PERFORMANCE ENHANCEMENT OF PIPELINE ADCs

A Dissertation
Submitted to the Graduate Faculty
of the
North Dakota State University
of Agriculture and Applied Science

By

Ting Li

In Partial Fulfillment
for the Degree of
DOCTOR OF PHILOSOPHY

Major Department:
Electrical and Computer Engineering

August 2014

Fargo, North Dakota

North Dakota State University
Graduate School

Title

Performance Enhancement of Pipeline ADCs

By

Ting Li

The Supervisory Committee certifies that this ***disquisition*** complies with North Dakota State University's regulations and meets the accepted standards for the degree of

DOCTOR OF PHILOSOPHY

SUPERVISORY COMMITTEE:

Dr. Chao You
Chair

Dr. Debasis Dawn

Dr. Sudarshan Srinivasan

Dr. Jun Kong

Approved:

08/20/2014
Date

Scott Smith
Department Chair

ABSTRACT

The pipeline ADC is mainstream architecture in wireless communication and digital consumer products because of its speed, resolution, dynamic performance, and power consumption. However, there are three areas of concern with the pipeline analog-to-digital converter (ADC): power consumption, accuracy, and convergence speed of the digital calibration. The traditional pipeline ADC includes a dedicated front-end sample-and-hold amplifier (SHA), which consumes a significant amount of power. This research presents a novel configuration of the front-end stage with a sample-and-hold function for a SHA-less architecture. In addition, the multi-bit front-end has multiple benefits. Interestingly, if one additional bit is resolved in the front-end stage, then the comparator offset correction ability of this stage is reduced by half. To address this problem, this research presents a novel domain-extended digital error correction algorithm to increase the comparator offset correction ability. In order to improve accuracy, a combination of techniques are used: communicated feedback-capacitor switching (CFCS), gain boost amplifiers, and low noise dynamic comparators. Here, the ADC uses the above mentioned techniques and is fabricated with AMIS 0.5 µm CMOS. The ADC, with an active area of 4.5 mm^2, consumes 264 mW when a 32 MHz input is at 75-MS/s sample rate.

The third area of concern is convergence time, which determines the quality of the digital calibration. The high resolution ADC can be achieved without calibration. Therefore, in order for a digital calibration to be useful, it should minimize the analog circuits and have a reasonable convergence time. The reduced accuracy due to minimized analog circuits can be complemented by the digital calibration. Therefore, the convergence time determines the quality of the digital calibration. In this research a new domain-extended dither-based algorithm increases the

convergence speed. Moreover, the novel variable-amplitude domain-extended dither-based algorithm further increases the convergence speed. Matlab simulations illustrate these improvements.

ACKNOWLEDGMENTS

I would like to thank my advisor, Dr. You, one of the most important people in my life. He gave me the opportunity to study and work at North Dakota State University. He has been a constant source of guidance, support, and encouragement both in my professional and personal lives. He also has trained me with valuable skills for research, writing, and conference presentation. Words are inadequate to express my heartfelt thanks.

I would also like to express my great gratitude to Dr. Dawn. I especially appreciate the in-depth explanation of fundamental principles about the CMOS analog design and the dedicated help during the chip fabrication. In addition, I also would like to thank the other members of my PhD committee, Dr. Srinivasan and Dr. Kong, for their assistance, guidance, and time contributions on my dissertation.

I am also thankful to MOSIS for chip fabrication. I would also like to thank Dr. Braaten and Jeffrey Erickson, for their very friendly lab and equipment help. In addition, I also would like to thank Kristina Caton, a consultant in NDSU Center for Writers, for her professional English review during paper and dissertation writing. I also thank my friends in both America and China for providing the support, encouragement, friendship that I needed.

DEDICATION

I dedicate this dissertation to my family:

to my husband, Songbo, for his constant love, understanding, and support;

to my son, Allen, for the fun time playing with him and his encouragement for me to work harder;

to my loving parents, Shuhe Li and Xiufang Liu, who have sacrificed their lives a lot for me and provided unconditional love and support;

to my grandpa for his encouragement and care.

I love you all dearly.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1. INTRODUCTION

1.1. Background

The rapid development of digital techniques promote the trend of processing the signal in the digital domain rather than in the analog domain. This is because processing the signal in the digital domain improves accuracy and speed, reduces power consumption, and introduces flexibility [1]. However, the real world signal is analog. This is because the real world temperature, pressure, gravity, etc. are all analog. The analog signal is continuous in both time and amplitude. In contrast, the digital signal is discrete in time and only has two amplitude values: 0 and 1. The ADC is a device that converts the continuous analog signal to the digital signal. Therefore, the ADC functions as an interface that connects the analog signal to the digital signal processing system.

Many ADC structures have been developed in order to best suit special applications. Each structure is designed for a particular combination of speed, accuracy, chip-area, and power consumption. The pipeline ADC has outstanding performance when speed, resolution, dynamic performance, and power consumption are comprehensively considered. This results in the pipeline ADC being widely used in many areas, such as communication and digital consumer products.

Traditionally, a pipeline ADC includes a front-end SHA [2]. The front-end SHA consumes a significant amount of power and contributes substantial noise and distortion to the entire ADC. Therefore, an elimination of the front-end SHA can save on power consumption and diminish noise contribution. Then, in order to reduce power consumption and noise distribution result from the front-end SHA, a SHA-less architecture ADC using the matching technique is developed [3] [4]. However, high quality matching is hard to achieve because the multiplying digital-to-analog

converter (MDAC) signal path and the sub-ADC signal path have significant differences in both design and layout.

In addition to the front-end stage consideration mentioned above, another consideration for the front-end stage of the pipeline ADC is that a multi-bit front-end is normally adopted by an ADC. This is because the multi-bit front-end is able to relax the capacitor matching requirements while yielding good power efficiencies in pipeline ADCs [5] [6]. However, the comparator offset toleration ability is reduced by half [6] if one more bit is resolved in the front-end stage through the use of the traditional digital error correction technique.

Furthermore, a high resolution pipeline ADC without calibration can be achieved [2] [4]. Therefore, in order for a digital calibration to be useful, it should minimize the analog circuits as well as have the reasonable convergence time. It is well known that the digital calibration complements the reduced accuracy results from the minimized analog circuits. Hence, convergence time is the major limitation of a digital calibration.

1.2. Objectives

This research is targeted to design a pipeline ADC with a good trade-off between power and performance. In other words, the purpose is to design a high accuracy ADC with a minimized power consumption. Therefore, this newly developed ADC aims to reduce power consumption by using a new front-end stage with a sample-and-hold function. Also, this ADC aims to improve accuracy through a novel domain-extended digital error correction algorithm. In addition, this research is targeted to develop dither-based digital calibration algorithms with faster convergence speeds.

1.3. Organization

Chapter 2 presents the overall pipeline ADC architecture and analysis. In this chapter some key error sources are analyzed and the performance parameters are introduced. Next, Chapter 3 analyzes the techniques that are used for high accuracy and low power consumption ADC design. These techniques are: SHA-less ADC technique, domain-extended digital error correction technique, and CFCS technique. Afterwards, Chapter 4 provides the detailed circuit configuration and simulations. Then, Chapter 5 illustrates the test setup, test method, experimental results, and results analysis. After that, Chapter 6 presents the two newly developed dither-based algorithms used to reduce the convergence time for digital calibrations. Finally, Chapter 7 concludes the research.

CHAPTER 2. PIPELINE ADC ARCHITECTURE AND ANALYSIS

2.1. Pipeline ADC architecture

An analog-to-digital converter (ADC) is a device that converts a continuous analog input signal to a discrete time digital representation. It is widely used in many areas such as music recording, healthcare instrumentation, and communication and radar systems. This is due to its outstanding performance when speed, resolution, dynamic performance, and power consumption are comprehensively considered.

Figure 2-1 shows the block diagram of a pipeline ADC. The pipeline ADC consists of sample-and-hold, time alignment, digital error correction, and clock signal generation circuits. In addition, the pipeline ADC also has several pipeline conversion stages. The individual pipeline conversion stages may or may not have the same resolutions. Each pipeline conversion stage, except the last stage, includes a sample-and-hold circuit, a sub-ADC, a sub-DAC, and an operational-amplifier. The last stage is a simple flash-ADC since the output of the last stage does not need further resolution.

The process through which the ADC resolves the analog input into the digital output is well known. The clock signal generator generates two opposite clock signals, Φ_1 and Φ_2. When Φ_1 is high, the ADC works in a sample phase. In contrast, when Φ_2 is high, the ADC works in an amplification phase. The input is first sampled and then held by a sample-and-hold circuit. Then, a sub-ADC quantize the input held by a sample-and-hold circuit to K1 bits. These K1 bits output are then fed to a K1-bit sub-DAC. Next, the analog output of the sub-DAC is subtracted from the input. The residue is then increased by a factor of 2^{K1-1} and fed to the next stage. The total output bits, T, can be calculated after the time alignment and digital error correction.

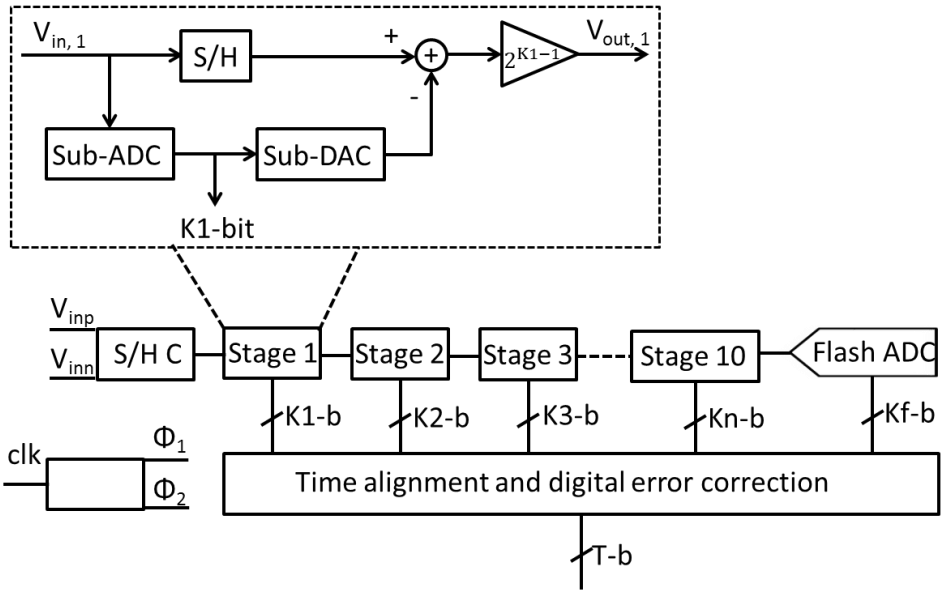


Figure 2-1. Pipeline ADC block diagram

2.2. Error sources of a pipeline ADC

2.2.1. Charge injection and clock feedthrough

A simple sample-and-hold circuit can be realized through an MOS transistor and a capacitor, as shown in Figure 2-2. This sample-and-hold circuit is implemented in the following way. The input voltage with the MOS transistor on is sampled on the capacitor. In contrast, with the MOS switch off, the input voltage is stored on hold on the capacitor.

However, this simple sample-and-hold circuit rarely meets real application requirements because of the charge injection and the clock feedthrough effects [7]. The charge injection effect is caused by the injection of the charges stored on the transistor's channel into the capacitor during the short interval transitioning from on to off. The clock feedthrough effect is caused by the coupling due to gate-drain overlap capacitance during the short interval transitioning from on to off.

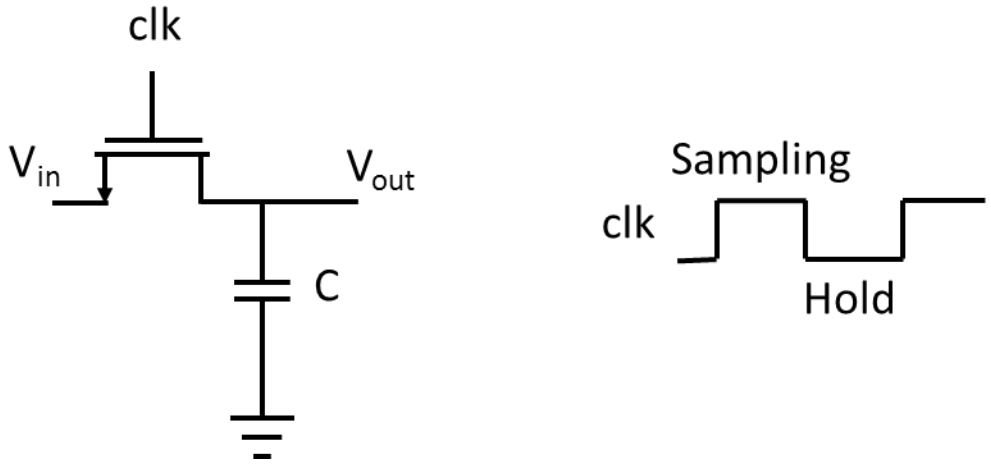


Figure 2-2. Simple sample-and-hold circuit

2.2.1.1. Charge injection effect

During the short interval transitioning from on to off, the channel charges of the transistor are injected into the source and drain. This modeled charge injection effect is shown in Figure 2-3. The channel charges of the transistor, Q, are represented by the following equation

$$Q = WLC_{ox}(V_{clk} - V_{in} - V_{th}), \tag{2.1}$$

where W represents the width of the MOS transistor, L represents the length of the MOS transistor, C_{ox} represents the oxide capacitance per unit area, V_{clk} represents the clock signal voltage, V_{in} represents the input voltage, and V_{th} represents the threshold voltage of the MOS transistor.

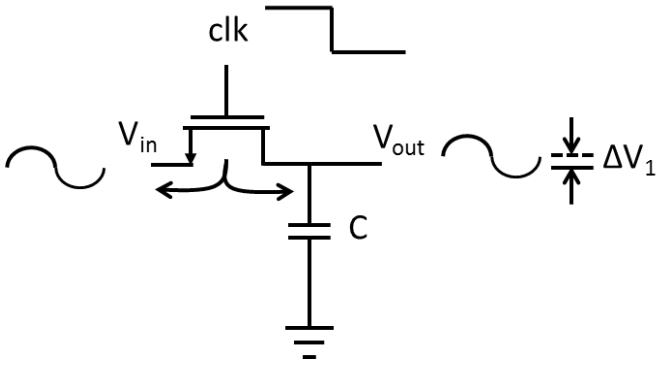


Figure 2-3. Charge injection effect

When the transistor is turned off, the channel charges enter the drain and source. The source voltage remains unchanged since the source is connected to the voltage source. However, the channel charges entering the drain introduce an error voltage to the capacitor. The error voltage can be represented as

$$\Delta V_1 = KQ/C = KWLC_{ox}(V_{clk} - V_{in} - V_{th})/C, \tag{2.2}$$

where K is a coefficient, which depends on the voltage and resistance between the drain and source and C is the capacitance of the sample-and-hold capacitor.

The signal dependency of the voltages V_{th}, V_{clk}, V_{in}, and the coefficient K result in the error voltage, ΔV_1, in response to input voltages. Therefore, the charge injection effect introduces nonlinear errors to the sample-and-hold circuit.

2.2.1.2. Clock feedthrough effect

Besides the charge injection effect, the clock feedthrough effect also changes the drain voltage. When the clock swings from high voltage to low voltage, the transistor turns off. During the short interval transitioning from on to off, the coupling through gate-drain capacitance influences the drain voltage. This clock feedthrough effect introduces an error voltage to the sample output as shown in Figure 2-4. This error voltage can be represented as

$$\Delta V_2 = -(V_{dd} - V_{ss})C_{GD}/(C_{GD} + C), \tag{2.3}$$

where V_{dd} represents the high voltage of clock signal, V_{ss} represents the low voltage of clock signal, and C_{GD} represents the gate-drain overlap parasitic capacitance.

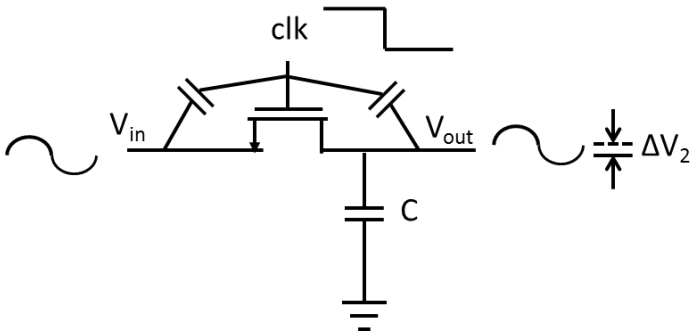


Figure 2-4. Clock feedthrough effect

2.2.2. Comparator offsets and gain errors

The performance of the switched-capacitor pipeline ADC is significantly limited by the linear errors due to comparator offsets, capacitor mismatches, and finite operational-amplifier gain. A pipeline ADC is composed of several stages. Each stage is composed of a low resolution sub-ADC, a low resolution sub-DAC, a sample-and-hold circuit, and a residue gain stage. The input of a stage is constrained by the output of the previous stage. The input range must be identical with the output range in order to prevent an ADC from code loss. Two non-idealities can cause code loss. The first one is comparator offset, and the other one is gain error, which is introduced by capacitor mismatches and finite operational-amplifier gain.

A sub-ADC includes one or more comparators. The comparator compares input voltage with a threshold voltage. Figure 2-5 shows the transfer characteristics of a 2-bit converter. The ideal transfer characteristics are shown with solid lines, while the transfer characteristics with comparator offsets are depicted with dashed lines. A comparator offset is caused by the component offset and the dynamic offset. The dynamic offset originates from some harmful effects such as clock coupling. Figure 2-5 illustrates how the comparator offset produces an output of the current stage exceeding the output range. However, this out-of-range output exceeds the input range of the subsequent stage, which leads to a code loss.

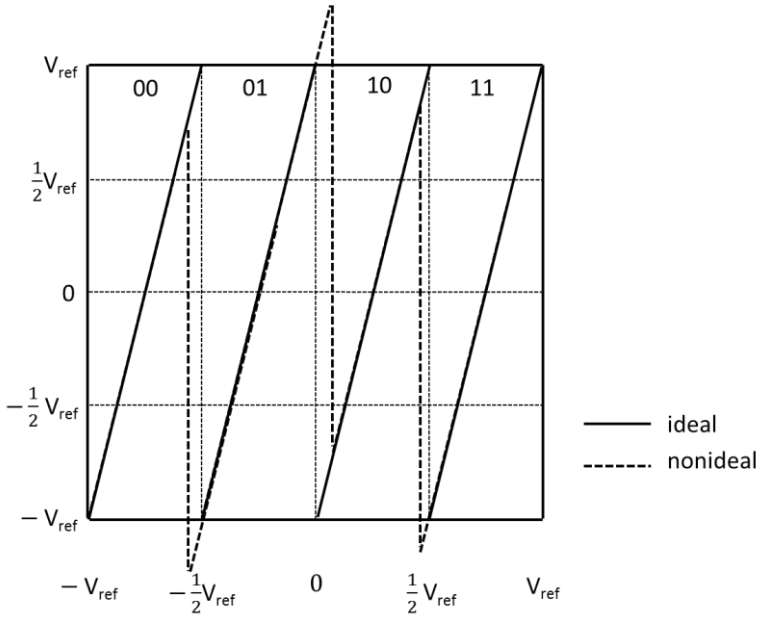


Figure 2-5. Transfer characteristics with comparator offsets

Besides the comparator offset, the gain error can also lead to a code loss. A residue gain stage includes switched-capacitors and an operational-amplifier. Figure 2-6 shows the transfer characteristics. The ideal transfer characteristics are shown with solid lines, while the transfer characteristics with gain errors are shown with dashed lines. The gain error is caused by switched-capacitor mismatches and finite open-loop operational-amplifier gain. As shown in Figure 2-6, because of this gain error, the output of the current stage exceeds the output range. Therefore, this out-of-range output exceeds the input range of the subsequent stage, which leads to a code loss. If a code loss occurs, it would significantly degrade the entire ADC performance.

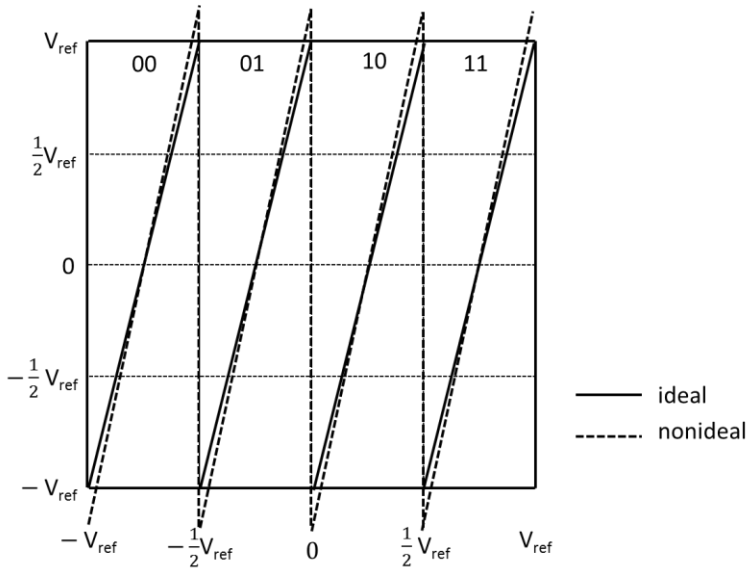


Figure 2-6. Transfer characteristic with gain errors

2.2.2.1. Comparator offset correction

In order to decrease the possibility of a code loss result from gain errors or comparator offsets, the inter stage gain should be decreased from four to two. However, this method cannot avoid the occurrence of miscode due to the comparator offsets. The digital error correction technique based on the residue bits can be used to correct the miscode due to the comparator offsets. As seen in Figure 2-7, for this method, the threshold voltages are changed to $-V_{ref}/4$ and $V_{ref}/4$ as shown in the dotted lines. The coded range is from $-V_{ref}$ to V_{ref}. Thus, the usable bits are decreased from 2 to 1.5. The residue bit, 0.5, is used to correct the comparator offsets. The transfer characteristics of a real ADC with comparator offsets are shown using dashed lines. In this case, the maximum comparator offset is $V_{ref}/4$ and the corresponding maximum output is equal to V_{ref}. As long as the absolute value of the comparator offsets are within the range of $0 \sim V_{ref}/4$， the comparator offsets can be corrected [8].

The implementation of this method adds the lowest bit of current stage to the highest bit of the subsequent stage. This realization of dislocation addition requires several addition blocks in the digital domain to correct the errors in the analog domain.

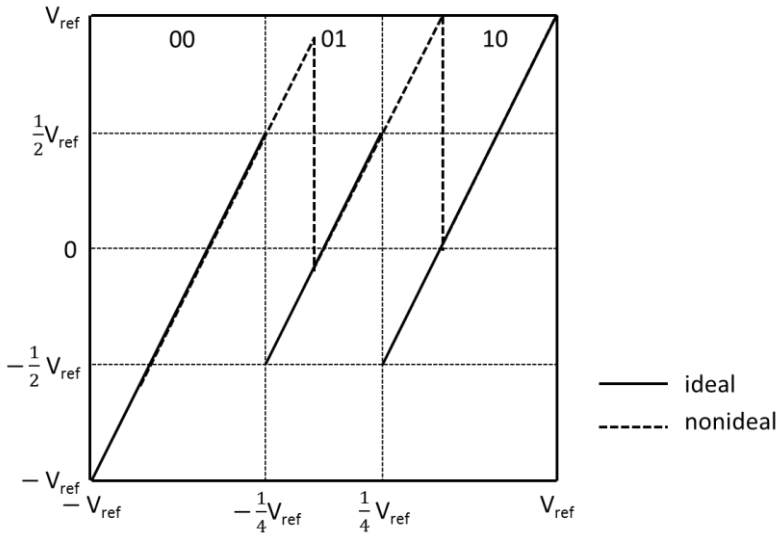


Figure 2-7. Transfer characteristics of the 1.5-bit/stage ADC

2.2.2.2. Gain error source: capacitor mismatches

In switched-capacitor pipeline ADCs, another error source is capacitor mismatches. The capacitor mismatches between feedback capacitor and non-feedback capacitor(s) introduce gain errors. In order to illustrate how the gain errors are affected by capacitor mismatches, the scheme of a single stage ADC on the amplification phase is shown in Figure 2-8.

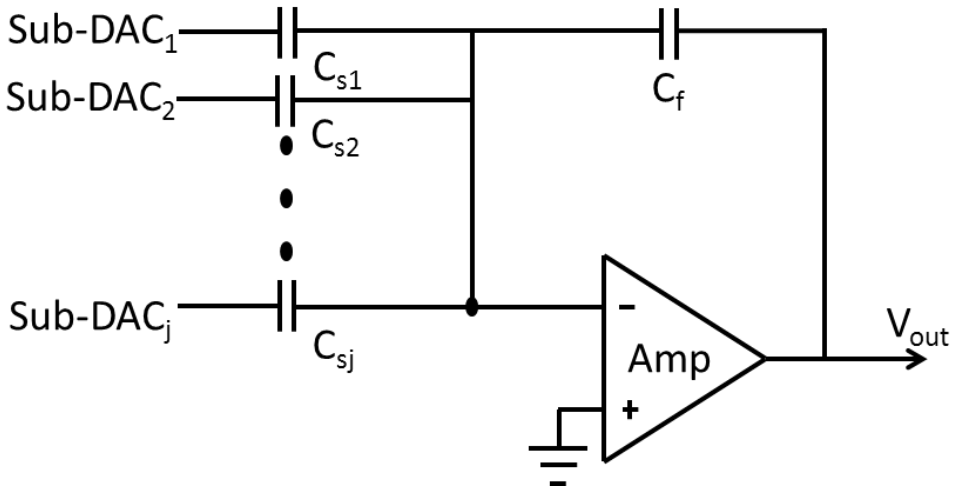


Figure 2-8. Single stage ADC works on the amplification phase

In this configuration, the feedback capacitor C_f has the nominal value C. Therefore, with the presence of capacitor mismatches, the gain is

$$G = \frac{C_{s1} + C_{s2} + ...C_{sj}}{C_f} = j + \Delta C_{s1} + \Delta C_{s2} + ...\Delta C_{sj} \,, \tag{2.4}$$

where j is the number of non-feedback capacitors and ΔC_{s1}, ΔC_{s2} … ΔC_{sj} represent the relative mismatches of C_{s1}, C_{s2} … C_{sj} with respect to C.

In order to formulate the transfer function with capacitor mismatches, the transfer function without capacitor mismatches must be first calculated using the following formula:

$$V_{out_i} = j \cdot V_{in_i} - V_{sub-DAC_i} \,, \tag{2.5}$$

where V_{outi} represents the output of the i^{th} stage, j is the total number of non-feedback capacitors, V_{ini} represents the input of the i^{th} stage, and $V_{sub\text{-}DACi}$ represents the sub-DAC output of the i^{th} stage.

In the above equation, the $V_{sub\text{-}DACi}$ is not affected by the capacitor mismatches. The $V_{sub\text{-}DACi}$ can be represented as

$$V_{sub-DAC_i} = V_{ref} \cdot D_i \,, \tag{2.6}$$

where $D_i \in \{0, \pm 1, \pm 2, \ldots \pm j\}$.

Thus, the transfer function with capacitor mismatches can now be calculated using the following formula:

$$V'_{out_i} = \left(j + \Delta C_{s1} + \Delta C_{s2} + \ldots \Delta C_{s2}\right) \cdot V_{in_i} - V'_{sub-DAC_i} \,. \tag{2.7}$$

In the above equation, the $V'_{\text{sub-DACi}}$ is affected by the capacitor mismatches. The $V'_{\text{sub-DACi}}$ can be represented as

$$V'_{sub-DAC_i} = V_{ref} \cdot D'_i \,, \tag{2.8}$$

where $D'_i \in \{0, \pm(1+\Delta C_{s1}), \pm(2+\Delta C_{s1}+\Delta C_{s2}), \ldots \pm(j+\Delta C_{s1}+\Delta C_{s2}+\ldots \Delta C_{sj})\}$.

The output error results from capacitor mismatches can be calculated through subtracting Equation (2.7) by Equation (2.5). Here, the output error is

$$E = V'_{out_i} - V_{out_i} = \left(\Delta C_{s1} + \Delta C_{s2} + \ldots \Delta C_{s2}\right) \cdot V_{in_i} - V_{ref} \cdot \Delta D_i \,, \tag{2.9}$$

where $\Delta D_i \in \{0, \pm(\Delta C_{s1}), \pm(\Delta C_{s1}+\Delta C_{s2}), \ldots \pm(\Delta C_{s1}+\Delta C_{s2}+\ldots \Delta C_{sj})\}$.

2.2.2.3. Gain error source: finite operational-amplifier open-loop gain

The operational-amplifier is a very important part of an ADC. Because the residue, which is generated by subtracting the sub-DAC analog output from the input signal, is amplified through the operational-amplifier and feed to the following stage. Besides the capacitor mismatches, the finite operational-amplifier open-loop gain also introduces the gain error. As stated above, the gain error is a critical source of linear error [9]. When the converter shown in Figure 2-8 works in an ideal condition, the transfer function is

$$V_{out_i} = j \cdot V_{in_i} - V_{sub-DAC_i} = j \cdot (V_{in_i} - \frac{1}{j} V_{sub-DAC_i}), \tag{2.10}$$

where V_{outi} represents the output of the i^{th} stage, j is the total number of non-feedback capacitors, V_{ini} represents the input of the i^{th} stage, and $V_{sub\text{-}DACi}$ is the sub-DAC output of the i^{th} stage.

In contrast to this ideal condition, when the converter shown in Figure 2-8 has a finite open-loop gain, the transfer function changes to

$$V'_{out_i} = \frac{j}{1 + j/\alpha} \cdot (V_{in_i} - \frac{1}{j} V_{sub-DAC_i}), \tag{2.11}$$

where V'_{outi} represents the output of the i^{th} stage when the finite open-gain is considered and α is the open-loop gain. Therefore, the gain error results from the finite open-loop gain is

$$E = j - \frac{j}{1 + j/\alpha}. \tag{2.12}$$

In the ADC design then, in order to realize the destination resolution, the gain error E must be less than 1/2 LSB.

2.2.3. Thermal noise error

Thermal noise is a result of the random motion of electronics. In a switched-capacitor pipeline ADC, the switched-capacitor configuration contributes a significant amount of thermal noise. During the sampling phase, the switch thermal noise is also sampled on the capacitor [10]. As shown in Figure 2-9, in order to analyze the thermal noise results from the switched-capacitor configuration, the modelled sample circuit is a RC circuit. In this figure, when the switch is on, the resistance of the switch is R and the capacitance of the capacitor is C.

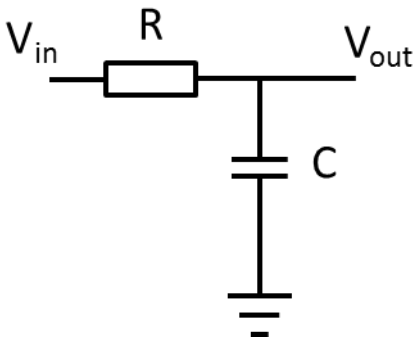


Figure 2-9. Modeled sample circuit with thermal noise

In order to access the total noise power, the power spectral density can be expressed as

$$S = 4KTR, \tag{2.13}$$

where K represents the Boltzman constant, T represents the thermodynamic temperature, and R represents the resistance of the switch when the switch is on. In addition, the transfer function of the RC circuit is

$$H(jw) = \frac{1}{1 + jwRC}, \tag{2.14}$$

where w, which represents the angular frequency with f representing the frequency, is $2\pi f$ and C represents the capacitance of the sampling capacitor.

Therefore, the total thermal noise power is calculated by the following formula:

$$P = \int_0^\infty 4KTR|H(jw)|^2 df = \int_0^\infty \frac{4KTR}{1 + (2\pi RCf)^2} df = \frac{KT}{C}. \tag{2.15}$$

For the fully differential configuration, the total noise power is calculated by multiplying Equation (2.15) by two. In order to eliminate power supply noise, charge injection effects, as well as clock feedthrough effects, a fully differential configuration needs to be adopted. As can be seen in Equation (2.15), the total noise power is proportional to KT/C, so the thermal noise is normally

called KT/C noise. This equation also illustrates that total thermal noise power is not related to the resistance R. In order to reduce total thermal noise, the capacitance need to be increased. However, the increased capacitance not only reduces input bandwidth but also increases power consumption. Therefore, in the analog-to-digital converter design, total thermal noise, speed, and power consumption need to be comprehensively considered.

2.3. Performance parameters of a pipeline ADC

2.3.1. Static performance parameters

Static performance parameters illustrate the differences between the actual quantification curve and the ideal quantification curve in the time domain. For the pipeline ADC, the normally considered static performance parameters are offset error, gain error, differential nonlinearity (DNL), and integral nonlinearity (INL).

2.3.1.1. Offset error

Figure 2-10 shows the ideal transfer curve and actual transfer curve with the offset error, respectively. The offset error is the difference between the starting points of the two curves.

Due to the offset error, the actual transfer curve is the one that moves the ideal transfer curve parallel to the right.

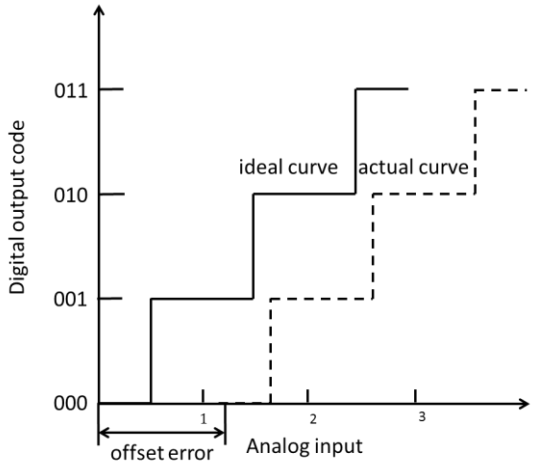


Figure 2-10. Offset error of a 3-bit ADC

2.3.1.2. Gain error

Figure 2-11 shows the ideal transfer curve and the actual transfer curve with the gain error, respectively. The gain error indicates how well the slope of the actual transfer curve matches the slope of the ideal transfer curve. The gain error can be measure by first coinciding the starting points and then measuring the difference between the ending points between two curves.

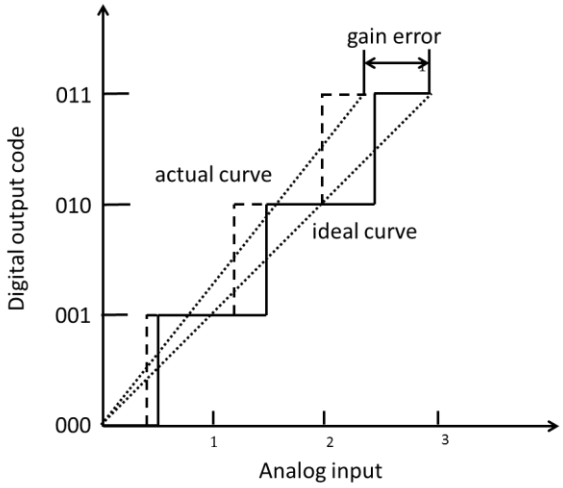


Figure 2-11. Gain error of a 3-bit ADC

2.3.1.3. Differential nonlinearity (DNL)

The DNL is defined as the actual step size deviation from the ideal step size (1-LSB) with the error normalized with respect to 1-LSB [11]. The DNL can be calculated by

$$DNL(\mathrm{k}) = \frac{V(\mathrm{k}) - LSB}{LSB}, \tag{2.16}$$

where $V(k)=V_{in}(k)-V_{in}(k-1)$ represents the real step size of the k^{th} step.

As can be seen from Equation (2.16), the DNL of the k^{th} step cannot be lower than -1, since the V(k) cannot be a negative number. If the DNL of the k^{th} step equals to -1, then it indicates a code loss, since the actual step size equals 0 in this case. In contrast, in the ideal case, DNL equals to 0 and V(k) equals to 1-LSB. The DNL(k) depends on the individual step size. Normally, in order to evaluate the static performance of a ADC, the DNL indicates the largest DNL(k).

2.3.1.4. Integral nonlinearity (INL)

The INL is the deviation between an actual transfer curve and an ideal curve as shown in Figure 2-12. This parameter shows the linearity of an ADC. In order to calculate the INL, a best-fit straight line or a line drawn between the end points of the transfer curve needs to be drawn [1] [12]. The end points method is usually used when addressing the INL for an ADC. The INL can be calculated by

$$INL(k) = \frac{V(k)}{LSB}, \tag{2.17}$$

where $V(k)=V_{in}(k)-V_{in}(k-1)$ represents the real step size of the k^{th} step.

For the k[th] step, the relationship between INL and DNL is calculated by

$$INL(\mathrm{k}) = \sum_{i=1}^{k} DNL_i \,, \tag{2.18}$$

where $i \in \{1, 2 \ldots \mathrm{k}\}$ represents the i[th] step.

Both offset and gain errors can be calibrated out using software or hardware. Therefore, DNL and INL are the two significant parameters needed in order to evaluate the static performance of an ADC. The code density histogram is normally used to calculate the DNL and INL of an ADC [13].

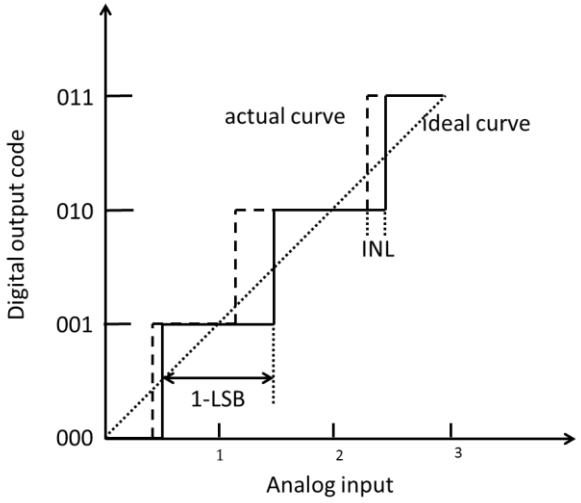


Figure 2-12. INL of an ADC

2.3.2. Dynamic performance parameters

In addition to the static errors, other error sources, called dynamic errors, also appear when an ADC works in a dynamic circumstance. Normally, the dynamic errors are frequency dependent and increase with the increase of frequency. The dynamic performance parameters are used to evaluate the dynamic errors of an ADC in dynamic circumstances.

The spurious-free dynamic range (SFDR) is the strength ratio of the fundamental signal to the strongest spurious signal in the output [14]. The SFDR can be calculated by

$$SFDR = 10\log_{10}\frac{P_{signal}}{P_{distortion\_\max}}, \tag{2.19}$$

where P_{signal} is the input signal power and $P_{distortion\_max}$ is the strongest spurious signal power in the output.

The total harmonic distortion (THD), a measurement of harmonic distortion, is defined as the ratio of the sum of the powers of all harmonic components to the power of the fundamental frequency [14]. Therefore, the THD is calculated by

$$THD = 10\log_{10}\frac{P_{distortion}}{P_{signal}}, \tag{2.20}$$

where $P_{distortion}$ is the total distortion power.

The signal-to-noise ratio (SNR) is defined as the ratio of signal power to noise power, often expressed in decibels. Therefore, the SNR is calculated by

$$SNR = 10\log_{10}\frac{P_{signal}}{P_{noise}}, \tag{2.21}$$

where P_{noise} is the noise power.

The effective number of bits (ENOB) is an alternative method to measure a specifying SNR [14]. The relationship between ENOB and SNR is calculated by

$$ENOB = \frac{SNR - 1.763}{6.02}.$$

The signal-to-noise and distortion ratio (SNDR) is a dynamic parameter that both THD and SNR are considered. This parameter comprehensively reflexes the dynamic property of the ADC. The SNDR is calculated by

$$SNDR = 10\log_{10}\frac{P_{\text{signal}}}{P_{noise} + P_{distortion}}. \tag{2.23}$$

Normally, the Harmonic Distortion ratio (HD) decreases with the increasing of harmonic order. The HD is calculated by

$$HD_2 = 10\log_{10}\frac{P_{distortion\_2}}{P_{\text{signal}}}, \tag{2.24}$$

$$HD_n = 10\log_{10}\frac{P_{distortion\_n}}{P_{\text{signal}}}, \tag{2.25}$$

where $P_{distortion\_2}$ is the power of the second harmonic component and $P_{distortion\_n}$ is the power of the n^{th} harmonic component. Since the first harmonic component is the input signal, HD_1 equals 0.

2.4. Conclusion

This chapter describes the pipeline ADC architecture, the error sources of the pipeline ADC, and the performance parameters. A pipeline ADC includes several pipeline stages. Each stage processes an input signal and produces an output signal for the following stage. In order to design a high performance ADC, the error sources must be analyzed. Theses error sources introduce non-idealities to the pipeline ADC. Therefore, several static and dynamic performance parameters are described in order to evaluate the pipeline ADC performance.

CHAPTER 3. TECHNIQUES FOR LOW POWER CONSUMPTION AND HIGH ACCURACY ADC DESIGN

3.1. ADC architecture and circuit implementation

The performance of the pipeline ADC is highly dependent on the number of bits resolved per stage. Two advantages for fewer bits per stage are lower accuracy requirements for the comparators and faster stage speeds due to large feedback factors. However, if fewer bits per stage are to be achieved, more stages are needed. Conversely then, the advantage for more bits per stage is that fewer stages are necessary. However, if more bits per stage are used, then the comparators in each stage need to have a higher accuracy due to the lower toleration to offset voltage, and the amplifiers require a larger bandwidth to meet the speed requirement [6]. If the ADC resolves one more bit per stage, the comparator offsets toleration will be reduced by half.

However, in the research presented here, the first two stages have less noise and gain errors than the later stages. Therefore it is reasonable to convert more bits for the first two stages in order to reduce the total number of stages. Here, a domain-extended digital error correction technique is introduced to increase the comparator offset toleration ability. In addition, the technique also provides crucial information on both overflow and underflow situations. Normally, a 2.5-bit/stage ADC only tolerates $\pm V_{ref}/8$ comparator offset using the traditionally digital error correction technique. This research proposes a domain-extended 2.5-bit/stage digital error correction technique. This proposed technique increases the comparator offsets toleration to $\pm 3V_{ref}/16$. The input signals may overflow/underflow the reference voltage. One way to realize overflow/underflow judgment is to add two more comparators to the first stage. This method is easy to implement, but has a very high accuracy requirement for these two comparators. In this

research, the overflow/underflow judgment bits can be accessed through the digital logic operations. Since the later stages of the pipeline ADC have increased noise and gain errors, the 1.5-bit/stage architecture has a higher error toleration ability than the 2.5-bit/stage and should be adopted. Therefore, the 1.5-bit/stage architecture is used from the third stage to the tenth stage. In addition, the gain boost amplifiers [6] are adopted to obtain a high open-loop gain and an excellent bandwidth with low power consumption.

In a MDAC, the mismatches between the non-feedback capacitor(s) and the feedback capacitor contribute substantially to the gain error and reference voltage offset. A multiple bit front-end relaxes the capacitor matching requirements and enhance power efficiencies [2] [5] [15]. In this research, the 2.5-bit/stage architecture is adopted by the first two stages and relaxes the capacitor matching requirement. The CFCS technique is used to reduce capacitor mismatches.

Figure 3-1 shows the block diagram of the proposed 14-bit ADC. This ADC eliminates a dedicated front-end SHA. It consists of two 2.5-bit/stage converters followed by eight 1.5-bit/stage converters and a flash ADC. The total output is 14-bit after digital error correction.

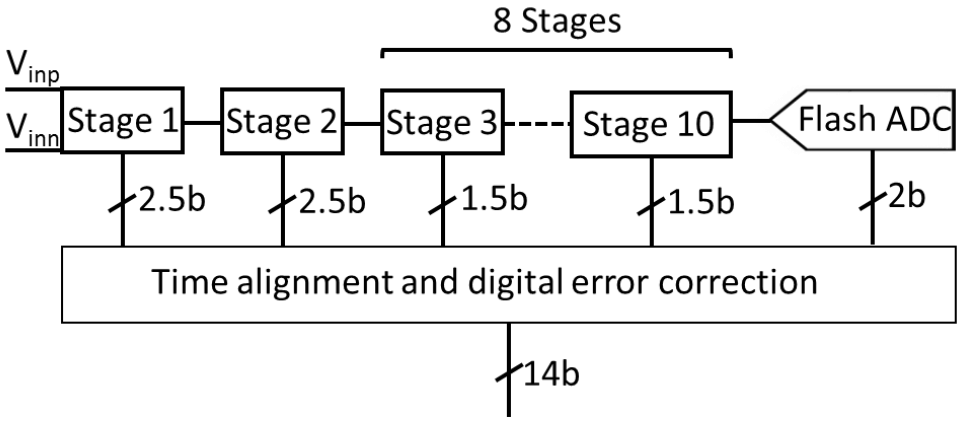


Figure 3-1. Block diagram of the proposed 14-bit ADC

In order to design a high accuracy and low power consumption ADC, multiple techniques are involved in this research. It is well known that a front-end SHA consumes a significant amount

of power and introduces a substantial noise and distribution. Therefore, a novel developed front-end stage with sample-and-hold function for SHA-less ADC is used to eliminate the power consumption as well as noise contribution of a front-end SHA. Another newly developed technique, domain-extended 2.5-bit/stage ADC, is used to relax the capacitor matching requirement and improve the comparator offset toleration ability. Furthermore, the CFCS technique is used to reduce the capacitor mismatches. The detailed implementation of these techniques are discuss in this chapter.

