Generalized Radix Design Techniques for Low-Power, Low-Voltage Pipelined and Cyclic Analog-Digital Converters.

by

Vivek Sharma

A THESIS

submitted to

Oregon State University

in partial fulfillment of
the requirement for the
degree of

Master of Science

Completed June 1, 2004

Commencement June 2005

ACKNOWLEDGEMENTS

Firstly, I would like to thank my advisor Dr. Gabor C. Temes for granting me the privilege of working under his guidance, and for his patience during the course of my work. Not only have I gained from his vast technical experience, but also he has helped hone my presentation and writing skills with constructive criticism and encouragement. Additionally, his down-to-earth manner of dealing with his students sets a truly wonderful example.

I would like to thank my co-major professor Dr. Un-Ku Moon for egging me on with critical suggestions and comments, and pushing me harder. Without his guidance, a substantial chunk of this work would never have materialised.

I also want to thank Prof. Huaping Liu, Roger Traylor and Prof. William M. Hetherington for serving on my committee.

Special thanks are due to Thirumalai Rengachari for his collaboration on the cytosensor project, and for his friendship and all our discussions.

Thanks to José Silva for patiently answering various doubts and offering solutions to a variety of technical problems that kept cropping up. Also, thanks for allowing us to put our chip alongside his.

I would like to thank Pavan Hanumolu, Robert Batten, Volodymyr Kratyuk, Zhimin Li, and Gil-Cho Ahn for answering various questions and offering useful tips.

Thanks also to Gowtham Vemulapalli, Anantha Nag Nemmani, Husni Habal, Ruopeng Wang, Yoshio Nishida, Yuhua Guo and Sirisha Adluri for all the discussions and help.

I also extend thanks to all my colleagues and ex-colleagues in Dearborn 211 and Dearborn 212 for all the help, guidance and camraderie.

Thanks to all the ECE office staff, especially Ferne Simendinger for helping with the reams of paperwork.

Thanks to Catalyst Foundation for funding my research activity.

Special thanks to Patrick Birrer and Stefan Meier for helping arrange my summer internship at HTA-Burgdorf, Switzerland, and many thanks for Dr. Franz Bachmann and Jean-Pierre Steger of HTA for their cheerful guidance and valuable advice. Thanks also to all my colleagues at HTA.

I would like to acknowledge the guidance of Dr. Baquer Mazhari and Dr. Utpal Das at IIT Kanpur, and Kallol Chatterjee at ST Microelectronics-India. Their guidance and advice helped me get started in the field of analog circuit design.

Special thanks to my friends Vinay Ramyead, Anurag Tiwari and Ragini Murugan for their cheerful friendship and all of our good times. Also, thanks to the entire Indian community for making life enjoyable with all the social activity and many dinner and lunch invitations.

Thanks to my parents and sister for their love, support and belief in me at all times, and for haggling me about my graduation every time I called home.

A very special note of thanks to my girlfriend Eva for her support, love and encouragement at all times.

There are many more people to be thanked and I may have forgotten to do so, but their help, friendship and advice is always appreciated.

Finally, thanks to God for all his blessings during good times and bad times.

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

Page

TABLE OF CONTENTS (Continued)

Page

LIST OF FIGURES

LIST OF FIGURES (Continued)

Figure Page

LIST OF FIGURES (Continued)

Generalized Radix Design Techniques For Low-Power, Low-Voltage Pipelined & Cyclic Analog-Digital Converters

1 Introduction

Pipelined ADCs offer the only feasible solution for high-speed, high-resolution applications. Thus, it is of great interest to optimize them at both the system level, and circuit level, in order to achieve the lowest possible power consumption. Several researchers have addressed this problem, and a quick overview of the optimization techniques is presented. In addition, a simple but accurate model is presented to provide understanding of the tradeoffs involved, and this is subsequently used to find optimal design parameters. A novel technique is then presented to allow realization of the optimal design. The efficacy of this technique is demonstrated through extensive simulations. As cyclic converters are fundamentally similar to pipelined converters, and are merely duals of the former, all the results apply to design of these as well.

1.1 Background

The ideas presented in this dissertation have evolved from research work directed towards design of high-performance, high-resolution ADCs capable of low-power, low-voltage operation.

The stage resolution of a pipelined ADC is a critical parameter affecting cost, performance and overall power consumption. This work attempts to find the optimal solution to the problem of designing a pipelined ADC under a given set of

constraints. A generalized radix implementation scheme is then presented to realize an optimal pipelined ADC. This technique is shown to be superior to the conventional scheme of realizing a pipelined ADC resolving an integer number of bits per stage.

The inherent duality between cyclic and pipelined ADCs allows application of similar techniques to optimization of cyclic ADCs as well.

1.2 Motivation

Aggressive process scaling as predicted by Moore's Law has established CMOS technology as the dominant force in the semiconductor industry, with exponentially increasing levels of monolithic integration, processing capability and speed. However, this picture is true largely for digital circuitry, which requires ADCs and DACs in order to interface with the analog world. Analog circuit design has not received such a shot in the arm from process scaling, and even in terms of raw speed, analog circuits have not scaled as fast as their digital counterparts. This has placed severe demands on ADC performance. With reducing supply voltage, short-channel effects, and increasing speed requirements, only pipelined ADCs are capable of meeting the requirements of high resolution at a reasonable cost. Even these are getting harder to design within a reasonable power budget owing to the demands placed on them, and the task will surely get harder with the drive towards 100nm gate length, and sub-1V power supply. Thus, the topic of optimizing power consumption of pipelined ADCs has evinced considerable interest among

researchers, with several case studies on the subject [1] [2] [3]. However, the existing solutions for implementation are still far from optimal.

These reasons have motivated this research effort in an attempt to find optimal system-level parameters in order to harness the potential of pipelined ADCs to the limit.

1.3 Interstage Gain As A Crucial Parameter

Pipelined ADCs are the converters of choice for high-speed, high-resolution applications because they offer low-cost solutions for high-performance applications. However, they also offer an additional feature which is crucial – design flexibility. The design flexibility arises from the ability to distribute the task of a high-resolution conversion across many stages, each performing a fast, low-resolution conversion. This distribution of the conversion process across different stages can be quantified in terms of the interstage gain, or the number of bits resolved by each stage. The manner in which this flexibility is exercised can significantly impact the overall cost, performance and power consumption of the ADC. Substantial amount of research has focussed on optimizing power consumption of a pipelined ADC through appropriate choice of the interstage gain[1][2][3]. However, the tendency to think *binary*, coupled with the lack of techniques to implement non-binary interstage gain designs has led to sub-optimal implementations. A novel technique is proposed to realize pipelines with an arbitrary integer interstage gain, which is henceforth referred to as generalized

radix design. The effectiveness of this technique is demonstrated through the use of extensive simulations while the simplicity of design is self-evident, and there is little or no increase in complexity of the system.

1.4 Purpose Of This Work

The purpose of this research was to find a way to design a power-optimized pipelined ADC. As part of this work, it was necessary to find or develop a simple model useful for optimization purposes. Several such models of varying complexity are cited in literature pertaining to pipelined ADCs [1][2]. More importantly, the purpose was to develop the means for realizing the optimal interstage gain, which frequently happens to be a non-binary integer. Here, the term *binary* represents a number of the form 2^N. This task is complicated by the need for digital redundancy in the stage design. Digital redundancy is an indispensable feature of pipelined ADC stages, which offers them immunity from a host of non-idealities and renders the design robust, leading to the low-cost advantage that pipelined ADCs offer. The concept of digital redundancy is explained in detail in Chapter 3.

An additional constraint was to ensure that the outputs of the generalized radix stages be easy to implement using conventional digital logic. This is vital or else the advantages offered on the analog side would be nullified by the added complexity on the digital side.

1.5 Dissertation Structure

Chapter 2 offers an introduction to the basics of A/D conversion, and traces the conceptual evolution of the pipelined ADC. The concept of digital redundancy and error correction are introduced at this stage.

Chapter 3 introduces the cyclic ADC as a low-power solution, and as a dual of the pipelined ADC. A few implementation details are covered. Further ideas for improving efficiency of cyclic ADCs are considered, and the idea of utilizing a non-binary gain is raised here.

Chapter 4 introduces a simplified model of the pipelined ADC, along with critical parameters and attempts to address the issue of optimizing pipelined ADCs for minimum power consumption. Most of this work derives from [1], although it has been modified and augmented a bit. The results of optimization are presented for a wide variety of cases.

Chapter 5 introduces the concept of generalized radix stages and treats the subject in detail. This chapter also presents a methodology for the design of such stages with provision for digital redundancy and error correction. Examples are used to illustrate the advantages of the proposed technique over existing schemes for realization of pipelined ADCs.

Chapter 6 briefly looks at a few recent developments in the field of design of pipelined and cyclic ADCs for low-power, low-voltage applications. Topics such as time-delayed CDS and chirp clocking are covered here.

Chapter 7 concludes this dissertation and presents the key results obtained in this work.

2 Introduction To Pipelined ADCs

In this chapter, the basic idea underlying A/D conversion is presented. The flash ADC is shown to be a natural implementation of the concept. Some of the issues affecting practical realizations of ADCs are discussed, and evolution of pipelined ADCs from flash ADCs is traced. In addition, the concept of digital redundancy and error correction is introduced.

2.1 Basics Of A/D Conversion

The process of A/D conversion consists of *sampling* the signal [1], *quantization* of the samples by comparing them with a set of reference levels, and *encoding* the output to facilitate digital signal processing. Thus, a real, continuous signal is converted into a dimensionless code, which is meaningful only when viewed as an equivalent analog value (2-1). Different ADC architectures combine and implement the above tasks in different ways. The simplest ADC is the flash ADC, and its operation is illustrated with the example of a 2-bit ADC.

$$V_{quantized}(k) = V_{REF} \cdot \sum_{i=1}^{N} b_i(k).2^{-(k-1)} \qquad \textbf{(2-1)}$$

[1] An anti-aliasing filter may be required to satisfy the Nyquist sampling criterion.

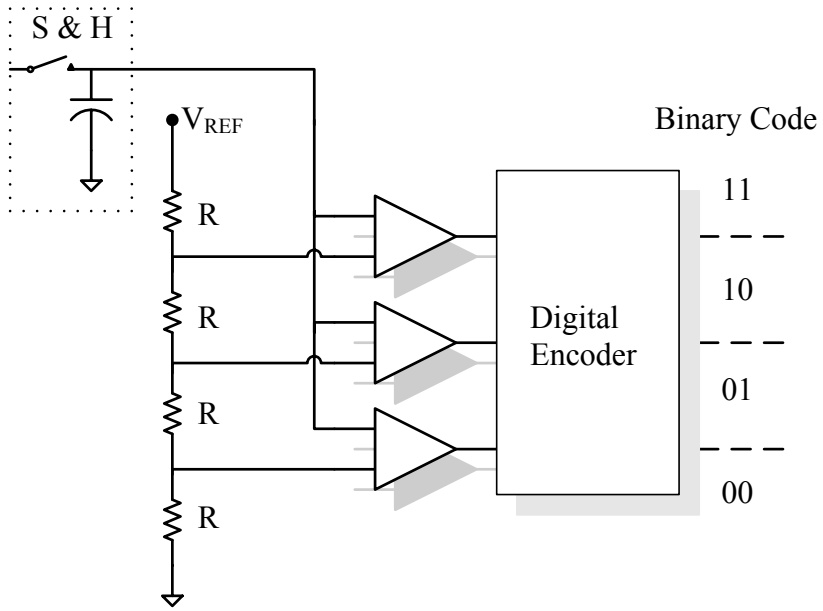


Figure 2-1 A 2-bit flash ADC

Typically, the sampled & held input is compared to a series of reference voltages generated using a resistor-string ladder, and the thermometer output code from the comparators is converted to the appropriate binary code using an encoder[2] (not shown in figure).

2.1.1 ADC Characteristics

It should be evident that the process of A/D conversion introduces a certain amount of additive *quantization noise* in the signal as a real, continuous-valued signal is now represented by a finite number of bits. Even an ideal A/D converter with a finite number of bits will possess this error (Fig. 2-2). In the presence of non-idealities[28][29], the output signal will be corrupted further and it is

[2] Typically, the output code is first converted to a Gray code and digitally processed to minimize the impact of *bubble errors* prior to conversion to the desired binary code.

customary to denote the *effective number of bits (ENOB)* of an ADC as the output SNR obtained when the input is a full-scale sinewave. For these conditions, the SNR of an ideal N-bit ADC is given by (2-2). The least count of the ADC is called the LSB (least significant bit).

$$SNR = 6.02N + 1.76(dB) \qquad \textbf{(2-2)}$$

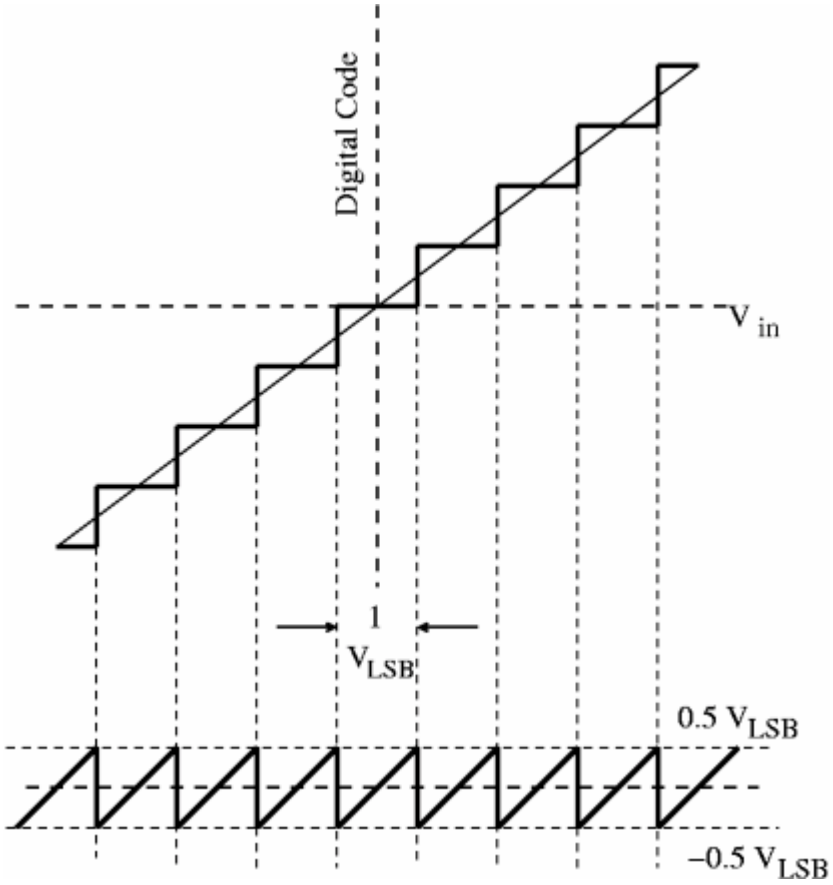


Figure 2-2 Ideal Transfer Curve & Quantization Noise for a 9-level ADC

2.1.2 Issues In Flash ADC Design

One of the most important issues in flash ADC design is the exponential increase in the required number of comparators and reference voltages, causing the cost factor to shoot up exponentially (~2^N). The large number of comparators presents a huge load capacitance to the sample & hold amplifier, making it hard to achieve high-speed operation. This can be remedied somewhat by the use of interpolating architectures [28][29].

The design of comparators also becomes an onerous task with the increasing number of levels, as the input offset voltage specifications get tighter. This does not allows the use of low-power, dynamic comparators, and increases the power consumption as static preamplifiers or other offset minimization techniques must be used [30]. Additionally, it becomes very hard to generate accurate reference voltages using the resistor string.

All these factors typically limit the use of flash ADCs to 8-10 bit applications, and the associated cost is still quite high. However, there is need for high-resolution, high-speed ADCs for video, broadband and other such applications, and no other scheme can meet these requirements. These difficulties led to the introduction of the two-step architecture.

2.1.3 The Two-Step ADC

The two-step architecture distributes the task of a high-resolution ADC conversion into two medium-resolution ADCs. Fig. 2-3 illustrates the operation of an 8-bit ADC working on the two-step principle. The MSB flash ADC determines the first four MSBs, and the quantization error of this ADC is computed by the DAC and subtractor, and passed on after appropriate scaling to the LSB flash ADC. This implementation requires only $2.2^{(N/2)}$ comparators, which is almost the square root of the original 2^N comparators. However, the latency of the ADC is now increased as the signal path has many more components. In addition, a DAC, a gain block and a subtractor are now required.

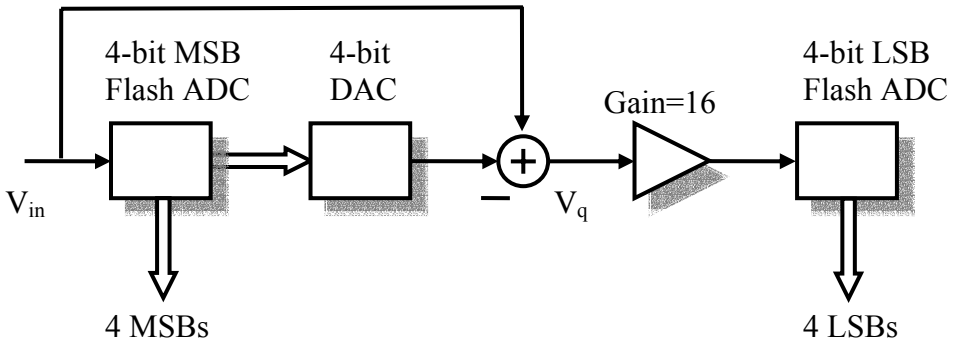


Figure 2-3 An 8-bit two-step ADC

This architecture requires all blocks to be 8-bit accurate, but using digital redundancy and error correction can ease these requirements. This concept will be discussed later. One can also see that it is possible to extend the two-step idea to a multiple-step ADC. However, this approach is not very desirable for high-speed

applications as one has to wait for the signal to ripple through the entire cascade which now includes several gain blocks, each realized using a compensated opamp. This makes the approach unattractive from the viewpoint of conversion rate of this ADC.

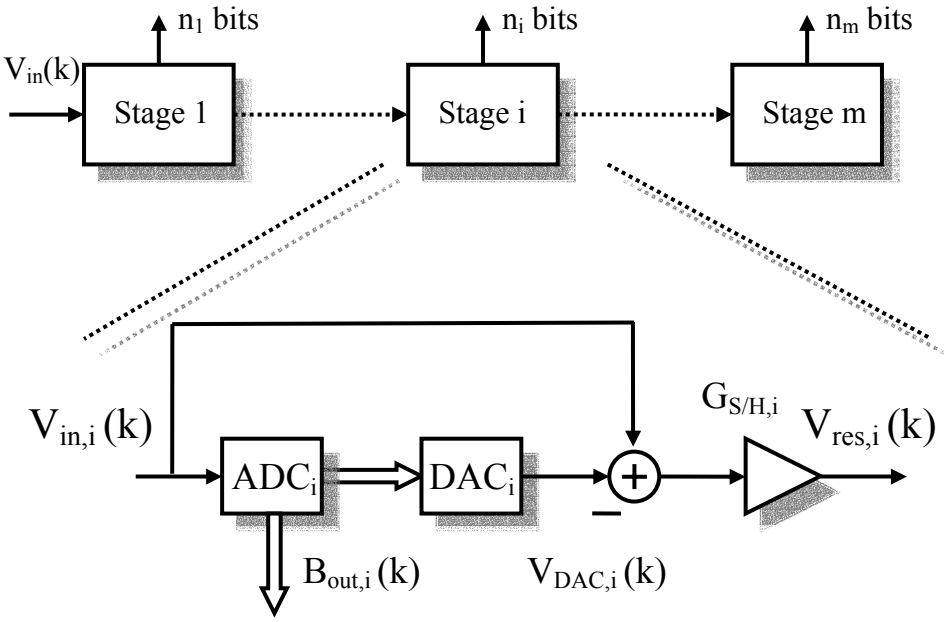


Figure 2-4 The Pipelined ADC

2.2 The Pipelined ADC

The restriction on conversion rate arises because the current signal sample must be completely processed by all stages of the ADC before the input signal is sampled again. However, introducing a S&H between every two stages can solve this problem. Now, each stage receives a scaled-up remainder from the previous stage and can quantize it independently, and pass on the remainder that it generates to the next stage for further resolution. Thus, the process of a high-resolution A/D conversion has been *pipelined* into several concurrently running low-resolution

A/D conversions. The latency of this ADC is still quite large and similar to the multi-step (also known as subranging) ADC discussed in the previous section. However, the presence of a S&H at the input of each stage has increased the throughput significantly, as the input can now be sampled again as soon as the current sample has been processed by the first stage.

We still have the difficult task of meeting the accuracy requirement for design of each stage, with the most difficult part being the accurate design of the first stage. These requirements are eased considerably by incorporating digital redundancy and error correction into the design. These concepts are now outlined.

2.2.1 Digital Redundancy & Error Correction

In the pipelined ADC scheme described so far, the accuracy requirements for most blocks are quite stringent. Consider the ADC in the first stage. This ADC must be as accurate as the entire converter, and any input offset in the comparator will shift the ADC decision thresholds by an equal amount. Thus, the comparator offsets for this stage must be very tightly controlled (< 0.5 LSB) if the overall resolution requirement is 10 bits or more. Even for moderate resolution requirements such as 6-8 bits, it is not easy to achieve the low values of comparator offset required with dynamic comparator designs, as these have offsets in the range of 100mV. Reducing supply voltages makes the task even harder. The accuracy requirement of the blocks reduces as we move down the pipeline because the required resolution there is lower. Nonetheless, it is still impractical to realize designs meeting such stringent requirements.

In the pipelined ADC, it is convenient to look at the transfer characteristic of the output residue $V_{res,i}(k)$ with respect to the input $V_{in,i}(k)$. Let us see consider the example of a pipelined ADC resolving 2 bits/stage. The residue transfer curve of this stage is shown in Fig. 2-5, with and without offset in the first comparator.

