampling and MSB bit generation. First switch is opened with SAR clock signal $clks$, second switch is still closed with ENs signal and total capacitance of the array is connected to virtual ground and comparator activates.

Last graphic (C) shows that when clock $clks$ opens and closes the switch constantly, the signal ENs keeps its switch open to immunize the comparator.

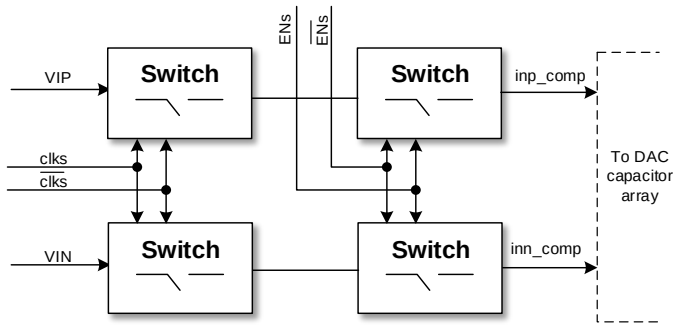

Figure 5.19. Implemented sampling switches

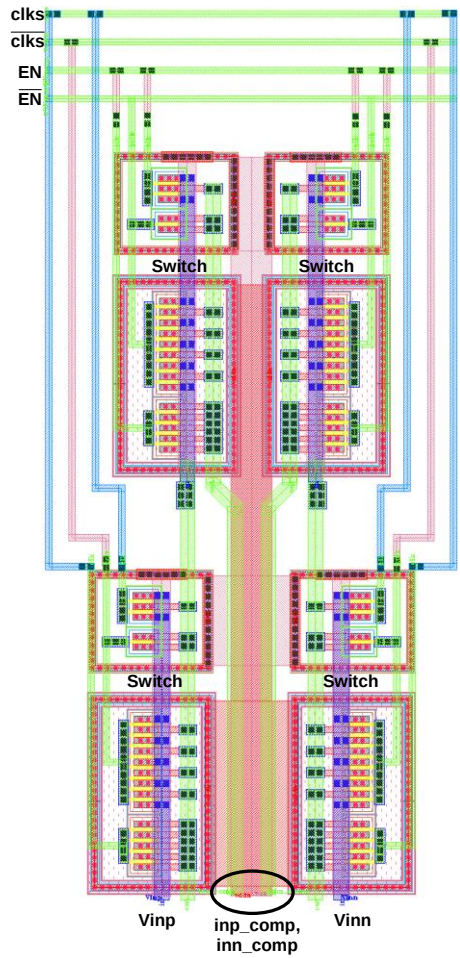

Figure 5.20. Switches block layout (50μm high x 22μm wide)

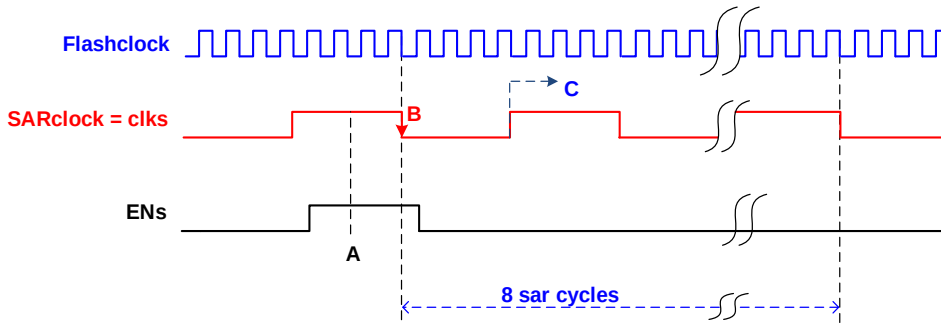

Figure 5.21. Sampling signals in SAR ADC

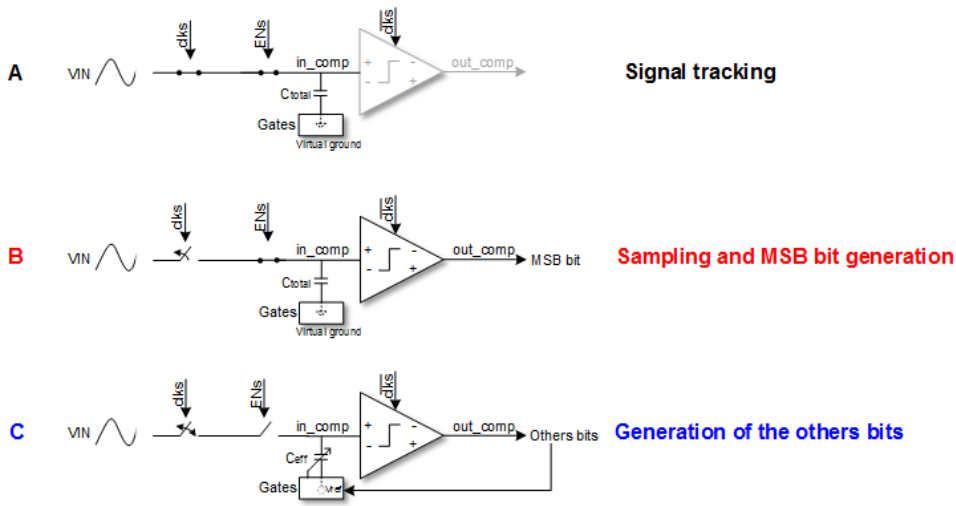

Figure 5.22. SAR sampling and bits generation

5.5.3. Comparator architecture

The employed comparator (depicted in Figure 5.23) is the same comparator as in flash ADC. Same DL and RS latch have been used. However, a preamplifier has been employed to reduce offset and mismatch problems, since preamplifier improves the precision of the comparator by dividing the offset effect by the preamplifier gain. The designed preamplifier is a NMOS differential pair with resistive load and gain of 10V/V.

There are two configuration signals that are externally controlled with the SP64b block.

First signal is $enSAR$, which enables the generation of the current bias of the comparator. If $enSAR = 1$, SAR logic and comparator are activated. Otherwise, SAR logic is disabled. Second signal is $ctrlIbComp[1:0]$, which selects the current bias of the preamplifier. Minimum possible current is $7\mu A$ and maximum is $12\mu A$.

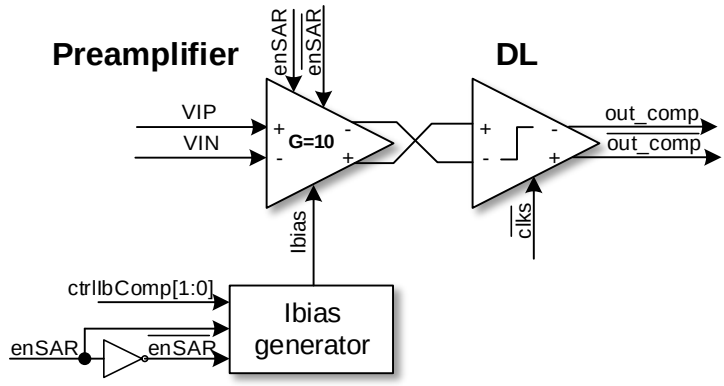

Figure 5.23. SAR comparator

Comparator output signal is shown in Figure 5.24. Input signal is sampled in falling edge of $clks$ and comparator starts comparing and outputs a value for out_comp. In rising edge of $clks$ comparison is finished and comparator output data could be employed in configuring SAR logic and generating enable signals in charge of connecting the NOR gates and inverters to select the capacitors connections, as depicted in DAC configuration in Figure 5.16.

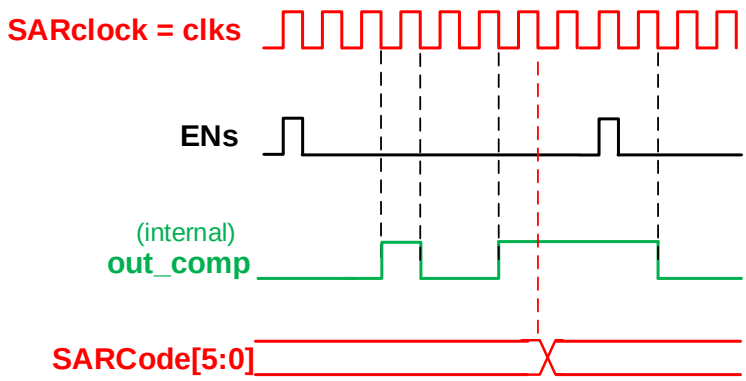

Figure 5.24. SAR ADC signals

SAR comparator layout is shown in Figure 5.25. Note that bias of preamplifier layout is greater that layout of the rest of the comparator. However, this was not a problem, since there was enough free area in our design to implement a preamplifier with a vast area and programmable bias current.

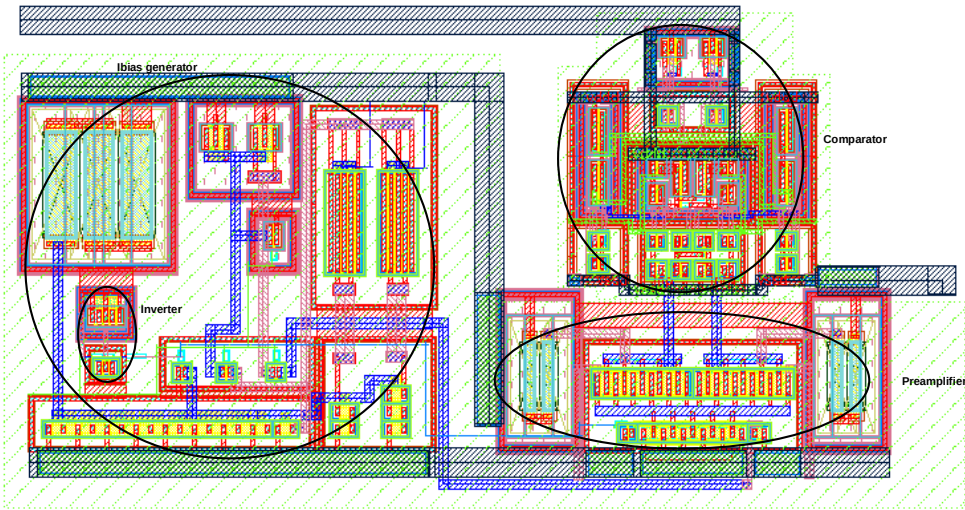

Figure 5.25. SAR comparator layout (45µm high x 92µm wide)

5.5.4. Layout and simulations

SAR layout is shown in Figure 5.26. This ADC has a height of $294\mu m$ and a width of $420\mu m$. Digital signals have been kept far from the analogue and mixed-signal parts. Remarkable is the symmetrical DAC capacitor array with a centroid structure and symmetrical routing. Dummy capacitors around DAC capacitor array are employed, maintaining the common-centroid structure. Those capacitors have been also added to the schematic. The LSB capacitors do no influence the SAR process, but only influence the parasitic extraction.

Further, SAR logic has been semi-custom designed at schematic level and later Faraday gates has been employed. Note that there is enough free area and, hence, a relaxed routing has been done. Also, DAC size is big, since no great effort has been done in reducing it due to lack of time.

NOR gates and inverters in DAC behave similarly to switches. Since the SAR resolution required to calibrate is low (only 2 bits above flash resolution are needed), SAR design and its layout are not required to be done so careful, but it has to be small and has a low power consumption. However, DAC switches are analogue switches with differential structure and their layout has to be completely symmetrical and careful done.

Post-layout simulations for the complete SAR ADC have been carried out at different clock and input frequencies. With an input signal of $3MHz$, SAR reaches an ENOB of 5.1 bits up to $12.5Msps$. SAR simulation performance is good enough to guarantee the offset calibration process.

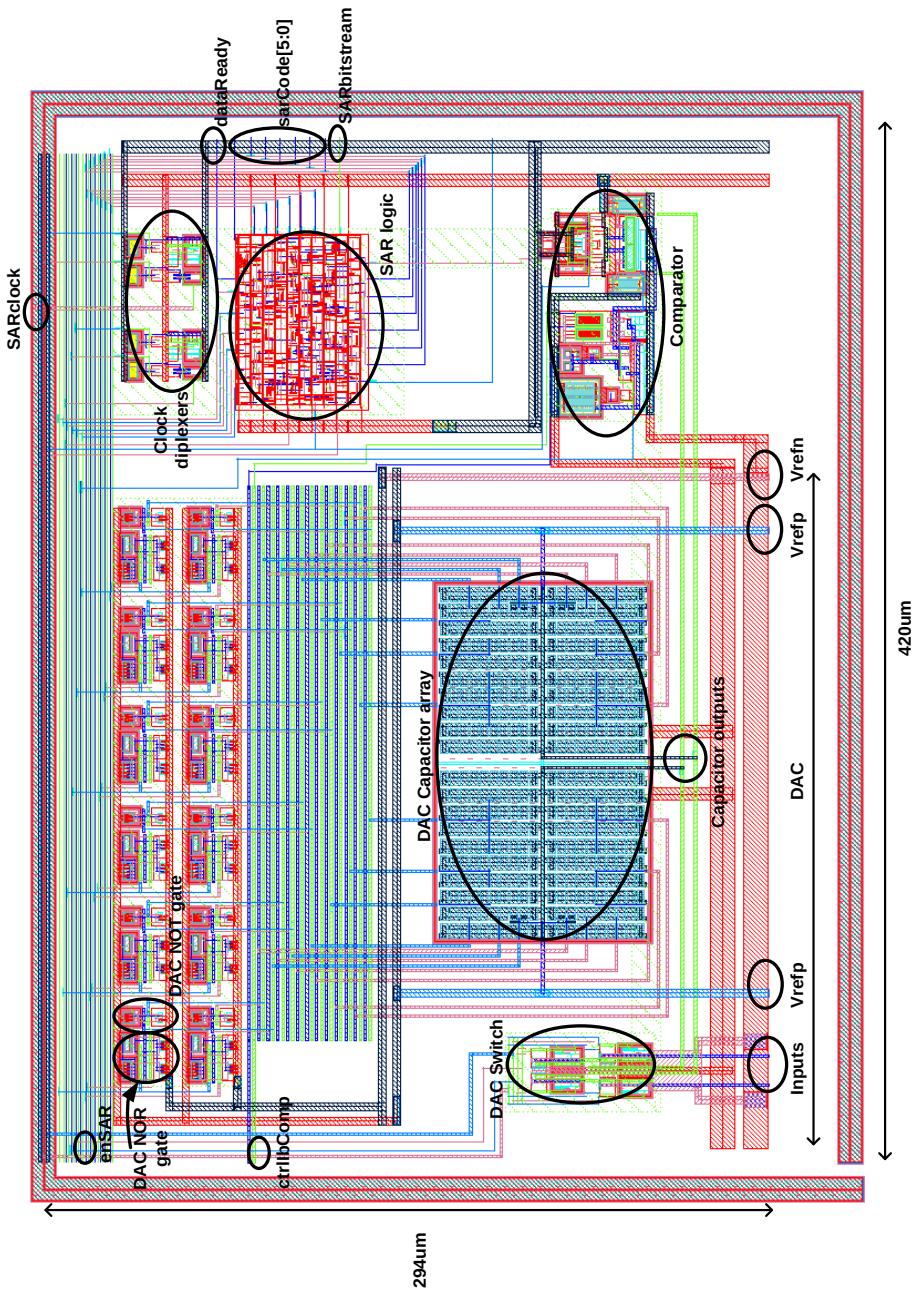

Figure 5.26. SAR layout (294µm high x 420µm wide)

5.6. Calibration block (COCL)

Calibration block (COCL) is in charge of calibrating offset in every flash comparator. The 28 output bits of the COCL (7 different calibration codes – one per comparator – and each calibration code with 4 bits) cannot be outputted in parallel due to the lack of pads. Nevertheless, thanks to the serial/parallel block (SP64b), the 4-bit calibration code can be read in series on demand.

As described in Section 3.4, COCL functionality is based on the idea of the adaptive digital background algorithm published in [105]. First, offsets in comparators are measured and then calibration codes shift the new transitions' locations to their ideal values, thus cancelling their offsets. Both tasks are implemented in the digital domain.

COCL block has been described in Verilog. To test the operation of COCL, great programmability has been included in this Verilog design, making the code rather complicated and its layout greater. A trade-off between Verilog complexity and programmability has been found. However, since such a great programmability is for test reasons, having a great layout area is not a problem. Programmability of COCL includes:

- The possibility of selecting the algorithm limits with parameter $selEQ$. If $selEQ = 0$, $SARCode \in (T_k, T_k + 1)$, and when $selEQ = 1$, $SARCode \in [T_k, T_k + 1)$, being T_k the SAR k-transition (see Section 3.4.1, equation (3-5)). Considering open or closed intervals could be of great importance because in low resolution the distance between SAR transitions is great. By default, $selEQ = 1$ is chosen.
- Selecting the accuracy in evaluating offset with parameters Δ_{down} and Δ_{up}. Offset (OFF) after calibration must be $-\Delta_{down} < OFF < \Delta_{up}$, taking Δ_{down} and Δ_{up} values from 0 to 3. By default, $-2 < OFF < 1$ is considered (see Section 3.4.2, equation (3-8)).
- Parameter $pcal[1:0]$, which allows increasing the internal processing arithmetic resolution up to 8 bits: the use of 6, 7 and 8 bits for estimating the offset in comparators is possible. By default, 8 bits are considered for internal processing.
- The choice of introducing an offset in $SARCode[5:0]$ with parameter $shiftX$ to

obtain fewer errors at quantization. Normal mode ($shiftX = 0$) centers the $SARCode$, obtaining errors in the interval $[0, LSB]$. With $shiftX = 1$, an offset is added to $SARCode$ and the waveform is shifted an amount of $LSB/2$ to the right, reducing its error from LSB to $LSB/2$. In this case, errors are in the interval $[-LSB/2\ , LSB/2]$. This is de default mode.

In addition to COCL algorithm, a block to monitor the $Ctrl$ code in real time has been included in the design, as depicted in Figure 5.27. This block is a decoder and with signal $sel\_Ctrl\_mon[2:0]$ the $Ctrl$ code to be monitored is selected.

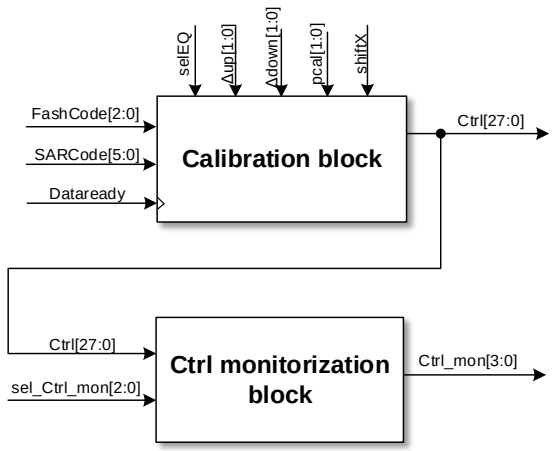

Figure 5.27. Control code monitoring and generation

A critical issue of COCL design is the delays control. COCL works at low frequency, since its clock ($Dataready$) comes from SAR logic. However, since algorithms are executed in parallel and $Ctrl$ codes are obtained in parallel, delays control is essential, especially at high frequencies, since COCL delays are added to gates and signals delays. If delays are not properly considered at COCL description, calibration synchronism could fail.

5.6.1. Layout and simulations

Layout of COCL is depicted in Figure 5.28. Done by automatic synthesis in Cadence Encounter, COCL layout is $187\mu m$ high x $202\mu m$ wide and it is composed of 1031 gates of which 110 are flip-flops. Its great size is due to the circuitry added for test reasons. A

new synthesis could be done with a simplified code description, containing only the best case implementation.

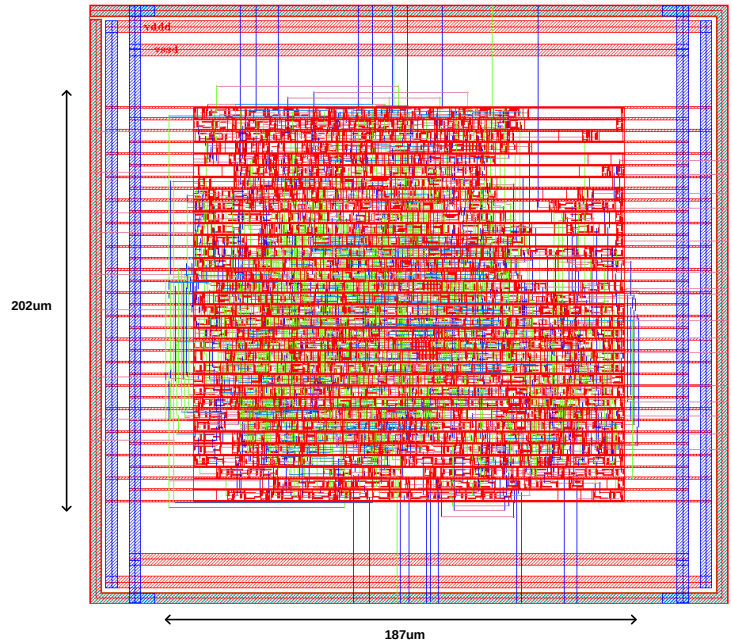

Figure 5.28. COCL layout (187μm high x 202μm wide)

To test the COCL block, post-layout simulations with Cadence Virtuoso environment are carried out. These transient simulations consider flash and SAR conversion frequencies $f_{sampflash} = f_{Flashclock} = 150MHz$ and $f_{sampSAR} = f_{ckMCOCL} = \frac{f_{sampflash}}{8} = 18.75MHz$, respectively, being frequency $f_{ckMCOCL}$ is the COCL working frequency. Input signal has a frequency of $567kHz$. COCL programmability is selected by default. An extremal external control word ($CtrlCal0$) is introduced in the comparators, adding offset voltages to their transitions. After a certain amount of time, calibration block COCL corrects the offset and gives a control bus $CtrlCalB$ that compensates for the introduced word $CtrlCal0$. Therefore, when $CtrlCal0 = -7$ is introduced, the COCL block will calibrate the offset with the word $CtrlCalB = 7$. To have a fast simulation, a ServoLoop test has been employed (a sine input signal will highly increase the simulation time). Simulation results are presented in Table 5.7.

