A Time-Based Energy-Efficient Analog-to-Digital Converter

by

Heemin Yi Yang

S.B., Massachusetts Institute of Technology (1998)
M.Eng., Massachusetts Institute of Technology (2000)

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2006

Author .....
Department of Electrical Engineering and Computer Science
January 27, 2006

Certified by .....
Rahul Sarpeshkar
Associate Professor of Electrical Engineering
Thesis Supervisor

Accepted by .....
Arthur C. Smith
Chairman, Department Committee on Graduate Students

A Time-Based Energy-Efficient Analog-to-Digital Converter

by

Heemin Yi Yang

Submitted to the Department of Electrical Engineering and Computer Science
on January 27, 2006, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

Dual-slope converters use time to perform analog-to-digital conversion but require 2^{N+1} clock cycles to achieve N bits of precision. We describe a novel algorithm that also uses time to perform analog-to-digital conversion but requires $5N$ clock cycles to achieve N bits of precision via a successive sub-ranging technique. The algorithm requires one asynchronous comparator, two capacitors, one current source, and a state machine. Amplification of two is achieved without the use of an explicit amplifier by simply doing things twice in time. The use of alternating *Voltage-to-Time* and *Time-to-Voltage* conversions provides natural error cancellation of comparator offset and delay, $1/f$ noise, and switching charge-injection. The use of few components and an efficient mechanism for amplification and error cancellation allow for energy-efficient operation: In a 0.35 μm implementation, we were able to achieve 12 bits of DNL limited precision or 11 bits of thermal noise-limited precision at a sampling frequency of $31.25k$Hz with 75μW of total analog and digital power consumption. These numbers yield a thermal noise-limited energy-efficiency of $1.17p$J per quantization level making it one of the most energy-efficient converters to date in the 10 to 12 bit precision range. This converter could be useful in low-power hearing aids after analog gain control has been performed on a microphone front-end. An 8 bit audio version of our converter in a 0.18μm process consumes $960n$W and yields an energy-efficiency of $0.12p$J per quantization level, perhaps the lowest ever reported. This converter may be useful in biomedical and sensor-network applications where energy-efficiency is paramount. Our algorithm has inherent advantages in time-to-digital conversion. It can be generalized to easily digitize power-law functions of its input, and it can be used in an interleaved architecture if higher speed is desired.

Thesis Supervisor: Rahul Sarpeshkar
Title: Associate Professor of Electrical Engineering

Acknowledgments

First and foremost, I would like to thank my advisor, Rahul Sarpeshkar. He provided a truly free-thinking environment where I was given the freedom to pursue all of my research interests. At the same time, none of this work would have been possible without his guidance.

Second, I would like to thank my committee members, Anantha Chandrakasan and James Roberge, who provided valuable insights into my research. I would also like to express my gratitude to the members of my research group, Analog VLSI and Biological Systems, for providing such a supportive and intellectually stimulating environment.

In addition, I would like to thank my friends and members of Unlockedgroove for their support. Pud, 6 Million Dollar Dan, Jamie, Local Fields, Keepalive, Smartypants, Isaac, Gys, and Eddie, thanks for keeping me sane through this.

And of course, none of this would have been possible without the incredible support from my family members Hannah, Heela, Michael, and Michael Jr.

Finally, I dedicate this thesis to my parents, Yang Young Chi and Yi Sook Chung, who inspire me to become a better person everyday.

Contents

1 Introduction **17**

1.1 Dual-Slope A/D Converters 19

1.2 A Novel Time-Based A/D Converter 20

2 Energy-Efficient Analog-to-Digital Conversion Techniques **21**

2.1 Algorithmic and Pipelined Converters 22

2.1.1 Basic Architecture 22

2.1.2 Modern Techniques 25

2.2 Successive-Approximation Converters 26

2.2.1 Basic Architecture 26

2.2.2 Modern Techniques 27

2.3 Over-sampling Converters 29

2.3.1 Quantization Error 30

2.3.2 Basic Architecture 32

2.3.3 Modern Techniques 33

3 A Novel Algorithm for Time-Based Analog-to-Digital Conversion **35**

3.1 The Algorithm 36

3.1.1 The MSB Stage (Dual-Slope Stage) 38

3.1.2 Quantizing the Residue (Successive Subranging Stage) 38

3.1.3 Positive and Negative Index Counting 41

3.2 Error Cancellation Properties 41

3.2.1 Basics of Error Cancellation 42

3.2.2 Comparator Delay and Offset 42
3.2.3 Switching charge-injection 44
3.2.4 Reducing Capacitor (Gain) Mismatch 47
3.2.5 State-Machine Delay 48
3.3 Clock-Straddling Residues 49
3.4 $1 + \varepsilon$ Algorithm 49
3.5 Determining the Over-Ranging Factor 52

4 Implementation and Analysis of a Thermal-Noise-Limited Low-Power 12 bit Converter 55

4.1 Design and Implementation 55
4.1.1 Asynchronous Comparator 56
4.1.2 Pulse-Width Control Circuit 59
4.1.3 Differential Current Switching 62
4.1.4 State-Machine 63
4.2 Noise Analysis 65
4.2.1 Temporal Jitter vs. Voltage Noise 65
4.2.2 Quantization Error 66
4.2.3 Thermal Noise Sources 66
4.3 Experimental Results 75
4.3.1 Static and Dynamic Performance Results 75
4.3.2 Validating Our Noise Models 77

5 A Sub-Microwatt Audio Converter 85

5.1 Design and Implementation 86
5.1.1 Asynchronous Comparator 86
5.1.2 State-Machine 88
5.2 Noise Analysis 88
5.2.1 Quantization Error 89
5.2.2 Comparator Noise 89
5.2.3 Integration Noise 91

5.2.4 Reset Noise 91
5.2.5 Total Noise 92
5.3 Experimental Results 92
5.4 Comparing Energy-Efficient A/D Converters 100
5.5 Scaling to 90nm Technology 103

6 Conclusions **105**
6.1 Time-to-Digital Conversion 106
6.2 Computational A/D's for Polynomial Functions 107
6.3 Possible Immediate Applications of this Work 108

A Dual-Slope vs. Our Converter: A Comparison in Efficiency **111**

B An Analysis of Voltage vs. Time Amplification **113**

C Capacitor Mismatch Reduction Technique **117**

D Detailed Noise Analysis **119**
D.1 Diminishing Contribution of Noise from the Comparator 119
D.2 Integration Noise Derivation 120

List of Figures

1-1 *FOM* versus year of publication 18
1-2 Block diagram of a dual-slope ADC 19
1-3 Timing diagram of a dual-slope ADC 20

2-1 Block diagram of an algorithmic ADC 23
2-2 Block diagram of a pipelined ADC 24
2-3 Block diagram of a successive-approximation ADC 26
2-4 A successive-approximation ADC based on charge redistribution . . . 27
2-5 Transfer function of an ideal ADC 29
2-6 Probability density function of a quantization error model 31
2-7 Spectral density function of a quantization error model 31
2-8 Block diagram of a first order $\Sigma\Delta$ modulator 32
2-9 Basic block diagram of a $\Sigma\Delta$ converter 33

3-1 Graphical illustration of our algorithm 37
3-2 Visual representation of signal variables in the algorithm 39
3-3 Error cancellation principle 42
3-4 Illustration of comparator delay cancellation 44
3-5 Illustration of charge-injection cancellation 46
3-6 A clock-straddling residue scenario 49
3-7 Conversion time vs. N_{Total} 51
3-8 Conversion time vs. N_{Total} with minimum residue 51
3-9 Conversion time vs. N_{MSB} (Ideal) 52
3-10 Conversion time vs. N_{MSB}(Non-zero comparator delay and pulse-width) 53

3-11 Conversion time vs. $N_{MSB}(1+\varepsilon)$ 54

4-1 Block diagram of our ADC 56
4-2 Pre-amplifier of the comparator 58
4-3 Gain/latch stage of the comparator 59
4-4 Pulse-width control mechanism 60
4-5 Pulse-width internal node voltages 61
4-6 Current switching topology 62
4-7 D flip-flop 63
4-8 State-machine flow chart 64
4-9 Matched-logic module 65
4-10 Circuit diagram of reset phase 72
4-11 Theoretical model's prediction of SNR vs. relative current level . . . 74
4-12 Layout of the high performance ADC 76
4-13 INL and DNL plots for high performance ADC 79
4-14 PSD of high performance ADC 80
4-15 ENOB versus input frequency of high performance ADC 81
4-16 PSD of the off-chip V-to-I converter 82
4-17 3-D plot comparing SNR noise model with measurements 83
4-18 High performance ADC's SNR vs. the pre-amplifier's bias current . . 84
4-19 High performance ADC's SNR vs. the gain/latch stage's bias current 84

5-1 Comparator for low-power ADC 87
5-2 A low-power low-voltage D flip-flop 88
5-3 Theoretical model's prediction of SNR vs. relative current level . . . 93
5-4 Layout of the low-power ADC 95
5-5 INL and DNL plots for low-power ADC 97
5-6 PSD of low-power ADC 98
5-7 ENOB versus input frequency of low-power ADC 99
5-8 Survey of A/D converters in ISSCC and JSSC from 2001-2005 102

6-1 Basic principle of a TDC using our algorithm 106
6-2 Timing diagram of polynomial coefficient digitization process 108
B-1 OTA in a G_m-C filter topology 114
D-1 Abstract representation of ADC as a cascade of amplifiers 120

List of Tables

3.1 Capacitor matching requirements 47

4.1 Summary of noise contributions for high performance ADC (Theory) 73

4.2 Summary of experimental results for high performance ADC 78

5.1 Summary of noise contributions for low-power ADC (Theory) 92

5.2 Summary of experimental results for low-power ADC 96

5.3 Comparison of energy-efficient A/D converters 101

Chapter 1

Introduction

The ever-growing demand for ultra-low-power systems has pushed the world of analog integrated circuit design to its limits. The world of analog-to-digital converters (ADC) is no exception. Over-sampling converters are constantly benefiting from the smaller feature sizes of new fabrication processes, and as a result, they are becoming more energy-efficient. Unfortunately, the same cannot be said for Nyquist-rate converters. Energy efficiency has improved somewhat slowly over the past decade as they have benefited from better processes as well [1]. A survey of over seventy A/D converters published in the *IEEE Journal of Solid-State Circuits* reveals that ADCs have become more energy-efficient over the years. Figure 1-1 shows a scatter plot of the Figure-of-Merit (FOM) versus year of publication. The FOM, which is shown in (1.1), allows one to compare data converters across different sampling speeds, f_{samp}, bit precision, N, and power consumption, P.

$$FOM = \frac{2^N f_{samp}}{P} \tag{1.1}$$

A higher count indicates a more energy-efficient converter. The solid red line follows the average FOM per year and indicates that converters are becoming more energy-efficient every year. Some of the improvements in efficiency can be attributed to better fabrication technologies. The market demand for energy-efficient converters has driven research into creating more efficient designs via circuit and topology

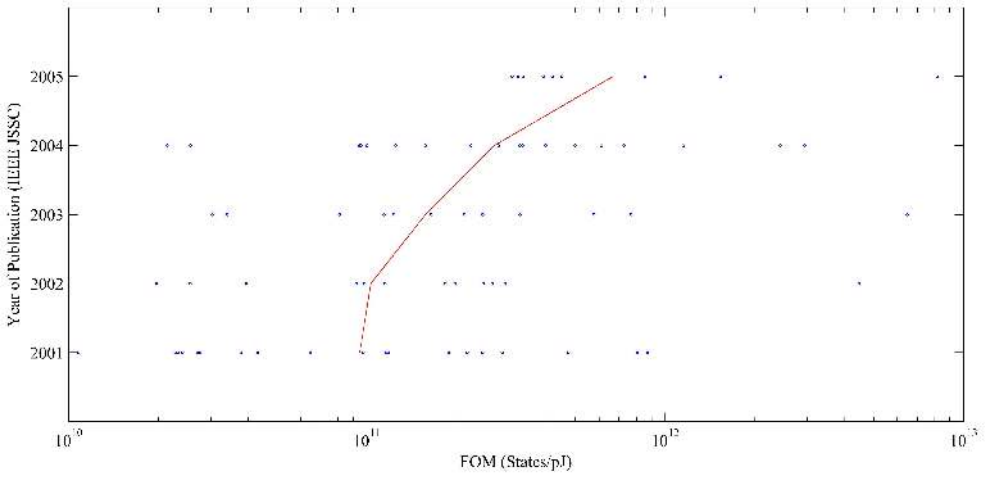

Figure 1-1: We plotted the FOM versus year of publication for over 70 A/D converters published in the *IEEE Journal of Solid-State Circuits*. The solid red line is an average of the FOM for that year and indicates that it is increasing virtually every year since 2001.

innovations as well.

When it comes to time-based A/D converters such as dual-slope converters, there have only been incremental improvements, and consequently, they have remained one of the least efficient designs. The novel time-based algorithm described in this thesis improves the efficiency of time-based A/D conversion from having a conversion time that scales exponentially in bit precision to scaling linearly. Our algorithm yields very energy-efficient designs. Before we describe this algorithm, we will first describe how dual-slope converters are built briefly in the introduction.

The organization of this thesis is as follows: In Chapter 2, we review prior work on energy-efficient A/D architectures. In Chapter 3, we describe our algorithm. In Chapter 4, we discuss circuits and data from a thermal-noise-limited 12 bit implementation of our algorithm that achieves an FOM of $\approx 10^{12}$. In Chapter 5, we describe a sub-microwatt 8 bit audio version that achieves an FOM near 10^{13} which may be the most energy-efficient converter ever reported to date. In Chapter 6, we conclude with possible applications and extensions of our work.

1.1 Dual-Slope A/D Converters

One of the first published documents on dual-slope analog-to-digital converters was available in a 1967 United States patent filing entitled, "Integrating Analog-to-Digital Converter" [2]. A block diagram of a basic dual-slope configuration is shown in Figure 1-2. The main principle behind the dual-slope ADC is simple: A current proportional to the input voltage is used to integrate a capacitor for a fix amount of time, after which a current proportional to a reference voltage subtracts charge from the same capacitor. The ratio of the fixed integration time of T_{in} to the variable integration time of T_{ref} is inversely proportional to the ratio of the input to the reference voltage, as shown in (1.2). The integration times are quantized by using a simple counter, as illustrated in Figure 1-3.

$$\frac{V_{IN}}{V_{REF}} = \frac{T_{ref}}{T_{in}} \tag{1.2}$$

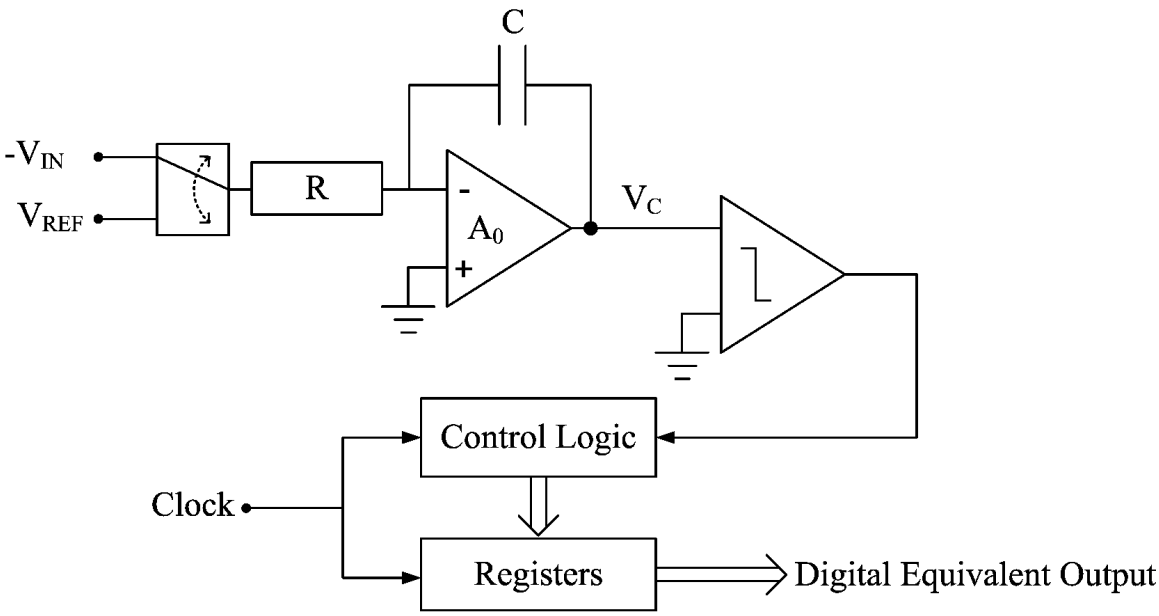

Figure 1-2: A block diagram of a basic dual-slope A/D converter is shown. More complex systems involve auto-zeroing the amplifier [3] and dual-ramp configurations [4].

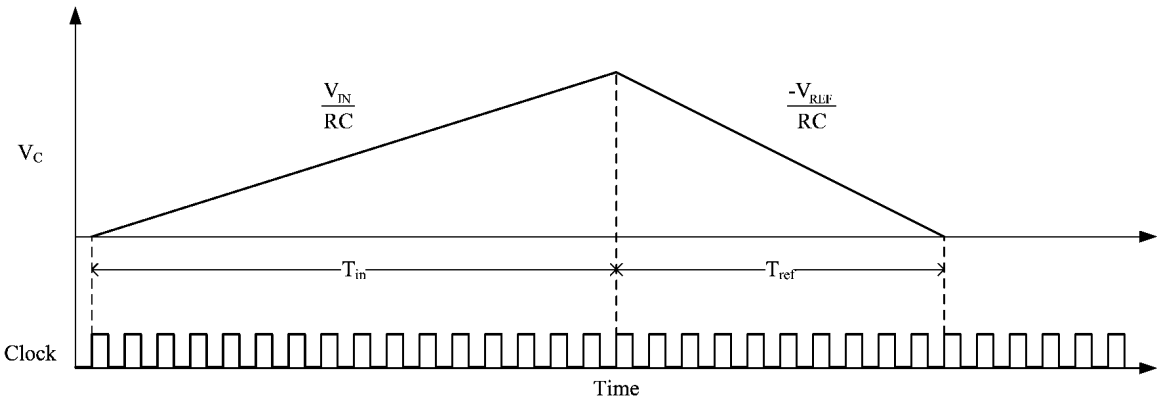

Figure 1-3: The timing variables in addition to the integration voltage, V_c, are shown.

1.2 A Novel Time-Based A/D Converter

Dual-slope converters are still commonly used in applications that require very precise digitization such as AC voltage and current measurements [5] [6] or extremely low power consumption such as bionic ear processors [3]. However, the dual-slope algorithm is inherently inefficient. As illustrated above, the conversion time scales exponentially with bit precision, and accordingly, algorithmic, pipelined, successive-approximation, and other recursive schemes are always more energy-efficient. Appendix A presents a detailed comparison between the new algorithm presented in this thesis and the dual-slope architecture and shows that our algorithm can be 10 to 19 times more energy-efficient.

In this thesis, we present a novel time-based algorithm that, like the dual-slope converter, uses time to perform analog-to-digital conversion. However, the algorithm utilizes time as an intermediate signal variable as opposed to voltage or current, and digitizes information in a recursive successive sub-ranging fashion such that the conversion time grows linearly with precision. In addition, the algorithm naturally alleviates errors such as comparator delay and charge-injection that often plague other converters. Most importantly, the converter performs amplification without an explicit amplifier, and consequently, it is extremely energy-efficient.

Chapter 2

Energy-Efficient Analog-to-Digital Conversion Techniques

In the past, many have focused on improving both the speed and precision of analog-to-digital converters. While the need for faster and more precise converters will always remain, the expanding market of wireless and portable electronic systems has created a strong demand for a new breed of energy-efficient converters. Many applications that demand such energy-efficient operation require moderate precision (8 to 12 bits) and conversion speed ($< 1M$Hz) such as in biomedical and sensor-network applications. Consequently, conversion techniques that are inherently more efficient are being utilized. Recent publications show that several architectures such as algorithmic, successive-approximation, and over-sampling converters can, if designed properly, achieve very good energy-efficiency. Digital calibration of relatively poor performance analog-to-digital converters is increasingly being used to achieve good energy-efficiency and performance [7]. This chapter will present a review of some of the important work in the area.

2.1 Algorithmic and Pipelined Converters

2.1.1 Basic Architecture

Both algorithmic and pipelined converters are based on the same principle: The input signal, V_{IN}, is compared to a reference voltage, V_{REF}, to generate a conversion bit, b_i. A residue from this comparison, e.g., $(V_{IN} - b_i V_{REF})$, is amplified. The amplified residue is then used as the input for the next comparison with the reference voltage, and the process is recursively repeated to get successive bits of the conversion. Algorithmic converters iteratively process the residue by using the same circuitry for each quantization loop. Pipelined converters, however, trade off area and power for faster conversion speeds by processing each residue in a serial fashion in a chain of blocks pipelined with S/H circuitry.

Algorithmic

Algorithmic A/D converters, also known as cyclic converters, are widely used in moderate speed, high accuracy systems. The basic principle behind the algorithm is illustrated in Figure 2-1. The system is composed of a sample-and-hold (S/H), a comparator, an amplifier, a subtractor, and a simple digital control system. The S/H front-end first samples the input, V_{IN}, and the following MUX directs the input to the comparator which goes high if the signal is greater than zero. The output of the MUX is also passed on to the subtractor which subtracts $V_{REF}/4$ if the comparator is high or adds $V_{REF}/4$ if the comparator is low. The output of the subtractor is amplified by two, and the result is then sent back to the MUX which routes the output of the amplifier back to the comparator for the remaining successive stages. A set of registers keeps track of the comparator's output for each iteration and stores the digital equivalent output. This algorithm iteratively uses the same comparator, subtractor, and amplifier and therefore requires little power and area. In most cases, the subtractor, amplifier, and S/H units are implemented in a single block [8]. The ultimate precision often relies on the accuracy of the amplifier and the subtractor [9]. Note that the algorithm can be generalized for multi-bit quantization per iteration.

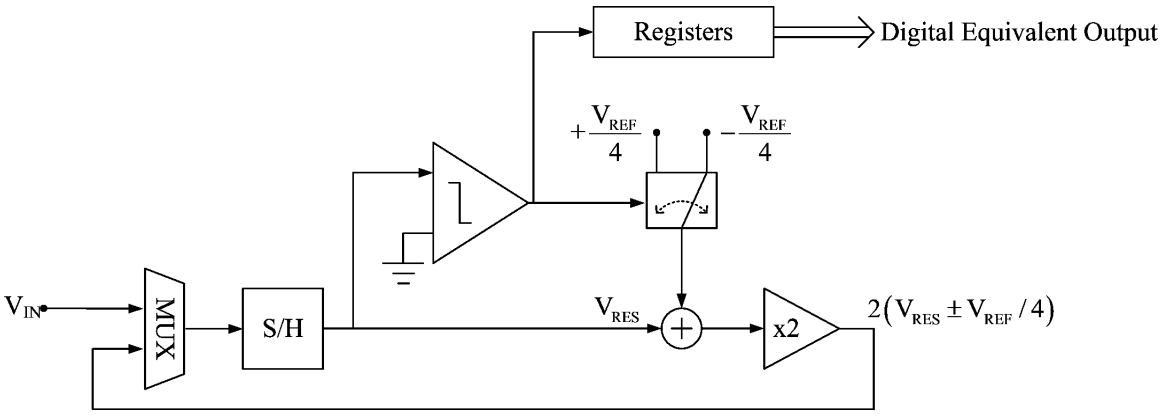

Figure 2-1: A simplified block diagram of an algorithmic A/D.

Pipelined

Pipelined analog-to-digital converters rely on multiple algorithmic stages with intermediate S/H blocks. For simplicity, a generalized block diagram of a 1 bit per stage pipelined converter is shown in Figure 2-2. Each stage consists of a S/H, a 1 bit A/D converter, a 1 bit D/A converter, and subtractor. Note that both the ADC and DAC can be generalized into a multi-bit system. The example given here is identical to the algorithmic converter shown in Figure 2-1 except that each iterative step is explicitly constructed as an independent block. After the input is sampled by the first S/H, the 1 bit A/D converter produces the Most Significant Bit (MSB). The 1 bit DAC recreates the analog representation of the MSB and then subtracts it from the input signal, producing a residue. The residue is amplified by a factor of 2 which is then sampled by the following S/H, and the successive stage repeats the A/D/A process. Once a stage has finished processing its input, the intermediate sample-and-hold allows the stage to immediately start processing the next sample. Consequently, the throughput is determined by the speed of an individual stage but the latency of the overall converter is set by the number of stages. Similar to the algorithmic converter, the overall precision of the converter relies on the accuracy of the DACs, subtractors, and gain stages. Each successive stage's weight to the quantization process diminishes down the pipeline, so the accuracy requirements on the components in the successive stages can be relaxed accordingly.

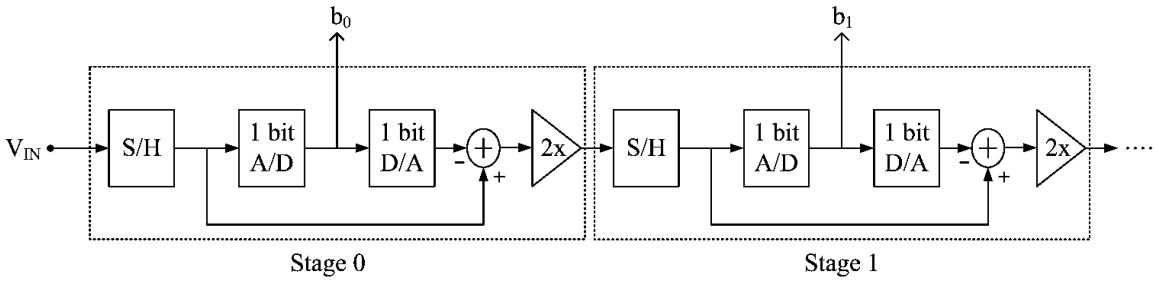

Figure 2-2: A simplified block diagram of a pipelined A/D.

Like the algorithmic converter, most pipeline architectures combine the S/H, subtractor, and amplifier into a classic switched-capacitor gain circuit whose topology has changed little over time [10] [11]. Although the generalized diagram in Figure 2-2 shows a 1 bit per stage single-ended topology, most pipelined converters implement a 1.5 bit per stage differential configuration [10] [11] [12]. The Redundant Signed Digit (RSD) algorithm is based on the Sweeny-Robertson-Tocher division principal [13] and implements a 1.5 bit/stage architecture by using two comparators in parallel to quantize the input into three levels between $\pm V_{REF}$. A common input/output relationship for the 1.5 bit stage is shown in (2.1).

$$V_{OUT} = \begin{cases} 2V_{IN} - V_{REF}, & \text{if} \quad V_{IN} > \frac{V_{REF}}{4} \\ 2V_{IN}, & \text{if} \quad -\frac{V_{REF}}{4} \leq V_{IN} \leq \frac{V_{REF}}{4} \\ 2V_{IN} + V_{REF}, & \text{if} \quad V_{IN} < -\frac{V_{REF}}{4} \end{cases} \tag{2.1}$$

The extra half-bit of information from each stage is used for digital error correction. Specifically, the use of RSD allows the successive stage to digitally compensate for comparator offset errors from the previous stage, as illustrated in [11]. And while the pipeline algorithm can be generalized for multi-bit stages, most converters maintain a 1 or 1.5 bit per stage architecture to allow high-speed throughput; the bandwidth of the switched-capacitor S/H & gain block often limits the speed of a stage, and therefore, a low gain of two minimizes the load capacitances on the amplifier, which in turn improves the bandwidth performance of the switch-capacitor circuit.

2.1.2 Modern Techniques

Since the precision of the DACs, amplifiers, and subtractors limit the accuracy of a pipelined converter, both analog and digital calibration techniques attempt to compensate for these sources of error. Analog calibration techniques often involve capacitor error averaging [14] and capacitor trimming [15] in the switched-capacitor amplifier. Digital calibration techniques, on the other hand, estimate and digitally correct DAC gain errors using DAC noise cancelation (DNC) [16]. The technique presented in [16] implements a continuous, background approach to calibration such that extra clock cycles and calibration states are not required. A recently published digital calibration technique for pipelined converters takes a similar approach [17]. However, [17] utilizes a slow but accurate algorithmic converter to digitally compensate for the amplifier and DAC gain errors and offsets. Furthermore, the algorithmic converter itself is calibrated, making the entire system's calibration completely nested. Noise requirements on the algorithmic converter can be lax because its outputs can be averaged over several cycles. As a result, they were able to achieve a signal-to-noise-and-distortion ratio (SNDR) of 70.8dB for a 20 Msamples/s sampling rate while consuming $254m$W of power. These figures translate to an FOM of approximately 2.2×10^{11}.