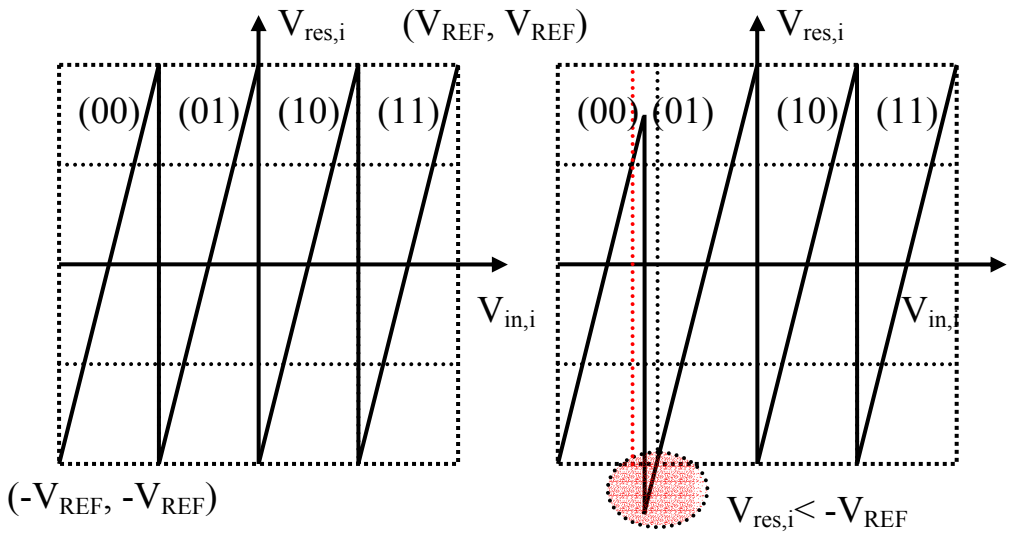


Figure 2-5 Effect of comparator offset on residue transfer curve

It is apparent that any comparator offset causes the residue to go out of range, thus introducing missing levels in the overall ADC transfer curve. Now, consider the scenario where the *interstage gain* G_i is reduced by half from 4 to 2. This is depicted in Fig. 2-6. Now, the residue output is bounded to half of the full-scale range if there is no comparator offset in this stage. Additionally, any comparator offset shifts the decision levels, but the output still stays within range provided the error is less than 0.5 LSB_i, where LSB_i is merely the LSB of this 2-bit stage. Thus, the ADC in each stage must be just as accurate as the *stage resolution*

n_i. Since the gain of the stage is reduced by a factor of 2 as compared to its nominal value, we say that the stage has 1 bit ($\log_2 2$) of redundancy. Now, let us look at the behavior of 2 such stages in cascade, assuming that the first stage suffers from comparator offsets, while the second one is ideal (Fig. 2-7).

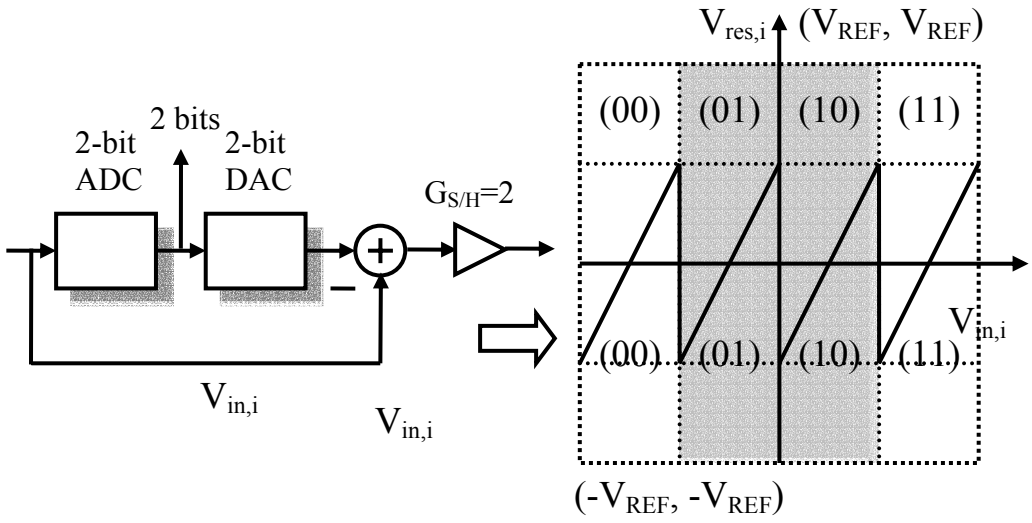


Figure 2-6 2-bit pipeline stage with reduced gain

If the first stage is ideal, then the input to the second stage is bounded between -0.5 V_{ref} and 0.5 V_{ref}, and its digital output code should be either (01) or (10). The input to the second stage should ideally fall in the *residue range* marked in gray. Any *overrange error* can be used to detect the nature of comparator offset in the first stage.

A negative offset in this comparator threshold produces a digital output code of (01) for an input which is slightly lower than 0.5 V_{ref}. The output code is 1 LSB greater than the actual value, and thus the output code must be corrected by subtracting 1 LSB from it. To do this, the next stage must be able to detect the

negative offset in this stage. This can be done as the next stage sees an input lower than -0.5 V_{ref} (left of gray region), and detects this as a negative overrange error.

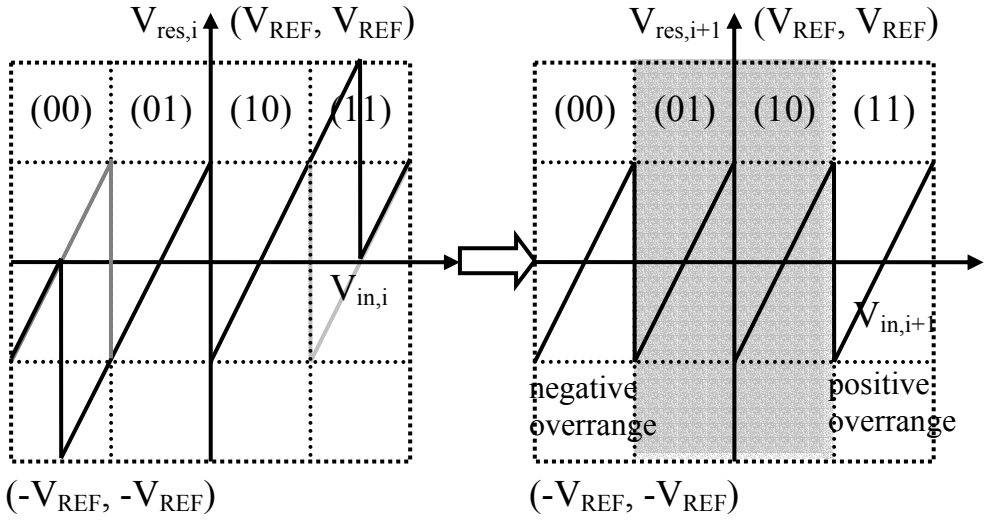


Figure 2-7 Illustration of the digital correction process

Similarly, a positive offset in the comparator threshold produces a digital output code of (00) for an input which is slightly larger than 0.5 V_{ref} (right of gray region). Since the correct output code would have been (01), this must be corrected by adding 1 LSB to the output code. Again, the presence of this positive offset can be detected in form of a positive overrange error in the second stage.

Similar arguments can be used to show that an offset in any comparator of the first stage can be detected provided it is less than 0.5 LSB_1, and the second stage is 2-bit accurate.

Thus, we have achieved our goal of rendering the design of the ADC intolerant to comparator offsets. Similar arguments can used to show that this scheme offers immunity from comparator offsets in the two-step ADC architecture

discussed earlier provided the offset is less than 0.5 LSB_i. The DAC does not benefit from this technique and must remain as accurate as the number of bits still to be resolved down the pipeline.Although we have managed to solve a big part of the problem, the digital error correction still remains a little complicated, as we need to first detect the nature of the overrange error (positive/negative), and then perform an addition/subtraction depending on this. This introduces two steps in the correction algorithm and also calls for two different blocks, each of which might be used only 50% of the time on average. Thus, it is desirable to modify the correction scheme.

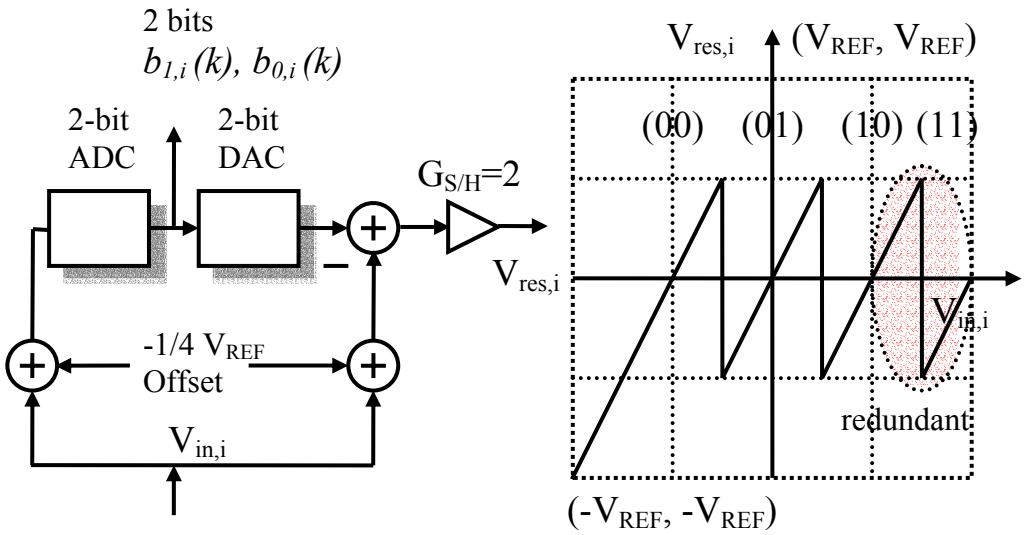


Figure 2-8 Modified pipeline stage with added offset

Let us consider the residue transfer curve shown in Fig. 2-8. Here, we have deliberately added an offset of 0.25 V_{ref} to the transfer curve depicted in Fig. 2-6. Thus, we have ensured that any error caused by a negative comparator offset is automatically absorbed. Also, the last comparator is now redundant, as a positive

overrange error can be detected by the next stage. So, this is thrown away leaving just 2 comparators. This gives us the *1.5-bit stage* (Fig. 2-9).

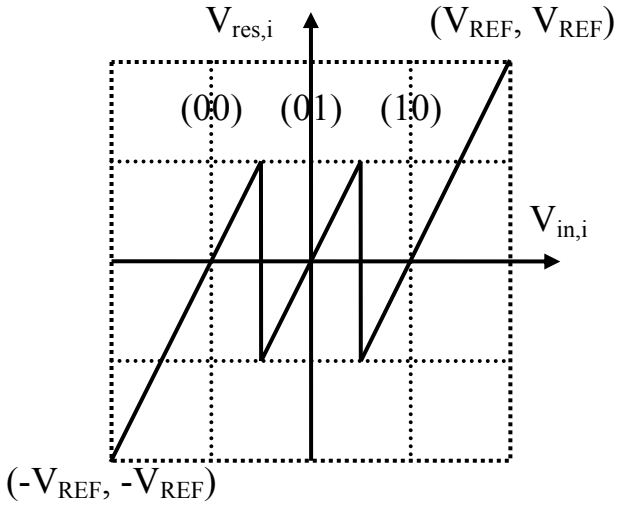


Figure 2-9 Residue transfer curve for 1.5-bit stage

It can be shown that the digital correction is performed automatically by the *overlap & add* scheme shown in (2-3). An important point to note is that this correction scheme simply accounts for the *analog interstage gain* (in this case, 2) in the digital domain while adding the output codes from the various stages. It can be compared to the scheme for adding bits from all the pipeline stages in the absence of digital redundancy and error correction (2-4). Thus, this is the simplest possible correction scheme, and cannot be improved any further. The significance of this observation will be clearer when the *generalized radix pipeline stage* is introduced.

In conclusion, digital redundancy helps improve the robustness of each stage to component non-idealities while ensuring that there is no excess overhead,

delay or inefficient resource usage. This scheme of implementing digital redundancy and error correction has been widely used to design high-performance, high-resolution ADCs [1][2][4][5][6], and is an indispensable aspect of pipelined ADC design. The overall ADC resolution can be written in terms of the individual stage resolutions as shown in (2-5).

$$\begin{array}{cccccccc} b_{1,1} & b_{0,1} & & & & & & \\ & b_{1,2} & b_{0,2} & & & & & \\ & & \cdots & \cdots & & & & \\ & & & & b_{1,m} & b_{0,m} & & \\ \hline B_0 & B_1 & \cdots & \cdots & B_{m-1} & B_m & \end{array}; \; Addition\ of\ bit\ outputs, G = 2 \tag{2-3}$$

$$\begin{array}{cccccccccc} b_{1,1} & b_{0,1} & & & & & & & & \\ & & b_{1,2} & b_{0,2} & & & & & & \\ & & & & \cdots & & & & & \\ & & & & & & b_{1,m} & b_{0,m} & & \\ \hline b_{1,1} & b_{0,1} & b_{1,2} & b_{0,2} & \cdots & b_{1,m} & b_{0,m} & & & \end{array}; \; Addition\ of\ bit\ outputs, G = 4 \tag{2-4}$$

$$N_{ADC} \le \sum_{i=1}^{m} n_i; \; Number\ of\ Stages = m \tag{2-5}$$

3 Cyclic ADCs As A Low-Power Solution

In this chapter, the cyclic ADC is introduced as the dual of the pipelined ADC, trading speed with cost. All the results derived for the pipelined ADC in the previous chapter are applicable to cyclic ADCs. The conventional implementation scheme for cyclic ADCs is shown.

3.1 The Cyclic ADC

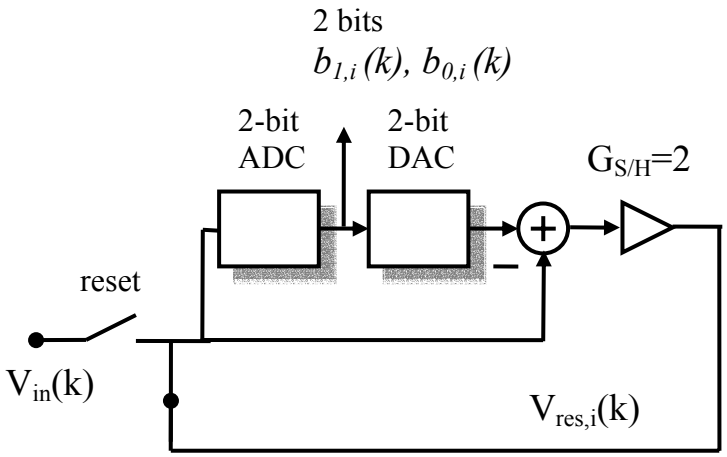


Figure 3-1 A 1.5-bit/stage pipeline modified to a cyclic ADC

Let us consider a modified form of the pipelined ADC as shown in Fig. 3-1. It has only one pipeline stage, which recycles its own residue, instead of passing it on to another stage. Digital redundancy is assumed to be in-built as seen from the *interstage gain* of 2 instead of 4, and the details are omitted for simplicity. The main input is sampled once during a *reset phase* and the loop is closed after this point for *m* cycles in which the stage resolves the current input, generates a residue and feeds it back as the next input, until the resolution is complete. After this, it

samples the main input V_{in} again. Thus, the operation is identical to an *m*-stage pipeline, except that the input conversion rate is now reduced.

The cyclic ADC is also known as the *algorithmic ADC*. However, this name often causes confusion. In reality, all ADCs follow a certain algorithm, which is usually a variation of long division implemented in a base 2 system. The cyclic ADC and the successive-approximation ADC (SAR ADC) work on two different variations of the *binary search* algorithm[3] known as the *multiplied remainder approach* and the *divided reference approach* respectively [29]. Thus, the underlying technique is the same for both. However, the SAR ADC requires the DAC to generate all the 2^N levels, and its maximum accuracy is typically limited by matching considerations in the resistor-string used to generate these levels. The cyclic ADC on the other hand will be shown to require only a handful of accurate reference voltages, and can be implemented conveniently using switched-capacitor circuits. With digital calibration and mismatch-shaping techniques, the cyclic ADC can provide 12-14 bits of resolution [19][21][22].

3.2 Implementation Techniques & Issues

The following discussion uses the example of a 1.5-bit stage. However, the results can be easily generalized to any arbitrary cyclic ADC or a pipelined ADC with each stage resolving an equal number of levels. As shown earlier, digital redundancy eases the accuracy requirements of the ADC. However, the design of

[3] Consider the operation of the ADC without digital redundancy

the DAC, the S&H, the gain block and the subtractor remains critical. With a switched-capacitor circuit implementation, all these functions can be lumped into one single circuit called the *multiplying DAC* or MDAC. A typical implementation is shown in Fig. 3-2. Single-ended representations are shown for simplicity.

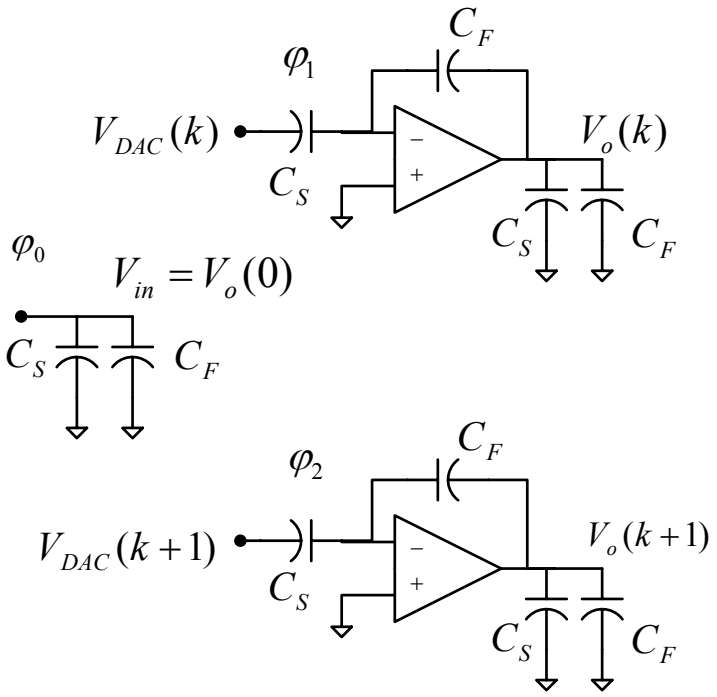


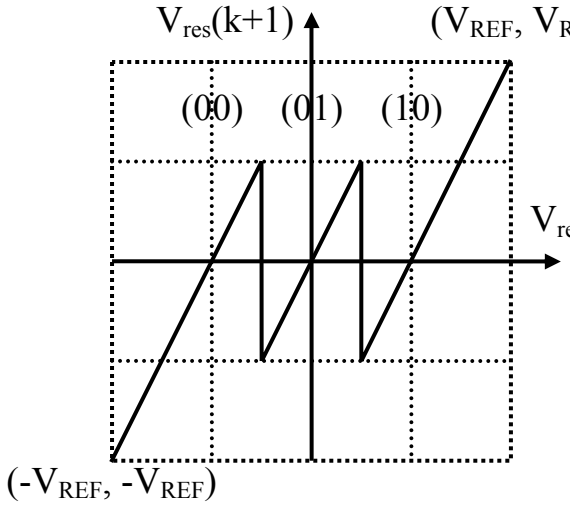
Figure 3-2 Conventional MDAC scheme

The design of an accurate gain stage is a difficult task as the gain of this stage must be nearly as accurate as the entire converter itself, and any non-idealities such as capacitor mismatch, insufficient settling time, charge-injection and clock feedthrough introduce non-linearities in the overall transfer curve of the ADC. Circuit techniques combined with fully differential design can eliminate the effect of the last two [27]. Section 3.3 summarizes the effect of a few such non-idealities on cyclic ADCs. Special circuits may be used to effect mismatch shaping

and to control the gain with great accuracy [16][17][18]. However, all these schemes typically require extra clock cycles and implement an N-bit conversion in 2N clock cycles.

$$V_{res}(k+1) = (1+\frac{C_s}{C_f}).V_{res}(k) - \frac{C_s}{C_f}.V_{DAC}(k) = 2.V_{res}(k) - V_{DAC}(k)$$
$$V_{res}(0) = V_{in} \qquad \textbf{(3-1)}$$

The sampling capacitor C_S is usually made with unit elements with value C. The DAC operation can then be realized easily by connecting the bottom plates of these capacitors to $+V_{REF}$, $-V_{REF}$ or 0 depending on the previous bit output. In the case of the 1.5-bit stage, there is only one unit capacitor connected to V_{DAC}.



$$V_{DAC}(k) = \begin{cases} +V_{REF}, B_{out} = 00 \\ 0, \quad B_{out} = 01 \\ -V_{REF}, B_{out} = 10 \end{cases}$$

The DAC can be realized using a few switches if $+V_{REF}/-V_{REF}$ are available.

Figure 3-3 Residue transfer characteristic for ADC

3.3 Effect Of Nonidealities On ADC Performance

The accuracy and performance of an ADC are affected by various circuit non-idealities [27][28]. These can be broadly classified into static and dynamic errors. Static errors are independent of the frequency of operation of the circuit and include comparator and amplifier offsets, capacitor mismatch, finite amplifier DC gain and amplifier nonlinearity. Dynamic errors include charge-injection, clock feedthrough, incomplete amplifier settling, comparator metastability, timing errors and other such effects. In addition, device noise is an important limiting factor. Only the important factors are addressed here, and their effect on circuit performance and specifications is quickly reviewed. A more detailed treatment may be found in [1][28][30].

3.3.1 Comparator Offset

As discussed in the earlier chapters and sections, in-built digital redundancy and error correction suppress the effect of comparator offset provided the offset is less than 0.5 $V_{LSB,i}$, where $V_{LSB,i}$ represents the LSB of the ADC stage ($V_{REF}/4$ in this case).

3.3.2 Opamp Offset

Let us review the operation of the MDAC again (Figs. 3-2). A simple construction (Fig. 3-4) shows that the opamp offset doubles when referred from the non-inverting input of the opamp to the input of the MDAC. This input-referred offset does not introduce any distortion in the overall transfer curve of the

ADC and merely reduces the dynamic range by a few LSBs. Thus, it is not too critical in most cases. If it is, then offset-cancellation techniques should be used [26].

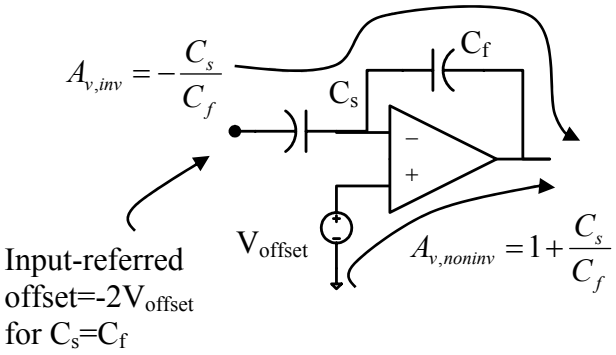


Figure 3-4 Effect of opamp offset

3.3.3 Finite Opamp Gain

Finite DC gain of the opamp in conjunction with parasitic capacitances at the virtual ground node cause the charge-transfer to deviate from the ideal scenario considered so far. Thus, the interstage gain is affected thereby introducing non-linearity in the system. The effect of these was computed using the MDAC model shown in Fig. 3-5. The result is shown in (3-2).

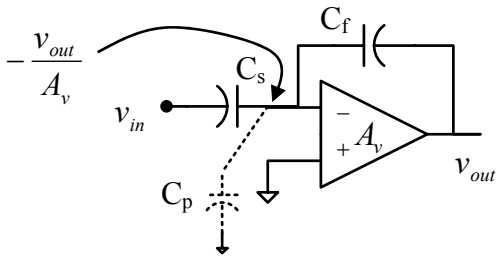


Figure 3-5 MDAC model for estimating effect of finite opamp gain

$$V_o(k+1) = \left[\frac{C_s + C_f}{C_s + \frac{C_s + C_f - C_p}{A_v}}\right] V_o(k) - \left[\frac{C_f}{C_s + \frac{C_s + C_f - C_p}{A_v}}\right] V_{DAC}(k) \quad \textbf{(3-2)}$$

$$V_o(k+1) \approx 2\left(1 - \frac{2}{A_v}\right) V_0(k) - \left(1 - \frac{2}{A_v}\right) V_{DAC}(k); \; for \; C_p = 0, C_s = C_f,$$

$$A_v >> 1;$$

3.3.4 Capacitor Mismatch

Systematic and random process variations introduce mismatch between the capacitors. While systematic variations may be corrected for by careful layout and common-centroid techniques, some variations still remain and unless some sort of mismatch shaping scheme is employed, these limit the matching accuracy to about 0.1%. The exact effect of capacitor mismatch on the converter linearity must be determined using behavioral simulations. Digital calibration techniques may be used to alleviate the effect of mismatch for high-resolution converters.