Table 5.7. Calibration block simulation results

Comparator	$CtrlCal0$ (external)	Nominal transition (mV)	Measured transition (MAX Corner) (mV)	$CtrlCalB$ (to Flash)
1	-7	-750	-749	7
1	7	-750	-746	-7
2	-7	-500	524	7
2	7	-500	-522	-7
3	-7	-250	-243	6, 7
3	7	-250	-260	-7
4	-7	0	0	7
4	7	0	0	-7
5	-7	250	267	7
5	7	250	246	-6
6	-7	500	525	7
6	7	500	496	-6
7	-7	750	746	7
7	7	750	746	-6

Results show that calibration block is able to correct properly the offset in each comparator. Same simulations are done with different COCL configuration parameters. Maximum flash working frequency is $250MHz$, maximum SAR working frequency is $50MHz$ and, hence, maximum COCL working frequency is also $50MHz$.

It is possible to measure the response time of each comparator as the sum of the settling time and calibration time. This represents the total amount of time that each comparator employs in calibrating. Each calibration bank has different capacitance, since transistors lengths luc are different, therefore some comparators could take longer than others in activating the control code at the COCL. Hence, delays are needed to ensure that all signals coming from every comparator will reach the COCL block at the same time. In

post-layout simulations, maximum delay in control code activation is only $9ns$, which do not go against with the SAR $Datareadу$ timing (in the order of 10^{-5}).

5.7. ASIC CALFLASH layout and simulations

5.7.1. Layout

The prototype chip CALFLASH was taped out as a proof of concept for the offset calibration in a low resolution flash ADC with no references. The chip was fabricated in UMC's 180 nm CMOS process. The total area of the die is $1525mm\ x\ 1525mm$ and it was packaged in a 32-pin plastic $5mm\ x\ 5mm$ QFN32 package. The active area of the flash ADC is $210\mu m\ x\ 350\mu m$, the area of the SAR ADC is $275\mu m\ x\ 450\mu m$ and that of the COCL is only is $200\mu m\ x\ 187\mu m$. Figure 5.29 exhibits the layout of the complete circuit with its differentiated blocks. Notice that analogue domain is separated from the digital part.

5.7.2. Bonding

Digital signals are inputted/outputted through single-ended pads. Some examples are flash and SAR output codes ($flashCode[2:0]$ and $sarCode[5:0]$), COCL enable signal ($Datareadу$), reset signal ($Resetn$) and reference and PLL clocks ($ckREF$ and $ckMpll$, respectively). However, critical analogue signals are differential signals and require two pads. Examples of differential signals are the ADCs analogue input ($VIN+$ and $VIN-$) and SAR external reference ($VREFSAR$ and GND).

Instead of using a differential LVDS driver clock block provided by Faraday Technology, the PLL given by the technology has smaller area and great versatility, this was the reason for choosing this block.

The circuit core operates from a 1.8V supply, but digital IO pins operate from 3.3V. Moreover, three independent power domains are differentiated for better noise behaviour: analogue power supply 1.8V, digital core supply 1.8V and PLL power supply 1.8V. To reduce the number of pads, all ground pins are shorter together and bonded to the die attach connection of QFN32 package. Figure 5.30 shows its bonding

diagram.

With these techniques, the total number of pins of the package is reduced to 32, while the number of pads on the silicon was 45. Hence, a smaller package (QFN32) is required and shorter bond wires are employed, improving the signal integrity.

5.7.3. Simulations

Exhaustive post-layout simulations has been carried out with Cadence Virtuoso. In order to do the simulations fast, a ServoLoop test has been used with a ramp input signal. The different blocks in the circuit have been checked with simulations at different levels of abstraction. Mixed-simulations have been done considering flash and SAR ADCs at schematic level, the digital part in HDL language and I/O pin as electric macromodel. In some simulations, digital blocks have been written in Verilog-A in order to speed up the process. The different programmable bits of COCL block have been also simulated, all of them reaching the offset cancellation. The complete circuit has been test at different frequencies, giving great results for flash offset calibration up to 200 Msps.

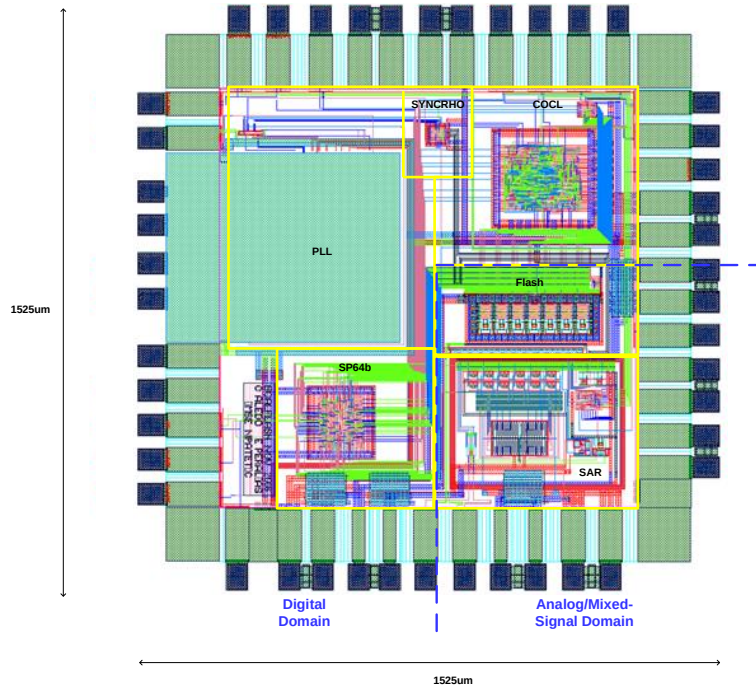

Figure 5.29. CALFLASH layout

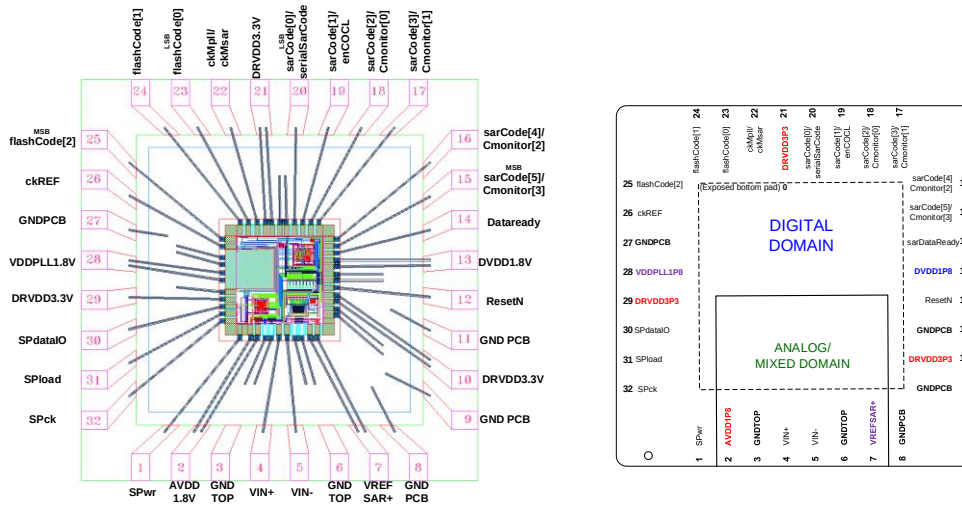

Figure 5.30. Bonding diagram to QFN32 package

6. EXPERIMENTAL RESULTS

This chapter presents the silicon test results of the CALFLASH test chip. The chip was fabricated in a 180nm CMOS process. PCB design is presented in Section 6.1. Test setup and communication interface are both explained in Sections 6.2 and 6.3, respectively. Finally, results for test chip are shown in Section 6.4.

6.1. PCB setup

The chip is tested on a dedicated printed circuit board (PCB) for testing. PCB design is presented in Figure 6.1 and its corresponding photograph of the PCB is shown in Figure 6.2.

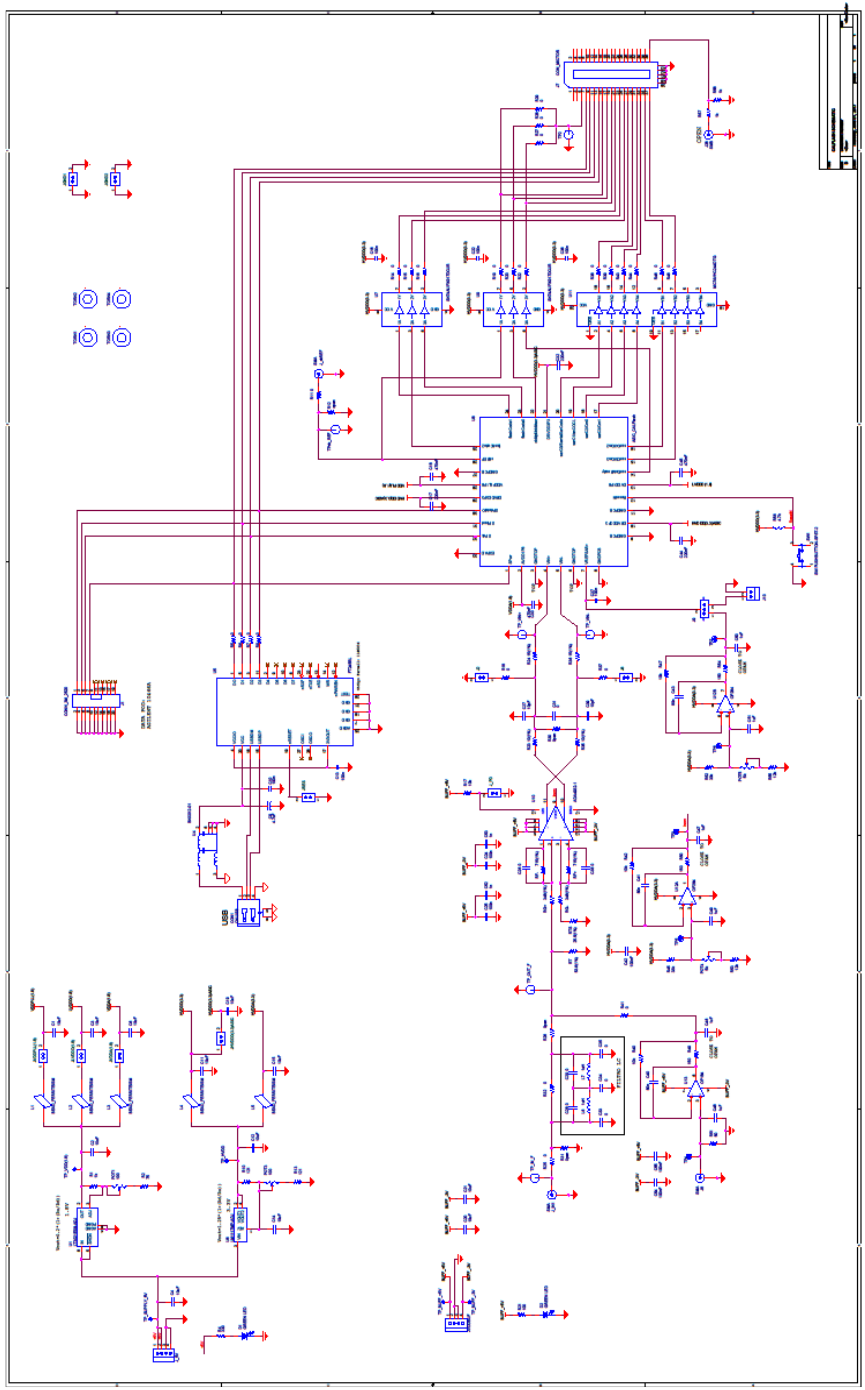

Figure 6.1. CALFLASH PCB schematic

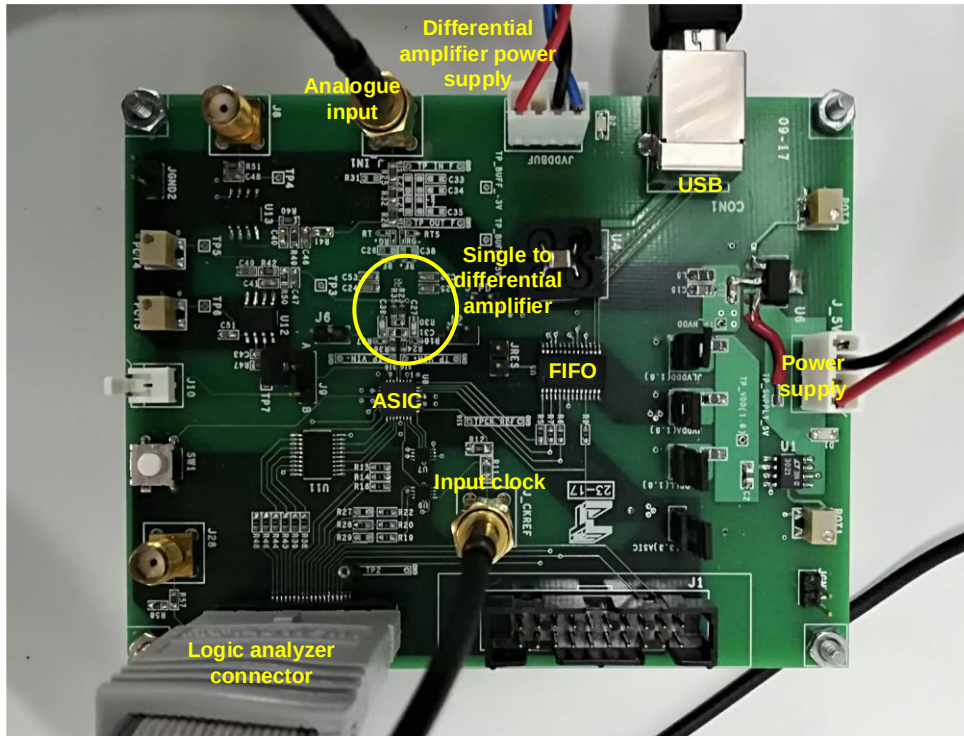

Figure 6.2. Photograph of CALFLASH PCB

6.2. Test setup

Figure 6.3 depicts the block diagram of the measurement setup for CALFLASH chip used in the laboratory. An arbitrary function generator Tektronix AFG3102 with two channels provides the input signal (fin) through channel 1 and clock reference signal ($ckREF$) through channel 2.

A power supply HP E3631A delivers +5V to the circuit. From this voltage, two regulators are used to generate VDD 3.3V for I/O and some blocks of the PBC and also 1.8V VDD for analogue, digital and PLL blocks. In addition, power supply gives +5V and -3V to the single-to-differential amplifier for analogue input.

A logic analyzer 16823A is employed to capture the digital signals outputted from the chip. Also, an oscilloscope is used to see the waveforms. Further, an Agilent 34410 multimeter is utilized to measure the transitions' locations with great precision.

Finally, a PC is required to communicate with the PCB and to send the data and control signals of the chip through MATLAB software. PCB is connected to PC via USB connection while the rest of the instruments are connected to the PC via GPIB.

A photograph of the test setup is shown in Figure 6.4.

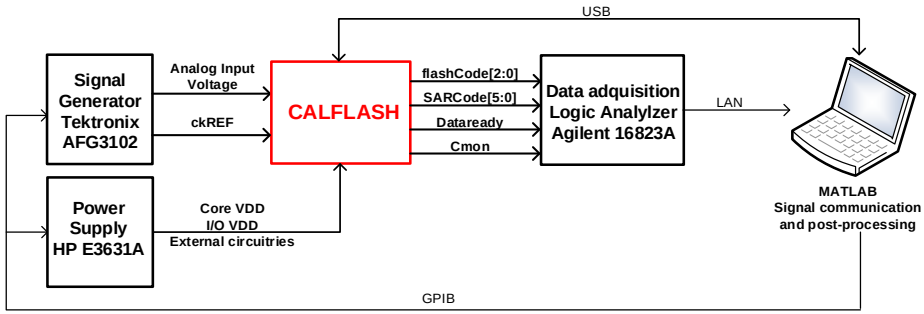

Figure 6.3. CALFLASH diagram block setup

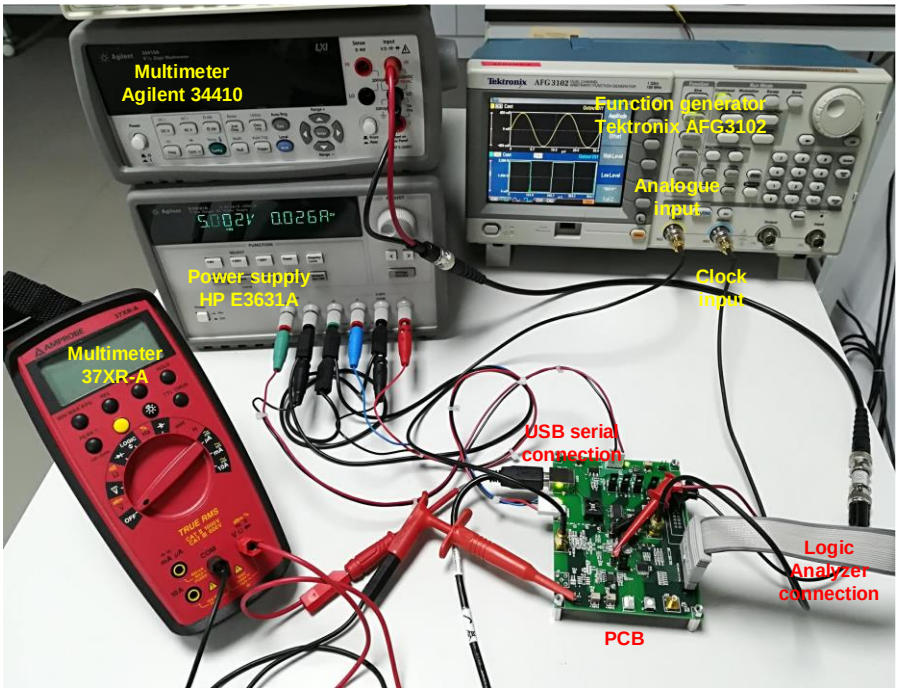

Figure 6.4. CALFLASH test setup

6.3. Communication and control

The communication with the circuit is performed through the 64-bit serial/parallel interface (SP64b), as depicted in Figure 6.5. A computer with MATLAB software has been used to send the different programming configuration and read the outputs of the circuit. PC is connected to PCB via USB port. Through MATLAB, programming values are sent to the circuit and SP64b block programs the different blocks of the circuit with the sent parameters.

Also, MATLAB software has been employed for post-processing the obtained data, for instance, for ADCs signal reconstruction and performance evaluation as well as for flash ADC calibration.

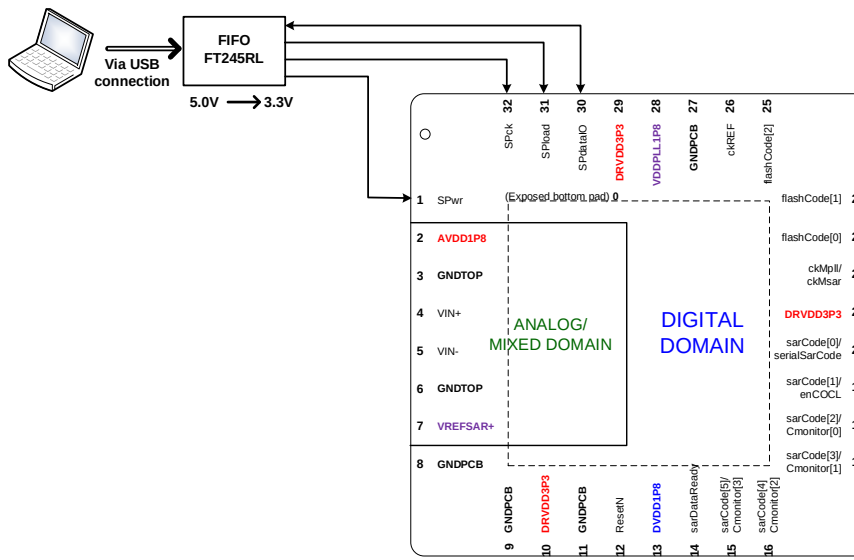

Figure 6.5. CALFLASH programming serial interface

6.4. Measured results

Three main blocks compose the CALFLASH circuit. Each of these blocks has been individually tested. First, both ADCs (flash and SAR) are proved to convert satisfactory. Later, the whole circuit is checked to verify that calibration is done properly.

All different circuit configurations have been programmed manually. Each of the configuration modes of the circuit is programmed by sending a succession of states to the chip which charges the wanted values of the different parameters to be configured. Through SP64b circuit these configurations have been sent. Four different configuration modes are distinguished:

- Normal conversion of flash ADC
- Normal conversion of SAR ADC
- Normal conversion of flash ADC with external offset applied
- Calibration of flash ADC

All of the configuration modes above are explained in the following subsections. In all tests, a sinusoidal input signal with frequency f_{in} and a squared external reference clock with frequency f_{ckREF} are assumed. The considered input frequency f_{in} must have enough amplitude to excite all the transitions in flash ADC, that is, an amplitude greater or equal to $2Vpp$. External reference clock frequency will be $f_{ckREF} = 10MHz$, unless otherwise indicated.

6.4.1. Normal conversion of flash ADC

A characterization of the flash ADC is carried out while SAR ADC is deactivated. Figure 6.6 shows the blocks that are turned on during the flash conversion. Disabled blocks are drawn in grey colour. Note that when flash sampling clock is low, there is no need for activating the PLL. However, to obtain greater clock frequencies, PLL activation is required.

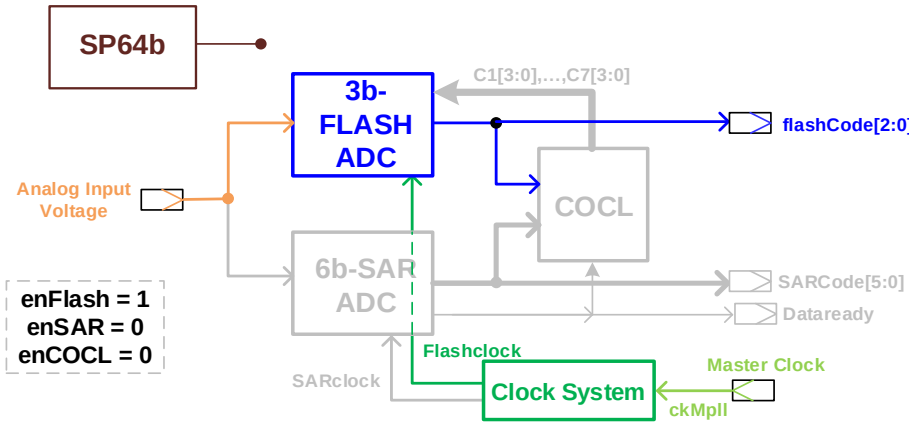

Figure 6.6. Flash ADC in normal conversion

Furthermore, to fulfil a complete characterization of the flash ADC, flash has been tested in its conversion mode not only while SAR is disabled, but also with SAR enabled. At power on, the default state of the circuit is set, both ADCs are enabled and then flash starts converting.

