# Accuracy Enhancement Techniques in Low-Voltage High-Speed Pipelined ADC Design

by

Jipeng Li

A DISSERTATION

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Doctor of Philosophy

Presented October 3, 2003
Commencement June 2004

# ACKNOWLEDGEMENTS

I would like to thank Professor Un-Ku Moon, my advisor, for his encouragement and insightful guidance throughout my Ph.D. studies. I am also grateful to Professors Gábor C. Temes, Karti Mayaram and Terri Fiez for their helps in both research and academic matters. I would also like to acknowledge all members of my committee, for taking the time to serve on my defense.

I would like to thank Dr. Byung-Moo in National Semiconductor Corporation for the sponsorship of the prototype ADC fabrication. I appreciate the helps on chip layout from Mr. Yong-In Park in Texas Instruments Corporation. This work was supported by NSF center for Design of Analog and Digital Integrated Circuits (CDADIC) and partly by Analog Devices Corporation.

All my colleagues in the Analog Group have not hesitated to offer their help to me. First, I would like to thank Jose Silva. He has offered great help to me from the beginning of my research in school to the end of my study. I would also like to thank Dong-Young Chang, Cil-Cho Ahn, Xuesheng Wang and Pavan Hanumolu for helpful discussions.

Last, I would like to thank all my friends for their understanding, support and sharing happiness with me.

***To***

***my parents Shukun Li, Yongzhi Gao***

***and***

***my wife Lanyue Xu***

## TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

Page

TABLE OF CONTENTS (Continued)

Page

# LIST OF FIGURES

LIST OF FIGURES (Continued)

LIST OF FIGURES (Continued)

LIST OF FIGURES (Continued)

# LIST OF TABLES

# ACCURACY ENHANCEMENT TECHNIQUES IN LOW-VOLTAGE HIGH-SPEED PIPELINED ADC DESIGN

## 1 INTRODUCTION

### 1.1 Background

Analog-to-digital converters (ADCs) are very important building blocks in modern signal processing and communication systems. Many good ADC architectures have been invented to satisfy different requirements in different applications. To name some: flash ADC, folding and interpolating ADC, two-step ADC, pipeline ADC, successive-approximation-register (SAR) ADC, delta-sigma ADC, integrating ADC etc. Among various ADC architectures, the pipelined ADC has the attractive feature of maintaining high accuracy at high conversion rate with low complexity and power consumption. Therefore it is used extensively in high-quality video systems, high-speed data acquisition systems and high performance digital communication systems where both precision and speed are critical. Some typical applications for pipelined ADCs are listed in Table 1.1.

## 1.2 Motivation

The rapid growth of these applications is driving the pipelined ADC design towards higher speed, higher precision, lower power consumption, lower supply voltage, smaller size and higher levels of integration along with the advancement of the fabrication technology. While continual speed improvement can still be achieved by using the advanced sub-micron or deep sub-micron CMOS processes, data converter designers find it more and more difficult to improve or even keep the accuracy of pipeline ADCs which reply on high gain operational-amplifier (opamp) and well matched components to produce high-precision converters. First, large open loop opamp gain is difficult to realize without sacrificing bandwidth given the continuing trend of submicron CMOS scaling which is coupled with lower power supply voltages. Second, there are some physical limits on the component matching due to process variation, so it can not be improved continually with CMOS technology scaling. Thus, while the state-of-the-art pipelined ADCs has exceeded 100MSPS (mega-samples-per-second) in CMOS technology [1]-[6], the commonly achieved resolution is still bound within the range of 8-12 effective-number-of-bits (ENOBs) due to the limitations set by component mismatches and finite opamp gain. Use of multi-bit-per-stage architecture and design optimization can achieve 14-bit performance as demonstrated in [7], but most pipelined ADCs with more than 12-bit resolution will usually require some kind of linearity enhancement techniques.

## 1.3 Existing Methods

Trimming is one such method but it cannot track variations over time caused by component aging and temperature changes despite the high cost of implementation. Some traditional analog techniques such as ratio-independent multiplication [8], reference-refreshing method [9], and capacitor error averaging [10][11] can also correct errors due to component mismatches. However, these techniques require at least one extra clock phase for correction, which equivalently reduces the conversion rate. To avoid this kind of speed penalty, self-calibration techniques, which measure errors by the converter itself and subtract the code error during the normal operation, can be used to improve the accuracy of high resolution ADCs. In many of the self-calibration techniques, although the code errors are calculated in the digital domain, they are actually subtracted in the analog domain using a separate calibration digital-to-analog converter (DAC) [12][13]. More recently, digital self-calibration techniques, which subtract the code errors in the digital domain, have been introduced to relieve the accuracy requirements of analog calibration circuits [14]-[17]. While digital self-calibration has the advantage of low complexity and high accuracy, most are calibrated in the foreground. That means the normal operation has to be interrupted to start the calibration cycle. Although the calibration can be done during the system power-up or standby, it is desirable to run the calibration at all times to track device and environmental variations.

To avoid the foreground interruptions, several background calibration schemes that are transparent to the normal operation have been proposed. A resistor-string DAC instead of a capacitor DAC was used in a 13-bit ADC, so that it could be calibrated in the background by a slow-but-accurate delta-sigma ADC [18]. This scheme can only be applied to an ADC with a resistor-string DAC. Redundant pipeline stages can be added to substitute the stages under calibration as shown in [19], so that the normal operation need not be stopped during calibration. This results in a large overhead for die area and power dissipation. To create the needed time slots for calibration, a skip-and-fill algorithm was proposed [20][21]. In this algorithm, the conversion of input samples is occasionally skipped, and a sample of the calibration signal is converted instead. The missing input samples are later filled in digitally via nonlinear interpolation of data. To avoid the complexity of the digital post-processing required by this skip-and-fill algorithm, a queue-based architecture was used in an algorithmic ADC [22]. The skip-and-fill method is relatively simple and accurate for implementation, but the input signal bandwidth has to be limited to avoid performance degradation due to interpolated regeneration of skipped samples [20][21].

Several correlation-based methods have been proposed for background calibration in pipelined ADCs [23]-[27]. These methods modulate the calibration signal or capacitor DAC errors with a pseudo-random noise sequence in the analog domain and then demodulate them in the digital domain to extract the errors from the processed input signal. The input signal bandwidth limitation and/or redundant analog hardware can be avoided using this method. Despite the added advantages, the previously reported correlation-based schemes are quite complex and slow to converge.

## 1.4 Proposed Approaches

Although many accuracy enhancement techniques are already available and work well under certain context, all of them have some drawbacks. Usually, either speed or power consumption or both have to be compromised. And most of them are difficult to implement in low-voltage and high-speed designs. Therefore, simple and efficient accuracy enhancement techniques which can be used in low-voltage and high-speed pipelined ADCs would be worthy of focused research.

In this work, two novel accuracy enhancement techniques are proposed. The first one is the *time-shifted* correlated-double-sampling (CDS) technique [28][29] which can be used to compensate finite opamp gain in the context of a 1.5-bit-per-stage pipelined ADC. This technique can significantly reduce the errors due to finite opamp gain without compromising conversion speed. The second approach is a fast and accurate correlation-based background digital calibration scheme [30]. The input signal magnitude needs not to be reduced to allow the injection of a pseudo-random calibration signal. The minimal addition of analog hardware for calibration keeps the original ADC design essentially unchanged. The correlation algorithm converges very quickly in the proposed two-channel ADC architecture because of the interference canceling scheme employed. The effectiveness of both methods is demonstrated in simulations as well as experiments.

## 1.5 Thesis Organization

The thesis is organized as follows. The basic operation and design techniques of pipelined ADCs are described in Chapter 2. An overview of some accuracy enhancement techniques in high performance pipelined ADC designs is given in Chapter 3. Next, two novel accuracy enhancement techniques are presented. In Chapter 4, the *time-shifted* correlated-double-sampling (CDS) technique which can efficiently compensate the finite opamp gain in pipeline ADCs is explained. In Chapter 5, a novel background digital calibration technique which can correct errors due to finite opamp gain as well as capacitor mismatch is described. The conclusions and a summary of research are given in Chapter 6.

| Resolution<br>(number of bits) | Applications |
|---|---|
| 8 | • Flat-panel displays<br>• Lab instrumentation<br>• HDTV<br>• Medical imaging (low-end portable)<br>• WLAN and WAN<br>• Radar |
| 10 | • Flat-panel displays<br>• HDTV<br>• Medical imaging such as ultrasound<br>• Cellular basestations (power amp linearization)<br>• High-data-rate radios (point-to-point microwave, LMDS, and MMDS)<br>• Cable headends (for digitizing cable modem uplinks) |
| 12 | • Cellular basestations<br>• Test equipment for ATE and communications<br>• Cable headends<br>• Professional HDTV cameras<br>• Medical imaging |
| 14 | • Cellular basestations, particularly 3G multicarrier systems<br>• ATE<br>• High-end instrumentation<br>• Military and aerospace |

Table 1.1 Typical applications for pipelined ADCs

# 2 REVIEW OF FPIPLINED ADC

Deep insight into of the practical design of a high-performance pipelined ADC is the cornerstones of developing a successful accuracy enhancement technique. In this chapter, the basic knowledge of pipelined ADCs and more advanced design techniques in CMOS technology are reviewed. First, the structure and operation of a typical pipelined ADC are introduced. Then the details of the building blocks design are described. The design issues which are critical to the function as well as performance are also discussed. A simple introduction of some advanced design techniques is given in the last section of this chapter.

## 2.1 Pipelined ADC Fundamentals

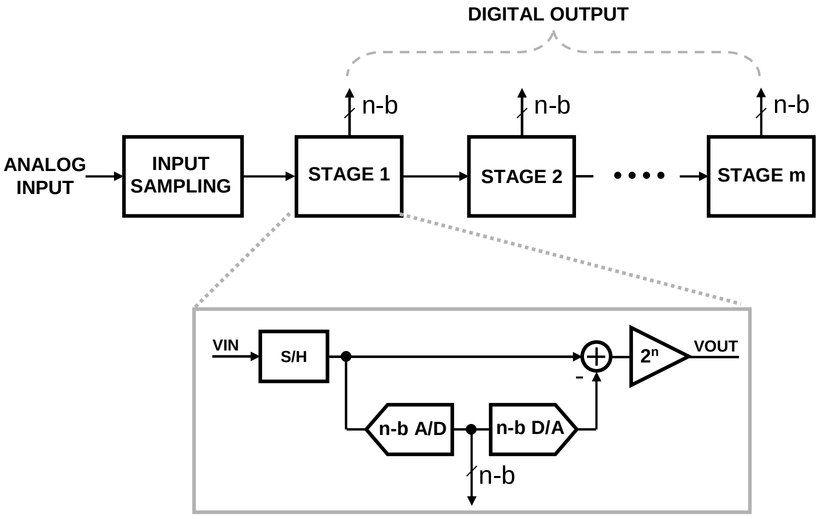


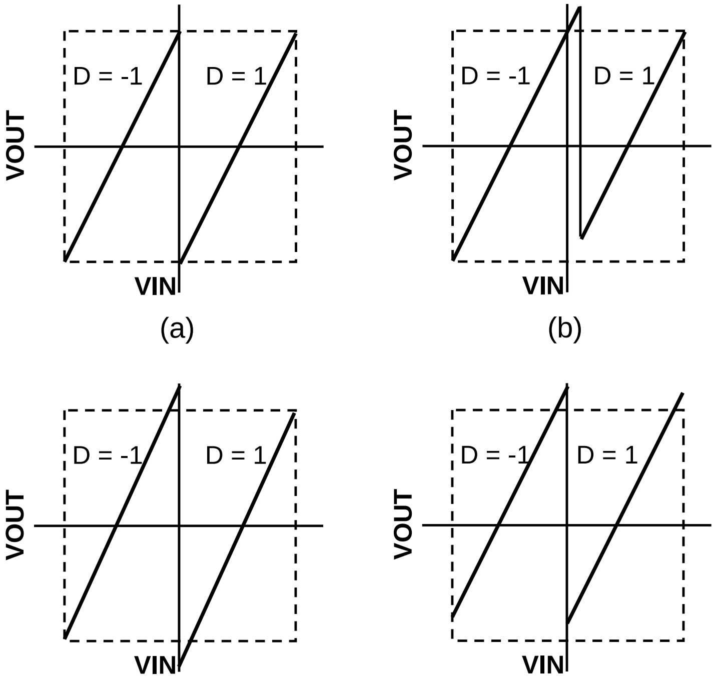
Fig.2.1 A typical pipelined ADC

Figure 2.1 shows the structure of a typical pipelined ADC. It includes several cascaded stages. In each stage, there is a sample and hold (S/H) block, a sub-ADC (usually low resolution flash type ADCs), a sub digital-to-analog converter (sub-DAC), a subtractor and an inter-stage gain amplifier. The operation of each stage is as follows. The sampled input signal is first quantized by the sub-ADC to produce the output digital code for this stage. Then the output digital code is converted back to an analog signal by the sub-DAC. This quantized analog signal is subtracted from the input signal, resulting in a residue that is amplified and then passed onto the next stage. A binary bit searching scheme is employed in the operation of pipelined ADCs. While the overall resolution of the pipelined ADC is the sum of the number of bits resolved in each stage, the throughput rate of the overall pipelined ADC is equal to each stage's throughput rate because of the pipelining. Of course, there is also a latency introduced by this pipelining. Fortunately, this is not an issue in most applications.

Two distinguished attributes of the pipelined ADC architecture can be observed from the simple introduction above. First, unlike the flash type ADC whose circuit complexity increases exponentially with resolution, the circuit complexity of a pipelined ADC increases only linearly with the converter's resolution because of the binary bit searching scheme involved. Second, the throughput rate does not increase with the number of stages because of the pipelining. Ideally, the pipeline ADC could be as fast as flash the ADCs and consume much less power. However, this is not completely true in practical designs particularly in low resolution and very high-speed converters. The opamps used in pipelined ADCs usually consume more power than the comparators and limit the speed of the overall ADC. This will be explained in detail in later sections.

Two other advantages of pipelined ADCs which maybe not be obvious are the input S/H function and the inter-stage gain function. The front S/H block makes the pipelined ADC insensitive to the frequency of the input signal processed. Actually, it is not unusual for a carefully designed pipeline ADC to maintain good performance at the Nyquist input frequency or higher (sub-sampling). The inter-stage gain makes the pipelined ADC less insensitive to the noise and non-idealities of the later stages in the pipeline. Therefore, scaling can be used to optimize the design for low power consumption and small area.

Another very important concept to mention here is the digital correction which is used in almost every pipelined ADC design. Digital correction dramatically reduces the pipelined ADC's sensitivity to the nonlinearities in the sub-ADCs, therefore makes the design of comparators in sub-ADCs much easier. While the detailed analysis of digital correction can be found in [31], a simple introduction is given below.



(a) (b) (c) (d)

Fig. 2.2 Effects of non-idealities in 1b/stage pipelined ADC

To understand the digital correction, some basic knowledge of the non-idealities in the pipelined ADC and their effects is necessary. Comparator and opamp offsets, capacitor mismatches, finite opamp gains, charge injections are some major error sources in pipelined ADCs. Their effects on the transfer curve of the pipeline ADC can be illustrated in Fig. 2.2. Figure 2.2(a) is the ideal stage input-output transfer curve of a 1bit/stage pipelined ADC. The comparator offset will shift the transition point and cause the residue to exceed the signal range as indicated in Fig. 2.2(b). The capacitor mismatch and finite opamp gain can change the inter-stage gain. This gain error will change the slope of the curve as shown in Fig. 2.2(c). The vertical shift of the transfer curve in Fig. 2.2(d) is caused by the charge injection offset. Thus the linearity of the comparator is corrupted. It can be seen that all the error sources can possibly cause the residue to go out of signal range. Once this happen, the missing levels will be produced and the linearity of the overall ADC will be corrupted. Although the example of a 1bit/stage ADC is given for simplicity, a similar issue exists in multi-bit/stage ADCs. To avoid the residue going out of signal range, a simple approach is to introduce some redundancy in the pipelined ADC design. In a practical design, this can be done by choosing smaller inter-stage gain factor. For instance, an inter-stage gain of 4 instead of 8 can be chosen in a 3bit/stage ADC design. So in the ideal case, the residue will be just half of the signal range. And it will not exceed the signal range even if there are some comparator offsets as far as they are less than ±1/2LSB of the stage resolution. However, this modification (adding redundancy) changes the encoding scheme of the overall pipelined ADC and a digital correction is needed to post-process the ADC's output code. The digital correction logic can be greatly simplified if some offsets are added to the sub-ADCs and the sub-DACs which results in a mid-tread transfer curve as shown in Fig. 2.3. Note only 6 comparators are needed to resolve 7 levels for a 3bit sub-ADC. Note one more comparator is needed for the last stage to give a true 3bit output representing 8 levels. Otherwise, there will be one missing code. Now the digital correction is just some simple bit-shift adding on the pipeline ADC's digital output (overlap the LSB of each stage and the MSB of its following stage) as demonstrated in Fig.2.4. Note the final number of bits is reduced after this digital correction. For example, a five stages pipelined ADC (3bit/stage) only give 11 bit output instead of 15bit after digital correction. However, this drawback is well compensated given the fact that fairly large comparator offsets can be tolerated without affecting the linearity of the overall ADC. As a result, the comparator design is much easier (usually no need for offset canceling) and the digital calibration of the inter-stage gain errors is feasible.

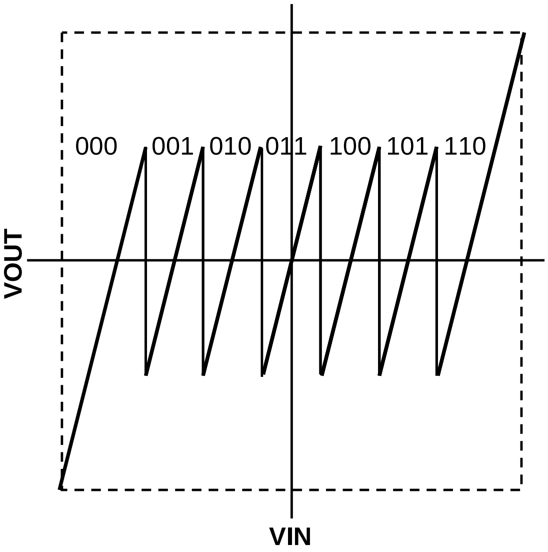


Fig. 2.3 Mid-tread transfer curve with redundancy

```
STAGE 1     010
STAGE 2       010
STAGE 3         010
STAGE 4           010
STAGE 5             111
------------------------------
ADC OUTPUT  01010101111
            ↑         ↑
           MSB       LSB
```

Fig. 2.4 Digital correction algorithm

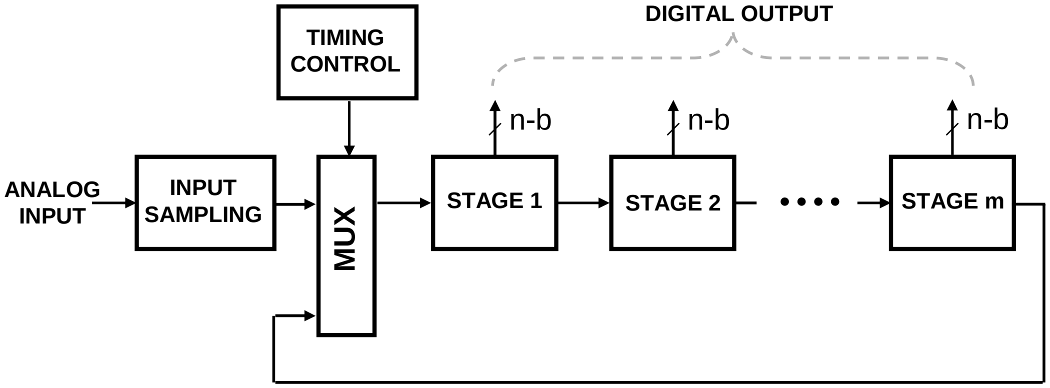


Fig. 2.5 General structure of cyclic ADCs

One variant of the pipelined ADC is the cyclic ADC or algorithm ADC which is shown in Fig. 2.5. It is almost the same as the pipeline ADC. But one multiplexer (MUX) is put in front of all the pipelined stages. Each input sample will be processed through all the stages in the same way of a pipelined ADC. However, the residue of the last stage will be feed back to the input of the pipeline to quantize again. This kind of "recycling" process can equivalently extend the length of the pipeline, and so increase the resolution. The price is the increased time to process an input sample. Actually, both the resolution and conversion time will increase proportionally with the number of "cycles" used in the conversion. So the cyclic ADC can be seen as one kind of pipeline ADC which trades speed with space and power consumption. And it can be used in those applications where the small die area and low power consumption are important but the conversion rate is relatively low.

## 2.2 Building Block Design

Although the highest performance monolithic pipelined ADCs are still built in BiCMOS or bipolar processes, the mainstream of pipelined ADC design has already shifted to CMOS process for lower cost and power. Since fine-line CMOS technology can provide good sampling switches and well matched capacitors, the switched-capacitor (SC) techniques based on charge transferring are used extensively in the CMOS pipelined ADC designs. In the following, the building blocks design of a typical CMOS pipelined ADC using switched-capacitor techniques will be described.

### 2.2.1 Input Sample and Hold Circuit

Most pipeline ADCs have an on-chip sample and hold circuit in front of the pipeline stages to buffer the input signal as indicated in Fig. 2.1. This front S/H circuit can give some isolation between the pipelined ADC and its driving circuit. So the driver suffers less kick-back noise from the comparators in the pipelined ADC. More importantly, since the S/H circuit keeps the sampled input signal constant during holding phase, it can eliminate the signal discrepancy between the input of sub-ADC and subtractor caused by the clock skew. Note this signal discrepancy adds equivalent comparator offsets to the pipelined ADC and it could be very large if the input signal frequency is very high (up to or higher than Nyquist frequency). Once the comparator offsets exceed the range of digital correction, huge amount of errors will happen to the conversion.

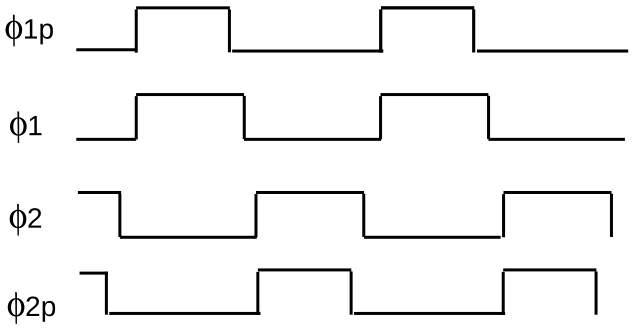


Fig. 2.6 Non-overlap and early falling clock scheme

Two CMOS S/H architectures are used widely in the pipelined ADC design. Both of them are fully differential circuits. And the classic four clock scheme (two non-overlap clock signals and two additional clock phases with early falling edge) for switched-capacitor circuits is employed. The clock waveforms are shown in Fig.2.6. The first one is referred as the charge redistribution S/H circuit as shown in Fig. 2.7. Four capacitors with the same size are used in this structure. During the sampling phase, the differential input signal is sampled into the two input sampling capacitors. Next, during the holding phase, the bottom plates of the two sampling capacitors are connected together. Thus only the differential charge is transferred to the feedback capacitors. As a result, this S/H can handle very large input common mode variation. The second one is the capacitor flip-over S/H as shown Fig. 2.8. No charge transferring happens in this scheme and only two capacitors are used. During the sampling phase, the differential input signal is sampled into the input capacitors in the same way as the first S/H. However, during the holding phase, the input capacitors are “flipped over” by connecting their bottom plates to the output the amplifier. By doing this, both the common mode and differential mode charge is transferred. Although the amplifier’s common mode feedback circuit will force the output common mode to the nominal value, the amplifier’s input common mode level will change according to the difference between the input signal’s common mode level and the amplifier output’s common mode level. That means the amplifier must be capable of handling large input common mode variation. Even with this drawback, the flip-over S/H is more popular than the charge redistribution S/H in the state of the art high-speed pipelined ADC designs because of its smaller size, lower noise and lower power consumption. These advantages stem from the large feedback factor and lower number of capacitors. The detailed analysis can be found in [7].

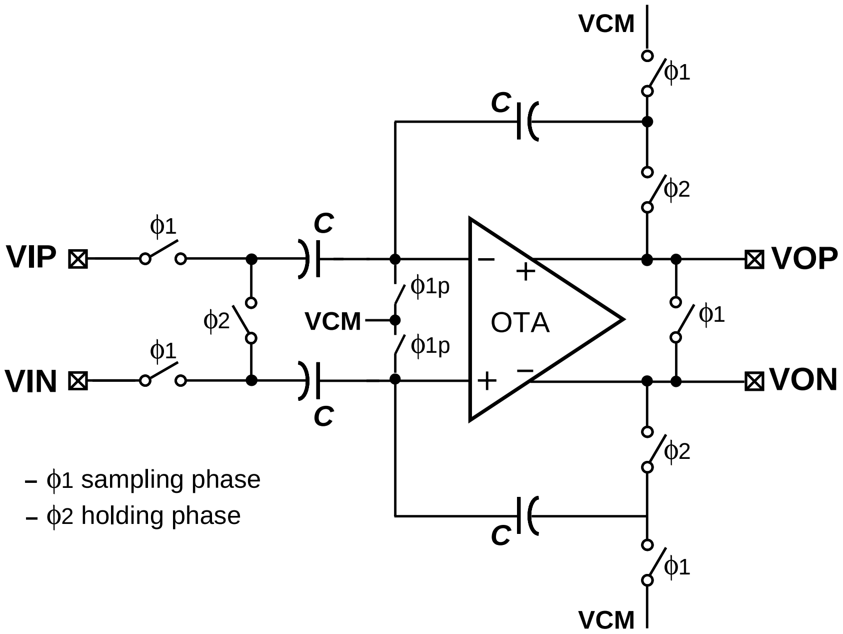


Fig. 2.7 Charge redistribution S/H circuit

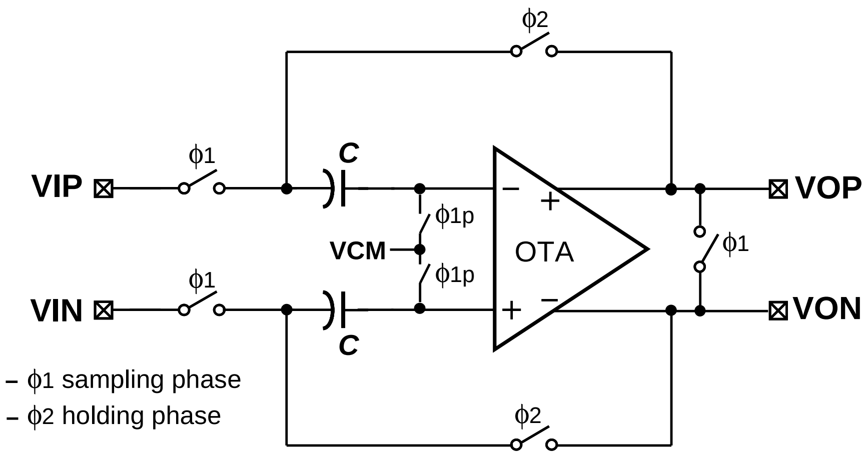


Fig. 2.8 Capacitor flip-over S/H circuit

The design of front S/H circuit is very critical to the overall performance of a pipelined ADC. The noise and linearity requirement should be the same or better than the overall noise and linearity requirement of the ADC. As a result, the S/H circuit usually takes large die area and consumes quite amount of power. To achieve low power consumption, a pipeline ADC without S/H circuit is proposed [32]. However, special efforts have to be put into the design to avoid the signal discrepancy issue mentioned above.

