Statistical based Piecewise Linear Calibration of Nonlinearity in SAR-ADC

Author
Mirhaj, Seyed Arash

Publication Date
2014

University of California

Los Angeles

Statistical based Piecewise Linear Calibration of Nonlinearity in SAR-ADC

A dissertation submitted in partial satisfaction

of the requirements for the degree

Doctor of Philosophy in Electrical Engineering

by

Seyed Arash Mirhaj

2014

Abstract of the Dissertation

Statistical based Piecewise Linear Calibration of Nonlinearity in SAR-ADC

by

Seyed Arash Mirhaj

Doctor of Philosophy in Electrical Engineering

University of California, Los Angeles, 2014

Professor Mau-Chung Frank Chang, Chair

In recent years, Successive Approximation Register Analog-to-Digital Converter (SAR-ADC) has received significant attention due to its well-known energy and hardware efficiency, which also benefits even more from technology scaling. Traditionally, designers have to pay extra attention to matching of the elements in the Digital-to-Analog Converter (DAC), commonly capacitors, as the heart of the ADC. In many cases this leads to overdesigning the DAC hence, directly and indirectly, wasting precious area and power efficiency.

In this research we have been seeking for a robust, hardware/energy efficient, non-invasive, and non-interrupting calibration scheme for SAR-ADC. Unlike current state of the art, proposed calibration does not require stringent assumption on layout considerations and it does not affect the throughput of the data. The prototype of the calibration has been implemented on a 12-bit SAR-ADC fabricated in 65 nm CMOS. Measurement results show more than 9 dB improvement in SNDR (from 58 dB without calibration to 67.2 with calibration) and about 19 dB enhancement in SFDR of the ADC (from 62.6 dB without calibration to 81.2 with calibration). In order to demonstrate effectiveness of the proposed calibration in a practical scenario with wide bandwidth input signals, the performance of the ADC is also tested for quantizing the samples of a 256-QAM signal. Measurement results show improvement of EVM of the output from about 42 dB without calibration to 50 dB with calibration.

The dissertation of Seyed Arash Mirhaj is approved.

Songwu Lu

Sudhakar Pamarti

Asad A. Abidi

Mau-Chung Frank Chang, Committee Chair

University of California, Los Angeles

2014

To my parents . . .

Table of Contents

List of Figures

List of Tables

Acknowledgments

First and foremost, I would like to express my deepest appreciation and gratitude to my advisor, Professor Mau-Chung Frank Chang, for his continuous support, inspiration, and guidance during past five years. Without his support and direction, this research would not be possible. I would also like to thank Professor Asad Abidi, Professor Sudhakar Pamarti, Professor Katsushi Arisaka, and Songwu Lu for their constructive comments during the research and taking their time to be on my committee.

I would also like to thank members of High Speed Electronic Lab, especially Zuow-Zun Chen, Boyu Hu, Wei-Han Cho, Yilei Li, Yen-Cheng Kuan and Yen-Hsiang Wang, for their fruitful discussions and encouragements.

My appreciation goes to all the members of Integrated Circuit and System Lab for warmly hosting me during the testing process of this research, particularly: Neha Sinha, Dejan Rozgic, Hariprasad Chandrakumar, Richard Dorrance, Vahagan Hokhikyan, Sina basir-Kazeruni and Soheil Golara. I wish you the best with renovating your Lab.

Many thanks to Sedigheh Hashemi, Jaewook (Ryan) Shin, and Arashk Norouzpour for their comments and suggestions on design, layout and testing procedures of the chip.

Also, I would like to express my appreciation to Janet lee and Kyle Jung for all their help and support. You guys are life savers!

I would like to thank my bright roommate, Ata Mahjoufbar, for all the eye opening thoughts that we shared during last four years which made me grow not only in technical knowledge but also in the big picture perspective of life.

A special thanks goes to my beloved girlfriend, Tara Nijhowne, whose love made the most challenging times of this journey delightful. Also I am grateful of the time and effort that she put into reading and editing this manuscript.

Lastly I would like to take advantage of this opportunity to show my appreciation to my parents whom I am indebted of their love forever. Although I have been thousands miles away from them during this work, their warm encouragements and support always gives me the energy to push further. I wish words could bear all my gratitude...

Vita

2006 B.Sc., Electrical Engineering, University of Tehran, Iran.

2008 M.Sc., Electrical Engineering, University of Tehran, Iran.

2009–2014 Research and Teaching Assistant, Electrical Engineering Department, University of California, Los Angeles, California.

CHAPTER 1

Introduction

1.1 Motivation

> “ . . . [If] the received signal is a definite function of the transmitted signal, then the effect may be called distortion. If this function has an inverse-no two transmitted signals producing the same received signal-distortion may be corrected at least in principal, by merely performing the inverse functional operation on the received signal.”
>
> – CLAUDE E. SHANNON

Analog to Digital Converter (ADC) can be also considered as an information channel. The transfer function of such a channel is not exactly invertible since the purpose of ADC is to map a range of the analog input to one digital value in the output. Still, the definition of distortion can be generalized to ADCs as well. In the case of uniform ADC (with a linear transfer function), the error in the output can be divided to two distinct categories:

1. Quantization Noise: Information loss due to mapping a range of the input to a single value at the output
2. Distortion: The deviation of the output from the ideal line

Although nowadays most ADCs are designed to be linear, which might be because of the ease of linear design, the desired transfer function of the ADC does not have to be linear. For instance, in many wideband applications the ideal function that transfers maximum information from input in analog domain to output in digital domain is exponential because of the Gaussian form of the probability distribution function (PDF) of the input. In this case distortion can be simply generalized to deviation of the transfer function curve from the desired curve. In this research we are looking for an efficient way to detect the distortion and correct for it, calibration.

The traditional approach in designing Successive Approximation Register Analog to Digital Converter (SAR-ADC) with binary weighted capacitive Digital to Analog Converter (DAC) is to make sure that the capacitors are large enough to meet the stringent matching requirement for the desired accuracy. In most cases, this approach (especially in range of mid-resolution ADCs) results in large capacitor requirement for DAC which is much higher than the thermal noise limit. It is much desired to reduce the size of the DAC capacitor since it directly and indirectly affects the total power and area of the ADC. The indirect impact on the power consumption is on the Low Dropout (LDO) Regulators that need to drive the DAC with high accuracy. Significant impact on the area is manily on the decoupling capacitors. Strict matching requirements also result in avoiding many high density geometries for capacitors. High density geometries exploit the vertical electrical field as well as lateral electrical field. Matching of the capacitors that use vertical field is worse because accuracy of the deposition is inferior to the accuracy of the lithography and etching[6]. In addition, in many practical situations it is desired to accommodate the ADC in odd shaped layout footprints which can negatively impact the matching of the elements.

The above mentioned reasons are among many reasons that it is desirable to be able to tolerate the mismatch between the DAC capacitors instead of avoiding it. If this is achieved there are many other opportunities on the horizon that can be also considered e.g. using mixed element DAC, combination of resistors and capacitors, for high throughput ADCs.

In this research we study how the mismatch of the capacitors of the DAC and other non-idealities in the ADC can convert to increase in quantization noise and how they affect the distortion. The outcome of this research is an efficient way of estimating the transfer function, its deviation from the ideal model, and explore the ways of correcting the distortion. It is very important for the proposed idea to keep the power and area overhead minimum and to make sure that the algorithms and methods are minimally invasive to the original system. In other words, we would like to avoid interrupts to the system in order to do the calibration. It is also desirable to be able to embed the algorithm of already well-known architectures without too much modifications of the original layout. Finally, the estimatation process should be stable, robust, and insensitive to other error mechanisms.

1.2 Prior Art

In this section a few of the state-of-the-art calibrations that are relative to the proposed algorithm in this work are briefly reviewed. Decision Boundary Gap Estimation (BDGE) [1] is a statistical based calibration which is implemented for pipeline ADCs. In this work the authors are looking for ways to detect the jumps in the characteristic of the ADC, named boundary gaps, in order to detect the error in the bitweights of the ADCs digital output. In order to do so it has been proposed to save the histogram of the codes of the output near the boundaries of the characteristic where a jump in the characteristic is expected. In this technique, it is assumed that the PDF of the signal is uniform near the boundaries since the range that is important for the calibration is small. Thus it is expected that the output codes of the ADC also have a uniform histogram distribution near the boundaries. However, if there is a gap in the histogram (Figure 1.1) the goal of the algorithm is to shift one side of the transfer function so the uniform histogram is achieved. A significant benefit of DBGE is that it is non-invasive to the original system and it can be done in the background. The main drawback of this work is that a lot of memory is needed for saving the data of the histogram. In addition this idea is suitable for ADCs in which redundancy exist in the output codes of the ADC similar to pipeline while in general SAR-ADC not necessarily has redundancy.

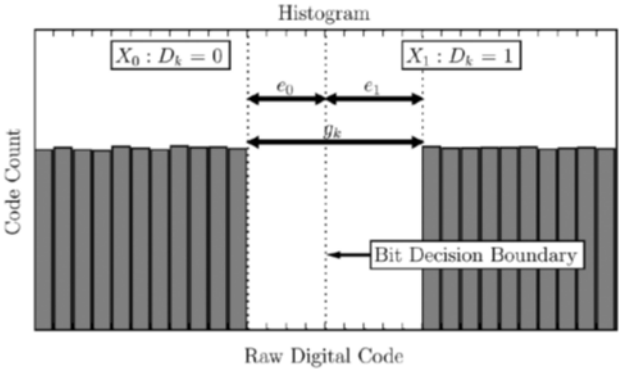


Figure 1.1: Example of a gap in the histogram of the output of the ADC in presence of nonidealities in the bitweights [1]

Another interesting state-of-the-art calibration technique is Dithering Method introduced in [2]. In this technique shown in Figure 1.2, inspired by dithering calibration in pipeline ADCs, a dithering method is proposed to detect the bitweights in SAR-ADC. In order to handle dithering the large MSB capacitor some extra clock cycles are added to the sequence of comparator decisions that reduces the sampling frequency of the ADC. In addition in order to be able to run the calibration in the background a Sample and Hold circuit is necessary.

Another very interesting calibration technique that has been implemented for SAR-ADC is introduced in [3]. In this work Figure 1.3 a pseudo random signal is injected along with the input to the ADC. The pseudo random signal equivalent is subtracted from output of the ADC. In a linear ADC with well selected pseudo random signal, there should be no trace of the pseudo random signal remaining in

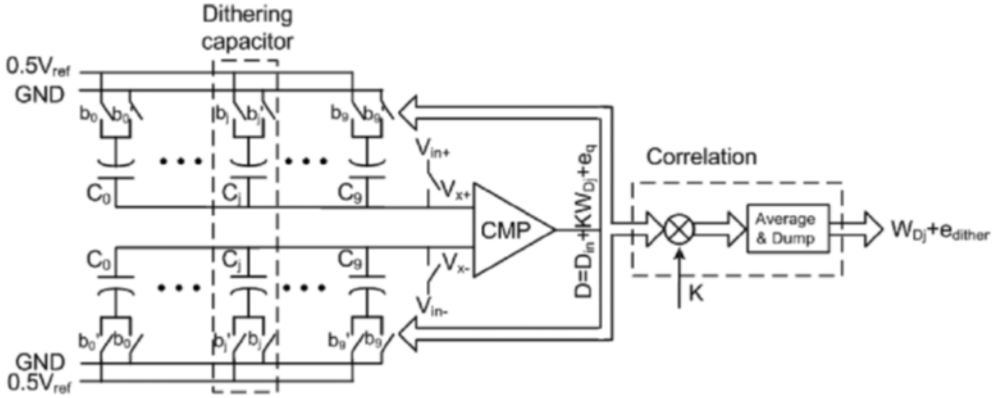


Figure 1.2: Dithering method in [2]

the ADCs output. However, if there is a nonlinearity in the transfer function of the ADC, traces of the pseudo random will be remaining which can be detected by correlating the output of the ADC with pseudo random signal. The result can be fed into a Least Mean Square (LMS) engine with feedback to the bitweights of the ADC in order to linearize the ADC. Although the technique is very powerful it is relatively complicated in terms of implementing the LMS engine. The convergence speed is also limited since there are two slow mechanism involved in the technique; First, a long correlation should be done to disengage the pseudo random signal from input signal. Second, LMS engine takes time to converge.

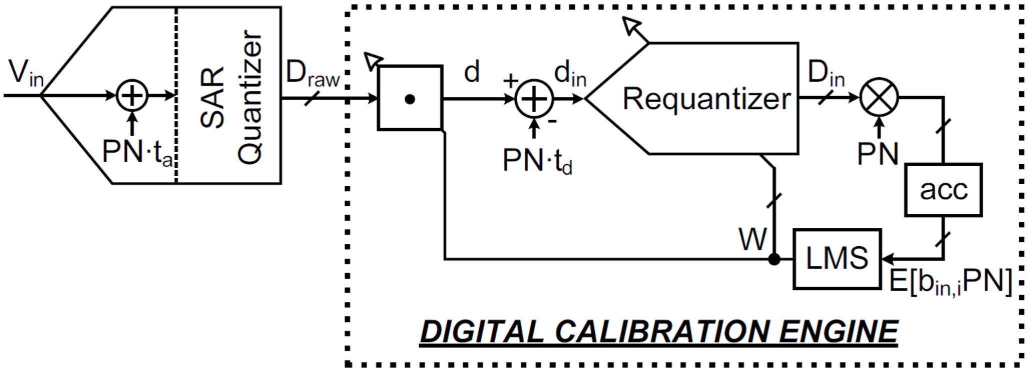


Figure 1.3: Block diagram of the calibration method introduced in [3]

CHAPTER 2

Proposed Calibration

2.1 Piecewise Linear Approximation

Piecewise Linear (PWL) approximation is a well-known technique to approximate a complicated curve in mathematics with simple lines as illustrated in Figure 2.1

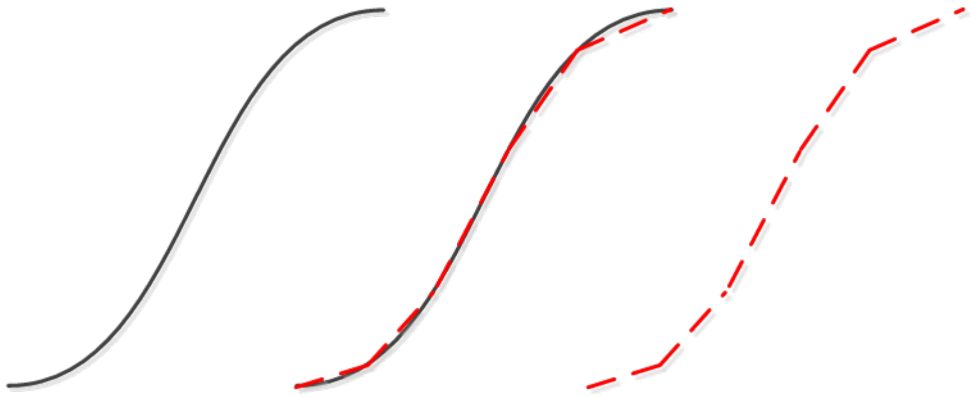


Figure 2.1: Transfer function of a system (on the left) can be approximated by pieces of straight lines (on the right) which can be a reasonable approximation for the transfer function of the system (in the middle)

PWL can be considered as the digitizer of the slope of the curve. That is, the amount of the information necessary to describe the curve with PWL is greatly reduced compared to the original curve. In this research we are interested to use PWL approximation of the transfer function of a system (for example SAR-ADC) and compare it with ideal model.

Because much less information is needed to describe PWL approximation of a system compared to the original system, any calculation for comparison to the PWL approximation of the model is also less hardware intensive which results in lower power dissipation and smaller area overhead. Fineness or roughness of the PWL approximation should be adjusted according to the target application and required accuracy. For large distortion contributors a course approximation is enough, however, for detecting small distortion the approximation intervals should be fine-tuned which can be done successively.

2.2 Statistical Estimation of PWL

Histogram is a great tool for measuring the slope of a function. If the input PDF is uniform, then the histogram of the output with equal bin-widths can reveal all the information required to estimate the PWL approximation of the transfer function. If the slope of the function is steep, the area of the input corresponding to a single bin is smaller compared to the case that the slope of the function is gradual. Hence, the count of the bin associated with steeper parts of the transfer function is smaller than the gradual parts of the function. Thus the relative height of each bin is inversely proportional to the slope of the transfer function. For example, if the function is linear, all the output bins will be equal.

An already well-known application of uniform input used for detecting the nonlinearity of the transfer function is measurement of static characteristics of ADCs, Differential NonLinearity (DNL) and Integral NonLinearity (INL), by means of a ramp signal as the input of the ADC.

The idea of measuring the nonlinearities of the transfer function by histogram of the output can be extended for non-uniform PDF signals. If the PDF of the signal is not uniform but *known*, then one can calculate what to expect as the histogram of the output and any deviation from the expectation can be considered as distortion of the transfer function. A famous example of application of such a method is measurement of the DNL and INL of the ADC by means of sinusoidal input to the ADC. This method is preferred over the ramp (uniform PDF) since generating an ideal ramp signal is very hard but generating a high quality sinusoidal signal is relatively easy and efficient with the help of narrow band filters.