Flash ADC working clock is $Flashclock$, which is a delayed version of the reference clock $ckREF$. Since flash samples in every clock cycle $Flashclock$, its sampling frequency ($f_{sampflash}$) is equal to the working frequency:

$$f_{sampflash} = f_{flashclock} \tag{6–1}$$

In this subsection, flash ADC characterization is done at low, medium and high sampling frequencies. This way, maximum sampling frequency of flash is obtained. Furthermore, in some cases, the clock system (formed of a PLL and a frequency divider) is enabled and flash sampling frequency is given by:

$$f_{sampflash} = pll_{mult} \cdot f_{ckREF} \tag{6–2}$$

where pll_{mult} is the multiplication parameter that introduces the activation of PLL. On the contrary, when PLL is disabled, sampling clock is directly the reference clock from the function generator:

$$f_{sampflash} = f_{ckREF} \tag{6–3}$$

On the other hand, when SAR is enabled, its working frequency is given by:

$$f_{SARclock} = \frac{f_{flashclock}}{freqdiv} = pll_{mult}\frac{f_{ckREF}}{freqdiv} \quad (6\text{–}4)$$

and SAR sampling frequency is:

$$f_{sampSAR} = \frac{f_{SARclock}}{8} = \frac{f_{flashclock}}{8 \cdot freqdiv} = pll_{mult}\frac{f_{ckREF}}{8 \cdot freqdiv} \quad (6\text{–}5)$$

In characterization of flash ADC, similar ENOB results are obtained when SAR is disabled or enabled. Figure 6.7 shows the ENOB of the flash ADC while SAR ADC is enabled for different sampling frequencies. In the picture, different input frequencies are considered: {1MHz, 10MHz, 30MHz, 60MHz}. In these cases, SAR sampling frequency is minimum, being $freqdiv = 32$. The graphic shows that ENOB lowers as input frequency and flash sampling frequency increase. Maximum flash ADC sampling frequency is around 220Msps, regardless of the value of the input frequency. For greater sampling frequencies, the ENOB sharply decreases due to the impossibility of detecting the clock generated by the PLL circuit because of the PCB and circuit setup. Note also that ENOB lowers its value at low flash sampling frequencies. Corner frequency increases as input frequency rises. The flash ADC achieves an ENOB around 2.9 bits up to 220Msps in its default state (no offset injected).

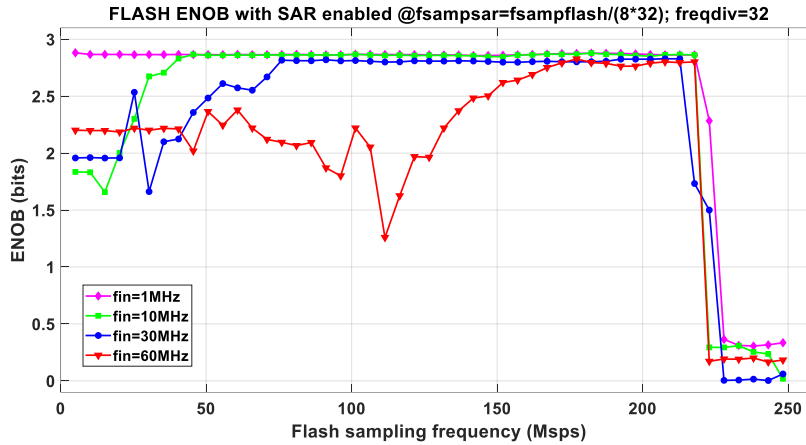

Figure 6.7. Flash ENOB with SAR enabled working at minimum frequency with symmetric sampling clock and PLL ON

Figure 6.8 depicts flash ENOB versus sampling frequency while SAR is working at higher frequencies than in last graphic. SAR sampling frequencies are considered between 5Msps and 10Msps and results show that SAR does not barely influence flash ENOB when flash is working in the considered range of frequencies, even if input frequency is great.

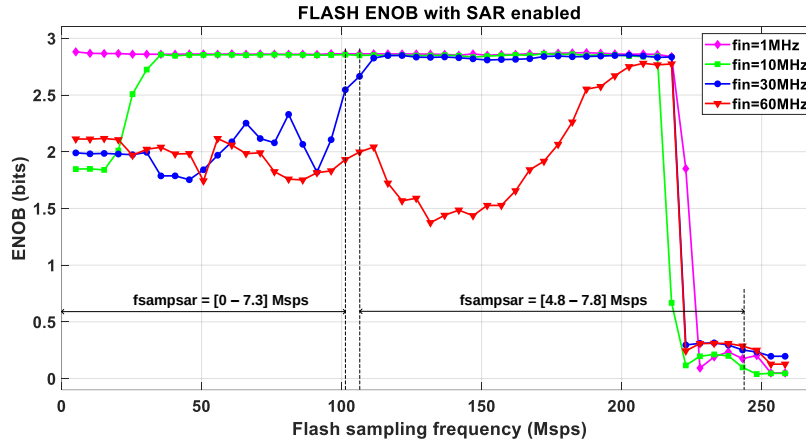

Figure 6.8. Flash ENOB with SAR sampling clock between 5 Msps and 10 Msps with symmetric sampling clock and PLL ON

As we can see from the two pictures of flash ENOB, when input frequency is 30MHz and flash sampling frequency is 50Msps, the achieved ENOB is lower than that when

the sampling frequency is greater, for example, 100Msps (in graphic Figure 6.7, the ENOB reaches 2.5bits at 50Msps and 2.8bits at 100Msps). This proves that when working with fast input signals, flash needs high sampling frequencies to have a correct working. The reason for this dynamic effect could be due to the relationship between the input and sampling frequencies. Indeed, the aperture error could have influence. A way of solving this effect is using an appropriate comparison width (that is, the pulse width of the comparator latch).

During the reset phase of the comparator, the input signal is cut and, hence, the signal does not affect the comparator performance. However, during the evaluation phase, the latch clock is high and the input signal reaches the comparator input, affecting the regenerative latch output (see latch topology in Figure 4.14). For greater comparison pulse widths, the time the input signal is disturbing the comparator signal is higher than that for lower widths. Therefore, the output code is longer affected by the input signal. This could be solved by using a narrow comparison width, since cutting the input signal reaching the comparator input is not an option in the implemented circuit.

In order to prove the effect of the input signal on the flash ENOB, a narrow comparison pulse has been used. A narrow pulse is obtained from an asymmetric clock. Clock signal has been directly generated from the arbitrary function generator, since an asymmetric pulse is required in this test and, in this case, activating the PLL is not possible because PLL output is always symmetric. This way of generating the clock limits the maximum sampling frequency to the maximum frequency achieved by the generator, which is 50MHz.

Chronogram depicted in Figure 6.9 shows the asymmetric pulse of the comparator clock. The comparator captures the data in the rising edge of the comparator clock *Compclock*. Flash sampling clock is an inverted clock from *Compclock* signal. Results for asymmetric comparison clocks with narrow widths of 10ns are shown in Figure 6.10. In this graphic, ENOB measured using symmetric comparison clock (from Figure 6.7) is also depicted. Notorious is that the ENOB remains regular for sampling frequencies lower than 50Msps when using asymmetric latch clock. This means that input frequency is independent of flash sampling frequency, since the ENOB keeps stable in the considered sampling frequency range.

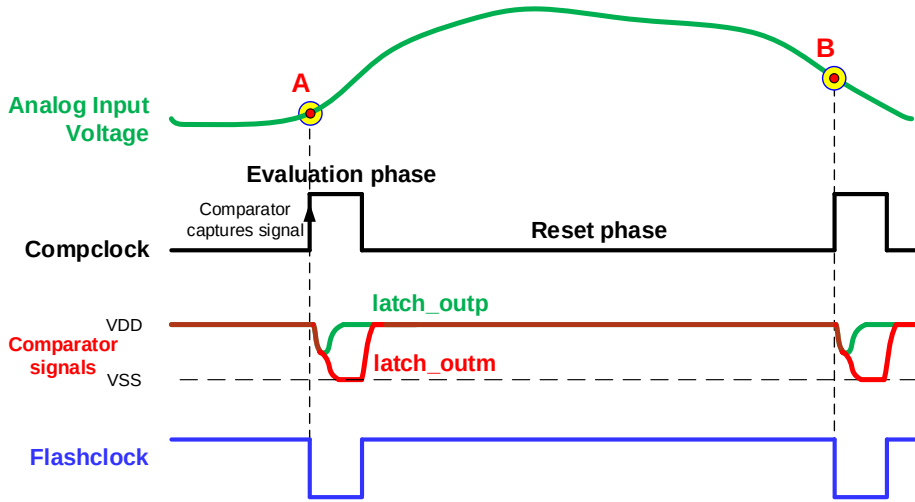

Figure 6.9. Chronogram with asymmetric comparison pulse

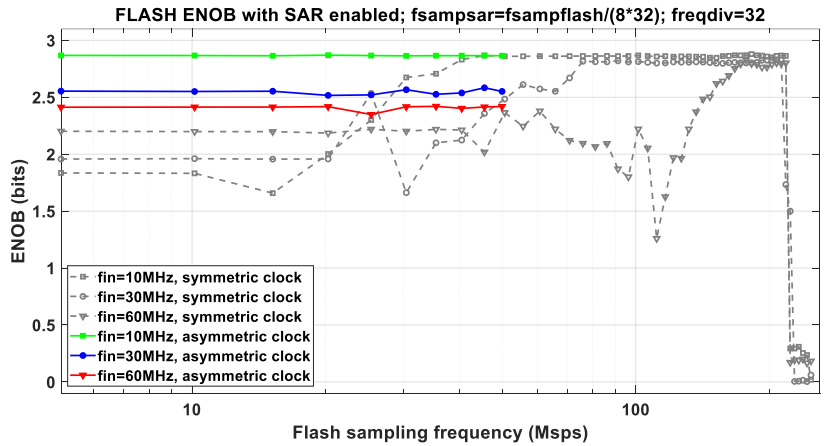

Figure 6.10. Flash ENOB with SAR enabled working at minimum frequency with asymmetric sampling clock with comparison pulse of 10 ns and PLL OFF

As already said, the input frequency is not dependent of the flash sampling frequency. Hence, this dynamic error must be due to the aperture error of the converter, in this case, of the comparator. In the design, the comparator captures the data in the rising edge of the clock. In practice, the comparator will take some time in capturing the data. If the input signal is fast, the input signal will experience a great change between the ideal rising edge and the real moment in which the data is captured, as depicted in Figure

6.11, instead of capturing signal in A1 or B1, it will captured in A2 or B2. This leads to greater error in the flash output as input frequency increases. In Figure 6.10 it is remarkable that the comparator does not capture data correctly with an input signal of 30MHz or 60MHz, while it does capture them properly with an input signal of 10MHz (ENOB of 2.8bits). Using a SH will solve these aperture problems, but our design has no SH. Also, each ADC has different aperture error. Even with a correct synchronisation of both ADCs, it is assumed to have different acquisition time for each of them. Hence, when input frequency increases, more discrepancies between both clocks will appear.

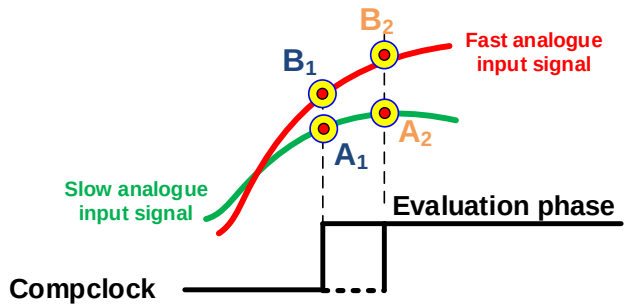

Figure 6.11. Aperture error effect in comparator

Furthermore, Figure 6.12 shows an example of the flash INL for flash sampling frequency of 150Msps, input frequency of 30MHz and SAR sampling frequency of 594ksps. INL values for each transition are between -0.15LSB and 0.25LSB.

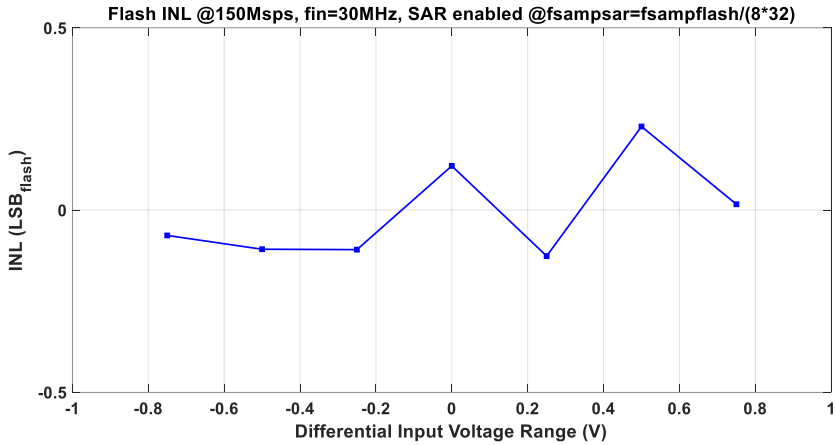

Figure 6.12. Flash INL @150Msps with SAR enabled @594ksps and input frequency of 30MHz

Note that transitions, ENOB and INL are obtained from histogram method. Moreover, the monotony of the transitions has been proved and INL error is verified to be always below half the least-significant bit, that is, $0.5LSB$ in all cases where flash is working properly.

In addition to measuring codes and transitions with histogram method, real locations of transitions of flash ADC have been also manually measured one by one for different values of input and clock frequencies. Transitions measured with both methods have similar values, appearing a maximum deviation of $10mV$ in transitions when input frequency increased over 55MHz.

All of the above described proves the correct conversion of the flash ADC.

6.4.2. Normal conversion of SAR ADC

SAR ADC acts as an ideal quantizer for calibrating flash ADC. Therefore, SAR must have a smaller LSB and, therefore, a greater resolution than flash (at least 2 more bits):

$$q_{effective,flash} \geq 4 \cdot q_{effective,SAR} \rightarrow q_{effective,SAR} \leq \frac{FS_{flash}}{2^{N_{flash}+2}} \qquad (6\text{–}6)$$

The selected SAR (with 6-bit resolution) should be able to convert the 64 transitions with no input saturation in the FS range of the flash ADC..

The characterization of the SAR ADC is performed while flash ADC is disabled. Figure 6.13 exhibits in different colours the blocks that are in use during the SAR conversion. Most of the tests are carried out with PLL enabled, thanks to its versatility to obtain the wanted frequency. Frequency $f_{SARclock}$ is the working frequency of SAR ADC, which depends on the flash working frequency $f_{flashclock}$, the pll_{mult} parameter and the frequency divider parameter ($freqdiv$). SAR ADC samples every 8 clock cycles, that is:

$$f_{sampSAR} = \frac{f_{SARclock}}{8} = \frac{f_{flashclock}}{8 \cdot freqdiv} = pll_{mult} \cdot \frac{f_{ckREF}}{8 \cdot freqdiv} \qquad (6\text{–}7)$$

Input data is captured in the falling edge of clock $SARclock$ and SAR ADC takes 8 cycles to convert the data. At $Dataready$ rising edge, data is already converted and is ready to be read. These points where $Dataready$ is high are the instants when the code is converted and ready to be used.

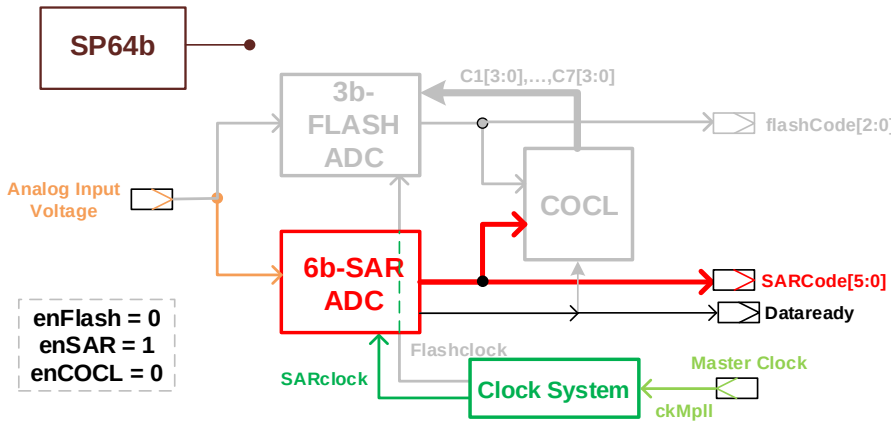

Figure 6.13. SAR ADC in normal conversion

ENOB and INL of SAR working at different frequencies are measured while flash is OFF as well as while it is enabled. Similar results for SAR are obtained when flash is disabled or activated.

Figure 6.14 shows the ENOB of the SAR ADC versus sampling frequency, considering flash enabled. In the picture, different input frequencies are evaluated. As SAR sampling clock raises, ENOB reduces. For a correct working of SAR ADC, maximum values for ENOB and sampling frequency are 5.6 bits and 15 Msps, respectively. For frequencies greater than 15 Msps, ENOB drops dramatically and circuit behaviour is not correct

because the PLL does not work properly at such a high frequency. According to the figure, below 7 Msps, SAR sampling frequency shows correct and stable ENOB values. Also, SAR ADC reaches an ENOB of 5.2 bits up to 7 Msps. Hence, in flash calibration, SAR sampling frequency must be chosen below this value. This good performance of SAR ADC is enough to guarantee the offset calibration process.

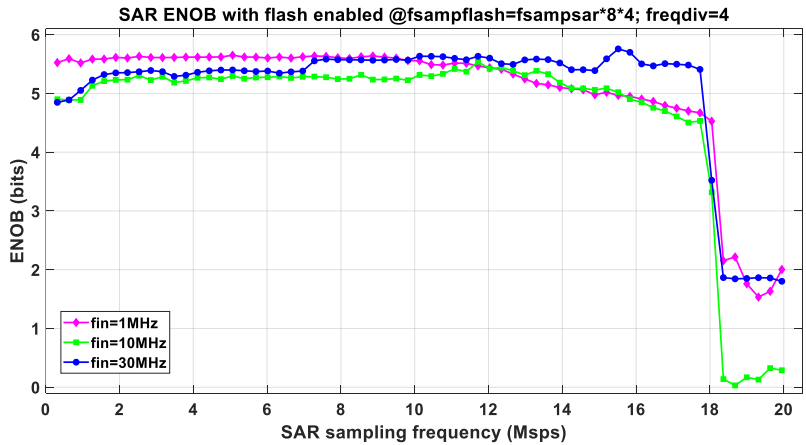

Figure 6.14. SAR ENOB with flash enabled and $f_{sampflash} = 32 \cdot f_{sampsar}$

Note that ENOB decreases at low sampling frequencies. This behavior has a twofold explanation. First and mainly, signals with input frequencies 10 MHz and 30 MHz have more noise aliased, since their input frequencies are much higher than sampling frequencies (condition for aliasing: $f_{in} > \frac{f_{sampsar}}{2}$). Second, the inherent losses of charge when transferring charges in SC circuits with small capacitors may also influence in ENOB decrease due to leakage errors at low frequencies.

The output code and INL are measured. Figure 6.15 shows the SAR linearity error INL when this ADC is working at 1Msps. Input frequency is chosen to be 1kHz and flash sampling frequency is 15Msps, with frequency divider set to 2. INL error is given as a function of the flash LSB ($LSB_{flash} = 250mV$) and remains always below $\pm 0.1 LSB_{flash}$.

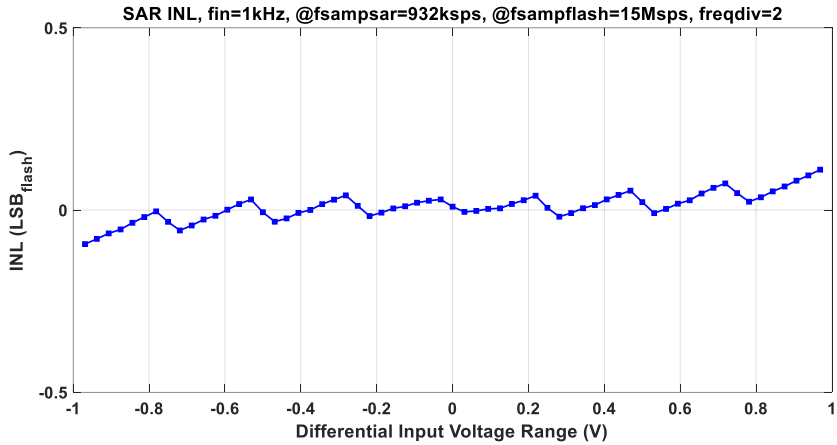

Figure 6.15. SAR INL @1Msps with flash @15Msps and input frequency of 1kHz

With these tests, SAR transitions have been obtained with histogram method, leading to similar values than those measured manually. Also, with histogram method the monotony of transitions has been checked.

6.4.3. Normal conversion of flash ADC with external offset applied

This procedure is similar to that in Section 6.4.1. A characterization of the flash ADC is done while SAR ADC is disabled. However, in this case, an artificial offset injection is applied to flash through the external control code bus $CtrlCal0$. Diagram of the whole circuit with the blocks in use in different colours is shown in Figure 6.16.