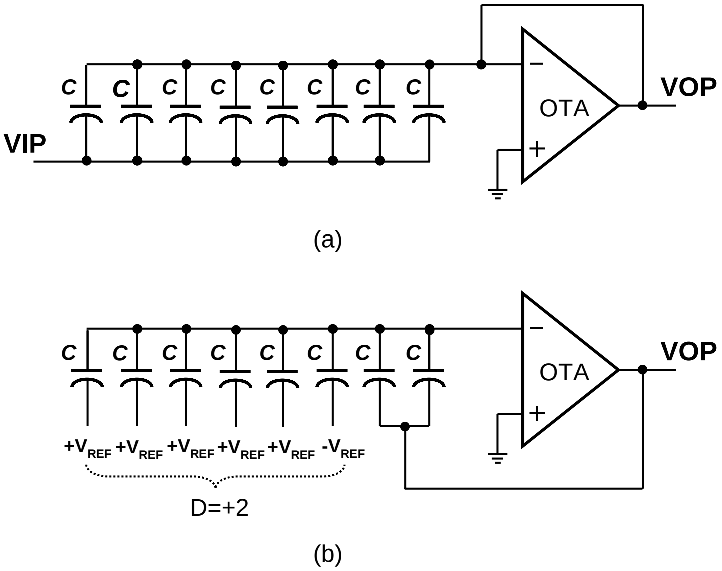
### 2.2.2 Multiplying Digital to Analog Converter

In the practical implementation of a pipelined ADC sown in Fig. 2.1, the sub-ADC, subtractor and inter-stage gain block in each stage are usually combined together and referred as multiplying-digital-to-analog converter (MDAC). The ideal input-output transfer function of an n-bit MDAC with digital redundancy can be expressed as:

$$V_{out} = V_{in} \cdot 2^{n-1} - D \cdot V_{REF} \,, \tag{2.1}$$

where $V_{in}, V_{out}$ and $V_{REF}$ are input voltage, output residue voltage and reference voltage respectively. And $D$ is $0, \pm 1, \pm 2 \cdots \pm (2^{n-1} - 1)$ digital code generated in sub-ADC depending on the input signal range.

MDAC is probably the most critical unit in a pipelined ADC design. It consumes most of the power dissipated by the ADC. And usually is the bottleneck for speed and accuracy improvement of a pipelined ADC. Therefore, almost all the low power, high speed and high precision techniques developed so far are focused on the MDAC design. And most of the design optimization work is also done in MDAC.



(a)

(b)

Fig. 2.9 A 3-b switched-capacitor MDAC

Despite its importance, the structure of a switched-capacitor MDAC is not very complex. Figure 2.9 illustrates the structure and operation of a typical 3-bit switched-capacitor MDAC whose input-output transfer curve is shown in Fig. 2.3. For simplicity, the single-ended version is shown although fully differential architecture is usually used in practical design. It takes two clock phases to fulfill the all the functions of a MDAC. During sampling phase, the input signal is sampled into eight unit size capacitors and the amplifier is reset as shown in Fig. 2.9(a). Note, in the mean time, the sub-ADC is sampling the input signal also and will give the output digital code and DAC control signal at the end of sampling phase. Next, during amplifying phase, two capacitors are flipped-over and have their bottom plates connected to the output of the amplifier while the bottom plates of the remaining capacitors are connected to $+V_{REF}$ or $-V_{REF}$ depending on the digital code (+2 for this case) from the sub-ADC output as shown in Fig.2.9(b).

The opamps used in MDACs and S/H circuits are actually operational transconductance amplifiers (OTAs) since they have high output impedance and only drive capacitor load. Opamp is the key part of the MDAC and worthy of most attention when designing a pipelined ADC. The main requirements on opamp design are high gain, large bandwidth, large slew rate, large signal swing, low noise, low power consumption and low power supply voltage. Besides these, high power supply noise rejection ratio (PSRR), high common mode noise rejection ratio (CMRR), small offset and small size are also desirable. Unfortunately, these factors can not be improved together and many trade-offs exist among them. For example, an increase in bandwidth almost always necessitates an increase in power consumption. So a good opamp design is the optimization of these factors according to the design specification.

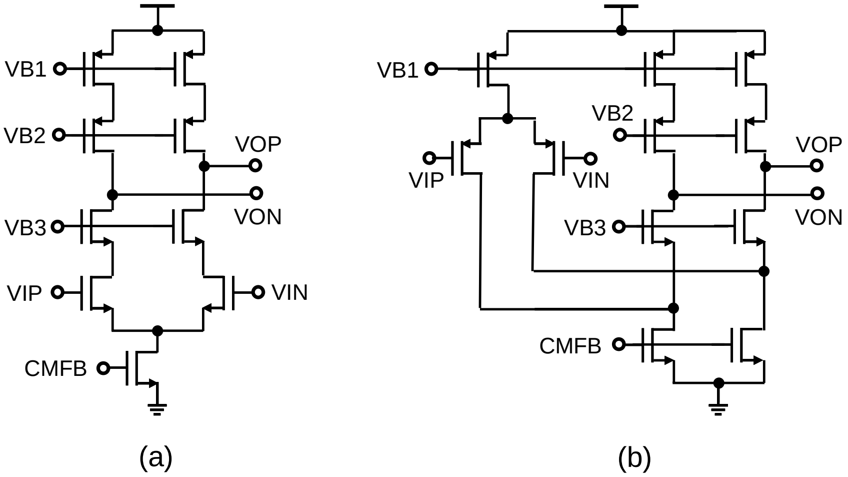


Fig. 2.10 Cascode opamp

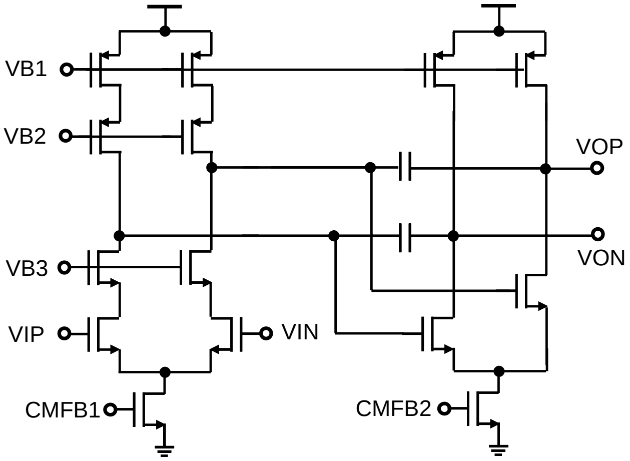


Fig. 2.11 Two-stage opamp

Many good opamps have been developed for different pipelined ADCs. Among them, the telescopic opamp shown in Fig. 2.10(a) and the folded cascode opamp shown in Fig. 2.10(b) are the most commonly used for their high gain and large bandwidth at given power consumption. While both opamp designs employ cascoded transistors to boost the opamp gain, they have many differences. The telescopic opamp has the advantages of higher speed and lower power consumption. But folded cacode opamp has large output signal swing and large input common mode range. Another good opamp design is the two-stage opamp design using miller compensation shown in Fig. 2.11. Although this two-stage opamp is usually slower than single stage designs, it can provide higher gain and large output signal swing, therefore is popular in low voltage pipelined ADC designs. Besides cascode and cascade (multi-stage opamp), another way to increase opamp gain is gain-boost technique which employs feedback scheme to enhance the output impedance of an opamp [33][34]. Because this technique does not compromise opamp bandwidth (like cascade) or signal swing (like cascode) and can be used with cascode and cascode together if more gain is needed, it is extensively used in current high-speed and high resolution pipelined ADC design. If highest bandwidth is of interest, the $G_m$ boost technique [35] which employs a wide band low gain preamplifier to boost the input equivalent $G_m$ and so bandwidth without increasing input parasitic capacitance can be used.

### 2.2.3 Sub-ADC

Another important building block in pipelined ADCs is the sub-ADC within each stage. Usually, the flash ADC architecture is employed and the resolution is less than 5-bit. Figure 2.12 shows the diagram of a typical sub-ADC. The input signal is fed into $2^N$ comparators in parallel directly (for an N-bit sub-ADC). The threshold voltages of these comparators are provided by a resistor string. And their outputs will go through bubble correction logic to minimize the effects of threshold error. Note the output word is thermometer code. So, after the bubble correction, an encoding circuit is needed to translate it to binary code to reduce the data bus width. Besides the function of a standard flash ADC, the sub-ADC has to provide the DAC control signal for MDAC. While the absolute delay of output binary code is relatively unimportant, the delay time of the DAC control signal is really critical and need to minimize in practical designs. Otherwise, it will take up the settling time of MDAC and limit the speed of overall ADC.

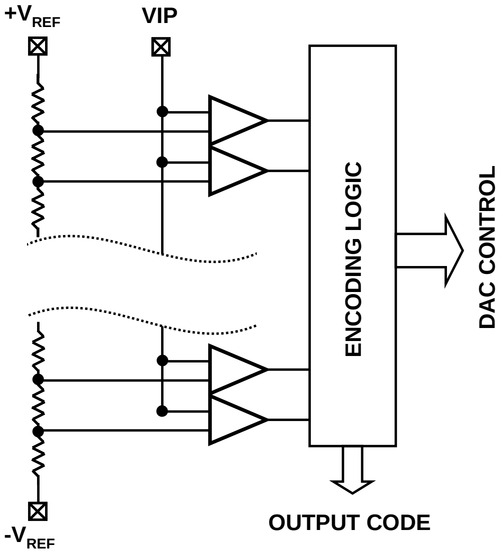


Fig. 2.12 Structure of a typical sub-ADC

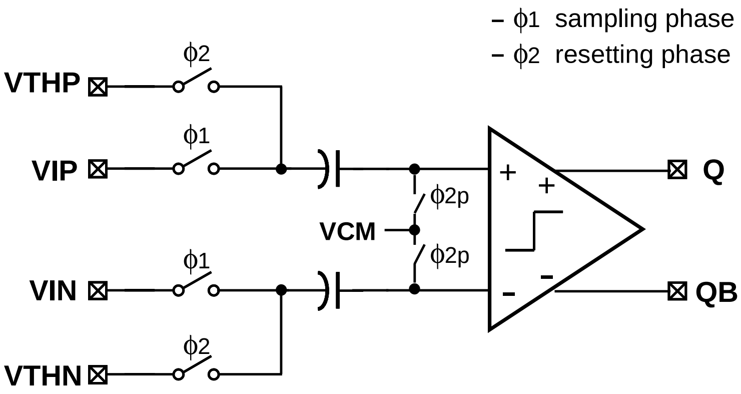


Fig. 2.13 Capacitive coupled comparator module

The capacitive coupled comparator shown in Fig. 2.13 is commonly used in sub-ADC designs because it can handle large input signal range and adopt offset cancellation easily. The non-overlap clock two phase clock is the same as shown before in Fig. 2.6. During the resetting phase (last sample), the input capacitors are reset to threshold voltage. Next, during the sampling phase (current sample), the input signal is sampled into the input capacitors. Thus the difference between input signal and threshold voltage will show up at the input of lathed comparator. Right before the end of the sampling phase, the latched comparator will be triggered, and the comparison result will be available to use by the encoding logic and DAC control logic in the beginning of the resettling phase.  Notice that the comparator latching time is only about the non-overlap time of the two phase clock. This time gap is very small compared to the clock period (usually less than 10%). For instance, the comparator needs to latch within 1ns even though the pipelined ADC's clock frequency is only about 100MHz. In very high speed pipelined ADC design, this stringent latching time requirement could pose a speed limitation. A simple solution to this problem is to reduce the sampling time of the comparator, so more latching time is assigned.

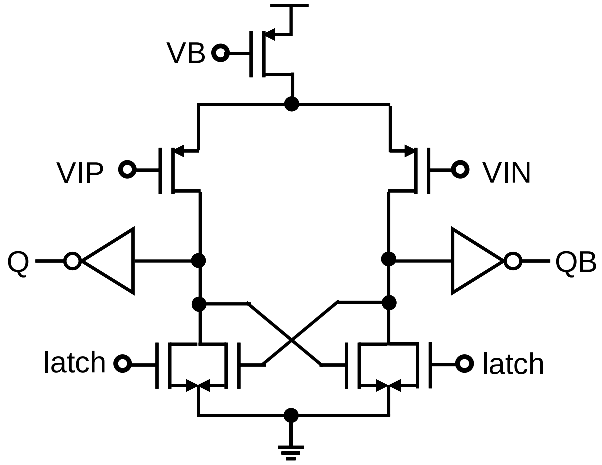


Fig. 2.14 Static latched comparator

The latched comparator used in the capacitive coupled comparator module could be just a single stage comparator without preamplifier since the accuracy requirement is very relaxed (usually less than 4bit). For example, in a 1.5bit/stage pipelined ADC, the static latched comparator shown in Fig.2.14 can be used in for its simplicity, or the dynamic comparators shown in Fig. 2.15 can be used for their ultra low power consumption. When the sub-ADC has resolution more than 4bit, multi-stage comparator design with offset cancellation is usually required to reduce kickback noise and increase accuracy.

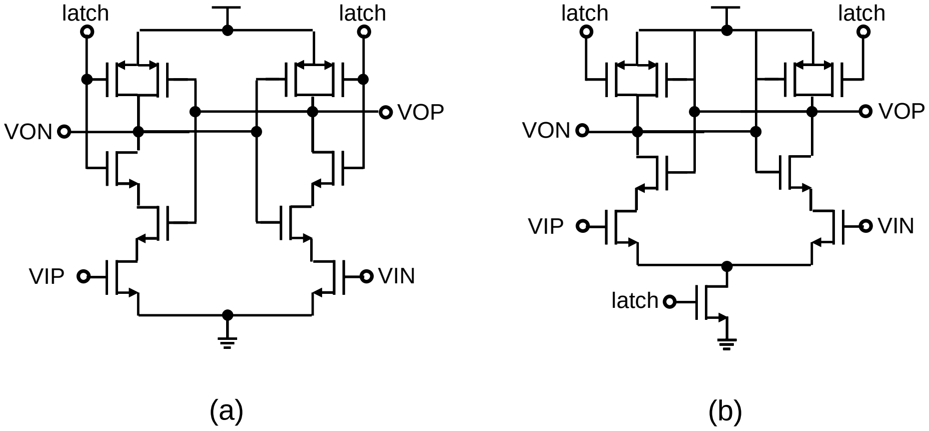


Fig. 2.15 Dynamic latched comparator

### 2.2.4 Clock Generator and Clock Buffer

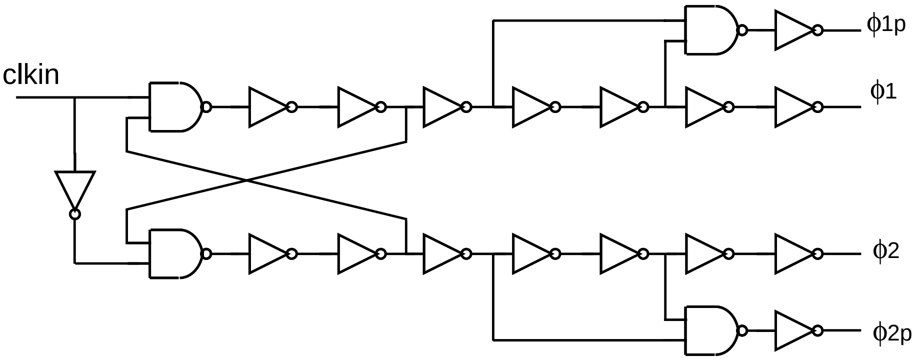


Fig. 2.16 Clock generator

The clock generator shown in Fig. 2.16 can be used to generate the four clock signals shown in Fig. 2.6 from the input reference clock. The non-overlap time and the early falling time can be adjusted easily by changing the number or the unit delay time of the delay cells in this clock generator. The size of these delay cells can be minimized to reduce the power consumption and the noise injection to the substrate and power supply. However, the jitter added in the clock generator will increase accordingly with smaller transistor size. Moreover, smaller transistor size makes the circuit more sensitive to the layout interconnection parasitic capacitance. In practical design, the transistor size should be optimized considering the trade-off mentioned above and the specific design requirement. Another thing needs to mention is that minimizing the input-output delay of the clock generator can help reduce the added jitter in clock generator.

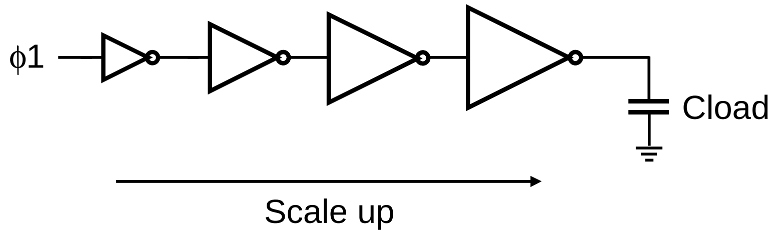


Fig. 2.17 Clock buffer

Except in those very low-speed switched capacitor circuit, usually clock buffers are needed between the outputs of the clock generator and their loads. These buffers can improve the driving capability of the clock generator so fast rising/falling edge clock waveforms can be maintained even with big capacitive loads. Moreover, the skew and clock rising/falling time can be also tuned by tweaking the clock buffer design. As shown in Fig. 2.17, a typical clock buffer is just a chain of inverters with their sizes scaled up. The scaling ratio is usually about 2.7 for minimum delay but can be varied in practical design depending specific requirement.

### 2.2.5 Reference Buffer

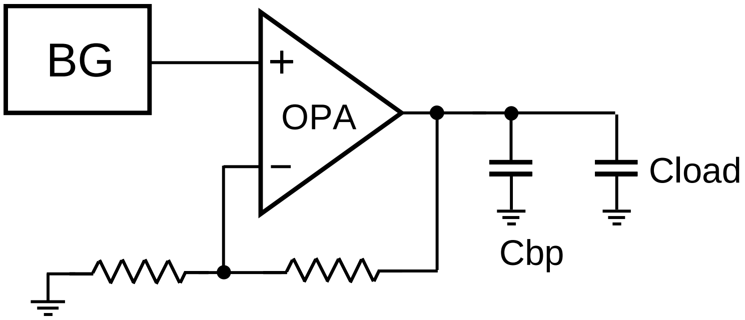


Fig. 2.18 Reference buffer

Although voltage reference circuit is not necessary a part of pipelined ADC, it is worthy of a simple introduction because it is very critical to the performance of the pipelined ADC. The voltage reference can be generated from a band-gap reference circuit amplified/buffered by a non-inverting amplifier shown in Fig. 2.18. While the gain of opamp and the matching of the resistor are usually not very important as far as one reference circuit is shared by all the stages of the pipelined ADC, the bandwidth requirement of a reference buffer is very critical and must be designed carefully. There are two opposite scheme for choosing the bandwidth. First, make the bandwidth very high to satisfy the settling requirement for given total capacitor load. Obviously, this scheme will consume quite amount of power (20%-30% of the overall power consumption a pipelined ADC). Sometime class-AB reference buffer are used to reduce the power consumption. The second scheme is to put a very big bypassing capacitor in parallel with the loading capacitor. Now the transient voltage change at the voltage reference node is approximately:

$$V_{tran} = V_{REF}\frac{C_{load}}{C_{byp}+C_{load}}, \tag{2.2}$$

where $V_{REF}$ is the average reference voltage. It can be seen that the settling accuracy is decoupled from the bandwidth of the reference buffer and is only a function a bypassing capacitor size. Thus, dramatic power saving can be achieved since the bandwidth can be very low. However, the required bypassing capacitor is usually too big to put on-chip and an external bypassing capacitor has to be used. This can limit the speed of the pipelined ADC because of the bonding wire inductance. A simple solution to this issue is to carefully design the capacitor switching scheme so $C_{load}$ takes the same amount of charge from $C_{byp}$ at each clock period. Therefore large transient reference voltage change is not a problem since it is always the same. As a result smaller on-chip bypassing capacitor can be used.

### 2.2.6 Other Circuitry

The other circuitry in the pipelined ADC includes bias generator, digital correction logic, switches, capacitors and resistors. These circuits or components are either very simple or less important and will be discussed wherever needed in the later sections or chapters.

## 2.3 Important Design Issues

### 2.3.1 MOS Switches

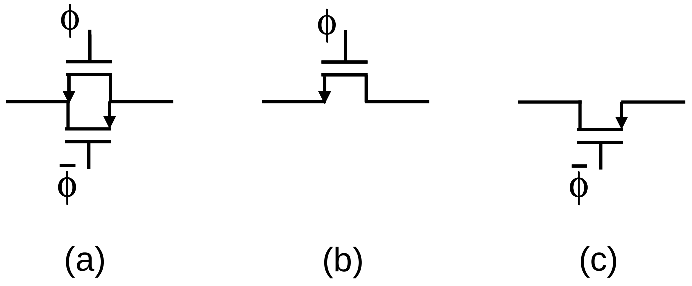


Fig. 2.19 MOS switches

The MOS switches used in SC circuits are shown in Fig. 2.19. Among them, NMOS/PMOS switches are usually used at the node where fixed voltage levels are applied (NMOS switches for lower voltage and PMOS switches for higher voltage), while CMOS switches are usually used in signal path where the voltage is changing between low level and high level. Two major concerns when designing a MOS switch are the on-resistance and charge injection. The on-resistance for a NMOS switch (assume $V_{ds} = 0$ ) can be written as:

$$R_{on} = r_{ds} \approx \frac{1}{\mu C_{ox}(\frac{W}{L})(V_{GS} - V_{tn})}, \tag{2.3}$$

And the charge injection voltage error is:

$$V_{ch} = \frac{Q_{ch}}{C_{eq}} = \frac{WLC_{ox}(V_{GS} - V_{th})}{C_{eq}}, \tag{2.4}$$

Where $C_{eq}$ is the overall capacitance associated with source/drain of the MOS switch. In practical design, the sizes of MOS switches need to be optimized to minimize both on-resistance and charge injection. Furthermore, the nonlinearity of on-resistance must be considered also when the switches are used as the sampling switches in the very first S/H circuit of the pipelined ADC. Otherwise, extra nonlinearity will be introduced by the nonlinear switches. This is particularly important in low voltage and high-speed design.

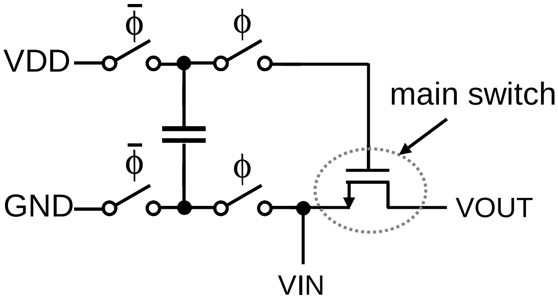


Fig. 2.20 Conceptual bootstrapped switch

To solve this issue, bootstrapped switches instead of simple CMOS switches are usually used as sampling switches to sample high frequency input signal [36][37][38]. Their operation can be illustrated in Fig. 2.20. A constant voltage (VDD in this case) is put across the gate and source of the main NMOS sampling switch to turn on this switch during sampling phase. Thus the on-resistance of the main sampling switch is kept constant even the input signal amplitude is changing quickly. As a result, the nonlinearity from sampling switches is eliminated. A practical implementation of bootstrapped switch proposed in [37] is shown in Fig. 2.21. It is much more complex than the common CMOS switch, therefore usually only used as the sampling switch in the S/H stage.

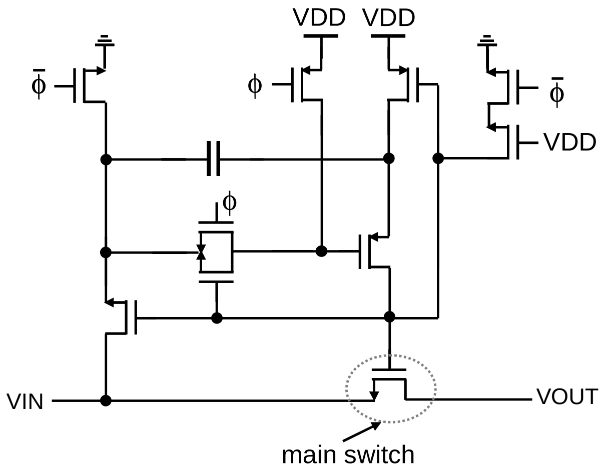


Fig. 2.21 Implementation of bootstrapped switch

### 2.3.2 Noise

Two type of noise can be added into the input signal during the A/D conversion. One is the external interfering noise such as digital switching noise and power supply noise. It can be coupled into the ADC through power supply, substrate and interconnection. The other is the internal circuit noise generated from the transistors and resistors used in the ADC design as well as the quantization noise due to the quantization. Since the quantization noise is fixed after the ADC's resolution is determined, here we only focus on how to reduce interference noise and circuit noise.

While the external interference can be suppressed effectively by good layout skills and fully differential architecture, the internal circuit noise must be taken care in the beginning of the pipelined ADC design. The total input referred noise in a pipelined ADC is:

$$V_{n\_in}^2 = V_{n\_SH}^2 + V_{n\_1}^2 + \frac{1}{2^{2m}} V_{n\_2}^2 + \frac{1}{2^{4m}} V_{n\_3}^2 + \cdots + \frac{1}{2^{2km}} V_{n\_k}^2 + \cdots, \qquad (2.5)$$

where $V_{n\_SH}^2$ is the noise in S/H stage, $V_{n\_k}^2$ is the noise in the k*th* stage and $m$ is the inter-stage gain of each stage. Since the noise from the following stages is suppressed by the inter-stage gain, the noise in S/H stage and the first stage pipeline is the dominant noise source and need to be controlled carefully in the pipelined ADC design.

The input referred noise in S/H or MDAC due to switch resistance and opamp can be calculated as:

$$V_n^2 \approx \frac{kT}{C_s} \cdot \frac{C_{total}}{C_s} + V_{opamp}^2(\omega) \cdot \omega_{noise} \cdot \frac{G_n^2}{G_{sig}^2} \qquad (2.5)$$

where $C_s$ is the sampling capacitor; $C_{total}$ is the total capacitor used in S/H or MDAC, $V_{opamp}^2(\omega)$ is the opamp's input-referred noise power density, $\omega_{noise}$ is the opamp noise bandwidth, $G_n$ is the opamp noise gain, $G_{sig}$ is the signal gain. If we only consider thermal noise since it is the dominant source in high-speed pipelined ADC and ignore the 1/f noise. Eq. (2.5) can be written as:

$$V_n^2 \approx \frac{kT}{C_s} \cdot \frac{C_{total}}{C_s} + \gamma \cdot \frac{\beta}{C_c} \cdot \frac{G_n^2}{G_{sig}^2} \qquad (2.6)$$

where $\beta$ is the feedback factor, $C_c$ is the opamp compensation capacitor and $\gamma$ is a constant determined by the actual opamp design. Several design guidelines for low noise operation can be observed from above equation. First, the circuit noise is determined by the sizes of sampling capacitor and compensation capacitor. Increasing the size of these capacitors can reduce the noise. However, it will increase the die area and power consumption. In practical design, the sizes of these capacitors should be optimized considering this power-noise tradeoff. Second, the capacitors which are not connected to the input as sampling capacitor during sampling phase will contribute extra noise. So it is not a good practice to have separate feedback capacitor or reference sampling capacitor during sampling phase in terms of low noise operation. Third, the opamp should be optimized to achieve small noise factor $\gamma$. For example, we can reduce the trans-conductance of the active load of the input differential pair in the opamp to reduce the noise contribution from these transistors.

