HIGH-RESOLUTION PIPELINED ANALOG-TO-DIGITAL CONVERSION

by

Sehat Sutarja

Memorandum No. UCB/ERL M88/27

2 May 1988

HIGH-RESOLUTION PIPELINED ANALOG-TO-DIGITAL CONVERSION

by

Sehat Sutarja

Memorandum No. UCB/ERL M88/27

2 May 1988

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

HIGH-RESOLUTION PIPELINED ANALOG-TO-DIGITAL CONVERSION

Ph.D. Sehat Sutarja Department of EECS

Chairman of Committee

ABSTRACT

This research addresses techniques for implementing area efficient medium- to high-speed and high-resolution CMOS analog-to-digital (A/D) converters for use in the receivers for digital telecommunication systems and for general purpose data acquisition applications. Although pipelined A/D conversion techniques show promise for minimizing the area utilization, the linearity and resolution of a straight-forward implementation of such converters, however, are usually limited to approximately 10 bits because of various component matching errors found in a typical monolithic implementation.

In this dissertation, simple analog and digital correction techniques for reducing the effects of most of the matching errors will be studied. These techniques can replace costly trimming and complicated self-calibrating techniques for many applications that require high resolution and medium linearity. To verify the effectiveness of these techniques, an experimental prototype pipelined A/D converter that was optimized for minimum area and power dissipation for applications that only require medium sampling rate (~200 ksamples/s) but high resolution (~13 bits) has been designed and fabricated using a standard 3-μm double-poly single-metal CMOS processing technology.

To my wife, Weili Dai, and my parents

for their love, encouragement, and support

ACKNOWLEDGEMENT

I would like to deeply thank Professor Paul R. Gray for his guidance and support throughout my years at U.C. Berkeley. I would also like to thank Professor David A. Hodges for his valuable advice.

I would also like to thank my fellow students, especially Joey Doernberg, Bosco Leung, Steve Lewis, Nan-Sheng Lin, Yu-Min Lin, Hyun-Jong Shin, Lawrence Tse, and Lee-Chung Yiu for the many stimulating discussions. Special thanks to Steve Lewis and Pantas Sutardja for their helps in editing my dissertation.

I would also like to thank Don Webber for fixing RELAX2, a computer program used for simulating the prototype A/D converter, and to Brian Richards for supplying computer programs that make the printing of this dissertation much easier.

This work was supported through grants from National Science Foundation, and partly through California MICRO program.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

Digital-signal-processing (DSP) techniques have been proven more suitable than their analog counterparts in many complex signal-processing applications such as high-speed modems,[1] speech and image recognition systems, and high-performance color TV applications. [2,3,4] With the advancements of CMOS processing technologies and DSP techniques, an increasing number of other applications currently implemented in traditional analog signal-processing techniques can be implemented in digital techniques to improve the overall system performance and to reduce the overall manufacturing and maintenance cost. Such systems receive analog input signals and thus require analog-to-digital (A/D) converters if digital techniques are to be used for their signal processing.

Depending on the exact application, an A/D converter that has a resolution and/or an accuracy of between 8 and 16 bits and a conversion rate in the range from several samples/second (s/s) to several hundreds of Ms/s might be needed. Most research and development on monolithic A/D conversion techniques in recent years, however, has been concentrated in two areas. The first is in high-resolution/accuracy A/D conversion techniques with sampling rates in the audio range for applications such as telephony and instrumentation. The second is in very high-speed but low-resolution A/D conversion techniques for applications such as video A/D interfaces and radar applications. The combination of high resolution (ie. >> 8 bits) and sampling rates far above the audio range has not received wide attention until recently. However, applications such as Integrated Service Digital Network (ISDN)[5] receivers, high-definition TVs, and satellite infra-red (IR) imagers require this type of performance. The research to be described here is aimed at a

conversion technique that is particularly suitable for implementing high-resolution and high-speed pipelined A/D converters.

The pipelined A/D converter architecture has historically been used in medium- to high-resolution, high-speed board-level A/D converters. In the past, however, such an architecture was not popular for monolithic CMOS implementations because of the opamp speed limitation. With the present wide availability of fine-line CMOS processing technologies, such an architecture is now attractive for high-speed A/D converter implementations. However, as in the case of the board-level implementations, the linearity and resolution of the monolithic implementations can potentially be limited to around 10 bits unless trimming or (self-) calibrating techniques[6] are also used.

The objectives of this thesis are to determine the basic limitations to the linearity and resolution of pipelined A/D converters and to investigate new circuit techniques that can be used to increase the maximum achievable resolution to levels beyond those previously achievable. One such technique is segmentation similar to that used in an inherently monotonic charge-redistribution A/D conversion[7] and in a segmented current D/A conversion [8,9] architecture. Using such a technique, the resolution (differential linearity) of a pipelined A/D converter can be made insensitive to component matching errors and finite opamp open-loop gain. However, since linearity correction techniques such as trimming and self-calibrating techniques are not used, the maximum achievable linearity of the pipelined A/D converter that will be described in this thesis is still limited by the component matching accuracies. So the other objective of this thesis is to investigate the maximum achievable linearity based on component matching alone. Layout techniques that can improve the linearity will be studied.

This thesis is organized as follows. Chapter 2 is a review of various A/D conversion architectures that are commonly implemented in CMOS processing technology. Chapter 3 covers the practical sources of linearity errors in a pipelined A/D converter and solutions to them. An impor-

tant topic that will be covered in this chapter is a technique that can be used to make the resolution of a pipelined A/D converter insensitive to component matching errors and finite opamp open-loop gain. Chapter 4 covers the high-level design considerations for optimum speed and area utilization. Chapter 5 covers circuit design considerations for a medium-speed but high-resolution prototype pipelined A/D converter that was optimized for minimum overall area. The experimental test setup and test results will then be described in chapter 6. Finally, the conclusions are given in chapter 7.

CHAPTER 2

OVERVIEW OF A/D CONVERSION ARCHITECTURES

Most of the currently used A/D conversion architectures can be categorized into three basic architectures: serial, successive-approximation, and parallel. The fundamental difference between these architectures is in the way analog comparisons are performed to obtain the digital representation of the analog input signal. These comparisons can be done by comparing all of the possible representations of the digital output codes one at a time, by successively comparing some of the representations of the digital output codes, or by simultaneously comparing all of the possible representations of the digital output codes; hence, the terms serial, successive-approximation, and parallel came from.

Other A/D conversion architectures, with the exception of the noise-shaping oversampling A/D conversion architectures, are extensions of these three basic architectures. For example, a subranging/multi-step architecture is actually a successive-approximation architecture in disguise, and so is a pipelined architecture. Neither architecture is usually referred to as the successive-approximation architecture because the term successive-approximation architecture is almost exclusively used to refer to a class of A/D converters that use the so-called binary search algorithms in the analog comparison steps. In a subranging/multi-step and pipelined architecture, all three basic architectures can in fact be combined to obtain a certain tradeoff between conversion speed, chip area, resolution, linearity, and many other design requirements.

In this chapter, the various architectures mentioned above will be briefly reviewed. Emphasis will be given on the number of clock cycles needed to complete each A/D conversion, and on the chip-area utilization as a function of the resolution for each architecture.

2.1. SERIAL ARCHITECTURE

In an A/D converter using the serial technique,[10] the A/D conversion process is done by possibly comparing all of the representations of the digital output codes to the analog input signal sequentially in time. There are several ways these representations can be generated. The simplest way is to generate a ramp signal by integrating a fixed current onto an integrator. The output voltage of the integrator is then compared to the analog input signal at some predetermined time intervals. The digital output code representation of the analog input signal can then be obtained easily by counting the number of time intervals (clock cycles) required for the ramp signal to equal the analog input signal from its zero state. Because most serial converters use the integrating technique, the serial architecture is usually referred to as the integrating architecture.

The most important advantage of the integrating converter is that an inherently monotonic A/D conversion can be obtained because an inherently monotonic ramp signal can be readily generated without the need of any component matching requirement. In fact, even high accuracy/linearity can be obtained by using a slightly modified version of the integrating architecture, the so-called dual-slope integrating A/D conversion architecture. A conceptual block diagram of a dual-slope integrating A/D conventer is shown in Figure 2.1.1. Because of the simplicity of the required circuit regardless of the number of bits that must be resolved, only a small amount of silicon area is needed to build such a converter. Unfortunately, the required number of clock cycles increases exponentially with the number of bits that must be resolved. The conversion speed of the integrating A/D converter is thus usually too slow for most applications other than slow instrumentation applications.

2.2. SUCCESSIVE-APPROXIMATION ARCHITECTURE

A conceptual circuit diagram of an A/D converter using the successive-approximation technique is shown in Figure 2.2.1. In such a converter, the A/D conversion is done by successively comparing the analog input signal to the representation of the approximated digital output code (the

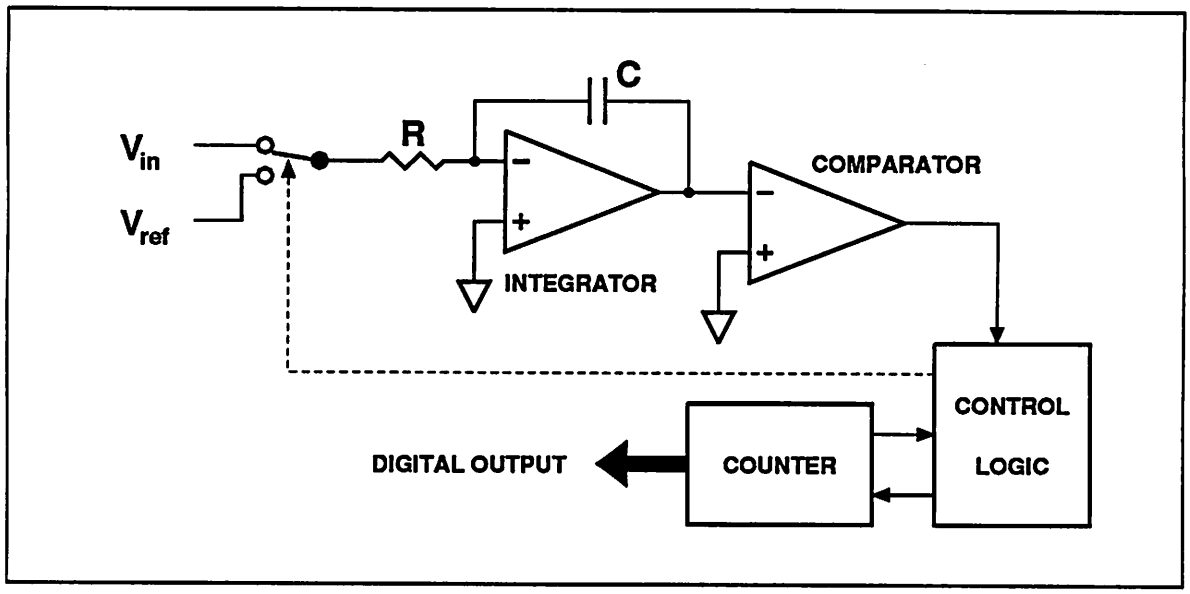

Figure 2.1.1 Conceptual Block Diagram of an Integrating A/D Converter.

output of the D/A converter) in a manner similar to a binary-search algorithm. As such, the number of comparison steps increases only linearly with the number of bits that must be resolved. In many implementations, only n clock cycles are needed for each n-bit conversion.

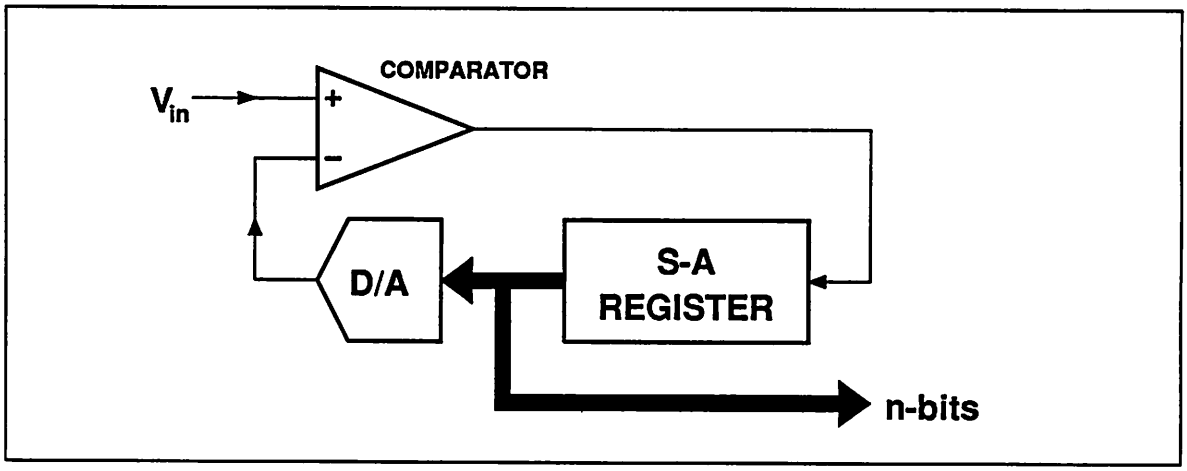

Figure 2.2.1 Conceptual Block Diagram of a Successive-Approximation A/D Converter.

Because there are many ways to implement the D/A converter function to produce the representation of the approximated digital output code, there are many variations of the

successive-approximation architecture. In this section, however, only two of them will be covered.

2.2.1. Charge-Redistribution Architecture

The charge-redistribution technique [11] is probably the most widely used successive-approximation technique for implementing high-resolution A/D converter functions in CMOS processes with high quality capacitors. A conceptual block diagram of an A/D converter using such a technique is shown in Figure 2.2.1.1. In a charge-redistribution A/D converter, the D/A converter function is usually implemented using an n-bit binary capacitor array that can be reconfigured as a voltage divider to obtain binary fractions of the reference voltage for comparisons to the analog input signal.

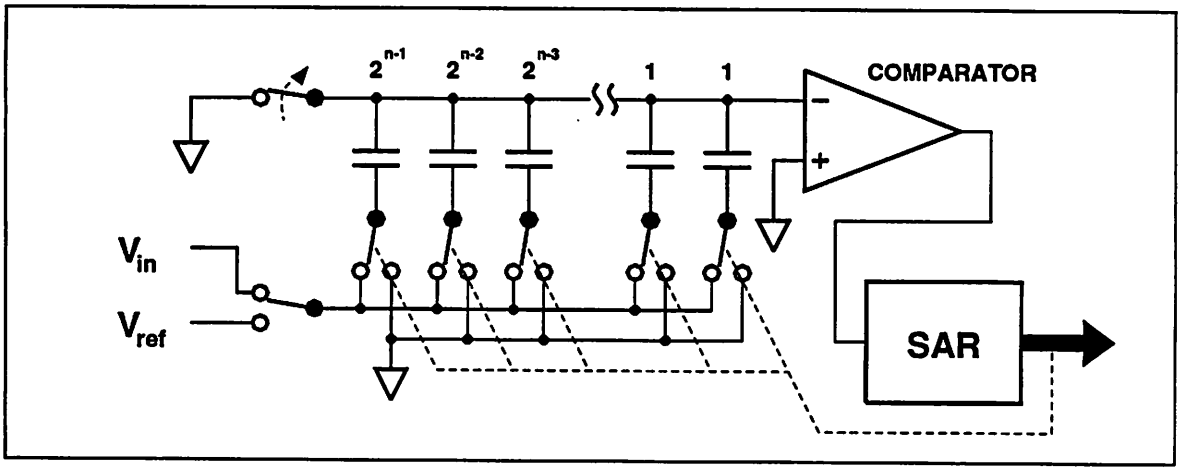

Figure 2.2.1.1 Block Diagram of a Charge-Redistribution A/D Converter.

A straight implementation of a high-resolution charge-redistribution A/D converter, however, requires a large amount of silicon area because the number of the required capacitors increases exponentially with the number of bits that must be resolved. Furthermore, because of mismatch in the capacitor array, the maximum achievable resolution is usually limited to 10 bits. One solution to these limitations is to employ the so-called inherently monotonic D/A converter structure which can be implemented using a resistive-main-DAC/capacitive-sub-DAC structure,[7]

as illustrated in Figure 2.2.1.2.

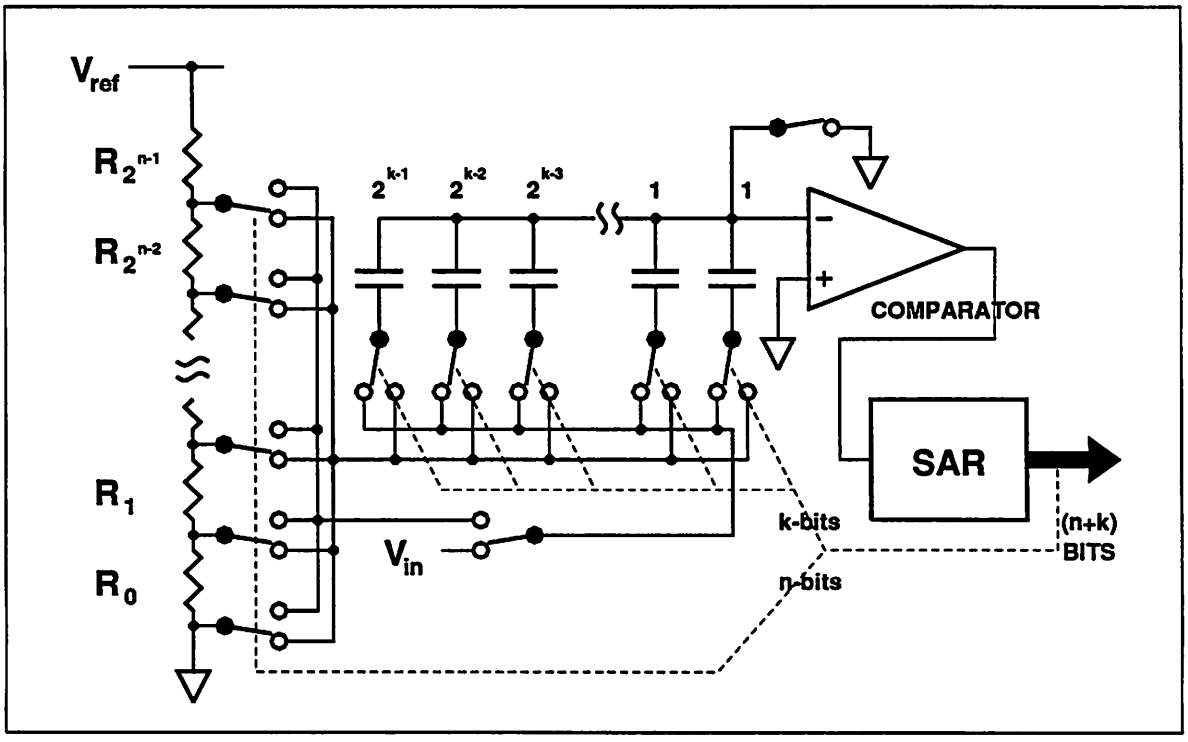

Figure 2.2.1.2 Block Diagram of the CR A/D Converter Using R-main-DAC/C-sub-DAC Structure

2.2.2. Algorithmic/Cyclic Architecture

An important successive-approximation technique for CMOS A/D converter implementation is the algorithmic [12] (cyclic) technique. A conceptual block diagram of the algorithmic A/D converter is shown in Figure 2.2.2.1. With the exception of the integrating A/D converter, the algorithmic A/D converter uses the smallest amount of silicon area compared to any other high-resolution converter implementation because it can be implemented using only two or three opamps and a comparator regardless of the number of bits that must be resolved. However, because the clock cycle time for each bit decision is determined by settling time of the opamp, the conversion speed of the algorithmic A/D converter is slower than that of the charge-redistribution A/D converter.

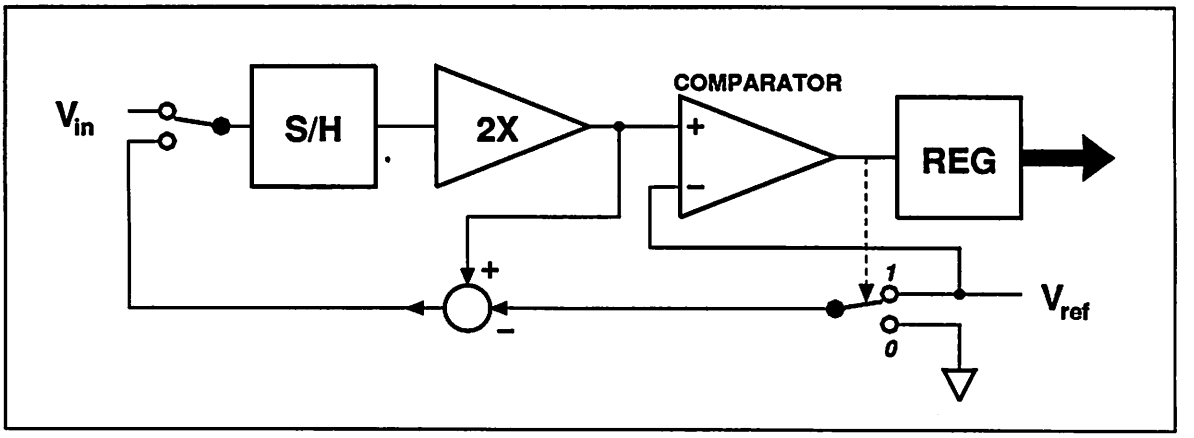

Figure 2.2.2.1 Block Diagram of an Algorithmic A/D Converter.

Furthermore, because errors that are accumulated during the A/D conversion process, the required component matching accuracy of the algorithmic A/D converter is, for the same resolution, usually twice that of the charge redistribution A/D converter. Solutions to this limitation are (self-) calibration which can significantly increase the manufacturing cost, and ratio-independent [13] and reference-refreshing [14] correction which can significantly reduce the conversion speed.

2.3. PARALLEL ARCHITECTURE

In an A/D converter using the parallel technique, the analog input signal is simultaneously compared to all of the n-bit digital code representations. An extremely fast conversion speed can be obtained because each A/D conversion can be carried out in one clock cycle. A flash A/D converter is one such converter. It is usually implemented using a bank of comparators and latches, and a resistor string, as illustrated in Figure 2.3.1.

Unfortunately, because the number of comparators and latches increases exponentially with the number of bits that must be resolved, flash A/D converters are at present only practical for applications that do not require more than 8 bits of resolution (eg. digital TV and other imaging applications).

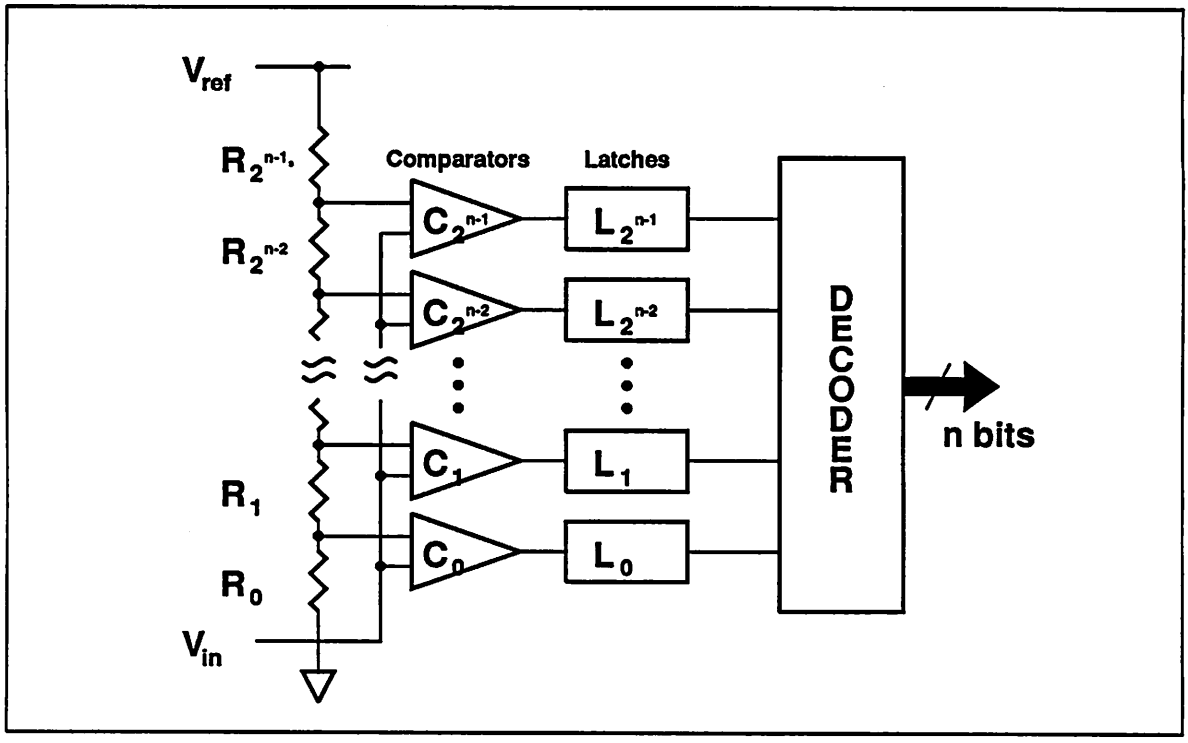

Figure 2.3.1 Conceptual Block Diagram of a Flash A/D Converter.

2.4. SUBRANGING/MULTI-STEP ARCHITECTURE

As mentioned earlier, a subranging A/D converter [15] is in principle a successive-approximation A/D converter. The difference is that more than 1 bit is resolved at a time to reduce the number of clock cycles needed for each A/D conversion. A conceptual block diagram of a typical subranging A/D converter is shown in Figure 2.4.1. Typically, flash A/D converters are used to implement the low-resolution stages of the subranging converter. As such, the subranging architecture is also usually referred to as the multi-step flash architecture.

The main advantage of the subranging architecture is that the area of a subranging A/D converter can be made relatively small because low-resolution flash A/D converters can be used in each stage.

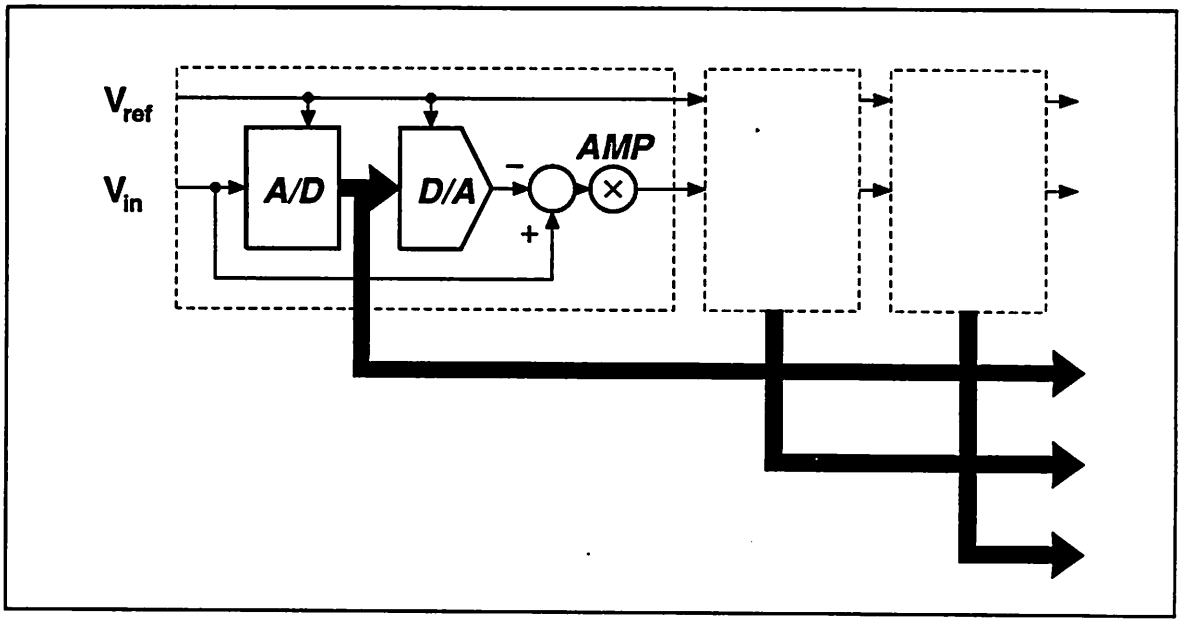

Figure 2.4.1 Conceptual Block Diagram of a subranging A/D Converter.

2.5. PIPELINED ARCHITECTURE

A pipelined A/D converter [16] is an extension of the subranging A/D converter. A generalized block diagram of a pipelined A/D converter is shown in Figure 2.5.1. It consists of at least two stages that are made from a sample-and-hold (S/H) circuit, a low-resolution A/D and D/A converter, a subtraction circuit, and an interstage amplifier.

The pipelined A/D converter draws its primary advantage from the fact that at any given time, the 1^{st} stage operates on the most current analog input sample, while the 2^{nd} stage operates on the amplified residue of the previous input sample, and so on. The inherent concurrency of operations thus results in an A/D converter with a conversion speed that is only limited by the time it takes to process the analog information in one stage. Because the only major speed limiting factors are the input-sampling and residue-amplification steps, only 2 clock cycles are needed for each A/D conversion.

Another advantage of the pipelined A/D conversion architecture is that because each pipelined stage can be made to have low-resolution without slowing down the conversion speed and

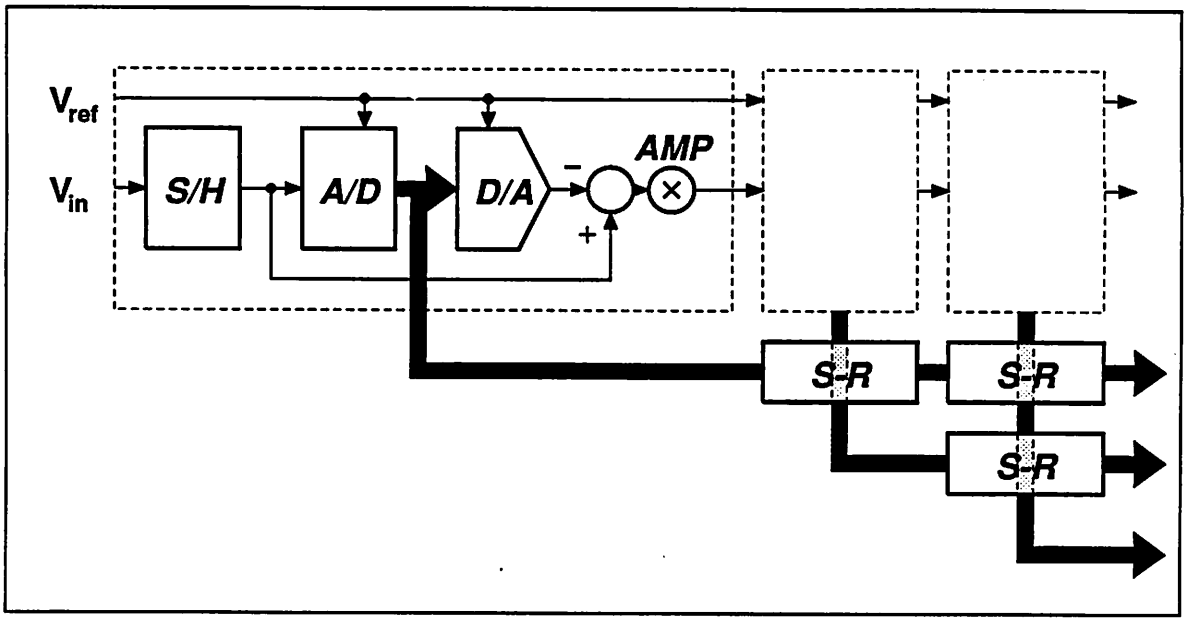

Figure 2.5.1 Conceptual Block Diagram of a Pipelined A/D Converter.

because the number of stages increases only approximately linearly with the resolution, the area utilization and power dissipation of a pipelined A/D converter can be made relatively small compared to that of high-resolution A/D converter implementations using other architectures.

2.6. OVERSAMPLING ARCHITECTURES

The oversampling A/D conversion architectures, especially the Sigma-Delta Modulators [17, 18, 19] (Σ–ΔM), have gained wide acceptance recently because the linearity of the Σ–ΔM can be made independent of the component matching accuracy in a single-bit-oversampling implementation. By using a technique referred to as noise shaping, a much lower oversampling ratio than that required in a straight oversampling converter can be used to obtain an A/D converter with the same effective resolution. By averaging the oversampled signal over many samples or more precisely by filtering the unwanted noise power spectrum, an accurate digital representation of the analog input signal can then be obtained. Two types of Σ–ΔM architectures will be discussed here.

2.6.1. 1^{st} Order Σ–ΔM

The simplest form of all Σ–ΔMs is of the 1^{st} order.[18] The block diagram of the 1^{st} order Σ–ΔM is shown in Figure 2.6.1.1. It can be shown that the noise power spectrum at the output of a 1^{st}-order Σ–ΔM that uses a 1-bit quantizer is shaped in such a way that when the noise above the frequency band of interest is filtered out, the achievable peak SNR depends only on the over-sampling ratio, R_{os}, by the relationship

$$SNR_{peak} = 9\,R_{os}\ dB \qquad (2.6.1.1)$$

Since the peak SNR is related to the resolution of an A/D converter by the following relationship:

$$SNR_{peak} = 6.02n + 1.72\ dB, \qquad (2.6.1.2)$$

the number of clock cycles per conversion, N_{c1}, required by the 1^{st} order Σ–ΔM can now be computed from the above two equations:

$$N_{c1} = 2^{R_{os}} \approx 2^{\frac{n}{1.5}}, \qquad (2.6.1.3)$$

where n is the equivalent overall number of bits to be resolved. Note that N_{c1} is the number that should be used for comparing the ultimate achievable conversion speed of the 1^{st} order Σ–ΔM architecture to that of the more traditional A/D conversion architectures when they are implemented using comparable processing technologies.

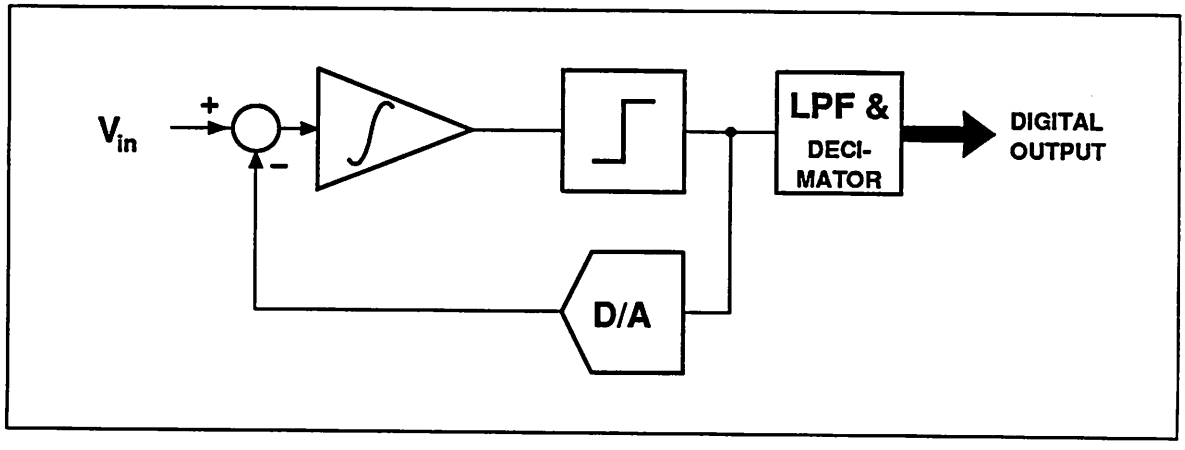

Figure 2.6.1.1 Conceptual Block Diagram of a 1^{st} Order Σ–ΔM.

Because the SNR is improved by only $9dB$ or equivalently the resolution is improved by $1.5bits$ every time the oversampling rate is doubled, the 1^{st} order Σ–ΔM still requires too many clock cycles per effective conversion, especially for high-resolution applications. For this reason, it has not been widely used.

2.6.2. 2^{nd} Order Σ–ΔM

In order to reduce the amount of oversampling ratio required for the Σ–ΔM so that a higher-resolution and higher-speed converter can be built, the in-band noise power of the oversampled signal must be further reduced. One way to do this without using a multiple bit D/A converter (which suffers from the mismatch problem as in the case of the more traditional A/D converters) in the feedback loop, is to increase the order of the loop. However, the 2^{nd} order loop is the highest loop order currently known to be unconditionally stable. A block diagram of a 2^{nd} order Σ–ΔM is shown in Figure 2.6.2.1. It can be shown that the noise power spectrum at the output of the 2^{nd} order Σ–ΔM is shaped in such a way that when the noise power above the frequency band of interest is filtered out, the effective peak SNR that can be obtained from such a converter is given by:

$$SNR_{peak} = (15\,R_{os} - x)\,dB \qquad (2.6.2.1)$$

In the above equation, x is the degradation of the peak SNR caused by system overloading and from the fact that the white noise model that is used to analyze the converter breaks down when the input signal is close to the peak range. The degradation factor x depends on the pole locations of the integrators, but is typically around 15.

Because the peak SNR is improved by $15dB$ or equivalently the effective resolution is improved by 2.5 $bits$ every time the oversampling rate is doubled, the 2^{nd} order Σ–ΔM requires much fewer clock clock cycles per effective conversion than that required by the 1^{st} order Σ–ΔM for the same resolution. The number of clock cycles needed for each effective A/D conversion step, N_{c2}, can similarly be computed from equation 2.6.2.1 and 2.6.1.2 to yield:

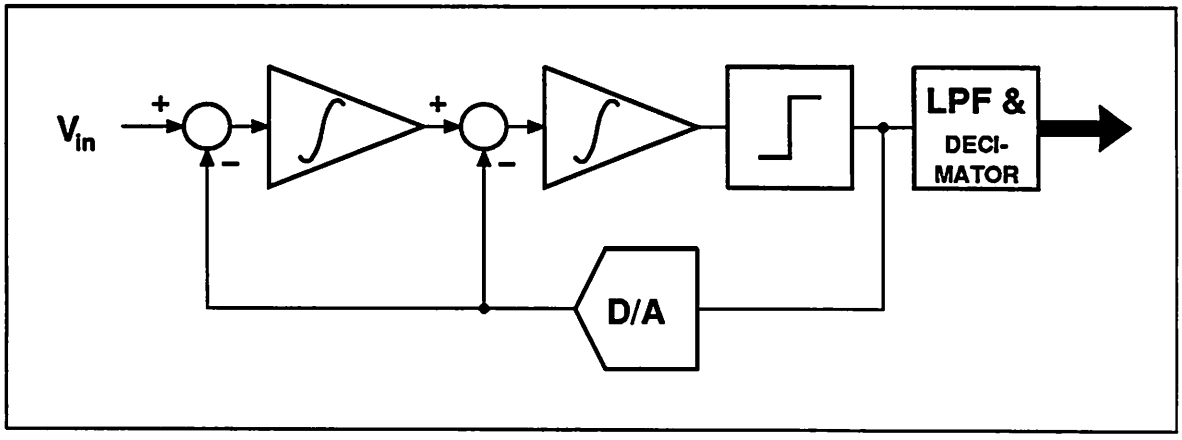

Figure 2.6.2.1 Conceptual Block Diagram of a 2^{nd} Order Σ–ΔM.

$$N_{c2} = 2^{\frac{n}{2.5}+1}, \tag{2.6.2.2}$$

where n is the equivalent overall number of bits to be resolved.

Although the number of clock cycles required by the 2^{nd} order Σ–ΔM is significantly fewer than that required by the 1^{st} order Σ–ΔM, it is still too many compared to other A/D conversion architectures. For this reason, the 2^{nd} order Σ–ΔM has been mostly used for slow- to medium-speed and high-resolution applications.