3.3.5 Thermal Noise

Thermal noise is present in all circuits owing to resistive elements. The two principal contributors to thermal noise are the opamp and the switches. The finite resistance of the switches contributes thermal noise, which is sampled onto the capacitors to yield the *kT/C noise* [27][30].

3.3.5.1 Thermal Noise Calculation

The following derivation assumes the use of single-stage amplifiers for both the conventional and proposed schemes. The current density is assumed to be constant, leading to a fixed unity gain frequency. More details regarding this model may be found in the next chapter, where it is used extensively and also in [1]. With these assumptions, the input-referred thermal noise power for the conventional scheme is written as:

$$\overline{V_{n,conv}^2} = \frac{4kT}{3C_{F,conv}} + \frac{4kT}{18C_{F,conv}} = \frac{14kT}{9C_{F,conv}} \quad \textbf{(3-3)}$$

The first term represents noise due to switches, while the second term represents the thermal noise of the opamp. The opamp output parasitic capacitance is ignored here as the load capacitance is assumed to be much larger.

The load capacitance seen by the opamp is $2.5C_{F,conv}$. The finite opamp bandwidth has been accounted for in this analysis. Similar analyses can be performed for other opamp configurations.

3.4 Conclusion

The cyclic ADC scheme offers potential for low-power operation. However, some more options are worth pursuing for design of power-efficient ADCs. Techniques such as *entropy coding* have been used successfully in SAR ADC design [25]. These schemes reduce the average number of conversions

performed by investing more information content per symbol. This suggests the possibility of using non-binary radix schemes to increase information efficiency per symbol or resolving more bits per stage as the bulk of the power dissipation is accounted for by the opamps used in the MDAC. However, resolving more bits per stage involves using an ADC with higher resolution per stage. We had started our discussion of pipelined ADCs in an attempt to break down a high-resolution flash conversion into simpler, more manageable and cost-efficient steps. Reverting to it may seem counterproductive. However, it is not all that simple. Detailed discussions in the future chapters will elucidate this point.

4 Optimizing Pipelined ADC Design

As outlined towards the end of the last chapter, it might be worthwhile pursuing the option of resolving more bits per stage to improve the power efficiency of a cyclic ADC. As explained earlier, the cyclic ADC is a dual of the pipelined ADC, and any results and analyses concerning one may be easily applied to the other, offering considerable benefits as the pipelined ADC finds much more widespread usage and its optimization is therefore a topic of considerable interest.

As this subject has been covered extensively by many researchers [1][2][3], this work does not aim to find new optimization methods but merely attempts to provide a quick overview of the topic, based largely on the excellent work by Cline [1][4]. For this purpose, the basics of pipelined ADC implementation are reviewed briefly in order to determine some of the critical parameters and the various tradeoffs associated with them. The interstage gain is then identified as one of the critical parameters, and attempts are made to find the optimal stage resolution in terms of this gain which is allowed to take *any integer value* for implementation purposes. This is a major departure from conventional research which being *binary* in thought, considers only implementations with an integral number of bits per stage, and gain of the form 2^L. This *generalized radix* implementation itself is the subject of the following chapter. Most of the optimization is carried out numerically. As emphasized earlier, the purpose of this work is not to provide a technique of optimization but merely to present it as a context for the following work.

4.1 Problem – Division Of Labor

Consider the problem of designing a 14-bit pipelined ADC. The task can be divided in many possible ways. A few schemes are shown in Fig. 4-1.

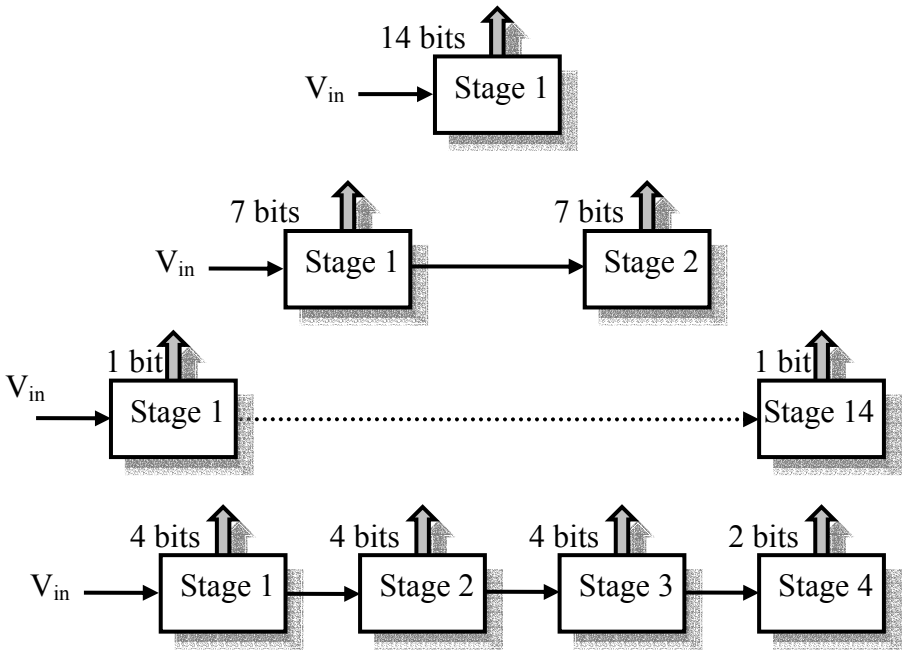
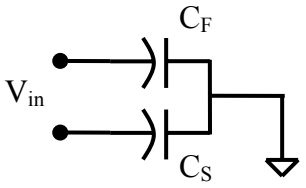


Figure 4-1 A few possible ways to realize a 14-bit pipelined ADC

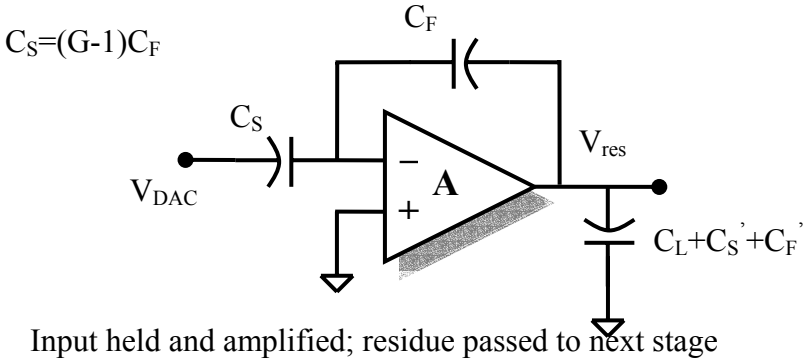
The above list is by no means exhaustive, and we are not even considering the possibility of resolving a fractional number of bits per stage which will be addressed in the following chapter (not to be confused with 1.5-bit/stage, which is merely a term). Clearly, the optimal configuration may be any of these choices, depending on the specifications, and it might offer substantial savings in cost and power consumption, and is worth pursuing for these reasons.

4.1.1 Basic Implementation Scheme

Let us consider a single pipeline stage. Once again, we focus on the S&H amplifier section as this is the limiting part for both speed and power consumption. The ADC within each stage contains just the comparators, and these are usually quite fast, and typically consume only dynamic power. Besides, the design of individual comparators is typically independent of the stage resolution. A S&H amplifier section is shown in Fig. 4-2. The load capacitance C_L accounts for the parasitic capacitance due to interconnects and comparator input capacitance. As the S&H output must drive the input of the next pipeline stage, C_S' and C_F' of the next stage are also added at the output.



Sampling of input from previous stage



Input held and amplified; residue passed to next stage

Figure 4-2 A Typical S&H Amplifier with gain G

4.1.2 Basic Implementation Issues

This section provides a qualitative understanding of the various parameters affecting the performance of a pipeline stage utlising the S&H amplifier shown above. These include choice of capacitor sizes, capacitance ratio and supply voltage. The opamp is assumed to be a single-stage, single-transistor amplifier, with an equivalent circuit as shown in Fig. 4-3.

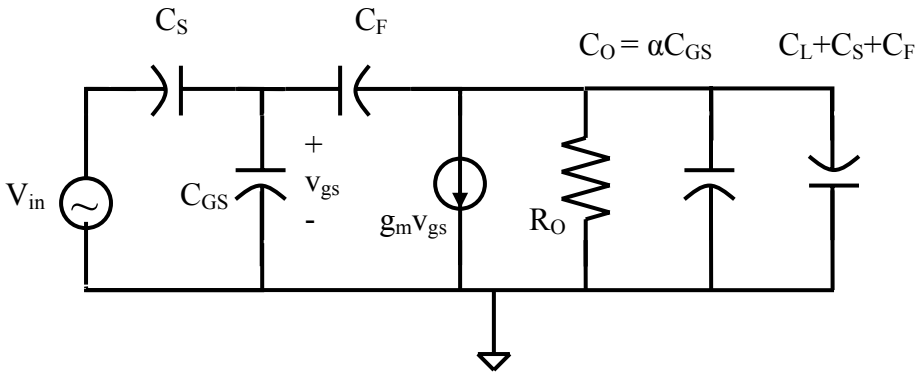


Figure 4-3 Equivalent circuit for S&H during amplification phase

Additionally, it is assumed that the current density for the transistor remains constant. This assumption sets the unity-gain frequency of the amplifier to a constant value for a given process, and a fixed channel length (L). This derivation is shown in (4-1). Now, the current consumption of the amplifier is proportional to C_{GS}^2. Additionally, all the operating speeds are normalized with respect to the unity-gain frequency for the process used. This makes the analysis technology-independent to a good approximation.

$$\rho = \frac{I_{DS}}{W}; C_{GS} = \left(\frac{2}{3}\right) WLC_{ox}$$
$$g_m = k_n^{'}\left(\frac{W}{L}\right)(V_{GS} - V_T) = \sqrt{\frac{2k_n^{'} WI_{DS}}{L}} \Rightarrow P = V_{dd} I_{DS} \propto g_m^2 \quad \textbf{(4-1)}$$
$$\omega_T = \frac{g_m}{C_{GS}} = \left(\frac{3}{2}\right)\left(\sqrt{\frac{2}{k_n^{'} LC_{ox}^2}}\right)\sqrt{\rho} \propto \sqrt{\rho}$$

4.1.2.1 Capacitor Sizing

In the S&H amplifier shown above, an important design parameter is the absolute value of the capacitances chosen. In high-resolution converters, thermal noise becomes crucial and determines the minimum allowable values of the capacitances. The opamp thermal noise varies depending on the choice of opamp topology. The input referred noise power can be written as shown in (4-2). The effect of kT/C noise for a fully differential configuration is summarized as an input referred noise power as shown in (4-3), and the total thermal noise power is shown in (4-4). This sets a minimum limit on the choice of (C_S+C_F). These capacitors are usually scaled down as we progress along the pipeline, thus saving power.

Although this analysis will focus only on the single-stage amplifier, it is useful to have a crude estimate for other cases. With this objective, additional scaling factors are introduced for output load capacitance (α), input-referred noise (β) and total power consumption (γ) in terms of corresponding quantities as found for the single-stage case. For the simplest case, these assume unit value.

$$\overline{v_{in,diff}^2} = \frac{4\beta kTC_F}{3\left[C_L\left(C_S + C_F\right) + C_S C_F\right]} \tag{4-2}$$

$$\overline{v_{in,diff}^2} = \frac{2kT}{C_S + C_F} \tag{4-3}$$

$$\overline{v_{in,diff}^2} = \frac{2kT}{C_S + C_F} + \frac{4\beta kTC_F}{3\left[C_L\left(C_S + C_F\right) + C_S C_F\right]} \tag{4-4}$$

The other factor important for sizing of capacitors is the speed requirement. The closed-loop bandwidth of the amplifer is dependent on the feedback factor *f*, which is controlled by C_F. As C_F goes up, *f* goes up, but so does the load capacitance. Beyond a certain point, the benefits diminish and the loading effect of C_F starts to dominate. Thus, there exists an optimal value of C_F for a given load capacitance C_L. It should be kept in mind that increasing C_F also leads to an increase in C_S since the nominal value of gain is to be kept constant.

$$f = \frac{C_F}{C_F + C_S + C_{GS}} \tag{4-5}$$

4.1.2.2 Interstage Gain & Supply Voltage

An appropriate choice of the interstage gain can help save power. Consider a high-resolution pipeline where capacitor sizes are limited by thermal noise. In such a design, the capacitors may be scaled as we progress down the length of the pipeline where the accuracy requirements are reduced. This reduction is proportional to the interstage gain. Also, the choice of capacitor values for the first stage is typically controlled by the overall resolution requirement of the converter, and largely independent of interstage gain. Thus, a larger interstage gain allows for a more rapid scaling of capacitor sizes down the pipeline, thus saving power and die area.

However, as one approaches the latter stages of the pipeline, the reduced accuracy requirements might mean that the required capacitor value may be too small to be reliably fabricated, and the designer may have to use the minimum sized capacitor (~50fF in most processes). Hence, matching requirements are the controlling factor now. Thus, a high-resolution pipelined converter may be viewed as comprising of two sections – a thermal noise limited front end, and a mismatch limited rear end (Fig. 4-4). Thus, the optimal scaling factor tends to be nonuniform, reducing as one reaches the end, where no scaling is performed.

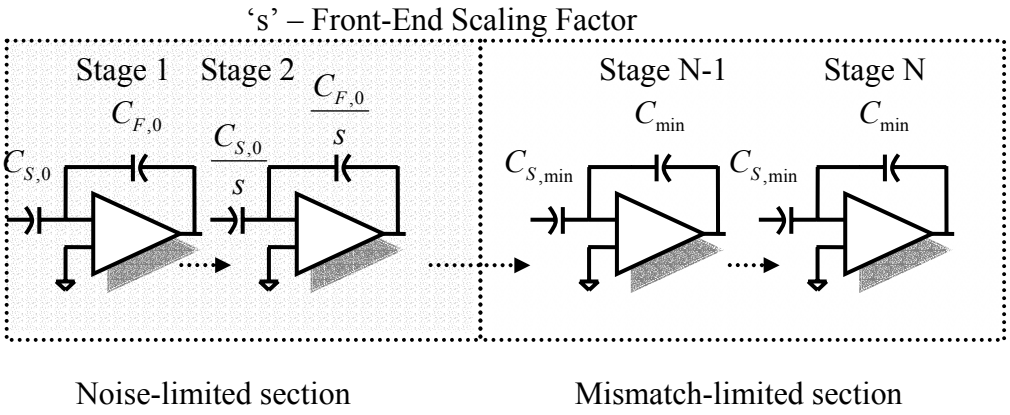


Figure 4-4 Non-uniform capacitor scaling in a pipelined ADC

Thermal noise is critical for converters with high resolution requirement and/or at low supply voltages. This is because the thermal noise itself is dependent only on the capacitor values (~$kT/C_{effective}$). The signal power on the other hand, is reduced by scaling of supply voltages (~V_{dd}^2).

In contrast, low-resolution converters are not limited by thermal noise, and the capacitor sizing (C_F) and current are dictated by speed requirements, with no room for scaling. Thus, it is advantageous to have lower supply voltage to minimize the power consumption (~V_{dd}.I) in such cases.

Higher supply voltages also offer better performance for analog blocks, and are thus more desirable. So, the designer should try and choose the appropriate supply voltage when the option is available. This freedom is not available in most cases, and with shrinking supply voltages, thermal noise usually is always a limiting factor.

4.2 Optimal Interstage Gain – Cline's Model

Now, the real problem of optimizing pipelined ADCs is considered for a variety of cases, beginning with the simplest case and progressing to increasingly complicated scenarios. Most of the results presented are cited from the work of Cline [1]. Others are based on modifications of these models, and the final case is analyzed numerically. The analysis becomes increasingly complex, and most of the advanced derivations may be skipped without loss of continuity, as the various plots provide sufficient insight.

4.2.1 Case I: Optimal Interstage Gain In The Absence Of Noise

This scenario is encountered in low to moderate resolution converters, and in cases where the available supply voltage is fairly large. This also applies to the later stages of a pipeline. The following model assumes a single-stage amplifier with fixed current density, and first-order settling. All assumptions are the same as in Section 4.1.2. The operating speed of the circuit is represented in terms of the unity-gain frequency ω_T. The time constant τ is a measure of the speed, and the normalized settling time θ can be written as $\omega_T\tau$. The comparators are assumed to load the opamps, but not to consume any static power. Thus, the power consumption of the ADC is proportional to the sum of the gate-source capacitances of all the opamps put together. No scaling is assumed and hence the sampling and feedback capacitors have the same value for all stages. With these assumptions, the time constant for the circuit of Fig. 4-3 is given by (4-6).

The minimum gate-source capacitance can then be derived using (4-5) and (4-6) and the result is shown in (4-7). This corresponds to minimum current.

$$\tau = \frac{C_S + C_F + C_L + C_F(1-f) + \alpha C_{GS}}{f g_m} \tag{4-6}$$

$$C_{GS,\min} = \frac{GC_L + 2(G^2 + G + 1)C_F}{\theta - (G+1) - \alpha G} \tag{4-7}$$

Combining (4-6) & (4-7) yields the following relations for C_F and C_{GS}:

$$\frac{C_F}{C_L} = \frac{\left[\theta - G - 1 - \alpha G\right]\left[1 + \sqrt{\frac{G\theta - 1}{G^2 + G + 1}}\right] + 2\alpha G}{\left[\theta - G - 1 - \alpha G\right]^2 - 4\alpha(G^2 + G + 1)} \tag{4-8}$$

$$\frac{C_{GS}}{C_L} = \frac{\left[\sqrt{\theta G - 1} + \sqrt{G^2 + G + 1}\right]^2 - \alpha G^2}{\left[\theta + G + 1 - \alpha G\right]^2 - 4\alpha(G^2 + G + 1)} \tag{4-9}$$

Now, we can use this information to find the sum of all the gate-source capacitances for a pipelined ADC with *n* bits per stage. The load capacitance for a typical pipeline stage with 1 bit of digital redundancy consists of the input capacitances of $2(2^n\text{-}1)$ comparators, and some interconnect capacitance (C_{int}),

which can be assumed to be equal to the input capacitance of one comparator C_{comp}. Thus, the capacitance loading an *n*-bit stage can be written as:

$$C_L = 2(2^n - 1)C_{comp} + C_{comp} \quad \textbf{(4-10)}$$

The number of stages $N_S=N_B/n$, where N_B is the overall resolution requirement of the ADC. The C_{GS} of each stage can be found by substituting (4-10) into (4-9), and this is multiplied by N_S to obtain the total input capacitance $C_{GS,total}$, which is given as:

$$\frac{C_{GS,total}}{N_B C_{comp}} = \frac{1}{n}\left[2(2^n - 1) + 1\right]\frac{\left[\sqrt{\theta G - 1} + \sqrt{G^2 + G + 1}\right]^2 - \alpha G^2}{\left[\theta + G + 1 - \alpha G\right]^2 - 4\alpha(G^2 + G + 1)} \quad \textbf{(4-11)}$$

This quantity is plotted in Fig. 4-5 for different speed and accuracy requirements. The optimal number of bits per stage is seen to be around 1 for almost all the cases, and this confirms the conjecture made in 4.1.2.2, where it was suggested that the lowest possible stage resolution may be optimal for a case where thermal noise is not an issue. Thus, dividing the ADC into 1-bit sections optimizes the overall performance in such cases.

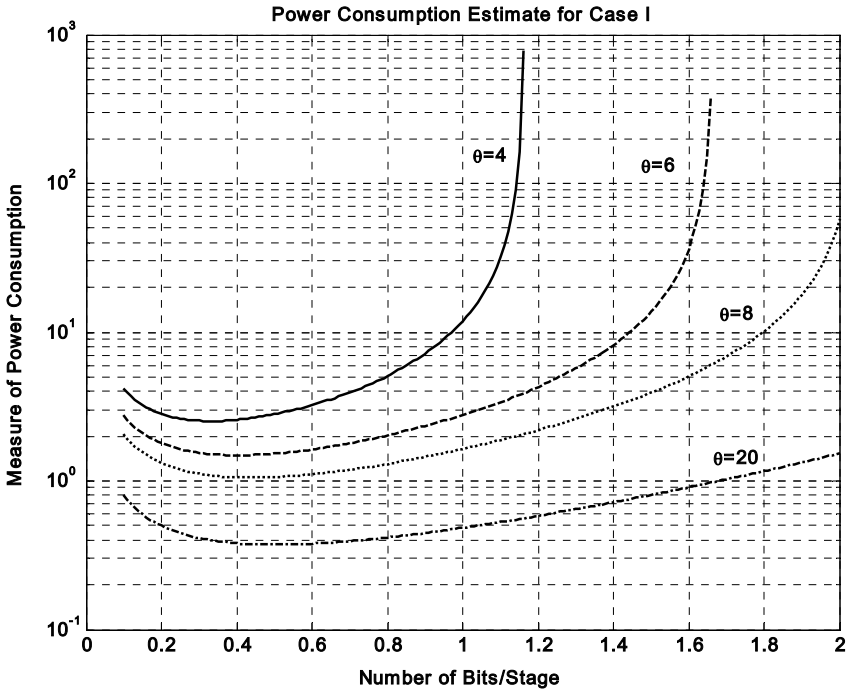


Figure 4-5 Power Consumption Vs. bits/stage for different speeds

4.2.2 Case II: Optimal Interstage Gain In The Presence Of Noise For High-Resolution, Low-Speed ADCs

In this section, thermal noise is taken into account. Minimizing the effect of thermal noise requires appropriate capacitor sizing. The capacitors may be scaled down as one progresses down the length of the pipeline, as shown in the first section of Fig. 4-4, and discussed earlier. This scaling factor depends on the gain. Let us consider the example of uniformly scaled stages, and look at the input-referred noise power. If these sources are assumed to be uncorrelated, then the input-referred noise power at the front-end of the pipeline can be written as:

$$\overline{v_{in,tot}^2} = \overline{v_{in,0}^2} + \frac{\overline{v_{in,1}^2}}{G^2} + \frac{\overline{v_{in,2}^2}}{G^4} + \ldots$$

(4-12)

$$\approx \overline{v_{in}^2}\left(\frac{G^2}{G^2-1}\right), \text{ with no scaling}$$

The thermal noise contribution is dependent only on the capacitor sizes, and if these are scaled down by a factor of G^2 as we progress along the pipeline, then the noise contribution of each stage at the front-end of the pipeline will be identical. A more aggressive scaling approach would make the noise performance of an intermediate stage more critical. This is an unacceptable scenario, and thus the capacitors should scale according to a factor G^y, where y < 2 in all cases. Practically, the *front-end scaling exponent* y is smaller than 2 as will be shown soon. Thus, the scaling factor *s* may be written as:

$$s = G^y \qquad \textbf{(4-13)}$$

Thus, the capacitor values of successive stages may be related as shown below:

$$C_{S,k+1} = \frac{C_{S,k}}{s} = \frac{C_{S,k}}{G^y} = \frac{C_{S,0}}{G^{(k+1)y}}$$
$$C_{F,k+1} = \frac{C_{F,k}}{s} = \frac{C_{F,k}}{G^y} = \frac{C_{F,0}}{G^{(k+1)y}} \quad \textbf{(4-14)}$$