3.2. SHA-less ADC

3.2.1. Disadvantages of the pipeline ADC with a SHA

Figure 3-2 shows the flip-around sample-and-hold circuit and its corresponding control signal [2]. The actual configuration is fully differential; here, the single-ended configuration is shown for the sake of simplicity. When Φ_1 is high, the sample-and-hold circuit works on the sample phase. During this time, the input voltage is sampled on the capacitor C. As stated in Chapter 2, in order to eliminate the charge injection and clock feedthrough effects, the fall-edge of clock signal Φ_{1p} arrives a little earlier than the fall-edge of clock signal Φ_1, from high to low. In contrast, when Φ_2 is high, the sample-and-hold circuit works on the amplification phase. During this time, the left panel of the sample-and hold capacitor C connects to the output. Since the sample-and-hold amplifier has a very high open-loop gain, the right panel of this capacitor is still connected to the virtual ground. Therefore, the output voltage is approximately equal to the input voltage.

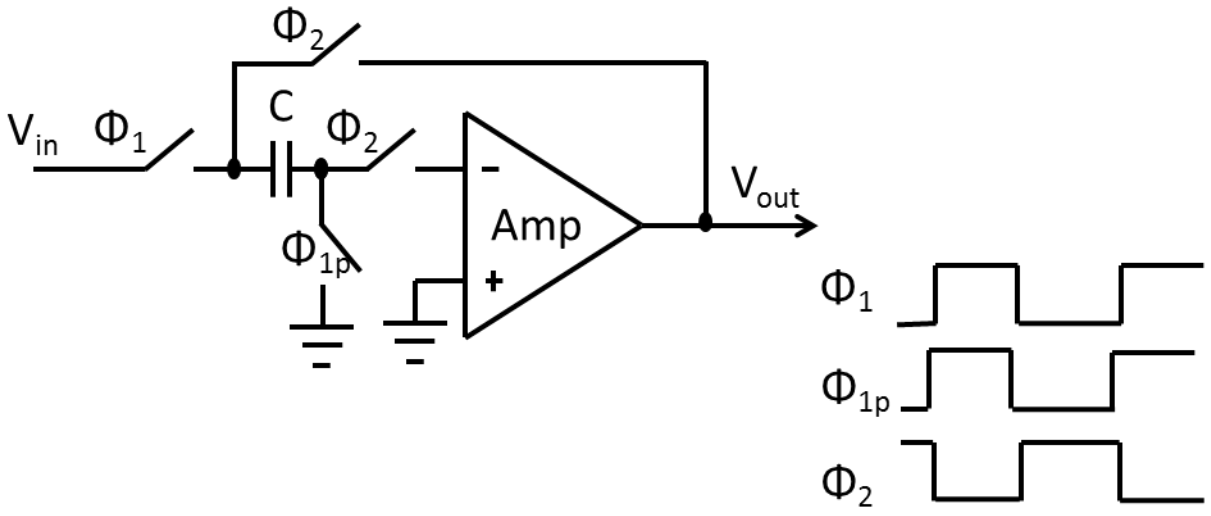


Figure 3-2. Flip-around sample-and-hold circuit

The front-end SHA is conventional used in pipeline ADCs. This SHA provides no signal gain but introduces noise and distortion to the entire ADC. In addition, the front-end SHA often needs to consume a significant amount of power to maintain the SNR as well as the linearity. Normally, at least half of the total power needs to be consumed by a SHA in order to have the SNR and linearity in an acceptable range [16]. Also, in the high speed and high resolution ADC design, the design of the front-end SHA is a very challenging task because the difficult tradeoffs between SNR, linearity, and power consumption.

3.2.2. Disadvantages of a SHA-less pipeline ADC using the matching technique

In order to reduce power consumption, noise distribution, and chip area, a SHA-less pipeline ADC using matching technique is reported [3] [4]. However, the different time constants of the MDAC signal path and the sub-ADC signal path may cause the aperture error become unacceptably large [3] [16]. Figure 3-3 shows the different time constants in the first stage of a SHA-less ADC. The aperture error caused by the different time constant is a critical problem for

SHA-less configuration. This is because that the first stage converter in the SHA-less ADC deals with a dynamic input rather than a held one. In order to eliminate the aperture error, the input signal needs to be sampled simultaneously by the sample-and-hold circuit of the MDAC and the sub-ADC. However, in practice, this synchronization is hard to achieve because of the different time constants of the two signal path.

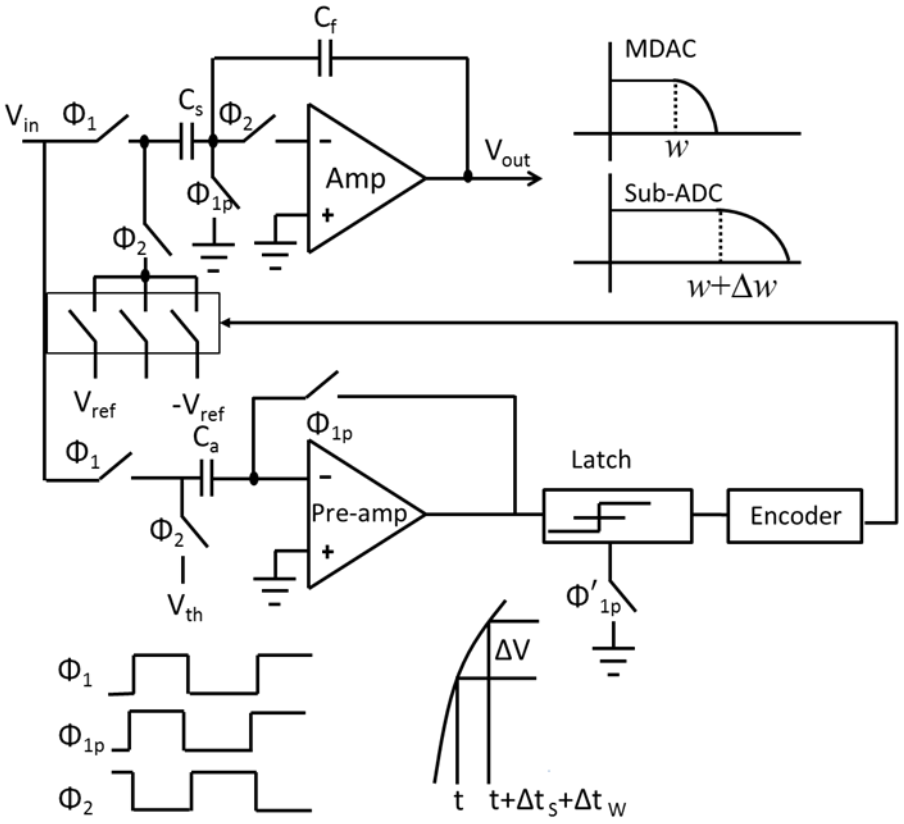


Figure 3-3. Different time constants in the first stage of a SHA-less ADC [17]

The different time constants of the two signal path result from these two signal path have differences in bandwidth as well as sampling instant. For example, the input signal is a sine wave, which can be expressed as

$$V_{in} = V_{ref} \sin w_{in} t \,, \tag{3.1}$$

where V_{ref}, the reference voltage, is equal to half of the signal full range and ω_{in} is the input frequency. The input bandwidths of the MDAC signal path and sub-ADC signal path are different. Hence assume the sampling network bandwidth of the MDAC and sub-ADC are ω and $\omega+\Delta\omega$, respectively. Also, assume the sampling instant of the MDAC and sub-ADC are t_1 and t_2, respectively. Thus, the sampled voltage through the MDAC path is [18]

$$V_1 = \frac{V_{ref}}{\sqrt{1+w_{in}^2/w^2}} \sin w_{in}\left(t_1 - \frac{\tan^{-1}(w_{in}/w)}{w_{in}}\right). \tag{3.2}$$

However, the sample voltage through the sub-ADC path is

$$V_2 = \frac{V_{ref}}{\sqrt{1+w_{in}^2/(w+\Delta w)^2}} \sin w_{in}\left(t_2 - \frac{\tan^{-1}(w_{in}/(w+\Delta w))}{w_{in}}\right). \tag{3.3}$$

Therefore, the sampled voltage mismatch between the MDAC signal path and sub-ADC signal path is

$$\begin{aligned}\Delta V = V_1 - V_2 = \\ \frac{V_{ref}}{\sqrt{1+w_{in}^2/w^2}} \sin w_{in}\left(t_1 - \frac{\tan^{-1}(w_{in}/w)}{w_{in}}\right) \\ - \frac{V_{ref}}{\sqrt{1+w_{in}^2/(w+\Delta w)^2}} \sin w_{in}\left(t_2 - \frac{\tan^{-1}(w_{in}/(w+\Delta w))}{w_{in}}\right)\end{aligned}. \tag{3.4}$$

In order to achieve good linearity, ω is normally choose to be much larger than ω_{in}. The largest sampled voltage mismatch is approximately equal to [18]

$$\Delta V_{\max} = V_{ref} \cdot w_{in} \cdot (\Delta t_s + \Delta t_w), \tag{3.5}$$

where $\Delta t_s=t_2-t_1$ is the sampling instant mismatch and $\Delta t_\omega=(1/(1+\omega_{in}^2/\omega^2))(\Delta\omega/\omega^2)$ is the propagation delay mismatch. The sampling instant mismatch Δt_s is normally less than 10^{-120} second and the propagation delay mismatch $\Delta t_\omega$is normally in 10^{-14} level [18]. Hence, the propagation delay dominates the time constant mismatch. Therefore, the aperture error result from the time constant mismatch can be almost completely eliminated through matching the MDAC and the sub-ADC signal paths in both design and layout.

For the SHA-less structure, carefully matching the time constant between the MDAC signal path and the sub-ADC signal path can reduce the aperture error [3] [4]. However this SHA-less structure involves two potential problems. The first potential problem is that clock buffers, routing, and sampling switches must be carefully matched in both design and layout, in order to reduce the aperture error caused by the clock skew and the bandwidth mismatches between the two signal paths. However, the two signal paths are very different in location and size, so it is very hard to achieve high-precision matching in bandwidth [6] [19]. The second potential problem is that the sampling network of the sub-ADC must have the similar topology as the one of MDAC. When matching is required for SHA-less ADC, a high-speed and low power consumption dynamic comparator is not suitable for the sub-ADC [20].

3.2.3. Novel SHA-less pipeline ADC architecture and implementation

This research proposes a novel configuration for SHA-less ADC without matching the MDAC and sub-ADC signal paths and thus avoids potential problems associated with the traditional SHA-less ADC. This structure also allows the adoption of the high-speed and low power consumption dynamic comparators in the sub-ADC.

The configuration of the first stage with a sample-and-hold function for a SHA-less pipeline ADC is shown in Figure 3-4 [21]. Although the real configuration is fully differential,

only the single-ended configuration is shown for the sake of clarity. The implementation of this configuration is discussed as following: the fall-edge of phase Φ_{1p} and Φ arrives at the same time. At this time, the input V_{in} is simultaneously sampled on capacitors C, C_1 and C_h. Capacitor C denotes a capacitor array, which includes several capacitors, C_2, C_3, and C_4. The detailed usage of C_1, C_2, C_3, and C_4 is discussed in Section 3.3. In order to prevent signal dependent charges from being injected into capacitors C, C_1, and C_h through the sample switches, clock signal ϕ_1 has a short delay compared to clock signal Φ_{1p}. Thus, the fall-edge of phase Φ_{1p} arrives a little earlier than the fall-edge of phase Φ_1. At the fall-edge of the phase Φ_{1p}, the charges stored on the capacitor C_h is $Q_h=C_h\times V_{in}$. At this time, and for the same reason, the charges stored on capacitors C and C_1 are $Q= C\times V_{in}$ and $Q_1= C_1\times V_{in}$, respectively. Since switches S' and S_h' are turned off after the fall-edge of phase Φ_{1p} and phase Φ, the voltage difference, $\Delta V= V_{in} =Q/C= Q_1/C_1= Q_h/C_h$, between the two panels for capacitors C, C_1, and C_h stays unchanged.

Afterwards, the raise-edge of phase Φ_2 and Φ arrives. At this time, since the right panel of capacitor C_h is connected to ground and because the charges stored on the capacitor C_h much be conserved, the voltage at node X reaches to a value equal to V_{in}. Since both the sampling times for MDAC and sub-ADC are defined by Φ_{1p}, there are no the usual problems associated with skewed sampling times, which would degrade performance at high frequencies.

At the same time, the left side of the sample-and-hold capacitor C_h is connected to the input of the dynamic comparator and the V_{in} is quantized through the sub-ADC. The input of the dynamic comparator is located at the gate of the NMOS as shown in Figure 4-5. Since the input impedance of the dynamic comparator is in meg-ohms range, the charges stored on the sample-and-hold capacitor are almost completely held on the capacitor. According to the resolved bits of the sub-

ADC, the left sides of the capacitor array are switched to the proper reference voltages in order to produce the residue and amplify the residue signal.

Moreover, this configuration eliminates the charge injection effect, which is caused by the charges stored on the switches being injected into the capacitors during the switches' off time. Also, this configuration eliminates the clock feedthrough effect, which is caused by the charges stored on the gate-drain overlap capacitance of the switches being injected into the capacitors during the switch's off time. Since the fall-edges of phaseΦ_{1p} andΦ arrive earlier than the fall-edge of phaseΦ_1, the switches $S^{'}$ and $S_h^{'}$ are turned off earlier than S, S_1, and S_h. Therefore, the charge injection and the clock feedthrough effects caused by the switches $S^{'}$ and $S_h^{'}$ will be discussed first. Since the injected charges are signal dependent, and because both source and drain of switches $S^{'}$ and $S_h^{'}$ are connected to constant voltages, the injected charges can only introduce a constant voltage change. This introduced constant voltage change can be eliminated through the fully differential configuration. Then, after a short delay, the fall-edge of phaseΦ_1 arrives. At this time the three switches S, S_1, and, S_h, which are all controlled by phaseΦ_1, are turned off. Second in this consideration is the charge injection and the clock feedthrough effects caused by these switches, S, S_1, and S_h. Due to the above mentioned reason, the charges on the capacitor C_h must be conserved. In addition, because the switch $S_h^{'}$ is switched to ground, the voltage at node X reaches to a value equal to V_{in}. Thus, the charge injection and the clock feedthrough effects introduced by switches S, S_1, and S_h cannot change the voltage at node X. Therefore, the problematic charge injection and the clock feedthrough effects are eliminated.

A front-end SHA consumes at least half of the total power and contributes noise and distortion to the ADC. In the proposed configuration, adding two sample-and-hold capacitors and

several switches reduces the power consumption and the chip area significantly. Also, the noise and distortion related to the front-end SHA can be eliminated.

In contrast, a SHA-less ADC based on matching techniques requires a substantial design effort in order to match the sub-ADC and the MDAC signal paths. Also, more power in the SHA-less ADC based on matching techniques is consumed due to the fact that the low power consumption dynamic comparator is not suitable for this configuration. However, the proposed configuration is not based on the traditional matching techniques; therefore, both power and design effort can be saved for the proposed SHA-less ADC.

As stated before, for the sake of clarity, a single-ended configuration is shown in Figure 3-4. However, the actual proposed configuration is fully differential. This proposed configuration involves minor analog modifications, which include adding two capacitors and several switches. In this configuration, both the sampling instants for MDAC and sample-and-hold circuit are defined byΦ_{1p} and the sample-and-hold switches S_h and S have the same configuration, so there are almost no clock skews that would degrade performance at high frequencies. Also, during the amplification phase, the left side of the capacitor C_h is connected to the input of the dynamic comparator. The input impedance of the dynamic comparator is in meg-ohms range, so the charges are almost completely held on the capacitor C_h. Furthermore, the charge injection and the clock feedthrough effects are eliminated.

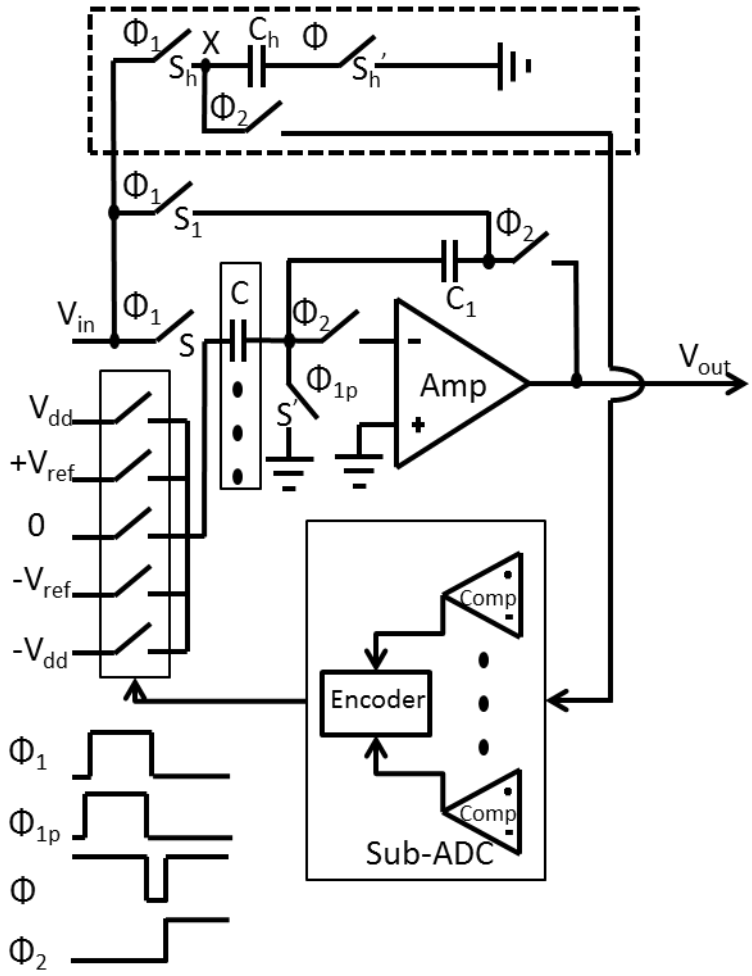


Figure 3-4. First stage configuration with the sample-and-hold function

3.3. Domain-extended 2.5-bit/stage ADC

3.3.1. Domain-extended 2.5-bit/stage ADC transfer function

Although a 2.5-bit/stage ADC has a lower comparator offset toleration ability than a 1.5-bit/stage ADC [22] [23] [24], the same resolution can be realized within fewer stages than a 1.5-bit/stage ADC. This means less delay time. In order to achieve higher resolution with fewer stages and less delay time, a 2.5-bit/stage architecture for the first two stages is preferable because the first two stages suffer from less noise and distortion than the later stages.

It is well known that the traditional 2.5-bit/stage ADC can correct the comparator offset within $\pm V_{ref}/8$. The error toleration ability of a 2.5-bit/stage ADC is reduced by half compared to a 1.5-bit/stage ADC. In this research, a domain-extended 2.5-bit/stage ADC is introduced to increase the comparator offset toleration ability to $\pm 3V_{ref}/16$.

In order to compare the comparator offset toleration ability of both the traditional and the domain-extended 2.5-bit/stage ADCs, the transfer characteristics of both configurations are show in Figure 3-5. The transfer characteristics of the traditional 2.5-bit/stage ADC are shown in Figure 3-5 (a). The ideal threshold voltages of the comparators are $\pm 5V_{ref}/8$, $\pm 3V_{ref}/8$, and $\pm V_{ref}/8$ shown with dotted lines in Figure 3-5 (a). The digital output of each domain from left to right are 000, 001, 010 …110 in Figure 3-5 (a). When one of the comparators has an offset of $V_{ref}/8$, the transfer characteristics are changed to the dashed lines and the maximum output is equal to V_{ref}. Since the output of the current stage is the input of the following stage, and the input range is from $-V_{ref}$ to V_{ref}, the out-of-range output leads to a code loss. To prevent code loss, the comparator offsets for the traditional 2.5-bit/stage ADC are within $\pm V_{ref}/8$. In this research, as shown in Figure 3-5 (b), a domain-extended 2.5-bit/stage ADC increases the comparator offset toleration ability to $\pm 3V_{ref}/16$. By adding two comparators with comparison voltages of $\pm 7V_{ref}/8$ to the upper and lower sides of the traditional 2.5-bit/stage ADC, the analog quantify domain is a nine-domain as shown in Figure 3-5 (b) rather than a seven-domain as shown in Figure 3-5 (a). The digital output of each domain from left to right are 0000, 0001, 0010 …1000 in Figure 3-5 (b). When one of the comparators has comparator offset of $3V_{ref}/16$, the transfer characteristics are changed to the dashed oblique lines and the maximum output is $10V_{ref}/8$. Since the input range is from $-10/8V_{ref}$ to $10/8V_{ref}$ after domain extension, the maximum absolute values of comparator offsets allowed without code loss are $|3V_{ref}/16|$.

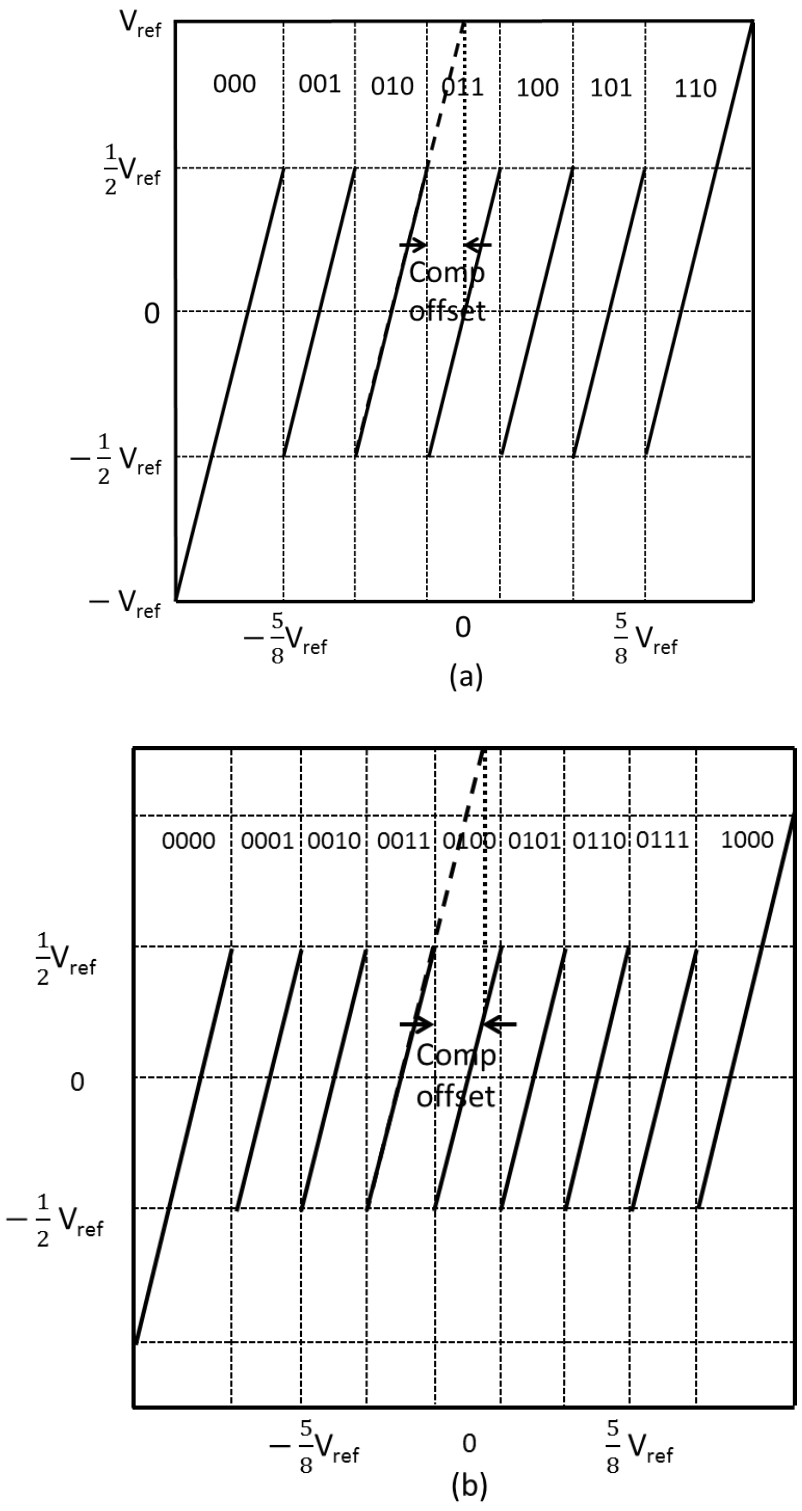


Figure 3-5. (a) Traditional 2.5-bit/stage ADC transfer characteristics and (b) domain-extended 2.5-bit/stage ADC transfer characteristics

3.3.2. Domain-extended 2.5-bit/stage ADC algorithm process

The domain-extended 2.5-bit/stage ADC not only increases the comparator offset toleration ability, but also realizes overflow/underflow judgement. The domain-extended 2.5-bit/stage ADC has two options for realizing overflow/underflow judgment. In [18], two comparators are utilized to observe out-of-range inputs; however, this method requires very high accuracy comparators to realize precise judgements. This is hard to achieve because it requires very high accuracy comparators. The other option is to generate the overflow/underflow bits through digital logic. This is a more accessible method. In this design, the latter option is used.

The algorithm process of this proposed domain-extend 2.5-bit/stage ADC is described in Figure 3-6. In order to have the digital output of this domain-extended 2.5-bit/stage consistent with the traditional 2.5-bit/stage, it needs to have the subtraction of one operation. However, a negative number will occur for 0000 minus 1. Adding a '1' in front of first stage digital output will resolve this problem. For the same reason, the second stage digital output also needs the subtraction of one operation. This dislocation addition should be implemented first, then the subtraction of one, so that a negative number can be avoided. In order to prevent code loss, the third stage is a domain-extended 1.5-bit converter [25]. Its transfer function and the circuit configuration are shown in Chapter 6. The first two bits of the digital output are utilized as the overflow/underflow bits. When they are '11' or '01', it means overflow and underflow, respectively; otherwise when they are '10', it means the remaining 14 bits are usable digital output. Figure 3-7 (a) and (b) are transfer curves based on the domain-extended 2.5-bit/stage and the traditional 2.5-bit/stage ADC, respectively. In the traditional method, even if analog input is out of $\pm V_{ref}$ range, the digital output cannot sense the overflow/underflow.

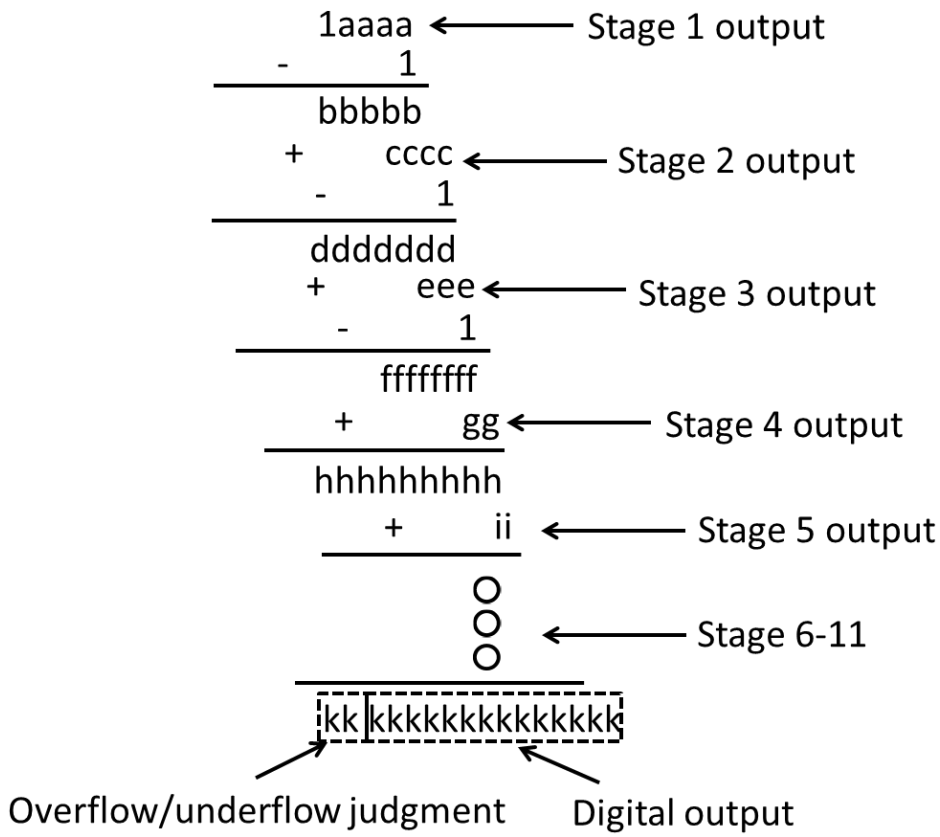


Figure 3-6. Domain-extended 2.5-bit/stage ADC algorithm processes

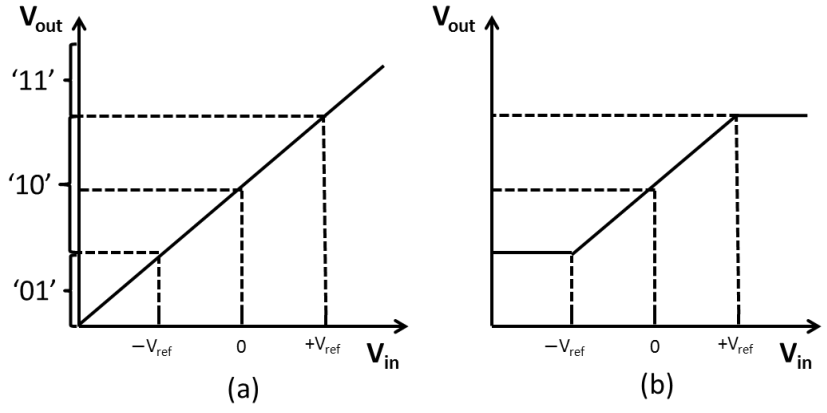


Figure 3-7. (a) Domain-extended 2.5-bit/stage ADC transfer curve and (b) traditional 2.5-bit/stage ADC transfer curve

3.3.3. Domain-extended 2.5-bit/stage ADC circuit implementation

In order to implement the above algorithm in the circuit level, the transfer function of the domain-extended 2.5-bit/stage pipeline ADC stage is given by the following equation

$$V_{\text{out}} = \begin{cases} 4V_{in}+4V_{ref}, & D=0000 \\ 4V_{in}+3V_{ref}, & D=0001 \\ 4V_{in}+2V_{ref}, & D=0010 \\ 4V_{in}+1V_{ref}, & D=0011 \\ 4V_{in}+0V_{ref}, & D=0100 \\ 4V_{in}-1V_{ref}, & D=0101 \\ 4V_{in}-2V_{ref}, & D=0110 \\ 4V_{in}-3V_{ref}, & D=0111 \\ 4V_{in}-4V_{ref}, & D=1000 \end{cases} . \tag{3.6}$$

The circuit level realization of Equation (3.6) is given by

$$V_{\text{out}} = \begin{cases} \frac{C_1+C_2+C_3+C_4}{C_1}V_{in}+\frac{2C_2+C_3+C_4}{C_1}V_{ref}, & D=0000 \\ \frac{C_1+C_2+C_3+C_4}{C_1}V_{in}+\frac{C_2+C_3+C_4}{C_1}V_{ref}, & D=0001 \\ \square\square\square & \\ \frac{C_1+C_2+C_3+C_4}{C_1}V_{in}-\frac{C_2+C_3+C_4}{C_1}V_{ref}, & D=0111 \\ \frac{C_1+C_2+C_3+C_4}{C_1}V_{in}-\frac{2C_2+C_3+C_4}{C_1}V_{ref}, & D=1000 \end{cases} , \tag{3.7}$$

where C_2, C_3 and C_4 are chosen to be equal to C_1 in order to provide a gain of two, three, or four for V_{ref}. C_2, C_3, and C_4 compose the capacitor array C as shown in Figure 3-4. When the required gain is two or three, the circuit level realization is the same as the traditional 2.5-bit/stage pipeline ADC, and C_2, C_3, and C_4 are connected to the relevant reference voltages. However, a gain of four for V_{ref} cannot be implemented through the traditional method since one of the four capacitors must be used as the feedback capacitor. Therefore, the maximum gain, three, is the sum of the

remainder three capacitors divided by the feedback capacitor. In this design, the proposed method achieves a gain of four for V_{ref}, as previously shown in Equation (3.7). V_{ref} is half of V_{dd}; therefore, the first and the last sub-equations of Equation (3.7) are

$$\begin{aligned}&\frac{C_1+C_2+C_3+C_4}{C_1}V_{in}+\frac{2C_2+C_3+C_4}{C_1}V_{ref}\\&=4V_{in}+\left(\frac{C_2}{C_1}V_{dd}+\frac{C_3+C_4}{C_1}V_{ref}\right)\\&=4V_{in}+4V_{ref}\end{aligned}, \tag{3.8}$$

$$\begin{aligned}&\frac{C_1+C_2+C_3+C_4}{C_1}V_{in}-\frac{2C_2+C_3+C_4}{C_1}V_{ref}\\&=4V_{in}-\left(\frac{C_2}{C_1}V_{dd}+\frac{C_3+C_4}{C_1}V_{ref}\right)\\&=4V_{in}-4V_{ref}\end{aligned}. \tag{3.9}$$

The circuit level configurations of the traditional 2.5-bit/stage ADC and the domain-extended 2.5-bit/stage ADC are shown in Figure 3-8 (a) and Figure 3-8 (b), respectively. Although the real configurations of both cases are fully differential, only the single-ended configurations are shown for the sake of clarity. These two converters work on a sample phase during phase Φ_1, while the converters work on an amplification phase during phaseΦ_2. As shown in Figure 3-8, in the amplification phase capacitor C_1 is the feedback capacitor, while other capacitors, C_2, C_3, and C_4, are connected to the relevant reference voltages according to the output of the sub-ADC.

Therefore, this proposed configuration involves minimal analog modifications. Compared to Figure 3-8 (a), the added components are shown in the dashed boxes in Figure 3-8 (b). Two extra comparators, with the comparison voltages of $\pm 7V_{ref}/8$, are added to provide two extra quantify domains. In addition, two references voltages, V_{dd} and $-V_{dd}$, are added in the sub-DAC to provide a gain of four for V_{ref}. Since the configurations are fully differential, $-V_{ref}$ can be realized

by connecting the ground to the positive input side and V_{ref} to the negative input side of the amplifier in both configurations. These findings show that the realization of $-V_{dd}$ is similar to the realization of $-V_{ref}$. In contrast, in the digital domain only several subtraction and dislocation blocks need to be added.

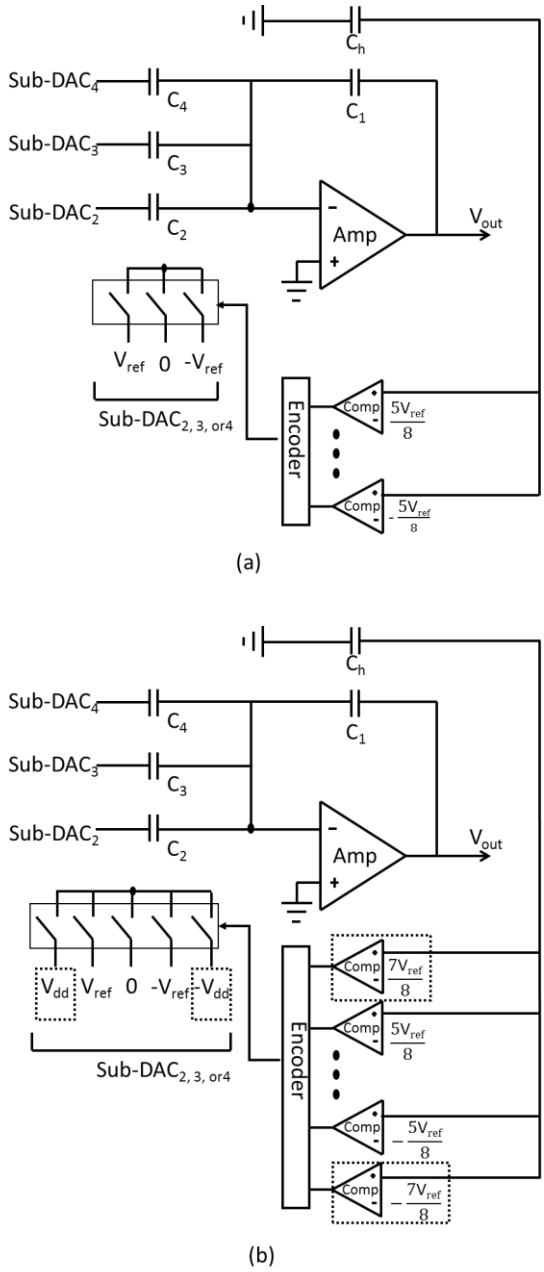


Figure 3-8. 2.5-bit/stage ADC circuit implementation using (a) the traditional technique and (b) the proposed domain-extended technique

3.3.4. Domain-extended 2.5-bit/stage ADC behaviour simulations

In order to demonstrate the comparator offset correction ability of the domain-extended 2.5-bit/stage ADC, Figure 3.9 (a) shows the ideal 3-bit ADC transfer characteristics and Figure 3.9 (b) shows the domain-extended 2.5-bit/stage ADC transfer characteristics with a comparator offset of $3V_{ref}/16$. According to the ideal transfer characteristic of the 3-bit ADC shown in Figure 3.9 (a), if the input voltage is $V_{ref}/16$, the ideal digital output is 100. However, as shown in Figure 3.9 (b), one of the comparators has an offset of $3V_{ref}/16$. The ideal threshold voltage of this comparator is $-V_{ref}/8$. Thus, the comparator threshold voltage is changed to $V_{ref}/16$. The worst case is that in which the input is $V_{ref}/16$ and its corresponding digital output code is 0011. The digital output code can be corrected back through the following procedure. First, according to the transfer characteristics shown in Figure 3.9 (b), the output of the initial stage is $10V_{ref}/8$. Hence, the digital output code of the second stage is 1000. Then, the output code of the third stage is 100. Finally, applying the algorithm process shown in Figure 3-6, the digital output code is 10100001. As stated before, the first two bits are overflow/underflow bits. Since the first two bits are 10, the remaining bits are usable. Therefore, the digital output code is 100001. The first three bits are 100, which are the same as the ideal output codes. Thus, the digital output codes are corrected.

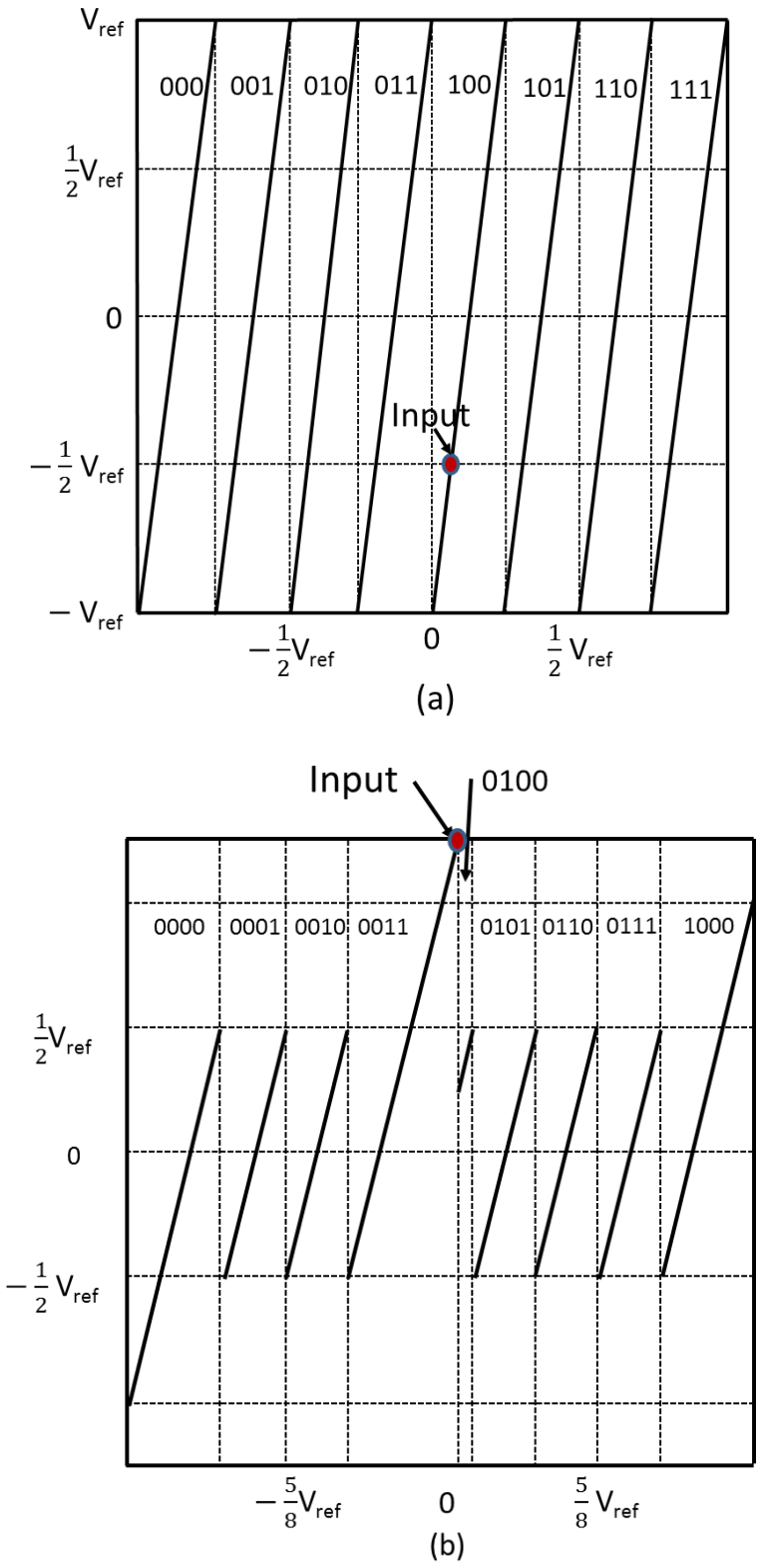


Figure 3-9. (a) Ideal 3-bit ADC transfer characteristics without compactor offset and (b) domain-extended 2.5-bit/stage ADC transfer characteristics with a comparator offset of $3V_{ref}/16$

The example stated above shows the worst case in which the ideal threshold voltage is $-V_{ref}/8$ and the comparator offset is $3V_{ref}/16$. Table 3-1 shows the worst cases in which the ideal threshold voltages are $\pm 7V_{ref}/8$, $\pm 5V_{ref}/8$, $\pm 3V_{ref}/8$, and $\pm V_{ref}/8$ and the comparator offset is $3V_{ref}/16$. Again, applying the algorithm process to the output codes of the first three stages, the first three bits of the final results are consistent with the ideal 3-bit output. In addition, using the same method mentioned above also demonstrates that the first two stages correct the comparator offset within $\pm 3V_{ref}/16$.

Table 3-1. Comparator offset equals $3V_{ref}/16$ in the first stage

Ideal threshold voltages	Real threshold voltages	Worst case input	Ideal output codes	Stage1 output codes	Stage2 output codes	Stage3 output codes	Result with O/U bits	Final result
-7/8	-11/16	-11/16	001	10000	1000	100	10001001	001001
-5/8	-7/16	-7/16	010	10001	1000	100	10010001	010001
-3/8	-3/16	-3/16	011	10010	1000	100	10011001	011001
-1/8	1/16	1/16	100	10011	1000	100	10100001	100001
1/8	5/16	5/16	101	10100	1000	100	10101001	101001
3/8	9/16	9/16	110	10101	1000	100	10110001	110001
5/8	13/16	13/16	111	10110	1000	100	10111001	111001
7/8	17/16	17/16	Overflow	10111	1000	100	11000001	Overflow

In order to further illustrate the comparator offset correction ability improvement for the domain-extended 2.5-bit/stage ADC, Matlab behaviour simulations are implemented. Two ideal 10-bit pipeline ADCs, except the comparator offsets of the first two stages, are simulated. The first ADC consists of two traditional 2.5-bit/stage converters followed by four 1.5-bit/stage converters and a flash ADC. The second ADC consists of two domain-extended 2.5-bit/stage converters followed by four 1.5-bit/stage converters and a flash ADC. For the second configuration, in order to prevent code loss, the third stage is a domain-extended 1.5-bit/stage converter [25]. This configuration is the same as the real chip configuration. In this simulation, the total output codes are 23176.