In a practical applications the pdf of the input is unknown. In this research we propose the following algorithm to extract the information about the slope of each piece of the characteristic.

1. The input of the system is dithered with a random offset. For the sake of simplicity of the explanation one can assume that the dither is a random signal with equal probability (50%) of being either a positive or a negative offset (±OS). (It is better for the value of the OS to be close to the width of each bin of the histogram at the input of the system).

2. The output of the system, depending on the value of the corresponding offset, is sorted to two separate histograms. Thus a certain bin number (e.g. bin number 4 of the histogram related to -OS in the Figure 2.2) is corresponding to the neighboring bin in the other histogram (bin number 5 of the histogram related to +OS in Figure 2.2)

3. The neighboring histogram bins in each histogram cover the same range of the input, and the probability of input to be added to positive offset and negative offset is equal. Therefore, if there is a difference between the numbers of counts of these two bins, it is the result of the different slope of the transfer function at two neighboring pieces. The ratio of the count of these two bins (#5 in the +OS and #4 in the OS histogram) is the ratio of the slope of the pieces of PWL approximation of the transfer function. This information is enough to compare the PWL of the actual characteristic with the PWL approximation of the desired model and detect any deviations.

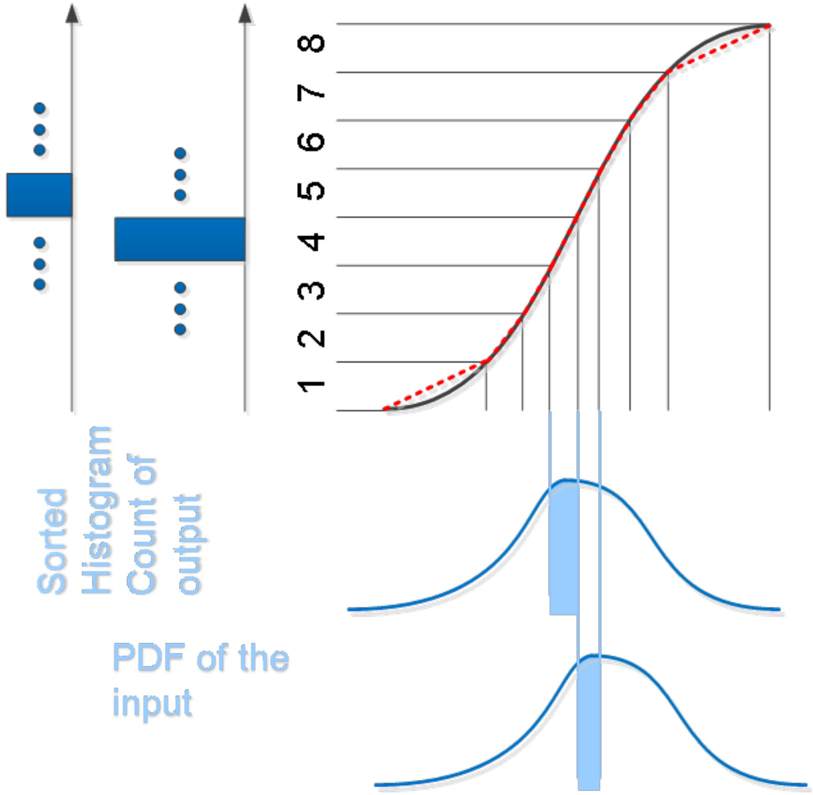


Figure 2.2: By introducing a random offset to the input of the system (on the bottom) and sorting the output of the system to separate histograms (on top left), Piecewise linear approximation of the system can be extracted

If the desired model is a straight line, any deviation from the ideal line can result in undesired harmonics at the output. The correction procedure can be designed to feedback the mean square root of the ratios of the histogram bins corresponding to the same input to the calibration system with the goal of reducing variation in the slope of the transfer function.

Binary weighted SAR-ADC is a very suitable system for the proposed calibration because of the following properties of its transfer function (shown in Figure 2.3):

1. It is inherently monotonic. Monotonicity is a requirement for correct estimation in every histogram based analysis [7]. It is noteworthy to mention that

a binary weighted DAC can be non-monotonic in the presence of mismatch in the elements. However, after the DAC is placed within a SAR-ADC the characteristic of the ADC will be monotonic but with missing codes.

2. It is Piecewise linear by nature.

3. Superposition holds for it. That is, if each bit error is independent of the state of the other bits, then the linearity error at any code is simply the algebraic sum of the errors of each bit in that code [7]. This property is also useful in reducing the hardware for extracting the error associated with each capacitor, especially the smaller MSB capacitors.

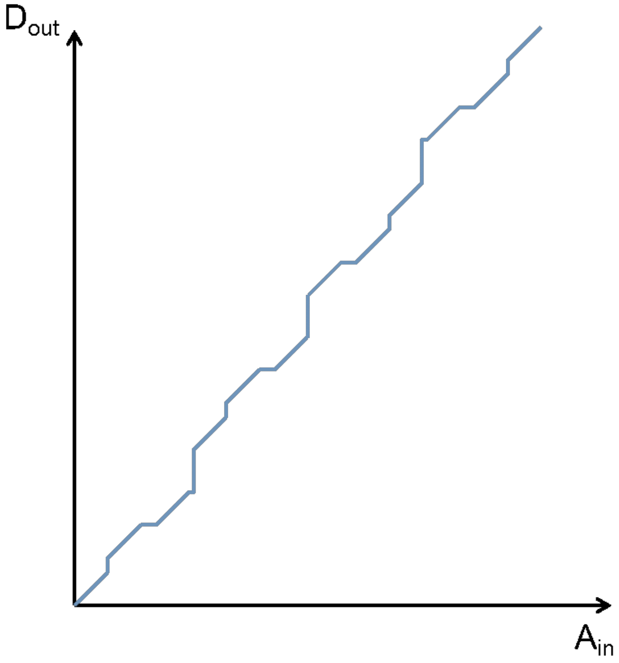


Figure 2.3: A typical Transfer function of SAR-ADC in presence of nonidealities

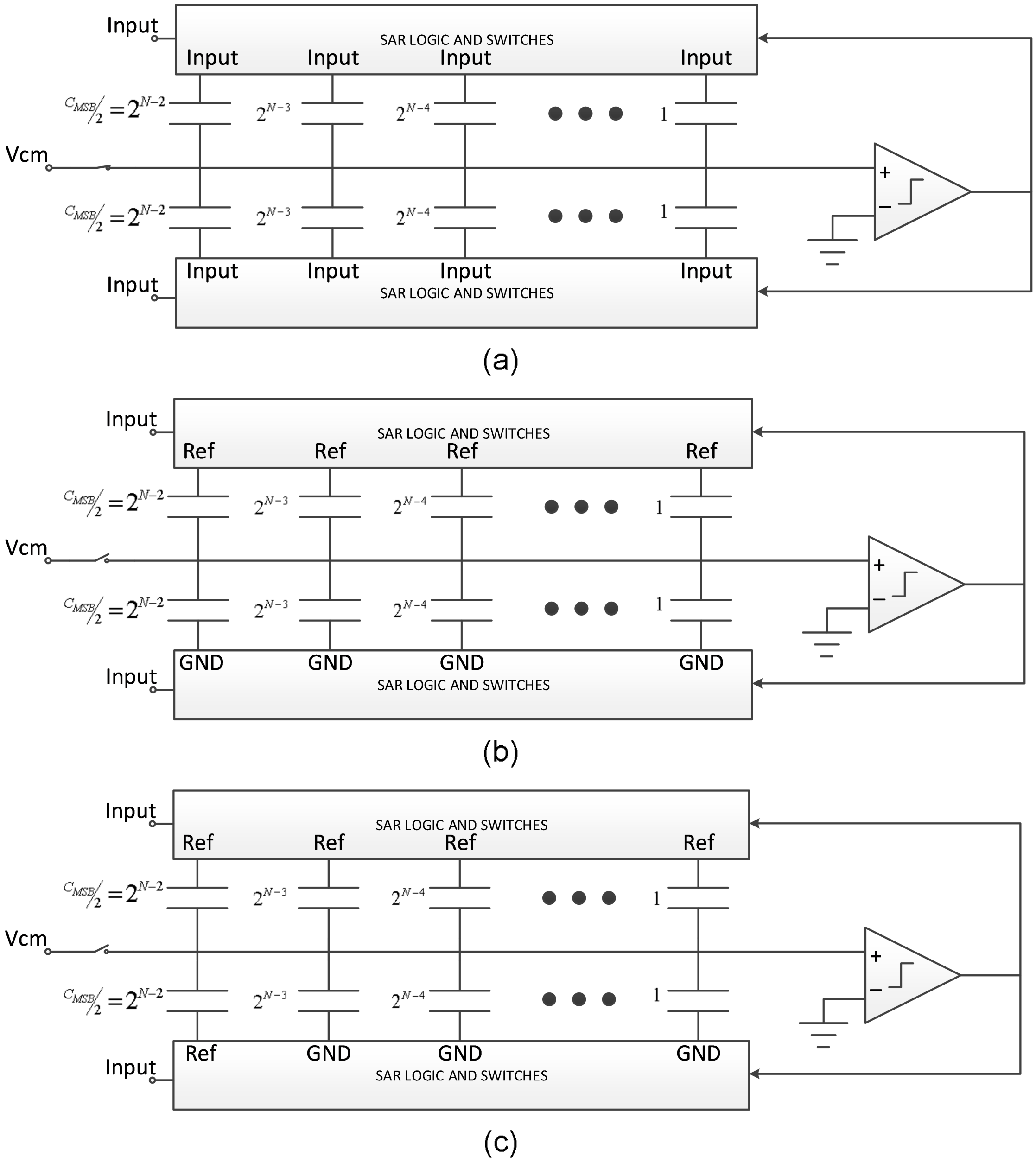
2.3 SAR-ADC and its error mechanisms

In this work we demonstrate the proposed calibration on a binary weighted split capacitor similar to the architecture proposed in [8]. In this architecture each capacitor of the binary bank is split into two equal pieces. As can be seen in Figure 2.4(a) the first step is to sample the input on all the capacitors. At this phase, all the bottom plates of the capacitors are connected to input and the top plates are connected to the common mode voltage. Figure 2.4(b) shows the configuration of the DAC for the first comparison (MSB). At this phase half of all the capacitors are connected to reference voltage (Ref) of the DAC and the rest are connected to ground (GND). After the first comparison is done, depending on the result of the first MSB, both pieces of the MSB capacitor will be connected to Ref (Figure 2.4(c)) or both will be connected to GND. The second MSB comparison can now be done and so on until the last capacitor is also switched. One last comparison is done to extract the LSB afterwards.

In this architecture of the DAC, each pair of capacitors (corresponding to each bitweight of the binary code in the output) can experience two error mechanisms.

1. After the conversion cycles are all done, both halves of each pair are either connected to Ref or GND. The ratio of each pair combined together, to the total capacitors in the DAC determines the weight of the bit associated to them. This bitweight might be different than the ideal bitweight. For example, as depicted in Figure 2.5, the MSB capacitors can have a mismatch of Δ_W to the ideal value C_{MSB}. The effect of such an error results in the transfer function to have a nonlinearity as shown in Figure 2.6(a) or Figure 2.6(b) depending on the sign of the error. This category of errors results in deviations from the straight line of the ADC for all the codes. They cause INL error trends and if the dynamic performance of the ADC is concerned, they result in concentrated power of the error in few harmonics.

2. The other error mechanism is the mismatch between two halves of each pair (Δ_S in Figure 2.5). The impact of such error is shown in Figure 2.6(c). In general, any temporary error that happens during the comparison of a



(a)

(b)

(c)

Figure 2.4: Sequence of a typical SAR-ADC with split capacitor architecture: Sampling phase (a), First comparison (b), Second Comparison (c)

particular bit during the SAR conversion but disappears soon after, results in an error in the transfer function as shown in Figure 2.6(c). In the illustrated case, Δ_S disappears after both half capacitors merge together. Temporary glitches in the reference voltage of the DAC or kickbacks have similar affect on the transfer function of the ADC. This category of errors are confined along a few LSBs in the transfer function in the neighborhood of the codes where the corresponding bit is switching. They result in DNL errors but the impact on INL is limited. The impact on the dynamic performance of the ADC is an error with small power which is spread among many harmonics, similar to quantization noise.

Figure 2.6(d)&(e) show the cases in which both bitweight errors and temporary errors occur.

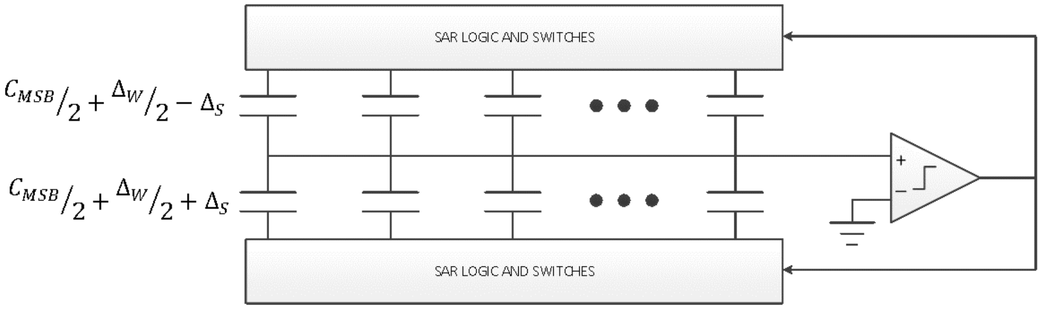

SAR LOGIC AND SWITCHES
$C_{MSB}/2 + \Delta_W/2 - \Delta_S$
$C_{MSB}/2 + \Delta_W/2 + \Delta_S$
+
−
SAR LOGIC AND SWITCHES

Figure 2.5: Possible error mechanisms in capacitors of split capacitor SAR-ADC

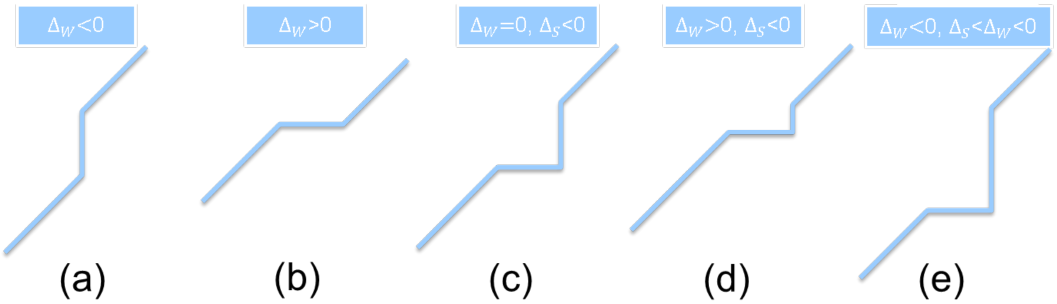

Δ_W<0
Δ_W>0
Δ_W=0, Δ_S<0
Δ_W>0, Δ_S<0
Δ_W<0, Δ_S<Δ_W<0
(a)
(b)
(c)
(d)
(e)

Figure 2.6: Impact of each error mechanism in capacitors of split capacitor SAR-ADC on the transfer function of the ADC

2.4 Methods of Calibration

As shown in Figure 2.7, calibration process can be considered as two separate steps; Estimation of the error and Correction.

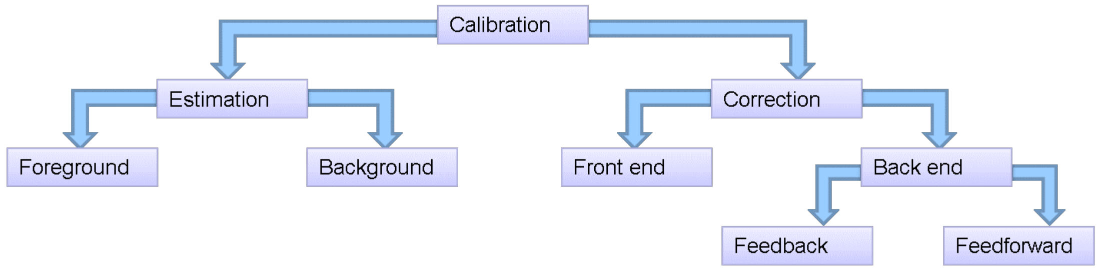


Figure 2.7: Calibration process is a two step process: Estimation and Correction. Each of these processes can be implemented in different ways

Estimation can be done in foreground or background:

- In foreground calibration, at the power up of the chip, a test signal with known properties is fed to the input of the ADC and the output is compared with the expected model, and the difference is considered as the error that has been cause by the ADC. After the foreground estimate is done it is assumed that the error mechanisms do not change. In some cases the throughput of the data is interrupted every often to do a calibration cycle. Foreground calibrations are usually faster in convergence, but there is a startup time associated with them that in some applications it might not be desirable. Also if the errors drift over time, the interrupts necessary for calibration might not be tolerable in some applications.

- Background calibration is done while in the ADC is doing its normal operation. A known signal is added to the input of the ADC along with the main input signal. At the output the known signal should be separated from the main signal and properties of the known signal are compared with expected properties according to the model to detect the error mechanisms. This process of separating two signals at the output requires long integration which results in slower calibration compared to foreground calibration. On the positive side, background calibrations do not have startup time associated with them which means ADC can start conversion of the main signal immediately (though not with full accuracy). Also if the error mechanisms change and drift overtime the calibration process keeps the track of the errors.