2.7. SUMMARY

The number of clock cycles needed in each A/D conversion step for all the A/D conversion architectures described is summarized graphically in Figure 2.7.1. Although this speed comparison does not include factors such as the minimum achievable cycle time for different architectures and the dependency of the cycle time on the resolution, it illustrates that most A/D conversion architectures require a large number of clock cycles for resolution above the 12-bit level. In particular, the most widely used architectures for high-resolution A/D implementations, the 2^{nd} order Σ–ΔM and the successive-approximation A/D conversion architecture, require around 64 and 12 clock cycles per conversion, respectively, for 12-bit resolution.

On the other extreme, the flash A/D conversion architecture is not suitable for high-resolution applications because the required area increases exponentially with the resolution. This leaves the pipelined A/D conversion architecture as the most suitable architecture for high-resolution and medium- to high-speed applications.

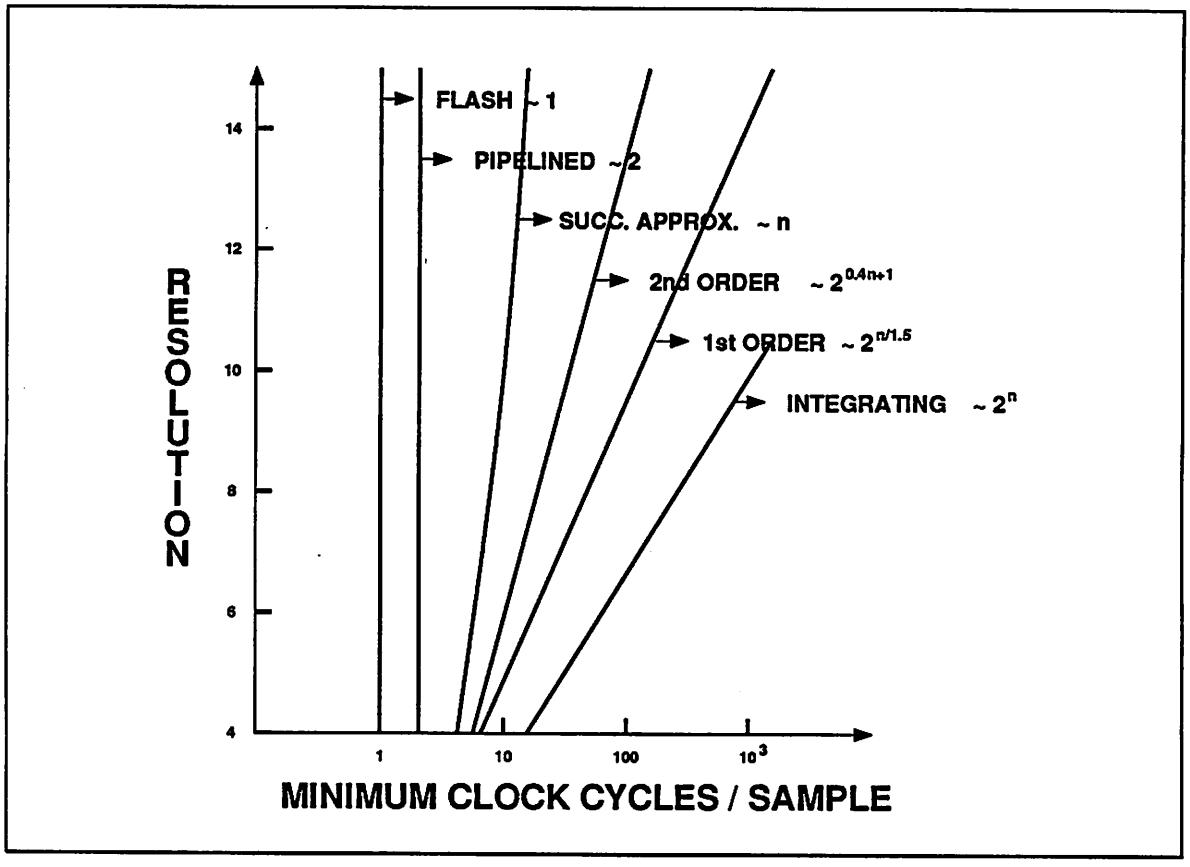

Figure 2.7.1 Architectural Speed Comparisons

The area comparison is not as straightforward as the speed comparison because, in most cases, different components are required by different architectures. Because information on the sizes of these components is not available, a similar plot for the area comparison is not given here.

CHAPTER 3

LINEARITY ERRORS AND SOLUTIONS

As in the case with most A/D converters, the pipelined A/D converter is susceptible to analog component imperfections such as offsets, gain errors, and nonlinearities. Unless these imperfections are corrected, they can limit the maximum achievable linearity and resolution to about 8-10 bits. This chapter covers the error sources that can affect the DC linearity and resolution of a conventional pipelined A/D converter implementation. The solutions to the problems will also be covered.

3.1. SUB-DAC NONLINEARITY

To illustrate the effect of the sub-DAC nonlinearity on the overall transfer characteristic of a pipelined A/D converter, a pipelined A/D converter that resolves two bits in each stage, such as the one illustrated in Figure 3.1.1, is used as an example. The sub-DAC nonlinearity in the example shown results from mismatch in the unit resistors of the sub-DACs. Only the sub-DAC nonlinearity of the 1^{st} stage is considered initially to simplify the analysis. In this example, it is further assumed (for illustration) that the nonlinearity is caused by the value of the first resistor, R_0, in the resistor string being smaller than its nominal value.

The effect of the 1^{st} stage sub-DAC nonlinearity can be studied by examining the plots of the overall digital output codes and the intermediate node signals of the 1^{st} stage vs. the original analog input signal. Figure 3.1.2 illustrates the plot of the sub-DAC output signal vs. the original analog input signal. Ideally, the staircase like plot should have uniform step sizes both in the horizontal and vertical directions. However, because of the sub-DAC nonlinearity, the step sizes are

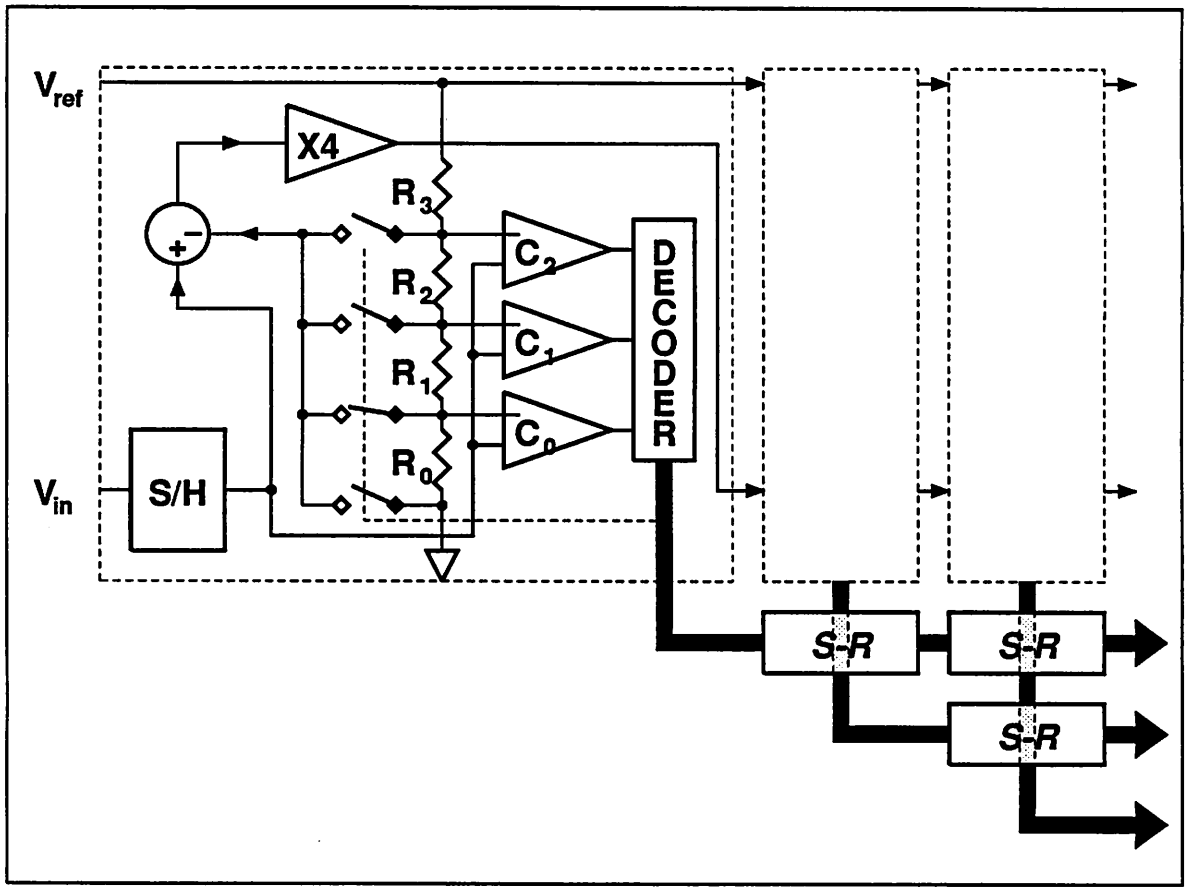

Figure 3.1.1 An Example Pipelined A/D Converter.

no longer uniform. The effect is that the residue signal, which is obtained by subtracting the sub-DAC output from the original analog input signal, no longer looks like an ideal sawtooth waveform, as illustrated in Figure 3.1.3. As a result, the range of the amplified residue signal does not fit the conversion range of the 2^{nd} stage, even if the interstage gain of the 1^{st} stage is still ideal.

When the amplified residue range is much smaller than the conversion range of the 2^{nd} stage, some of the codes near the full-scale conversion region of the 2^{nd} stage are never produced (ie. missing codes). On the other hand, when the amplified residue range is much larger than the conversion range of the 2^{nd} stage, the overall A/D converter transfer characteristic will contain large horizontal jumps, as illustrated in Figure 3.1.4.

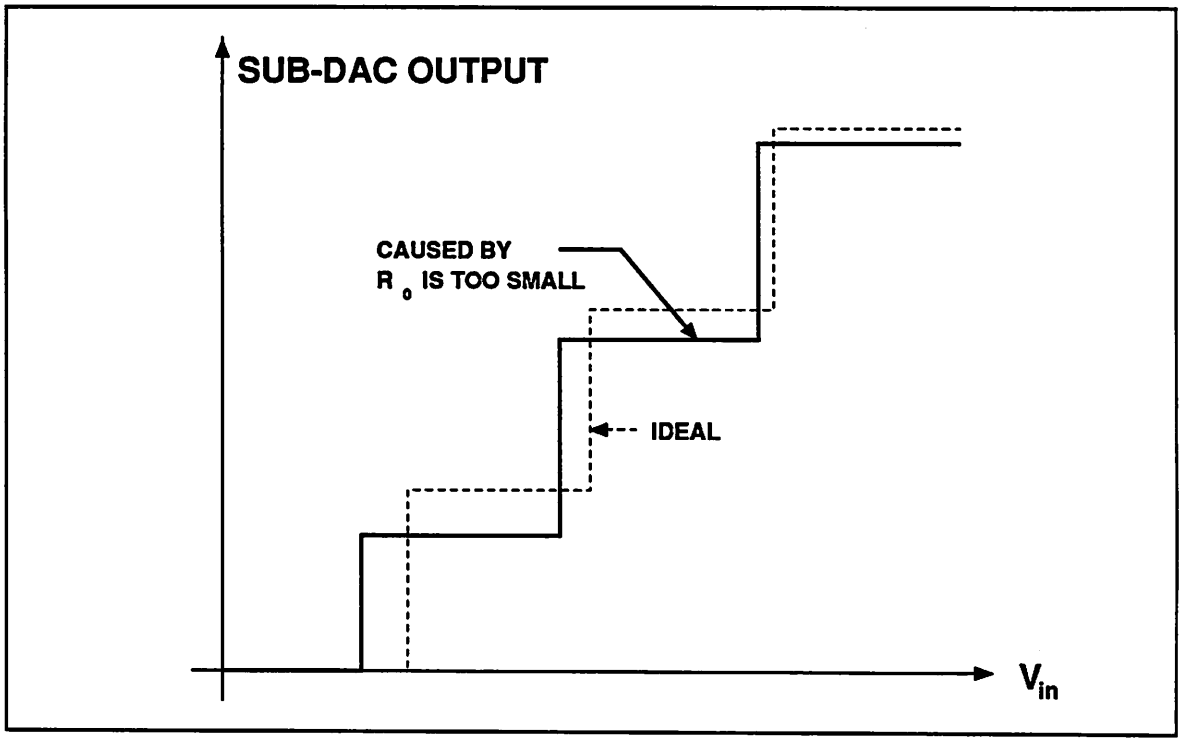

Figure 3.1.2 Plot of the Sub-DAC Transfer Function.

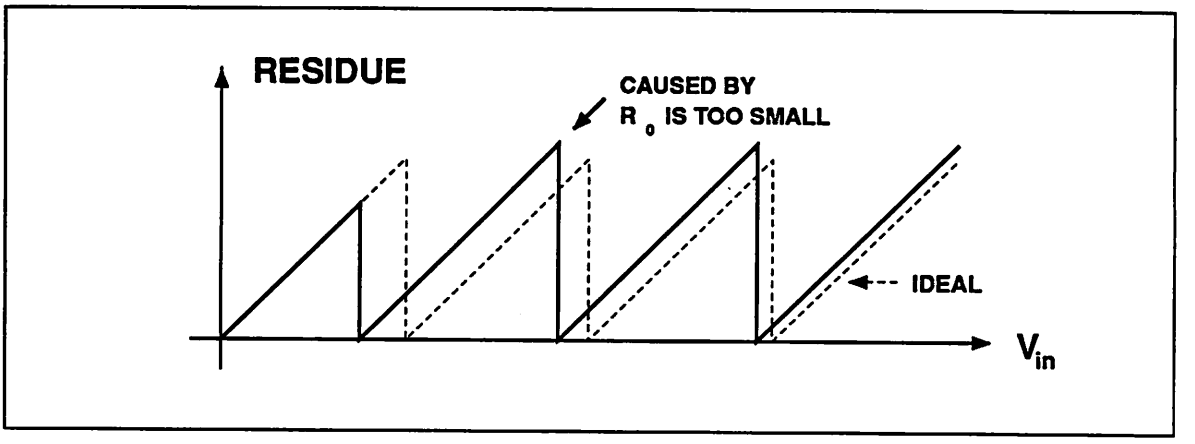

Figure 3.1.3 Plot of the Residue Signal.

Large jumps and missing codes are not usually a problem for resolutions on the order of 8-10 bits because the corresponding requirement on the sub-DAC linearity can be readily attained by relying on the inherent matching properties of resistors [20] or MOS capacitors [21,22] in typical CMOS processes. However, at a higher resolution level, especially in the range of 12 bits and

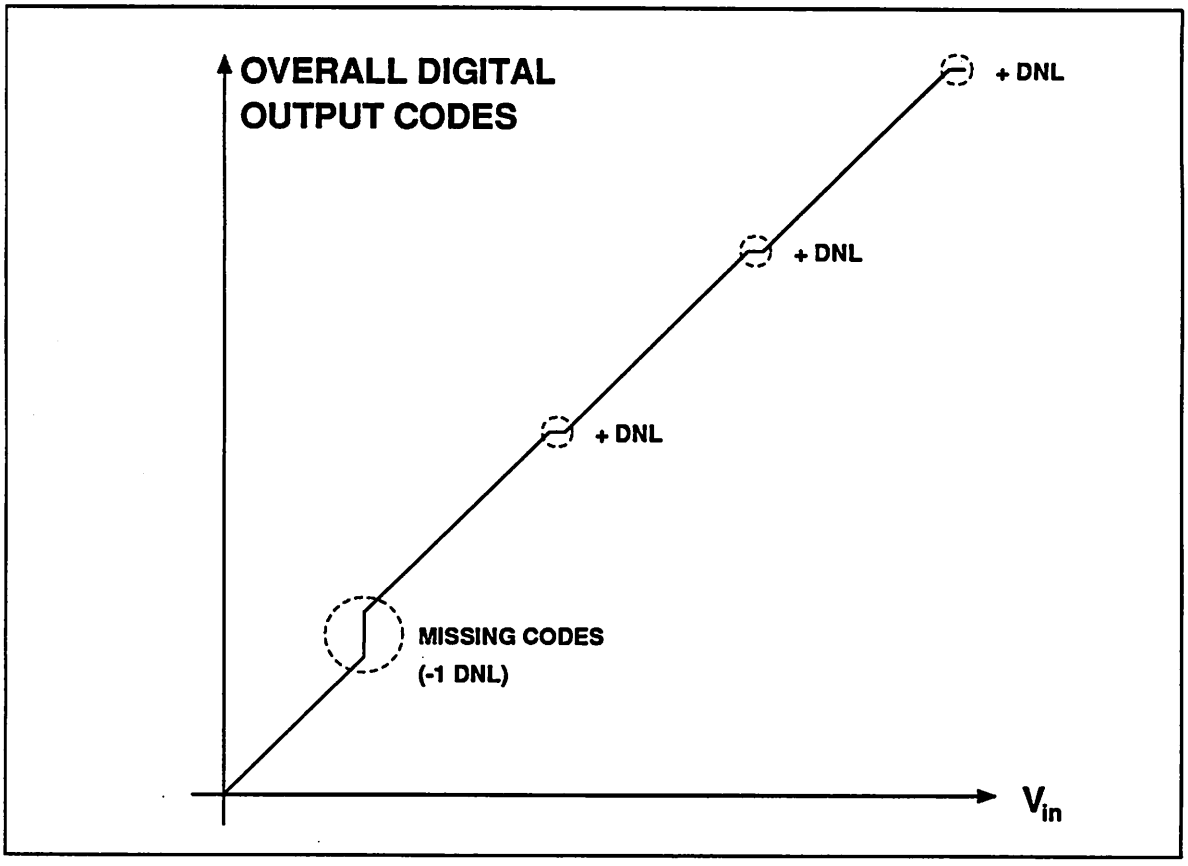

Figure 3.1.4 Plot of the Overall A/D Converter Transfer Function.

above, it may be impossible to obtain the required sub-DAC linearity without using costly trimming or self-calibrating techniques.

3.1.1. Effect of Sub-DAC Nonlinearity from the Later Stages

The sub-DAC nonlinearity from the later stages of a pipelined A/D converter can also introduce nonlinearity to the overall A/D converter transfer function. However, if enough bits are resolved in the 1^{st} stage, the effects of errors including that from the sub-DAC nonlinearity of the later stages will be negligible. This is because the errors of the later stages are scaled down by the gain of the preceding interstage amplifiers when they are referred to the input. The input referred errors can therefore be arbitrarily scaled down by increasing the interstage gain value.

3.2. INTERSTAGE AMPLIFIER GAIN ERROR

The interstage amplifier is another possible source of error in a pipelined A/D converter. The interstage amplifier is used to scale the range of the residue signal of one stage to equal the conversion range of the subsequent stages. Typically in MOS implementations, an interstage amplifier consists of an opamp, a sampling capacitor, and an integrating capacitor, as illustrated in Figure 3.2.1. Errors in the interstage amplifier can therefore be caused by either a finite opamp open-loop gain, an error in the sampling to integrating capacitor ratio, or an opamp offset voltage.

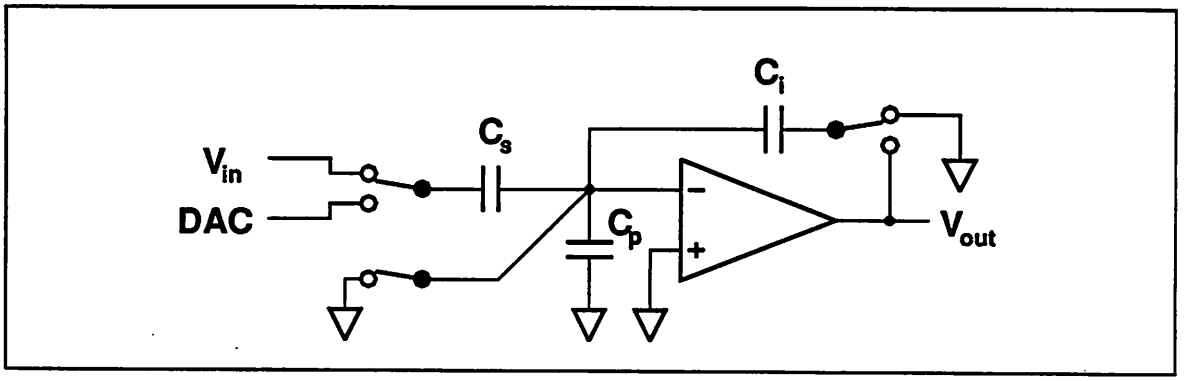

Figure 3.2.1 Typical Interstage Amplifier Implementation.

3.2.1. Effect of Finite Opamp Open-Loop Gain

Because of the finite opamp open-loop gain, the value of the interstage gain obtained from an actual implementation, $a_{o,actual}$, is somewhat lower than the ideal value, $a_{o,ideal}$. In particular,

$$a_{o,actual} = \frac{A_o C_s}{C_s + C_p + (A_o+1)C_i}, \tag{3.2.1.1}$$

and

$$a_{o,ideal} = \frac{C_s}{C_i}, \tag{3.2.1.2}$$

where C_s, C_i, and C_p are the sampling, integrating, and summing-node parasitic capacitance respectively, and A_o is the open-loop gain of the opamp. As can be seen from the nonideal gain equation, the interstage gain error is increased by a lower opamp open-loop gain and a larger

parasitic capacitance. Therefore, a minimum opamp open-loop gain requirement has to be established to keep the error that is caused by the finite opamp open-loop gain small.

It can be shown that the opamp open-loop gain requirement of a particular stage does not depend on the number of bits resolved in that stage, but only depends on the number of bits that must be resolved from that stage on. Although the required interstage gain accuracy is lowered by increasing the number of bits that are resolved in a stage, the increase in the interstage gain value causes the product of the loop-gain of the interstage amplifier and the accuracy requirement to stay approximately constant. Therefore, if the tolerable amount of error in a stage is limited to 0.25 LSB of the total resolution left to be resolved from that stage on, the minimum opamp open-loop gain requirement is:

$$A_{k,min} = 2^{n_k+2}\left[\frac{C_s}{C_i}+1\right], \qquad (3.2.1.3)$$

where n_k is the total number of bits that must be resolved from the k^{th} stage on. For an example, an opamp that has an open-loop gain of at least equal to 32k is required in the 1^{st} stage of a 13 bit pipelined A/D converter to achieve less than 0.25 LSB of error.

Although opamps that have such a high open-loop gain can be realized in a typical CMOS process, the required high open-loop gain can only be achieved at the expense of slower settling time. Typically if a higher gain is needed, longer-channel devices and/or lower-quiescent current are needed, causing a significant reduction in the bandwidth of the opamp.

3.2.2. Effect of Capacitor Ratio Error

As can be seen from equation 3.2.1.2, the accuracy of the interstage gain is limited by the accuracy of the sampling to integrating capacitor ratio, especially that of the 1^{st} stage. To reduce the effect of the ratio accuracy error on the overall linearity to less than 0.25 LSB, the error in the sampling to integrating capacitor ratio must be a factor of 4 smaller than $\frac{1}{2^{n_k}}$, where n_k is the number of bits left to be resolved from the k^{th} stage on. Therefore, if a traditional pipelined A/D

converter implementation is chosen, it is important to minimize the number of bits that must be resolved in the later stages. However, as it will be shown in the next chapter, an optimum implementation of a pipelined A/D converter requires that the number of bits resolved in each stage to be limited to no more than 3 to 4 bits typically. Therefore, a substantial number of bits are left to be resolved after the first stage. For an example, if 4 bits are resolved in the 1^{st} stage of a 13 bit pipelined A/D converter, 9 more bits must be resolved in the following stages. Therefore, a gain accuracy of around 11 bits is still needed to limit the error from the first interstage gain to 0.25 LSB. Unfortunately, such an accuracy requirement is hard to attain from component matching alone.

3.2.3. Offset Voltage of the Interstage Amplifier

The offset voltage of the interstage amplifier can also cause nonlinearity in the overall converter transfer function. The offset voltage shifts the residue signal, causing missing codes and large horizontal jumps in the overall A/D converter transfer characteristic. With the use of a technique that will be covered in detail in section 3.3, however, the opamp offset voltage can be made to only affect the input referred offset. Further discussion on how this can be done will be covered later in the section on Sample-and-Hold(S/H) error sources because offset in the residue signal can be considered as offset in the S/H circuit.

3.3. SUB-ADC NONLINEARITY

The sub-ADC nonlinearity in general results from comparator offset voltages and hysteresis, and errors in the comparison reference levels. Nonlinearity in the sub-ADC causes the sub-ADC to make a wrong estimation of the analog input signal, especially when the input signal is close to a comparison reference level. In a pipelined A/D converter, sub-ADC decision errors cause certain portions of the residue signal to lie outside the normal residue conversion range, as illustrated in Figure 3.3.1. For simplicity of the discussion, it is assumed that only the sub-ADC of the 1^{st}

stage is nonlinear. As can be seen from Figure 3.3.2, missing codes may result if no correction is performed.

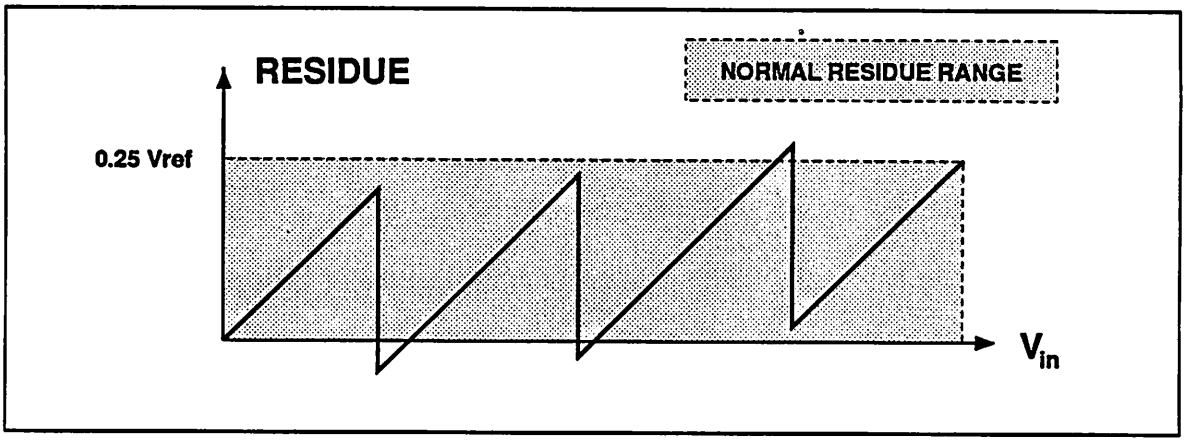

Figure 3.3.1 Plot of Residue Signal in the Presence of Sub-ADC Nonlinearity.

3.4. SAMPLE/HOLD ERRORS

The sample-and-hold(S/H) circuit is another source of errors in a pipelined A/D converter. Errors from the S/H circuit include offset, gain, and linearity error. Since errors from the S/H circuit of the 1^{st} stage are directly referred to the input, only the linearity error from the S/H of the 1^{st} stage is critical to the linearity of the overall pipelined A/D converter. If the linearity error is small, however, it does not effect the overall converter resolution. Therefore, a good S/H circuit is only needed for applications that require high linearity. Circuit design techniques for obtaining S/H circuit that has high linearity, low offset, and small gain error will be covered in a later chapter.

Errors from the S/H circuits of the later stages, with the exception of a small and constant offset voltage, can, however, cause linearity errors in the overall A/D converter transfer characteristic because they distort the residue signal. Nevertheless, errors from the later stages are scaled down by the subsequent interstage gain, and thus their effects can be made negligible by resolving more bits in the 1^{st} stage.

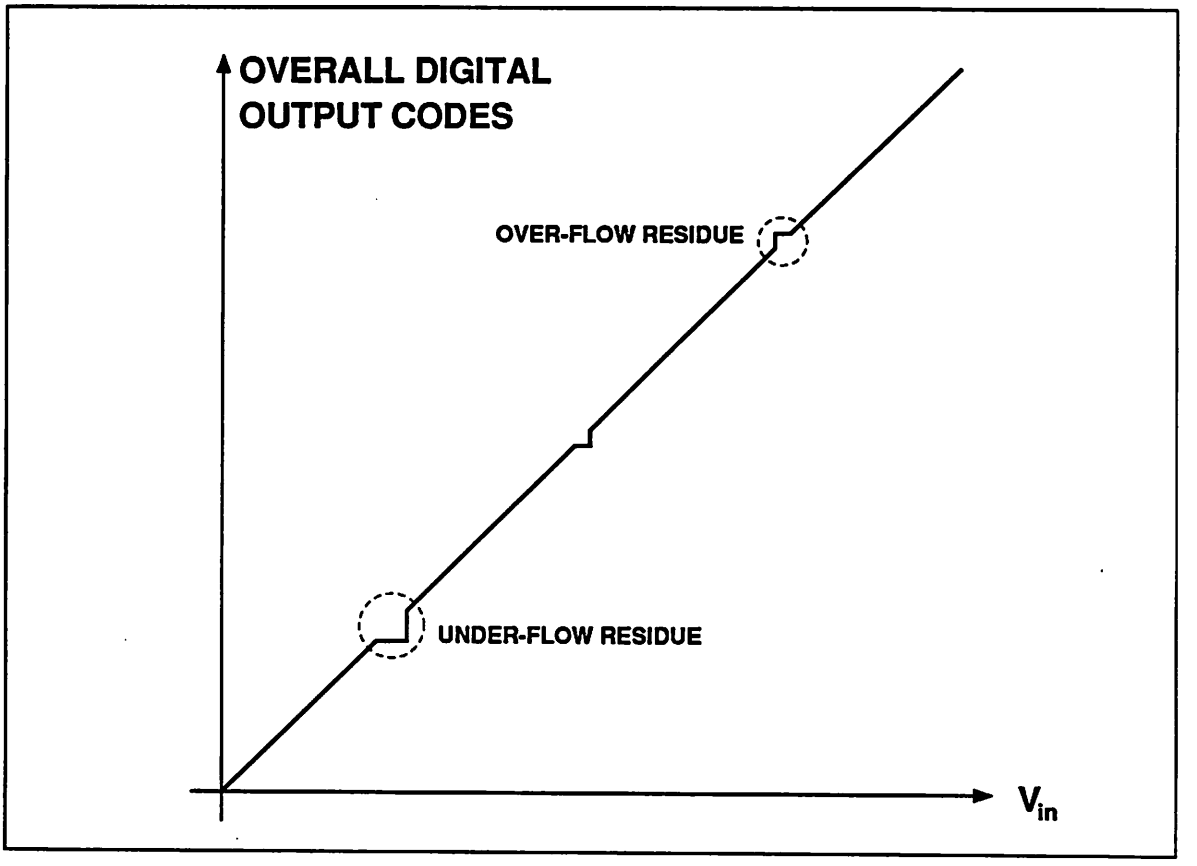

Figure 3.3.2 The Corresponding Plot of the Overall ADC Transfer Function.

3.5. OTHER SOURCES OF ERRORS

Other sources of errors that have not been discussed so far are mostly related to dynamic and speed related imperfections. These types of errors are mostly implementation dependent, so they will be covered in the detail implementation. It is important to note, however, that only the 1^{st} stage S/H circuit is subjected to large and fast variations in the input signal. Therefore, as long as the rest of the pipelined converter can work at the required speed, the only factor that limits the frequency response of a pipelined A/D converter is the S/H circuit of the 1^{st} stage.

3.6. NEW SOLUTION TO THE D/A NONLINEARITY

If the main objective is to obtain high resolution (differential linearity) but not high (integral) linearity, however, a solution without the use of trimming or calibration is possible. Previously,

A/D and D/A converters with high resolution (differential linearity) but limited (integral) linearity have been shown to be possible to obtain even in the presence of large component matching errors by using the so-called segmentation techniques.[7,8,9] The realization of this technique in a pipelined A/D converter is obtained by making the conversion range of any stage equal to the amplified residue range of the stage in front of it, as opposed to making the amplified residue range of a stage equal to the conversion range of the stage after it. The latter can only be obtained using trimming or self-calibrating correction techniques. In the example used here, this means that the conversion range of the second and subsequent stages must be made equal to the amplified residue range of the 1^{st} stage, since errors are assumed to come from the 1^{st} stage only.

This approach, which is called the reference-feedforward-correction technique, is much easier and cheaper to implement because it can be implemented by only modifying the reference voltage of the second and subsequent stages to equal the amplified sub-DAC segment voltage that corresponds to the current residue signal range. In the example used here, there are four possible sub-DAC segment voltages that can be used for the new reference voltage of the 2^{nd} stage because 2 bits are resolved in the 1^{st} stage. However, only one of them is correct and a different segment voltage might be needed for a different input signal. For example, when the analog input signal is in the range of the 1^{st} segment of the sub-DAC, the 1^{st} segment voltage of the sub-DAC is the only voltage that should be amplified and sent to the 2^{nd} stage as the new reference voltage. In general, the i^{th} sub-DAC segment voltage must be used whenever the original analog input signal is in the range of the i^{th} sub-DAC segment.

The correct selection of the sub-DAC segment voltage is possible because the same resistor string is used in both the sub-ADC and the sub-DAC. As long as offset voltages of the comparators used in the sub-ADC are canceled out, the resultant digital output codes of the sub-ADC can be used to directly select the correct sub-DAC segment voltage. If, however, large comparator offsets are still present, or if different resistor strings are used in the sub-DAC and sub-ADC, or if

capacitors are used in the sub-DAC, the resultant digital output codes of the sub-ADC must first be corrected before the sub-DAC segment selection is performed because the selection may be incorrect when the input signal is close to the segment boundaries. The detail implementation of this correction will be covered later.

The effect of the reference-feedforward correction on the linearity of the overall converter in the presence of the sub-DAC nonlinearity is shown in Figure 3.6.1. Because the nonlinearity of the sub-DAC is not actually corrected, the transfer function of the overall converter is at best as linear as that of the sub-DAC used in the 1^{st} stage. To achieve the highest possible linearity, it is therefore necessary to use components that have the best matching properties, such as capacitors instead of resistors because capacitors usually match better than that of resistors in standard CMOS processes. Based on previous published reports on component matching, an overall linearity of around 10 bits can be readily achieved using capacitors,[22] as opposed to only around 8 bits using resistors.[20] Layout techniques that can potentially improve the linearity will be discussed in a later chapter.

3.6.1. Speed Penalty

As the corrected reference voltage depends on the value of the analog input signal being converted, the conversion speed of a pipelined A/D converter that uses the reference-feedforward correction will be lower than that of the one that does not. In particular, because it takes two time slots to sample and amplify a sub-DAC segment voltage during the reference-feedforward-correction phase, the effective A/D conversion time is increased by that amount. If flash sub-ADCs are used, reference-feedforward correction will increase the conversion time by a factor of two. The reason is that only two time slots are normally needed in the fastest pipelined A/D converter implementation. However, if slower sub-ADCs are used, the increase in the conversion time will be less.

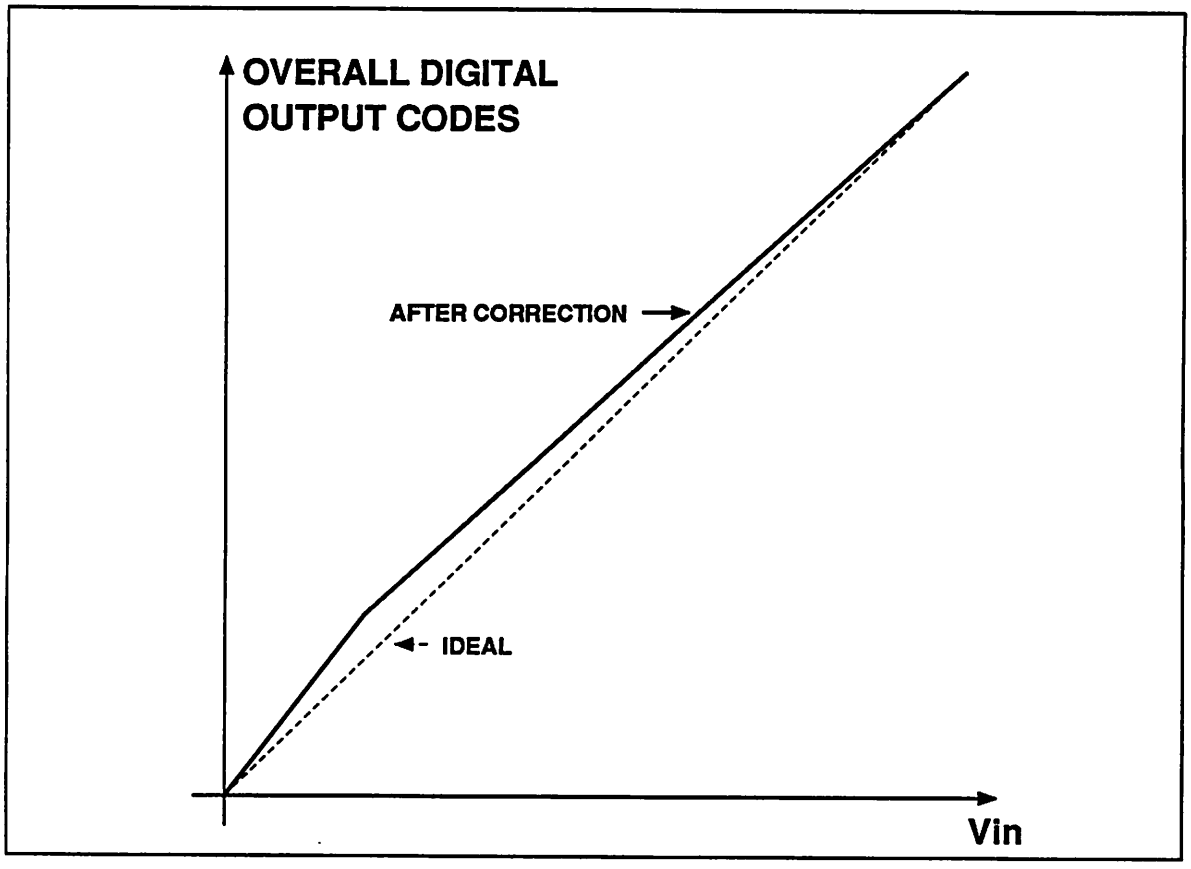

Figure 3.6.1 Plot of the Overall A/D Converter Transfer Function After Correction.

3.7. SOLUTION TO THE GAIN ERROR

Fortunately, the effect of the interstage gain error can be canceled out if the reference-feedforward correction is used. If both the residue and reference signal are amplified through the same interstage amplifier, the full scale amplified residue signal of a stage that uses the reference-feedforward correction will always be equal to the full scale conversion range of the next stage. In fact, the effect of any slow variation in the opamp open-loop gain, such as caused by temperature drift, is automatically canceled out because the new reference voltage is always obtained immediately after the residue signal. Therefore, opamps that are optimized for a high frequency response can be used without sacrificing the overall A/D converter linearity.

3.7.1. Effect of Offset of the Interstage Amplifier

The offset voltage of the interstage amplifier will cause nonlinearity in the overall converter transfer function if the reference-feedforward correction is used. This is because an offset voltage in the reference signal is effectively equivalent to a gain error, while an offset voltage in the residue signal is not. The reference signal path will therefore have a different gain than that in the residue signal path. Consequently, if the reference-feedforward correction is used, the offset voltage of the interstage amplifier has to be canceled out.