Ignoring the C_{GS} and C_O of each opamp, we can express the load capacitance at each stage output as:

$$C_{L,k} = GC_{F,k+1} = G^{1-y}C_{F,k} \quad \textbf{(4-15)}$$

For y~1, the load capacitance is almost equal to the feedback capacitance, and the input-referred thermal noise (4-4) of the k^{th} stage may be written as:

$$\overline{v_{in,diff,k}^2} = \frac{2kT}{fG^2C_{F,k}}\left(1 + \frac{2\beta}{3\left(1 - f + G^{1-y}\right)}\right) \quad \textbf{(4-16)}$$

The contribution of the k^{th} stage is scaled down by all the intervening gain stages when referred to the input of the converter. Thus, the overall input thermal noise may be written as:

$$\overline{v_{in,diff,total}^2} = \sum_{k=0}^{N_S-1} \overline{v_{in,diff,k}^2} \approx \sum_{k=0}^{\infty} \overline{v_{in,diff,k}^2} = \frac{2kT}{fG^2 C_{F,0}} \left(1 + \frac{2\beta}{3\left(1 - f + G^{1-y}\right)}\right)\left(\frac{1}{1 - G^{y-2}}\right) \quad \textbf{(4-17)}$$

If the specification on input thermal noise is available, then one can use this value to compute the front-end feedback capacitance $C_{F,0}$ writing it as:

$$C_{F,0} = \frac{2kT}{fG^2 \overline{v_{in,diff,total}^2}} \left(1 + \frac{2\beta}{3\left(1 - f + G^{1-y}\right)}\right)\left(\frac{1}{1 - G^{y-2}}\right) \quad \textbf{(4-18)}$$

Using (4-6), we can relate the input transconductance of the opamp g_m to the value of the feedback capacitance (4-19), neglecting the effect of opamp capacitances for the while. This transconductance is a measure of the power consumption, as the current density is constant, and we just need to minimize it to find the optimal value of the scaling exponent. This calculation is shown below.

$$g_{m,k} = \frac{(C_{L,0} + C_{F,0}(1 - f))G^{-ky}}{f\tau} = \frac{C_{F,0}(G^{1-y} + 1 - f)G^{-ky}}{f\tau} \quad \textbf{(4-19)}$$

Thus, total input transconductance can be estimated as:

$$g_{m,total} \approx \sum_{k=0}^{\infty} g_{m,k} = \frac{C_{F,0}(G^{1-y} + 1 - f)}{f\tau}\left(\frac{1}{1-G^{-y}}\right) \quad \textbf{(4-20)}$$

Substituting the value of $C_{F,0}$ (4-18) into (4-20), we have

$$g_{m,total} = \frac{1}{f\tau}\left(\frac{2kT}{f\overline{v_{in,diff,total}^2}G^2}\right)\frac{\left(2 - f + \frac{2\beta}{3}\right)}{\left(1-G^{-y}\right)\left(1-G^{y-2}\right)} \quad \textbf{(4-21)}$$

Minimizing this with respect to y yields the following value for s_{opt}:

$$s_{opt} = G^{y_{opt}} = \left(\frac{G^2}{G-1}\right)\left(-1 + \sqrt{1 + G - \frac{1}{G}}\right) \quad \textbf{(4-22)}$$

The overall power consumption for the ADC and the resulting capacitor sizing are shown in Figs. 4-6, 4-7, 4-8, 4-9. It can be seen clearly that the optimal interstage gain and scaling exponent go up as the speed requirement is lowered. Next, we consider the case of high-resolution, high-speed converters, ignoring effects of parasitic capacitance at first.

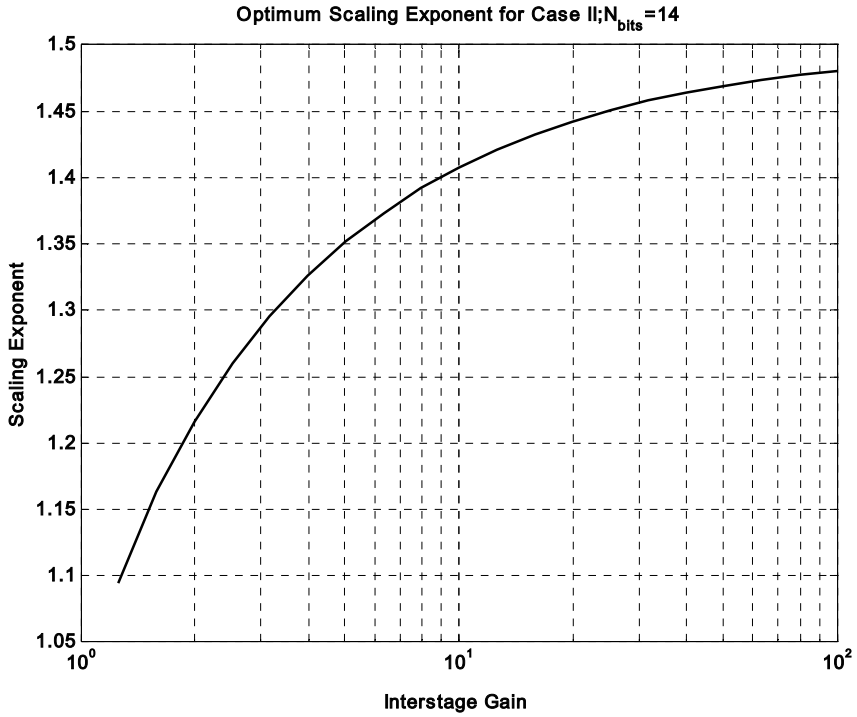


Figure 4-6 Optimal Scaling Factor Vs. G

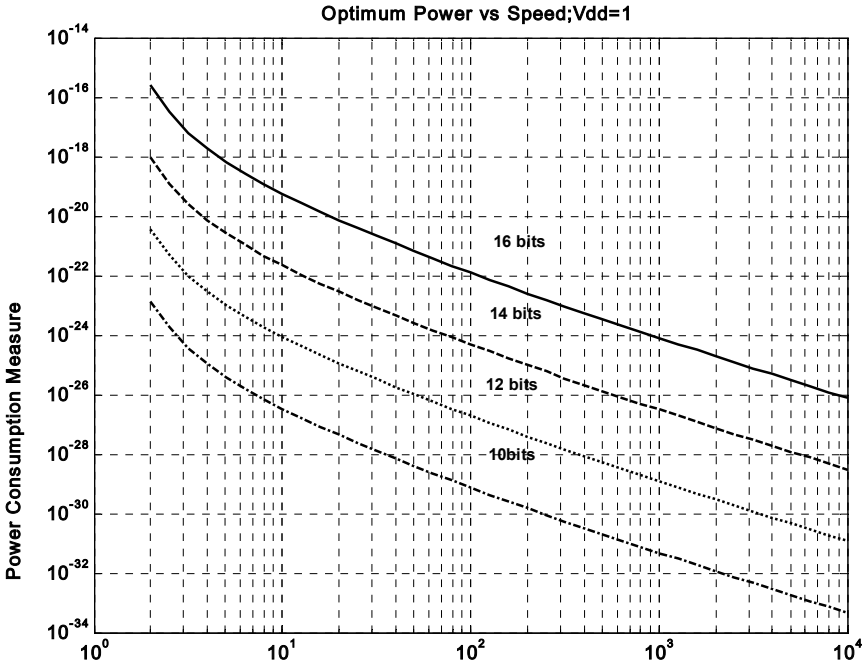


Figure 4-7 Power Consumption Vs. θ

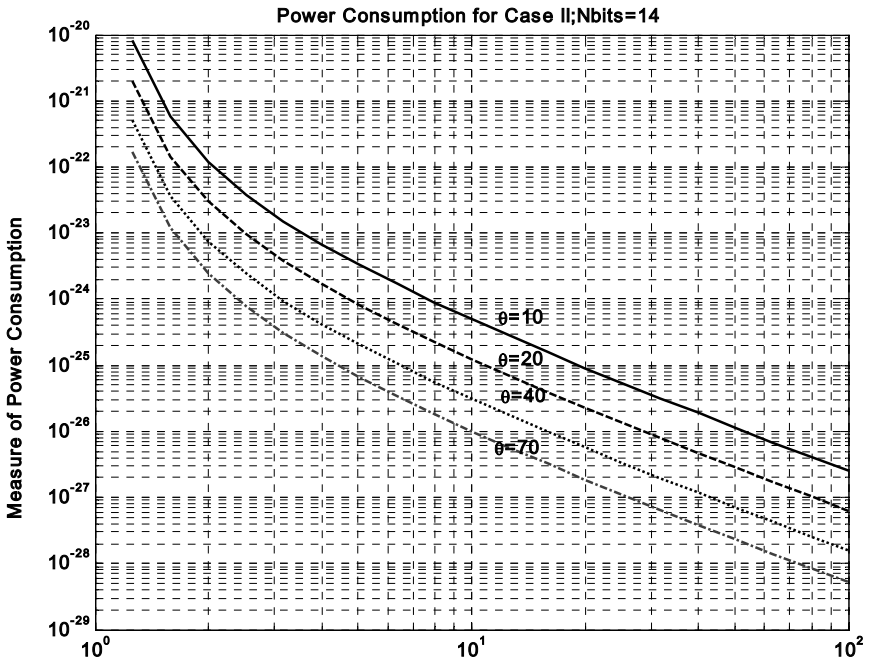


Figure 4-8 Power Consumption Vs. G

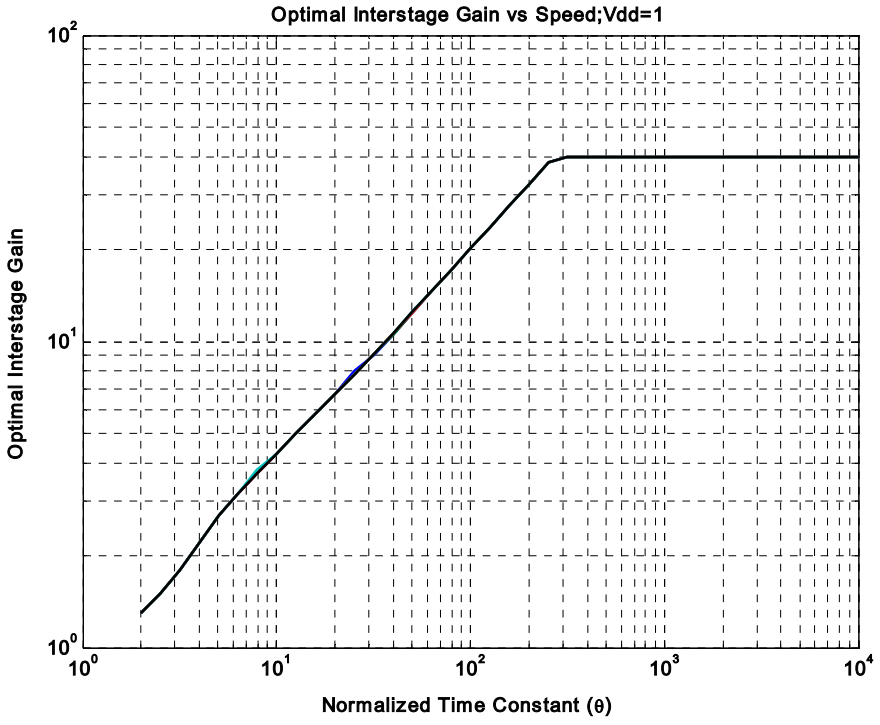


Figure 4-9 Optimal Interstage Gain Vs. θ

4.2.3 Case III: Optimal Interstage Gain In The Presence Of Noise For High-Resolution, High-Speed ADCs

In this section, the analysis is similar to the previous one except that the effect of C_O is now taken into account. The comparator input capacitance and the interconnect capacitance are still ignored. (4-6) is now written as:

$$g_{m,k} = \frac{C_{L,k} + C_{F,k}(1-f) + \alpha C_{GS,k}}{f\tau} \tag{4-23}$$

Rewriting (4-1) and (4-5), we have:

$$\begin{aligned} g_{m,k} &= \omega_T C_{GS,k} \\ f &= \frac{1}{G + \dfrac{C_{GS,k}}{C_{F,k}}} \end{aligned} \tag{4-24}$$

The above equations can be solved for the feedback factor to obtain the following expression:

$$f = \frac{\left[\theta + \alpha G - (G^{1-y} + 1)\right]}{2(\theta G - 1)}\left[1 + \sqrt{1 - \frac{4\alpha(\theta G - 1)}{\left[\theta + \alpha G - (G^{1-y} + 1)\right]^2}}\right] \tag{4-25}$$

The front-end feedback capacitance is now written in terms of the input-referred noise specification as:

$$C_{F,0} \approx \frac{2kT}{fG^2\overline{v_{in,diff,total}^2}}\left(1+\frac{2\beta}{3\left(1-f+G^{1-y}\right)\left(1+\frac{\alpha}{f\theta-\alpha}\right)}\right)\left(\frac{1}{1-G^{y-2}}\right) \qquad \textbf{(4-26)}$$

Since the current consumption is proportional to the total gate-source capacitance, we need to compute this quantity. This can be done by rearranging (4-24) to obtain:

$$C_{GS,k} = \frac{\left(G^{1-y}+1-f\right)C_{F,k}}{f\theta-\alpha} = \frac{\left(G^{1-y}+1-f\right)C_{F,0}G^{-ky}}{f\theta-\alpha} \qquad \textbf{(4-27)}$$

The total gate-source capacitance can be found by substituting (4-26) into (4-27) to obtain the following expression:

$$C_{GS,total} \approx \left(\frac{2kT}{fG^2\overline{v_{in,diff,total}^2}}\right)\frac{1}{\left(1-G^{y-2}\right)\left(1-G^{-y}\right)}\left\{\left(1+\frac{2\beta}{3\theta}\right)\frac{1}{f}-G\right\} \qquad \textbf{(4-28)}$$

The above expression must be minimized with respect to the scaling exponent y in order to find the optimal scaling factor. This exercise yields the following result:

$$s_{opt} = \left(\frac{G}{K_1}\right)\left\{\sqrt{K_1^2 + 1 + K_1\left(G + \frac{1}{G}\right)} - 1\right\}$$
$$K_1 = 1 - f + \frac{2\beta}{3\left(1 + \frac{\alpha}{f\theta - \alpha}\right)} \qquad \textbf{(4-29)}$$

This optimal value of the scaling exponent is then substituted back into the earlier equations to find the quantities of interest namely, the total gate-source capacitance, the optimal gain and the individual feedback capacitance values. Analytical calculations are omitted as the equations become overly cumbersome. Results from numerical calculations are shown in Figs. (4-10), (4-11), (4-12) and (4-13). The first three figures are plotted for various speeds, and thus, θ is a parameter for all these.

Once again, it can be seen that the optimal scaling exponent is higher for larger interstage gain, indicating the possibility of more aggressive scaling. The expected trends are displayed including improved power efficiency for lower speed converters with higher interstage gain. The optimal interstage gain is low for

high-speed requirements. As the power consumption is much higher at higher speeds, even minor changes in the interstage gain have a significant bearing on the overall power consumption. On several occasions, this optimal gain appears to be *non-binary*, e.g. 3,5,6,7,11 etc. Conventional methods provide no way for realization for such pipelined ADCs, and thus sub-optimal designs are still in vogue.

Additionally, Fig. (4-13) shows that the optimal gain remains the same for converters of varying resolution requirements. This is owing to our simplistic analysis where the comparator input capacitances are ignored. In more realistic cases analyzed in the next section, this plot will change markedly.

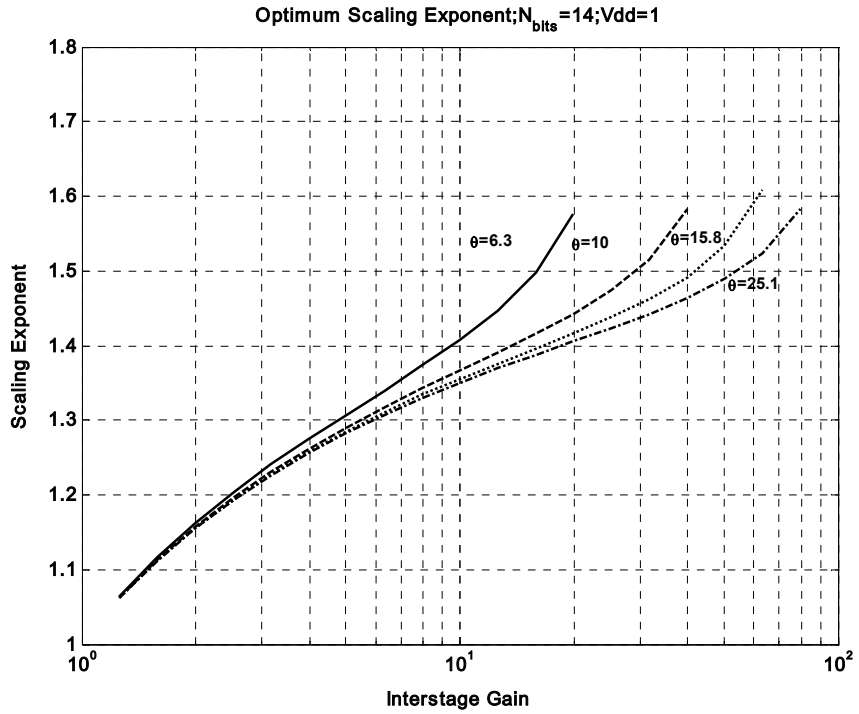


Figure 4-10 Scaling exponent Vs. G

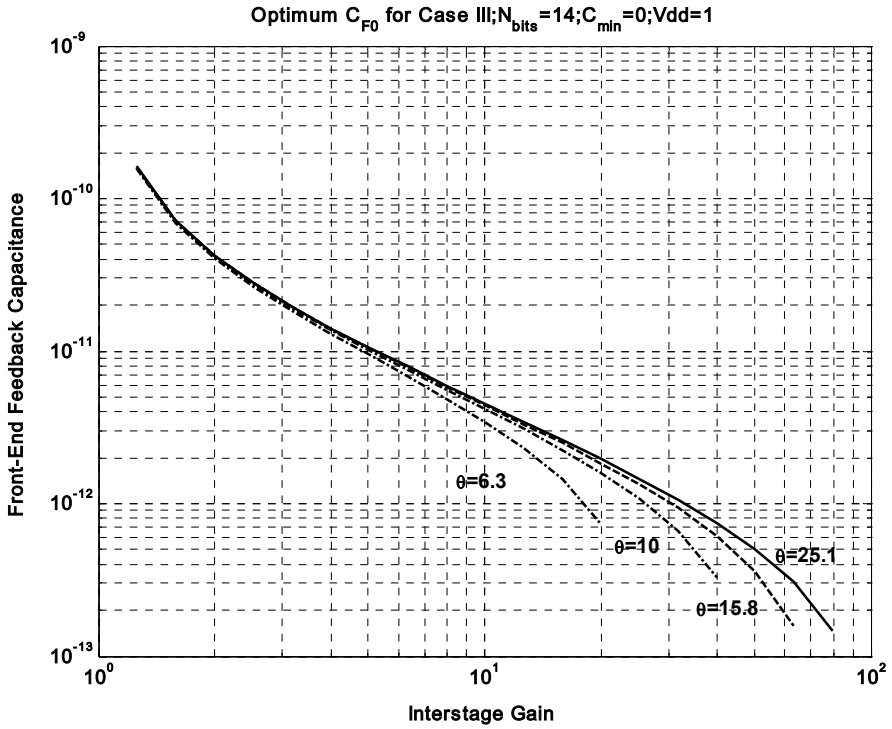


Figure 4-11 Front-end Feedback Capacitance Vs. G

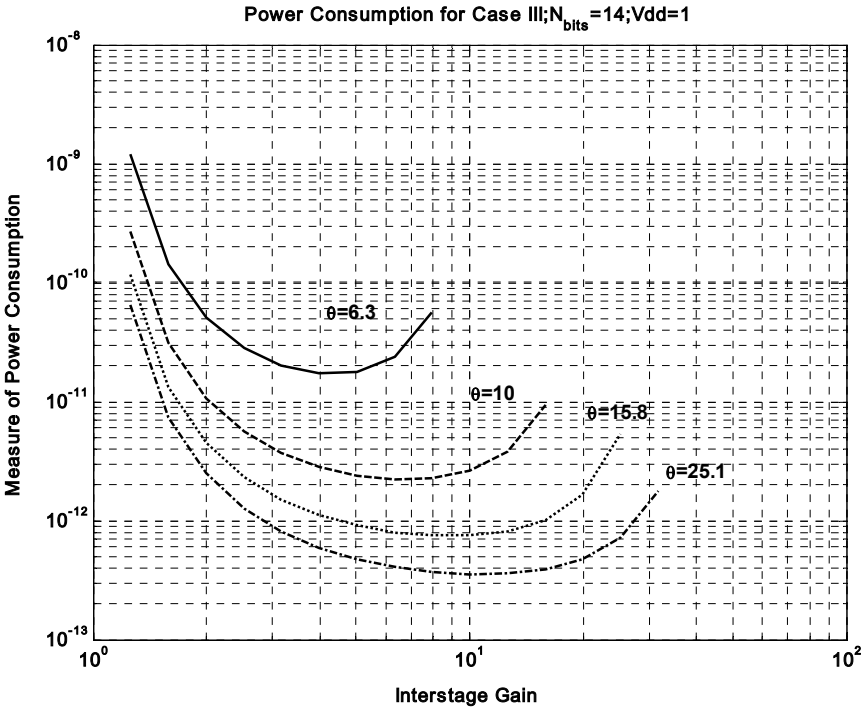


Figure 4-12 Power Consumption Vs. G

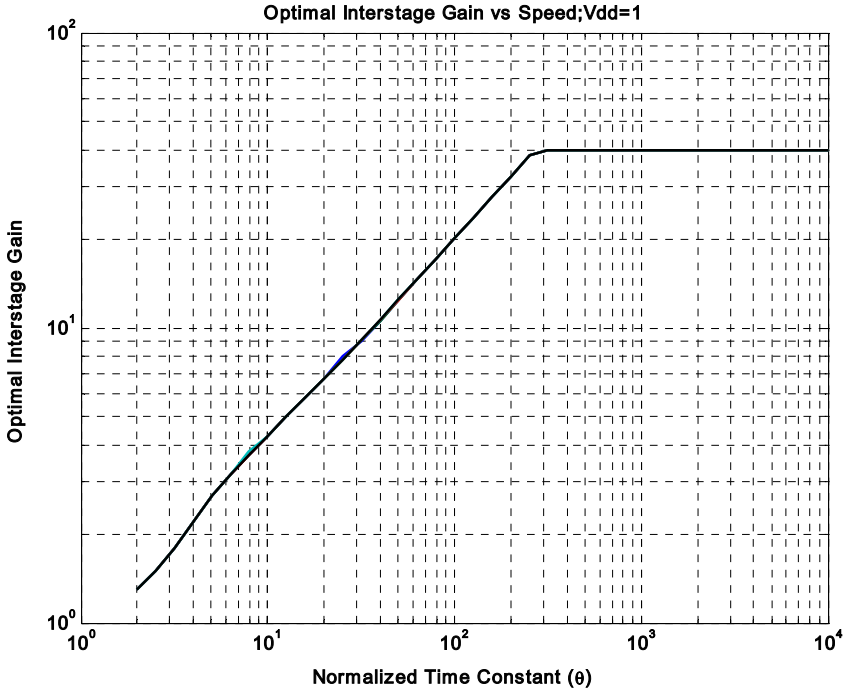


Figure 4-13 Optimal Interstage Gain Vs. θ

4.2.4 Case IV: Optimal Interstage Gain For High-Resolution, High-Speed ADCs With Parasitic Capacitances And Matching Constraints

The effect of comparator input capacitances (C_{comp}) and interconnect capacitances (C_{int}) was neglected so far in the analysis of high-resolution converters. This assumption is good to some extent as the capacitors are typically sized with a large value to meet thermal noise requirements. However, these parasitics do have an effect, especially in the later stages of the pipeline where the thermal noise constraints are eased, and the optimally sized capacitors become quite small with scaling. Here, the effect of these parasitics cannot be ignored anymore. Moreover, the optimal capacitor size may be lower than the minimum size capacitor (C_{min}) that can be reliably fabricated, as discussed earlier in Section 4.1.2.2, and depicted by Fig. (4-4), which shows the pipeline divided into two sections, with the front end limited by noise, and the rear end limited by matching and speed requirements. In reality, the demarcation is not so abrupt, and the scaling factor must ease off gently.