One thing need to mention is that the noise from the resetting switches and the switches connecting the feedback capacitor to the opamp's output during holding/amplifying phase is assumed to be very small and ignored in Eq. (2.5). However, this is not always true in practical circuit design. In reality, the noise of these switches is probably comparable to the noise due to the sampling capacitor.

Some designers take it into account by adding another ${kT}/{C_s}$ noise in Eq. (2.5). This is probably a pessimistic estimate since this noise will be filtered by the opamp which usually has lower bandwidth than the RC network formed by the switches and associated capacitors. In practical pipelined ADC design, the more accurate noise analysis can be obtained by indirect simulation using AC noise analysis or direct simulation by PSS and PNOISE analysis in SPECTRE.

### 2.3.3 Opamp Gain and Bandwidth

Finite opamp DC gain will create a non-zero voltage at the virtual ground, therefore makes the charge transfer-function of S/H or MDAC inaccurate. This effect can be modeled as adding a voltage error to the ideal opamp output:

$$V_{out} \approx V_{out\_ideal} + V_{err\_opamp} \tag{2.7}$$

While this error voltage is proportional to the input voltage, its largest value is bounded by the opamp's output signal range:

$$V_{err\_opamp} \leq \frac{V_{ref}}{a\beta} \tag{2.8}$$

where $a$ is the opamp DC open loop gain, $\beta$ is the feedback factor and $V_{ref}$ is the reference voltage value which determine the signal range. Eq. (2.8) can be used to set the spec. for each stage's opamp DC gain. For example, in a 10b pipelined ADC with 1.5b/stage, the first stage's opamp gain should be at least 60dB if the feedback factor is 0.5. In practical design, at least 70dB or even 80dB gain should be achieved for opamp design to guarantee robust operation. Similar to the noise calculation, the errors due to finite opamp gain in the following stages will be suppressed by the interstage gain of the pipeline ADC. Thus, the opamp gain requirement is scaled down with the pipeline.

The opamp's closed-loop bandwidth determines the output settling accuracy of S/H or MDAC. The settling error due to opamp's finite bandwidth (single pole system) can be written as:

$$V_{err\_set} \approx e^{-t_{set}/\tau} V_{set} = e^{-t_{set}\omega_{CL}} V_{set} \qquad (2.9)$$

where $t_{set}$ is the available settling time which is around half period of the clock cycle, $\tau$ is the settling time constant determined by the opamp's closed-loop bandwidth $\omega_{CL}$, $V_{set}$ is the opamp's ideal output voltage step for settling which is usually in the range of 0 to $V_{ref}$. Eq. (2.9) can be used to set the bandwidth requirement for the opamp design. For example, again in a 10b pipelined ADC with 1.5b/stage, if the clock frequency is 100MHz, the clock period will be 10ns, so the maximum settling is about 5ns. The settling error of the first stage needs to be less than 0.1% to give 10b accuracy. Then the required opamp closed-loop bandwidth is:

$$f_{CL} = \frac{1}{2\pi} \cdot \frac{-\ln(1/2^{10})}{T/2} = \frac{1}{2\pi} \cdot \frac{7}{5\times10^{-9}} = 223\text{MHz} \qquad (2.10)$$

The relationship between opamp's unit-gain bandwidth and opamp's closed-loop bandwidth is:

$$f_{CL} = f_{UGB} \cdot \beta \tag{2.11}$$

So to maximize opamp gain-bandwidth and feedback factor can lead to maximum closed-loop bandwidth and then minimum settling time or settling error.

### 2.3.4 Capacitor Mismatch

Capacitor mismatch is another major error source in pipelined ADC and can affect the linearity directly. Therefore must be considered from the very beginning of the ADC design. The differential non-linearity (DNL) due to capacitor mismatch of the first stage, normalized to the least significant bit (LSB), can be written as [7]:

$$DNL = \frac{\lambda \cdot 2^{N - m/2}}{\sqrt{C_{total}}} \tag{2.12}$$

where $N$ is the resolution (bits) of the overall ADC, $m$ is the resolution (bits) of the first stage, $C_{total}$ is the total sampling capacitor in the first stage and $\lambda$ is the constant related to the random variation of capacitance. From Eq. (2.12), it can be seen that the DNL improves by a factor of $\sqrt{2}$ with every extra bit in the first stage or doubling the total capacitance in the first stage. This also means the DNL can be improved by resolving more bits in the first stage for given total capacitance. That's probably the main reason that most people prefer multibit/stage architecture for high resolution pipeline ADC design. However, the integral non-linearity (INL) is not affected by the stage resolution and only depends on the total capacitance:

$$INL = \frac{\lambda \cdot 2^N}{\sqrt{C_{total}}} \tag{2.13}$$

The interesting truth here is that INL follows the same rule as the ${}^{kT}\!/_{C}$ noise: both of them can only improved by increasing the total capacitance. Like the finite gain effect, the requirement on capacitor matching is also scaled down with the pipeline.

### 2.3.5 Charge Injection and Clock Feed-through

Charge injection and clock feed-through can cause differential signal error and/or common mode level shift. The signal dependant charge injection due to the switches connected to the bottom plate of the sampling capacitor can be eliminated by turning off the switches connecting the top plate to ground node earlier. Fully differential architecture can cancel the errors due to clock feed-through. And it is always helpful to insert dummy switches at sensitive nose such as virtual ground to absorb the charge injection. Finally, it is a good practice to try to use as less switches as possible and make them as small as possible (but enough to satisfy on-resistance requirement).

## 2.4 Advanced Design Techniques

As with most analog circuits, pipelined ADC designs suffer from trade-offs among various parameters. The main trade-off exists among power, speed and dynamic range as illustrated in Fig. 2.22. In addition, the die area, design complexity and cost are also need to be considered. Facing these trade-offs, a good designer should adopt appropriate techniques to optimize the design for the best figure-of-merit. Besides the standard pipelined ADC design techniques described earlier, there are many more advanced techniques developed for optimizing the overall design and enhancing the performance of pipelined ADCs.

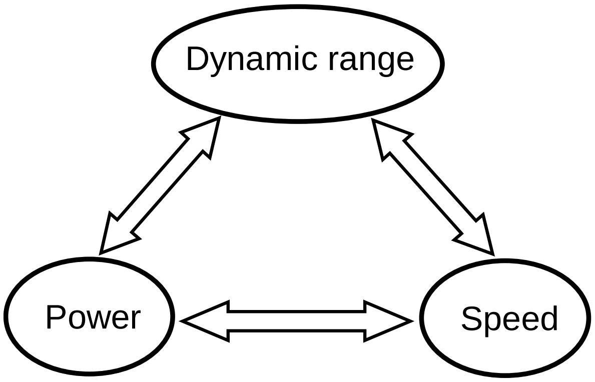


Fig. 2.22 Dynamic range, power and speed trade-off

The most commonly used design optimization technique in pipeline ADC design is capacitor scaling. As described earlier, the capacitor size is determined by $kT/C$ noise and matching requirements which are decreased down the pipeline. So it is very natural to scale down the sizes of the capacitors and their associated opamps from the first stage to the last stage in the pipeline. As a result, 20%-30% power and area saving can be achieved easily even in 10b ADC. And more power and area saving can be expected for a higher resolution design.

While 1.5bit/stage architecture is very popular for its simplicity and high speed, many designers choose multibit/stage architecture for better DNL and lower power consumption, particularly in high resolution ADCs (more than 12b). Then the choosing of stage resolution becomes a critical issue. Higher bits/stage means less number of stages. However, each opamp's feedback factor will be small and consumes more power to keep the same speed. Moreover, the number of comparators in each stage will increase exponentially with stage resolution. So it is not easy to determine the optimum stage resolution. This issue gets more complicated when entangled with the capacitor scaling. While the detailed analysis can be found in [39][40], some architecture level simulation incorporating accurate process information can be done to get the optimum point.

Another very effective power saving technique is opamp sharing [41][42][43] which is proposed based on the observation that, in standard pipeline ADC design, the opamps are only resetting during one clock phase. So they can be shared between two cascaded stages by serving each stage alternately during two clock phases. This technique can reduce the opamp number. As a result drastic power and area save can be achieved. The main drawback of this method is the memory issue to opamps since they are never reset. A new opamp sharing configuration is just developed recently which can reduce the memory voltage to 1/3 of conventional configuration [6]. Another minor issue in opamp sharing technique is, since opamp are being switched back and forth between two stages, more switches are needed and charge injection is increased.

While the power saving can be achieved by applying the approaches introduced above, the speed of pipelined ADCs can be improved dramatically by using multi-channel architecture with time interleaved operation [44][45]. Although the area and power also increase in this case, it is still worth if the conversion rate is the primary concern and is limited by the process. The main design issue in multi-channel ADC design is the channel mismatches including offset, gain and timing mismatch. Offset mismatch can produce tone at ${}^{f_s}\!/_{M}$ , where $f_s$ is the sampling frequency and M is the numbe of channels. Gain and timing mismatch can generate the tone at the image frequency of the input signals. Usually certain calibration is needed to eliminate these mismatches even at 10-b level design [46].

Besides the power optimization techniques and speed boosting technique, there are many accuracy enhancement techniques to improve the accuracy of pipelined ADCs which is limited mainly by component mismatches and finite opamp gains, for example, digital self-calibration techniques, background calibration techniques and other analog techniques. The details of them will be described in the next chapter.

# 3 ACCURACY ENHANCEMENT TECHNIQUES

The accuracy of pipelined ADCs is limited mainly by finite opamp gains and capacitor mismatches. During the not very long history of monolithic pipelined ADCs, many correction techniques have been developed to exceed these technology limits. And these accuracy enhancement techniques can be categorized into three types: analog correction, digital self-calibration and background calibration. The key concepts of them will be explained in this chapter.

## 3.1 Analog Correction

In analog correction techniques, the errors due to component mismatches and finite opamp gains are corrected in analog domain with extra analog circuitry. The advantage of analog correction techniques is low complexity. But like any other analog circuit, they are usually sensitive to process and environment variation. And the conversion rate may be compromised. Ratio-independent multiplication, reference-refreshing, capacitor error averaging, analog trimming and correlated double sampling (CDS) technique are such kind of classical analog correction techniques.

Ratio independent multiplication [8] can be illustrated in Fig. 3.1. The input signal is sampled twice by the same capacitor. The charge collected during the first sampling is transferred and stored temporarily in another capacitor. After the second sampling, this stored charge is transferred back to the sampling capacitor again to realize voltage amplification. Note the accuracy of this amplification is independent of capacitor ratio because there is no net charge transferring between two capacitors. However, four steps are required to implement this method. As a result, the conversion rate is almost reduced by half given the same opamp settling time.

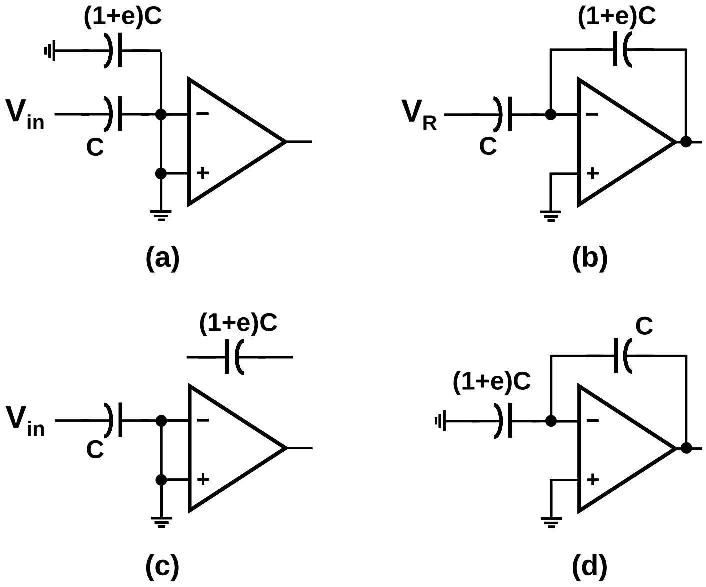


Fig.3.1 Ratio independent multiplication

While the ratio independent multiplication approach tries to avoid the errors due to capacitor mismatches, the error averaging technique [10] cancels them in three steps with two switched capacitor amplifiers as illustrated in Fig.3.2. During the sampling phase, the input signal is sampled by the first amplifier. Next, during the amplification phase, the sampled input signal is amplified by the first amplifier like regular MDAC. In the mean time, the output of the first amplifier is sampled by the second amplifier. Note this output contains the error due to capacitor mismatch.

Finally, during error averaging phase, the two capacitors in the first amplifier exchange their positions. As a result, the error component in the first amplifier's output changes the polarity. This inverting error passes through the second amplifier and cancels out the sampled error component during amplification phase. So the final output of the second amplifier will be free of error due to the capacitor mismatch in the first amplifier. However, the price is the conversion rate is reduced by 1/3, the power consumption is doubled and the noise also increases significantly.

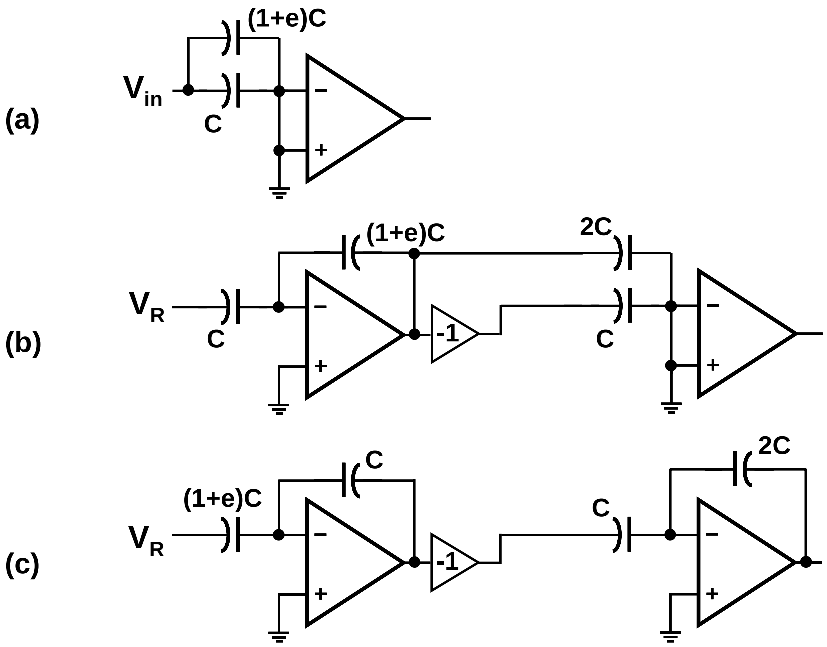


Fig.3.2 Capacitor error averaging

Another analog correction technique is the capacitor trimming [13][47]. It is a very straight forward solution to the capacitor mismatch problem as shown in Fig.3.3. The capacitance of one capacitor in regular MDAC can be tuned to match the other capacitor by connecting a small trimming capacitor to it. In practical circuit implementation, this “trimming capacitor” is usually an array of capacitors controlled by digital logic. The accuracy of this capacitor trimming technique is limited by the accuracy of the small trimming capacitors, therefore sensitive to process variation. Another thing need to mention is the capacitor trimming technique is a foreground correction method which means the normal A/D conversion has to be interrupted to do the trimming.

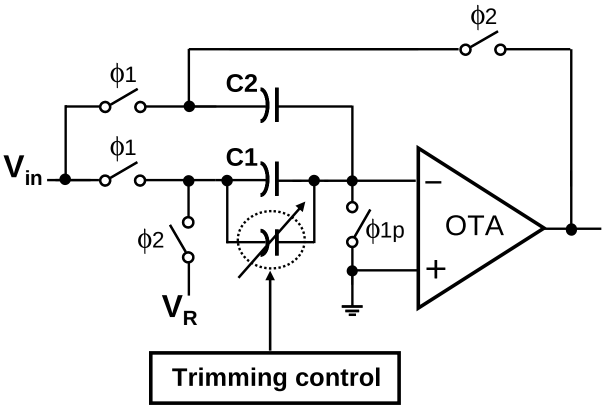


Fig.3.3 Capacitor trimming

An indirect but equivalent solution to capacitor mismatch and finite opamp gain problem is the reference refreshing technique [9]. Instead of making the MDAC as accurate as possible, this technique varies the reference voltage injected at each MDAC to compensate the errors due to capacitor mismatches and finite opamp gain. This can be explained by a simple equivalent transformation:

$$\begin{aligned}
&G_i(V_{in}^{[i]} - D_i V_{ref}^{[i]}) - D_{i+1} V_{ref}^{[i+1]} \\
&= G_i(V_{in}^{[i]} - D_i V_{ref}^{[i]} - D_{i+1}\frac{V_{ref}^{[i+1]}}{G_i}) \\
&= G_i(V_{in}^{[i]} - D_i V_{ref}^{[i]} - D_{i+1}\frac{V_{ref}^{[i]}}{2})
\end{aligned} \tag{3.1}$$

where

$$\begin{aligned}
G_i &= 2(1+e_i) \\
V_{ref}^{[i+1]} &= V_{ref}^{[i]}(1+e_i).
\end{aligned} \tag{3.2}$$

It can be seen that the reference voltage of next stage can be adjusted to compensate the inter-stage gain error caused by capacitor mismatches and finite opamp gain. The accurate adjustment of reference voltage can be done by letting the voltage reference go through the same path as the input signal. However, the implementation is usually complex and requires more clock phases.

While most analog correction methods described above mainly deal with capacitor mismatches, the correlated double sampling (CDS) technique is a very effective approach to reduce the errors due to finite opamp gain and make it inversely proportional to the square of opamp gain [48]-[53]. Equivalently, this doubles the opamp gain in dB. Moreover, the opamp offset is cancelled and 1/f noise is suppressed. The linearity of the amplifier can be also improved dramatically [53]. There are several different CDS schemes. But their principles are similar: stores the non-zero opamp input voltage due to finite opamp gain in a capacitor, then this pre-stored error is used to cancel out the error in the next clock phase. Note there is very small added power consumption in CDS technique since only passive components such as switches and capacitors are needed. This technique will be discussed with more details in the next chapter.

### 3.2 Digital Self-calibration

In the early period of the development on accuracy enhancement techniques, analog correction is the focus of the study. However, with the rapid improvement of fine line CMOS technology and digital signal processing techniques, people realized that it maybe a better solution to measure the errors by ADC itself and correct them in digital domain for the robustness. As a result, several digital self-calibration techniques [14][15][16] were developed to incorporate these two features. And they have been the most extensively used calibration techniques in high resolution pipelined ADC designs so far because of their superior performance.

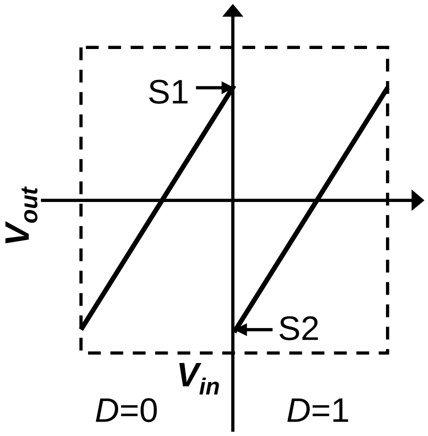


Fig. 3.4 Transfer curve of 1b MDAC

The operation of digital self-calibration can be illustrated in the context of a 1b/stage pipelined ADC for simplicity although it can be applied to multibit/stage ADC as well. Fig.3.4 shows the transfer curve of a 1bit MDAC. S1 and S2 correspond to the quantized residue voltage when $V_{in} = 0$ with $D = 0$ and $D = 1$ respectively, where $D$ is the bit decision of this stage. After S1 and S2 are measured by the following pipeline ADC stages, the calibration of this stage can be done in the digital domain by correcting the raw code:

$$\begin{aligned} &Y = X\text{, if } D = 0 \\ &Y = X + S1 - S2\text{, if } D = 1 \end{aligned} \tag{3.3}$$

where $X$ is the raw code (the digital output of this stage and its following stages) and $Y$ is the final code. This correction algorithm eliminates the gap in the ADC’s transfer curve at $V_{in} = 0$. As a result, a highly linear A/D conversion is achieved even with imperfect MDAC. Note there is still probably a gain error to the overall ADC, but that is usually not a problem in most applications. In the calibration algorithm just described, it is assumed that the following stages are linear. This can be guaranteed by doing the calibration from the last stage to the first stage.

Unlike the analog correction, the digital self-calibration requires no extra analog circuit. This makes the original MDAC design (the most critical part in pipelined ADC) essentially unchanged. So there is no compromise of conversion rate or power consumption due to calibration. Some extra digital circuitry is added, but with the sub-micron CMOS technology, the resulting power consumption and die area overhead will be negligible. More importantly, the calibration is very robust and insensitive to process and environment variation. However, there is one limit: the normal ADC's operation has to be interrupted for the error measurement. Usually, the calibration is done during the system power-up or standby mode to avoid this interruption. However, it is desirable to run the calibration at all time to track the device and environment variation. That necessitates the concept of background calibration which will be introduced in the next section.

## 3.3 Background Calibration

Background calibration is a much more difficult job than foreground calibration. Usually, much more hardware is needed and the performance of ADC is compromised. Here, the main issue is how to isolate the calibration process from the normal operation of ADC. Many clever schemes are developed so far on this isolation. Unfortunately, they all have their limitations. Therefore, background calibration is still a good topic worthy of continuing study.

A very straightforward background calibration scheme is to replace the pipeline stage under calibration with an identical redundant stage. So the normal A/D conversion will not be interrupted [19]. And any suitable foreground calibration approach can be employed to correct the stage being substituted. The main drawbacks of this method is the large overhead of power consumption and die area overhead since one or more extra stages are added. The extra stages also produce troubles to the chip layout which is very critical to the high-speed pipelined ADC design.

Another simple background calibration scheme is to skip some input samples occasionally to create the time slots for calibration [20]. The missing input samples are later filled in digitally via nonlinear interpolation of data. The digital self-calibration technique described earlier can be employed to do the calibration during the idle conversion time slots. Therefore this background calibration scheme has all the desirable features of digital self-calibration: no extra analog circuitry, no need for change of original MDAC design and high calibration accuracy. The only additional circuitry compared to the digital self-calibration is the digital filter block for nonlinear data interpolation. But the resulting overhead for power consumption and die area will be negligible if implemented in sub-micro CMOS process. The main drawback of this method is the input signal frequency has to be limited due to the interpolated regeneration of skipped samples. The detailed discussion of this limit can be found in [20].

A different way for generating the calibration time slots is the use of queue-based architecture [22] shown in Fig.3.5. An analog queue of sample-hold-amplifiers (SHA) is inserted between the input signal and ADC. Its timing scheme is shown in Fig.3.6. The conversion rate of the ADC is slightly higher than the sampling frequency of the first SHA. So the number of samples in the queue decreases overtime until the last SHA is empty. At this time, the ADC can be calibrated. Note the input signal is still uniformly sampled and stored in the queue. So there is no need for the nonlinear data interpolation like skip-and-fill algorithm. But the input signal bandwidth is still limited since the ADC's speed has to be faster than the input sampling frequency. One practical design issue is that the ADC must be capable of fast acquisition because sometimes the available acquisition time is much shorter than half clock period.

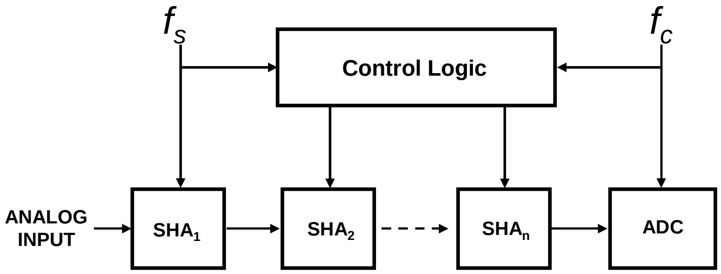


Fig. 3.5 Queue based ADC architecture

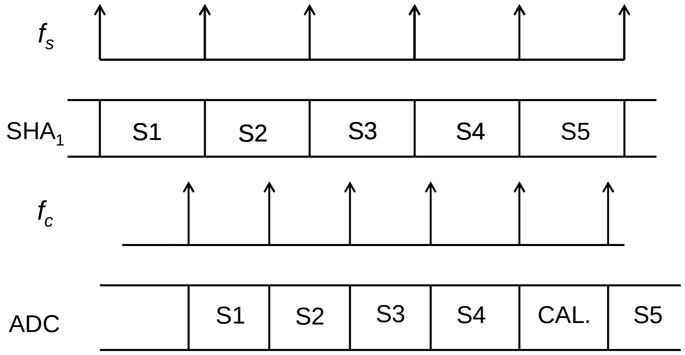


Fig. 3.6 Timing of queue based calibration

If the resistor string MDACs instead of the charge redistribution MDACs are used in a pipelined ADC design, the mismatches of those resistors can be measured by an oversampling delta-sigma ADC and corrected in digital domain without interfering the normal A/D conversion [18]. Since component mismatches change very slowly, the delta-sigma ADC for measurement can be running at low frequency with large oversampling ratio. The resulting overhead for die area and power consumption is very small. And there is no input signal bandwidth limit like skip-and-fill algorithm and queue based calibration. However, it is very difficult to extend this background calibration scheme to charge redistribution MDAC which is more power efficient and widely used.

More recently, correlation algorithm has been adopted in the background calibration of pipeline ADCs to realize the isolation between the calibration process and the normal A/D conversion [23]- [27]. The principle is very simple: the calibration signal is first modulated with a pseudo-random noise sequence in the analog domain. Then it passes through the ADC together with the input signal. Finally it is demodulated in the digital domain to extract the errors in ADC. One example of capacitor mismatch calibration using correlation [25] is shown in Fig.3.7. One extra scrambler is inserted between the sub-ADC and the sub-DAC in regular pipeline stages. This scrambler randomly picks up the unit capacitors in the MDAC based on the output of sub-ADC and the input pseudo-random control sequence. As a result, the difference between each unit capacitor and the average capacitance of all unit capacitors is modulated with a pseudo-random sequence. We can extract this error in digital domain by multiplying the backend ADC’s output with the same pseudorandom sequence. At a first glance, this method seems a perfect solution to background calibration: no extra limit of input signal bandwidth, no redundant analog hardware, no limit on the choosing of MDAC architecture. However, despite the added advantages, the time for error extraction by using correlation is quite long compared to the other background calibration techniques. The previously reported correlation-based schemes are quite complex and slow to converge. This issue will be discussed in more details in chapter 5.