Algorithmic and pipelined converters require accurate and precise amplifiers, and as such expend most of their power budget on the amplifier [18]. However, a recent technique presented in [7] uses the same design as in [18] but replaces the precise, high-power amplifier with an imprecise, low-power, open-loop amplifier. A digital background calibration technique then compensates for the non-linear characteristics of the amplifier. Almost all algorithmic and pipelined converters use a two-stage, high gain operational transconductance amplifiers (OTA) with a feedback capacitor to achieve the desired precision and accuracy during the first stage of the pipelined converter. However, [7] uses an open-loop, low-power OTA with an imprecise gain term during the first stage of the pipelined conversion. The digital calibration scheme takes a statistics-based approach to continuously compensate for the most significant

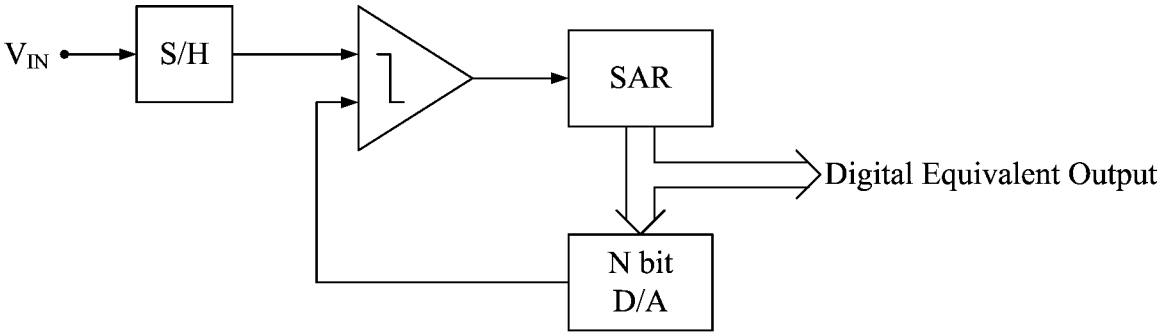

Figure 2-3: A simplified block diagram of a successive-approximation A/D. The successive-approximation register (SAR) updates and stores the binary output while providing the digital code for the D/A converter.

nonlinear characteristics of an open-loop OTA. For simplicity, only the 3rd order distortion term within the differential pair model from [19] needs to be corrected by the calibration block to achieve the required precision. Consequently, [7], compared to [18], achieved a power reduction in the amplifier of $33m$W (62%) while the digital calibration mechanism only added $\approx 10m$W. As a result, [7] achieves an FOM of approximately 5.4×10^{11}.

2.2 Successive-Approximation Converters

Algorithmic converters have proven to be highly accurate and consume very low power and area for moderate speed conversion rates. Pipelined converters, on the other hand, provide high precision at very fast conversion speeds but consume significantly more power and area. And while both architectures have been shown to be energy-efficient, successive-approximation converters are proving to be one of the most energy-efficient architectures available.

2.2.1 Basic Architecture

Successive-approximation converters operate on a binary search principle: the input, whose range is limited to V_{REF}, is initially compared to a reference voltage of $V_{REF}/2$. If greater, then the input is compared to $3V_{REF}/4$. Otherwise, the input is compared to $V_{REF}/4$. Each cycle produces an estimate that is one bit more precise. Thus, the

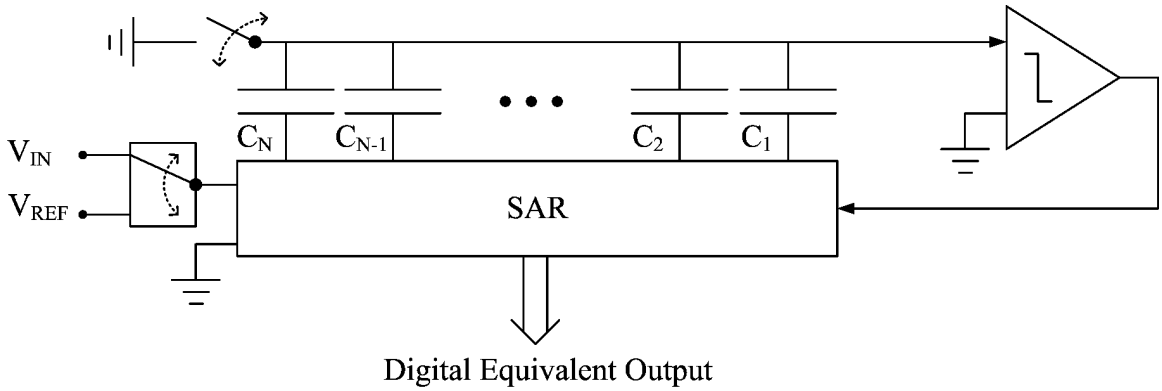

Figure 2-4: A D/A converter is realized using a capacitor array and the charge-redistribution principle. C_1 is the unit size capacitor, and $C_{i+1} = 2C_i$. The SAR switches each capacitor to V_{IN}, V_{REF}, or GND.

process repeats for N cycles to produce N bits of precision.

A generalized block diagram of a successive-approximation converter is shown in Figure 2-3. The converter is comprised of a S/H, comparator, successive-approximation register (SAR), and a D/A converter. The SAR, in conjunction with the comparator, continually updates the input to the D/A converter such that the DAC's output approaches the input voltage to within the desired Least-Significant-Bit (LSB). In some cases, the D/A converter is realized using a standard R-2R ladder architecture [20] [21] or current steering architecture [22]. But in most low-power implementations of the successive-approximation algorithm, the D/A converter is based on the charge-redistribution principle which is implemented as a simple switched-capacitor network [23]. In some cases, the S/H is integrated into the capacitor network such that the entire capacitor array acts as the sampling capacitor [24]. A generalized block diagram of a successive-approximation converter based on this principle is illustrated in Figure 2-4.

2.2.2 Modern Techniques

As stated earlier, applications that require low-speed, moderate precision, energy-efficient A/D converters are turning towards the successive-approximation architecture as a solution. Several factors are responsible for this trend and are itemized below:

- No amplifier. Only a single synchronous comparator is required.

- Scales well with technology. Capacitor matching is improving for smaller unit size capacitance.

- Low-voltage operation. Slight modifications to the algorithm allow for low power supply voltages.

- Extremely simple architecture. Very few components are required to implement the algorithm.

Some of the latest, most energy-efficient converters ever reported are based on the charge-redistribution successive-approximation architecture and take advantage of these properties [23] [24]. The converter presented in [24] operates on a 1V power supply which also serves as the reference voltage. The converter in [23] also uses the power supply voltages (V_{DD} and V_{SS}) as the reference voltages and operates on an even lower power supply of 0.5V. Driving the gates of the switch transistors is a potential problem at this voltage level. However, unlike [24], they elected to implement an explicit sample-and-hold such that the capacitors in the network are always switching with respect to a reference voltage as opposed to the input and a reference voltage. Therefore, the switches are always driven by a constant gate-to-source voltage. Both designs also reduce power consumption by using a small unit size capacitor, minimizing the amount of total charge required for the charge-redistribution network. Essentially, both designs illustrate that one can achieve very energy-efficient operation with the successive-approximation architecture because of the principles itemized above. Using these techniques, [24] achieved 47.8dB of SNDR at a sampling rate of $100k$Hz while consuming 3.1μW of total power resulting in an FOM of approximately 6.5×10^{12}. The converter in [23] achieved 43.3dB of SNDR for a $4.1k$Hz sampling rate while consuming 0.85μW of total power resulting in an FOM of approximately 4.5×10^{12}.

2.3 Over-sampling Converters

So far, all of the analog-to-digital conversion techniques discussed can be classified as Nyquist-rate converters where the input signal is band-limited to within one-half the sampling frequency. Over-sampling converters sample at a rate much greater than the Nyquist sampling rate and use averaging and noise shaping to increase precision. In the past 15 years, over-sampling converters, also known as $\Sigma\Delta$ converters, have gained popularity in the medium to high precision and moderate speed category. The architecture is composed of mostly digital filters with very simple analog components, and consequently, the rapid scaling of fabrication technology has made $\Sigma\Delta$ converters more and more energy-efficient.

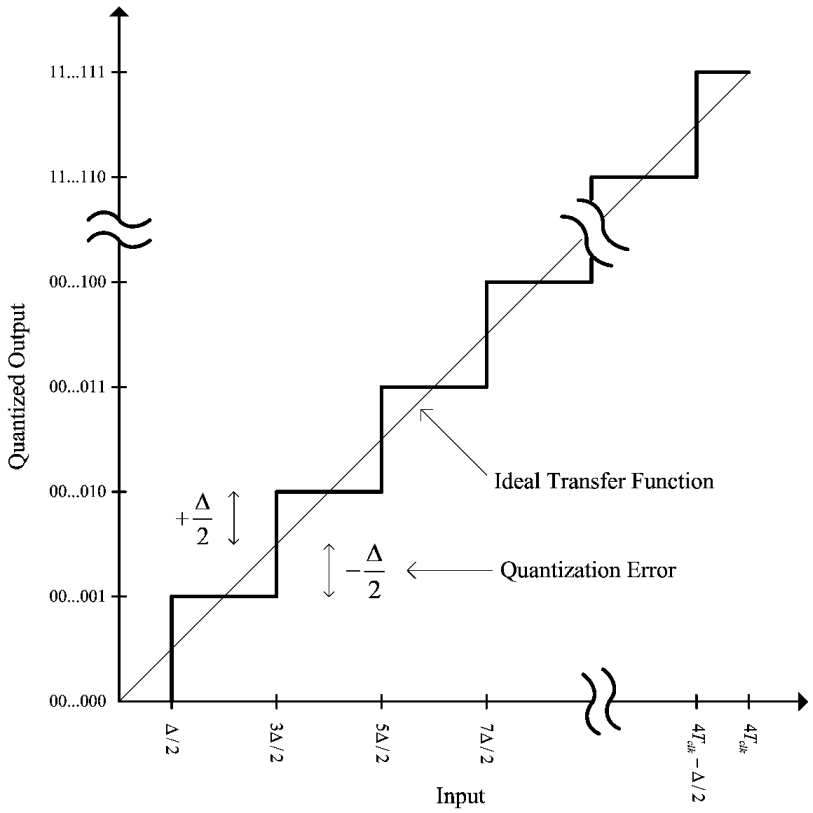

Figure 2-5: The transfer function of an ideal A/D converter shifted by $\Delta/2$ shows that the quantization error is bound by $\pm\Delta/2$.

2.3.1 Quantization Error

The basic principle of over-sampling converters is based on a white-noise assumption on the quantization error. Therefore, before we proceed, let us briefly review quantization error and the white-noise approximation model.

First, let us assume that our converter is free of thermal noise sources and is ideal such that the transfer function shown in Figure 2-5 applies to all input signals within our range. As illustrated in the figure, the error between the quantized output and its ideal value will be bound by $\pm\Delta/2$, where Δ is defined as the LSB. Specifically, for an N bit converter with a full-scale voltage of V_{FS},

$$\Delta = \frac{V_{FS}}{2^N}. \tag{2.2}$$

If we digitize a signal that is dynamic enough, the resulting quantization error can be approximated as a stochastic "noise" source. In other words, we can model the quantization error as a uniformly distributed random variable with a probability density function, $f_q(x)$, shown in Figure 2-6. Then the energy of the quantization error, $V_{NZ,q}^2$, is defined as the variance of $f_q(x)$, or specifically,

$$V_{NZ,q}^2 = \frac{\Delta^2}{12}. \tag{2.3}$$

In essence, we are making a white-noise approximation for the quantization error such that for a sampling rate of f_s, the spectral density of the quantization error, $S_q(f)$, would be a constant spread between $\pm f_s/2$, as illustrated in Figure 2-7. The height of the spectral density function is then determined by the fact that

$$\int_{-\frac{f_s}{2}}^{+\frac{f_s}{2}} S_q^2(f)df = \frac{\Delta^2}{12}. \tag{2.4}$$

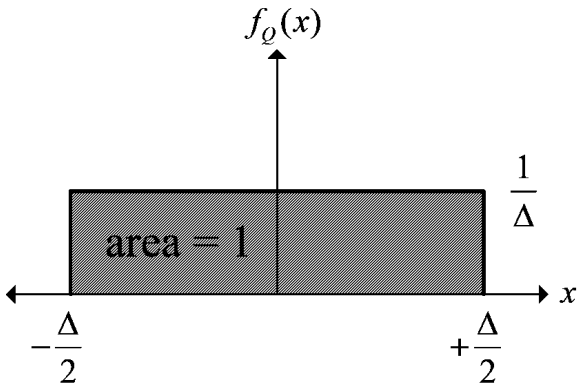

Figure 2-6: We can approximate the quantization error as a random variable with probability density function, $f_Q(x)$, that has a uniform distribution between $\pm\Delta/2$.

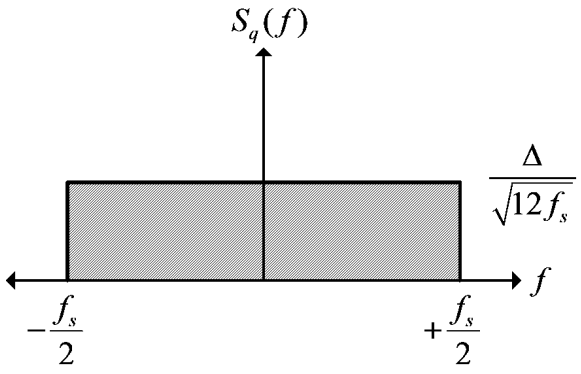

Figure 2-7: We can approximate the spectral density of the quantization error as a white-noise model.

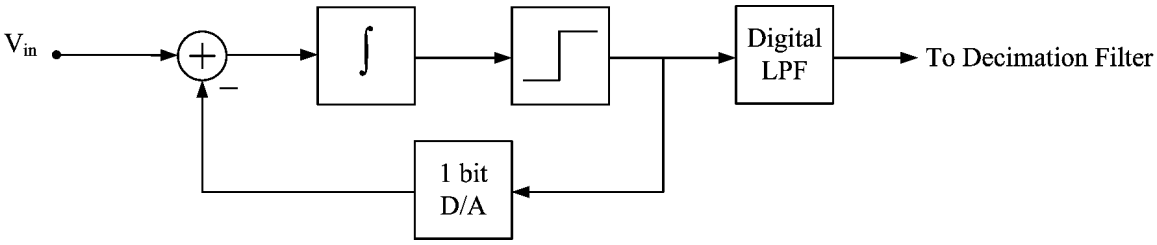

Figure 2-8: A block diagram of a first order ΣΔ modulator is shown.

2.3.2 Basic Architecture

One key idea behind ΣΔ converters is that by filtering the output of the quantizer to within a bandwidth that is much less than the sampling rate, we can remove much of the quantization noise. Specifically, the power of the quantization error is reduced to

$$\frac{\Delta^2}{12}\left(\frac{2f_o}{f_s}\right), \tag{2.5}$$

where f_o is defined as the bandwidth of the filter. Consequently, the overall precision, as defined by the signal-to-noise ratio (SNR), improves by

$$10log_{10}\left(\frac{f_s}{2f_o}\right). \tag{2.6}$$

In effect, we are averaging the input signal, and since the power of the signal adds linearly while the power of quantization error adds as the square root of the sum of squares, the SNR increases.[1] Furthermore, by using first, second, or higher order filters in a single or multi-loop feedback system, we can improve the precision even more by modulating much of the noise into a higher frequency band beyond the bandwidth of the system [25]. A single-order modulator is shown in Figure 2-8. This method, otherwise known as "noise shaping," is at the heart of over-sampling converters. A generalized block-diagram of a complete over-sampling converter is shown in Figure 2-9. The anti-alias filter at the front-end simply band-limits the input signal to within half the sampling frequency, and the decimation filter, which

[1]The derivation of SNR and its effective bit precision from all deterministic and non-deterministic sources of error will be discussed in Section 4.2.

is comprised of a down-sampler, band-limits the output of the modulator to f_o. The design of the decimation filters, in addition to the modulators, is an extremely broad area and is beyond the scope of this thesis. However, we will briefly discuss systems presented in recent literature that have achieved very energy-efficient operation.

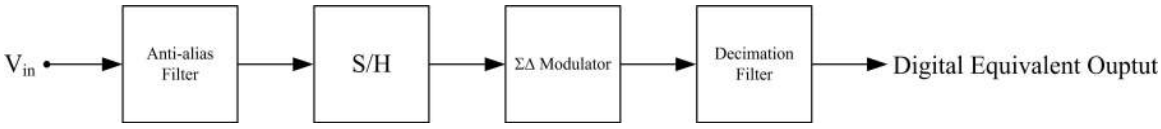

Figure 2-9: A generalized block diagram of a ΣΔ converter is shown. The anti-alias filter limits the input frequency to $f_s/2$. The decimation filter is comprised of a digital low-pass filter and a down-sampler. The digital low-pass filter limits the output of the modulator to $f_o/2$.

2.3.3 Modern Techniques

Since the linearity of the modulator is required to be within the overall LSB of the converter, many architectures utilize a 1 bit ADC and DAC; the advantages of a 1 bit quantizer in ΣΔ converters are well known [26]. However, multi-bit modulators are not uncommon, but the DAC in the feedback path often requires calibration and/or complex algorithms such as dynamic element matching (DEM) to achieve precise linearity [27] [28]. Several converters such as [29] and [30] achieve energy-efficient operation using 1 bit modulators. An extremely energy-efficient converter presented in [31] uses a 2 bit modulator but avoids the linearity requirements of the DAC by using the 2nd bit for only rare, over-ranging conditions. In addition, all three converters utilize 3^{rd} or 4^{th} order modulators to achieve excellent noise-shaping properties. And while [29] operates on a 3.3V supply, both [30] and [31] achieve lower power consumption by operating on a 1V and 1.25V supply, respectively. Another common theme among efficient ΣΔ designs is that the OTA used for the switched-capacitor integrators are all single-stage designs with simple modifications to improve gain/bandwidth performance without consuming additional power. But more importantly, all of these converters were able to achieve energy-efficient operation simply because of better and smaller fabrication technologies, especially [30] which

achieved 81dB of SNDR at a sampling rate of $20k$Hz while consuming 140μW of total power on a 90nm technology. These figures yield an FOM of approximately 2.9×10^{12}. The converter presented in [31] was able to achieve 84dB of SNR at a sampling rate of $11k$Hz while consuming 260μW of total power, allowing it to achieve an FOM of approximately 5.5×10^{11}.

Chapter 3

A Novel Algorithm for Time-Based Analog-to-Digital Conversion

Our converter is a current-mode analog-to-digital converter that functions like a successive sub-ranging or pipelined converter. However, unlike other known analog-to-digital converters, this converter uses time as an intermediate signal variable for the gain and subtraction routines. In addition, the converter naturally alleviates potential errors due to charge-injection, comparator delays and offsets through a unique algorithm that utilizes a single comparator and reference current. Amplification by two is done by simply doing things twice in time. Thus, we build an amplifying ADC without an explicit amplifier and saving power.[1] Our ADC consists of two matched capacitors, a reference current, a single comparator with finite pulse-width control, a simple state-machine, and N one-bit counters, where N is the total bit count of the converter. Only the two integrating capacitors need to match to within the LSB precision. This matching can be achieved or improved by calibration [32]. Since only a single comparator and one reference current source are used for the entire conversion process, the ADC consumes minimal power. A good fraction of the analog and digital power consumption scale with technology.

[1] Appendix B compares the energy-efficiency of time amplification versus voltage amplification.

3.1 The Algorithm

In order to simplify the presentation of the algorithm, we will ignore the effects of comparator delay and switching charge-injection. Section 3.2 presents in detail the mechanism that minimizes these sources of error. The algorithm consists of two distinct stages: During the first stage, we calculate the first two MSBs in a method identical to a dual-slope converter. During the second stage, we iteratively calculate the remaining bits by performing a subtraction-and-amplification process for each extra bit in a manner similar to that of a successive sub-ranging converter, but using time as the signal variable.

Figure 3-1 shows both stages of the conversion process. On the first clock cycle an input current charges a capacitor to create a voltage. Then a reference current charges another capacitor until it reaches the same voltage. By counting the number of clock edges within the latter charging period, we calculate the MSB corresponding to the I_{IN}/I_{REF} ratio as in a dual-slope converter. In our converter, this is at most 4 clock cycles since I_{IN} is required to be less than $4I_{REF}$ for maximum efficiency in the conversion process (See Section 3.5). At the end of the first stage of conversion, a residual time, denoted $0.5 + \varepsilon$ clock cycles in the figure, encodes the remaining bits of I_{IN}/I_{REF}. These bits are quantized in the second iterative stage of the conversion process described in the next paragraph. In this example, ε is positive such that the overall $0.5 + \varepsilon$ residue is greater than a half clock cycle.

We first convert the residual time, $0.5 + \varepsilon$, into a residual voltage. Then we charge another capacitor until it reaches this residual voltage and repeat the process again such that the residual time before the clock edge, $0.5 + \varepsilon$, is doubled to $2(0.5 + \varepsilon) = (1 + 2\varepsilon)$ after the clock edge. In essence, we amplify the residual time by two by doing things twice in time. The '1' in '$1 + 2\varepsilon$' provides quantization information revealing that the previous residue was indeed greater than a half clock cycle such that a clock edge is seen during the amplification, and the 2ε is automatically encoded as a residue referenced to this clock edge for the next stage of the conversion. Thus, subtraction of intermediate quantized bits is automatic in the algorithm because they manifest as

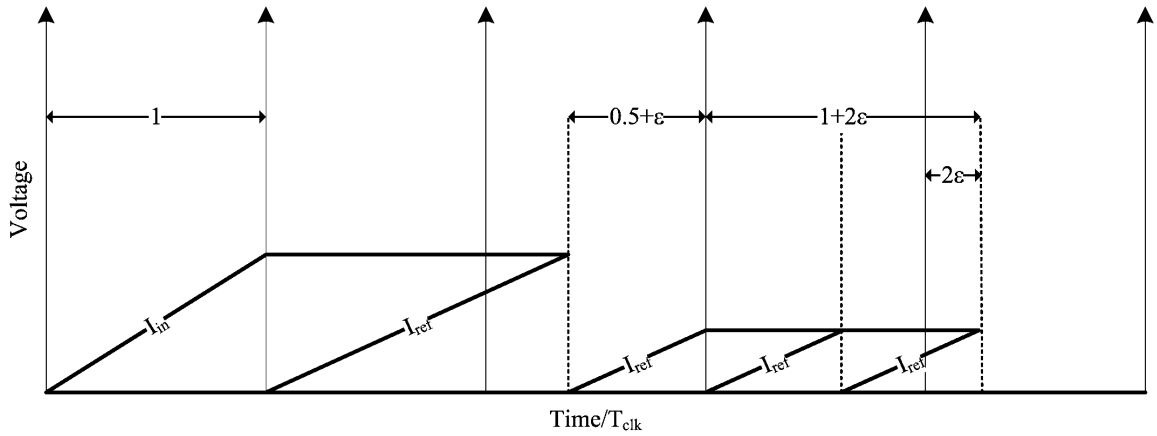

Figure 3-1: Time is normalized with respect to T_{clk}. The residual time, $0.5 + \varepsilon$, leading up to the next clock edge is at least as large as one-half the clock period. After amplification by two and automatic subtraction of the intermediate quantization bit '1', the original residue is transformed to a new residue, 2ε, which is used in the next stage of conversion.

an integer number of clock edges (or not), and amplified residues are always encoded with respect to the last seen edge.

The amplification, quantization, and subtraction of the residues are iteratively repeated to obtain successive bits: The 2ε residue is converted to a $1 - 2\varepsilon$ residue by converting the time from the end of amplification in Figure 3-1 to the next clock edge into a voltage. We can then recursively repeat the overall $\varepsilon_{n+1} \leftarrow (1 - 2\varepsilon_n)$ process to get successive conversion bits. The overall scheme ensures correct treatment of all residues whether quantization edges occur or do not occur during amplification. However, it causes alternating sign changes in the residues which are easily digitally corrected in the quantization bits (See Section 3.1.3).

We now expand the previous intuitive discussion into a more detailed description of the algorithm: Section 3.1.1 describes how we quantize the first two MSBs. Section 3.1.2 discusses the recursive processes performed during the sub-ranging process. Section 3.1.3 explains the converter's unique binary output code. For future reference, all clock edges refer to positive clock edges.

3.1.1 The MSB Stage (Dual-Slope Stage)

Figure 3-2 shows that during the first stage of the converter, the input current, I_{IN}, charges a capacitor, C_1, for one clock period, T_{clk}. Note that since the input current is integrated only during the first clock period, the MSB stage can behave like a sample-and-hold stage. During the next clock period, we charge another capacitor of equal value, C_2, with a reference current, I_{REF}, until the voltages on the two capacitors are equal. The latter comparison can be achieved by using a simple asynchronous comparator. Let us define T_{ref} as the time required to charge C_2 up to the voltage on C_1. In a manner similar to that of a dual-slope converter, we obtain the ratio of I_{IN} to I_{REF} if we quantize the ratio of T_{clk} to T_{ref} [33]. Specifically,

$$\frac{I_{IN}}{I_{REF}} = \frac{T_{ref}}{T_{clk}}. \tag{3.1}$$

If we use a two-bit counter to count the number of clock edges during T_{ref}, we can quantize the input current to within two bits of the reference current assuming that I_{IN} is less than $4I_{REF}$. This limitation is needed to maintain the linear relationship between conversion time and precision. At the end of T_{ref}, we define a residual time, T_{res}, as the time remaining beyond the last clock edge, as illustrated in Figure 3-2. Note that this temporal residue is analogous to the residual voltage in a successive sub-ranging converter. If we can quantize this residual time to within $N-2$ bits of precision, it follows that we have successfully quantized the input current to within N bits of the reference current.

3.1.2 Quantizing the Residue (Successive Subranging Stage)

Once we have obtained the first two bits, we need a method for subtracting the quantized signal from T_{ref} in order to operate on the residual time, T_{res}. Furthermore, we need a method for amplifying T_{res} such that we can quantize this residue to within one bit.

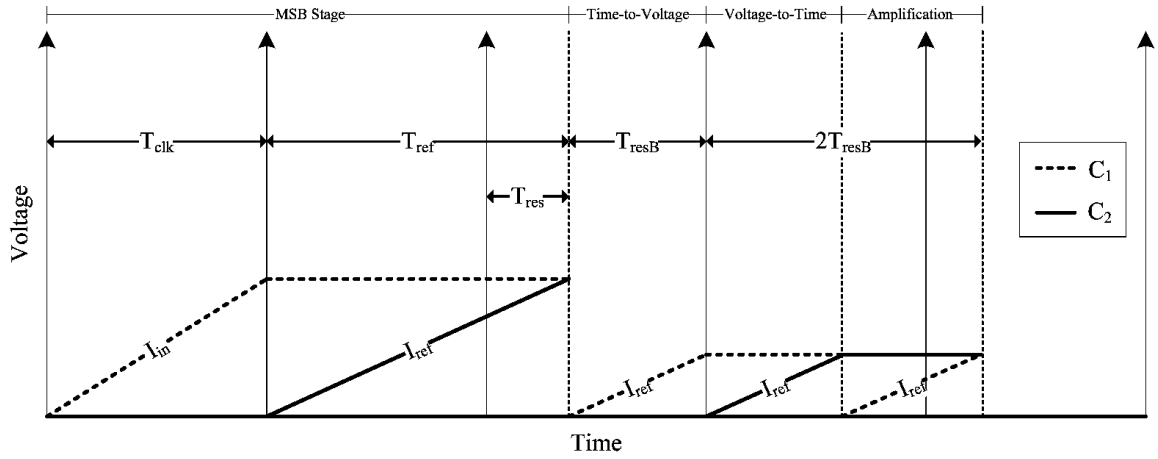

Figure 3-2: These waveforms represent the voltages on the two integrating capacitors, C_1 and C_2. For now, we are assuming that the comparator has zero offset and delay, i.e., the output is triggered as soon as the two voltages are exactly equal. In addition, we are ignoring any switching charge-injection. We also illustrate the timing variables T_{ref}, T_{res}, and T_{resB}, and the different stages of the converter.

The Subtraction Routine

In Figure 3-2, T_{res} is the result of subtracting the quantized signal from T_{ref}. Subtraction of intermediate quantization results is automatic in the algorithm if we ignore the integer number of quantization clock edges that have already passed (or not) and always reference our residual time signal to the next neighboring clock edge. Then the residual time after the last clock edge is defined as

$$T_{res} = T_{ref} - MT_{clk} \quad \forall \quad M \in [0, 4). \tag{3.2}$$

Let us also define T_{resB} as the time from the end of T_{ref} up to the next clock edge. Then,

$$T_{resB} = T_{clk} - T_{res}. \tag{3.3}$$

Quantizing T_{resB} is then equivalent to quantizing T_{res}: Quantizing x and $1 - x$ are equivalent as long as we can digitally compensate for referencing with respect to 1 rather than with respect to 0. We show how this is easily done in Section 3.1.3. Thus, by taking advantage of the precision of a low phase-noise clock with jitter less than $T_{clk}/2^N$ and always referencing the current residue to the next clock edge, we can

operate on T_{resB} in each stage of the conversion process.

Amplification: *Time-to-Voltage* and *Voltage-to-Time*

At the end of the MSB stage, the comparator's output goes high. This event signals a state-machine to reset both capacitors, C_1 and C_2, to zero and to rearrange the analog circuitry such that I_{REF} is now redirected to charge C_1. We can convert T_{resB} into a voltage by integrating C_1 with I_{REF} from the end of T_{ref} up to the next clock edge. Then the voltage on C_1 is

$$V_{C1} = \frac{I_{REF} T_{resB}}{C_1}. \tag{3.4}$$

We define the conversion of T_{resB} to V_{C1} as our *Time-to-Voltage* conversion. At this time, we switch I_{REF} over to charge C_2 until the voltage on C_2 is equal to the voltage in (3.4). The latter comparison can be achieved by using the same comparator used during the MSB stage. We define this process of reconverting V_{C1} to a time as our *Voltage-to-Time* conversion. As soon as the two voltages are equal, the voltage on capacitor C_1 is reset to zero, and we repeat the same charge integration on C_1. This integration cycle is defined as the amplification stage because at the end of this comparison, we have successfully amplified T_{resB} by two. By counting the number of clock edges seen within $2T_{resB}$, we can quantize T_{resB} to within one bit of T_{clk}. After we quantize $2T_{resB}$, we need to subtract this quantized value from $2T_{resB}$ to produce a new residue for successive conversions. To do so, we repeat our "subtraction" routine by encoding the time from the end of $2T_{resB}$ to the next clock edge as the new residue.

At the end of the amplification stage, we reset both capacitors to zero, and the state-machine reconfigures the analog circuitry such that I_{REF} is now set to charge C_2.[2] The elements are now in place to repeat the previous subtraction and amplification processes except that we are now doing *Time-to-Voltage* conversions on C_2 and *Voltage-to-Time* conversions using C_1. Each successive subtraction-and-amplification

[2] At this point in time, we are free to choose either capacitor C_1 or C_2 for the next *Time-to-Voltage* conversion. However, alternating the capacitors after each successive stage helps reduce the effect of capacitor mismatch, as discussed in Section 3.2.4 and Appendix C

process recursively yields one more bit in our converter.