3.8. SOLUTION TO THE A/D NONLINEARITY

The sub-ADC nonlinearity in the first few stages of a pipelined A/D converter is fortunately not fundamental in the sense that it can be perfectly canceled out using simple circuit design techniques. As long as the sub-DAC and the interstage gain are ideal, the only error that is introduced to the residue signal is portions of the residue signal are shifted by an amount that is equal to an integer multiple of the sub-DAC segment voltages. Therefore, if the conversion range of the subsequent stages is expanded to cover the shifts in the residue signal, a correct conversion of the residue signal can still be performed.

The extra bits that are obtained from the extended residue conversion range can then be used to correct the resultant digital output code obtained from the earlier stage using a technique that is usually referred to as the digital-error correction. [23, 16, 24, 25] The technique works as follow. If the residue signal is detected to be in the overflow conversion range, the resultant digital output code from the earlier stage is incremented because the residue signal can only be too big if the sub-ADC believes that the analog input signal is lower than what it actually is. On the other hand, if the residue signal is detected to be in the underflow conversion range, the resultant digital output code of the earlier stage is decremented. The amount of this digital correction depends on the size of the error the sub-ADC makes, or simply the nonlinearity of the sub-ADC.

The degree of the sub-ADC nonlinearity in a typical implementation is fortunately much smaller compared to the quantization error of the sub-ADC because of the small number of bits that are typically resolved in each stage. Therefore, only small portions of the residue signal around the sub-ADC decision levels are shifted. Furthermore, they are shifted by at most one sub-DAC segment voltage as can be seen in Figure 3.8.1. Consequently, only a small portion of the conversion range of the subsequent stages must be expanded, and the resultant digital output code of the earlier stage must be incremented or decremented by only one count.

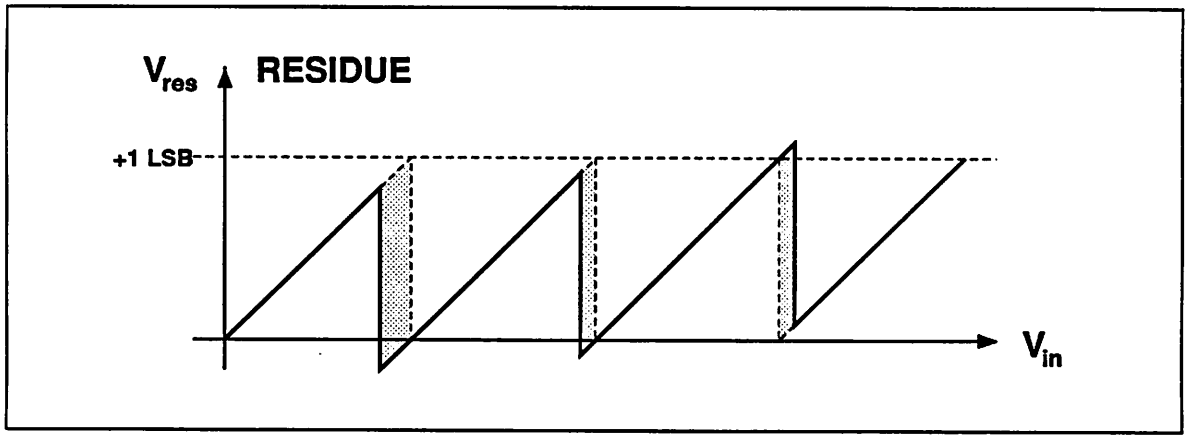

Figure 3.8.1 Shifting in the Residue Signal because of Sub-ADC Nonlinearity.

Because it is more convenient to increase the residue conversion range by a multiple factor of 2, and because both ends of the residue conversion range must be extended, effectively 2 extra bits are needed to implement the digital-error correction normally. To keep the number of additional bits required to implement the digital-error correction to only one bit, it is necessary to shift the residue signal down (by shifting the sub-DAC up) so that random shifting in the sub-ADC can be contained by only doubling the residue conversion range. An alternative solution is to intentionally introduce a known shift in the sub-ADC. Since the shifts in the sub-ADC are random, the optimum amount of intentional shift in either the sub-DAC or sub-ADC should be set to half of the segment size.

3.8.1. Sub-DAC Shifting

Shifting the sub-DAC levels up by half of the segment size causes the normal residue conversion range to be shifted down by an equal amount, as illustrated in Figure 3.8.1.1. A nonlinearity of the sub-ADC up to +/- 0.5 LSB can therefore be tolerated because the residue signal will still be in the conversion range of the 2^{nd} stage if the conversion range of the 2^{nd} stage is doubled or if the interstage gain is halfed, as illustrated in Figure 3.8.1.2. Digital-error correction is then performed by incrementing and decrementing the previous digital output code by one count, as explained earlier.

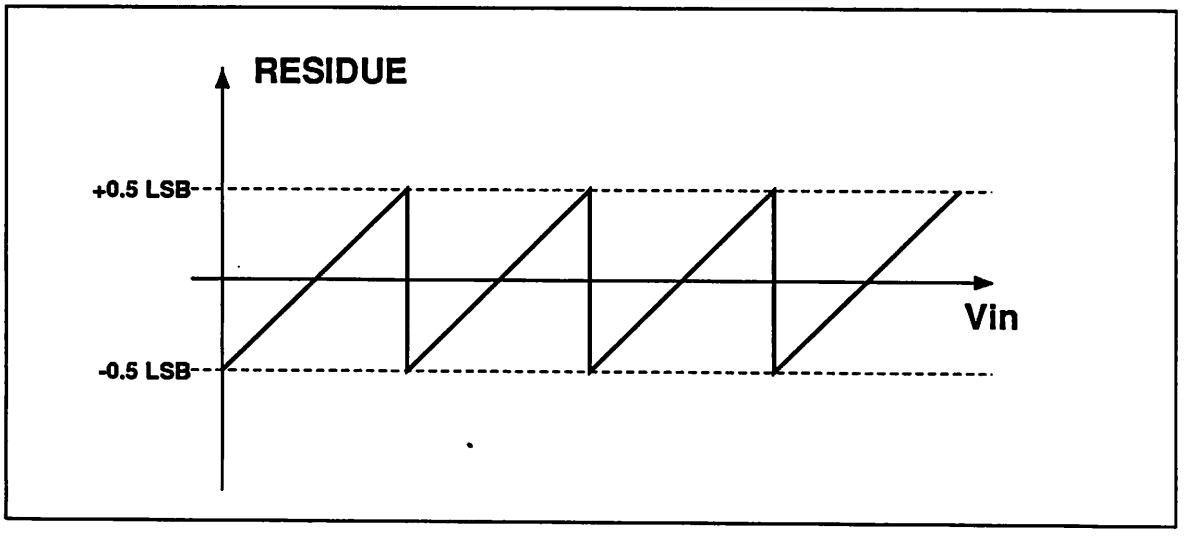

Figure 3.8.1.1 Normal Residue Signal Range with Sub-DAC Shifting .

3.8.2. Sub-ADC Shifting

A slightly better way of implementing the digital-error correction is to shift the sub-ADC decision levels instead of the sub-DAC output levels. By shifting the sub-ADC up or down by half of the segment size, decision errors are intentionally made during half of the residue range under the ideal condition, as illustrated in Figure 3.8.2.1. Up to +/- 0.5 LSB of the sub-ADC nonlinearity can be tolerated this way because the direction of the intentional error will not change, as can be seen in Figure 3.8.2.2. Therefore, digital-error correction can still be performed.

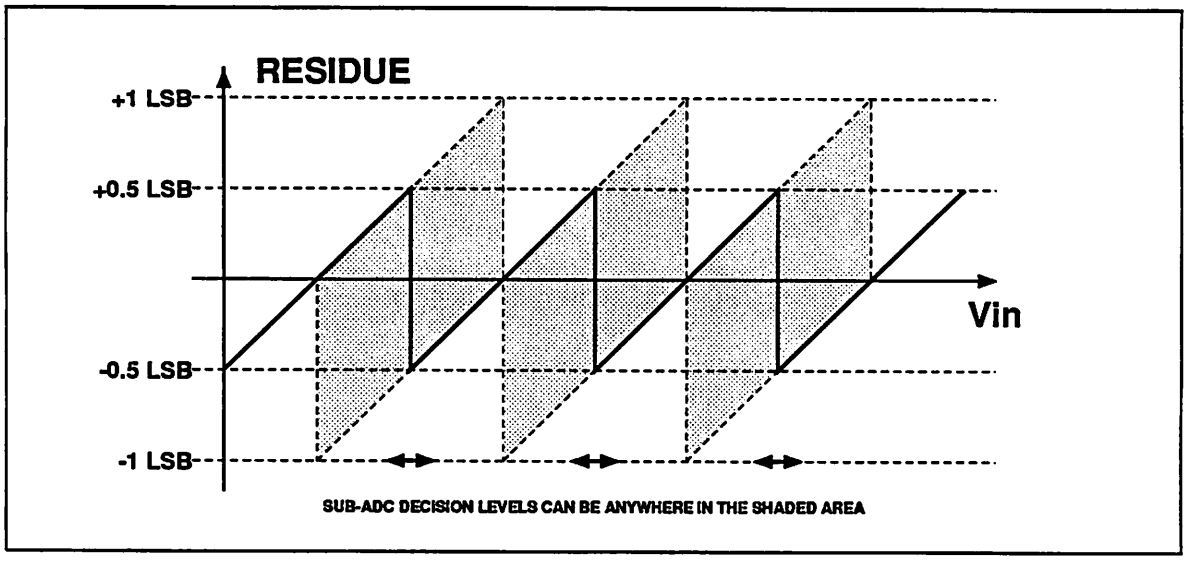

Figure 3.8.1.2 Residue Signal Range with Sub-DAC Shifting and Sub-ADC Nonlinearity.

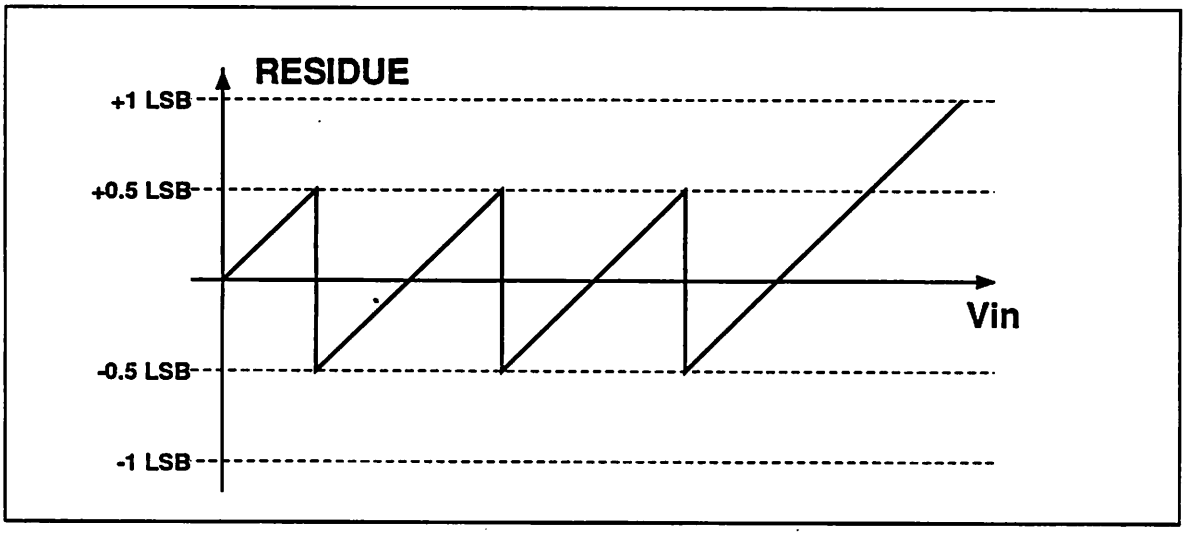

Figure 3.8.2.1 Residue Signal Range with Negative Sub-ADC Shifting.

The disadvantage of shifting the sub-ADC levels is that under the ideal condition, the inter-stage amplifier has to be able to swing to twice the full scale amplitude of that in the sub-DAC shifting in the same amount of time. This can be solved easily, fortunately, by adding another comparator in the sub-ADC. The major advantage of shifting the sub-ADC levels instead of the sub-DAC levels is that only an incrementer or decrementer, instead of both, is needed.

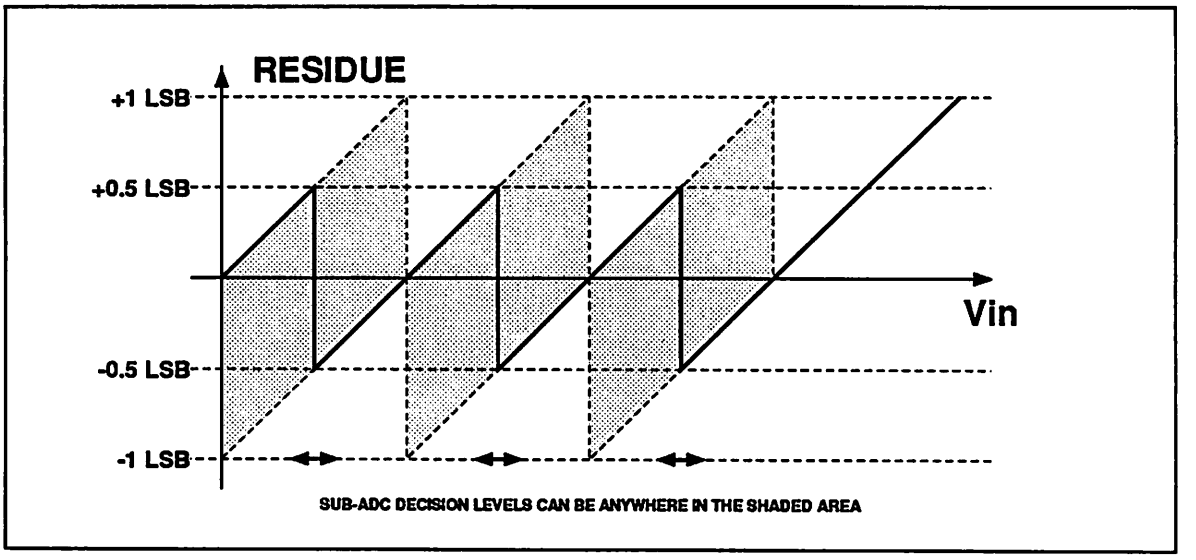

Figure 3.8.2.2 Residue Signal Range with Negative Sub-ADC Shifting and Nonlinerity.

3.8.3. Limitation of Digital-Error Correction

The reason that it is possible to increment or decrement the resultant digital output code to get the correct code is that the residue signal is assumed to shift by exactly one ideal step size of the sub-DAC. If the sub-DAC is nonlinear (nonideal), however, digital-error correction will not work. Trimming has been universally used to overcome this problem in most board-level high-resolution pipelined A/D converter.

3.8.4. Reference-Feedforward and Digital-Error Correction

Digital-error correction can still be used even if the sub-DAC is nonlinear, if the reference-feedforward correction discused earlier is also used. There is a complication, however, when both techniques are used at the same time. This is because correct selection of the sub-DAC segment voltages can only be performed if the sub-ADC tracks the sub-DAC perfectly. Since a shift in either the sub-DAC or sub-ADC levels is necessary to implement the digital-error correction, the resultant sub-ADC output code will sometimes point at a wrong sub-DAC segment voltage.

In order to make the reference-feedforward correction work properly, the digital output

code of the sub-ADC must be corrected first before the sub-DAC segment selection is performed. Unfortunately, digital-error correction cannot be performed unless the sub-DAC is ideal. To overcome this problem, the resultant digital output code can be temporarily corrected by looking at the sign of the residue signal. Small mistakes in this temporary correction step are acceptable since they can only happen when the residue signal is closed to zero. The original digital output code can then be properly corrected at the end of the pipelined stages, where errors are much less important than that in the earlier stages.

3.8.5. Effect of Offset Voltage under Digital-Error Correction

The offset voltage of the S/H from the later stages can normally cause nonlinearity in the overall converter transfer function, but if the digital-error correction is used, the offset voltage is scaled down and referred to the input. The reason is that the S/H offset voltage of the 2^{nd} stage, for an example, is actually equivalent to the offset voltages of the 1^{st} stage sub-ADC and S/H, but scaled down by the interstage gain factor, as illustrated in Figure 3.8.5.1. The sub-ADC offset voltage can then be corrected by the digital-error correction, while the S/H offset voltage is directly referred to the input. For applications that require a very small input-referred offset voltage, offset cancellation techniques such as the auto-zero offset cancellation [26] technique can also be used.

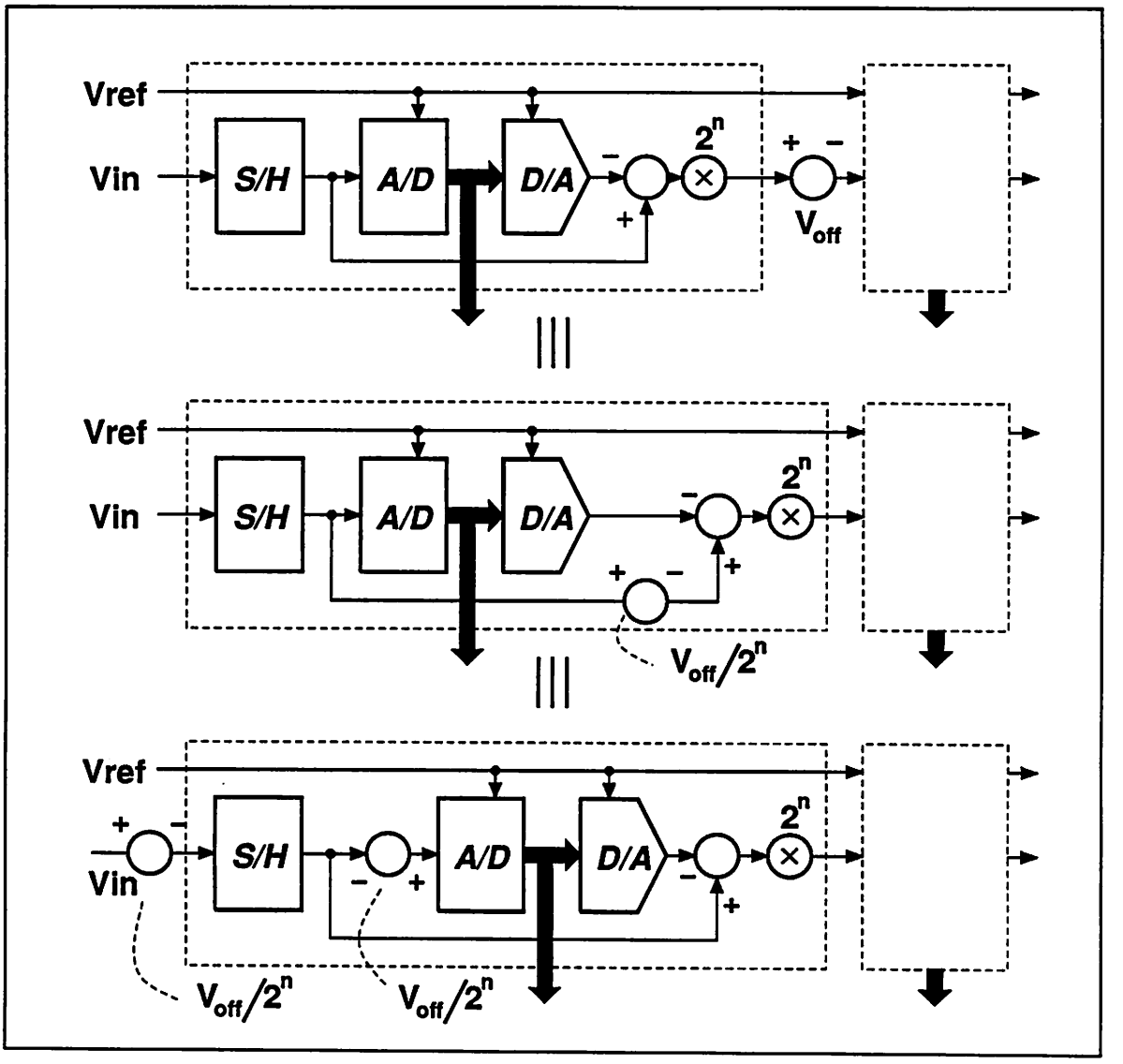

Figure 3.8.5.1 Equivalent Representations of Offset Voltage of the 2^{nd} Stage S/H.

3.9. SUMMARY

As can be seen from the preceding discussion, nonlinearity in the sub-DAC is the major limiting factor in obtaining a high-resolution pipelined A/D converter. The solutions to this error are trimming, self-calibrating, or reference-feedforward-correction techniques. There is another class of solutions [13, 14, 27] that are not discussed in this dissertation. These solutions rely on using 1-bit stages to ensure inherent linearity on the sub-DAC.

CHAPTER 4

SPEED AND AREA OPTIMIZATION

In this chapter, high-level design considerations for maximizing the conversion speed of a pipelined A/D converter will be discussed. An equally important topic that will be discussed later in this chapter is the consideration for a minimum area utilization in applications that do not require a very high conversion speed. In fact, the prototype pipelined A/D converter that will be described in a later chapter is designed not for a maximum possible conversion speed but for a minimum silicon area and power dissipation when implemented using an inexpensive 3-μm CMOS processing technology.

4.1. MAXIMUM CONVERSION SPEED

As mentioned in chapter 2, the conversion speed of a pipelined A/D converter is limited mostly by the speed of the input sampling and the residue amplification steps. It is thus important to minimize the time spent on these two steps in order to achieve a maximum conversion speed.

Consider a case in which the interstage amplifier is constructed from a single-stage opamp, such as a folded-cascode opamp. A single-stage opamp is chosen in this discussion because it usually can be made faster than a two-stage opamp. A typical set up of such an interstage amplifier is shown in Figure 4.1.1. It can be shown that the bandwidth of the interstage amplifier, ω, is:

$$\omega = f_b \, \omega_u \tag{4.1.1}$$

where ω_u is the unity-gain bandwidth of the opamp, and f_b is the amount of feedback at the unity-gain frequency (f_b is usually referred to as the feedback factor in feedback theory). The

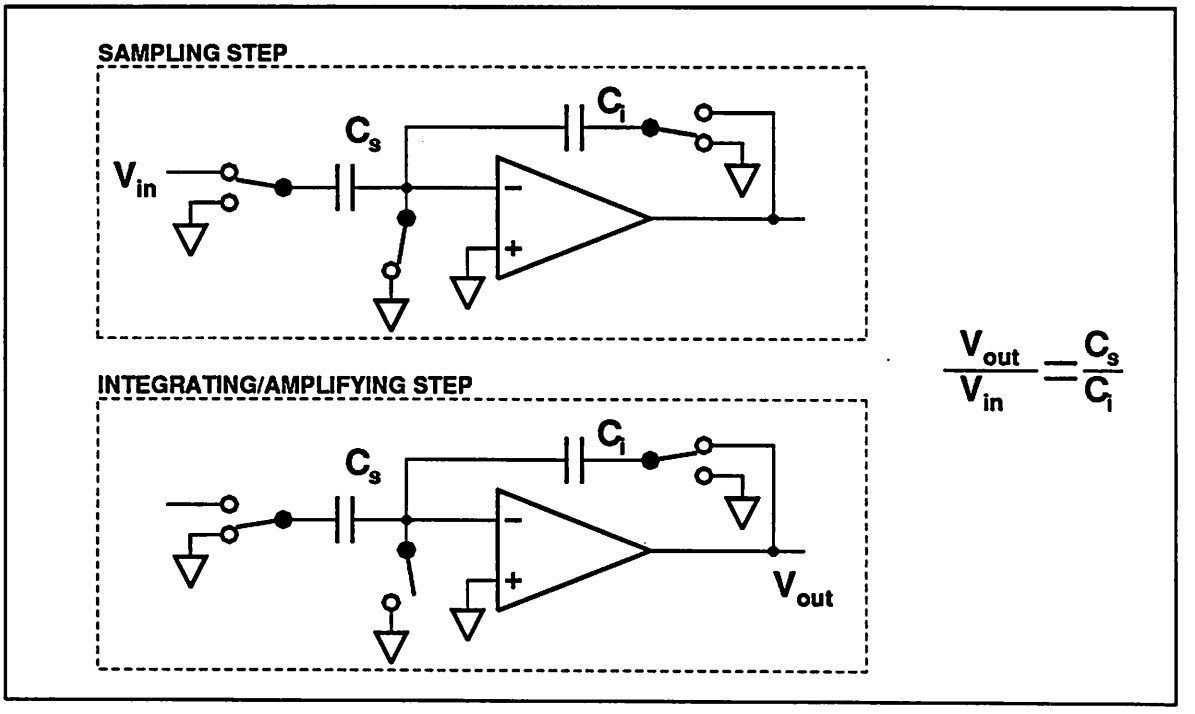

Figure 4.1.1 Typical Implemenation of an Interstage Amplifier in CMOS Technology.

unity-gain bandwidth of a single-stage opamp is given by:

$$\omega_u = \frac{g_m}{C_L}, \tag{4.1.2}$$

where g_m is the input transconductance of the opamp, and C_L is the total output loading capacitance. Assuming for simplicity that identical pipelined A/D converter stages are employed, the total output loading capacitance can be calculated from Figure 4.1.2 to give:

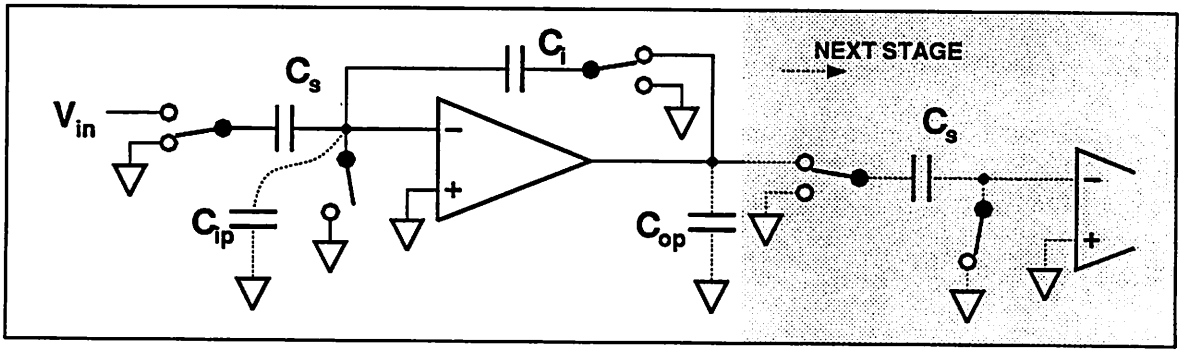

Figure 4.1.2 Parasitic Capacitances in Interstage Amplifier.

$$C_L = C_{op} + C_s + \frac{(C_s + C_{ip})\, C_i}{C_s + C_{ip} + C_i} \tag{4.1.3}$$

where C_s, C_i, C_{ip}, and C_{op} are the sampling, integrating, input parasitic, and output parasitic capacitance respectively. Similarly the feedback factor, f_b, can be calculated from Figure 4.1.2 to give

$$\begin{aligned} f_b &= \frac{C_i}{C_s + C_i + C_{ip}} \\ &= \frac{1}{1 + \frac{C_s}{C_i} + \frac{C_{ip}}{C_i}} \\ &= \frac{1}{1 + 2^k + \frac{C_{ip}}{C_i}}, \end{aligned} \tag{4.1.4}$$

where 2^k is the interstage gain value. As can be seen from these equations, a significant loss of bandwidth can result if too many bits are resolved in each stage because the feedback factor decreases almost exponentially with the number of bits that are resolved in each stage. Therefore, as few bits as possible should be resolved in each stage in order to achieve a maximum possible conversion speed. In other words, only 1 bit (or 2 bits with digital-error correction) should be resolved in each stage unless the reference voltages for the subsequent stages are scaled down so that a smaller gain value can be used in the interstage amplifier to reduce the settling time.

In order to further maximize the bandwidth of the interstage amplifier, it is necessary to maximize the unity-gain bandwidth of the opamp, ω_u. This can be achieved by maximizing the input transconductance and minimizing the output parasitic capacitance of the opamp. Unfortunately, ω_u cannot be maximized independently of f_b without affecting the latter. In an actual implementation of the interstage amplifier, the input parasitic capacitance, C_{ip}, of the opamp would normally increase with g_m. Even though the unity-gain bandwidth of the opamp can be increased by using a large g_m, the corresponding high input parasitic capacitance offsets the advantage of a large g_m so that the overall bandwidth of the interstage amplifier may not be increased.

The bandwidth of the interstage amplifier, ω, can only be correctly maximized by simultaneously optimizing the sizes and current levels of the transistors in the opamp. Instead of first specifying the values of the sampling and integrating capacitor and optimizing the opamp for these values, an alternative approach is proposed in this section. The optimization is divided into two steps. The first step is to design the best possible opamp that has a reasonable g_m using standard circuit design techniques. An exact g_m is not yet needed because if it turns out that the resulting opamp is too large or too small, the opamp can then be scaled accordingly. The most important goal of this step is to minimize C_{ip} and C_{op} for the given g_m, output voltage swing, power supply condition, minimum open-loop gain requirement, etc.

The second step is to maximize the overall interstage bandwidth by selecting an optimum value of C_i (or C_s, which is related by the closed-loop gain). This can be obtained by setting the partial derivative of ω with respect to C_i to zero. The resulting values of C_i and C_s can then be compared to the desired values, which are usually determined by the $\frac{kT}{C}$ [28] noise limitation and/or the capacitor matching requirement. If the optimum integrating and sampling capacitor values are found out to be smaller than the desired values, the initial opamp design can be scaled up to maintain the maximum closed-loop bandwidth. On the other hand, if the resulting optimum capacitor values are found out to be larger than the desired values, the initial opamp design can be scaled down to save area and power dissipation.

For the case where identical 1-bit pipelined A/D converter stages are used, the partial derivative of ω with respect of C_i is:

$$\frac{\partial \omega}{\partial C_i} = \frac{g_m}{8C_i^2 + 3C_{op}C_i + 3C_{ip}C_i + C_{ip}C_{op}} - \frac{g_m C_i(16C_i + 3C_{op} + 3C_{ip})}{(8C_i^2 + 3C_{op}C_i + 3C_{ip}C_i + C_{ip}C_{op})^2} = 0 \qquad (4.1.5)$$

Solving the above equation, the optimum value of the integrating capacitor for minimum settling time is:

$$C_i = \sqrt{\frac{1}{8} C_{ip} C_{op}} \tag{4.1.6}$$

The maximum closed-loop bandwidth can be solved by substituting the resulting optimum integrating capacitor value back into equation 4.1.1:

$$\omega_{max} = \frac{g_m}{C_{ip}} \frac{1}{4\sqrt{2\rho + 3\rho + 3}}, \tag{4.1.7}$$

where ρ is given by:

$$\rho = \frac{C_{op}}{C_{ip}} \tag{4.1.8}$$

Assuming that the input parasitic capacitance is mostly due to the gate capacitance of the input transistor of the opamp, the maximum bandwidth of the interstage amplifier can be written as:

$$\omega_{max} \approx 2\pi f_{T,input} \frac{1}{4\sqrt{2\rho + 3\rho + 3}}, \tag{4.1.9}$$

where $f_{T,input}$ is the f_T of the input transistor of the opamp. Plots of the bandwidth of the interstage amplifier and the bandwidth reduction factor (which is defined as the ratio of f_T over the bandwidth of the interstage amplifier) as a function of ρ are shown in Figure 4.1.3. Depending on the exact ratio of the output over the input parasitic capacitance, the maximum closed-loop bandwidth that can be obtained from a typical CMOS processing technology can be anywhere between 10 to 20 times smaller than the f_T of the input transistor of the opamp. Because the f_T that can be obtained from a typical 3-μm CMOS processing technology under the usual analog biasing conditions is less than 600 MHz, and because the bandwidth reduction factor is rather high, it is hard to build a video-rate pipelined A/D converter using a 3-μm CMOS processing technology.

4.2. ALTERNATIVE INTERSTAGE AMPLIFIER CONFIGURATION

One way to further maximize the bandwidth of an interstage amplifier is to reduce the number of unit capacitors to further increase the feedback factor. In particular, by eliminating the integrating capacitor and by using one of the unit sampling capacitors as the integrating capacitor during the integrating/amplifying phase, the feedback factor can be increased appreciably. This is especially

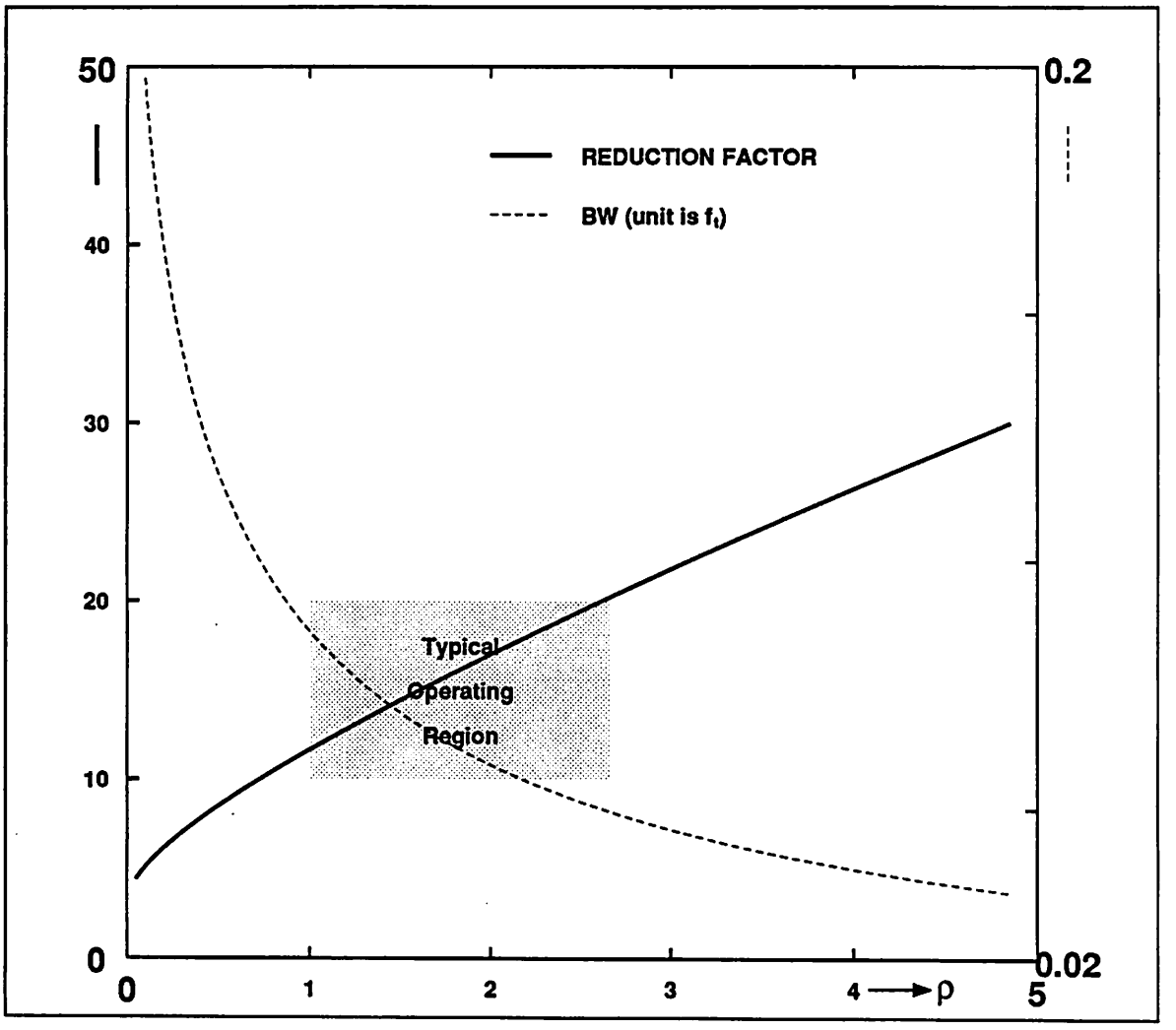

Figure 4.1.3 Effects of ρ on the Bandwidth of the Interstage Amplifier.

attractive when only 1 bit (or 2 bits with digital-error correction) is resolved in each stage, as the increase in the feedback factor is smaller when more bits are resolved in each stage. An example of such an interstage amplifier configuration for the 1-bit stages is shown in Figure 4.2.1. In such a case, the optimum value of the unit capacitor is given by:

$$C = \sqrt{\frac{1}{5} C_{ip} C_{op}}, \tag{4.2.1}$$

and the corresponding maximum obtainable bandwidth is:

$$\omega_{max} = \frac{g_m}{C_{ip}} \frac{1}{2\sqrt{5\rho + 2\rho + 3}} \tag{4.2.2}$$

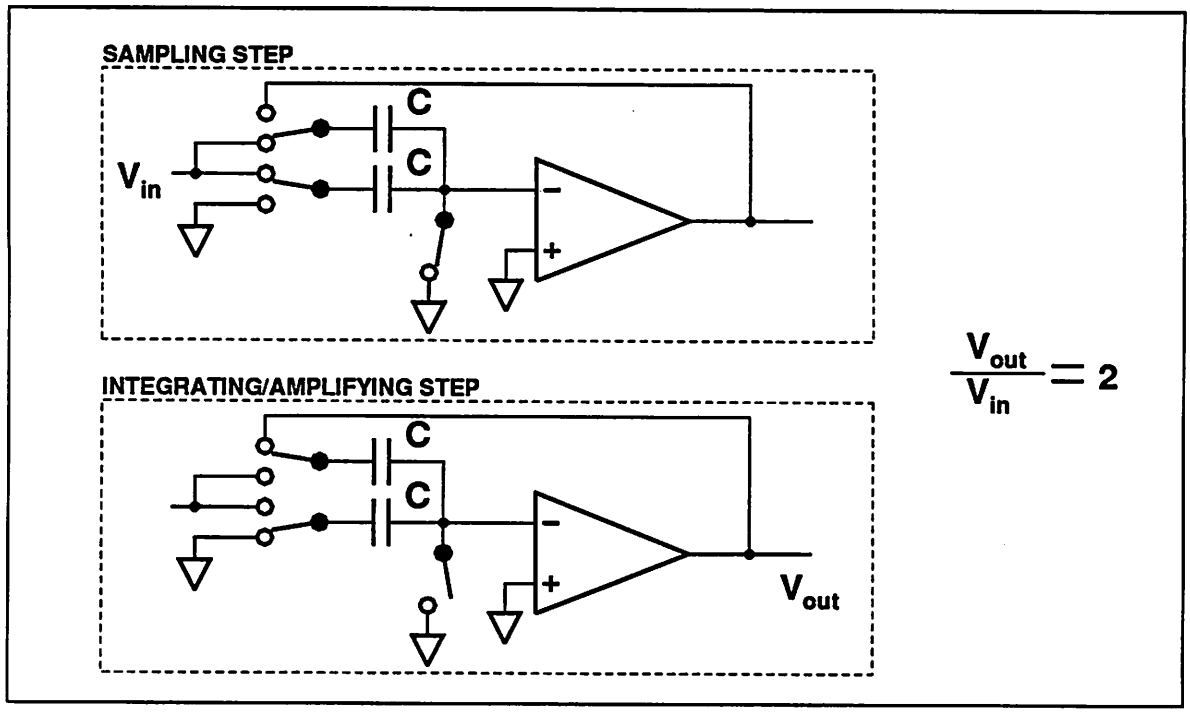

Figure 4.2.1 Faster Interstage Amplifier.

$$\approx 2\pi f_{T,input} \frac{1}{2\sqrt{5\rho + 2\rho + 3}} \tag{4.2.3}$$

As a comparison to the interstage amplifier configuration shown earlier, the bandwidth of the interstage amplifier and the reduction of bandwidth from that of the f_T of the input transistor as a function of ρ are plotted in Figure 4.2.2. Compared to the results shown in Figure 4.1.3 for ρ in the range of 1 to 2, the improved interstage amplifier configuration is typically about 25% faster.