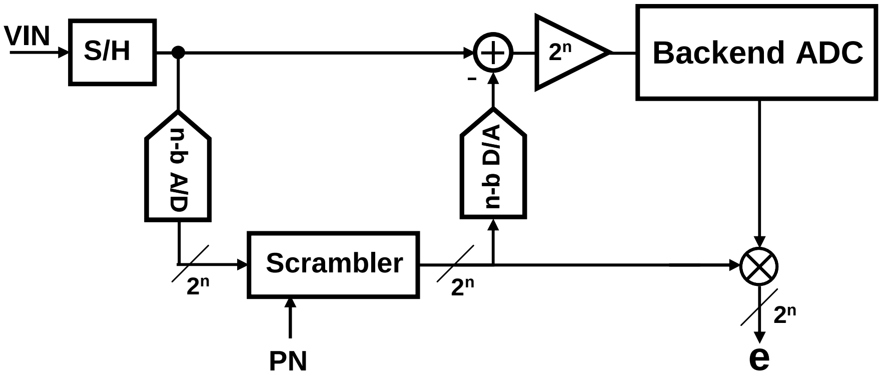


Fig.3.7 Background calibration of capacitor mismatch using correlation

# 4 TIME-SHIFTED CDS TECHNIQUE

In this chapter, a *time-shifted* correlated double sampling (CDS) technique is presented in the context of low-voltage high-speed pipelined ADC design. This technique can significantly reduce the finite opamp gain error without compromising the conversion speed, allowing the active opamp block to be replaced by a simple cascoded CMOS inverter. Both high speed and low power operation can be achieved at the same time without compromising the accuracy requirement. Fabricated in a 0.18μm CMOS process, the prototype 10-bit pipeline ADC achieves 65dB SFDR and 54dB SNDR at 100MSPS. The total power consumption is 67mW at 1.8-V supply.

## 4.1 Introduction

The pipelined ADC architecture has been adopted into many high-speed applications including high performance digital communication systems and high quality video systems. The rapid growth in these application areas is driving the design of ADCs towards higher operating speed and lower power consumption. This trend poses great challenges to conventional pipelined ADC designs which rely on high-gain operational amplifiers to produce high-accuracy converters. Given the continuing trend of submicron CMOS scaling which is coupled with lower power supply voltages and the demand for increased clock speed, large open loop opamp gain is difficult to realize without sacrificing bandwidth [54]. As a result, the finite opamp gain is becoming a major hurdle in achieving both high speed and high resolution. To address this issue, a time-shifted correlated double sampling (CDS) technique is proposed in this chapter. The proposed technique is highly effective for finite opamp gain compensation in the context of low-voltage and high-speed pipelined ADCs. Due to this effective gain compensation, the time-shifted CDS technique has enabled a successful implementation of a low power and high-speed pipelined ADC that uses simple cascoded CMOS inverters in place of traditional operational amplifiers.

## 4.2 Time-shifted CDS Technique

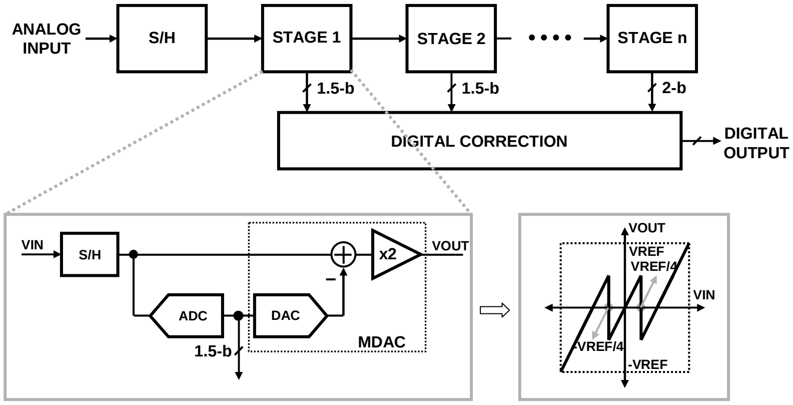


Fig. 4.1 1.5-bit-per-stage pipelined ADC

One of the simplest implementations of pipelined ADCs incorporating digital correction/redundancy is based on the 1.5-bit-per-stage architecture shown in Fig. 4.1.

This architecture is widely used to maximize the conversion speed [55]. Fig. 4.2 shows a typical multiplying digital-to-analog converter (MDAC) structure used in this type of pipeline ADC architecture.

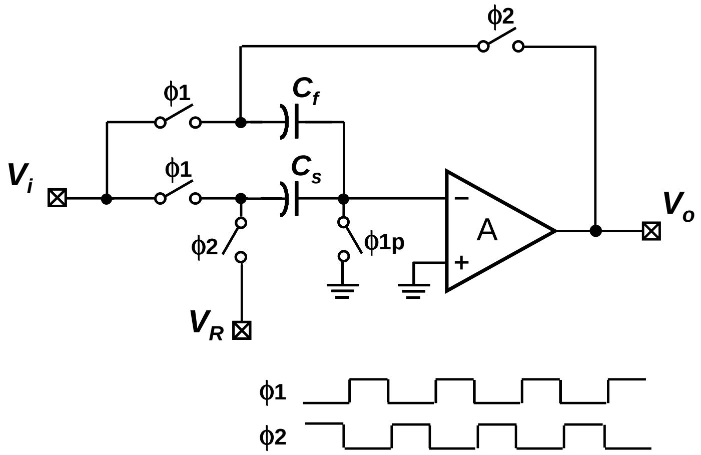


Fig. 4.2 Typical 1.5-bit-per-stage MDAC

The output of this MDAC at the end of the amplification phase ($\phi_2$) is:

$$\phi_2 = \left(\frac{C_s + C_f}{C_f}\right) \cdot V_i - \left(\frac{C_s}{C_f}\right) \cdot V_R + e\,, \tag{4.1}$$

where $V_i$ is the sampled input ($\phi_1$),$V_R$ is $\pm V_{ref}$,0 that depends on the result of the sub-ADC conversion of the sampled input, and the error resulting from the finite opamp gain is:

$$e = \frac{-1}{A}(1 + \frac{C_s}{C_f}) \cdot V_o\,. \tag{4.2}$$

This error *e* is inversely proportional to the opamp gain A, directly deteriorating the overall linearity of the ADC.

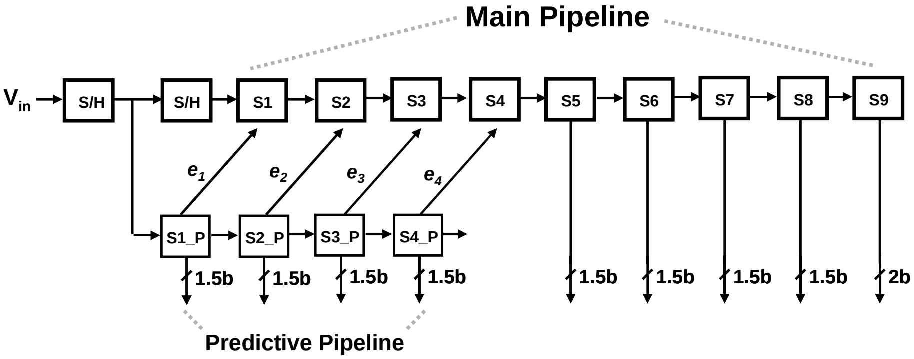


Fig. 4.3 Proposed pipelined ADC architecture

Figure 4.3 illustrates the proposed 10b pipelined ADC architecture employing the time-shifted CDS technique. This architecture realizes two pipelined paths working in parallel for the first few stages. One path represents the *predictive* path which only operates for the first four stages, and the other path represents the main signal path which operates for all nine stages necessary for the 10-bit conversion. The first four stages of the main signal path are very similar to their corresponding stages in the predictive path, and they share the same set of active stages (opamps and comparators). Both signal paths (main and predictive) process the same input signal from the first sample-and-hold (S/H) stage, but the main signal path is delayed a half clock cycle (one phase) by an additional S/H (shares the same opamp) following the first S/H. The input signal is first processed by the predictive pipeline and the finite opamp gain error is stored on a capacitor. The stored error is used to correct the corresponding stage in the main pipeline in the following clock phase (half clock cycle delay). As both signal paths (predictive and main) share the same opamp, this operation is easily achieved with added switches and capacitors.

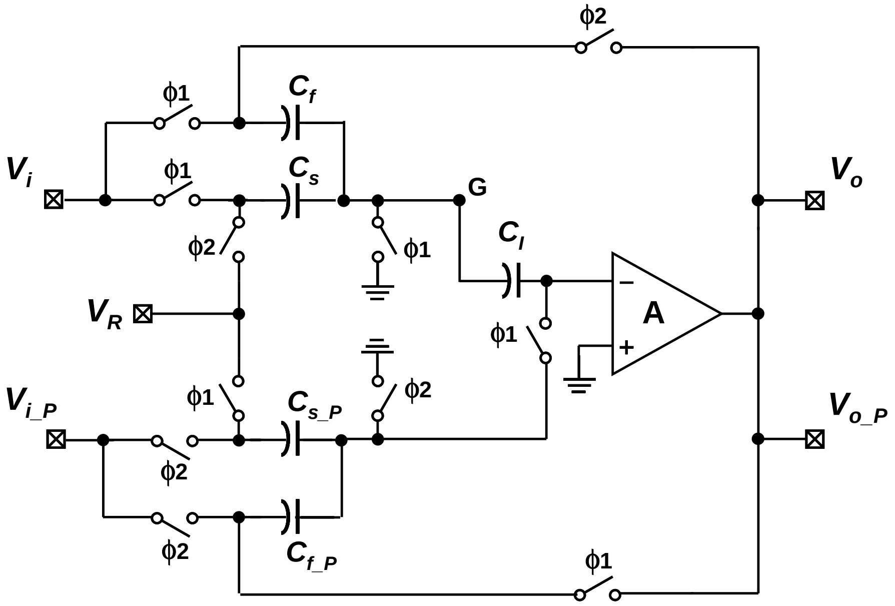


Fig. 4.4 Proposed MDAC architecture

The switched-capacitor implementation of this MDAC operation merging both the predictive pipeline and the main pipeline is shown in Fig. 4.4. $V_i$ and $V_o$ are the input/output of the main pipeline, whereas $V_{i\_P}$ and $V_{o\_P}$ are the input/output of the corresponding predictive pipeline. The capacitors are chosen such that $C_{s\_P} = C_{f\_P} = C_s = C_f$ . In the proposed time-shifted CDS scheme, the sampling and amplifying operation is actually performed twice. The initial/first operation is done by $C_{s\_P}$ and $C_{f\_P}$, and the non-zero error voltage due to finite opamp gain at the negative input of opamp is stored in $C_I$. The following/second operation is done by $C_s$ and $C_f$, with $C_I$ connected between the negative input of opamp and the common node of $C_s$ and $C_f$ (node G). An accurate virtual ground is created at node G. While the operation of this time-shifted CDS technique appears similar to conventional CDS techniques [50], it performs without the additional capacitive load to the opamp and/or the extra clock phase(s) to the ADC operation. Any possible speed penalty due to CDS operation is completed avoided, which is critical in achieving the low-power and high-speed ADC performance.

Some design considerations of the proposed architecture are in the following. First, because the inputs of MDAC are not the same for the predictive path and the main path (with the exception of the very first MDAC), the effect of error correction will not be as good as the conventional CDS techniques (e.g. [50]). The output error at stage *i* in the main pipeline is approximately by

$$e_i = \frac{-i}{A^2}\left(1+\frac{C_s}{C_f}\right)\cdot\left[\left(1+\frac{C_s}{C_f}\right)\cdot V_{oi}(n)-\left(\frac{C_I}{C_f}\right)\cdot V_{oi}\left(n-{}^{1}\!/_{2}\right)\right], \tag{4.3}$$

where $V_{oi}(n)$ is the current output in the main pipeline, and $V_{oi}(n-{}^{1}\!/_{2})$ is the output of previous clock phase (predictive pipeline). Note that the error is inversely proportional to $A^2$. However, this error will increase from stage to stage down the pipeline. This is because the discrepancy between the outputs of the predictive path and the main path will become larger from stage to stage down the pipeline. The second design issue is that the time-shifted CDS will add extra offset to sub-ADCs in the main pipeline. The reason is that the MDACs in the main pipeline need to use the digital code generated by the sub-ADCs in the predictive pipeline. This is equivalent to putting a signal-dependent offset to the sub-ADCs in the main pipeline. Fortunately, digital redundancy of the pipelined ADC is able to correct for the offset, whether signal-dependent or not, as long as the amount of the offset is within the correctable range ($\pm V_{ref}/4$ for 1.5-bit-per-stage MDAC).

Some behavior simulations have been done to verify the effectiveness of the proposed architecture. Simulation results of a 10-bit pipelined ADC (1.5-bit-per-stage) are shown in Fig. 4.5 – Fig. 4.8. In simulation, opamp gain was chosen to be 40dB, the capacitor mismatch was assumed to be less than 0.1\%, and the random offsets of sub-ADCs are assumed to be less than $\pm V_{ref}/8$ . Fig. 4.5 shows the results of the architecture using conventional CDS technique (recall the extra capacitive load and extra clock phase overhead). Fig. 4.6 shows the results of the proposed architecture using time-shifted CDS technique in the first five stages. It can be seen their performance are very close in terms of SNDR, although the proposed architecture does face small but increasing degradation of error correction down the pipeline. The reason why this presumed degradation does not degrade the overall performance too much is that the opamp gain requirement is also reduced down the pipeline as MSBs are resolved. For comparison, the simulation results of regular pipelined ADC without any gain error correction is shown in Fig. 4.7. Note that the SNDR is only about 43dB, which is 16dB lower than the SNDR of architectures with gain error correction. In order to see the effect of hidden degradation of gain error correction that exists in the proposed architecture, the simulation results of the proposed architecture with one more stage using CDS is shown in Fig. 4.8. The SNDR decreases by about 6dB in this case. This indicates that the errors accumulated have overflowed the bounds of digital redundancy in sub-ADCs.

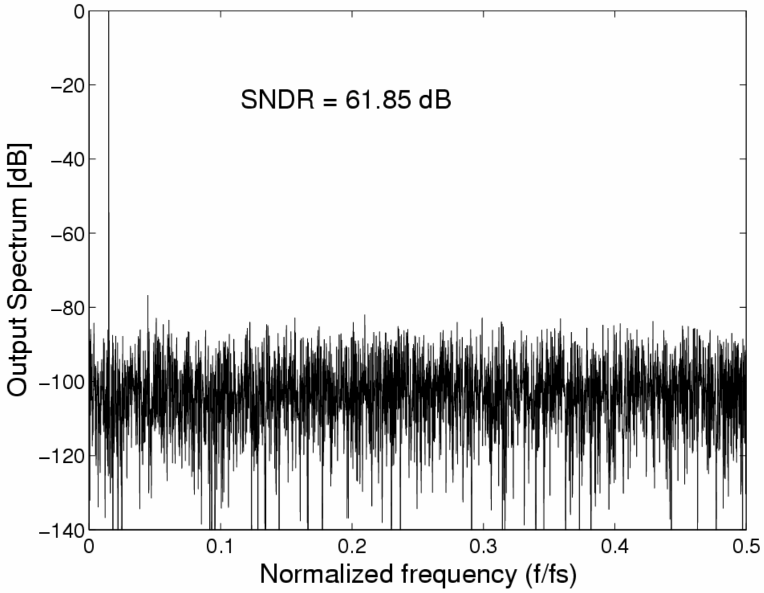


Fig. 4.5 Pipelined ADC employing conventional CDS technique

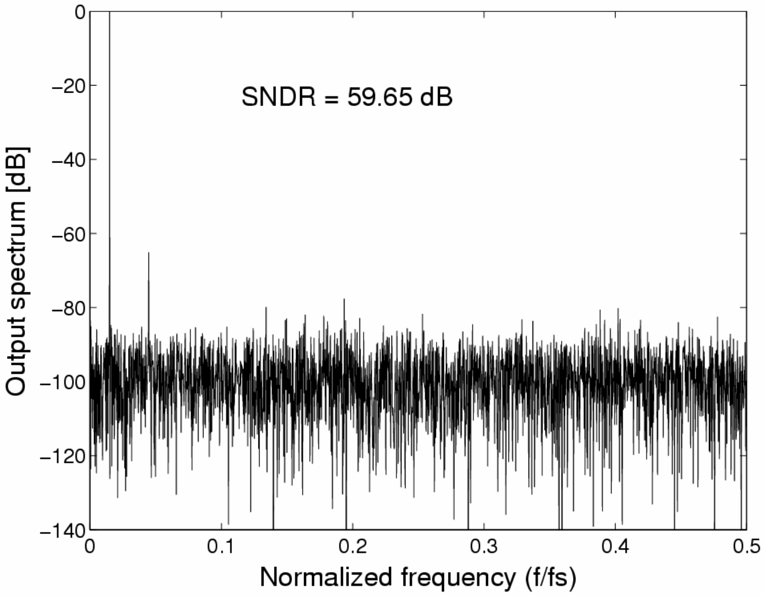


Fig. 4.6 Pipelined ADC applying *time-shifted* CDS to the first five stages

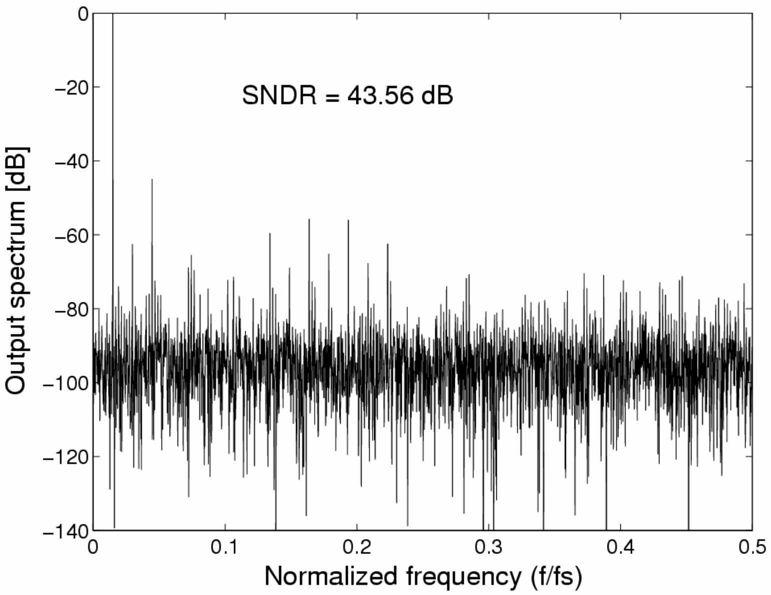


Fig. 4.7 Pipelined ADC without any gain error correction

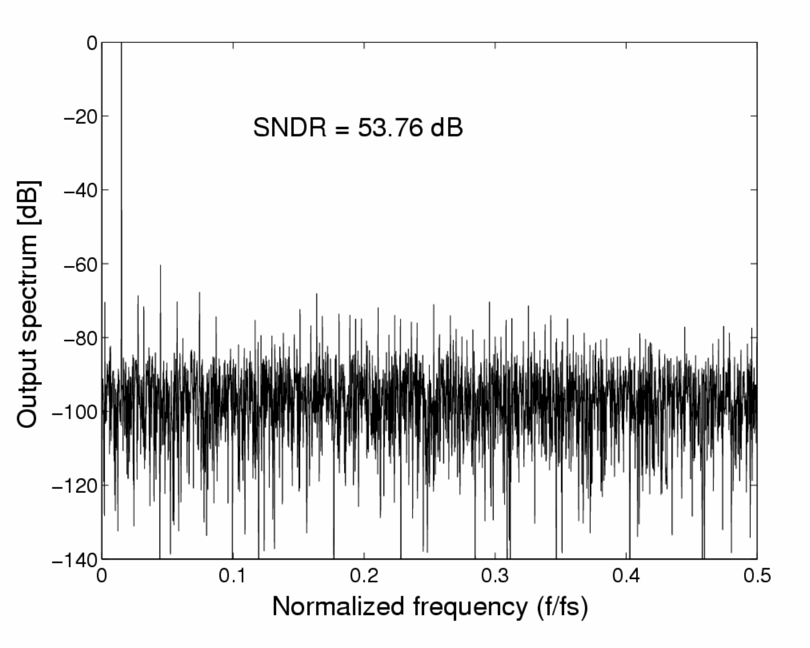


Fig. 4.8 Proposed pipelined ADC applying *time-shifted* CDS to the first six stages

### 4.3 Prototype Pipelined ADC Design

To verify the effectiveness of the proposed time-shifted CDS technique experimentally, a prototype 10-b pipelined ADC was designed in 0.18μm CMOS technology. This design targets faster than 100MSPS operation with minimum power consumption at 1.8V power supply voltage. Besides the time-shifted CDS technique, several other techniques for reducing power consumption and improving robustness have been adopted in this prototype design. The details of main building blocks design will be presented in the following.

### 4.3.1 Opamp Design

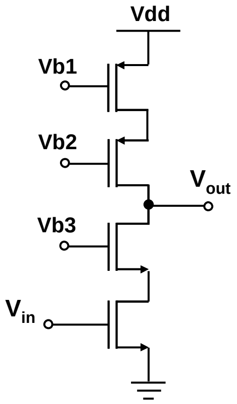


Fig. 4.9 CMOS cascode inverter

Opamp is the most critical building block in pipeline ADCs. The opamp dc gain and bandwidth determine the achievable accuracy and conversion rate. For a 10-bit pipeline ADC, open-loop opamp gain needs to be well over 60dB. It is not uncommon to see 80dB gain in practical design examples. To design such high gain opamp at low supply voltage is quite challenging because traditional stacking of cascode transistors are not feasible. Use of compensated multi-stage opamps will lead to a considerably increased power consumption and reduced speed. In the prototype IC implementation, we used simple cascoded (both the NMOS input and PMOS current source) CMOS inverters shown in Fig. 4.9. Replacing opamps with these inverters allowed wide swing, large bandwidth, and low power consumption. Simulation results have indicated approximately 43dB open-loop dc gain in the 0.18μm CMOS process as shown in Table 4.1. This level of dc gain is insufficient for a 10-bit accuracy pipeline ADC, but we are able to tolerate the low dc gain due to the enhancements achieved from the time-shifted CDS technique described in the above.

| Technology | 0.18μm CMOS |
|---|---|
| DC gain | 43dB |
| UGB | 2.4GHz (with 0.5pF load) |
| PM | $60^o$ |
| Input Cap. | 0.2pF |
| Total Current | 1mA |
| Power supply voltage | 1.8V |
| Output swing | 1Vpp |

Table 4.1 Simulation results of CMOS inverter

### 4.3.2 Pseudo-differential MDAC Design

The use of inverters in place of opamps implies inherently single-ended design. We have adopted pseudo-differential configuration throughout the pipelined ADC design. In other words, two single-ended MDACs in parallel are used to build a pseudo-differential MDAC. As in fully differential circuits, all pseudo-differential structures require some sort of equivalent common-mode feedback (CMFB) operation. Without the equivalent CMFB function, any common-mode error in the pipeline would be amplified just the same way the differential input signal is amplified (residue amplification). This can cause single-ended opamps (inverters) to saturate down the pipeline. Without implementing a traditional CMFB with large amount of overhead, a new pseudo-differential MDAC that uses differential *float* sampling scheme is proposed. This is shown in Fig. 4.10 (time-shifted CDS not shown for simplicity). The differential gain of this MDAC is still two but the common-mode gain is just one due to one of the input capacitor pairs ($C_2$ and $C_3$) that is differentially sampled without a specific common-mode reference (thus floating). This equivalent CMFB operation is achieved with no speed penalty.

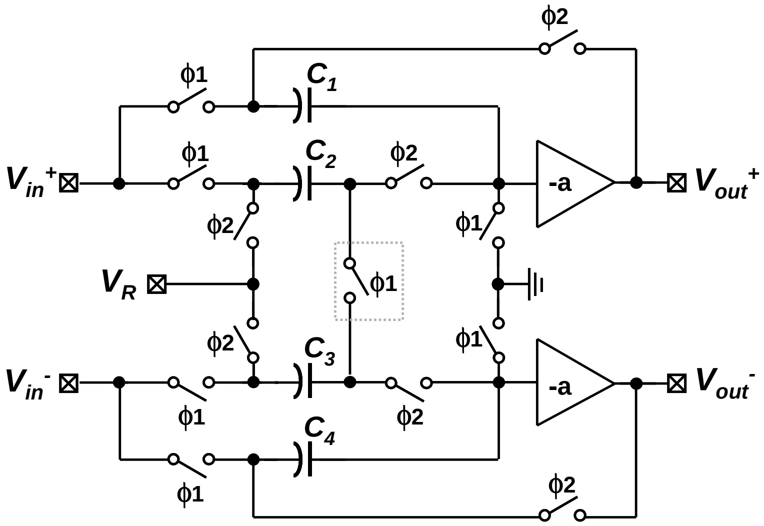


Fig. 4.10 Proposed pseudo-differential MDAC

### 4.3.3 Double Sampling S/H Design

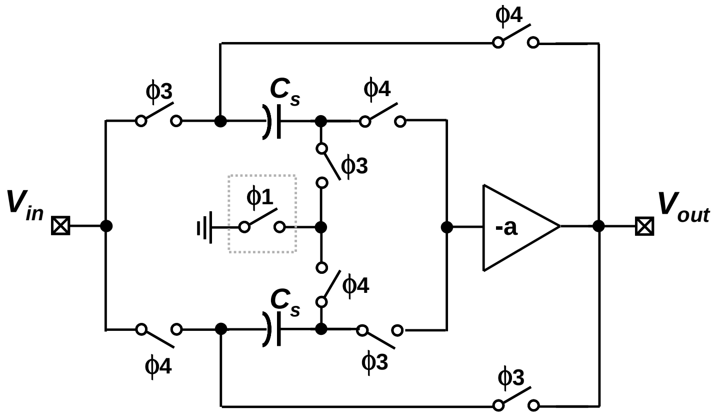


Fig. 4.11 Double sampling front S/H circuit

The proposed architecture shown in Fig. 4.3 indicates that two S/H blocks are needed to apply the time-shifted CDS technique. To minimize power consumption and kT/C noise, a timing skew insensitive double sampling S/H circuit [56] shown in Fig. 4.11 (single-ended illustrated for simplicity) is employed. There are two sets of sampling switches and capacitors for this time-interleaved operation, and they operate at half the speed of the overall ADC. So the output of this S/H circuit will provide a sampled output (hold operation) for the two sampling phases of the first stage pipeline employing the time-shifted CDS. This double sampling S/H circuit is insensitive to the timing skew due to the series master sampling switch [56]. The opamp offset and gain (memory) mismatches are manageable at the 10-bit level. Any capacitor mismatches are alleviated due to inherently voltage-mode operation (i.e. sampled input voltage is “flipped” to the output).

### 4.3.4 Comparator Design

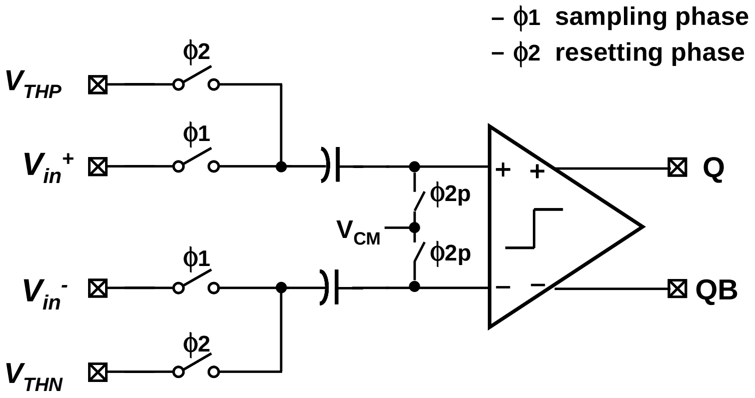


Fig. 4.12 Capacitive coupling comparator

The capacitive coupling comparator shown in Fig. 4.12 is adopted in the sub-ADC design. The input capacitor size is 0.1pF. No offset cancellation scheme is employed because large comparator offsets can be tolerated in 1.5b/stage pipelined ADCs. Time-shifted CDS technique makes this tolerance smaller. However there is no obvious performance degradation even with up to $\pm V_{ref}/8$ comparator offsets as demonstrated in behavior simulation.