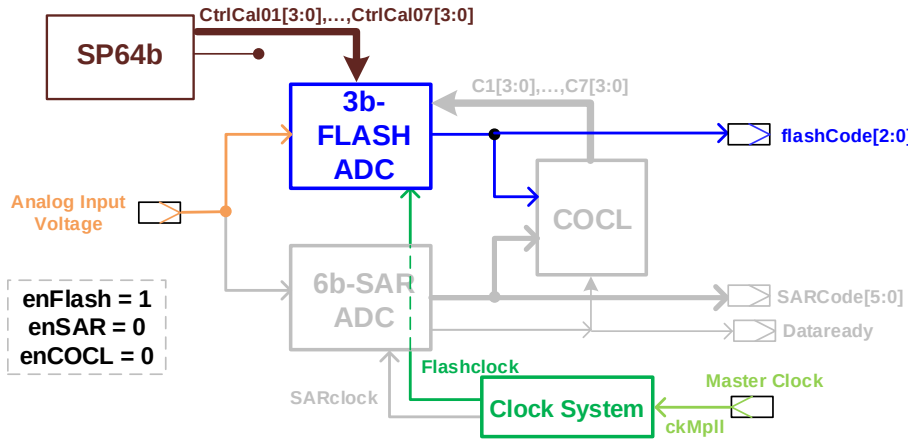

Figure 6.16. Control code injection in flash ADC

As in previous characterization, SAR is deactivated. In these tests, the default state of the circuit is set at power on. Then, a control code $CtrlCal0 \neq 0$ is sent through the SP64b block, injecting offset in one (or more) of the comparators and, hence, shifting its transitions. After, the flash ADC starts the conversion and the new location of the transition is measured.

To see how SAR activation affects to flash performance, let us consider that SAR ADC is enabled in the following tests. First, different control codes are injected in each comparator of the flash to introduce offset errors in them. Later, the calibration logic will correct these errors, as explained in 6.4.4

Figure 6.17 shows a pattern of injected control codes in flash ADC and the respective measured INL. In this case, flash and SAR sampling frequencies are 150Msps and 4.7Msps, respectively. The input frequency is 1MHz and frequency divider is set to 4. The pattern injected corresponds to control words $\{C_1 = 7, C_2 = 4, C_3 = -1, C_4 = -5, C_5 = -2, C_6 = 3, C_7 = 7\}$. Remarkable is that introducing an almost symmetric pattern causes a certain symmetry in the INL of the internal transitions. However, in extremal transitions appears some asymmetry, especially when extremal control codes are considered. For instance, first code ($C_1 = 7$) injects the code 7 into the first transition, t_1, leading to an INL of $-2.3LSB_{flash}$, while last code ($C_7 = 7$) injects code 7 into transition t_7, shifting the transition an amount of $-1LSB_{flash}$ from its ideal value. INL values are measured from histogram method. This asymmetry has been shown in

tuning curves (see Figure 5.14.)

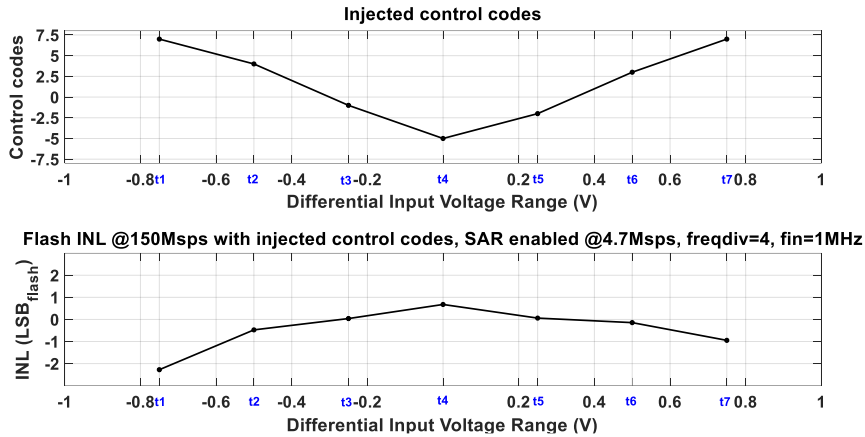

Figure 6.17. a) Injected control codes and b) flash INL versus input voltage, with flash @150Msps, SAR @4.7Msps and input frequency of 1MHz

Also, some other different patterns are applied to the flash comparators with same conditions as before. Results are shown in Figure 6.18. The graphic above exhibits the different injected patterns versus the input signal. The picture below depicts the corresponding INL for each transition (before calibrating). The injected patterns are the following:

- Pattern 0 corresponds to the nominal case, that is, $C_{1,2,\ldots,7} = 0$
- Pattern 1: $\{C_1 = 7, C_2 = 4, C_3 = -1, C_4 = -5, C_5 = -2, C_6 = 3, C_7 = 7\}$
- Pattern 2: $\{C_1 = -6, C_2 = -3, C_3 = -1, C_4 = 2, C_5 = 5, C_6 = 7, C_7 = 4\}$
- Pattern 3: $\{C_1 = -7, C_2 = -5, C_3 = -3, C_4 = -1, C_5 = 1, C_6 = 4, C_7 = 6\}$
- Pattern 4: $\{C_1 = 6, C_2 = 3, C_3 = 1, C_4 = -2, C_5 = -5, C_6 = -7, C_7 = -4\}$

From the graphics, it is verified that when a control code $CtrlCal0 > 0$ is sent, transition shifts to the left (it becomes lower) and if $CtrlCal0 < 0$, transition location increases.

The INL measures the distortions in transitions and, consequently, a linear pattern should exhibit a constant measured INL for all transitions. This is the case of pattern 3, where the measured INL remains almost constant, as shown in Figure 6.18.

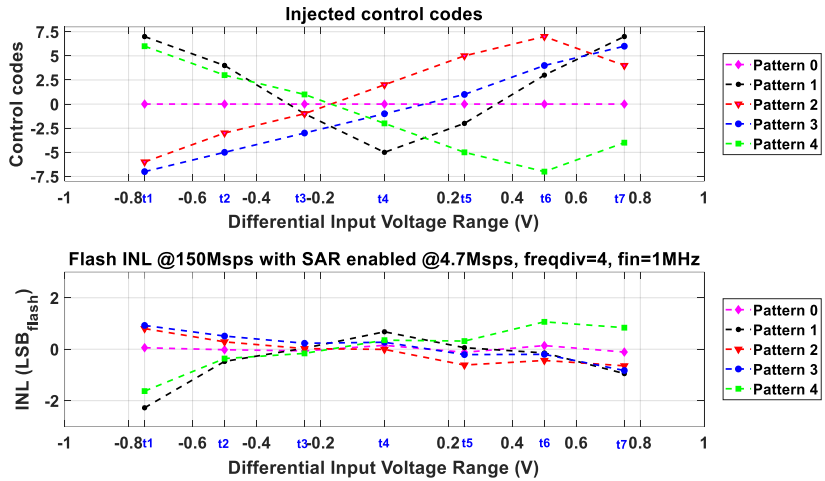

Figure 6.18. a) Different injected patterns and b) flash INL versus input voltage, with flash @150Msps, SAR @4.7Msps and input frequency of 1MHz

Moreover, Figure 6.19 gathers the flash INL for every transition of the flash considering different input frequencies (1MHz, 10MHz and 30MHz). Same conditions as before are considered: sampling frequency of flash and SAR are 150Msps and 4.7Msps, respectively. Similar results are obtained when SAR is disabled. Little discrepancies are found between transitions measured at different input frequencies.

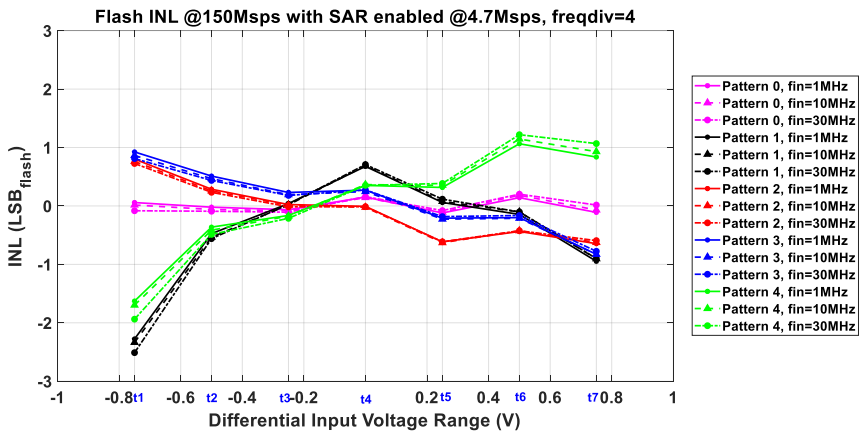

Figure 6.19. Flash INL versus input voltage for different patterns injected and input frequencies, with flash @150Msps and SAR @4.7Msps

6.4.4. Calibration of flash ADC

In flash calibration, the three main blocks are enabled (flash and SAR ADCs and calibration block COCL). Also, PLL is activated, due to its versatility. Calibration process is shown in Figure 6.20 and calibration diagram block is depicted in Figure 6.21. The whole procedure starts with the initial estimation of the flash transitions. Then, an external offset is applied to the comparators through $CtrlCal0$ bus. Therefore, transitions are shifted to new locations. Later, calibration is enabled (that is, SAR and COCL blocks are activated) and, after few seconds, flash output and control codes Ci are measured and evaluated, moving the transitions back to their initial locations. Some signals are visualized in real time. For instance, flash output code (FlashCode) is outputted in a parallel manner, while SAR output code is outputted in series (SARbitmon signal). In addition, control code evolution could be also checked in real time (through $Cmon$ signal) as well as the COCL activation signal ($enCOCLmon$) (see Figure 5.27).

In calibration process, in order to obtain a correct performance, SAR sampling frequency must be chosen below 7 Msps, as explained in Section 6.4.2.

ENOB and transitions are evaluated with histogram method for different input and clock frequencies. Furthermore, real positions of flash transitions have been also manually measured.

In the following subsections, the flash tuning curves are shown. Also, ENOB and INL performance before and after calibration is applied. Moreover, dynamic evolution of transitions and ENOB are studied. Finally, different values for COCL parameters are analyzed.

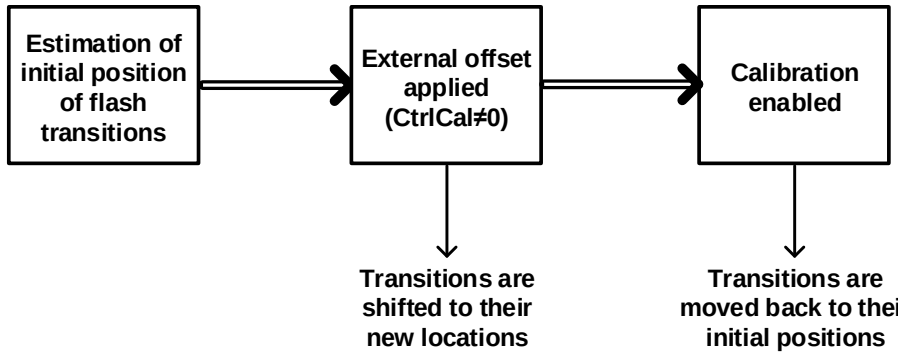

Figure 6.20. Calibration process

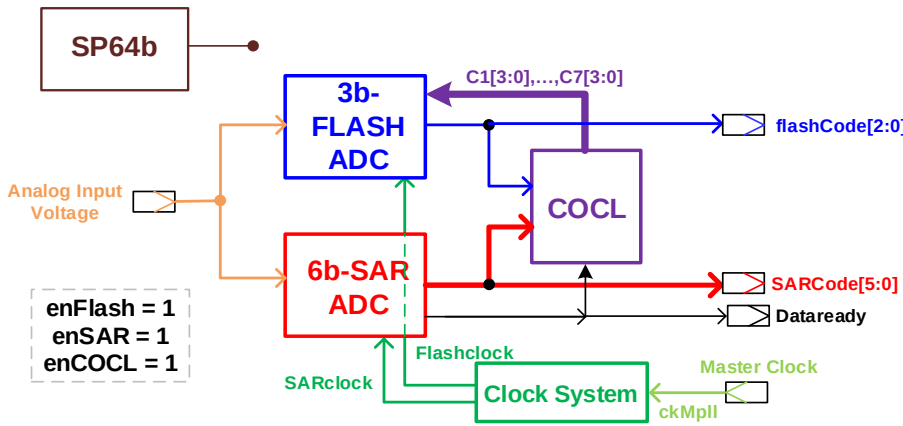

Figure 6.21. Calibration of flash ADC

Tunning curves

In Figure 6.22 tuning curves are depicted. Tuning curves represent transitions shifts with respect to their ideal values versus the different control codes. All different control codes for each of the transitions in flash ADC have been applied and new transitions' locations measured. The figure shows flash tuning curves at 100Msps, SAR sampling frequency of 6.3MSps (frequency divider set to 2) and input frequency of 1MHz. The obtained tuning curves could be compared with the tuning curves obtained in post-layout simulations (see Figure 5.14).

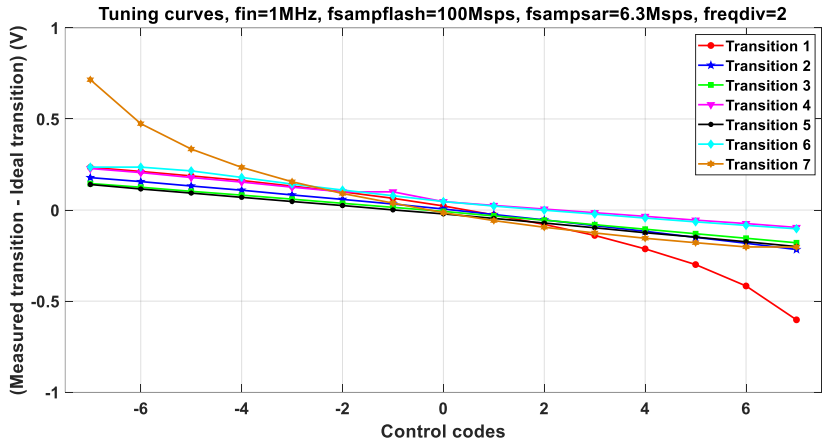

Figure 6.22. Flash tuning curves @100Msps and SAR @6.3MSps for input frequency of 1MHz

Observed is that extremal transitions (1 and 7) are not linear as separating from the nominal control code $C = 0$. In these extremal control codes the flash ADC is forced due to its design and circuit could not be working properly since transitions are far from their ideal values and out of range (from -1V to 1V by definition in design). Fortunately, calibration is able to correct the offset of these transitions, approaching them to their ideal values.

INL and ENOB performance

Figure 6.23 shows the flash linearity performance at 150 Msps with an input frequency of 1MHz and SAR ADC enabled with sampling frequency of 4.7 Msps. The graphic above shows the INL error of each flash comparator when the pattern 1 error (from Figure 6.18) is injected in the comparators. The figure below shows the error obtained after applying the calibration method and correcting the injected offset. Also, calibration codes achieved after calibrating the offset are also represented for each transition. It is observed that before calibration INL error achieved almost $\pm 2LSB_{flash}$, while after applying calibration, the maximum value for INL is $0.07LSB_{flash}$. This rather good performance of this offset calibration method improves flash accuracy and speed, by relaxing the architecture complexity design. Also, both power consumption and area are lessened due to the simpler architecture. Similar results are obtained for input frequencies of 1MHz, 10MHz and 60MHz.

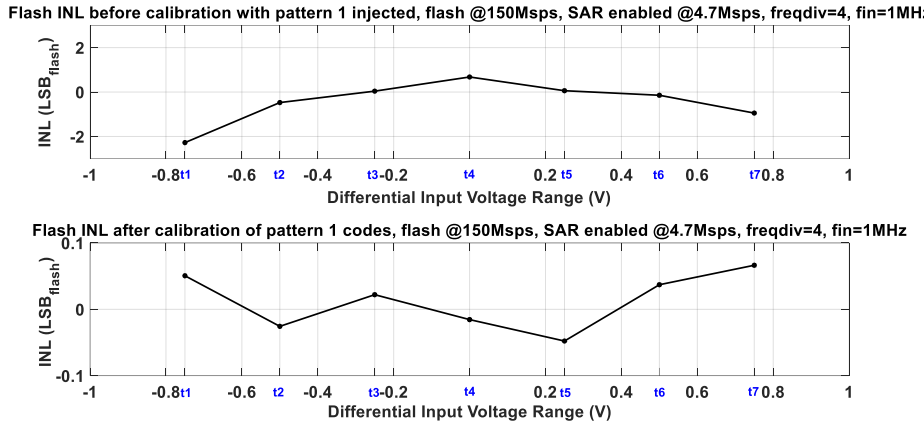

Figure 6.23. Flash INL before and after calibration when pattern 1 was injected in flash transitions @150Msps with SAR ADC enabled @4.7Msps and input frequency of 1MHz

The obtained control words after calibration are $\{C_1 = -7, C_2 = -4, C_3 = 0, C_4 = 7, C_5 = 1, C_6 = -1, C_7 = -7\}$, as shown in the figure. When introducing a control word $C_1 = 7$ in transition 1, this transition is shifted to a new location, where it reaches an INL of more than $2LSB_{flash}$. Then, calibration method starts and moves the transition back to its ideal position applying a control code of $C_1 = 7$, opposite to the injected code. Same procedure occurs with the other transitions.

The control words obtained with calibration process are robust if they remain the same and if the ENOB does not experiment a significant variation when input frequency changes. Figure 6.24 depicts the flash ENOB versus input frequency for flash at 100Msps and SAR at 6.25Msps. Results for uncalibrated pattern 1 are shown in blue colour and ENOB for the calibrated pattern 1 is depicted in green. Also, flash ENOB of the integration by default (when all control words are $C = 0$) is depicted in pink.

Control codes obtained after calibrating pattern 1 injection at different input frequencies have always the same values below 31MHz. Note that up to 31MHz, flash ENOB keeps below 2 bits when pattern 1 is applied (uncalibrated), while the ENOB value remains greater than 2.9 once the calibration method is employed. This way, the robustness of the control words is successfully proven when input frequency experiences variations. Moreover, calibrated offset leads to a better ENOB that chip default state without any calibration applied.

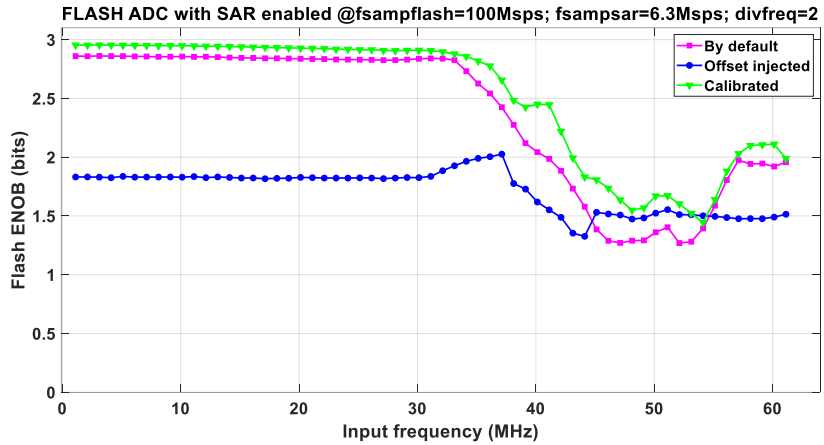

Figure 6.24. Flash ENOB of default state, with patter 1 applied without calibration and with calibration with sampling frequency of 100Msps and SAR enabled working at 6.33Msps (frequency divider set to 2), with PLL ON and symmetric sampling clock with comparison pulse of 10ns

As shown in the Figure 6.7 and Figure 6.8, flash ENOB decreases as input frequency increases. The reason for this loss was considered a dynamic effect associated to symmetric clock. However, the fact that flash is not calibrated is also affecting the flash performance, as we can see in Figure 6.24. It can be observed that ENOB improves when calibration is applied. Moreover, the dynamic problems are still exhibited in the graphics for frequencies greater than 30MHz.

Different offset patterns in Table 6.1 are applied to the flash while working in background calibration mode and the flash ENOB has been measured after calibration in Figure 6.25. In the graphic, the ENOBs of the uncalibrated flash for the different tests have been also depicted.

As shown in Figure 6.25, in a background mode calibration, different offset patterns have been applied to the circuit, giving rather good results, since ENOB is always above 2.9 bits when flash is calibrated.

Table 6.1. Different offset patterns injected in flash ADC

Test	$C_1, C_2, \ldots, C_7$
1	{-2, 3, 7, -1, -6, 5, 1}
2	{-2, 3, 7, -2, -6, 5, 1}
3	{2, 1, 4, -1, -4, 3, 2}
4	{4, -6, -3, 5, 2, -4, -3}
5	{-3, 7, -6, -4, 7, -1, -5}
6	{0, 0, 0, 0, 0, 0, 0}
7	{0, -4, 5, 7, 3, 1, 4}
8	{-4, -1, -4, 3, 0, 6, -2}
9	{1, -5, 0, -7, 5, 3, -7}
10	{5, 2, -5, 6, -1, -5, 0}
11	{-1, -3, 1, -3, -5, 0, -1}

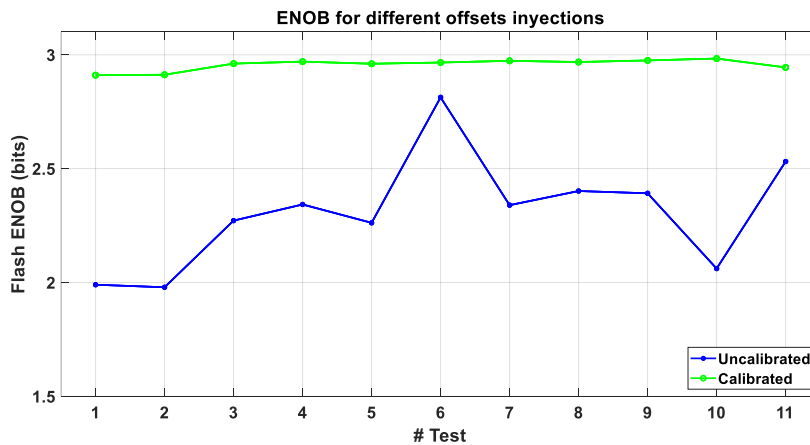

Figure 6.25. Measured flash ENOB for different offset pattern injections before (uncalibrated) and after (calibrated) calibration is applied.

Dynamic evolution of ENOB and transitions

In order to see the dynamic evolution of the flash transitions, pattern of test 1 in Table 6.1 has been considered: $\{C_1 = -2, C_2 = 3, C_3 = 7, C_4 = -1, C_5 = -6, C_6 = 5, C_7 = 1\}$. This pattern is injected into the circuit through external $CtrlCal0$ bus. The circuit now has an offset, which must be corrected with the calibration method. Calibration performance starts when signal $enCOCL$ goes high. Figure 6.26 shows the evolution of the different transitions of the flash during calibration process. Also, the evolution of the ENOB and the gain of the flash are both shown.