4.3. MINIMUM AREA UTILIZATION

To reduce the fabrication cost of a pipelined A/D converter, it is necessary to reduce the area utilization as much as possible. Obviously, the area will depend on the processing technology. In the case where the processing technology is the limiting factor, nothing much can be done aside from using the fastest pipelined A/D converter configuration as described in the previous section, and optimizing the opamp size for the minimum acceptable value of the unit capacitors. On the other hand, when the technology is fast enough for the target applications, there are other approaches to

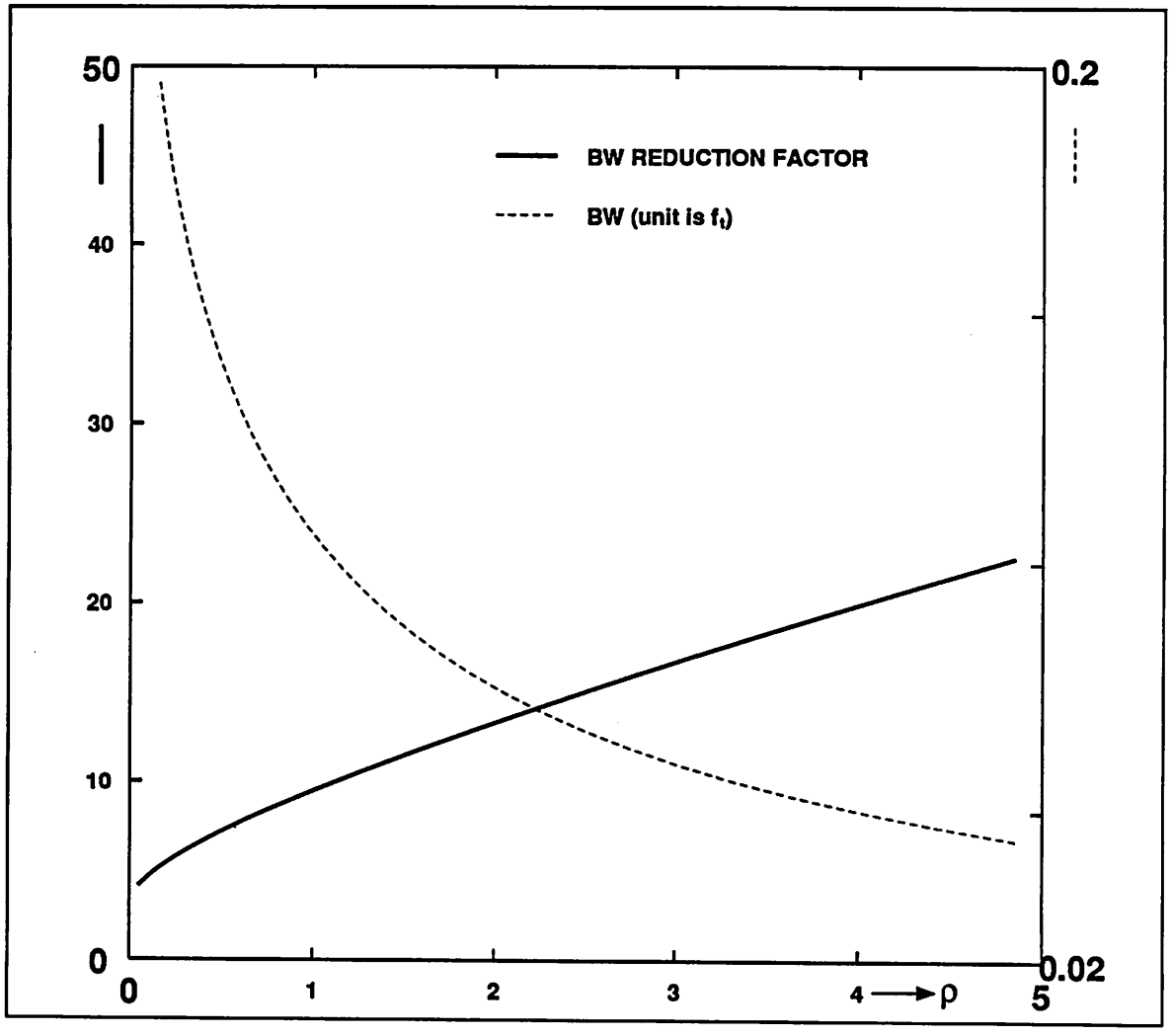

Figure 4.2.2 Effects of ρ on the Bandwidth of the Interstage Amplifier.

minimize the overall converter area besides reducing the size of the opamps. For an example, for applications that require medium to high-conversion speed, it is possible to minimize the area by having each stage resolves more bits to reduce the number of stages and therefore the number of opamps required. For even slower applications, it is possible to further minimize the area by using successive-approximation sub-ADCs and by resolving even more bits in each stage to further reduce the number of stages.

4.3.1. Minimum Area with Flash Sub-ADCs

In the fastest configuration of the pipelined A/D converter, only 1 bit (or 2 bits with digital-error correction) can be resolved in each stage. Therefore, its area is mainly determined by the total area of the opamps. Because n opamps (or $2n-1$ opamps if reference-feedforward correction is also used) are needed for an n-bit pipelined A/D converter, the overall area of such a converter can be rather large. Intuitively, a smaller overall area can be obtained if the number of opamps is reduced by having each stage resolves more bits. However, it is not clear how many more bits can still be resolved in each stage to get the minimum area because obviously the area of the comparators can grow out of hand if too many bits are resolved in each stage, leading to an increase in the overall area.

Therefore, it is necessary to write down the overall area utilization as a function of the number of stages and the area of the individual components to analytically solve for the minimum overall area. Unfortunately, this information is not usually available. In order to simplify the area optimization, only the opamps and the comparators will be considered. The result of this simplified treatment should not deviate significantly from that of a more detail and complete treatment because these components utilize the most area in a typical pipelined A/D converter.

Assuming that the optimum number of bits that are resolved in each stage is not too many so that it is not appropriate to ignore the area taken by the linear-to-binary encoding circuit, the overall area of a pipelined A/D converter can be approximately written as

$$A_{total} \approx \frac{n}{k}(A_{opamp} + 2^k A_{comp}), \tag{4.3.1}$$

or

$$A_{total} \approx \frac{n-1}{k-1}(A_{opamp} + 2^k A_{comp}), \tag{4.3.2}$$

if digital-error correction is used, or as

$$A_{total} \approx \frac{n-1}{k-1}(2A_{opamp} + 2^k A_{comp}), \quad (4.3.3)$$

if both digital-error correction and reference-feedforward correction are used. In the above equations, k is the number of bits that are resolved in each stage, n is the total number of bits to be resolved, A_{opamp} is the area of one opamp, and A_{comp} is the area of one comparator.

Plots of the overall area utilization for the above three cases as a function of the number of bits that are resolved in each stage for a 13-bit pipelined A/D converter are shown in Figure 4.3.1.1, Figure 4.3.1.2, and Figure 4.3.1.3. Two curves are shown in each figures. One curve is for the case where the opamp is only 5 times bigger than the comparator, and the other is for the case where the opamp is 10 times bigger than the comparator. These values are selected because the relative size of a typical opamp over a typical comparator falls in between these two numbers.

As can be seen from these plots, 3 or 4 bits per stage is about the optimum number of bits that should be resolved in each stage to achieve a minimum area utilization. However, the smallest implementation probably employs stages that resolve 3 bits because smaller opamps can be used. A small saving in the overall area can additionally be obtained using non-identical stages at the expense of a longer design time. Because the accuracy requirement at the later stages of a pipelined A/D converter is not as high as that in the earlier stages, stages that uses smaller capacitors, opamps, etc can be used in the later stages to reduce the overall area.

4.3.2. Minimum Area with Successive-Approximation Sub-ADCs

When successive-approximation sub-ADCs are substituted for the normally used flash sub-ADCs, a significant saving in the area and the power dissipation can be obtained because only one comparator in each stage is needed as opposed to 7 or 15 of them when flash sub-ADCs are used. Although successive-approximation registers (SARs) are also needed, because the number of successive-approximation registers increases only approximately linearly with the overall resolution, a much smaller overall A/D converter can still be obtained by having each stage of the pipe-

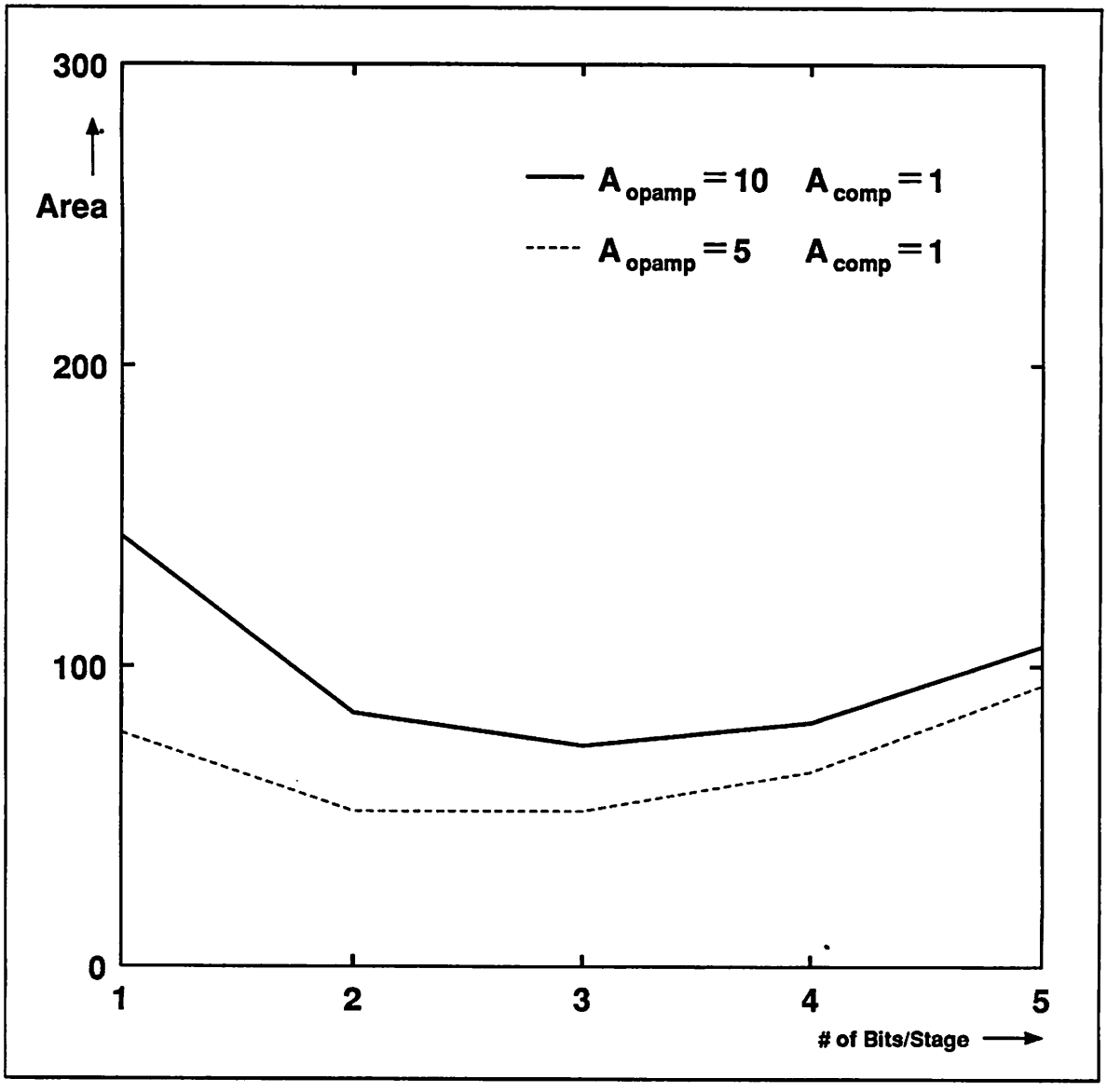

Figure 4.3.1.1 Overall Area vs. # of Bits/Stage in a straight-forward implementation.

lined A/D converter to resolve more bits to reduce the number of stages and opamps.

In order to minimize the area of a pipelined A/D converter that uses successive-approximation sub-ADCs, some assumptions have to be made. First, the overall area is assumed to vary only with the number of opamps and sub-DACs because the number of SARs is approximately constant for a given total resolution. Second, the area of each sub-DAC is assumed to be constant when few bits are resolved in each stage because of noise and matching accuracy

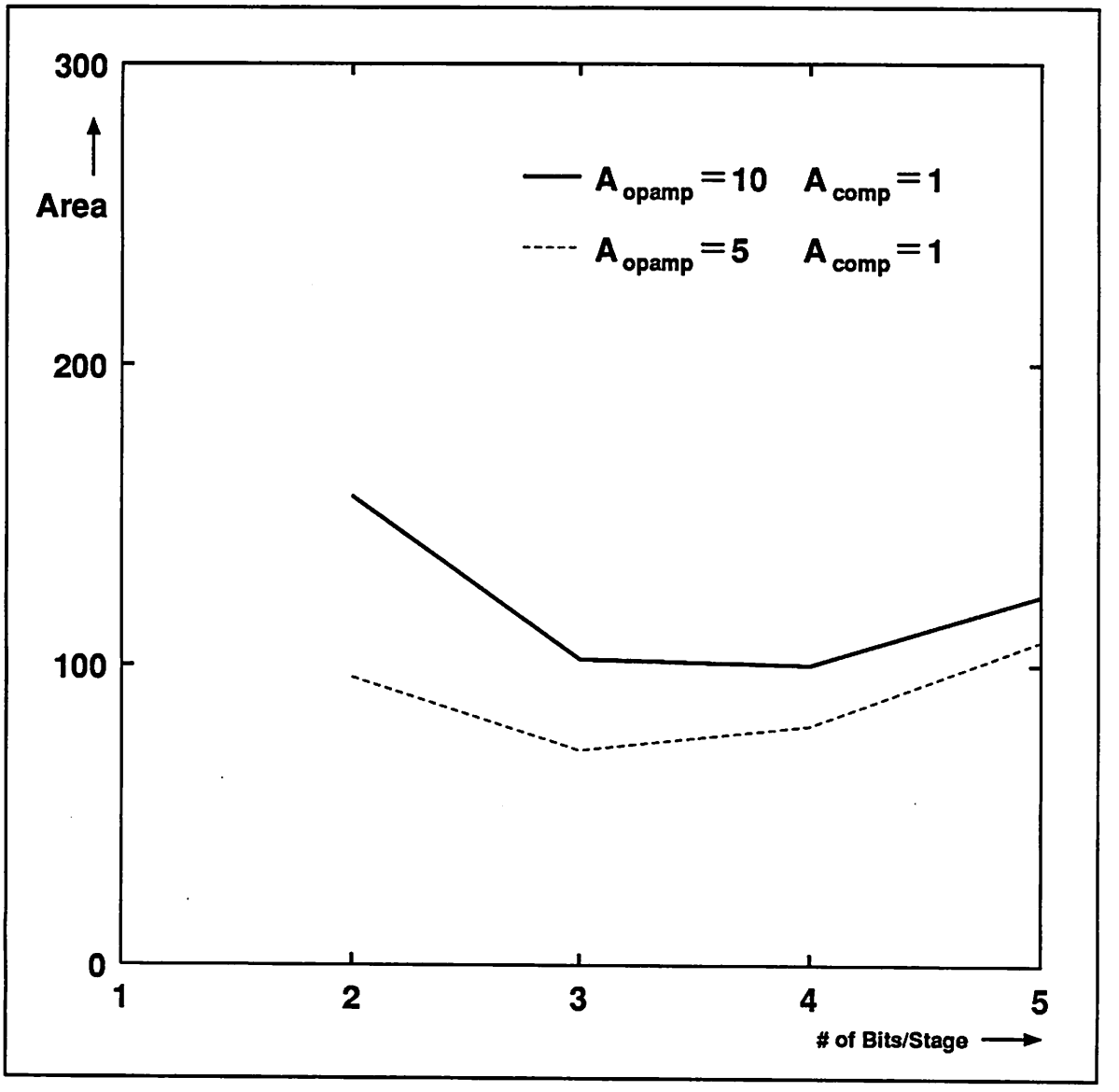

Figure 4.3.1.2 Overall Area vs. # of Bits/Stage when Digital-Error Correction is used.

requirement. When too many bits are resolved in each stage, however, the area of each sub-DAC is assumed to increase exponentially. Based on the results of a prototype pipelined A/D converter that will be discussed in the next two chapters, the area taken by the sub-DACs can be assumed to be constant when up to 5 bits are resolved in each stage for an overall linearity of around 11 to 12 bits. Third, for simplicity, it is assumed that identical stages are used.

First, consider the case in which less than 5 bits are resolved in each stage. The overall area

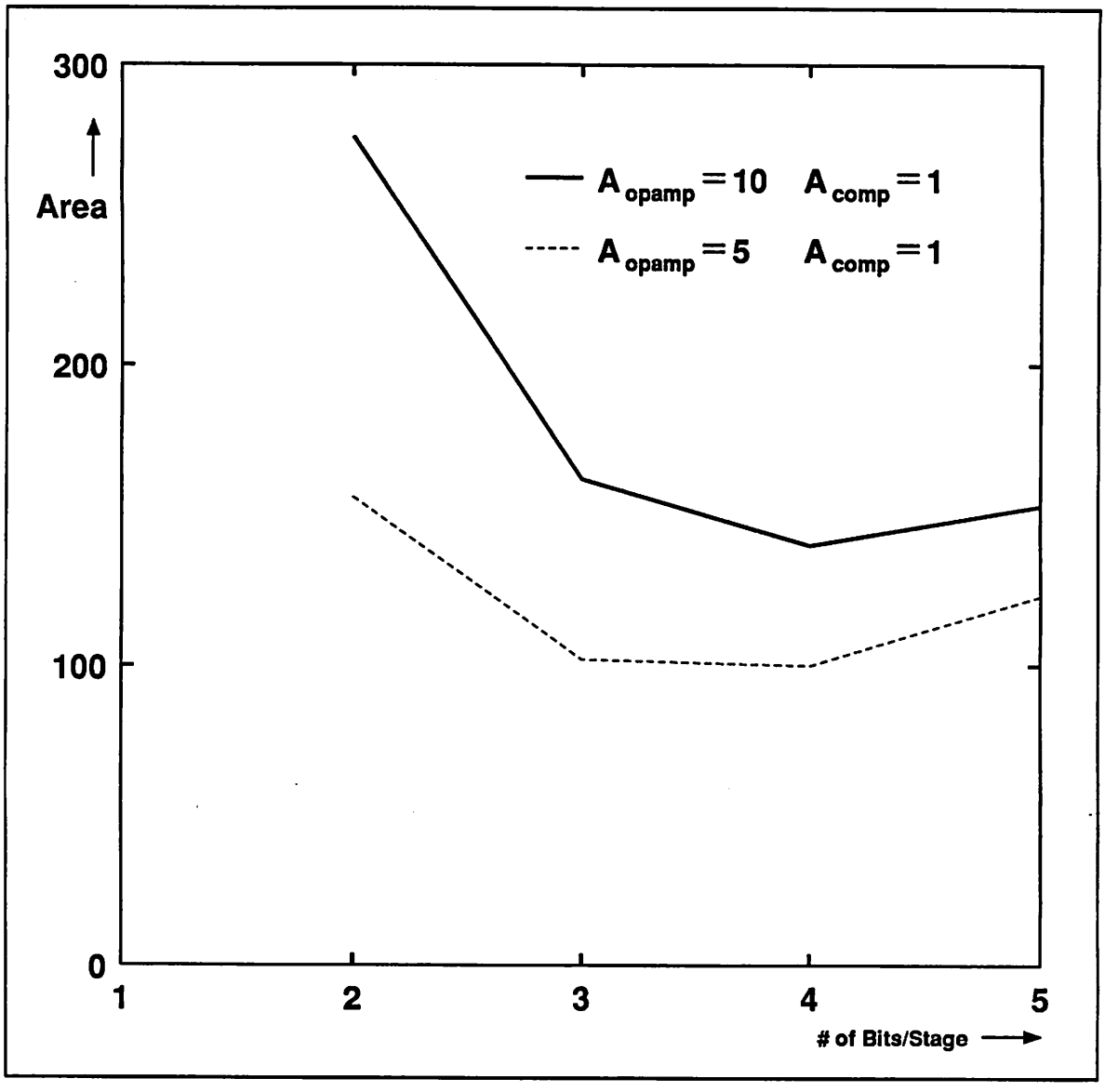

Figure 4.3.1.3 Overall Area vs. # of Bits/Stage when both Digital-Error and Reference-Feedforward Correction are used.

of a pipelined A/D converter that uses successive-approximation sub-ADCs can be written as

$$A_{total} = \frac{n}{k}(A_{opamp} + A_{smallest\_DAC}) + A_{SARs}, \tag{4.3.2.1}$$

or

$$A_{total} = \frac{n-1}{k-1}(A_{opamp} + A_{smallest\_DAC}) + A_{SARs}, \tag{4.3.2.2}$$

when digital-error correction is used, or as

$$A_{total} = \frac{n-1}{k-1}(2A_{opamp} + A_{smallest\_DAC}) + A_{SARs}, \quad (4.3.2.3)$$

when both digital-error correction and reference-feedforward correction are used. In all cases, A_{opamp} is the area of a single opamp, $A_{smallest\_DAC}$ is the area of a single smallest sub-DAC, and A_{SARs} is the area of all SARs.

When the number of bits that are resolved in each stage is more than 5, however, the overall area utilization of the converter is now

$$A_{total} = \frac{n}{k}(A_{opamp} + 2^{k-5}A_{smallest\_DAC}) + A_{SARs}, \quad (4.3.2.4)$$

or

$$A_{total} = \frac{n-1}{k-1}(A_{opamp} + 2^{k-5}A_{smallest\_DAC}) + A_{SARs}, \quad (4.3.2.5)$$

when digital-error correction is used, or

$$A_{total} = \frac{n-1}{k-1}(2A_{opamp} + 2^{k-5}A_{smallest\_DAC}) + A_{SARs}, \quad (4.3.2.6)$$

when both digital-error correction and reference-feedforward correction are used. Plots of the overall area utilization for these three cases (excluding that of the SARs) as a function of the number of bits that are resolved in each stage for a 13 bit pipelined A/D converter that uses successive-approximation sub-ADCs are shown in Figure 4.3.2.1, Figure 4.3.2.2, and Figure 4.3.2.3. In all cases, A_{opamp} is assumed to be equal to $A_{smallest\_DAC}$.

As can be seen from these figures, the overall area utilization decreases as the number of bits resolved in each stage increases. However, when more than 5 bits are resolved in each stage, the overall area starts to increase again. Therefore, the optimum number of bits that should be resolved in each stage should be around 5 bits.

The tradeoff in minimizing the area and the power dissipation is a slower overall A/D conversion. This kind of A/D converter, however, can still be faster than that of an equivalent but straight-forward implementation of a successive-approximation A/D converter, especially if digital-error correction is used. Digital-error correction allows shorter clock periods for the

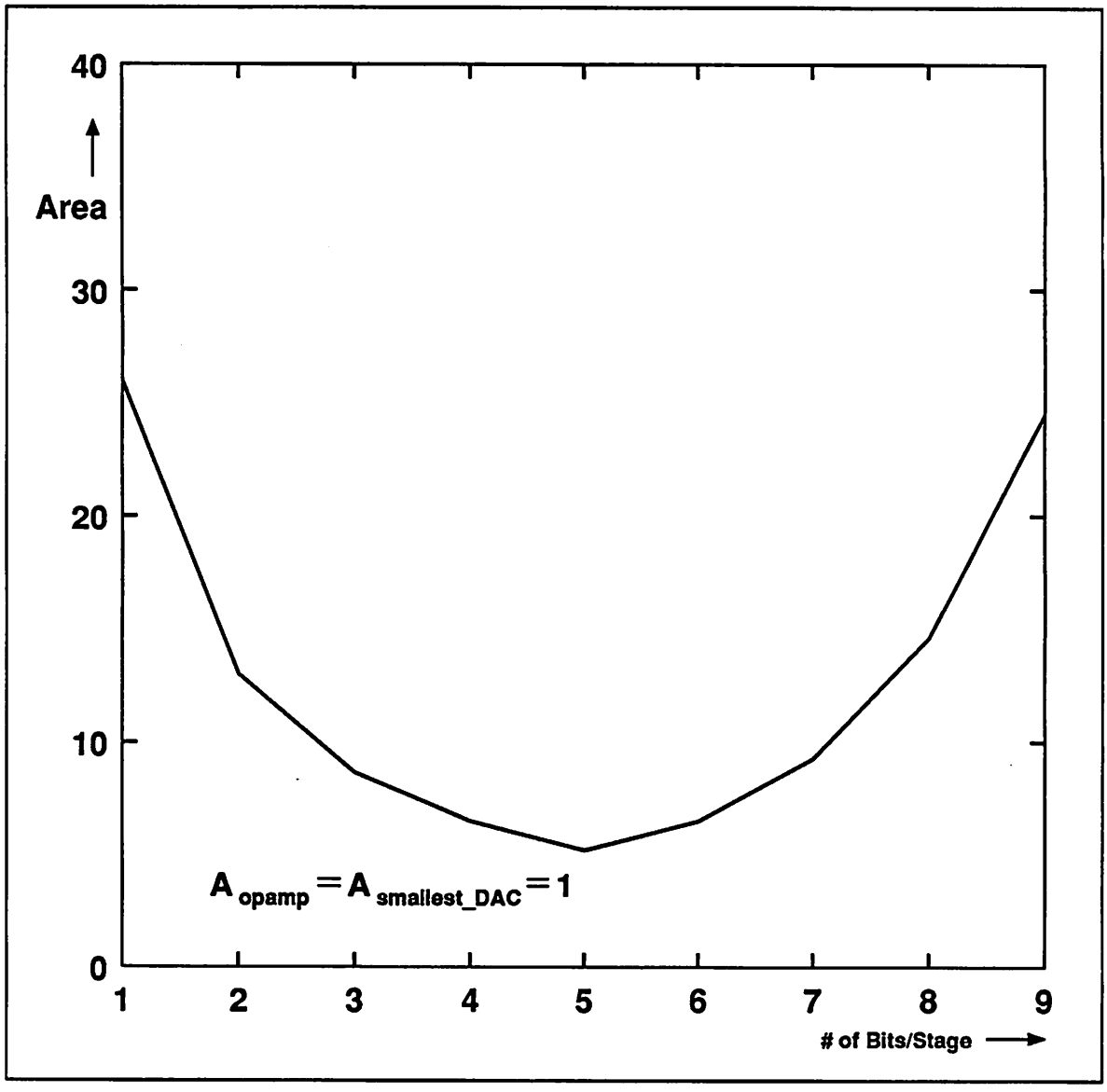

Figure 4.3.2.1 Overall Area vs. # of Bits/Stage in a straight-forward implementation.

successive-approximation search to be used since the coarse A/D subconversion has to be accurate to only the number of bits that are resolved in that stage alone instead of to the full accuracy or resolution of the overall pipelined A/D converter.

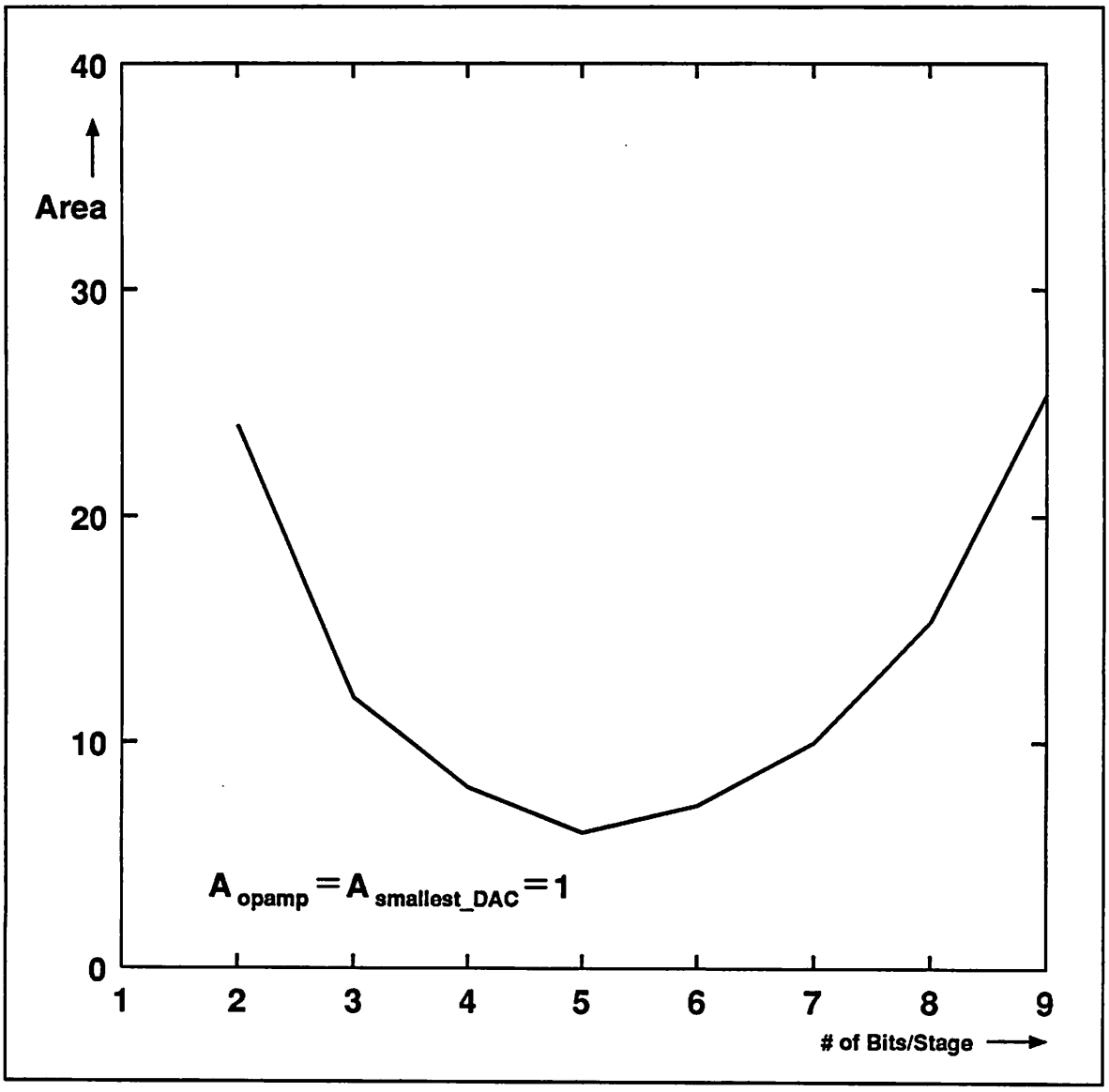

Figure 4.3.2.2 Overall Area vs. # of Bits/Stage when Digital-Error Correction is used.

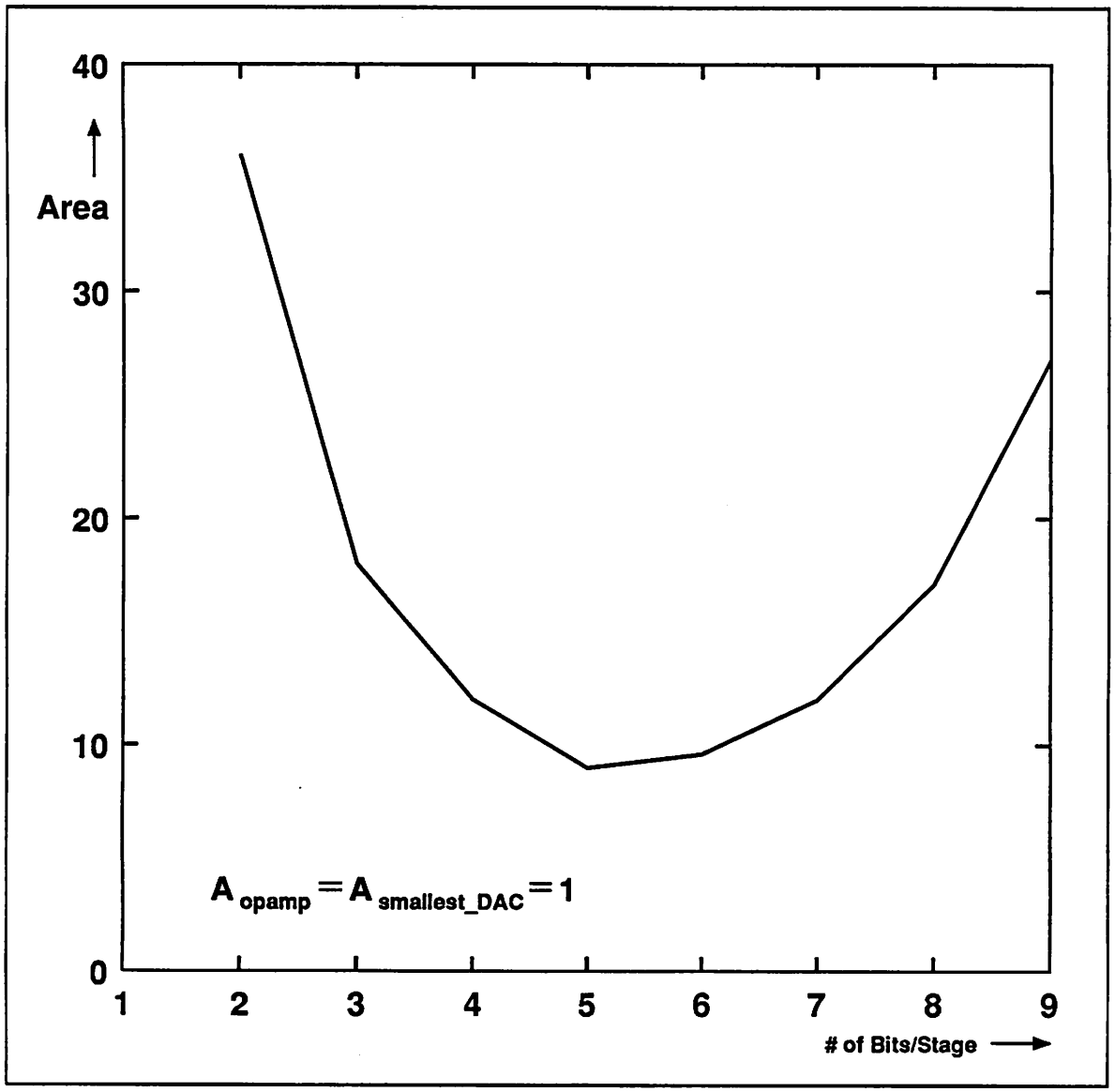

Figure 4.3.2.3 Overall Area vs. # of Bits/Stage when both Digital-Error and Reference-Feedforward Correction are used.

CHAPTER 5

PROTOTYPE DESCRIPTION

One application of high-resolution and medium- to high-speed A/D converter is the ISDN A/D interface function. In such an application, a resolution of 12-13 bits and a sampling rate of 100-200 ks/s are usually required. [5,29] Previously, various prototypes of this function have been implemented using the 2^{nd} order Σ–ΔM architecture. However, such implementations are only practical in a 2-μm or better CMOS processing technology because of the high oversampling ratio required.

Using the pipelined A/D conversion architecture, such a converter function can be easily realized as a few number of effective clock cycles is needed for each A/D conversion. The goal of this project is to illustrate the use of reference-feedforward correction in a pipelined A/D converter to achieve a resolution higher than that normally possible. The other objectives are to achieve as low-power dissipation and small-area utilization, and as high linearity as possible. To illustrate the speed advantage of the pipelined approach, the prototype is implemented using a standard 3-μm, 5-V only CMOS processing technology.

5.1. ARCHITECTURE DESCRIPTION

The typical pipelined A/D converter implementation using flash sub-ADCs is not optimum for the ISDN application as the required conversion speed is not as high as that needed in video-rate applications. Even when the optimum number of stages is used, up to 12 opamps and 48 comparators are still required to meet the 13-bit resolution requirement when both digital-error correction and reference-feedforward correction are used. On the other hand, when successive-

approximation sub-ADCs are used, a much smaller and lower-power overall pipelined converter may be realized because only one comparator is needed in each stage regardless of the number of bits resolved in that stage.

To minimize the area utilization as much as possible, a three-stage pipelined A/D converter was chosen. A three-stage implementation needs only three opamps; two opamps for the inter-stage amplifiers in the first two stages, and an opamp for the S/H that is needed to store the new reference voltage for the 2^{nd} and 3^{rd} stage. Although a two-stage version needs one less opamp than a three-stage version, a slightly larger overall chip area would result as 7-bit sub-DACs would be required instead of 5-bit ones. Even if the area is not a major concern, due to the much higher interstage gain needed, a much slower interstage amplifier would result in the two-stage version. This in turn reduces the conversion speed of the A/D converter.

5.2. CIRCUIT DESCRIPTION

The successive-approximation sub-ADCs are implemented using the charge-redistribution architecture. There are two major reasons for this. First, when a charge-redistribution sub-ADC is used, the only major additional circuit needed to form a pipelined stage is an opamp for residue amplification. Other required functions such as the S/H, sub-DAC, and subtraction are inherently available as part of the charge-redistribution A/D converter. Second, the S/H circuit that is needed to store the new reference voltage for the next stage must drive only capacitive loads. Therefore, a simple, single-stage, high-output impedance opamp can be used in the S/H circuit.

A slightly simplified circuit diagram of the actual implementation of the 1^{st} stage of the prototype pipelined A/D converter is shown in Figure 5.2.1. A fully differential configuration was chosen to maximize the power-supply rejection and to simplify the design of offset-voltage cancellation.