This analysis and the optimization are too complicated to be performed analytically, and have been done numerically. The results are shown in Figs. (4-14), (4-15), (4-16), (4-17). The expected trends are again confirmed in these plots.

The scaling exponent is high for higher resolution and lower speeds. The front-end feedback capacitance is more or less independent of speed and dependent only on overall resolution requirement.

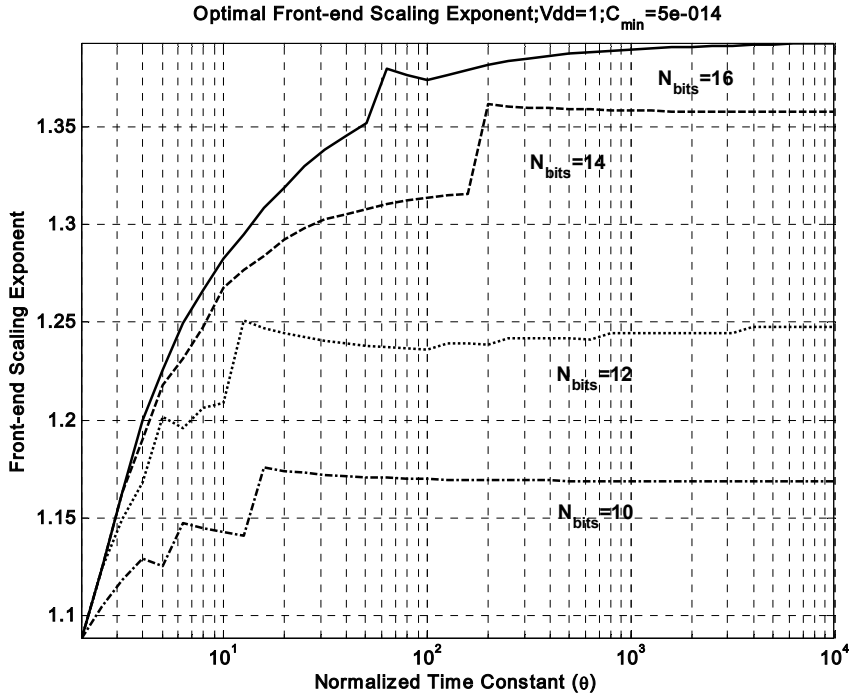


Figure 4-14 Scaling Exponent Vs. θ

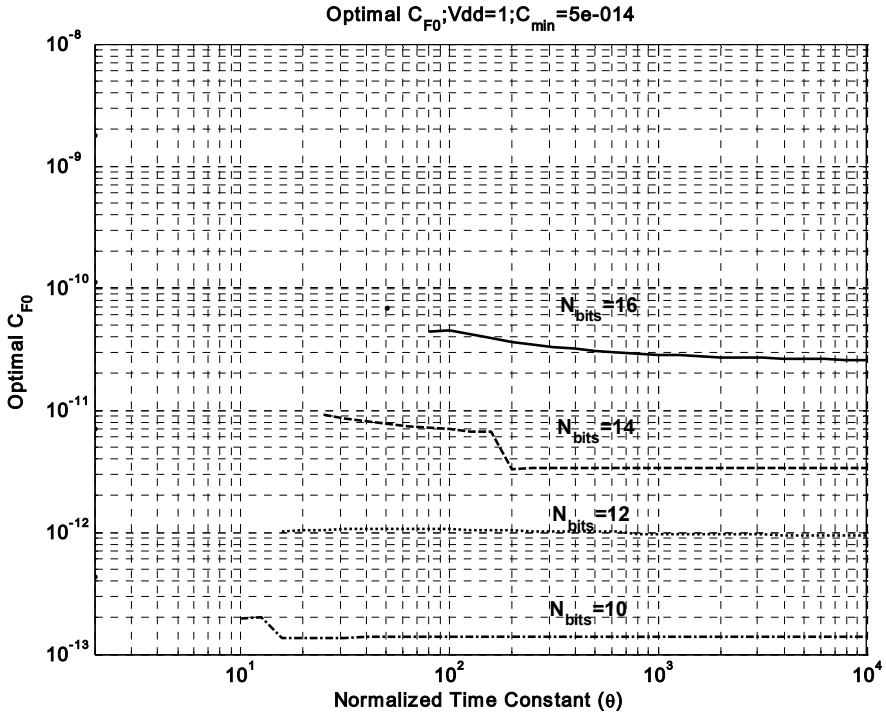


Figure 4-15 Front-end Feedback Capacitance Vs. θ

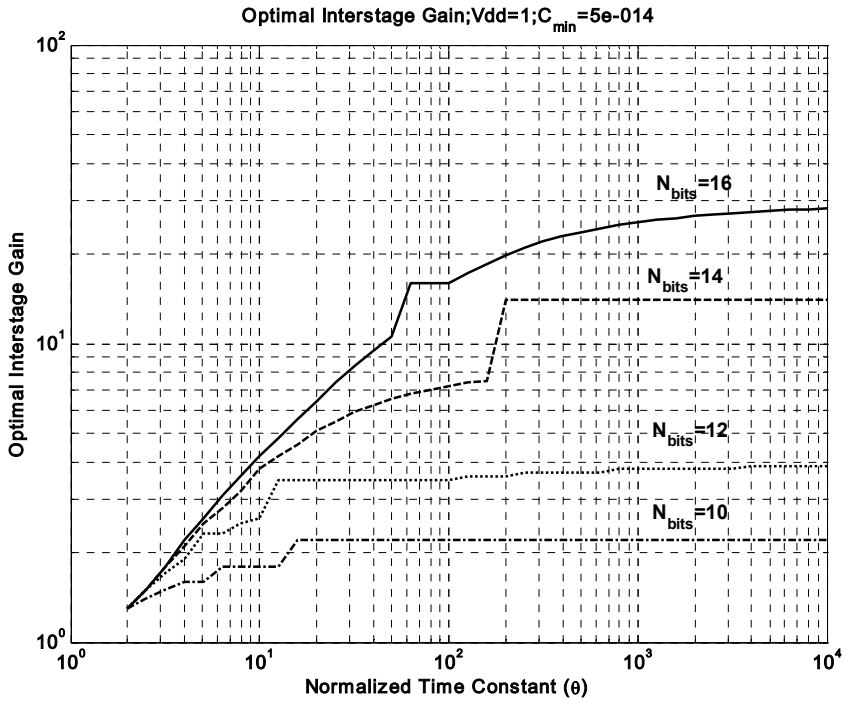


Figure 4-16 Optimal Interstage Gain Vs. θ

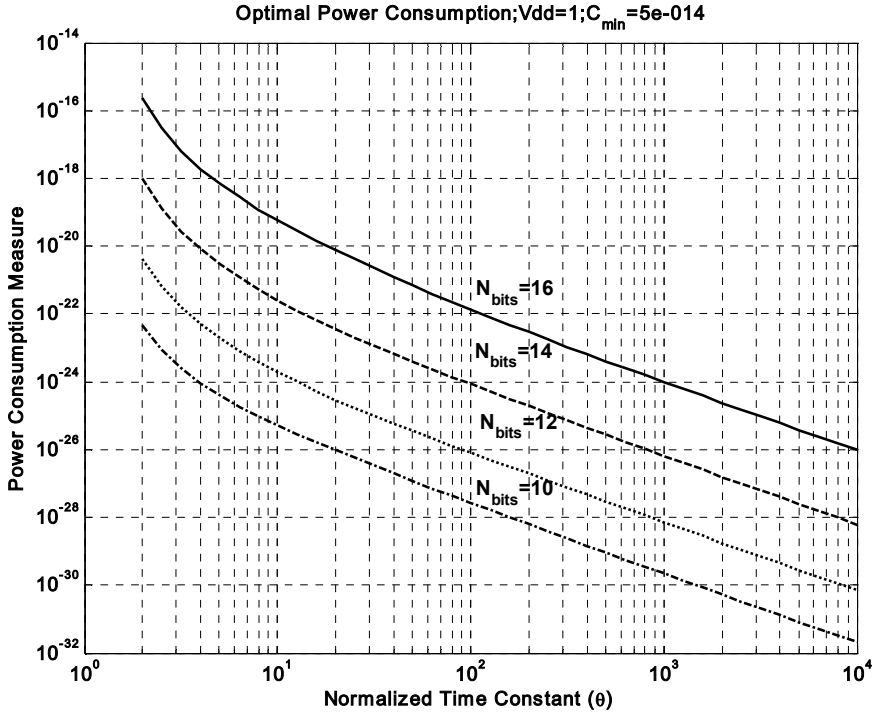


Figure 4-17 Optimized ADC Power Consumption Vs. θ

4.2.5 Case V: Optimizing Interstage Gain Vs. Supply Voltage

As discussed earlier in Section 4.2, the choice of supply voltage can significantly impact the performance of analog circuits, and affect the power consumption requirement. With the current migration towards reducing on-chip supply voltages, the power penalty for choosing a sub-optimal design goes up as the power consumption levels reach new highs, especially in the face of the high-speed, high-resolution requirements. Thus, an analysis was performed to estimate the optimal interstage gain and minimum power consumption for different supply voltages. The analysis was performed numerically, and the results are shown in Figs. (4-18), (4-19), (4-20), (4-21).

The expected trends are confirmed, as the optimal interstage gain rises sharply with shrinking supply voltages. Also, the overall power consumption rises sharply, and so does the power penalty for choosing a suboptimal design. As the optimal interstage gain rises, finding a binary gain close to this value becomes harder. For instance, if the optimal interstage gain is 12, then the closest neighbours are 8 and 16. Choosing any of the two will force additional tradeoffs, and lead to increased inefficiency.

So far, we have ignored the power consumed by the comparators as we expect these to consume no static power, and their overall power consumption to be significantly lower than those of the opamps, which have large static power consumption, owing to the constraints of thermal noise and speed. However, a small correction term may be added to account for this if required.

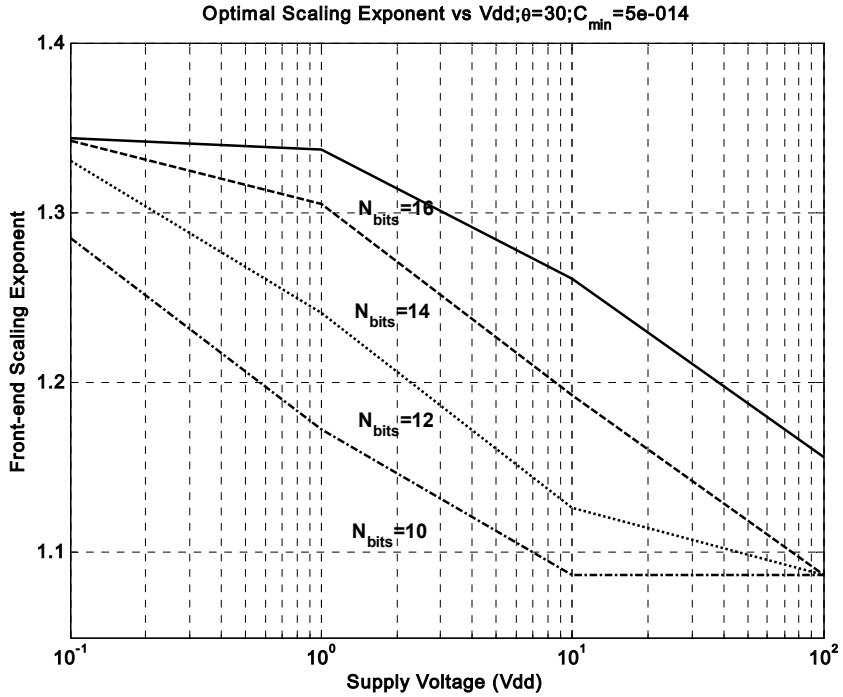


Figure 4-18 Optimal Scaling Exponent Vs. Supply Voltage

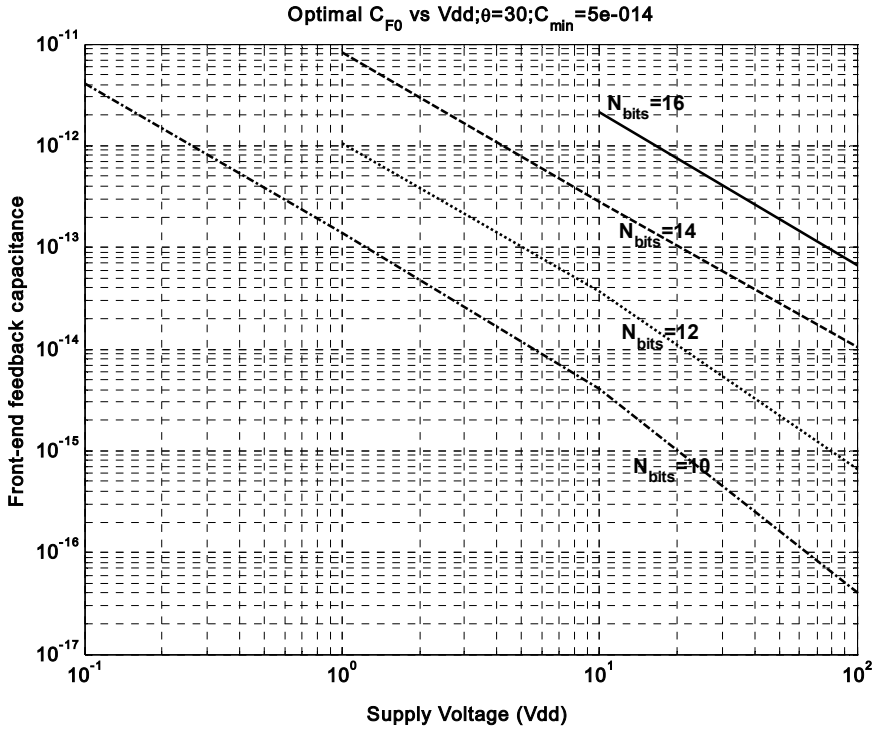


Figure 4-19 Front-end Feedback Capacitance Vs. Supply Voltage

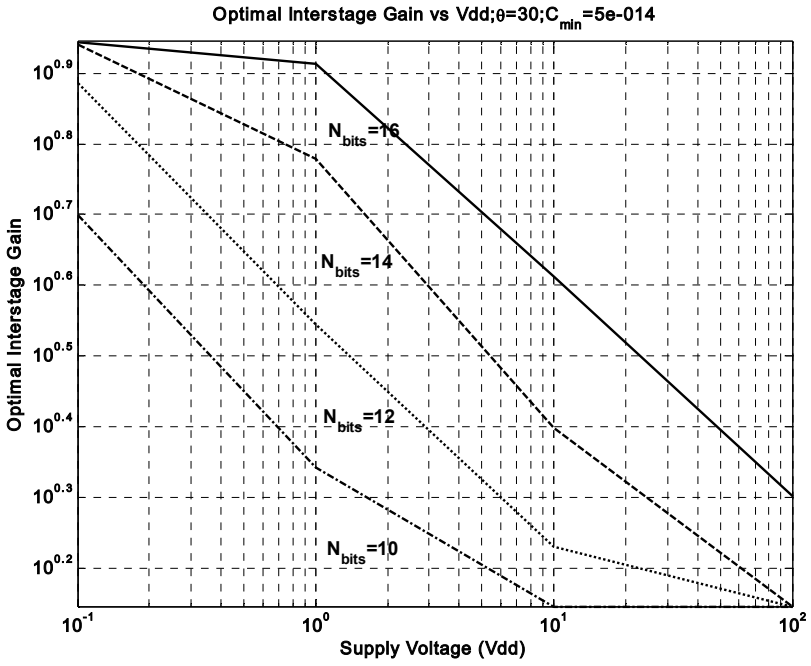


Figure 4-20 Optimal Interstage Gain Vs. Supply Voltage

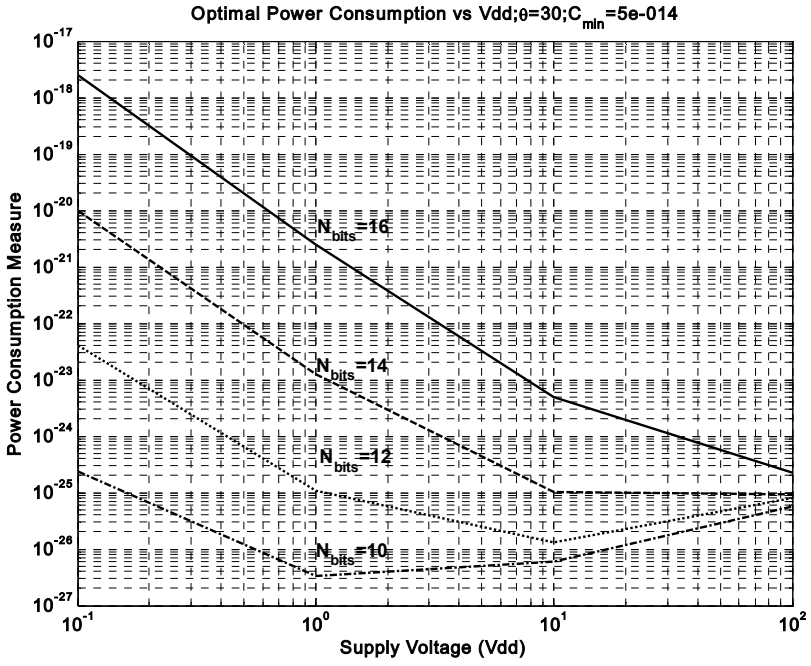


Figure 4-21 Optimized ADC Power Consumption

4.3 Conclusions

The various trends predicted from the qualitative analysis of Section 4.1 are found to hold good. These are summarized briefly here.

Shrinking power supply voltages increase the analog power consumption significantly, and place constraints on thermal noise performance of the ADC. Also, a higher interstage gain is required to achieve optimal performance in high-resolution converters. This interstage gain is usually non-binary in nature, and cannot be implemented using any of the existing methods.

Scaling and optimization are absolutely necessary owing to the high penalty associated with use of sub-optimal designs. Using identical stages is a bad idea, even though it saves design time as the final product will have very poor performance. The analysis presented in 4.2.3 can be easily adapted for the numerical analysis performed in 4.2.4 in order to automate the task of scaling.

When the speed requirement of the ADC approaches the technology limits, low interstage gains have to be used, and there is less room for scaling.

4.3.1 Further Improvements In Optimization

The models used above are fairly simple, owing to the difficulty associated with optimizing for more complicated cases. However, it still offers valuable insights into the task at hand. Further improvements can be made by fine tuning the model used above. For this purpose, the three scaling parameters α, β, γ may be

used as explained earlier in Section 4.1. These are particularly useful when the single-stage design does not provide sufficient gain to meet the requirements of high-resolution conversion, thereby necessitating the use of two-stage and three-stage opamps. However, the migration towards low supply voltages has rendered the design of most opamp topologies much harder. Therefore, it is advisable to try and utilise some system level features to tide over the lack of sufficient gain in single-stage opamps. One such technique will be discussed in Chapter 7.

More elaborate optimization may also be performed considering the effects of finite power consumption in comparators. However, these have been ignored for the time being. The current optimization scheme as used in Section IV and V attempts to divide the pipeline into two sections limited by thermal noise and matching requirements as described earlier. The parameters are interpolated between these two regions leading to some unevenness in the plots for optimal interstage gain and scaling exponent. Nonetheless, the overall power consumption metric shows a smooth behavior, hinting that the problem is not severe. However, this may be improved upon using more complicated interpolation techniques.

Finally, it might be necessary to alter some of the basic equations if the channel lengths used are too small. However, the optimization problem becomes nonintuitive and intractable for such cases and is not treated in this work.

5 Design Of Generalized Radix Pipeline Stages

The last chapter treated the subject of optimizing pipelined ADCs. Although the models used may not be applicable to all cases, they offer a few important insights. One of these is that the migration towards lower supply voltages and higher performance requirements might necessitate designing pipelined ADCs that have a non-binary interstage gain, or to put it in a slightly different fashion, the accuracy of each stage might have to be a fractional number of bits. While conventional design has proceeded along the lines of rounding off to the closest binary gain, such an approach is both unnecessary and wasteful.

With this in mind, this chapter presents a technique for realizing a pipeline stage with a generalized radix interstage gain that can assume any integer value. Although it is very easy to realize an ADC which resolves an arbitrary number of levels (Fig. 2-2 shows an ADC capable of resolving 9 levels), such methods do not incorporate any provision for digital redundancy and error correction, which is an indispensable requirement for realization of high-resolution pipelined ADCs.

The optimization in the previous chapter assumes a 1 bit redundancy in all the pipeline stages. Typically, this is sufficient for practical applications, and thus, the proposed approach will be developed with 1 bit redundancy. Should larger amounts of digital redundancy be required, these can be achieved by following guidelines similar to those shown in the development of the proposed technique.

For the sake of simplicity, this approach will be developed for the simplest case where the interstage gain is 3. One of the optimization examples results in an optimal gain of 3 (Fig. 4-12 reproduced here as Fig. 5-1 shows this). Conventional binary logic gates can be used for implementation of the digital circuitry, thereby saving the trouble of implementing a multiple-level logic system. The validity of this approach and the proposed digital correction technique will be demonstrated through worst-case simulations performed in MATLAB. Finally, it will be shown that the proposed technique does not entail any extra overhead in terms of design complexity or cost, and blends seamlessly with design methodologies for binary cases, which are seen as specific cases of this more generic approach.