As for the correction part of the calibration process, it can be applied either at the front-end or at the back-end:

- In the front-end method, after the error mechanisms are detected, they are corrected at the source. For example, if a particular capacitor is detected to be smaller than the ideal value, some extra capacitors can be added to it from a capacitor bank or a designated calibration DAC can add or subtract some amount of charge to the main DAC as decided by estimation process. In theory this method of correction can correct the errors completely. However, in practice the calibration DAC or other elements used for correction, have non-idealities also. In addition, in cases similar to SAR-ADC applying the decisions of correction in the front-end means extra gates and delays in the main loop of the ADC, reducing the maximum achievable frequency.

- In contrast, back-end calibration accepts that some errors have happened during the ADC conversions but the purpose of such calibration is to interpret the process correctly to avoid adding more errors. Since here is no modification to the original system to accommodate the corrections in the front-end, this method of correction introduces no loading on the system. Also, because it is done at the back-end in the digital domain, it can benefit from all properties of digital system such as robustness and programmability. Figure 2.8 shows how a back-end calibration can fix misinterpretations and fix some of the errors. It is worthy to note that any back-end calibration can only shift segments of the characteristic in the vertical direction. As can be seen in Figure 2.8 after the back-end correction all the segments of the characteristic are inline. However, there are some notches in the characteristic that result from the fact that some information is lost because of the mistakes that have been made at the front-end.

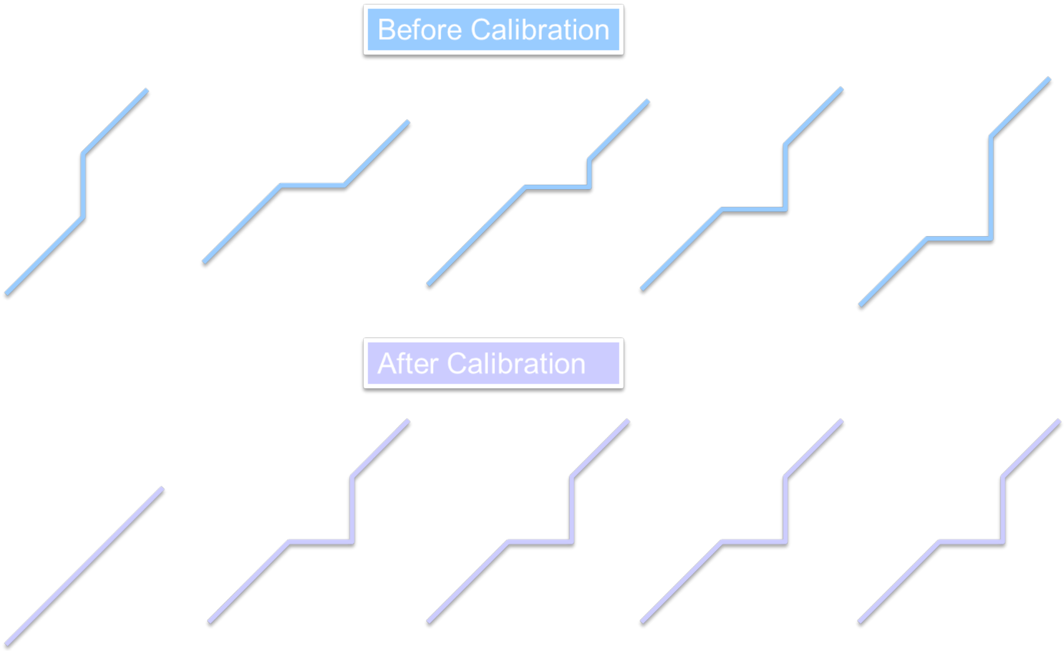


Figure 2.8: Impact of Back-end calibration on transfer function of the ADC

The impact of the information loss due to these notches on the ENOB of the ADC and the impact of the misinterpretation can be seen in Figure 2.9. In order to generate each curve, it is assumed that one capacitor of the DAC has an error (that is swept on the horizontal axis, Δ_W) while all the other capacitors are at their nominal value. In these families of plots each color coresponds to a different capacitor in the DAC (dark blue is the MSB). The bottom plots show if each capacitor of the DAC has an absolute error (normalized to unit capacitor) how it would affect the ENOB without any calibration. The top plots show if an ideal backend calibration is applied to the ADC how the ENOB would be improved, which shows the merit and limitation of back-end calibration.

It is interesting to note that the MSB capacitor is an exception, since when it is smaller than nominal value the drop in the ENOB is by far slower than when it is larger than the nominal value. This is because negative Δ_W after calibration becomes a straight line but positive Δ_W remains a notch behind (as shown in Figure 2.8). For other capacitors, error in the bitweight propagates to the larger capacitors and results in notches at larger bits. The MSB capacitor is an exception since it only changes the bitweight of the MSB but other capacitors change all the bitweights larger than themselves.

Another observation in these plots is that the sensitivity of the ENOB to the error in the LSBs is much more than MSBs. As a result, any error mechanism that results in notches in LSBs should be carefully controlled. As mentioned before, temporary changes in the references voltage during the conversion and unfinished settling can cause similar effects as the notches. However, during the LSB conversion, settling is usually fast enough. Reference voltage changes should be well controlled during LSB conversions.

A very effective technique to mitigate the notches in the transfer function is the use of redundancy. Figure 2.10 shows how redundancy can affect the transfer function by extending the length of each segment of the transfer function. As shown in the top row, when segments of the transfer function are not aligned due to error in the bitweights, introducing redundancy is not helpful. However, with presence of back-end calibration when all the segments are inline, redundancy can solve the problem of loss of information.

As shown in Figure 2.7 back-end correction can be implemented through a feedback loop or by the means of feedforward error correction. We will discuss about their differences and application more in section 2.6.

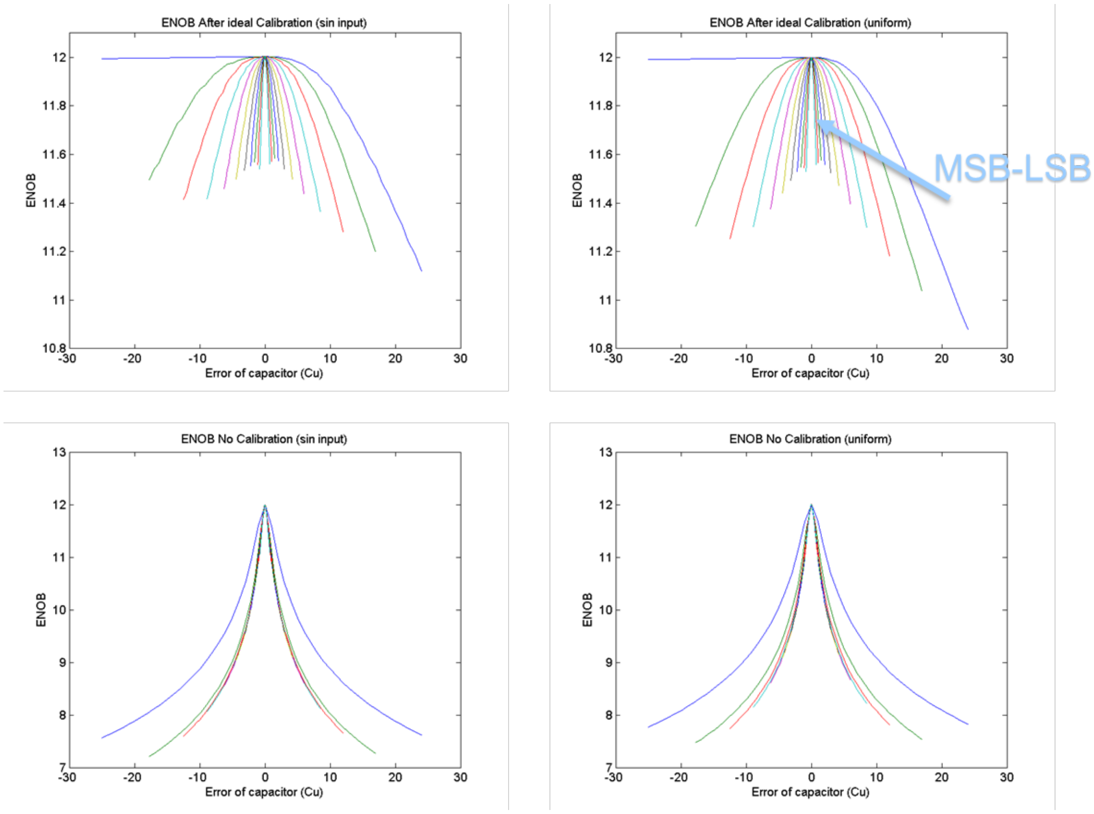


Figure 2.9: Impact of error in the value of each capacitor in the ENOB, if the input of the ADC is a sinusoidal (left plots) or if the input is a random signal with uniform PDF (right plots). Back-end claibration can improve the ENOB (top plots) compared to no calibration (bottom plots)

2.5 Implementation of Estimation

In order to illustrate the implementation of the estimation process in SAR-ADC, we assume the ADC has nonlinearity only due to the MSB capacitor ($\Delta_W > 0$). In this case, nonlinearity results in "wide code" (Shown in Figure 2.11) which is often avoided in prior art because of hardship in estimation and correction of it. As mentioned before, a part of the proposed calibration is to obtain a histogram of the output of the ADC. Therefore, the output codes are sorted in histogram bins with Nominal Bin Width (NBW) e.g. 64 LSB. Meanwhile, the input is dithered with an offset of either positive (+OS) or negative (-OS). Assuming the pick-to-

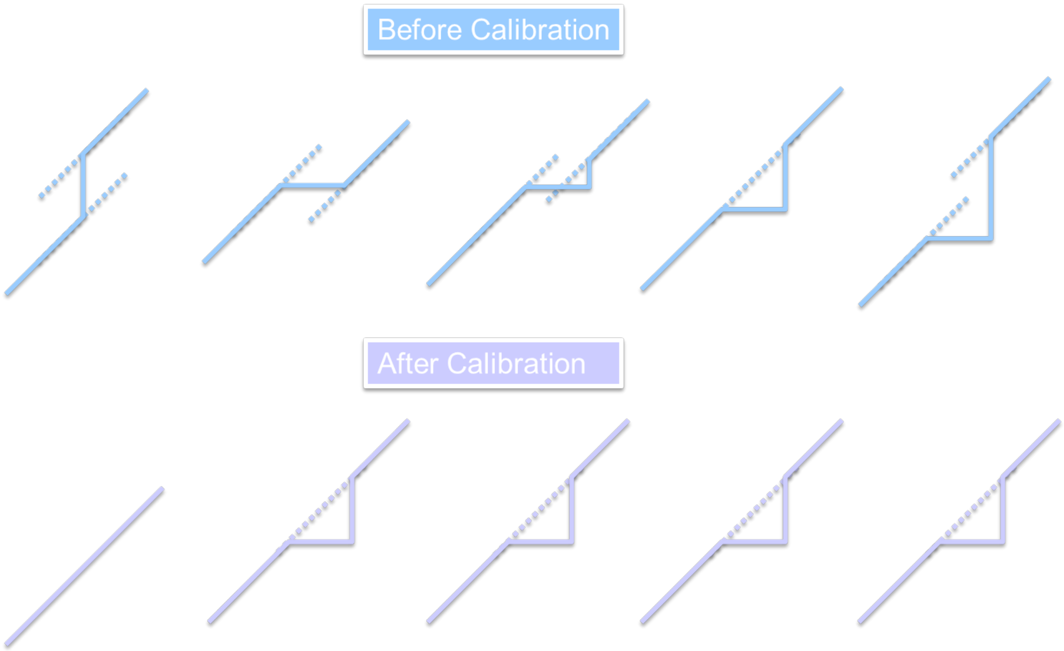


Figure 2.10: Redundancy can recover the lost information if all segments of the ADC are inline (bottom plots). However, if the segments of the transfer function are not inline, redundancy can not help (top plots)

pick amplitude of the dither (2*OS) is equal to NBW, after sorting the output codes of the ADC into two separate histograms, count number of N1 in the positive dither histogram and count number of N0 in the negative dither histogram are associated to the same input range. Thus the value of the error in the bitweight of the MSB (Δ) can be calculated as:

$$\frac{N0}{NBW + \Delta} = \frac{N1}{NBW} \tag{2.1}$$

or

$$\Delta = NBW \times \left(\frac{N0 - N1}{N1} \right) \tag{2.2}$$

Circuit implementation of this equation can be very simple by making some smart choices. The subtract operation in the nominator can be easily implemented by an UP/DN counter. Whenever the dither is positive and the output is detected to be in the range of the 4th bin, the counter counts down and whenever the dither is positive and the output is detected in the 3rd bin the counter counts one up. The denominator is a simple counter. If the value of the NBW is chosen to be a power of two number (2^m) and the counter of the denominator is a k-bit counter, then the carry out of the denominator counter can be used as a signal to a shift register to fetch the output of the nominator counter and shift it to the right for m-k times.

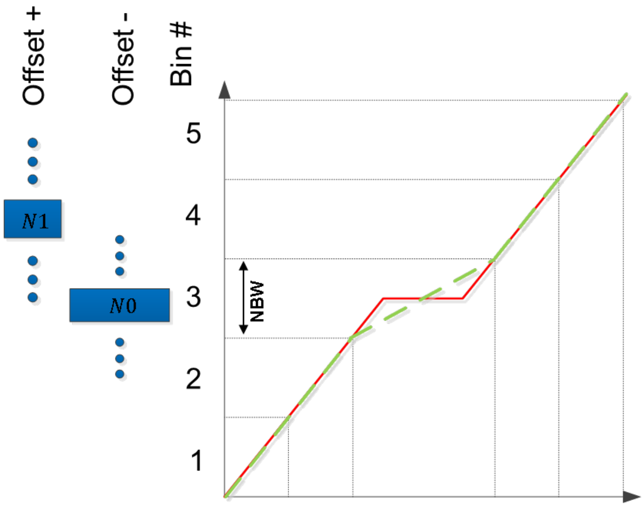


Figure 2.11: By sorting the output of the ADC in separate histograms depending on the value of the offset in the dither (on the left) a Piecewise linear approximation (green curve) of the transfer function (red curve) of the ADC can be obtained

An important step in the proposed calibration that can greatly affect power and hardware efficiency is the process of detecting the histogram bins. In theory, the PDF of the dither signal is not required to be uniform, however choosing it to be uniform can result in avoiding a normalizing step of the histogram sets which is a costly division process.

Another power and area saving solution is to combine as many histogram bins as possible together in order to reduce the number of memory cells. For example, we can assume there is an error in the third MSB. The transfer function experiences discontinuity at 7 points (Figure 2.12). At four of these point, marked green, all the larger MSBs are the same on both sides of the discontinuity. Thus the value of Δ calculated at these 4 points depends only on smaller capacitors. If the lower bitweights are accurate (or equivalently their bitweights are calibrated) Δ can be acurately calculated.

The three remaining discontinuity points are not used for current Δ calculation (the 3$^{\text{rd}}$ MSB in this example) but they will be used in subsequent steps to calculate for Δ of larger MSBs.

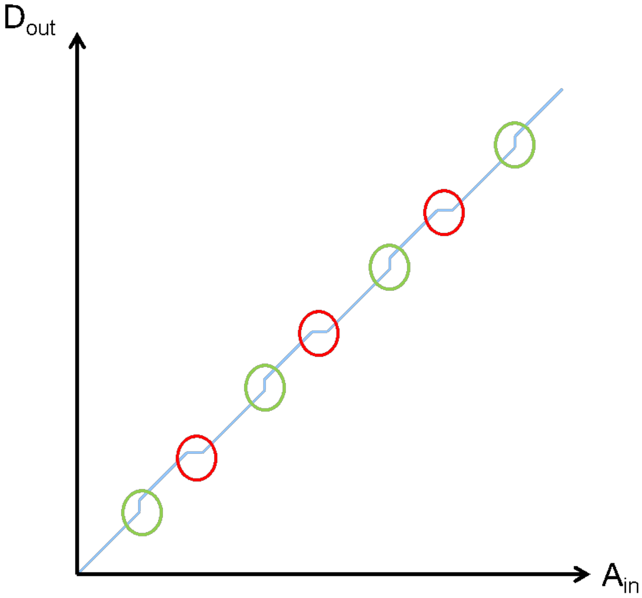


Figure 2.12: An error in the 3$^{\text{rd}}$ MSB capacitor can cause discontinuity in 7 points. Four of these points (green marks) are used to extract the bitweight of the 3$^{\text{rd}}$ MSB, the rest (red markes) are used to estimate the larger MSBs

In order to reduce the hardware, especially the registers, it is necessary to avoid storing the information related to every nonlinearity point of the transfer function separately. Thus we combine all the histogram bins associated to the same bitweight (green marks in Figure 2.12). To further explain this idea, output of the ADC can be considered as its bit sequence:

$$\begin{aligned} Raw\_ADC\_output\_code = {} & D_{MSB}D_{MSB-1}D_{MSB-2}\dots D_{cal}D_m \\ & D_{m-1}D_{m-2}\dots D_{NBW-1}D_{NBW-2}D_{NBW-3}\dots D_{LSB} \end{aligned} \quad (2.3)$$

where D_{Cal} is the bitweight under calibration (3^{rd} MSB in the example of Figure 2.12), D_{MSB}'s are the larger bitweights in which all of their permutations need to be collected but they should be constant within each bin. D_m's are the lower bitweights that are either accurate enough or have been calibrated. These bits are all the same for the codes near the discontinuities and they are opposite to the D_{cal} when located in the green marks. D_{NBW}'s are LSBs that all their permutations should be collected to form the NBW wide bin.