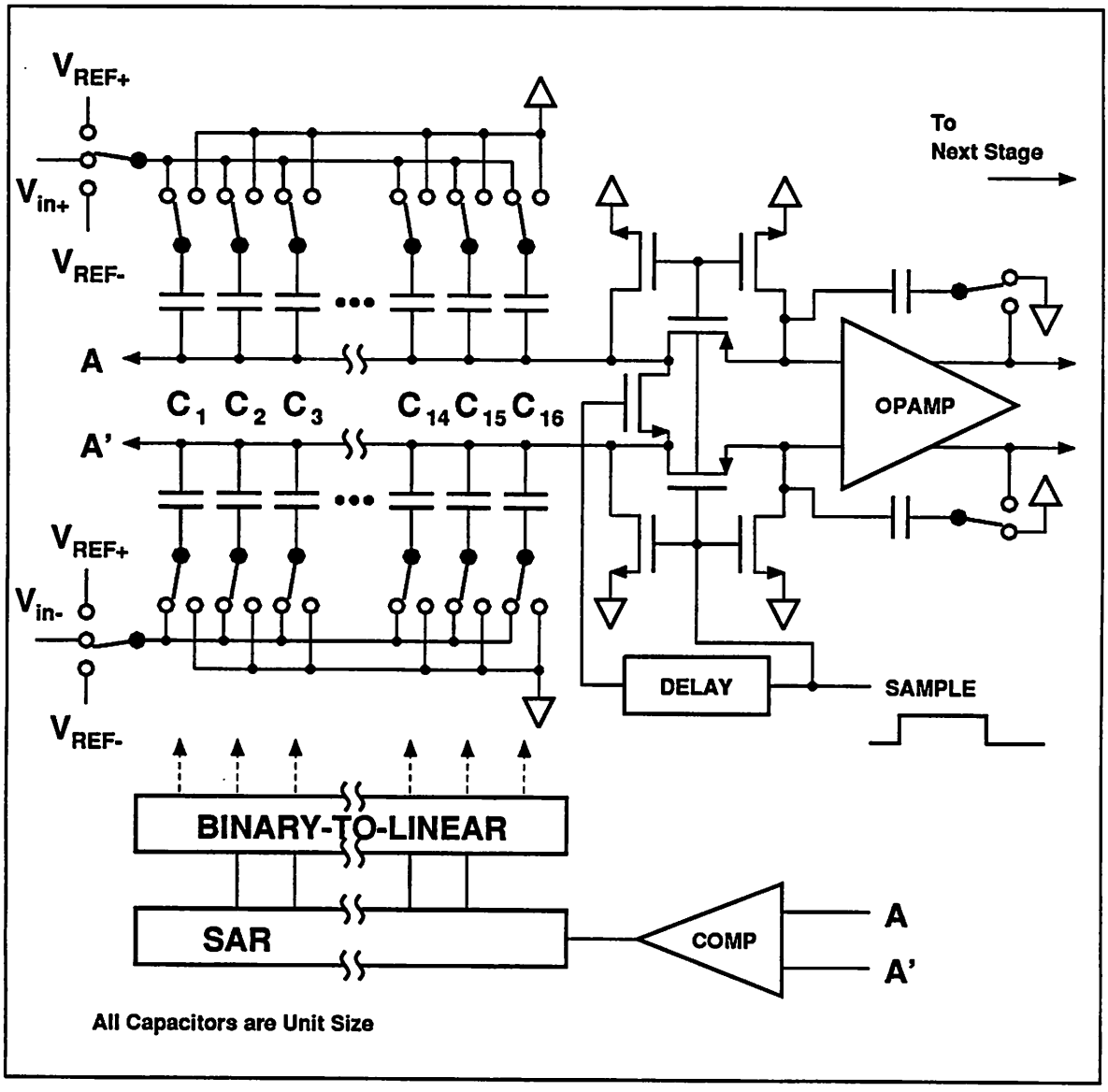

Figure 5.2.1 Simplified Circuit Diagram of the 1st Stage.

5.3. DETAILS OF OPERATION

In principle, the conversion process of the 1st stage can be divided into 5 major phases as illustrated in Figure 5.3.1.

5.3.1. Sampling Phase

The first phase is the analog input sampling phase. The input signal is sampled by connecting all

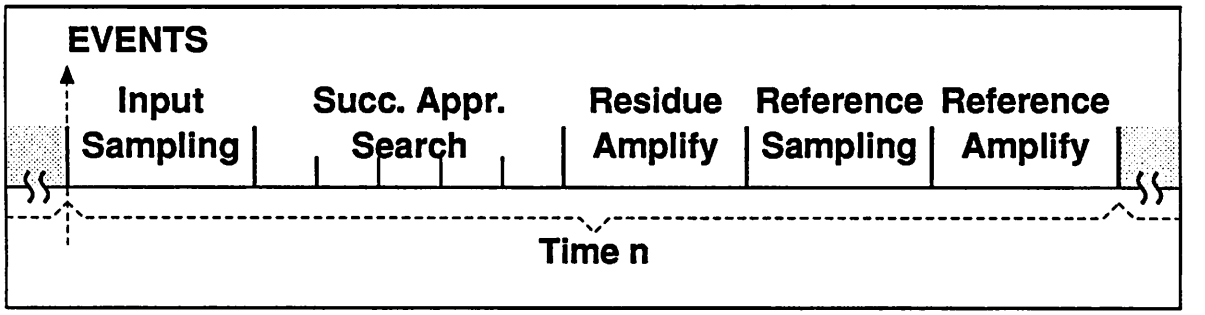

Figure 5.3.1 Timing Diagram of Operation of the 1st Stage.

of the bottom plates of the input capacitor array to the analog input path. At the same time, the top plates of the capacitor array are shorted together and to the ground (the $\frac{1}{2}V_{dd}$ point in a single-supply environment) to provide the input charging current path and to establish the input common-mode voltage for the opamp and the comparator. The input signal is then held on the capacitor array by opening all of the sampling switches (including the grounding switches) at the end of this phase.

Notice from Fig. 5.2.1 that a few things are different from an ordinary implementation of the charge-redistribution ADC. Most implementations of the charge-redistribution ADC use a binary capacitor array. However, a linear capacitor array was chosen here, even though a binary capacitor array will work almost as well. Specific reasons to this choice will be covered in section 5.7.

Another difference is that there are more than just one pair of input sampling switches. The sampling switches in traditional charge-redistribution ADCs usually consist of a pair of MOS switches that are tied around an opamp/comparator so that the opamp/comparator offset voltage is canceled out. There are a couple of limitations to the latter approach. First, the opamp/comparator has to be unity-gain stable for this to work. This is not optimum because the opamps in the prototype converter cannot be optimized for fast settling time with a closed-loop gain of 16 if it has to be unity-gain stable. Second, only the offset voltage of the opamp can be

canceled this way, while the offset voltage from the charge-injection mismatch between the sampling switches cannot be canceled. In the past, small sampling switches have been used to reduce this charge-injection mismatch. As a result, however, the high-frequency response of the S/H is degraded. Incidentally, the so-called multi-stage auto-zero offset-cancellation technique cannot be used without adding more clock cycles to solve this offset problem if the input signal is moving rapidly.

To overcome the above limitations, the capacitor array of the prototype A/D converter is detached from the opamp during the input sampling phase so that both the frequency response of the input sampling and opamp offset cancellation can be optimized independently. Improved offset cancellation schemes that can reduce the residual charge-injection mismatch can now be used because the opamp is not subjected to any moving input signal. This includes the so-called multi-stage offset cancellation scheme and the auxiliary offset cancellation scheme. Reduced charge-injection error from the sampling switches, on the other hand, can be obtained by using a differential sampling switch which terminals are connected to the top plates of the capacitor array in addition to the two sampling (grounding) switches. This differential switch is opened after the other sampling switches are opened so that the charge-injection mismatch from the grounding switches is eliminated by the differential sampling switch. Because only a single switch is effectively used, the charge-injection mismatch should in principle be reduced significantly.

5.3.2. A/D Subconversion

The second phase is for the A/D subconversion steps. Because successive-approximation sub-ADCs are used, this phase actually consists of a number of sub-phases for the successive-approximation search. Therefore, the conversion speed of the overall A/D converter can be easily limited by the A/D subconversion steps if too many bits are resolved in each stage. To speed up the successive-approximation A/D subconversion, short clock sub-phases can be used, fortunately. A/D subconversion decision errors that result in the early stages of a pipelined A/D

converter due to the limited time that is available during the successive-approximation steps can be corrected using digital-error correction in the later stages as long as the error is bounded within +/- 0.5 LSB of the coarse sub-ADCs.

5.3.3. Residue Amplification

The third phase is for the residue-amplification step. The amplified residue voltage can be obtained by integrating the residue charge that is left on the top plates of the input capacitor array to a unit capacitor that is usually referred to as the integrating capacitor at the end of the successive-approximation steps. Specifically, this is done by connecting the integrating capacitor around the opamp while keeping the positions of all of the bottom plate switches unchanged from that of at the end of the charge-redistribution successive-approximation A/D subconversion.

5.3.4. Reference Sampling

The fourth phase is for the reference-sampling step. The sub-DAC segment voltage for the new reference voltage can be obtained by sampling the reference voltage onto the unit capacitor that corresponds to the latest result of the A/D subconversion, as illustrated in Figure 5.3.4.1 for the case of input voltages that fall in the range of the 2^{nd} segment of the D/A subconverter. The illustrated technique can only be used with a linear capacitor array, however.

Another way to obtain the sub-DAC segment voltage is to take the difference between two adjacent sub-DAC levels. This can be done by first sampling the reference voltage onto those capacitors that are represented by the next higher digital output code of the latest result of the A/D subconversion, and then subtracting the same reference voltage through those capacitors that are represented by the unincremented digital output code in the reference amplification phase. This technique works whether a linear capacitor array or a regular binary capacitor array is used. An illustration on how this is done in a binary capacitor array is shown in Figure 5.3.4.2

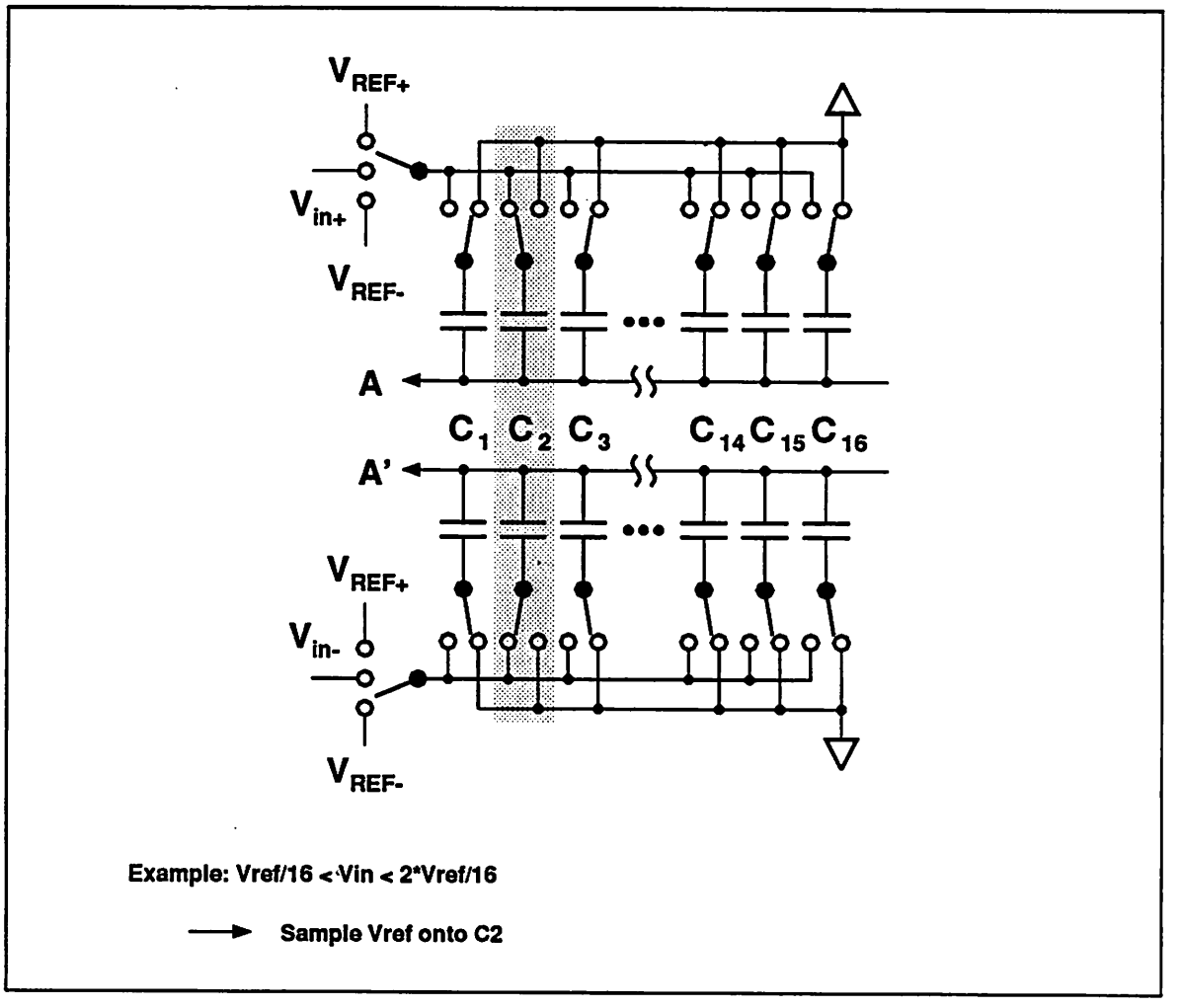

Figure 5.3.4.1 Reference-Sampling Step for a Linear Capacitor Array.

5.3.5. Reference Amplification

The final phase is for the reference-amplification step. The new reference voltage for the next stage can be obtained by connecting the integrating capacitor around the opamp while throwing all of the bottom plate switches to the ground, as illustrated in Figure 5.3.5.1 when a linear capacitor array is used. When the difference method is used, however, the bottom plate switches should be configured in such a way that only those capacitors that are represented by the latest digital output code are connected to the reference voltage and the rest are connected to the ground. An example on how this is done for a binary capacitor array is illustrated in Figure

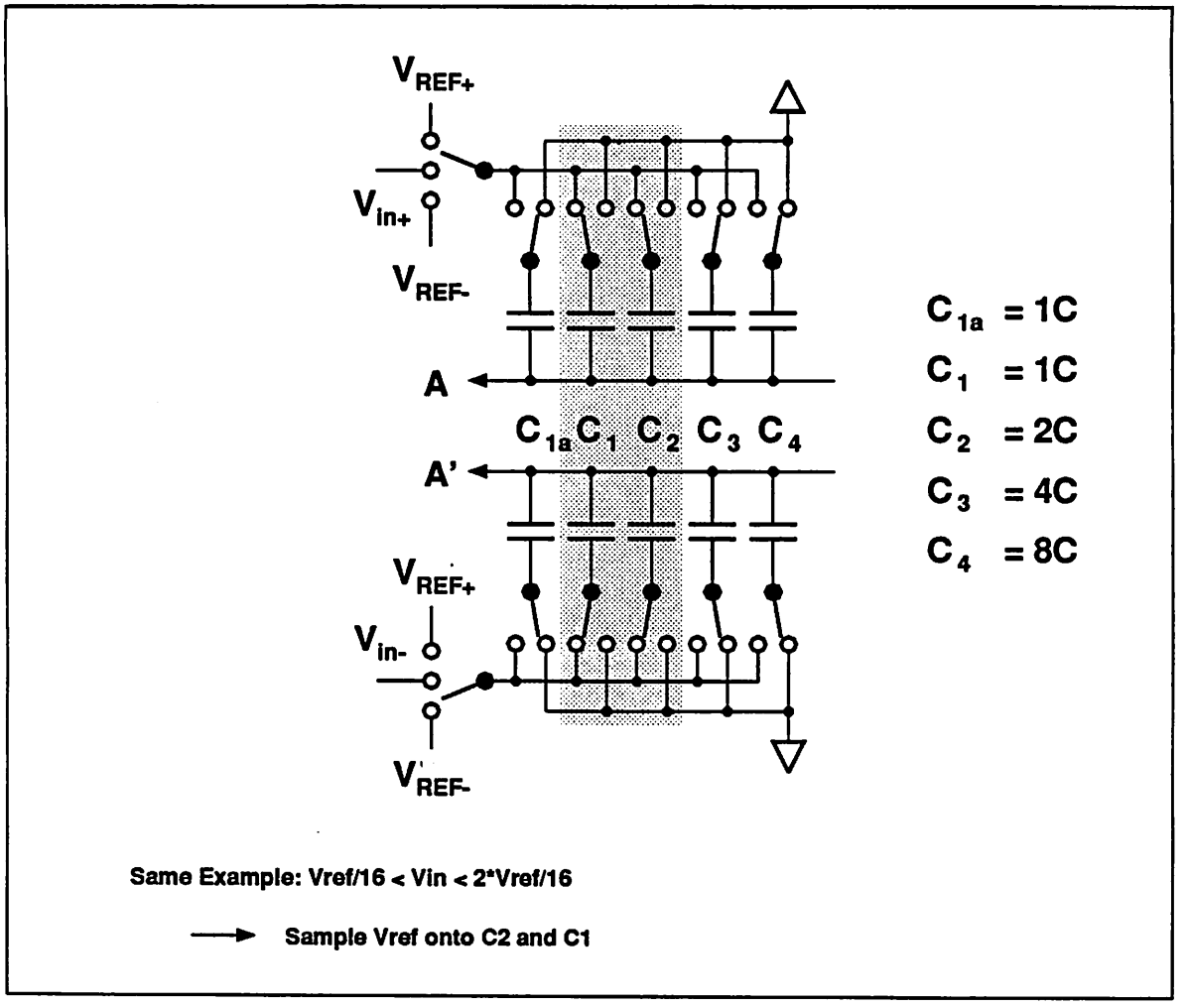

Figure 5.3.4.2 Alternative Reference-Sampling Step for a Binary Capacitor Array.

5.3.5.2.

Notice that because the same opamp and the same capacitors are used throughout the residue and reference-amplification steps, any gain error caused by either the opamp finite open-loop gain or the ratio error between the sampling and integrating capacitor will only affect the absolute magnitude of the amplified residue and the new reference voltage. It will not, however, affect the relative ratio between the full-scale amplified residue and the new reference voltage. Therefore, the effects of gain error are canceled out with the use of reference-feedforward correction.

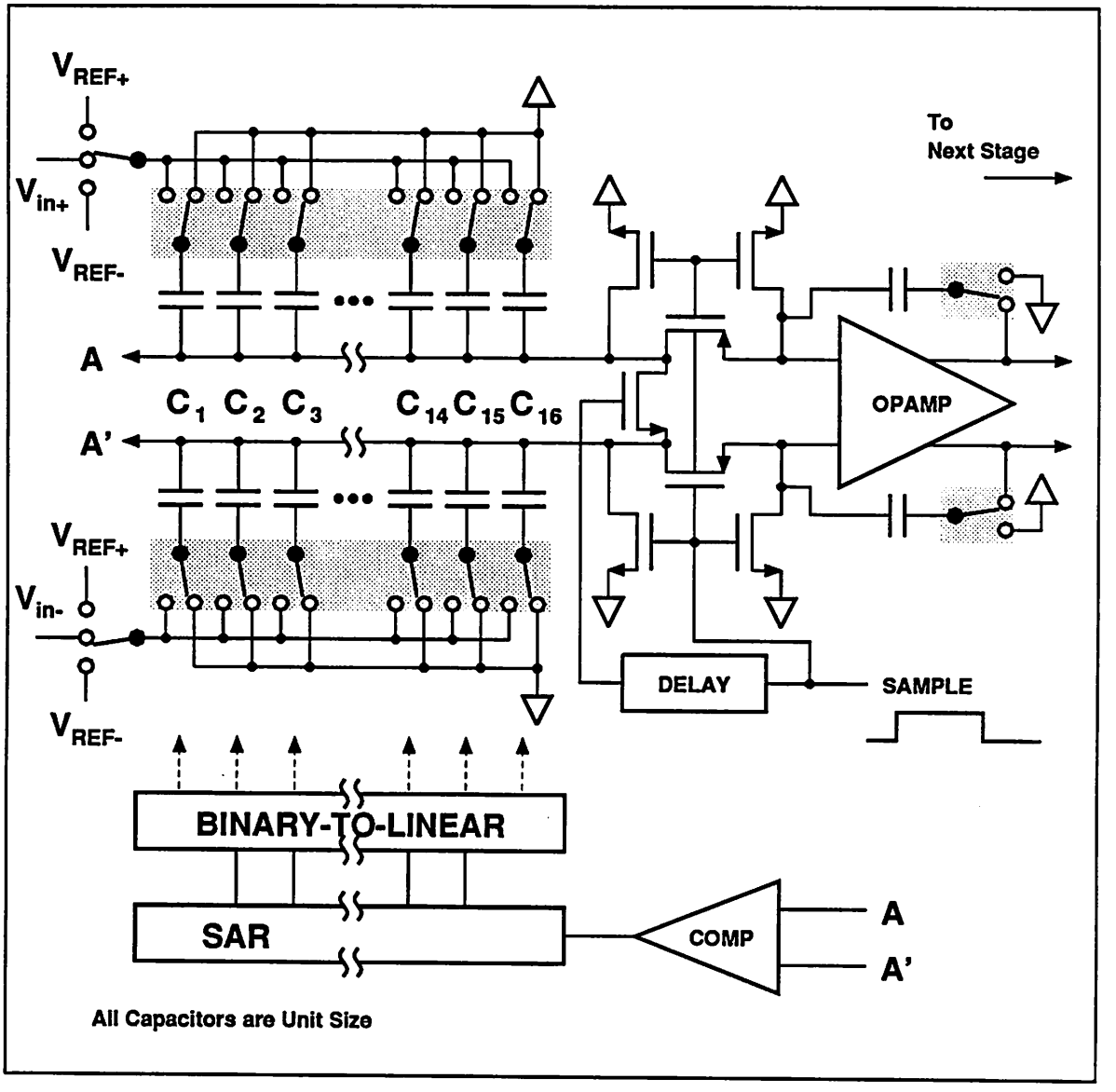

Figure 5.3.5.1 Reference-Amplification Step for a Linear Capacitor Array

5.4. OPAMP

As discussed in chapter 3, because of the use of the reference-feedforward correction, it is not necessary to maintain exact interstage gain to achieve a high linearity. This means that the ratio of the sampling over intergrating capacitor does not have to be precise. Furthermore, an opamp with a high open-loop gain is not needed. However, this does not mean that the opamp does not have to settle to the normally required accuracy. Since 9 bits still have to be resolved after the 1st

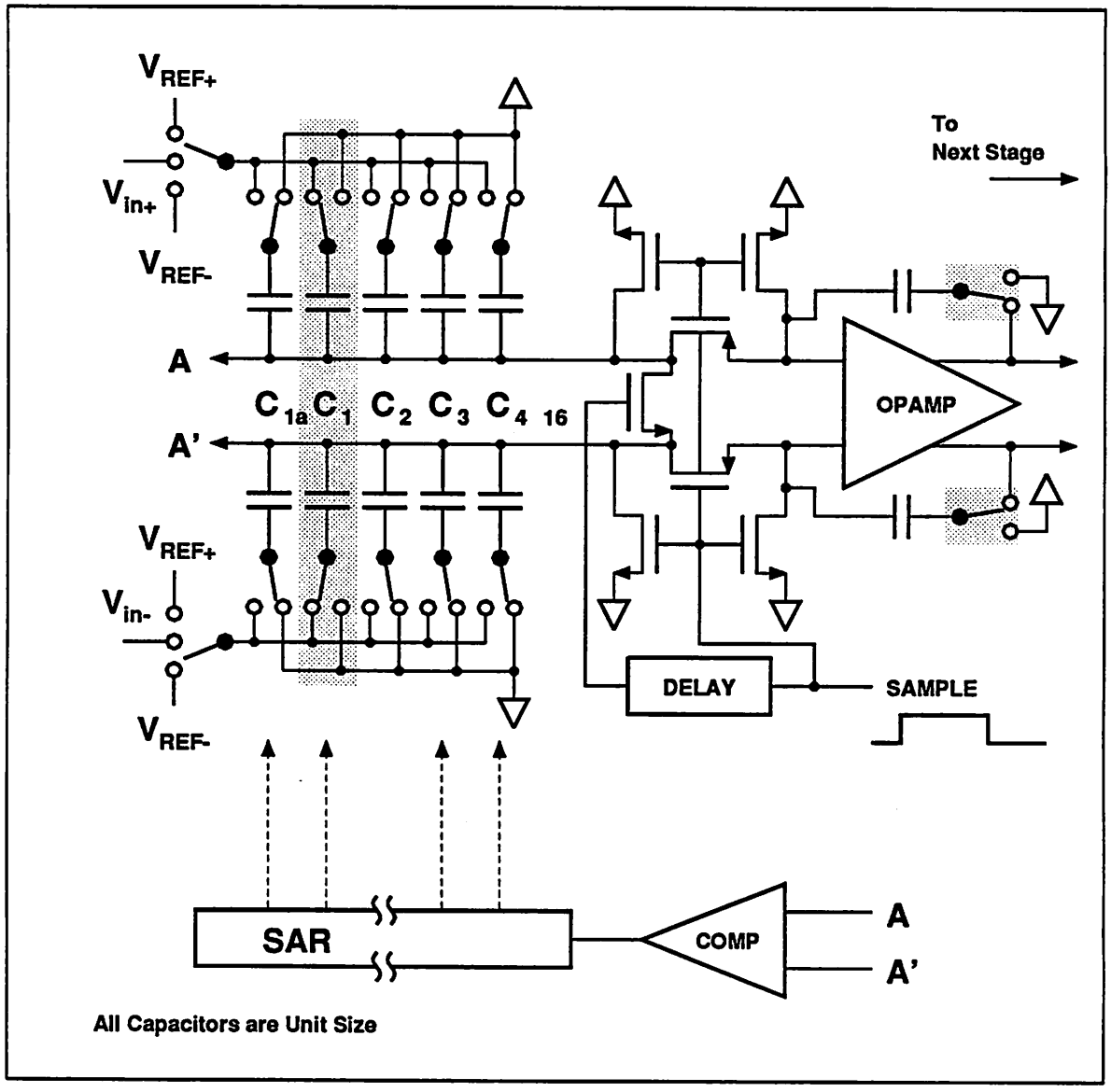

Figure 5.3.5.2 Reference-Amplification Step for a Binary Capacitor Array

stage, the opamp in the first interstage amplifier still have to settle to at least 11 bits from the final value to reduce the error contribution of the opamp to less than ¼ LSB, whether reference-feedforward correction is used or not. Based on the timing diagram described earlier, to achieve a sampling rate of up to 300 ks/s, only about 500 ns is available for the opamp to settle to the required accuracy. However, in order to guarantee that the opamp would still settle under a wide range of operating conditions and various processing variations, it is necessary to require the

opamp to settle in only half of the available time under typical operating and processing conditions.

Because approximately 7 time constants are required for the opamp to settle to 11-bit accuracy from the final value and only about half of the 500ns is available, an opamp that has a closed-loop bandwidth of approximately 4 MHz is needed. This corresponds to a gain bandwidth (GBW) product requirement of:

$$GBW = \frac{4}{f_b} MHz, \tag{5.4.1}$$

where the feedback factor, f_b, is at most:

$$f_b = \frac{1}{2^{(n_1-1)}+1}$$

with n_1 being equal to 5 in the prototype converter. Therefore, an opamp that has a GBW product of at least 58 MHz is needed. As discussed in chapter 3, because of the parasitic capacitance at the opamp summing node, a lower feedback factor than suggested by equation 5.4.1 results. This in turn causes the GBW product requirement to increase. In fact, a GBW product of as much as 100 MHz might be needed considering that the input summing node parasitic capacitance of the prototype converter can be as much as 3-4 pF and the sampling capacitance is only about 5 pF.

5.4.1. Opamp Implementation

Early designs of CMOS opamps have used two high-gain stages with an internal pole-splitting compensation capacitor. But most of the new high-speed opamp designs have used a single high-gain stage architecture, especially in the fully differential implementations. There are two reasons to this switch. First, a single-stage opamp can be made faster[30] because there are inherently fewer non-dominant poles in a single-stage opamp than in a two-stage opamp. Second, because there is only one high-gain stage in a single-stage opamp as opposed to two high-gain stages in a two-stage opamp, it is simpler to design the common-mode-feedback circuit of a single-stage opamp.

Consider the most commonly used single-stage opamp based on the folded-cascode architecture, as illustrated in Figure 5.4.1.1. The GBW product of such an opamp can be written as:

$$GBW = \frac{g_{m,in}}{2\pi C_{total\_load}} \quad (H_z),$$

where $g_{m,in}$ is the input transconductance of the opamp, and C_{total_load} is the sum of the sampling capacitance of the next stage, junction, common-mode feedback, integrating, and routing capacitance. Even with a reasonable loading capacitance of 7 pF, a $g_{m,in}$ of 4 mS is needed to meet the large GBW product requirement stated earlier. This $g_{m,in}$ requirement is about ten times of that can be normally obtained from a typical 3-μm CMOS opamp implementation. To achieve such a high input transconductance, it is necessary to scale both the transistor sizes and bias current levels up by ten times from the normally used amount. However, aside from the large-area and high power dissipation, such a solution is not attractive because the resulting high input capacitance further degrades the feedback factor, f_b.

The solution selected in the prototype converter was to add a low-gain but high bandwidth preamp in front of the folded-cascode opamp to boost the GBW product of the folded-cascode opamp to the required value. A conceptual block diagram of this opamp is illustrated in Figure 5.4.1.2. The effective input transconductance of this opamp is:

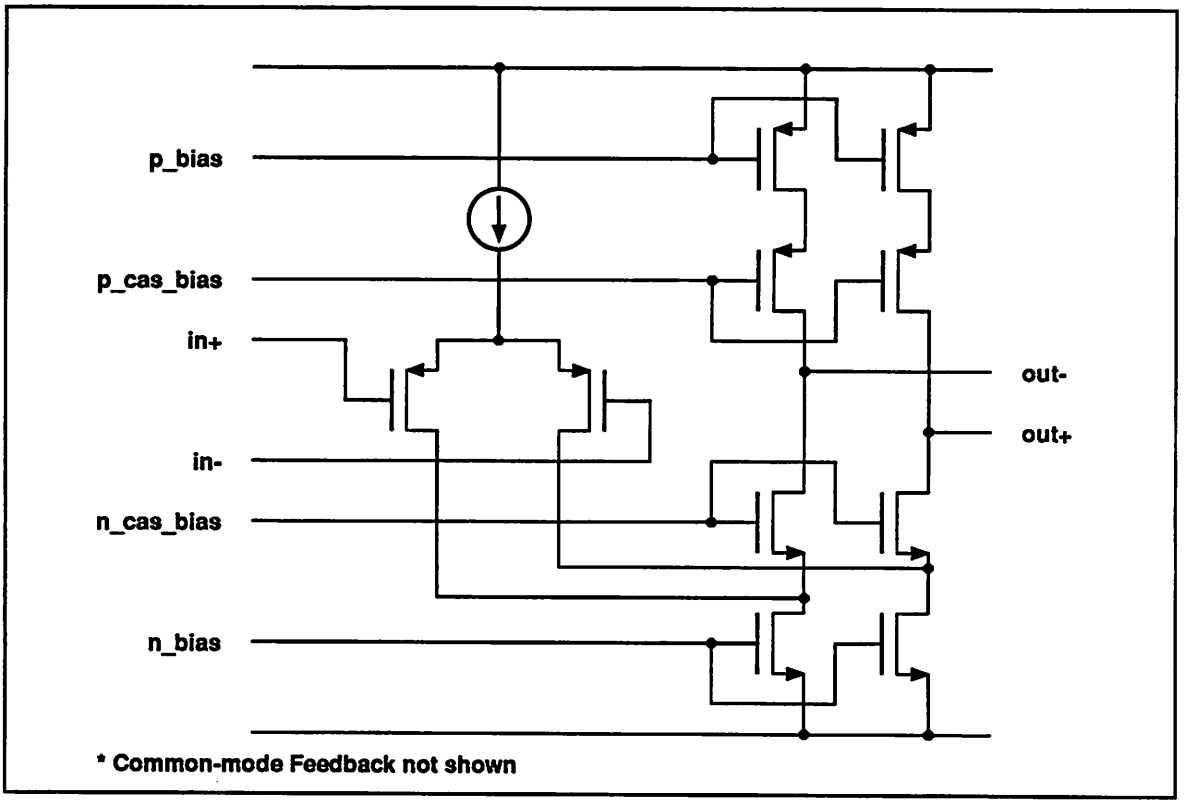

Figure 5.4.1.1 Folded-Cascode Opamp.

$$g_{m,eff} = A_{pre-amp}\, g_{m,in}, \tag{5.4.1.1}$$

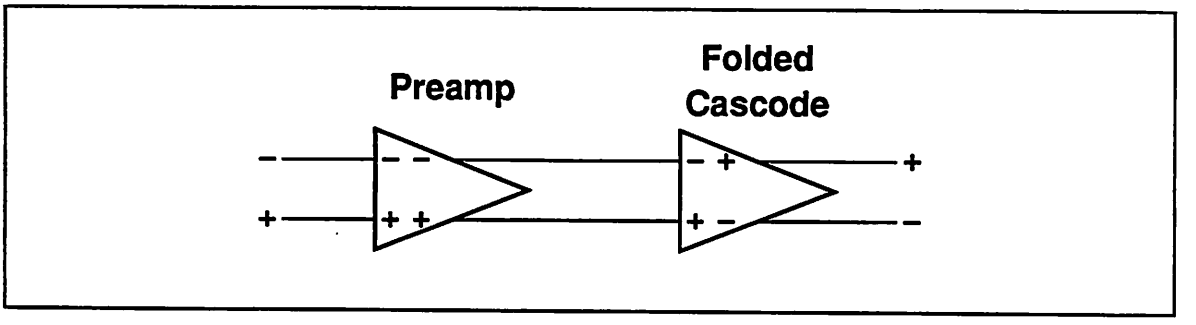

Figure 5.4.1.2 Conceptual Block Diagram of the Prototype Opamp.

where $A_{pre-amp}$ is the gain of the pre-amp and $g_{m,in}$ is the input transconductance of the folded-cascode opamp, which will be referred for simplicity from now on as the "second stage". This type of opamp is not normally used because of the large phase shift near the unity-gain bandwidth. However, in this application it is not the unity-gain bandwidth that is important but

rather the closed-loop bandwidth, which can still be significantly lower than the non-dominant pole frequencies.

5.4.2. Preamp

The circuit diagram of the preamp used in the prototype converter is shown in Figure 5.4.2.1. The gain of the preamp is designed to be around 10 to boost the second-stage transconductance by ten times. Diode connected loads are used instead of resistors because the gain of the preamp can be more tightly controlled. However, because of the square-law behavior of MOS transistors, to get a gain of even only ten, it is necessary to simultaneously use small load devices and inject some extra current to the input devices. To avoid loading the preamp by the input capacitance of the second stage, the preamp is fully buffered using MOS source followers.

5.4.3. Second Stage

The "second stage" of the two-stage opamp uses a slightly modified folded-cascode architecture. To improve the slew rate and open-loop gain while requiring only a slight increase in area and power dissipation, a push-pull [31] configuration was chosen. The circuit diagram of the "second stage" opamp is illustrated in Figure 5.4.3.1.

5.4.4. Offset Cancellation

The opamp offset voltage is usually canceled by storing an equal but opposite amount of voltage on capacitors that are placed at the input of the opamp.[26] In the case of the charge-redistribution stages, these capacitors are the sampling capacitors. The correct offset voltage can be found by putting the opamp in a unity-gain mode. However, because the GBW of the opamp that is used here is much higher than the non-dominant poles of the preamp and the second stage, the opamp will be most likely unstable in the unity-gain mode. It is, however, still possible to put the second stage in the unity-gain mode and to store both the preamp and second-stage offset voltage on a capacitor that is placed in between the preamp and the second stage. This technique is normally

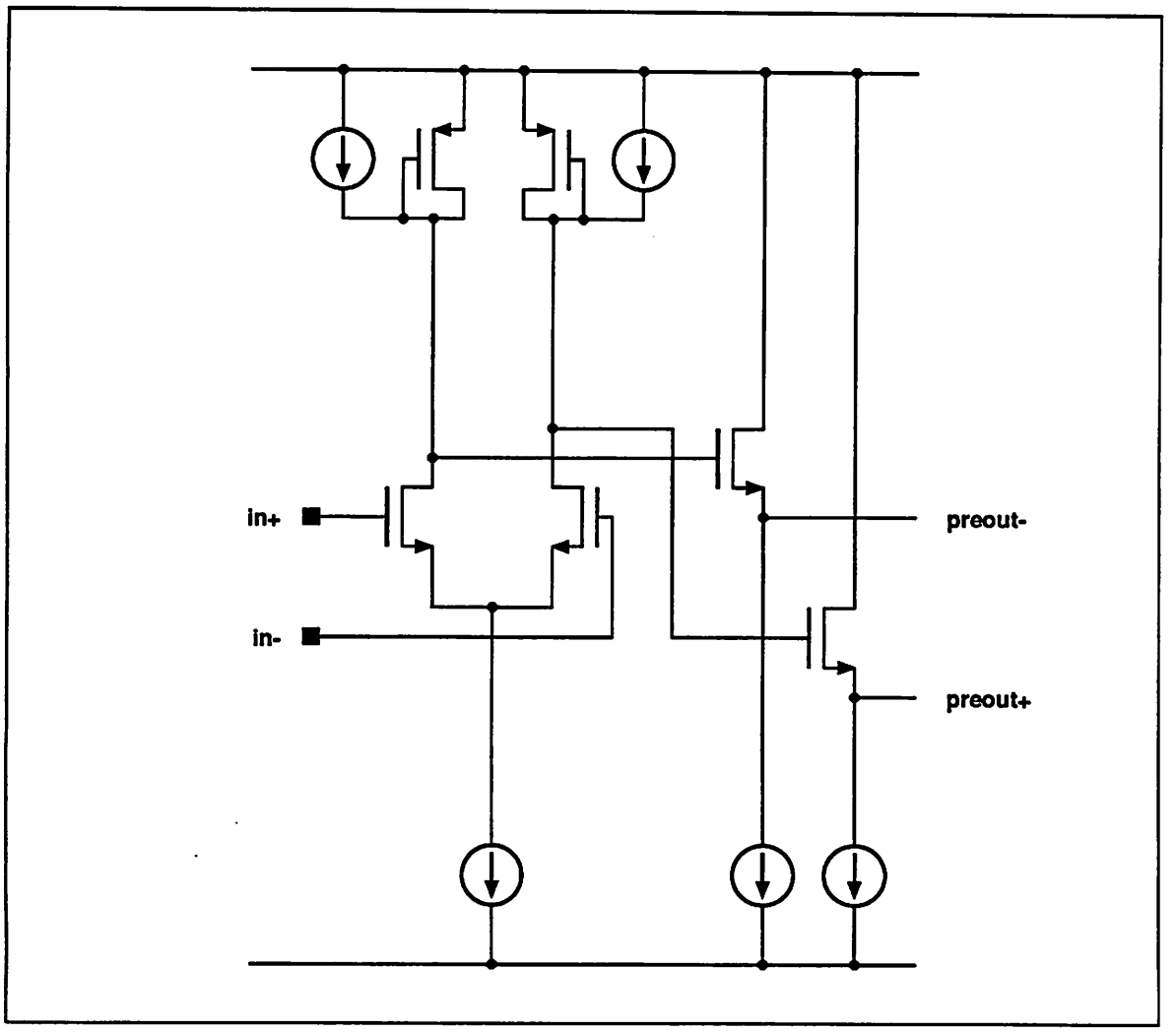

Figure 5.4.2.1 Simplified Circuit Diagram of the Preamp.

used in high-precision multistage comparator designs to achieve a very low residual offset voltage.