Figure 3-10 shows the total miscode numbers and their related comparator offsets. Figure 3-10 illustrates that when the comparator offsets of the domain-extended 2.5-bit/stage converters are within $\pm 3V_{ref}/16$ range, no miscode exists for the second ADC. However, for the first ADC, miscodes happen when the comparator offsets of the first two stage are out of $\pm V_{ref}/8$ range.

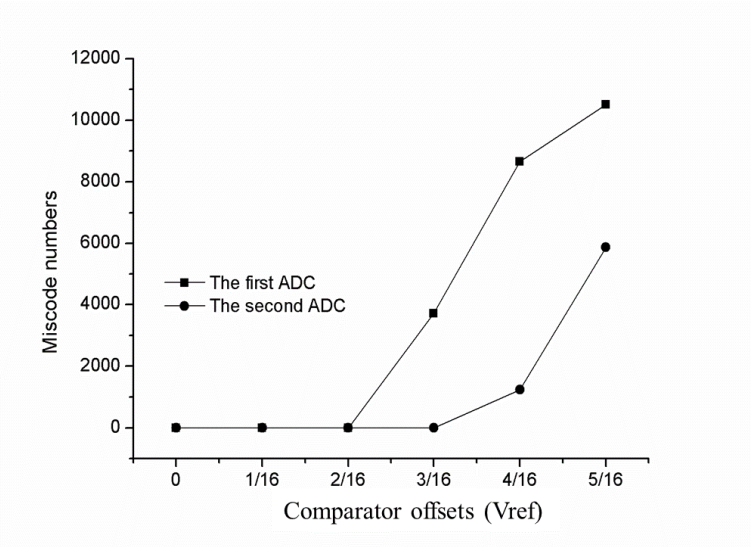


Figure 3-10. Total miscode numbers for various comparator offsets

In order to demonstrate the effectiveness of this technique, a comparison between the simulated dynamic performances of the two ADCs is shown in Figure 3-11. As shown in Figure 3-11 (a), SFDR, THD, and SNR decrease dramatically when comparator offsets are higher than $1/8V_{ref}$, since the traditional digital error correction technique can only correct the compactor offset within $\pm V_{ref}/8$. As shown in Figure 3-11 (b), SFDR, THD, and SNR stay almost the same, from the low comparator offset to the comparator offset of $3V_{ref}/16$, since the domain-extended 2.5-bit/stage digital error correction corrects the comparator offset within $\pm 3V_{ref}/16$.

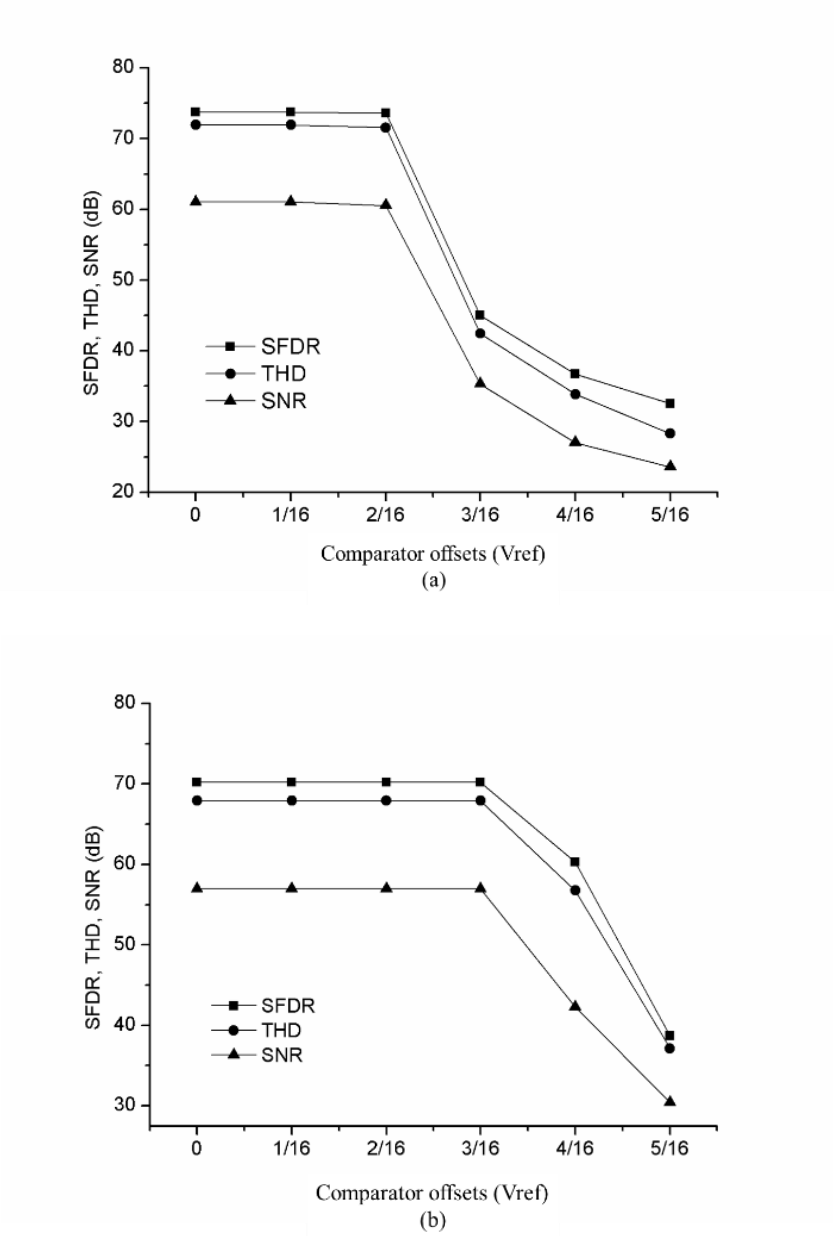


Figure 3-11. Dynamic performance of the ADC based on (a) the traditional 2.5-bit/stage architecture and (b) the domain-extended 2.5-bit/stage architecture

3.4. Communicated feedback-capacitor switching

A number of techniques have been developed to reduce capacitor mismatching requirements, such as the series capacitor technique [17], the average capacitor error technique [18], and the digital error correction technique [19]. The CFCS technique implements two

capacitors alternatively as the feedback capacitors, thereby significantly relaxing capacitor mismatching requirements [18]. Only one digital block and several switches added per stage are required for the CFCS technique, which in conjunction with digital error correction improves ADC accuracy.

The detailed implementation of the CFCS technique is as follows. The transfer characteristics are shown in Figure 3-12. In this figure, the feedback capacitors of the ADC using CFCS technique are shown in the dashed areas. The circuit level implementation of the traditional and the CFCS 1.5bit/stage ADC are shown in Figure 3-13 (a) and Figure 3-13 (b), respectively. As shown in Figure 3-13 (a), C_5 is always used as the feedback capacitor in the traditional method. In contrast, in Figure 3-13 (b), C_5 and C_6 are used alternatively as feedback capacitors. When the digital output is "00" or "10," C_5 is the feedback capacitor. Otherwise, when the digital output is "01," C_6 is the feedback capacitor. Points a and b are junction points between "00" and "01," assuming that $C_5 > C_6$, $C_7 = (C_5 + C_6)/2$, $C_5 = C_7(1+\varepsilon_1)$, $C_6 = C_7(1+\varepsilon_2)$, where ε_1 and ε_2 are capacitor mismatches for C_5 and C_6 compared to ideal capacitor C_7.

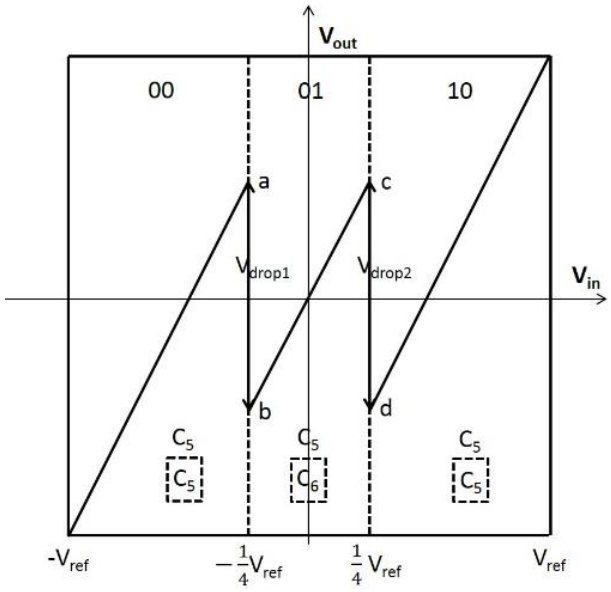


Figure 3-12. Transfer characteristics of the pipeline ADC

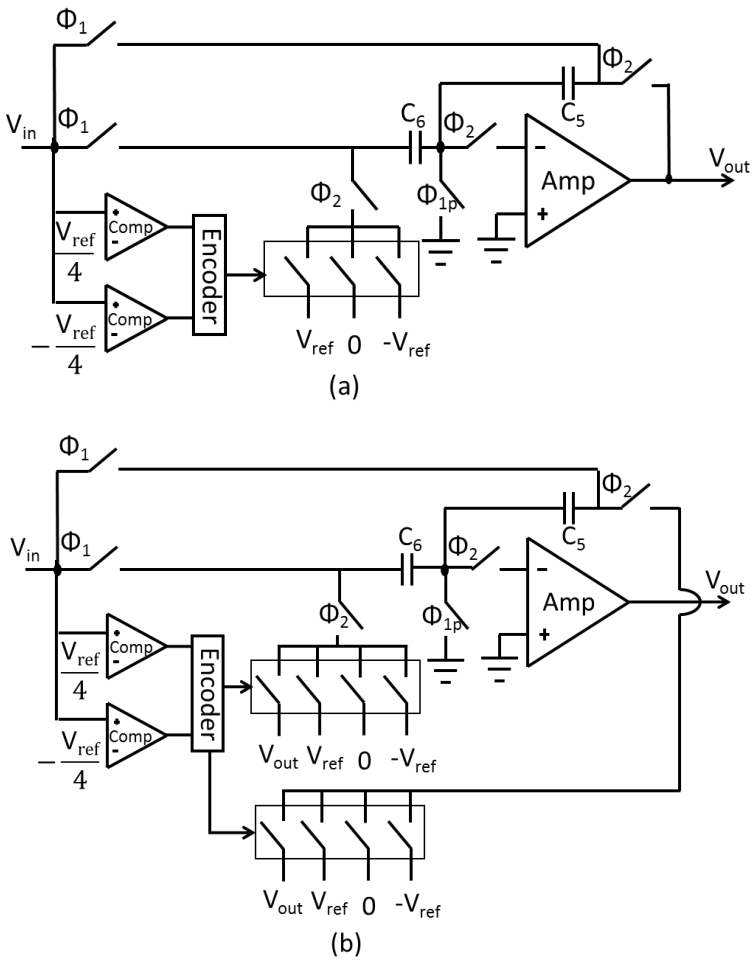


Figure 3-13. 1.5-bit/stage ADC circuit implementation using (a) the traditional technique and (b) the CFCS technique

The transfer function of a 1.5-bit/stage pipeline ADC stage using the traditional technique is given by [20]

$$V_{out}=\begin{cases}\dfrac{C_5+C_6}{C_5}V_{in}+\dfrac{C_6}{C_5}V_{ref}, & V_{in}\le\dfrac{-V_{ref}}{4}\\[2ex]\dfrac{C_5+C_6}{C_5}V_{in}, & \dfrac{-V_{ref}}{4}<V_{in}\le\dfrac{V_{ref}}{4}\\[2ex]\dfrac{C_5+C_6}{C_5}V_{in}-\dfrac{C_6}{C_5}V_{ref}, & V_{in}>\dfrac{V_{ref}}{4}\end{cases}. \tag{3.10}$$

Substituting both V_a and V_b equal to $-V_{ref}/4$ into Equation (3.10) gives

$$V_a = \frac{2}{1+\varepsilon_1}\left(\frac{-V_{ref}}{4}\right) + \frac{1+\varepsilon_2}{1+\varepsilon_1}V_{ref}, \tag{3.11}$$

$$V_b = \frac{2}{1+\varepsilon_1}\left(\frac{-V_{ref}}{4}\right), \tag{3.12}$$

$$V_{drop1} = V_a - V_b = \frac{1+\varepsilon_2}{1+\varepsilon_1}V_{ref} \approx (1+\varepsilon_2-\varepsilon_1)V_{ref}. \tag{3.13}$$

Substituting V_c and V_d equal to $V_{ref}/4$ into Equation (3.10) gives

$$V_{drop2} \approx (1+\varepsilon_2-\varepsilon_1)V_{ref}. \tag{3.14}$$

The error introduced due to mismatch is $(\varepsilon_2 - \varepsilon_1)V_{ref}$, which makes DNL worse.

The transfer function of a 1.5-bit/stage pipeline ADC stage using the CFCS technique is given by

$$V_{out} = \begin{cases} \frac{C_5+C_6}{C_5}V_{in} + \frac{C_6}{C_5}V_{ref}, & V_{in} \le \frac{-V_{ref}}{4} \\ \frac{C_5+C_6}{C_6}V_{in}, & \frac{-V_{ref}}{4} < V_{in} \le \frac{V_{ref}}{4} \\ \frac{C_5+C_6}{C_5}V_{in} - \frac{C_6}{C_5}V_{ref}, & V_{in} > \frac{V_{ref}}{4} \end{cases}. \tag{3.15}$$

Using the same method as above, we then get

$$V_a = \frac{2}{1+\varepsilon_1}\left(\frac{-V_{ref}}{4}\right) + \frac{1+\varepsilon_2}{1+\varepsilon_1}V_{ref}, \tag{3.16}$$

$$V_b = \frac{2}{1+\varepsilon_1}\left(\frac{-V_{ref}}{4}\right), \tag{3.17}$$

$$V_{drop1} = V_a - V_b \approx \left(1+\frac{\varepsilon_2-\varepsilon_1}{2}\right)V_{ref}, \tag{3.18}$$

$$V_{drop2} \approx (1+\frac{\varepsilon_2 - \varepsilon_1}{2})V_{ref} \cdot \quad (3.19)$$

The error introduced due to mismatch is $((\varepsilon_2 - \varepsilon_1)/2)V_{ref}$, which is half of the error using the traditional method. Thus, DNL is improved.

3.5. Conclusion

This chapter describes multiple techniques that are used to design a low power consumption and high accuracy ADC. In order to reduce power consumption, a novel configuration of the front-end stage with a sample-and-hold function is developed. Moreover, in order to improve accuracy, a domain-extended 2.5-bit/stage architecture is also developed to improve the comparator offset toleration ability. In addition, the CFCS technique is adopted to reduce the error due to the capacitor mismatches.

CHAPTER 4. CIRCUIT CONFIGURATIONS AND SIMULATIONS

In Chapter 3, two techniques are proposed, and one is adopted, for the design of a high accuracy ADC with minimized power consumption. The overall concept of the techniques and circuit implementations are also introduced in Chapter 3. However, the detailed circuit configurations and simulations are not demonstrated in Chapter 3. Therefore, the major purpose of Chapter 4 is to discuss the detailed circuit configurations and simulations. In this pipeline ADC design, the relationship between the ADC and the ADC components is shown in Figure 4-1. Chapter 4 is organized as follows: the configuration of the sub-ADC is presented in Section 4.1; the configuration of the MDAC is presented in Section 4.2; the configuration of the flash-ADC is presented in Section 4.3; the configuration of the clock is presented in Section 4.4; the configuration of the digital logic is presented in Section 4.5; and the conclusion is presented in Section 4.6.

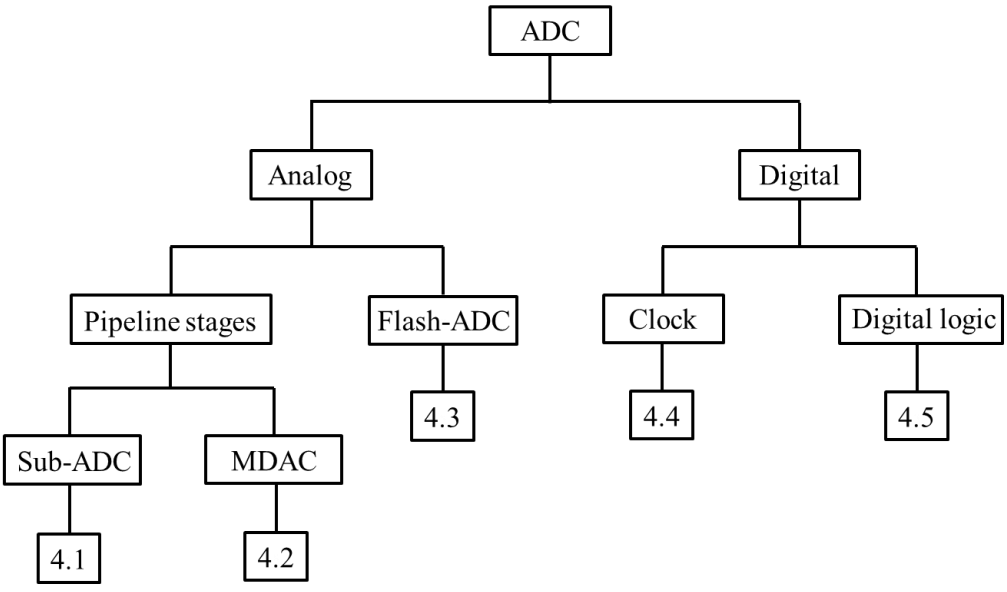


Figure 4-1. Relationship between the ADC and the ADC components

4.1. Sub-ADC configuration

In the pipeline ADC, each pipeline stage has a sub-ADC. The function of the sub ADC is to convert an analog input to a digital output. A general ADC includes several comparators with specific threshold voltages and an encoder.

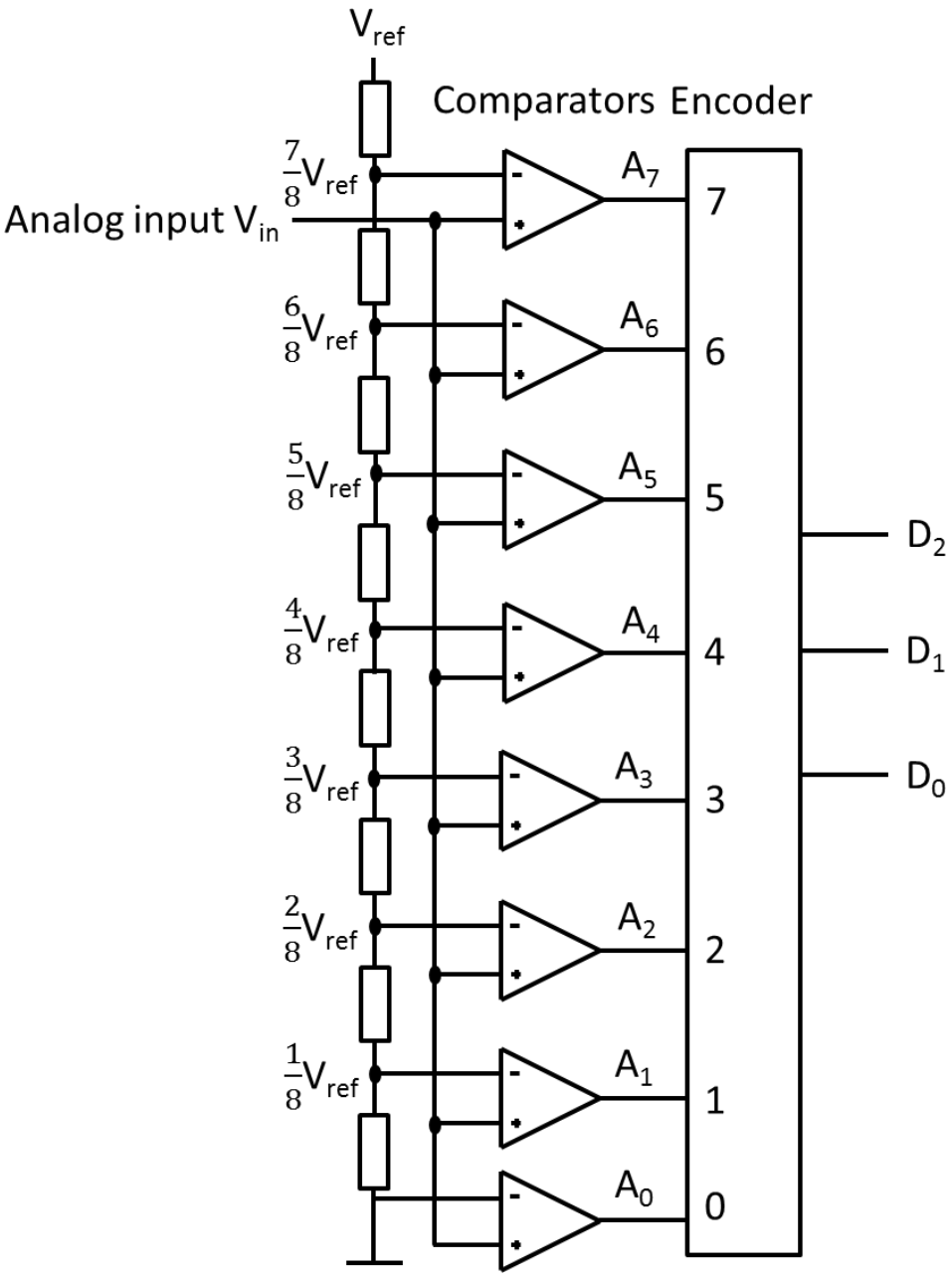


Figure 4-2. Modelled sub-ADC

In order to demonstrate the general configuration of a sub-ACD, a modelled sub-ADC is shown in Figure 4-2. In this figure, a voltage divider is composed of eight same value resistors. This voltage divider provides eight threshold voltages for the comparators. The threshold voltages are: 0, $V_{ref}/8$, $2V_{ref}/8$, $3V_{ref}/8$, $4V_{ref}/8$, $5V_{ref}/8$, $6V_{ref}/8$, and $7V_{ref}/8$. These threshold voltages are applied to the negative input sides of the comparators. In order to compare the analog input with the threshold voltages, the analog input is applied to the positive input sides of the comparators. When the input voltage is higher than a threshold voltage of a comparator, the output of the comparator is 1; otherwise, the output of the comparator is 0. The direct output of the comparator is called the temperature code. For ease of use, an encoder is used to transfer the temperature codes to the binary digital output codes. Table 4-1 shows the relationships between temperature codes and binary digital output codes.

Table 4-1. The relationships between temperature codes and binary digital output codes

Analog input voltage	Temperature codes	Binary digital output codes
V_{in}	$A_7\ A_6\ A_5\ A_4\ A_3\ A_2\ A_1\ A_0$	$D_2\ D_1\ D_0$
$0 \sim V_{ref}/8$	0 0 0 0 0 0 0 1	0 0 0
$V_{ref}/8 \sim 2V_{ref}/8$	0 0 0 0 0 0 1 1	0 0 1
$2V_{ref}/8 \sim 3V_{ref}/8$	0 0 0 0 0 1 1 1	0 1 0
$3V_{ref}/8 \sim 4V_{ref}/8$	0 0 0 0 1 1 1 1	0 1 1
$4V_{ref}/8 \sim 5V_{ref}/8$	0 0 0 1 1 1 1 1	1 0 0
$5V_{ref}/8 \sim 6V_{ref}/8$	0 0 1 1 1 1 1 1	1 0 1
$6V_{ref}/8 \sim 7V_{ref}/8$	0 1 1 1 1 1 1 1	1 1 0
$7V_{ref}/8 \sim 8V_{ref}/8$	1 1 1 1 1 1 1 1	1 1 1

The sub-ADC designed in this research is similar to the general sub-ADC described above. In this research, the sub-ADC also includes several dynamic comparators followed by an encoder. However, in contrast to the threshold voltages provided by the output of the voltage divider, the dynamic comparators in the proposed sub-ADC have built-in threshold voltages. Like the general sub-ADC described above, the encoder for the proposed sub-ADC, is used to transfer the temperature codes to the binary codes.

4.1.1. Dynamic comparator configuration

The static latched comparator with a pre-amplifier has both high accuracy and low comparator offset. However, this kind comparator consumes more power and occupies more chip area than the low accuracy dynamic comparator. Since a lot of comparators need to be used in the pipeline ADC, the power consumption and chip area occupation of the comparators significantly affect the total power consumption and chip area occupation. Fortunately, with the digital error correction technique based on redundancy bits, the pipeline ADC lowers the requirement for the comparators, which enables the popular usage of low power consumption dynamic comparators [26]. For example, in 1.5-bit/stage ADC, the ADC can tolerate the comparator offset within $V_{ref}/4$ [27].

As shown in Figure 4-3, the Lewis-Gray comparator is the typical dynamic comparator [28]. When the clock signal V_{latch} is high, the MOS transistors, M1, M2, M3, and M4, work in the linear region. In addition, the parallel transconductance of M1 and M2 is G_a and the parallel transconductance of M3 and M4 is G_b. Therefore, when the MOS transistors, M1, M2, M3, and M4, work in the linear region, the transconductance of G_a and G_b can be calculated by the following equations:

$$G_a = \mu_N C_{ox}[\frac{W_1}{L}(V_{inn} - V_{th}) + \frac{W_2}{L}(V_{refp} - V_{th})], \quad (4.1)$$

$$G_b = \mu_N C_{ox}[\frac{W_4}{L}(V_{inp} - V_{th}) + \frac{W_3}{L}(V_{refn} - V_{th})]. \quad (4.2)$$

The parameters consist of the following where: μ_N represents the mobility of charge carriers; C_{ox} represents oxide capacitance per unit area; W_1, W_2, W_3, and W_4 represent the widths of the MOS transistors M1, M2, M3, and M4; L represents the length of the MOS transistors; V_{inn} represents the input voltage of the negative input side; V_{refp} represents the reference voltage of the positive reference side; V_{inp} represents the input voltage of the positive input side; V_{refn} represents the reference voltage of the negative reference side; and V_{th} represents the threshold voltage of the MOS transistors.

In Equation (4.1) and Equation (4.2), W_1 is equal to W_4 and W_2 is equal to W_3. Therefore, in this situation, the threshold voltage can be calculated if Equation (4.1) and Equation (4.2) are equal. The threshold voltage is

$$V_{inp} - V_{inn} = \frac{W_1}{W_3}(V_{refp} - V_{refn}). \quad (4.3)$$

As can be seen, the threshold voltage can be adjusted through changing the ratio of W_1/W_3. Therefore, the threshold voltage of this type of comparator has the benefit of linear changeability.

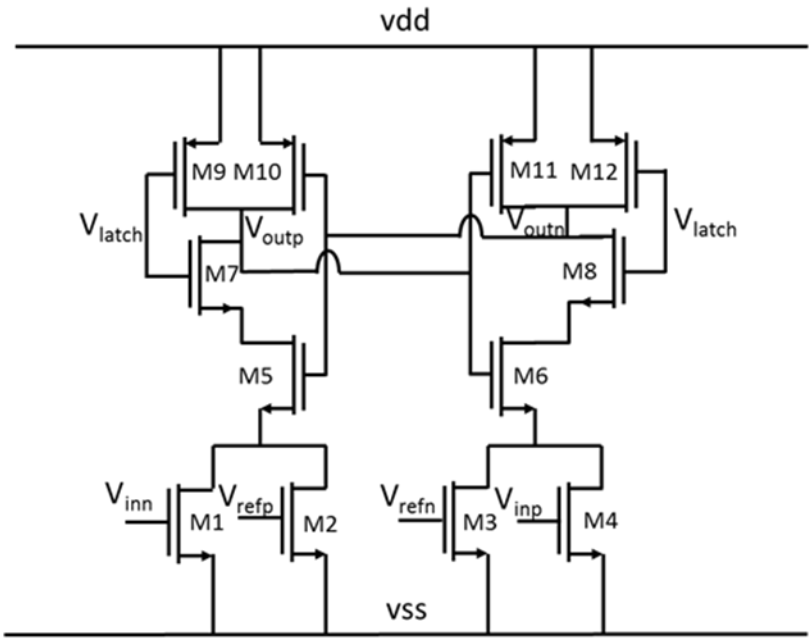


Figure 4-3. Lewis-Gray comparator

Figure 4-4 shows an alternative dynamic comparator, which called the differential pair comparator [20]. The major difference between the differential pair comparator and the Lewis-Gray comparator is that the MOS transistors, M1, M2, M3, and M4, work in saturation region in the differential pair comparator rather than in the linear region in the Lewis-Gray comparator. Therefore, the differential pair comparator has the advantage of being less sensitive to transistor mismatches. In this configuration, the current flowing through M1, M2, M3, and M4 are I_1, I_2, I_3, and I_4, respectively. Also, $V_{in}=V_{inp}-V_{inn}$, $V_{ref}=V_{refp}-V_{refn}$, $W_1=W_4$, and $W_2=W_3$, gives

$$I_4 - I_1 = \frac{1}{2}\mu_N C_{ox}\frac{W_1}{L}V_{in}\sqrt{\frac{2I_a}{\frac{1}{2}\mu_N C_{ox}\frac{W_1}{L}} - V_{in}^2}, \tag{4.4}$$

$$I_2 - I_3 = \frac{1}{2}\mu_N C_{ox}\frac{W_3}{L}V_{in}\sqrt{\frac{2I_b}{\frac{1}{2}\mu_N C_{ox}\frac{W_3}{L}} - V_{ref}^2}\,, \tag{4.5}$$

where I_a and I_b are the currents flowing through the MOS transistors M_7 and M_8, respectively. Therefore, in this situation, the threshold voltage can be calculated if Equation (4.4) and Equation (4.5) are equal. However, in this case, the threshold voltage is not linearly changeable with the width ratio of the MOS transistor. Hence, this method increases the design effort.

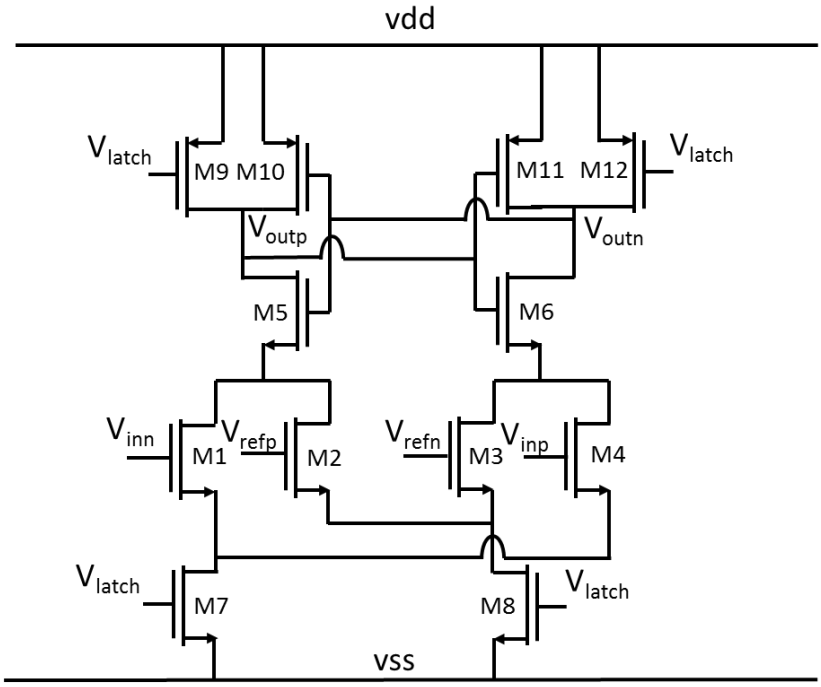


Figure 4-4. Differential pair comparator

In the pipeline ADC, the comparators are required to provide a variety of threshold voltages. Since the threshold voltages of Lewis-Gray comparators can be changed linearly by adjusting the width ratio of the MOS transistors, a significant amount of design effort can be reduced using the Lewis-Gray comparator. Although the Lewis-Gray comparator is sensitive to the transistor

mismatches, the proposed domain-extension digital error correction technique can correct the comparator offset results from the sensitivity.

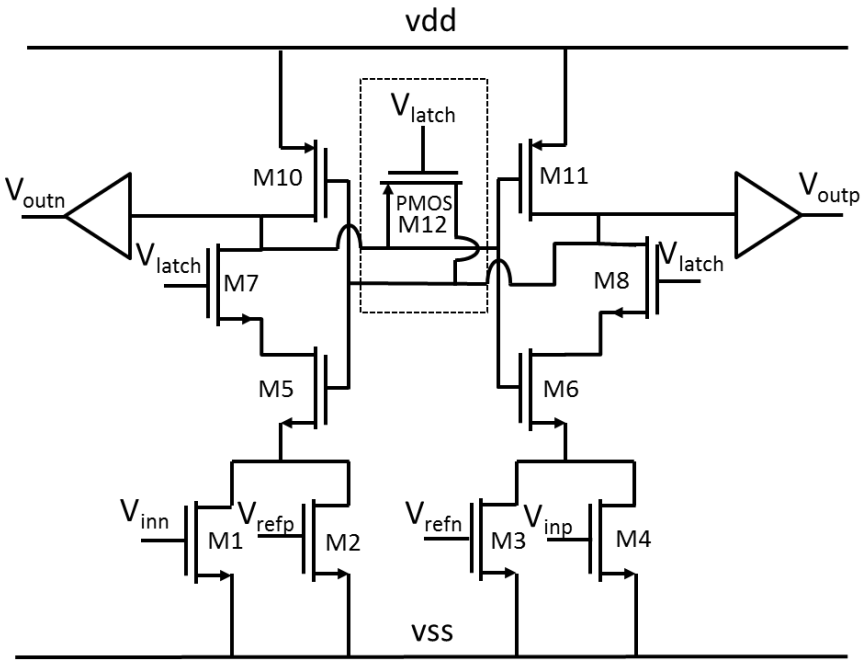


Figure 4-5. Proposed dynamic comparator

Figure 4-5 shows the dynamic comparator used in the proposed ADC. This dynamic comparator retains the benefits of the traditional Lewis-Gray configuration, which are its zero static power consumption and its linearly changeable threshold voltage [20]. In this proposed configuration, two switches in the Lewis-Gray configuration are replaced by one PMOS switch, which reduces power consumption, as shown in the dashed area in Figure 4-5. This PMOS switch is used to reset the comparator after comparison. In order to reduce the loading effect and the noise from the periphery circuit, two buffers are added to the positive and negative outputs of the comparator. Thus, this configuration reduces noise as compared to the Lewis-Gray configuration. Figure 4-6 shows the setup of the dynamic comparator simulation. In this simulation, both negative and positive inputs are ramp signals. Since the configuration is fully differential, input voltage

equals positive input voltage minus negative input voltage. Also, in this simulation, both negative and positive reference voltages are DC signals. Again, since the configuration is fully differential, reference voltage equals positive reference voltage minus negative input voltage. The parameter setup of the comparator simulation is shown in Table 4-2. Moreover, Figure 4-7 shows the simulation results of the dynamic comparators with various threshold voltages. In Figure 4-7, the threshold voltages of the dynamic comparators are shown in the upper left corner.

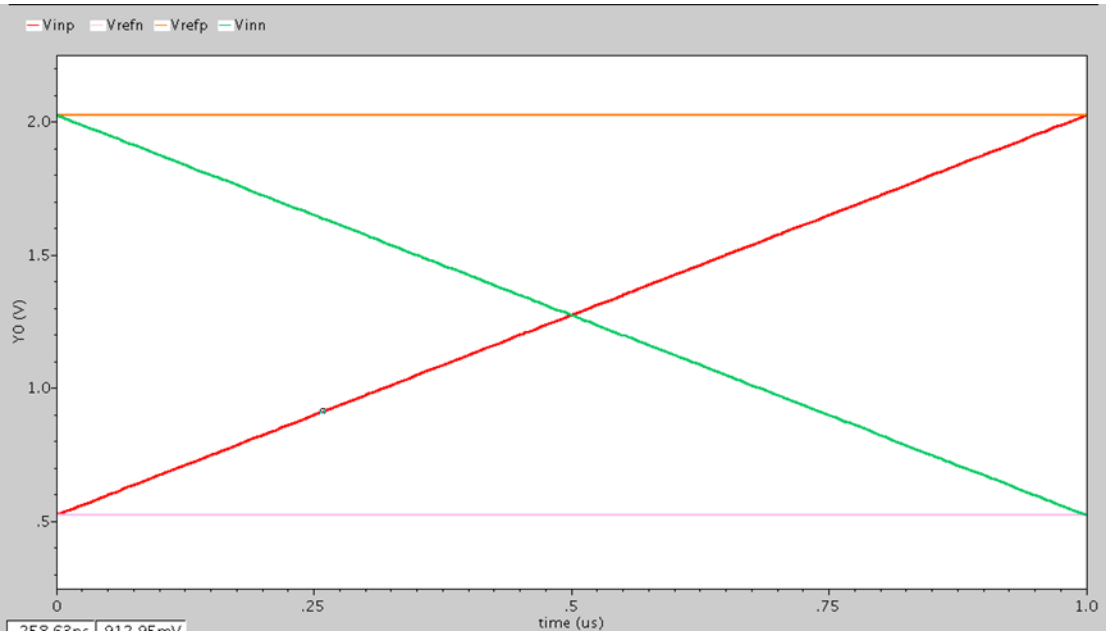


Figure 4-6. Setup of the dynamic comparator simulation

Table 4-2. Parameter setup of the dynamic comparator simulation

Name	Voltage
$V_{ref} = V_{refp}-V_{refn}$	1.5
$V_{in} = V_{inp}-V_{inn}$	-1.5~1.5
V_{refp}	2.025V
V_{refn}	0.525V
V_{inp}	0.525~2.025V
V_{inn}	2.025~0.525V

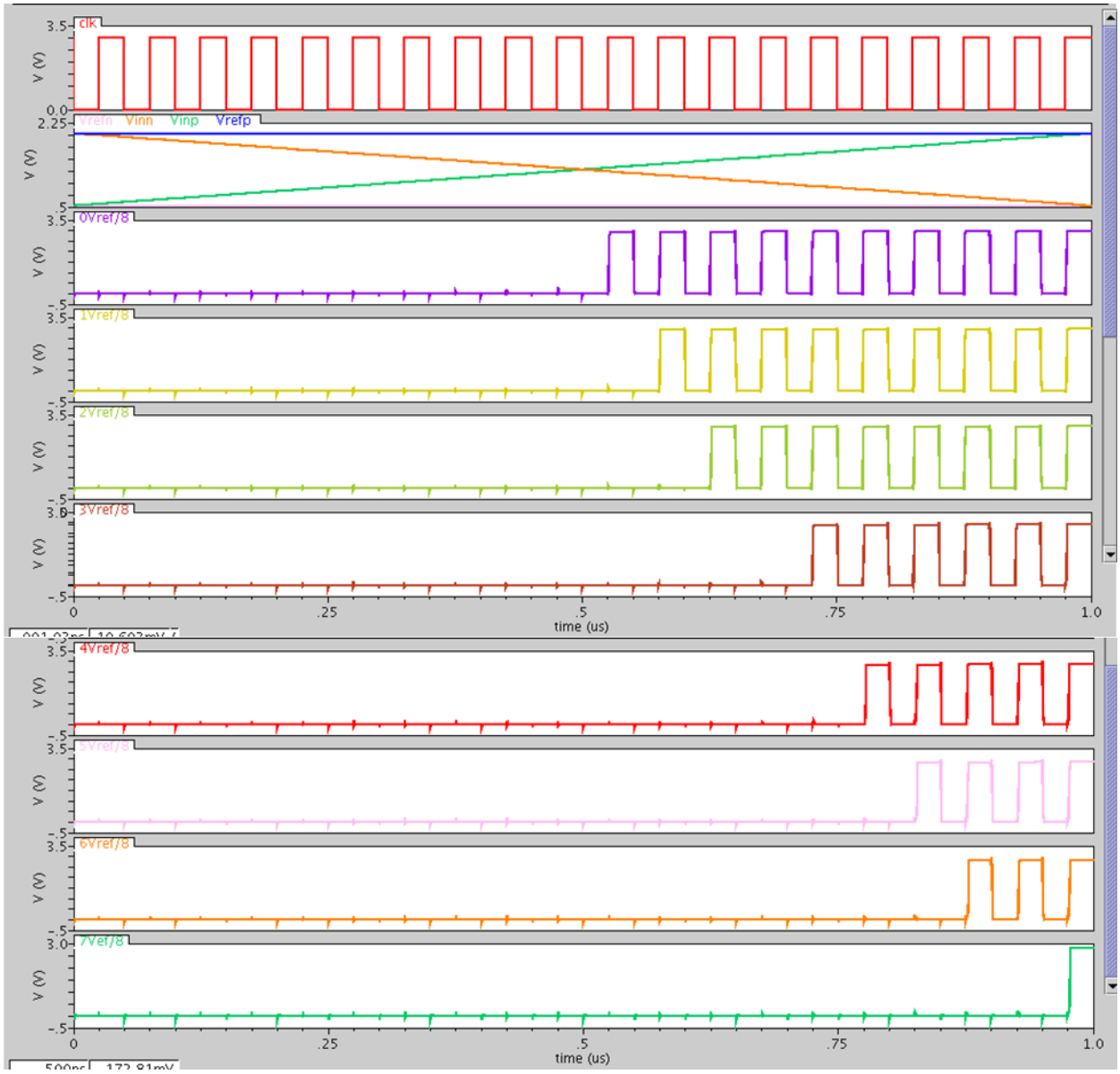


Figure 4-7. Simulation results of the dynamic comparators

4.1.2. Design of the sub-ADC

4.1.2.1. Design of the 1.5-bit/stage sub-ADC

The transfer function of the 1.5-bit/stage ADC is calculated by

$$V_{\text{out}} = \begin{cases} 2V_{in} + 1V_{\text{ref}}, & \text{D=00} \\ 2V_{in} + 0V_{\text{ref}}, & \text{D=01} \\ 2V_{in} - 1V_{\text{ref}}, & \text{D=10} \end{cases}, \tag{4.6}$$

where V_{out} is the output voltage, V_{in} is the input voltage, V_{ref} is the reference voltage, and D is the digital output the sub-ADC. As analysed before, the direct output of the comparators are the temperature codes rather than the required digital output codes. The relationships between temperature codes and required binary digital output codes are shown in Table 4-3.

Table 4-3. The relationships between temperature codes and binary digital output codes

Analog input voltage	Temperature codes	Digital output codes
V_{in}	A_1 A_0	D_1 D_0
$-V_{ref} \sim -V_{ref}/4$	0 0	0 0
$-V_{ref}/4 \sim V_{ref}/4$	0 1	0 1
$V_{ref}/4 \sim V_{ref}$	1 1	1 0

Therefore, the transfer functions between the temperature codes and the digital output codes are as follows:

$$D_1 = A_1, \tag{4.7}$$

$$D_0 = \overline{A_1} \cdot A_0 = \overline{(A_1 + \overline{A_0})} \cdot \tag{4.8}$$

According to the transfer function stated above, the sub-ADC can be designed using the dynamic comparators and the encoder. In the sub-ADC simulation, the setups of the input and reference signal are the same as in Table 4-2. The simulation results of digital output codes D_1 and D_0 are shown in Figure 4-8. As shown in Figure 4-8, when the input ranges from $-V_{ref}$ to $-V_{ref}/4$, the digital output codes are D_1D_0=00; when the input ranges from $-V_{ref}/4$ to $V_{ref}/4$, the digital output codes are D_1D_0=01; and when the input ranges from $V_{ref}/4$ to V_{ref}, the digital output codes are D_1D_0=10.