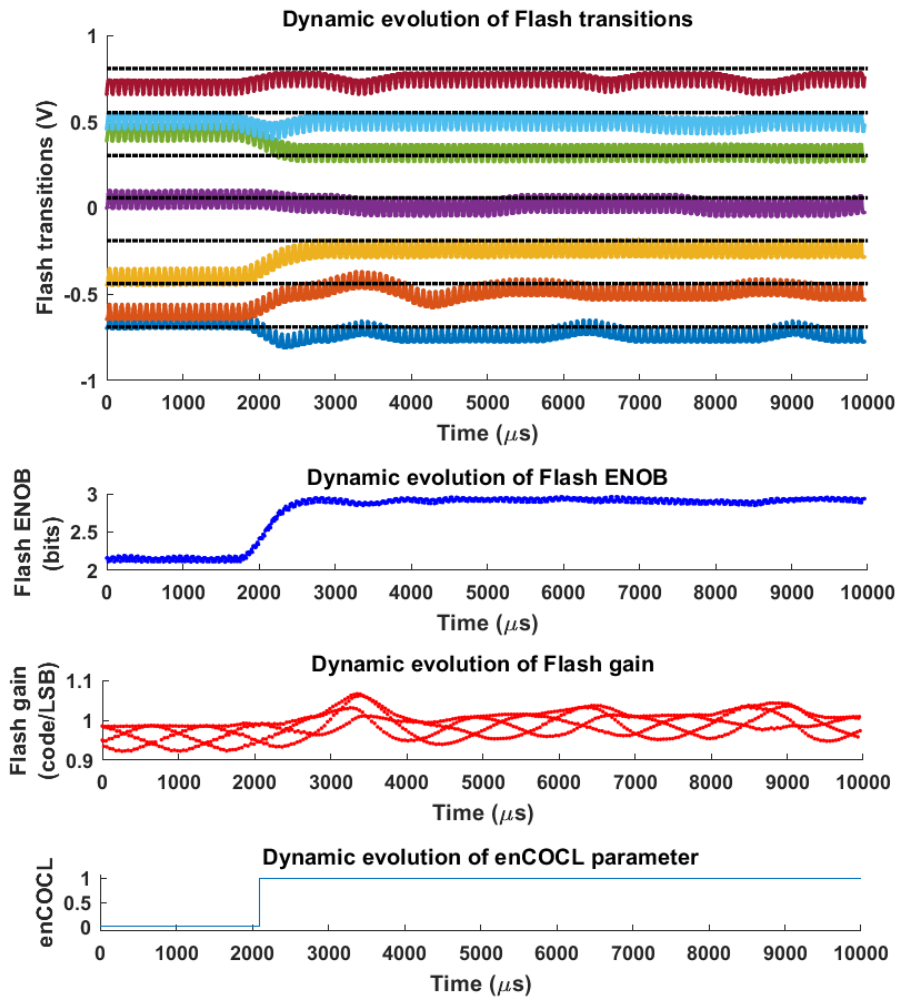

Figure 6.26. Dynamic evolution of transitions, ENOB and gain of flash when offset pattern of test 1 is injected and calibration is activated ($enCOCL = 1$).

At the beginning of the experiment, the offset is applied and transitions are shifted to new locations, having an ENOB of 2.1 bits (uncalibrated flash). After certain time (at 2092μs in the graphic), the $enCOCL$ signal is set to high and calibration method starts. Then, the transitions go back to their ideal positions and the ENOB achieves 2.9 bits, which means the offset has been corrected and the flash has a good performance again. Moreover, the flash gain does not suffer from significant changes.

Now dynamic evolution of calibration is studied. In order to prove that calibration is stable in time, consider calibration activated (that is, parameter $enCOCL = 1$) and test 11 (see Table 6.1) applied. At the beginning, calibration is not working and signal $enCOCL$ is set to 0, having the flash an ENOB of 2.5 bits. When calibration starts ($enCOCL$ changes to 1), ENOB achieves a value of 2.95 bits. Figure 6.27 shows the stability of the flash ENOB while calibration is activated (background mode). ENOB has been measured many times during one day and remains stable at 2.95 bits after 24 hours. Note that calibration does not influence the ENOB performance and, in case voltage and temperature variations affected the circuit, they have been corrected with calibration method. Also, the algorithm used in calibration method is stable and does not degrade in time.

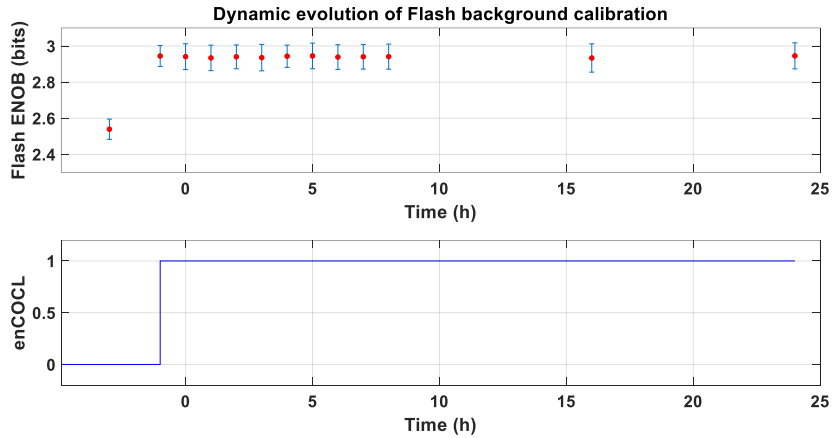

Figure 6.27. Time evolution of background calibration.

Assume now that after calibration of pattern of test 11 (see Table 6.1), calibration is turn off (signal $enCOCL$ goes low). What happens now with the calibrated flash ADC after a certain time? Figure 6.28 represents the ENOB of the flash when pattern of test 11 is applied and the $enCOCL$ signal evolution. First, the pattern 11 is forced and flash ENOB goes to 2.5 bits when calibration is off. After activating calibration ($enCOCL$ signal to 1), ENOB goes high up to 2.95 bits. Then, calibration is again disconnect and ENOB has been measured during the next 24 hours. The graphic shows a stable ENOB, which means that circuit is always corrected, even after a long time and even if calibration is switched off.

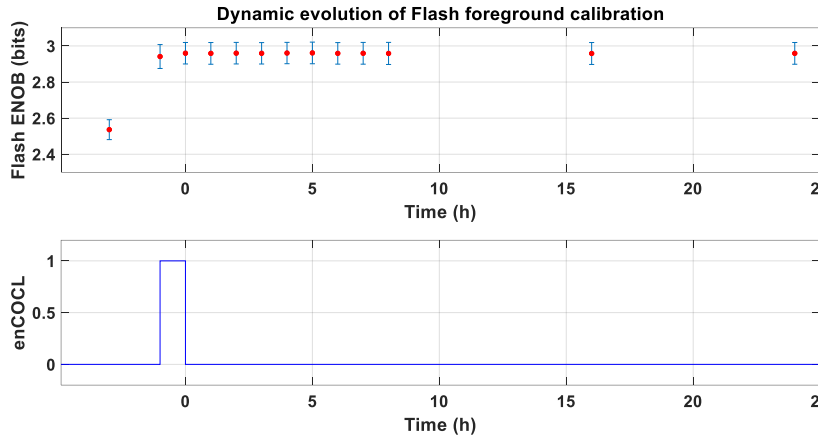

Figure 6.28. Time evolution of flash ENOB without background calibration activated.

Since 24 hours could be not enough time to see the influence of voltage and temperature variations on the calibrated circuit, we are going to speed up this process to uncalibrate the circuit. A way to do this is forcing an internal variation in the circuit, in this case, by varying the internal reference (REF) of the SAR. Remember that external REF is set to 1.3V. Two variations of REF have been considered: REF1 = REF – 15 mV and REF2 = REF + 15 mV. Figure 6.29 depicts the evolution of ENOB when REF has been modified when background calibration is activated, that is, $enCOCL$ signal remains high. At the beginning, calibration is disconnected ($enCOCL$ signal is low) SAR reference (REF) has a value of 1.3V and pattern 11 (see Table 6.1) is applied. Flash ENOB and gain achieve 2.54 bits and 0.96 LSB/code, respectively. Before time 0 h, calibration is activated ($enCOCL$ signal is changed to 1) and offset is corrected, leading to an ENOB value of 2.95 bits and a gain of 1 LSB/code. ENOB and gain have been measured during 24 hours and their values have not changed significantly. Then, SAR reference REF has been modified to value REF1 (REF – 15 mV = 1.285 mV) and both ENOB and gain measured during 24 hours. Later, REF has been changed to REF2 (REF + 15 mV = 1.315 mV) and measurements have been taken again. In this test, calibration is continually working, that is, $enCOCL$ signal is always set to 1. Therefore, ENOB is continuously corrected and it remains stable around 2.95 bits, as shown in Figure 6.29. When background calibration is activated, a variation of SAR reference changes the transitions regularly, that is, INL error remains constant and ENOB should be always the same. On the contrary, flash gain is affected by SAR reference variation, as depicted in Figure 6.30. Hence, a change

in flash gain do not bring to a distortion error.

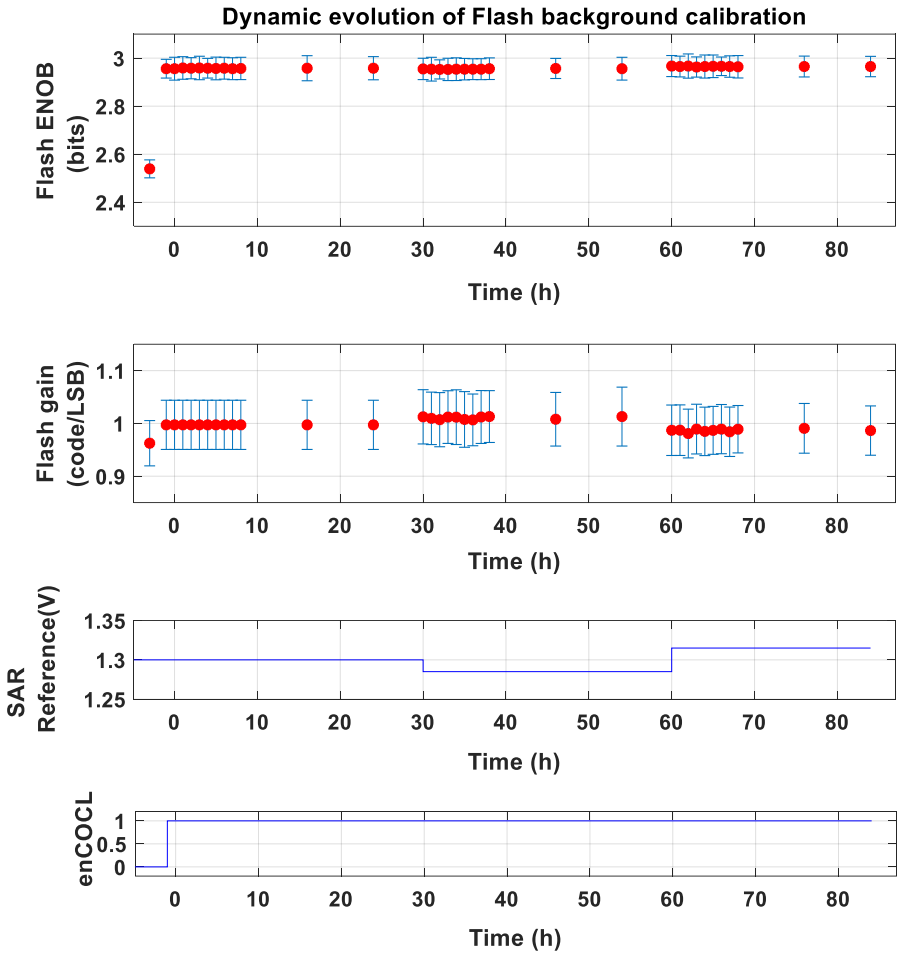

Figure 6.29. Time evolution of background calibration when an internal variation is forced in the circuit.

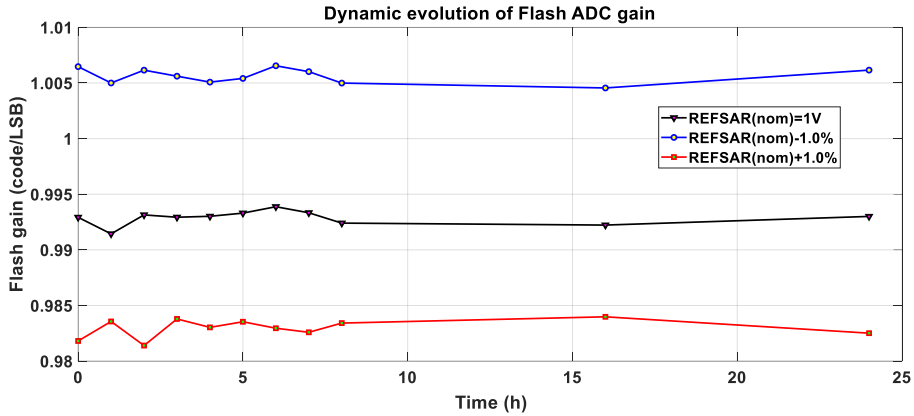

Figure 6.30. Time evolution of flash ADC gain.

COCL parameters

COCL block has been designed with great programmability, as explained in Section 5.6. The programmable parameters are:

- Δ_{up} and Δ_{down}: After calibration, offset must be $-\Delta_{down} < OFF < \Delta_{up}$, taking Δ_{down} and Δ_{up} values from 0 to 3. By default, $\Delta_{down} = -2$ and $\Delta_{up} = 1$.
- $pcal[1:0]$: It allows increasing the internal processing arithmetic resolution from 6 to 8 bits. By default, $pcal = 8\ bits$.
- $selEQ$: It selects the algorithm limits: If $selEQ = 0, SARCode \in (T_k, T_k + 1)$, and when $selEQ = 1$, $SARCode \in [T_k, T_k + 1)$, being T_k the SAR k-transition. By default, $selEQ = 1$.
- $shiftX$: When $shiftX = 1$ (default value), an offset is added to $SARCode$ and the waveform is shifted an amount of $LSB/2$ to the right, being the errors in the interval $[-LSB/2\ , LSB/2]$. With $shiftX = 0$, $SARCode$, is centered, obtaining errors in the interval $[0, LSB]$.

In the following test, nominal pattern ($C_{1,2,\dots,7} = 0$) has been injected in the circuit. COCL parameters have been set and calibration is turned on. The dynamic evolution of flash ENOB and their transitions have been studied for the different parameters values in order to find the best performance of the flash calibration. In this test, flash is working at 10 Msps, PLL is off and frequency divider is set to 2, being the SAR frequency of 630

ksps. A low input frequency (12 kHz) has been considered. Table 6.2 gathers the different parameters values used in each test. From Figure 6.31 to Figure 6.34 the dynamic evolution of flash transitions and ENOB are depicted for the different values of the parameters shown in Table 6.2. Not every case has been represented, since similar results as depicted in Figure 6.34 are found for the rest of cases in Table 6.2.

Note that calibration is activated at time 1700μs, and from this moment, ENOB and transitions are corrected, bringing them back to their ideal values. The evolution of transitions gives an idea of the stability of the calibration and in the graphic of the ENOB evolution the settling time of the calibration could be observed.

From the graphics above it can be concluded calibration is working properly, regardless of the value of the different parameters. However, there are some parameters which make calibration slightly faster and others do have influence in settling time, making it greater.

For instance, parameter $shiftX = 1$ shows more stability in the graphic of transitions evolution, since less variability is observed than in graphic with $shiftX = 0$ (see Figure 6.31). Also, overshooting in ENOB evolution appears when $shiftX = 0$. Hence, calibration is slower than using $shifX = 1$.

Moreover, in Figure 6.32 $selEQ$ parameter exhibits more variability in transitions when it has high value ($selEQ = 1$). However, in ENOB evolution less overshooting is observed and, hence, calibration is faster than using $selEQ = 0$.

In addition, Figure 6.33 shows for $pcal = 0$ no overshooting in ENOB evolution when calibration starts. However, with $pcal = 1$, a slight overshooting appears, and with $pcal = 2$ (maximum resolution), the peak seems to be overcompensated. Then, as $pcal$ increases, the calibration goes slower. Also, transitions evolution are less stable as $pcal$ rises.

Furthermore, as shown in Figure 6.34, transitions with $\Delta_{down} = 3$ immediately become stable (transition 7 has more variability than the rest). Using $\Delta_{down} = 2$ the stability of transitions last more time. With Δ_{down} of 1 or 0 transitions take even more time to stabilize, due to their variations. That means, as Δ_{down} decreases, transitions are less stable. Further, ENOB evolution seems to be slower and has more overshooting for low Δ_{down} values. Indeed, with high Δ_{up}, transitions are more stable. On the other hand, the ENOB shows that for high Δ_{up} calibration takes longer than for low values.

Table 6.2. COCL different parameters employed in test

Case	Δ_{up}	Δ_{down}	$pcal$	$selEQ$	$shiftX$
1 (default)	1	2	2	1	1
2	1	2	2	1	0
3	1	2	2	0	1
4	1	2	1	1	1
5	1	2	0	1	1
6	1	3	2	1	1
7	1	1	2	1	1
8	1	0	2	1	1
9	2	3	2	1	1
10	2	2	2	1	1
11	2	1	2	1	1
12	2	0	2	1	1
13	3	3	2	1	1
14	3	2	2	1	1
15	3	1	2	1	1
16	3	0	2	1	1
17	0	3	2	1	1
18	0	2	2	1	1
19	0	1	2	1	1
20	0	0	0	1	1

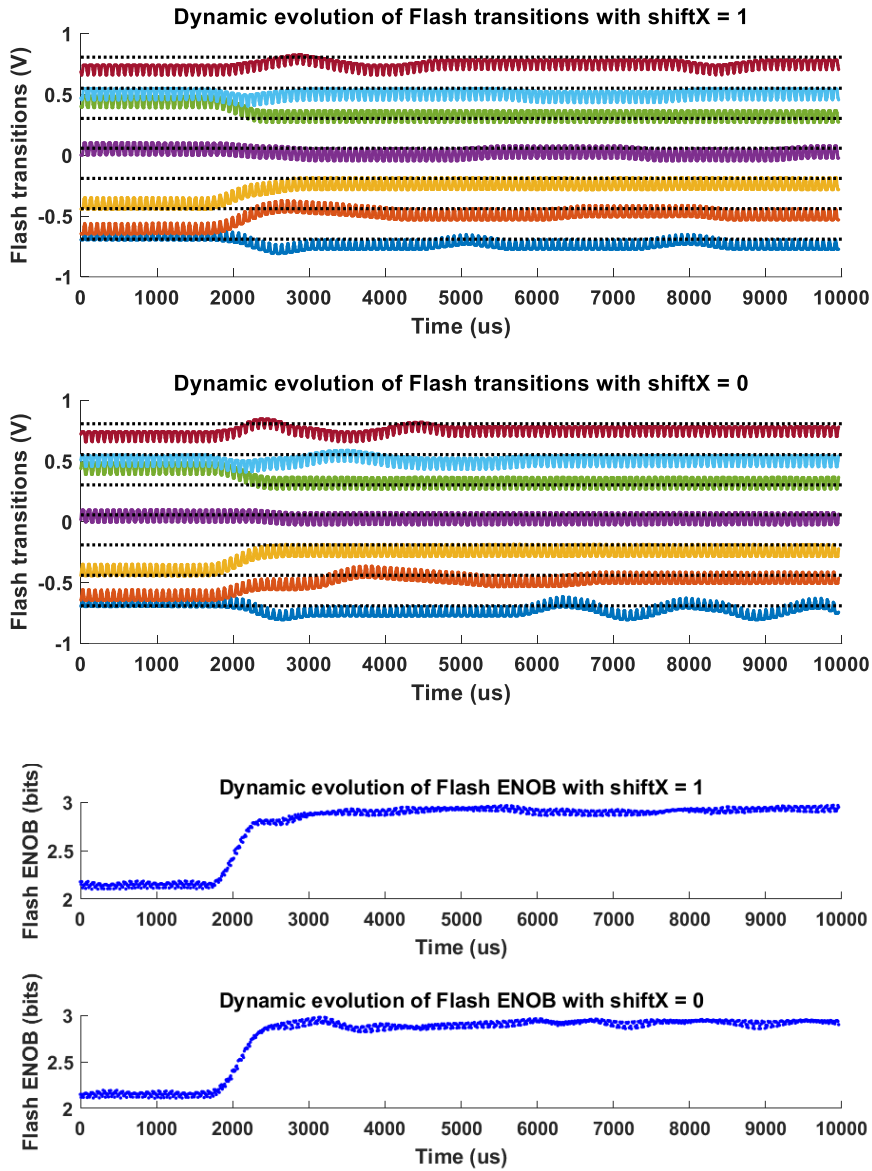

Figure 6.31. Dynamic evolution of transitions (above) and ENOB (below) for $shiftX$ *values 1 and 0, with parameters* $\Delta_{up} = 1, \Delta_{down} = 2, pcal = 2, selEQ = 1$ *(cases 1 and 2).*