5.4.5. Common-Mode Feedback

Dynamic common-mode feedback [32] is used for two reasons. First, continuous common-mode feedback schemes such as that used in [13] cannot be used in a single 5 volt power supply environment without severely limiting the output voltage swing of the opamp. Second, a much faster common-mode feedback response time can be obtained using the dynamic scheme since

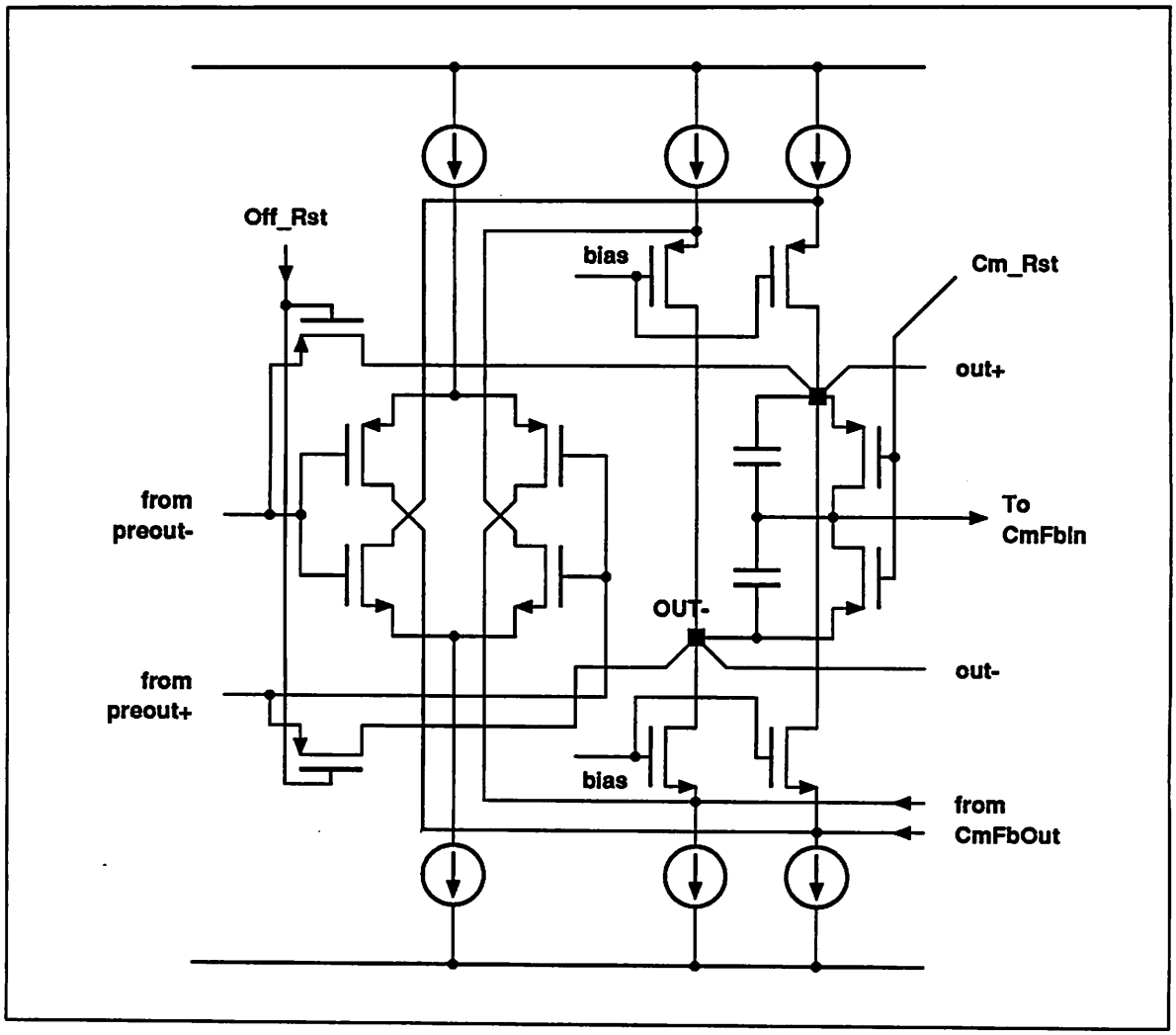

Figure 5.4.3.1 Simplified Circuit Diagram of the 2nd Stage Opamp.

bigger common-mode feedback devices can be used.

As can be seen from Figure 5.4.5.1, the dynamic common-mode feedback is achieved by summing the differential output voltages through a pair of common-mode feedback capacitors. This voltage is then compared to the ½Vdd point using a modified differential pair whose outputs are used to control the common-mode currents of the second-stage opamp.

5.5. BIASING CIRCUIT

Proper operation of the opamp described in the previous section is also determined by the kind of

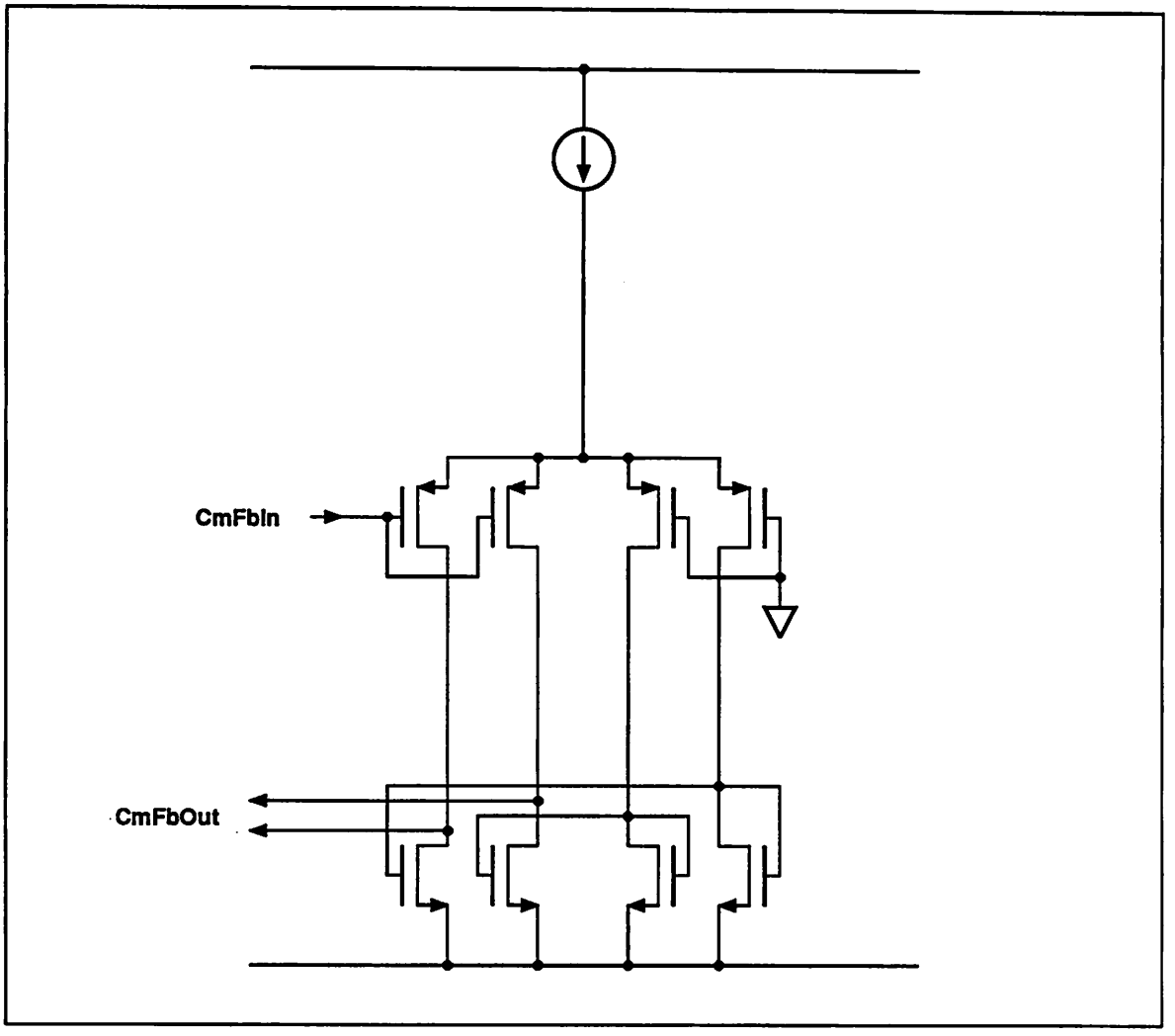

Figure 5.4.5.1 Dynamic Common-Mode Feedback Circuit.

biasing circuit used. As can be seen from Figure 5.4.3.1, the output voltage swing of the opamp can be severely limited by poor selections of the cascode transistor biasing voltages, especially in a single 5 volt power supply environment. In order to maximize the output voltage swing, the so-called high-swing [33] biasing circuit is therefore needed.

Early high-swing biasing circuit designs, however, are rather sensitive to variations in the threshold voltage(V_T) of MOS transistors caused by the body effect(γ) parameter because it is not normally possible to have all of the cascode transistors isolated in separate wells to keep the threshold voltage of the cascode transistors independent of the the body-effect parameter. Even if

it is possible to have each cascode transistor in a separate well, it is still desirable to not rely on it because the pole locations of the cascode transistors would be reduced by the parasitic well-substrate capacitance. Furthermore, a larger opamp than normally possible will result because of the restrictions in the well design rules from typical CMOS processes.

An example of the early high-swing biasing circuit[33] is illustrated in Figure 5.5.1. The difference between this circuit and the simple stacked diode-connected biasing circuit is that the gate of the cascode transistor in this high-swing biasing circuit is set at a voltage that is just enough to keep the bottom current-mirror transistor in the saturation region. This voltage is obtained by subtracting one threshold voltage from the drain voltage of the reference cascode transistor (M_1).

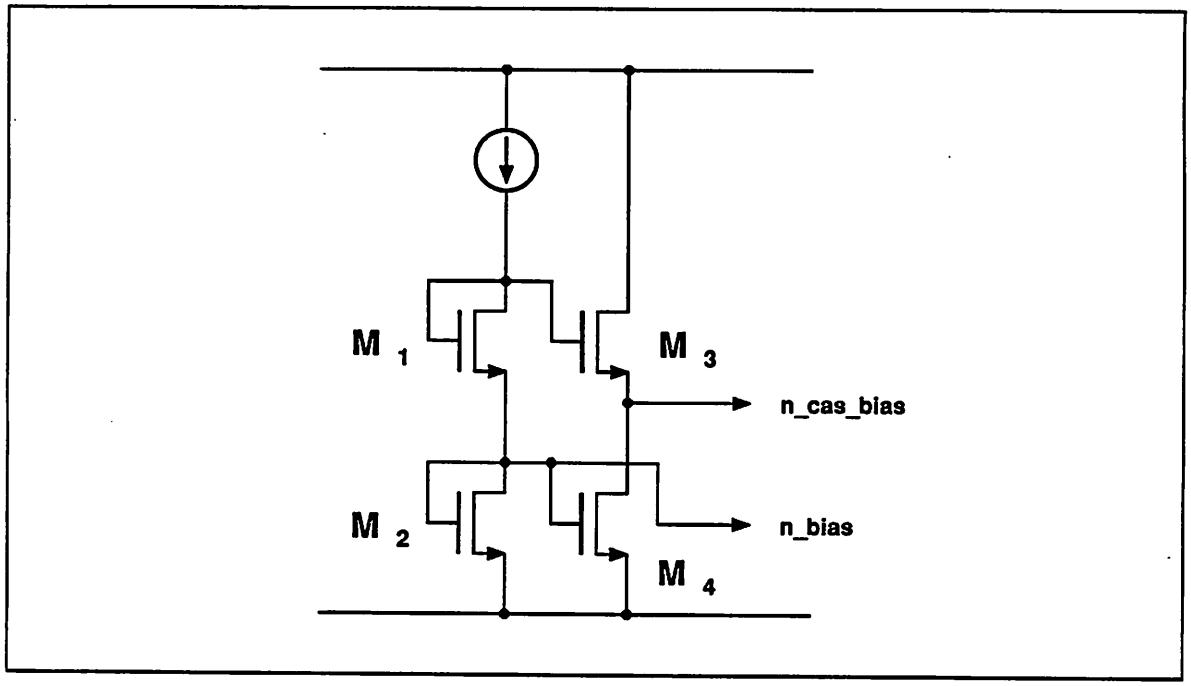

Figure 5.5.1 Simple High-Swing Biasing Circuit.

The correct drain voltage of the reference cascode transistor (M_1) can be obtained by scaling the reference cascode transistor down from the other transistors. Assuming that the MOS transistors follow the square-law behavior, the body-effect parameter is zero, and the current

levels are identical, the optimum gate voltage of the cascode transistor can be obtained if the reference cascode transistor is 4 times smaller than the other transistors. If the body-effect parameter is not zero, however, an even smaller reference cascode transistor is needed to keep the current-mirror transistor in the saturation region. How much the reference cascode transistor has to be made smaller depends on the actual process parameters such as the body effect(γ), zero-bias threshold voltage(V_{T_o}), source/drain to gate overlap(L_d), k', and the actual current level. Because these parameters are usually hard to control, it is necessary to design the reference cascode transistor smaller than otherwise necessary so that any reasonable variation in the process parameters will not drive the current mirror transistors into the linear operating region. In other words, it is necessary to sacrifice the output voltage swing of the opamp to guarantee that all transistors stay in the saturation region over different process runs.

Another limitation of this particular high-swing biasing circuit is that the finite output impedance of current-mirror transistors causes the current level of the current mirror branches to be slightly lower than that of the biasing circuit. Because the drain-to-source voltages of the current-mirror transistors also depend on many process parameters, it is impractical to compensate the current level mismatch by adjusting the device sizes. Although this is usually not a problem in a single-ended opamp, in a fully differential opamp, this will result in an appreciable systematic output common-mode error.

5.5.1. New Solution

To obtain an optimum cascode biasing voltage, a slight modification to the above high-swing cascode biasing circuitry can be made. One possible modification is illustrated in Figure 5.5.1.1. The required voltage drop to keep the current mirror transistor (M_4) in the saturation region is obtained by operating the reference transistor (M_2) in the linear region instead of by subtracting two different voltages used in the previously mentioned biasing circuit. This voltage can be obtained by properly sizing the reference transistor (M_2). It can be shown that if the body-effect

parameter is assumed to be zero, the correct reference transistor size is:

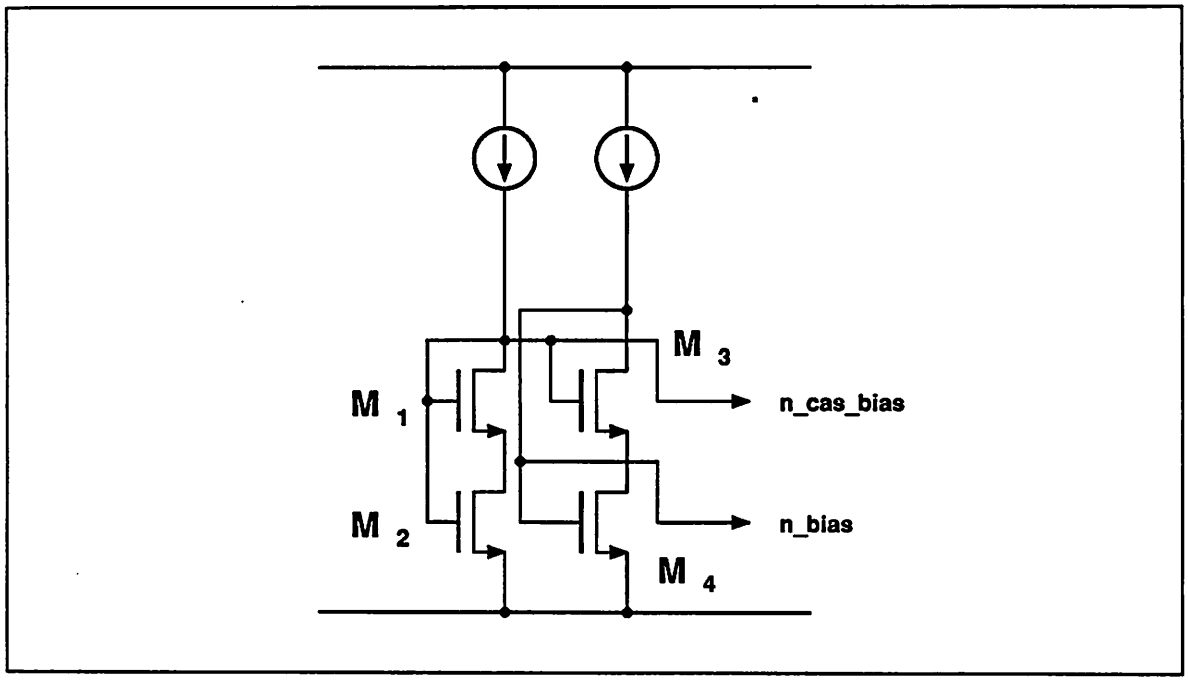

Figure 5.5.1.1 Proposed High-Swing Biasing Circuit.

$$\left[\frac{W}{L}\right]_2 = \frac{1}{(2\alpha+\alpha^2)}\left[\frac{W}{L}\right]_4, \tag{5.5.1.1}$$

where $\alpha\Delta V_4$ is the voltage needed to keep the current mirror transistor (M_4) reasonably in the saturation region, and $\Delta V_4 = (V_{DSsat_4})$. ($\alpha$ -1) is the amount of safety margin to keep the current-mirror transistor in the saturation region. Although this equation is only accurate when γ is zero, the effect of γ on α is found out to be negligible from Spice simulations.

Because all of the cascode transistors are now biased at the same voltage and current level, any variation in the threshold voltage is automatically canceled out. Furthermore, because all of the current mirror transistors in both the biasing circuit and the opamp are biased at the same drain-to-source voltages, the effect of a finite output impedance of the current mirror transistors is eliminated.

5.5.2. Limitation

To ensure proper operation of the biasing circuit, it is necessary to guarantee that the voltage drop across the drain and source of transistor M_3 to be larger than ΔV. This means that the following equation has to be satisfied.

$$V_{T_0} > \alpha \Delta V \tag{5.5.2.1}$$

If a large ΔV is needed, or if the threshold voltage of the MOS transistor is too low, it might be necessary to modify the biasing circuit so that the cascode transistor is kept in the saturation region. One way to do this is shown in Figure 5.5.2.1. A source follower is added to the original circuit so that transistor M_3 can be operated deep into the saturation region. The disadvantages of this approach are larger area and higher power dissipation.

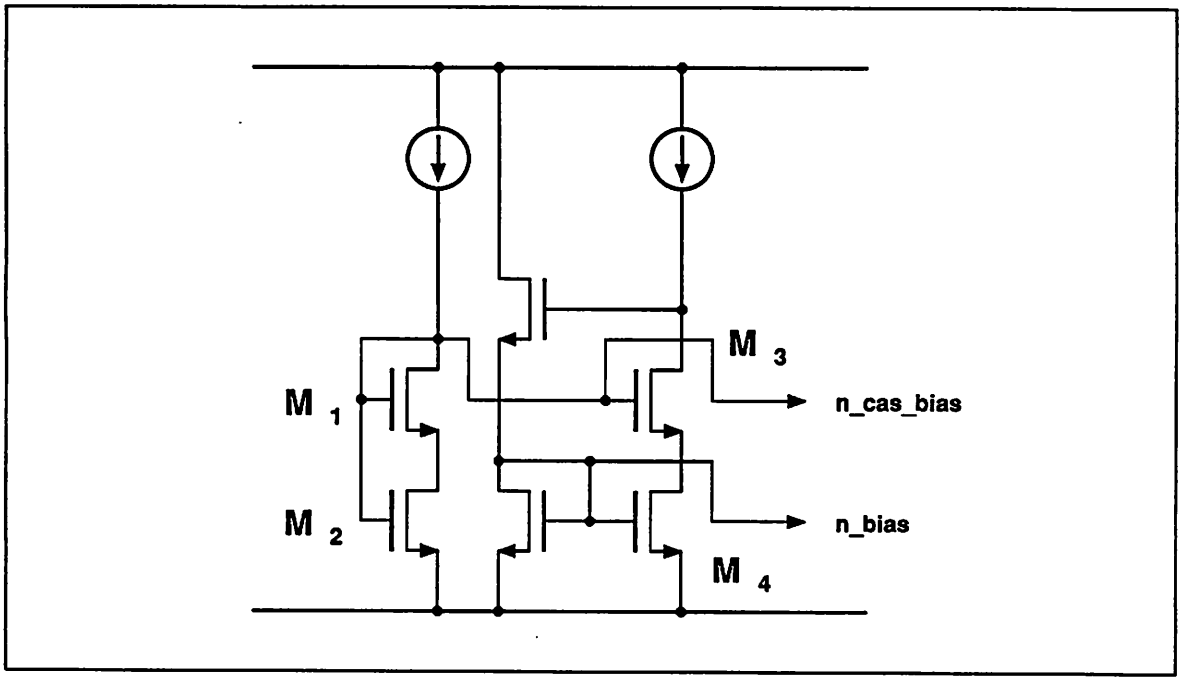

Figure 5.5.2.1 Modified High-Swing Biasing Circuit for a large ΔV or a small V_T.

5.5.3. Alternative Solution

A similar biasing circuit to the one just shown that uses only one (instead of two) current refer-

ence is shown in Figure 5.5.3.1. The basic circuit is the same as the circuit shown in Figure 5.5.1.1, except that the cascode biasing circuit is obtained from a voltage divider that is stacked on top of transistor M_3. This circuit works only with a high voltage power supply, however, because of the large number of stacked transistors.

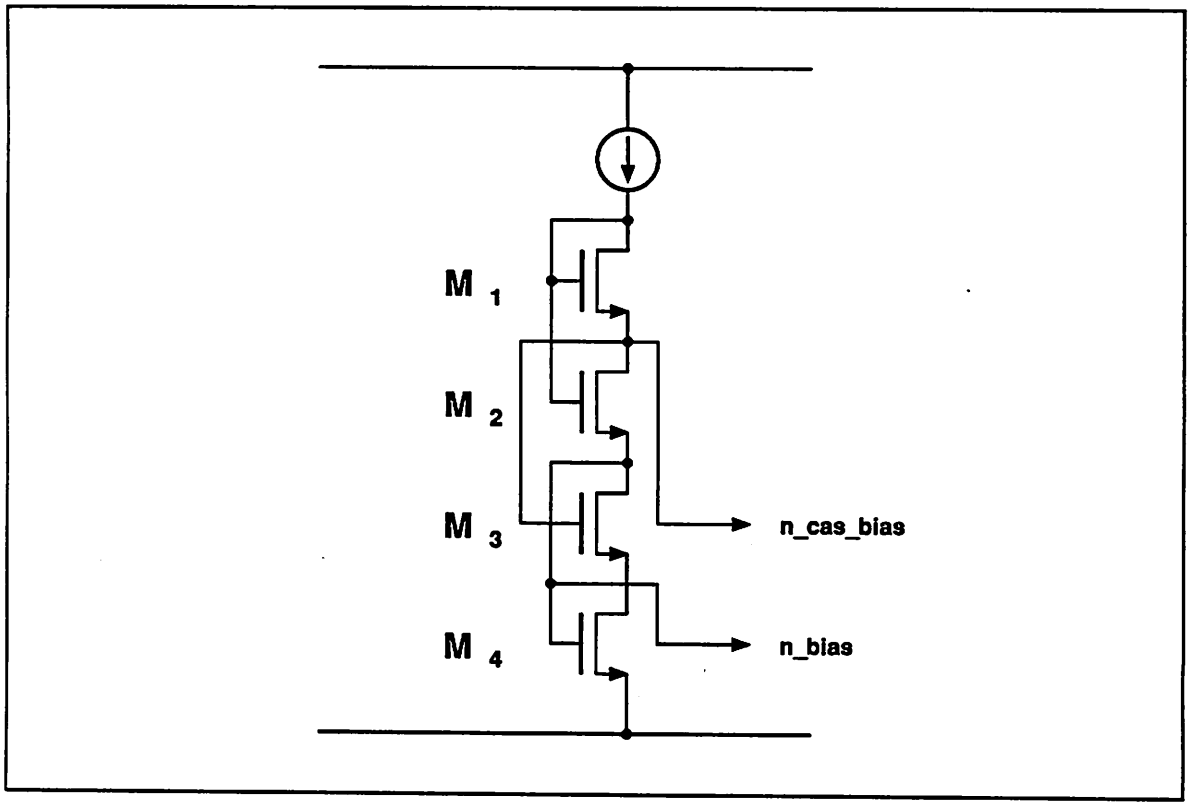

Figure 5.5.3.1 Simpler High-Swing Biasing Circuit for High Power Supply Voltage Applications.

5.6. COMPARATOR

The accuracy requirement of the comparators in each stage of a direct implementation of a pipelined A/D converter is at least a couple of bits more than the number of bits that are left to be resolved from that stage on. This means that extremely accurate comparators are normally required in the first few stages of a high-resolution pipelined A/D converter. Such comparators, however, require unnecessarily large amount of silicon area and high power dissipation because they can only be obtained using multi-stage offset cancellation techniques.[34]

The biggest problem of all is that even though accurate comparators are available, accurate A/D subconversion still can only be performed after all of the analog circuits settle to the required accuracy. This can be the major speed limiting factor in a pipelined A/D converter that uses successive-approximation sub-ADCs, especially when a large number of bits must be resolved in each stage. For example, in the prototype pipelined A/D converter, because 5 bits are resolved in each stage, 5 long clock cycles would be normally needed to obtain an accurate A/D subconversion.

To speed the A/D subconversion, digital-error correction can be used. As discussed in chapter 3, digital-error correction allows the comparators to make errors of up to +/–0.5 coarse LSB of the subconverter. Therefore, the comparators in the prototype A/D converter need only to be accurate to around 5-6 bits instead of around 15 bits without digital-error correction. Furthermore, a much shorter clock cycle time can be used in the successive-approximation subconversion because the analog circuits have to settle to only around 6 bits during the A/D subconversion cycles. A comparator with such a low-accuracy requirement can be readily obtained without the use of any offset cancellation. As such, the comparator can be optimized for fast response time as opposed to for high accuracy.

5.6.1. Solution

A simple comparator design, illustrated in Figure 5.6.1.1, was chosen to meet the much relaxed accuracy requirement. The comparator consists of a single-stage, low-gain differential pair followed by a self-regeneration latch made from strobed cross-coupled inverters. To speed up the overload recovery time from the previous decision cycle, a reset switch is added to equalize the outputs of the cross-coupled inverters before a new comparison is made.

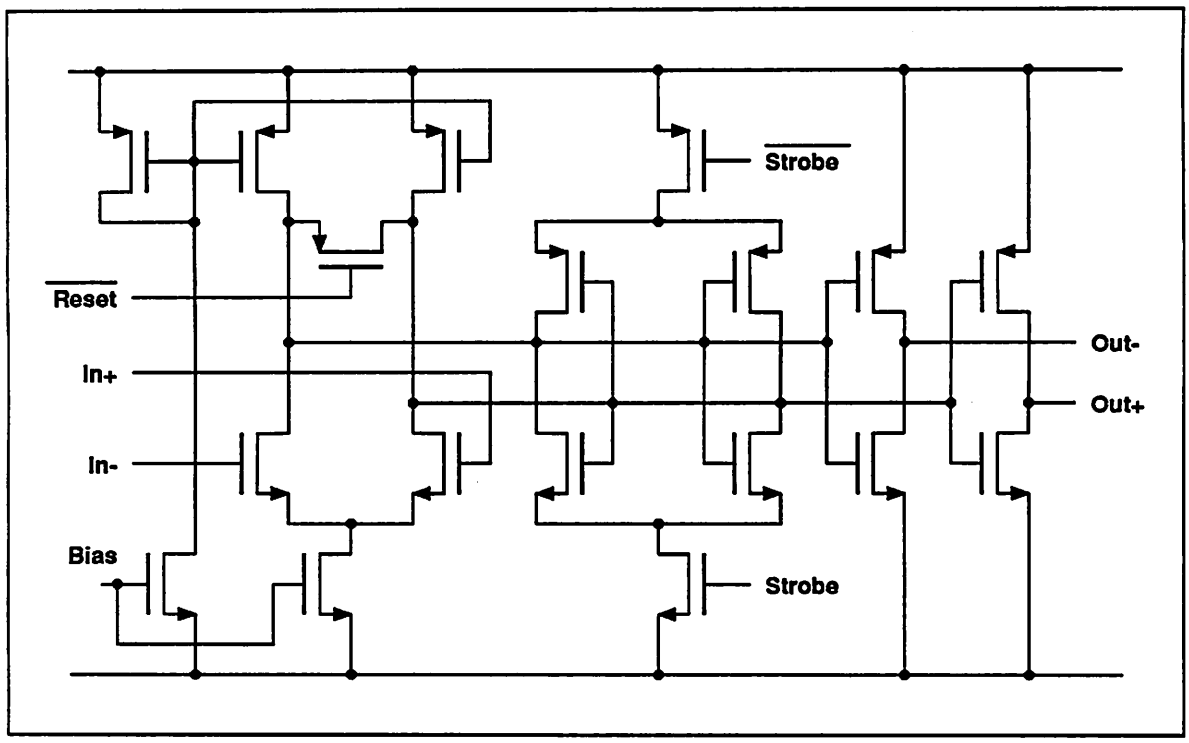

Figure 5.6.1.1 Proposed Comparator Circuit Diagram.

5.6.2. Cross-Coupled Inverters

A strobed cross-coupled inverter latch [35] was selected because it can produce a fast regeneration time and be made relatively insensitive to mismatch in the transistor sizes and parasitic loading capacitances. The cross-coupled inverter latch can achieve a fast regeneration time mainly because of the large current available during the regeneration time and the small capacitive loading at the output nodes. Extremely fast regeneration time can be obtained by making the strobing switches large and/or by turning on these strobing switches hard as fast as possible. The disadvantage of turning the switches on hard too fast is that small transistor and capacitive loading mismatch can result in a large offset voltage error. The faster the strobing switches are turned on, the larger the offset voltage will be.

To overcome the offset voltage caused by the various sources of mismatch and to maintain a fast response time, the strobing switches can be initially turned on slowly to let the cross-

coupled inverters to regenerate in the direction of the initial state, which is set up by the outputs of the differential pair. Once enough voltage difference is developed across the two outputs of the cross-coupled inverters, the strobing switches can then be turned on harder and/or faster. One way to facilitate this two-way strobing technique is to use multiple strobing switches. A smaller strobing switch can be turned on during the initial phase of the regeneration time, and a bigger switch can be turned on later to provide large latching current. Alternatively, the control signals to the strobing switches can be turned on in such a way that the strobing switches are turned on lightly at first and harder later.

To minimize the offset further, two identical output buffers are connected to the output of the cross-coupled inverters so that parasitic mismatch on the external loads does not directly affect the cross-coupled inverters. These buffers are also used to isolate the external loading capacitances from the cross-coupled inverter to minimize the regeneration time.

5.6.3. Low Gain Differential Pair

A differential pair is used to isolate the high-impedance nodes of the capacitor array in the charge-redistribution sub-ADCs from the low-impedance nodes of the cross-coupled inverters. It is also used to reduce the random offset voltage of the cross-coupled inverters. A relatively small gain is chosen so that a common-mode feedback circuit is not needed. The use of a common-mode-feedback circuit should be avoided in the comparator to minimize the loading capacitances. Besides, a large open-loop gain will not help if a fast response time is needed. This is because for a fixed and short period of time that is available for the differential pair to amplify the small input signal, the accuracy performance of either the high-gain and low-gain differential pair is to a first order approximation determined only by the amount of loading capacitances and input transconductance of the differential pairs. Therefore, unless extremely accurate comparators are required and plenty of time is available, a low-gain differential pair is sufficient.

5.7. CAPACITOR MATCHING CONSIDERATION

As mentioned earlier, although high differential linearity can be obtained using the reference-feedforward correction, the maximum achievable integral linearity of the overall converter would still be limited by the linearity of the sub-DACs, especially that of the 1^{st} stage. When charge-redistribution subconverters are used, the linearity of the sub-DAC can be mostly determined by the mismatch between the unit capacitors in the capacitor array. The smaller the mismatch, the more linear the sub-DAC will be. To effectively minimize the mismatch, it is important to identify the various types and sources of mismatch and to apply appropriate measures.

In principle, most sources of mismatch can be categorized into random and systematic mismatch. Random mismatch is caused by, for an example, random variations in the oxide thickness and edge etching rate. Unfortunately, there is nothing much that can be done to overcome this type of mismatch aside from using costly trimming or self-calibrating techniques, or using a bigger capacitor array. However, because the random mismatch drops only by a factor two every time the size of the unit capacitors is quadrupled, increasing the capacitor size is not always practical for most applications that require an extremely linear A/D converter. Not only is the area taken by the capacitors too large, more importantly the conversion speed can be too slow unless everything else is scaled up accordingly. Therefore, above a certain linearity requirement, it might make more sense to just use trimming or self-calibrating techniques.

The other type of mismatch is the systematic mismatch. Systematic mismatch can be caused by, for example, shifting in one side of the capacitor layers with respect to the other side (to be more precise, this is caused by mask alignment errors). This can cause the main and parasitic capacitance at different sites of the capacitor array to be different, especially in a large capacitor array. Systematic mismatch can also be caused by the etching rate variation due to proximity effects. For an example, the etching rate in the middle of the capacitor array might be different from that at the array boundary. Another possible source of systematic mismatch is the intercon-

nect layer of the capacitors. This might be important in a large capacitor array because it is usually impractical to make the capacitor in the middle of a large array have the same amount of interconnect parasitic capacitance to that of the one at the outer part of the array.

Unlike the random mismatch, however, it is relatively simple to reduce the effects of systematic mismatch, especially when the capacitor array is made from very few unit capacitors, such as when only 5 bits are resolved in each stage. Since it is practical to arrange up to 32 capacitors in a single or double-row arrangement, it is easy to ensure that each unit capacitor has equal parasitic capacitance. It is also practical to arrange the unit capacitors so that the effects of capacitor-mask alignment errors and linear processing gradients can be reduced significantly. One such possible capacitor arrangement is illustrated in Figure 5.7.1. However, because it only uses a first-order twisted arrangement, only the ratio of the first half over second half of the capacitor array is guaranteed to be unaffected by systematic etching error or linear processing gradient. A slightly improved structure that uses a second-order twisted capacitor arrangement is illustrated in Figure 5.7.2.

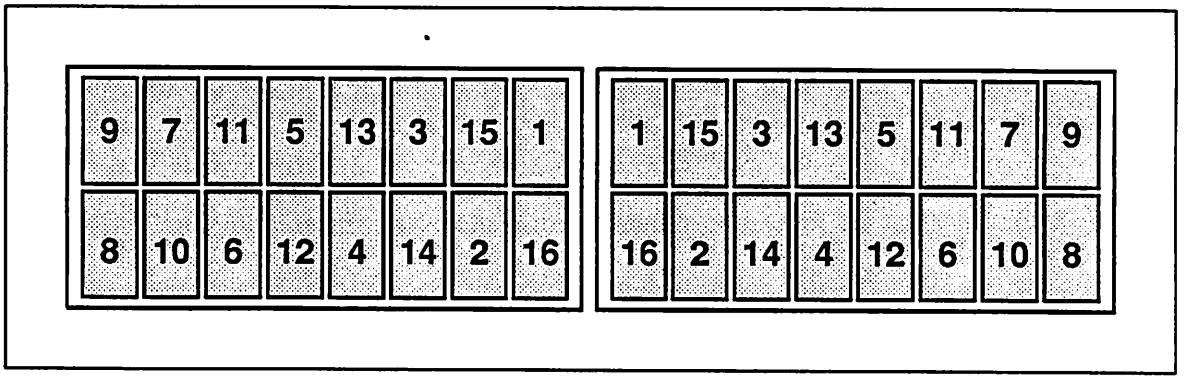

Figure 5.7.1 Layout Arrangement of the Capacitor Array.

To minimize the mismatch caused by proximity effects, dummy capacitors can be used. Dummy capacitors, however, can slow the converter down and also reduce the voltage swing that is available to the input of the comparator because the capacitance of the dummy capacitors

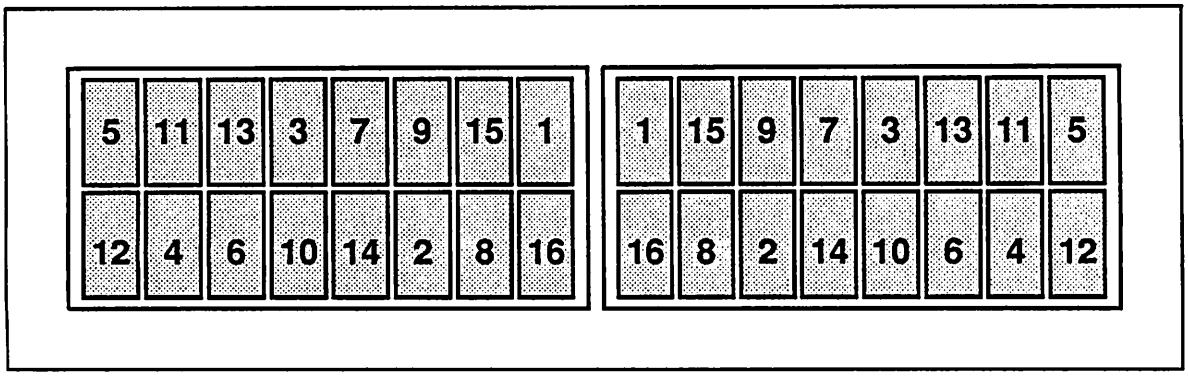

Figure 5.7.2 Improved Layout Arrangement of the Capacitor Array.

directly contributes to the summing node parasitic capacitance, unless the dummy capacitors are floating. The solution is to use dummy capacitors only at the end of the capacitor array where they are mostly needed.

5.7.1. Binary Capacitor Placement

It is also possible to reduce the effects of systematic mismatch of the capacitor array even if a binary capacitor array is used. All layout techniques that apply to the linear array also apply to the binary array. For example, if not too many unit capacitors are used, it is also possible to arrange the first half of the capacitor array so that it is equal to the other half of the array regardless of any systematic mismatch and linear processing gradient. A possible arrangement of the binary capacitor array that minimizes the systematic mismatch is illustrated in Figure 5.7.1.1.

5.7.2. Linear vs. Binary Array

It is well understood that the maximum integral nonlinearity usually occurs around the major carry. This is true whether a binary capacitor array or a linear capacitor array is used. In this prototype, a linear capacitor array was chosen despite the fact that it requires many more switches than that required in a binary capacitor array. This is because when a linear capacitor array is used, the resultant integral nonlinearity of the overall A/D converter changes more gradually with

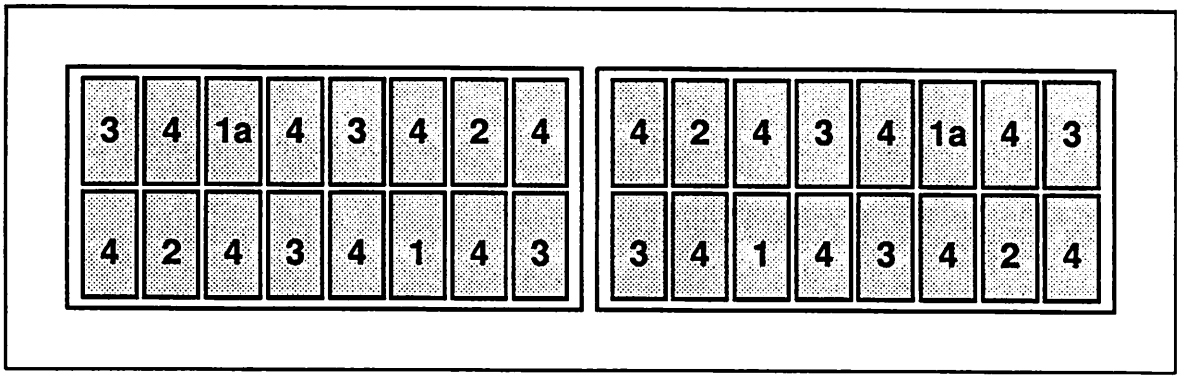

Figure 5.7.1.1 Layout Arrangement of the Binary Capacitor Array.

changes in the digital code. This is important in some data communication applications such as the ISDN in which intermodulation errors caused by heavy nonlinearity cannot be canceled easily.