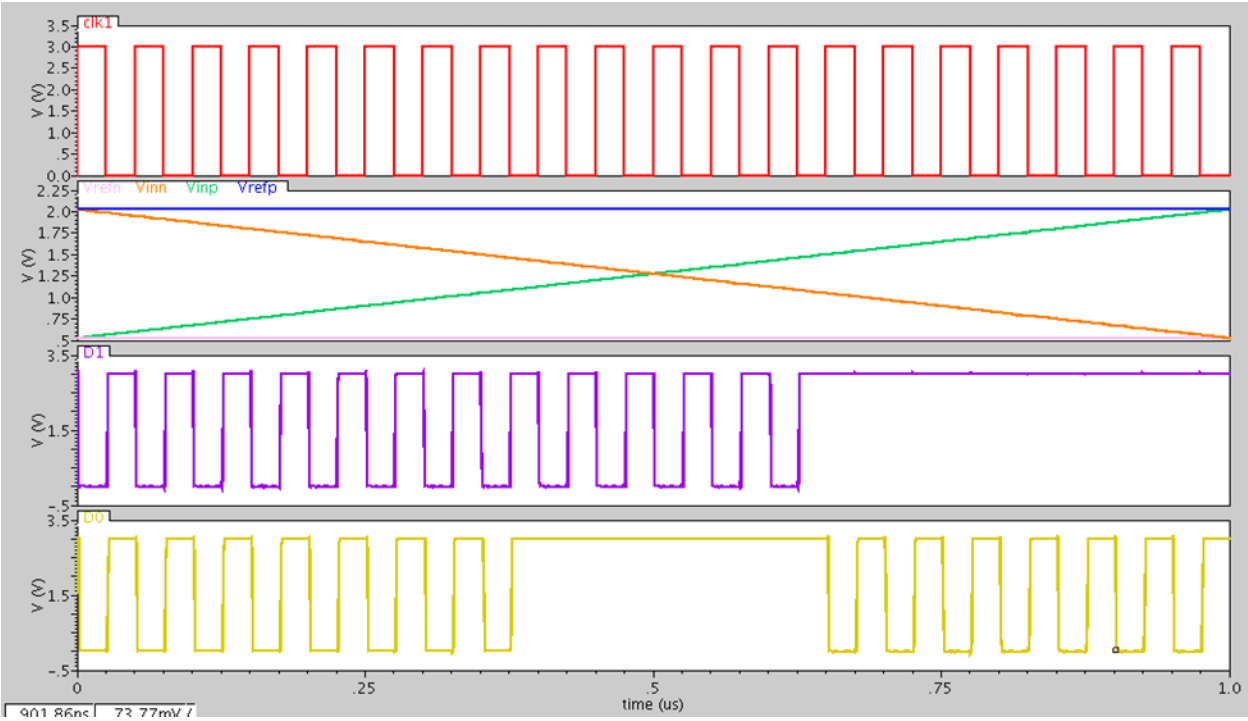


Figure 4-8. Simulation results of the 1.5-bit/stage sub-ADC

4.1.2.2. Design of the 2.5-bit/stage sub-ADC

The transfer function of the domain-extended 2.5-bit/stage ADC is calculated by

$$V_{\text{out}} = \begin{cases} 4V_{in} + 4V_{\text{ref}}, & \text{D=0000} \\ 4V_{in} + 3V_{\text{ref}}, & \text{D=0001} \\ 4V_{in} + 2V_{\text{ref}}, & \text{D=0010} \\ 4V_{in} + 1V_{\text{ref}}, & \text{D=0011} \\ 4V_{in} + 0V_{\text{ref}}, & \text{D=0100} \\ 4V_{in} - 1V_{\text{ref}}, & \text{D=0101} \\ 4V_{in} - 2V_{\text{ref}}, & \text{D=0110} \\ 4V_{in} - 3V_{\text{ref}}, & \text{D=0111} \\ 4V_{in} - 4V_{\text{ref}}, & \text{D=1000} \end{cases} \tag{4.9}$$

where V_{out} is the output voltage, V_{in} is the input voltage, V_{ref} is the reference voltage, and D is the digital output of the sub-ADC. Again, in this example, the direct output of the comparators are the temperature codes rather than the required digital output codes. The relationships between temperature codes and required digital output codes are shown in Table 4-4.

Table 4-4. The relationships between temperature codes and digital output codes

Analog input voltage	Temperature code	Digital output code
V_{in}	A_7 A_6 A_5 A_4 A_3 A_2 A_1 A_0	D_3 D_2 D_1 D_0
$<-7V_{ref}/8$	0 0 0 0 0 0 0 0	0 0 0 0
$-7V_{ref}/8 \sim -5V_{ref}/8$	0 0 0 0 0 0 0 1	0 0 0 1
$-5V_{ref}/8 \sim -3V_{ref}/8$	0 0 0 0 0 0 1 1	0 0 1 0
$-3V_{ref}/8 \sim -V_{ref}/8$	0 0 0 0 0 1 1 1	0 0 1 1
$-V_{ref}/8 \sim V_{ref}/8$	0 0 0 0 1 1 1 1	0 1 0 0
$V_{ref}/8 \sim 3V_{ref}/8$	0 0 0 1 1 1 1 1	0 1 0 1
$3V_{ref}/8 \sim 5V_{ref}/8$	0 0 1 1 1 1 1 1	0 1 1 0
$5V_{ref}/8 \sim 7V_{ref}/8$	0 1 1 1 1 1 1 1	0 1 1 1
$>7V_{ref}/8$	1 1 1 1 1 1 1 1	1 0 0 0

Therefore, the transfer functions between the temperature codes and the digital output codes are as follows:

$$D_3 = A_7, \tag{4.10}$$

$$D_2 = \overline{(A_7 \oplus \overline{A_3})}, \tag{4.11}$$

$$D_1 = \overline{(A_7 \oplus \overline{A_5})} + \overline{(A_3 \oplus \overline{A_1})}, \tag{4.12}$$

$$D_0 = \overline{(A_7 \oplus \overline{A_6})} + \overline{(A_5 \oplus \overline{A_4})} + \overline{(A_3 \oplus \overline{A_2})} + \overline{(A_1 \oplus \overline{A_0})}. \tag{4.13}$$

According to the transfer function stated above, the sub-ADC can be designed using the dynamic comparators and the encoder. As before, in the sub-ADC simulation, the setups of the

input and reference signals are the same as Table 4-2. The simulation results of digital output codes D_3, D_2, D_1, and D_0 are shown in Figure 4-9.

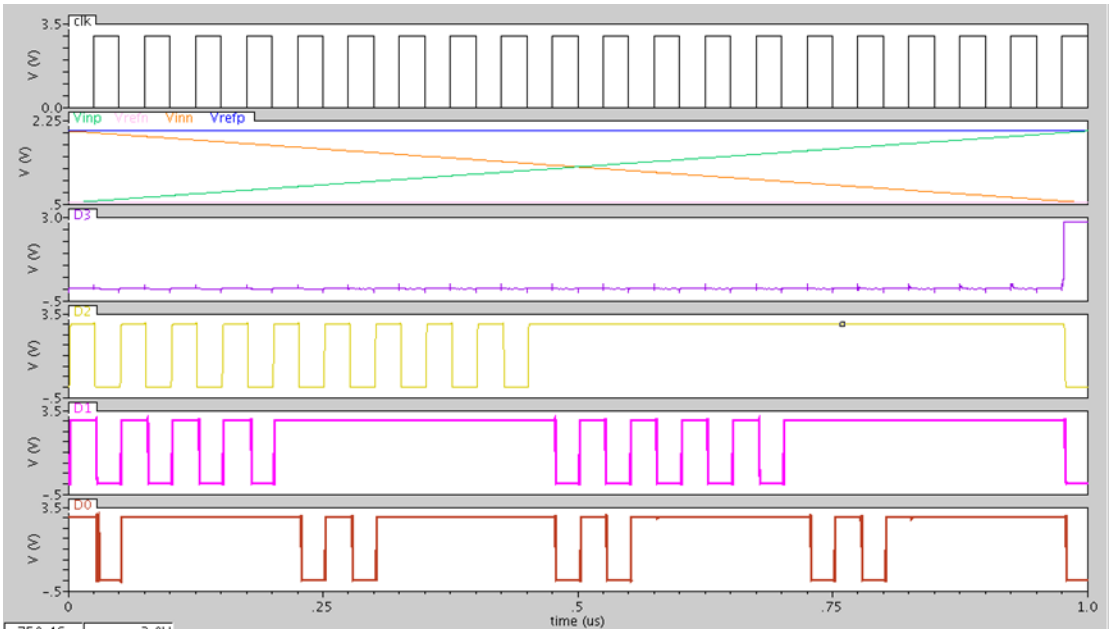


Figure 4-9. Simulation results of the 2.5-bit/stage sub-ADC

4.2. MDAC configuration

Figure 4-10 shows the MDAC configuration. In this configuration, Φ_1 and Φ_2 are two non-overlap clock signals. As shown in Figure 4-11, when Φ_1 is high, the MDAC works in the sample phase. During this time, input is simultaneously sampled on all capacitors C_f, C_{s1}, C_{s2},…, and C_{sj}. In contrast, as shown in Figure 4-12, when Φ_2 is high, the MDAC works in the amplification phase. During this time, the residue, which is generated by the input signal minus the output of sub-DAC, is amplified. As stated in Chapter 3, in order to eliminate the charge injection effect, the fall-edge of the clock signal Φ_{1p} arrives earlier than the fall-edge of phase Φ_1. The neighbouring two pipeline stages have opposite clock signals. For example, if a stage works in the sample phase, then the pipeline stages before and after this stage will work in the amplification phase.

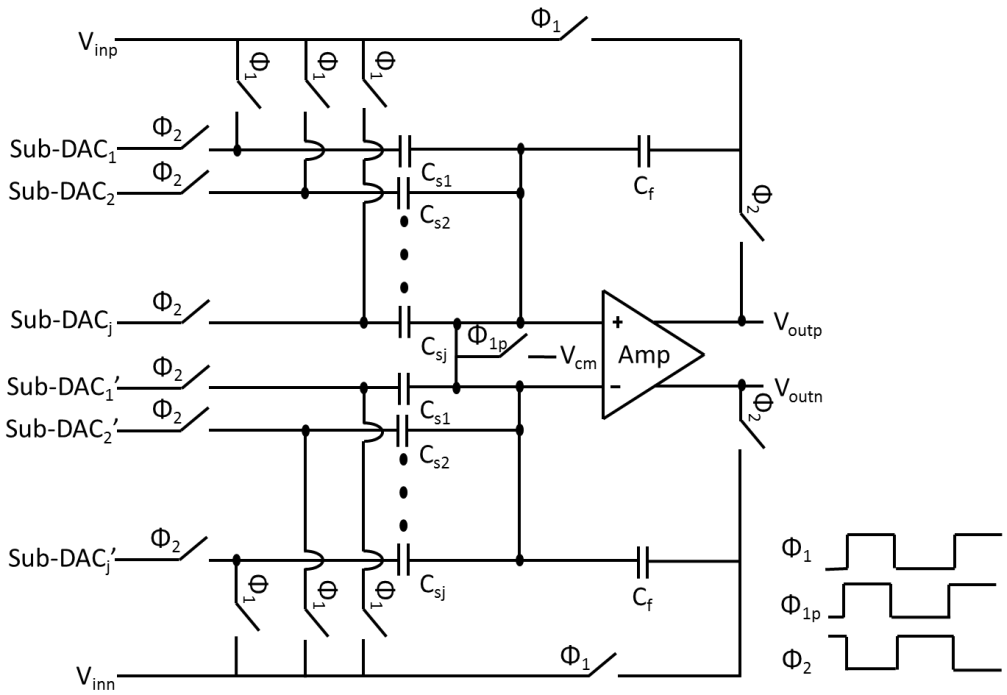


Figure 4-10. MDAC configuration

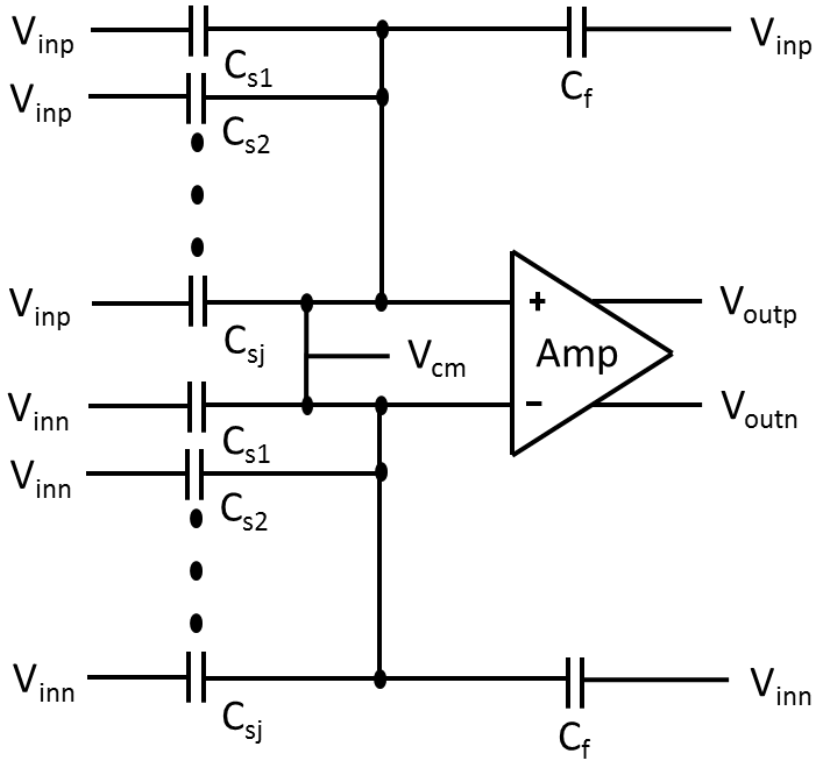


Figure 4-11. MDAC configuration in the sample phase

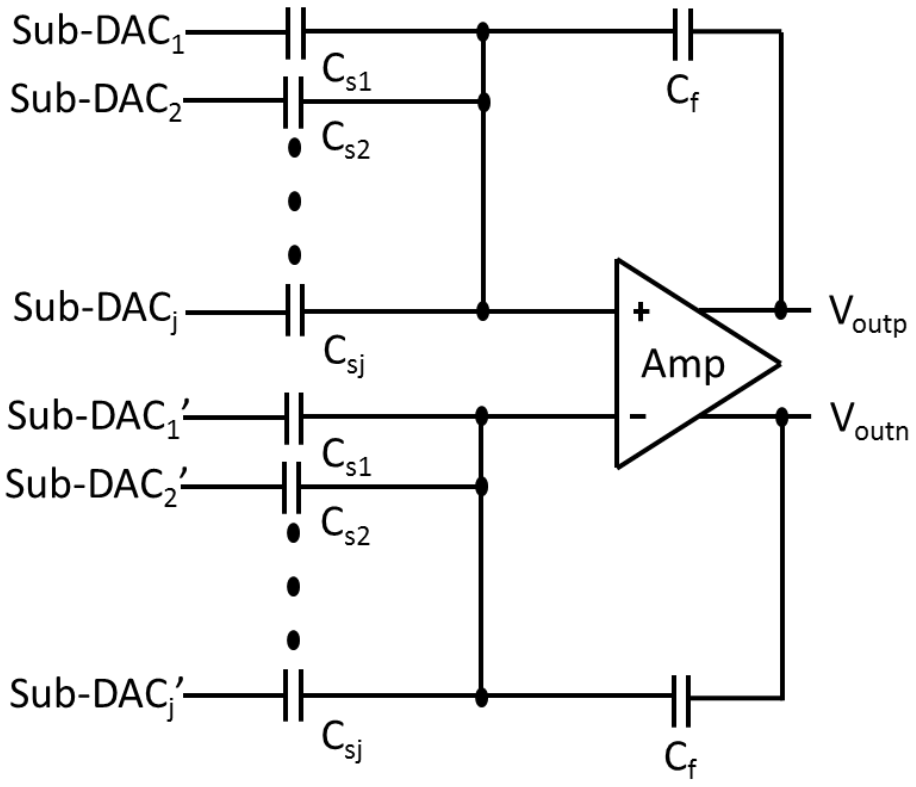


Figure 4-12. MDAC configuration in the amplification phase

When the clock signal Φ_1 is high, as in Figure 4-10, the resulting MDAC configuration is shown in Figure 4-11. During this time, all the capacitors of the MDAC sample the input signal. Since the configuration of the MDAC is fully differential, we can calculate the input voltage, the reference voltage, and the output voltage by

$$V_{in} = V_{inp} - V_{inn}, \tag{4.14}$$

$$V_{ref} = V_{refp} - V_{refn}, \tag{4.15}$$

$$V_{out} = V_{outp} - V_{outn}. \tag{4.16}$$

Therefore, the charge stored in the capacitor C_{s1}, C_{s2}, … C_{sj}, and C_f are shown in the following equations:

$$Q_{s1} = C_{s1} \cdot V_{in}, \tag{4.17}$$

$$Q_{s2} = C_{s2} \cdot V_{in}, \tag{4.18}$$

$$Q_{sj} = C_{sj} \cdot V_{in}, \tag{4.19}$$

$$Q_f = C_f \cdot V_{in} \cdot \tag{4.20}$$

When the clock signal Φ_2 is high, as in Figure 4-10, the resulting MDAC configuration is shown in Figure 4-12. During this time, the capacitor C_f is the feedback capacitor and other capacitors, Cs1, Cs2, …, and Csj, connect to the output of the sub-DAC. Both negative and positive input sides of the amplifier are mutually virtual ground. For example, when all the non-feedback capacitors, which located at the positive input side of the amplifier, connect to the V_{refp} and when all the non-feedback capacitors, which located at the negative input side of the amplifier, connect to the V_{refn}, the charge is stored in the capacitors C_{s1}, C_{s2},…, C_{sj}, and C_f, and can be calculated using the following equations:

$$Q_{s1} = C_{s1} \cdot V_{ref}, \tag{4.21}$$

$$Q_{s2} = C_{s2} \cdot V_{ref}, \tag{4.22}$$

$$Q_{sj} = C_{sj} \cdot V_{ref}, \tag{4.23}$$

$$Q_f = C_f \cdot V_{out} \cdot \tag{4.24}$$

Since the charge is stored in the capacitors must be conserved, we can derive the following equation:

$$\begin{aligned} &C_{s1} \cdot V_{in} + C_{s2} \cdot V_{in} + \ldots C_{sj} \cdot V_{in} + C_f \cdot V_{in} = \\ &C_{s1} \cdot V_{ref} + C_{s2} \cdot V_{ref} + \ldots C_{sj} \cdot V_{ref} + C_f \cdot V_{out} \end{aligned} \cdot \tag{4.25}$$

According to Equation (4.25), the output voltage V_{out} can be represented by

$$V_{out} = \left(1 + \frac{C_{s1} + C_{s2} + \ldots C_{sj}}{C_f}\right) \cdot V_{in} - \frac{C_{s1} + C_{s2} + \ldots C_{sj}}{C_f} \cdot V_{ref} \cdot \tag{4.26}$$

For a 1.5-bit/stage ADC, the number of non-feedback capacitor j equals 1 and the capacitance of the capacitor C_{s1} equals the capacitance of the capacitor C_f. Therefore, Equation (4.26) is simplified to

$$V_{out} = 2 \cdot V_{in} - V_{ref} \cdot \tag{4.27}$$

For a 2.5-bit/stage ADC, there are three non-feedback capacitors and one feedback capacitor. All the capacitors, including non-feedback and feedback, have the same capacitance. Therefore, Equation 4.26 becomes

$$V_{out} = 4 \cdot V_{in} - 3 \cdot V_{ref} \cdot \tag{4.28}$$

Also, for both the 1.5-bit/stage and the 2.5-bit/stage converters, with the different sub-DAC outputs, the transfer function of the 1.5-bit/stage and the 2.5-bit/stage converters can also be calculated. The detailed configuration of the sub-DAC is discussed in Section 4.2.1 and the detailed configuration of the amplifier is discussed in Section 4.2.2.

4.2.1. Sub-DAC configuration

4.2.1.1. Design of the 1.5-bit/stage sub-DAC

The output of the 1.5-bit sub-DAC has three possible values: $-V_{ref}$, 0, and V_{ref}. The relationships between the digital outputs of sub-ADC and the outputs of sub-DAC is shown in Table 4-5.

Table 4-5. Relationships between the sub-ADC output and the sub-DAC output

Analog input voltage	Sub-ADC output	sub-DAC output
V_{in}	D_1 D_0	V_{dac}
$-V_{ref} \sim -V_{ref}/4$	0 0	$-V_{ref}$
$-V_{ref}/4 \sim V_{ref}/4$	0 1	0
$V_{ref}/4 \sim V_{ref}$	1 0	V_{ref}

As stated in Chapter 3, the CFCS technique is used to reduce the capacitor mismatches. Therefore, the implementation of the 1.5-bit/stage's transfer function becomes

$$V_{out} = \begin{cases} \frac{C_5+C_6}{C_5}V_{in} + \frac{C_6}{C_5}V_{ref}, & D=00 \\ \frac{C_5+C_6}{C_6}V_{in}, & D=01 \\ \frac{C_5+C_6}{C_5}V_{in} - \frac{C_6}{C_5}V_{ref}, & D=10 \end{cases}, \tag{4.29}$$

where C_5 and C_6 are the sample-and-hold capacitors.

As can be seen in Equation (4.29), the sample-and-hold capacitors C_5 and C_6 are alternatively used as the feedback capacitor. When the digital output of the sub-ADC are 00 and 10, the sample-and-hold capacitor C_5 is the feedback capacitor; otherwise, when the digital output of the sub-ADC is 01, C_6 is the feedback capacitor. Table 4-6 shows the connections of the capacitors C_5 and C_6.

Table 4-6. Connections of the sample-and-hold capacitors during the amplification phase

Analog input voltage	Sub-ADC output	Capacitor	Capacitor
V_{in}	D_1 D_0	C_5	C_6
$-V_{ref} \sim -V_{ref}/4$	0 0	V_{out}	$-V_{ref}$
$-V_{ref}/4 \sim V_{ref}/4$	0 1	0	V_{out}
$V_{ref}/4 \sim V_{ref}$	1 0	V_{out}	V_{ref}

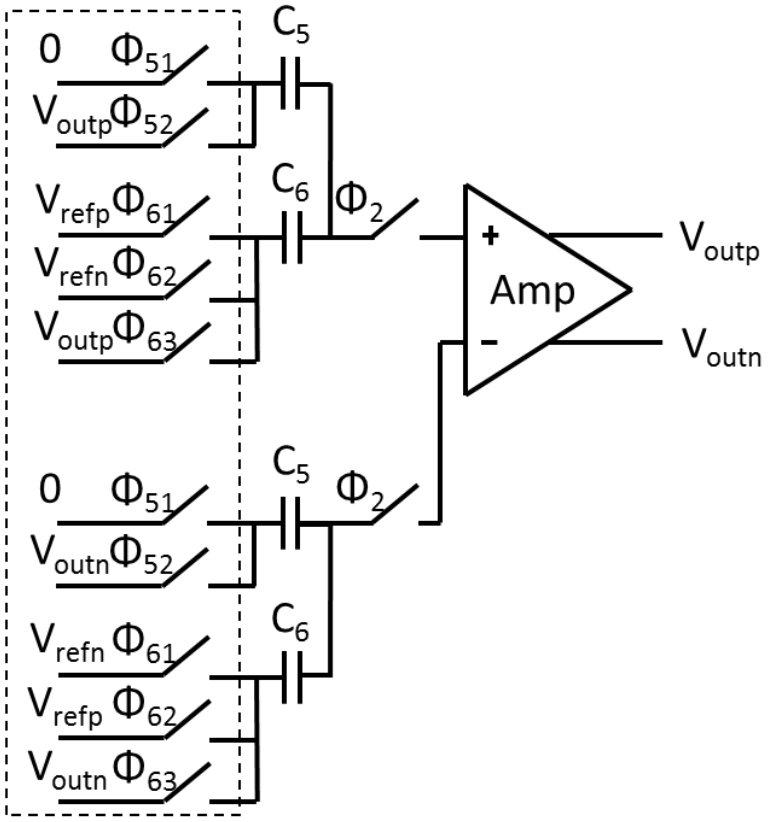


Figure 4-13. Circuit implementation of the 1.5-bit sub-DAC

The detailed circuit implementation of the sub-DAC based on the CFCS technique is shown in the dashed box of Figure 4-13. As also shown in Figure 4-13, the configuration of the 1.5-bit/stage DAC is fully differential. During the amplification phase, each sample-and-hold capacitor

has four connection possibilities: V_{refp}, 0, V_{refn}, and $V_{outp(n)}$. When a capacitor is connected to the $V_{outp(n)}$, it indicates that this capacitor is the feedback capacitor. Since the circuit configuration is fully differential, V_{ref} and $-V_{ref}$ in Table 4-6 can be calculated using

$$V_{ref} = V_{refp} - V_{refn}, \tag{4.30}$$

$$-V_{ref} = V_{refn} - V_{refp}. \tag{4.31}$$

The connections of the capacitors are controlled by switches. The on/off state of the switches depend on the relationships between the sub-ADC digital output codes and the capacitor connection requirements. Therefore, the switches' control signals can be calculated through the following equations:

$$\Phi_{51} = D_0 \cdot \Phi_2, \tag{4.32}$$

$$\Phi_{52} = \overline{D_0} \cdot \Phi_2, \tag{4.33}$$

$$\Phi_{61} = D_1 \cdot \Phi_2, \tag{4.34}$$

$$\Phi_{62} = \overline{D_1 \oplus D_0} \cdot \Phi_2, \tag{4.35}$$

$$\Phi_{63} = D_0 \cdot \Phi_2. \tag{4.36}$$

4.2.1.2. Design of the 2.5-bit/stage sub-DAC

The output of the domain extended 2.5-bit sub-DAC has nine possible values: $-4V_{ref}$, $-3V_{ref}$, $-2V_{ref}$, $-V_{ref}$, 0, V_{ref}, $2V_{ref}$, $3V_{ref}$, and $4V_{ref}$. The relationships between the digital outputs of sub-ADC and the outputs of sub-DAC are shown in Table 4-7.

Table 4-7. The relationships between the sub-ADC outputs and the sub-DAC outputs

Analog input voltage	Sub-ADC outputs	Sub-DAC outputs
V_{in}	D_3 D_2 D_1 D_0	V_{dac}
$< -7V_{ref}/8$	0 0 0 0	$-4V_{ref}$
$-7V_{ref}/8 \sim -5V_{ref}/8$	0 0 0 1	$-3V_{ref}$
$-5V_{ref}/8 \sim -3V_{ref}/8$	0 0 1 0	$-2V_{ref}$
$-3V_{ref}/8 \sim -V_{ref}/8$	0 0 1 1	$-V_{ref}$
$-V_{ref}/8 \sim V_{ref}/8$	0 1 0 0	0
$1V_{ref}/8 \sim 3V_{ref}/8$	0 1 0 1	V_{ref}
$3V_{ref}/8 \sim 5V_{ref}/8$	0 1 1 0	$2V_{ref}$
$5V_{ref}/8 \sim 7V_{ref}/8$	0 1 1 1	$3V_{ref}$
$>7V_{ref}/8$	1 0 0 0	$4V_{ref}$

As stated in Chapter 3, the domain-extended algorithm tolerates more comparator offsets. Thus, according to Equation (3-8) and Equation (3-9), when the required sub-DAC output is $4V_{ref}$ or $-4V_{ref}$, the power source $V_{dd}=2V_{ref}$ is connected to the capacitor in order to provide an extra reference voltage. Therefore, the implementation of the 2.5-bit/stage's transfer function becomes:

$$V_{\text{out}} = \begin{cases} \frac{C_1+C_2+C_3+C_4}{C_1} V_{in} + \left(\frac{C_2}{C_1} V_{dd} + \frac{C_3+C_4}{C_1} V_{ref}\right), & D=0000 \\ \frac{C_1+C_2+C_3+C_4}{C_1} V_{in} + \frac{C_2+C_3+C_4}{C_1} V_{ref}, & D=0001 \\ \frac{C_1+C_2+C_3+C_4}{C_1} V_{in} + \frac{C_2+C_3}{C_1} V_{ref}, & D=0010 \\ \frac{C_1+C_2+C_3+C_4}{C_1} V_{in} + \frac{C_2}{C_1} V_{ref}, & D=0011 \\ \frac{C_1+C_2+C_3+C_4}{C_1} V_{in}, & D=0100 \\ \frac{C_1+C_2+C_3+C_4}{C_1} V_{in} - \frac{C_2}{C_1} V_{ref}, & D=0101 \\ \frac{C_1+C_2+C_3+C_4}{C_1} V_{in} - \frac{C_2+C_3}{C_1} V_{ref}, & D=0110 \\ \frac{C_1+C_2+C_3+C_4}{C_1} V_{in} - \frac{C_2+C_3+C_4}{C_1} V_{ref}, & D=0111 \\ \frac{C_1+C_2+C_3+C_4}{C_1} V_{in} - \left(\frac{C_2}{C_1} V_{dd} + \frac{C_3+C_4}{C_1} V_{ref}\right), & D=1000 \end{cases}, \tag{4.37}$$

where C_1, C_2, C_3, and C_4 are sample-and-hold capacitors.

According to Equation (4.37) then, during the amplification phase the connections of the sample-and-hold capacitors are shown in Table 4-8.

Table 4-8. Connections of the sample-and-hold capacitors during amplification phase

Analog input voltage	Sub-ADC output	Capacitor	Capacitor	Capacitor	Capacitor
V_{in}	D_3 D_2 D_1 D_0	C_1	C_2	C_3	C_4
$< -7V_{ref}/8$	0 0 0 0	V_{out}	$-V_{dd}$	$-V_{ref}$	$-V_{ref}$
$-7V_{ref}/8 \sim -5V_{ref}/8$	0 0 0 1	V_{out}	$-V_{ref}$	$-V_{ref}$	$-V_{ref}$
$-5V_{ref}/8 \sim -3V_{ref}/8$	0 0 1 0	V_{out}	$-V_{ref}$	$-V_{ref}$	0
$-3V_{ref}/8 \sim -V_{ref}/8$	0 0 1 1	V_{out}	$-V_{ref}$	0	0
$-V_{ref}/8 \sim V_{ref}/8$	0 1 0 0	V_{out}	0	0	0
$1V_{ref}/8 \sim 3V_{ref}/8$	0 1 0 1	V_{out}	V_{ref}	0	0
$3V_{ref}/8 \sim 5V_{ref}/8$	0 1 1 0	V_{out}	V_{ref}	V_{ref}	0
$5V_{ref}/8 \sim 7V_{ref}/8$	0 1 1 1	V_{out}	V_{ref}	V_{ref}	V_{ref}
$>7V_{ref}/8$	1 0 0 0	V_{out}	V_{dd}	V_{ref}	V_{ref}

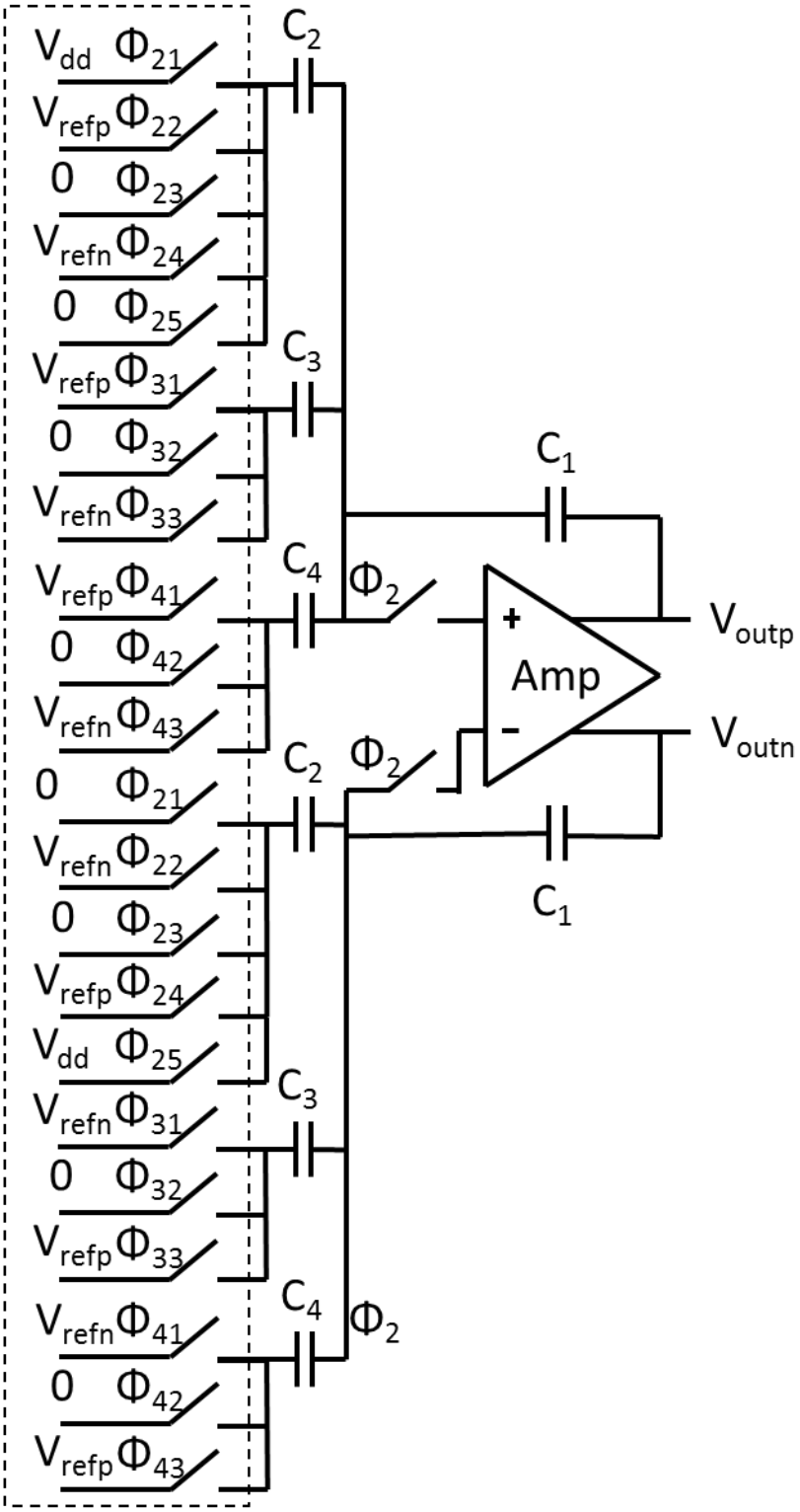


Figure 4-14. Circuit implementation of the 2.5-bit sub-DAC

The detailed circuit implementation of the domain-extended 2.5-bit sub-DAC is shown in the dashed box in Figure 4-14. This configuration of the 2.5-bit/stage ADC is fully differential. During the amplification phase, one side of the non-feedback capacitors C_2, C_3, and C_4 are connected to either negative or positive input side of the amplifier. However, the other side of the non-feedback capacitors is connected to the relevant reference voltage. The non-feedback capacitor C_2 has four connection possibilities: V_{refp}, 0, V_{refn}, and V_{dd}. The other non-feedback capacitors C_3 and C_4 have three connection possibilities: V_{refp}, 0, V_{refn}. In contrast, C_1 only connects to V_{out}; this indicates that capacitor C_1 is the feedback capacitor. Since the circuit configuration is fully differential, the voltages, V_{ref}, $-V_{ref}$, V_{dd}, and $-V_{dd}$, can be obtained through

$$V_{ref} = V_{refp} - V_{refn}, \tag{4.38}$$

$$-V_{ref} = V_{refn} - V_{refp}, \tag{4.39}$$

$$V_{dd} = V_{dd} - 0, \tag{4.40}$$

$$-V_{dd} = 0 - V_{dd}. \tag{4.41}$$

The connections of these capacitors are controlled by switches. The on/off state of the switches depend on the relationship between the sub-ADC digital output code and the capacitor connection requirements. Therefore, the switches' control signals can be calculated through the following equations:

$$\Phi_{21} = A_7 \cdot \Phi_2, \tag{4.42}$$

$$\Phi_{22} = A_4 \cdot \overline{A_7} \cdot \Phi_2, \tag{4.43}$$

$$\Phi_{23} = (A_4 \otimes A_3) \cdot \Phi_2, \tag{4.44}$$

$$\Phi_{24} = \overline{A_3} \cdot A_0 \cdot \Phi_2 , \tag{4.45}$$

$$\Phi_{25} = \overline{A_0} \cdot \Phi_2 , \tag{4.46}$$

$$\Phi_{31} = A_5 \cdot \Phi_2 , \tag{4.47}$$

$$\Phi_{32} = A_2 \cdot \overline{A_5} \cdot \Phi_2 , \tag{4.48}$$

$$\Phi_{33} = \left(\overline{A_2}\right) \cdot \Phi_2 , \tag{4.49}$$

$$\Phi_{41} = A_6 \cdot \Phi_2 , \tag{4.50}$$

$$\Phi_{42} = A_1 \cdot \overline{A_6} \cdot \Phi_2 , \tag{4.51}$$

$$\Phi_{43} = \overline{A_1} \cdot \Phi_2 . \tag{4.52}$$

The digital output of the sub-ADC is converted to an analog signal through the sub-DAC. Residue is then generated by the input signal minus the output of the sub-DAC. Afterwards in the residue gain stage, the residue is increased by a factor of two for the 1.5-bit/stage converter and by a factor of four for the 2.5-bit/stage converter.

4.2.2. Amplifier configuration

It is well known that the amplifier is the crucial constituent part of the switched capacitor MDAC. The detailed configuration of the amplifier is discussed in this section. In a pipeline analog-to-digital converter, the design of the high speed and high accuracy amplifier is very important. Both speed and accuracy criteria for the amplifier are determined by the amplifier's settings. A fast setting speed depends on a large unity gain bandwidth and a high setting accuracy depends on a high DC gain [29]. Furthermore, a stable work status depends on a sufficient phase

margin [30]. Therefore, in this design, the requirement for the amplifier includes: a high DC gain, a large unity gain bandwidth, and a sufficient phase margin.

The performance of the amplifier directly determines the realization of the MDAC functions. Therefore, in this design, a gain-boosted transconductance operational amplifier is adopted. This section is organized as follows: first, the principal of the gain-boosted technique is presented; second, the configuration and consideration of the amplifier used in this design is discussed; and third, the amplifier simulation results are presented.

4.2.2.1. Principal of the gain boosting technique

Figure 4-15 shows a cascode amplifier. In order to increase the output impedance, the amplifier adopts the gain boosting technique. Also, in this amplifier, the MOS transistor M1 is an input device, the MOS transistor M2 is a cascode device, and additional amplifier A1 is a gain boost device. This amplifier also includes a current-voltage feedback system. In this configuration, the MOS transistor M1 works as a feedback resistor. This MOS transistor detects the output current and transfers the current to a voltage. Then, this voltage feeds back to the output current through the additional operational amplifier A_1 and the MOS transistor M2. For example, if one assumes that the gain of the additional amplifier is A_{add}, then according the current-voltage feedback principle, the output impedance is increased by a factor of A_{add} [31]. Therefore, the output impedance is calculated with Equation (4.53) and the DC gain is calculated with Equation (4.54),

$$R_{out} \approx A_{add} \cdot g_{m2} \cdot r_{ds2} \cdot r_{ds1}, \tag{4.53}$$

$$A_{dc} = g_{m1} \cdot R_{out} \approx g_{m1} \cdot A_{add} \cdot g_{m2} \cdot r_{ds2} \cdot r_{ds1}, \tag{4.54}$$

where g_{m1} is the transconductance of the transistor M_1, g_{m2} is the transconductance of the transistor M_2, r_{ds2} is the output resistance of the transistor M_2, and r_{ds1} is the output resistance of the transistor M_1.

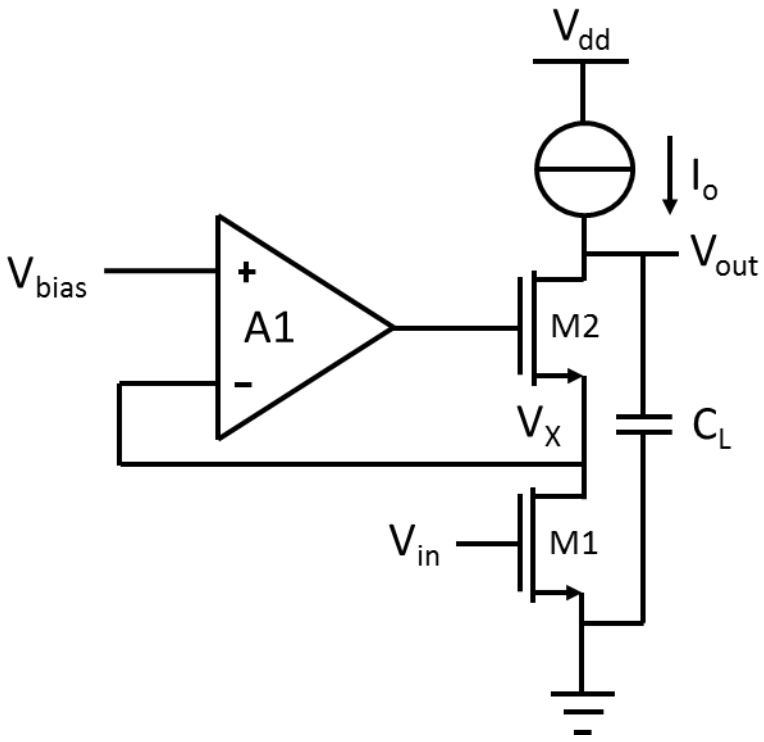


Figure 4-15. Amplifier using gain boosting technique

Through this gain boosting technique, the output resistance as well as the DC gain are increased by a factor of A_{add}. However, the additional amplifier changes the gain boost amplifier's frequency characteristics by the presence of a pole-zero doublet [31]. The pole-zero doublet degrades the setting time because the doublet appears as a slow exponential term in the step response of the amplifier [29]. In order to analyze the doublet introduced by the additional amplifier, the parameters are defined as follows: ω_1 is the dominant pole of the gain boost amplifier, ω_2 is the dominant pole of the additional amplifier, ω_3 is the dominant pole of the operational amplifier without the additional amplifier, ω_4 is the unity gain bandwidth of the additional amplifier, ω_5 is the unite gain bandwidth of the gain boost amplifier, andω_6 is the first non-dominant pole of the gain boost amplifier.

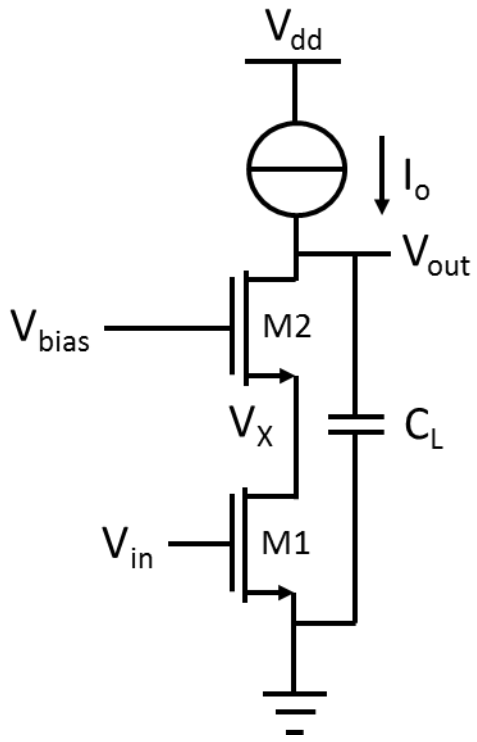


Figure 4-16. Cascode amplifier without using the gain boosting technique

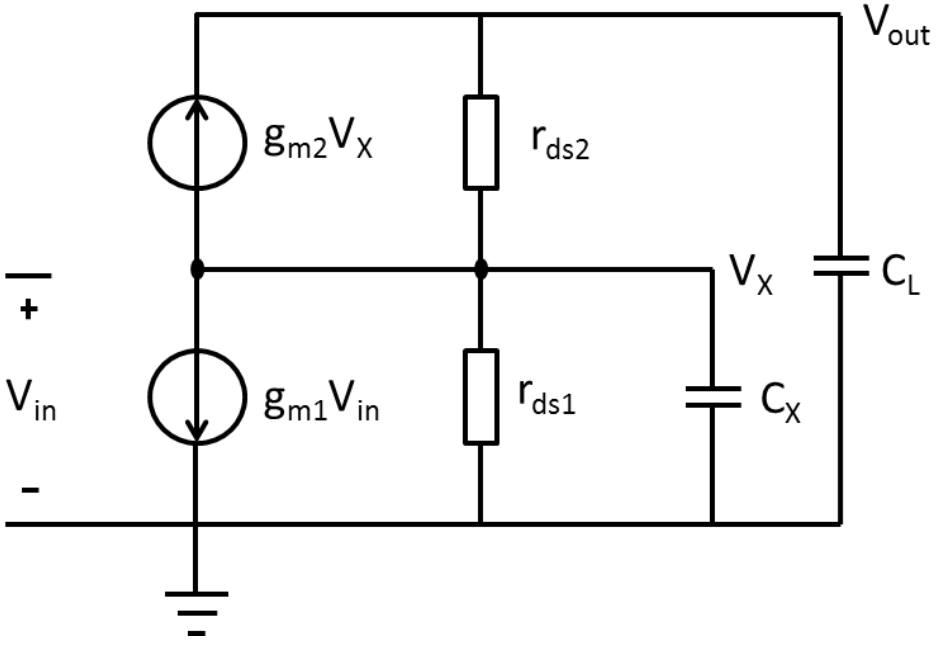


Figure 4-17. Small signal model of the cascode amplifier without using the gain boosting technique

In the process of analysing the effect of the pole-zero doublet introduced by the additional amplifier, the frequency characteristics of the cascode amplifier without the gain boosting technique is discussed first. Figure 4-16 shows a cascode amplifier without the additional amplifier and Figure 4-17 shows the small signal model of this cascode amplifier. The transfer function of the cascode amplifier is derived from the small signal analysis. Therefore, the transfer function is

$$A(s) = \frac{V_{out}(s)}{V_{in}(s)} = -g_{m1} \cdot [R_{out}(s) \| \frac{1}{s \cdot C_L}], \tag{4.55}$$

where g_{m1} is the transconductance of the transistor M_1, C_L is the load capacitance, and R_{out} is the output resistance, which can be calculated by

$$R_{out} = g_{m2} \cdot r_{ds2} \cdot r_{ds1}. \tag{4.56}$$

According to the small signal analysis, the dominant pole of this cascode amplifier, ω_3, locates the output node of the cascode amplifier. The frequency of the dominant pole is determined by the output resistance and load capacitance. Therefore, the frequency of the dominant pole is

$$\omega_3 = \frac{1}{R_{out} C_L}. \tag{4.57}$$

Furthermore, according to the small signal analysis, the first non-dominant pole of the cascode amplifier locates at node V_x. The frequency of the first non-dominant pole is determined by the equivalent resistance $1/g_{m2}$ at node V_x and the equivalent parasitic capacitance C_x at node V_x. Therefore, the frequency ω_6 of the first non-dominant pole is

$$\omega_6 = \frac{g_{m2}}{C_x}. \tag{4.58}$$

In order to have a sufficient phase margin, the first non-dominant pole of the cascode amplifier should be located outside of the unity gain bandwidth through the amplifier's optimization. In this case, the unity gain bandwidth of the cascode amplifier GBW_{main} is calculated by

$$GBW_{main} = \frac{g_{m1}}{2 \cdot \pi \cdot C_L} = \omega_5 \cdot \tag{4.59}$$

After analysing the cascode amplifier without using the gain boosting technique, one is in the position to analyze the gain boost cascode amplifier. The performance of the gain-boosted cascode amplifier is affected by the additional amplifier. The small signal model of the gain-boosted cascode amplifier is shown in Figure 4-18.