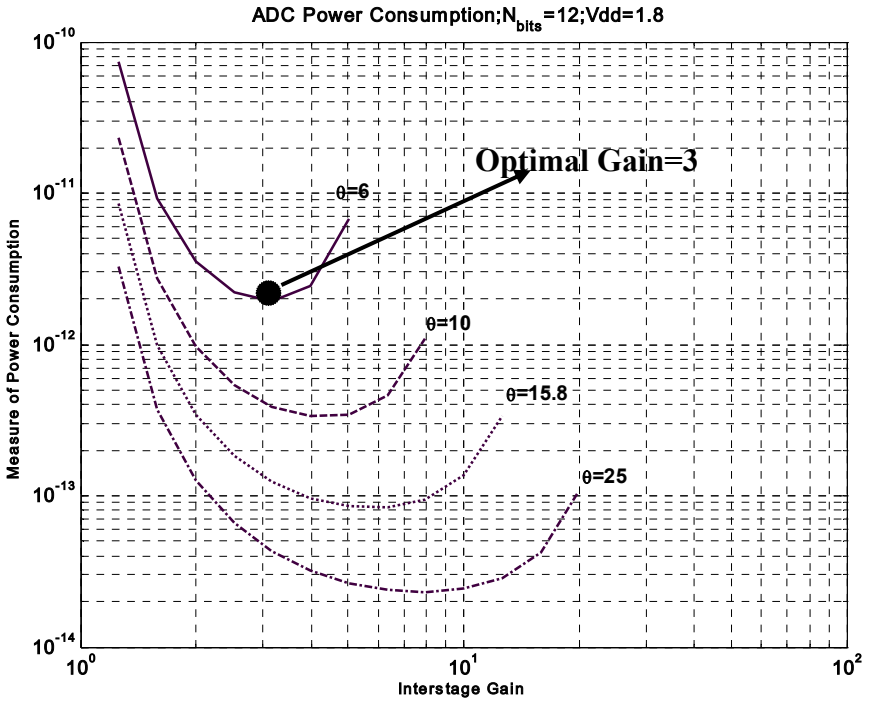


Figure 5-1 Example showing an optimal gain of 3

5.1 Design Methodology For A Generalized Radix Stage

The goal is to design a pipelined ADC stage with interstage gain of 3 and digital redundancy of 1 bit. 1 bit corresponds to a gain of 2^1=2. Therefore, we must start with a stage that resolves 3 x 2 = 6 levels, and halve the gain to get the interstage gain of 3. The residue transfer curve of the original stage is shown in Fig. 5-2. This has an interstage gain of 6 and no digital redundancy. The number of comparators is one less than the number of levels resolved (6-1 = 5 comparators). Any comparator offset here will cause the residue to go out of range leading to missing levels in the overall converter transfer characteristic.

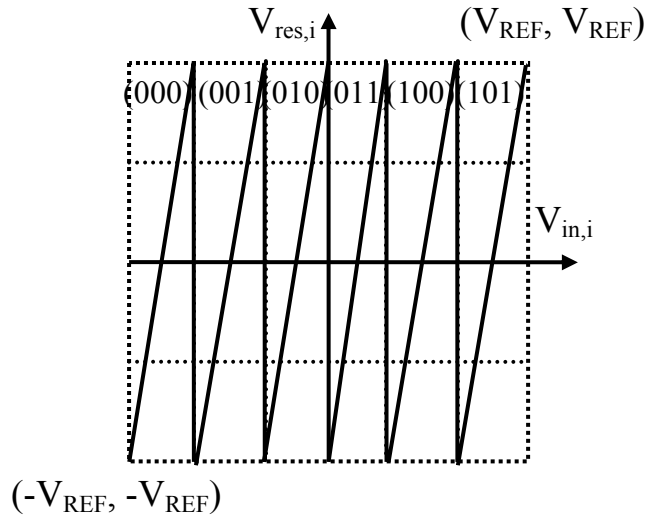


Figure 5-2 Residue transfer curve for 6-level ADC stage

Now, the gain is halved to incorporate the desired 1 bit redundancy. The modified residue transfer curve is shown in Fig. 5-3. With this, the residue output of the stage is ideally bounded between -0.5 V_{REF} and 0.5 V_{REF}. Thus, any comparator offset less than 0.5 V_{REF}/(2x3) in magnitude can be tolerated.

Arguments similar to those presented in our earlier discussion on digital redundancy (Section 2.2.1) can be used to show that the digital error correction remains similar and we need to add 1 LSB to the output code to correct for negative overrange errors, and subtract 1 LSB to correct for positive overrange errors. For reasons explored earlier, this is undesirable, and we would like to just perform one of these operations.

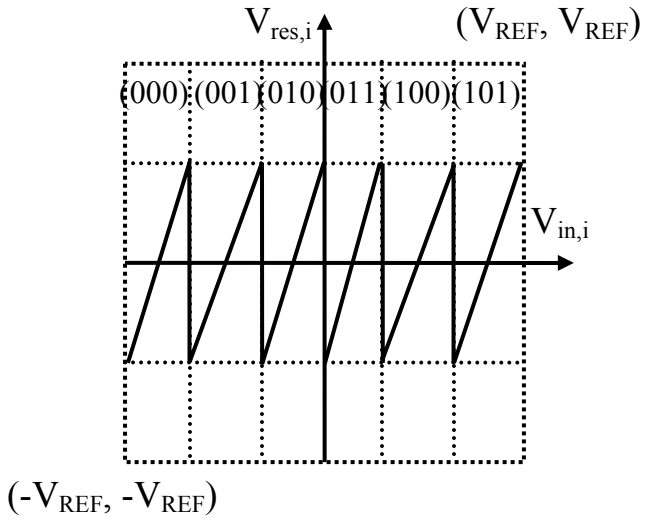


Figure 5-3 6-level ADC stage with 1 bit of digital redundancy

With this aim, we artificially introduce an offset of 0.5 V_{LSB} in the signal path to shift the residue transfer curve (Fig. 5-4) so that it resembles the one shown in Fig. 2-8, and again the last comparator is redundant and can be discarded. Now, the number of comparators is (6-2 = 4 comparators). The resulting residue transfer curve is shown in Fig. 5-5. Thus, digital redundancy has been introduced in the *ternary* (gain=3) stage. It can be similarly proved by construction that for an arbitrary interstage gain G, the number of comparators required to implement 1 bit of digital redundancy will be (2G-2), and the maximum comparator offset that can

be tolerated is $V_{REF}/(2G)$. It can be seen that this number would correspond to $2(2^n-1)$ comparators for the case of *binary* interstage gain, and thus this scheme proves to be a generic one of which, only a few specific instances have been realized so far. Let us now address the issue of digital error correction to see if a scheme similar to the *overlap & add* scheme of Chapter 2 can be developed.

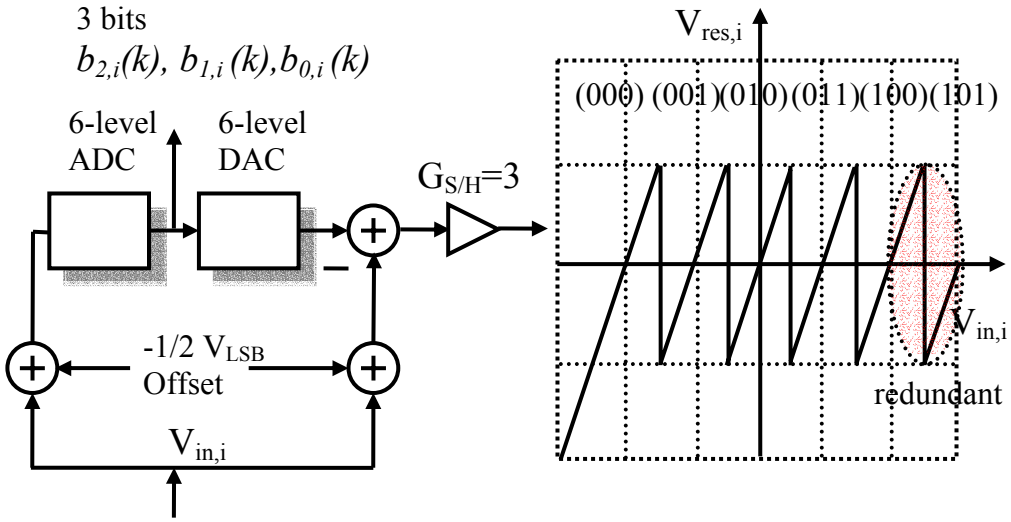


Figure 5-4 Modified 6-level ADC stage with 1-bit redundancy

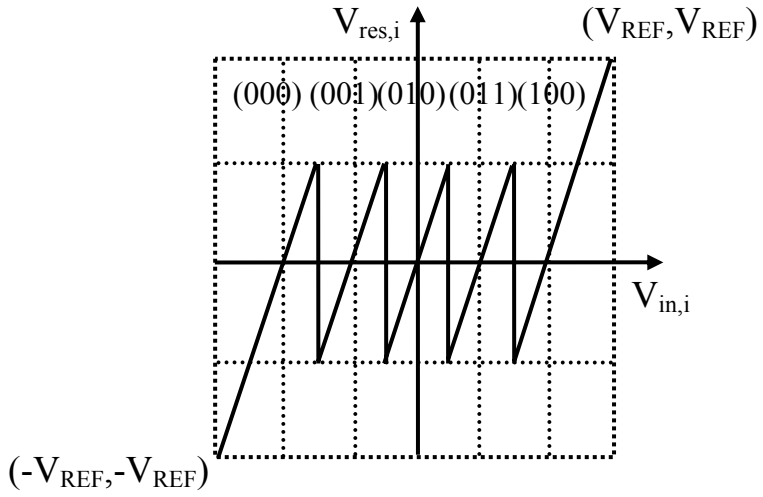


Figure 5-5 Residue transfer curve for pipeline stage with G=3

5.2 Digital Error Correction

Towards the end of Section 2.2.1, the digital error correction technique for the 1.5-bit stage was demonstrated using the *overlap & add* scheme, and in comparing it with the native scheme lacking digital redundancy, we had observed that the digital error correction merely accounted for the *analog interstage gain* in the digital domain, and the bit outputs were added after accounting for this gain. For the sake of clarity, the scheme shown in (2-3) and (2-4) is shown again in (5-1) and (5-2).

This suggests that if the digital outputs were to be added with the effect of the interstage gain of 3 accounted for, then the desired results would be obtained (5-3). This is indeed the case and the interested reader may verify this by performing the detailed calculations. These are omitted here. However, the technique is used to simulate the performance of various generalized radix stages, and the results will be presented shortly.

To illustrate the modified overlap & add technique, an example is presented in Section 5.2.1 showing digital correction in a two-stage pipelined ADC for different interstage gains. One can see that there are many ways of doing this and two possible methods are shown to realize the interstage gain of 3. Both of these are equally effective and either one may be used.

$$\begin{array}{cccccccc} b_{1,1} & b_{0,1} & & & & & & \\ & b_{1,2} & b_{0,2} & & & & & \\ & & \ldots & \ldots & & & & \\ & & & & b_{1,m} & b_{0,m} & & \\ \hline B_1 & B_2 & \ldots\ldots & & B_m & B_{m+1} & & \end{array}; \textit{Addition of bit outputs}, G = 2 \quad \textbf{(5-1)}$$

$$\begin{array}{cccccccccc} b_{1,1} & b_{0,1} & & & & & & & & \\ & & b_{1,2} & b_{0,2} & & & & & & \\ & & & & \ldots & & & & & \\ & & & & & & b_{1,m} & b_{0,m} & & \\ \hline b_{1,1} & b_{0,1} & b_{1,2} & b_{0,2} & \ldots & & b_{1,m} & b_{0,m} & & \end{array}; \textit{Addition of bit outputs}, G = 4 \quad \textbf{(5-2)}$$

$$\begin{array}{ccccccccc} b_{2,1} & b_{1,1} & b_{0,1} & & & & & & \\ & b_{2,1} & b_{1,1} & b_{0,1} & & & & & \\ & b_{2,2} & b_{1,2} & b_{0,2} & & & & & \\ & & b_{2,2} & b_{1,2} & b_{0,2} & & & & \\ & & . & . & . & & & & \\ & & & b_{2,m-1} & b_{1,m-1} & b_{0,m-1} & & & \\ & & & & b_{2,m} & b_{1,m} & b_{0,m} & & \\ \hline B_0 & B_1 & \ldots\ldots & \ldots\ldots & \ldots\ldots & & B_{m-1} & B_m & \end{array}; \textit{Addition of bit outputs}, G = 3 \quad \textbf{(5-3)}$$

5.2.1 Overlap & Add Example

Consider a 2-stage pipelined ADC with the output words from the first and second stage denoted by D_1 and D_2 respectively. Four cases are shown:

$$\begin{array}{l} D_1 = 01, D_2 = 10 \\ \quad 0\ \ 1 \\ +\quad\ \ 1\ \ 0 \\ \hline \quad 0\ \ 1\ \ 1\ \ 0 \end{array} ; Case(i) - G = 4, no\ redundancy \qquad \textbf{(5-4)}$$

$$\begin{array}{l} D_1 = 01, D_2 = 10 \\ \quad\ \ 0\ \ 1 \\ +\quad 1\ \ 0 \\ \hline \quad 1\ \ 0\ \ 0 \end{array} ; Case(ii) - G = 2, 1 - bit\ redundancy \qquad \textbf{(5-5)}$$

$$\begin{array}{l} D_1 = 011, D_2 = 100 \\ \quad 0\ \ 1\ \ 1 \quad 3.\mathrm{D}_1 + \mathrm{D}_2 \\ \quad\ \ 0\ \ 1\ \ 1 \\ +\quad 1\ \ 0\ \ 0 \\ \hline \quad 1\ \ 1\ \ 0\ \ 1 \end{array} ; Case(iii) - G = 3 = (2+1), 1 - bit\ redundancy \qquad \textbf{(5-6)}$$

$$\begin{array}{l} D_1 = 011, D_2 = 100 \\ \quad 0\ \ 1\ \ 1 \\ -\quad\ \ 0\ \ 1\ \ 1 \\ +\quad 1\ \ 0\ \ 0 \\ \hline \quad 1\ \ 1\ \ 0\ \ 1 \end{array} ; Case(iv) - G = 3 = (4-1), 1 - bit\ redundancy \qquad \textbf{(5-7)}$$

5.2.2 Implementation Details

Let us consider a generalized radix pipelined ADC with an arbitrary interstage gain G. An implementation is shown in Fig. 5-6. This can be viewed as a combination of an analog pipeline and a digital pipeline.

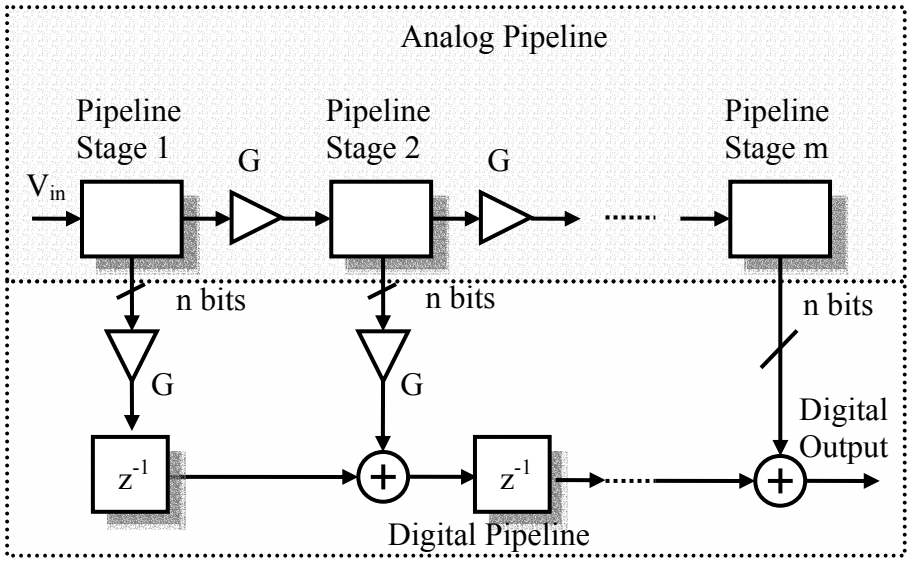


Figure 5-6 Pipelined ADC System Block Diagram

The design of the analog blocks remains unchanged. The design of comparators and the opamp is unaffected, while the only change in the S&H amplifier (Fig. 4-2) is in the ratio of capacitors, which are sized to reflect the interstage gain required.

On the digital side, the thermometer-binary encoder remains unchanged. The only difference is that the interstage gain is now non-binary, and hence cannot be realized just by bit-shifts alone. For the case of G=3, we need to perform 2 different bit-shifts, and add/subtract these before passing them on. These are used

even in the design of conventional pipelined ADCs, where an interstage gain of 4 or 8 may be realized by shifting a word to the left by 2 or 3 bits respectively. Table 5-1 shows various ways to implement values of gain from 2 till 8. Performing subtraction does not complicate things much as the digital output from each stage is often encoded in 2s-complement format.

Table 5-1 Example implementations of various digital gain values

Gain	Implementation
2	x2
3	x2 + x1, x4 – x1
4	x4
5	x4 + x1
6	x4 + x2, x8 – x2
7	x4 + x2 + x1, x8 –x1
8	x8

This increase in hardware is moderate and the savings in power resulting from the use of the optimal interstage gain offer more than enough benefits to compensate for this.

5.3 Further Examples & Simulation Results

The design methodology illustrated in the earlier sections was used for system-level design of several different generalized radix pipeline stages, for different overall converter resolutions. These were then simulated under worst-case conditions of comparator offset and the effectiveness of the proposed digital redundancy and error correction techniques was tested by measuring the non-linear distortion resulting from these offsets. The design methodology was found to work perfectly in line with expectations and the overall accuracy of the converters was seen to meet the required specification each time.

Thus, the design methodology is verified to be correct. Owing to the complex nature of the problem and the need for a large number of simulations, the task was performed in MATLAB using behavioral simulations. Some results are shown in the following figures. First, a few examples of residue transfer curves are shown for different generalized radix stages, along with the worst case residue transfer curves when the comparator offsets are at their maximum value.

Then, the simulation results demonstrate that the worst-case INL and DNL for each of these cases is within stipulated limits after digital error correction.

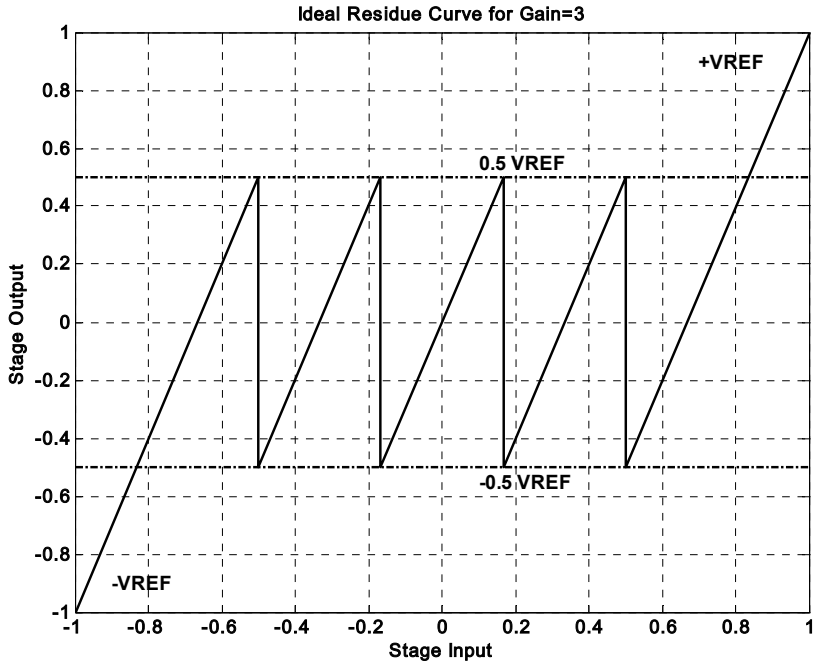


Figure 5-7 Ideal Residue Curve for G=3

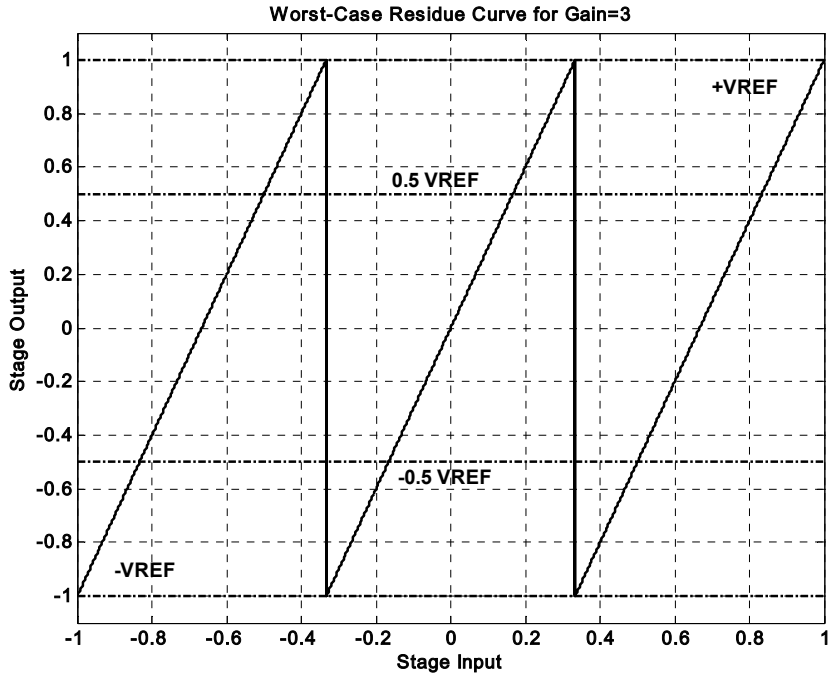


Figure 5-8 Worst Case Residue Curve for G=3

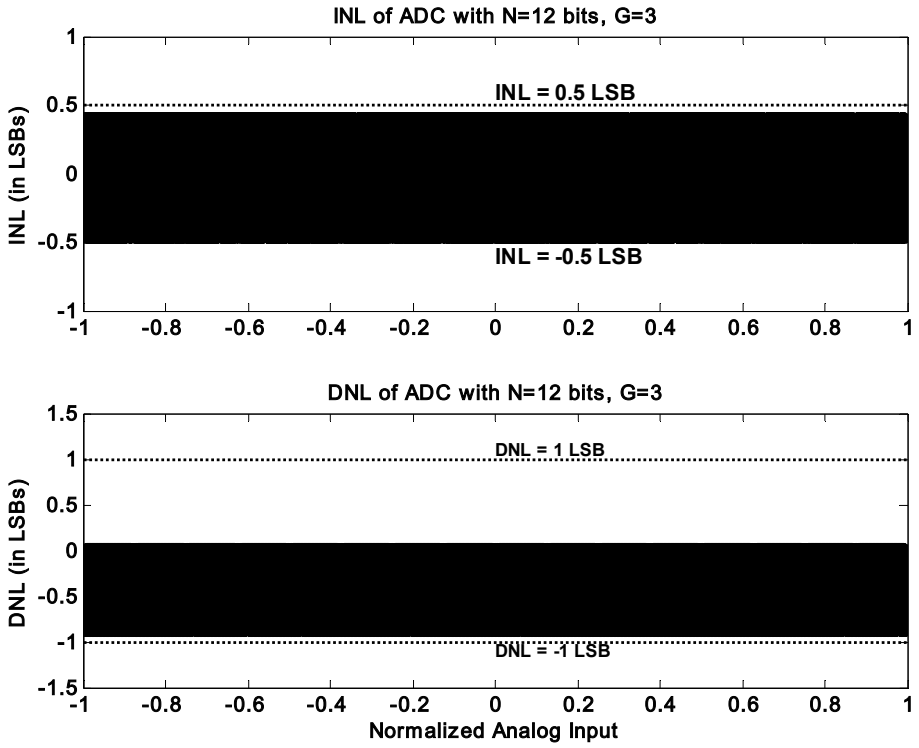


Figure 5-9 INL & DNL for G=3 after digital correction

The integral nonlinearity error (INL) must remain within +/- 0.5 LSB for the converter accuracy to be equal to the desired value. Alternatively, the differential nonlinearity error (DNL) must remain within +/- 1 LSB [29]. This requirement is clearly satisfied in case of the proposed technique. Similar results are presented for the case of a few more generalized radix stages. As can be seen from 5-7 and 5-8, the worst-case scenario is approached when two adjacent comparator thresholds overlap leading to a residue curve, which is similar to the original residue curve without any digital redundancy. Any extra error beyond this point cannot be corrected digitally.