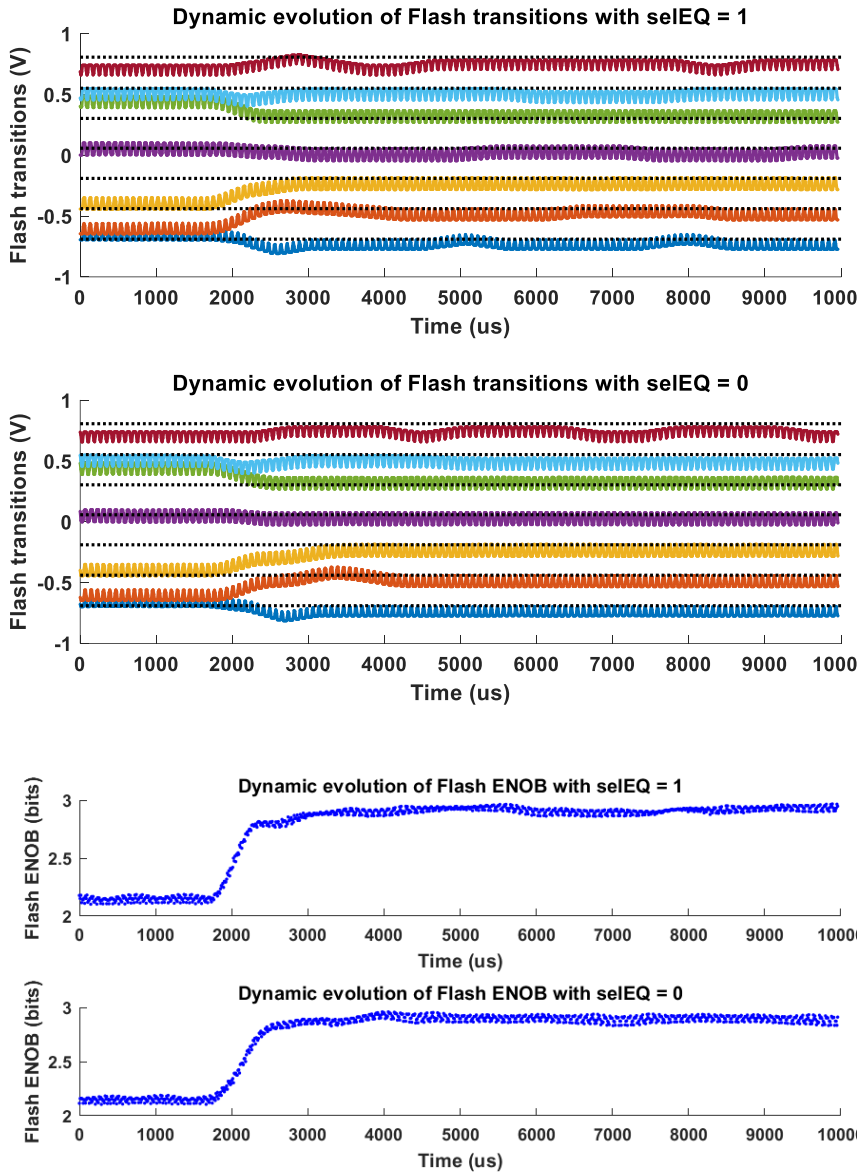

Figure 6.32. Dynamic evolution of transitions (above) and ENOB (below) for $selEQ$ *values 1 and 0, with parameters* $\Delta_{up} = 1, \Delta_{down} = 2, pcal = 2, shiftX = 1$ *(cases 1 and 3)*

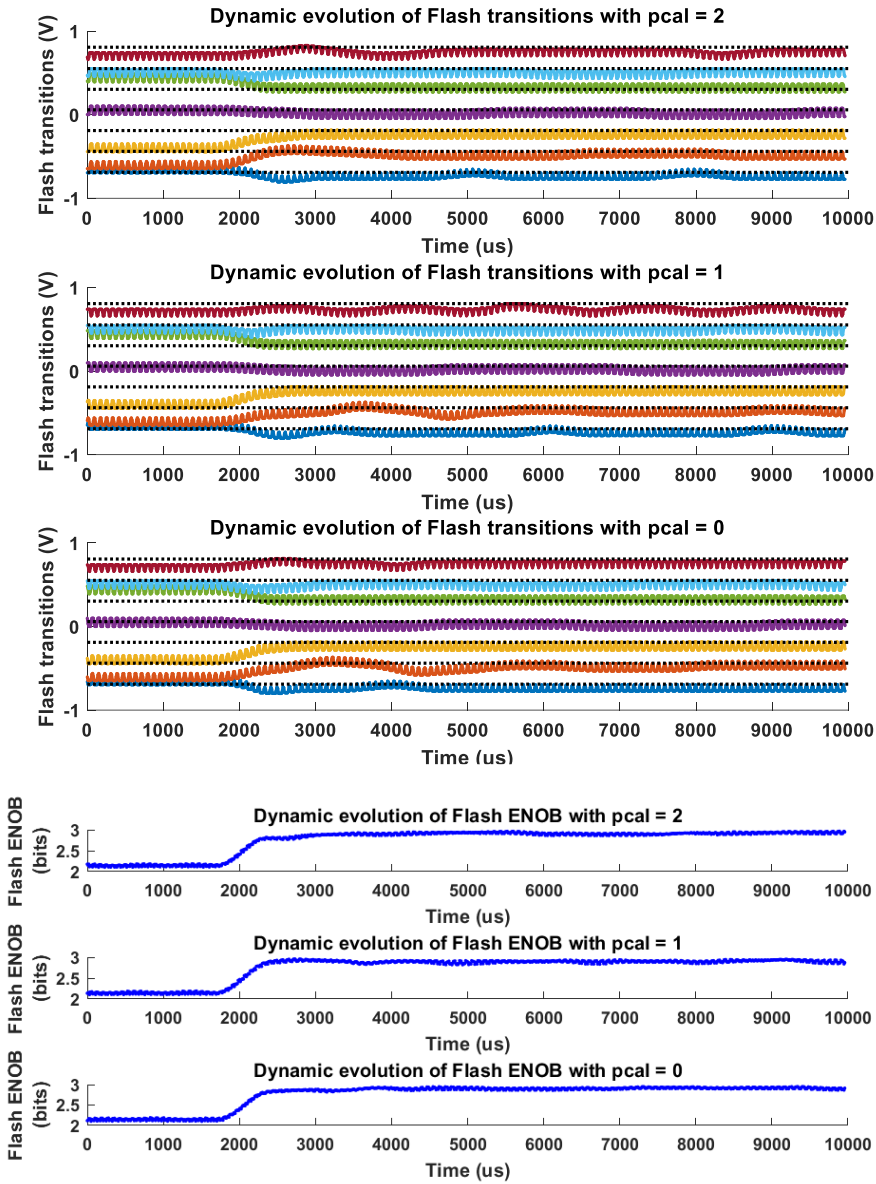

Figure 6.33. Dynamic evolution of transitions (above) and ENOB (below) for $pcal$ *values 2, 1 and 0, with parameters* $\Delta_{up} = 1, \Delta_{down} = 2, selEQ = 1, shiftX = 1$ *(cases 1, 4 and 5).*

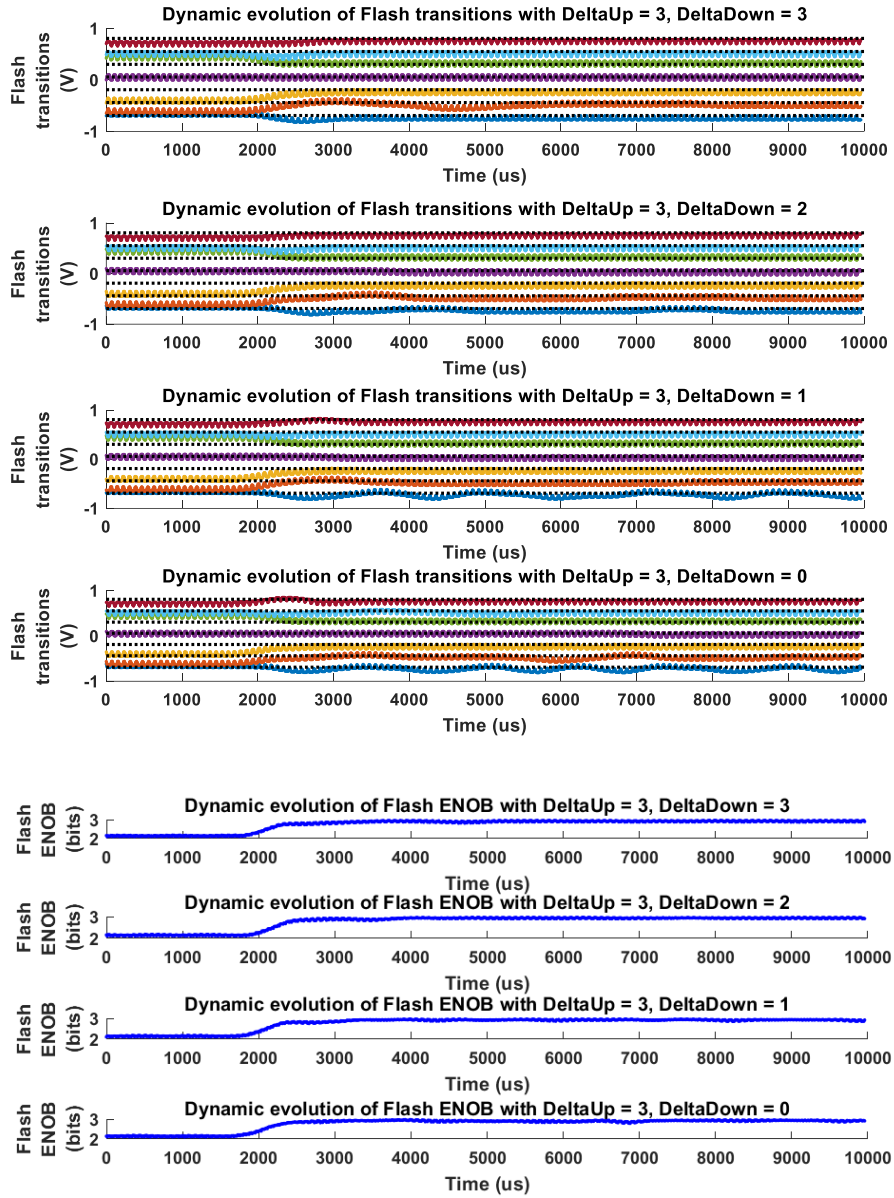

Figure 6.34. Dynamic evolution of transitions (above) and ENOB (below) for Δ_{down} *values 3, 2, 1 and 0, with parameters* $\Delta_{up} = 3,\ pcal = 2, selEQ = 1, shiftX = 1$ *(cases 13, 14, 15 and 16).*

All the above conclusions are gathered in Table 6.3. The table summarize the performance of the transition and ENOB dynamic evolution for the different parameters as the values of the parameters increase. For example, for high $shiftX$ values, that is, $shiftX = 1$, the graphic of dynamic evolution of transitions shows more stable transitions, with less variability, than those in graphic with $shiftX = 0$.

Table 6.3. Summary of parameters values performance as parameter value increases

	$shiftX$	$selEQ$	$pcal$	Δ_{up}	Δ_{down}
More stable **transitions**, less variability in graphic	X			X (except for transition 7)	X
Less stable **transitions**, more variability		X	X		
Faster **ENOB** evolution, less variability, less overshooting in graphic	X	X			X
Slower **ENOB** evolution, more variability, higher overshooting			X	X	

In conclusion, all different values of COCL parameters give good results in calibration. For a proper, faster calibration, using the parameters shown in Table 6.4 is highly recommended, which differs from the default state.

Table 6.4. Parameters choice

Parameter	Default	Best case	Best case explanation
$shiftX$	1	1	Symmetrical quantization error
$selEQ$	1	1 or 0	$SARCode \in (T_k, T_k + 1)$ or $[T_k, T_k + 1)$
$pcal$	2	0	Internal processing arithmetic resolution of 6 bits
Δ_{up}	1	2 or 1	$-3 < OFF < 2$
Δ_{down}	2	3	

7. CONCLUSIONS AND FUTURE WORK

In this chapter the main conclusions arisen from this thesis work are given in Section 7.1. Some of the analyses, studies and developments that remains to be covered are summarized in Section 7.2 as well as those that could create by their own new lines of research.

7.1. Conclusions

- Amongst the different techniques for offset calibration of comparators in flash A/D converters, those converters that employ digital adaptive technique have been analysed, with a special focus on background type methods.
- A specific background adaptive technique, initially proposed in bibliography in [105] to be implemented in pipeline A/D converters, has been deeply studied and analysed in order to apply it in flash ADCs with dynamic latch comparators and no external references.
- The electric performances of a set of seven different topologies for comparators based on dynamic latches in 180nm CMOS technology have been analysed. In this analysis, some parameters and figures-of-merit have been identified, which allow

selecting the most efficient topology for its integration in a bank of comparators of a flash ADC with no references and with the background calibration technique previously referred.

- A SAR type A/D converter with low speed and high frequency has been proposed as auxiliary ADC to perform the background calibration process. SAR ADC is suitable for this task thanks to its great compaction, low power consumption and reduced cost in terms of area and design, and its excellent level of precision, compared to typical resolutions of flash converters in CMOS technologies.
- A solution for achieving synchronization in the process of analogue sampling between flash A/D converter under calibration and the auxiliary SAR ADC has been presented, without the need to add a common sample-and-hold circuit to both blocks.
- A conversion system A/D flash type has been designed and implemented in an IC which works simultaneously with a background self-calibration process based on an adaptive technique that corrects the effective location of transitions of flash comparators. This adaptive technique operates with two interlaced feedback loops, one in charge of the offset estimation and the second one to cancel the offset.

  Implementation of the circuit has been developed in UMC 180nm CMOS technology. Converter under calibration is a 3-bit flash ADC without external references, full-scale of 2Volts and maximum sampling frequency of 500Msps. Auxiliary ADC is a 6-bit SAR ADC with full-scale of 2Volts and maximum frequency of 10Msps. Both converters work up to 300Msps of flash ADC with their sampling instants synchronized. In order to control the adaptive algorithm, a digital processor has been designed. This block estimate and correct the location of the 7 transitions of flash ADC in a parallel manner and it has been automatically synthetized. An external control signal allows introducing some variations in the calibration algorithm.
- The complete integrated system has been proved in laboratory with complete and satisfactory results.
- The experiments have demonstrated that the selected technique is valid and fully stable, even if external variations occur.

- The analysed technique and implemented system turn out to be rather efficient to be integrated in conversion applications with low/medium resolution and high speed, where great accuracy in the uniformity of quantization levels is required, such as in some digital communication systems.
- Finally, we want to highlight that the complete work done in this thesis has led us to a certain specialization in the design of integrated circuits, the test of mixed-signal systems and the better understanding of many of the problems that are currently presented in the state-of-the-art.

7.2. Future work

- Design of a A/D flash converter with higher resolution, 5 or 6 bits, with the same background calibration system, but some great differences are proposed:
    - Consider calibration of transitions one by one in a serial manner (not in parallel)
    - Incorporate a system to reorder effective transitions and, hence, employ a two-level-gate thermometric-to-binary encoder, avoid using a Wallace Tree type.
    - Employ a finer control adjustment. Instead of designing one calibration DAC for all flash comparator, a different DAC could be implemented for each of them. Also, if one comparator is eliminated from flash ADC structure, two different calibration banks could be designed, one for positive and one for negative transitions, both centred in the central transition of each group.

LIST OF FIGURES

LIST OF TABLES

List of Symbols and Acronyms

A/D	Analogue-to-digital
ADC	Analogue-to-digital converter
DL	Dynamic latch
DL-CS	Conventional dynamic latch with current source
DL-CS-INR	Dynamic latch with current source and internal nodes resets
DL-NoCS	Dynamic latch with no current source
DL-NoCS-INR	Dynamic latch with no current source and internal nodes reset switches
DL-NoCS-LS	Dynamic latch with no current source which incorporates switches to open the regenerative latch
DL-NoCS-LS2	Dynamic latch with no current source and switches to open the regenerative latch and regenerative latch NMOS in parallel to NMOS input
DNL	Differential non-linearity
ENOB	Effective number of bits
FFT	Fast Fourier Transform
FoM	Figure-of-merit
FS	Full-scale
INL	Integral non-linearity
LSB	Least-significant bit
MSB	Most-significant bit
PDL-CS	PMOS-input dynamic latch with current source
PLL	Phase-locked loop

SADL	Stand-alone dynamic latch
SC	Switched-capacitor
SFDR	Spurious-free dynamic range
SH	Sample and hold
SNDR or SINAD	Signal-to-noise-distortion ratio
SNR	Signal-to-noise ratio
THD	Total harmonic distortion

REFERENCES

[1] P. Desgreys, F. Ghanem, G. Pham, H. Fakhoury and P. Loumeau, "Beyond 3G wideband and high linearity ADCs," 2011 Faible Tension Faible Consommation (FTFC), pp. 59-62, Marrakech, 2011.

[2] M. Gustavsson, J. Jacob Winker and N. Nick Tan, CMOS Data Converters for Communications, Springer US, 2000.

[3] Y. M. Lin, B. Kim and P. R. Gray, "A 13-b 2.5-MHz self-calibrated pipelined A/D converter in 3-μm CMOS," IEEE Journal of Solid-State Circuits, vol. 26, no. 4, pp. 628-636, Apr 1991.

[4] S. T. Ryu, S. Ray, B. S. Song, G. H. Cho and K. Bacrania, "A 14-b linear capacitor self-trimming pipelined ADC," IEEE Journal of Solid-State Circuits, vol. 39, no. 11, pp. 2046-2051, Nov. 2004.

[5] Y. Chiu, "Inherently linear capacitor error-averaging techniques for pipelined A/D conversion," IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, vol. 47, no. 3, pp. 229-232, Mar 2000.

[6] K. Ohhata, H. Takase, M. Tateno, M. Arita, N. Imakake and Y. Yonemitsu, "A 1-GHz, 17.5-mW, 8-bit subranging ADC using offset-cancelling charge-steering amplifier," 2012 IEEE Asian Solid State Circuits Conference (A-SSCC), pp. 149-152, Kobe, 2012.

[7] Y. Chiu, P. R. Gray and B. Nikolic, "A 14-b 12-MS/s CMOS pipeline ADC with over 100-dB SFDR," IEEE Journal of Solid-State Circuits, vol. 39, no. 12, pp. 2139-2151, Dec. 2004.

[8] H. S. Chen, B. S. Song and K. Bacrania, "A 14-b 20-Msamples/s CMOS pipelined

ADC," IEEE Journal of Solid-State Circuits, vol. 36, no. 6, pp. 997-1001, Jun 2001.

[9] Y. Nakajima, A. Sakaguchi, T. Ohkido, N. Kato, T. Matsumoto and M. Yotsuyanagi, "A Background Self-Calibrated 6b 2.7 GS/s ADC With Cascade-Calibrated Folding-Interpolating Architecture," IEEE Journal of Solid-State Circuits, vol. 45, no. 4, pp. 707-718, April 2010.

[10] J. A. McNeill, M. C. W. Coln, D. R. Brown and B. J. Larivee, "Digital Background-Calibration Algorithm for "Split ADC" Architecture," IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 56, no. 2, pp. 294-306, Feb. 2009.

[11] H. S. Chen, K. Bacrania and B. S. Song, "A 14 b 20 MSample/s CMOS pipelined ADC," 2000 IEEE International Solid-State Circuits Conference. Digest of Technical Papers, pp. 46-47, San Francisco, CA, USA, 2000.

[12] B. S. Song, M. F. Tompsett and K. R. Lakshmikumar, "A 12-bit 1-Msample/s capacitor error-averaging pipelined A/D converter," IEEE Journal of Solid-State Circuits, vol. 23, no. 6, pp. 1324-1333, Dec 1988.

[13] M. Ding, P. Harpe, Y. H. Liu, B. Busze, K. Philips and H. Groot, "A 46uW 13 b 6.4 MS/s SAR ADC With Background Mismatch and Offset Calibration," IEEE Journal of Solid-State Circuits, vol. 52, no. 2, pp. 423-432, Feb. 2017.

[14] C. Shih and P. R. Gray, "Reference refreshing cyclic analog-to-digital and digital-to-analog converters," IEEE Journal of Solid-State Circuits, vol. 21, no. 4, pp. 544-554, Aug 1986.

[15] S. Sutarja and P. R. Gray, "A pipelined 13-bit 250-ks/s 5-V analog-to-digital converter," IEEE Journal of Solid-State Circuits, vol. 23, no. 6, pp. 1316-1323, Dec 1988.

[16] K. Dyer, D. Fu, S. Lewis and P. Hurst, "Analog background calibration of a 10 b 40 Msample/s parallel pipelined ADC," 1998 IEEE International Solid-State Circuits Conference. Digest of Technical Papers, ISSCC. First Edition, pp. 142-143, San Francisco, CA, USA, 1998.

[17] J. M. Ingino and B. A. Wooley, "A continuously calibrated 12-b, 10-MS/s, 3.3-V A/D converter," IEEE Journal of Solid-State Circuits, vol. 33, no. 12, pp. 1920-1931, Dec 1998.

[18] F. Rabuske, T. Rabuske and J. Fernandes, "A 5-bit 300–900-MS/s 0.8–1.2-V Supply Voltage ADC With Background Self-Calibration," IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 64, no. 1, pp. 1-5, Jan. 2017.

[19] B. Razavi, Principles of Data Conversion System Design, Wiley-IEEE Press, 1994.

[20] G. Miller, M. Timko, H. S. Lee, E. Nestler, M. Mueck and P. Ferguson, "An 18 b 10 mu s self-calibrating ADC," 1990 37th IEEE International Conference on Solid-State

Circuits, pp. 168-169, San Francisco, CA, USA, 1990.

[21] S. C. Dong and B. S. Carlson, "A 10-bit pipelined ADC for high speed, low power applications," Conference Record of the Thirty-First Asilomar Conference on Signals, Systems and Computers (Cat. No.97CB36136), no. 1, pp. 744-748, Pacific Grove, CA, USA, 1997.

[22] S. Sheikhaei, S. Mirabbasi and A. Ivanov, "A 4-bit 5 GS/s flash A/D converter in 0.18 μm CMOS," 2005 IEEE International Symposium on Circuits and Systems, vol. 6, pp. 6138-6141, 2005.

[23] K. Deguchi, N. Suwa, M. Ito, T. Kumamoto and T. Miki, "A 6-bit 3.5-GS/s 0.9-V 98-mW Flash ADC in 90-nm CMOS," IEEE Journal of Solid-State Circuits, vol. 43, no. 10, pp. 2303-2310, Oct. 2008.

[24] F. Maloberti, Data Converters, Pavia University: Springer, 2007.

[25] S. M. Jamal, D. Fu, N. C. J. Chang, P. J. Hurst and S. H. Lewis, "A 10-b 120-Msample/s time-interleaved analog-to-digital converter with digital background calibration," IEEE Journal of Solid-State Circuits, vol. 37, no. 12, pp. 1618-1627, Dec 2002.

[26] D. Fu, K. C. Dyer, S. H. Lewis and P. J. Hurst, "A digital background calibration technique for time-interleaved analog-to-digital converters," IEEE Journal of Solid-State Circuits, vol. 33, no. 12, pp. 1904-1911, Dec 1998.

[27] F. Lang, M. Grözing and M. Berroth, "Design of a 4 GS/s radix-1.75 single channel pipeline ADC in 28 nm CMOS technology with foreground calibration," 2014 International Symposium on Integrated Circuits (ISIC), pp. 87-90, Singapore, 2014.