3.1.3 Positive and Negative Index Counting

The subtraction routine forces us to quantize T_{resB} as opposed to the actual residue, T_{res}. As a result, each successive quantization stage weighs differently on the overall digital equivalent output. Specifically, if we define n_i as the i^{th} bit, then the quantized representation of I_{IN} with respect to I_{REF} is

$$I_{IN} = I_{REF}\left[n_0 + \frac{n_1}{2} + \sum_{i=2}^{N}(-1)^{i-1}\frac{(n_i+1)}{2^i}\right] \tag{3.5}$$

The first two bits n_0 and n_1 are simply the MSB and LSB outputs, respectively, of the two-bit counter during the MSB stage. However, unlike a traditional binary weighted summation, each successive bit during sub-ranging either adds or subtracts from the output. For example, the first subtraction-and-amplification stage of the sub-ranging process amplifies and quantizes T_{resB} as opposed to T_{res}. Therefore, the more clock edges we see during $2T_{resB}$, the smaller T_{res} is. Thus, we define this stage as a *negative-index* stage. The next stage of the sub-ranging process operates on the negative of the negative residue. We label this stage as a *positive-index* stage. The *negative-index* and *positive-index* stages alternate throughout the successive sub-ranging process.

Hardware implementation of *negative* and *positive-indexing* to create a traditional binary code is straightforward: Following the MSB stage, the one-bit counter in each *negative-index* stage inverts its bit, while the one-bit counter in each *positive-index* stage keeps its bit intact.

3.2 Error Cancellation Properties

So far, we have ignored several potential sources of error such as comparator offset and delay, switching charge-injection, state-machine delays, and capacitor mismatch. We will address each of these issues and introduce techniques that minimize these

effects on the performance of our converter.

3.2.1 Basics of Error Cancellation

The main idea behind our error cancellation technique is simple: Whatever elements "add" error to our signal during *Time-to-Voltage* conversion also "subtract" the same error from our signal during *Voltage-to-Time* conversion. Figure 3-3 provides an illustration of this basic principle and sections 3.2.2 through 3.2.5 provide further details.

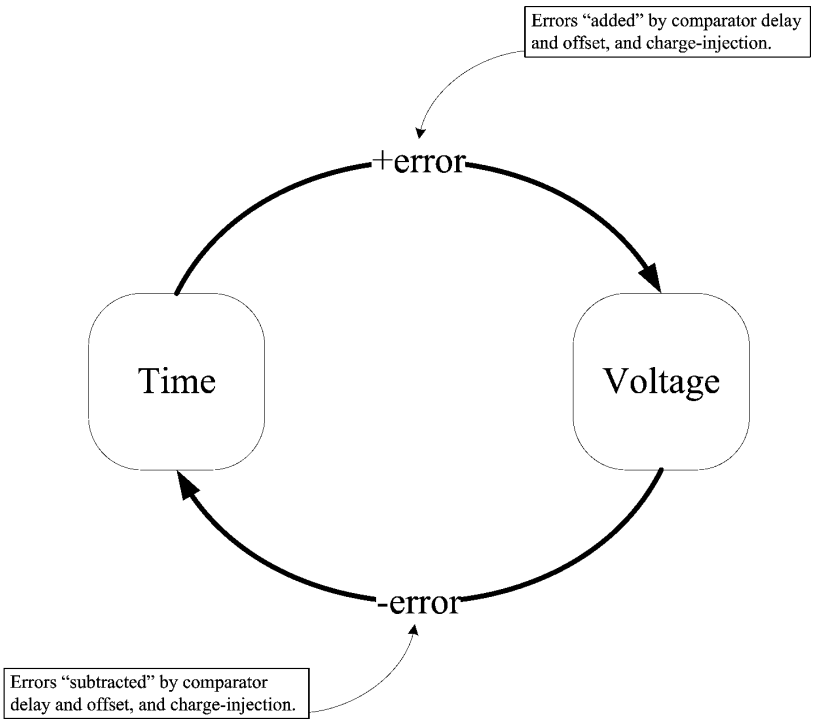

Figure 3-3: Whatever elements "add" error to our signal during *Time-to-Voltage* conversion also "subtract" the same error from our signal during *Voltage-to-Time* conversion.

3.2.2 Comparator Delay and Offset

Up to this point, we have assumed that the comparator has zero delay and offset, i.e., that the comparator output goes high as soon as the two voltages on the two capacitors are exactly equal. Since we are dealing with time as an intermediate signal variable, a voltage offset in our comparator can simply be translated into a comparator

delay. Therefore, our discussion below for minimizing the effects of comparator delay can also be applied to minimizing the effects of comparator offset.

At the end of the MSB stage, the comparator requires a finite amount of time to signal that the voltages on C_1 and C_2 are equal. Then T_{res} will be larger than its ideal value, which translates into a smaller T_{resB} through (3.3). If we define the first comparator delay as $T_{cd}(m_1)$,[3] then the adjusted residual time, T'_{resB}, is simply

$$T'_{resB} = T_{resB} - T_{cd}(m_1). \tag{3.6}$$

Naturally, the *Time-to-Voltage* conversion will produce a voltage on C_1 that is smaller by ΔV_{cd}, where

$$\Delta V_{cd} = \frac{I_{REF} T_{cd}(m_1)}{C_1}. \tag{3.7}$$

During the *Voltage-to-Time* conversion, I_{REF} charges C_2 up to the smaller voltage on C_1, i.e., V_{C1} is converted back into time with a gain term of C_2/I_{REF}. But the comparator again takes a finite amount of time before it signals that the two voltages are equal and adds a delay of $T_{cd}(m_2)$. Thus, the *Voltage-to-Time* conversion produces a temporal value of

$$\left(\frac{C_2}{C_1}\right) T_{resB} - \left(\frac{C_2}{C_1}\right) T_{cd}(m_1) + T_{cd}(m_2). \tag{3.8}$$

If we assume that capacitors C_1 and C_2 are equal, then (3.8) simply reduces to T_{resB}. In other words, at the end of the *Time-to-Voltage-to-Time* conversion, we have successfully cancelled out the first comparator delay and created a perfect replication of T_{resB}.

At this point, the ideal value of T_{resB} is encoded as a voltage on C_2. When we integrate C_1 with I_{REF} for our amplification stage, we reintroduce the comparator delay. Reintroduction of the delay is required for the successive stage to cancel, and the process starts anew. Figure 3-4 provides an illustration of comparator delay

[3]The comparator delay is approximately a first-order function of the slope of the input signal. This slope is denoted as m_i for a charging rate of I_{REF}/C_i. The DC value of the comparison affects the comparator delay to second-order.

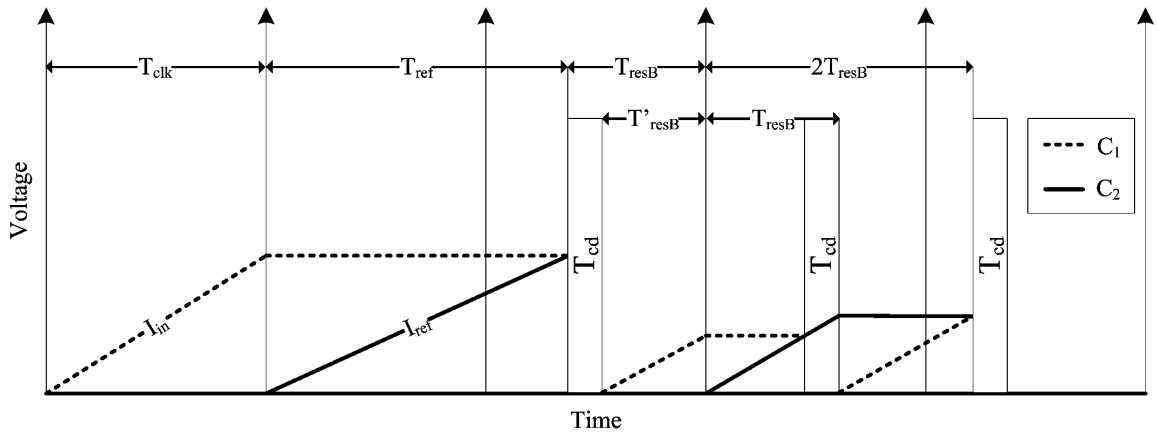

Figure 3-4: The block labeled T_{cd} represents the comparator delay. T'_{resB} is defined as $T_{resB} - T_{cd}$. Even though the comparator delay is a function of the input charging rate, T_{cd} is constant in any stage of the conversion and we may ignore effects caused by its variation since we always use the same charging rate of I_{REF}/C (assuming $C_1 = C_2 = C$). At the end of the *Voltage-to-Time* conversion, the threshold voltage on V_{C2} encodes the true residual time, T_{resB}, as opposed to T'_{resB}. As a result, when the voltages on the two capacitors are equal at the end of the amplification stage, we have produced an exact representation of $2T_{resB}$. The last comparison reintroduces T_{cd} for subsequent cancellation in the next stage of conversion.

cancellation.

3.2.3 Switching charge-injection

There exist two instances where switching charge-injection could potentially affect the converter. First, each capacitor requires a switch to reset its charge to zero, and each time the reset switch turns off, it releases some charge onto its respective capacitor. However, the voltages around the reset switches do not vary, i.e., $V_{GS} = V_{DD}$ and $V_{DS} = 0$. Therefore, the amount of charge-injection is constant and can be treated as a simple DC offset on the capacitor's "zero" value. Second, each time a current source switches away from a capacitor, it introduces a finite amount of charge onto the capacitor that varies with the voltage drop across the gate-source junction of the switch MOSFET [34]. And while the gate voltage is always constant, the source voltage varies throughout the conversion process. As a result, the amount of this charge-injection varies. However, as we show below, the effects of this charge-injection are minimized in a manner similar to that of the comparator delay. Figure 3-5 provides

a graphical illustration of this error cancellation.

During the MSB stage, when I_{IN} stops charging C_1, charge-injection increases the voltage on C_1 by some voltage, V_{sw}.[4] Thus, T_{ref} will be larger than its ideal value by

$$\frac{V_{sw}C_2}{I_{REF}}, \tag{3.9}$$

or equivalently,

$$T'_{resB} = T_{resB} - \frac{V_{sw}C_2}{I_{REF}}. \tag{3.10}$$

During the *Time-to-Voltage* conversion, I_{REF} charges C_1 for a slightly shorter time producing a smaller voltage on C_1. But when I_{REF} stops charging C_1 and switches over to charge C_2, a similar amount of charge is dumped onto C_1 such that the time-converted voltage on C_1 increases by V_{sw}. Specifically,

$$V_{C1} = \left(\frac{I_{REF}}{C_1}\right) T'_{resB} + V_{sw}. \tag{3.11}$$

Assuming that $C_1 = C_2$ and that both amounts of charge-injection are equal, we can combine (3.10) and (3.11) such that V_{C1} reduces to

$$\frac{I_{REF}T_{resB}}{C_1}. \tag{3.12}$$

In other words, V_{C1} encodes the ideal residual time such that the following *Voltage-to-Time* conversion perfectly reconstructs T_{resB}. During the amplification stage, I_{REF} stops charging C_2 and switches over to charge C_1 which reintroduces charge-injection for the subsequent stage to cancel.

If charge-injection was constant, this cancellation would be perfect. Since the voltages at which the MOSFETs switch vary from stage to stage, the amount of charge dumped onto the capacitors varies as well and the cancellation is imperfect. If we model the variation in charge-injection as a function of the switching voltage, we can determine a simple constraint on the size of the integrating capacitor for any

[4]We are assuming PMOS current switches such that the sign of the charge-injection is positive.

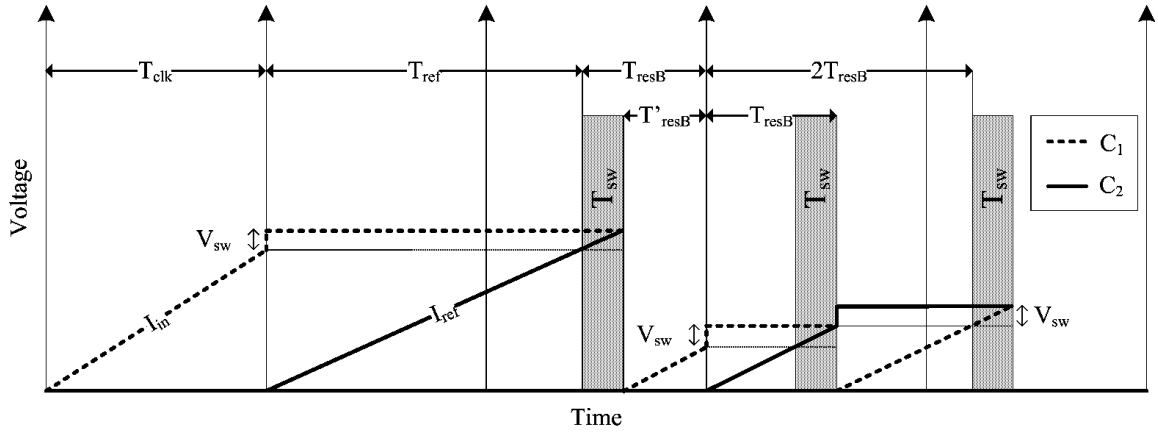

Figure 3-5: The change in the capacitor voltage, V_{sw}, is due to the charge-injection from switching a current source away from the threshold capacitor. The block labeled T_{sw} is the net change in time as a result of the charge-injection. Note that this time is a function of the charging rate, I_{REF}/C_i. Since this charging rate remains constant throughout the process, we can ignore slope-dependent variations. T'_{resB} is defined as $T_{resB} - T_{sw}$. At the end of the *Voltage-to-Time* conversion, we have successfully reproduced the ideal residual time, T_{resB}. We then reintroduce the charge-injection error such that the next successive stage can cancel it.

required precision. Equation (3.13) is a MOSFET charge-injection model from [35] where W and L are the width and length, respectively, of the switch MOSFET, C_{ox} is the oxide capacitance, V_T is the threshold voltage, C_{ov} is the overlap capacitance, and V_{GS} is the gate-source voltage.

$$Q_{sw} = C_{ox}WL(V_{GS} - V_T) + C_{ov}WV_{GS} \tag{3.13}$$

Observing that V_{GS} can never vary by more than V_{DD}, we can show that the variation of this charge from the MSB stage to the successive stage is bound by

$$\Delta Q_{sw} < V_{DD}(C_{ox}WL + C_{ov}W). \tag{3.14}$$

Equivalently, the bound on the differential voltage due to the varying charge-injection is

$$\Delta V_{sw} < \frac{V_{DD}(C_{ox}WL + C_{ov}W)}{C_i}. \tag{3.15}$$

For N bits of precision, we must ensure that

$$\Delta V_{sw} < \frac{V_{DD}}{2^{N-2}}, \tag{3.16}$$

where our algorithm accounts for 2 bits of over-ranging on the input. Combining (3.15) and (3.16), we can derive the following constraint on the size of the integrating capacitors, C_i;

$$C_i > 2^{N-2}(C_{ox}WL + C_{ov}W). \tag{3.17}$$

Table 3.1 summarizes the technologies used in Chapters 4 and 5 and their respective constraints on capacitor sizing as a function of precision. The specific device parameters were taken from fabrication run T35L for the 0.35μm process and T46U for the 0.18μm process.

Table 3.1: Capacitor matching requirements

Technology	MOSIS TSMC 0.35μm	MOSIS TSMC 0.18μm
C_{ox} (PMOS)	$4.564 f\text{F}/\mu\text{m}^2$	$8.271 f\text{F}/\mu\text{m}^2$
C_{ov} (PMOS)	$0.355 f\text{F}/\mu\text{m}$	$0.66 f\text{F}/\mu\text{m}$
Minimum Size	$W = 0.6\mu\text{m}$, $L = 0.35\mu\text{m}$	$W = 0.27\mu\text{m}$, $L = 0.18\mu\text{m}$
$C_{i,min}(N)$	$2^{N-2}(1.083) f\text{F}$	$2^{N-2}(0.521) f\text{F}$
	$0.128 p\text{F}$ (N=12)	**$33.3 f\text{F}$ (N=8)**

3.2.4 Reducing Capacitor (Gain) Mismatch

So far, we have assumed that the two integrating capacitors are exactly equal. And while capacitor matching can exceed 12 bits of precision without calibration [32], our algorithm attempts to minimize the effects of capacitor mismatch through a simple capacitor-alternating scheme.

If we assume that C_2 is slightly larger than C_1, then T_{res} is slightly larger at the end of the MSB stage, and inversely, T_{resB} is slightly smaller. At this point in time, we are free to choose either C_1 or C_2 for the successive *Time-to-Voltage* conversion. If we integrate with C_1, V_{C1} encodes the smaller value of T_{resB} with a smaller voltage. But when we integrate with the larger capacitor, C_2, for the *Voltage-to-Time* conversion,

we increase the "mirrored" value of T_{resB} such that we are now closer its ideal value. We then integrate with C_1 for the amplification stage and reintroduce the capacitor mismatch error. Again, we are free to choose either capacitor for the successive *Time-to-Voltage* conversion. But instead of integrating with C_1 like we did for the previous *Time-to-Voltage* conversion, we integrate with C_2, and the error minimization process begins anew.

Unfortunately, unlike the previous error cancellation schemes, this error minimization technique is far from perfect. The gain errors are heavily dependent on the absolute values of the timing variables, and consequently, the effectiveness of this error minimization technique varies widely from stage to stage. Nevertheless, the capacitor-alternating scheme prevents capacitor mismatches from aggregating from stage to stage. A detailed analysis is provided in Appendix C.

3.2.5 State-Machine Delay

Each time the state-machine enters a new state, there exists a small delay, T_{sm}, before the converter can reconfigure the analog circuit topology. There exist two classes of state-machine delays, each of which could potentially affect the precision of our converter. The first class of state-machine delays involves transitions that are triggered by a clock edge. The second class of state-machine delays involves transitions that are triggered by the falling edge of the comparator's output. Let us first examine the effect of T_{sm} after clock edge-triggered state transitions. These delays can be ignored because we can simply view this as a "phase shift" in the clock waveform. As a result, it does not affect our algorithm. Note that the lack of a "phase shift" in the intermediate clock edges, i.e., edges that do not cause state transitions, does not affect the converter.

The effects from T_{sm} after comparator edge-triggered state transitions are perfectly analogous to the effects of the comparator delay. These state-machine delays simply add to the comparator delay, and as a result, the analysis in Section 3.2.2 also applies to these state-machine delays.

3.3 Clock-Straddling Residues

The introduction of comparator delays and its finite pulse-width outputs creates scenarios where the amplified temporal signal can be greater than two clock cycles. Specifically, at the end of the amplification phase when the voltages on the two integrating capacitors are exactly equal, the ideal residue up to the following clock edge, T_{resB}, could be smaller than the sum of the comparator delay and the pulse-width output, T_{cd+pw}. Figure 3-6 illustrates such an example where $T_{resB} < T_{cd+pw}$. The following stage then doubles the ideal residue in addition to the extra clock cycle. As a result, the two extra clock cycles signal to the registers that the previous stage's residue was in fact smaller than the comparator delay and pulse-width output and digitally subtracts 1 from the previous stage's counter while resetting the current stage's counter to 0. Since the digital correction propagates in the reverse direction, the correction must be performed after all of the desired bits have been computed.

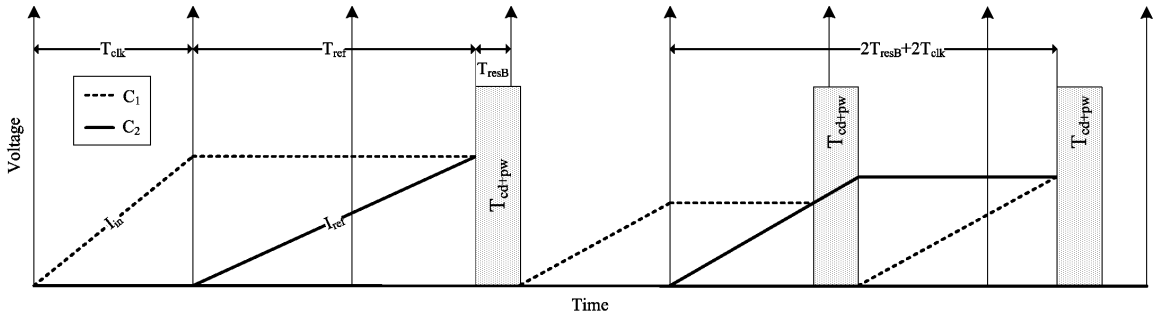

Figure 3-6: In this example, the residue is smaller than the comparator delay and pulse-width output ($T_{resB} < T_{cd+pw}$) such that the following stage amplifies the residue in addition to a clock cycle.

3.4 $1 + \varepsilon$ Algorithm

A practical limitation to our algorithm is the possibility of an infinitesimally small residue in T_{resB}. Then our *Time-to-Voltage* conversion would produce an infinitesimally small voltage to operate on. This is a problem inherent to many time-to-digital converter (TDC) designs [36] [37]. Therefore, in a manner similar to other TDC's, we

instituted a $1+\varepsilon$ algorithm where we guarantee a minimum voltage in our *Time-to-Voltage* process by always integrating for an extra clock period. For example, instead of integrating for T_{resB}, we integrate for an extra clock cycle such that the time-converted voltage reflects $T_{resB}+T_{clk}$. Note that this modification to the algorithm is analogous to the clock straddling scenario presented in Section 3.3. We can then guarantee that the voltage that results from *Time-to-Voltage* conversion will be at least V_{FS}, where

$$V_{FS}=\frac{I_{REF}T_{clk}}{C_i}. \tag{3.18}$$

We can digitally subtract the two extra clock cycles generated during amplification with minimal overhead in our counter. We can also view this modification as a common-mode voltage offset such that we are always operating V_{FS} above zero volts.

Unfortunately, the average conversion time will increase by an additional $3N$, as illustrated in Figure 3-7. In order to minimize the impact of the $1+\varepsilon$ algorithm on the conversion time of the converter, we can implement a level-dependent algorithm that integrates for an extra clock cycle if the *Time-to-Voltage* conversion produces a minimum threshold voltage. A simple implementation would involve utilizing a separate, imprecise, low-power comparator that signals to the state-machine if the *Time-to-Voltage* conversion exceeds some threshold voltage. The counters in the registers would make the appropriate modifications depending on the outcome of this comparison. Figure 3-8 shows that the conversion time can be reduced to $4N$ by simply setting the minimum threshold voltage to be $V_{FS}/2$. Another approach is to implement the $1+\varepsilon$ algorithm for the first n bits past the 2 MSBs. The value of n can be modified by the user until the desired precision is achieved.

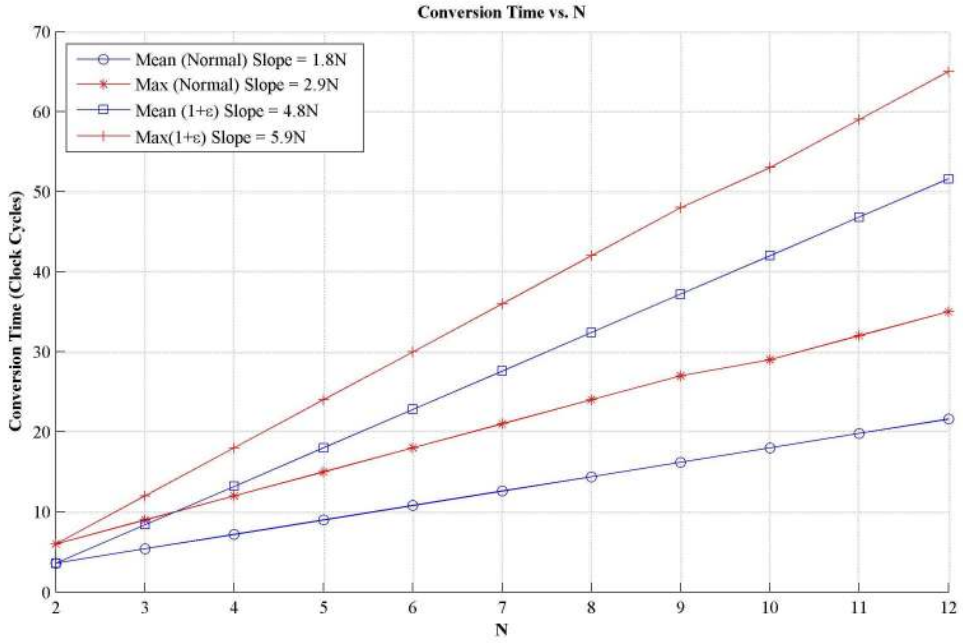

Figure 3-7: The conversion time as function of the total bit count of the converter is shown for both the unmodified (normal) algorithm and the $1+\varepsilon$ algorithm. Both the maximum and average conversion times increase by $3N$ for the $1+\varepsilon$ algorithm. Note that this conversion time model includes comparator delays and non-zero comparator pulse-widths.

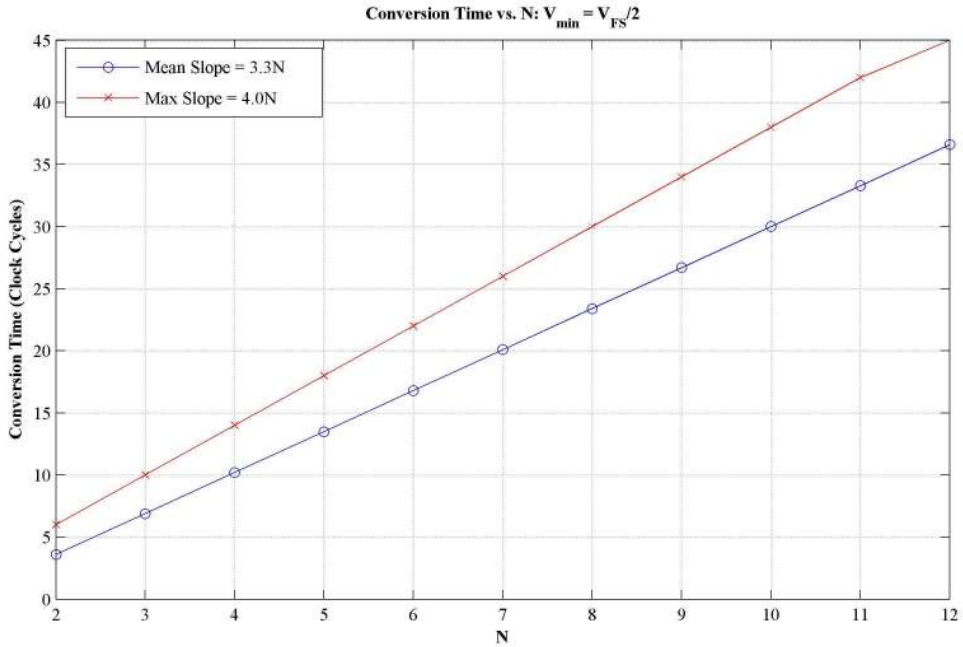

Figure 3-8: The conversion time is reduced from $5.9N$ to $4N$ by implementing a minimum threshold voltage requirement of $V_{FS}/2$.

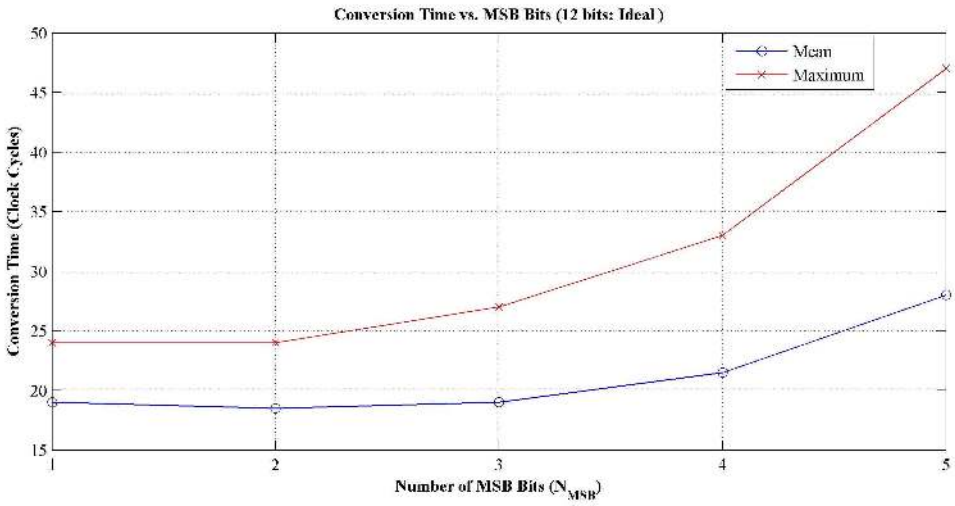

Figure 3-9: We have ignored the $1+\varepsilon$ algorithm and comparator delays and finite pulse-width outputs such that the smallest conversion time is achieved when $N_{MSB} = 1$ or $N_{MSB} = 2$.

3.5 Determining the Over-Ranging Factor

So far, we have assumed that the input current is limited to $4I_{REF}$. This upper-bound is determined by several factors: The conversion time of the converter, the power consumption, and the noise performance all play a role in determining the over-range factor. Let us define N_{MSB} as the number of bits computed during the MSB phase.

If we ignore the $1+\varepsilon$ algorithm and assume that the comparator has zero delay and an infinitesimally narrow pulse-width output, then the maximum conversion time grows proportionally to $2N$. Figure 3-9 plots the average and maximum conversion times for different values of N_{MSB} and shows that $N_{MSB} = 1$ or $N_{MSB} = 2$ produces the smallest maximum conversion time.

However, due to non-zero comparator delays and finite pulse-width outputs, certain input values to the converter will create clock-straddling scenarios during successive stages such that the maximum conversion time grows proportionally to $2.9N$ (See Figure 3-7). Then $N_{MSB} = 2$ produces the smallest maximum conversion time, as shown in Figure 3-10.