Also, if NBW is chosen to be a power of two number, all the LSBs that toggle within one NBW will not affect the calculated Δ.

Table 2.1 shows the binary representative of the output codes near the discontinuities of Figure 2.12. The green codes are the output codes that are necessary to be collected in order to calibrate the 3^{rd} MSB with NBW of 4 in an 8-bit ADC. The logic for this operation is shown in Figure 2.13.

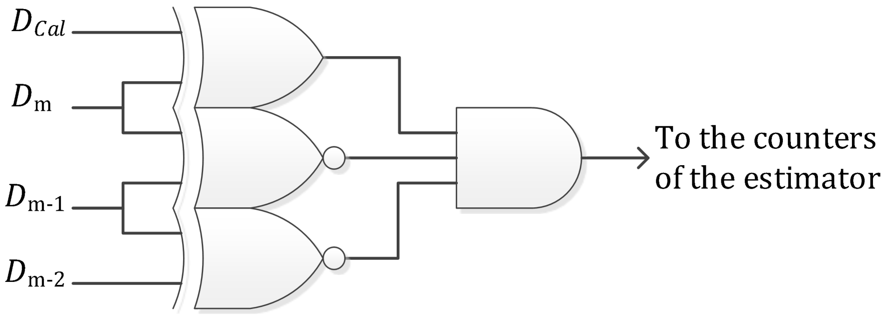
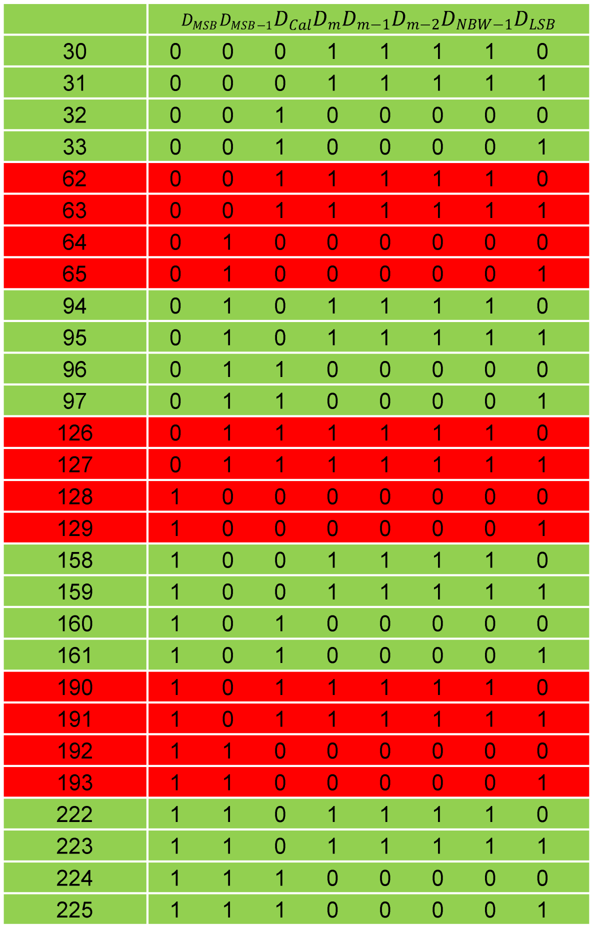


Figure 2.13: A typical logic to detect the number of counts in combined bins of the histogram

The flowchart diagram of Figure 2.14 shows the implementation of the estimation. On the Left, the raw data from the output of the ADC and the pseudo

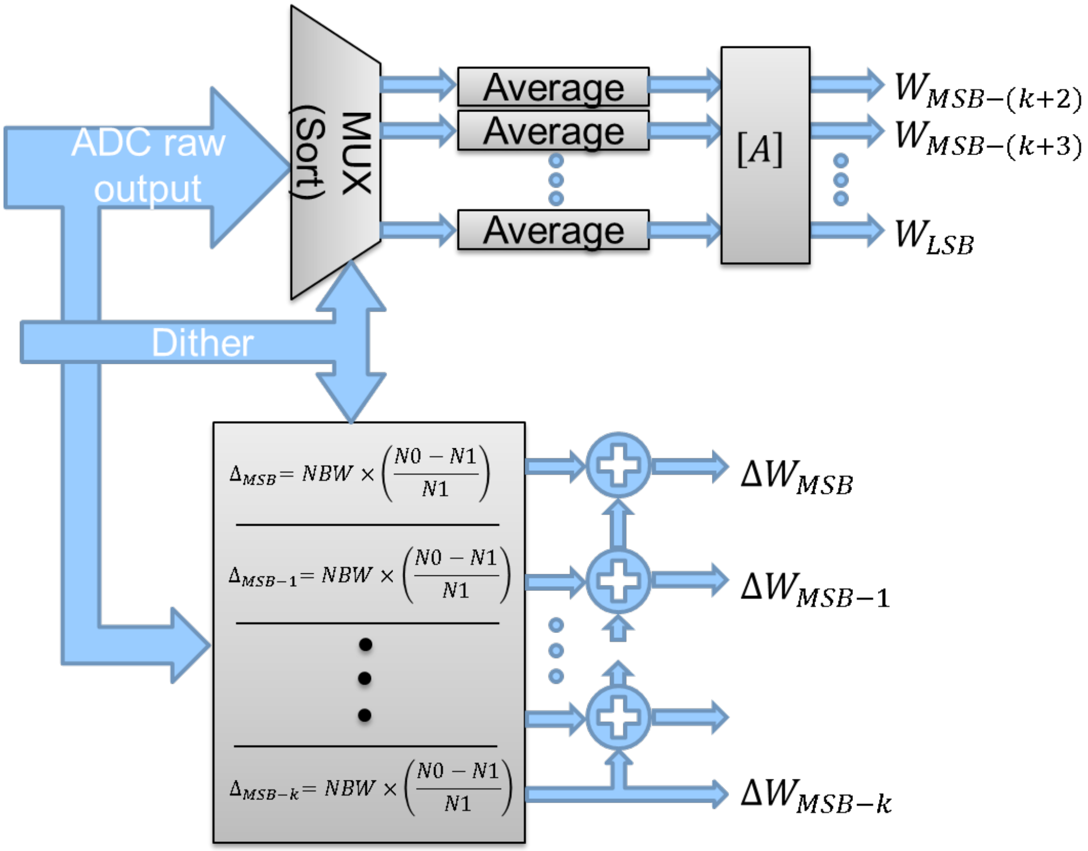
Table 2.1: sample of output codes near the discontinuities



	D_{MSB}	D_{MSB-1}	D_{Cal}	D_m	D_{m-1}	D_{m-2}	D_{NBW-1}	D_{LSB}
30	0	0	0	1	1	1	1	0
31	0	0	0	1	1	1	1	1
32	0	0	1	0	0	0	0	0
33	0	0	1	0	0	0	0	1
62	0	0	1	1	1	1	1	0
63	0	0	1	1	1	1	1	1
64	0	1	0	0	0	0	0	0
65	0	1	0	0	0	0	0	1
94	0	1	0	1	1	1	1	0
95	0	1	0	1	1	1	1	1
96	0	1	1	0	0	0	0	0
97	0	1	1	0	0	0	0	1
126	0	1	1	1	1	1	1	0
127	0	1	1	1	1	1	1	1
128	1	0	0	0	0	0	0	0
129	1	0	0	0	0	0	0	1
158	1	0	0	1	1	1	1	0
159	1	0	0	1	1	1	1	1
160	1	0	1	0	0	0	0	0
161	1	0	1	0	0	0	0	1
190	1	0	1	1	1	1	1	0
191	1	0	1	1	1	1	1	1
192	1	1	0	0	0	0	0	0
193	1	1	0	0	0	0	0	1
222	1	1	0	1	1	1	1	0
223	1	1	0	1	1	1	1	1
224	1	1	1	0	0	0	0	0
225	1	1	1	0	0	0	0	1

random dither offset are the inputs of the calibration block. The block on the bottom part of the chart calibrates the MSB capacitors. As mentioned before, detection of the error in smaller bitweights is not dependent on the larger MSBs, but the value of the error of the smaller MSBs should be added to the larger MSBs to extract the actual error. All the capacitors, except capacitors which participate in the NBW and dithering, can be calibrated this way.

The top part of the chart is used to extract the bitweights of the LSB capacitors, which participate in pseudo random dithering. In order to do this, the raw output data of the ADC is sorted according to the value of the dither. Until this point our examples of the dither were only ±OS, however, it is possible to use more value for the pseudo random signal. Thus after sorting the raw output of the ADC according to the value of the pseudo random, the output of the ADC is fed to a different accumulator/average block. From the relationship between the outputs of the "average" blocks, bitweights of the LSB capacitors can be calculated. The calculations can be substantially simplified if the possible values of the dither is designed to be only powers of two. In this case, matrix A has only two nonzero values at each row: value "1" on the dignal of maxtrix A and value of "-1" of the colum associated with the dither zero. This makes the implementation of matrix A very simple.



ADC raw output
MUX (Sort)
Average
Average
Average
$[A]$
$W_{MSB-(k+2)}$
$W_{MSB-(k+3)}$
W_{LSB}
Dither
$\Delta_{MSB} = NBW \times \left(\frac{N0 - N1}{N1}\right)$
$\Delta_{MSB-1} = NBW \times \left(\frac{N0 - N1}{N1}\right)$
$\Delta_{MSB-k} = NBW \times \left(\frac{N0 - N1}{N1}\right)$
ΔW_{MSB}
ΔW_{MSB-1}
ΔW_{MSB-k}

Figure 2.14: Block diagram of the estimation block of the proposed calibration; Inputs are (on the left) ADC raw output codes and dither. The outputs of this block diagram are the error in the MSB bitweights and the value of the LSB bitweights (on the right)

It is a very important condition for NBW to be large enough to cover all of the discontinuity as shown in Figure 2.15. Although, the larger the NBW, the lesser the accuracy of the estimator. Thus it is beneficial to have an estimate of the statistics of the possible errors in the ADC to make sure that the NBW is just large enough to cover the discontinuities at the worst case.

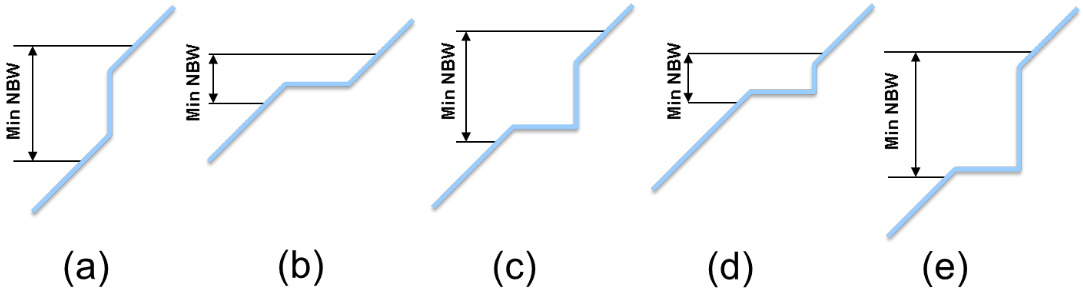


Figure 2.15: NBW must be large enough to cover all of the discontinuity at each point

Since the error of the lower MSBs are used in all the higher MSBs, any inaccuracy in the detection of the error propagates up to the rest. It is beneficial to use smaller NBW for lower MSBs to increase the accuracy.

2.6 Implementation of Correction

As was mentioned previously, after the estimation block detects the errors in the bitweights and the decision has been made to do the correction at the back-end, there are two options to follow: doing the correction in feedforward or in feedback as shown in Figure 2.16.

Feedback has the advantage of being less sensitive to the accuracy of the dither signal or if the PDF of the signal has sharp transitions. However, because of the method of the detection proposed in Figure 2.16 (that the errors of MSBs depend on error of the smaller capacitors), the feedback topology results in many loops

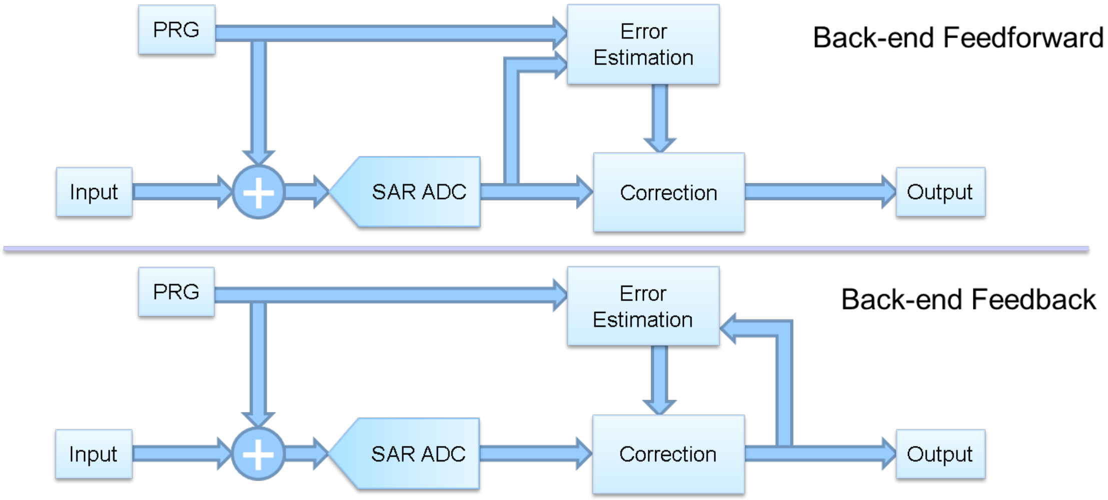


Figure 2.16: Feedforward implementation of the back-end calibration uses the raw outputs of the ADC as the input of the estimation block (top) while feedback implementation uses the corrected codes at the output of the correction block as the input of the estimation engine (bottom)

inside each other, which makes it very hard to stabilize the system. To compound the problem, the time index of each of the loops are random too, which makes it extra difficult to control. This arises from the fact that occurrence of output within each bin associated to different bitweights are random,

Feedforward method is stable, but care should be taken to make sure histogram bins of two neighbors and therfore the pieces of the PWL approximation do not overlap. Also, large variations of PDF near the discontinuity point can compromise the accuracy of the detection. However, since the PDF of thermal noise is convolved with the PDF of the input signal it is realistic to assume the PDF has small and smooth variations within a couple of LSBs.

A Matlab simulation on a random capacitive 12-bit SAR-ADC shown in Figure 2.17 shows the difference between the convergence of the feedforward and feedback implementation of the back-end correction. In practice, stability issue of the feedback scheme occur when the denominator counter of the lower MSBs fills up at

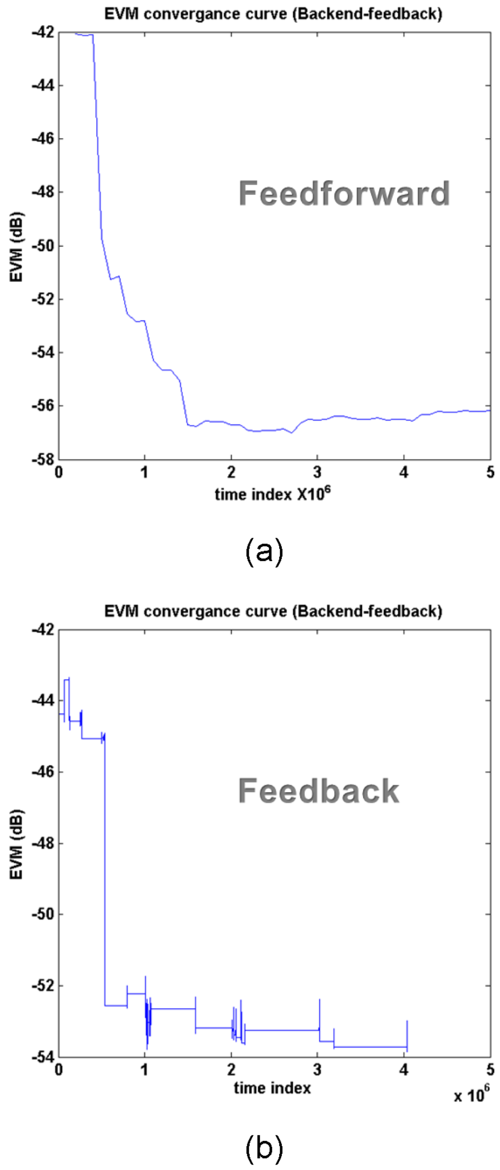


Figure 2.17: Convergance of the back-end calibration in an EVM test over time for feedback implementation (bottom) and feedforward implementation (top)

a much faster rate than the larger MSBs. Since every time the lower MSBs update their bitweight the MSBs should restart their counters to count with the updated lower bitweights, the MSBs never have a chance to be updated. These issues can be controlled by imposing constraints of how often each bitweight updates compared to the rest of the bitweights which in essence is the famous gain-bandwidth tradeoff of feedback loops.