[28] J. Li and U. K. Moon, "Background calibration techniques for multistage pipelined ADCs with digital redundancy," IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, vol. 50, no. 9, pp. 531-538, Sept. 2003.

[29] H. S. Lee, "A 12-b 600 ks/s digitally self-calibrated pipelined algorithmic ADC," IEEE Journal of Solid-State Circuits, vol. 29, no. 4, pp. 509-515, Apr 1994.

[30] A. N. Karanicolas, H.-S. Lee and K. L. Barcrania, "A 15-b 1-Msample/s digitally self-calibrated pipeline ADC," IEEE Journal of Solid-State Circuits, vol. 28, no. 12, pp. 1207-1215, Dec 1993.

[31] E. J. Peralías Macías, "Autocalibración digital y autotest en convertidores A/D multietapa," Ph. D. Thesis (only in Spanish), University of Seville, Spain, July 1999.

[32] W. Ellersick, C. K. K. Yang, M. Horowitz and W. Dally, "GAD: A 12-GS/s CMOS 4-bit A/D converter for an equalized multi-level link," 1999 Symposium on VLSI Circuits. Digest of Papers (IEEE Cat. No.99CH36326), pp. 49-52, Kyoto, Japan, 1999v.

[33] P. M. Figueiredo , P. Cardoso, A. Lopes, C. Fachada, N. Hamanishi, K. Tanabe and J.

Vital, "A 90nm CMOS 1.2v 6b 1GS/s two-step subranging ADC," 2006 IEEE International Solid State Circuits Conference - Digest of Technical Papers, pp. 2320-2329, San Francisco, CA, 2006.

[34] O. E. Erdogan, P. J. Hurst and S. H. Lewis, "A 12-b digital-background-calibrated algorithmic ADC with -90-dB THD," IEEE Journal of Solid-State Circuits, vol. 34, no. 12, pp. 1812-1820, Dec 1999.

[35] W. Liu and Y. Chiu, "An equalization-based adaptive digital background calibration technique for successive approximation analog-to-digital converters," 2007 7th International Conference on ASIC, pp. 289-292, Guilin, 2007.

[36] S. H. Lee and B. S. Song, "A direct code error calibration technique for two-step flash A/D converters," IEEE Transactions on Circuits and Systems, vol. 36, no. 6, pp. 919-922, Jun 1989.

[37] S. U. Kwak, B. S. Song and K. Bacrania, "A 15-b, 5-Msample/s low-spurious CMOS ADC," IEEE Journal of Solid-State Circuits, vol. 32, no. 12, pp. 1866-1875, Dec 1997.

[38] B. Murmann, "A/D converter circuit and architecture design for high-speed data communication," Proceedings of the IEEE 2013 Custom Integrated Circuits Conference, pp. 1-78, San Jose, CA, 2013.

[39] C. Y. Chen, M. Q. Le and K. Y. Kim, "A Low Power 6-bit Flash ADC With Reference Voltage and Common-Mode Calibration," IEEE Journal of Solid-State Circuits, vol. 44, no. 4, pp. 1041-1046, April 2009.

[40] M. El-Chammas and B. Murmann, "A 12-GS/s 81-mW 5-bit Time-Interleaved Flash ADC With Background Timing Skew Calibration," IEEE Journal of Solid-State Circuits, vol. 46, no. 4, pp. 838-847, April 2011.

[41] M. P. Flynn, C. Donovan and L. Sattler, "Digital calibration incorporating redundancy of flash ADCs," IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, vol. 50, no. 5, pp. 205-213, May 2003.

[42] C. Donovan and M. P. Flynn, "A "digital" 6-bit ADC in 0.25-μm CMOS," IEEE Journal of Solid-State Circuits, vol. 37, no. 3, pp. 432-437, Mar 2002.

[43] D. C. Daly and A. P. Chandrakasan, "A 6b 0.2-to-0.9V Highly Digital Flash ADC with Comparator Redundancy," 2008 IEEE International Solid-State Circuits Conference - Digest of Technical Papers, pp. 554-635, San Francisco, CA, 2008.

[44] I. Bogue and M. P. Flynn, "A 57 dB SFDR digitally calibrated 500 MS/s folding ADC in 0.18 μm digital CMOS," 2007 IEEE Custom Integrated Circuits Conference, pp. 337-340, San Jose, CA, 2007.

[45] G. Keskin, J. Proesel, J. O. Plouchart and L. Pileggi, "Exploiting Combinatorial Redundancy for Offset Calibration in Flash ADCs," IEEE Journal of Solid-State

Circuits, vol. 46, no. 8, pp. 1904-1918, Aug. 2011.

[46] T. Ogawa, H. Kobayashi, M. Hotta, Y. Takahashi, H. San and N. Takai, "SAR ADC algorithm with redundancy," APCCAS 2008 - 2008 IEEE Asia Pacific Conference on Circuits and Systems, pp. 268-271, Macao, 2008.

[47] S. Park, Y. Palaskas and M. P. Flynn, "A 4-GS/s 4-bit Flash ADC in 0.18- μm CMOS," IEEE Journal of Solid-State Circuits, vol. 42, no. 9, pp. 1865-1872, Sept. 2007.

[48] W. Liu, P. Huang and Y. Chiu, "A 12-bit, 45-MS/s, 3-mW Redundant Successive-Approximation-Register Analog-to-Digital Converter With Digital Calibration," IEEE Journal of Solid-State Circuits, vol. 46, no. 11, pp. 2661-2672, Nov. 2011.

[49] T. Okazaki, D. Kanemoto, R. Pokharel, K. Yoshida and H. Kanaya, "A design technique for a high-speed SAR ADC using non-binary search algorithm and redundancy," 2013 Asia-Pacific Microwave Conference Proceedings (APMC), pp. 506-508, Seoul, 2013.

[50] A. H. Chang, H. S. Lee and D. Boning, "A 12b 50MS/s 2.1mW SAR ADC with redundancy and digital background calibration," 2013 Proceedings of the ESSCIRC (ESSCIRC), pp. 109-112, Bucharest, 2013.

[51] M. Ding, P. Harpe, Y. H. Liu, B. Busze, K. Philips and H. Groot, "26.2 A 5.5fJ/conv-step 6.4MS/S 13b SAR ADC utilizing a redundancy-facilitated background error-detection-and-correction scheme," 2015 IEEE International Solid-State Circuits Conference - (ISSCC) Digest of Technical Papers, pp. 1-3, San Francisco, CA, 2015.

[52] S. Haenzsche, S. Höppner, G. Ellguth and R. Schüffny, "A 12-b 4-MS/s SAR ADC With Configurable Redundancy in 28-nm CMOS Technology," IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 61, no. 11, pp. 835-839, Nov. 2014.

[53] T. Li, Y. Wang, R. Li and K. Li, "A bidirectional overflow digital correction algorithm with a single bit redundancy used in the pipeline A/D converters," 2009 IEEE 8th International Conference on ASIC, pp. 238-241, Changsha, Hunan, 2009.

[54] S. H. Lewis, H. S. Fetterman, G. F. Gross, R. Ramachandran and T. R. Viswanathan, "A 10-b 20-Msample/s analog-to-digital converter," IEEE Journal of Solid-State Circuits, vol. 27, no. 3, pp. 351-358, Mar 1992.

[55] J. Guerber, M. Gande and U. K. Moon, "The Analysis and Application of Redundant Multistage ADC Resolution Improvements Through PDF Residue Shaping," IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 59, no. 8, pp. 1733-1742, Aug. 2012.

[56] C. H. Chan, Y. Zhu, U. F. Chio, S. W. Sin, U. Seng-Pan and R. P. Martins, "A reconfigurable low-noise dynamic comparator with offset calibration in 90nm CMOS," IEEE Asian Solid-State Circuits Conference 2011, pp. 233-236, Jeju, 2011.

[57] K. L. J. Wong and C. K. K. Yang, "Offset compensation in comparators with minimum input-referred supply noise," IEEE Journal of Solid-State Circuits, vol. 39, no. 5, pp. 837-840, May 2004.

[58] M. Miyahara, Y. Asada, D. Paik and A. Matsuzawa, "A low-noise self-calibrating dynamic comparator for high-speed ADCs," 2008 IEEE Asian Solid-State Circuits Conference, pp. 269-272, Fukuoka, 2008.

[59] C. H. Chan, Y. Zhu, U. F. Chio, S. W. Sin, S. P. U and R. P. Martins, "A voltage-controlled capacitance offset calibration technique for high resolution dynamic comparator," 2009 International SoC Design Conference (ISOCC), pp. 392-395, Busan, 2009.

[60] P. Nuzzo, C. Nani, C. Armiento, A. Sangiovanni-Vincentelli, J. Craninckx and G. Van der Plas, "A 6-Bit 50-MS/s Threshold Configuring SAR ADC in 90-nm Digital CMOS," IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 59, no. 1, pp. 80-92, Jan. 2012.

[61] M. Zlochisti, S. A. Zahrai and M. Onabajo, "Digitally programmable offset compensation of comparators in flash ADCs for hybrid ADC architectures," 2015 IEEE 58th International Midwest Symposium on Circuits and Systems (MWSCAS), pp. 1-4, Fort Collins, CO, 2015.

[62] H. Okada, Y. Hashimoto, K. Sakata, T. Tsukada and K. Ishibashi, "Offset calibrating comparator array for 1.2-V, 6bit, 4-Gsample/s flash ADCs using 0.13/spl mu/m generic CMOS technology," ESSCIRC 2004 - 29th European Solid-State Circuits Conference (IEEE Cat. No.03EX705), pp. 711-714, Estoril, Portugal, 2003.

[63] M. El-Chammas and B. Murmann, "A 12-GS/s 81-mW 5-bit Time-Interleaved Flash ADC With Background Timing Skew Calibration," IEEE Journal of Solid-State Circuits, vol. 46, no. 4, pp. 838-847, April 2011.

[64] Y. Asada, K. Yoshihara, T. Urano, M. Miyahara and A. Matsuzawa, "A 6bit, 7mW, 250fJ, 700MS/s subranging ADC," 2009 IEEE Asian Solid-State Circuits Conference, pp. 141-144, Taipei, 2009.

[65] J. Mei, X. Shen, H. Zhou, F. Ye and J. Ren, "A low kickback noise and offset calibrated dynamic comparator for 2B/C SAR ADC," 2014 12th IEEE International Conference on Solid-State and Integrated Circuit Technology (ICSICT), pp. 1-3, Guilin, 2014.

[66] D. G. Chen and A. Bermak, "A low-power dynamic comparator with digital calibration for reduced offset mismatch," 2012 IEEE International Symposium on Circuits and Systems, pp. 1283-1286, Seoul, Korea (South), 2012.

[67] H. Jeon, Y. B. Kim and M. Choi, "Offset voltage analysis of dynamic latched comparator," 2011 IEEE 54th International Midwest Symposium on Circuits and Systems (MWSCAS), pp. 1-4, Seoul, 2011.

[68] Y. L. Wong, M. H. Cohen and P. A. Abshire, "A 1.2-GHz comparator with adaptable offset in 0.35-μm CMOS," IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 55, no. 9, pp. 2584-2594, Oct. 2008.

[69] J. Goes, J. C. Vital and J. E. Franca, "Systematic design for optimization of high-speed self-calibrated pipelined A/D converters," IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, vol. 45, no. 12, pp. 1513-1526, Dec 1998.

[70] S. H. Lee and B. S. Song, "Digital-domain calibration of multistep analog-to-digital converters," IEEE Journal of Solid-State Circuits, vol. 27, no. 12, pp. 1679-1688, Dec 1992.

[71] H. S. Lee, D. A. Hodges and P. R. Gray, "A self-calibrating 15 bit CMOS A/D converter," IEEE Journal of Solid-State Circuits, vol. 19, no. 6, pp. 813-819, Dec. 1984.

[72] C. R. Grace, P. J. Hurst and S. H. Lewis, "A 12-bit 80-MSample/s pipelined ADC with bootstrapped digital calibration," IEEE Journal of Solid-State Circuits, vol. 40, no. 5, pp. 1038-1046, May 2005.

[73] C. S. G. Conroy, D. W. Cline and P. R. Gray, "An 8-b 85-MS/s parallel pipeline A/D converter in 1-μm CMOS," IEEE Journal of Solid-State Circuits, vol. 28, no. 4, pp. 447-454, Apr 1993.

[74] I. E. Opris, L. D. Lewicki and B. C. Wong, "A single-ended 12-bit 20 Msample/s self-calibrating pipeline A/D converter," IEEE Journal of Solid-State Circuits, vol. 33, no. 12, pp. 1898-1903, Dec 1998.

[75] D. Y. Chang, J. Li and U. K. Moon, "Radix-based digital calibration techniques for multi-stage recycling pipelined ADCs," IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 51, no. 11, pp. 2133-2140, Nov. 2004.

[76] T. Ito and T. Itakura, "A 3-GS/s 5-bit 36-mW flash ADC in 65-nm CMOS," 2010 IEEE Asian Solid-State Circuits Conference, pp. 1-4, Beijing, 2010.

[77] B. Murmann, "Digitally Assisted Data Converter Design," in 2013 Proceedings of the ESSCIRC (ESSCIRC), Bucharest, 2013.

[78] U. K. Moon and B. S. Song, "Background digital calibration techniques for pipelined ADCs," IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, vol. 44, no. 2, pp. 102-109, Feb 1997.

[79] E. J. Siragusa and I. Galton, "Gain error correction technique for pipelined analogue-to-digital converters," Electronics Letters, vol. 36, no. 7, pp. 617-618, 30 Mar 2000.

[80] J. Ming and S. H. Lewis, "An 8-bit 80-Msample/s pipelined analog-to-digital converter with background calibration," IEEE Journal of Solid-State Circuits, vol. 36, no. 10, pp. 1489-1497, Oct 2001.

[81] B. Murmann and B. E. Boser, "A 12 b 75 MS/s pipelined ADC using open-loop residue amplification," 2003 IEEE International Solid-State Circuits Conference, 2003. Digest of Technical Papers. ISSCC., vol. 1, pp. 328-497 , San Francisco, CA, USA, 2003.

[82] I. Galton, "Digital cancellation of D/A converter noise in pipelined A/D converters," IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, vol. 47, no. 3, pp. 185-196, Mar 2000.

[83] A. J. Ginés, E. J. Peralias and A. Rueda, "Digital background gain error correction in pipeline ADCs," Proceedings Design, Automation and Test in Europe Conference and Exhibition, vol. 1, pp. 82-87, 2004.

[84] R. G. Massolini, G. Cesura and R. Castello, "A fully digital fast convergence algorithm for nonlinearity correction in multistage ADC," IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 53, no. 5, pp. 389-393, May 2006.

[85] R. Jewett, K. Poulton, K. C. Hsieh and J. Doernberg, "A 12 b 128 MSample/s ADC with 0.05 LSB DNL," 1997 IEEE International Solids-State Circuits Conference. Digest of Technical Papers, pp. 138-139, San Francisco, CA, USA, 1997.

[86] J. McNeill, M. Coln and B. Larivee, "Split-ADC" architecture for deterministic digital background calibration of a 16-bit 1-MS/s ADC," IEEE J. Solid-State Circuits, vol. 40, no. 12, pp. 2437-2445, Dec. 2005.

[87] J. Li, G. C. Ahn, D. Y. Chang and U. K. Moon, "A 0.9-V 12-mW 5-MSPS algorithmic ADC with 77-dB SFDR," IEEE Journal of Solid-State Circuits, vol. 40, no. 4, pp. 960-969, April 2005.

[88] J. L. Fan, C. Y. Wang and J. T. Wu, "A Robust and Fast Digital Background Calibration Technique for Pipelined ADCs," IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 54, no. 6, pp. 1213-1223, June 2007.

[89] C. H. Law, P. J. Hurst and S. H. Lewis, "A Four-Channel Time-Interleaved ADC With Digital Calibration of Interchannel Timing and Memory Errors," IEEE Journal of Solid-State Circuits, vol. 45, no. 10, pp. 2091-2103, Oct. 2010.

[90] C. C. Huang and J. T. Wu, "A background comparator calibration technique for flash analog-to-digital converters," IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 52, no. 9, pp. 1732-1740, Sept. 2005.

[91] H. Wang, X. Wang, P. J. Hurst and S. H. Lewis, "Nested Digital Background Calibration of a 12-bit Pipelined ADC Without an Input SHA," IEEE Journal of Solid-State Circuits, vol. 44, no. 10, pp. IEEE Journal of Solid-State Circuits, Oct. 2009.

[92] A. Tahmasebi, A. Kamali, Z. D. K. Kanani and J. Sobhi, "A Simple Background Interstage Gain Calibration Technique for Pipeline ADCs," 2009 International Conference on Signal Acquisition and Processing, pp. 221-224, Kuala Lumpur, 2009.

[93] J. Sobhi, Z. K. Kanani, A. Tahmasebi and M. Yousefi, "A mixed mode background calibration technique for pipeline ADCs," 2009 4th IEEE Conference on Industrial Electronics and Applications, pp. 2452-2455, Xian, 2009.

[94] Y. Chiu, C. W. Tsang, B. Nikolic and P. R. Gray, "Least mean square adaptive digital background calibration of pipelined analog-to-digital converters," IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 51, no. 1, pp. 38-46, Jan. 2004.

[95] A. J. Ginés, E. J. Peralías and A. Rueda, "Digital background calibration technique for pipeline ADCs with multi-bit stages," 16th Symposium on Integrated Circuits and Systems Design, 2003. SBCCI 2003. Proceedings., pp. 317-322, 2003.

[96] H. Mafi, M. Yargholi and M. Yavari, "Digital Blind Background Calibration of Imperfections in Time-Interleaved ADCs," IEEE Transactions on Circuits and Systems I: Regular Papers, vol. PP, no. 99, pp. 1-11.

[97] S. Saleem and C. Vogel, "Adaptive Blind Background Calibration of Polynomial-Represented Frequency Response Mismatches in a Two-Channel Time-Interleaved ADC," IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 58, no. 6, pp. 1300-1310, June 2011.

[98] C. Vogel, S. Saleem and S. Mendel, "Adaptive blind compensation of gain and timing mismatches in M-channel time-interleaved ADCs," 2008 15th IEEE International Conference on Electronics, Circuits and Systems, pp. 49-52, St. Julien's, 2008.

[99] S. J. Liu, P. P. Qi, J. S. Wang, M. H. Zhang and W. S. Jiang, "Adaptive Calibration of Channel Mismatches in Time-Interleaved ADCs Based on Equivalent Signal Recombination," IEEE Transactions on Instrumentation and Measurement, vol. 63, no. 2, pp. 277-286, Feb. 2014.

[100] S. Liu, H. Ma and J. Cui, "Adaptive semiblind background calibration of timing mismatches in an M-channel time-interleaved analog-to-digital converter," 2015 IEEE 11th International Conference on ASIC (ASICON), pp. 1-4, Chengdu, 2015.

[101] B. Xu and Y. Chiu, "Comprehensive Background Calibration of Time-Interleaved Analog-to-Digital Converters," IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 62, no. 5, pp. 1306-1314, May 2015.

[102] N. T. Abou-El-Kheir, M. Abbas and M. E. Khedr, "An adaptive digital background calibration technique using variable step size LMS for pipelined ADC," 2014 9th International Symposium on Communication Systems, Networks & Digital Sign (CSNDSP), pp. 835-840, Manchester, 2014.

[103] A. J. Ginés, E. J. Peralias and A. Rueda, "Blind Adaptive Estimation of Integral Nonlinear Errors in ADCs Using Arbitrary Input Stimulus," IEEE Transactions on Instrumentation and Measurement, vol. 60, no. 2, pp. 452-461, Feb. 2011.

[104] A. J. Ginés, E. J. Peralias, C. Aledo and A. Rueda, "Power optimization and stage op-amp linearity relaxation in pipeline ADCs with digital comparator offset calibration," Design of Circuits and Integrated Systems, pp. 1-6, Madrid, 2004.

[105] A. Ginés, E. Peralías and A. Rueda, "Background Digital Calibration of Comparator Offsets in Pipeline ADCs," IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 23, no. 7, pp. 1345-1349, July 2015.

[106] A. Ginés, E. Peralías, C. Aledo and A. Rueda, "Fast Background Calibration of Comparator Offsets in Pipeline ADCs with Self-repairing Thermometer to Binary Encoder," International Journal of Circuit Theory and Application, vol. June, 2019.

[107] J. I. Huircán, Conversores Análogo-Digital y Digital-Análogo: Conceptos básicos.

[108] W. Kester, "Which ADC architecture is right for your application?," Analog Dialog 39-06, pp. Available: http://www.analog.com/media/en/analog-dialogue/volume-39/number-2/articles/the-right-adc-architecture.pdf, 2005.

[109] T. Chan Carusone, D. A. Johns and K. W. Martin, Analog Integrated Circuit Design, Toronto: John Wiley & Sons Inc, 2012.

[110] B. Murmann, "ADC Performance Survey 1997-2016 [Online]. Available: http://web.stanford.edu/~murmann/adcsurvey.html," 2016.

[111] B. Murmann, "The Race for the Extra Decibel: A Brief Review of Current ADC Performance Trajectories," IEEE Solid-State Circuits Magazine, vol. 7, no. 3, pp. 58 - 66, Sept. 2015.

[112] S. Shahramian, S. P. Voinigescu and A. C. Carusone, "A 35-GS/s, 4-Bit Flash ADC With Active Data and Clock Distribution Trees," IEEE Journal of Solid-State Circuits, vol. 44, no. 6, pp. 1709-1720, June 2009.

[113] V. H. C. Chen and L. Pileggi, "A 69.5mW 20GS/s 6b time-interleaved ADC with embedded time-to-digital calibration in 32nm CMOS SOI," 2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC), pp. 380-381, San Francisco, CA, 2014.

[114] J. I. Kim, D. R. Oh, D. S. Jo, B. R. S. Sung and S. T. Ryu, "A 65 nm CMOS 7b 2 GS/s 20.7 mW Flash ADC With Cascaded Latch Interpolation," IEEE Journal of Solid-State Circuits, vol. 50, no. 10, pp. 2319-2330, Oct. 2015.

[115] B. Verbruggen, J. Craninckx, M. Kuijk and P. Wambacq , "A 2.2 mW 1.75 GS/s 5 Bit Folding Flash ADC in 90 nm Digital CMOS," IEEE Journal of Solid-State Circuits, vol. 44, no. 3, pp. 874-882, March 2009.