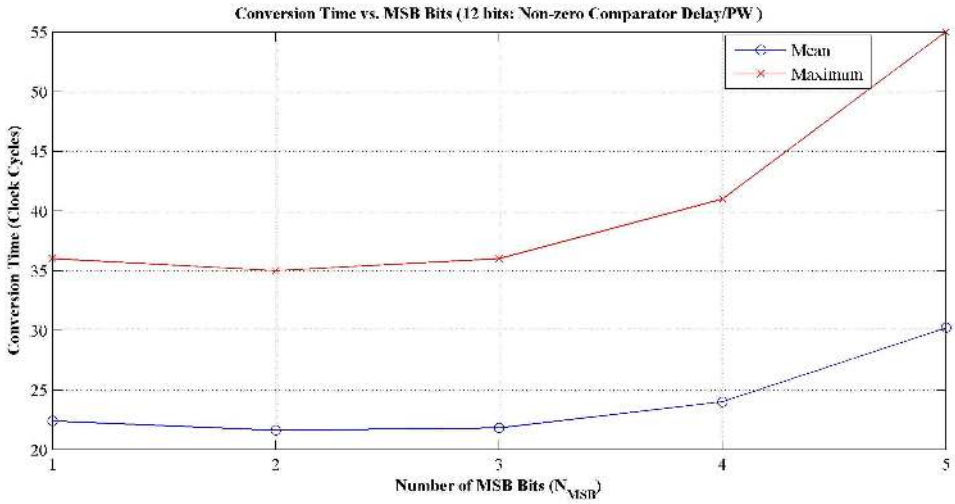

Figure 3-10: We have ignored the $1+\varepsilon$ algorithm but included the effects of comparator delays and finite pulse-width outputs such that the smallest conversion time is achieved when $N_{MSB} = 2$.

Unfortunately, the addition of the $1+\varepsilon$ algorithm increases the conversion time significantly such that it grows proportionally to $5.9N$. Figure 3-11 shows the conversion time when the $1+\varepsilon$ algorithm is implemented throughout the successive stages. In this scenario, $N_{MSB} = 3$ produces the smallest maximum conversion time.

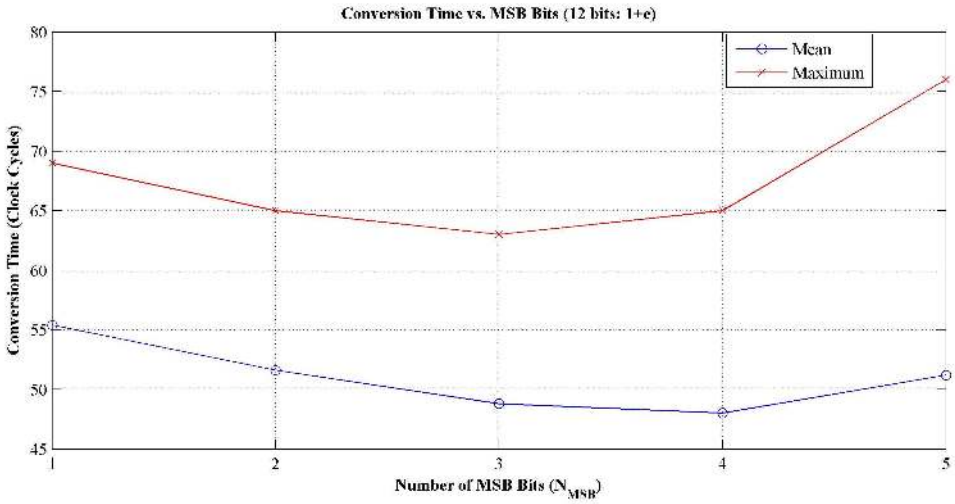

Figure 3-11: We have implemented the $1+\varepsilon$ algorithm throughout the conversion process and included the effects of comparator delays and finite pulse-width outputs. Since the maximum conversion time grows proportionally to $5.9N$, the minima is achieved when $N_{MSB} = 3$.

Chapter 4

Implementation and Analysis of a Thermal-Noise-Limited Low-Power 12 bit Converter

This chapter is partitioned into three sections: First, we discuss the implementation of the key building blocks of our converter such as the comparator, state-machine, and registers. Second, we present a novel time-based noise analysis that utilizes timing variables as opposed to voltage or current. Finally, we present experimental results from a Very Large-Scale Integration (VLSI) implementation.

4.1 Design and Implementation

The overall design of our A/D converter consists of two capacitors, an analog switching network, a reference current, an asynchronous comparator, a state-machine, and registers and is shown in Figure 4-1. Using the control signals from the state-machine, the analog switch network routes the currents to the appropriate capacitors and the capacitors to the appropriate input terminals of the comparator. Note that the comparator includes a finite pulse-width control mechanism, which is discussed in Section 4.1.2. The state-machine also controls the behavior of the registers which count and store the output bits. The timing requirements between the state-machine and reg-

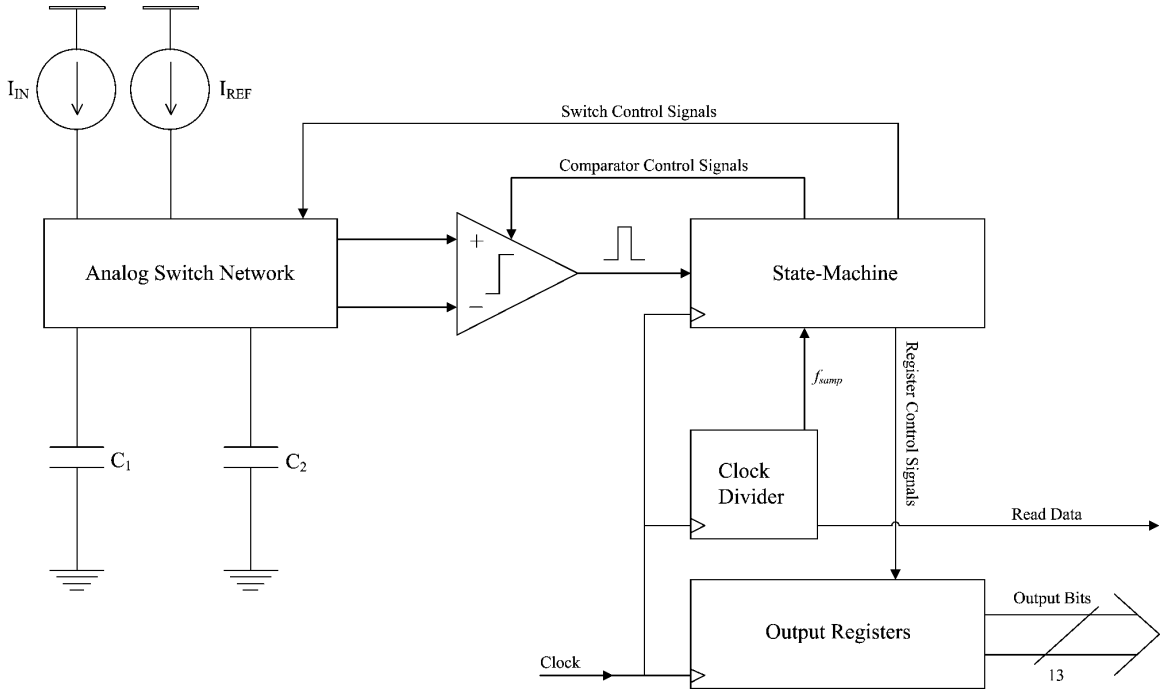

Figure 4-1: Block diagram of our ADC

isters are non-trivial and require careful consideration during the design of both of these blocks. The clock-divider block generates a control signal that sets the sampling frequency of the converter.

4.1.1 Asynchronous Comparator

Like many A/D converters, the most critical component of our converter is the comparator. However, the performance requirements for the comparator in our converter are rather unique. Since our algorithm is immune to comparator delay and offset, the constraints on these two parameters are generous. In essence, the natural offsets inherent to comparator designs due to transistor mismatches are negligible in our converter. Furthermore, the absolute value of the comparator delay has little effect on our converter other than increasing the conversion time. Instead, the variations of the comparator delay from comparison to comparison can limit the precision of the converter. But more importantly, we will show in Section 4.2 that the thermal noise characteristics of the comparator proves to be the limiting factor in the overall precision of the converter. In general, the lower the power budget of a converter, the

lower is the precision at which it becomes thermal-noise-limited since there are fewer number of electrons to average over a given bandwidth.

The Pre-amplifier

The comparator is comprised of two stages: the pre-amplifier and the gain/latch stage. The pre-amplifier is a simple resistively loaded differential pair and forms the first stage of the comparator (See Figure 4-2). The pre-amplifier provides low gain and kick-back isolation for the input from the second stage of the comparator, a gain/latch stage. In Figure 4-2, the two capacitors labeled C_p are parasitic capacitors introduced by the poly-to-substrate capacitance for the poly resistors, R_p. While unavoidable, these capacitors actually help limit the noise-bandwidth of our pre-amplifier. The geometry of the resistors can be shaped to find the optimal capacitance for minimizing noise while avoiding slew-limit conditions. The bias current, I_{BIAS}, is generated by transistor $M1$ which is cascoded by $M2$. The cascode structure provides high Common-Mode-Rejection-Ratio (CMRR) which is required for minimizing delay variations that are dependent on the input's common-mode voltage. In addition, the differential PMOS transistors $M3$ and $M4$ have their substrates connected to V_{DD} in order to minimize common-node capacitance and feed-through [38]. The outputs of the pre-amplifier, $V+$ and $V-$, are passed onto the second stage of the comparator, the gain/latch stage.

The Gain/Latch Stage

The gain/latch stage of the comparator is a wide-output swing operational transconductance amplifier with a unidirectional positive feedback mechanism and is shown in Figure 4-3. As $V+$ approaches $V-$, the output voltage, D_{out}, starts increasing from close to 0 volts. As D_{out} increases above the threshold voltage of transistor $M17$, this NMOS briefly pushes a large, increasing current into the output capacitor, C_L, via the current mirror formed by transistors $M13$ and $M14$. Transistors $M15$ and $M16$ provide a mechanism for the state-machine to reset and disable the output via the control signal, *Disable*. Again, the differential pair PMOS transistors $M7$ and $M8$

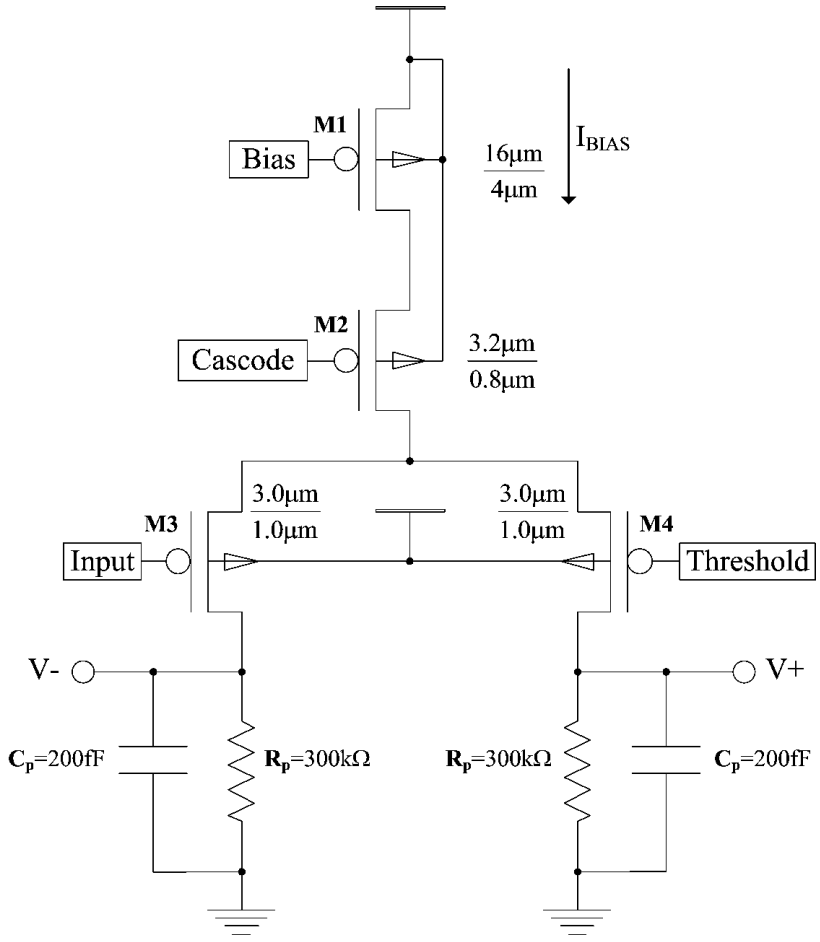

Figure 4-2: The pre-amplifier of the comparator is shown with circuit parameters from the high performance ADC.

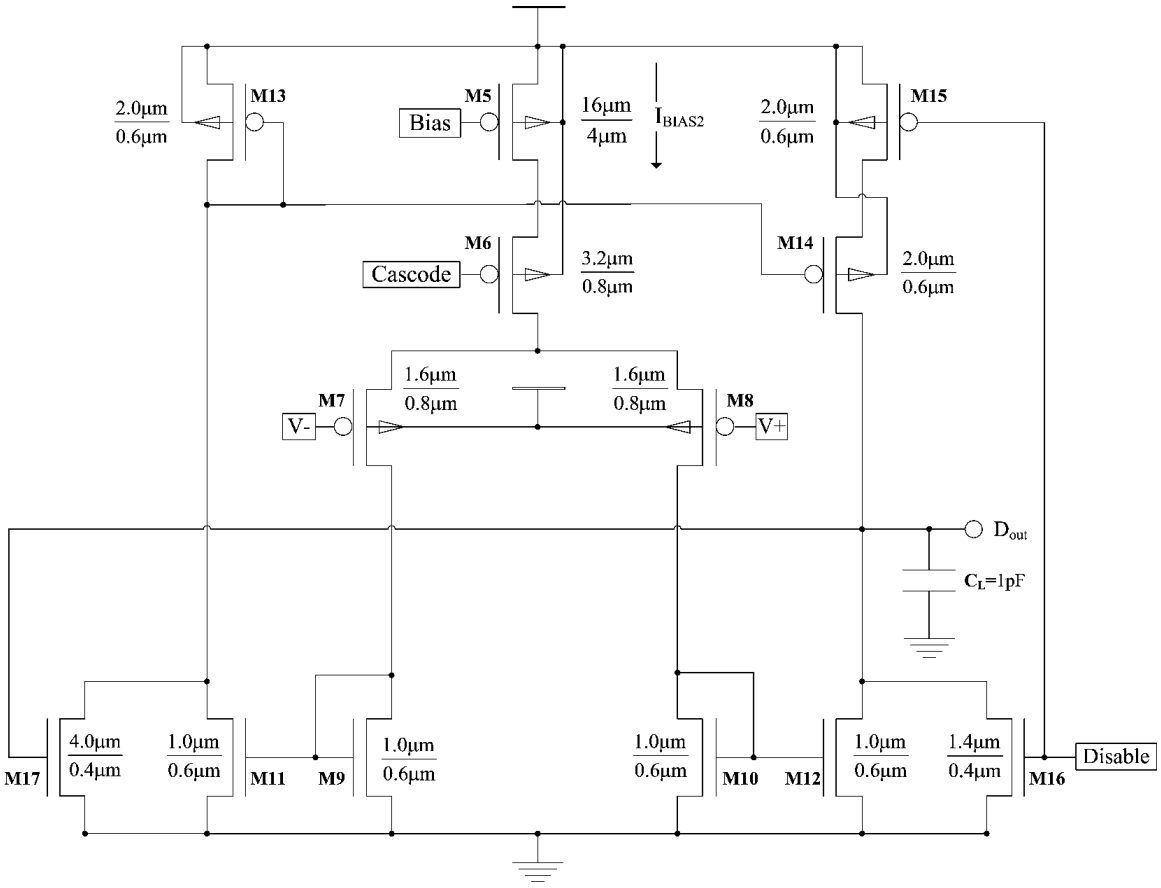

Figure 4-3: The gain/latch stage of the comparator is shown with circuit parameters from the high performance ADC.

have their substrates connected to V_{DD}. The output of the gain/latch stage is then passed onto the pulse-width control mechanism.

4.1.2 Pulse-Width Control Circuit

The pulse-width control circuit is an adaptation of the spiking neuron circuits in [39] [40] and is shown in Figure 4-4. When the pulse-width control circuit is inactive, nodes X, H, and Out are low and nodes Y and $Outb$ are high. Note that node H is pinned low through transistor $M13$. As soon as the output of the gain/latch stage, $Dout$, increases past the threshold of inverter $INV1$, the pulse-width control circuit is activated. First, the output of the gain/latch stage is "sharpened" by inverters $INV1$ and $INV2$. Since Out is low and $Outb$ is high, the "sharpened" signal is passed directly onto node X through transmission gate A which in turn sets node Y

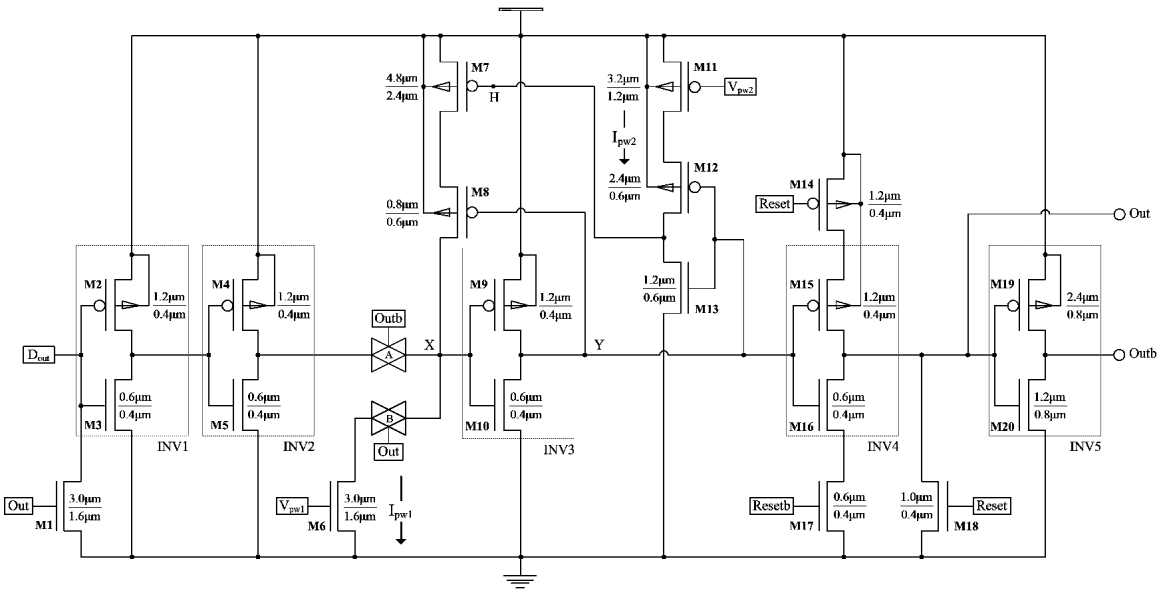

Figure 4-4: The output of the comparator, D_{out}, is passed as the input to the pulse-width control circuit above. The width of the pulse is controlled by two currents, I_{pw1} and I_{pw2}. The state-machine resets the pulse-width circuit with the complementary control signals *Reset* and *Resetb*.

to zero through inverter $INV3$ and the positive-feedback transistor $M8$. Assuming that the state-machine has set the control signals *Reset* and *Resetb* to the correct values, node *Out* goes high and *Outb* goes low through the final two inverters, $INV4$ and $INV5$. The change in these two signals then initiates the *refractory* phase of the pulse-width circuit. Transmission gate A prevents the gain/latch stage from influencing the pulse-width circuit during the *refractory* phase by detaching inverters $INV1$ and $INV2$ from the rest of the circuit. Alternatively, transmission gate B connects current source $M6$ to node X. Initially, this current source cannot overcome the strong positive feedback loop around inverter $INV3$. However, as current source $M11$ charges node H, transistor $M7$ slowly turns off, reducing transistor $M8$'s hold on node X. Eventually, current I_{pw1} pulls down node X below the threshold of inverter $INV3$ and the pulse-width circuit returns to its initial state. Figure 4-5 provides an illustration of the intermediate node voltages. The pulse-width circuit is an electronic analog of inactivating positive feedback in Na^+ channels and delayed negative feedback in K^+ channels that determine the pulse width of action potentials in biological neurons.

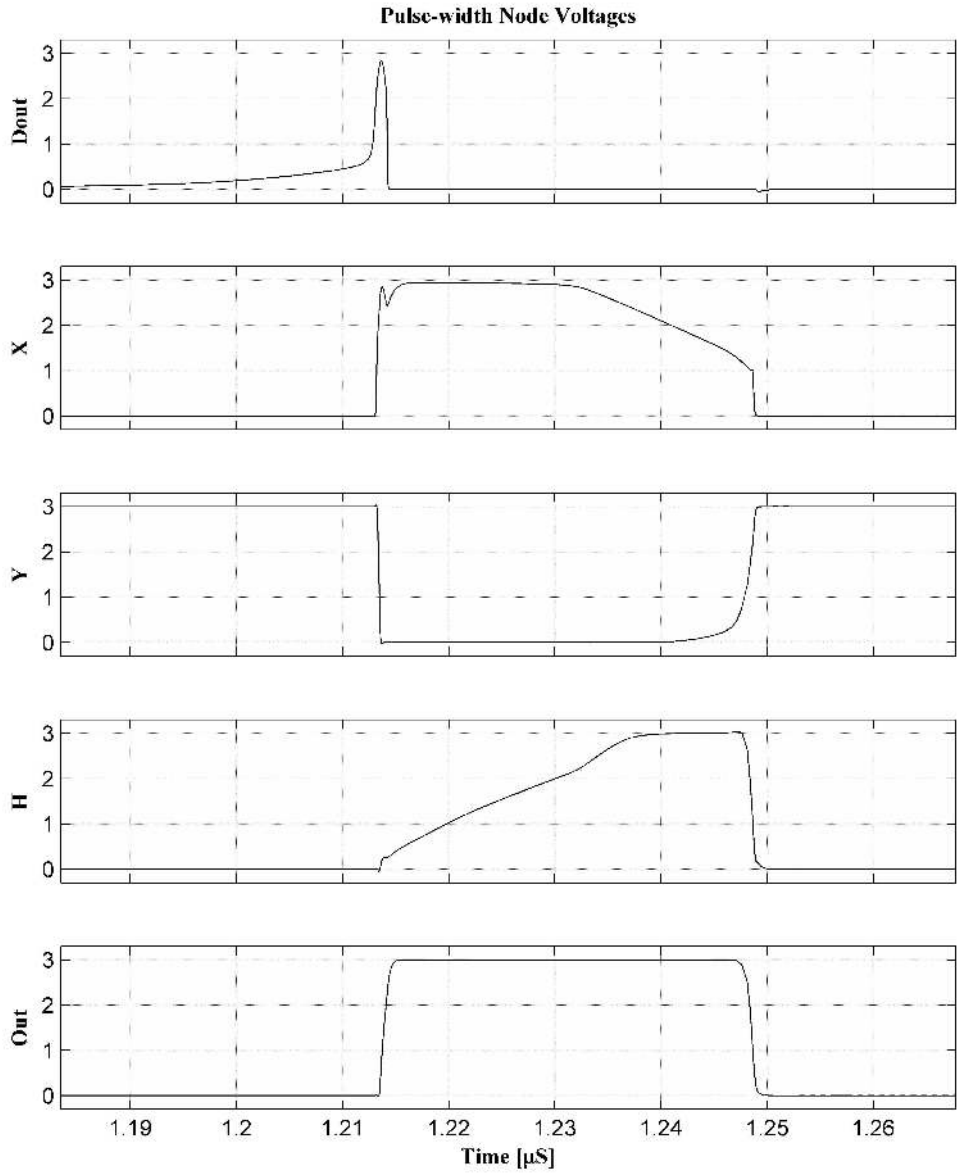

Figure 4-5: The internal node voltages from a simulation of the pulse-width control circuit are shown.

4.1.3 Differential Current Switching

In order to minimize the absolute level of charge-injection and maximize the switching speed of the reference current source, we utilized a cascoded differential current switching topology often used in current-mode Digital-to-Analog converters (DAC) and is shown in Figure 4-6. The reference current source is always connected to C_1, C_2, or ground. As long as the maximum voltage on both C_1 and C_2 satisfy the $V_{D,SAT}$ condition of transistor Msc, the reference current source transistor, $Mref$, will always remain in saturation. Even when the current is sinked to ground and is in its "off" state, $Mref$ will remain in saturation allowing for high speed operation. Furthermore, the gate voltages to the switch transistors swing from V_{DD} to a cascode voltage, V_{CASC}, such that the switching charge-injection is minimized and the output resistance of the current source is increased.

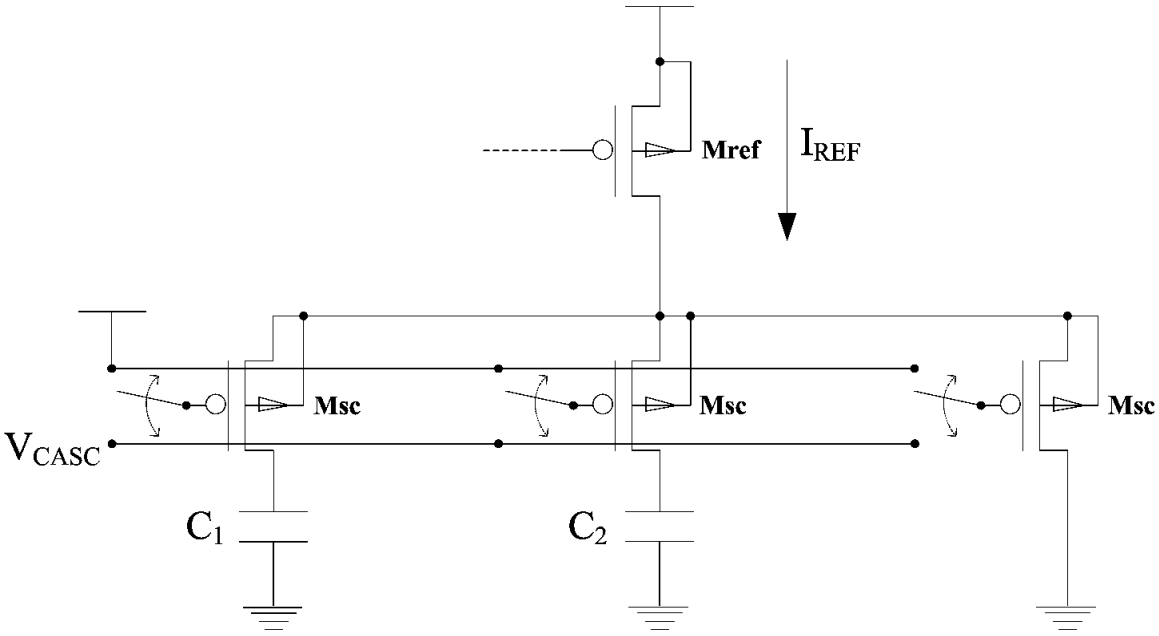

Figure 4-6: The reference current source is switched to C_1, C_2, or ground via the switch transistors, Msc. The gates to these switches transition from V_{DD} to a cascode voltage, V_{CASC}, which minimizes switching charge-injection. In addition, the current transistor, $Mref$, is always operating in saturation even in its "off" state, i.e., when the current is sinked to ground, which allows for high speed operation.

4.1.4 State-Machine

A state machine controls the charging and discharging of capacitors and coordinates the overall operation of the converter. We implemented the state-machine using true single-phase edge-triggered flip-flops (See Figure 4-7). State transitions may be triggered by clock edges or the falling edge of the comparator's pulse-width output, depending on the state. Using the falling edge of the pulse-width output allows us to treat the pulse-width output as a comparator delay which effectively cancels its temporal affect. Due to the recursive nature of the algorithm, we were able to implement a 9 state controller with 6 recursive states, as illustrated in Figure 4-8.

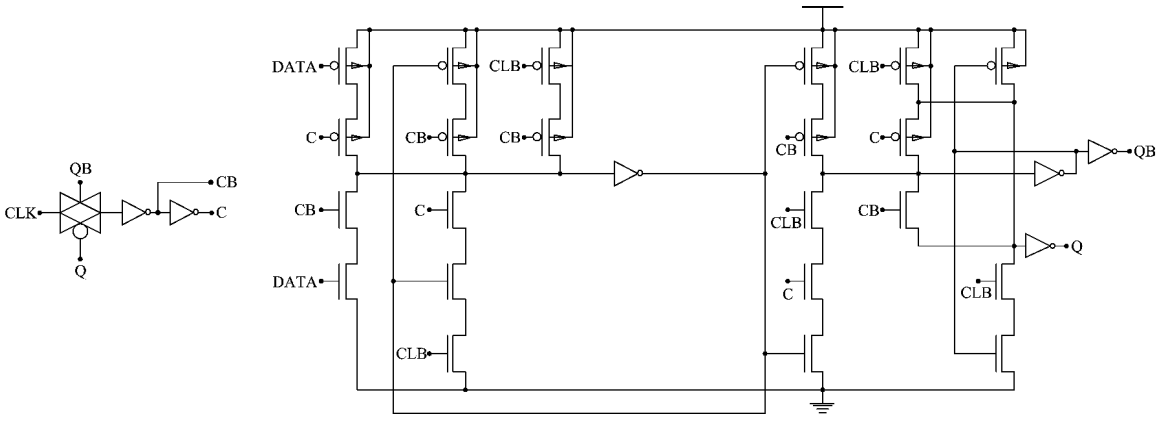

Figure 4-7: A classic master-slave flip-flop was utilized in both the state-machine and registers.

Different combinations of the state signals are passed onto a set of matched-logic modules. Each module consists of a latch with output buffers and reset transistors, as shown in Figure 4-9. Additional reset transistors, Msw, are added for "OR"ing of state signals. These transistors must be sized so that they can overcome the positive feedback mechanism of the latch. The use of matched-logic modules as opposed to traditional logic gates allows us to control the timing of the complementary output signals, $SWITCH$ and $SWITCHb$, such that the corresponding complementary switches in the analog switch network transition at approximately identical times.