The critical part of this comparator module is the latched comparator which is shown in Fig. 4.13. It includes three stages: input amplifier (M1 and M2), NMOS and PMOS regeneration latches (M5-M8), output S-R latch (M13-M20). The input amplifier is just a simple NMOS differential pair with 300uA bias current, which not only amplifies the input signal but also suppresses the kick-back noise from the regeneration latches. The NMOS switches (M3 and M4) will turn off the input differential pair during regeneration time to save power consumption. And it can also help to reduce kick-back noise from the regeneration latches. The combination of PMOS and NMOS regeneration latches can speed up the regeneration compared to the PMOS latches only scheme. The regeneration latches are reset to a voltage close to power supply by M11 and M12 during the sampling/resetting phase. One additional reset switch M10 across the differential latching node can reduce the offset due to the mismatch of M11 and M12. The NMOS switch M9 will disable the NMOS regeneration latch during resetting phase to avoid large DC current to ground. The output S-R latch will hold the comparison result during the whole clock period for the convenience of following encoding logic. With about 0.3mW at 1.8V, this latched comparator achieves less than 250ps regeneration time under 2mV differential input signal, which is short enough for 100MHz clock with 500ps non-overlap time.

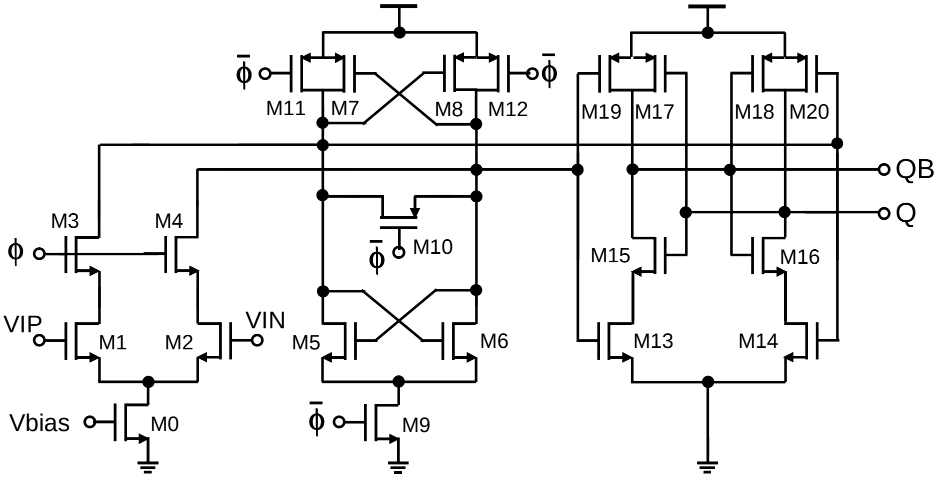


Fig. 4.13 High-speed latched comparator

### 4.3.5 Clock Generator Design

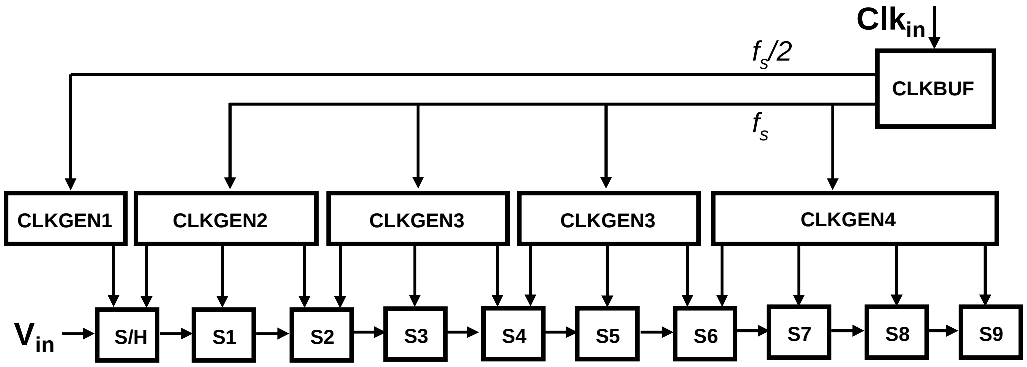


Fig. 4.14 Distributed clock generator

The distributed internal clock generator scheme shown in Fig. 4.14 is used in this design to reduce the load of clock drivers from the parasitic capacitance of the interconnection wires. It also helps to reduce the delay skew due to the interconnection wires. Note two internal clock references are generated from the single input clock signal: one is half rate for the local clock generator which generates the clocks for the input double sampling S/H stage. The other one is full rate for the other local clock generators. The delay matching of these clock signals is very critical to the function as well as the performance. Extensive design and layout optimization (inserting the dummy load and matching the length of clock lines) have been done to minimize the delay skew. The maximum clock skew in the final design is less than 30ps. The typical clock rising/falling time is about 50ps, and the typical clock non-overlap time is about 400ps.

### 4.4 Experiment Results

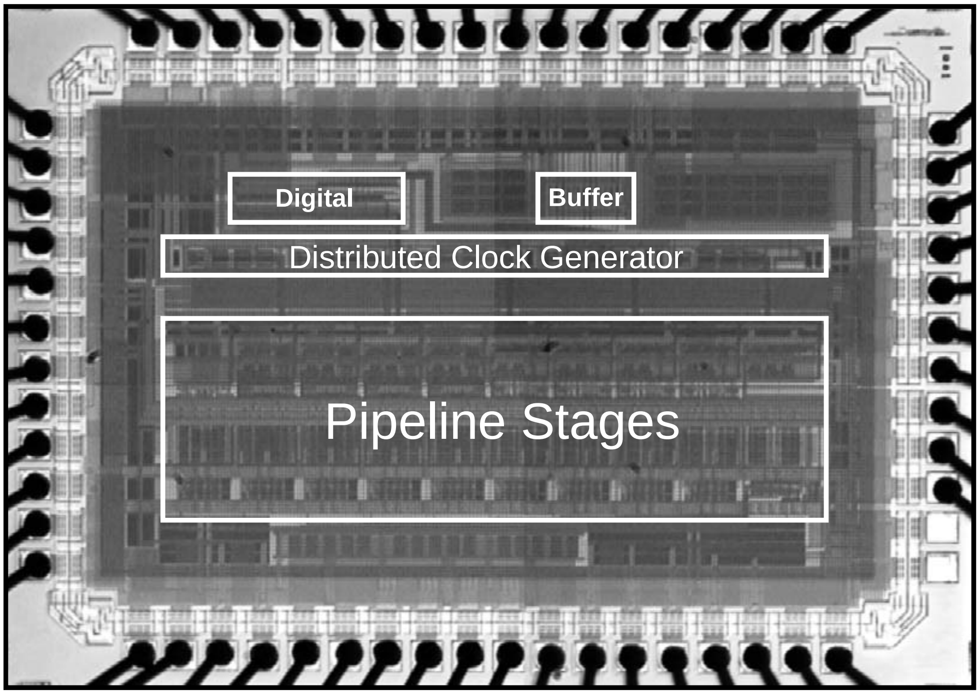


Fig. 4.15 Die photo of prototype pipelined ADC

The prototype ADC was fabricated in a 0.18μm CMOS process. The die photograph is shown in Fig. 4.15, where the active die area is 1.2mm×2.1mm. The total power consumption is 67mW at 1.8-V supply and 100MHz sampling frequency. Analog portion consumes 45mW. The measured DNL and INL are 0.8LSB and 1.6LSB as shown in Fig. 4.16.

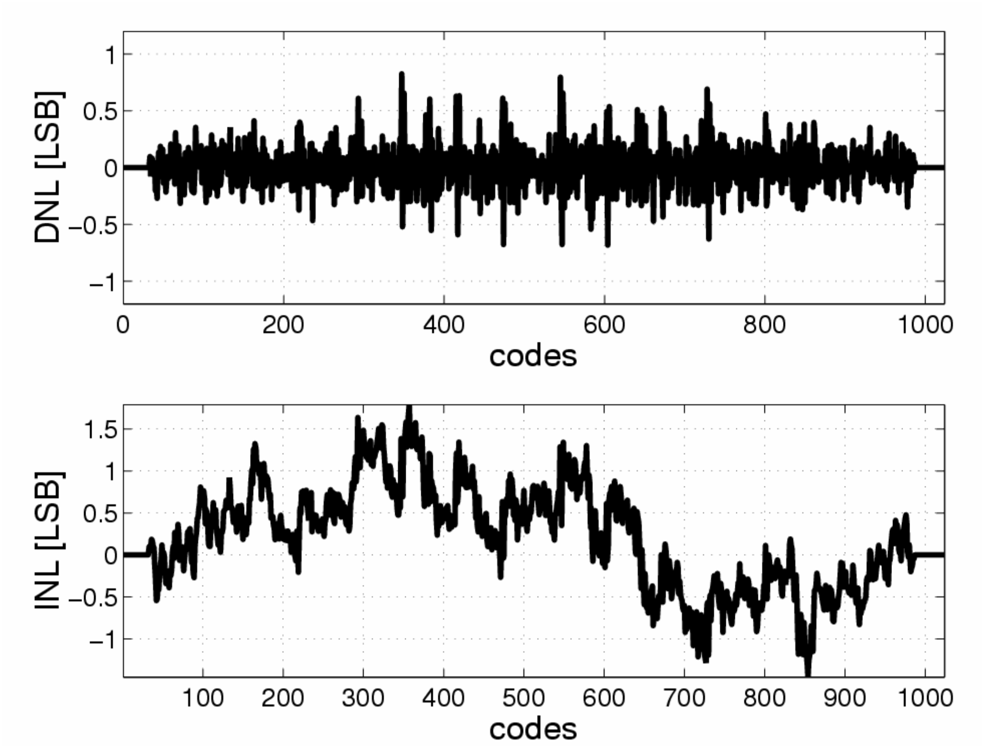


Fig. 4.16 Measured DNL and INL

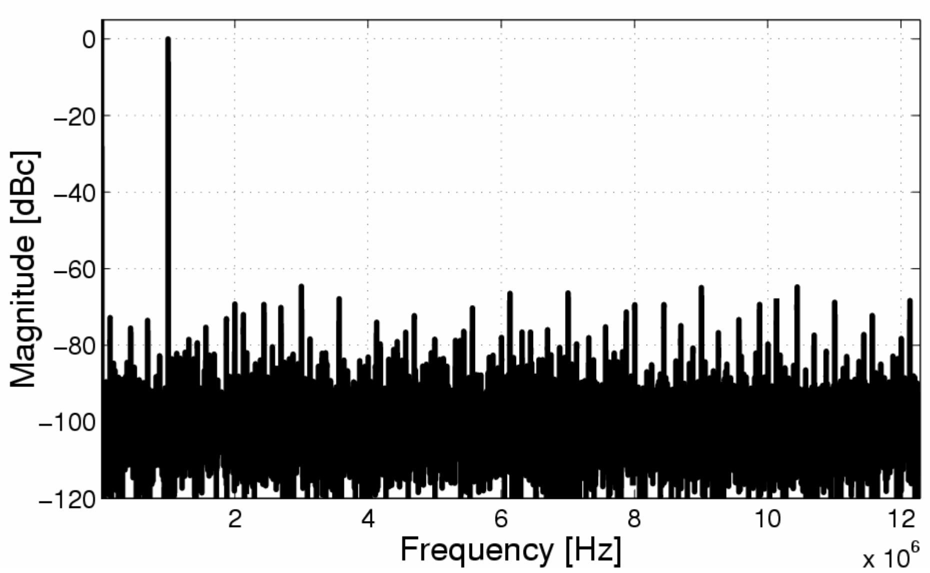


Fig. 4.17 Measured ADC output spectrum at 100MSPS

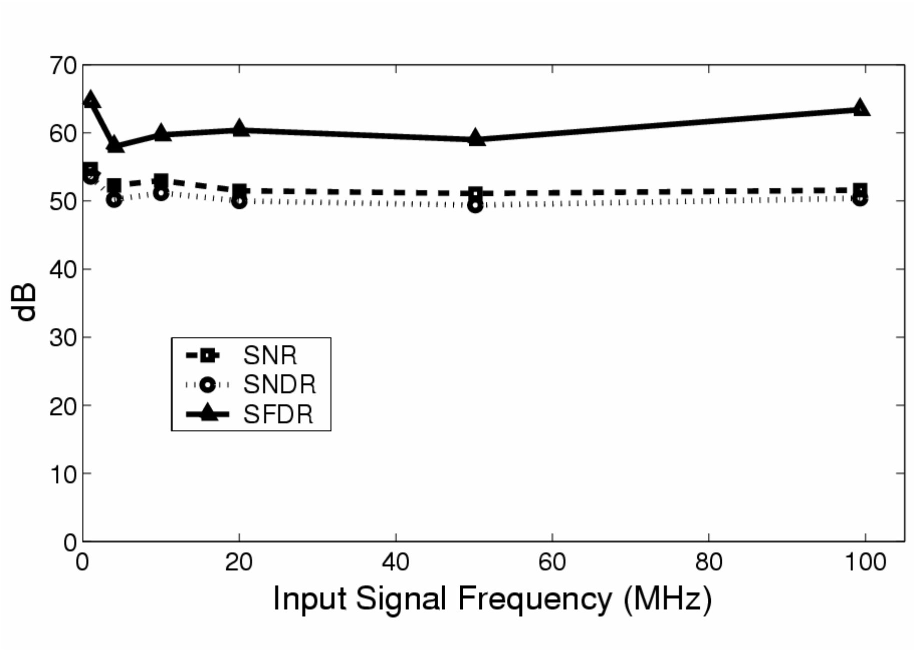


Fig. 4.18 Dynamic measurements vs. input frequency

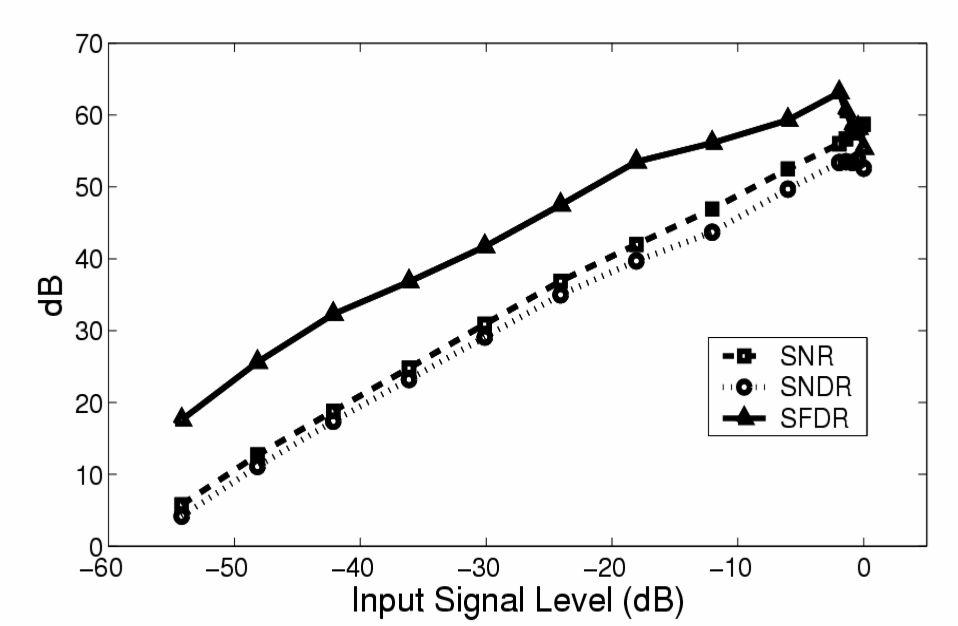


Fig. 4.19 Dynamic measurements vs. input amplitude

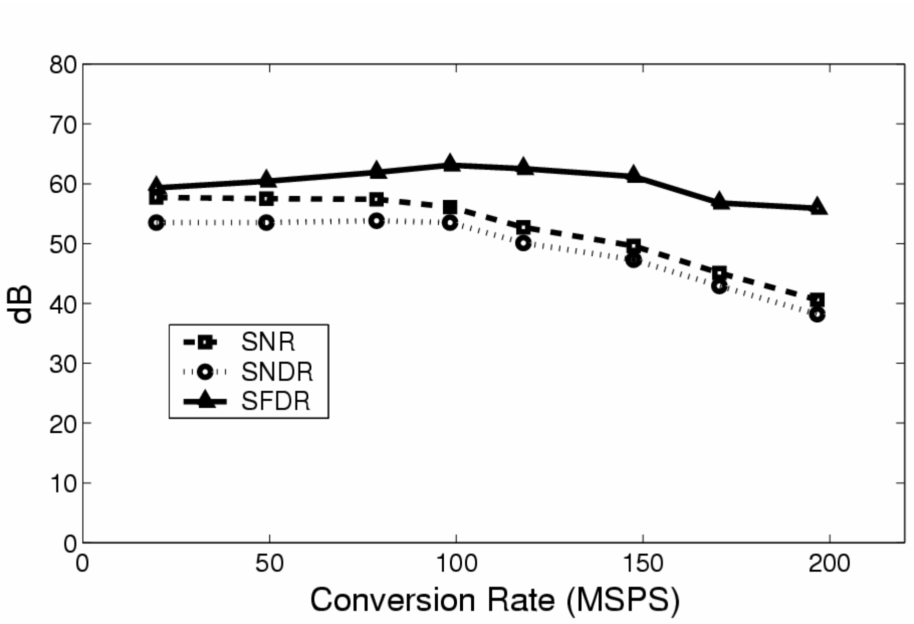


Fig. 4.20 Dynamic measurements vs. conversion rate

With 1MHz input and 100MSPS, the measured SFDR, SNR, and SNDR are 65dB, 55dB, and 54dB, respectively. Figure 4.17 shows a typical measured frequency spectrum at 1MHz input and 100MSPS (the digital output of the ADC is decimated/down-sampled by four on chip for testing purposes). Figure 4.18 shows the dynamic performance versus input frequency at 100MSPS. The measured SFDR, SNR, and SNDR at the 99MHz input frequency are 63dB, 52dB, and 51dB. Figure 4.19 shows the dynamic performance versus input amplitude. The performance is consistent up to -1dB of the target maximum input amplitude. Figure 4.20 shows the dynamic performance versus conversion/clock rate. The performance degrades past 100MSPS. The measurement results are summarized in Table 4.2.

| | |
|---|---|
| Resolution | 10-bit |
| Sampling Rate | 100MSPS |
| Technology | 0.18μm CMOS |
| Supply Voltage | 1.8V |
| Active Die Area | 1.2mm×2.1mm |
| Power Consumption | 67mW |
| DNL/INL | 0.8LSB/1.6LSB |
| SNR/SNDR/SFDR | 55dB/54dB/65dB |

Table 4.2 Performance summary of the prototype ADC

### 4.5 Summary

A time-shifted CDS technique which compensates for the finite amplifier gain of inverter-based pipelined ADC is described. The proposed technique enables low-power high-speed operation by allowing significantly reduced amplifier gain without the added overhead of increased power dissipation or extra clock phase(s). Prototype IC measurements demonstrate 67mW 10-bit 100MSPS performance. The achieved results indicate that a design incorporating effective CDS techniques (e.g. time-shifted CDS) in combination with simplistic active stages (e.g. inverter) can achieve significant speed improvement while maintaining, or even lowering, the overall power consumption.

# 5 BACKGROUND DIGITAL CALIBRATION TECHNIQUE

In this chapter, an efficient background digital background calibration technique is presented. The proposed digital background calibration scheme, applicable to multi-stage (pipelined or algorithmic/cyclic) analog-to-digital converters (ADCs), corrects the linearity errors resulting from capacitor mismatches and finite opamp gain. A high-accuracy calibration is achieved by re-calculating the digital output based on each stage's equivalent radix. The equivalent radices are extracted in the background by using a digital correlation method. The proposed calibration technique takes advantage of the digital redundancy architecture inherent to most pipelined ADCs. In the proposed method, the SNR is not degraded from the pseudo-random noise sequence injected into the system. A two-channel ADC architecture with negligible overhead is also proposed to significantly improve the efficiency of the digital correlation. The effectiveness of the proposed calibration technique is demonstrated in simulation as well as experiment.

## 5.1 Introduction

Pipelined ADCs have been used extensively in high performance digital communication systems. While the speed of these state-of-the-art pipelined ADCs has exceeded 100MSPS (mega-samples-per-second) in CMOS technology [1]-[6], the commonly achieved resolution is still bound within the range of 8-12 effective-number-of-bits (ENOBs) due to the limitations set by component mismatches. Use of multi-bit-per-stage architecture and design optimization can achieve 14-bit performance as demonstrated in [7], but most pipelined ADCs with more than 12-bit resolution will usually require some kind of linearity enhancement techniques. Although many accuracy enhancement techniques are already available and work well under certain context, all of them have some drawbacks as discussed in chapter 3. Usually, either speed or power consumption or both has to be compromised. And most of them are difficult to implement in low-voltage and high-speed design. Therefore, simple and efficient accuracy enhancement technique which can be used in low-voltage and high-speed pipelined ADCs would be worthy of focused research.

In the following, the details of a fast and accurate correlation-based background digital calibration scheme in the context of a 1.5-bit-per-stage pipelined or cyclic ADC architecture will be presented. The input signal magnitude needs not to be reduced to allow the injection of pseudo-random calibration signal. The minimal addition of analog hardware for calibration keeps the original ADC design essentially unchanged. The correlation algorithm converges very quickly in the proposed two-channel ADC architecture because of the interference cancelling scheme employed.

## 5.2 Proposed Background Calibration

In the proposed calibration scheme, the errors due to capacitor mismatches and finite opamp gain are corrected by re-calculating the digital output based on the equivalent radix value of each stage. The equivalent radices are extracted on-line using a correlation-based algorithm. To minimize the interference from the input signal in the correlation-based radix extraction, a two-channel ADC architecture (but same total capacitance and power consumption) is also proposed.

### 5.2.1 Radix-based Digital Self-calibration

For the widely used "capacitor-flip-over" MDAC shown in Fig. 4.2, assuming that the ADC is ideal, the analog output of each stage is given by:

$$\begin{aligned} V_o &= 2V_i - D \cdot V_{ref} \\ &= 2(V_i - D \cdot V_{ref}/2), \end{aligned} \tag{5.1}$$

where $D$ is $\pm 1$ or 0 depending on the input voltage level (i.e. the sub-ADC output). In the transistor level circuit implementation, the non-ideal effects such as capacitor mismatches and finite opamp gain will add errors to the conversion. Figure 5.1 shows the functional diagram of a pipeline ADC in the presence of non-ideal terms, $\alpha_i$, $\beta_i$ and $\delta_i$, denoting the errors caused by capacitor mismatches and finite opamp gain. Since the signal-independent charge injection and opamp offset only result in an overall offset to the pipeline ADC, they are not shown in Fig. 5.1. The resulting analog output of an MDAC is:

$$V_o = (1+\delta)((2+\alpha)\cdot V_i - D\cdot(1+\beta)\cdot V_{ref}). \tag{5.2}$$

For a non-ideal pipeline ADC with the error terms mentioned above, the conversion will be inaccurate and calibration is needed for an improved performance.

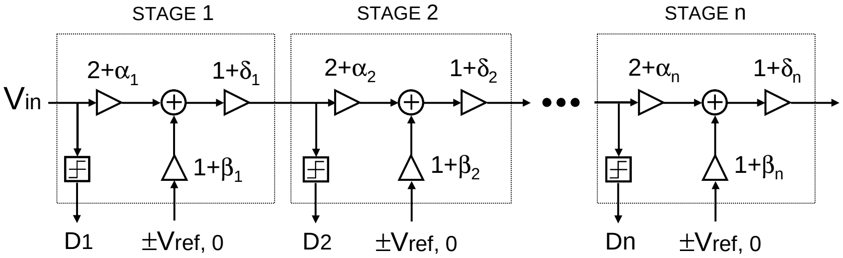


Fig. 5.1 Functional diagram of a non-ideal pipelined ADC (1.5b/stage)

For single stage cyclic/algorithmic ADC, we can rewrite the Eq. (5.2) as:

$$V_o = (1+\delta)(2+\alpha)(V_i - D\cdot\frac{1+\beta}{2+\alpha}\cdot V_{ref}). \tag{5.3}$$

Note that the Eq. (5.3) is equivalent to Eq. (5.1) if we take $(1+\delta)(2+\alpha)$ as the new radix (instead of 2) and $\frac{2(1+\beta)}{2+\alpha}\cdot V_{ref}$ as the redefined reference voltage. The correct digital output of the ADC can now be obtained by using a simple radix calculation based on the modified radix value [22]:

$$D_{out} = D_n + D_{n-1}\cdot(ra) + D_{n-2}\cdot(ra)^2 + \cdots + D_2\cdot(ra)^{n-2} + D_1\cdot(ra)^{n-1}, \tag{5.4}$$

where *ra* is the modified (from 2) radix number taking into account the effects of all error terms. While the single-stage algorithm ADC displays this favorable feature, this algorithm can not be applied to a multi-stage ADC, because the redefined reference voltages (function of capacitor mismatches and finite opamp gain) will be different from stage to stage.