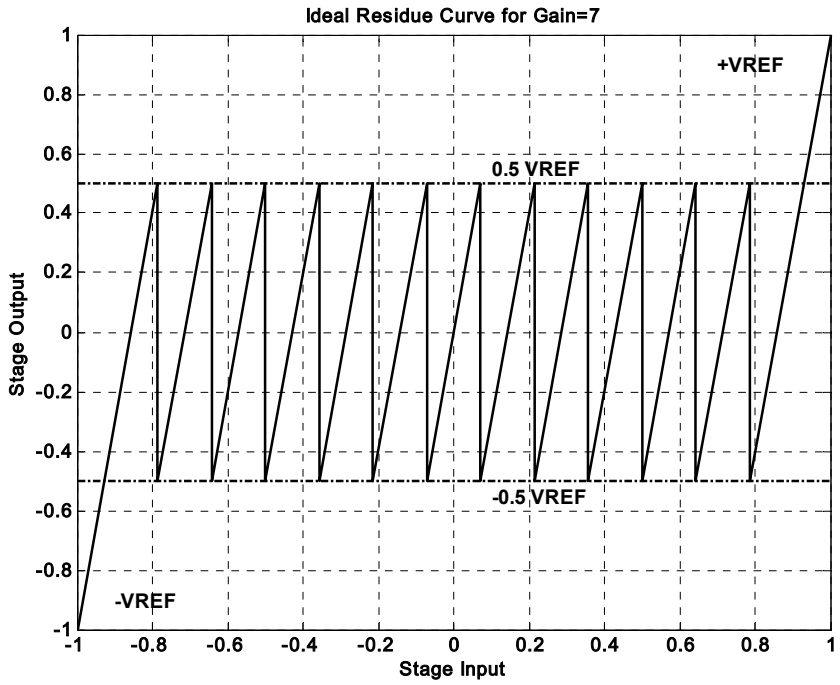


Figure 5-10 Ideal Residue Curve for G=7

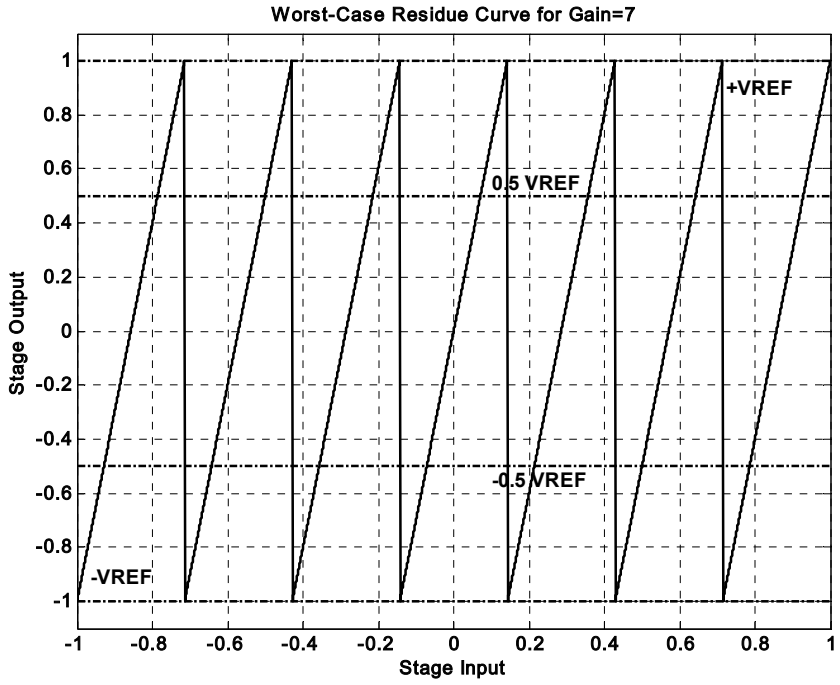


Figure 5-11 Worst Case Residue Curve for G=7

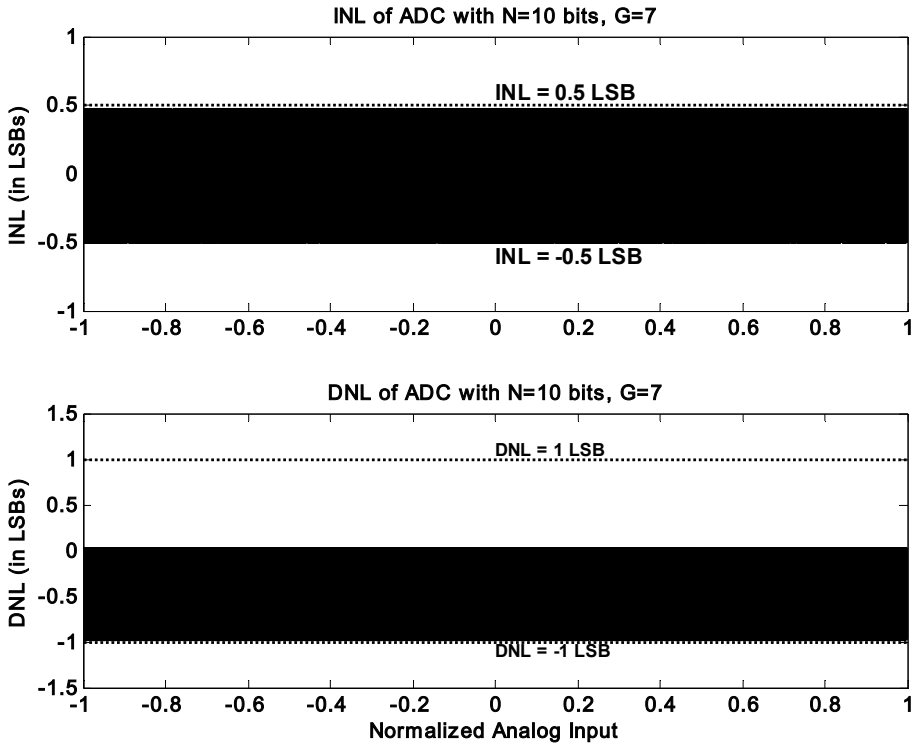


Figure 5-12 Worst Case INL & DNL for G=7 after digital correction

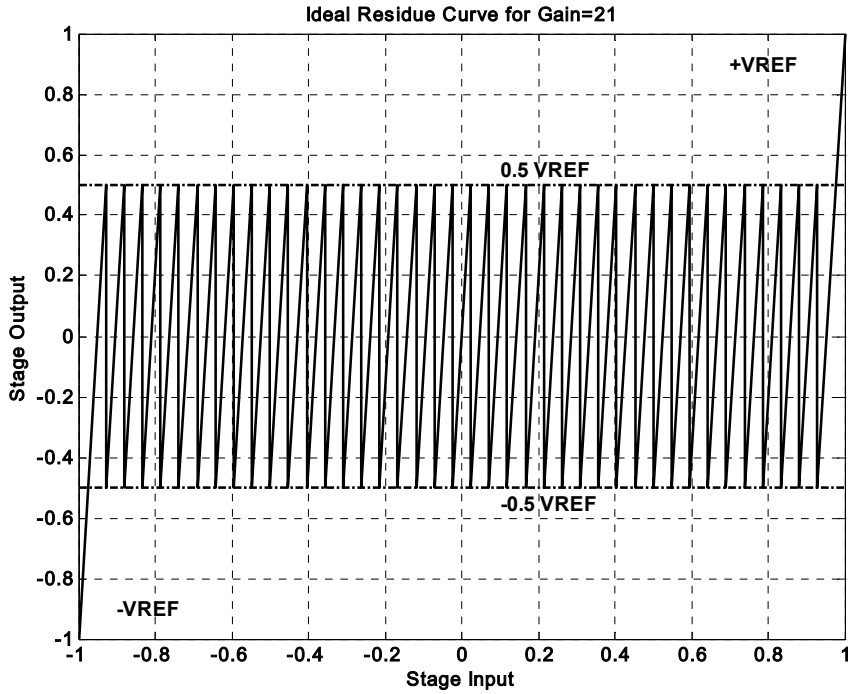


Figure 5-13 Ideal Residue Curve for G=21

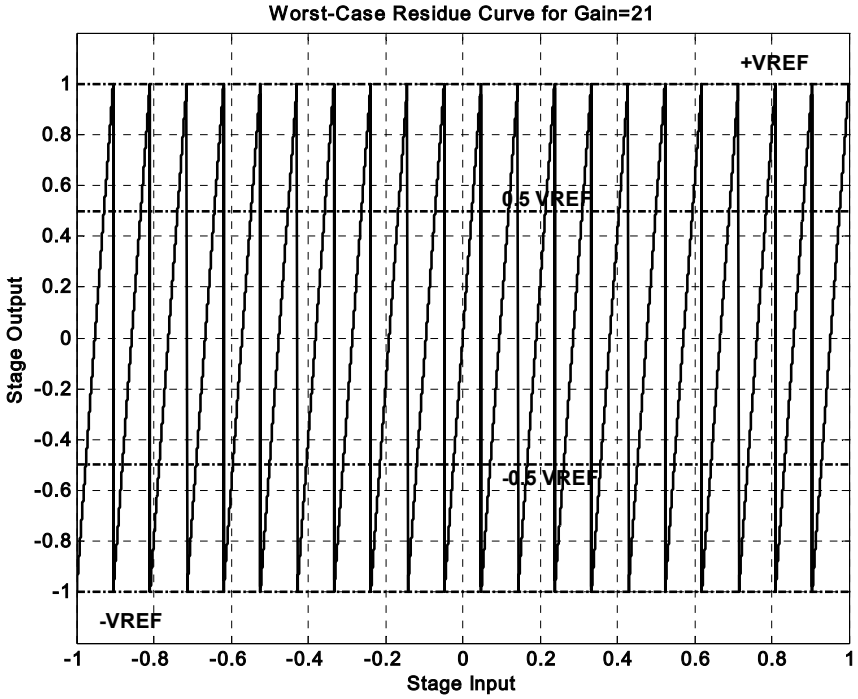


Figure 5-14 Worst Case Residue Curve for G=21

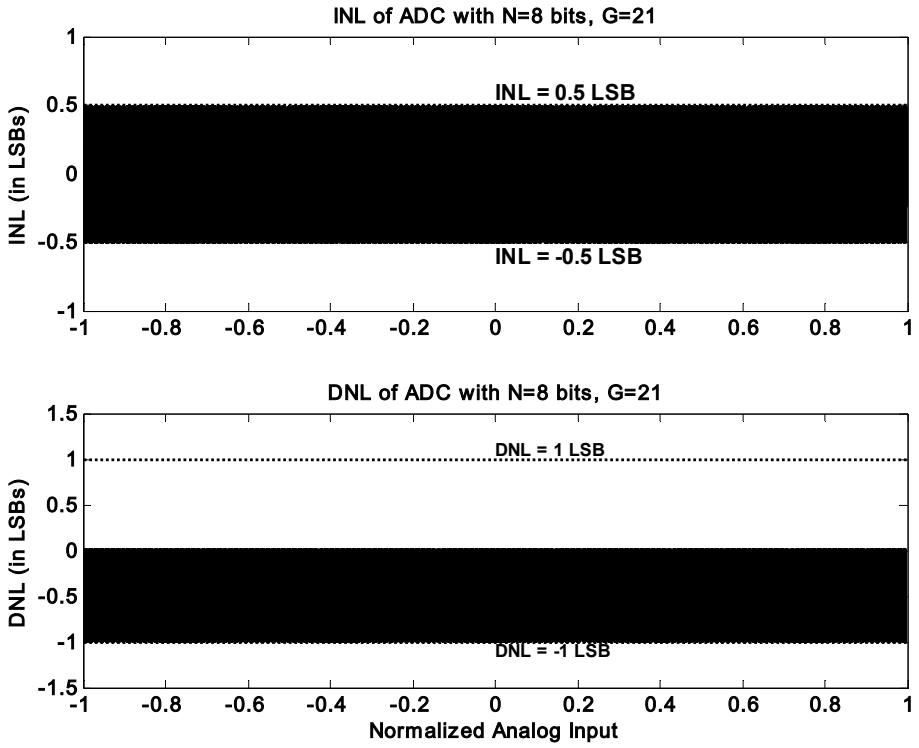


Figure 5-15 Worst Case INL & DNL for G=21 after digital correction

5.4 Relevance Of Generalized Radix Designs

This section presents a few examples showing the relevance of generalized radix stage designs. The chief aim is to highlight scenarios where the generalized radix schemes offer significant advantages over the conventional binary schemes. This is helpful in deciding whether or not such a scheme needs to be used for a given set of constraints on the performance of the ADC.

The performance of a 12-bit ADC is optimized here for three different operating speeds with the supply voltage assumed to be 1.8V. All stages are assumed to resolve an identical number of bits in this analysis. In many

commercial implementations, this is not valid as different stages typically resolve different number of bits. Typically, the later stages are designed to resolve fewer bits than the first stage. This is consistent with the arguments presented earlier as the required resolution is lower in the later stages and a higher interstage gain is less attractive. However, such cases are not considered here. As expected from the analysis shown in Chapter 4, the optimal interstage gain should increase with a reduction in the speed requirements. Additionally, the penalty incurred due to choice of a non-optimal interstage gain will also decrease when the speed requirement is lower. The three cases are chosen so as to yield a non-binary value for the optimal interstage gain. The results are shown in Figs. 5-16, 5-17, 5-18.

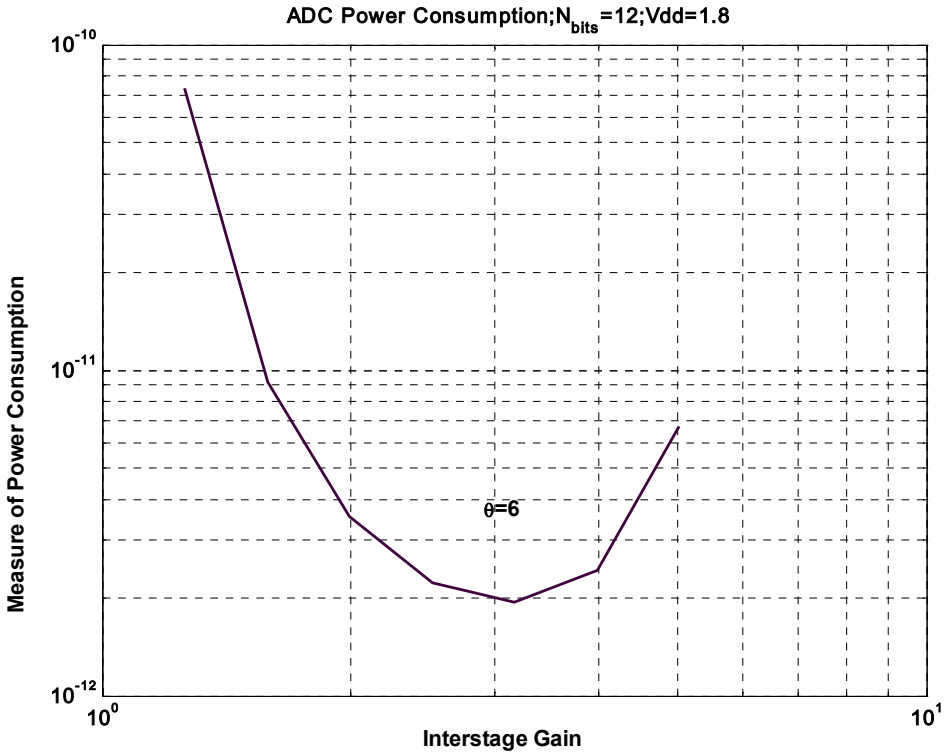


Figure 5-16 Optimization Example 1: G_{opt}=3

The optimization example presented above shows an optimal interstage gain of 3. The penalty incurred upon choosing an interstage gain of 2 or 4 for this design would be 80% and 30% respectively. If we assume a process f_T of about 2.4GHz, then the above example would correspond to an ADC running at ~50Msps. Typical power consumption of such an ADC would be in the range of a few hundred milliwatt and a 30% reduction in power is quite significant. Thus, choosing the non-binary interstage gain is highly recommended for such a case.

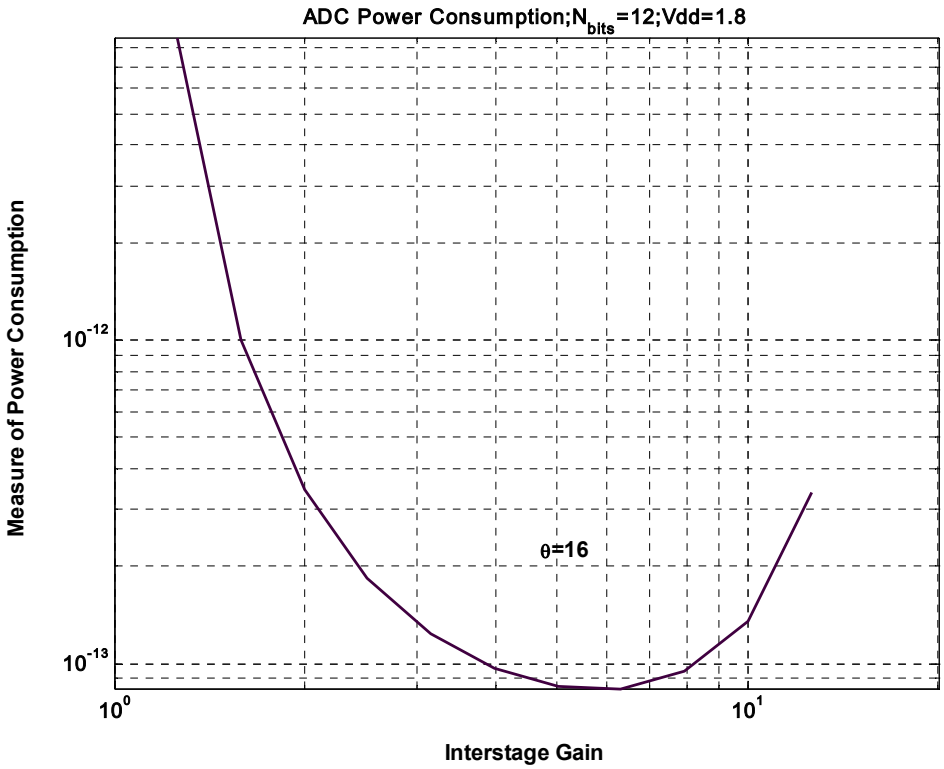


Figure 5-17 Optimization Example 2: G_{opt}=6

The above example shows an optimal interstage gain of 6 with a penalty of about 20% if a gain of 4 or 8 is used instead of the optimal gain. The above example would correspond to an operating speed of ~20Msps. Since the savings in

power are significant, the use of the non-binary interstage gain is advisable even here.

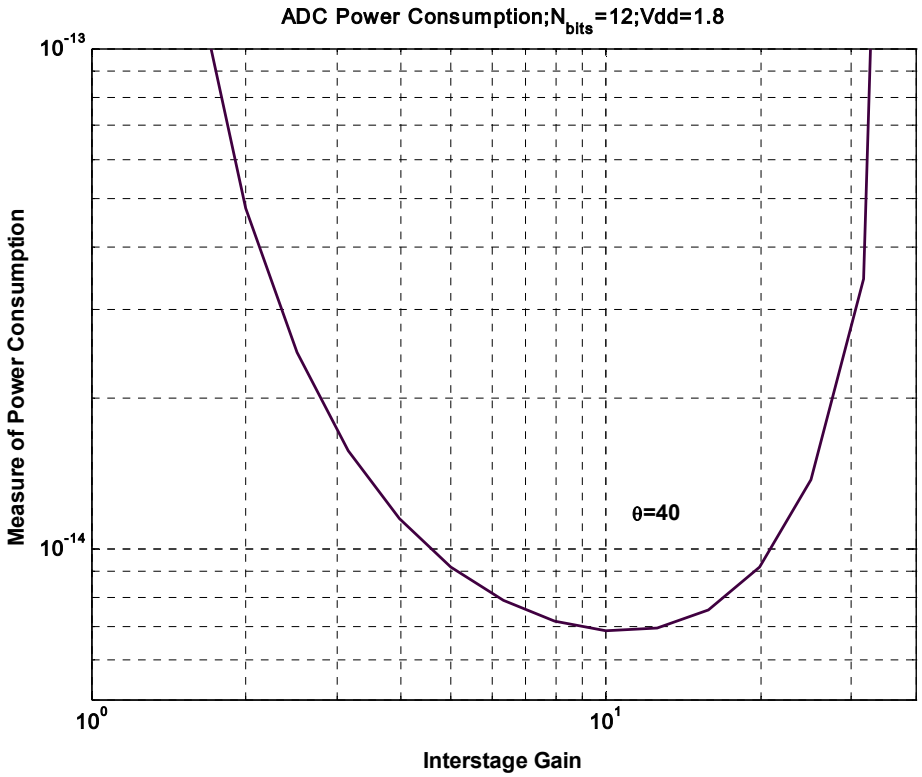


Figure 5-18 Optimization Example 3: G_{opt}=10

The above example yields an optimal interstage gain of 10. The operating speed in this case is ~7 Msps, and the penalty incurred on choosing the non-optimal interstage gain of 8 is about 5%, while choosing an interstage gain of 16 yields a penalty of about 7%. Additionally, the curve shows a flatter minima allowing more potential candidates to be chosen depending on convenience. Thus, the use of a non-binary radix is not so critical in this case. Many more cases can be considered. However, the general conclusions remain the same, and suggest that the generalized radix technique will become increasingly relevant with escalating performance requirements.

5.5 Conclusions & Comments

This chapter presents a novel technique for designing optimal pipelined ADCs by utlising a generalized radix pipeline stage. The implementation methodology for such a scheme is shown in detail with provision for incorporating digital redundancy. The scheme is illustrated using an example of a pipeline with an interstage gain of 3 (resolving $\log_2 3 = 1.585$ bits/stage) with digital redundancy of 1 bit. The validity of the method is demonstrated through MATLAB simulations conducted for the worst case scenario for various examples, showing the efficacy of digital correction.

Additionally, a few examples are presented highlighting the relevance of the generalized radix methodology to contemporary ADC design. It is shown that the tradeoffs and penalties incurred in designing high-performance, high-resolution converters are more significant, and the generalized radix scheme offers substantial savings when compared to the existing state-of-art. Thus, the techniques presented in this chapter are of immediate relevance to analog designers attempting the design of high-resolution, high-performance ADCs. The ternary (G=3) stage is of special importance as it offers an alternative to the 1.5-bit and 2.5-bit stages currently in use when ADCs must be designed to operate in the 100Msps range.

An example was presented to show that G=3 is the optimal choice for a 12-bit ADC for a 1.8V 0.18u CMOS process running at about 50Msps. These

specifications represent state-of-art pipelined ADCs, which currently use interstage gains of 2 or 4, and consume a few hundred milliwatt of power. The use of the generalized radix scheme offers a direct benefit in such an application with significant power savings.