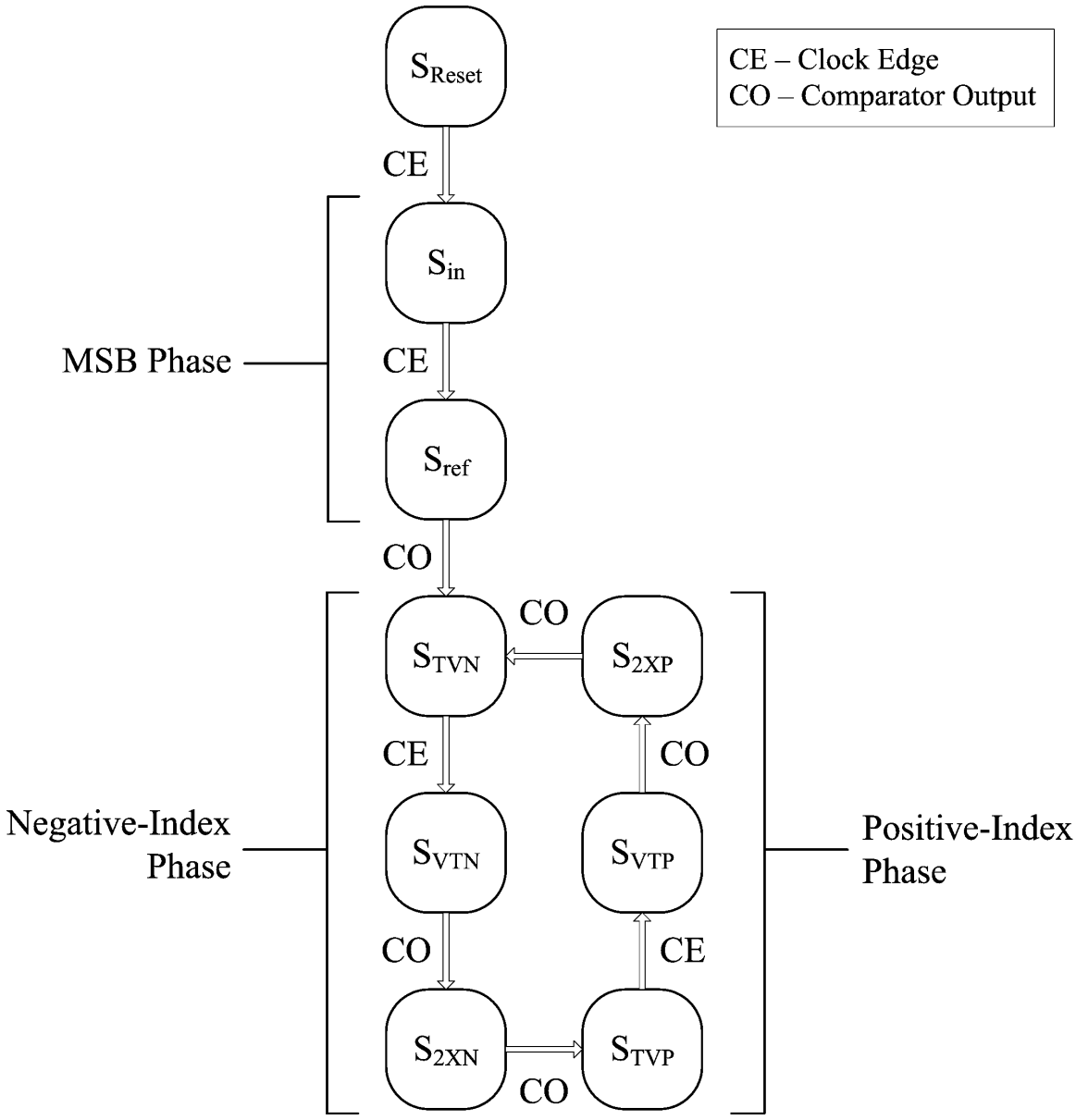

Figure 4-8: This is a simplified representation of the state-machine where we have ignored states that implement an enhanced version of the algorithm termed "$1+\varepsilon$" and exit states. In any state, either a positive clock edge or the falling edge of the comparator's finite pulse-width output can trigger a state transition.

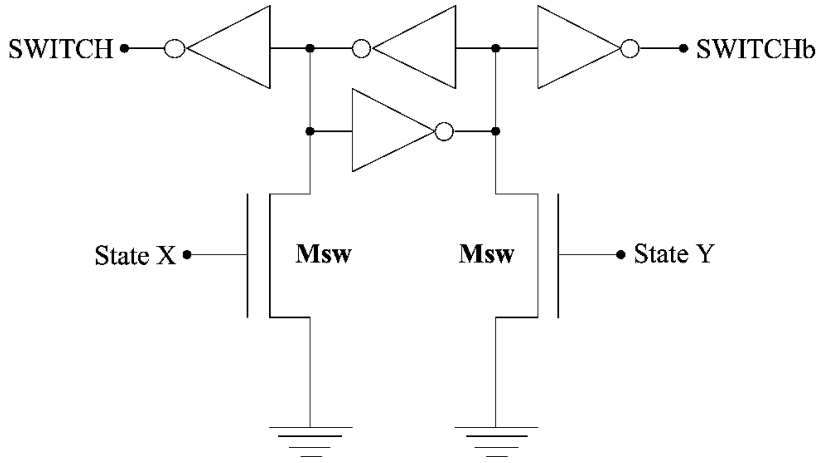

Figure 4-9: A combination of state variables are used in matched-logic modules whose complementary output signals, $SWITCH$ and $SWITCHb$, are passed onto the analog switch network. Additional switch transistors, Msw, are added accordingly to account for multiple state variable inputs.

4.2 Noise Analysis

In order to quantify the maximum achievable precision for our architecture, we must analyze all potential sources of error. If designed properly, the converter's ultimate precision should be limited by thermal noise and quantization error. In this section, we will explore how quantization error and thermal noise affect our system. However, since our converter uses a novel time-based algorithm, we first introduce a unique time-based noise analysis method.

4.2.1 Temporal Jitter vs. Voltage Noise

Since we use time as our primary signal variable, it is best to look at noise in neither the voltage nor current domain, but in the time domain. In traditional voltage-mode designs, we can determine our signal-to-noise ratio by comparing a full-scale voltage to some voltage noise. In our case, we compare our full-scale signal, $4T_{clk}$, to the temporal jitter induced by a voltage noise. For instance, at the end of the MSB stage, the comparator's output will jitter, causing T_{resB} to vary by some amount. Voltage and current noise from several different components in our system contribute to this jitter, and we can quantify that contribution by examining how voltage noise translates

to temporal jitter in our system. In the case of the comparator's pre-amplifier, we calculate the equivalent temporal jitter by dividing the input-referred voltage noise by its respective input-voltage ramp. This method also applies to the calculation of the integration noise. However, due to the non-linear nature of the gain/latch stage, we need to employ a slightly different noise calculation method. The total output voltage noise of the gain/latch stage is superimposed on its slew-limited output-voltage ramp. By dividing the total output-voltage noise by the output-voltage ramp's slope, we obtain an equivalent temporal jitter.

4.2.2 Quantization Error

As is true for all data converters, the maximum achievable SNR is determined by the quantization error. As discussed in Section 2.3.1, there is exists a noise energy associated with quantizing to within N bits of precision. For the case of a full-scale signal of $4T_{clk}$, the LSB needs to be redefined as

$$\Delta = \frac{4T_{clk}}{2^N}. \tag{4.1}$$

And in the same manner as (2.3), we can show that the temporal noise energy due to quantization error is

$$\begin{aligned} T_{NZ,q}^2 &= \frac{\Delta^2}{12} \\ &= \left(\frac{4}{3}\right)\frac{T_{clk}^2}{2^{2N}}. \end{aligned} \tag{4.2}$$

The quantization error's energy will be a contributing factor when calculating the total noise energy in Section 4.2.3.

4.2.3 Thermal Noise Sources

Several sources of thermal noise contribute to the overall limitation on our precision. We will primarily focus on the following dominant noise sources: The comparator's pre-amplifier noise, the comparator's latch noise, white noise due to current integra-

tion, and the integrating capacitor's reset noise. For the following discussion, we model the MOSFET's current noise as

$$I_{NZ}^2(f) = 4\gamma kT g_{mi}\Delta f \tag{4.3}$$

where k is Boltzmann's constant, T is absolute temperature, g_{mi} is the transconductance of the MOSFET Mi, Δf is the bandwidth, and $\gamma = 2/3$ for above-threshold operation and $\gamma = 1/2\kappa$ for sub-threshold operation. The κ coefficient is from the "κ approximation" described in [41]. For all practical purposes, we are using the same "κ approximation" for both sub and above-threshold operation. If we define I_{DS} as the DC bias current through MOSFET Mi, the transconductance is defined as the following:

$$g_{mi} = \begin{cases} \frac{\kappa I_{DS}}{\frac{kT}{q}} & \text{sub-threshold} \\ \\ \sqrt{2\kappa\mu C_{ox}\frac{W_i}{L_i}I_{DS}} & \text{above-threshold} \end{cases} \tag{4.4}$$

The $1/f$ noise in our transistors contributes negligibly to our design because, like comparator delay and offset, it behaves like a nearly constant offset voltage across successive clock cycles and is cancelled. As long as the corner frequency is less than a couple clock cycles (order of 1μs), flicker noise will appear as a constant offset across clock cycles and cancel like comparator delay.

The Pre-amplifier

The first stage of the comparator is a resistively loaded differential pair that provides low-gain and kick-back isolation for the input from the second stage of the comparator. Figures 4-2 and 4-3 show the two stages of the comparator. Using (4.3), we can

calculate that the total input-referred noise for the pre-amplifier is

$$V_{NZ,pa}^2 \approx 2\left(\frac{5}{3}\right)\frac{4kT\left(\gamma g_{m3,4}+1/R_p\right)}{g_{m3,4}^2}\left(\frac{1}{2\pi}\right)\left(\frac{\pi}{2}\right)\left(\frac{1}{R_pC_p}\right)$$

$$\approx \frac{10kT\left(\gamma g_{m3,4}+1/R_p\right)}{3g_{m3,4}^2R_pC_p} \tag{4.5}$$

assuming that R_p is much smaller than the output resistance of the input PMOS transistor. The factors of $1/2\pi$ and $\pi/2$ arise from integrating noise per unit bandwidth over a single-pole low-pass bandwidth of $1/(R_pC_p)$. The factor of 2 comes from the fact that both halves of the pre-amplifier are active during the comparison and therefore contribute input-referred noise to both the negative and positive inputs. The 5/3 coefficient arises from the geometric sum of the diminishing contribution of successive stages to the overall precision of the converter (See Appendix D.1). We can obtain the equivalent temporal jitter's energy, $T_{NZ,pa}^2$, by dividing (4.5) with the input slope squared, as shown in (4.6).

$$T_{NZ,pa}^2 \approx \frac{10kT\left(\gamma g_{m3,4}+1/R_p\right)}{3g_{m3,4}^2R_pC_p}\left(\frac{C_i}{I_{REF}}\right)^2 \tag{4.6}$$

A typical bias current is on the order of 5μA, and therefore, we can assume that all transistors are operating above-threshold. Using the transconductance equation for an above-threshold MOSFET from (4.4), we can rewrite (4.6) as

$$T_{NZ,pa}^2 \approx \frac{10kT\left(\gamma\sqrt{2\kappa\mu C_{ox}\frac{W}{L}I_{BIAS}}+1/R_p\right)}{3\kappa\mu C_{ox}\frac{W}{L}I_{BIAS}R_pC_p}\left(\frac{C_i}{I_{REF}}\right)^2. \tag{4.7}$$

The Gain/Latch Stage

The gain/latch stage consists of a nine transistor, wide-output-swing operational transconductance amplifier with a positive feedback latch, as shown in Figure 4-3. As the output node D_{out} increases above the threshold voltage of transistor $M17$, this NMOS briefly pushes a large, increasing current into C_L via the current mirror formed by transistors $M13$ and $M14$. The total output voltage noise can be shown

to be

$$V_{NZ,gl}^2 \approx \left(\frac{5}{3}\right) \frac{\gamma kT \left(g_{m5} + g_{m9} + g_{m11} + g_{m13} + g_{m14}\right)}{C_L g_{o14}} \tag{4.8}$$

where g_{o14} is the output conductance of transistor $M14$. In the operating region of interest, only one-half of the OTA is contributing noise. Specifically, when the output is slew-limited, transistors $M5$, $M9$, $M11$, $M13$, and $M14$, which have significant currents flowing through them, contribute noise to the output. Transistors $M6$ and $M7$ function as cascode transistors, self-shunt most of their current source noise, and contribute negligibly to the overall output noise: $M15$, which functions as a switch, also contributes negligible noise. The 5/3 coefficient is required to reflect the contributions of the successive conversion stages as previously discussed (See Appendix D.1). In order to obtain the temporal jitter's energy, we divide (4.8) by the latch's output slew rate squared, as shown in (4.9).

$$T_{NZ,gl}^2 \approx \left(\frac{5}{3}\right) \frac{\gamma kT \left(g_{m5} + 2g_{m9,11} + 2g_{m13,14}\right)}{C_L g_{o14}} \left(\frac{C_L}{I_{BIAS2}}\right)^2 \tag{4.9}$$

Note that we can lump the mirror transistors together, i.e., $M9$ with $M11$ and $M13$ and $M14$. Once again, all of the transistors involved in the noise analysis are operating above-threshold, and as a result, we can use (4.4) to elaborate on (4.9). The output conductance of transistor Mi is termed

$$g_{oi} = \lambda_i I_{DS}, \tag{4.10}$$

where λ_i, the channel length modulation coefficient, is defined as

$$\frac{1}{L_i} \frac{\partial \chi}{\partial V_{DS}}, \tag{4.11}$$

and χ is the reduction in the channel length due to V_{DS}. Then we can rewrite (4.9) as

$$T_{NZ,gl}^2 \approx \frac{10kTC_L^2 \sqrt{2\kappa\mu C_{ox}} \left(\sqrt{\frac{W_5}{L_5}} + 2\sqrt{\frac{W_{9,11}}{L_{9,11}}} + 2\sqrt{\frac{W_{13,14}}{L_{13,14}}}\right)}{9\lambda_{14} C_L \left(I_{BIAS2}\right)^{\frac{3}{2}}}. \tag{4.12}$$

Integration Noise

The final source of noise in our converter is due to the thermal noise in the integration currents. Given an upper bound on our full-scale voltage, we can collect only a certain amount of electrons, regardless of capacitor size since we are integrating our input current for a finite amount of time. Note that integrating a current for a fixed amount of time is analogous to filtering with a *sinc* filter in the frequency domain [42]. Therefore, we need to calculate the bandwidth, Δf, for a *sinc* filter. If we integrate for a fixed time of T_i, then the equivalent bandwidth for our noise model is $1/(2T_i)$. The derivation assumes that the transconductance term is independent of frequency, which is a good approximation for input signals whose frequencies are well below T_i.

Using this bandwidth estimate and our current noise model in (4.3), we can show that the total voltage noise squared due to the above-threshold integrating currents is

$$V_{NZ,int}^2 \approx \frac{4kT}{3C_i^2} T_{clk} \left(g_{min} + \frac{I_{IN}}{I_{REF}} g_{mref} + \frac{3}{2} g_{mref} \right). \tag{4.13}$$

Note that (4.13) accounts for the two MSB integration cycles with I_{IN} and I_{REF} and the three integration cycles with I_{REF} during each successive stage that contribute in a geometrically diminishing manner (See Appendix D.2). Accordingly, the temporal jitter's energy can be derived by dividing (4.13) by the square of the integration rate, $(I_{REF}/C_{ref})^2$. Specifically,

$$T_{NZ,int}^2 \approx \frac{4kT}{3} T_{clk} \left(\frac{g_{min}}{I_{REF}^2} + \frac{I_{IN} g_{mref}}{I_{REF}^3} + \frac{3 g_{mref}}{2 I_{REF}^2} \right). \tag{4.14}$$

If the input current is at its full-scale value of $4I_{REF}$, then $g_{min} = \sqrt{4} g_{mref}$ (assuming both $Mref$ and Min have the same W/L ratio) and (4.14) reduces to the following:

$$\begin{aligned} T_{NZ,int}^2 &\approx \frac{10kT g_{mref}}{I_{REF}^2} T_{clk} \\ &\approx \frac{10kT \sqrt{2\kappa\mu C_{ox} \frac{W}{L}}}{(I_{REF})^{\frac{3}{2}}} T_{clk} \end{aligned} \tag{4.15}$$

The detailed derivations of (4.14) and (4.15) are shown in Appendix D.2.

Reset Noise

After every voltage comparison in our algorithm, switch transistors reset either one or both of the integrating capacitors to zero volts. The equivalent circuit diagram is shown in 4-10. Like thermal noise in CMOS image sensors, this reset phase introduces uncertainty on the reset voltage of the capacitors [43]. If we assume that the reset phase is much longer than the settling time of the reset transistor, the noise calculation is rather straight-forward [43]. In our converter, the comparator output's pulse-width determines the duration of the reset phase, which can be varied to satisfy this approximation. The resulting voltage noise for a single reset phase is then given by

$$\sqrt{\frac{kT}{C_i}}. \tag{4.16}$$

In a manner similar to the integration and comparator noise calculations, we need to account for the multiple reset phases throughout the conversion cycle. We perform two reset operations during the MSB phase and three reset operations during each successive phase, and thus, (4.16) needs to be scaled accordingly. At the beginning of the MSB phase, there are two instances of reset noise since both capacitors have been reset to zero prior to the conversion process. The temporal noise due to each reset cycle is derived from scaling (4.16) by C_i/I_{REF}. During each successive phase, the capacitors are reset three times, and consequently, contribute noise in a way identical to the successive integration cycles illustrated in Appendix D.2. If we assume that each reset noise contribution is statistically independent, then the total equivalent temporal noise energy, $T^2_{NZ,re}$, is defined as

$$\frac{kT}{C_i}\left[\left(\frac{C_i}{I_{REF}}\right)^2+\left(\frac{C_i}{I_{REF}}\right)^2+3\left(\frac{C_i}{I_{REF}}\right)^2\right]. \tag{4.17}$$

Thus,

$$T^2_{NZ,re} \approx \frac{5kTC_i}{I^2_{REF}}. \tag{4.18}$$

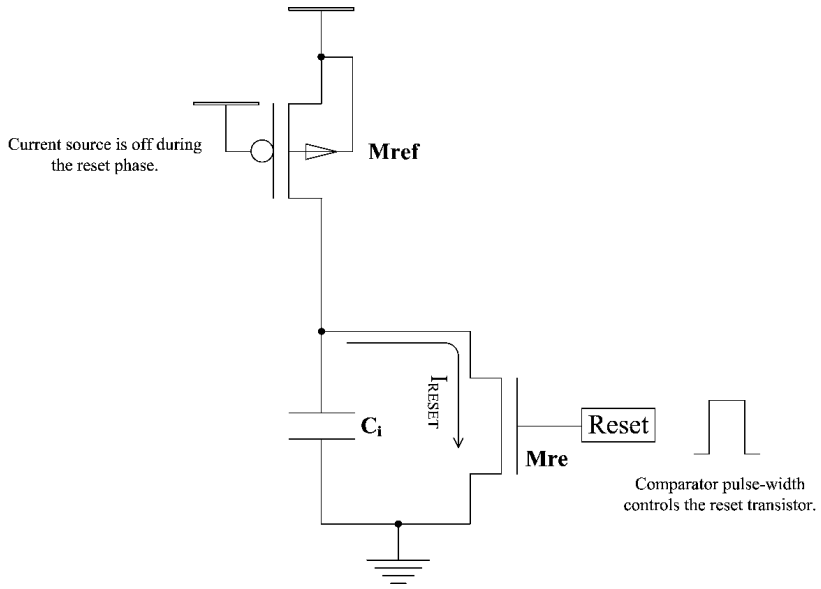

Figure 4-10: The equivalent circuit diagram of the reset phase of the converter is shown. The noise approximation is proportional to $\sqrt{kT/C_i}$.

Total Noise

In order to calculate the overall precision of the converter as defined by the quantization error and the thermal noise of the pre-amplifier, gain/latch stage, the integrating currents, and reset phase, we sum the energy of all the temporal noises and compare it to the energy of the full-scale signal, $4T_{clk}$. To calculate the SNR of our converter, we must assume that the input signal is a full-scale sinusoid at a frequency below the Nyquist-rate, as shown in (4.19).

$$2T_{clk}\sin(\omega t) \quad \text{where } \omega \leq 2\pi f_{samp} \tag{4.19}$$

We can then compare the energy of this full-scale input signal to the sum of the energy of all the noise sources to obtain the SNR. The energy of (4.19) is simply

$$\int_0^{\frac{2\pi}{\omega}} [2T_{clk}\sin(\omega t)]^2 dt = 2T_{clk}^2. \tag{4.20}$$

Therefore, the maximum SNR of our system is defined as

$$10log_{10}\frac{2T_{clk}^2}{T_{NZ,q}^2+T_{NZ,pa}^2+T_{NZ,gl}^2+T_{NZ,int}^2+T_{NZ,re}^2} \tag{4.21}$$

The absolute and relative noise contribution from each source is summarized in Table 4.1. For the integration noise, we evaluated the worst-case scenario where the input current is at its full-scale value of $4I_{REF}$. The band-limiting capacitance in the pre-amplifier, C_p, is dominated by the poly resistor's parasitic capacitance to substrate, which we calculated to be approximately $200f$F. Figure 4-11 shows our model's prediction on the SNR upper bounds set by each noise source for various current levels. Our converter was designed for 13 bits of quantization, and as a result, the quantization error places an upper limit of 80dB. For $I_{REF} = 10\mu$A and $I_{BIAS2} = 4\mu$A, the pre-amplifier noise dominates all other noise sources for most values of I_{BIAS}.

Table 4.1: Summary of noise contributions for high performance ADC (Theory)

	Quantization	Pre-Amplifier	Gain/Latch	Integration	Reset
Parameters	$T_{clk} = 0.5\mu$sec	$I_{BIAS} = 5\mu$A $R_p = 300k\Omega$ $C_p = 200f$F	$I_{BIAS2} = 4\mu$A $C_L = 1p$F	$I_{REF} = 10\mu$A $I_{IN} = 40\mu$A $C_i = 22p$F	
Temporal Noise	$0.071n$sec	$0.215n$sec	$0.174n$sec	$0.138n$sec	$0.068n$sec
Equivalent SNR	80.0dB	70.3dB	72.2dB	74.2dB	80.35dB
ENOB	13	11.39	11.70	12.03	13.05

Effective Number of Bits

As we did in Section 4.2.2, let us assume for a moment that quantization error is the only source of noise in our converter. Then we can compute the maximum achievable SNR solely as a function of N. In this special case, the SNR is

$$log_{10}\left(\frac{2T_{clk}^2}{\Delta^2/12}\right). \tag{4.22}$$

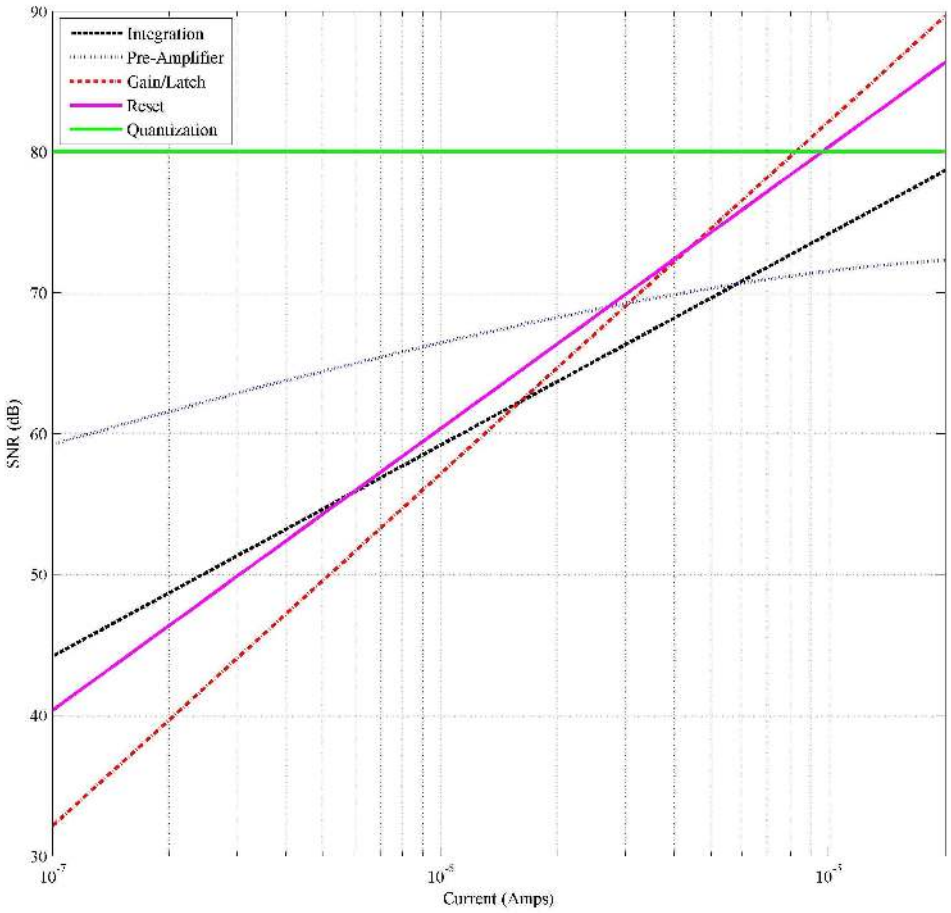

Figure 4-11: For the noise models presented in Section 4.2.3, we swept I_{REF} for the integration and reset noises, I_{BIAS} for the pre-amplifier's noise, and I_{BIAS2} for the gain/latch noise and calculated their respective SNR limitations. For $N = 13$ bits, the quantization error places an upper bound of 80dB. For $I_{REF} = 10\mu$A (74.2dB) and $I_{BIAS2} = 4\mu$A (72.2dB), the pre-amplifier noise dominates the total noise in our converter for almost all I_{BIAS} levels.

Combining (4.1) with (4.22), we can show that

$$SNR(dB) \approx 6.02N + 1.76. \tag{4.23}$$

Inversely, we can compute the Effective Number of Bits (ENOB) given a measurement of SNR, as shown in (4.24).

$$ENOB = \frac{SNR(dB) - 1.76}{6.02} \tag{4.24}$$

In essence, the ENOB formula treats all sources of noise as a single source of quantization error. This is a useful tool in estimating the dynamic precision of a converter.

4.3 Experimental Results

We fabricated our converter in the MOSIS TSMC 0.35μm mixed-signal process. We explored different integrating capacitor sizes as well as different comparator and state-machine topologies. The converter that proved to be the optimal configuration consumes approximately $0.45mm^2$ and is shown in 4-12. The design was not optimized for area, but improvements to layout of the digital circuitry and capacitors can potentially cut the area in half. The optimal configuration included the $1 + \varepsilon$ modification for the entire conversion cycle.

4.3.1 Static and Dynamic Performance Results

The overall experimental results are summarized in Table 4.2. For our static measurements, the input was swept from 1μA to 40μA with the reference current held at 10μA. The limited dynamic range can be attributed to an intentional offset current of 1μA in the input stage. Figure 4-13 shows the Integral Nonlinearity (INL) and Differential Nonlinearity (DNL) with respect to 12 bits. For our dynamic measurements, we used 4000 samples of a 1kHz sinusoidal input sampled at 31.25kHz to perform a Fast-Fourier Transform (FFT). The resulting Power Spectral Density (PSD), which

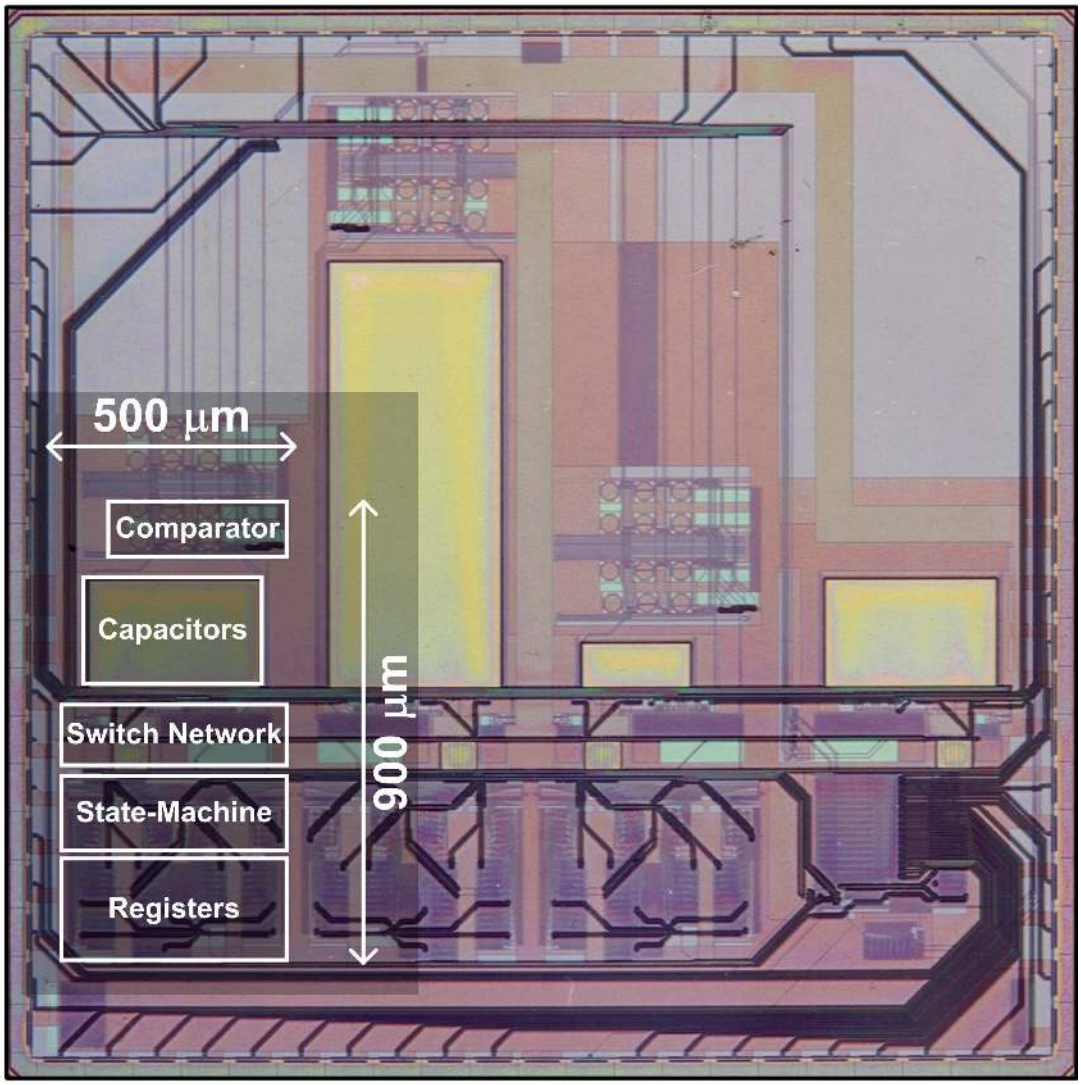

Figure 4-12: The chip was fabricated on a MOSIS TSMC 0.35μm process. The area of interest is 500μm by 900μm.

is shown in Figure 4-14, shows an SNR of approximately 69dB. In addition, Figure 4-15 shows the experimental measurements of the ENOB for various input frequencies ranging from $1k$Hz to $15k$Hz.