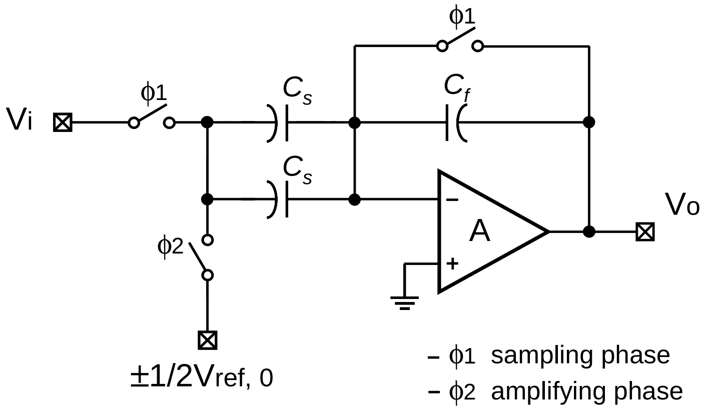


Fig. 5.2 Non-capacitor-flip-over MDAC

To solve this problem, a "non-capacitor-flip-over" MDAC shown in Fig. 5.2 was used in [57]. In this modified MDAC scheme, the analog input $V_i$ and the reference $V_{ref}$ will see identical error terms, so the analog input and output relation can be written as [57]:

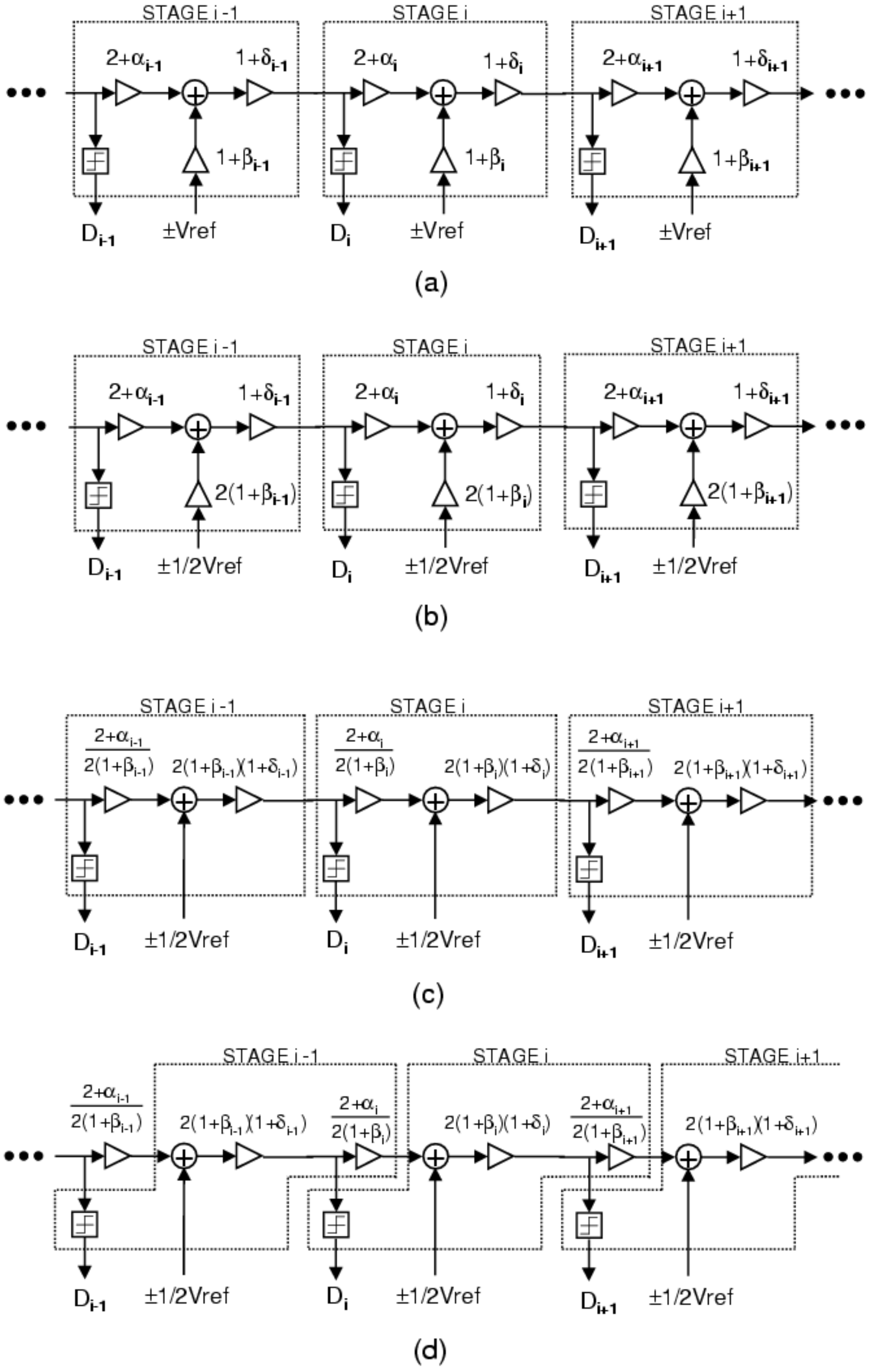
$$V_o = ra \cdot (V_i - D \cdot V_{ref} / 2) . \tag{5.5}$$

Note that the radix number *ra* varies from stage to stage but the reference voltages ($V_{ref}/2$) are same for all stages. The correct digital output of the ADC can be calculated by [57]:

$$\begin{aligned} D_{out} = D_n + D_{n-1} \cdot ra_{n-1} + D_{n-2} \cdot ra_{n-1} \cdot ra_{n-2} + D_{n-2} \cdot ra_{n-1} \cdot ra_{n-2} \\ + \cdots + D_1 \cdot ra_{n-1} \cdot ra_{n-2} \cdots ra_2 \cdot ra_1 \end{aligned} . \tag{5.6}$$

Despite the simplicity found in the “non-capacitor-flip-over” MDAC of Fig. 5.2, the conventional “capacitor-flip-over” MDAC has the speed advantage due to the large feedback factor in the loop settling. Thus it would be desirable to find a solution which can be applied to both MDAC structures.

Such a solution indeed exists and can be achieved easily after some systematic manipulations [58]. To understand how this general solution works, we can apply some equivalent transformations to the pipeline ADC's functional diagram of Fig. 5.1. The transformation procedure is shown in Fig. 5.3. Figure 5.3(a) represents the functional diagram of a pipeline ADC with all error terms. If we change $V_{ref}$ to $V_{ref}/2$ and adjust the gain factor of the reference voltage accordingly, we get the equivalent block diagram Fig. 5.3(b). Then we merge the gain factor of the reference voltage into the gain factors of the input and the output to arrive at Fig. 5.3(c). Now we simply redefine the stage's input and output so that each stage's input gain factor is merged into its previous stage's output gain factor, and the next stage's input gain factor is merged into output gain factor. The resulting equivalent ADC is shown in Fig. 5.3(d). Note that Fig. 5.3(d) is equivalent to the functional block diagram of a pipeline ADC where MDAC’s input and output relationship can be written in the form of Eq. (5.5). The equivalent radix of each stage is now given by:



(a)

(b)

(c)

(d)

Fig. 5.3 Equivalent transformation of a pipelined ADC

$$ra_i = (1+\beta_i)(1+\delta_i)(\frac{2+\alpha_{i+1}}{1+\beta_{i+1}}) , \qquad (5.7)$$

This means we can now use Eq. (5.6) to calibrate multi-stage ADCs with “capacitor-flip-over” MDAC. As a result this radix-based digital calibration scheme becomes a general calibration technique and can be applied to any 1-bit-per-stage pipelined or algorithmic/cyclic ADC without the limitation on the structure of MDAC. The key point in the “capacitor-flip-over” MDAC was to redefine the stage's input and output so that the desired form of equivalent radix can be achieved. One resulting issue with this redefinition is that the comparators (sub-ADC) still see the original input voltage before the redefinition took place. It equivalently adds signal-dependent offset to the comparator (sub-ADC) input. Fortunately, this is not a problem because the added offset is small and can be compensated by the digital redundancy of the pipeline ADC without any performance degradation.

Although this kind of radix-based digital calibration scheme prefers 1-bit-per-stage ADC architecture for its simplicity, it can easily be applied to the 1.5-bit-per-stage architecture with little modification. We can still use Eq. (5.6) to re-calculate the ADC’s digital output. The digital output of each stage is now a three level value (±1 or 0) instead of the two level value (±1) of the 1-bit-per-stage ADC. This is the only modification that we need to make in order for this radix-based calibration scheme to be adapted from the 1-bit-per-stage ADC to the 1.5-bit-per-stage ADC. In an uncalibrated 1.5-bit-per-stage ADC where all radices are “2”, the Eq. (5.6) is equivalent to a commonly used digital correction logic.

### 5.2.2 Background Equivalent Radix Extraction

In [57], the radix extraction/measurement was employed in the foreground to obtain the equivalent radices in a non-ideal pipeline ADC. The ADC operation has to be interrupted to perform this radix extraction/measurement. Because of this drawback, it is highly desirable to develop a background equivalent radix extraction scheme.

Among various background calibration techniques, the correlation-based schemes [23]-[27] are most promising because they involve minimum additional analog circuitry. In the calibration system, the small error terms resulting from capacitor mismatches and finite opamp gain are modulated by a pseudo-random sequence in the analog domain. Then they are converted to digital code along with the input signal to the ADC. These small error terms are detected in the digital domain by correlating the ADC digital output with the same pseudo-random sequence. Applying the correlation algorithm to the radix-based calibration described in the previous section, we can develop a simple and robust background digital calibration technique to pipelined ADCs.

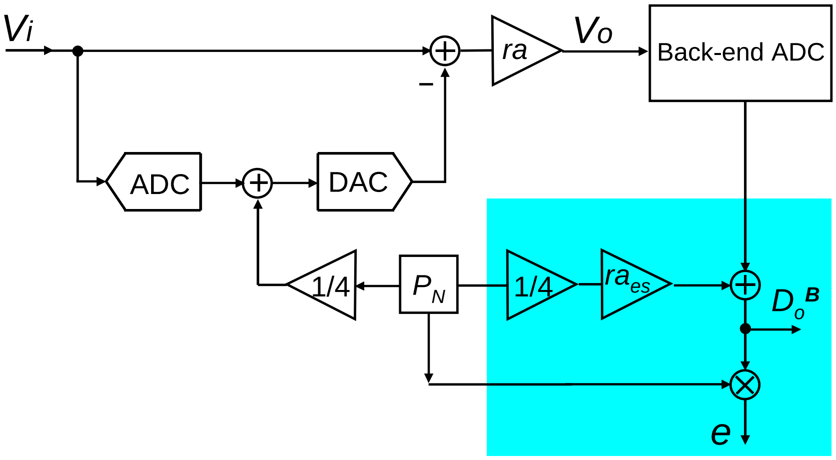


Fig. 5.4 Background equivalent radix extraction scheme 1

Figure 5.4 illustrates one possible (not optimum) background equivalent radix extraction scheme based on correlation method. This calibration scheme incorporates a ±1 pseudo-random noise sequence, which is scaled by a constant (1/4 in this example) and then added to the input of the sub-DAC. This pseudo-random noise travels through the interstage gain block which contains the actual radix number. Then it is quantized by the back-end ADC. Finally, to maintain the same SNR, this added pseudo-random noise is subtracted from the back-end ADC's output in the digital domain. An estimated radix number has to be provided to do this pseudo-random noise cancellation. The resulting final digital output of the back-end ADC is given by:

$$\begin{aligned} D_o^B &= (1/4)P_N \cdot ra_{es} - (Q_N + (1/4)P_N) \cdot ra + O_N \\ &= -Q_N \cdot ra + (1/4)P_N \cdot (ra_{es} - ra) + O_N \end{aligned}, \tag{5.8}$$

where $D_o^B$ is the final digital output of the back-end ADC, $P_N$ is the pseudo-random noise sequence, $ra$ is the actual radix, $ra_{es}$ is the estimated radix, $Q_N$ is the quantization noise of this stage, and $O_N$ includes all other noise sources such as thermal noise and quantization noise of the back-end ADC. Note that the sub-DAC's error has been merged into the actual radix from the radix-based calibration concept described earlier, so it can be seen as a noise-free block. Also, if the estimated radix $ra_{es}$ is not equal to the actual radix $ra$, the added pseudo-random noise will not be cancelled completely. If we now correlate the back-end ADC digital output with the same pseudo-random sequence, we can get the difference between the actual radix and the estimated radix:

$$e = (1/4)(ra_{es} - ra) - P_N \otimes (Q_N \cdot ra - O_N)\,, \tag{5.9}$$

where $e$ is the result of the correlation and $\otimes$ is the symbol for correlation. Ideally, since $P_N$ is uncorrelated with $Q_N$ or $O_N$, their correlation products will approach zero as we increase the length of the pseudo-random sequence. We can ignore their correlation products if the pseudo-random sequence is long enough, and the actual radix can be calculated as:

$$ra = ra_{es} - 4e\,. \tag{5.10}$$

Similar schemes can be found in [24][26] in a slightly different context.

There are some practical issues about the correlation scheme described in the above. The first issue is that the amplitude of the ADC's input signal has to be reduced when injecting this pseudo-random noise. The reason is that each stage's analog output has a maximum full signal range. Therefore the adding of pseudo-random noise can make the ADC stage's analog output to go out of the full-scale signal range if the input signal's magnitude is not reduced. The second issue is how to inject a *scaled* pseudo-random noise sequence in the analog domain with a very accurately known magnitude (i.e. the accuracy of 1/4 in the analog domain). This is not a trivial task in a practical integrated circuit (IC) implementation. The methods proposed in [24][26] add quite a bit of complexity to resolve this issue. For example, a slow-but-accurate delta-sigma ADC was employed to measure the changed reference voltage after injecting pseudo-random sequence [24]. Finally, beyond these two issues, any uncorrected (incompletely cancelled) pseudo-random sequence injected into the system directly degrades the SNR of the ADC under calibration.

In pipelined ADCs which incorporate digital redundancy, there is another way to do the background radix detection. This proposed method is illustrated in Fig. 5.5. Here, the scaled (by approximately 1/4) pseudo-random noise sequence is injected at the input of the sub-ADC instead of the sub-DAC. The resulting ADC digital output (combining this stage ADC and its back-end ADC) is:

$$\begin{aligned} D_o &= (V_i + Q_N + (1/4)P_N) \cdot ra_{es} - (Q_N + (1/4)P_N) \cdot ra + O_N \\ &= V_i \cdot ra_{es} + Q_N \cdot (ra_{es} - ra) + (1/4)P_N \cdot (ra_{es} - ra) + O_N \end{aligned} . \tag{5.11}$$

If we correlate the ADC digital output with the same pseudo-random sequence, we get:

$$e = (1/4)(ra_{es} - ra) - P_N \otimes (V_i \cdot ra_{es} + Q_N \cdot (ra_{es} - ra) + O_N)\,. \tag{5.12}$$

By similar theoretical reasoning, since $V_i$, $Q_N$ and $O_N$ are uncorrelated with $P_N$, we can ignore their correlation products if the pseudo-random sequence is long enough. And the actual radix can be calculated using Eq. (5.10).

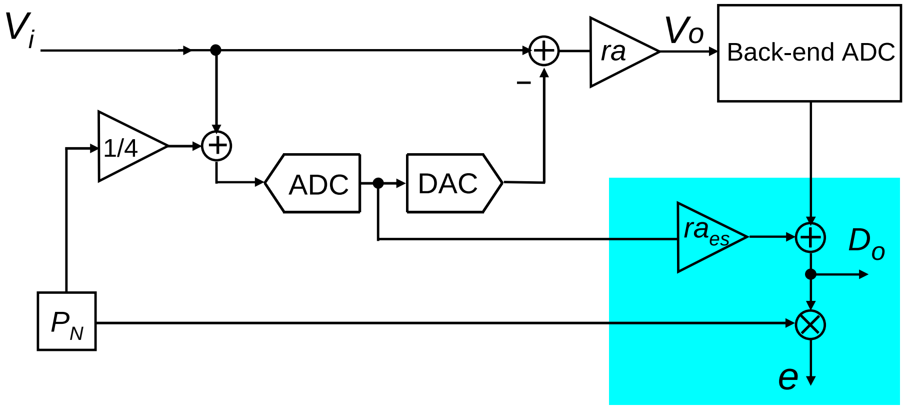


Fig. 5.5 Background equivalent radix extraction scheme 2

It may be seen that the second equivalent radix detection scheme is similar to the first one. The key modification here is to inject the pseudo-random sequence at the input of the sub-ADC instead of the sub-DAC. But this change gives us many advantages as we do not suffer from the practical issues mentioned in the first scheme (Fig. 5.4). First, the injected pseudo-random noise can be seen as being equivalent to time-varying (but limited/bounded) comparator offsets, thus it can be compensated by the digital redundancy of the pipeline ADC without any performance degradation. Therefore it is unnecessary to reduce the input signal magnitude when injecting the pseudo-random noise. Second, we do not need to face the complications of needing an accurate scaling (1/4 value) in the analog domain. (Even though we loosely use the scaled value approximately 1/4, a practical implementation would most likely need a value closer to 1/8 so that the digital redundancy would accommodate for actual circuit-level offsets as well as the pseudo-random signal.) Third, in the circuit implementation, the injection of pseudo-random noise can be done by randomly varying the comparator threshold level (comparator dithering), as long as this variation/dithering does not exceed the bounds of digital redundancy.

There is no change to the MDAC, which is the most critical and sensitive block in the pipeline ADC design. As a result, we can expect faster operation and less noise coupling from this radix detection scheme. Because of these advantages, this “comparator dithering” equivalent radix detection scheme is proposed as the most preferred digital background calibration scheme.

### 5.2.3 Interference Canceling in Calibration

Although the proposed correlation-based algorithm of Fig. 5.5 can be employed “as is” to extract the equivalent radix, the process would be slower and less accurate than the foreground algorithm in [57]. The main reason is the strong interference from the input signal to the ADC. Specifically, when we correlate the ADC's digital output with the pseudo-random sequence to detect the radix error, the input signal is transformed into noise at the same time (because the input signal is uncorrelated to the pseudo-random sequence). This input signal transformed noise will interfere with the radix error detection, making it less accurate or very slow.

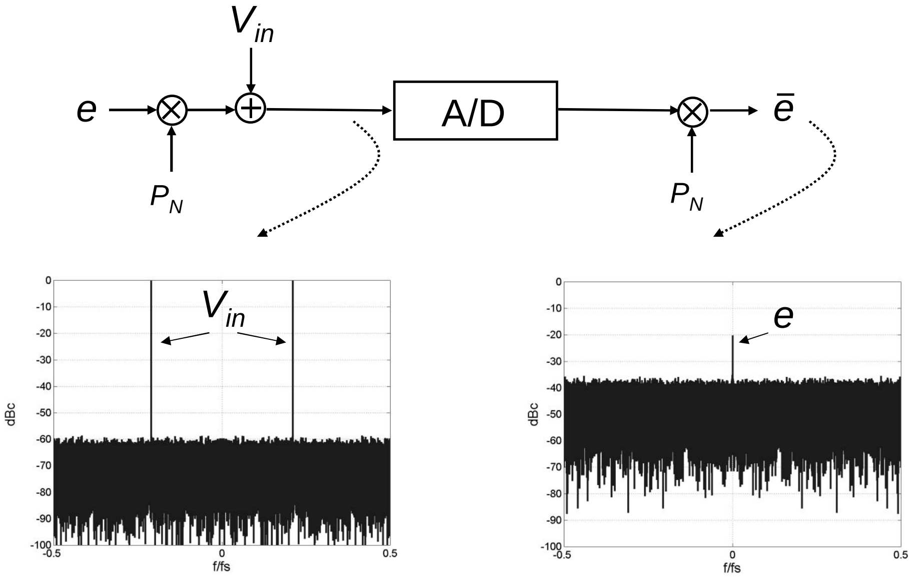


Fig. 5.6 Interference in the radix error detection

This situation is illustrated in Fig. 5.6. As shown in the figure, *e* is the small radix error that needs to extracted. It is first modulated by a pseudo-random sequence $P_N$ then added to the ADC’s input signal $V_{in}$. After A/D conversion, it is demodulated in the digital domain. This illustration directly represents Fig. 5.5 with Eqs. (5.11) and (5.12). Comparing the spectrum before and after demodulation, we can observe an obvious rise of the noise floor that is due to the input signal being transformed into noise. Although other noise sources such as thermal noise, 1/f noise, and back-end ADC’s quantization noise also affect the detection, the signal transformed noise is the dominant interference because the signal is usually strong (e.g. full scale input), and other noise sources are much smaller in comparison. This interference issue is common to all correlation-based calibration techniques. And it will limit the calibration accuracy. This problem is usually minimized by increasing the length of the pseudo-random sequence, but it is not very effective because the noise level only goes down by a 3dB for each doubling of the sequence length.

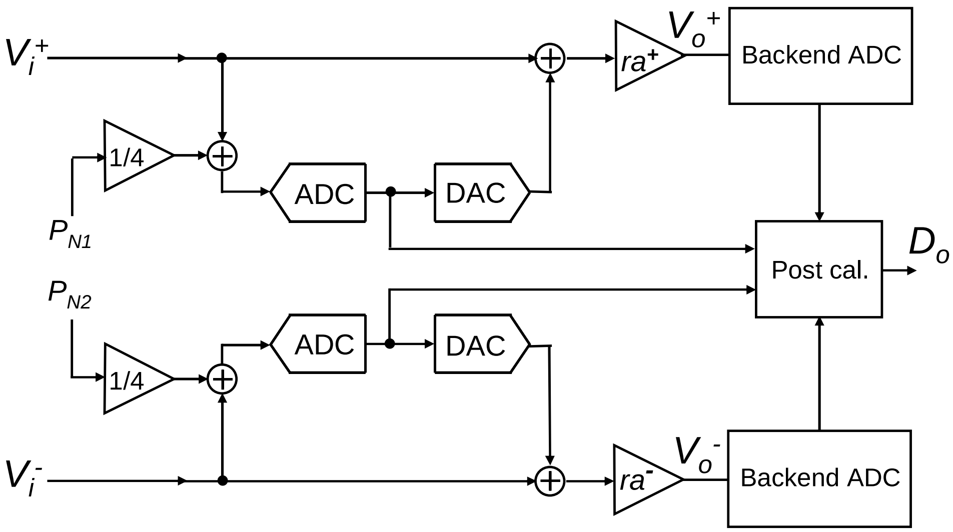


Fig. 5.7 Two-channel ADC architecture

To mitigate this problem, we propose a two-channel ADC architecture shown in Fig. 5.7. Instead of a single channel ADC, we put two identical ADCs in parallel to build a two-channel ADC. The two ADC channels are not time-interleaved. Instead, they take the same input signal but with opposite polarity. The final digital output of this two-channel ADC is given by:

$$D_o = D_o^+ - D_o^-, \tag{5.13}$$

where $D_o^+$ is the positive (*P*) channel ADC's digital output and $D_o^-$ is the negative (*N*) channel ADC's digital output. When we apply the proposed background radix extraction algorithm to this two-channel ADC, we can use two uncorrelated pseudo-random sequences to extract both radices in *P* channel and *N* channel at the same time. Despite this overhead, the ADC’s input signal can be cancelled by adding the two channels’ digital output together when we do the equivalent radix extraction using correlation. The computational efficiency of the proposed background radix extraction process can be improved dramatically because the majority of input signal component is removed in the correlation process.

If the two ADC channels are perfectly matched, the efficiency of this background radix extraction algorithm could be as good as a foreground algorithm that is performed in the absence of the input signal. In practical designs, channel mismatches such as offset mismatch and gain mismatch will limit performance. While these mismatches can severely degrade the SNDR in *time-interleaved* two-channel ADCs, here they only mildly reduce the efficiency of the background radix extraction algorithm because of the incomplete canceling of the ADC's input signal in the correlation process. No tones will be produced because the operations of these two ADC channels are parallel and synchronized, not time-interleaved. To speed up the radix detection algorithm even more, some channel mismatch calibration methods could also be employed with added circuitry. If necessary, in this proposed two-channel ADC architecture, the mismatches can also be calibrated out easily in the background as the two ADC channels are essentially processing the same signal but opposite polarity.

At first glance, it would seem that the two-channel ADC architecture will double the die area as well as power dissipation. But in reality, it just increases the die area taken up by active devices such as the opamps and comparators. The reason is similar to why fully-differential designs will not double the die size in comparison to a single-ended design given the same SNR requirement. In high accuracy switched-capacitor circuits, the die size is dominated by the total capacitor area. And the capacitor size is determined by the *kT/C* noise requirement. Given the same SNR requirement and the same input signal magnitude, we can reduce each ADC channel's capacitor size by half, so the total capacitor size of this proposed two-channel ADC is equal to the total capacitor size of a conventional single channel ADC. Although the resulting noise power would be four times higher in the proposed two-channel ADC, the equivalent input signal power is also four times higher since the equivalent input signal magnitude is doubled (we take both ADC channels’ digital output). Thus the SNR will not change. The main die size overhead is due to the fact that the number of comparators and opamps is doubled (even though smaller) in the proposed two-channel ADC architecture.

### 5.2.4 Overall Calibration Scheme

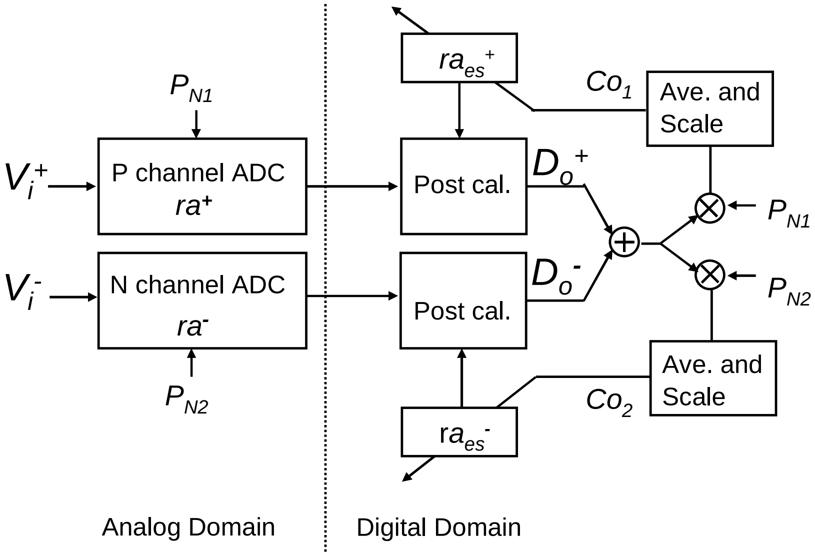


Fig. 5.8 Overall ADC with proposed calibration scheme

Figure 5.8 shows the overall ADC with proposed background calibration scheme using two uncorrelated pseudo-random sequences *P<sub>N1</sub>* and *P<sub>N2</sub>* to calibrate both channels simultaneously. To achieve a very robust operation, we use an iterative approach to extract the equivalent radix instead of using Eq. (5.10). First, we give an initial value to the estimated radix.

Then we start an iteration to approach the actual radix value. The radix update equations are:

$$\begin{aligned} ra_{es}^{+}[n+1] &= ra_{es}^{+}[n] - \Delta \cdot ((D_o^{+} + D_o^{-}) \otimes P_{N1}) \\ ra_{es}^{-}[n+1] &= ra_{es}^{-}[n] - \Delta \cdot ((D_o^{+} + D_o^{-}) \otimes P_{N2}) \end{aligned}, \tag{5.14}$$

where $ra_{es}^{+}$ and $ra_{es}^{-}$ are the estimated radices of $P$ and $N$ channel ADC stages, respectively; $P_{N1}$ and $P_{N2}$ are the two pseudo-random noise sequences used in radix extraction; $n$ is the iteration index; and $\Delta$ is the step size. After a certain number of steps, the estimated radix will converge to the actual equivalent radix. The main advantage of this iterative method is that the calibration of each stage is insensitive to the errors of its back-end ADC. That makes it very accurate, robust, and easy to implement.

When there are multiple stages that need to be calibrated, we can choose to use more pseudo-random noise sequences to extract the radices simultaneously. To reduce the number of pseudo-random noise sequences, we can share the same pseudo-random noise sequence among different stages and do the multi-stage ADC calibration by stepping through one stage at a time. Pushing this to the extreme, ultimately only one pseudo-random sequence is required, which can then be stepped through one block at a time (shared among channels as well as stages).

### 5.2.5 Simulation Results

Some behavioral simulations have been performed to verify the proposed calibration scheme. The prototype ADC in simulation was a 17-bit two-stage cyclic/algorithmic ADC as shown in Fig. 5.9. Note that the pseudo-random sequences are shared between two stages in this example to reduce the number of pseudo-random sequence generators. Gaussian distributed random capacitor mismatches of $\sigma = 0.1\%$

and 60dB opamp gain were assumed for this ADC. The two ADC channels were also given a fixed 1% $V_{ref}$ offset and 1% gain mismatches. In the equivalent radix extraction, the number of total samples was $2^{20}$, and the radix update step size Δ was $2^{-24}$. One typical output spectrum of the prototype ADC is shown in Fig. 5.10. In this case, the equivalent radices calculated by using Eq. (5.7) were 2.002 and 1.996 for the P channel, and 2.004 and 1.998 for the N channel. The values extracted by applying the proposed algorithm were 2.001990 and 1.996029 for the P channel, and 2.003979 and 1.998014 for the N channel. Figure 5.10(a) shows the output spectrum of the prototype ADC before calibration. The SNDR before calibration is 69dB. Figure 5.10(b) shows the simulation results of the same ADC after the proposed background calibration is applied. The SNDR is improved to 102dB.