For converters operating at low speeds, the tradeoffs may not be as significant, and conventional design techniques should still work well enough. However, the use of generalized radix stages entails very little overhead and offers a viable solution even in these cases. Additionally, generalized radix stages may be used as part of a non-uniform pipeline. As mentioned earlier, commercial piplined ADCs typically use a first stage with a slightly higher resolution so as to ease the requirements on the later stages. However, this is not a universal trend and different manufacturers follow different practices.

The more generic problem of optimizing a non-uniform pipelined ADC is rather intractable and cannot be addressed directly. It must be solved on a case-by-case basis, and is the domain of engineers with several years of experience in designing such ADCs. It is hoped that the work presented in this thesis will be of tremendous interest to the designers of high-performance, high-resolution ADCs.

6 Recent Advances In The Design Of Low-Power, Low-Voltage Pipelined & Cyclic ADCs

The discussion in the chapters so far has focused on the system-level aspects of pipelined and cyclic ADCs. The entire optimization process has proceeded largely at the system level, making some assumptions about the analog circuitry to be used. However, it is equally important to consider the circuit-level aspects to see if these conditions can be satisfied.

In this chapter, some recent ideas proposed by various authors will be reviewed. These ideas use ingenious system-level and circuit-level techniques to realize ADCs capable of low-power, low-voltage operation. These ideas are highly relevant to the discussion presented so far as they provide support to some of the crucial assumptions made in the optimization process discussed in Chapter 4.

One of the ideas involves the use of *chirp clocking* which allows the use of results from the optimization of pipelined ADCs to the implementation of cyclic ADCs. This truly makes the cyclic ADC the dual of the pipelined ADC. Another assumption was that the opamp used is a single-stage, single-transistor opamp. Typically, the high gain requirements necessitate the use of multi-stage opamps. However, some recent research presents a way to realize a pipelined ADC with single-transistor amplifiers and uses *time-shifted correlated double sampling* to enhance the moderate gain offered by these.

The last idea deals with common-mode control for pseudo-differential circuits. These circuits are widely used for low-voltage operation, where designing a common-mode feedback circuit is quite difficult..

6.1 Chirp Clocking Scheme For Cyclic ADCs

The cyclic ADC discussed in Chapter 3 uses just one stage, and therefore, the capacitors must be sized so as to satisfy the noise specifications for the entire converter. Thus, the freedom of scaling capacitors from one stage to the next is no longer available, and the opamp is always loaded by this maximum capacitance and dissipating large amounts of power.

The conventional clocking scheme for cyclic ADCs uses a fixed clock frequency and each conversion step is allocated the same amount of time for completion. However, the minimum settling time required per conversion reduces constantly (6-1) as the conversion proceeds because the accuracy requirements are reduced as the number of bits to be resolved goes down. This excess time slows down the conversion rate and also leads to waste of power as the opamp is still consuming DC power. This scenario is illustrated in Fig. 6-1.

$$t_{SHA} = \frac{(N+1-k)\ln 2}{\omega_{SHA}} \qquad \textbf{(6-1)}$$

In (6-1), N is the required converter resolution, while k represents the number of the conversion step and ω_{SHA} represents the closed-loop bandwidth of the S&H amplifier. Although the converter processes a residue every clock phase, the diagram shows an output once in every clock cycle for simplicity.

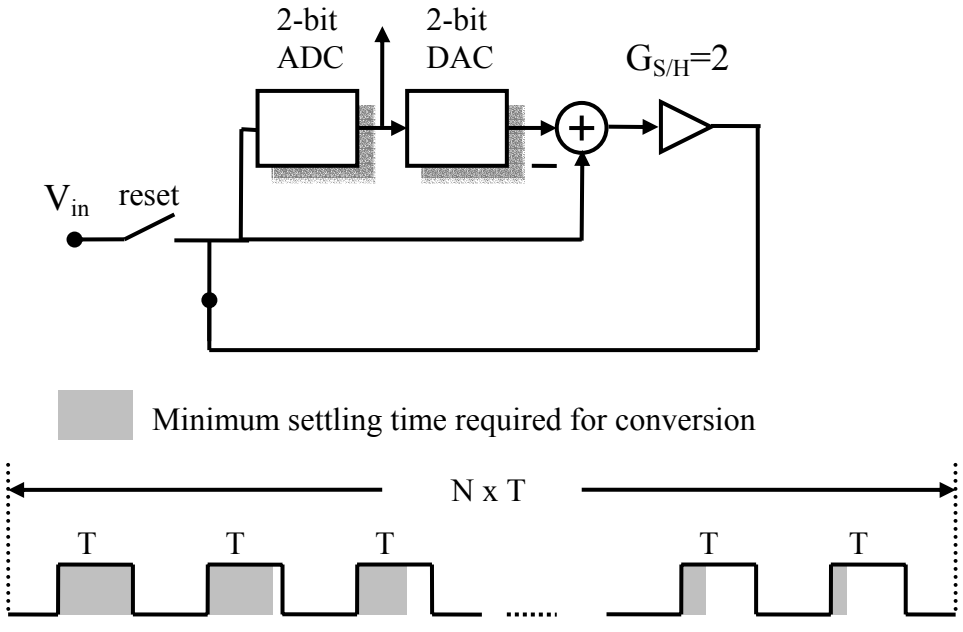


Figure 6-1 1.5-bit/stage cyclic ADC with conventional clocking scheme

The *chirp clocking* scheme proposed in [23] uses a clock whose time period is reduced as more and more bits are resolved. This scheme is shown in Fig. 6-2. This results in improvements of upto 82% in the conversion rate of the ADC, and 45% improvement in power efficiency per input sample. The complicated *chirp clocking* scheme can be realized using many different alternatives, depending on convenience. One can generate the required clock if N pre-determined phases are available. These can be combined using an edge combiner. Such a clock would need to run at a frequency N time greater than the one used in

conventional clocking. Alternatively, the various phases can be generated from a low-frequency clock using a PLL/DLL and combined using an edge combiner to generate these phases. Further details are available in [23].

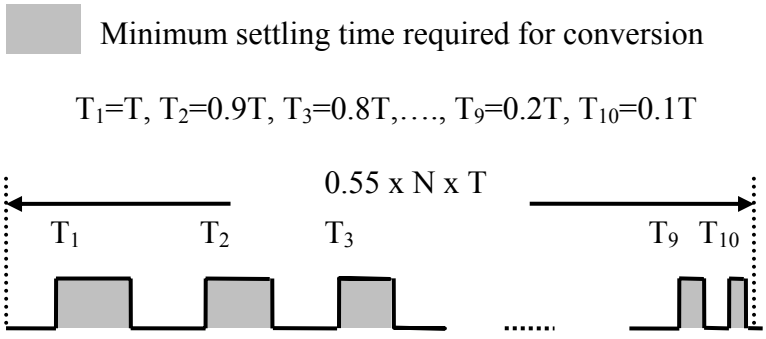


Figure 6-2 Chirp clocking scheme for the 1.5-bit/stage cyclic ADC

6.2 Time-Shifted Correlated Double Sampling

Pipelined ADCs rely on precision opamps to perform high-resolution A/D conversion. However, it is very difficult to achieve high opamp gain at low supply voltages. Let us examine the effect of finite opamp gain on the MDAC once more. The MDAC shown in Chapter 4 is redrawn here in Fig. 6-4. The output of the MDAC at the end of the amplification phase is given as:

$$V_{res} = \left(\frac{C_S + C_F}{C_F}\right)V_{in} - \left(\frac{C_S}{C_F}\right)V_{DAC} + e; \quad e = \frac{-1}{A}\left(\frac{C_S + C_F}{C_F}\right)V_{res} \qquad \textbf{(6-2)}$$

The error term *e* worsens with low opamp gain.

One can think of the error as the result of the opamp inverting node not being a *virtual ground* as is assumed for the ideal opamp. As seen in the equation above, this error is signal-dependent and causes nonlinear distortion.

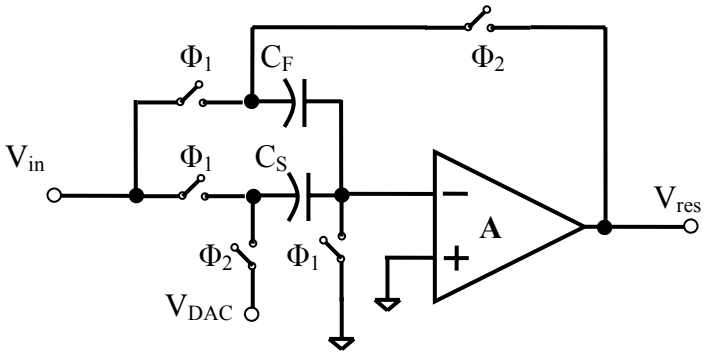


Figure 6-3 Conventional MDAC structure

Techniques like *correlated double sampling* (CDS) [26] can alleviate the effect of finite opamp gain to a considerable extent, but at the cost of halving the speed of operation. However, a recent approach [14][15] implements *time-shifted* CDS to achieve high-performance design using single-stage opamps. The scheme is shown in Fig. 6-3.

This scheme uses two pipelined paths working in parallel for the first few stages. One path represents the main signal path which operates for the entire length of the pipeline. The other path is the *predictive* path, which operates only for the first few stages where the opamp gain requirement is very high. The two paths share the same set of active stages (opamps and comparators) and process the same input signal from the first S&H, but the main signal path is delayed by one clock phase during which the predictive path processes the signal and stores

the finite opamp gain error on a capacitor. In the next phase, the main signal path uses this information to correct for this signal-dependent error and realize a more ideal virtual ground. The error term e_i is now shown to be proportional to $1/A^2$, which is a significant improvement (6-3). The MDAC implementing the scheme is shown in Fig. 6-5.

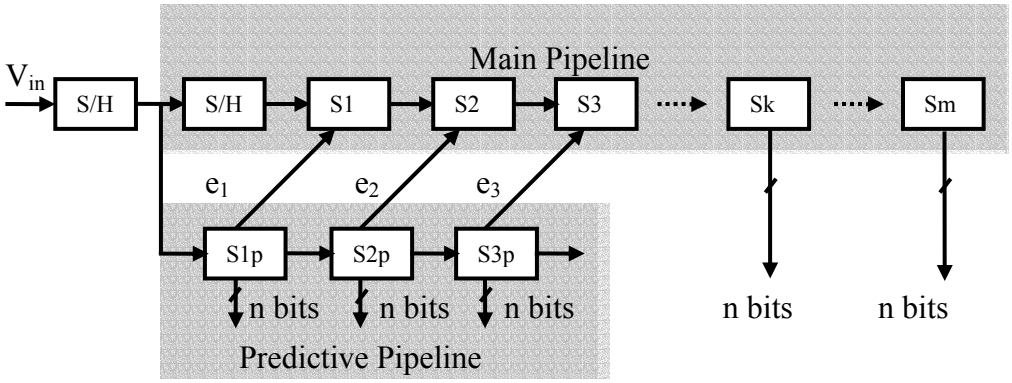


Figure 6-4 Modified Pipelined ADC Architecture

This scheme avoids the speed penalty imposed by conventional CDS and allows substantial savings in power. This has been used to design a 10-bit pipelined ADC achieving a 65dB SFDR and 54dB SNDR at 100Msps while consuming 67mW from a 1.8V supply. This design used cascoded CMOS inverters as amplifiers [14][15].

$$e_i = \frac{-i}{A^2}\left(\frac{C_S + C_F}{C_F}\right)\left[\left(\frac{C_S + C_F}{C_F}\right)V_{res,i}(n) - \left(\frac{C_I}{C_F}\right)V_{res,i}\left(n - 1/2\right)\right] \qquad \textbf{(6-2)}$$

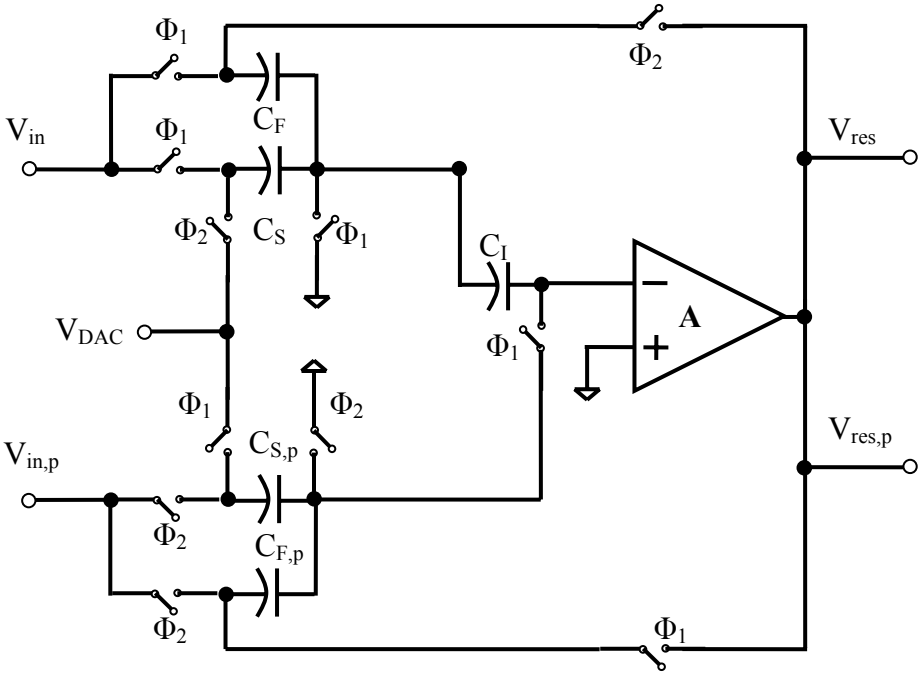


Figure 6-5 Modified MDAC architecture

6.3 Common Mode Control For Pseudo-Differential Designs

Design of fully differential circuits often becomes very difficult at low supply voltages, and pseudo-differential designs are often used, as in the case of the approach demonstrated in the earlier section. Unlike fully differential designs, which strongly reject any common mode signal, these designs amplify it along with the signal, and this may cause the later stages to saturate. Thus, some form of common mode feedback or control is necessary. A simple technique entailing no overhead in terms of speed or cost is proposed in [31]. This is called the differential *float* sampling scheme, as the MDAC operation does not involve sampling the common mode signal on the sampling capacitors (Fig. 6-6).

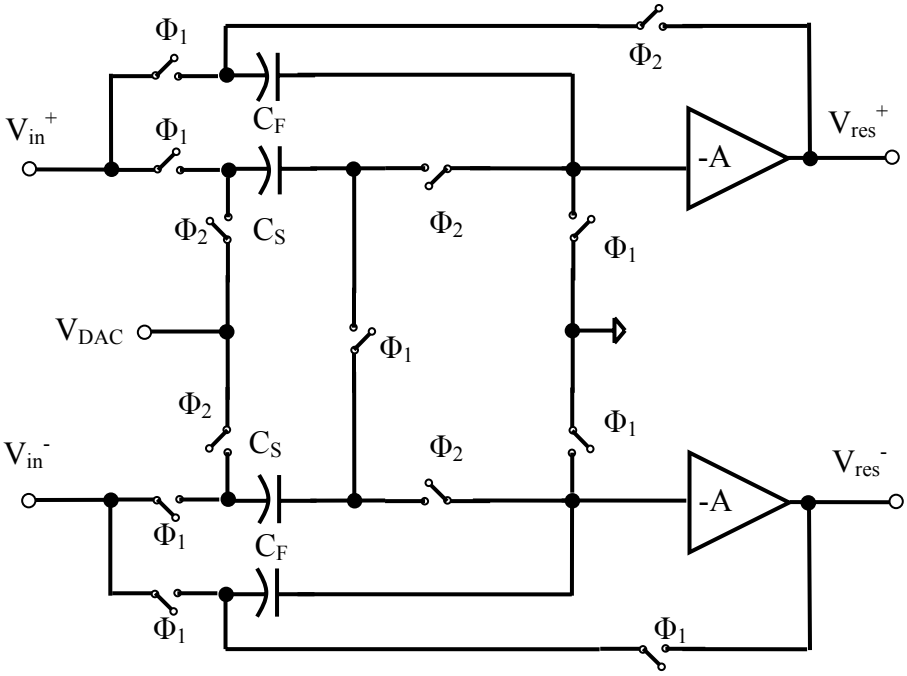


Figure 6-6 Pseudo-Differential *float* sampling scheme for MDAC

In this scheme, the common mode signal always sees a unit gain, while the differential signal is amplified. Additionally, the common mode gain is realized through voltage mode operation, and does not vary with capacitor mismatch or inaccuracy. Thus, the common mode signal applied at the input of the pipelined ADC propagates unchanged throughout the length of the pipeline, and problems associated with stabilizing the CMFB loop are absent in this purely feedforward scheme.

6.4 Conclusion

This chapter presents a few recently proposed techniques for improving the performance of pipelined and cyclic ADCs. Additionally, design of high resolution ADCs is limited by the matching accuracy of capacitors. This is typically limited to about 0.1 % accuracy with careful layout. Thus, calibration techniques must be used to compensate for capacitor mismatch. Several techniques have been proposed that accomplish this task [8][9][10][11][12][19][21][22].

7 Conclusion

This dissertation presents two key ideas:

Firstly, it demonstrates that optimal performance in a pipelined ADC may require a design where each pipeline stage has a non-binary gain and resolves a fractional number of bits.

Secondly, it proposed a scheme for realizing the pipelined ADC with a generic, non-binary interstage gain. This scheme is the generalized radix scheme and is shown to offer the optimum solution for the design of power-efficient pipelined ADCs. The entire methodology for realizing such a generalized radix pipeline with provision for digital redundancy and error correction is presented in detail. The efficacy of the scheme is demonstrated through extensive simulations.

The most significant advantage of the scheme is its ability to achieve an optimal tradeoff among various competing factors such as power consumption, speed and resolution, while retaining a very simple implementation, and entailing no additional overhead in terms of design complexity, design cost or speed.

The ideas presented in this thesis are easily applied to optimization of cyclic ADCs as well. However, a few modifications may be necessary to adapt the optimization scheme presented here.

Bibliography

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

[2] S.H. Lewis, "Video-rate analog-to-digital conversion using pipelined architectures," Ph.D. dissertation, U.C. Berkeley, 1987.

[3] S.H. Lewis, "Optimizing the stage resolution in pipelined, multistage, analog-to-digital converters for video-rate applications," *IEEE Trans. Circuits & Systems II*, vol. 39, no. 8, pp. 516-523, Aug. 1992.

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

[5] K. Nagaraj *et al.*, "A 250-mW, 8-b, 52-Msamples/s parallel pipelined A/D Converter with Reduced Number of Amplifiers," *IEEE J. Solid-State Circuits*, vol. 32, no. 3, pp. 312-320, Mar. 1997.

[6] T. Cho and P.R. Gray, "A 10-bit, 20-Msample/s, 35 mW pipelined A/D converter," *IEEE J. Solid-State Circuits*, vol. 30, no. 3, pp. 166-172, Mar. 1995.

[7] B. Song, M. Tompsett, K. Lakshmikumar, "A 12-b, 1-Msample/s capacitor error-averaging pipelined A/D converter," *IEEE J. Solid-State Circuits*, vol. 23, no. 6, pp. 1324-1333, Dec. 1988.

[8] U. Moon and B. Song, "Background digital calibration techniques for pipelined ADCs," *IEEE Trans. Circuits & Systems II*, vol. 44, no. 2, pp. 102-109, Feb. 1997.

[9] D. Chang and U. Moon, "Radix-based digital calibration technique for multi-stage ADC," *IEEE Int. Symp. Circuits & Systems.*, vol. II, pp 796-799, May 2002.

[10] J. Li and U. Moon, "Background calibration techniques for multi-stage pipelined ADCs with digital redundancy," *IEEE Trans. Circuits & Systems II*, pp. 531-538, Sep. 2003.

[11] Jipeng Li and U. Moon, "An extended radix-based digital calibration technique for multi-stage ADC," *IEEE Int. Symp. Circuits & Systems*, vol. I, pp. 829-832, May 2003.

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

[13] Y. Chiu, "Inherently linear capacitor error-averaging technique for pipelined A/D converters," *IEEE Trans. Circuits & Systems II*, vol. 47, no. 3, pp. 229-232, Mar. 2000.

[14] J. Li and U. Moon, "A 1.8-V 67mW 10-bit 100MSPS pipelined ADC using time-shifted CDS technique," *IEEE Custom Int. Circuits Conf.*, pp. 413-416, Sep. 2003.

[15] J. Li and U. Moon, "High-speed pipelined A/D converter using time-shifted CDS technique," *IEEE Int. Symp. Circuits Syst.*, vol. I, pp. 909-912, May 2002.

[16] P. Li, M. Chin, P.R. Gray, R. Castello, "A ratio-independent algorithmic analog-to-digital conversion technique," *IEEE J. Solid-State Circuits*, vol. 19, no. 6, pp. 828-836, Dec. 1984.

[17] Z. Zheng, U. Moon, J. Steensgaard, B. Wang, and G.C. Temes, "Capacitor mismatch error cancellation technique for a successive-approximation A/D converter," *IEEE Int. Symp. Circuits & Systems.*, vol. II, pp. 326-329, May 1999.

[18] Z. Zheng, B. Min, U. Moon, and G.C. Temes, "Efficient error-cancelling algorithmic ADC," *IEEE Int. Symp. Circuits & Systems*, vol. I, pp. 451-454, May 2000.

[19] S. Lee and B. Song, "Digital-domain calibration of multi-step analog-to-digital converters," *IEEE J. Solid-State Circuits*, vol. 27, no. 12, pp. 1679-1688, Dec. 1992.

[20] T. Rengachari, "A 10-bit algorithmic A/D converter for a bio-sensor," M.S. Thesis, Oregon State University, 2004.

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

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

[23] M. Kim, G. Ahn, U. Moon, "An improved algorithmic ADC clocking scheme," *IEEE Int. Symp. Circuits & Systems.*, vol. I, pp. 589-592, May 2004.

[24] S. Chin and C. Wu, "A CMOS ratio-independent and gain-insensitive algorithmic analog-to-digital converter," *IEEE J. Solid-State Circuits*, vol. 31, no. 8, pp. 1201-1207, Aug. 1996.

[25] R. Peck, D. Schroeder, "A low-power entropy-coding analog/digital converter with integrated data compression," *Proc. ESSCIRC*, pp. 173-176.

[26] C. Enz and G. C. Temes, "Circuit techniques for reducing the effects of opamp imperfections: autozeroing, correlated double sampling and chopper stabilization," *Proc. IEEE*, pp.1584-1614, Nov. 1996.

[27] R. Gregorian and G. C. Temes, "Analog MOS integrated circuits for signal processing," John Wiley & Sons, New York, 1986.

[28] B. Razavi, "Principles of data conversion design," John Wiley & Sons, New York, 1994.

[29] D.A. Johns and K. Martin, "Analog integrated circuit design," John Wiley & Sons, New York, 1997.

[30] M. Gustavsson, J.J. Wikner, N.N. Tan, "CMOS data converters for communications," Kluwer Academic Publications, Boston, 2000.

[31] J. Li, "Accuracy enhancement techniques in low-voltage, high-speed pipelined ADC design," Ph.D. Thesis, Oregon State University, 2003.