[116] J. Wu and e. al., "27.6 A 4GS/s 13b pipelined ADC with capacitor and amplifier

sharing in 16nm CMOS," 2016 IEEE International Solid-State Circuits Conference (ISSCC), pp. 466-467, San Francisco, CA, 2016.

[117] A. J. Ginés, E. J. Peralías and A. Rueda, "Black-Box Calibration for ADCs With Hard Nonlinear Errors Using a Novel INL-Based Additive Code: A Pipeline ADC Case Study," IEEE Transactions on Circuits and Systems I: Regular Papers, vol. PP, no. 99, pp. 1-12, 2017.

[118] G. Mitteregger, C. Ebner, S. Mechnig, T. Blon, C. Holuigue and E. Romani, "A 20-mW 640-MHz CMOS Continuous-Time \Sigma \Delta ADC With 20-MHz Signal Bandwidth, 80-dB Dynamic Range and 12-bit ENOB," IEEE Journal of Solid-State Circuits, vol. 41, no. 12, pp. 2641-2649, Dec. 2006.

[119] B. Murmann, "The successive approximation register ADC: a versatile building block for ultra-low- power to ultra-high-speed applications," IEEE Communications Magazine, vol. 54, no. 4, pp. 78-83, April 2016.

[120] K. Doris, E. Janssen, C. Nani and A. Zanikopoulos, "A 480 mW 2.6 GS/s 10b Time-Interleaved ADC With 48.5 dB SNDR up to Nyquist in 65 nm CMOS," IEEE Journal of Solid-State Circuits, vol. 46, no. 12, pp. 2821-2833, Dec. 2011.

[121] L. Kull, T. Toifl, M. Schmatz, P. A. Francese, C. Menolfi, M. Braendli, M. Kossel, T. Morf, T. M. Andersen and Y. Leblebici, "A 90GS/s 8b 667mW 64× interleaved SAR ADC in 32nm digital SOI CMOS," 2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC), pp. 378-379, San Francisco, CA, 2014.

[122] C. C. Lee and M. P. Flynn, "A SAR-Assisted Two-Stage Pipeline ADC," IEEE J. Solid-State Circuits, vol. 46 , no. 4, pp. 859-869, Apr. 2011.

[123] J. Wu, F. Li, W. Li, C. Zhang and Z. Wang, "A 14b 200MHz power-efficient pipelined flash-SAR ADC," in 2014 12th IEEE International Conference on Solid-State and Integrated Circuit Technology (ICSICT), Guilin, 2014.

[124] A. Devices, Analog-Digital Conversion, 2004.

[125] P. Scholtens and M. Vertregt, "A 6-b 1.6-Gsample/s flash ADC in 0.18-μm CMOS using averaging termination," IEEE Journal of Solid-State Circuits, vol. 37, no. 12, pp. 1599-1609, Dec. 2002.

[126] C. Sandner, M. Clara, A. Santner, T. Hartig and F. Kuttner, "A 6-bit 1.2-GS/s low-power flash-ADC in 0.13-μm digital CMOS," IEEE Journal of Solid-State Circuits, vol. 40, no. 7, pp. 1499-1505, July 2005.

[127] J. Lin and B. Haroun, "An embedded 0.8 V/480 μW 6b/22 MHz flash ADC in 0.13-μm digital CMOS process using a nonlinear double interpolation technique," IEEE Journal of Solid-State Circuits, vol. 37, no. 12, pp. 1610-1617, Dec 2002.

[128] R. C. Taft, C. A. Menkus, M. R. Tursi, O. Hidri and V. Pons, "A 1.8-V 1.6-

GSample/s 8-b self-calibrating folding ADC with 7.26 ENOB at Nyquist frequency," IEEE Journal of Solid-State Circuits, vol. 39, no. 12, pp. 2107-2115, Dec. 2004.

[129] M. Saberi, R. Lotfi, K. Mafinezhad and W. A. Serdi, "Analysis of Power Consumption and Linearity in Capacitive Digital-to-Analog Converters Used in Successive Approximation ADCs," IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 58, no. 8, pp. 1736-1748, Aug. 2011.

[130] N. Verma and A. P. Chandrakasan, "An Ultra Low Energy 12-bit Rate-Resolution Scalable SAR ADC for Wireless Sensor Nodes," IEEE Journal of Solid-State Circuits, vol. 42, no. 6, pp. 1196-1205, June 2007.

[131] J. Sauerbrey, D. Schmitt-Landsiedel and R. Thewes, "A 0.5-V 1-μW successive approximation ADC," IEEE Journal of Solid-State Circuits, vol. 38, no. 7, pp. 1261-1265, July 2003.

[132] H. C. Hong and G. M. Lee, "A 65-fJ/Conversion-Step 0.9-V 200-kS/s Rail-to-Rail 8-bit Successive Approximation ADC," IEEE Journal of Solid-State Circuits, vol. 42, no. 10, pp. 2161-2168, Oct. 2007.

[133] Y. Zhu, C. H. Chan, U. F. Chio, S. W. Sin, S. P. U, R. P. Martins and F. Maloberti, "A 10-bit 100-MS/s Reference-Free SAR ADC in 90 nm CMOS," IEEE Journal of Solid-State Circuits, vol. 45, no. 6, pp. 1111-1121, June 2010.

[134] E. Culurciello and A. G. Andreou, "An 8-bit 800- \mu\hbox {W} 1.23-MS/s Successive Approximation ADC in SOI CMOS," IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 53, no. 9, pp. 858-861, Sept. 2006.

[135] B. P. Ginsburg and A. P. Chandrakasan, "500-MS/s 5-bit ADC in 65-nm CMOS With Split Capacitor Array DAC," IEEE Journal of Solid-State Circuits, vol. 42, no. 4, pp. 739-747, April 2007.

[136] B. P. Ginsburg and A. P. Chandrakasan, "An energy-efficient charge recycling approach for a SAR converter with capacitive DAC," 2005 IEEE International Symposium on Circuits and Systems, vol. 1, pp. 184-187, 2005.

[137] Yan Zhu, U-Fat Chio, He-Gong Wei, Sai-Weng Sin, Seng-Pan U and R.P. Martins, "A power-efficient capacitor structure for high-speed charge recycling SAR ADCs," 2008 15th IEEE International Conference on Electronics, Circuits and Systems, pp. 642-645, St. Julien's, 2008.

[138] H. K. Hong, W. Kim, S. J. Park, M. Choi and H. J. Par, "A 7b 1GS/s 7.2mW nonbinary 2b/cycle SAR ADC with register-to-DAC direct control," Proceedings of the IEEE 2012 Custom Integrated Circuits Conference, pp. 1-4, San Jose, CA, 2012.

[139] P. Harpe, C. Zhou, Y. Bi, N. van der Meijs, X. Wang, K. Philips, G. Dolmans and H. Groot, "A 26 \mu W 8 bit 10 MS/s Asynchronous SAR ADC for Low Energy Radios," IEEE Journal of Solid-State Circuits, vol. 46, no. 7, pp. 1585-1595, July 2011.

[140] T. Jiang, W. Liu, F. Y. Zhong, C. Zhong, K. Hu and P. Y. Chiang, "A Single-Channel, 1.25-GS/s, 6-bit, 6.08-mW Asynchronous Successive-Approximation ADC With Improved Feedback Delay in 40-nm CMOS," IEEE Journal of Solid-State Circuits, vol. 47, no. 10, pp. 2444-2453, Oct. 2012.

[141] P. Harpe, C. Zhou, X. Wang, G. Dolmans and H. Groot, "A 30fJ/conversion-step 8b 0-to-10MS/s asynchronous SAR ADC in 90nm CMOS," 2010 IEEE International Solid-State Circuits Conference - (ISSCC), pp. 388-389, San Francisco, CA, 2010.

[142] V. Tripathi and B. Murmann, "An 8-bit 450-MS/s single-bit/cycle SAR ADC in 65-nm CMOS," 2013 Proceedings of the ESSCIRC (ESSCIRC), pp. 117-120, Bucharest, 2013.

[143] H. Chen, J. Xiang, C. Chen, F. Ye, J. Xu and J. Ren, "A foreground digital calibration by switching control scheme for A 12-bit SAR ADC," 2014 12th IEEE International Conference on Solid-State and Integrated Circuit Technology (ICSICT), pp. 1-3, Guilin, 2014.

[144] W. Li, T. Wang and G. C. Temes, "Digital foreground calibration methods for SAR ADCs," 2012 IEEE International Symposium on Circuits and Systems, pp. 1054-1057, Seoul, Korea (South), 2012.

[145] F. Siqiang, H. Tang, H. Zhao, X. Wang, A. Wang, B. Zhao and G. C. Zhang, "Enhanced offset averaging technique for flash ADC design," Tsinghua Science and Technology, vol. 16, no. 3, pp. 285-289, June 2011.

[146] J. Doernberg, P. R. Gray and D. A. Hodges, "A 10-bit 5-Msample/s CMOS two-step flash ADC," IEEE Journal of Solid-State Circuits, vol. 24, no. 2, pp. 241-249, April 1989.

[147] G. Van der Plas, S. Decoutere and S. Donnay, "A 0.16pJ/Conversion-Step 2.5mW 1.25GS/s 4b ADC in a 90nm Digital CMOS Process," 2006 IEEE International Solid State Circuits Conference - Digest of Technical Papers, pp. 566-567, San Francisco, CA, 2006.

[148] Y. S. Shu, J. Y. Tsai, P. Chen, T. Y. Lo and P. C. Chiu, "A background calibration technique for fully dynamic flash ADCs," 2013 International Symposium onVLSI Design, Automation, and Test (VLSI-DAT), pp. 1-4, Hsinchu, 2013.

[149] T. Shih, L. Der, S. H. Lewis and P. J. Hurst, "A fully differential comparator using a switched-capacitor differencing circuit with common-mode rejection," IEEE Journal of Solid-State Circuits, vol. 32, no. 2, pp. 250-253, Feb 1997.

[150] J. P. Oliveira, J. Goes, B. Esperanca and N. Paulino , "Low-Power CMOS Comparator with Embedded Amplification for Ultra-high-speed ADCs," in 2007 IEEE International Symposium on Circuits and Systems, New Orleans, LA,, 2007.

[151] J. Yang, Y. Chen, H. Qian, Y. Wang and R. Yue, "A 3.65 mW 5 bit 2GS/s flash ADC with built-in reference voltage in 65nm CMOS process," 2012 IEEE 11th International Conference on Solid-State and Integrated Circuit Technology, pp. 1-3, Xi'an, 2012.

[152] J. Goes, J. C. Vital and J. E. Franca, "Systematic design for optimization of high-speed self-calibrated pipelined A/D converters," IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, vol. 45, no. 12, pp. 1513-1526, Dec 1998.

[153] T. C. Carusone, D. A. Johns and K. W. Martin, Analog Integrated Circuit Design (ISBN 978-047077010-8), John Wiley & Sons, Inc., 2012.

[154] Yao-Jen Chuang, Hsin-Hung Ou and Bin-Da Liu, "A novel bubble tolerant thermometer-to-binary encoder for flash A/D converter," 2005 IEEE VLSI-TSA International Symposium on VLSI Design, Automation and Test, 2005. (VLSI-TSA-DAT), pp. 315-318, 2005.

[155] A. Chunn and R. K. Sarin, "Comparison of thermometer to binary encoders for flash ADCs," 2013 Annual IEEE India Conference (INDICON), Mumbai, pp. 1-4, 2013.

[156] C. Vudadha, G. Makkena, M. V. S. Nayudu, P. S. Phaneendra, S. E. Ahmed, S. Veeramachaneni, N. M. Muthukrishnan and M. Srinivas, "Low-Power Self Reconfigurable Multiplexer Based Decoder for Adaptive Resolution Flash ADCs," 2012 25th International Conference on VLSI Design, Hyderabad, pp. 280-285, 2012.

[157] V. H. Bui, S. Beak, S. Choi, J. Seon and T. T. Jeong, "Thermometer-to-binary encoder with bubble error correction (BEC) circuit for Flash Analog-to-Digital Converter (FADC)," International Conference on Communications and Electronics 2010, Nha Trang, Vols. 102-106, 2010.

[158] D. Lee, J. Yoo, K. Choi and J. Ghaznavi, "Fat tree encoder design for ultra-high speed flash A/D converters," The 2002 45th Midwest Symposium on Circuits and Systems, 2002. MWSCAS-2002, vol. 2, pp. II-87-II-90, 2002.

[159] E. Sail and M. Vesterbacka, "A multiplexer based decoder for flash analog-to-digital converters," 2004 IEEE Region 10 Conference TENCON 2004, vol. 4, pp. 250-253, 2004.

[160] B. Van Hieu, S. Choi, J. Seon, Y. Oh, C. Park, J. Park, H. Kim and T. Jeong, "A new approach to thermometer-to-binary encoder of flash ADCs- bubble error detection circuit," 2011 IEEE 54th International Midwest Symposium on Circuits and Systems (MWSCAS), Seoul, pp. 1-4, 2011.

[161] N. Agrawal and R. Paily, "An improved ROM architecture for bubble error suppression in high speed flash ADCs," 2008 Annual IEEE Student Paper

Conference, Aalborg, pp. 1-5, 2008.

[162] F. Kaess, R. Kanan, B. Hochet and M. Declercq, "New encoding scheme for high-speed flash ADC's," Circuits and Systems, 1997. ISCAS '97., Proceedings of 1997 IEEE International Symposium on, vol. 1, pp. 5-8, 1997.

[163] T. Pardhu, S. Manusha and K. Sirisha, "A low power flash ADC with Wallace tree encoder," 2014 Eleventh International Conference on Wireless and Optical Communications Networks (WOCN), pp. 1-4, Vijayawada, 2014.

[164] I. S. A. Halim, S. L. M. Hassan, N. D. b. M. Akbar and A. A. A. Rahim, "Comparative study of comparator and encoder in a 4-bit Flash ADC using 0.18μm CMOS technology," in 2012 International Symposium on Computer Applications and Industrial Electronics (ISCAIE), Kota Kinabalu, 2012.

[165] B. N. Fang and J. T. Wu, "A 10-Bit 300-MS/s Pipelined ADC With Digital Calibration and Digital Bias Generation," IEEE Journal of Solid-State Circuits, vol. 48, no. 3, pp. 670-683, March 2013.

[166] I. Ahmed, J. Mulder and D. A. Johns, "A Low-Power Capacitive Charge Pump Based Pipelined ADC," IEEE Journal of Solid-State Circuits, vol. 45, no. 5, pp. 1016-1027, May 2010.

[167] B. G. Lee and R. M. Tsang, "A 10-bit 50 MS/s Pipelined ADC With Capacitor-Sharing and Variable-gm Opamp," IEEE Journal of Solid-State Circuits, vol. 44, no. 3, pp. 883-890, March 2009.

[168] N. Sasidhar , Y. J. Kook, S. Takeuchi, K. Hamashita, K. Takasuka, P. V. Hanumolu and U. K. Moon, "A Low Power Pipelined ADC Using Capacitor and Opamp Sharing Technique With a Scheme to Cancel the Effect of Signal Dependent Kickback," IEEE Journal of Solid-State Circuits, vol. 44, no. 9, pp. 2392-2401, Sept. 2009.

[169] B. D. Sahoo and B. Razavi, "A 12-Bit 200-MHz CMOS ADC," IEEE Journal of Solid-State Circuits, vol. 44, no. 9, pp. 2366-2380, Sept. 2009.

[170] L. Brooks and H. S. Lee, "A 12b 50MS/s fully differential zero-crossing-based ADC without CMFB," 2009 IEEE International Solid-State Circuits Conference - Digest of Technical Papers, pp. 166-167,167a, San Francisco, CA, 2009.

[171] S. Devarajan, L. Singer, D. Kelly, S. Decker, A. Kamath and P. Kamath , "A 16b 125MS/s 385mW 78.7dB SNR CMOS pipeline ADC," 2009 IEEE International Solid-State Circuits Conference - Digest of Technical Papers, pp. 86-87,87a, San Francisco, CA, 2009.

[172] S. Ray and B. S. Song, "A 13-b Linear, 40-MS/s Pipelined ADC With Self-Configured Capacitor Matching," IEEE Journal of Solid-State Circuits, vol. 42, no. 3, pp. 463-474, March 2007.

[173] I. Ahmed and D. A. Johns, "A 50-MS/s (35 mW) to 1-kS/s (15 μW) power scaleable 10-bit pipelined ADC using rapid power-on opamps and minimal bias current variation," IEEE Journal of Solid-State Circuits, vol. 40, no. 12, pp. 2446-2455, Dec. 2005.

[174] J. Li and U. K. Moon, "A 1.8-V 67-mW 10-bit 100-MS/s pipelined ADC using time-shifted CDS technique," IEEE Journal of Solid-State Circuits, vol. 39, no. 9, pp. 1468-1476, Sept. 2004.

[175] S. Lee, A. P. Chandrakasan and H. S. Lee, "A 12b 5-to-50MS/s 0.5-to-1V voltage scalable zero-crossing based pipelined ADC," 2011 Proceedings of the ESSCIRC (ESSCIRC), pp. 355-358, Helsinki, 2011.

[176] L. Picolli, A. Rossini, P. Malcovati, F. Maloberti, F. Borghetti and A. Baschirotto, "A Clock-Less 10-bit Pipeline-Like A/D Converter for Self-Triggered Sensors," IEEE Journal of Solid-State Circuits, vol. 43, no. 2, pp. 312-320, Feb. 2008.

[177] M. J. Choe, B. S. Song and K. Bacrania, "An 8-b 100-MSample/s CMOS pipelined folding ADC," IEEE Journal of Solid-State Circuits, vol. 36, no. 2, pp. 184-194, Feb 2001.

[178] M. J. Choe, B. S. Song and K. Bacrania, "A 13-b 40-MSamples/s CMOS pipelined folding ADC with background offset trimming," IEEE Journal of Solid-State Circuits, vol. 35, no. 12, pp. 1781-1790, Dec. 2000.

[179] C. J. Tseng, H. W. Chen, W. T. Shen, W. C. Cheng and H. S. Chen, "A 10-b 320-MS/s Stage-Gain-Error Self-Calibration Pipeline ADC," IEEE Journal of Solid-State Circuits, vol. 47, no. 6, pp. 1334-1343, June 2012.

[180] B. M. Min, P. Kim, F. W. Bowman, D. M. Boisvert and A. J. Aude, "A 69-mW 10-bit 80-MSample/s Pipelined CMOS ADC," IEEE Journal of Solid-State Circuits, vol. 38, no. 12, pp. 2031-2039, Dec. 2003.

[181] Z. Huang and P. Zhong, "An adaptive analog-to-digital converter based on low-power dynamic latch comparator," 2005 IEEE International Conference on Electro Information Technology, p. 6, Lincoln, NE, 2005.

[182] D. Schinkel, E. Mensink, E. Klumperink, E. Tuijl and B. Nauta, "A Double-Tail Latch-Type Voltage Sense Amplifier with 18ps Setup+Hold Time," 2007 IEEE International Solid-State Circuits Conference. Digest of Technical Papers, pp. 314-605, San Francisco, CA, 2007.

[183] S. Babayan-Mashhadi and R. Lotfi, "Analysis and Design of a Low-Voltage Low-Power Double-Tail Comparator," IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 22, no. 2, pp. 343-352, Feb. 2014.

[184] Y. Lin, K. Doris, H. Hegt and A. Roermund, "A dynamic latched comparator for

low supply voltages down to 0.45 V in 65-nm CMOS," 012 IEEE International Symposium on Circuits and Systems, pp. 2737-2740, Seoul, 2012.

[185] H. J. Jeon and Y. B. Kim, "A low-offset high-speed double-tail dual-rail dynamic latched comparator," ACM Great Lakes Symposium on VLSI, 2010.

[186] H. Jeon and Y. B. Kim, "A CMOS low-power low-offset and high-speed fully dynamic latched comparator," 23rd IEEE International SOC Conference, pp. 285-288, Las Vegas, NV, 2010.

[187] A. J. Ginés, E. J. Peralías, G. Leger and A. Rueda, "Closed-loop simulation method for evaluation of static offset in discrete-time comparators," 2014 21st IEEE International Conference on Electronics, Circuits and Systems (ICECS), pp. 538-541, Marseille, 2014.

[188] "Maxim Integrated Products, Inc.," 27 Oct 2009. [Online]. Available: https://www.maximintegrated.com/en/app-notes/index.mvp/id/4602.

[189] J. E. Proesel, Flash Analog-to-Digital Converter Design Based on Statistical Post-Silicon Calibration, 2010.

[190] B. Murmann, "A/D converter trends: Power dissipation, scaling and digitally assisted architectures," in 2008 IEEE Custom Integrated Circuits Conference, San Jose, CA, 2008.

[191] P. E. Allen and D. R. Holberg, CMOS Analog Circuit Design, Oxford University Press, 2002.

[192] B. Razavi and B. A. Wooley, "Design techniques for high-speed, high-resolution comparators," IEEE Journal of Solid-State Circuits, vol. 27, no. 12, pp. 1916-1926, Dec 1992.

[193] K. Souri, Y. Chae and K. A. A. Makinwa, "A CMOS Temperature Sensor With a Voltage-Calibrated Inaccuracy of +\- 0.15º C (3 \sigma) From - 55º C to 125º C," IEEE Journal of Solid-State Circuits, vol. 48, no. 1, pp. 292-301, Jan. 2013.

[194] A. Varzaghani, A. Kasapi, D. N. Loizos, S. H. Paik, S. Verma, S. Zogopoulos and S. Sidiropoulos, "A 10.3-GS/s, 6-Bit Flash ADC for 10G Ethernet Applications," IEEE Journal of Solid-State Circuits, vol. 48, no. 12, pp. 3038-3048, Dec. 2013.

[195] S. Park, Y. Palaskas, A. Ravi, R. E. Bishop and M. P. Flynn, "A 3.5 GS/s 5-b Flash ADC in 90 nm CMOS," IEEE Custom Integrated Circuits Conference 2006, pp. 489-492, San Jose, CA, 2006.

[196] S. Bardsley , C. Dillon, R. Kummaraguntla, C. Lane, A. M. A. Ali, B. Rigsbee and D. Combs, "A 100-dB SFDR 80-MSPS 14-Bit 0.35-um BiCMOS Pipeline ADC," IEEE Journal of Solid-State Circuits, vol. 41, no. 9, pp. 2144-2153, Sept. 2006.