These experimental results yield an efficiency of $1.17p$J per quantization level. The second harmonic limited our Spurious-Free Dynamic Range (SFDR) to 63dB due to dynamic limitations of the off-chip V-to-I converter used in our implementation. Figure 4-16 shows the PSD of a $1k$Hz sinusoid signal that was generated with the experimentally measured V-to-I transfer curve of our off-chip V-to-I converter and shows that the second order harmonic peaks at approximately 63dB.[1] For future iterations, we plan on implementing a wide-linear-range transconductance amplifier as our V-to-I converter using techniques described in [44] to achieve sufficient bandwidth and dynamic range while consuming minimal power. Using (4.24), we calculate that for an SNR of 69dB, the ENOB is approximately 11 bits.

4.3.2 Validating Our Noise Models

Using (4.21) with the parameters from Section 4.1, our noise model predicts an SNR of 66.8dB, which is close to our experimental measurement of 69dB. However, we wanted to take a closer look at our noise models for the pre-amplifier and gain/latch stage. Therefore, we varied both the pre-amplifier's bias current, I_{BIAS}, and the gain/latch stage's bias current, I_{BIAS2}, and measured the SNR for a full-scale sinusoidal input current. Figure 4-17 compares our SNR noise model with the experimental measurements over the I_{BIAS} and I_{BIAS2} matrix. Our model appears to follow the behavior of the converter over the entire spectrum. Note that the SNR noise model includes a fixed integration and reset noise source for $I_{REF} = 10\mu$A and quantization error for $N = 13$ bits.

Figure 4-18 shows the experimental SNR measurements for different values of I_{BIAS} with I_{BIAS2} fixed at 4μA. The experimental values follow our noise model over almost 2 orders of magnitude. Conversely, Figure 4-19 shows the experimental

[1]The higher noise floor in the V-to-I PSD is attributed to the limitations of the device used to measure the V-to-I transfer curve and does not reflect the noise of the system.

Table 4.2: Summary of experimental results for high performance ADC

Performance Metric	Value
Technology	MOSIS TSMC 0.35μm
Voltage Supply	
Analog	3 Volts
Digital	3 Volts
Reference Current	10μA
Input Current Range	1μA to 40μA
Input Offset Current	1μA
Integrating Capacitor	22pF
T_{clk}	0.5μsec
Sampling Rate	31.25kHz
INL	$\leq \pm$1.4 LSB [12 bits] typical
DNL	$\leq \pm$0.6 LSB [12 bits] typical
SNR	69dB
SFDR	63dB
ENOB	11 bits
Power Dissipation	
Analog	60μW
Digital	15μW
Thermal Noise-Limited	
Energy per Quantization Level	1.17pJ
Active Area	0.45mm^2

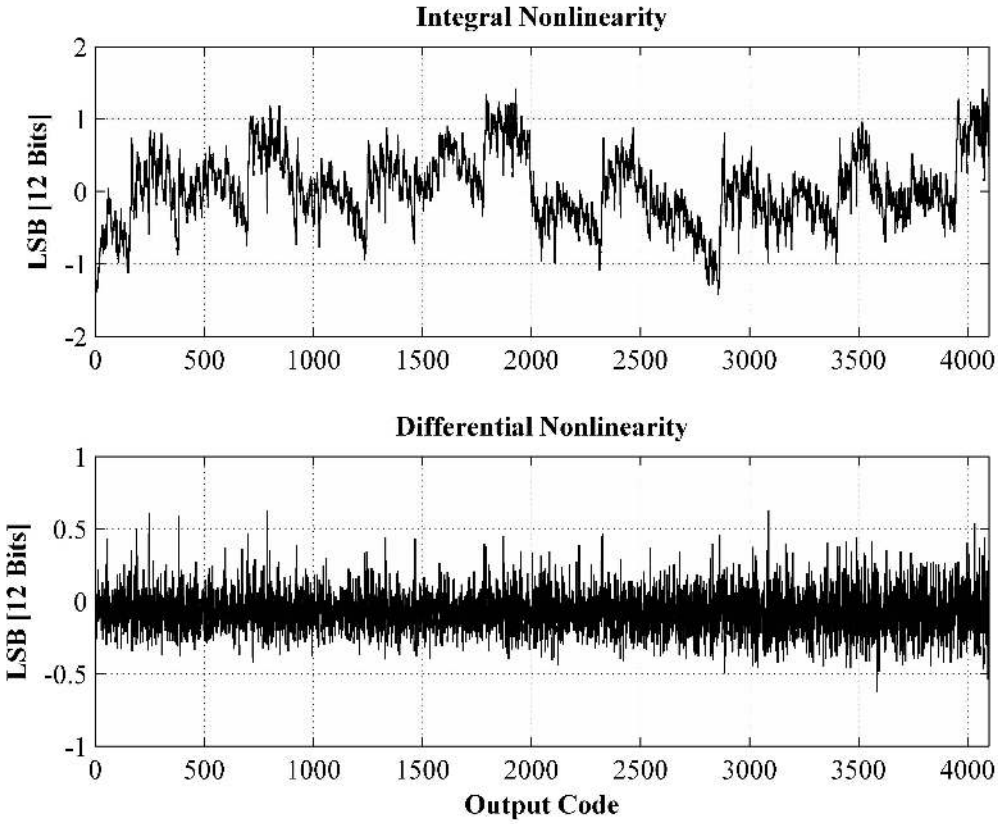

Figure 4-13: The INL was obtained using a least-squared-error approximation.

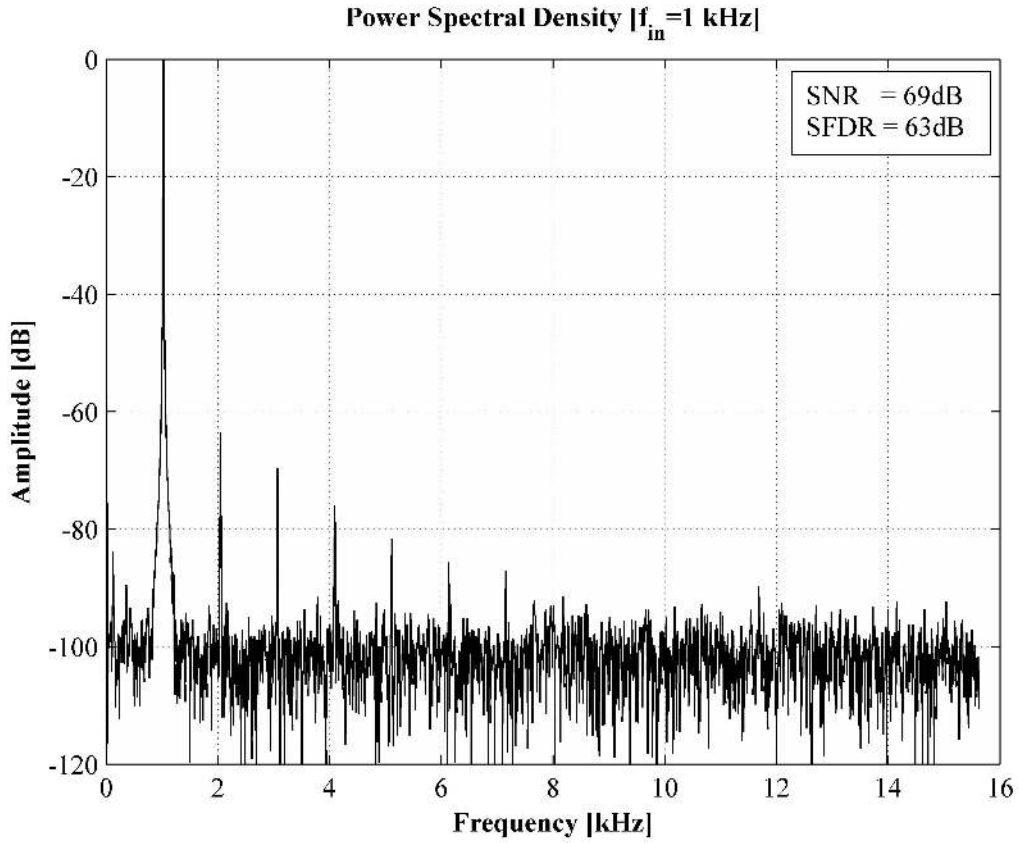

Figure 4-14: For a full-scale sinusoidal input at $1k$Hz, the SNR is 69dB while the second harmonic limits the SFDR to 63dB.

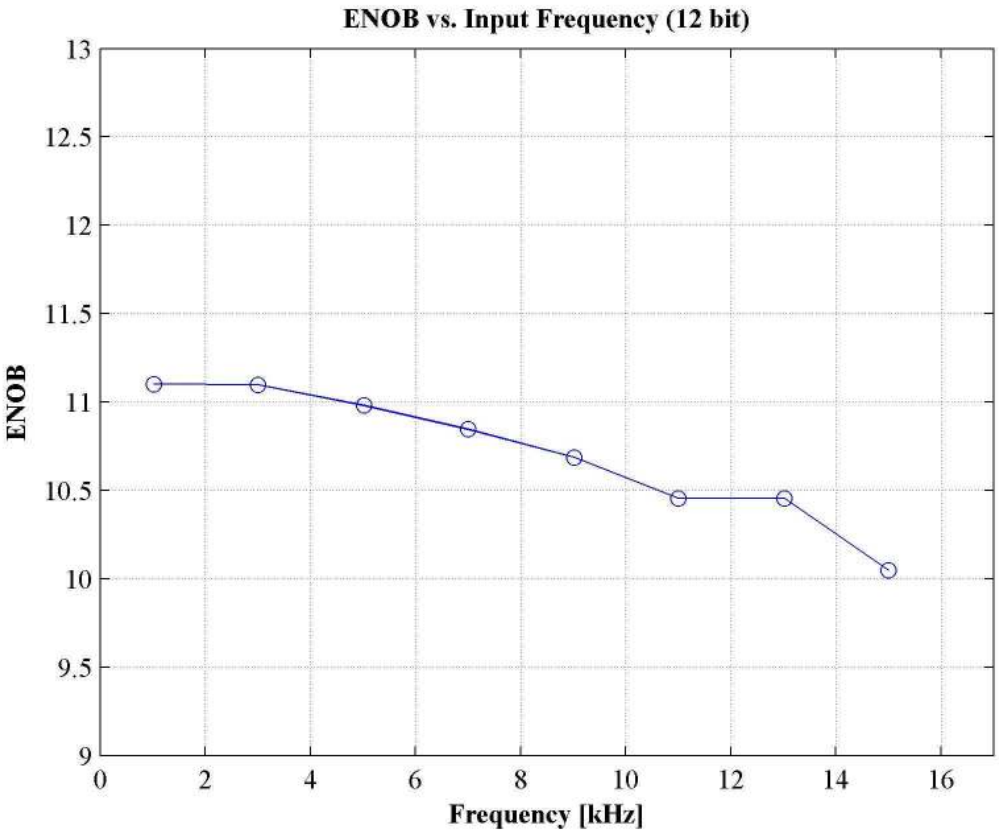

Figure 4-15: A full-scale sinusoidal input was swept from 1kHz to 15kHz, and the resulting ENOBs are shown.

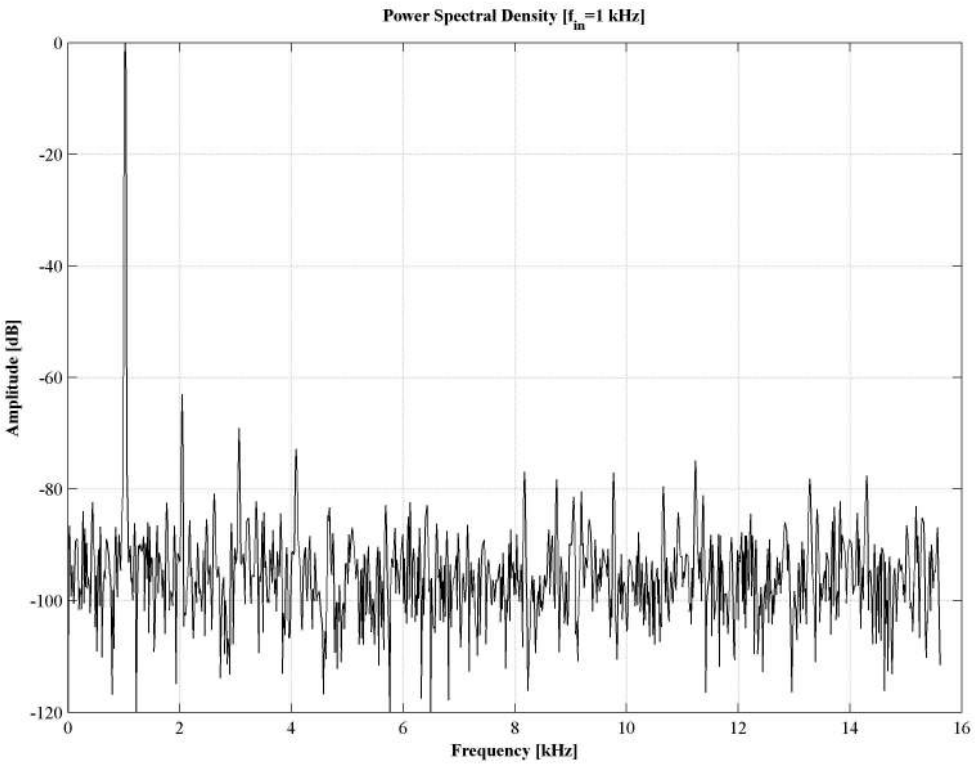

Figure 4-16: A full-scale sinusoidal input at 1kHz was generated from an experimentally measured V-to-I transfer curve. The resulting PSD shows that the second harmonic peaks at approximately 63dB.

SNR measurements for different values of I_{BIAS2} with I_{BIAS} fixed at 5μA. Although our model is not as accurate for the gain/latch stage, the overall behavior of the experimental data appears to match our model over 1.5 orders of magnitude. The noise model flattens out around 67dB due to the noise floor set by the pre-amplifier.

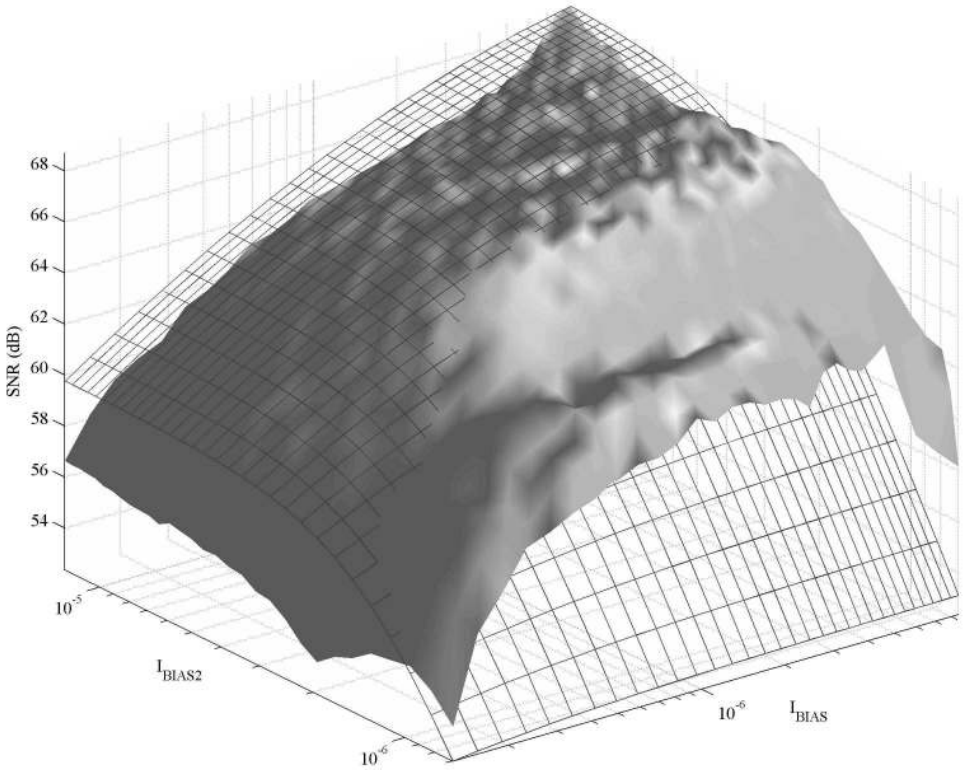

Figure 4-17: The experimental SNR measurements (grid-plane) follows the behavior of our SNR noise model (solid-plane) for the matrix of I_{BIAS} and I_{BIAS2} values. The model includes a fixed integration noise source for $I_{REF} = 10\mu$A, quantization error for $N = 13$ bits, and reset noise.

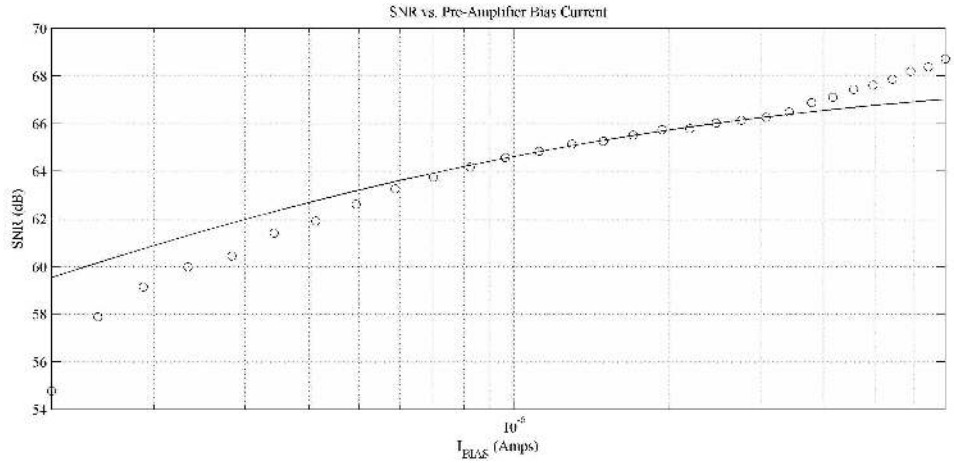

Figure 4-18: The experimental SNR measurements (markers) closely follow our theoretical model's prediction (solid line). For our noise model, we included a fixed gain/latch stage noise source modeled with $I_{BIAS2} = 4\mu$A and a fixed integration noise modeled with $I_{REF} = 10\mu$A and $I_{IN} = 40\mu$A.

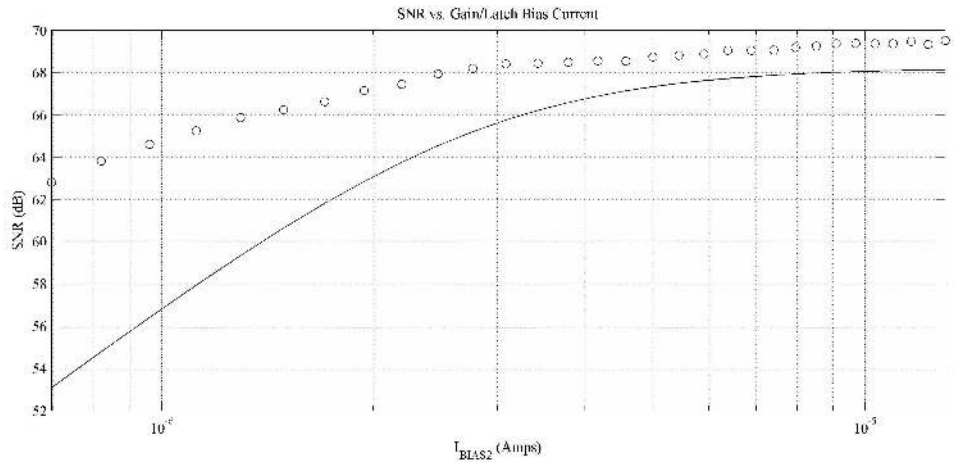

Figure 4-19: The experimental SNR measurements (markers) roughly follow our theoretical model's prediction (solid line). For our noise model, we included a fixed pre-amplifier noise source modeled with $I_{BIAS2} = 5\mu$A and a fixed integration noise modeled with $I_{REF} = 10\mu$A and $I_{IN} = 40\mu$A. The knee in the curve represents the SNR upper-bound imposed by the pre-amplifier's noise.

Chapter 5

A Sub-Microwatt Audio Converter

In Chapter 4, we explored a 12 bit, 33kHz, 75μW, thermal-noise-limited implementation of our algorithm. We were able to demonstrate energy-efficient operation of a 69dB SNR A/D converter. Unfortunately, we paid a significant price in power consumption to achieve high SNR performance. The pre-amplifier and gain/latch stage required significant bias currents, and the reference current, along with the integrating capacitors, had to be fairly large. Furthermore, we had to pay close attention to the timings of the state-machine transitions and the control signals that were passed onto the analog switch network. And while an energy-efficiency of 1.12pJ per quantization-level is extremely low, there exist applications that would benefit from an even more energy-efficient converter. Systems such as wireless sensor networks and wireless medical devices require low precision, moderate speed converters that consume very little power. As a result, there has been more focus lately on converters that meet this criteria. Section 5.4 presents a list of recently published converters that explore this spectrum.

In this chapter, we present a sub-microwatt 8 bit audio converter where we attempted to maximize its energy-efficiency. In effect, we were able to reduce power consumption drastically by eliminating many elements that were required for the high precision converter. Furthermore, we were able to reduce the bias and integrating currents and capacitors significantly due to the lower precision constraint.

5.1 Design and Implementation

We tailored our design of the converter around the MOSIS TSMC 0.18μm mixed-signal process. This technology has low device threshold voltages of approximately 0.50V for both PMOS and NMOS devices. Since we were targeting a much lower precision of 8 bits, we did not require as much dynamic range in our analog components. Therefore, we were able to reduce the analog voltage supply to 1.2V while maintaining all of our devices in saturation. Furthermore, the digital components required less stringent timing specifications. Therefore, the gate delays incurred by reducing the digital voltage supply to 0.75V were insignificant.

Two analog components of the 12 bit converter consumed the most significant amount of power (the comparator and the integrating current source), and we focused on reducing these.

5.1.1 Asynchronous Comparator

In order to save additional power, we used the gain/latch stage presented in Section 4.1.1 as a single-stage asynchronous comparator. Shown in Figure 5-1, the comparator has a few minor modifications from the original gain/latch stage. We eliminated the switch transistor, $M15$, from the original design. With a very small sub-threshold bias current, the short circuit path created by the disable transistor, $M12$, and current mirror transistor, $M10$, can be ignored. Therefore, the power-savings from $M15$ disabling this short-circuit path is insignificant, and $M15$ is no longer needed. We also eliminated the explicit 1pF capacitor at the output node, D_{out}. Instead, the load capacitor, C_L, is defined as the input capacitance to the inverter at the front-end of the pulse-width control circuit. In addition, the wide-output current mirror stage is tied to the digital voltage supply, DV_{DD}, to minimize the power consumption during the latching, high-current phase of the comparison. The same pulse-width control mechanism shown in Figure 4-4 was used with slightly different device sizes to accommodate for the smaller technology.

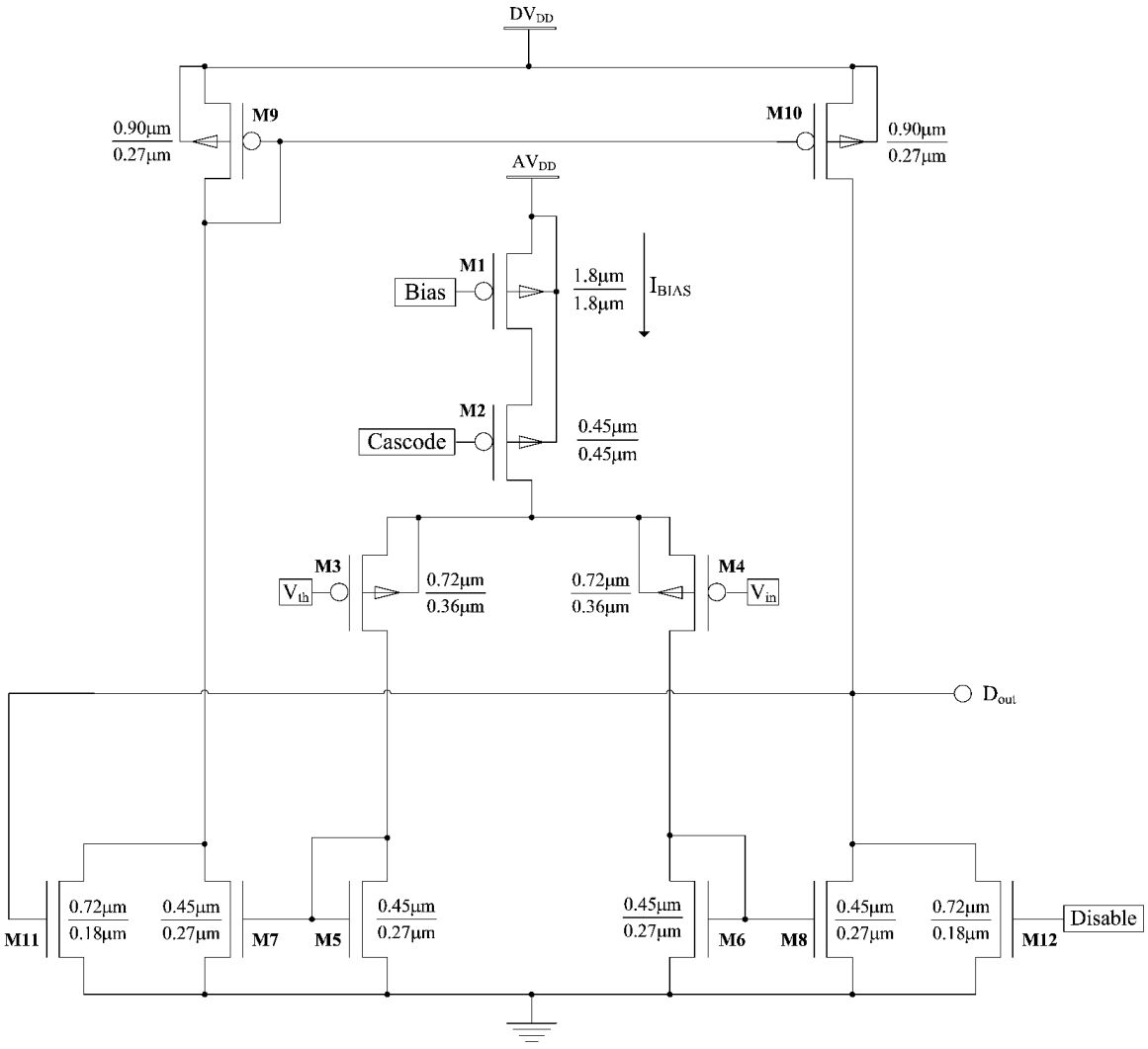

Figure 5-1: The low-power ADC uses a wide-output swing OTA with positive feedback and staggered rails as a single-stage comparator.

5.1.2 State-Machine

The number of iterative states in the state-machine was reduced from 6 to 3 states by eliminating the states required for the capacitor mismatch reduction method described in Section C. The tracking of the positive-index and negative-index phases was done in the registers. In addition, compared to the high precision design, we drastically reduced the number of gates in a control signal's path by eliminating buffers that were used to match their timings. However, we still used the matched-logic switching technique presented in Section 4.1.4. Furthermore, we used a modified version of the D flip-flop presented in [45] and is shown in Figure 5-2. An AND gate was added to ensure true edge-triggered behavior. This true single-phase flip-flop has 50 percent fewer gates than then the one presented in Section 4.1.4 and as a result, consumes significantly less power.