Figure 2.18 shows a Monte Carlo simulation of 400 ADCs with 6% accuracy of the unit capacitor. The input of the ADCs is a simulated samples of an 802.11ac 256-QAM including the preambles. Effectiveness of the calibration process can be observed by noticing that the far end of the distribution of the EVM is improved about 10dB.

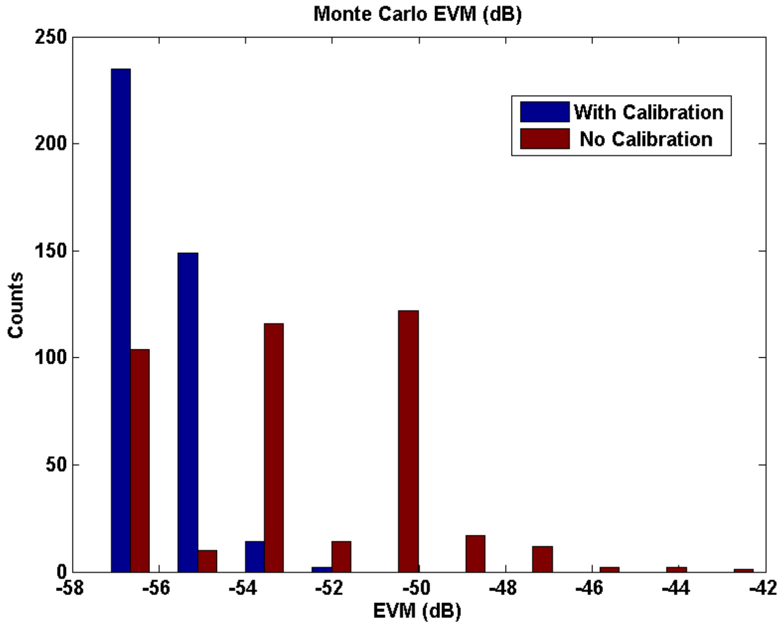


Figure 2.18: Monte Carlo simulation result of 400 SAR-ADC with 12-bit resolution and gaussian random mismatch with σ =6% in the unit capacitor

Since the calibration process depends on the support of the input signal to detect the nonlinearities, a valid concern, is “What if the input signal does not cover all of the transfer function of the ADC?” The answer is to this concern is pretty simple: “If it rarely happens it barely matters.”

For example, if the support of the input signal is around the mid-point of the ADC (experiencing the nonlinearity of the MSB capacitor) but not large enough to cover more than half of the ADC dynamic range (the discontinuity points of second MSB are not covered) the calibration does not correct the bitweight of the second MSB independantly. Instead, any error in the bitweight of the second MSB will be added to the first MSB. In other words, MSB capacitor and second MSB are not switching independently and as far as the calibration is concerned, their lumped weight is being calibrated.

Anywhere the signal has higher probability, the calibration converges faster and more accurately and anywhere that the input signal is less probable the calibration is slower and less accurate. This is acceptable since the lower the probability of the signal, the lower the probability of error at that point, hence the lower impact on the signal to noise ratio.

CHAPTER 3

Circuit Implementation

3.1 Digital to Analog Converter

Figure 3.1 shows the high level circuit implementation of the SAR-ADC prototype that has been fabricated in CMOS TSMC 65nm GP to evaluate the effectiveness of the proposed calibration. The architecture is an asynchronous 12-bit split capacitor DAC with one bit redundancy (inspired by [9]) at 64 LSB. Dither is a 6 bit code that is generated from a 16-bit linear feedback pseudo random (LFSR) generator. The length of the LFSR is chosen long so in case the dither is not removed completely from the output the remaining leakage does not introduce a tone in the output spectrum.

During the sampling Phase (Figure 3.1(a)), the input is sampled on all MSB capacitors while the LSB capacitors are charged with the value of the dither. This way the dither offset necessary for the calibration of MSBs is introduced to the ADCs transfer function and the bitweight of the LSBs also can be extracted. It should be mentioned that there is a penalty in the dynamic range of the ADC which in this case is 64 LSB, only -0.14dB.

Figure 3.1(b) shows the configuration of the capacitors for the first MSB comparison. All the MSB capacitors are connected according to the split capacitor architecture while the LSB capacitors are connected according to monotonic architecture [10]. In monotonic architecture, if the output of the comparator is high, both halves of the corresponding capacitor at the positive side of the DAC,

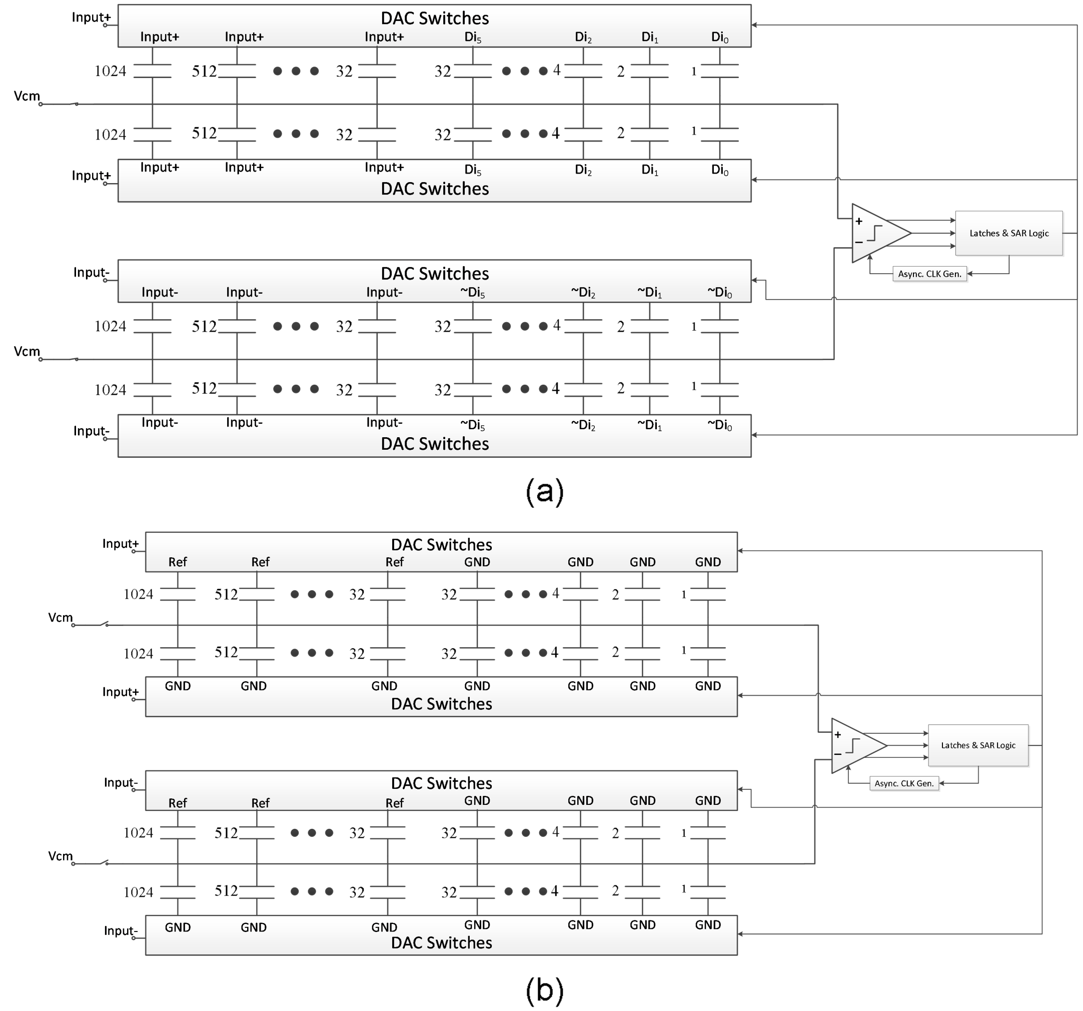


Figure 3.1: Sampling phase (a) and MSB comparison phase (b) of the proposed ADC. Dither signal is injected to the input of the ADC by controlling the LSB capacitors

switch from GND to Ref. If the output of the comparator is low, corresponding capacitors at the negative side switch from GND to Ref.

Monotonic architecture is used in LSBs to reduce the logic delay in the LSB conversion. The well-known drawback of the monotonic architecture is variations of the common mode voltage at the input of the comparator that can result in different comparator offset in different offsets. This mechanism of error results in similar error to Δ_S in Figure 2.5. Since these errors happen after the redundant cycle, they won't be corrected. Since this method is only used in LSB capacitors, the change in the common mode is only 32 LSBs which will not result in significant errors.

3.2 Comparator and SAR-Logic

Figure 3.2 shows the connection between the comparator, the SAR-Logic, and Asynchronous clock generator. The output of the comparator is connected to the input of all the latches (in this case 13 latches). Since the SAR-ADC is implemented with asynchronous architecture, a valid signal from the comparator also indicates when the data at the output of the comparator is valid. This signal is used as the clock for the latches to pick up the result of the comparison.

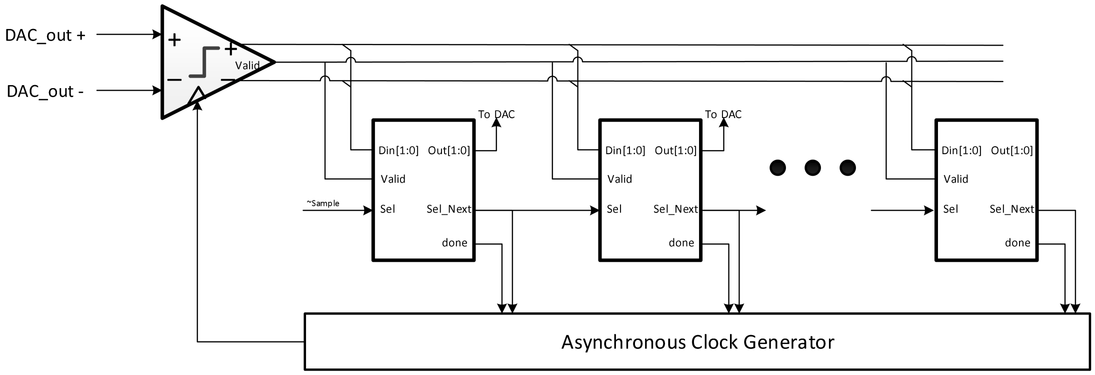


Figure 3.2: Connection between the comparator, latches, and the asynchronous clock generator in the proposed ADC

3.2.1 Latches

Circuit implementation of the clocked latches is shown in Figure 3.3. The latches fulfil both duties of storing the output of the comparator for the DAC and triggering the next latch similar to a one-hot state machine.

During the sampling cycle of the ADC both outputs of the latch are reset to VDD. After the sampling is done the output of each latch remains in this state (‘11’), which through a simple decoder connects half of the capacitor of the corresponding bit to Ref and the other half to GND. At each moment only the select signal (Sel) of one latch is high, which makes this latch fetch the output of the comparator at the rising edge of valid. Depending on the output of the comparator either the state of the latch converts to ‘10’ or ‘01’, which decodes to proper switching in the DAC. Meanwhile, a done signal is generated which is used to create the asynchronous clock of the comparator. Done signal also makes the latch opaque to the comparator for the rest of the comparisons. After the asynchronous clock propagates through the comparator, valid signal goes low and the current latch signals the Sel input of the next latch to take the turn of fetching the output of the comparator. This handshaking procedure guarantees that only one latch at a time is transparent to the output of the comparator and saves power by eliminating the need for a separate state machine to control the latches.

3.2.2 Comparator

As seen in Figure 3.2, the outputs of the comparator are connected to many latches which results in a large capacitor at these nodes. As it has been pointed out in literature [4][11][12] in a dynamic latch (e.g. Strong-ARM showin in 3.4) this capacitor is actually useful in reducing the input referred noise of the comparator. However, in a Strong-ARM latch comparator two distinct duties with contrasting tradeoffs are carried at the same time: amplifying the input to increase the signal-to-noise-ratio (SNR) and regenerating to full swing digital values at the output. Strong-ARM does not consume static power, which is the reason that it became so popular compared to other comparator architectures. However, because of the large voltage swing at the output nodes that need to have large capacitors, the dynamic power consumption (fCV^2) is large.

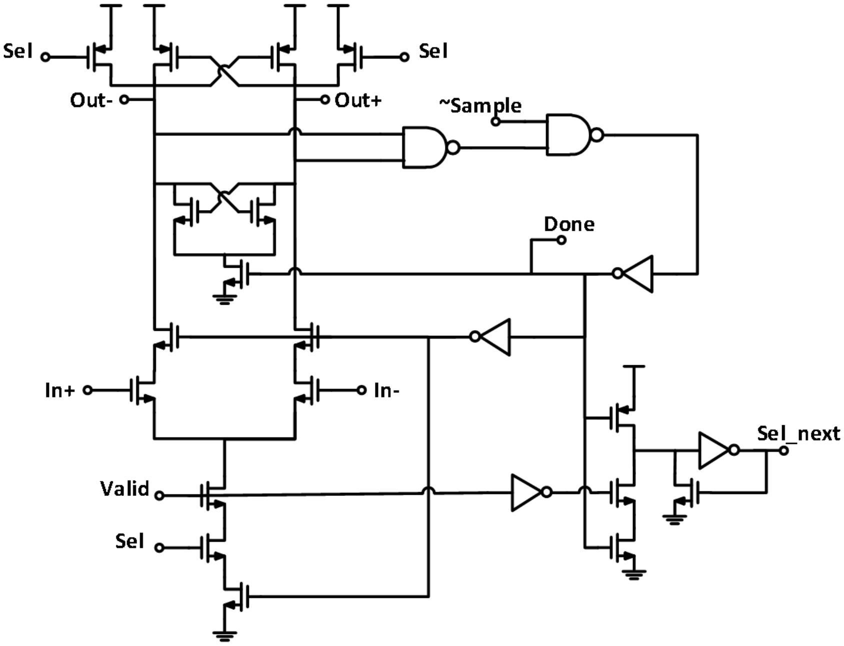


Figure 3.3: Architecture of the proposed latch, including the logic for handshaking with other latches, comparator and the asynchronous clk generator

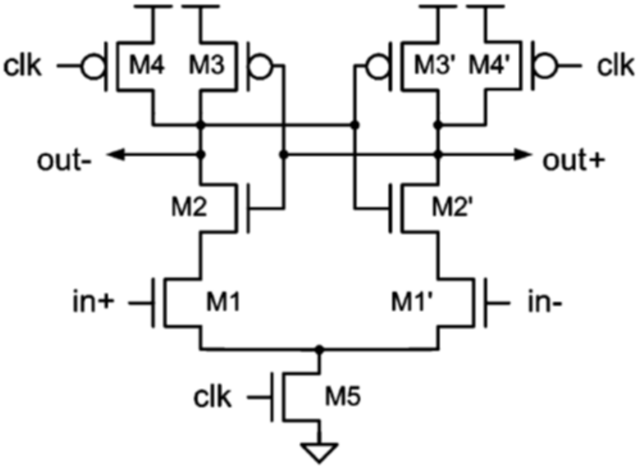


Figure 3.4: Strong-ARM Latch [4]

Figure 3.5 from [4] shows a the signal trends in a dynamic comparator which is the main inspiration of the proposed comparator in this work. Since Impulse Sensitivity Function (ISF) of the comparator (also explained in detail in [13]) is limited in time it means that it is not necessary for the input differential pair to consume any more power after ISF returns to zero. Also the Bit Error Rate (BER) of the comparator, which eventually determines the input referred noise of the comparator, depends on the SNR and the end of sampling phase when ISF returns to zero. After this time regeneration starts that does not affect the input referred noise of the comparator [12]. Since the latch of Figure 3.3 completes the regeneration process and ensures digital full swing levels at the output suitable for the DAC, there is no need for the comparator to consume energy to regenerate the voltages to full swing on the heavily capacitive nodes of its output. We propose the comparator of Figure 3.6 which can also be considered as a dynamic preamplifier rather than comparator since the output is a high SNR signal but not a full swing digital signal.