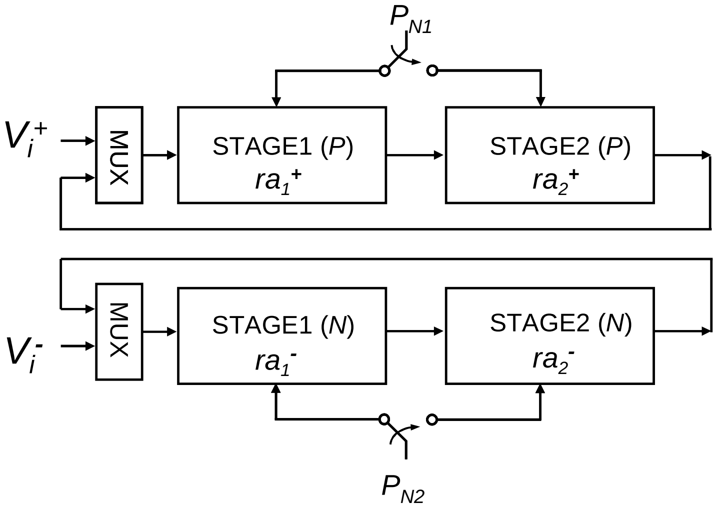
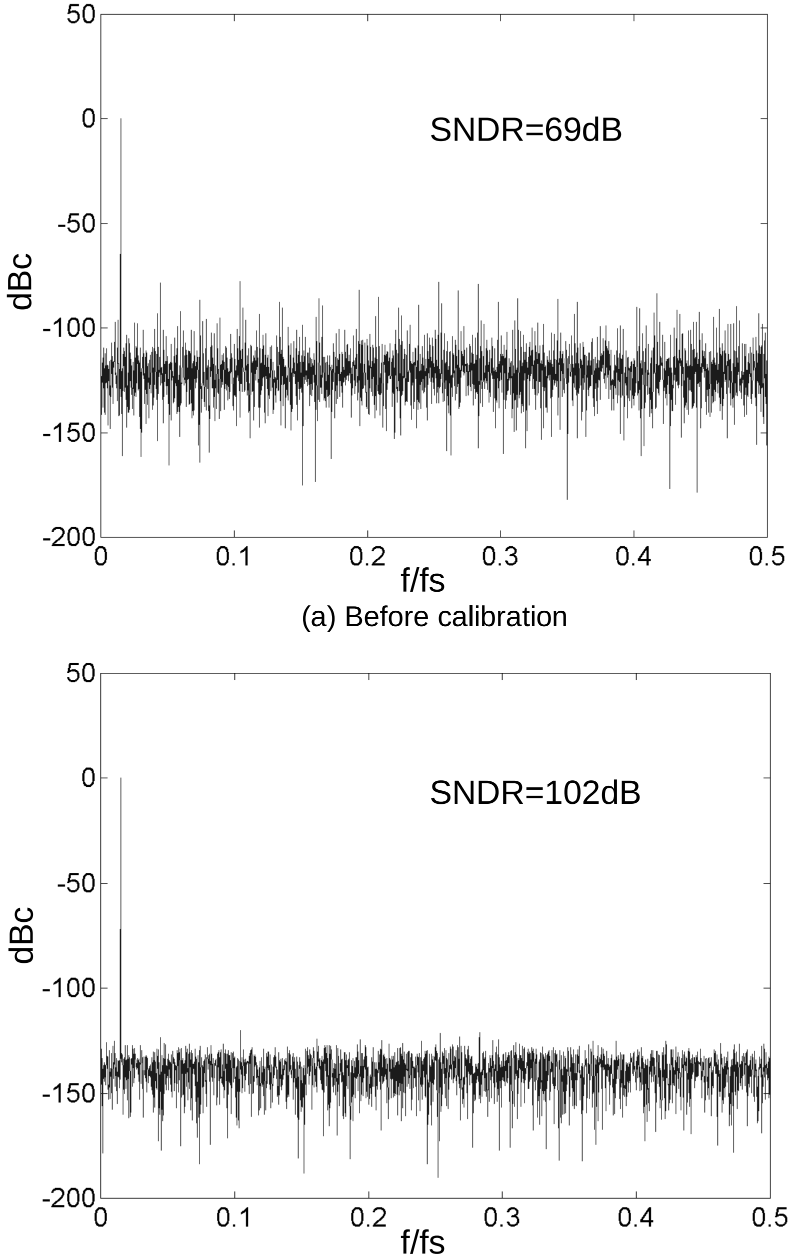


Fig. 5.9 Two stage cyclic ADC with calibration



(a) Before calibration

(b) After calibration

Fig. 5.10 Simulated output spectrum

### 5.3 Prototype Cyclic ADC Design

To verify the effectiveness of the proposed background calibration technique experimentally, a prototype two-channel cyclic ADC was designed in 0.18μm CMOS technology. Each channel includes two stages (1.5bit/stage). Consuming less than 20mW at 1V, this ADC employs the Opamp-Reset-Swiching-Technique (ORST) [59] and pseudo-differential architecture to solve the design issues from such ultra-low power supply voltage. The target clock frequency is 20MHz. The resolution can be programmed to change among 12bit, 14bit and 16bit corresponding to 6, 7 and 8 clock cycles to resolve an input sample. The details of main building blocks design will be presented in the following.

#### 5.3.1 Low-voltage Pseudo-differential MDAC Using ORST

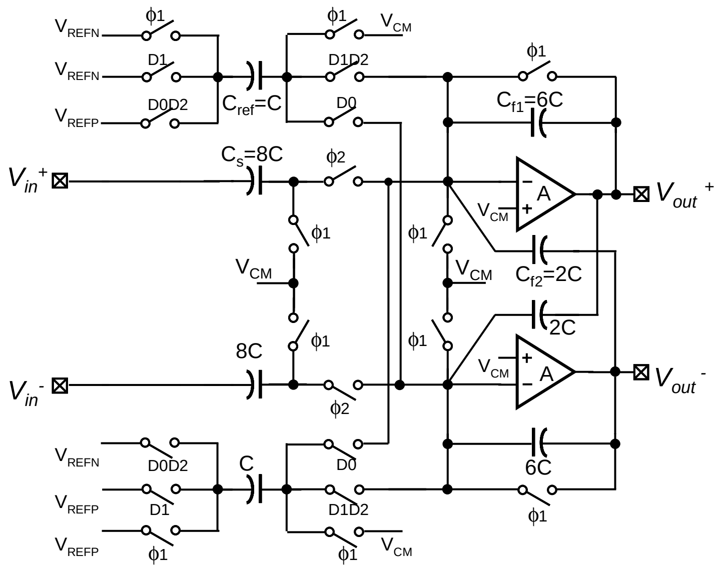


Fig. 5.11 Low-voltage pseudo-differential MDAC

The most critical design issue in ultra-low-voltage switched capacitor circuit design is that the clock over-drive voltage applied on the MOS switches scales down with the power supply voltage. As a result, the on-resistance of MOS switches increases dramatically. In the worst case, the CMOS switch may not be turned on if the over-drive voltage is less than the sum of the threshold voltage of NMOS transistor and PMOS transistor and the signal is close to half of the over-drive voltage. Several approaches have been proposed to overcome this limit [36][59][60][61][62]. In this prototype design, the Opamp-Reset-Switching-Technque (OSRT) [59][63][64] is chosen for its true low-voltage operation and higher speed potential.

Figure 5.11 shows the low-voltage MDAC based on ORST. The conventional floating CMOS switches in signal path are eliminated. The sampling capacitor is discharged by the resetting opamps in the previous stage. Here, one drawback is the "capacitor-flip-over" MDAC architecture can not be used. And the reference voltage has to be injected through a separate capacitor $C_{ref}$ which is scaled down to 1/8 of the sampling capacitor $C_s$ for lower extra noise and smaller speed penalty (due to the decrease of feedback factor). Note the reference voltages are scaled correspondingly ($V_{REFP}$ = 1V, $V_{REFN}$ = 0 for signal range from 250mV to 750mV). This also enables the use of NMOS or PMOS switches.  For the same purpose, the virtual ground voltage level is set to be 125mV.

Because it is very difficult to implement the common-mode-feedback (CMFB) circuit required in fully-differential circuit under ultra-low power supply voltage, a pseudo-differential MDAC architecture is used, as shown in Fig. 5.11. Here, the well-known common mode error amplification issue in pseudo-differential pipelined ADC is alleviated by adding extra cross-coupling positive feedback capacitors [65]. This makes the common mode gain of MDAC is one while keeping the differential gain still two. Therefore the common mode voltage error will not be amplified from stage to stage down the pipeline. However, this scheme does increase the opamp's settling time, which is mainly caused by the reduction of feedback factor in differential mode operation.

$$\beta^{DM} = \frac{C_{f1} - C_{f2}}{C_s + C_{f1} + C_{f2} + C_{ref} + C_{opamp}}. \tag{5.15}$$

where $C_{opamp}$ is the input parasitic capacitance. Ignoring $C_{opamp}$ and $C_{ref}$, the feedback factor calculated by Eq. (5.15) for the configuration shown in Fig. 5.11 is 1/4, which is about 25% smaller than the case of regular MDAC without splitting feedback capacitor which has the feedback factor of 1/3. When considering $C_{opamp}$ and $C_{ref}$ in practical design, this reduction is less significant. In addition to the feedback factor, this feedback capacitor splitting scheme also changes the equivalent load seen by the opamp in differential mode:

$$C_{L}^{DM} = \frac{(C_s + C_{ref} + C_{opamp})(C_{f1} + C_{f2}) + 4C_{f1} \cdot C_{f2}}{C_s + C_{f1} + C_{f2} + C_{ref} + C_{opamp}} + C_L \, . \tag{5.16}$$

where $C_L$ is the sampling capacitance of the next stage. For comparison, the feedback factor and equivalent load in common mode operation can be calculated as:

$$\beta^{DM} = \frac{C_{f1} + C_{f2}}{C_s + C_{f1} + C_{f2} + C_{ref} + C_{opamp}} \, . \tag{5.17}$$

$$C_{L}^{DM} = \frac{(C_s + C_{ref} + C_{opamp})(C_{f1} + C_{f2})}{C_s + C_{f1} + C_{f2} + C_{ref} + C_{opamp}} + C_L \, . \tag{5.18}$$

Another issue needs to be addressed in this MDAC design is that the opamp’s feedback factor during resetting phase is quite different from the feedback factor in the amplification phase if only one resetting switch (between opamp’s output and negative input) is used. This makes the opamp’s compensation very difficult. To solve this problem, an extra switch is put between opamp virtual ground and common mode voltage reference so the feedback factor is reduced during the resetting phase. However, large on-resistance is necessary for the resetting switches to avoid too much drop of opamp DC gain.

### 5.3.2 Low-voltage Opamp Design

Figure 5.12 shows the single-ended opamp design used in the MDAC. Two-stage architecture is chosen for large output swing and high DC gain under low power supply voltage. The first stage is a folded-cascode gain stage suitable for low power supply voltage. The PMOS input differential pair enables low input common mode level (125mV in this design), so NMOS switches can be used at the virtual ground to minimize the clock feed-through and parasitic capacitance. Smaller 1/f noise compared to NMOS input differential pair can be also expected. However, the thermal noise maybe higher since the NMOS transistors M3-M6 will have larger thermal noise current than the case of using PMOS transistors. The second stage is just a simple inverter formed by M11 and M12, so the output signal swing is maximized. The cascode compensation scheme instead of conventional miller compensation is used in this design to achieve faster settling with lower power consumption. Moreover, the size of compensation capacitor $C_C$ is also smaller by using cascode compensation [66][67]. As a result, the slew rate is also improved significantly. The simulated opamp parameters are summarized in Table 5.1.

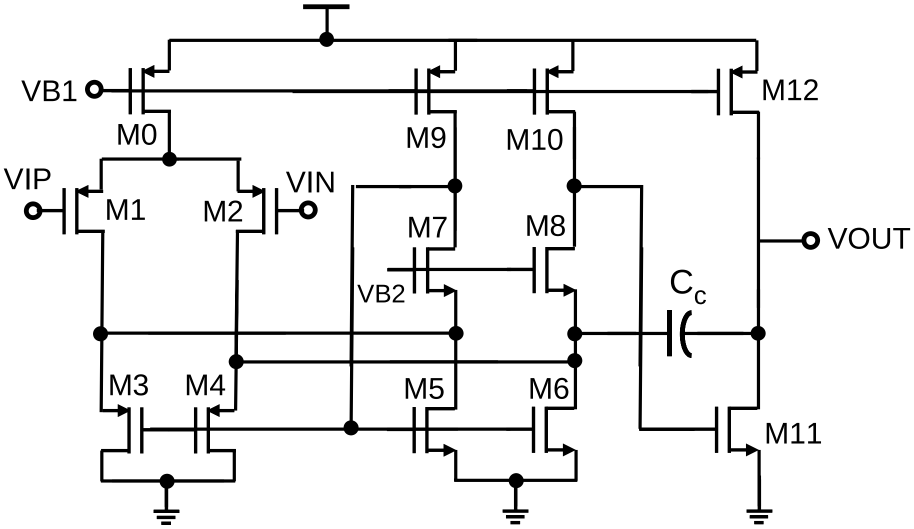


Fig. 5.12 Two-stage opamp with cascode compensation

| Technology | 0.18μm CMOS |
|---|---|
| DC gain | 66dB |
| Typical settling time in MDAC (14bit accuracy) | 13ns |
| Input Cap. | 0.2pF |
| Total Current | 1.5mA |
| Power supply voltage | 1V |
| Output swing | 0.5Vpp |

Table 5.1 Summary of opamp design

### 5.3.3 Input Sampling Circuit

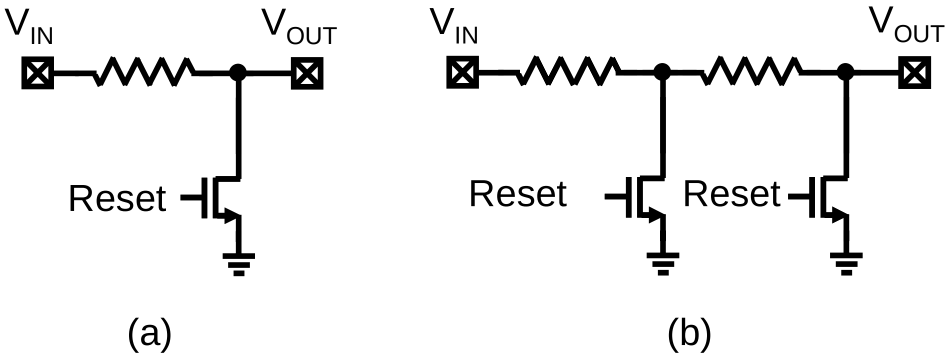


Fig. 5.13 Passive input sampling branch

Usually, a pipeline/cyclic ADC requires a front “Track-and-Reset” circuit to incorporate the Opamp-Reset-Switching-Technique or Switched-Opamp technique [68]. An active “Tack-and-Reset” circuit with very good linearity was proposed in previous work to meet this requirement [69]. For low input frequency application, a simple passive “Track-and-Reset” circuit shown in Fig. 5.13(a) can be used instead [70][71]. It has the advantages of low noise, ultra-low power consumption and very small die area. However, there is an input signal leakage during resetting phase since the input node is always connected to the sampling capacitor, and this leakage will cause distortion because of the nonlinear on-resistance of resetting switch. To reduce the input signal leakage and minimize the distortion, the sampling resistor usually has much large resistance than the resetting switch. As a result, the bandwidth of this “Track-and-Reset” circuit is limited. To alleviate this problem, a passive “Cascade-Track-and-Reset” circuit shown in Fig. 5.13(b) is employed in this design. It is very similar to the track and reset circuit described earlier, but two cascaded track-and-reset branches instead of one are used. This simple modification can dramatically reduce the input signal leakage and improve the linearity of track-and-reset stage with very small overhead. For example, if the each input resistor is 5kΩ and the on-resistance for each reset switch is about 100Ω, the "Cascade-Track-and-Reset" can improve the linearity by about 30dB.

### 5.3.4 Low-voltage Comparator with Dither

Figure 5.14 shows the comparator module used in this design. It has several signal paths to inject input signal, threshold voltage, dither signal and common mode voltage adjustment signal. The sizes of capacitors are scaled to reduce the spread ratio and avoid large amplitude drop of input signal due to the voltage division. Note there is no floating switch in signal paths to enable ultra-low voltage operation. Two ploy resistors are inserted at the differential input path to reduce the kick-back noise seen by the MDAC input.

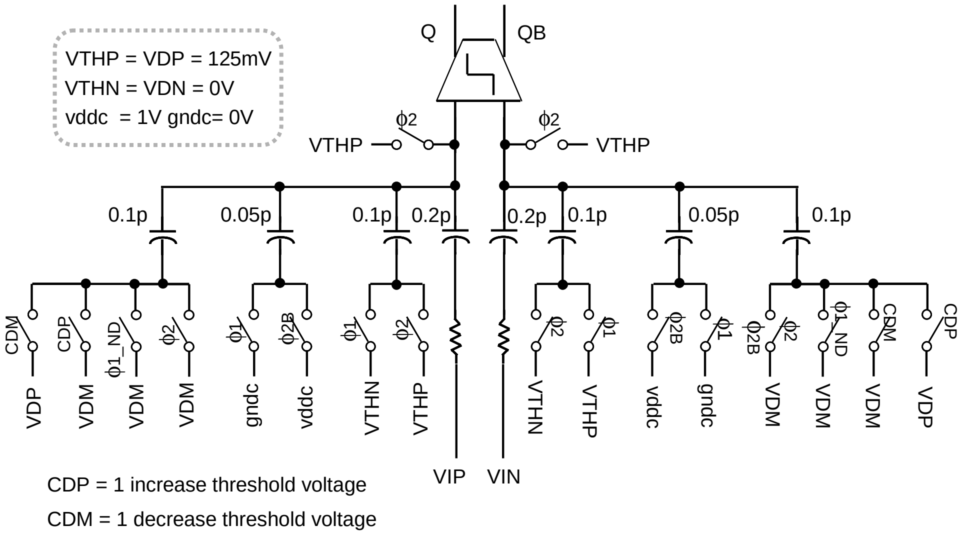


Fig. 5.14 Low-voltage comparator module with dither

The low voltage latched comparator design is shown in Fig. 5.15. Simple static comparator scheme is adopted. The static power consumption is 50uW at 1V power supply voltage. The combination of PMOS input differential pair and NMOS latch has the advantages of smaller 1/f noise as well as offset and faster latching. The typical latching time is about 800ps, which is enough for 20MHz ADC. Two dummy PMOS transistors M1 and M2 are added to reduce the differential kick-back noise and they are half size of the input pair.

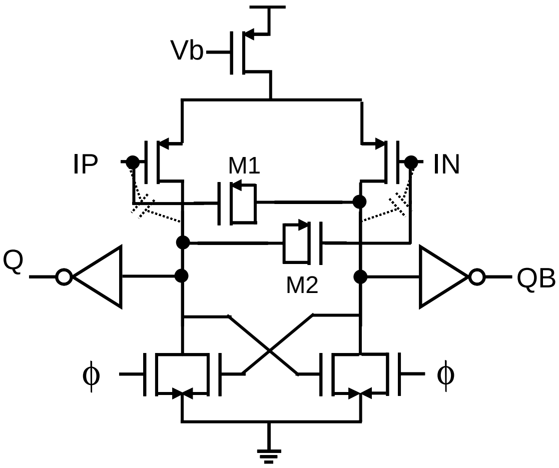


Fig. 5.15 Simple static comparator

### 5.3.5 Pseudo-random Sequence Generator

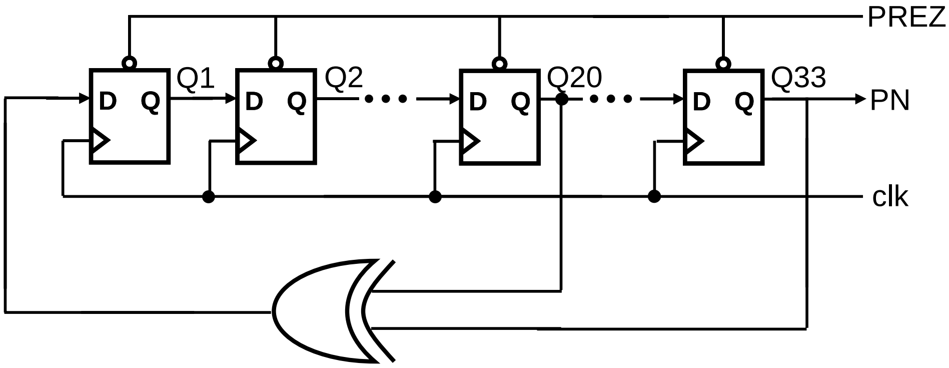


Fig. 5.16 Pseudo-random sequence generator

The on-chip pseudo-random sequence generator is shown in Fig. 5.16. It consists of 33 D flip-flops and one XOR gate. The output sequence will not repeat itself until $2^{33}$ clock cycles. Note the clock frequency of this block is the same as the sampling frequency which is the ADC's clock frequency divided by the number of cycles to process each input sample. The minimum allowed sizes are chosen for these D flip-flops and XOR gate to minimize the power consumption and digital noise.

### 5.4 Experiments Results

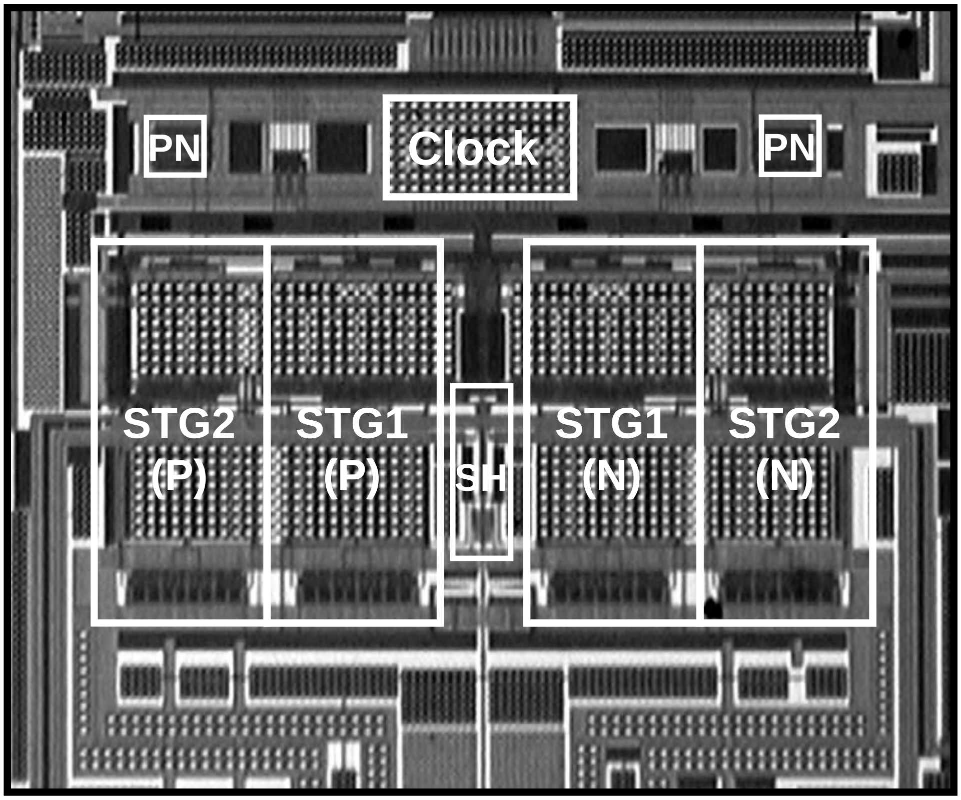


Fig. 5.17 Die photo of the prototype cyclic ADC

The prototype ADC was fabricated in a 0.18μm CMOS process. The die photograph is shown in Fig. 5.17, where the active die area is 1.6mm×1.1mm. The total power consumption is 12mW at 0.9-V supply and 2MHz sampling frequency (12MHz clock frequency).

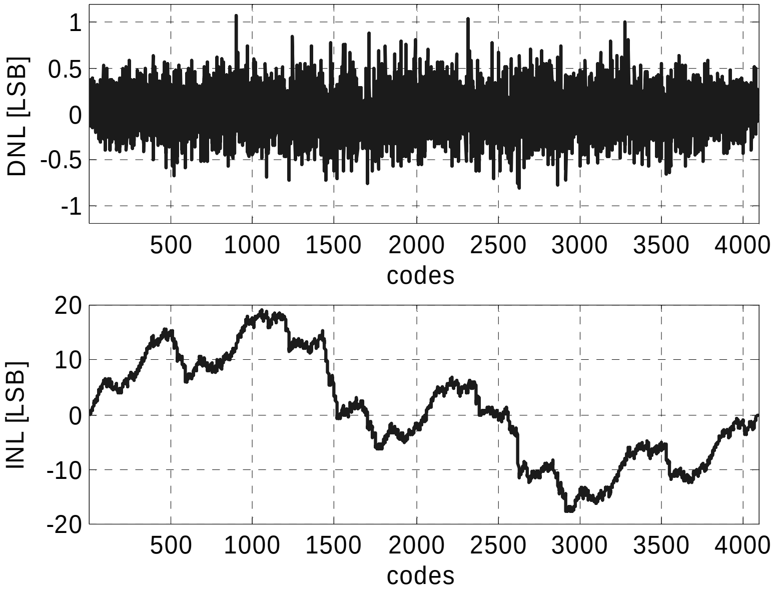


Fig. 5.18 DNL and INL before calibration

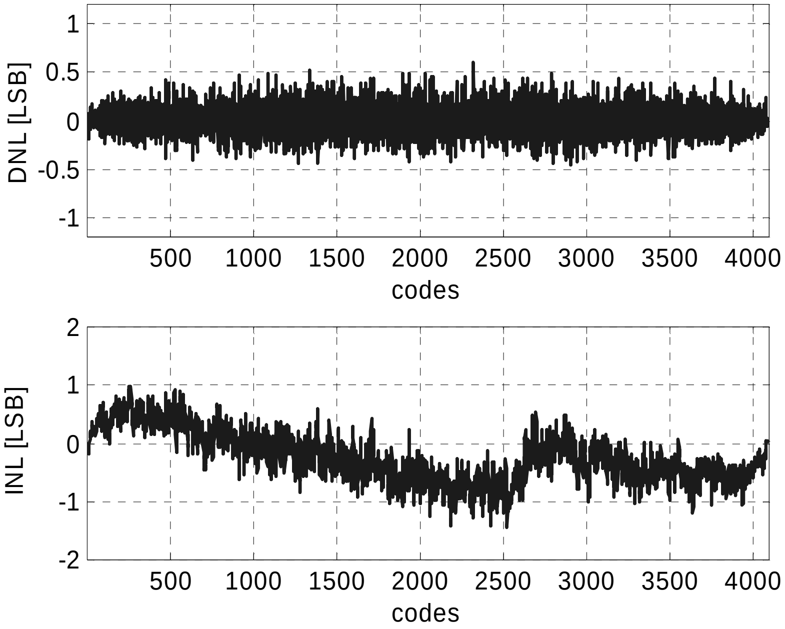


Fig. 5.19 DNL and INL after calibration

Before calibration, the measured DNL and INL are 1.1LSB and 19LSB at 12-bit level as shown in Fig. 5.18. After calibration, the DNL and INL are reduced to 0.6LSB and 1.4LSB respectively as shown in Fig. 5.19. Figure 5.20 shows a typical measured frequency spectrum at 80kHz input and 2MSPS (the clock frequency is 12MHz). Before calibration, the measured SFDR, SNR, and SNDR are 48dB, 51dB, and 46dB respectively. After calibration, the SFDR, SNR and SNDR are improved to 81dB, 55dB, and 55dB respectively.