The other reason is that in order to reduce the effects systematic mismatch as much as possible, it is necessary to layout the capacitor array so that each unit capacitor has its own switches, whether a linear capacitor array is used or not. The only difference in the area utilization would then be the extra area of the binary-to-linear digital decoding circuitry that is needed in a linear capacitor array. But, this difference is insignificant for a 5-bit capacitor array.

5.8. INPUT-REFERRED NOISE

Because of the relatively large number of bits that are resolved in the 1^{st} stage, the equivalent input referred noise voltage is theoretically:

$$v_{n;eq}^2 \approx v_{n;S/H}^2 + v_{n;opamp}^2 \tag{5.8.1}$$

where $v_{n;S/H}^2$ is the equivalent noise power of the input S/H capacitors and $v_{n;opamp}^2$ is the equivalent noise power of the opamp in the 1^{st} stage. For a fully differential S/H, the equivalent noise power of the input S/H capacitors is:

$$v_{n;S/H}^2 = 2\,\frac{kT}{C_s}, \tag{5.8.2}$$

where k is the Boltzmann's constant, T is the absolute temperature, and C_s is the total value of

one side of the input sampling capacitors. Assuming that the noise contributions from the current source pullup transistors, source followers, and the second stage of the two-stage opamp are negligible compared to that of the input transistors and the diode connected loads, the equivalent input noise power of the two-stage opamp is:

$$v_{n;opamp}^2 \approx 2\,\pi\, f_{BW} \left[\frac{C_s + C_i + C_{ip}}{C_s} \right]^2 \left[\frac{2}{3} \frac{4kT}{g_{m;input}} \right], \tag{5.8.3}$$

where C_s, C_i, and C_{ip} are the sampling, integrating, and parasitic summing node capacitance respectively. With a typical C_s of around 5 pF, $g_{m;input}$ of around 1 mS, and a maximum closed-loop bandwidth, f_{BW}, of around 10 MHz, the equivalent input referred noise voltage is expected to be around:

$$v_{n;opamp} = 60\,\mu V \tag{5.8.4}$$

In comparison, the design value for 1 LSB (the smallest step) is $1mV$.

5.9. BINARY-TO-LINEAR DECODING CIRCUIT

A binary-to-linear decoding circuit is needed to control the bottom-plate switches of a linear capacitor array. Because the truth table of such a decoding circuit normally resembles a bar-graph and thermometer decoding circuit, this type of decoding circuit is usually referred to as a bar-graph or thermometer decoding circuit. The binary-to-linear decoding circuit that is used in this prototype A/D converter, however, also has to individually control the unit capacitors during the reference-feedforward-correction cycles. To minimize the silicon area utilization of the decoding circuit, a multi-level decoding circuit was used. This circuit is illustrated in Figure 5.9.1.

5.10. SUMMARY

Using the circuit design techniques described above, a prototype pipelined A/D converter was laid out in a 3-μm double-poly CMOS processing technology provided by MOSIS. Currently, only the essential parts of the converter such as the opamps along with their biasing circuitry, comparators, analog switches along with their decoding circuitry, capacitor arrays, and binary-

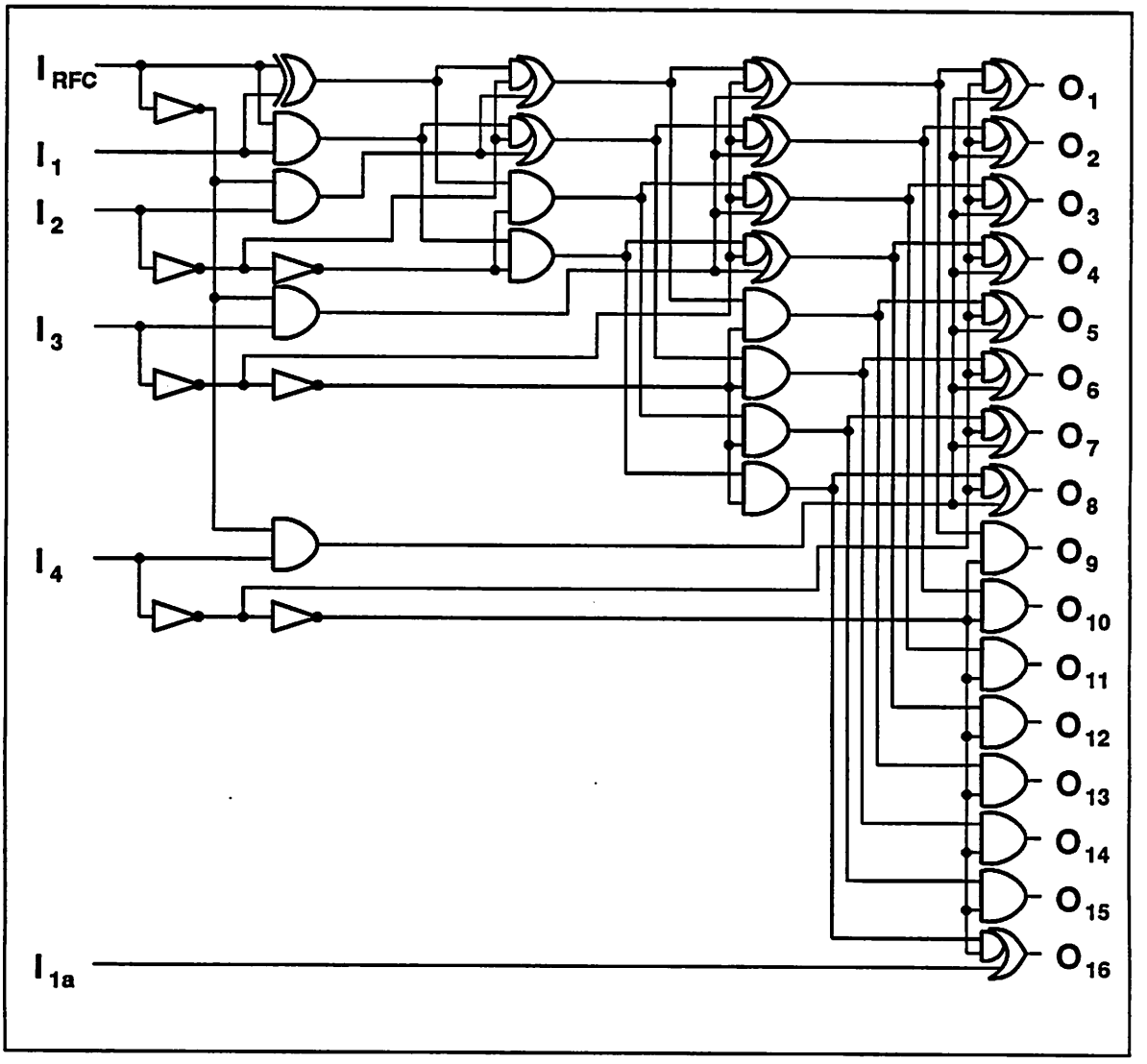

Figure 5.9.1 Binary-to-Linear Decoding Circuit Including Reference Correction Control.

to-linear decoding circuitry have been integrated in the prototype converter. The prototype converter was laid out as symmetrical as possible to avoid the effects of geometrical processing variations on the overall converter linearity and input-referred offset voltage. As can be seen from Figure 5.10.1, the prototype converter has been laid-out using an area that has an almost 1:1 aspect ratio. This is done in accordance to the industry standard layout practice to keep the aspect ratio as close to unity as possible.

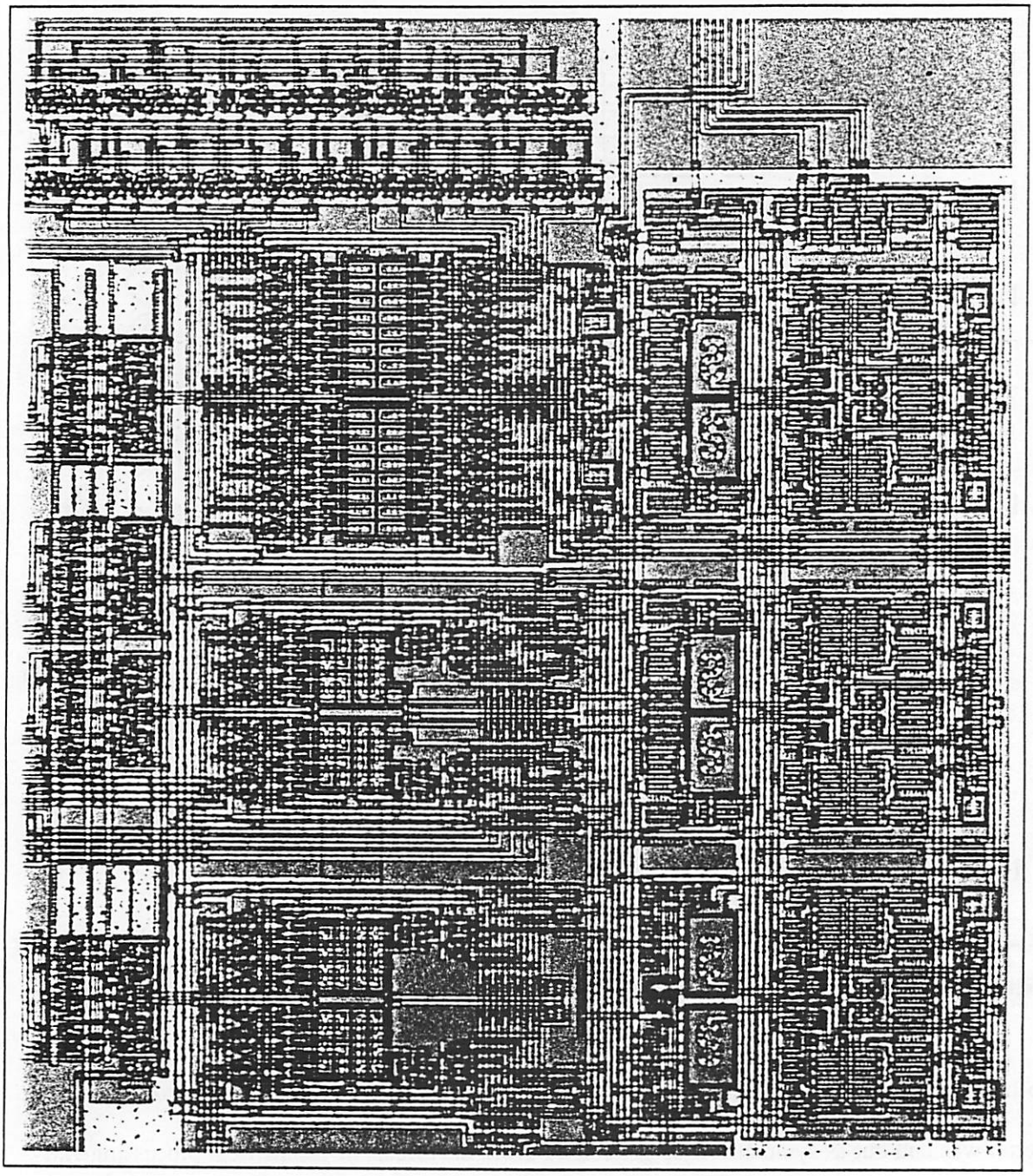

Figure 5.10.1 Chip Photograph of the Prototype 3-stage Pipelined A/D converter.

CHAPTER 6

EXPERIMENTAL RESULTS

The rest of the circuitry needed to complete the prototype A/D converter will be described in this chapter. Test results from the prototype converter will then be given. The prototype converter was tested using both code-density [36,37] and signal-to-noise ratio tests.

6.1. CONTROL CIRCUITRY

The control circuitry is grouped into three separate blocks. They consist of a clock sequencer block, a successive-approximation-register(SAR) block, and an incrementer block.

The function of the clock sequencer block is to generate the control signals for the on-chip switches. Its implementation was based on a shift-register approach similar to that illustrated in Figure 6.1.1. The shift-register based approach was chosen instead of the standard ROM based approach for the following reasons. First, because many control signals last for only one or two clock cycles, it is much easier to decode the appropriate shift-register tap signals than to decode the ROM outputs. In fact, those signals that last only for one clock cycle do not need any decoding at all, while those that last for two or three clock cycles can be decoded using simple OR(NOR) gates, and those that last for much longer clock cycles can be decoded using SET-RESET(SR) Flip-Flops(FF). Second, because of the simplicity of the decoding circuitry, the shift-register approach has the potential of reducing the delay variations between different control signals. Third, non-overlapping control signals can be easily generated by decoding the intermediate tap outputs of the shift registers. In fact, it is possible to fine-adjust the nonoverlapping time by simply changing the relation between the two phases of the main clocks. Finally, the

shift-register based approach is simple even when implemented using external parts.

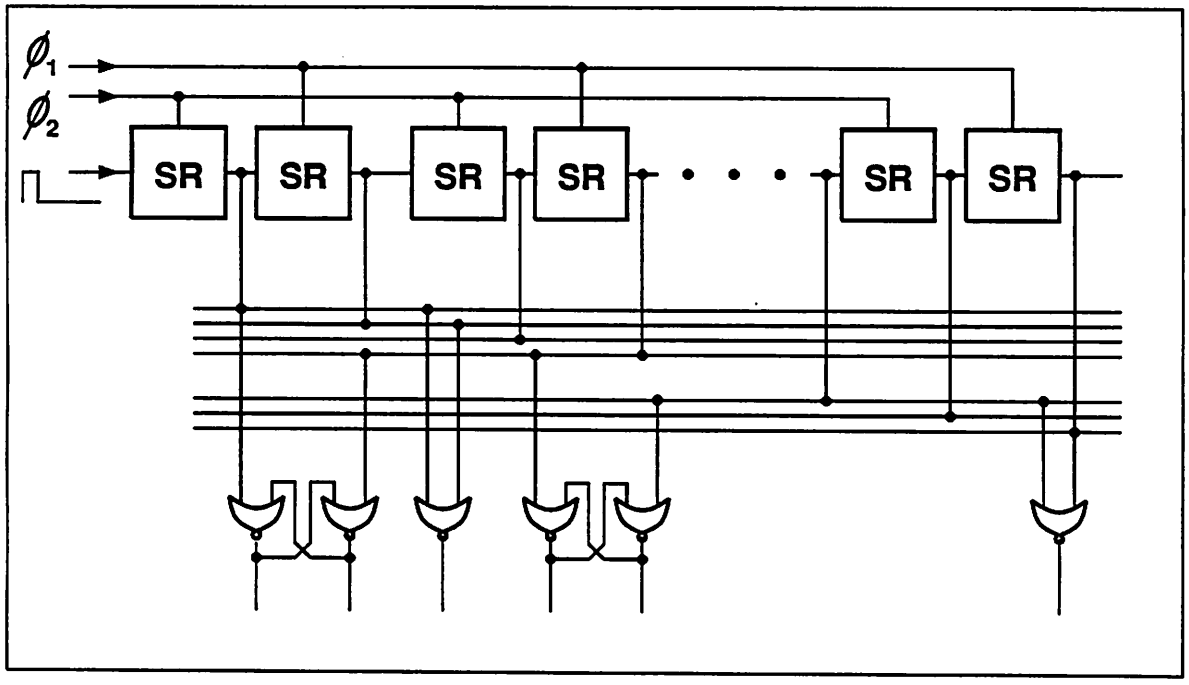

Figure 6.1.1 Shift-Register Sequencer.

The on-chip SARs can be designed using multiple input transmission-gate latches that utilize the shift-register sequencer in CMOS processing technology to minimize the silicon area. The on-chip incrementers can be designed using ripple carry half-bit adders to also minimize the silicon area utilization. An example of such SARs is shown in Figure 6.1.2, while the incrementer is shown in Figure 6.1.3. External SARs and incrementers used in the prototype A/D converter were not implemented using these approaches, however, because they would require too many low-level TTL functions. Instead, these functions are implemented using programmable-array logics (PALs).

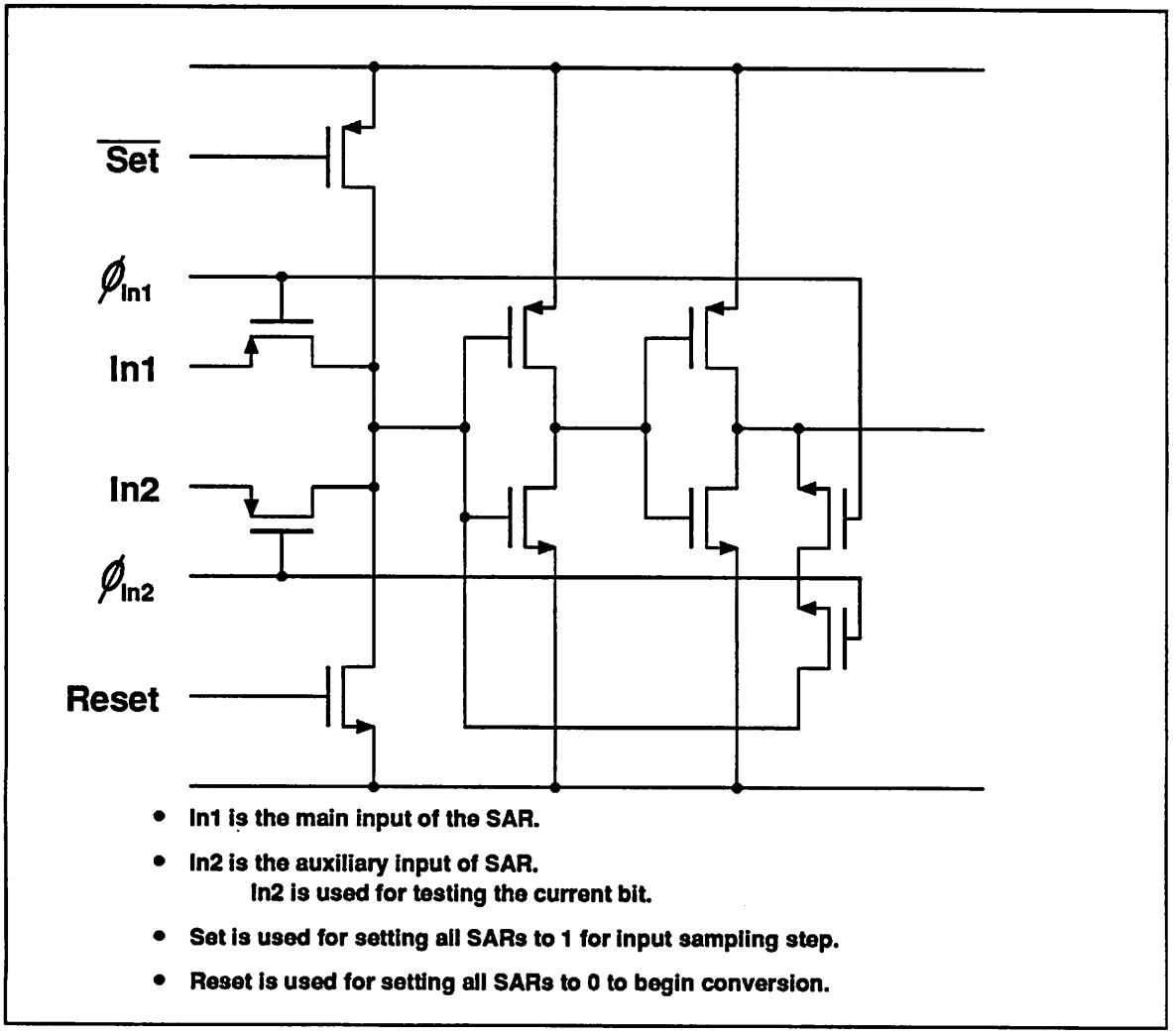

Figure 6.1.2 On-Chip SAR Implementation.

6.2. OTHER EXTERNAL CIRCUITS

Other support circuitry such as analog interface circuitry is needed to test the prototype A/D converter. Because the prototype A/D converter was designed in a fully differential configuration and all of the available signal generators in our test laboratory are single ended, a single-ended to fully differential conversion circuit is needed. Such a circuit is usually made of two opamps that are connected in a positive and negative unity-gain configuration, as illustrated in Figure 6.2.1.

A slight modification to this circuit is required, however, for the prototype A/D converter because the prototype is designed to run on a single 5-V power supply instead of the usual

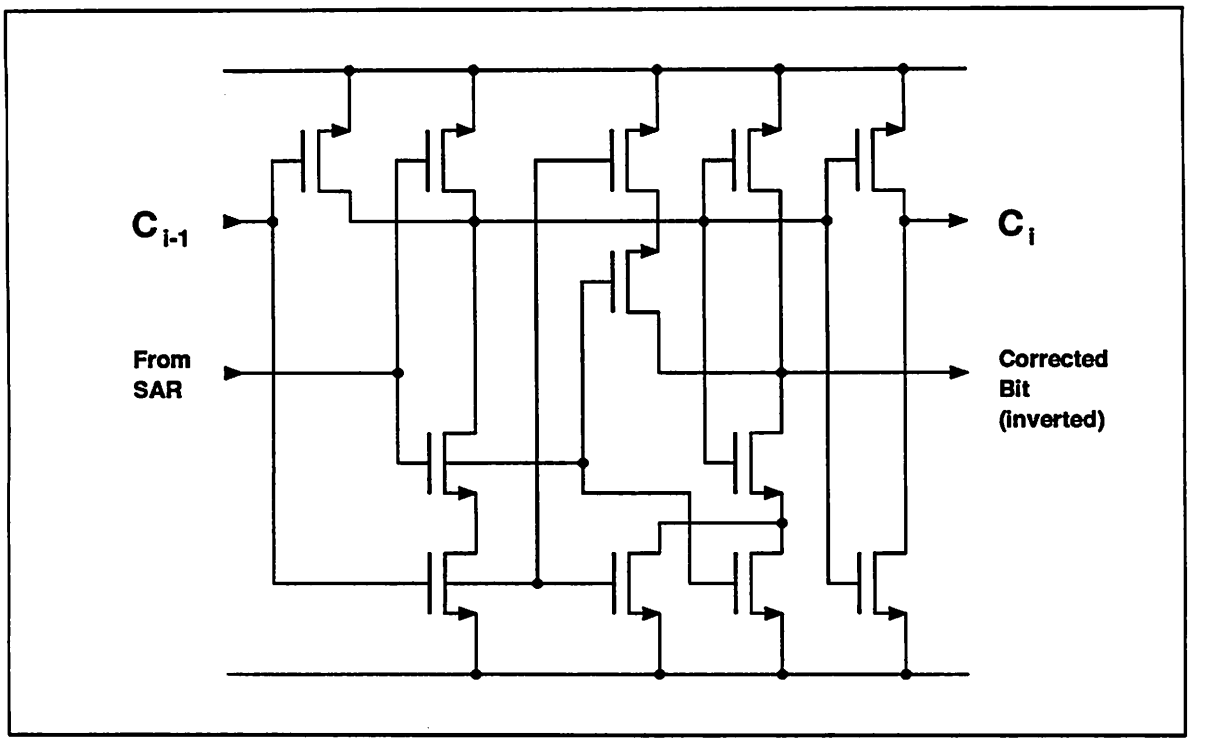

Figure 6.1.3 On-Chip Incrementer Implementation.

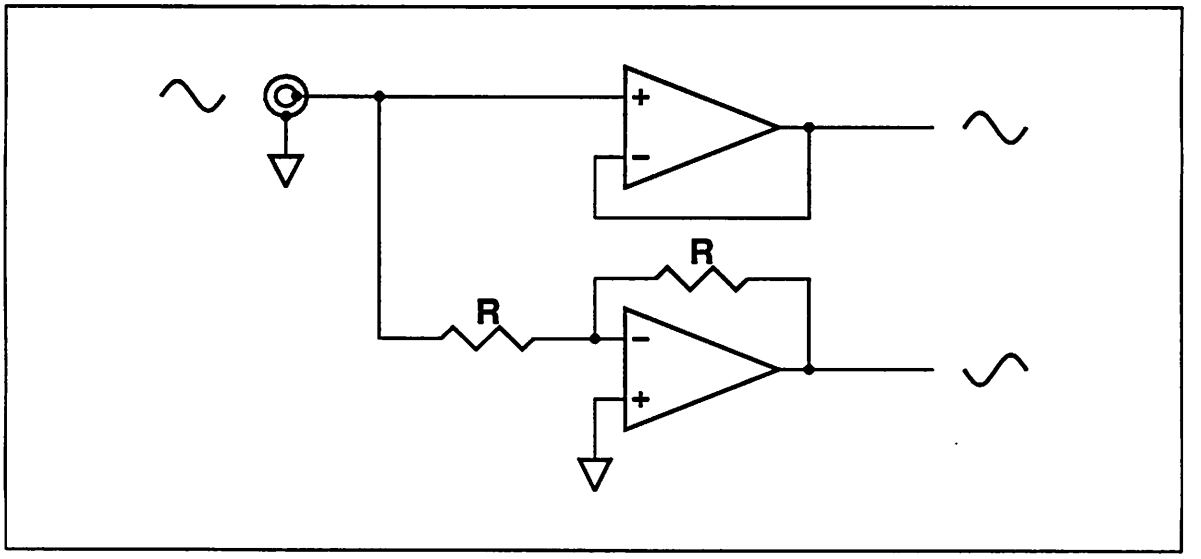

Figure 6.2.1 Typical Single-Ended-to-Fully-Differential Implementation.

dual/split power supplies. Therefore, a level-shifting function is needed to shift the input signal up by half of the power supply voltage. One way to do this is to use a capacitive-coupling

approach. Such a single-ended to fully differential conversion circuit is shown in Figure 6.2.2.

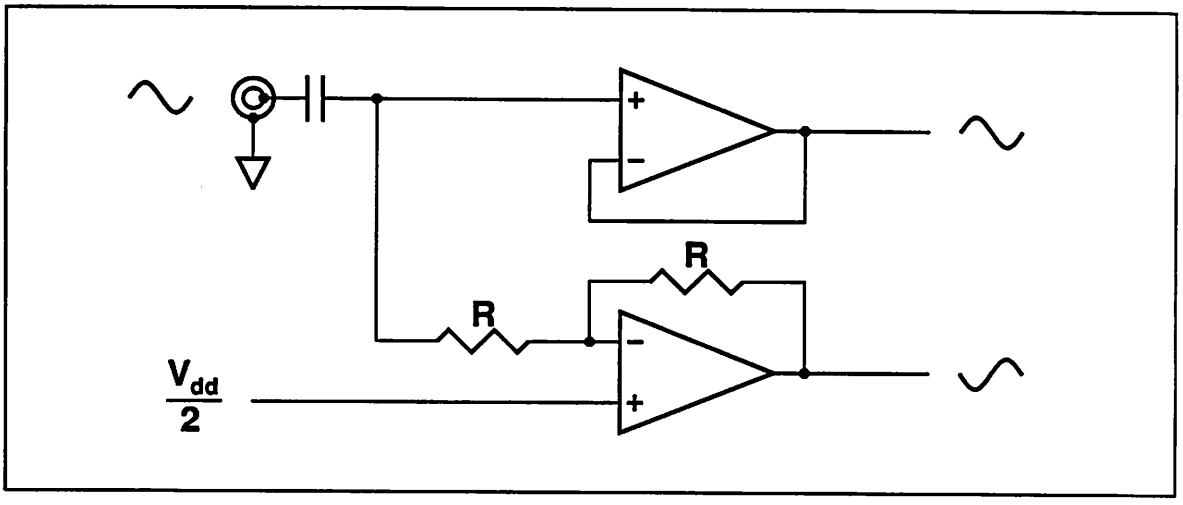

Figure 6.2.2 Single Power Supply Single-Ended-to-Fully-Differential Circuit Diagram.

Because of the use of opamps in the above single-ended to fully differential conversion circuits, however, the linearity of the opamps may limit the linearity of the signal that is fed to the A/D converter under test. To ensure that the nonlinearity measured at the output of the A/D converter is only due to the A/D converter, opamps that have a large open-loop gain are needed. This is not usually a problem at low signal frequencies. However, because of the low unity-gain bandwidth of most opamps, the opamp open-loop gain at frequencies of even around 100 kHz is typically so low that the nonlinearity of the single-ended to fully differential circuit can be easily worse than that of the A/D converter alone.

One way to avoid the use of op-amps in the single-ended to fully differential conversion circuit is to use a transformer. A bifiliar winding can be used in the secondary section of the transformer to generate both the in and out of phase sinusoidal signal. Unfortunately, we were not able to find such transformers, especially those that are linear to more than 12-13 bits in the frequency range of between 10kHz to 200kHz.

Another needed support circuitry is a single-ended to fully differential conversion circuit for

the reference voltage. However, because the reference voltage is DC instead of AC, the capacitive based level shifting or transformer coupled circuit cannot be used. To solve this problem, an active addition and subtraction circuit, illustrated in Figure 6.2.3, is employed. This kind of circuit can also be used for the input signal, but it requires more opamps than that required by the simpler capacitive-coupling approach.

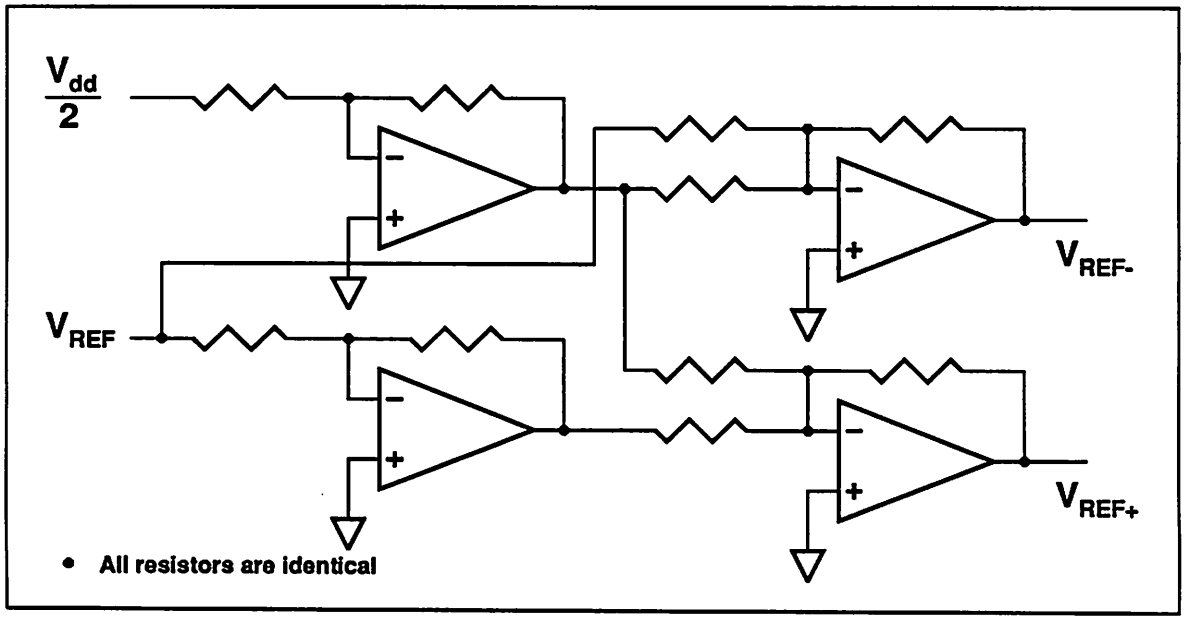

Figure 6.2.3 Fully-Differential Conversion Circuit for the Reference Voltage.

6.3. TESTING METHODS

To avoid the need of using an external D/A converter to test the A/D converter, direct digital means were used. The prototype A/D converters were tested using both code-density and SNR tests. The code-density tests are used to dynamically characterize the differential and integral nonlinearity of the A/D converter as a function of the digital output code. Because these tests do not rely on the input signal frequencies, it is possible to test the performance of the prototype A/D converter under demanding high input signal frequencies. In contrast, most commercial A/D converters are usually characterized by comparing the analog-reconstructed digital output code to the original analog input signal. As such, they can only be done with a very low-frequency (DC)

input signal. The input referred offset and noise voltage can also be tested using the code-density tests with an addition of a simple circuit that will be described later.

SNR tests are used to measure the linearity of the A/D converter indirectly. They are used more often nowadays to characterize A/D converters for signal processing applications. SNR tests are important because they can accurately measure the quantization noise and distortion components of the A/D converter in a reasonably short amount of time. Furthermore, they can show other nonidealities such as sampling jitter and extra noise which otherwise would be averaged out in the code-density tests.

6.4. CODE-DENSITY TESTS

The code-density test setup is shown in Figure 6.4.1. The digital output lines of the prototype A/D converter are interfaced to an LSI-11 which is used to accumulate the digital output codes of the A/D converter into the form of a histogram. The histogram data is then transferred to a main computer that is used to compute and generate the plots of the differential and integral nonlinearity. The main bottleneck in using this approach is that the digital interface board that we have in the LSI-11 can only accept digital data from the A/D converter at around 10 ks/s sampling rate. Fortunately, since the objective is to collect the statistic of digital output code distribution, it is possible to run the A/D converter at full speed, but to ignore most of the digital output samples and to only accept the digital output samples at a lower sampling rate.

A typical plot of the differential nonlinearity (DNL) of the prototype A/D converter as a function of the digital output code is shown in Figure 6.4.2. As can be seen from this plot, the worst case differential nonlinearity is less than +/–½LSB. This is typical of most of the working chips tested so far. Because the negative DNL never reaches -1LSB, this shows that the converter does not show any missing code.

The corresponding integral nonlinearity (INL) of the A/D converter tested above is plotted

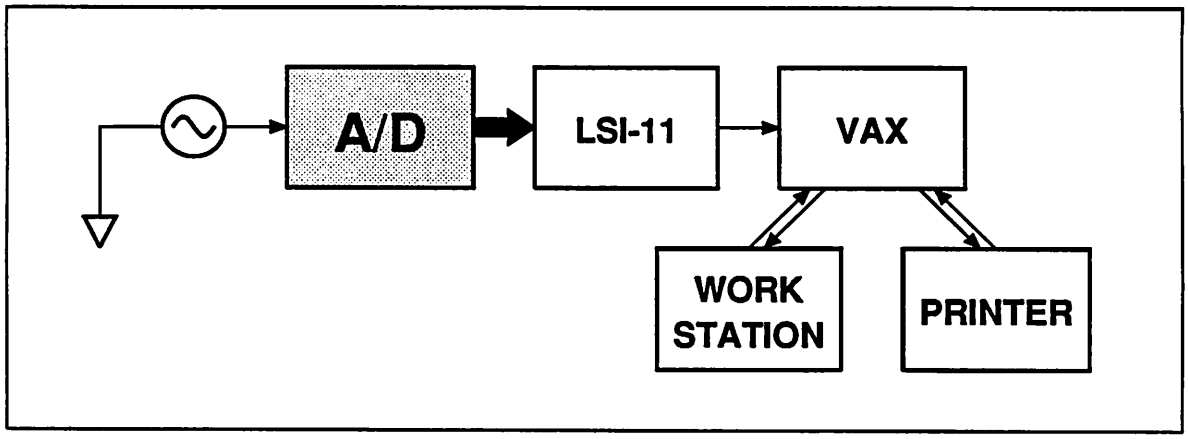

Figure 6.4.1 Code-Density Test Set-Up.

in Figure 6.4.3. The limitation of taking the digital output codes at a 10ks/s rate is that it takes too long a time to obtain the large number of samples needed for an accurate integral nonlinearity measurement. A much more troublesome factor is that during the long period of time that the A/D converter is tested, the amplitude of the test signal may drift. This worsen the plot of the integral nonlinearity. This is the major reason why the integral nonlinearity plot from the code-density tests for a high resolution A/D converter is not usually given.

Notice that the worst case INL observed from the prototype A/D converter is much better than previously observed for the size of sampling capacitors used in the A/D converter. For the total of 10pF of sampling capacitors used in the 1^{st} stage of the prototype, an INL of around 9 or at best of around 10 bits is usually expected. Instead, close to 12 bits of linearity had been observed repeatedly.

From a statistical point of view, it is further expected that the INL should vary from chip to chip. However, results obtained from many working chips that have been tested so far (especially among chips fabricated from the same process run) show that there are negligible variations in the INL. The fact that chips coming from one process run have similar nonlinearity can be explained from systematic matching errors in among some of the unit sampling capacitors of the

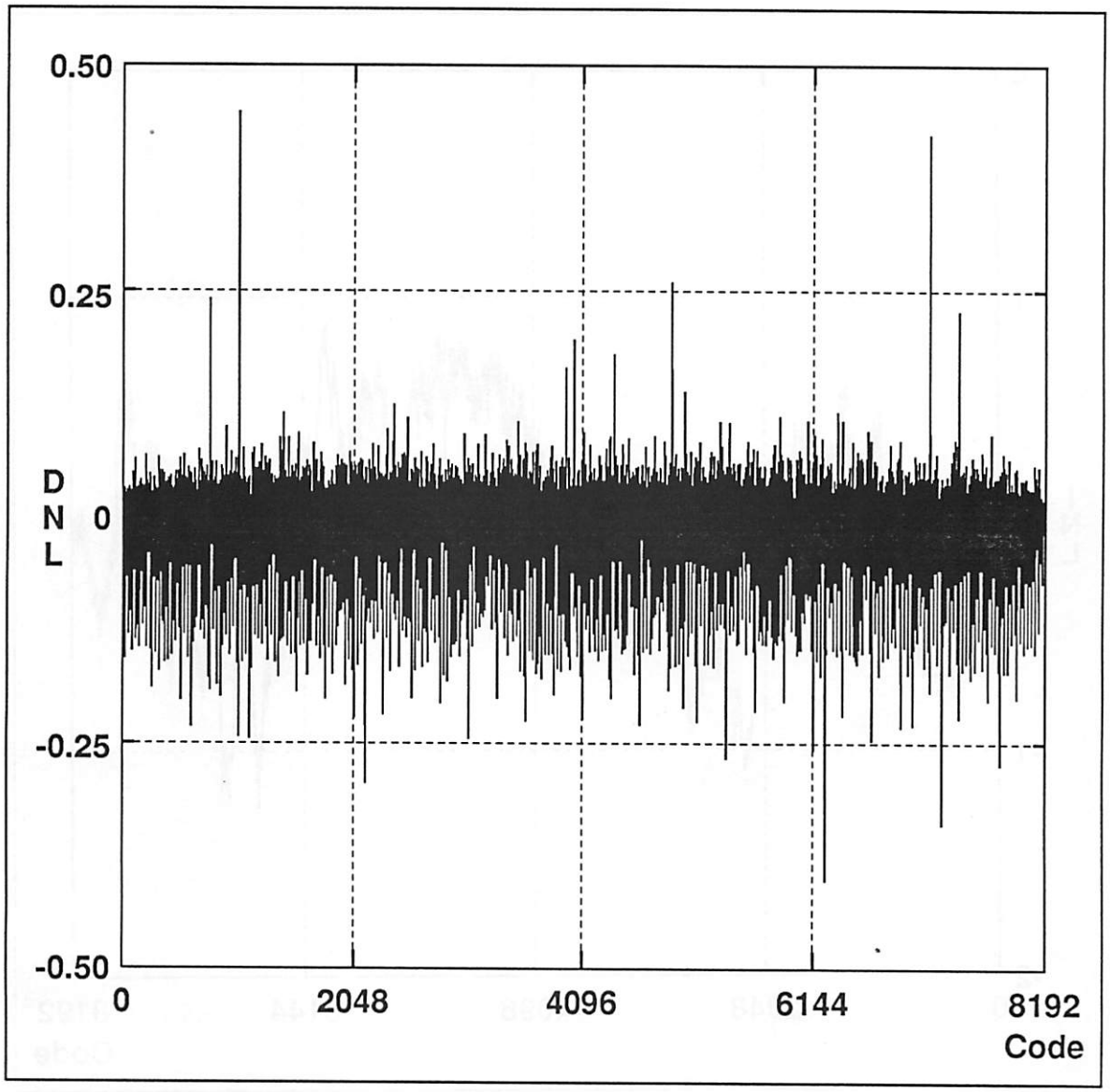

Figure 6.4.2 Differential Nonlinearity vs. Digital Output Code.