The outputs of the comparator are reset to VDD during the reset phase. After the rise of the CLK signal, the outputs of the comparator (with large capacitors due to input capacitors of the latches, routing, and noise considerations) experience similar trends of common mode and differential as [12]. When output common mode reaches one PMOS threshold voltage below VDD, the cross couple turns on and regeneration starts. At this time a common-mode sensing circuit detects the drop in the common-mode voltage and turns off the tail of the differential input. The same signal with some delay can also be used as valid signal for the latches. Figure 3.7 (from top to bottom respectively) shows differential input of the comparator, common mode of the input of the comparator, Asynchronous CLK, comparator outputs, and valid signal. As can be seen in this figure, the voltage swing at the outputs of the comparator (especially at LSBs) are reduced which results in significant dynamic power savings. The common mode of the output nodes returns toward VDD after the tail is turned off which is desirable since it reduces the required reset time to avoid the hysteresis in the comparator. But, the differential mode of the outputs keeps growing which is due to the fact

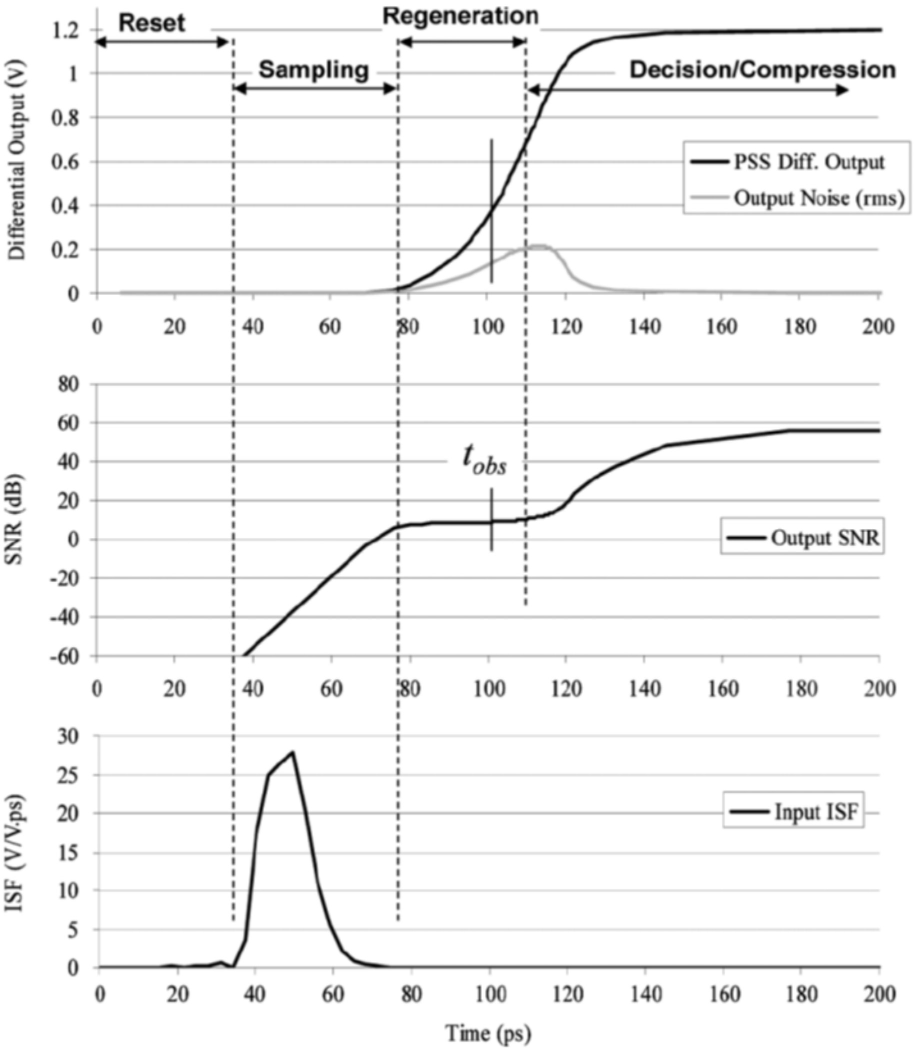


Figure 3.5: Simulated waveforms of the strong-ARM comparator [4]

that PMOS cross couple is regenerating, though this regeneration phase does not impose power penalty.

In low frequency application, this power free regeneration phase can be exploited to further reduce the shoot-through power of the latches since when latches drive with larger inputs, they remain less time at metastable phase which means less power consumption. In high frequency application, care must be taken to insure the comparator increases SNR to an acceptable value so that the noise

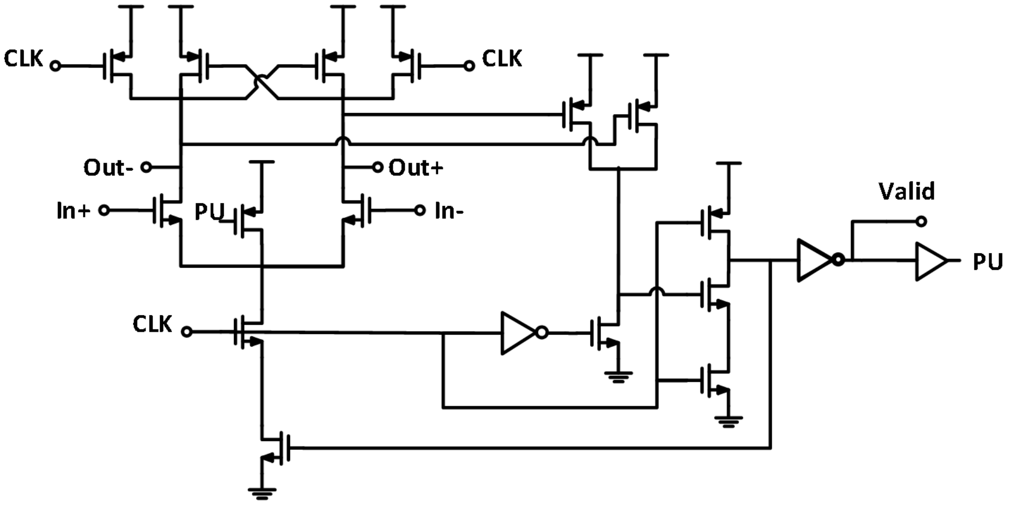


Figure 3.6: Proposed comparator

of the latches do not become significant contributor to the input referred noise. In addition, the proposed comparator has much less kickback owing to the low voltage swings.

3.3 Clock Generator

The asynchronous clock generator is shown in Figure 3.8. The "done" signal of each latch generates a pulse that pulls down the gate of a tri-state inverter that consequently pulls down the asynchronous CLK of the comparator. As described before in the comparator circuit when the CLK of the comparator is low, it will be in reset mode thus the valid signal falls. At the falling edge of the valid signal the corresponding latch that just finished fetching the output of the comparator and its "done" raised up will generate the "Sel_next" signal. "Sel_next" serves two purposes, first, it enables the next latch and second, it generates a positive pulse to the pull down NMOS of the tri-state inverter in the CLK generator circuit. As can be seen all the outputs of the tri-state inverters are wired-OR together and each latch subsequently triggers the comparator. Figure 3.9 shows an example of the timing signals between the comparator and two consecutive latches (m and

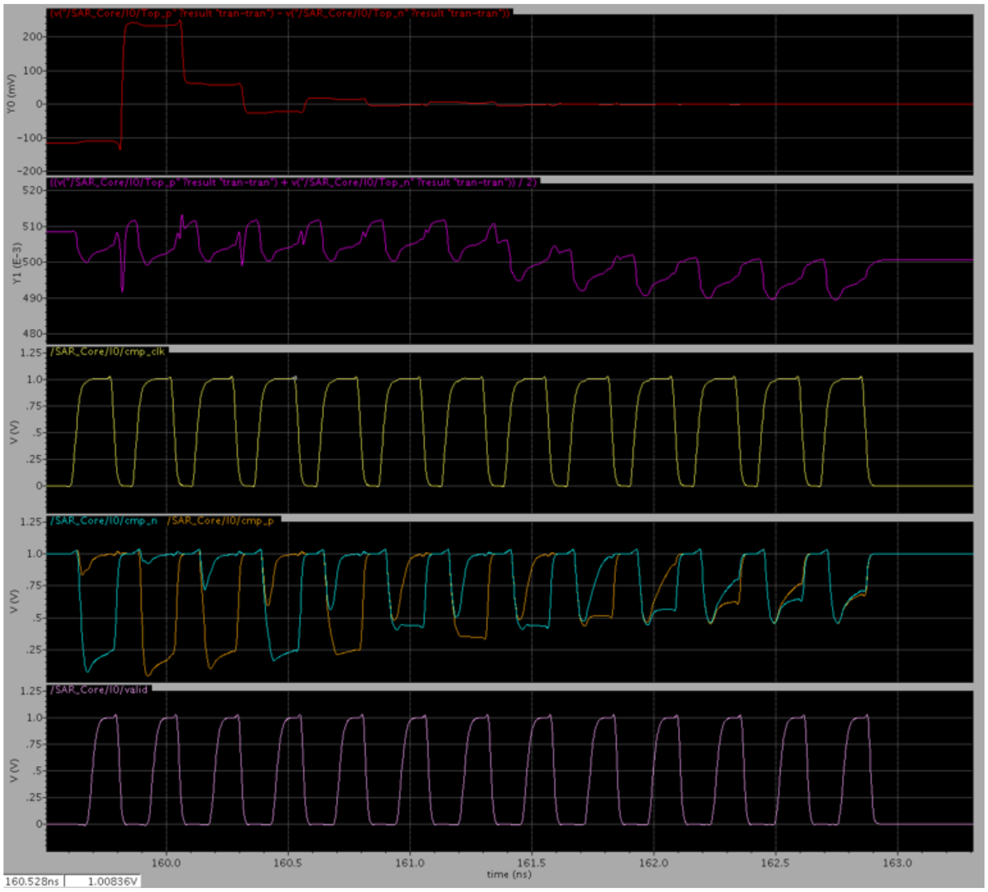
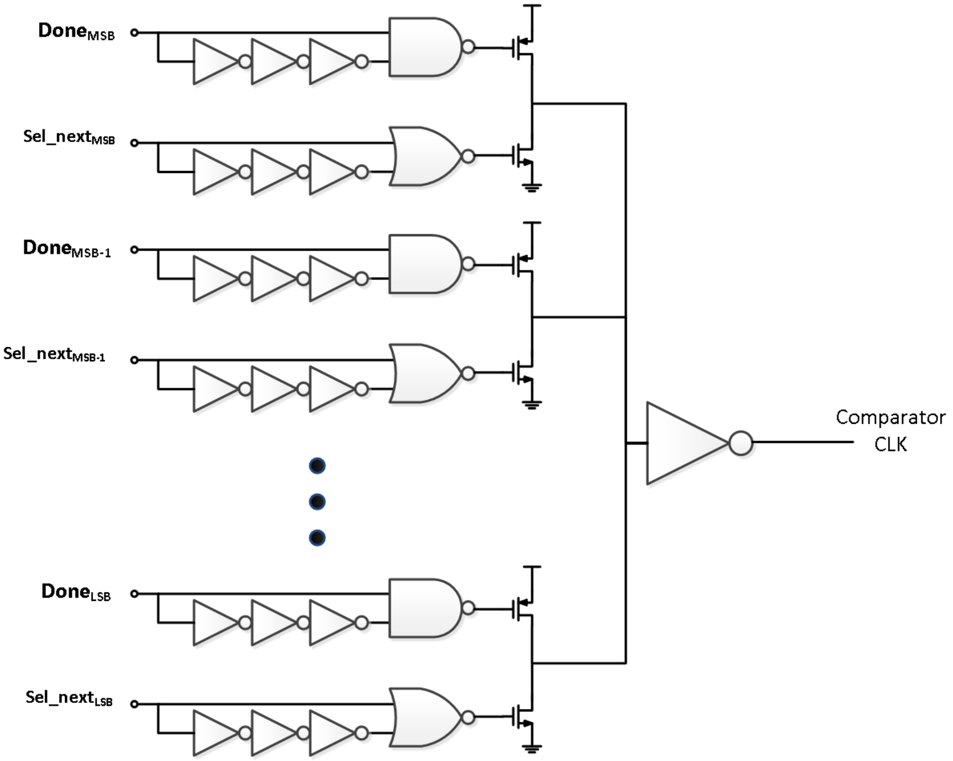


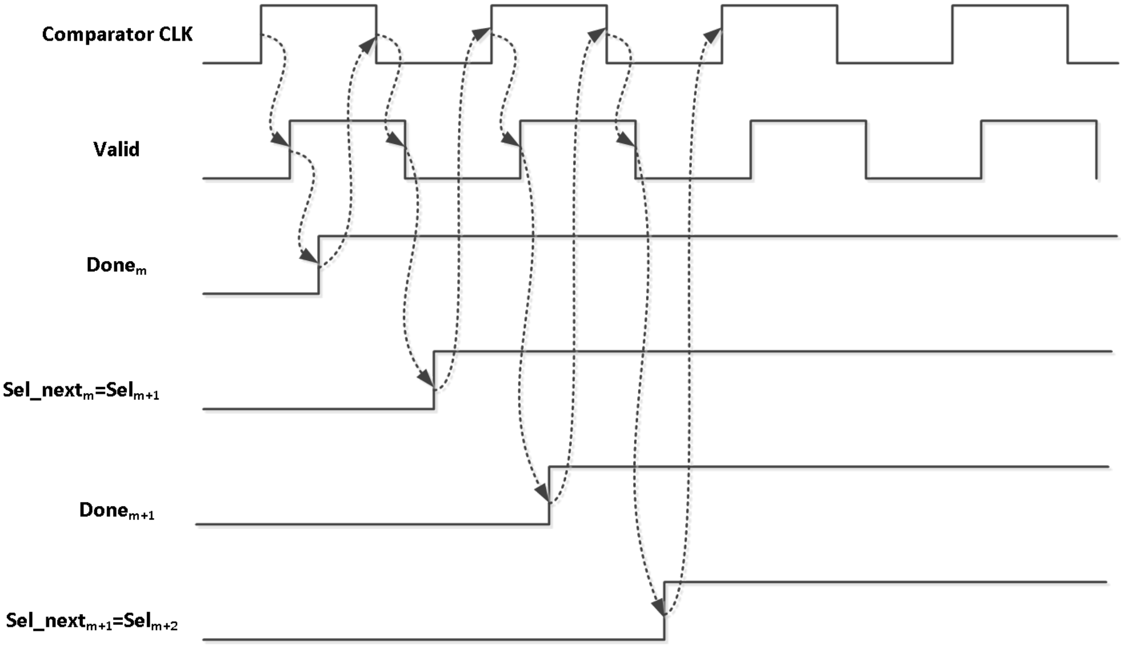
Figure 3.7: Simulated waveforms of the proposed comparator (from top to bottom respectively) differential input of the comparator, common mode of the input of the comparator, Asynchronous CLK, comparator outputs, and valid signal

m+1).



Done_{MSB}
$\text{Sel\_next}_{\text{MSB}}$
$\text{Done}_{\text{MSB-1}}$
$\text{Sel\_next}_{\text{MSB-1}}$
Done_{LSB}
$\text{Sel\_next}_{\text{LSB}}$
Comparator CLK

Figure 3.8: Asynchronous clock generator



Comparator CLK
Valid
Done_{m}
$\text{Sel\_next}_{m}=\text{Sel}_{m+1}$
Done_{m+1}
$\text{Sel\_next}_{m+1}=\text{Sel}_{m+2}$

Figure 3.9: Timing diagram of handshaking between comparator and latches

CHAPTER 4

Measurement Results

The SAR-ADC architecture explained in the previous chapter has been fabricated in CMOS tsmc65 nm GP. Figure 4.1 shows the layout of the fabricated chip and Figure 4.2 shows the chip micrograph of the ADC core on the fabricated die. The unit capacitor is 1fF to satisfy the $\frac{KT}{C}$ noise requirement of the sampling capacitor. To be able to show the effectiveness of the proposed calibration against mismatch of the capacitors, the unit capacitor of the LSBs are implemented with lateral field between "metal2" traces. However, the unit capacitor for MSBs consist of both "metal2" and "metal3".