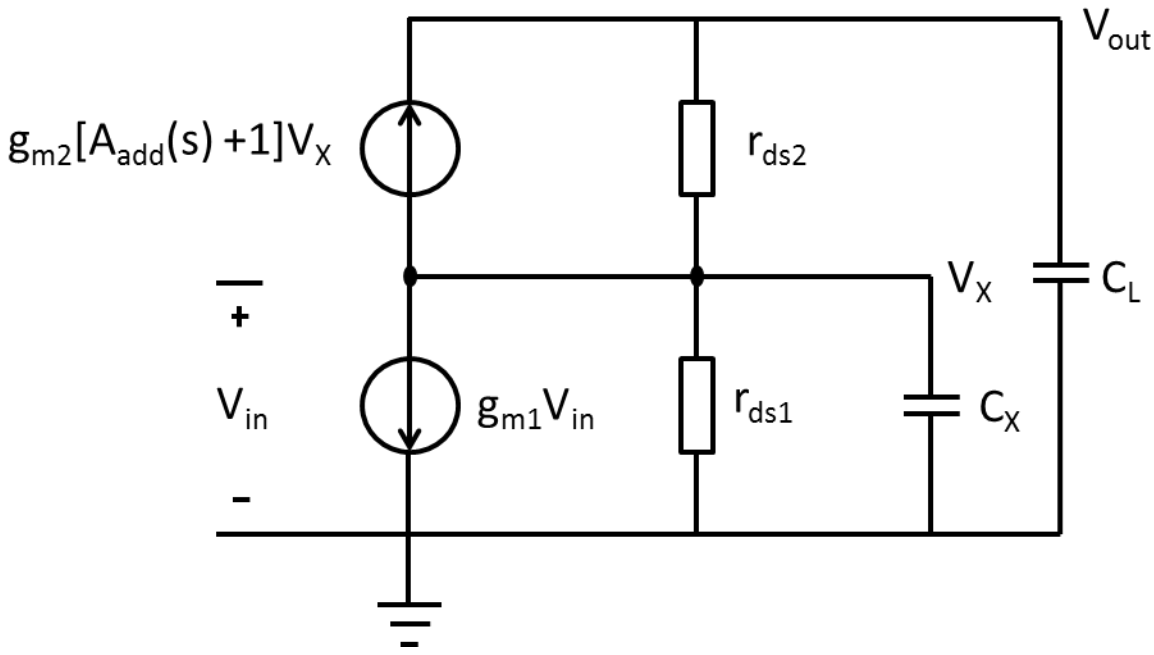


Figure 4-18. Small signal model of the gain boost cascode amplifier

In Figure 4-18, the gain of the additional amplifier $A_{add}(s)$ is a function of frequency. For simplicity, assume that the additional amplifier is a single pole amplifier, then, the transfer function would be

$$A_{add}(s) = \frac{A_{add}}{1 + s/\omega_2}, \tag{4.60}$$

where A_{add} is the DC gain of the additional amplifier and ω_2 is the dominant pole of the additional amplifier. In addition, the unity gain bandwidth of the additional amplifier GBW_{add} is

$$GBW_{add} = \omega_4 = A_{add} \cdot \omega_2 . \tag{4.61}$$

According to the above small signal analysis of the additional amplifier, the output impedance of the gain boost amplifier would be

$$R_{out}(s) = [A_{add}(s)+1]\, g_{m2} \cdot r_{ds2} \cdot r_{ds1} . \tag{4.62}$$

Therefore, the gain of the gain-boosted amplifier is now

$$\begin{aligned} A_{open-loop}(s) &= \frac{V_{out}(s)}{V_{in}(s)} = -g_{m1} \cdot [R_{out}(s) \parallel \frac{1}{s \cdot C_L}], \\ &= -g_{m1} \cdot \{[A_{add}(s)+1]\, g_{m2} \cdot r_{ds2} \cdot r_{ds1} \parallel \frac{1}{s \cdot C_L}\} \end{aligned} \tag{4.63}$$

where g_{m1} is the transconductance of transistor M_1, C_L is the load capacitance, and $R_{out}(s)$ is the output resistance. Since Equation (4.63) represents the open-loop gain of the gain boost amplifier, this open-loop gain is represented by $A_{open\text{-}loop}$.

The corresponding conductance is

$$G_{out}(s) = \frac{1}{R_{out}(s)} = \frac{1}{(A_{add}(s)+1)\, g_{m2} \cdot r_{ds2} \cdot r_{ds1}} . \tag{4.64}$$

Substituting Equation (4.64) for Equation (4.63), gives

$$A_{open-loop}(s) = \frac{-g_{m1}}{s \cdot C_L + 1/R_{out}(s)} = \frac{-g_{m1} \cdot [A_{add}(s)+1]}{s \cdot C_L \cdot [A_{add}(s)+1] + 1/(g_{m2} \cdot r_{ds2} \cdot r_{ds1})} . \quad (4.65)$$

In order to analyze the frequency property of the gain boost amplifier, substitute Equation (4.60) for Equation (4.65) and then simplify the equation to get

$$A_{open-loop}(s) \approx \frac{-g_{m1} \cdot [A_{add} + s/\omega_2]}{s^2 C_L/\omega_2 + s[A_{add} C_L + 1/(g_{m2} \cdot r_{ds2} \cdot r_{ds1})] + 1/(g_{m2} \cdot r_{ds2} \cdot r_{ds1})} . \quad (4.66)$$

It can be seen in Equation (4.66) that the zero locates in the left half plane; the value of this zero is

$$\omega_z = A_{add} \cdot \omega_2 = GBW_{add} = \omega_4 . \quad (4.67)$$

Furthermore, according to Equation (4.66), the dominant pole and the first non-dominant poles are

$$\omega_{p1} \approx \frac{1}{(A_{add}+1) \cdot g_{m2} \cdot r_{ds2} \cdot r_{ds1} \cdot C_L} , \quad (4.68)$$

$$\omega_{p2} \approx A_{add} \cdot \omega_2 + \frac{1}{g_{m2} \cdot r_{ds2} \cdot r_{ds1} \cdot C_L} . \quad (4.69)$$

As can be seen in Equation (4.69), the first term, which equalsω_4, is the unity gain bandwidth of the additional amplifier and the second term, which equalsω_3, is the dominant pole of the cascode amplifier without using the gain boosting technique. When the first term is much larger than the second term, that is $\omega_4 > \omega_3$, the first non-dominant pole can be expressed by

$$\omega_{p2} \approx A_{add} \cdot \omega_2 = \omega_z . \quad (4.70)$$

Based on the analysis above, the additional amplifier introduces the pole-zero doublet. The pole-zero doublet can be considered cancelled out by locating the zero and pole very close to each other. In this case, the pole-zero doublet introduced by the additional amplifier can be ignored. Hence, the gain boost amplifier unity gain bandwidth is approximately equal to the unity gain bandwidth of the cascode amplifier without using the gain boosting technique. It is well known that the unity gain bandwidth is the product of the gain and the bandwidth. Since the gain is increased by a factor of $A_{add,}$ the dominant pole of the gain boost amplifier should be decreased by a factor of A_{add}.

Based on the above amplitude-frequency property analysis, the relationship of the amplitude-frequency property to the main amplifier (the amplifier without using the gain boosting technique), the additional amplifier, and the gain boost amplifier is shown in Figure 4-19.

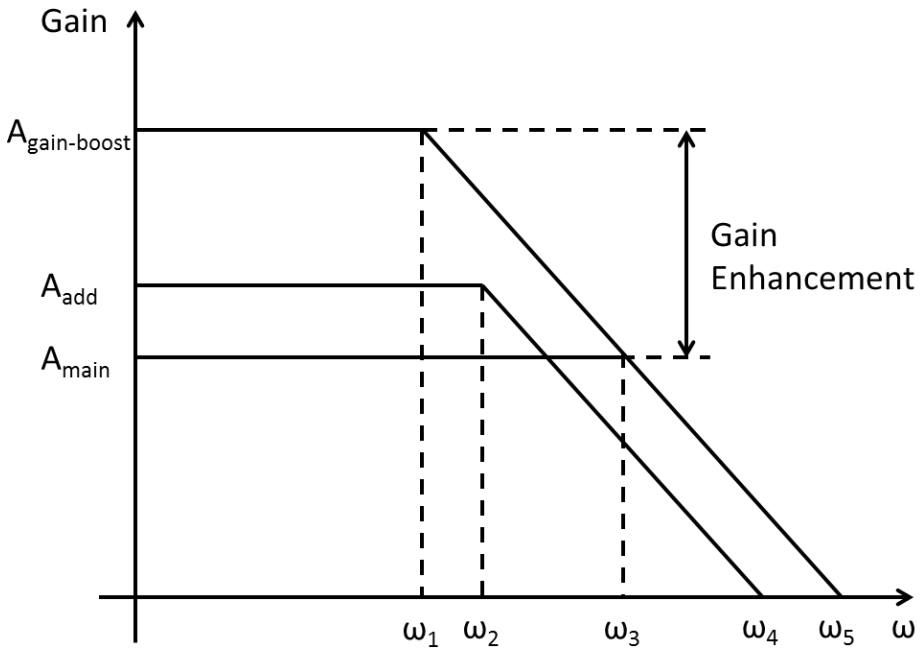


Figure 4-19. Relationship between the main, the additional, and the gain boost amplifiers

While the requirement for the additional amplifier used in the open-loop circuit has already been discussed, never the less, it is still necessary to analyze that requirement for the additional amplifier in a close-loop system. In the residue gain stage, the amplifier is in a close-loop system with the feedback factor of β. The gain of the close-loop system is

$$A_{close-loop}(s) = \frac{A_{close-loop}(s)}{1+\beta \cdot A_{close-loop}(s)} = \frac{A_0 \cdot (1+ s/\omega_z)}{(1+ s/\omega_{pa})(1+ s/\omega_{p2})}, \tag{4.71}$$

where $\omega_{pa} = \beta \cdot GBW_{main}$, $\omega_{p2} = GBW_{add}$, GBW_{main} is the unity gain bandwidth of the main amplifier, and GBW_{add} is the unity gain bandwidth of the additional amplifier.

The step response of the close-loop system is calculated by

$$V_{out}(t) = \mathrm{V}[1 - k_1 \exp(-\omega_{pa} \cdot t) + k_2 \exp(-\omega_{p2} \cdot t)], \tag{4.72}$$

where k1 and k2 are parameters. According to Equation (4.72), the settling time is extended due to the zero-pole doublet [32]. In order to reduce the settling time, ω_{pa} needs to be smaller than ω_{p2}, that is

$$\beta \cdot GBW_{main} < GBW_{add} \cdot \tag{4.73}$$

The equivalent equation of Equation (4.73) is

$$\beta \cdot \omega_5 < \omega_3 \cdot \tag{4.74}$$

In order to optimize the amplifier, both open-loop and close-loop requirements of the additional amplifier need to be satisfied. When both criteria are satisfied, the unity gain bandwidth of the gain boost amplifier is mainly determined by the main amplifier, that is

$$GBW_{total} \approx GBW_{main} = \frac{g_{m1}}{2 \cdot \pi \cdot C_L} \cdot \tag{4.75}$$

It can be seen from the above analysis, in order to save power, the unity gain bandwidth of the additional amplifier does not need to be large. Normally, the power consumption of the additional amplifier is only 1/8 ~ 1/4 of the main amplifier. Therefore, the extra power computation due to the gain-boosted technique is not significant.

4.2.2.2. Design of the amplifier

The gain boost operational amplifier used in the pipeline ADC is shown in Figure 4-20. This gain boost operational amplifier includes one main amplifier and two additional amplifiers. The main amplifier is a telescopic cascode amplifier because this scheme has the benefit of large unity gain bandwidth and sufficient phase margin. The additional amplifiers are folded cascode amplifiers because the scheme has the benefit of a large DC gain. Figure 4-21 shows the scheme of the additional amplifier. For the common mode feedback, the main amplifier uses a discrete time common mode feedback while the additional amplifiers use a continuous time common mode feedback.

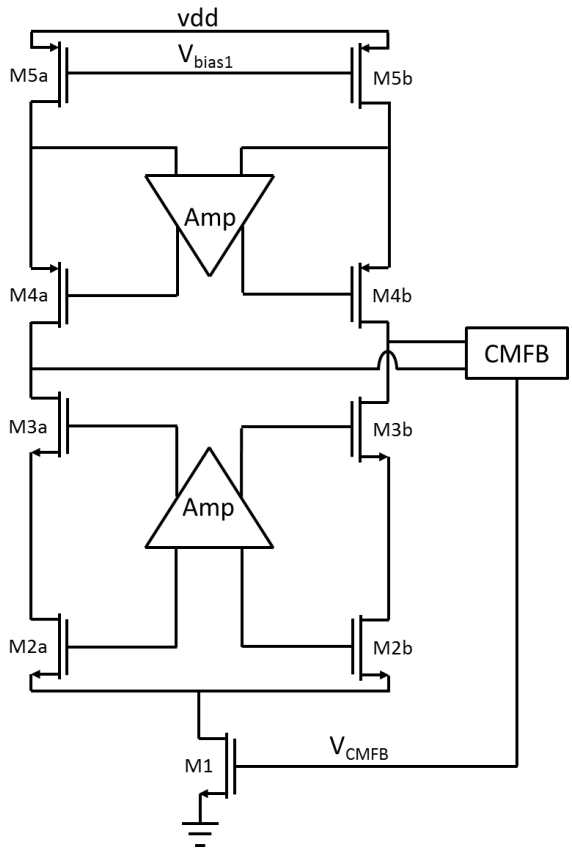


Figure 4-20. Gain boost amplifier

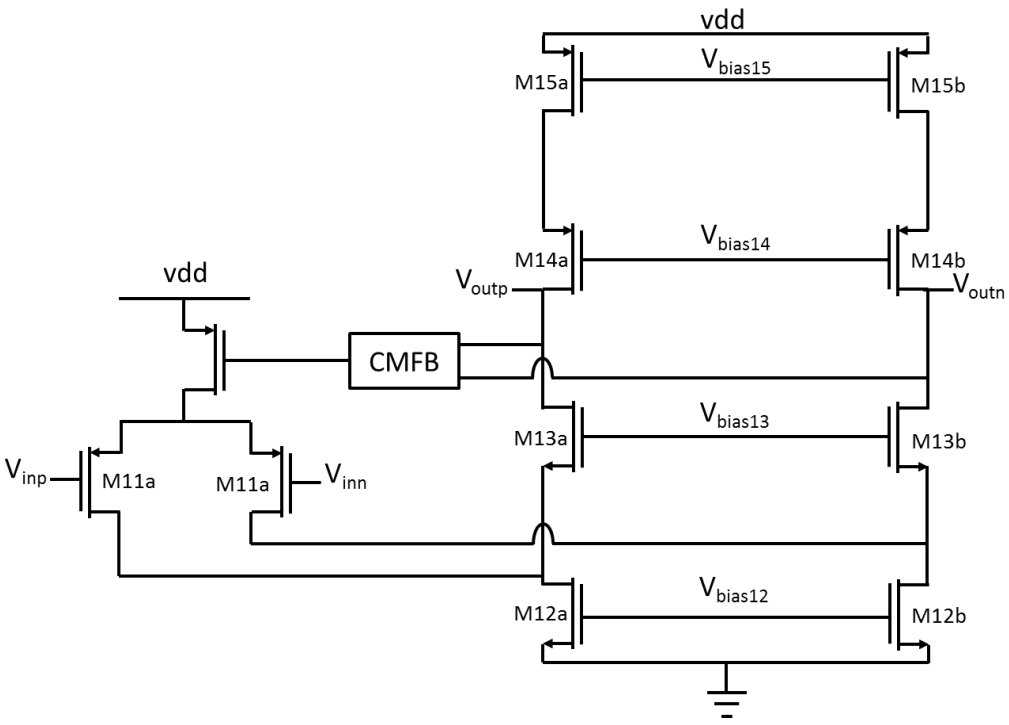


Figure 4-21. Additional amplifier

The total DC gain of this gain boost amplifier is calculated by

$$R_{out} = g_{m2} \cdot (g_{m3} \cdot r_{ds3} \cdot r_{ds2} \cdot A_{add} \square g_{m4} \cdot r_{ds4} \cdot r_{ds5} \cdot A_{add}), \tag{4.76}$$

where g_{m2} is the transconductance of the transistor M2a and M2b; g_{m3} is the transconductance of the transistor M3a and M3b; g_{m4} is the transconductance of the transistor M4a and M4b; r_{ds2} is the output resistance of the transistor M2a and M2b; r_{ds3} is the output resistance of the transistor M3a and M3b; r_{ds4} is the output resistance of the transistor M4a and M4b; r_{ds5} is the output resistance of the transistor M5a and M5b; and A_{add} is the DC gain of the additional amplifier.

4.2.2.3. Simulation results of the amplifier

Figure 4-22 shows the gain boost operational-amplifier simulation results. For a high accuracy requirement, the differential nonlinearity should be less than 1/2LSB (1/32768). Therefore, in order to realize the 14-bit resolution, the DC gain of the gain boost amplifier must

be at least 96dB. According the simulation results, DC gain, unity gain bandwidth, and phase margin are all within the acceptable ranges. Table 4-9 shows the summarized the simulation results.

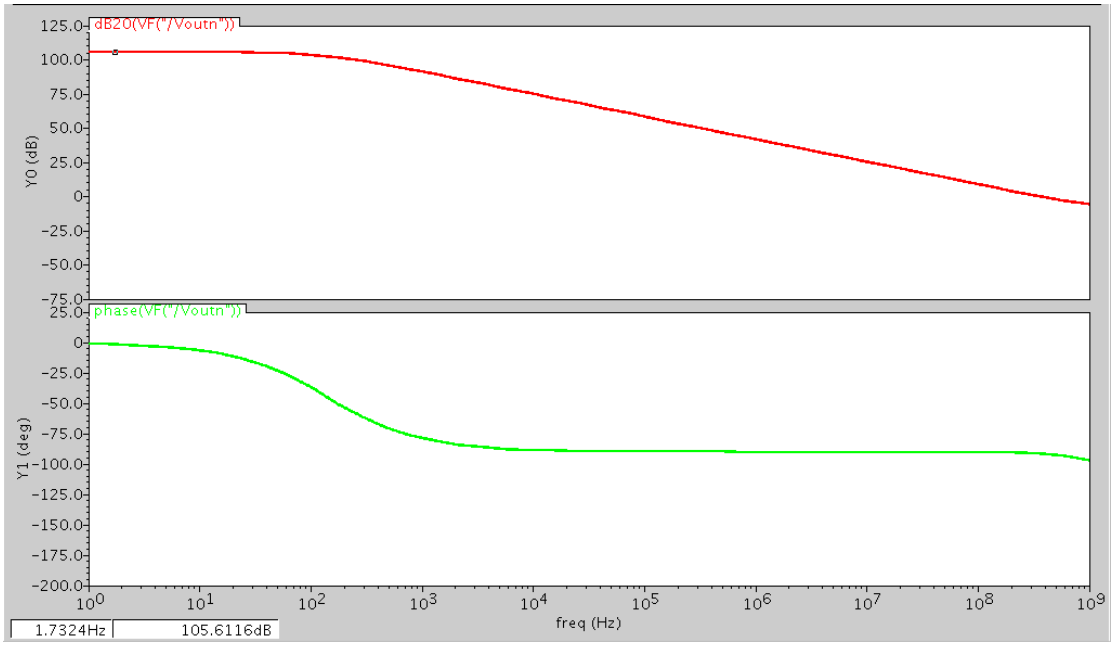


Figure 4-22. Gain boost operational amplifier simulation results

Table 4-9. Summarized amplifier simulation results

DC gain	105 dB
Unity gain bandwidth	340 MHz
Phase margin	87 Degree

4.3. Flash-ADC configuration

In the pipeline ADC, the last stage's configuration is different from the other pipeline stages. Since the output of the last stage does not need to be further processed, the last stage omits the residue gain stage. Also, since the only purpose of the last stage is to provide digital output, the fast and simple flash-ADC topology is adopted. No further stage is used to correct the digital output

of the last stage; therefore, the digital output of the last stage provides a 2-bit output rather than the 1.5-bit output in the earlier stages. In order to generate the 2-bit output, three dynamic comparators with the threshold voltages of $-V_{ref}/2$, 0, and $V_{ref}/2$ are used in the flash-ADC. However, the direct output of the dynamic comparator is the temperature code. An encoder is needed to transfer the temperature codes to the binary digital output codes. Table 4-10 shows the relationships between temperature codes and binary digital output codes.

Table 4-10. The relationships between temperature codes and binary digital output codes

Analog input voltage	Temperature codes	Binary digital output codes
V_{in}	A_2 A_1 A_0	D_1 D_0
$-V_{ref}$~$-V_{ref}/2$	0 0 0	0 0
$-V_{ref}/2$~0	0 0 1	0 1
0~ $V_{ref}/2$	0 1 1	1 0
$V_{ref}/2$ ~ V_{ref}	1 1 1	1 1

Therefore, the transfer functions between the temperature codes and the digital output codes are

$$D_1 = A_1, \tag{4.77}$$

$$D_0 = A_2 + \overline{A_1} \cdot A_0. \tag{4.78}$$

According to the transfer function shown above, the flash-ADC can be designed using the dynamic comparators and the encoder. In the flash-ADC simulation, the setups of the input and reference signals are the same as Table 4-2. The simulation results of the binary digital output code D_1 and D_0 are shown in Figure 4-23. As shown in Figure 4-23, when the input ranges from $-V_{ref}$ to

-V_{ref}/2, the binary digital output codes are D_1D_0=00; when the input ranges from -V_{ref}/2 to 0, the binary digital output codes are D_1D_0=01; when the input ranges from 0 to V_{ref}/2, the binary digital output codes are D_1D_0=10; and when the input ranges from V_{ref}/2 to V_{ref}, the binary digital output codes are D_1D_0=11.

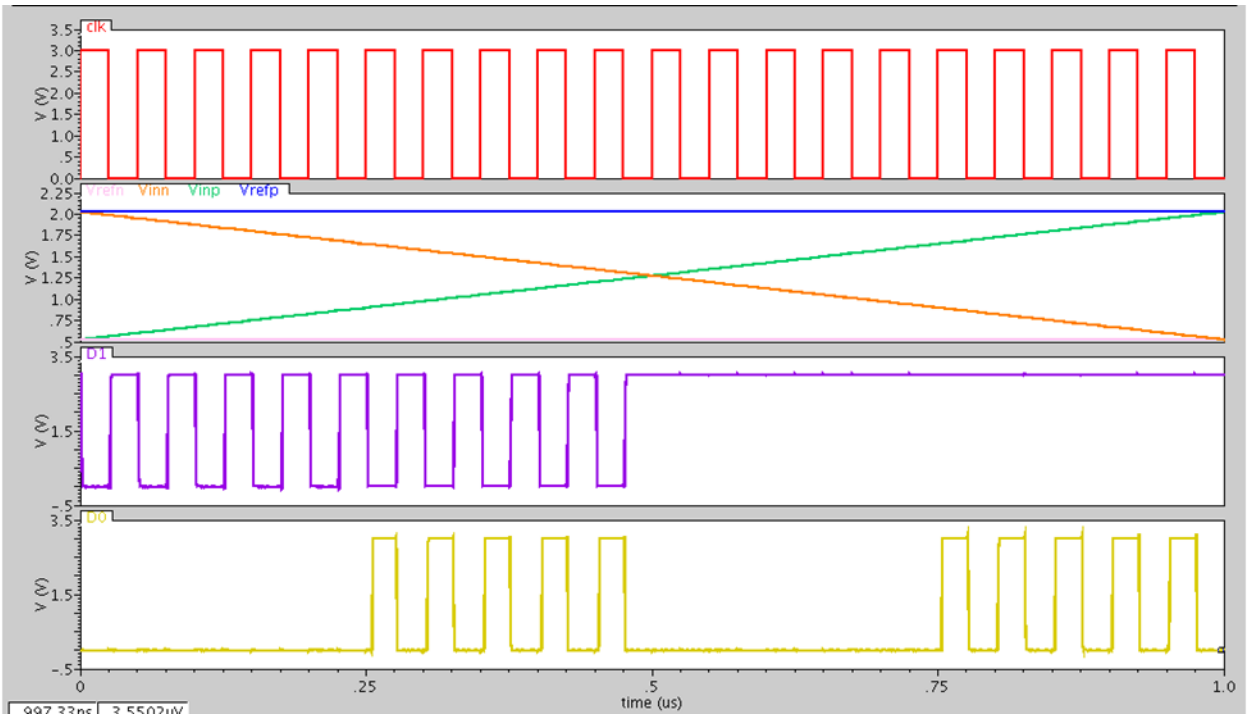


Figure 4-23. Flash-ADC simulation results

4.4. Clock generation circuit

All the pipeline stages are controlled by two-phase non-overlap clock signals. However, the two neighboring pipeline stages are controlled by opposite clock signals. When the odd number pipeline stages work in the sample phase, the even number pipeline stages work in the amplification phase. In order to ensure the synchronization of all the pipeline stages, the clock tree structure is adopted to drive all the pipeline stages. This clock signal generation circuit is shown in Figure 4-24.

The clock signal generation circuit has one input signal clk and five output signals: Φ, Φ_1, Φ_{1p}, Φ_2, and Φ_{2p}. The clock signalΦ is used to control the sample-and-hold circuit in the first pipeline stage. The other clock signalsΦ_1, Φ_{1p}, Φ_2, and Φ_{2p} can be separated into two groups. The clock signals Φ_1 and Φ_{1p} are in one group while clock signalsΦ_2 and Φ_{2p} are in the other group. The clock signals in one group do not overlap with the clock signals in the other group. The non-overlap of the two groups of signals is guaranteed by a feedback delay provided by the inverters as shown in Figure 4-24. The two clock signals in a group have a small phase shift in order to eliminate the charge injection effect. This function is realized by the two clock signal paths, which have different numbers of inverters.

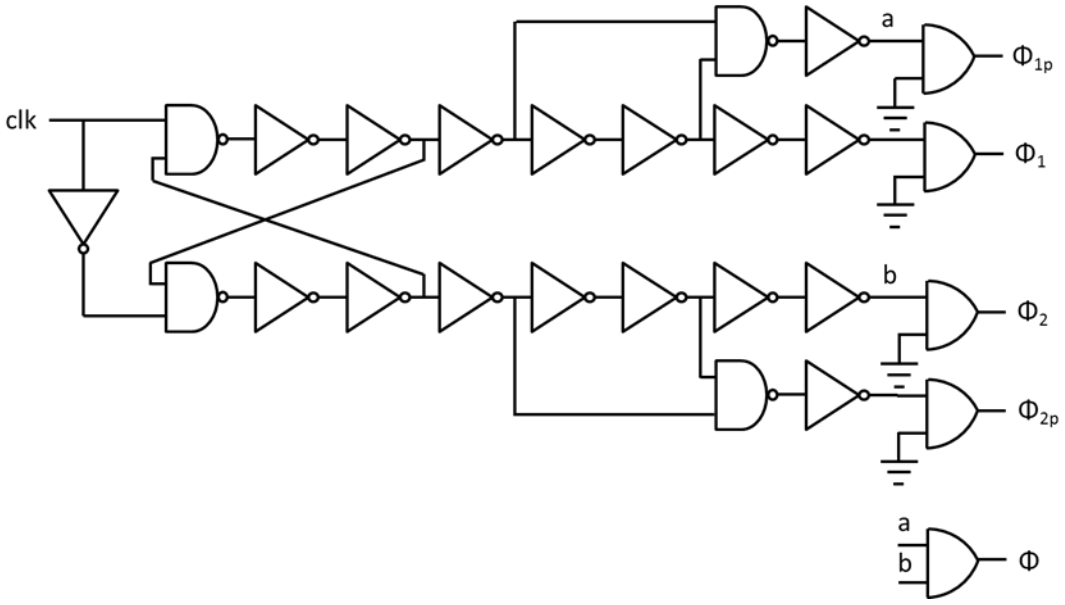


Figure 4-24. Clock signal generation circuit

Figure 4-25 shows the simulation results of the clock generation circuit. As can be seen in the simulation results, the clock signal output satisfies all the requirements: 1. the two group signals do not overlap; 2. the two signals in each group have phase shifts.

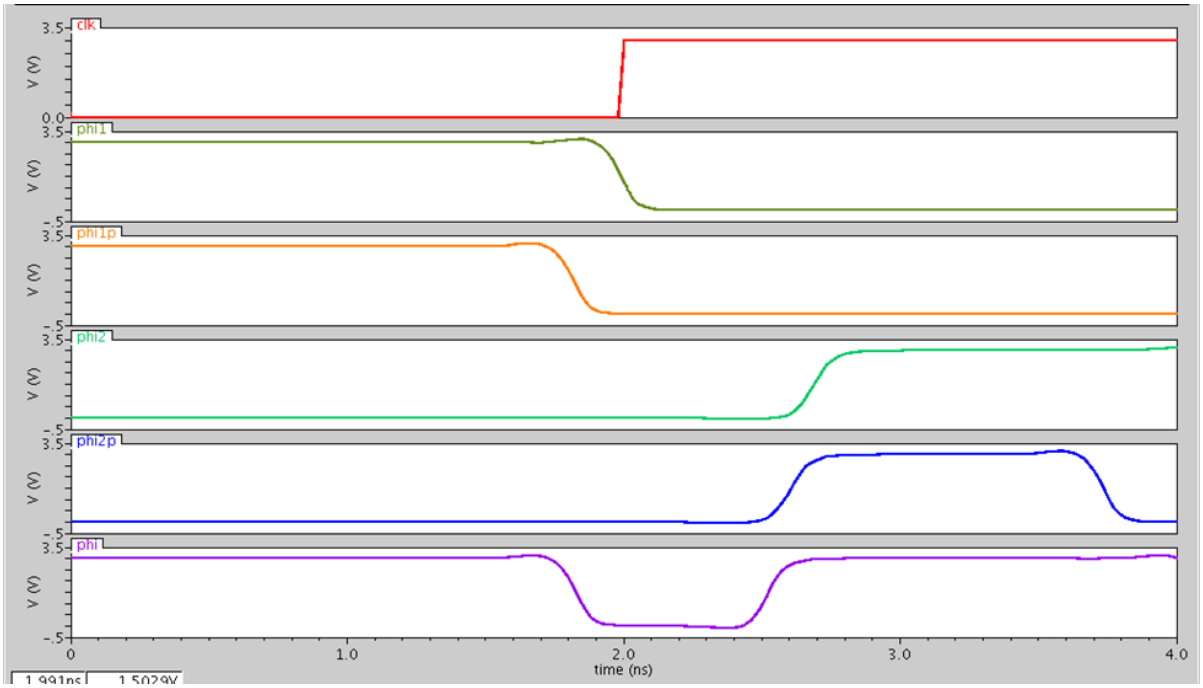


Figure 4-25. Clock signal simulation results

In order to guarantee that all the pipeline stages are controlled by the synchronization clock signals [33], the clock tree structure is adopted. Figure 4-26 shows the clock tree structure.

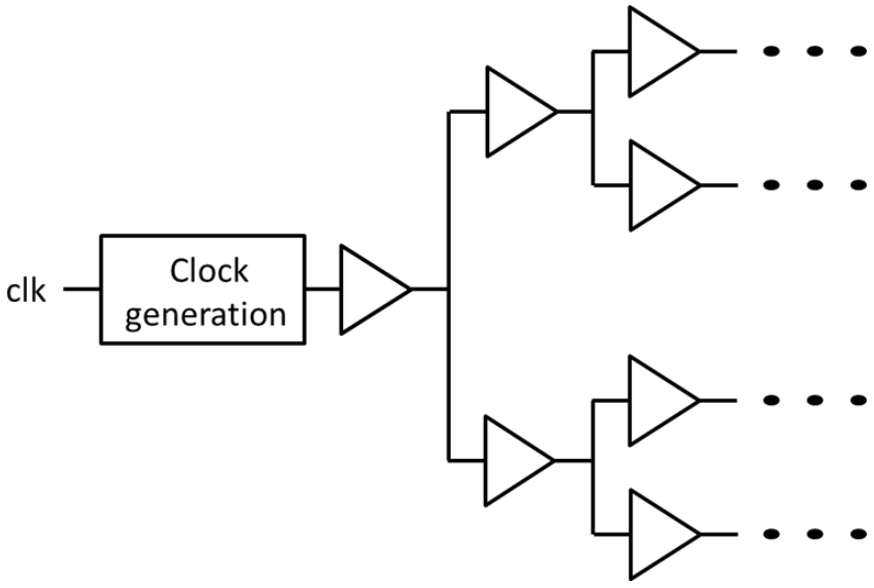


Figure 4-26. Clock tree structure

4.5. Digital logic circuit configuration

The digital logic circuit includes two parts: 1. the time alignment circuit and 2. the digital error correction logic circuit. The time alignment circuit is used to align all the digital output from different pipeline stages generated at different times into one time. In contrast, the digital error correction logic circuit is used to operate digital error correction algorithm in order to correct the comparator offsets.

For a pipeline ADC, an input signal is first processed by the front-end stage, the output of the front-end stage is then fed to the following stage, etc. The input signal is then processed through all the pipeline stages. Therefore, the digital output of an input signal from two neighbouring stages will have half clock cycle time differences. In this design, there are eleven stages total. In order to acquire the digital output from all the pipeline stages, the ADC needs 5.5 clock cycles. Therefore, before the digital output codes are sent to the digital error correction logic, a time alignment circuit is necessary to align all the digital output codes from the different pipeline stages. Figure 4-27 shows the time alignment circuit.

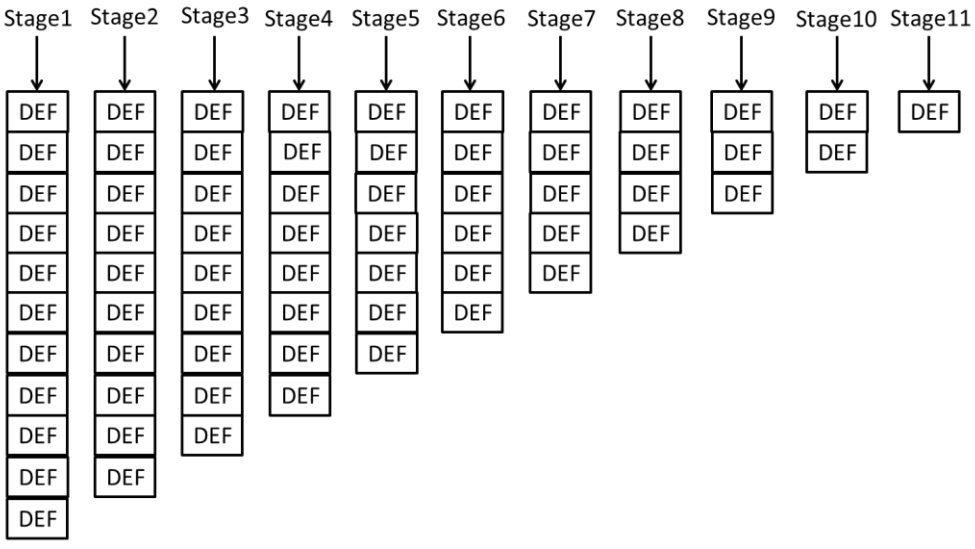


Figure 4-27. Time alignment circuit

This DEF module, shown in Figure 4-27, includes several D flip-flops. The number of D flip-flops in a DEF module depends on the number of bits resolved per stage. For example, if the first stage resolves 4 bits and it then has 4 D flip-flops. The configuration of the D flip-flop used in the DEF module is shown in Figure 4-28. When the clock signal Φ_1 is high, the digital output code is applied to node D and when the clock signalΦ_2 is high, the digital output code arrives at node Q. Before the next raise-edge of clock signalΦ_1, the output code keeps its value at node Q. Therefore, the DEF module has the function of data storage.

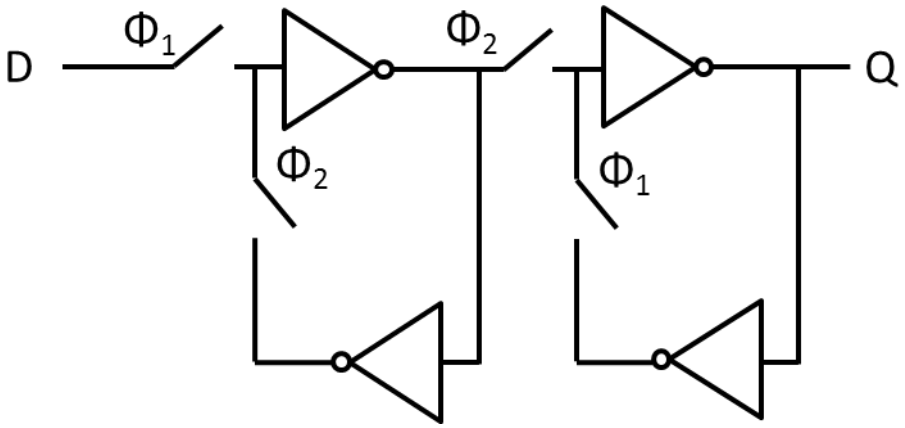


Figure 4-28. D flip-flop configuration

After the time alignment circuit, the digital output codes need to be further processed using the digital error correction technique. As described in Chapter 3, the domain-extended digital error correction algorithm involves minus one operation. Also, as described in Chapter 3, the output codes 0000 minus one is a negative number. In order to avoid a negative number, adding 1 in front of the first stage digital output is necessary. The minus operation can be realized by the following method: a number can be first calculated by complementing the binary digital output codes bit-by-bit; then, the calculated number is added to twice the binary digital output codes; finally, the binary

digital output codes minus one is obtained by deleting the bit on the left. The truth table of this algorithm is shown in Table 4-11. In this table, the first column shows the binary digital output codes and the last column shows the calculated results using the above stated algorithm. As can be seen in Table 4-11, the binary digital output codes shown in the first column minus one equals the binary codes shown in the last column.

Table 4-11. Minus one algorithm truth table

$D_4\ D_3\ D_2\ D_1\ D_0$	$\overline{D_4}\ \overline{D_3}\ \overline{D_2}\ \overline{D_1}\ \overline{D_0}$	$2\cdot(D_4D_3D_2D_1D_0)$	$A_4\ A_3\ A_2\ A_1\ A_0$
1 0 0 0 0	0 1 1 1 1	1 0 0 0 0 0	0 1 1 1 1
1 0 0 0 1	0 1 1 1 0	1 0 0 0 1 0	1 0 0 0 0
1 0 0 1 0	0 1 1 0 1	1 0 0 1 0 0	1 0 0 0 1
1 0 0 1 1	0 1 1 0 0	1 0 0 1 1 0	1 0 0 1 0
1 0 1 0 0	0 1 0 1 1	1 0 1 0 0 0	1 0 0 1 1
1 0 1 0 1	0 1 0 1 0	1 0 1 0 1 0	1 0 1 0 0
1 0 1 1 0	0 1 0 0 1	1 0 1 1 0 0	1 0 1 0 1
1 0 1 1 1	0 1 0 0 0	1 0 1 1 1 0	1 0 1 1 0
1 1 0 0 0	0 0 1 1 1	1 1 0 0 0 0	1 0 1 1 1

The circuit level implementation of the above algorithm is shown in Figure 4-29, Figure 4-30, and Figure 4-31. As shown in Figure 4-29, in order to realize the minus one operation, the binary digital output codes are processed through the minus one block. The detailed circuit of the above minus one block is shown in Figure 4-30. The minus one block includes ten full adders. Each full adder has three input pins: A, B, and carry-in C_{in} and two output pins: sum S and carry-

out C_{in}. The design of the minus one block is based on the algorithm described above. Furthermore, Figure 4-31 shows the full adder configuration.

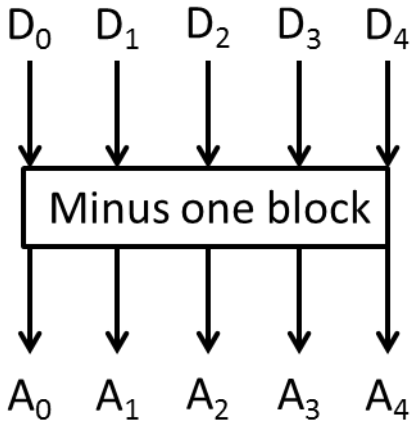


Figure 4-29. Minus one block

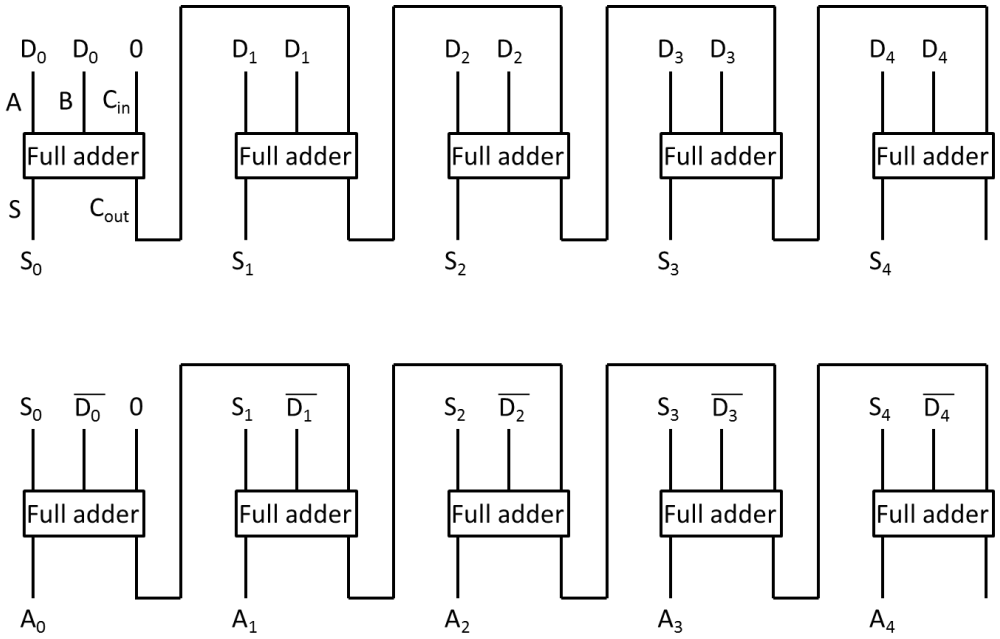


Figure 4-30. Detail circuit of the minus block

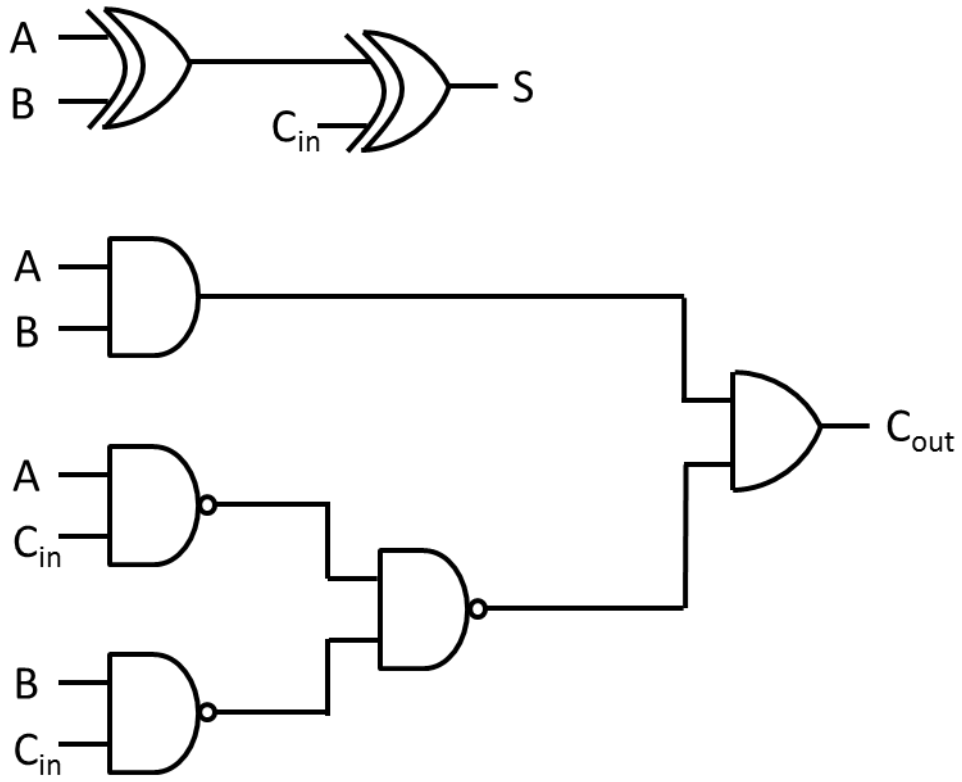


Figure 4-31. Full adder configuration

According to Figure 4-31, the sum S and the carry-out C_{out} can be calculated through the following equations:

$$S = A \otimes B \otimes C_{in}, \tag{4.79}$$

$$C_{out} = A \cdot B + \overline{\overline{A \cdot C_{in}} \cdot \overline{B \cdot C_{in}}} \cdot \tag{4.80}$$

Like mentioned in Chapter 3, in order to realize domain-extended digital error correction technique, some other minus one blocks and dislocation addition blocks are needed. Therefore, the minus operation described in Chapter 3 can be realized through the algorithm described in this chapter. Moreover, the dislocation addition operation can be realized by connecting several full adders serially.