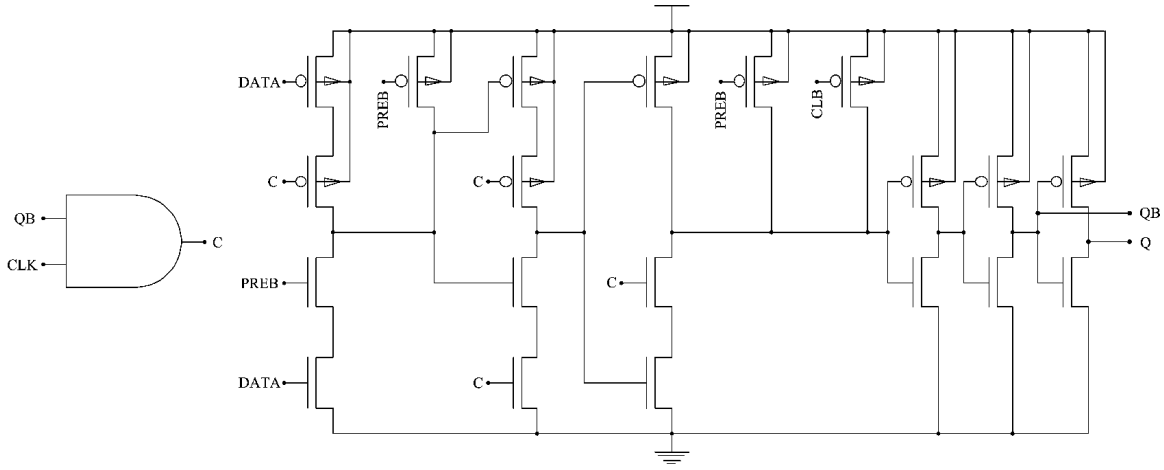

Figure 5-2: A low-power true single-phase edge-triggered D flip-flop was used both in the state-machine and the register. This flip-flop has a maximum number of 4 stacked transistors as opposed to 5 in the original flip-flop. In addition, it has 50 percent fewer gates.

5.2 Noise Analysis

The noise analysis for our 8 bit converter is similar to the analysis done for our high performance converter in Section 4.2. We need to quantify the noise contributions from the comparator, integrating currents, quantization error, and reset phase. For

8 bits of precision, jitter in the digital control signals induced by power-supply noise should not be a concern, even for power supply voltages below 1V. Also, we will show that reset noise does not factor into the overall precision, even for a low integration capacitor size of $500fF$. Instead, if designed properly, the overall precision of the converter should be determined by the white noise from the integration currents and/or the thermal noise of the comparator.

5.2.1 Quantization Error

The converter was designed for 8 bits of precision with an extra bit reserved for error correction. Therefore, using (4.2), we can compute the energy of the quantization error for 9 bits to be $(2.26n\text{sec})^2$.

5.2.2 Comparator Noise

Although the single stage comparator described in Section 5.1.1 and shown in Figure 5-1 is virtually identical to the gain/latch stage in Section 4.1.1, it requires a slightly different noise analysis. The inverter tied to the output of the comparator has a very low threshold around 0.4V due to the low digital power supply of 0.75V. Since the positive feedback transistor, $M11$, has a threshold voltage around 0.5V, the comparator will trigger the inverter before transistor $M11$ has an opportunity to inject significant current to the output node. Consequently, we can analyze the comparator noise in a manner similar to the pre-amplifier's noise analysis in Section 4.2.3. The positive feedback transistor simply reduces the short circuit current through the output inverter by increasing the speed at which the output node transitions through the inverter's linear region.

We will first need to calculate the total input-referred noise, $V_{NZ,comp}$, in order to derive the equivalent temporal noise. Using the transistor noise model from (4.3), we

find that

$$V_{NZ,comp}^2 \approx 2\left(\frac{5}{3}\right)\frac{4kT\gamma\left(g_{m3,4}+g_{m5,6}+g_{m7,8}+g_{m9,10}\right)}{g_{m3,4}^2}\left(\frac{1}{2\pi}\right)\left(\frac{\pi}{2}\right)\left(\frac{g_{o8}+g_{o10}}{C_L}\right)$$

$$\approx \frac{10kT\gamma\left(g_{m3,4}+g_{m5,6}+g_{m7,8}+g_{m9,10}\right)\left(g_{o8}+g_{o10}\right)}{3g_{m3,4}^2 C_L}. \tag{5.1}$$

Once again, the factors of $1/2\pi$ and $\pi/2$ account for a low-pass single-pole integration, the factor of 2 accounts for both halves of the comparator, and the 5/3 coefficient arises from the geometric sum of the diminishing contribution of successive stages. We can derive the equivalent temporal noise energy, $T_{NZ,comp}^2$, by dividing (5.1) by the square of the input slope, as shown in (5.2).

$$T_{NZ,comp}^2 = V_{NZ,comp}^2\left(\frac{C_i}{I_{REF}}\right)^2$$

$$= \frac{10kT\gamma\left(g_{m3,4}+2g_{m5,6,7,8}+g_{m9,10}\right)\left(g_{o8}+g_{o10}\right)C_i^2}{3g_{m3,4}^2 C_L I_{REF}^2}. \tag{5.2}$$

Note that we can lump the mirror transistors together, i.e., $M5$ and $M6$ with $M7$ and $M8$. If we assume that all transistors are operating in sub-threshold, all of the transconductance terms are equal, and (5.2) can be reduced even further to

$$\frac{20kT\left(g_{o8}+g_{o10}\right)C_i^2}{3\kappa g_m C_L I_{REF}^2}. \tag{5.3}$$

Using both (4.4) and (4.10) with a DC bias current of $I_{BIAS}/2$, we can show that

$$T_{NZ,comp}^2 = \frac{40(kT)^2(\lambda_8+\lambda_{10})C_i^2}{3\kappa^2 q C_L I_{REF}^2}, \tag{5.4}$$

which is independent of I_{BIAS}.

5.2.3 Integration Noise

The integration noise model in Section 4.2.3 assumes that the MOSFETs in the integrating currents paths are operating well above threshold. However, in our low-power converter, both the input and reference current sources are operating in sub-threshold. Therefore, the equation for the total voltage noise energy, $V^2_{NZ,int}$, needs to be revised to the following:

$$V^2_{NZ,int} = \frac{kT}{\kappa C_i^2} T_{clk} \left(g_{min} + \frac{I_{IN}}{I_{REF}} g_{mref} + \frac{3}{2} g_{mref} \right). \tag{5.5}$$

Then the equivalent temporal noise energy, $T^2_{NZ,int}$, is

$$\frac{kT}{\kappa} T_{clk} \left(\frac{g_{min}}{I^2_{REF}} + \frac{I_{IN} g_{mref}}{I^3_{REF}} + \frac{3 g_{mref}}{2 I^2_{REF}} \right). \tag{5.6}$$

If we assume that the input is at the full-scale value of $4I_{REF}$, then $g_{min} = 4g_{ref}$ and (5.6) reduces to

$$T^2_{NZ,int} = \frac{19kT g_{mref} T_{clk}}{2\kappa I^2_{REF}}. \tag{5.7}$$

And since

$$g_{mref} = \frac{\kappa I_{REF}}{\frac{kT}{q}}, \tag{5.8}$$

(5.7) simplifies to

$$T^2_{NZ,int} = \frac{19qT_{clk}}{2I_{REF}}. \tag{5.9}$$

For $T_{clk} = 1\mu$sec and a reference current of $80n$A, the temporal noise is approximately $6.2n$sec.

5.2.4 Reset Noise

Using the same reset noise formula from Section 4.2.3, we calculate that the temporal noise for an integration current of $80n$A and capacitor of $500f$F is $1.28n$sec. This equates to an SNR of approximately 60.9dB for a 1μsec clock cycle, or an ENOB of 9.82 bits. Reset noise should not be a contributing factor in determining the overall

precision of the 8 bit converter.

5.2.5 Total Noise

If we combine the temporal noise energies from the quantization error, integration currents, and comparator, we can predict the maximum achievable SNR as a function of the different circuit parameters. Combining (5.2) with (5.7), the quantization error, and reset noise, the SNR for a full-scale input of $4T_{clk}$ is bound by

$$SNR_{max} = 10log_{10}\frac{2T_{clk}^2}{T_{NZ,q}^2 + T_{NZ,comp}^2 + T_{NZ,int}^2 + T_{NZ,re}^2}. \tag{5.10}$$

Table 5.1 summarizes the temporal noise contribution from each noise source for the given circuit parameters. Figure 5-3 shows the bounds on the SNR set by each noise source for various current levels. Using (5.10), we calculate that the maximum achievable SNR for the parameters in Table 5.1 is 45.9dB.

Table 5.1: Summary of noise contributions for low-power ADC (Theory)

	Quantization	Comparator	Integration	Reset
Parameters	$T_{clk} = 1\mu$sec	$I_{BIAS} = 100n$A $C_L = 5n$F	$I_{REF} = 80n$A $I_{IN} = 320n$A $C_i = 500f$F	
Temporal Noise	2.26$n$sec	2.51$n$sec	6.20$n$sec	1.28$n$sec
Equivalent SNR	55.9dB	55.0dB	47.2dB	60.9dB
ENOB	9	8.85	7.54	9.82

5.3 Experimental Results

We implemented our sub-microwatt design in the MOSIS TSMC 0.18μm mixed-signal technology. A die photo of the VLSI implementation is shown in Figure 5-4. The layout includes six complete converters where we varied the integrating capacitor sizes and explored different V-to-I topologies. In the figure, the converter of interest

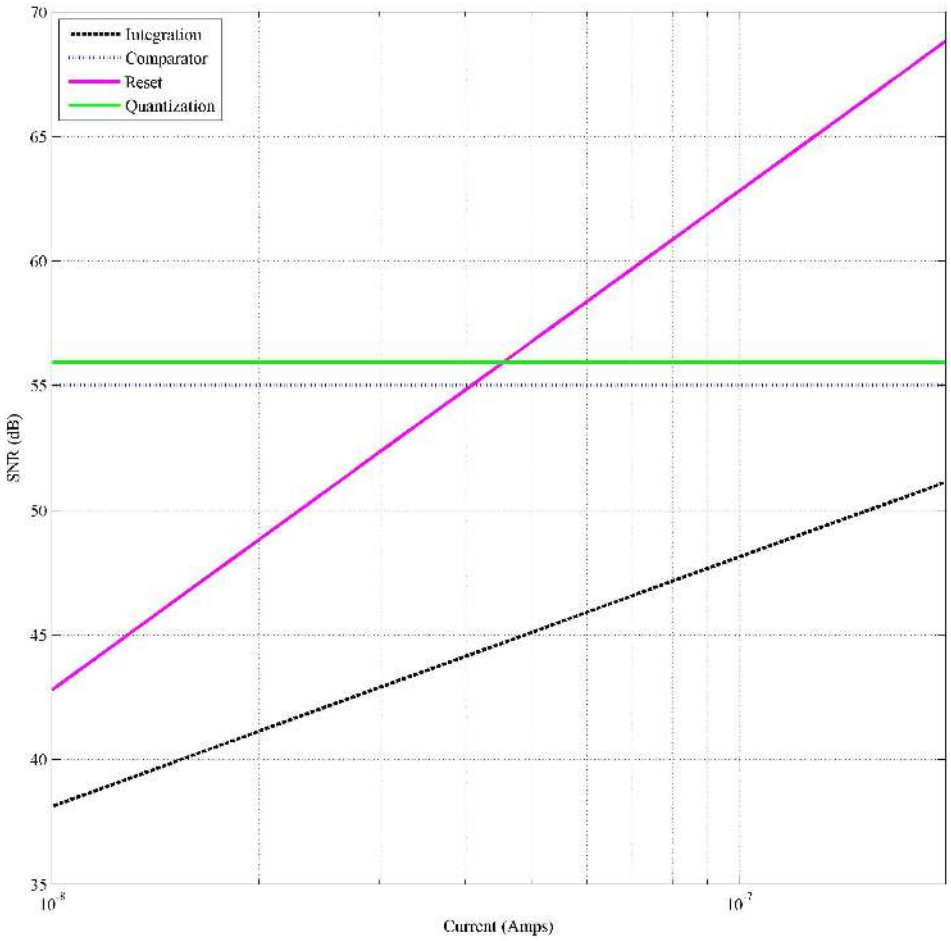

Figure 5-3: Using the sub-threshold integration noise model, we calculated the SNR for various values of I_{REF}. For the comparator noise model in Section 5.2.2, we swept I_{BIAS} and calculated its SNR. For $N = 9$ bits, the quantization error places an upper bound of 55.9dB.

is highlighted with its separate components labeled. All six converters share the same set of registers, and therefore, the layout is not optimized for area. Nevertheless, the converter of interest only consumes an area of $130n$m by $160n$m. A layout for a single converter can be optimized to consume even less area.

The experimental results are summarized in Table 5.2. Compared to the high performance converter presented in Section 4, this converter consumes significantly less power at $960n$W. The state-machine was configured to perform the $1+\varepsilon$ algorithm on the first three bits past the first two MSBs. Consequently, the conversion time averaged to approximately 22 clock cycles. For our static measurements, we swept the input current from $10n$A to $320n$A with a reference current of $80n$A. The limited dynamic range can be attributed to a $10n$A offset current in the input stage. The INL and DNL data with respect to 8 bits are presented in Figure 5-5. The INL is bound by ± 1 LSB's while the DNL is limited to ± 0.8 LSB's. For our dynamic measurements, we sampled a full-scale $3k$Hz sinusoidal input at $45k$Hz to obtain 512 sample points. We then performed an FFT analysis to obtain the PSD presented in Figure 5-6. Using (4.24), the SNR of 46dB translates to an ENOB of 7.4 bits. The SFDR is limited to 51dB by the second harmonic. Furthermore, Figure 5-7 shows the experimental measurements of the ENOB for various input frequencies ranging from $1k$Hz to $20k$Hz.

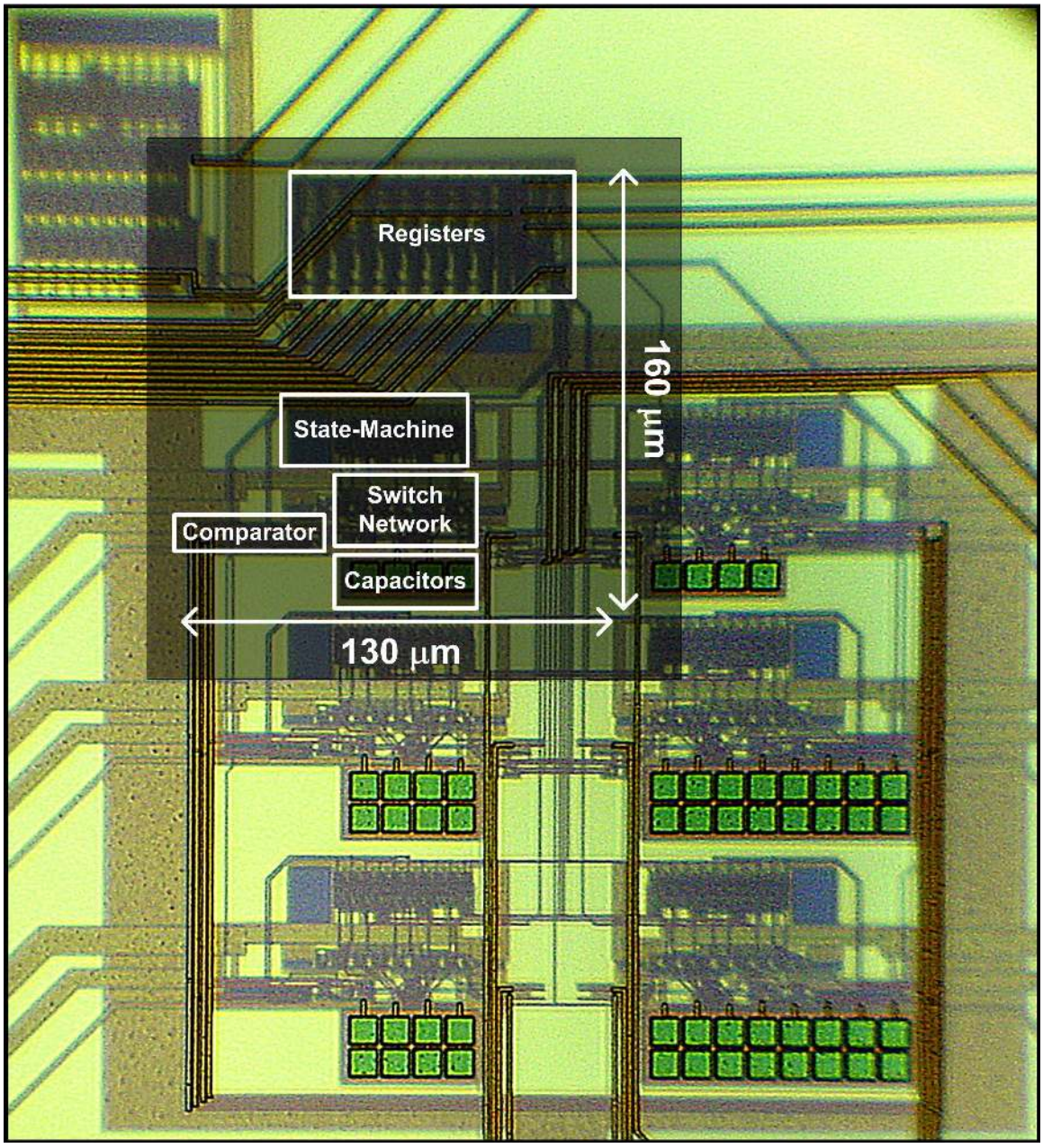

Figure 5-4: The chip was fabricated on the MOSIS TSMC 0.18μm process. The area of interest is 130μm by 160μm.

Table 5.2: Summary of experimental results for low-power ADC

Performance Metric	Value
Technology	MOSIS TSMC 0.18μm
Voltage Supply	
Analog	1.2 Volts
Digital	0.75 Volts
Reference Current	80nA
Input Current Range	10nA to 320nA
Input Offset Current	10nA
Integrating Capacitor	500fF
T_{clk}	1μsec
Sampling Rate	45kHz
INL	$\leq \pm$1.0 LSB [8 bits] typical
DNL	$\leq \pm$0.8 LSB [8 bits] typical
SNR	47dB
SFDR	51dB
ENOB	7.4 bits
Power Dissipation	
Analog	360nW
Digital	600nW
Thermal Noise-Limited	
Energy per Quantization Level	0.12pJ
Active Area	0.021mm^2

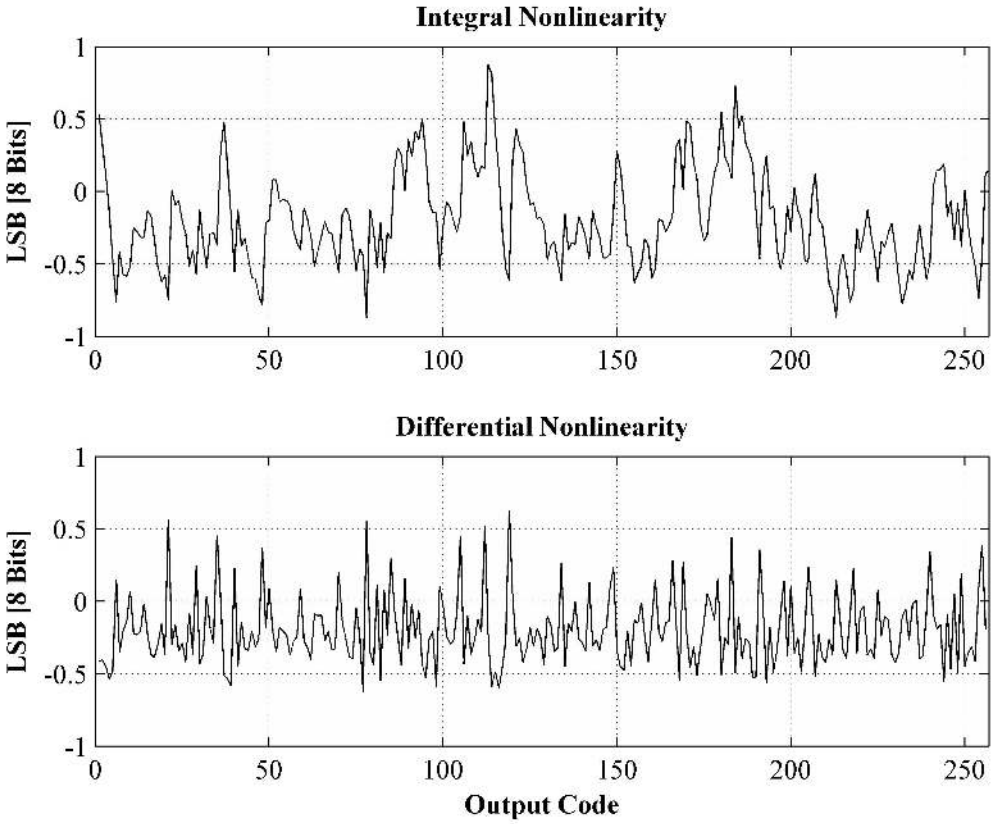

Figure 5-5: The INL was obtained using a least-squared-error approximation.

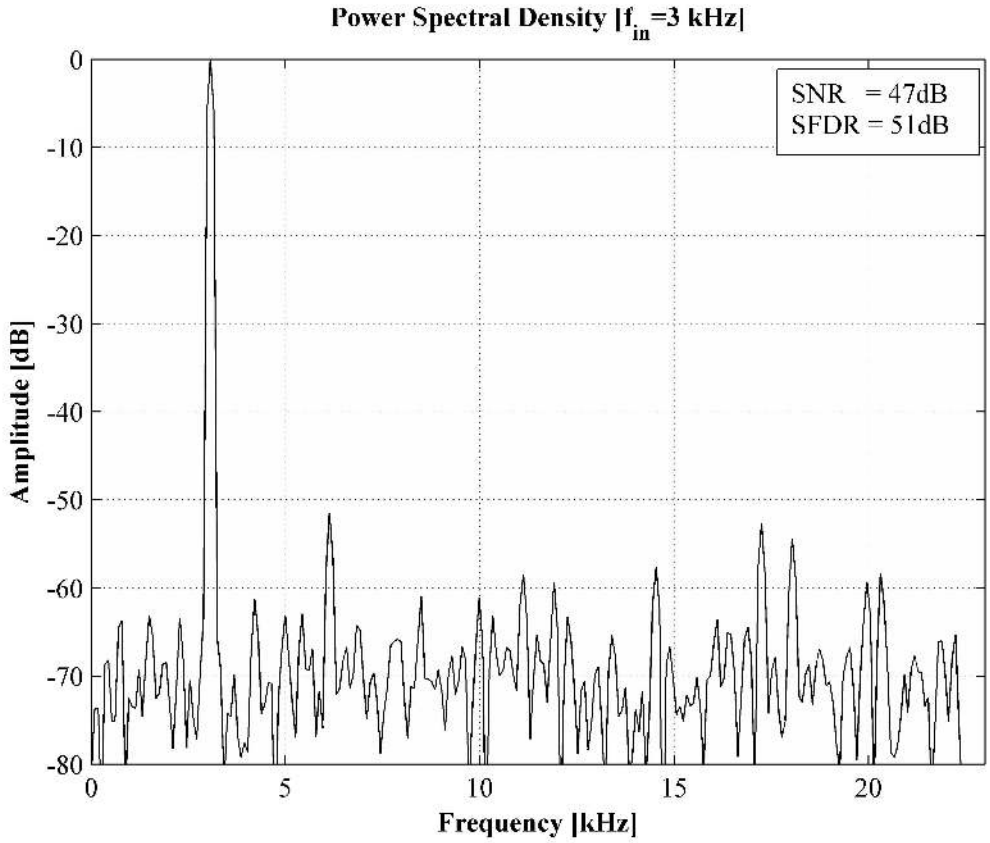

Figure 5-6: For a full-scale sinusoidal input at 3kHz, the SNR is 47dB while the second harmonic limits the SFDR to 51dB.

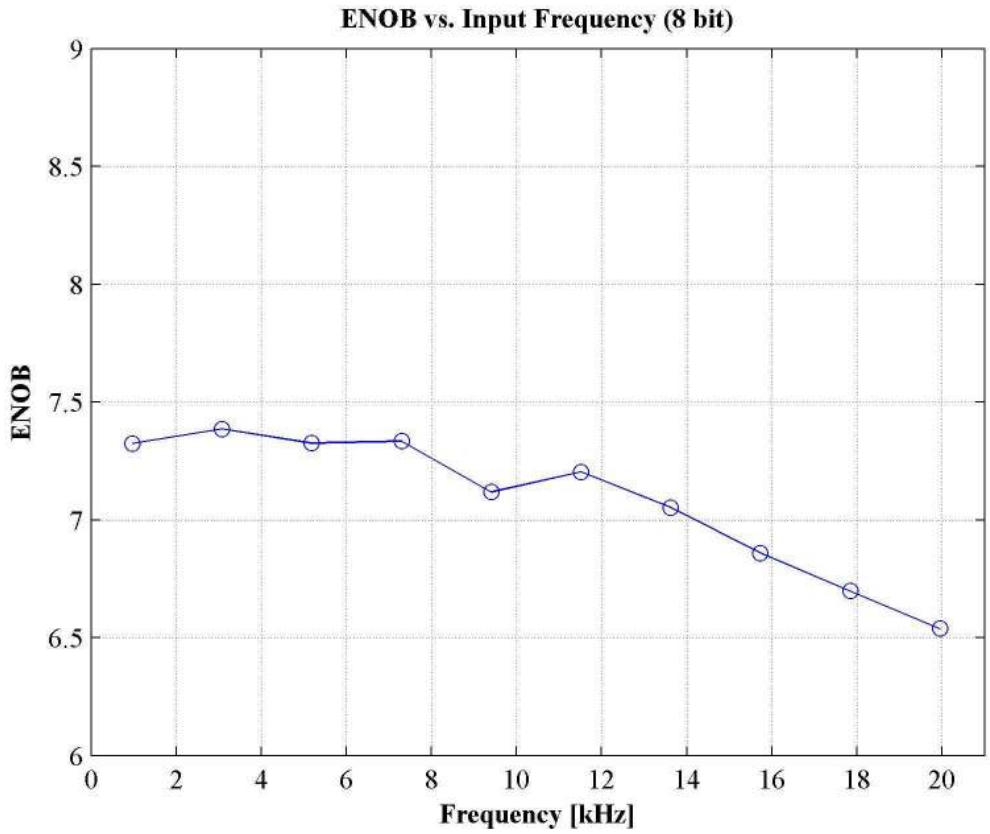

Figure 5-7: A full-scale sinusoidal input was swept from 1kHz to 20kHz, and the resulting ENOBs are shown.

5.4 Comparing Energy-Efficient A/D Converters

The performance specifications such as sampling rate, precision, and power consumption can vary widely among A/D converters. For obvious reasons, each converter's design is optimized for different needs; some designs stress high SNR performance while others pay close attention to total power consumption. In this section, we will focus on comparing energy-efficient converters, i.e., converters with moderate or high precision with moderate sampling rates that consume very low power. However, even among this specific class of converters, the performance metrics can vary enough such that it's often practical to utilize a normalized performance metric that combines all three of these parameters. First introduced in Chapter 1, (5.11) is commonly referred to as the Figure-of-Merit, or FOM, and is discussed in detail in [46].

$$FOM = \frac{2^{ENOB} f_{samp}}{P} \tag{5.11}$$

We refer to the inverse of (5.11) as En, the energy-per-quantization-level and is shown in (5.12). Essentially, En demonstrates the overall efficiency of an A/D or D/A converter. En reflects the energy required to increase the number of quantization levels by one.

$$En = \frac{P}{2^{ENOB} f_{samp}} \tag{5.12}$$

In Section 4, we presented a 12 bit converter that attempted to maximize precision while maintaining low total power consumption. Using the thermal noise-limited ENOB of 11 bits, the En for this converter is 1.17pJ/State. In Section 5.1, we presented a moderate precision converter that consumes less than 1μW. Using it's thermal noise-limited ENOB of 7.4 bits, we calculate that En is 0.12pJ/State. We compare our converters to some of the latest, most energy-efficient converters reported in Table 5.3.

Our low-power converter's 0.12pJ/State is the lowest En ever reported. In addition, the converter requires 20pJ per conversion as opposed to 31pJ per conversion reported in [24]. Our low-power converter also consumes significantly less area at

	High Precision	Low Power	[23]	[30]	[24]	[47]	[48]
Technology	0.35μm	0.18μm	0.25μm	0.09μm	0.18μm	0.09μm	0.8μm
ENOB	11	7.4	8.6	14.3	7.6	15.0	13.0
Input Frequency	$1k$Hz	$3k$Hz	$6.4k$Hz	$11k$Hz	$1.67k$Hz	$1k$Hz	$1k$Hz
Sampling Rate	$31.25k$Hz	$45k$Hz	$150k$Hz	$20k$Hz	$100k$Hz	$10k$Hz	$16k$Hz
Power Dissipation	75μW	0.96μW	30μW	140μW	3.1μW	73μW	150μW
En (pJ/State)	1.17	0.12	0.53	0.34	0.16	0.22	1.15
Active Area	$0.45mm^2$	$0.021mm^2$	$0.11mm^2$	$0.18mm^2$	$0.053mm^2$	$1mm^2$	$1.3mm^2$

Table 5.3: Comparison of energy-efficient A/D converters

$0.021mm^2$. All of the converters listed in Table 5.3 are either successive-approximation [24] [23], $\Sigma\Delta$ [30] [47], or a combination of $\Sigma\Delta$ and algorithmic architectures [48]. In order to obtain a better understanding of the trend towards energy-efficient A/D converters, we surveyed over ninety converters published in the *IEEE Journal of Solid-State Circuits* and *IEEE International Solid-State Circuits Conference* from year 2001 through 2005.

Figure 5-8 shows a plot of the FOM versus the bandwidth of the converters. The solid red line in the figure represents the least-squared-error approximation of the FOM versus bandwidth in the logarithmic domain. The regression analysis reveals a trend that higher-speed converters are less energy-efficient.