Figure 4.3 shows the Power Spectral Density (PSD) and histogram of the output of the ADC under dynamic characterization test without calibration or dithering. In this figure, there are many harmonics in the PSD of the output which degrades the Signal to Noise and Distortion (SNDR) of the ADC to 58 dB, equivalent to the Effective Number of bits (ENOB) of 9.34 bits. Spurious Free Dynamic Rage (SFDR) of the ADC in this setup is about 62.6 which is the result of mismatch in the capacitors of the DAC. Figure 4.4 shows the dynamic performance of the ADC after just the dithering is turned on but not the calibration. In this case the dithering is Detrimental rather than helpful as it actually degrades the SNDR of the ADC to 55 dB. This occurs because the dithering signal used in the ADC is very small and dithering cannot result in enough averaging. Also, since the bitweights of the LSBs (the capacitors which inject the dither to the ADC) are unknown, subtraction of dithered signal is not complete. Thus the power

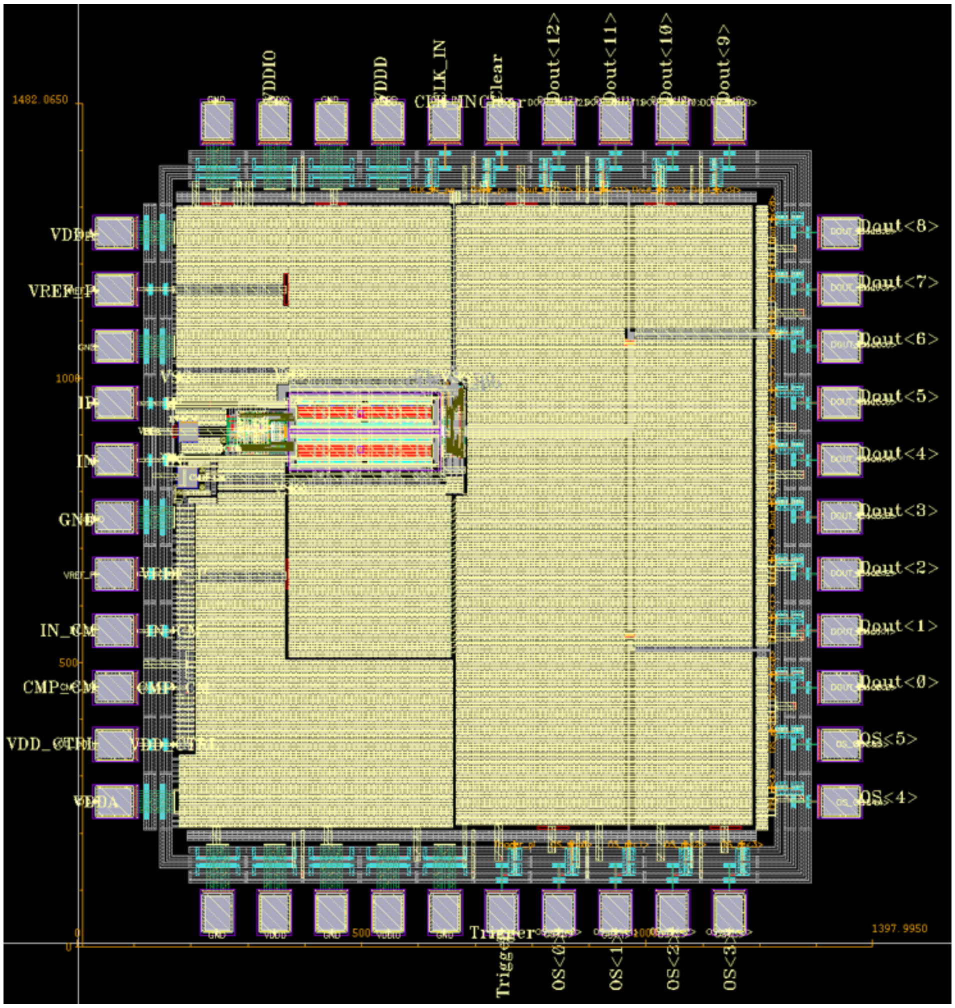


Figure 4.1: Layout of the impelemented SAR-ADC

of the remaining dither adds to the noise of the ADC. Larger dithering is not desirable since it will significantly reduce the dynamic range of the ADC even if it is completely subtracted from the output.

Figure 4.5 exhibits the effectiveness of the calibration. This figure shows PSD of the output of the ADC in presence of dithering and calibration together. It is noticeable that the SNDR of the ADC improves to 67.2 dB equivalent to ENOB of 10.9 bits. Also the SFDR improves to more than 81 dB.

Figure 4.6 shows the PSD of the output of the ADC when the calibration is done but the dithering is turned off after calibration. A small improvement in

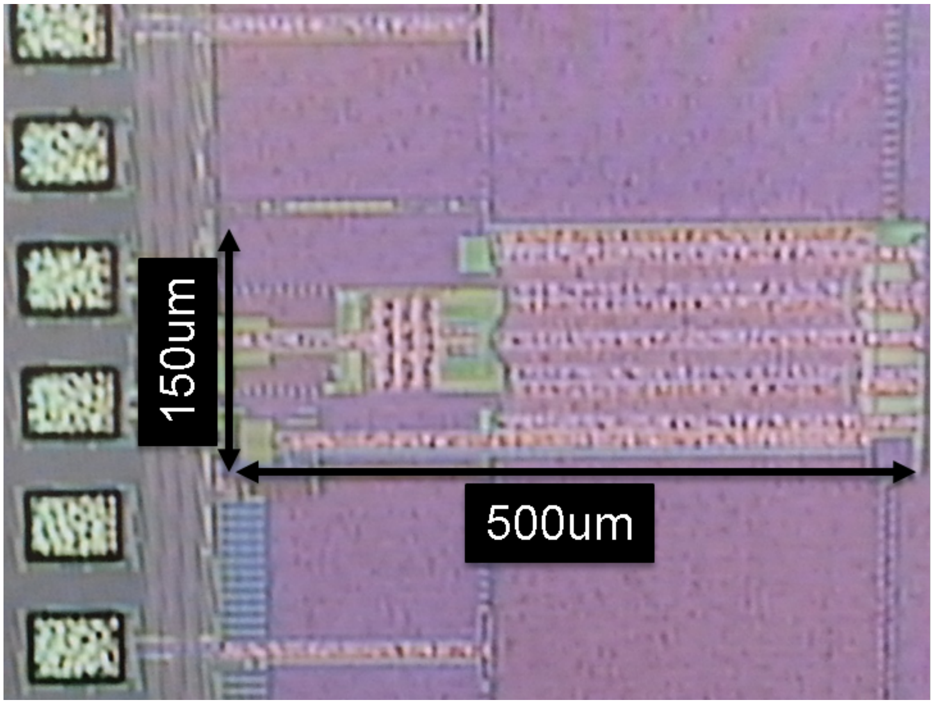


Figure 4.2: Chip micrograph of the SAR-ADC core

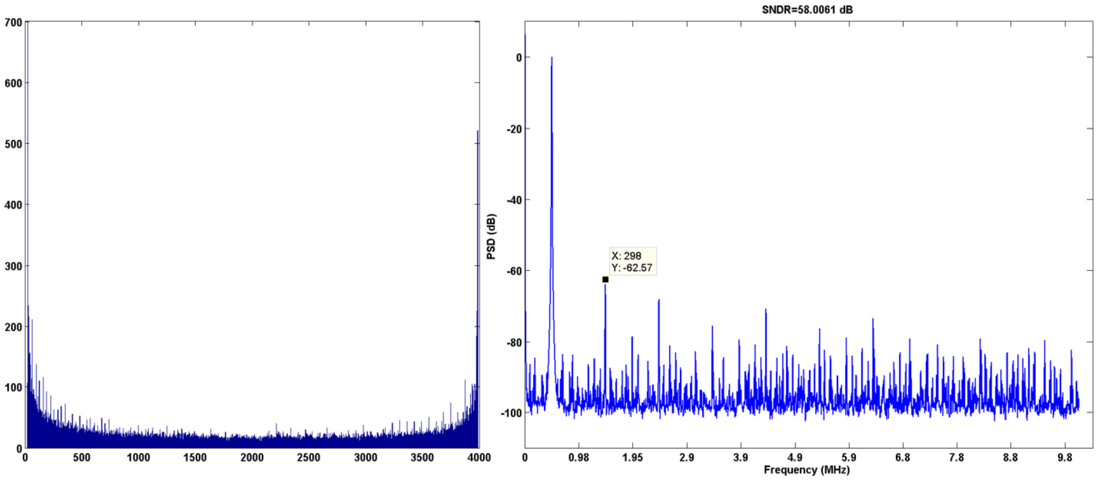


Figure 4.3: PSD of the output of the ADC under test with 20MHz sampling rate and 487KHz input signal with no calibration and no dithering

the SNDR compared to Figure 4.5 can be observed in the SNDR (from 67.2 to 67.5). However, some spurs in the PSD are now apparent. This is because of the

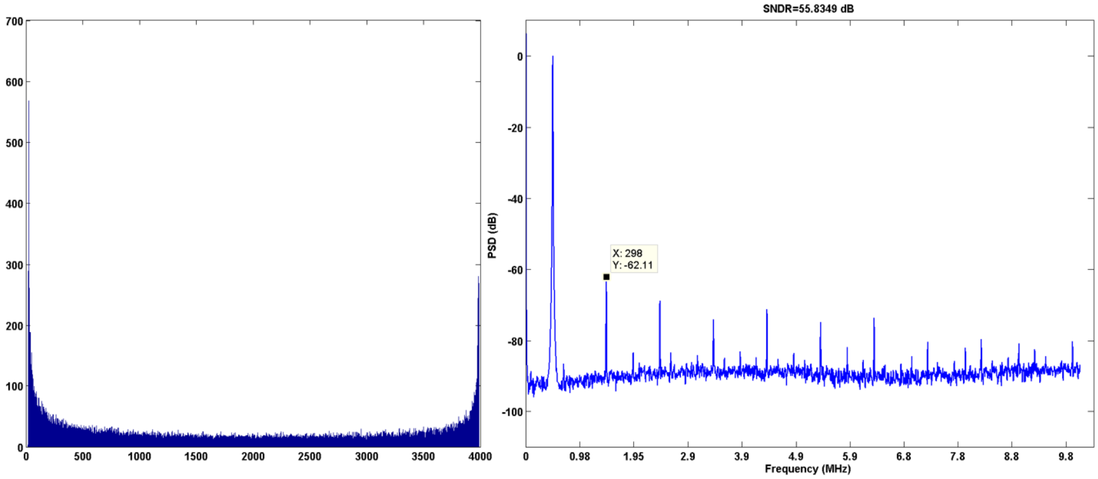


Figure 4.4: PSD of the output of the ADC under test with 20MHz sampling rate and 487KHz input signal with only dithering and no calibration

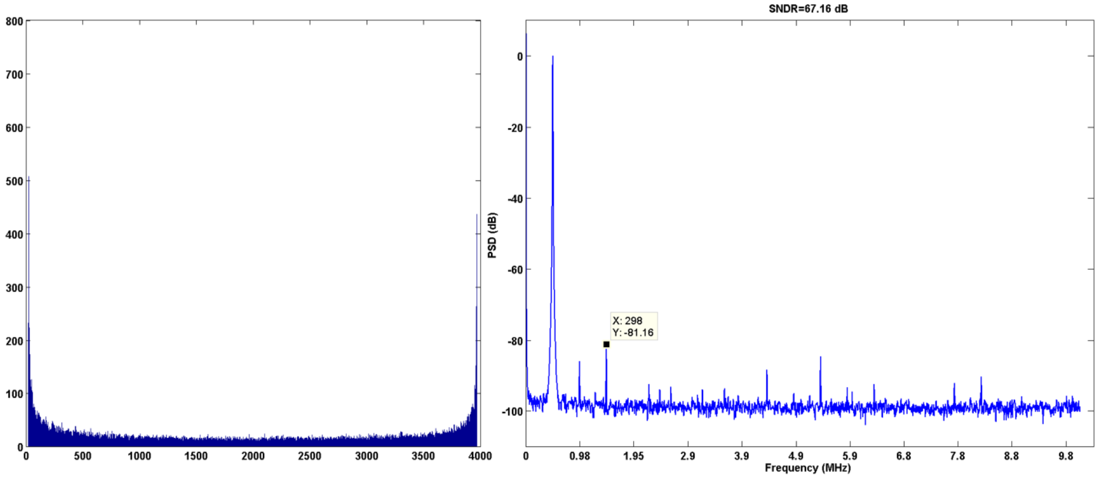


Figure 4.5: PSD of the output of the ADC under test with 20MHz sampling rate and 487KHz input signal with both calibration and dithering

fact that the redundant cycle of the ADC is at 64LSB thus the temporary errors similar to 2.5(c) that happen in the LSBs will not be corrected. Dithering can help these errors to be averaged, hence it improves the small spurs in the PSD.

Static performance of the ADC without calibration of dithering is shown in

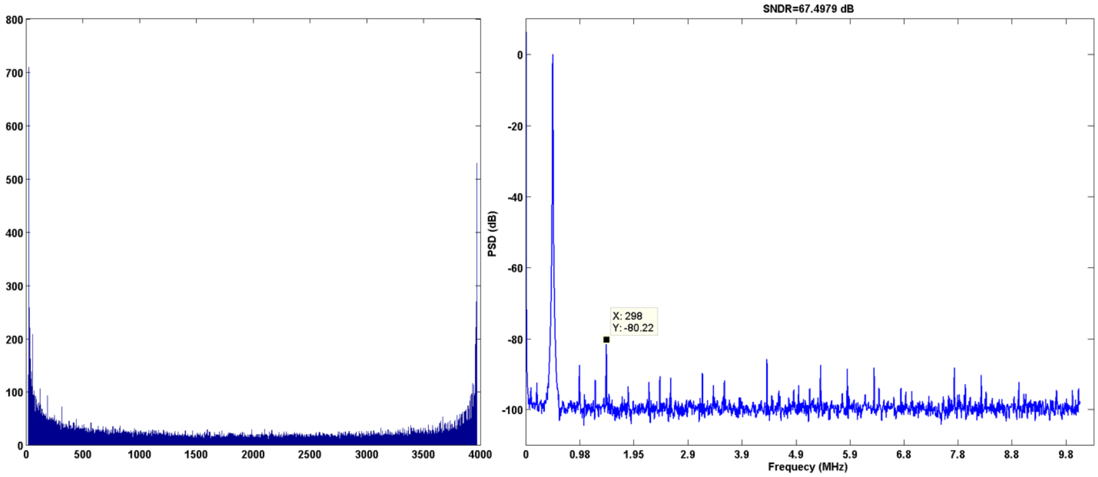


Figure 4.6: PSD of the output of the ADC under test with 20MHz sampling rate and 487KHz input signal with calibration, but turning off the dithering

Figure 4.7. As can be seen in this figure, DNL is as large as 3.1 LSB and maximum INL is 5.9 LSB. From this figure the error in the bitweight of the MSB is clearly visible.

Figure 4.8 shows the static performance of the ADC when calibration detects the correct bitweights but dithering has been turn off after calibration (similar scenario to the test condition of Figure 4.6). The calibration improves the DNL to 2.6 LSB but more importantly the INL improves to 2.1 LSB which is the reason that the SNDR and the SFDR of the ADC improved so much in the dynamic test as explained in section 2.3.

Figure 4.5 shows INL and DNL of the ADC when both calibration and dithering are on. It can be seen that the averaging feature of the dithering helps reduce the maximum of DNL to 1.4 LSB.

A valid concern about any algorithm that relies on PDF of the input signal is how it behaves in a practical scenarios. In order to test the performance of the proposed ADC in practical scenario, samples of a 802.11ac signal with 256-QAM modulation is used as the input of the ADC to measure the EVM of the output

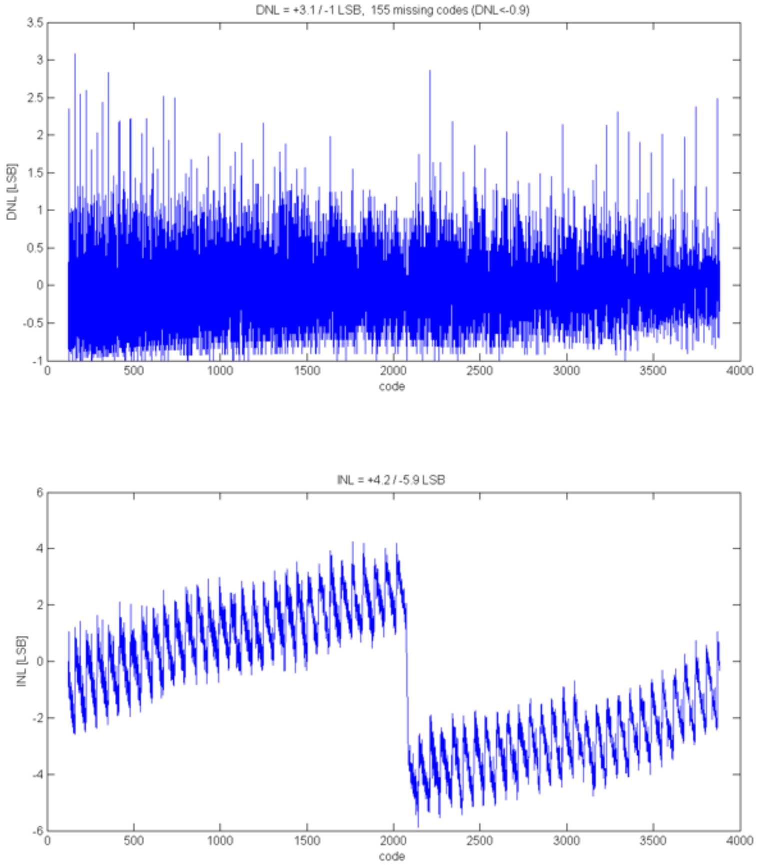


Figure 4.7: INL and DNL plots of the ADC when no calibration and no dithering is used

signal. In order to use the same ADC for both channels I&Q, an Arbitrary Waveform Generator feeds the ADCs input with samples of channel I and channel Q serially. Figure 4.10 shows the EVM of the output of the ADC without calibration (a) and with calibration (b). As can be seen in this figure, the EVM improves from -41.9 dB to -50.2 dB.