4.6. Conclusion

The chapter presents the detailed circuit configurations and simulations. The pipeline ADC design includes the analog design and the digital design. For the pipeline stages one through ten, the design includes sub-ADC design and MDAC design. However, since the signals do not need to be further processed, a simple flash-ADC needs to be designed for the eleventh and last stage. In contrast to the analog design, the digital design mainly includes both clock design and digital logic circuit design. The simulation results show that these designed components satisfy the pipeline ADC requirement.

CHAPTER 5. EXPERIMENTAL RESULTS AND DISCUSSIONS

5.1. Experimental hardware

The proposed ADC is fabricated with an AMIS 0.5 μm double-poly triple-metal CMOS process. Figure 5-1 shows the die micrograph with a total die area of 4.5 mm^2. The measured power consumption is 264 mW at a 3-V power supply and a 75-MS/s sample rate [21].

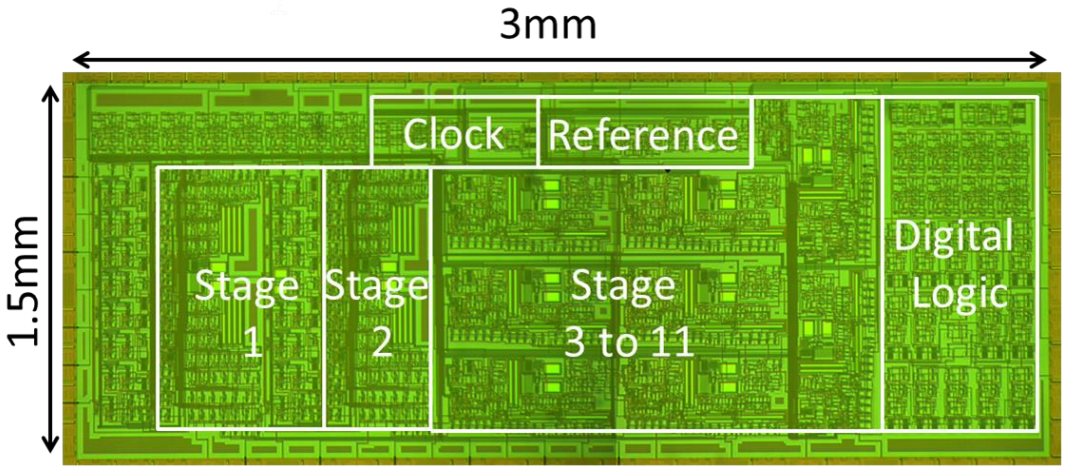


Figure 5-1. Die micrograph

Figure 5-2 shows the test system and Table 5-1 shows the parameters of the test instruments. Function generator1 can generate multi-waves with an output frequency of up to 300-MHz. Therefore, the function generator 1 provides the sine-wave inputs. Function generator 2 can generate square waves with an output frequency of up to 100-MHz. Therefore, function generator 2 provides the clock signals. The outputs of the tested ADC have two signal levels: 3V or 0V, which represent the digital output code of 1 or 0. The outputs of the tested ADC feed into the logic

analyzer. The logic analyzer has the function of an analog-to-digital conversion. It has 32 channels, which are enough to recode the ADC outputs. The threshold voltage of the logic analyser is set at 1.5V. If an input signal is higher than 1.5V, the logic analyzer recodes that input signal as 1; conversely, when an input signal is lower than 1.5 V, the logic analyser records that input signal as 0. Therefore, the stored data in the logic analyser is ready to be processed by computer software. The outputs are then fed to the computer. The computer operates Histogram testing and calculation in order to evaluate the tested ADC's static performance. In addition, the computer runs a FFT analysis and calculation in order to evaluate the tested ADC's dynamic performance.

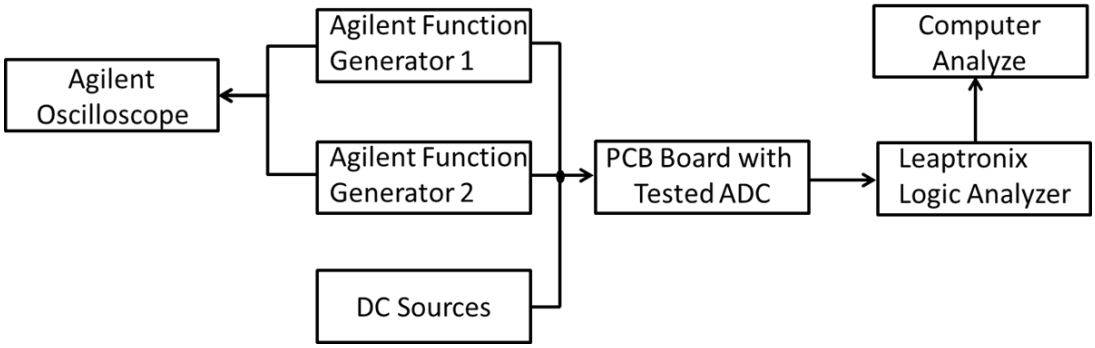


Figure 5-2. Chip test system

Table 5-1. Test instruments parameters

Name	Brand	Parameter
Function Generator1	Agilent	300-MHz multiple waves
Function Generator2	Agilent	100-MHz square wave
Logic Analyzer	leaptronix	32 channels
Oscilloscope	Agilent	
DC Sources		

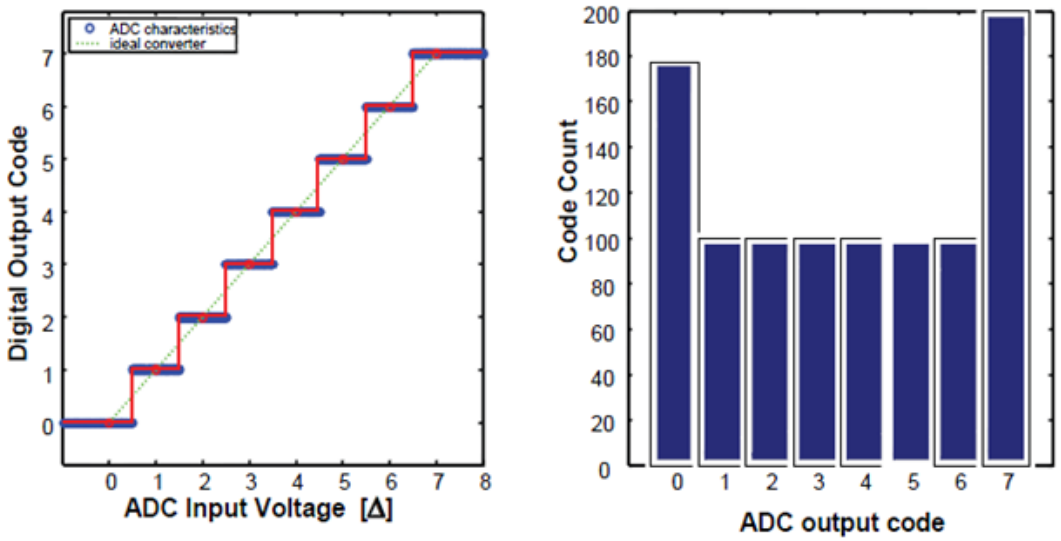
5.2. Experimental software

5.2.1. Static performance measurement method

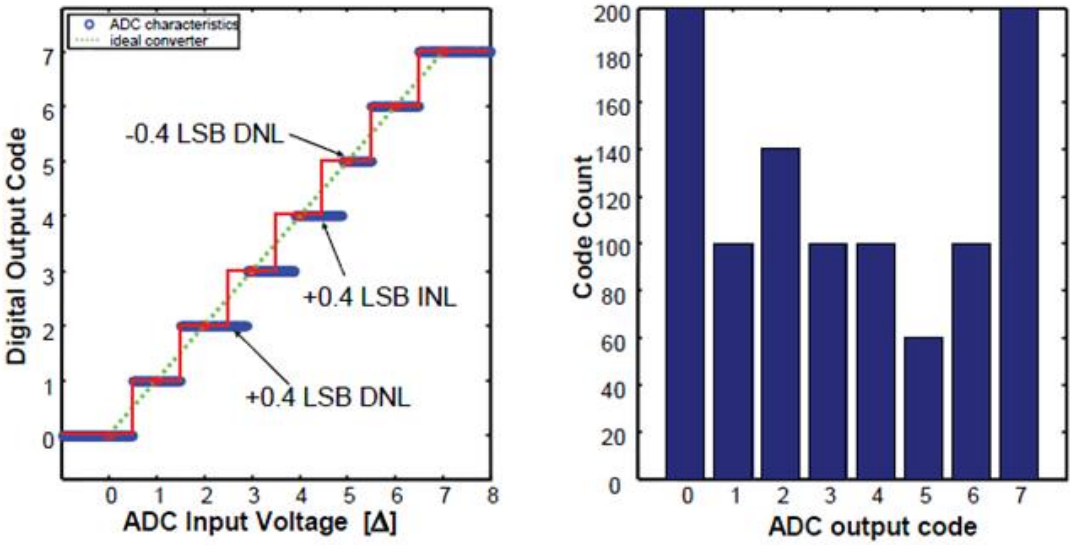
As stated before, in this research, the static performance is measured using Histogram. Using this method, the static performance can be evaluated using two parameters: DNL and INL. The DNL is defined as the actual step size deviation from the ideal step size (1-LSB) with the error normalized with respect to 1-LSB. The INL is the sum of the DNLs. In this research, Matlab is adopted to analyse the data and calculate the static parameters.

For the sake of simplicity, a 3-bit ADC is used to illustrate the measurement method. In order to contrast the Histogram generated by the ideal 3-bit and the real 3-bit ADCs, Figure 5-3 (a) shows the ADC transfer characteristics for the ideal 3-bit converter and the transfer characteristics' corresponding Histogram; Figure 5-3 (b) shows the ADC transfer characteristics for the real 3-bit converter with non-idealities and the transfer characteristics' corresponding Histogram. In the Histogram shown in Figure 5-3 (a), the counts in each bin are equal except the first and the last over range bins. This indicates no DNL and INL errors in the ideal 3-bit ADC. However, in the Histogram shown in Figure 5-3 (b), the counts in each bin may be different except the first and the last bins. This indicates that there are DNL and INL errors in the real 3-bit ADC. Then, Figure 5-3 (c) and Figure 5-3 (d) show the method to extract the DNL and INL errors. The first and the lasts bins are over range bins. As shown in Figure 5-3 (c), the over range bins are removed first. The average counts of the non-over range bins can then be calculated by dividing the total non-over range counts by the total non-over range bins. In this case, the average counts in each non-over range bin are 100. The bins with the digital output codes of 2 and 4 have 140 and 60 counts rather than the average count of 100. This indicates that the DNL error exits at these two points. Therefore, the DNL and INL errors can be extracted, as shown in Figure 5-3 (d). In this

research, the DNL and INL errors of the 14-bit ADC can also be extracted through the same process using the Histogram.

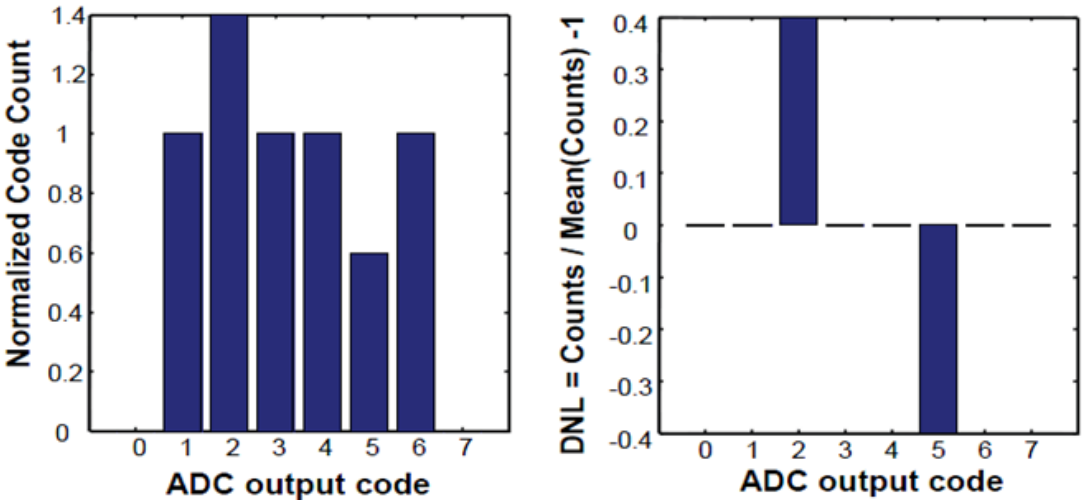


(a)

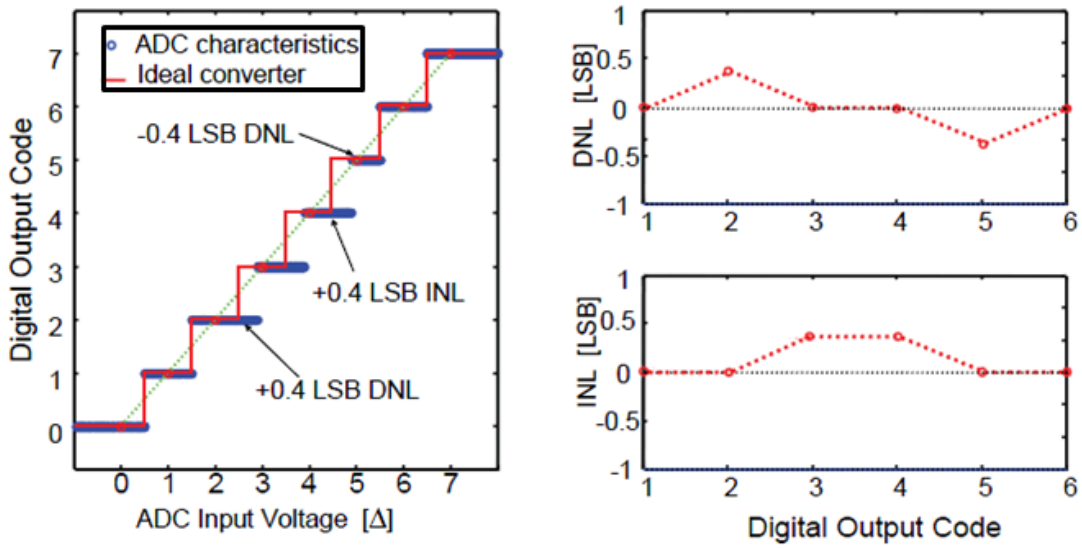


（b）

Figure 5-3. DNL and INL extract process



(c)



(d)

Figure 5-3. DNL and INL extract process (continued)

5.2.2. Dynamic performance measurement method

The acquired digital output codes need to be further processed in order to evaluate the dynamic performance of the ADC. The dynamic performance can be evaluated using the parameters: SNR, SNDR, THD, SFDR, and ENOB, etc. In this research, Matlab is adopted to analyse the data and calculate the dynamic parameters.

The flowchart shown in Figure 5-4 describes the process of calculating the dynamic parameters and drawing the FFT plot. In order to recover the input wave in the digital domain, the digital outputs need to be converted to decimal numbers. Then, a Hanning window is added to reduce the spectrum leakage. For the purpose of calculating dynamic parameters, the time domain data needs to be transferred to the frequency domain. Therefore, applying FFT to the data realizes the domain transfer. The power spectrum can then be calculated, according to the frequency spectrum acquired by applying the FFT. As stated in Chapter 2, the dynamic parameters SNR, SNDR, ENOB, SFDR, and THD can be calculated through the power spectrum.

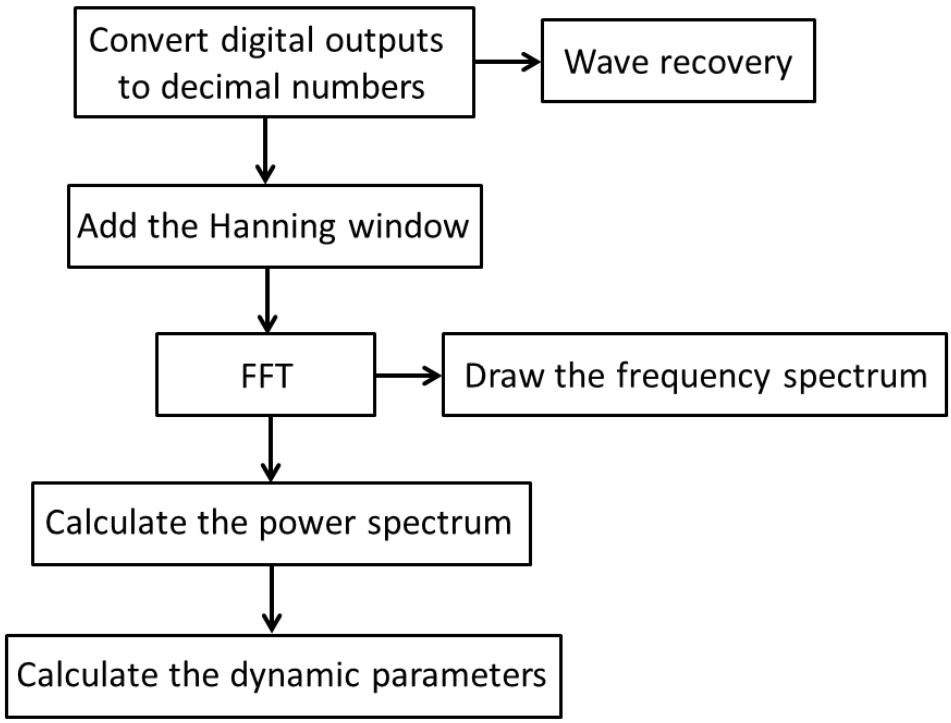


Figure 5-4. Flowchart of the dynamic parameter calculation

The following Matlab program shows the detailed process of calculating dynamic parameters and drawing FFT plot.

```
numpt=2^14;  % the number of samples
```

```
fclk=75e6;    % sample rate is 75-MS/s
numbit=14;   % the ADC resolution is 14 bits
for j=1:numpt;
       Vo(j,15)=0;
      for i=1:numbit;
         Vo(j,15)=Vo(j,15)+Vo(j,i)*(2^(numbit-i));
       end  % Converting the digital outputs to decimal numbers
  Vo(j,16)=Vo(j,15)-(2^numbit-1)/2; % Center the digital sine wave
end
Dout=Vo(:,16);
a=hanning(numpt); % Call the Hanning function
Dout=Dout'; % code
Doutw=Dout.*a;  % Add the Hanning window
Dout_spect=fft(Doutw);  % Perform the Fast Fourier Transform
Dout_dB=20*log10(abs(Dout_spect));  % Recalculate to dB
figure;
maxdB=max(Dout_dB(1:numpt/2));
% The input signal has the maximum amplitude, find the maximum amplitude in dB
plot([0:numpt/2-1].*fclk/numpt, Dout_dB(1:numpt/2)-maxdB);
% frequencies and their corresponding output amplitude in dB
grid on;
title('FFT PLOT');
xlabel('Frequency (MHz)');
ylabel('Amplitude (dB)');
a1=axis;

fin=find(Dout_dB(1:numpt/2)==maxdB);  % find the frequency of the input signal
span=max(round(numpt/200),5); % span of the input frequency on each side
spanh=2;  % Approximate search span for harmonics on each side
spectP=(abs(Dout_spect)).*(abs(Dout_spect)); % Determine the power spectrum
Pdc=sum(spectP(1:span)); % DC power
Ps=sum(spectP(fin-span:fin+span)); % Input signal power
Fh=[];
Ph=[];
for har_num=1:5
   tone=rem((har_num*(fin-1)+1)/numpt,1);
   if tone>0.5
      tone=1-tone;
% Input tones greater than 0.5fs (Nyquist bandwidth) are reflected
   end
    Fh=[Fh tone];
   har_peak=max(spectP(round(tone*numpt)-spanh:round(tone*numpt)+spanh));
har_bin=find(spectP(round(tone*numpt)-spanh:round(tone*numpt)+spanh)==har_peak);
   har_bin=har_bin+round(tone*numpt)-spanh-1;
% ensure the folded back high order harmonics do not overlap with DC or lower order %
harmonics
```

```
   Ph=[Ph sum(spectP(har_bin-1:har_bin+1))];
end
Pd=sum(Ph(2:5)); % determine the total distortion power
Pn=sum(spectP(1:numpt/2))-Pdc-Ps-Pd; %determine the noise power
format;
A=(max(Dout)-min(Dout))/2^numbit;
AdB=20*log10(A);
SNDR=10*log10(Ps/(Pn+Pd));
SNR=10*log10(Ps/Pn);
disp('ENOB(SINAD) is related to SINAD');
ENOB=(SINAD-1.76)/6.02
disp('ENOB(SNR) is related to SNR');
ENOB=(SNR-1.76)/6.02
disp('THD is calculated from 2nd through 5th order harmonics');
THD=10*log10(Pd/Ph(1))
SFDR=10*log10(Ph(1)/max(Ph(2:5)))
disp('Signal & Harmonic Power Components:')
HD=10*log10(Ph(1:5)/Ph(1))
hold on;
plot( Fh(2)*fclk, 0, 'bd',Fh(3)*fclk, 0, 'cx', Fh(4)*fclk, 0, 'r+',Fh(5)*fclk, 0,'bs');
% distinguish the harmonic locations within the FFT plot
hold off;
```

5.3. Experimental results and discussion

The measured DNL and INL without calibration and dithering are within ±0.5 and ±2 LSB, respectively, as shown in Figure 5-5.

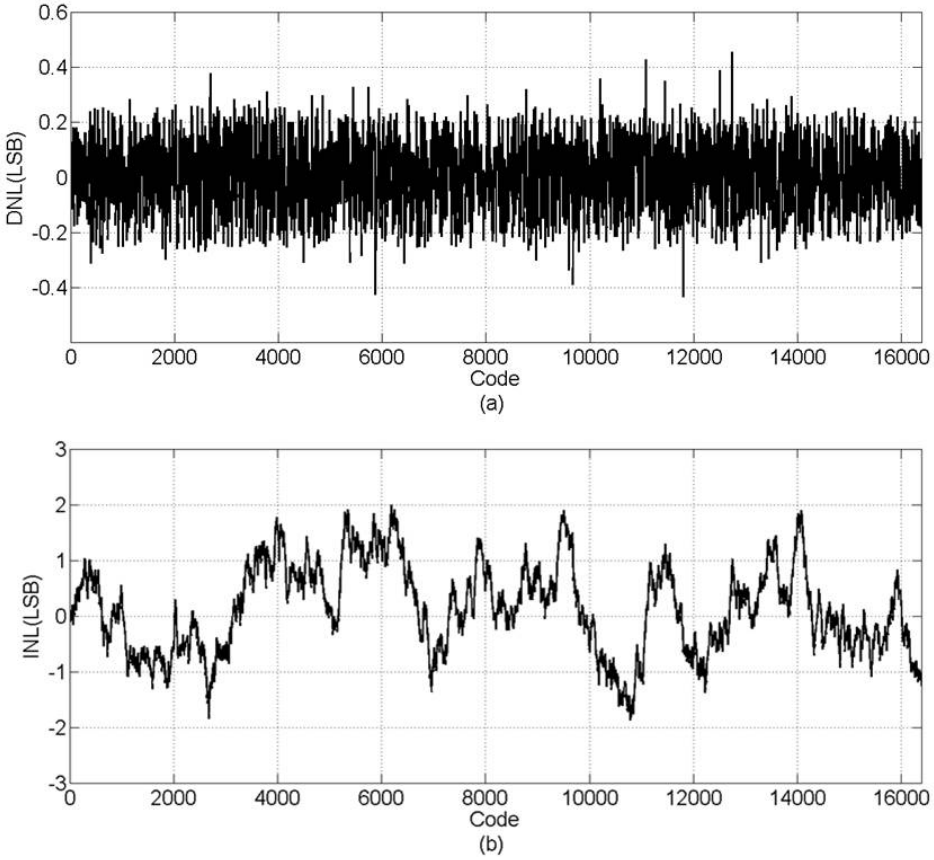


Figure 5-5. Measured (a) DNL and (b) INL

Figure 5-6 (a) shows a FFT plot with a 32 MHz input at a 3-V power supply and a 75-MS/s sample rate. It achieves 93.72-dB SFDR, 90.74-dB THD, and 78.42-dB SNR. Also, the first five Harmonic Distortion ratios are: 0, 93.72-dB, 98.23-dB, 95.98-dB, 108.06-dB, the SNDR is 78.17-dB, and the effective number of bits (ENOB) is 12.8.

Figure 5-6 (b) shows the FFT plot with an 88 MHz input at the same power supply and sample rate as above. It achieves 83.45-dB SFDR, 81.84-dB THD, and 68-dB SNR. Also, the first five Harmonic Distortion ratios are: 0，93.85-dB, 89.24-dB, 83.44-dB, 93.70-dB, and the SNDR is 66.87-dB.

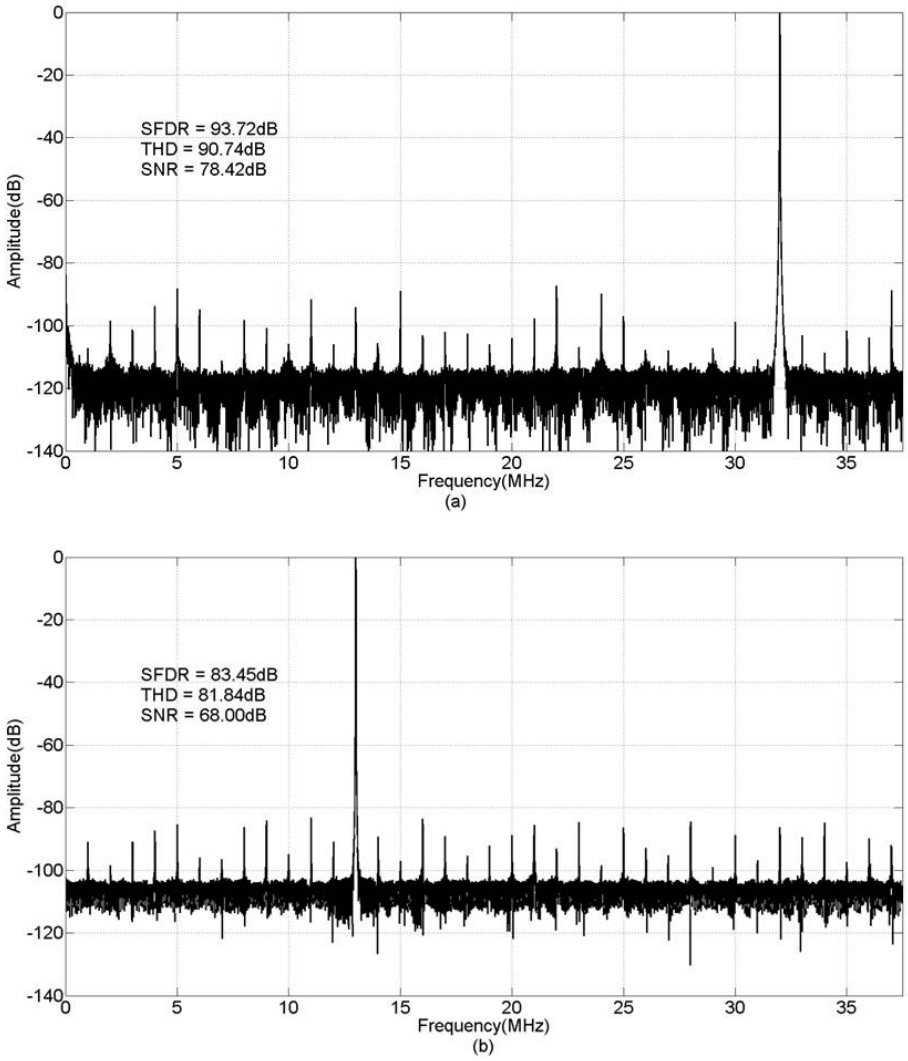


Figure 5-6. FFT plots (a) test at 32 MHz input frequency and (b) test at 88 MHz input frequency

In Figure 5-7, the harmonic components are distinguished in the FFT plot. The FFT plot is symmetrical around the Nyquist frequency, so only half of the plot needs to be drawn. Also, as can be seen in Figure 5-7, the fundamental has the maximum amplitude in the FFT plot. It is well known that the fundamental component has the same frequency as the input signal frequency and the frequency of the harmonic components are integer multiples of the fundamental frequency. If the frequencies of the harmonic components are higher than that of the Nyquist frequency, then

the harmonic components need to be reflected. Furthermore, the folded back high order harmonic components should not overlap with the fundamental component or the harmonic components already existent.

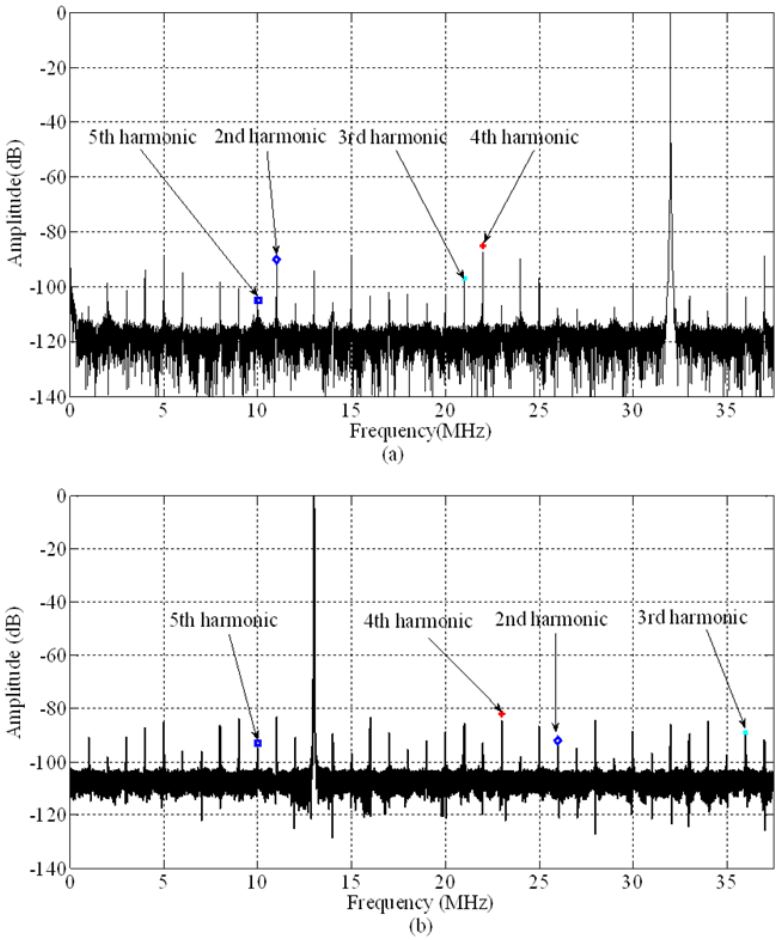


Figure 5-7. Distinguish the harmonic components in (a) 32 MHz input frequency FFT plot and (b) 88 MHz input frequency FFT plot

The measured dynamic performance of the ADC at 3-V power supply and 75-MS/s sample rate is summarized in Figure 5-8. The dynamic performance degrades gradually from low frequency to 100 MHz input. SFDR is over 90-dB with a low input frequency, and decreases to about 80-dB with an input of 100 MHz. In addition, the clock jitter leads to SNR degradation. The

obvious drop in SNR when the input frequency increases from 32-MHz to 42-MHz is due to the poor performance of the multi-bit structure at high input frequencies [16]. Although, a multi-bit structure demonstrates better power efficiency and relaxes the capacitor matching requirement, never the less, the built-in redundancy is half of the 1.5-bit/stage structure [16]. This means that although a domain-extended 2.5-bit/stage increases the built-in redundancy by 50%, it is still much less redundant than the 1.5-bit/stage structure.

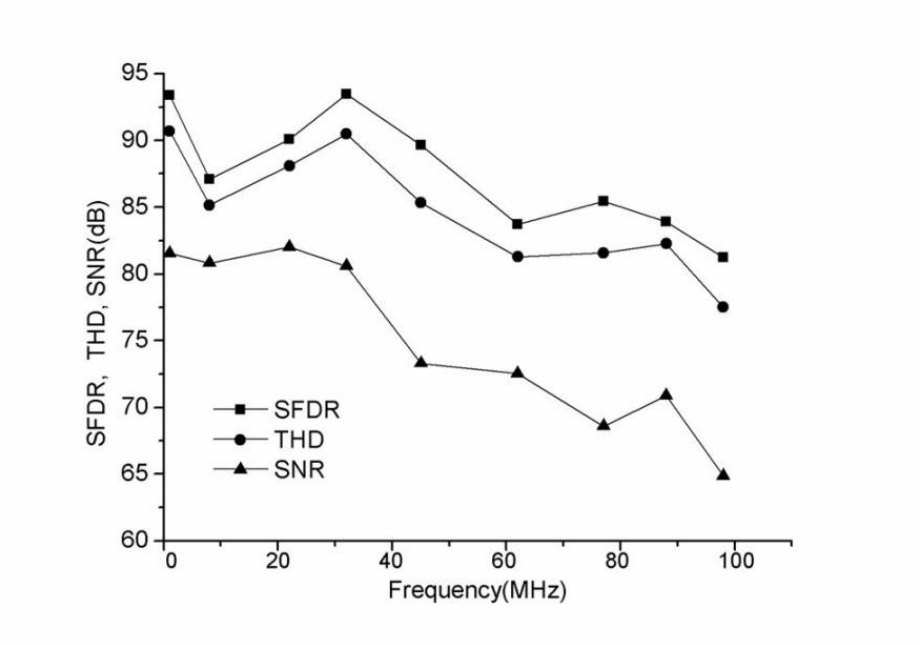


Figure 5-8. Measured dynamic performance at the sample rate of 75-MS/s

The performance of the proposed ADC is summarized in Table 5-2.

Table 5-2. Performance summary

Parameter	Proposed
Process (μm)	0.5
Sampling rate (MS/s)	75
Resolution (bit)	14
DNL (LSB)	±0.5
INL (LSB)	±2
SFDR (dB) @ 32MHz input	93.72
SNR (dB) @ 32MHz input	78
THD (dB) @ 32MHz input	90.74
SINAD (dB) @ 32MHz input	78
ENOB (bits) @ 32MHz input	12.8
Power (mW) @ 32MHz input	264
Chip area (mm^2)	4.5

A comparison of the proposed ADC to other ADCs with sample rates above 1MS/s is shown in Table 5-3. To evaluate the efficiency of the ADC, the figure of merit (FOM) can be calculated by the following equation:

$$FOM = \frac{Power}{10^{SNDR/20} \cdot 2 \cdot f_{in}}, \tag{5.1}$$

where Power is the total power consumption, SNDR is the signal-to-noise and distortion ratio, and f_{in} is the input frequency at which the SNDR is measured. It is well known that a smaller FOM indicates a more efficient ADC. As shown in Table 5-3, the presented ADC achieves a good trade-off between power and performance compared to other ADCs, only surpassed by the ADC shown in [34].

Table 5-3. Performance comparison

Parameter	Ref. [2]	Ref. [4]	Ref. [35]	Ref. [36]	Ref. [34]	Ref. [18]	This work
Process(μm)	0.35	0.35	0.35	0.35	0.18	0.09	0.5
Sampling rate(MS/s)	75	125	75	80	125	100	75
Resolution(bit)	14	14	12	14	16	14	14
SFDR(dB)	85	100@10MHz 82@400MHz	71.2	84.8	96@30MHz 87@151MHz	71	93.7
SNDR(dB)	73	75@10MHz 72 @400MHz	63.5	72	78.7@30MHz 76.7@151MHz	55	78
Power(mW)	341	1850	273	303	385	12.2	264
f_{in} (MHz)	38	400	37.5	36,.7	151	20	32
FOM (Pj/step)	1	0.5	2.4	1	0.2	0.5	0.5

5.4. Conclusion

The ADC is fabricated with an AMIS 0.5 μm double-poly triple-metal CMOS process. According to the experimental results, the ADC exhibits both a high static and a high dynamic performance. In addition, FOM is a parameter that can evaluate the efficiency of an ADC. Compared to other ADCs presented in articles, this ADC achieves a good balance between power and performance.

CHPATER 6. DITHER-BASED DIGITAL BACKGROUND CALIBRATION FOR PIPELINE ADCS

The switched-capacitor pipeline ADC is mainstream architecture in wireless communication and digital consumer products. The performance of the switched-capacitor pipeline ADC is significantly limited by the linear errors due to comparator offsets, capacitor mismatches, and finite operational-amplifier gains. A trend in modern ADC design is to utilize digital background calibration to calibrate both capacitor mismatches and finite operational-amplifier gains [37] - [9]. In contrast, a 14-bit level performance without calibration has been achieved by carefully matching the capacitors, using high open-loop DC gain operational-amplifiers and a multi-bit first stage [2] [4] [21]. Therefore, in order for a digital calibration to be useful, it should minimize the analog circuits as well as have the reasonable convergence time. Fortunately, digital circuits have higher speeds and lower power consumption as compared to an analog counterpart [38].

Indeed, many digital calibration algorithms have been reported. For example, the parallel-ADC algorithm uses a high accuracy but slow ADC to calibrate the errors of a high speed ADC [39]. This architecture requires an extra ADC, which costs both more power consumption and chip area. On the other hand, the statistic-based algorithm [40] requires both a large memory to store statistic data and an extremely low convergence speed. However, in the dither-based algorithm [41] [42], the gain errors are modulated with a pseudorandom noise sequence (PN sequence) in the analog domain and the gain errors are demodulated in the digital domain in order to extract the errors from the processed input signal. In [9], the injection of dithers, together with comparator offsets, makes the output voltage of the current stage exceed the input range of the following stage.

This out-of-range input leads to a code loss. Two options are reported to prevent an ADC from a code loss. The first option is to reduce the amplitude of the input signal, which causes the transmitting signal's SNR to decrease [9]. The other option is to reduce the amplitude of the dither, which leads to a longer convergence time [43] [44].

In order to relieve these two conflicting issues, a domain-extended dither-based algorithm is proposed in this research. This proposed algorithm extends the input range; therefore, both the amplitudes of signal and dither are not necessary to be reduced. In this case, the extended domain results in reasonable redundancy space for comparator offsets. This proposed algorithm ensures that the average amplitude of injecting dither stays high, which results in fast convergence speed. The algorithm also avoids reducing the amplitude of the signal, which means the SNR is not harmed.

Moreover, in order to further increase the convergence speed, this research also introduces a variable-amplitude domain-extended dither-based algorithm to calibrate gain errors for ADCs. This variable-amplitude domain-extended dither-based algorithm ensures a higher dither amplitude than the previous domain-extended dither-based algorithm; therefore, the convergence speed is much faster. A similar scheme is used in the traditional 1.5-bit/stage pipeline ADC to calibrate gain errors [45]. However, this newly developed variable-amplitude domain-extended dither-based algorithm has more advantages compared to the calibration method in [45] as is discussed later in this chapter.

This Chapter is organized as follows: The dither-based digital background calibration algorithm is described in Section 6.1. The domain-extended dither-based algorithm to correct linear errors is described in Section 6.2. Simulation results from the modelled 12-bit ADC using the domain-extended dither-based algorithm are discussed in Section 6.3. The variable-amplitude

domain-extended dither-based algorithm to correct linear errors is described in Section 6.4. Simulation results from the modelled 12-bit ADC using the variable-amplitude domain-extended dither-based algorithm are discussed in Section 6.5. Section 6.6 concludes the chapter.

6.1. Dither-based digital background calibration algorithm

A pipeline ADC consists of several low resolution conversion stages. Figure 6-1 shows a generalized single stage converter that includes a tri-level MDAC in two signal phases. Figure 6-2 (a) and Figure 6-2 (b) show the specific two phases, sample phase topology and the amplification phase topology, of the generalized single stage converter, respectively. In each stage, the residue is generated through subtracting the output of sub-DAC from the sample-and-hold input. This residue is then amplified through an operational-amplifier. The amplified residue is then fed into the next stage of the pipeline ADC [46].

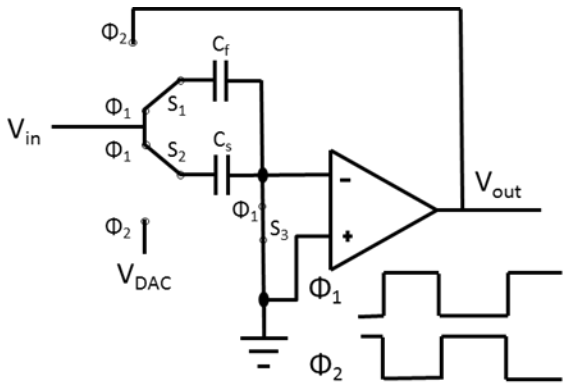


Figure 6-1. Single stage converter in two phases

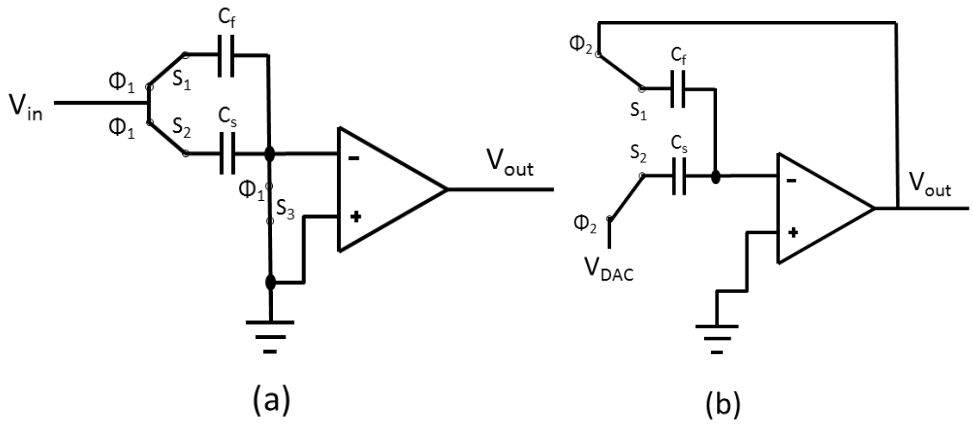


Figure 6-2. (a) Sample phase topology of the MDAC and (b) amplification phase topology of the MDAC

The performance of an ADC is highly dependent on the amplifier residue, because the comparator offset can be corrected by the digital error correction technique using redundancy bits [47]. The two major non-idealities which affect the amplifier residue are finite open-loop gains and capacitor mismatches. When both capacitor mismatches and finite open-loop gains are take into consideration, the amplifier residue of a stage equals to [30]

$$V_{res} = \frac{1}{\beta}\frac{1}{1+\frac{1}{\alpha\cdot\beta}}\left(V_{in}-\frac{1}{2}V_{DAC}\right)$$
$$= \frac{C_S+C_f}{C_f}\frac{1}{1+\frac{\left(C_S+C_f\right)}{\alpha\cdot C_f}}\left(V_{in}-\frac{1}{2}V_{DAC}\right). \quad (6.1)$$

Equation (6.1) takes both capacitor mismatches and finite open-loop gain into consideration. Therefore, both non-ideal factors that affect the amplifier residue can be combined into one error, called the gain error. Thus, the real close-loop gain is

$$A_{real} = \frac{C_S + C_f}{C_f} \frac{1}{1 + \frac{\left(C_S + C_f\right)}{\alpha \cdot C_f}}. \tag{6.2}$$

The equivalent model for a single pipeline stage considering the non-ideal factors mentioned above is shown in Figure 6-3.