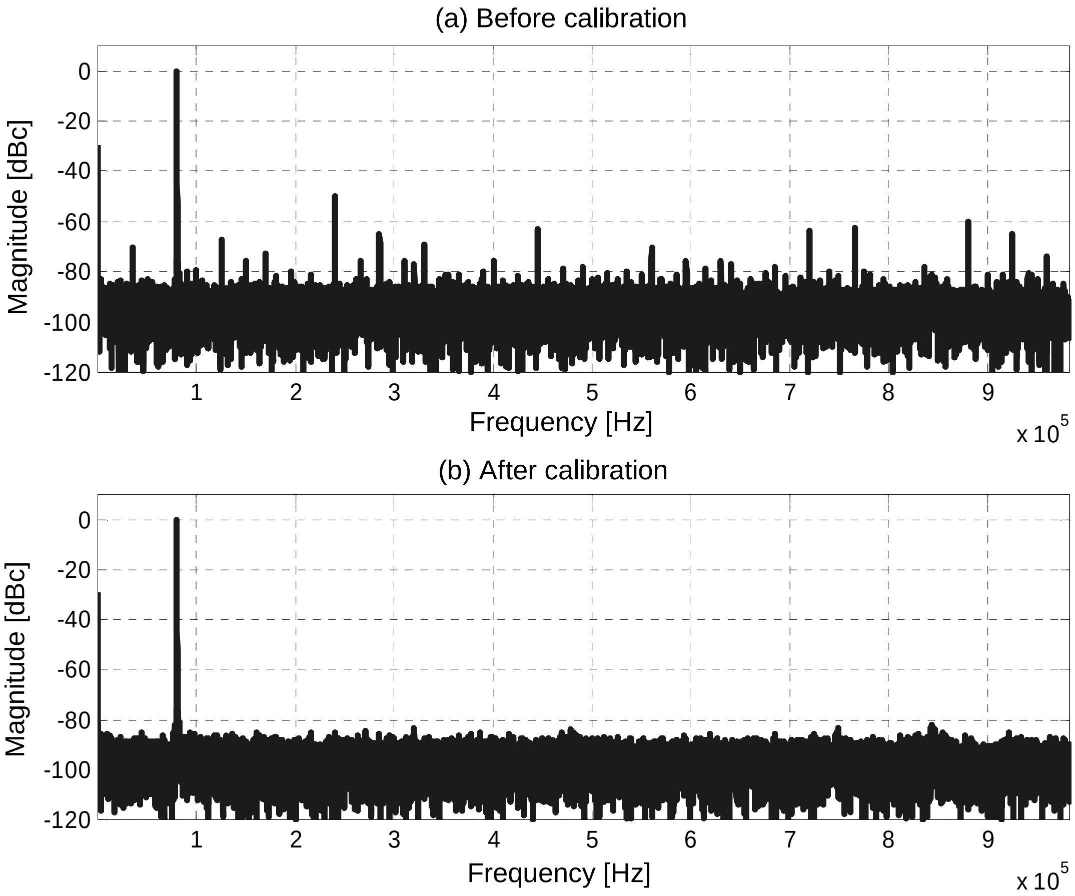


Fig. 5.20 Measured ADC output spectrum at 2MSPS

Figure 5.21 shows the dynamic performance versus conversion rate (clock frequency is 6× conversion rate). The SFDR remains exceeding 80dB up to 3.2MSPS and degrades to 77dB at 5MSPS. After 5MSPS, the SFDR drops quickly mainly due to settling time limit. Figure 5.22 shows the dynamic performance versus power supply voltage. From 0.9V to 1.3 V power supply voltage, the dynamic performance is almost constant and slightly better at higher supply voltage because of larger headroom and signal swing. Figure 5.23 shows the dynamic performance versus input amplitude. The performance is consistent up to -0.5dB of the target maximum input amplitude. The measurement results are summarized in Table 5.2.

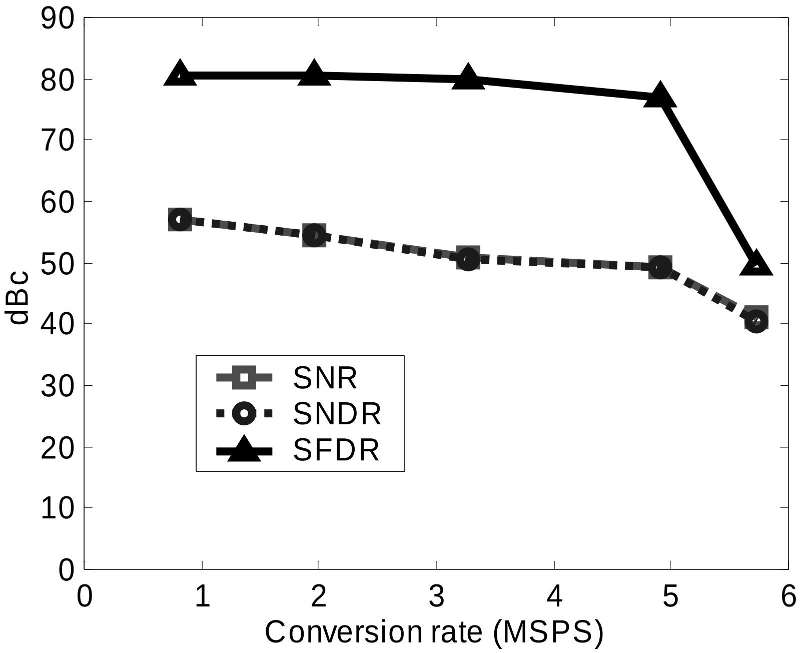


Fig. 5.21 Dynamic performance vs. conversion rate

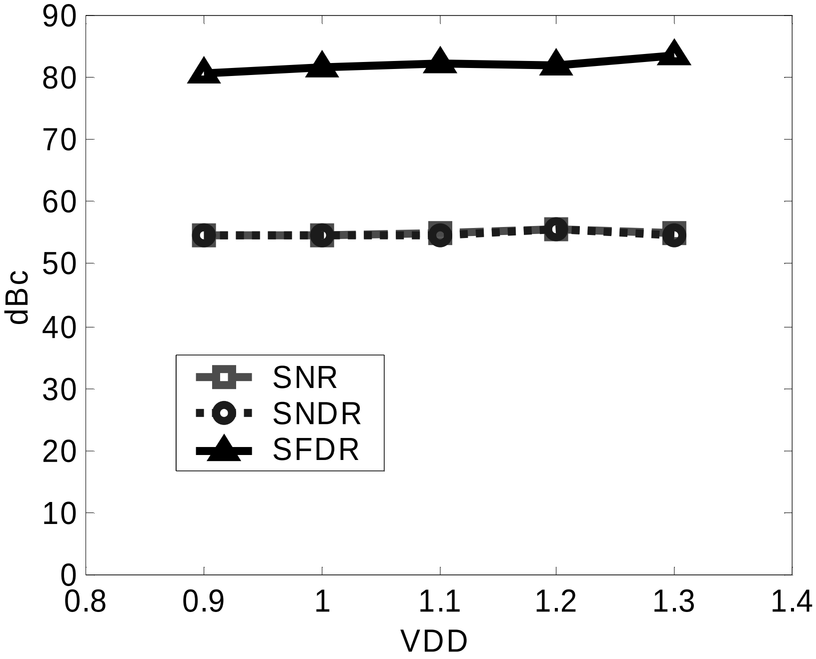


Fig. 5.22 Dynamic performance vs. power supply voltage

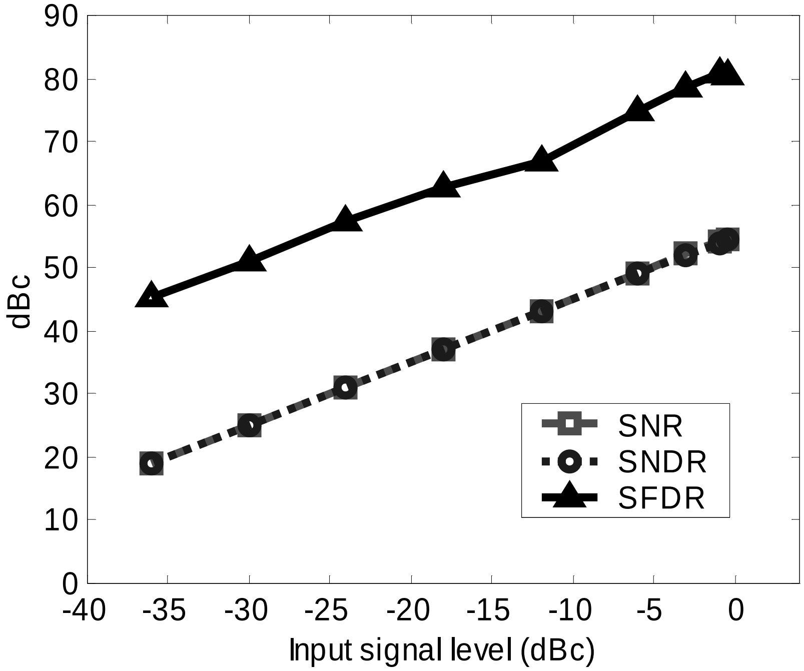


Fig. 5.23 Dynamic performance vs. input signal level

| | |
|---|---|
| Resolution | 12-bit |
| Sampling Rate | 2MSPS |
| Technology | 0.18μm CMOS |
| Active Die Area | 1.6mm×1.1mm |
| Supply Voltage | 0.9V |
| Power Consumption | 12mW |
| DNL/INL | 1.1LSB/19LSB (before cal.)<br>0.6LSB/1.4LSB (after cal.) |
| SNR/SNDR/SFDR | 50.5dB/45.8dB/47.7dB (before cal.)<br>54.6dB/54.5dB/80.6dB (after .cal) |

Table 5.2 Performance summary of the prototype cyclic ADC

### 5.5 Summary

A background digital self-calibration technique for multi-stage pipelined or cyclic/algorithmic ADCs has been described. This technique can correct the errors resulting from capacitor mismatches and finite opamp gain. An accurate calibration is achieved by re-calculating the digital output based on each stage's equivalent radix. These equivalent radices are extracted in the background by applying a correlation process using pseudo-random noise sequence. The pseudo-random noise sequence is injected at the input of the sub-ADC such that the input signal is unaltered by the added noise. The inherent digital redundancy of the pipelined ADC architecture compensates for the added pseudo-random noise without reducing the input signal range and without degrading the SNR of the ADC system. The efficiency of the correlation algorithm can be improved significantly by using a two-channel ADC architecture, where total capacitance and power consumption remain unchanged in comparison to a single-channel ADC architecture. Simulation results indicate that a significant SNDR improvement can be achieved by using the proposed calibration technique. Although the 1.5-bit-per-stage architecture is chosen to demonstrate the operation of the proposed calibration technique, the proposed technique can be generally applied to other multi-bit-per-stage architectures. In the case of a multi-bit-per-stage architecture, a binary-weighted DAC (instead of thermometer DAC) should be used to reduce the number of variables in calibration.

# 6 CONCLUSIONS

Two accuracy improvement techniques to overcome the accuracy limit set by analog building blocks (opamps and caps) are presented. One is the time-shifted CDS technique which addresses the finite opamp gain effect and the other is the radix-based background digital calibration technique which can take care of both finite opamp gain and capacitor mismatch. These methods are simple, effective, easy to implement and power efficient. And they are compatible with the state of the art low-voltage high-speed pipelined ADC design techniques using deep sub-micron CMOS technology. The effectiveness of the proposed techniques is demonstrated in simulation as well as in experiment.

# BIBLIOGRAPHY

[1] K. Kim, N. Kusayanagi, and A. Abidi, “A 10-b 100MS/s CMOS A/D Converter,” *IEEE Custom Int. Circuits Conf.*, pp. 419-422, May 1996.

[2] D. G. Nairn, “A 10-b, 3V, 100MS/s pipelined ADC,” *IEEE Custom Int. Circuits Conf.*, pp. 257-260, May 2000.

[3] Y. Park, S. Karthikeyan, F. Tsay, and E. Bartolome, “A 10-b 100MS/s CMOS Pipelined ADC with 1.8V power supply,” *ISSCC Digest of Tech. Papers*, pp. 130-131, Feb. 2001.

[4] S. Yoo, T. Oh, J. Moon, S. Lee, and U. Moon, “A 2.5V 120MSample/s CMOS Pipelined ADC with high SFDR,” *Proc. IEEE Custom Integrated Circuit Conf.*, vol. 23, pp. 441-444, May 2002.

[5] S. Yoo *et al.*, “A 10b 150MS/s 123mW 0.18μm Pipelined ADC,” *ISSCC Dig. Tech. Papers*, pp. 326-327, Feb. 2003.

[6] B. Min, et al., “A 69mW 10b 80MS/s Pipelined CMOS ADC,” *ISSCC Digest of Technical Papers*, pp. 324-325, Feb. 2003

[7] W. Yang, D. Kelly, I. Mehr, M. Sayuk, and L.signer, “A 3-V 340mW 14-b 75Msample/s CMOS ADC with 85-dB SFDR at Nyquist Input,” *IEEE J. Solid-State Circuits*, vol. 36, pp. 1931-1936, Dec. 2001.

[8] P. Li, M. Chin, and P. Gray, “A ratio independent algorithmic analog-to-digital conversion technique,” *IEEE J. Solid-State Circuits*, vol. sc-19, pp. 828-836, Dec. 1984.

[9] C. Shih and P. Gray, “Reference refreshing cyclic analog-to digital and digital-to-analog converters,” *IEEE J. Solid-State Circuits*, vol. sc-21, pp. 544-554, Aug. 1986.

[10] B. Song, M. Tompsett, and K. Lakshmikumar, “A 12-b 1-Msample/s capacitor error-averaging pipelined A/D converter,” *IEEE J. Solid-State Circuits*, vol. 23, pp. 1324-1333, Dec. 1988.

[11] Y. Chiu, “Inherently linear capacitor error-averaging technique for pipelined A/D converter,” *IEEE Trans. Circuits Syst. II*, vol. 47, pp. 229-232, Mar. 2000.

[12] H. Lee, D. Hodges, and P. Gray, “A self-calibrating 15-b CMOS A/D converter,” *IEEE J. Solid-State Circuits*, vol. SC-19, pp. 813-819, Dec. 1984.

[13] Y. Lin, B. Kim, and P. Gray, "A 13-b 2.5-MHz self-calibrated pipelined A/D converter in 3u CMOS," *IEEE J. Solid-State Circuits*, vol. SC-26, pp. 628-636, Apr. 1991.

[14] S. Lee and B. Song, "Digital-domain calibration of multi-step analog-to-digital converter," *IEEE J. Solid-State Circuits*, vol. 27, pp. 1679-1688, Dec. 1992.

[15] A. Karanicolas and H. Lee, "A 15-b 1-Msample/s digitally self-calibrated pipeline ADC," *IEEE J. Solid-State Circuits*, vol. 28, pp. 1207-1215, Dec. 1993.

[16] S. Lee and B. Song, "Interstage gain proration technique for digital-domain multi-step ADC calibration," *IEEE Trans. Circuits Syst. II*, vol. 41, pp. 12-18, Jan. 1994.

[17] H. Lee, "A 12-b 600ks/s digitally self-calibrated pipelined algorithmic ADC," *IEEE J. Solid-State Circuits*, vol. 29, pp. 509-515, Apr. 1994.

[18] T. Shu, B. Song, and K. Bacrania, "A 13-b, 10-Msample/s ADC digitally calibrated with oversampling delta-sigma converter," *IEEE J. Solid-State Circuits*, vol. 32, pp. 1866-1875, Dec. 1997.

[19] J. Ingino and B. Wooley, "A continuously calibrated 12-b 10-Ms/s 3.3-V A/D converter," *IEEE J. Solid-State Circuits*, vol. 33, pp. 1920-1931, Dec. 1998.

[20] U. Moon and B. Song, "Background digital calibration techniques for pipelined ADCs," *IEEE Trans. Circuits Syst. II*, vol.44, pp.102-109, Feb. 1997.

[21] S. Kwak, B. Song, and K. Bacrania, "A 15-b, 5-Msample/s low-spurious CMOS ADC," *IEEE J. Solid-State Circuits*, vol. 32, pp. 1866-1875, Dec. 1997.

[22] O. Erdogan, P. Hurst, and S. Lewis, "A 12-b digital-background-calibrated algorithmic ADC with -90-dB THD," *IEEE J. Solid-State Circuits*, vol. 34, pp. 1812-1820, Dec. 1999.

[23] R. Jewett, K. Poulton, K. Hsieh, and J. Doernberg, "A 12-b 128Msamples/s ADC with 0.05LSB DNL," *Proc Int. Solid-State Circuits Conf.*, pp. 138-139, Feb. 1997.

[24] J. Ming and S. Lewis, "An 8-bit 80-Msample/s Pipelined Analog-to-Digital Converter With Background Calibration," *IEEE J. Solid-Sate Circuits*, vol. 36, pp. 1489-1497, Oct. 2001.

[25] I. Galton, "Digital Cancelation of D/A Converter Noise in Pipelined A/D Converters," *IEEE Trans. Circuits Syst. II*, Vol.47, pp. 185-196, Mar. 2000.

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

[27] B. Murmann and B. Boser, “A 12-b 75MS/s pipelined ADC using open-Loop residue amplifier,” *ISSCC Digest of Tech. papers*, pp. 330-331, Feb. 2003.

[28] J. Li and U. Moon, “High-speed pipelined ADC using time-shifted CDS technique,” *IEEE Int. Symp. Circuits Syst.*, vol. I, pp. 90-912, May 2002.

[29] J. Li and U. Moon, “A 1.8-V 67mW 10-bit 100MSPS pipelined ADC using time-shifted CDS technique,” *Proc. IEEE Custom Integrated Circuit Conf.*, pp. 413-416, Sep. 2003.

[30] J. Li and U. Moon, “Background calibration technique for multi-stage pipelined ADCs with digital redundancy,” To appear in *IEEE Trans. Circuits Syst. II*, Sep. 2003.

[31] S. Lewis, “Video-rate analog-to-digital conversion using pipelined architectures,” Ph.D. dissertation, U.C. Berkeley, 1987.

[32] Mehr and L. Singer, “A 55-mW 10-bit, 40-Msample/s Nyquist-rate CMOS ADC,” *IEEE J. Solid-Sate Circuits*, vol. 35, pp. 318-325, Mar. 2000.

[33] K. Bult *et al.*, “A fast-settling CMOS opamp for SC circuits with 90dB DC gain,” *IEEE J. Solid-Sate Circuits*, Vol. 25, No. 12, Dec. 1990.

[34] M. Das, “Improved design criteria of gain-boosted CMOS OTA with high-speed optimizations,” *IEEE Tans. Circuits Syst. II*, vol. 49, No.3, pp. 204-207, Mar. 2002.

[35] T. Cho and P. Gray, “A 10-bit, 20-Msample/s 35mW pipelined A/D converter,” *IEEE J. Solid-Sate Circuits*, vol. 30, pp. 166-172, Mar. 1995.

[36] A. Abo and P. Gray, “A 1.5-V 10-bit 14.3MS/s CMOS pipeline ADC,” *IEEE J. Solid-Sate Circuits*, Vol. 34, pp. 599-606, May. 1999.

[37] M. Dessouky and A. Kaiser, “Very low-voltage digital-audio ΔΣ modulator with 88-dB dynamic range using local switch bootstrapping,” *IEEE J. Solid-Sate Circuits*, Vol. 36, pp. 349-355, Mar. 2001.

[38] L. Singer *et al.* “A 12b 65MS/s CMOS ADC with 82dB SFDR at 120MHz,” *ISSCC Digest of Technical Papers*, pp. 38-39, Feb. 2000.

[39] S. Lewis, “Optimizing the Stage resolution in pipelined, multistage, analog-to-Digital converters for video-rate applications,” *IEEE Tans. Circuits Syst. II*, vol. 39, no.8, pp. 516-523, Aug. 1992.

[40] D. Cline, “Noise, speed, and power trade-offs in pipelined analog to digital converters,” Ph.D. dissertation, U. C. Berkeley, 1995.

[41] P. Yu and H. Lee, “2.5-V, 12-b, 5MS/s pipelined CMOS ADC,” *IEEE J. Solid-Sate Circuits*, Vol. 31, No. 12, pp. 1854-1861, Dec. 1996.

[42] K. Nagaraj *et al.*, “A 250-mW, 8-b, 52Msamples/s parallel pipelined A/D converter with reduced number of amplifiers,” *IEEE J. Solid-Sate Circuits*, Vol. 32, No. 3, pp. 312-320, Mar. 1997.

[43] S. Kulhalli, V Penkota, and R. Asv, “A 30mW 12b 21MS/s Pipelined CMOS ADC,” *ISSCC Digest of Technical Papers*, pp. 312-313, Feb. 2002

[44] W. Black and D. Hodges, “Time Interleaved Converter Arrays,” *IEEE J. Solid-Sate Circuits*, Vol. 36, pp. 1022-1029, Dec. 1980.

[45] C. Conroy *et al.*, “An 8-b 85-MS/s parallel Pipeline A/D converter in 1μm CMOS,” *IEEE J. Solid-Sate Circuits*, pp. 447-454, Apr. 1993.

[46] L. Sumanen, M Waltari, and K. Halonen, “A 10-b 200-MS/s CMOS parallel pipeline A/D converter,” *IEEE J. Solid-Sate Circuits*, Vol. 36, pp. 1048-1055, Jul. 2001.

[47] H. Lee, D. Hoges, and P.Gray, “A self-calibrating 15bit CMOS A/D converter,” *IEEE J. Solid-Sate Circuits*, Vol. SC-19, pp.813-819, Dec, 1984.

[48] C. Enz and G. Temes, “Circuit techniques for reducing the effects of opamp imperfections: audozeroing, correlated double sampling and chopper stabalization,” *Proc. Of IEEE,* pp.1584-1614, Nov. 1996.

[49] G. Temes and K. Haug, “Improved offset compensation schemes for switched capacitor circuits,” *Electro. Lett.*, Vol. 20, pp. 508-509, 1984.

[50] K. Nagaraj *et al.*, “Switched-Capacitor circuits with reduced sensitivity to finite amplifier gain,” *Proc. IEEE Int. Symp. Cir. Sys.*, pp.618-621, 1986

[51] L. Larson, K. Martin, and G. Temes, “GaAs switched-capactior circuits for high-speed signal processing,” *IEEE J. Solid State Circ.*, Vol. 22, pp. 971-981, 1987.

[52] H. Yoshizawa, Y. Huang, and G. Temes, “Improved SC amplifiers with low sensitivity to op-amp imperfections,” *Electro. Lett.*, Vol. 34, pp. 508-509, 1998.

[53] Y. Huang, P. Ferguson, G. Temes, “Reduced nonlinear distortion in circuits with correlated double sampling,” *Proc. IEEE Int. Symp. Cir. Sys.*, pp.159-162, 1996.

[54] A. Ali and K. Nagaraj, “Correction of operational amplifier gain error in pipelined A/D Converters,” *IEEE Int. Symp. Circuits Syst.*, pp. 1.568-1.571, May 2001.

[55] S. Lewis *et al.*, “A 10-b 20MSamples/s analog to digital converter,” *IEEE J. Solid-State Circuits*, Vol. 27, No. 3, pp. 351-358, Mar. 1992.

[56] M. Waltari and K. Halonen, “Timing skew insensitive switching for double sampled circuits,” *IEEE Int. Symp. Circuits Syst.*, vol. II, pp. 61-64, May 1999.

[57] D. Chang and U. Moon, “Radix-based digital calibration technique for multi-stage ADC,” *IEEE Int. Symp. Circuits Syst*., vol. II, pp. 796-799, May. 2002.

[58] J. Li and U. Moon, “An extended radix-based digital calibration technique for multi-Stage ADC,” *IEEE Int. Symp. Circuits Syst*., May. 2003.

[59] E. Bidari *et al*., “Low-voltage switched capacitor circuits,” *IEEE Int. Symp. Circuits Syst*., pp. 49-52, May. 1999.

[60] Y. Atachi, A. Ishinawa, A. Barlow, and K. Taksuka, “A 1.4-V switched capacitor filter,” *Proc. IEEE Custom Integrated Circuit Conf.*, pp. 821-824, May 1990.

[61] Y. Nakagome *et al.*, “Experimental 1.5-V 64-Mb DRAM,” *IEEE J. Solid-Sate Circuits*, Vol. 26, pp. 465-472, Apr. 1991.

[62] J. Crols and M. Steyaert, “Switched opamp: an approach to realize full ery low-voltage digital-audio ΔΣ modulator with 88-dB dynamic range using local switch bootstrapping,” *IEEE J. Solid-Sate Circuits,* Vol. 36, pp. 349-355, Mar. 2001.

[63] M. Keskin, U. Moon, and G Temes, “A 1-V 10MHz clock rate 13-bit CMOS ΔΣ modulator using unit-gain-reset opamps,” *IEEE J. Solid-Sate Circuits*, Vol. 37, pp. 817-823, July. 2002.

[64] D. Chang and U. Moon, “A 1.4-V 10-bit 25-MS/s pipelined ADC using opamp-reset switching technique,” *IEEE J. Solid-Sate Circuits*, Vol. 38, pp. 1401-1404, July. 2003.

[65] J. Wu, Y. Chang, and K. Chang, “1.2-V CMOS switched-capacitor circuits,” *ISSCC Digest of Technical Papers*, pp. 388-389, Feb. 1996.

[66] B. Ahuja, “An improved frequency compensation technique for CMOS operational amplifier,” *IEEE J. Solid-Sate Circuits*, Vol. SC-18, No.6, pp.629-633, Dec. 1983.

[67] D. Ribner and M. Copeland, “Design techniques for cascaded CMOS Opamps with improved PSRR and common mode input range,” *IEEE J. Solid-Sate Circuits*, Vol. SC-19, No.6, pp.919-925, Dec. 1984.

[68] A. Baschirotto, R. Castello, and G. Montagna, “Active series switch for switched-opamp circuits,” *Electron. Lett.*, Vol. 34, No. 14, pp. 1365-1366, July. 1998.

[69] D. Chang and U. Moon, “A 1V input smapling circuit with 100dB SFDR,” *Electron. Lett.*, Vol. 34, No. 14, pp. 1365-1366, Jul. 2001.

[70] G. Ahn, personal communication, 2002.

[71] M. Keskin, “A novel low-voltage switched-capacitor input branch,” *IEEE J. Solid-Sate Circuits*, *IEEE Trans. Circuits Syst. II*, Vol. 50, No.6, pp. 315-317, Jul. 2003.