The pie chart of Figure 4.11 shows the percentage distribution of the power consumption in the ADC. As can be seen in the chart, the comparator consumes 30% of the energy of the ADC which is mainly noise limited. The latches and

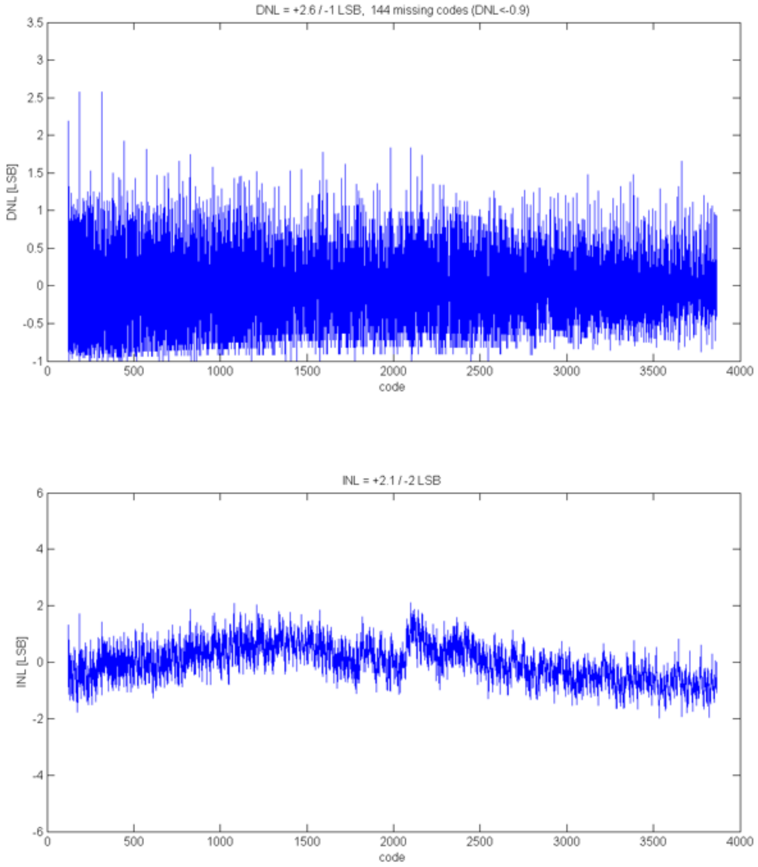


Figure 4.8: INL and DNL plots of the ADC when calibration is used but no dithering

digital circuitry of the DAC consume 60% of the total energy, which is digital switching power. Analog power of the DAC is the power of the switching of the capacitors of the DAC that draw necessary charge from reference voltage for the SAR algorithm to operate which is also noise limited. The power consumption of the calibration algorithm (except for the LFSR pseudo random generator) is not included in the pie chart of Figure 4.11 since it has been implemented in MATLAB. Our estimate for the calibration circuits by the count of the necessary registers is 150 μW which is about 16% of the total power consumption.

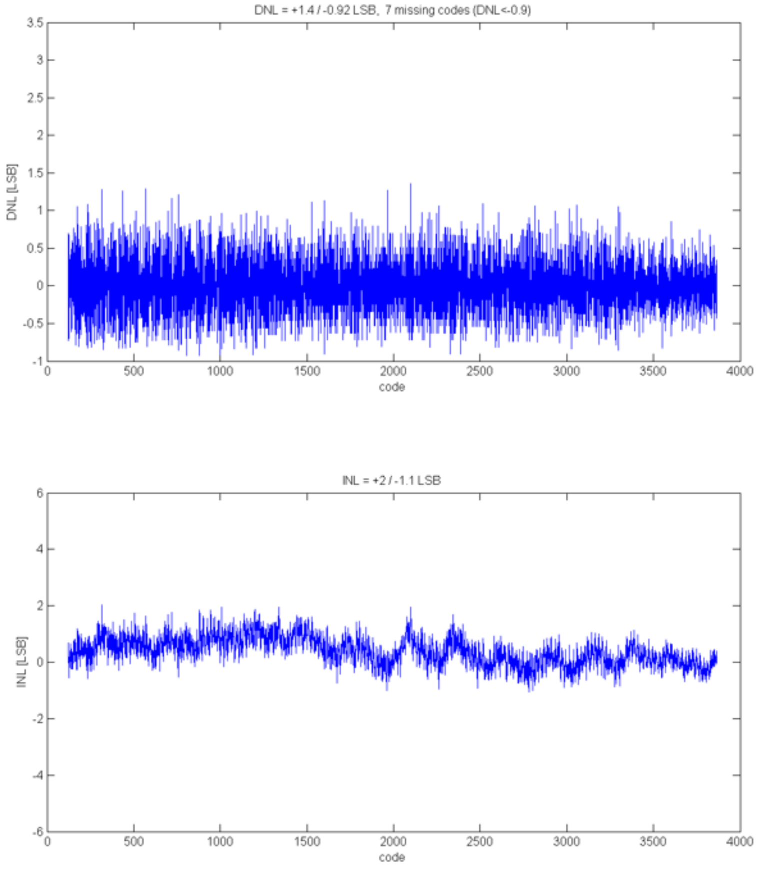


Figure 4.9: INL and DNL plots of the ADC when both calibration and dithering are used

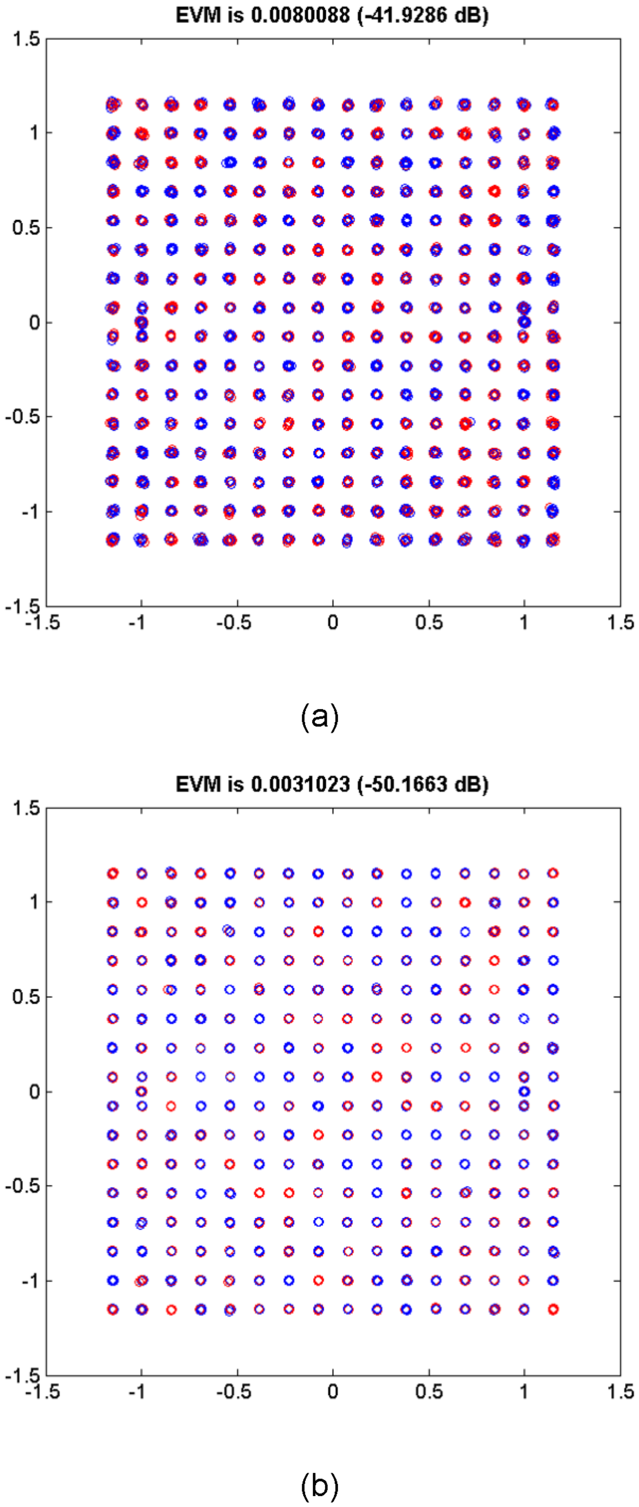
Finally, Figure 4.12 compares the proposed ADC with the prior art from ADC survey of [5].

Table 4.1 lists the summary of the specifications of the prototype SAR-ADC in this research.

Table 4.1: Summary of the specifications

Technology (nm)		**65**
Supply Voltage (V)		1
Resolution (bit)		12
Sampling Rate (MS/s)		20
ENOB (bit)	**w/ Calibration**	10.9
	wo/ Calibration	9.3
SNDR (dB)	**w/ Calibration**	67.2
	wo/ Calibration	58.0
SFDR (dB)	**w/ Calibration**	81.2
	wo/ Calibration	62.6
Power (mW)		920uW+150uW*
FOM (fJ/Conv.-step)		28
Area (mm^2)		0.075

*Estimated Power of the Calibration



(a)

(b)

Figure 4.10: EVM measurement of a 256-QAM at the output of the ADC (a) when there is no calibration applied to the ADC and (b) when the proposed calibration is used

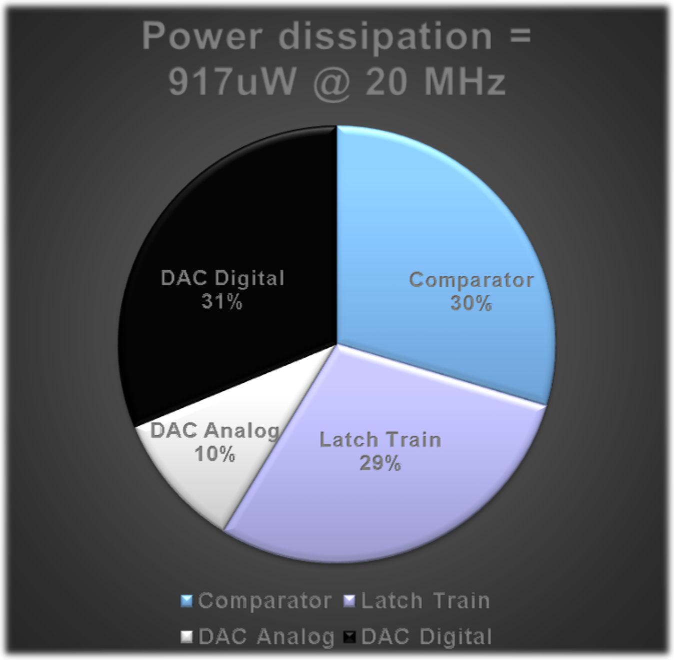


Figure 4.11: Pie diagram of the power consumption of different blocks of the ADC

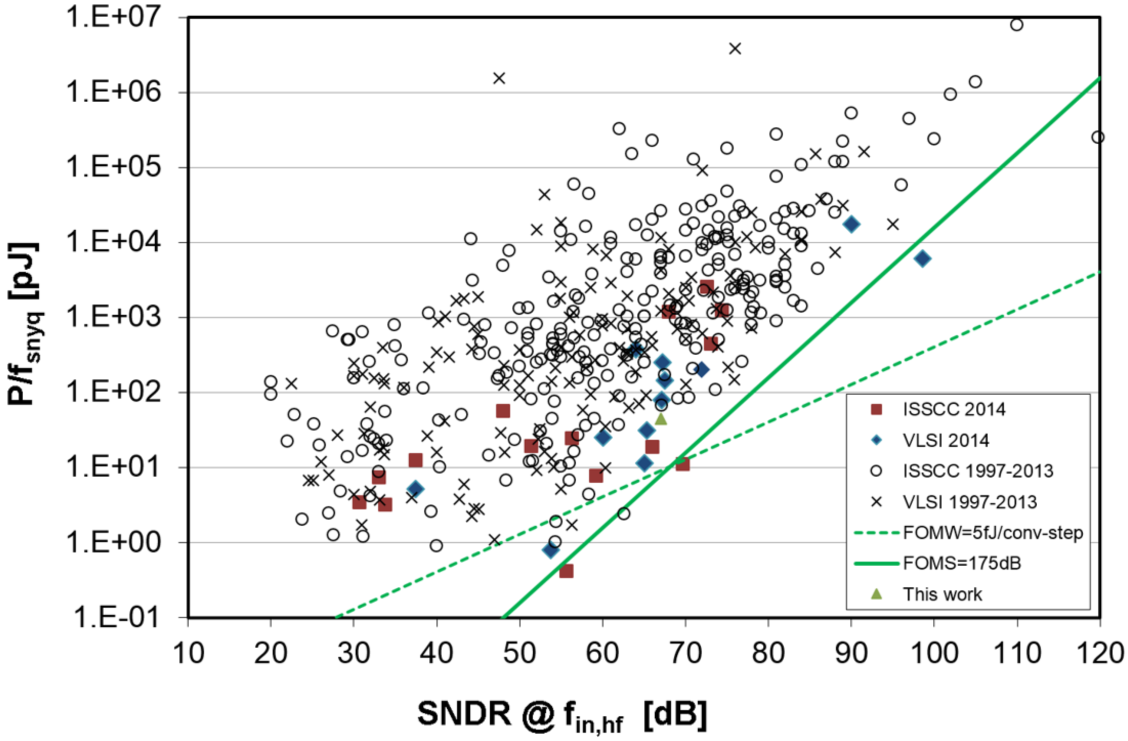


Figure 4.12: Prototype SAR-ADC of this work compared to the prior art, plot obtained from [5]

CHAPTER 5

Conclusion and Future Work

In this research a new calibration technique is introduced. The effectiveness of the proposed calibration has been successfully demonstrated by improving the performance of the fabricated SAR-ADC prototype. The prototype is a 12-bit SAR-ADC with 1 bit redundancy. The SNDR of the ADC before calibration is 58 dB which is equivalent to the ENOB of 9.3 bits and SFDR of the ADC is 62.6 dB. The proposed calibration improves the SNDR of the ADC to 67.2 dB (equivalent to 10.9 bits) and enhances the SFDR to over 81 dB. The effectiveness of the ADC is also tested on a practical situation with a 256-QAM modulated signal samples and it has been shown that the calibration improves the EVM of output from 42 dB to 50 dB.

The implementation of the calibration has been done in MATLAB. However, it has been shown that the implementation of the logic can be simplified by wise choices of the dither signal and NBW.

In addition, a new architecture for the comparator and SAR logic has been introduced. In this architecture redundant attempts of generating high swing voltages, which result in waste of power, have been avoided. The output of the comparator has been kept with low swing signals and the regeneration task is done by the SAR latches.

For future work, we believe the proposed calibration has a greater potential to be implemented on other ADC architectures. Pipeline ADC is a great candidate to take advantage of this calibration. Since this calibration is a background calibration, it can track any drift in the gain of the gain-stages of pipeline ADC which is useful in architectures with open-loop amplifiers or amplifiers with low gain that are sensitive to PVT.

It is also beneficial to investigate use of the proposed estimation technique to detect the nonlinearity of systems other that ADC. The only necessary conditions needed to use the proposed estimation on any system is the ability to introduce an offset to the input of the system and with use of a handful of comparators, generate a histogram of the output signal.

References

[1] J. Chu, L. Brooks, and H.-S. Lee, “A zero-crossing based 12b 100ms/s pipelined adc with decision boundary gap estimation calibration,” in *VLSI Circuits (VLSIC), 2010 IEEE Symposium on*, pp. 237–238, June 2010.

[2] R. Xu, B. Liu, and J. Yuan, “Digitally calibrated 768-ks/s 10-b minimum-size sar adc array with dithering,” *Solid-State Circuits, IEEE Journal of*, vol. 47, no. 9, pp. 2129–2140, Sept 2012.

[3] W. Liu, P. Huang, and Y. Chiu, “A 12-bit 50-ms/s 3.3-mw sar adc with background digital calibration,” in *Custom Integrated Circuits Conference (CICC), 2012 IEEE*, pp. 1–4, Sept 2012.

[4] J. Kim, B. Leibowitz, J. Ren, and C. Madden, “Simulation and analysis of random decision errors in clocked comparators,” *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 56, no. 8, pp. 1844–1857, Aug 2009.

[5] B. Murmann. Adc performance survey 1997-2014. [Online]. Available: http://web.stanford.edu/ murmann/adcsurvey.html

[6] R. Aparicio and A. Hajimiri, “Capacity limits and matching properties of integrated capacitors,” *Solid-State Circuits, IEEE Journal of*, vol. 37, no. 3, pp. 384–393, Mar 2002.

[7] W. A. Kester, *Data conversion handbook*. Newnes, 2005.

[8] B. Ginsburg and A. Chandrakasan, “An energy-efficient charge recycling approach for a sar converter with capacitive dac,” in *Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on*, pp. 184–187 Vol. 1, May 2005.

[9] C.-C. Liu, S.-J. Chang, G.-Y. Huang, Y.-Z. Lin, C.-M. Huang, C.-H. Huang, L. Bu, and C.-C. Tsai, “A 10b 100ms/s 1.13mw sar adc with binary-scaled error compensation,” in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2010 IEEE International*, pp. 386–387, Feb 2010.

[10] C.-C. Liu, S.-J. Chang, G.-Y. Huang, and Y.-Z. Lin, “A 10-bit 50-ms/s sar adc with a monotonic capacitor switching procedure,” *Solid-State Circuits, IEEE Journal of*, vol. 45, no. 4, pp. 731–740, April 2010.

[11] P. Nuzzo, F. De Bernardinis, P. Terreni, and G. Van der Plas, “Noise analysis of regenerative comparators for reconfigurable adc architectures,” *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 55, no. 6, pp. 1441–1454, July 2008.

[12] A. Abidi and H. Xu, “Understanding the regenerative comparator circuit,” in *Custom Integrated Circuits Conference (CICC), 2014 IEEE Proceedings of the*, pp. 1–8, Sept 2014.

[13] M. Jeeradit, J. Kim, B. Leibowitz, P. Nikaeen, V. Wang, B. Garlepp, and C. Werner, “Characterizing sampling aperture of clocked comparators,” in *VLSI Circuits, 2008 IEEE Symposium on*, pp. 68–69, June 2008.