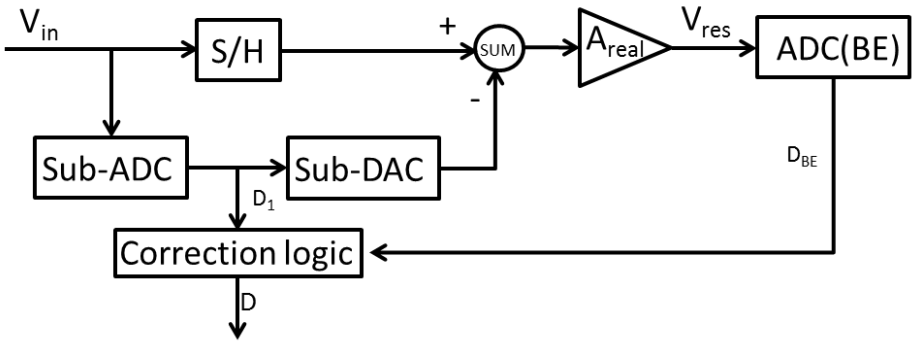


Figure 6-3. Modelled single stage converter

In contrast to the non-ideal condition, in ideal conditions, the open-loop gain of an operational amplifier is infinite and the capacitors are perfectly matched. Thus, when $(C_s+C_f)/C_f = N$ in the ideal condition, the close-loop gain is equal to N. In the non-ideal condition, however, the expression for the output word is [9]

$$D = D_1 + \frac{1}{A_{real}} \cdot D_{BE}, \tag{6.3}$$

where A_{real} is the real close-loop gain, D_1 is the digital output of sub-ADC with the dither, and D_{BE} is the output code of the back-end ADC.

The estimation of $m=1/A_{real}$ is m_{est}, Equation (6.3) can be rewritten as

$$D = D_1 + m_{est} \cdot D_{BE}\,, \tag{6.4}$$

$$D_{BE} = V_{res} = A_{real} \cdot (V_{in} - D_1)\,. \tag{6.5}$$

Subscribing Equation (6.5) into Equation (6.4) yields

$$D = D_1 + m_{est} \cdot A_{real} \cdot (V_{in} - D_1) = V_{res} = V_{in} \cdot (1+e) - D_1 \cdot e\,, \tag{6.6}$$

where

$$e = \frac{m_{est} - m}{m}\,. \tag{6.7}$$

In Equation (6.7), error coefficient e demonstrates the deviation of the estimate value m_{est} from the real value m. For example, when m_{est} is equal to m, the error coefficient e is equal to zero. In order to extract the error coefficient e, one must access D_1, which is the digital output of sub-ADC with the dither. The digital output D_1 can be calculated by adding D_a, the digital output of sub-ADC without the dither, to the dither $PN \cdot \Delta S$. In order to prevent the dither from disturbing the normal signal conversion, the dither must be signal independent. In order for the dither to be signal independent, the PN, a signal independent pseudorandom noise sequence, must belong to {-1, 1}. This PN sequence has an average value of zero. Therefore, the dither can be obtained by multiplying the dither amplitude ΔS and a PN sequence. Thus, the digital output of sub-ADC with the dither can be calculated by the following equation:

$$D_1 = D_a + PN \cdot \Delta S\,, \tag{6.8}$$

where D_a is the output word of the sub-ADC without the dither and ΔS is the amplitude of the dither. Substituting Equation (6.8) into Equation (6.6) gives

$$D=V_{in}\cdot(1+e)-D_a\cdot e-PN\cdot\Delta S\cdot e. \tag{6.9}$$

The error coefficient from Equation (6.9) can be extracted by adding a PN·D sequence then calculating the average. However, because D is highly dependent on the input signal, a new parameter is defined to reduce D's dependency. The new parameter equation is shown in Equation (6.10):

$$W(n)=D(n)\text{-}D_a(n), \tag{6.10}$$

where n is the number of conversions.

Substituting Equation (6.6) and Equation (6.8) into Equation (6.10), for ease of error coefficient e calculation, gives

$$W(n)=\left(V_{in}(n)-D_a(n)\right)\cdot(1+e)-PN\cdot\Delta S\cdot e. \tag{6.11}$$

In Equation (6.11), only ΔS·e multiplies with the PN pseudorandom noise sequence. In order to extract the error coefficient e, multiplying Equation (6.11) with the same PN sequence gives

$$PN(n)\cdot W(n)=PN(n)\cdot\left(V_{in}(n)-D_a(n)\right)\cdot(1+e)-PN^2\cdot\Delta S\cdot e. \tag{6.12}$$

According to the statistical practices for averaging an infinite number of cumulative sums (represented by symbol E[.]) of PN, the result is zero. Therefore, the first term in Equation (6-12) will equal zero and will yield

$$E(PN(n)\cdot W(n))=-\Delta S\cdot e. \tag{6.13}$$

The error coefficient e in the analog domain is extracted in the digital domain as shown in Equation (6.13). There are two ways to eliminate the error coefficient e. The one is to use complicated digital calculations with the help of DSP [43]. The other way is to apply least mean square (LMS) iterative calculation [44]. With the reduction of the error coefficient e, the m_{est} keeps renewing until it approaches $1/A_{real}$ as described in Equation (6.14)

$$m_{est}(n)=m_{est}(n-1)+\mu\cdot PN(n-1)\cdot W(n-1), \tag{6.14}$$

where μ is the step size for the LMS iterative. For a smaller step size, a longer convergence time needed for an ADC to be stable. However, for a larger step size, larger stead-state errors existed for an ADC.

6.2. Domain-extended dither-based algorithm

In order to gain more amplitude for dithering, the digital background calibration using a domain-extended dither-based algorithm is proposed in this research. In the domain-extended dither-based algorithm, two extra threshold voltages are used to divide the input range to more domains. For a generic N+0.5-bit using the traditional digital error correction technique, the input range can be divided into $(\sum_{i=1}^{N} 2^i) + 1$ domains. For example, when N is 1 the input range is divided into 3 domains, and when N is 2 the input is divided into 7 domains. In contrast, for a generic N+0.5-bit using the proposed domain-extension algorithm, the input range can be divided into $(\sum_{i=1}^{N} 2^i) + 3$ domains. For example, when N is 1 the input range is divided into 5 domains, and when N is 2 the input is divided into 9 domains.

This algorithm also extends the coded input range. For a generic N+0.5-bit using the traditional digital error correction technique, the input range is from $-V_{ref}$ to V_{ref}. For example, when N is 1 the input range is from $-V_{ref}$ to V_{ref}, and when N is 2, the input range is also from -

V_{ref} to V_{ref}. In contrast, for a generic N+0.5-bit using the proposed domain-extension algorithm, the input range is from $-(1+1/2^{N+1})V_{ref}$ to $(1+1/2^{N+1})V_{ref}$. For example, when N is 1 the input range is from $-(5/4)V_{ref}$ to $(5/4)V_{ref}$, and when N is 2, the input range is from $-(9/8)V_{ref}$ to $(9/8)V_{ref}$. However, the input range of the domain-extended 2.5-bit/stage ADC shown in Chapter 3 is from $-(10/8)V_{ref}$ to $(10/8)V_{ref}$. The reason causes this range difference is that the dither is not injected to the ADC shown in Chapter 3 while the dither is injected to the one shown in this chapter. As can be seen in the transfer characteristics of the domain-extended 2.5-bit/stage ADC shown in Chapter 3, if a fixed dither is injected to the ADC between the range of $-(10/8)V_{ref}$ ~ $-(9/8)V_{ref}$ and $(9/8)V_{ref}$ ~ $(10/8)V_{ref}$, the amplitude of signal plus dither would exceed the input range of the following stage; therefore, a code loss would happen. In addition, the same situation holds for the domain-extended 1.5-bit/stage ADC.

A multi-bit front-end is widely used in the pipeline ADCs because a multi-bit front-end can relax the capacitor matching requirement as well as provide a better power efficiency. However, if one more bit per stage were to be resolved, then the comparator offset correction ability is reduced by half. As stated before, this proposed algorithm improves the comparator offset correction ability, thus relieving the conflict between the multi-bit front end and the reduced comparator's offset correction ability. Also, the extended input range leads to an extended output range. Therefore, more space is allowed for dithering, which results in a faster convergence speed. Based on the analysis stated above, the choice of a domain-extended 2.5-bit front-end is preferred. For later pipeline stages, with their accumulation of noise and distortion, the domain-extended 1.5-bit pipeline stages are preferred. The transfer characteristics of the traditional 2.5-bit/stage ADC and the proposed domain-extended 2.5-bit/stage ADC are shown in Figure 6-4 (a) and Figure 6-4 (b), respectively. The coded range of the traditional seven-domain 2.5-bit/stage ADC is from $-V_{ref}$

to V_{ref}. Since the output of a stage is the input of the following stage, the output range should not exceed the $-V_{ref}$ to V_{ref} range in order to prevent code loss. In order to acquire the maximum convergence speed of the traditional technique, the sum of the signal plus dither equals the full-scale $-V_{ref}$ to V_{ref} and no redundancy space is left for comparator offsets. However, when some space for redundancy space is required for comparator offsets, the dither amplitude has to be reduced to leave some redundancy space. In this case, the convergence speed is slower due to the reduced dither amplitude.

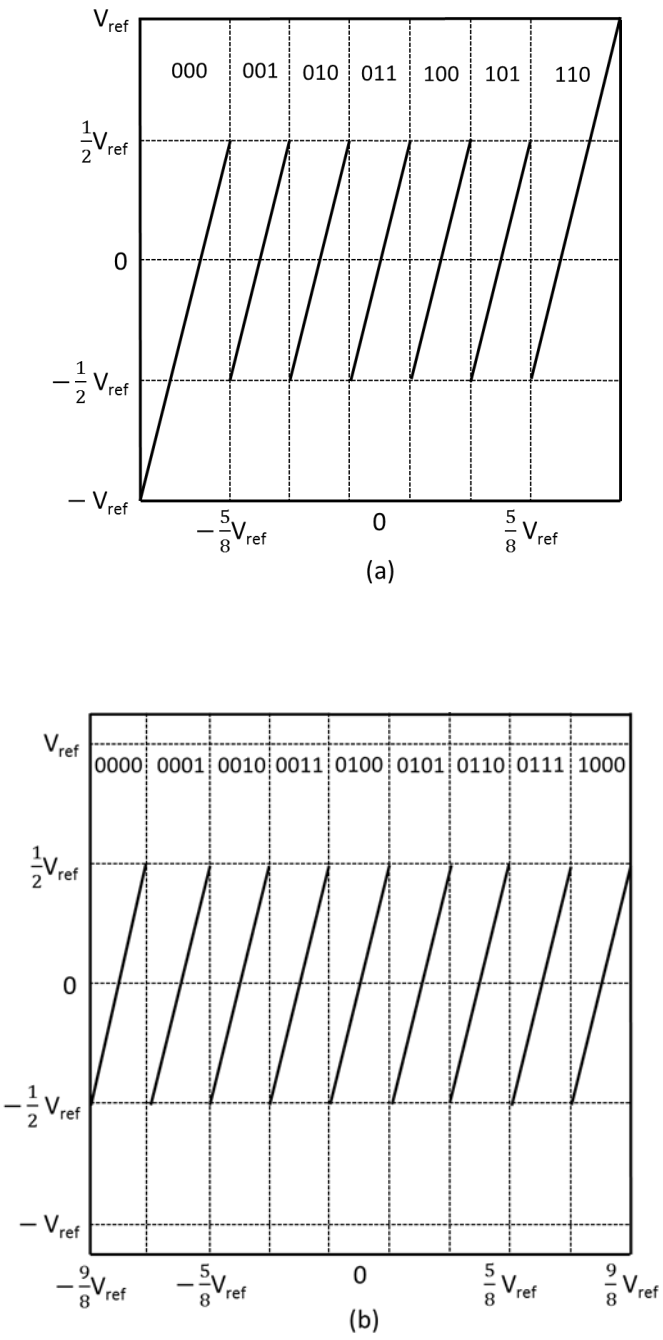


Figure 6-4. (a) Transfer characteristics of the traditional 2.5-bit/stage ADC and (b) transfer characteristics of the domain-extended 2.5-bit/stage ADC

The first two stages have the proposed 2.5-bit/stage configuration. This configuration has an ideal inter-stage gain of four, which is the same as the traditional 2.5-bit/stage configuration. But the proposed domain-extended 2.5-bit/stage configuration uses eight comparators to separate

nine input levels, referred to as nine-domain 2.5-bit/stage architecture. These eight comparators have threshold voltages of $\pm 7V_{ref}/8$, $\pm 5V_{ref}/8$, $\pm 3V_{ref}/8$, and $\pm V_{ref}/8$. Each input falls into a certain range and has a corresponding digital output. The digital output code is shown in the upper part of Figure 6-4 (b). For example, if the input is 0, this input falls into the $-V_{ref}/8$ and $V_{ref}/8$ range. Therefore, the digital output code is 0100. The coded range of the proposed configuration is from $-9V_{ref}/8$ to $9V_{ref}/8$. The extended input range allows the output range to be extended to the same range as input without a code loss. Therefore, the output range is also from $-9V_{ref}/8$ to $9V_{ref}/8$.

The dither-based transfer characteristics of the 2.5-bit/stage architecture are shown in Figure 6-5. As described in Section 6.1, D_a is the output word of the sub-ADC without the dither. $D_a \in \{-V_{ref}, -3V_{ref}/4, -V_{ref}/2, -V_{ref}/4, 0, V_{ref}/4, V_{ref}/2, 3V_{ref}/4, V_{ref}\}$ correspond to the centers of each input domain. In this case, the sum of the signal plus dither is within the $-V_{ref}$ to V_{ref} range. Since the output range is from $-9V_{ref}/8$ to $9V_{ref}/8$, this leaves the redundancy space of $\pm V_{ref}/8$ for comparator offsets.

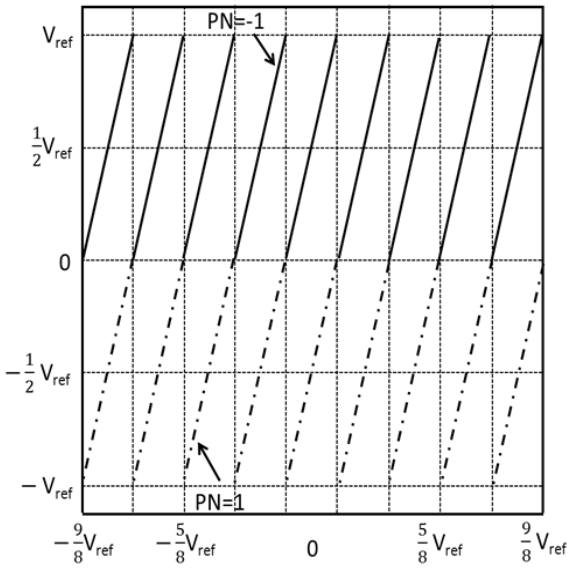
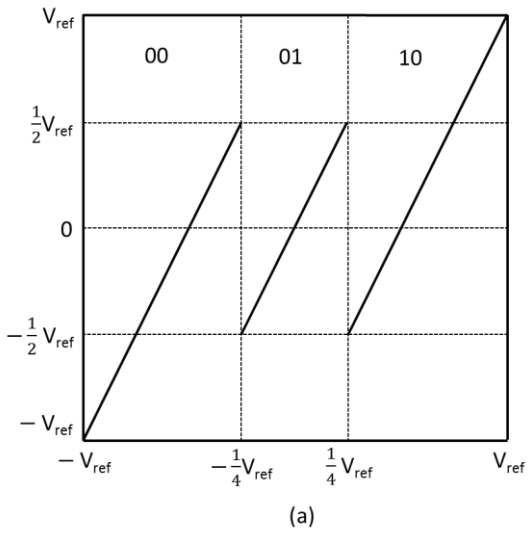


Figure 6-5. Dither-based transfer characteristics of the domain-extended 2.5-bit/stage ADC

The remaining pipeline stages are domain-extended 1.5-bit/stage ADC with a gain of two. The transfer characteristics of a traditional 1.5-bit/stage ADC are shown in Figure 6-6 (a). The threshold voltages are $-V_{ref}/4$ and $V_{ref}/4$. The input range is from $-V_{ref}$ to V_{ref}. Since the output of the current stage is the input of the following stage of the pipeline ADC, the output range needs to be the same as the input range to avoid a code loss. Therefore, like the input range, the output range is also from $-V_{ref}$ to V_{ref}. In contrast, the modified transfer characteristics of a domain-extended 1.5-bit/stage ADC are shown in Figure 6-6 (b). After two threshold voltages are added, the threshold voltages are $-3V_{ref}/4$, $-V_{ref}/4$, $V_{ref}/4$, and $3V_{ref}/4$. The input range is from $-5V_{ref}/4$ to $5V_{ref}/4$. Like in the traditional architecture, in order to prevent the ADC from a code loss in the modified architecture, the output range needs to be the same as the input range: from $-5V_{ref}/4$ to $5V_{ref}/4$. The performance of the ADC is limited by these 1.5-bit stages when the calibration is only applied to the first two 2.5-bit stages. For this reason, gain error calibration is also performed on the first two of the 1.5-bit stages. At the same time, comparator offset corrections are applied to all 1.5-bit stages.



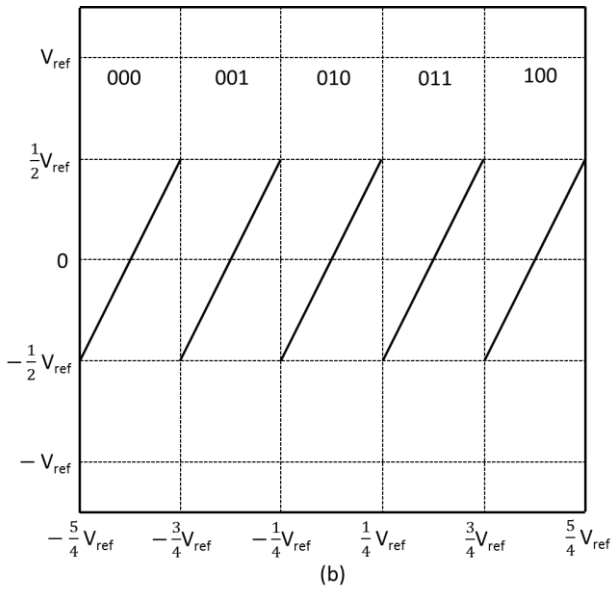


Figure 6-6. Transfer characteristics of (a) the traditional 1.5-bit/stage ADC and (b) the domain-extended 1.5-bit/stage ADC

Figure 6-7 shows the transfer characteristics with dithering. As shown in Figure 6-6 (b) the code range is from -5V_{ref}/4 to 5V_{ref}/4. In this case, $D_a \in \{-V_{ref}, -V_{ref}/2,\ 0,\ V_{ref}/2, V_{ref}\}$ correspond to the centers of each input domain. In this case, it leaves V_{ref}/4 redundancy for comparator offsets. A lookup table can be generated by a background multiplier to implement the gain calibration [40] [48]. This requires high resolution adders for pipeline stages.

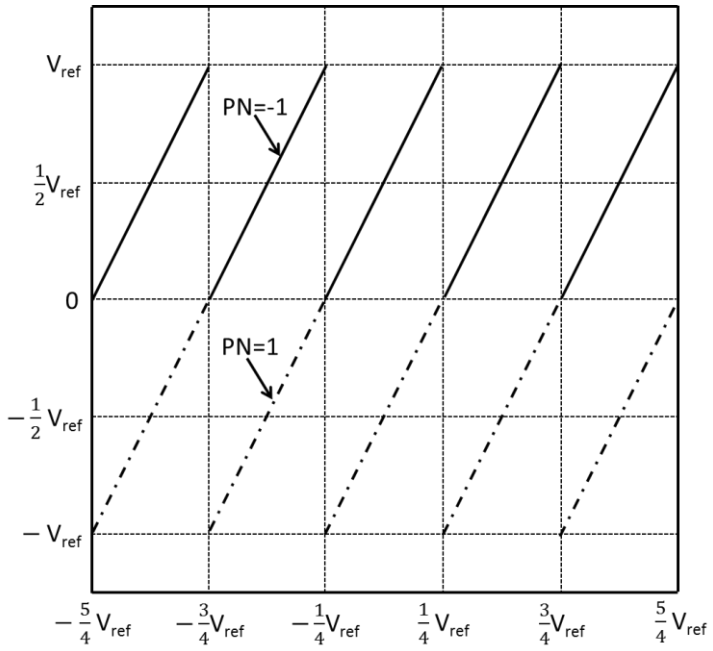


Figure 6-7. Dither-based transfer characteristics of the domain-extended 1.5-bit/stage ADC

The transfer function of a domain-extended 1.5-bit/stage pipeline ADC is given by the following equation [25]

$$V_{\text{out}} = \begin{cases} 2V_{in} + 2V_{\text{ref}}, & \text{D=000} \\ 2V_{in} + 1V_{\text{ref}}, & \text{D=001} \\ 2V_{in} + 0V_{\text{ref}}, & \text{D=010} \\ 2V_{in} - 1V_{\text{ref}}, & \text{D=011} \\ 2V_{in} - 2V_{\text{ref}}, & \text{D=100} \end{cases} \tag{6.15}$$

As shown in Equation (6.15), the gain of two for V_{ref} is required. However, the gain of two for V_{ref} cannot be realized through the traditional method, since one of the sample-and-hold capacitors has to be used as the feedback capacitor. The maximum gain allowed for V_{ref} using the traditional method is one. In order to avoid an additional reference voltage to provide the gain of two for V_{ref}, V_{ref} is set to be half of V_{dd}. Here, V_{dd} is the power source voltage. The circuit level realization of the above equation is given by

$$V_{\text{out}} = \begin{cases} \frac{C_s+C_f}{C_f}V_{in} + \frac{C_s}{C_f}V_{dd}, & \text{D=000} \\ \frac{C_s+C_f}{C_f}V_{in} + \frac{C_s}{C_f}V_{\text{ref}}, & \text{D=001} \\ \frac{C_s+C_f}{C_f}V_{in} + \frac{C_s}{C_f}GND, & \text{D=010} \\ \frac{C_s+C_f}{C_f}V_{in} - \frac{C_s}{C_f}V_{\text{ref}}, & \text{D=011} \\ \frac{C_s+C_f}{C_f}V_{in} - \frac{C_s}{C_f}V_{dd}, & \text{D=100} \end{cases}, \tag{6.16}$$

where C_s and C_f are sample-and-hold capacitors shown in Figure 6-8.

The first and the last sub-equations of Equation (6.16) are now

$$\frac{C_s+C_f}{C_f}V_{in} + \frac{C_s}{C_f}V_{dd} = 2 \cdot V_{in} + 1 \cdot 2 \cdot V_{ref}, \tag{6.17}$$

$$\frac{C_s+C_f}{C_f}V_{in} - \frac{C_s}{C_f}V_{dd} = 2 \cdot V_{in} - 1 \cdot 2 \cdot V_{ref}. \tag{6.18}$$

Figure 6-8 (a) and (b) show the circuit configurations of the traditional 1.5-bit/stage ADC and the proposed domain-extended 1.5-bit/stage ADC, respectively. The added reference voltage is provided by the power source V_{dd} as shown in the dashed areas. Also, for the proposed 2.5-bit/stage converter, the realization of the gain of four for V_{ref} is similar to the 1.5-bit/stage ADC configuration.

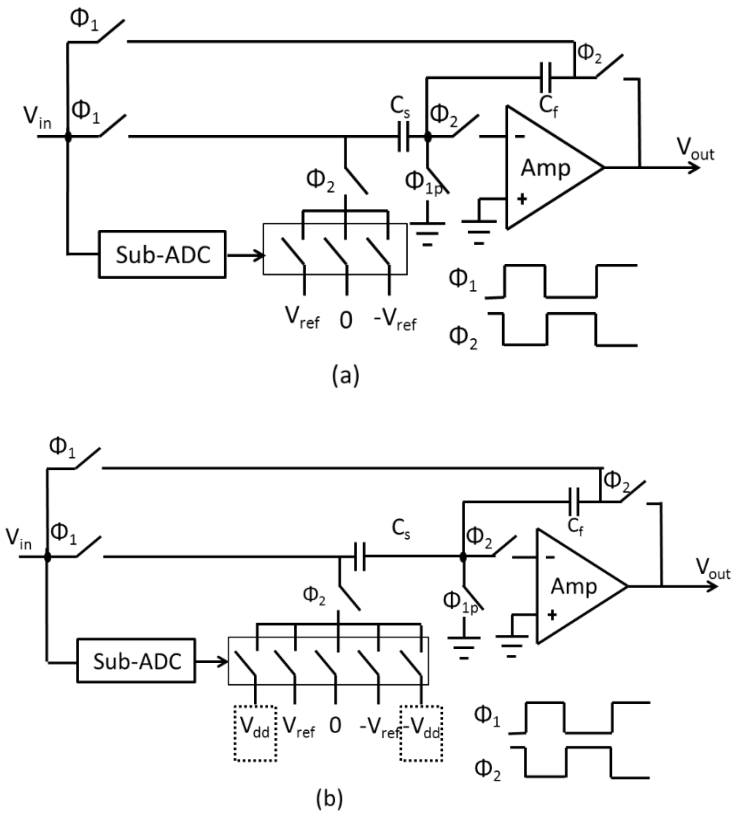


Figure 6-8. Circuit configurations of (a) the traditional 1.5-bit/stage ADC and (b) the domain-extended 1.5-bit/stage ADC

6.3. Simulation results using the domain-extended dither-based algorithm

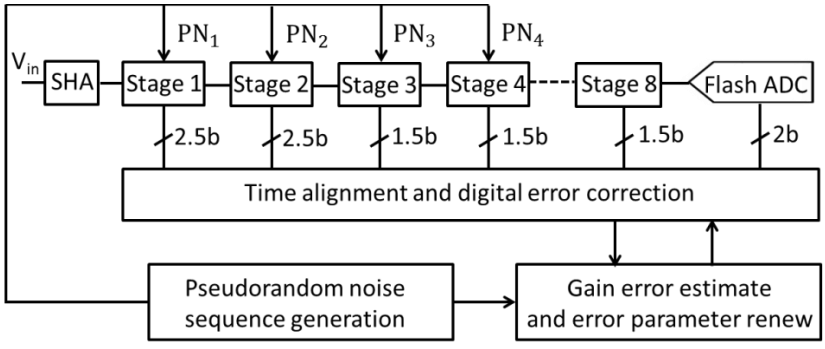


Figure 6-9. Block diagram of the system architecture

To demonstrate the effectiveness of the proposed ADC calibration algorithm, a 12-bit pipeline ADC is simulated in MATLAB. The ADC consists of two 2.5-bit/stage converters followed by six 1.5-bit/stage converters and a 2-bit flash ADC with the configuration as the simplified system block diagram shown in Figure 6-9. Gain errors, which include capacitor mismatches and finite gains, are calibrated on the first four stages. Comparator offsets are corrected on all stages. All stages have gain errors chosen between 2% ~5% and comparator offsets chosen between 2% ~5%. The initial calibration coefficient m_{est}, is set to be the standard value the stage. Full-scale sinusoid is used as the test input.

Figure 6-10 shows the convergence of the error coefficient $e = \frac{(m_{est}-m)}{m}$ for the 12-bit pipeline ADC. As stated in Section 6.1, μ is the step size of the LMS iterative. When μ is large, the convergence has large steady-state errors. On the other hand, when μ is small, the convergence speed is slow. Therefore, the convergence time is different for different step sizes. In this case, the step size μ equals 0.00001. The calibrations based on the proposed algorithm and the conventional algorithm are simulated under the same conditions. Using the proposed algorithm, convergence is reached after approximately 8×10^5 conversion samples. At a conversion rate of 100MS/s, the convergence time is approximately 8 ms. However, using the conventional algorithm with the same redundancy space as the proposed one, convergence is reached after approximately 14×10^5 conversion samples. At a conversion rate of 100MS/s, the convergence time is approximately 14 ms. Therefore, the calibration based on the proposed algorithm has a faster convergence speed than the calibration based on the conventional algorithm.

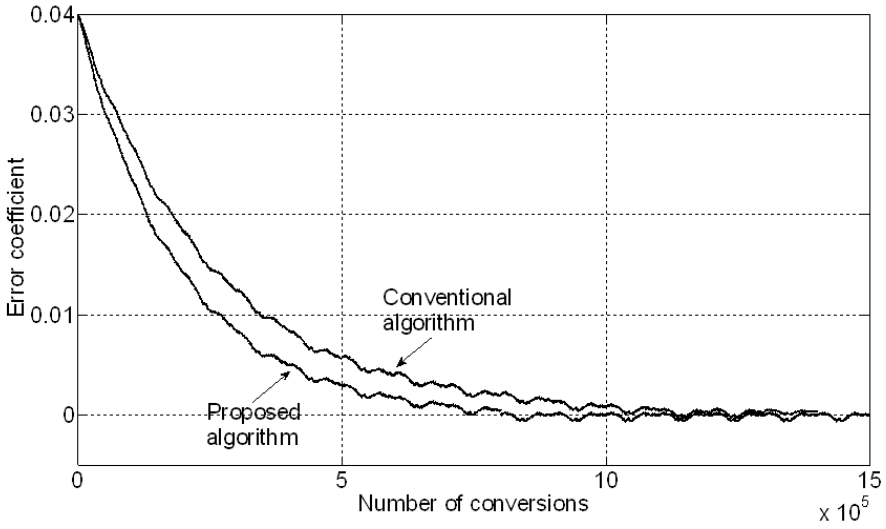
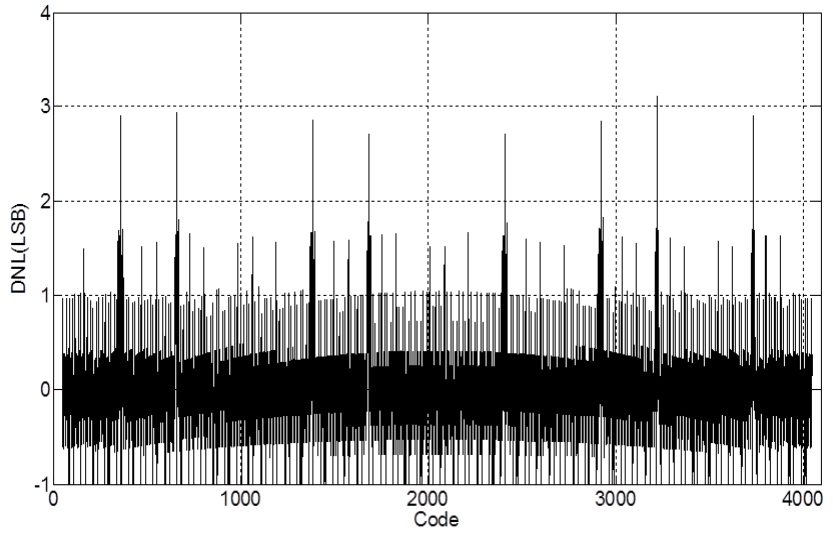
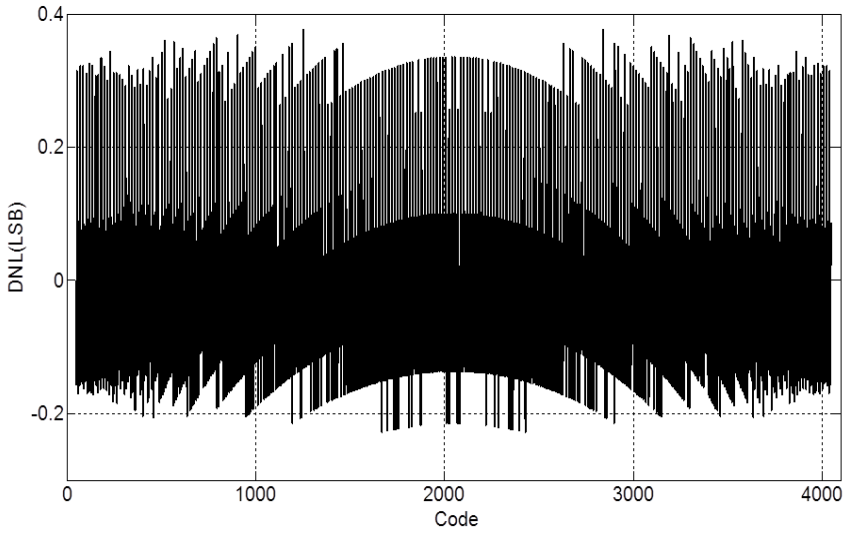


Figure 6-10. Convergence of the error coefficient for the 12-bit converter

Figure 6-11 and Figure 6-12 show the DNL and the INL, respectively, of the 12-bit converter without the digital calibration (a) and with the digital calibration (b). A full-scale sinusoidal input with slow input frequency is used to test the static performance. Assisted by the proposed algorithm, the peak DNL is reduced from 3.1 LSB to 0.4 LSB, while the peak INL is reduced from 16.6 LSB to 0.3 LSB.



(a)



(b)

Figure 6-11. 12-bit ADC DNL: (a) without calibration and (b) with calibration

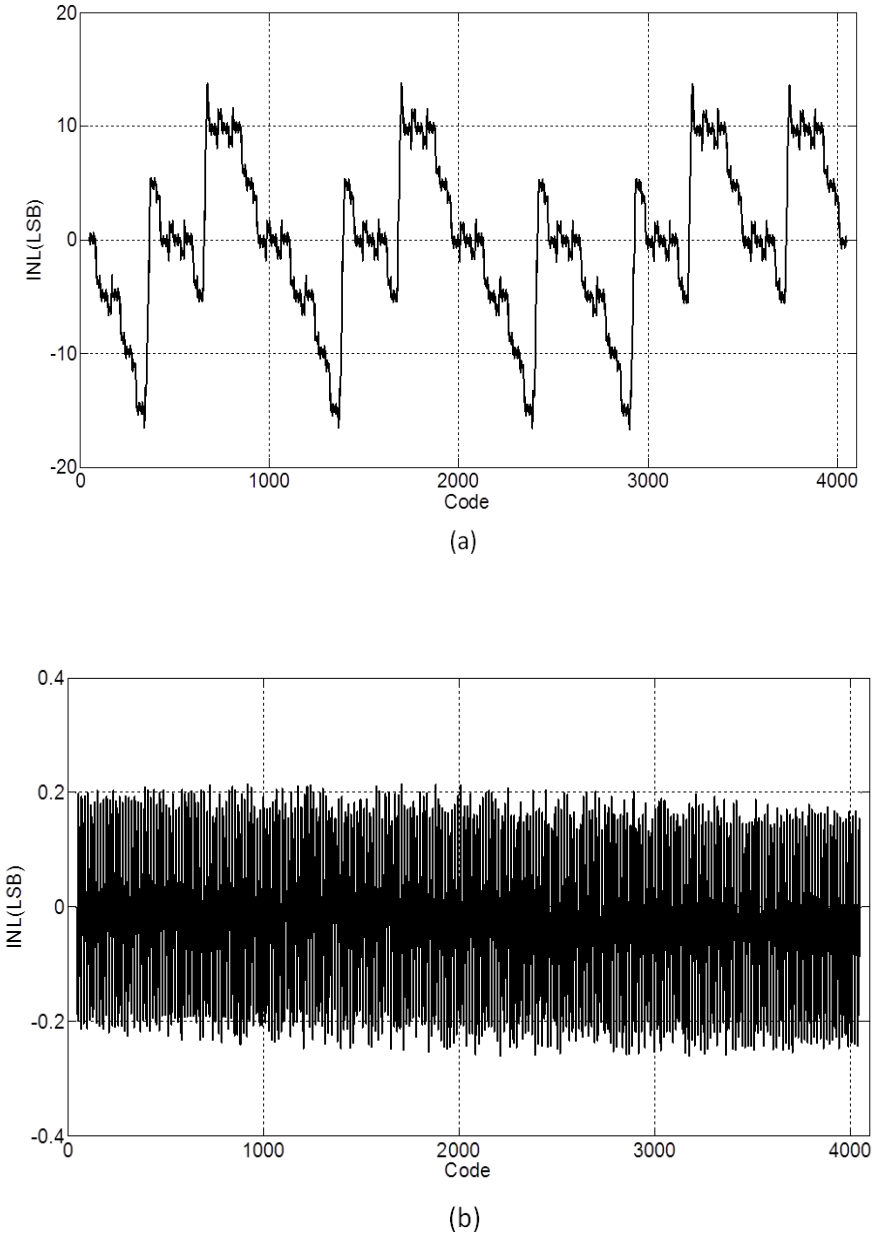


Figure 6-12. 12-bit ADC INL: (a) without calibration and (b) with calibration

Simulation results for the dynamic performance are shown in Figure 6-13. Full-scale sinusoidal input of 45-MHz and the sample rate of 100-MHz are used to test the dynamic performance. Figure 6-13 (a) shows the output spectrum without the digital calibration enabled; the SFDR is 57.69 dB, while the SNR is 47.92 dB, which corresponds to an ENOB of 7.61. Figure

6-13 (b) shows the output spectrum with the digital calibration enabled; the noise floor has been greatly lowered. In the calibrated case, the SFDR is improved to 100.55 dB and SNR to 73.23 dB. Performance characters of both cases in steady state are summarized in Table 6-1.

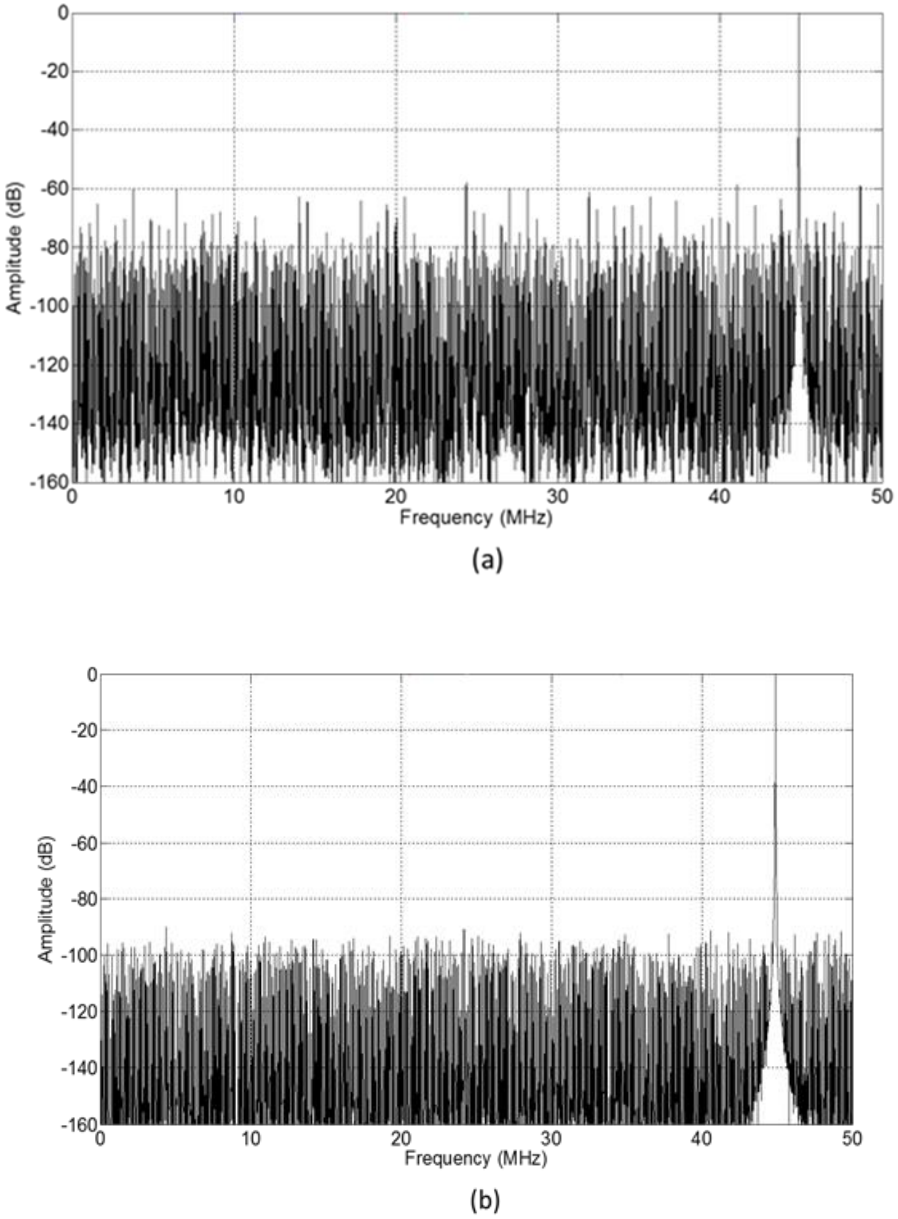


Figure 6-13. 12-bit ADC output spectrum: (a) without calibration and (b) with calibration

Table 6-1. Performance of the 12-bit ADC without and with calibration

Parameter	Without calibration	With calibration
Sampling rate(MS/s)	100	100
Resolution(bit)	12	12
DNL (LSB)	-1~3.1	-0.3~0.4
INL (LSB)	-16.6~13	-0.3~0.3
SFDR(dB)	57.69	100.55
SNR(dB)	47.62	73.23
THD(dB)	56.04	97.48
ENOB(bit)	7.61	11.87

A digital background calibration algorithm has been described that corrects linear errors, which include capacitor mismatches and finite op-amp open-loop gains. The existence of the digital redundancy bits of the pipeline ADC corrects the comparator offsets. In the traditional dither-based calibration, the amplitude of the sum of the signal plus dither has to be within a limited range, in order to ensure the redundancy space for comparator offsets to prevent code loss. However, the reduction of the amplitude of the sum of the signal plus dither results in either reduction of the signal amplitude or reduction of the dither amplitude. The reduction of the signal amplitude leads to the declining of SNR of the pipeline ADC. In contrast, the reduction of the dither amplitude results in the increasing of the convergence time. The proposed domain-extent dither-based algorithm relieves these two conflicting issues, as well as ensures the redundancy space for comparator offsets.

6.4. Variable-amplitude domain-extended dither-based algorithm

In order for a digital calibration to be useful, the convergence speed has to be fast. As stated in Section 6.3, the calibration using the domain-extended dither-based algorithm increases the convergence speed compared to the calibration using the traditional dither-based algorithm. In order to further increase the convergence speed, a variable-amplitude domain-extended dither-based algorithm is proposed [49]. The amplitude of the dither is variable according to the signal level. In this case, the amplitude of the dither is greatly increased without a reduction of the amplitude of the signal; therefore, the convergence speed is much faster without harming the SNR of the ADC. Moreover, the existence of redundancy space in the domain-extended architecture allows for comparator offsets.

The domain-extended 1.5-bit/stage ADC allows for more space for comparator offsets, for higher amplitude of the signal, or for higher amplitude of the dither, as compared to the traditional 1.5-bit/stage ADC. For domain-extended 1.5-bit/stage ADC, the total amplitudes of the signal and the dither can be set within the $-V_{ref}$ to V_{ref} range. Therefore, some space is available for comparator offsets. However, in contrast, the traditional 1.5-bit/stage ADC needs the total amplitudes of the signal and the dither to be set within the range of $-0.8V_{ref}$ to $0.8V_{ref}$ in order to have redundancy space for comparator offsets. The reduction of the total amplitudes of the signal and the dither results in either a decline of SNR or a slower convergence speed.

The transfer characteristics of the domain-extended 1.5-bit/stage ADC with variable-amplitude dithering are shown in Figure 6-14 (a). Since the transfer characteristics are periodic, each domain has the similar dither injection method as the other domains. Therefore, for the sake of clarity, a variable-amplitude dither injected into one domain is analysed. For example, the variable-amplitude dithers are injected between $-3V_{ref}/4$ and $-V_{ref}/4$, and three more comparators

with threshold voltages equal to $-5V_{ref}/8$, $-4V_{ref}/8$, and $-3V_{ref}/8$ are added. A dither amplitude is chosen from $-5V_{ref}/4$, $-4V_{ref}/4$, $-3V_{ref}/4$, $-2V_{ref}/4$, $2V_{ref}/4$, $3V_{ref}/4$, and $4V_{ref}/4$ depending on the PN value and the signal levels as shown in Figure 6-14 (a). The total amplitudes of the signal and the dither are within $\pm V_{ref}$, and so have redundancy space for comparator offsets. Therefore, the signal pulse dither between $-3V_{ref}/4$ and $-V_{ref}/4$ is, in effect, a constant-amplitude dither of $\pm 7V_{ref}/8$ with a small signal within the range of $\pm V_{ref}/8$, as shown in Figure 6-14 (b). The actual signal stays unchanged, which means the SNR has not declined. The dither amplitude is greatly increased, which leads to a fast convergence speed.