1^{st} stage. As can be seen from Figure 6.4.3, the peaks of the INL are found at around the second major carry. These peaks are not caused by random variations in the capacitor values, but are probably caused by systematic etching errors in the unit capacitors at the array boundary.

Measurement results obtained from many chips suggest that the 8^{th} and 9^{th} unit capacitor of the 1^{st} stage are always smaller by about 0.4% compared to the average of all unit capacitors. On the other hand, the random matching errors of the unit capacitors are only about $\frac{1}{4}$ smaller

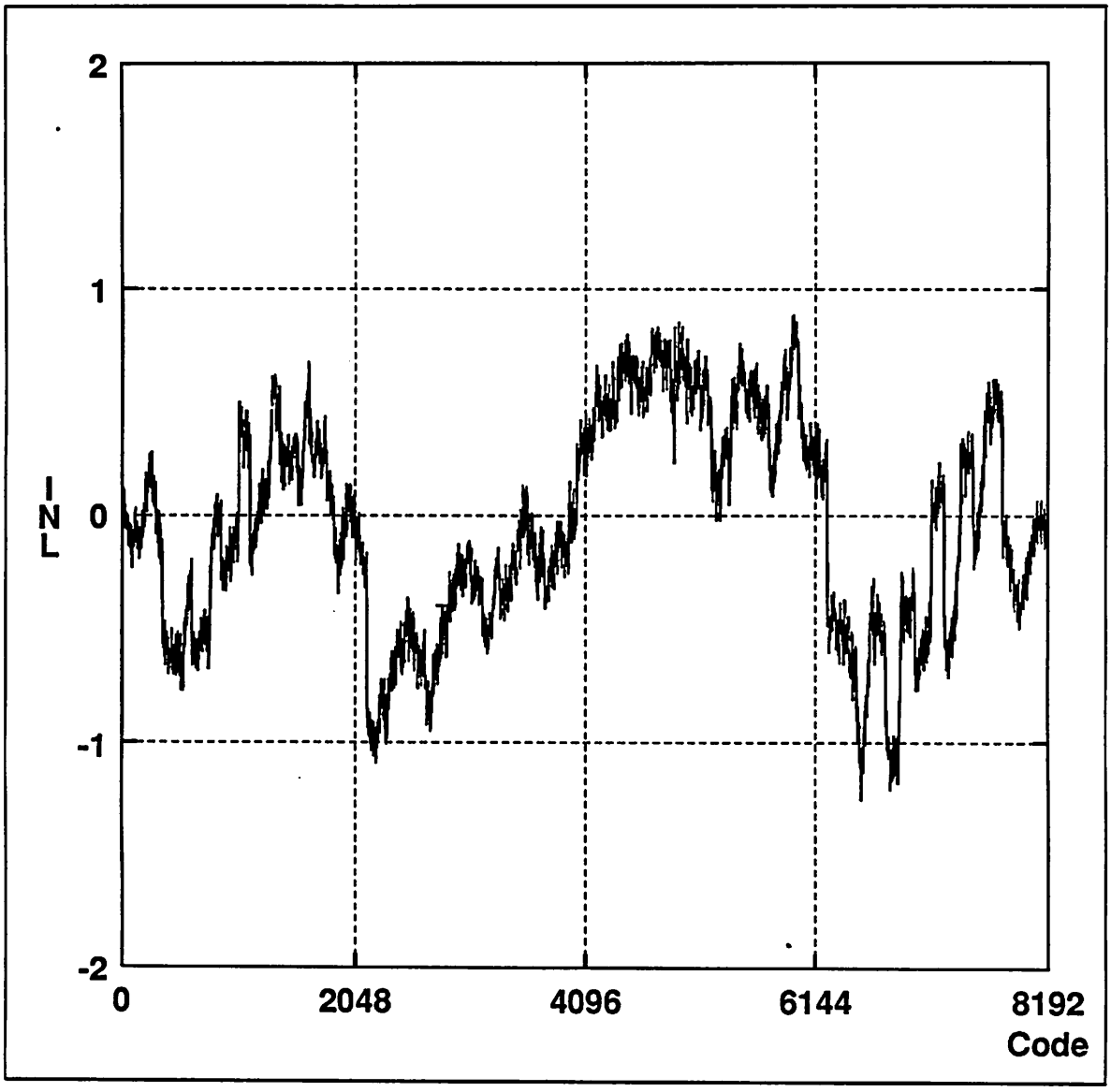

Figure 6.4.3 Integral Nonlinearity vs. Digital Output Code.

than the systematic matching errors. As a result, the systematic matching errors dominate the overall INL, and variations in the overall linearity are negligible.

6.5. SIGNAL-TO-NOISE RATIO (SNR) TESTS

An indirect way of measuring the nonlinearity of an A/D converter is to measure the distortion components produced by the A/D converter when it is being fed by a pure sinusoidal waveform signal. It is possible to do this because an ideal A/D converter would only produce quantization

noise which is white, while a nonlinear A/D converter would produce not just the quantization noise but also the distortion components at the harmonics of the applied sinusoidal input signal. There are two ways to do this measurement. One obvious approach is to first convert the digital output code of the A/D converter back into its corresponding analog signal using an extremely linear D/A converter and then to look at the converted signal through a spectrum analyzer. A better approach is to compute the distortion components directly in digital domain using Fourier Transform techniques. Such digital SNR tests are typically obtained by running a DFT (Discrete Fourier Transform) or an FFT (Fast Fourier Transform) on a sequence of stored digital output codes of the A/D converter.

The main disadvantage of using the SNR tests is that no specific nonlinearity measurements as a function of the digital output code of the A/D converter can be produced. This is because there is an infinite number of possible kinds of nonlinearity that can have the particular distortion components observed from the digital output codes of the A/D converter under test. However, the main advantage of using the SNR tests is that very fast measurements can be done because usually only a few thousand samples of the digital output codes are needed as opposed to a few hundred million samples for each code-density test. Correspondingly, the SNR tests do not suffer from the amplitude drift problems associated with the code-density tests. The SNR tests, however, are sensitive to frequency drift in the input signal and sampling clock, as well as jitter in the sampling clock. Therefore, a good test setup is important.

A plot of the output of the FFT tests from one of the prototype A/D converters for a full scale sinusoidal input signal is shown in Figure 6.5.1. This plot is typical for all of the working chips that have been tested so far. This plot is obtained from taking 2^{15} digital output codes for high plotting resolution in the frequency axis.

The above test was repeated for various input signal amplitudes and the results are summarized in the plot of SNR vs. analog input level in Figure 6.5.2.

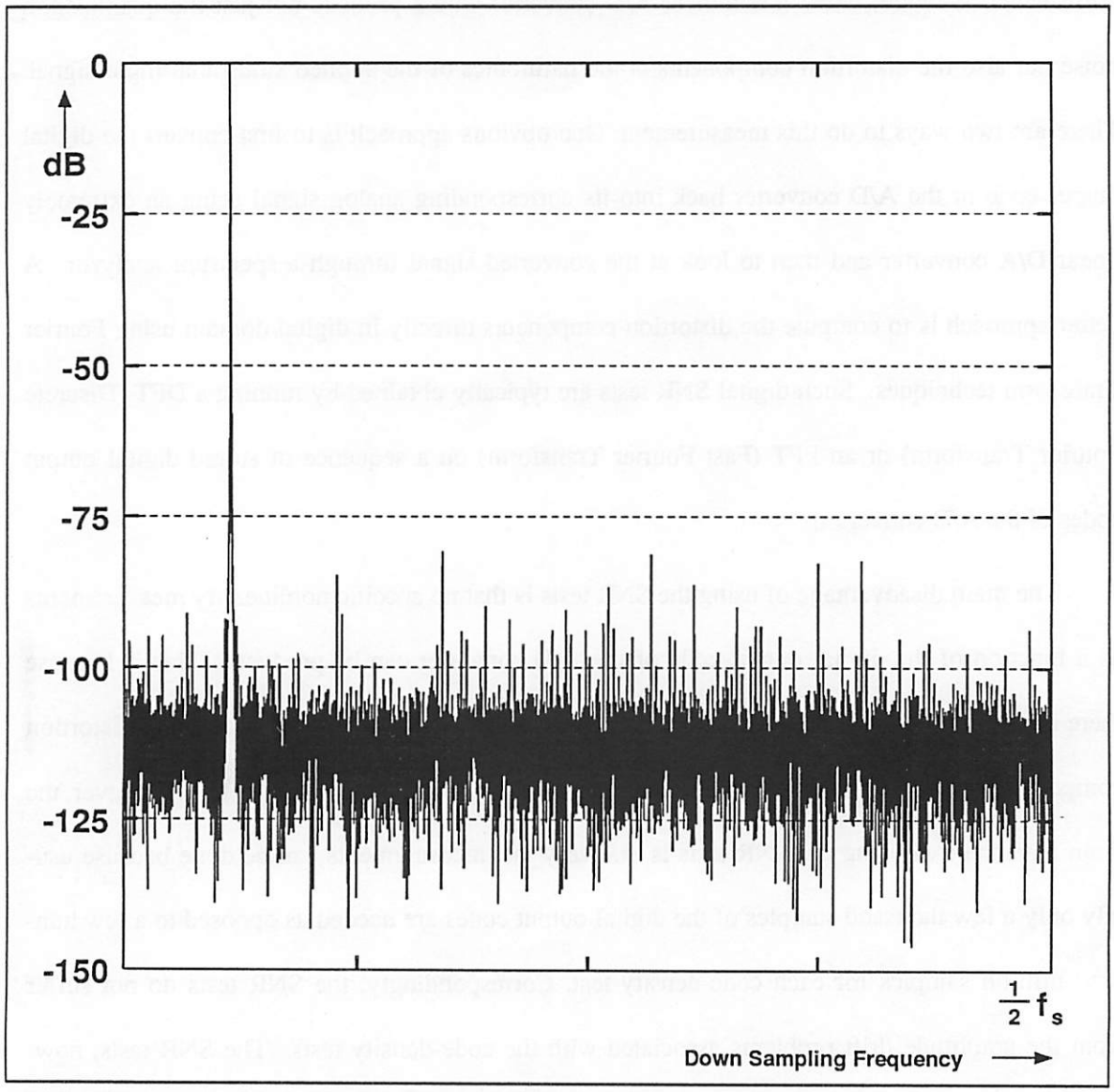

Figure 6.5.1 FFT Plot for the Full Scale Input Voltage.

6.6. POWER SUPPLY REJECTION RATIO (PSRR) TESTS

The power supply rejection ratio is a measurement of the amount of the unwanted signal or noise from the power supply line that gets through to the output of the A/D converter. These tests are usually done by modulating the power supply line with a known signal such as a sinusoidal waveform signal. The problem in performing this test is that it is hard to modulate the power supply voltage since the impedance of the power supply is usually very low.

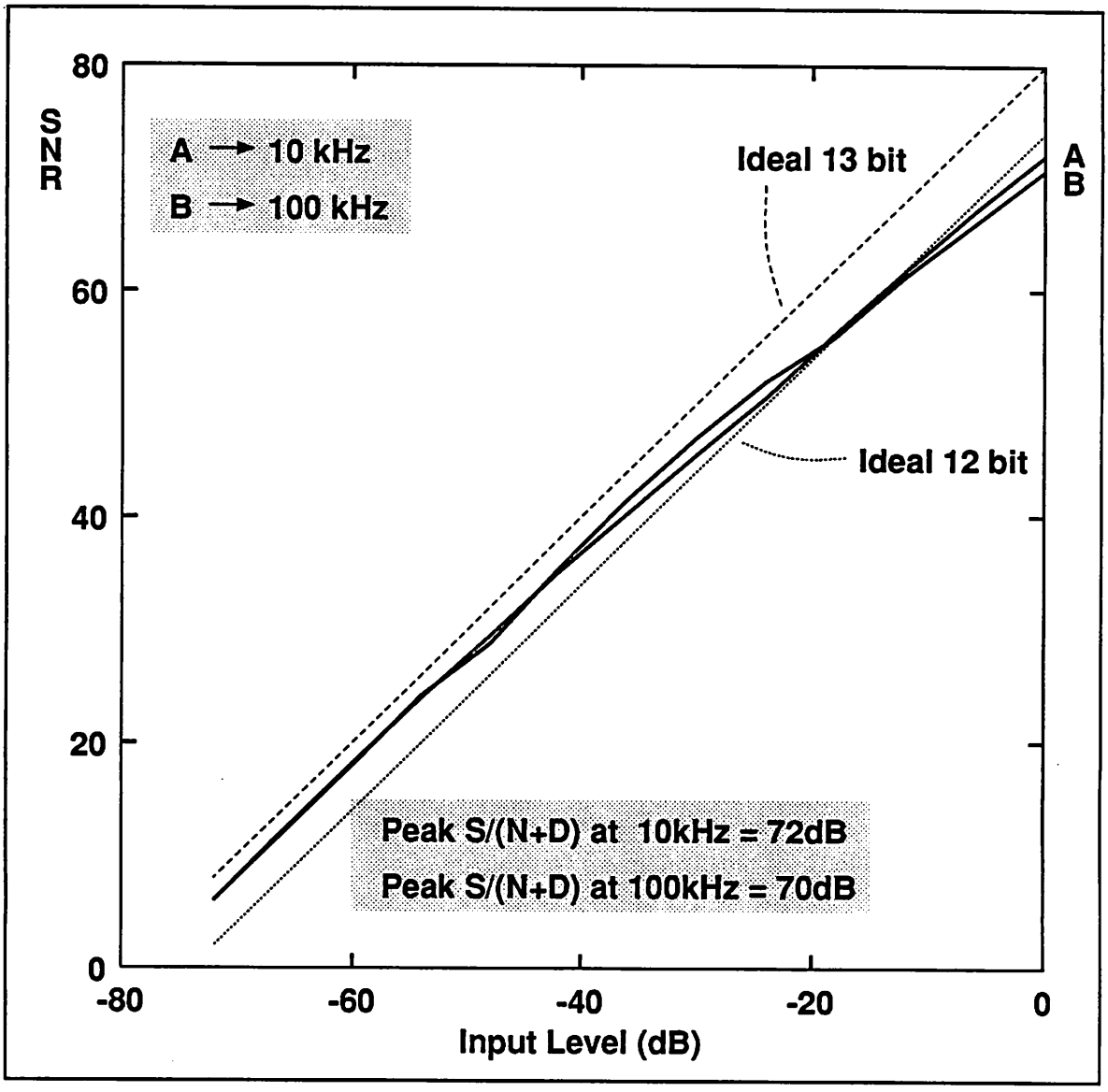

Figure 6.5.2 SNR vs. Analog Input Level.

To overcome the close to zero line impedance of the power supply, a circuit shown in Figure 6.6.1 is used. The advantage of such a circuit is that the output impedance of the test source can be made much lower than if it has to drive the power supply line directly. Because of the large number of decoupling capacitors used on the test board, however, the power supply rejection tests are limited to only tens of kHz.

Normally power supply rejection tests are done while the actual input of the A/D converter

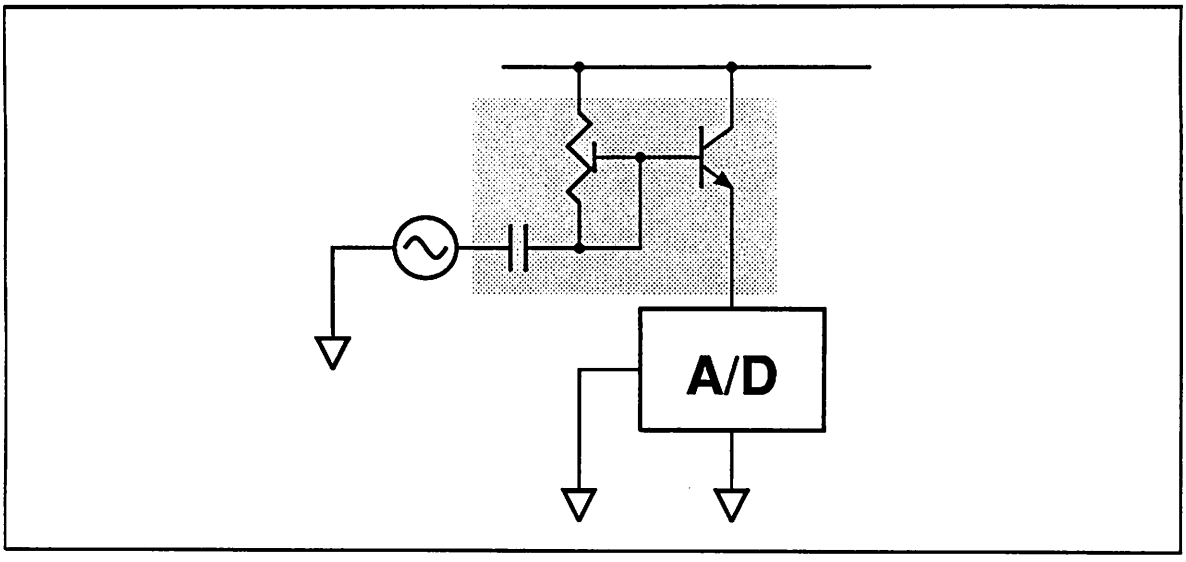

Figure 6.6.1 PSRR Test Set-Up.

is grounded. However, the PSRR result is usually incorrect, especially when an extremely large PSRR is expected. This is because when the power supply feedthrough signal is smaller than that of the smallest step of the A/D converter, the feedthrough signal component may not appear depending on what exactly the input-referred offset voltage is.

To overcome this problem, the PSRR test is done by applying two separate input signals that are different in frequency to the input path and to the power supply modulating circuit simultaneously, such as illustrated in Figure 6.6.2. The resultant digital output codes can then be processed digitally using the same FFT techniques employed earlier. The power supply rejection ratio can then be simply computed as the difference between the two main output signal components. A sample plot of such a test is shown in Figure 6.6.3. Typically, the PSRRs for most of the chips that have been tested fall in the range of between 50 to 60 dB at low (audio-range) frequencies. Notice that when an extremely high power supply rejection is expected, it might be necessary to scale the amplitude of the signal that is applied to the main input path to avoid its distortion components to be accidentally mistaken as the power supply feed-through components.

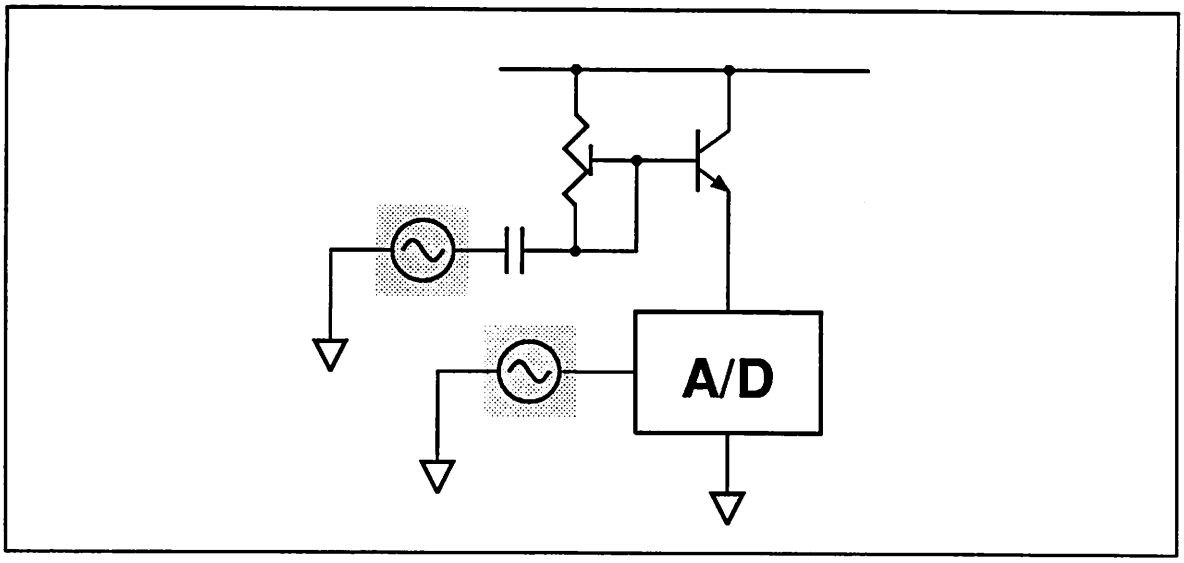

Figure 6.6.2 Simpler Technique to do PSRR Measurements.

6.7. INPUT REFERRED OFFSET VOLTAGE

The input referred offset voltage can be measured readily by applying an equivalent offset voltage to the input of the A/D converter such that the time average of the digital output codes of the A/D converter equals to zero. The fastest way to obtain the equivalent offset voltage is to reconstruct and filter the digital output codes of the A/D converter and feed the filtered signal back to the input of the A/D converter. A simplified block diagram of such a test setup is shown in Figure 6.7.1.

Another way to obtain the equivalent input offset voltage is to apply a DC voltage to the input of the A/D converter and to adjust this voltage such that the code-density output of the A/D converter (the histogram output) has symmetrical distribution around the zero code. However, this approach is rather time consuming. We chose to use this approach nevertheless because it requires a minimum number of additional components.

The typical input-referred offset voltages obtained from the prototype A/D converters are skewed unexpectedly toward the negative side with a mean of -1 LSB and a standard deviation of around 0.3 LSB. Out of more than 50 working chips that have been tested, none of them has

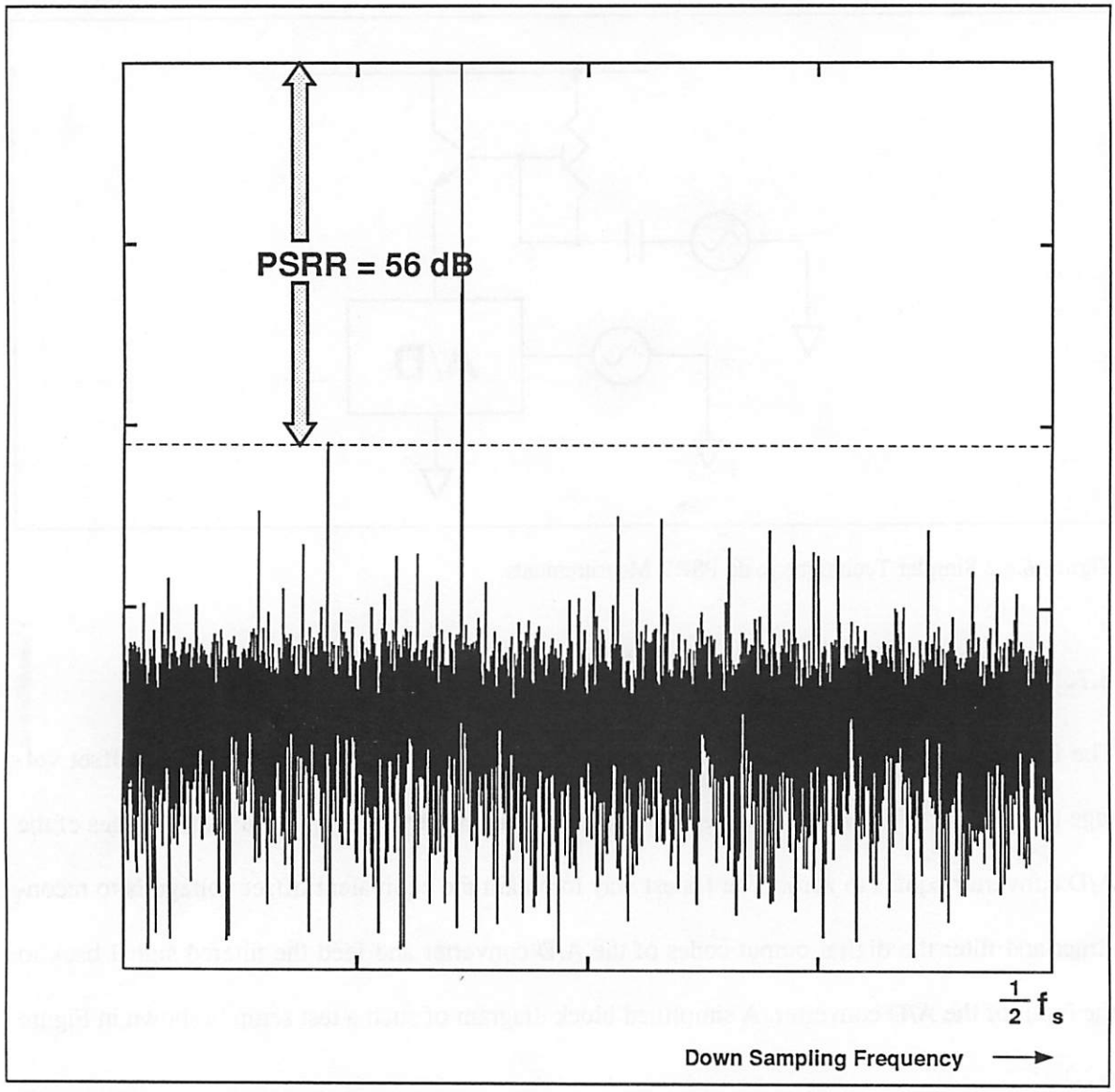

Figure 6.6.3 FFT Plot of the PSRR Test.

positive input-referred offset voltage. This is rather surprising because the layout is very symmetrical in many respects. On the other hand, the low random distribution of the offset voltage shows that the single shorting switch in the 1^{st} stage S/H circuit and the auto-zero offset cancellation circuits work as expected.

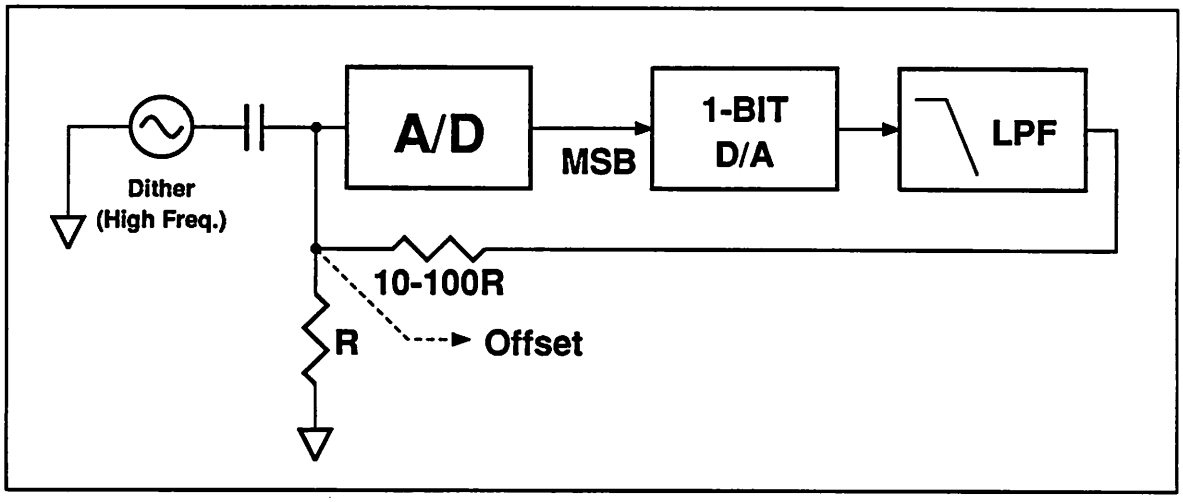

Figure 6.7.1 Input-Referred Offset Voltage Test Set-Up.

6.8. NOISE MEASUREMENT

A similar method to that of the offset voltage measurements was used to measure the equivalent input-referred noise voltage (excluding that of the inherent quantization noise) of the prototype A/D converter. By slowly varying the DC voltage that is applied to the A/D converter, the code-density distribution of the A/D converter can be tabulated. The standard deviation of a typical noise distribution, that is the equivalent RMS input-referred noise voltage, is found out to be about 150 $\mu Volts$.

As can be seen from the estimated noise data given in the previous chapter, the actual noise performance of the prototype converter is slightly worse than expected. This is probably caused by noise injected from on-chip digital circuitry such as the binary-to-linear decoding circuit and latches for the comparators which unfortunately were laid out to use the analog on-chip power supply.

6.9. SUMMARY OF TEST RESULTS

Some of the above tests were performed at different frequencies and the results are summarized in table 6.9.1.

Sampling rate		250 kHz
Differential nonlinearity		+/- 0.5 LSB
Integral nonlinearity		+/- 1.5 LSB
Peak S(N+D)R	10 kHz	72 dB, +/- 1dB
	100 kHz	70 dB, +/- 1dB
Dynamic range		80 dB
PSRR	DC	50-60 dB
	10 kHz	50-60 dB
Input referred offset		-1 LSB
Equivalent input noise		150 uV RMS
Power dissipation		15 mW
Area		3600 sq. mils

TEST CONDITIONS

- 25 °C
- 5V Single Power Supply
- Vref = +/- 2V

Table 6.9.1 Summary of Test Results.

CHAPTER 7

CONCLUSIONS

The pipelined A/D conversion architecture has been shown to be of considerable interest because of the promise of small-area utilization and low-power dissipation for implementing medium- to high-speed and high-resolution A/D converter functions. The two most salient features of the pipelined architecture are: First, because of the inherent concurrency of the operations among the individual pipelined stages, the conversion speed of a pipelined A/D converter is only limited by the time needed to process the analog information in one stage. This time is usually determined only by the opamp settling time. Second, because the number of stages of the pipelined A/D converter increases almost linearly with the resolution, and because the area of the individual stages is small, the pipelined A/D conversion architecture is suitable for implementing high-resolution A/D converters. The important results obtained from this work are summarized next.

As shown in chapter 4, the optimum pipelined architecture depends on the target applications and the available technology. For applications that require absolute maximum conversion speed, the optimum pipelined A/D converter implementation should employ stages that resolve only 1 bit in each stage or 2 bits if digital-error correction is used because the interstage amplifier settles fastest when the interstage gain is minimum. For applications that only require somewhat less conversion speed, the optimum pipelined A/D converter implementation should employ stages that resolve around 3 bits in each stage because the overall area is typically at a minimum under such a condition. In both cases, the sub-ADCs are assumed to be implemented using the flash architecture. For applications that only require medium conversion speed, the optimum pipelined A/D converter implementation should instead employ successive-approximation stages

that resolve around 5 bits in each stage because even smaller overall area can be obtained this way. The experimental prototype pipelined A/D converter is of the third category.

To desensitize the conversion to offset errors, digital-error correction is used in the prototype. Digital-error correction allows non-auto-zeroed comparators to be used without sacrificing the resolution and linearity of the overall A/D converter. Digital-error correction also improves the overall conversion speed because the comparators can be optimized for speed rather than for accuracy and because A/D comparisons can be made without having to wait for the rest of the analog circuitry to settle to a high accuracy. Digital-error correction also removes the effects of uncanceled opamp offset voltage on the linearity of the overall A/D converter. In summary, because of the use of digital-error correction, the resolution and linearity of the overall A/D converter are limited only by the linearity of the sub-DACs and the accuracy of the interstage amplifiers.

To reduce the effects of these remaining error sources, reference-feedforward correction can be used. This correction technique can significantly reduce the effects of the sub-DAC nonlinearity and interstage gain error on the overall resolution with only a slight increase in the overall area and with at most a factor of two reduction on the conversion speed. The linearity of a pipelined A/D converter that uses the reference-feedforward correction, however, is still limited by the linearity of the sub-DACs, especially that of the 1^{st} stage.

To improve the achievable linearity, an improved layout technique is therefore needed. A twisted capacitor layout structure has been used successfully in the experimental prototype pipelined A/D converter to achieve this goal. Even with a total input sampling capacitor value of less than 5pF and with small dummy capacitors, an overall linearity of close to 12 bits has been achieved repeatedly from three separate fabrication runs. An even better linearity would have been attainable if the residual systematic matching errors on the unit capacitors at the boundary of the input capacitor array could have been reduced.

One important observation that should be noted here is the extremely good matching seen between half of the capacitor array and the other half. As can be seen from the integral nonlinearity plot in the previous chapter, the observed matching accuracy would allow the realization of extremely fast 12-bit pipelined A/D converters using single-bit stages that employ the improved interstage amplifier configuration shown in chapter 3. More extensive experiments on different processing lines are needed, however, to confirm that it is indeed possible to attain such high linearity based on component matching alone when the capacitors are laid out using the twisted arrangement and when the capacitors are grouped into two halves.

REFERENCES

1. R.J. Apfel, "Digital Signal Processing Modems," *Design of MOS VLSI Circuits for Telecommunications*, pp. 458-511.

2. M. Yoshimoto, S. Nakagawa, K. Murakami, S. Asai, Y. Akasaka, Y. Nakajima, and Y. Horiba, "A Digital Processor for Decoding of Composite TV Signals using Adaptive Filtering," *IEEE Int'l Solid-State Circuits Conf. Digest of Tech. Papers*, pp. 152-153, 1986.

3. S. Matsumoto, I. Nakagawa, K. Kondo, and N. Kojima, "A Video Signal Processing 20ns 2K x 8 Multi-Funtional Memory," *IEEE Int'l Solid-State Circuits Conf. Digest of Tech. Papers*, pp. 186-187, 1987.

4. R. Schewer, P. Baker, T. Fisher, and H. Pfeifer, "Digital Television," *Design of MOS VLSI Circuits for Telecommunications*, pp. 567-612.

5. O.E. Agazzi, Large Scale Integration of Hybrid-method Digital Subscriber Loops, University of California at Berkeley, May, 1982. Ph. D. Thesis

6. H.-S. Lee, D. A. Hodges, and P. R. Gray, "A Self-Calibrating 15-bit A/D Converter," *IEEE Journal of Solid-State Circuits*, pp. 813-819, December, 1984.

7. B. Fotouhi and D.A. Hodges, "High-Resolution A/D Conversion in MOS/LSI," *IEEE Journal of Solid-State Circuits*, pp. 920-926, December, 1979.

8. J.A. Schoeff, "An Inherently Monotonic 12b DAC," *IEEE Int'l Solid-State Circuits Conf. Digest of Tech. Papers*, pp. 178-179, 1979.

9. J.A. Schoeff, "An Inherently Monotonic 12b DAC," *IEEE Journal of Solid-State Circuits*, pp. 904-911, Dec, 1979.

10. A.B. Grebene, *Analog Integrated Circuit Design,* Van Nostrand Reinhold, 1972.

11. J.L. McCreary and P.R. Gray, "All-MOS Charge Redistribution Analog-to-Digital Conversion Techniques-Part I," *IEEE Journal of Solid-State Circuits*, pp. 371-379, December, 1975.

12. R.H. McCharles, V.A. Saletore, W.C. Black Jr., and D.A. Hodges, "An Algorithmic Analog-to-Digital Converter," *IEEE Int'l Solid-State Circuits Conf. Digest of Tech. Papers*, pp. 96-97, 1977.

13. P.W. Li, M.J. Chin, P.R. Gray, and R. Castello, "A Ratio-Independent Algorithmic Analog-to-Digital Conversion Technique," *IEEE Journal of Solid-State Circuits*, pp. 828-836, December, 1984.

14. C.-C. Shih, Precision Analog to Digital and Digital to Analog Conversion Using Reference Recirculating Algorithmic Architectures, pp. 77-85, University of California at Berkeley, July 25, 1985. Ph. D. Thesis

15. A. G. F. Dingwall and V. Zazzu, "An 8MHz CMOS Subranging 8bit A/D Converter," *IEEE Journal of Solid-State Circuits*, pp. 1138-1143, December, 1985.

16. S. H. Lewis and P. R. Gray, "A Pipelined 5MHz 9b ADC," *Digest of Technical Papers, 1987 IEEE International Solid-State Circuits Conferences, New York, N. Y.*, pp. 210-211, February, 1987.

17. J.C. Candy, "Using Triangularly Weighted Interpolation to Get 13-bit PCM from a Sigma-Delta Modulator," *IEEE Transaction on Communications*, pp. 1259-1275, Nov, 1976.

18. M. Hauser and R.W. Brodersen, "MOS ADC-Filter Combination That Does Not Require Precision Analog Components," *IEEE Int'l Solid-State Circuits Conf. Digest of Tech. Papers*, pp. 80-81, 1985.

19. J. C. Candy, "A Use of Double Integration in Sigma Delta Modulation," *IEEE Transac-*

tions on Communications, pp. 249-258, March, 1985.

20. A.R. Armade, "A Single Chip All-MOS 8-bit A/D Converter," *IEEE Journal of Solid-State Circuits*, pp. 785-791, Dec, 1978.

21. J. L. McCreary, "Matching Properties and Voltage and Temperature Dependence of MOS Capacitors," *IEEE Journal of Solid-State Circuits*, pp. 608-616, December, 1981.

22. J.-B. Shyu and G. Temes, "Random Error Effects in Matched MOS Capacitors and Current Sources," *IEEE Journal of Solid-State Circuits*, pp. 948-955, December, 1984.

23. O. A. Horna, "A 150 Mbps A/D and D/A Conversion System," *Comstat Technical Review*, pp. 52-57, 1972.

24. S. Taylor, High Speed Analog-to-Digital Conversion, pp. 30-42, University of California at Berkeley, 1978. Ph. D. Thesis

25. W. J. Pratt, "High Linearity and Video Speed Come Together in A-D Converters," *Electronics*, October 9, 1980.

26. R. Poujois, B. Baylac, D. Barbier, and J.M. Ittel, "Low-Level MOS Transistor Amplifier Using Storage Techniques," *IEEE Int'l Solid-State Circuits Conf. Digest of Tech. Papers*, pp. 152-153, 1973.

27. B.-S. Song and M.F. Tompsett, "A 12b 1MHz Capacitor Error Avaraging Pipelined A/D Converter," *IEEE Int'l Solid-State Circuits Conf. Digest of Tech. Papers*, pp. 226-227, 1988.

28. K.-C. Hsieh, Noise Limitations in Switched-Capacitor Filters, pp. 9-10, University of California at Berkeley, May 18, 1982. Ph. D. Thesis

29. Nan-Sheng Lin, Partial Response Coding in Digital Subscriber Loops, University of California at Berkeley, September, 1986. Ph. D. Thesis

30. R.T. Kaneshiro, Circuit and Technology Considerations for High Frequency Switched-Capacitor Filters, University of California at Berkeley, July 18, 1983. Ph. D. Thesis

31. L.C. Yiu, Ratio-Independent Pipelined Analog-to-Digital Convertion, University of California at Berkeley, November 15, 1984. Doctoral Qualifying Examination Proposal

32. R. Castello and P. R. Gray, "A High-Performance Micropower Switched-Capacitor Filter," *IEEE Journal of Solid-State Circuits*, pp. 1122-1132, December, 1985.

33. P. R. Gray and R. G. Meyer, *Analysis and Design of Analog Integrated Circuits, Second Edition*, John Wiley & Sons, New York, 1984.

34. Y.S. Yee, L.M. Terman, and L.G. Heller, "A 1 mV MOS Comparator," *IEEE Journal of Solid-State Circuits*, pp. 294-298, June, 1978.

35. W.T. Lynch and H.J. Boll, "Optimization of the Latching Pulse for Dynamic Flip-FLop Sensors," *IEEE Journal of Solid-State Circuits*, pp. 49-55, April, 1974.

36. J. Doernberg, H. Lee, and D. A. Hodges, "Full-Speed Testing of A/D Converters," *IEEE Journal of Solid-State Circuits*, pp. 820-827, December, 1984.

37. J. Doernberg, H. Lee, and D. A. Hodges, Full-Speed Testing of A/D Converters, University of California at Berkeley, May 2, 1984. Memo Number UCB/ERL M84/38