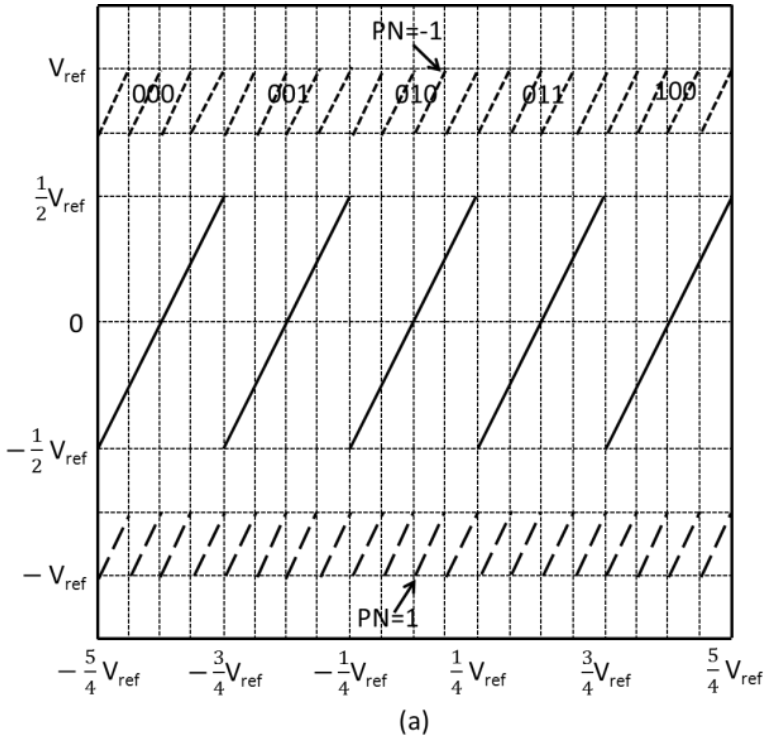
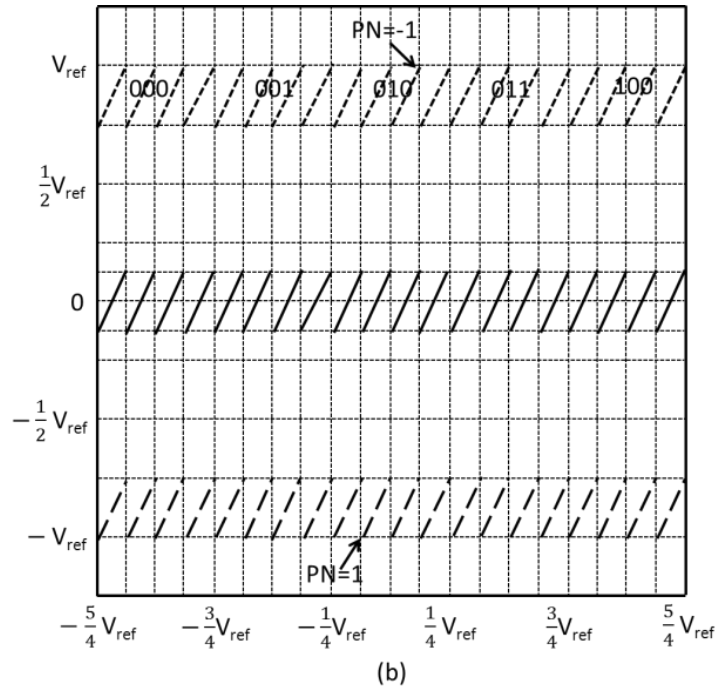
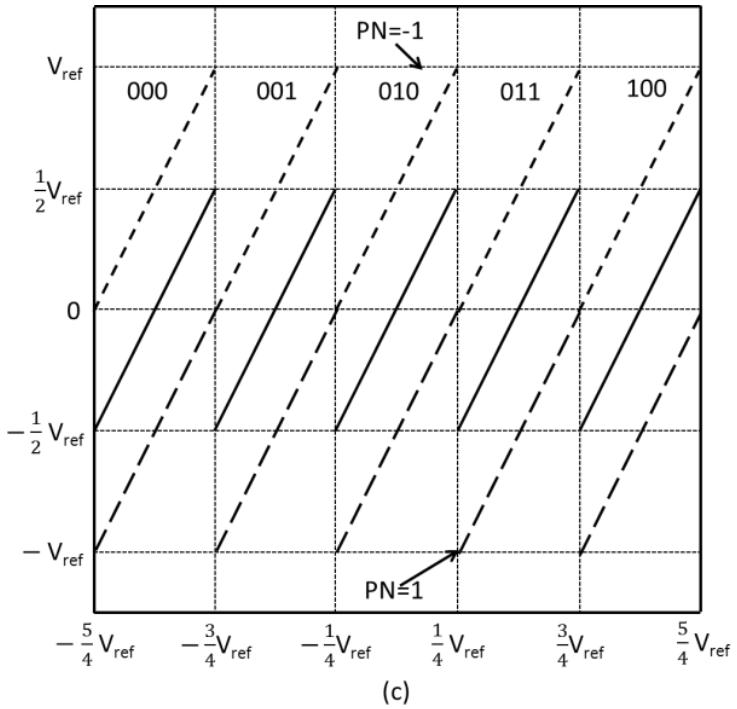


Figure 6-14. (a) Transfer characteristics of the domain-extended 1.5-bit/stage ADC with variable-amplitude dithering, (b) equivalent constant-amplitude PN dithering of (a), and (c) transfer characteristics of the domain-extended 1.5-bit/stage ADC with constant-amplitude dithering



(b)



(c)

Figure 6-14. (a) Transfer characteristics of the domain-extended 1.5-bit/stage ADC with variable-amplitude dithering, (b) equivalent constant-amplitude PN dithering of (a), and (c) transfer characteristics of the domain-extended 1.5-bit/stage ADC with constant-amplitude dithering (continued)

However, for the domain-extended 1.5-bit/stage ADC with constant-amplitude dithering, the dither amplitude is only $V_{ref}/2$ as shown in Figure 6-14 (c). As mentioned above, in the case of one signal level divided into four sub-levels, the dither amplitude is increased from $V_{ref}/2$ to $7V_{ref}/8$ without reducing the signal amplitude. Therefore, the variable-amplitude dithering calibration allows higher amplitude of the dither without the decline of SNR.

In dither-based architecture shown in previous studies [45] [38], when the signal stays at a high level all the time, more dither cannot be added. Therefore, the error coefficient e cannot be extracted in the digital domain according to Equation (6.12), (6.13), and (6.14). This is because the first term of Equation (6.12) equals zero after an infinite number of cumulative sums and the average is calculated. The second term always equals zero because ΔS equals zero. Therefore, the LMS iterative is not applicable for this condition. This circumstance constrains the application. However, the proposed variable-amplitude dithering does not suffer from the limitation. The proposed variable-amplitude dithering also offers substantial savings in convergence times, even though the signal stays at high level at all times.

These concepts are realized on the circuit level in the following concrete example. A MDAC works in two phases: the sample phase and the amplification phase. During the sample phase, the sample input signal is administered to all capacitors. Figure 6-15 shows how the MDAC works in the amplification phase with a signal range from $-3V_{ref}/4$ to $-V_{ref}/4$. Variable-amplitude dithering is realized by adding three more comparators and splitting a capacitor into four capacitors, C_1, C_2, C_3, and C_4 with each capacitor valued at $C_f/4$. The amplitudes of the dithers, in all sub-domains of the main domain from $-3V_{ref}/4$ to $-V_{ref}/4$, are shown in Table 6-2. Dither injections are controlled through switches dependant on both the output of the encoder and the PN value.

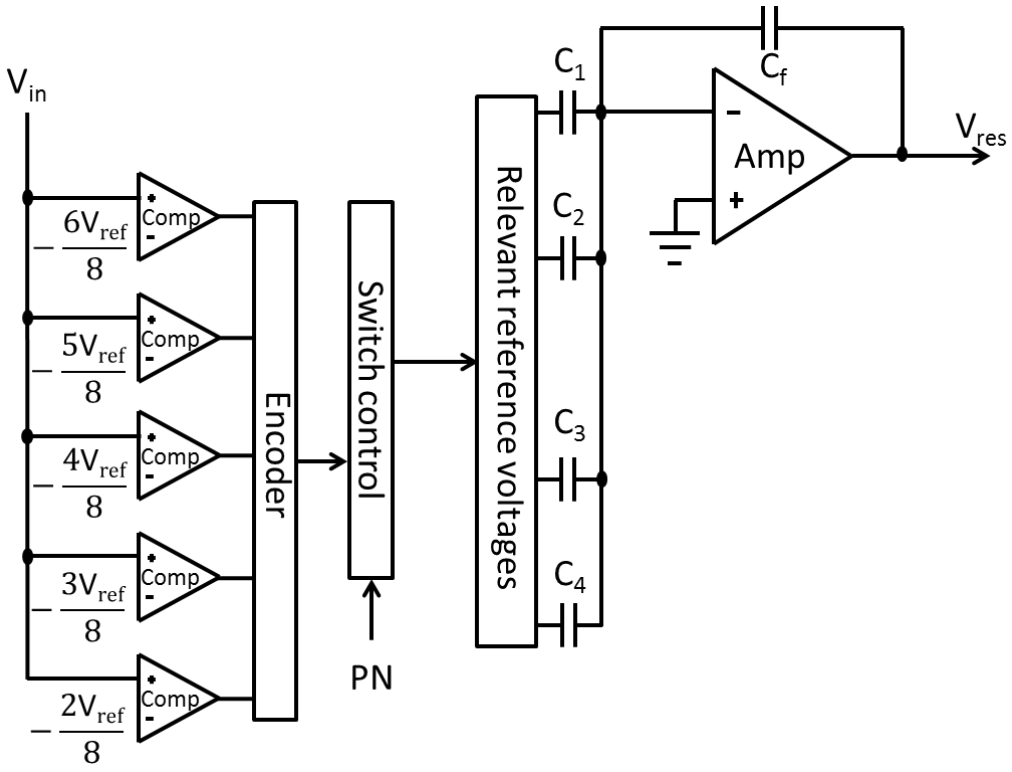


Figure 6-15. MDAC architecture in amplification phase with the signal range from $-3V_{ref}/4$ to $-V_{ref}/4$

Table 6-2. Amplitude of the dithers in all sub-domains of the main domain from $-3V_{ref}/4$ to $-V_{ref}/4$

V_{in}		Amplitude of dithers	
Main level	Sub-levels	PN=-1	PN=1
$-\frac{3}{4}V_{ref} \sim -\frac{1}{4}V_{ref}$	$-\frac{3}{4}V_{ref} \sim -\frac{5}{8}V_{ref}$	$\frac{5}{4}V_{ref}$	$-\frac{1}{2}V_{ref}$
	$-\frac{5}{8}V_{ref} \sim -\frac{1}{2}V_{ref}$	V_{ref}	$-\frac{3}{4}V_{ref}$
	$-\frac{1}{2}V_{ref} \sim -\frac{3}{8}V_{ref}$	$\frac{3}{4}V_{ref}$	$-V_{ref}$
	$-\frac{3}{8}V_{ref} \sim -\frac{1}{4}V_{ref}$	$\frac{1}{2}V_{ref}$	$-\frac{5}{4}V_{ref}$

6.5. Simulation results using the variable-amplitude domain-extended dither-based algorithm

To illustrate the effectiveness of the calibration using the variable-amplitude domain-extended dither-based algorithm, a 12-bit pipeline ADC is simulated in Matlab. This ADC consists of ten 1.5-bit/stage converters followed by a 2-bit flash ADC with digital error correction, digital calibration, and PN sequence generation as shown in Figure 6-16. The gain errors and comparator offsets in all stages are chosen between 2% ~ 5%. The initial calibration coefficient m_{est} set to be its standard value, which is m_{est}=0.5. Full-scale sinusoid is used as the test input. The gain calibration is applied to the first four stages, and the comparator offset corrections are applied to all 1.5-bit/stage converters.

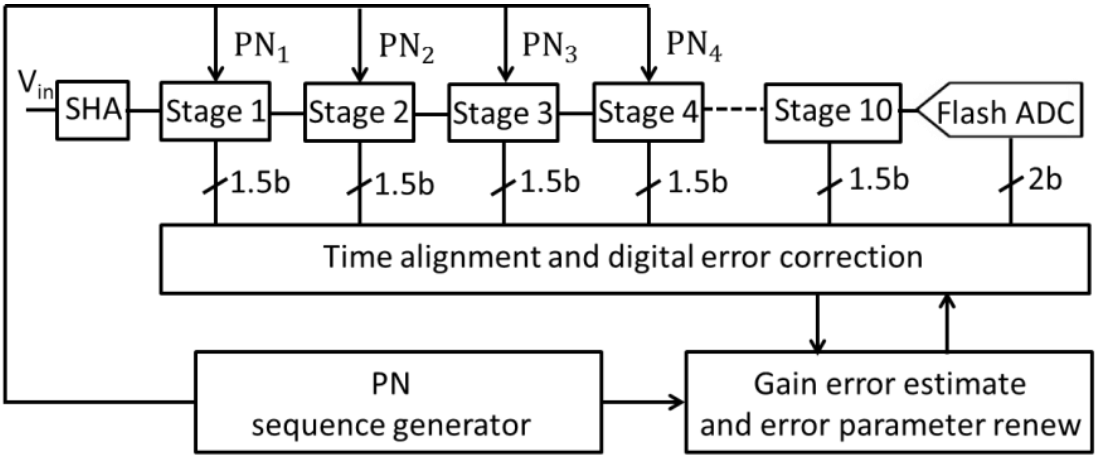


Figure 6-16. Block diagram of the system architecture

The convergence of the error coefficient $e = \frac{m_{est}-m}{m}$ for a 12-bit pipeline ADC using the conventional dither-based algorithm, the domain-extended dither-based algorithm, and the variable-amplitude domain-extend dither-based algorithm are shown in Figure 6-17.

As stated in Section 6.1, μ is the step size of the LMS iterative. When μ is large, the convergence has large steady-state errors. On the other hand, when μ is small, the convergence speed is slow. Therefore, the convergence time is different for the various step sizes. In this case, the step size μ equals 0.00001. The calibrations based on these three algorithms are simulated under the same conditions. The redundancy space left for comparator offsets in these three simulations are all $V_{ref}/4$. Different ADC structures also have the different convergence speed; therefore, it is more precise to compare these three algorithms using the same ADC structure than to compare the proposed algorithm shown in this section with the one shown in Section 6.3. Using the conventional algorithm, convergence is reached after approximately 17×10^5 conversion samples. At a conversion rate of 100MS/s, the convergence time is approximately 17 ms. In addition, using the domain-extended dither-based algorithm, convergence is reached after approximately 9×10^5 conversion samples. At a conversion rate of 100MS/s, the convergence time is approximately 9 ms. However, using the variable-amplitude domain-extended dither-based algorithm, convergence is reached after approximately 7×10^5 conversion samples. At a conversion rate of 100MS/s, the convergence time is approximately 7 ms. Also, as can be seen in Figure 6-17, the steady-state error for the first two cases are larger than the case using the variable-amplitude domain-extended dither-based algorithm. However, in order to realize a similar steady-state error as that of the case using the variable-amplitude domain-extended dither-based algorithm, the first two cases require a much smaller step size μ. As mentioned before, the decrease of the step size increases the convergence time. Therefore, the calibration using the variable-amplitude domain-extended dither-based algorithm has a faster convergence speed than the calibration using the domain-extended dither-based algorithm.

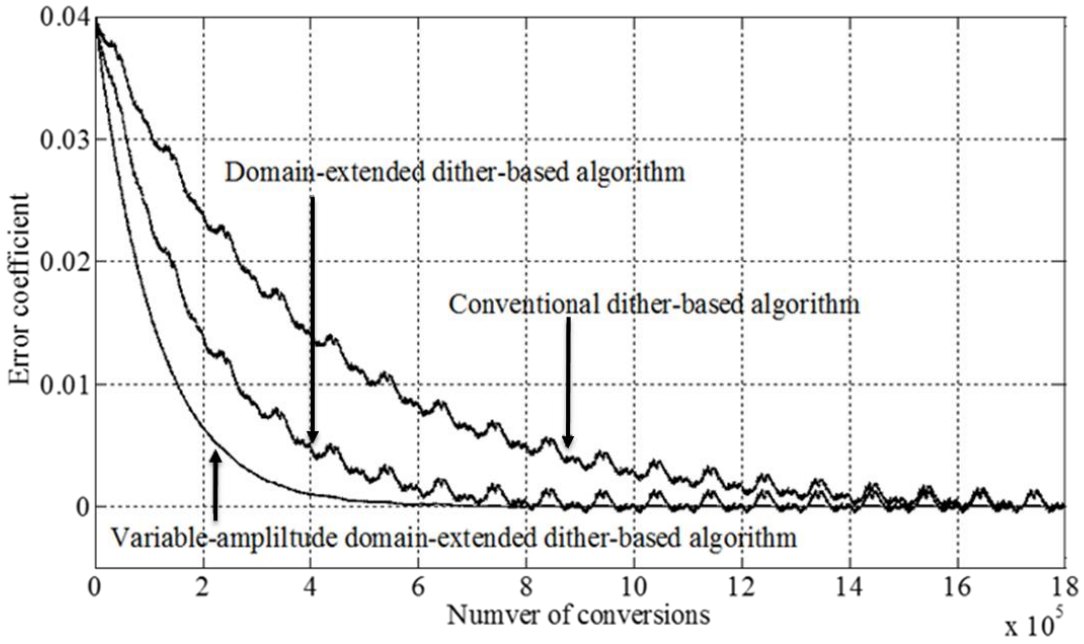


Figure 6-17. Convergence of the error coefficient

A full-scale sinusoidal input with low frequency is used to test the static performance. Figure 6-18 (a) and Figure 6-18 (b) show the DNL, of a 12-bit pipeline ADC, without calibration and with calibration, respectively. The peak DNL is reduced from 3.1 LSB in the un-calibrated case to 0.5 LSB in the calibrated case. Figure 6-19 (a) and Figure 6-19 (b) show the INL without calibration and with calibration, respectively. The peak INL is reduced from 31 LSB in the un-calibrated case to 0.5 LSB in the calibrated case.

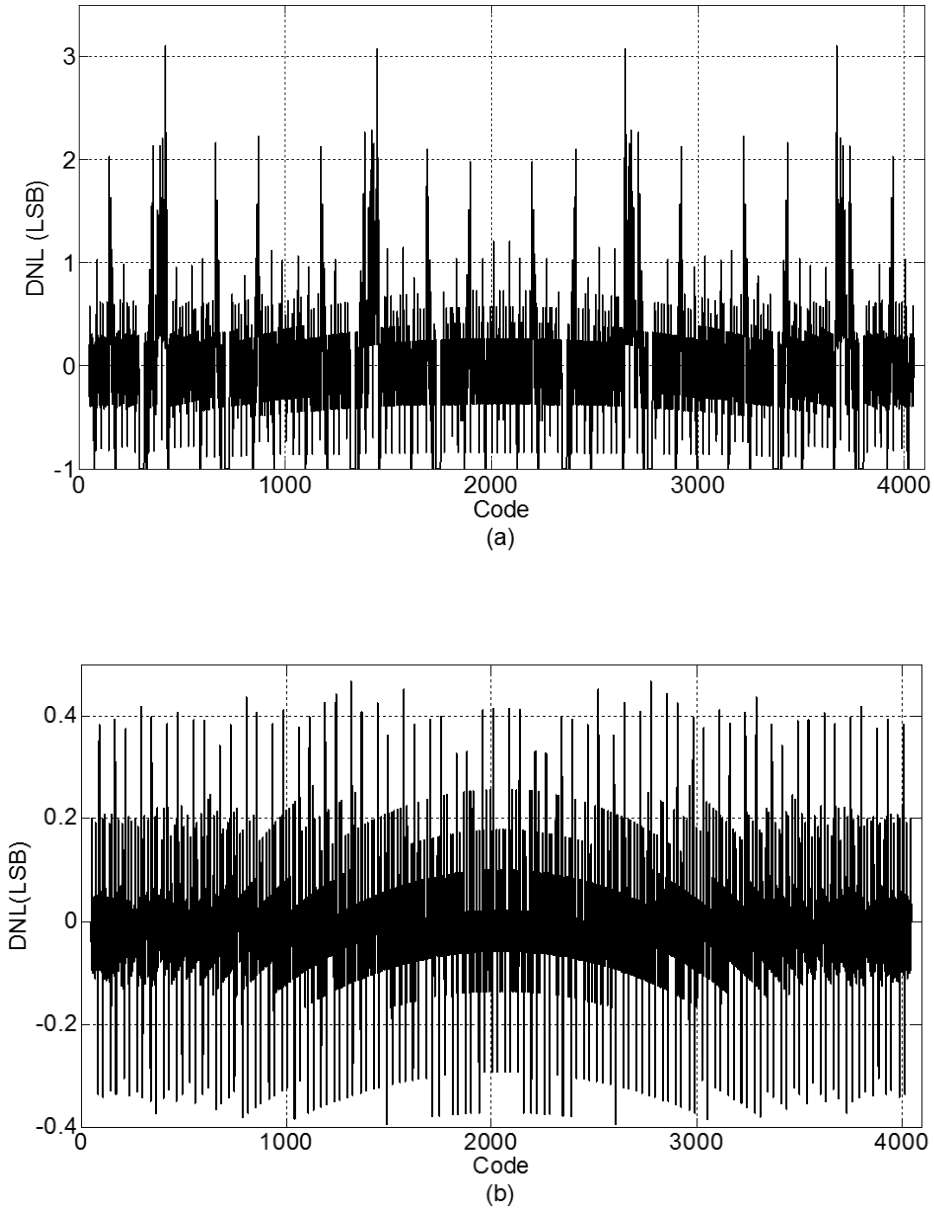


Figure 6-18. 12-bit ADC DNL: (a) without calibration and (b) with calibration

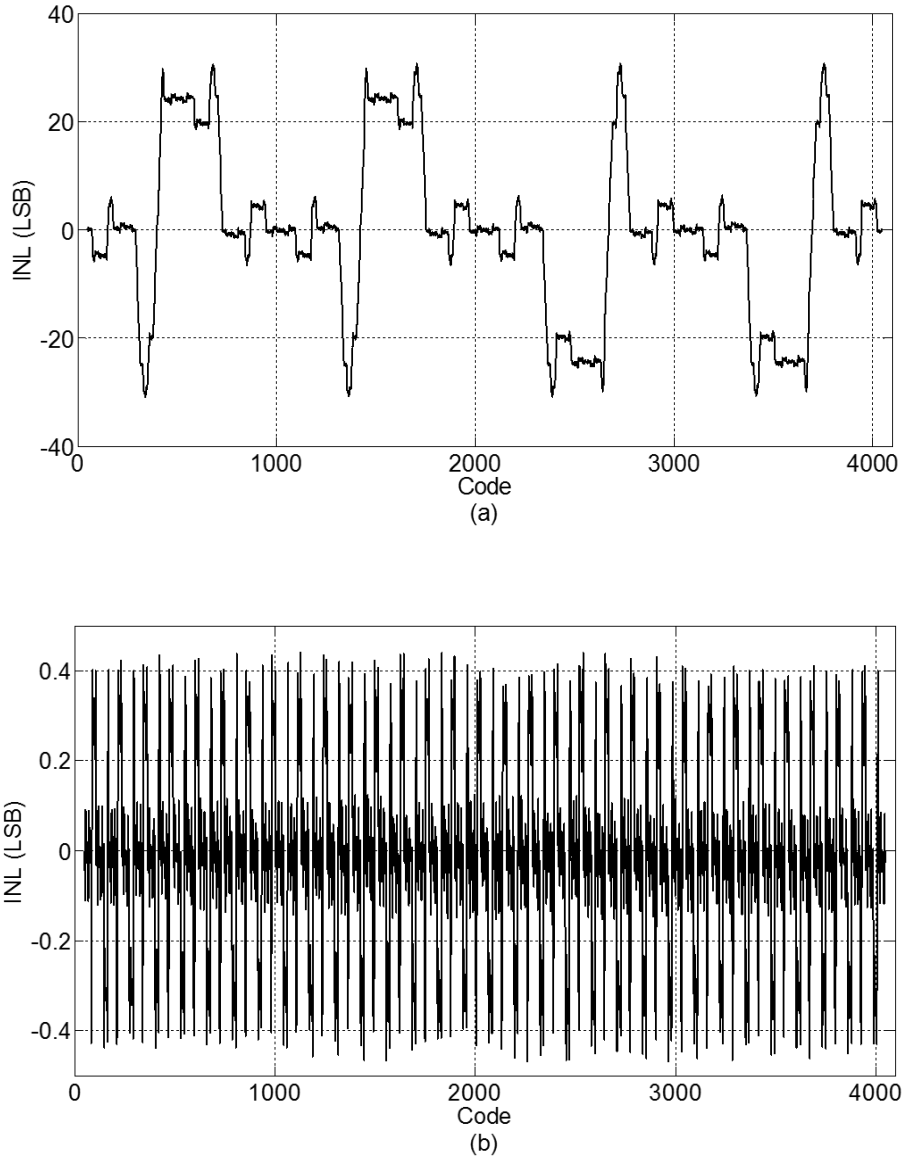


Figure 6-19. 12-bit ADC INL: (a) without calibration and (b) with calibration

A full-scale sinusoidal with an input frequency of 45 MHz and a sample-rate of 100 MS/s are used to test the dynamic performance. The output spectrum of the ADC without calibration is shown in Figure 6-20 (a). The SFDR in this un-calibrated case is 51 dB, while the SNR is only 40.4 dB, which corresponds to an ENOB of 6.4. Also, in this case, the THD is 51 dB. The output spectrum of the ADC with the proposed calibration is shown in Figure 6-20 (b). The tone is

reduced, and the SFDR is 87.7 dB. Also the noise floor is lowered, which shows a 69.3 dB SNR with an ENOB of 11.2. The THD is 86.2 dB in the calibrated case. Table 6-3 shows the performance of the 12-bit ADC without calibration and with calibration.

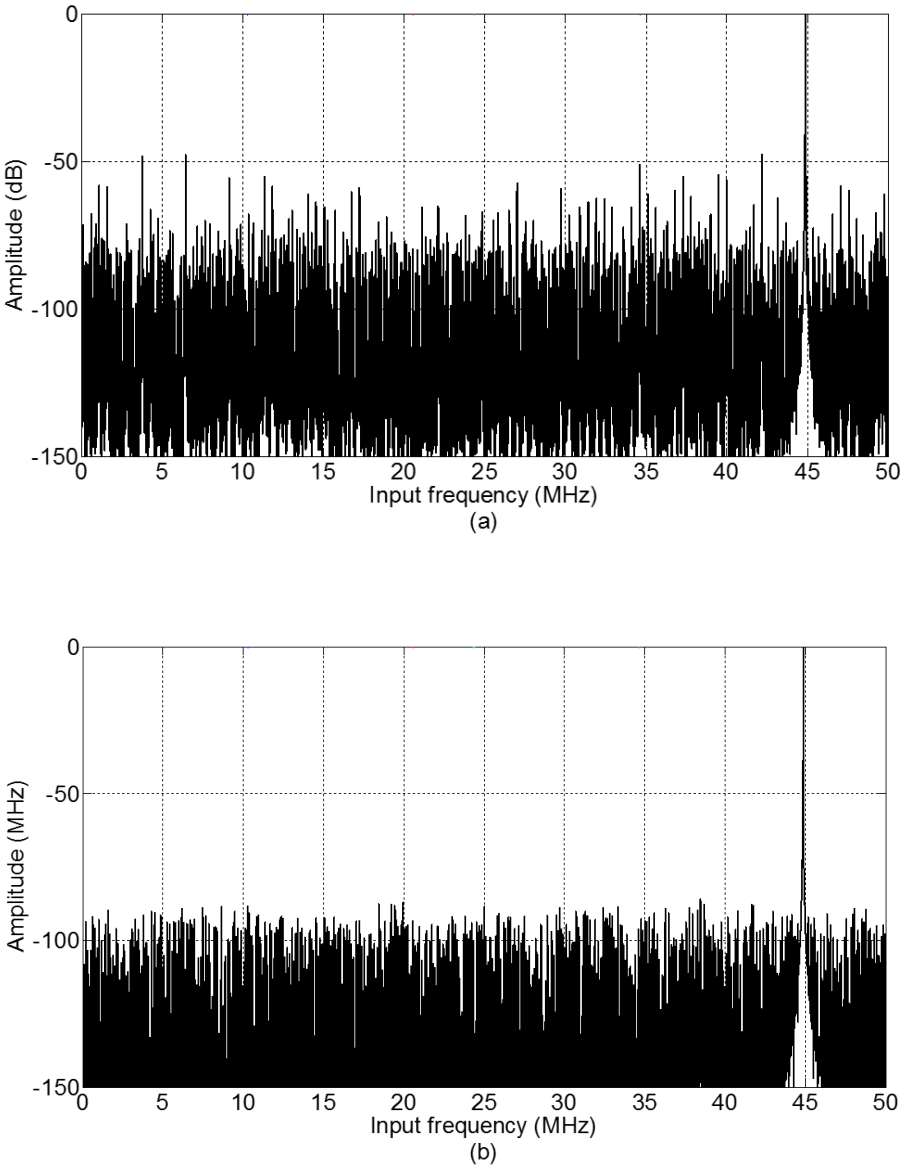


Figure 6-20. 12-bit ADC output spectrum: (a) without calibration and (b) with calibration

Table 6-3. Performance of the 12-bit ADC without and with calibration

Parameter	Without calibration	With calibration
Sample- rate(MS/s)	100	100
Resolution(bit)	12	12
DNL (LSB)	-1~3.1	-0.4~0.5
INL (LSB)	-31~30.8	-0.5~0.4
SFDR(dB)	51	87.7
SNR(dB)	40.4	69.3
THD(dB)	51	86.2
ENOB(bit)	6.4	11.2

A digital calibration using variable-amplitude domain-extended dither-based algorithm is applied to a pipeline ADC. This algorithm allows for large average amplitude dither injections without scarifying the signal amplitude, which means fast convergence speed while keeping the SNR not changed. For the dither-based calibrations [45] [38], the calibrations cannot proceed if the signal stays at high level all the time. This constrain limits the application. The introduction of domain-extended architecture allows the digital calibration to proceed even though the signal stays at high level all the time. Also, the redundancy space plus the total amplitude of the signal and the dither is limited by the quantify range. The domain-extended architecture extends the quantify range, which results in higher SNR, faster convergence speed, or more redundancy space for comparator offsets. The simulated ADC demonstrates good dynamic and static performance within a practical time without paying significant penalty in the circuit complexity.

6.6. Conclusion

A high resolution ADC can be achieved without digital calibration. Therefore, in order for a digital calibration to be useful, it should minimize the analog circuits as well as have a reasonable convergence time. The reduced accuracy due the minimized analog circuit can be complemented by the digital calibration. Hence, the convergence time determines the digital calibration's quality. This chapter presents two algorithms to reduce the convergence time. According to the simulation results, the domain-extended dither-based algorithm reduces the convergence time compared to the conventional dither-based algorithm. Based on the domain-extended dither-based algorithm, another algorithm, a variable-amplitude domain-extended dither-based algorithm, has been developed. The simulation results for this second algorithm show that the variable-amplitude domain-extended dither-based algorithm has an even faster convergence speed than the previous developed domain-extended dither-based algorithm. Furthermore, when these two algorithms are simulated under the same step size, the variable-amplitude domain-extended dither-based algorithm has a much smaller steady-state error.

CHAPTER 7. CONCLUSION

One of the targets of this research is the design of a pipeline ADC with a good trade-off between power and performance. In other words, the purpose is to design a high accuracy ADC with a minimized power consumption. In order to realize low power consumption, the front-end stage with a sample-and-hold function for the SHA-less ADC is developed. In addition, in order to realize high accuracy, the domain-extended 2.5-bit/stage architecture is developed to improve the comparator offset correction ability. Moreover, in order to improve accuracy, this research also uses a combination of techniques, such as CFCS technique, gain boost amplifiers, and low noise dynamic comparators.

The ADC chip using the above mentioned techniques is fabricated with AMIS 0.5 μm CMOS. The ADC, with an active area of 4.5 mm^2, consumes 264 mW, when a 32 MHz input is at a 75-MS/s sample rate. The measured DNL and INL without calibration and dithering are within ±0.5 and ±2 LSB, respectively. The measured SFDR, THD, and SNR, with a 32 MHz input at 3-V power supply and a 75-MS/s sample rate, are 93.72-dB, -90.74-dB, and 78.42-dB, respectively. In comparison with other ADCs, this proposed ADC achieves a good trade-off between power and performance.

As stated before, the high resolution ADC can be achieved without calibration. Therefore, in order for a digital calibration to be useful, it should minimize the analog circuits as well as have reasonable convergence time. The dither-based digital calibration corrects linear errors, which include capacitor mismatches, and finite op-amp open-loop gains. Hence, the reduced accuracy of the minimized analog circuits can be complemented by the digital calibration. As stated before, the convergence time determines if a digital calibration is useful. The first proposed domain-

extended dither-based algorithm consumes less convergence time than the traditional dither-based algorithm. The second proposed variable-amplitude domain-extended dither-based algorithm consumes even less convergence time than the previous proposed domain-extended dither-based algorithm. In addition, the variable-amplitude domain-extended dither-based algorithm has significantly less steady-state errors than the domain-extended dither-based algorithm using the same step size. Furthermore, the traditionally variable-amplitude dither-based algorithm is not applicable when the input signal stays at high levels at all times. However, the newly proposed variable-amplitude domain-extended dither-based algorithm is not limited by this restriction. Therefore, this algorithm expands the range of application.

REFERENCES

[1] L. Dongtian, "High Speed CMOS ADC for UWB Receiver," Hong Kong University of Science and Technology, 2007.

[2] W. Yang, D. Kelly, I. Mehr, M. T. Sayuk, and L. Singer, "A 3-V 340-mW 14-b 75-Msample/s CMOS ADC With 85-dB SFDR at Nyquist Input," *IEEE J. Solid-State Circuits*, vol. 36, no. 12, pp. 1931–1936, 2001.

[3] I. Mehr and L. Singer, "A 55-mW, 10-bit, 40-Msample/s Nyquist-rate CMOS ADC," *IEEE J. Solid-State Circuits*, vol. 35, no. 3, pp. 318–325, Mar. 2000.

[4] A. Ali, C. Dillon, R. Sneed, A. Morgan, S. Bardsley, J. Kornblum, and L. Wu, "A 14-bit 125 MS / s IF / RF Sampling Pipelined ADC With 100 dB SFDR and 50 fs Jitter," *IEEE J. Solid-State Circuits*, vol. 41, no. 8, pp. 1846–1855, 2006.

[5] S. H. Lewis, "Optimizing the Stage Resolution in Converters for Video-Rate Applications," *IEEE Trans. Circuits Syst. Analog Digit. Signal Process.*, vol. 39, no. 8, pp. 516–523, 1992.

[6] J. Li, S. Member, X. Zeng, L. Xie, J. Chen, J. Zhang, and Y. Guo, "A 1.8-V 22-mW 10-bit 30-MS/s Pipelined COMS ADC for Low-Power Subsampling Applications," *IEEE J. Solid-State Circuits*, vol. 43, no. 2, pp. 321–329, 2008.

[7] J. Yu, "Charge Injection and Clock Feedthrough," San Jose State University, 2010.

[8] H.-S. Lee, "A 12 – b 600 ks / s Digitally Self-Calibrated Pipelined Algorithmic ADC," *IEEE J. Solid-State Circuits*, vol. 29, no. 4, pp. 509–515, 1994.

[9] J. P. Keane, P. J. Hurst, and S. H. Lewis, "Background Interstage Gain Calibration Technique for Pipelined ADCs," *IEEE Trans. Circuits Syst. Regul. Pap.*, vol. 52, no. 1, pp. 32–43, 2005.

[10] Y. Lin, "Yield and performance enhancement of analog and mixed signal circuits," Iowa State University, 2006.

[11] J.-L. Huang, C.-K. Ong, and K.-T. Cheng, "A BIST scheme for on-chip ADC and DAC testing," in *Proceedings of the conference on Design, automation and test in Europe*, 2000, pp. 216–220.

[12] S. Max, "Fast Accurate and Complete ADC Testing," in *International Test Conference*, 1989, pp. 111–117.

[13] D. H. Joey Doernberg, Hae-seung Lee, “Full-Speed Testing of A / D Converters,” *IEEE J. Solid-State Circuits*, vol. sc - 19, pp. 820–827, 1984.

[14] D. Dallet and J. Machado, *Dynamic Characterization of Analog-to-Digital Converters*. Beijing: Science Press, 2007, pp. 10–100.

[15] D. G. Nairn, “A 10-bit, 3V, 100M§/s Pipelined ADC,” in *IEEE Custom Integrated Circuits Conf.*, 2000, no. 8, pp. 257–260.

[16] K. Gulati, M. S. Peng, A. Pulincherry, C. E. Muñoz, M. Lugin, A. R. Bugeja, J. Li, and A. P. Chandrakasan, “A Highly Integrated CMOS Analog Baseband Transceiver With 180 MSPS 13-bit Pipelined CMOS ADC and Dual 12-bit DACs,” *IEEE J. Solid-State Circuits*, vol. 41, no. 8, pp. 1856–1866, 2006.

[17] L. Zhao, Y. Yang, Z. Zhu, and L. Liu, “SHA-less architecture with enhanced accuracy for pipelined ADC,” *J. Semicond.*, vol. 33, no. 2, pp. 1–5, Feb. 2012.

[18] P. Huang, S. Hsien, V. Lu, P. Wan, S. Lee, W. Liu, B. Chen, Y. Lee, W. Chen, T. Yang, G. Ma, and Y. Chiu, “SHA-Less Pipelined ADC With In Situ Background Clock-Skew Calibration,” *IEEE J. Solid-State Circuits*, vol. 46, no. 8, pp. 1893–1903, 2011.

[19] Gilbert Promitzer, “12 Bit Low Fully Differential Switched Capcitor Non-Calibrating Successive Approximation ADC with 1MS/s,” *IEEE J. Solid-State Circuits*, vol. 36, no. 7, pp. 1138 – 1143, 2001.

[20] L. Sumunen, M. Wulturi, and K. Hulonen, “A Mismatch Insensitive CMOS Dynamic Comparator for Pipeline A / D Converters,” in *IEEE Int. Conf. Electronics, Circuits and Systems*, 2000, pp. 32 – 35.

[21] T. Li and C. You, “A 3-V 14-bit 75-MS / s CMOS pipeline analog-to-digital converter with 93 . 72-dB spurious-free dynamic range,” *Int. J. Electr. Electron. Comput. Syst.*, vol. 19, no. 1, pp. 871 –879, 2014.

[22] Y. Chiu, “High-Performance Pipeline A/D Converter Design in Deep-Submicron CMOS,” 2004.

[23] L. A. Singer and T. L. Brooks, “A 14-Bit 10-MHz Calibration-Free CMOS Pipelined,” in *VLSI Dig. Tech. Papers*, 1996, vol. 27, no. 6, pp. 94–95.

[24] L. Singer, S. Ho, M. Timko, and D. Kelly, “A 12b 65MSample / s CMOS ADC with 82dB SFDR at 120MHz,” in *ISSCC Dig. Tech. Papers*, 2000, vol. 32, pp. 38 –39.

[25] T. Li and C. You, “A Domain Extension Algorithm for Digital Error Correction of Pipeline ADCs,” *Circuits Syst.*, vol. 5, no. 2, pp. 39–44, 2014.

[26] S. Hamami, L. Fleshel, and O. Yadid-Pecht, “CMOS image sensor employing 3.3V 12bit 6.3MS/s pipelined ADC,” *Sensors and Actuators*, vol. 135, no. 1, pp. 119–125, Mar. 2007.

[27] M. Waltari and K. a. I. Halonen, “1-V 9-bit pipelined switched-opamp ADC,” *IEEE J. Solid-State Circuits*, vol. 36, no. 1, pp. 129–134, 2001.

[28] J. He, S. Zhan, D. Chen, S. Member, and R. L. Geiger, “Analyses of Static and Dynamic Random Offset Voltages in Dynamic Comparators,” *IEEE Trans. Circuits Syst. Regul. Pap.*, vol. 56, no. 5, pp. 911–919, 2009.

[29] R. Musa, Y. Yusoff, T. K. Yew, and M. R. Ahmad, “Design of Single-Stage Folded-Cascode Gain Boost Amplifier for 100mW 10-bit 50MS/s pipelined Analog-to-Digital Converter,” in *ICSE2006 Proc.*, 2006, no. 1, pp. 800–804.

[30] S. Franco, *Design with Operational Amplifiers and Analog Integrated Circuits*, vol. 28, no. 12. 2001, pp. 1–63.

[31] E. Sackinger and W. Guggenbuhl, “A High-Swing,Hhigh-Impedance MOS Cascode Circuit,” *IEEE J. Solid-State Circuits*, vol. 25, no. 1, pp. 289–298, 1990.

[32] M. Das, “Improved Design Criteria of Gain-Boosted CMOS OTA With High-Speed Optimizations,” *IEEE Trans. Circuits Syst. Analog Digit. Signal Process.*, vol. 49, no. 3, pp. 204–207, 2002.

[33] 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 J. Solid-State Circuits*, vol. 44, no. 6, pp. 1709–1720, 2009.

[34] S. Devarajan, L. Singer, D. Kelly, S. Decker, A. Kamath, and P. Wilkins, “A 16-bit, 125 MS/s, 385 mW, 78.7 dB SNR COMS Pipeline ADC,” *IEEE J. Solid-State Circuits*, vol. 44, no. 12, pp. 3305–3313, 2009.

[35] E. Iroaga and B. Murmann, “A 12-Bit 75-MS/s Pipeline ADC Using Incomplete Settling,” *IEEE J. Solid-State Circuits*, vol. 42, no. 4, pp. 748–756, 2007.

[36] H. Cai, P. Li, Y. Cen, and Z. Zhu, “A 14-bit 80 MS/s CMOS ADC with 84.8 dB SFDR and 72 dB SNDR,” *J. Semicond.*, vol. 33, no. 2, p. 025012, Feb. 2012.

[37] U. Moon and B. Song, “Background Digital Calibration Techniques for Pipelined ADC’s,” *IEEE Trans. Circuits Syst. Analog Digit. Signal Process.*, vol. 44, no. 2, pp. 102–109, 1997.

[38] S. Yang, J. Cheng, and P. Wang, “Variable-amplitude dither-based digital background calibration algorithm for linear and high-order nonlinear error in pipelined ADCs,” *Microelectronics J.*, vol. 41, no. 7, pp. 403–410, Jul. 2010.

[39] C. Tsang, Y. Chiu, J. Vanderhaegen, S. Hoyos, C. Chen, R. Brodersen, and B. Nikoli, "Background ADC Calibration in Digital Domain," *Proc. IEEE 2008 Cust. Integr. Circuits Conf.*, no. Cicc, pp. 301–304, 2008.

[40] B. Murmann and B. E. Boser, "A 12-bit 75-MS / s Pipelined ADC A 12-b 75-MS/s Pipelined ADC Using Open-Loop Residue Amplification," *IEEE J. Solid-State Circuits*, vol. 38, no. 12, pp. 2040–2050, 2003.

[41] E. Siragusa and I. Galton, "Gain error correction technique for pipelined analog-to-digital converters," *Electonics Lett.*, vol. 36, pp. 617–618, 2000.

[42] E. Siragusa and I. Galton, "A Digitally Enhanced 1.8-V 15-bit 40-MSample/s CMOS Pipelined ADC," *IEEE J. Solid-State Circuits*, vol. 39, no. 12, pp. 2126–2138, 2004.

[43] A. Panigada and I. Galton, "Digital Background Correction of Harmonic Distortion in Pipelined ADCs," *IEEE Trans. Circuits Syst. Regul. Pap.*, vol. 53, no. 9, pp. 1885–1895, 2006.

[44] A. Meruva and B. Jalali, "Digital Background Calibration of Higher Order Nonlinearities in Pipelined ADCs," *2007 IEEE Int. Symp. Circuits Syst.*, pp. 1233–1236, May 2007.

[45] Y. Shu and B. Song, "A 15-bit Linear 20-MS / s Pipelined ADC Digitally Calibrated With Signal-Dependent Dithering," *IEEE J. Solid-State Circuits*, vol. 43, no. 2, pp. 342–350, 2008.

[46] B. Song, S. Lee, and M. Tompsett, "A 10-b 15-MHz CMOS Recycling Two-step A/D Converter," *IEEE J. Solid-State Circuits*, vol. 25, no. 6, pp. 1328 – 1338, 1990.

[47] S. H. Lewis and P. R. Gray, "A pipelined 5-Msample/s 9-bit analog-to-digital converter," *IEEE J. Solid-State Circuits*, vol. 22, no. 6, pp. 954–961, Dec. 1987.

[48] J. Li and U. Moon, "Background Calibration Techniques for Multistage Pipelined ADCs With Digital Redundancy," *IEEE Trans. Circuits Syst. Analog Digit. Signal Process.*, vol. 50, no. 9, pp. 531–538, 2003.

[49] T. Li and C. You, "A Digital Calibration Algorithm with Variable -Amplitude Dithering for Domain-extended Pipeline ADCs," *Int. J. VLSI Des. Commun. Syst.*, vol. 5, no. 1, pp. 1–13, 2014.