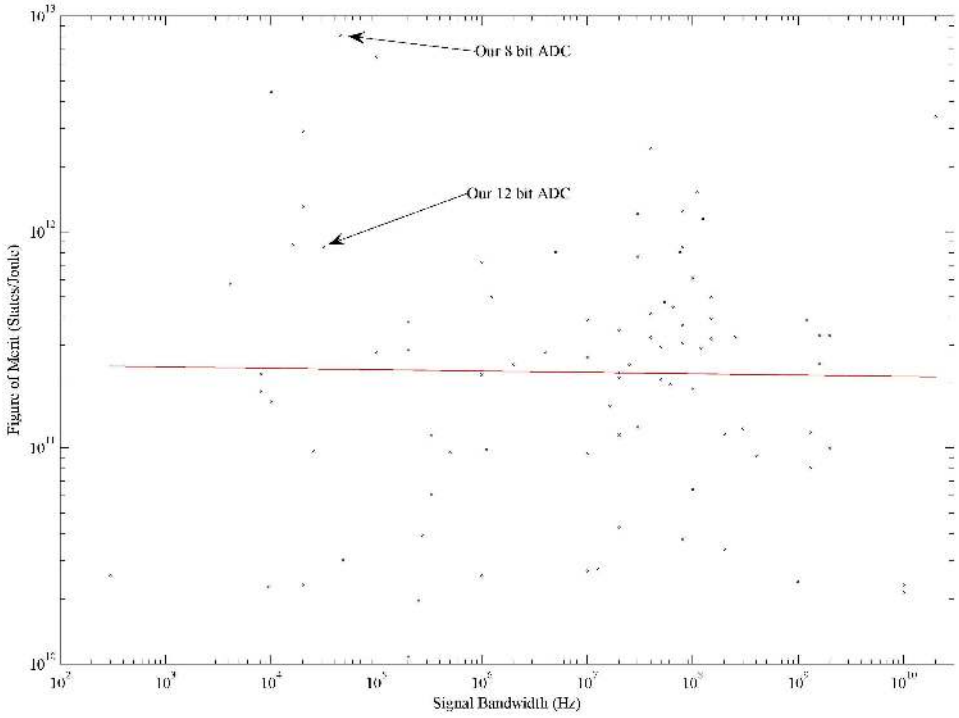

Figure 5-8: A survey of 92 A/D converters from the *IEEE Journal of Solid-State Circuits* and *IEEE International Solid-State Circuits Conference* from 2001 to 2005 is shown. A plot of the FOM vs. bandwidth indicates that lower speed often equates to better energy-efficiency. The solid line represents a linear least-squared-error approximation between FOM and bandwidth in the log domain.

5.5 Scaling to 90nm Technology

In our converter, the analog power consumption will not scale well at these power levels, but the digital power consumption should scale well with technology. Therefore, since two-thirds of the total power consumption is digital, the overall power consumption should scale fairly well. Simulations in 90nm technology show that the digital power consumption scales from 600nW to 350nW.[1] Furthermore, since the threshold voltages are lower at 400mV, the analog power supply can be reduced from 1.2V to 1.0V, reducing the analog power consumption to 300nW. Thus, the total simulated power consumption is approximately 650nW for the same precision and sampling rate as our 8 bit audio converter. If we assume that only the digital subsystem scales with technology, the entire A/D converter will take approximately 0.008mm^2 in area.

[1] We utilized the low-leakage, high threshold voltage MOSFET SPICE models from the UMC 90nm mixed-mode process.

Chapter 6

Conclusions

We have presented a completely novel method for performing analog-to-digital conversion that uses time as an intermediate signal variable as opposed to current or voltage. And unlike and other previously reported time-based converters, our algorithm is the first to trade off time and precision in a linear fashion. We have shown two different implementations that illustrate energy-efficient operation. And further improvements to the algorithm and design of the analog and digital blocks can improve the efficiency even more. For instance, in the design of the pre-amplifier in the high precision converter, we can improve its noise performance by increasing the g_m/I ratio. We can also increase the sampling rate by implementing a conditional $1 + \varepsilon$ algorithm and reduce power consumption by turning down the bias current of the comparator when it is not needed. These techniques, along with others, can be used to push its energy-efficiency even lower.

However, there exists other natural directions for our research. We can explore specific applications where our converter possesses a inherent advantage over other techniques. We will briefly discuss two areas where we believe our algorithm can provide a unique advantage: Time-to-digital conversion and digitization of polynomial coefficients. We will also describe possible immediate applications for our work.

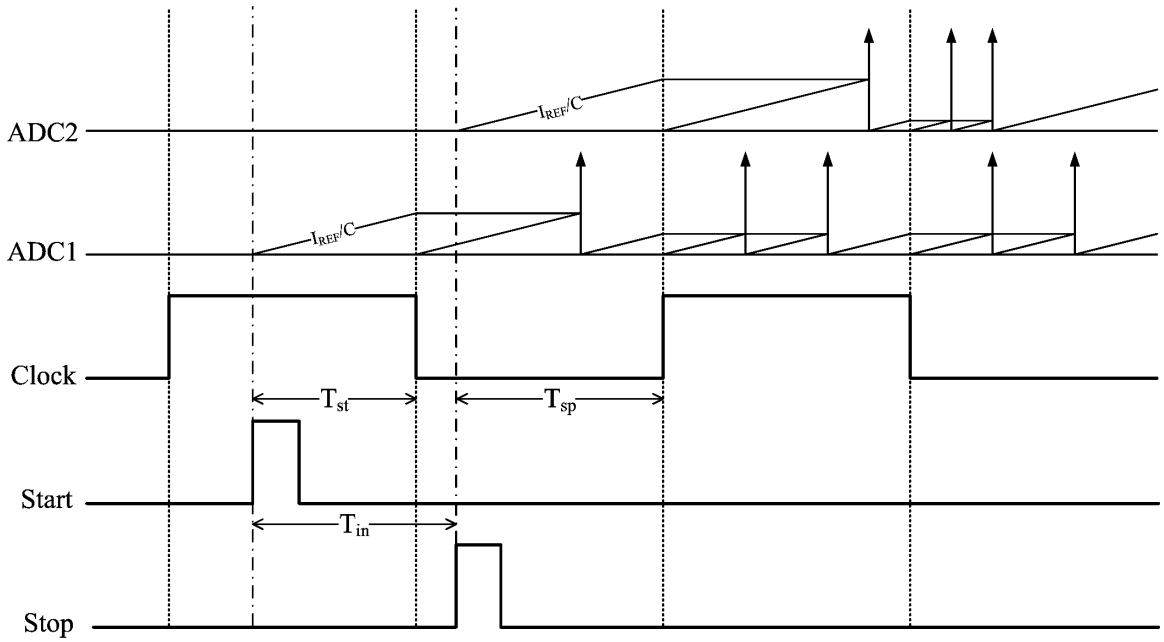

Figure 6-1: The basic approach to time-to-digital conversion using our time-based algorithm is shown. The reference currents and capacitors in ADC1 and ADC2 do not need to match.

6.1 Time-to-Digital Conversion

Our algorithm utilizes time as a signal variable, so it's only natural to explore time-to-digital conversion as a potential application. Areas such as laser-based range-finders, global-positioning-satellite (GPS) devices, and positron emission topography (PET) systems all require time-to-digital converters. And recent publications indicate the growing need for energy-efficient TDCs [49] [50] [51]. However, most TDCs utilize inefficient conversion techniques such as dual-slope [52], Vernier delay-line [53], or a combination of a latter architecture with a flash-based converter [51]. Using two of our converters in parallel, we can construct an energy-efficient asynchronous TDC that can detect sub-nanosecond time-intervals in a single measurement. Figure 6-1 illustrates the basic principle behind the construction of a TDC using our architecture. ADC1 digitizes the time interval between the start pulse and the following clock edge, and ADC2 digitizes the time interval between the stop pulse and its following clock edge. Then the time interval of interest, T_{in}, can be obtained by simply subtracting the final outputs of the two ADCs. If T_{in} is greater than one clock cycle, a simple

counter can be utilized to keep track of "over-ranging" conditions. And if we need to digitize synchronous time intervals, only a single converter is required. A prototype TDC has yet to be developed, but considering the simplicity of our ADC architecture and the ease in which it can be modified to perform time-to-digital conversion, it has the potential to become one of the world's most energy-efficient time-to-digital converters.

6.2 Computational A/D's for Polynomial Functions

Another potential application for our algorithm is the computation and digitization of polynomial functions in a sequential Taylor-series fashion: We can perform a "translinear" type calculation on timing variables in order to quantize successive terms of an M^{th} order polynomial function. Figure 6-2 illustrates the principle behind the time-multiplication process. ADC1 quantizes the first timing variable, T_{x1}, in the same manner as the original algorithm. Therefore, using (3.1), we know that

$$\frac{I_{IN}}{I_{REF}} = \frac{T_{x1}}{T_{clk}}. \tag{6.1}$$

ADC2, however, begins integrating I_{IN} at the end of the second comparator output of ADC1 such that it digitizes the timing interval, T_{x2}. The reason for beginning the integration process of ADC2 here instead of immediately after the first clock cycle is that the *Time-to-Voltage* and *Voltage-to-Time* process eliminates the comparator delays and charge-injection errors and because it references T_{x1} to a following instead of preceding clock edge. Specifically, T_{x1} labeled in ADC1 is identical to T_{x1} labeled in ADC2. Then the quantized output of ADC2 will reflect

$$\frac{I_{IN}}{I_{REF}} = \frac{T_{x2}}{T_{x1}}. \tag{6.2}$$

Combining (6.1) with (6.2), we can see that

$$\left(\frac{I_{IN}}{I_{REF}}\right)^2 = \frac{T_{x2}}{T_{clk}}. \tag{6.3}$$

The same analysis can be performed on the successive ADCs such that we can derive the following generalized formula:

$$\left(\frac{I_{IN}}{I_{REF}}\right)^m = \frac{T_{xm}}{T_{clk}} \tag{6.4}$$

The weighting of each order of the polynomial can be adjusted by changing the reference current for that converter. For instance, if the reference current in ADC2 is $I_{REF}/3$, then (6.3) changes to

$$3\left(\frac{I_{IN}}{I_{REF}}\right)^2 = \frac{T_{x2}}{T_{clk}}. \tag{6.5}$$

While we have not implemented such computational A/D's, our scheme appears to show promise because of its simplicity.

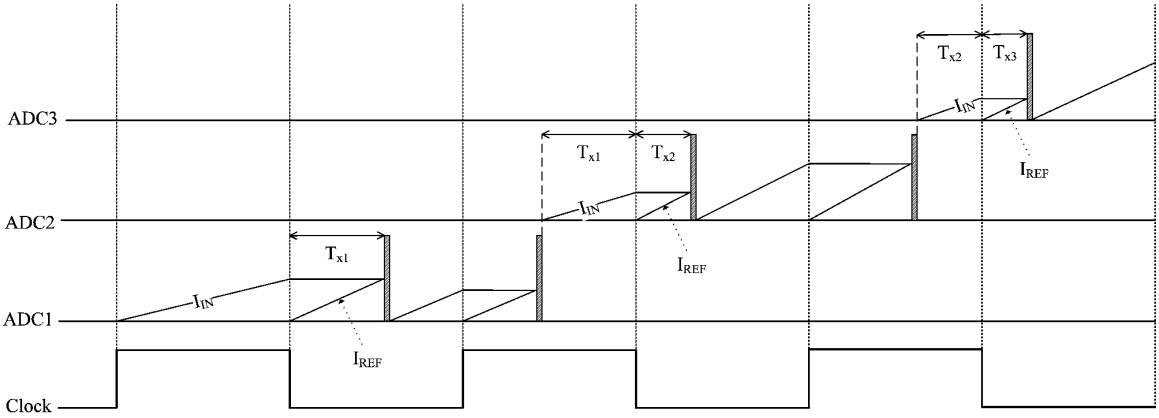

Figure 6-2: Each converter in the chain digitizes the coefficient for that order of the polynomial. The example shown above simply computes $(I_{IN}/I_{REF})^m$, but varying the reference current per successive converter can result in non-unity coefficient digitization.

6.3 Possible Immediate Applications of this Work

The precision and bandwidth requirements on A/D converters for most biomedical applications are rather modest. While electrocardiography (ECG) and electroencephalography (EEG) systems may require small minimum detectable signals (1μV and 0.3μV, respectively), the necessary dynamic range in a linear system corresponds

to 8 bits, and the bandwidth requirements range from 100Hz to 500Hz [54]. The converter utilized in the acquisition of cardiac signals in implantable pacemakers requires only 8 bits of precision at a bandwidth of 250Hz [55]. Pulse-oximetry systems require 7 bits of precision and 100Hz of bandwidth to digitize their outputs if analog preprocessing is used [56]. Our 8 bit audio converter appears to be the most energy-efficient converter reported to date, and therefore, it could be quite useful in these biomedical applications where high speed and high precision are not required, but where energy-efficiency is paramount. Sensor networks have similar requirements and would also benefit [24]. When combined with a microphone front-end and automatic gain control (AGC) [57], our 12 bit converter could be used in low-power hearing aids and cochlear-implant applications.

Appendix A

Dual-Slope vs. Our Converter: A Comparison in Efficiency

Using the same analog circuit parameters from our A/D architecture, we can estimate the total power consumption of a dual-slope architecture for the same sampling rate and precision. For instance, the two integrating current levels and the two different comparator topologies satisfy the thermal noise requirements for the 8 and 12 bit architectures, respectively. Therefore, for the same precision levels and sampling rates, the digital power consumption will increase due to the exponentially faster clock frequency that would be required for the equivalent dual-slope architecture. Specifically, a dual-slope architecture will require a clock that is $2^{N+1}/\beta N$ faster where β is defined as the number of clock cycles required per successive bit in our converter.

The digital power consumption for the dual-slope converter, $P_{DIGITAL,DS}$, is equal to

$$\left(\frac{1}{2}\right) C_{L,DS} V_{DD}^2 f_{clk,DS} \tag{A.1}$$

where $C_{L,DS}$ is defined as the total effective load capacitance of the digital subsystem. We can approximate $C_{L,DS}$ as the total gate capacitance of two LSB registers.[1]

The total effective load capacitance of our converter can be approximated as the

[1]The first two LSB registers are comprised of a single register clocked at the maximum frequency plus the geometric summation of the higher order registers.

total gate capacitance of three LSB registers.[2]. Therefore, using the fact that $f_{clk,DS}$ is $2^{N+1}/\beta N$ times faster than our clock, f_{clk}, we can show that

$$P_{DIGITAL,DS} = P_{DIGITAL}\left(\frac{2}{3}\right)\left(\frac{2^{N+1}}{\beta N}\right), \tag{A.2}$$

where $P_{DIGITAL}$ is the total digital power of our converter.

For 12 bits of precision and $\beta = 5.3$,[3] $P_{DIGITAL,DS}$ is approximately 86 times greater than $P_{DIGITAL}$. Consequently, the total power consumption of a dual-slope converter with the same sampling rate and precision is $1.4m$W ($60m$W [Analog] + $1.3m$W [Digital]). This is approximately 19 times greater than 75μW, the total power consumption of our 12 bit converter.

For 8 bits of precision and $\beta = 2.7$,[4] $P_{DIGITAL,DS}$ is approximately 16 times greater than $P_{DIGITAL}$. Thus, the total power consumption of the equivalent dual-slope converter is 9.4μW ($360n$W [Analog] + 9μW [Digital]) which is approximately 10 times greater than $960n$W, the total power consumption of our 8 bit converter.

[2]Once again, the first two LSB registers are comprised of a single register clocked at the maximum frequency plus the geometric summation of the higher order registers. The extra LSB register in our converter is attributed to our more complex state-machine being clocked at the maximum frequency.

[3]The conversion time is equal to 64 clock cycles in our 12 bit converter.

[4]The conversion time is equal to 22 clock cycles in our 8 bit converter.

Appendix B

An Analysis of Voltage vs. Time Amplification

A basic assumption we have made throughout this thesis is that time amplification is inherently more energy-efficient than voltage amplification. This assumption is based on the simple fact that we are utilizing a single device worth of noise whereas an amplifier usually has at least four devices worth of noise. In this section, we will present a more rigorous analysis comparing the energy-efficiency of time amplification versus voltage amplification.

In order to validate our claim, we must first establish a metric that fairly compares the performance of the two amplification methods. In Chapter 5, we introduced a figure of merit for A/D converters that incorporated the sampling rate (f_{samp}), precision (N), and power consumption (P) of the converter. Let us define $FOM_{amp,t}$ as

$$\frac{2^{2N} f_{samp}}{P}. \tag{B.1}$$

Equation (B.1) is similar to the FOM presented in Chapter 5 except that we use 2^{2N} for the precision to reflect the thermal-noise-limiting aspect of the converter. In addition, the sampling rate is determined by the time required for a single amplification phase, and the power is dominated by the integration current.[1] For a voltage

[1] We are assuming that the majority of the power budget is reserved for the integrating current source.

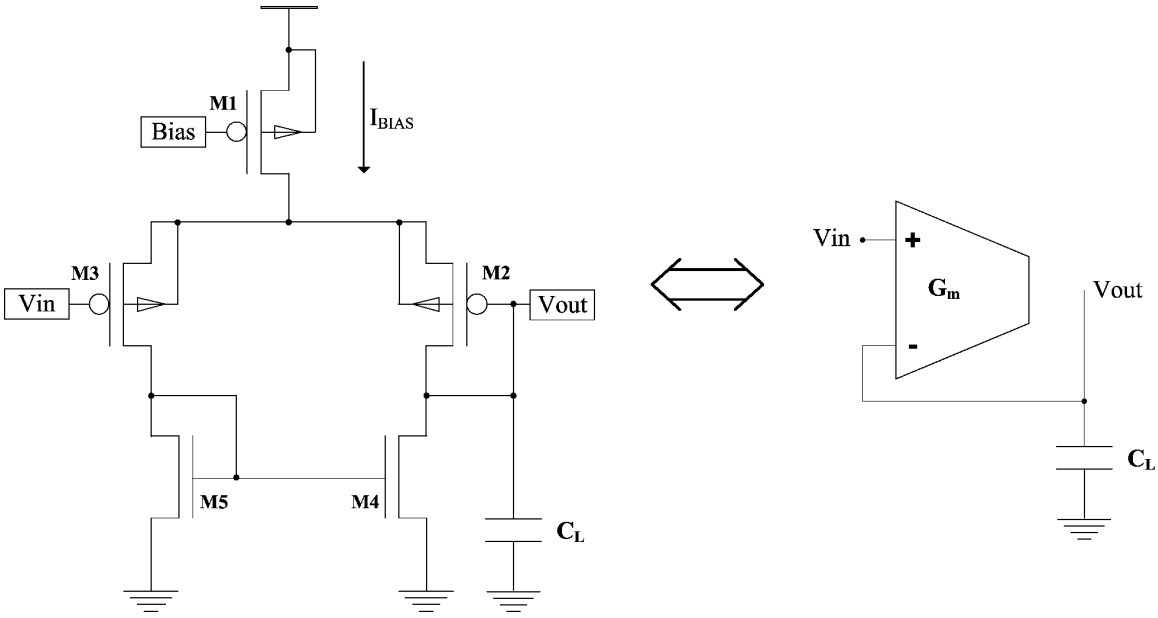

Figure B-1: A standard five transistor OTA is shown in a negative-feedback unity-gain G_m-C filter topology.

amplifier, we introduce an equivalent metric defined as $FOM_{amp,v}$ that incorporates the gain/bandwidth product (GBW), dynamic range (D_R), and power consumption (P) and is shown in (B.2).

$$FOM_{amp,v} = \frac{D_R GBW}{P} \tag{B.2}$$

In our analysis, we will compare our time amplification method to a negative-feedback unity-gain G_m-C filter topology that utilizes a simple five transistor OTA in sub-threshold operation (See Figure B-1). The dynamic range of an OTA is often defined as

$$\frac{V_L^2}{V_{nz}^2}, \tag{B.3}$$

where V_L is the linear range and V_{nz} is the input referred noise. If we define I_{BIAS} as the OTA's bias current, M as the number of noise sources, and C_L as the output capacitance, it can be shown that the dynamic range for subthreshold operation is

$$D_R = \frac{2C_L I_{BIAS}}{MqG_m}. \tag{B.4}$$

The gain/bandwidth product for a gain of two amplifier can be approximated by the bandwidth of the unity-gain amplifier in Figure B-1 and is defined as

$$GBW = \frac{G_m}{2\pi C_L}. \tag{B.5}$$

Since the power consumption of the OTA is $I_{BIAS}V_{DD}$, we can use (B.4) and (B.5) and the fact that $M = 4$ in the OTA to reduce $FOM_{amp,v}$ to

$$\frac{1}{4\pi q V_{DD}}. \tag{B.6}$$

For the time amplification process, the analysis for deriving $FOM_{amp,t}$ is straightforward. In Section 4.2.3, we introduced the current noise model for a MOSFET. If we assume that the current source MOSFET is operating in subthreshold, the current noise squared per unit bandwidth shown in (4.3) simplifies to

$$2qI_{REF}\Delta f, \tag{B.7}$$

where I_{REF} is the integration current. If T_{ref} is the integration time for current I_{REF}, then the equivalent bandwidth is $1/2T_{ref}$. Thus, the square of the total current noise is

$$\frac{qI_{REF}}{T_{ref}}. \tag{B.8}$$

For a temporal gain of two, we integrate the current two times,[2] and consequently, the square of the total noise of the system is $2qI_{REF}/T_{ref}$. Therefore, the ratio of I_{REF}^2 to the square of the total noise produces the square of the number of quantization levels, i.e.,

$$2^{2N} = \frac{I_{REF}T_{ref}}{2q}. \tag{B.9}$$

Using (B.9) and the fact that the sampling rate of the time amplification system is at most $1/2T_{ref}$ and that the power consumption is $I_{REF}V_{DD}$, $FOM_{amp,v}$ simplifies

[2]We have assumed two integration cycles instead of the standard three since the third cycle is required for residue subtraction and error cancellation, which we have ignored in this analysis.

to

$$\frac{1}{4qV_{DD}}. \tag{B.10}$$

Therefore, we can see that our time amplification method is approximately 3 times more energy-efficient than the equivalent voltage amplification method that uses a 5 transistor OTA in a G_m-C filter topology.

Appendix C

Capacitor Mismatch Reduction Technique

Let us define the following capacitances:

$$\begin{aligned} C_1 &= C \\ C_2 &= C + \Delta C \end{aligned} \tag{C.1}$$

During the MSB stage, we integrate the input current with C_1 and the reference current with C_2. Therefore, at the end of the MSB stage, T_{res} increases by ΔT_{res}, where

$$\Delta T_{res} = \left(\frac{\Delta C}{C} \right) T_{ref} \tag{C.2}$$

and

$$T_{ref} = M T_{clk} + T_{res} \quad \forall \quad M \in [0, 4). \tag{C.3}$$

Equation (C.3) is simply the quantification of the residue subtraction routine inherent to the algorithm. At this point in time, we are free to use either C_1 or C_2 for the successive *Time-to-Voltage* conversion.

If we use C_1, the *Time-to-Voltage* conversion has a gain term of I_{REF}/C, whereas the *Voltage-to-Time* conversion uses C_2 and has a gain term of $(C+\Delta C)/I_{REF}$. Thus,

the time after the *Time-to-Voltage-to-Time* conversion is

$$T_{resB} + G_{12}, \tag{C.4}$$

where

$$\begin{aligned} G_{12} &= \left(T_{clk} - T_{res} - T_{ref}\left(1 + \frac{\Delta C}{C}\right)\right)\frac{\Delta C}{C} \\ &\simeq ((1-M)T_{clk} - 2T_{res})\,\frac{\Delta C}{C}. \end{aligned} \tag{C.5}$$

If instead of C_1 we use C_2 for the *Time-to-Voltage* conversion, the order of the gain terms reverse and the time after the *Time-to-Voltage-to-Time* conversion is

$$T_{resB} + G_{21}, \tag{C.6}$$

where

$$G_{21} = -\left(\frac{\Delta C(M+1)}{C + \Delta C}\right)T_{clk}. \tag{C.7}$$

By comparing (C.5) and (C.7), we can show that the capacitor-alternating scheme is less prone to capacitor mismatch errors. Basically, we can demonstrate that

$$|G_{12}| < |G_{21}|. \tag{C.8}$$

If we expand on (C.8), we can derive the following capacitor matching requirement;

$$\frac{\Delta C}{C} < \frac{2MT_{clk} + 2T_{res}}{(1-M)T_{clk} - 2T_{res}}. \tag{C.9}$$

For reasonable values of $\Delta C/C$, we can see that G_{12} will always be less than G_{21} for all values of $M \in [0, 4)$ and $T_{res} \in [0, T_{clk})$. In other words, as long as we have reasonable capacitor matching, the capacitor-alternating scheme will produce less errors due to capacitor mismatch.

Appendix D

Detailed Noise Analysis

D.1 Diminishing Contribution of Noise from the Comparator

Let us define x to be the total output noise of the comparator. In a manner similar to a successive sub-ranging converter, comparator noise is added before each gain stage in the cascaded set of amplifiers, as show in Figure D-1. The noise in each successive sub-ranging stage is added twice while it is added only once in the MSB stage: this is due to the fact that the MSB stage utilizes the comparator only once whereas the successive stages utilize the comparator twice. In order to calculate the total input-referred noise of the converter, we divide the noise at each point by the total gain up to that point. Since the gain of each successive stage is fixed at 2, the total input-referred noise (squared) of the converter can be shown to be

$$x^2 + 2\left(\frac{x}{2^1}\right)^2 + 2\left(\frac{x}{2^2}\right)^2 + 2\left(\frac{x}{2^3}\right)^2 + \ldots + 2\left(\frac{x}{2^N}\right)^2 \tag{D.1}$$

which is bounded by

$$\left(\frac{2}{1-\frac{1}{4}} - 1\right)x^2 = \frac{5}{3}x^2 \tag{D.2}$$

as N approaches ∞.

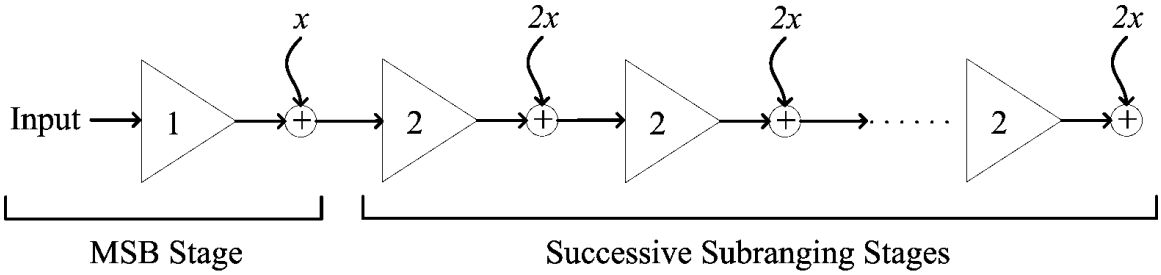

Figure D-1: Comparator noise, x, is added to our system after each gain stage. Since we use the comparator once during the MSB stage and twice during the successive stages, x is weighted appropriately.

D.2 Integration Noise Derivation

Using our current noise model in (4.3), we can define the voltage noise due to white noise in an integrating current, I_i, as

$$\begin{aligned} \Delta V_{NZ,i}^2 &= I_{NZ,i}^2 \left(\frac{T_i}{C_i}\right)^2 \\ &= 4kT\gamma g_{mi}\Delta f \left(\frac{T_i}{C_i}\right)^2 \end{aligned} \tag{D.3}$$

Since each current integration process is statistically independent, the total voltage noise variance due to the current integration in the MSB stage is $\Delta V_{NZ,in}^2 + \Delta V_{NZ,ref}^2$. The integration time for the input current is fixed at T_{clk} such that the bandwidth is approximately $1/(2T_{clk})$. The integration time for the reference current is a function of the input and is defined as T_{ref}, and the bandwidth is accordingly $1/(2T_{ref})$. Therefore, the total voltage noise variance at the end of the MSB stage is

$$\Delta V_{NZ,in}^2 + \Delta V_{NZ,ref}^2 = \frac{2kT\gamma}{C_i^2}\left(g_{min}T_{clk} + g_{mref}T_{ref}\right) \tag{D.4}$$

where g_{min} is the transconductance of the input current source transistor and g_{mref} is the transconductance of the reference current source transistor. Since T_{ref} is a

function of I_{IN} from (3.1), we can rewrite (D.4) as

$$\frac{2kT\gamma}{C_i^2}T_{clk}\left(g_{min}+g_{mref}\frac{I_{IN}}{I_{REF}}\right) \tag{D.5}$$

During each successive stage, we integrate the reference current three times. Hence, the total voltage noise variance is $3\Delta V_{NZ,res}^2$, where

$$\Delta V_{NZ,res}^2 = \frac{2kT\gamma}{C_i^2}T_{res}g_{mref} \tag{D.6}$$

for some residual time, T_{res}. For simplicity, we can approximate that T_{res} will on average be $3T_{clk}/2$ if we always use the $1+\varepsilon$ algorithm for the *Voltage-to-Time* conversion.[1] Then (D.6) reduces to

$$\Delta V_{NZ,res}^2 = \frac{2kT\gamma}{C_i^2}T_{clk}g_{mref}\frac{3}{2} \tag{D.7}$$

As we showed in Appendix D.1, the noise after a successive stage needs to be divided by the total gain up to that stage in order to calculate the input-referred noise. Therefore, the total input-referred voltage noise from the successive stages is

$$3\Delta V_{NZ,res}^2\left[\left(\frac{1}{2^1}\right)^2+\left(\frac{1}{2^2}\right)^2+\left(\frac{1}{2^3}\right)^2+\ldots+\left(\frac{1}{2^N}\right)^2\right] \tag{D.8}$$

which is bounded by

$$3\Delta V_{NZ,res}^2\left(\frac{1}{1-\frac{1}{4}}-1\right)=\Delta V_{NZ,res}^2 \tag{D.9}$$

as N approaches ∞. Thus, we compute that the total input-referred voltage noise variance due to the successive stages is simply bounded by (D.7). Therefore, the total

[1]For the low-power 8 bit converter, we implement the $1+\varepsilon$ algorithm only for the 3 bits following the 2 MSBs. Even so, using this approximation for the remaining LSB cycles is still a good estimate because their contribution to the overall precision is very small.

voltage noise due to all integration cycles, $V^2_{NZ,int}$, is defined as

$$V^2_{NZ,int} = \frac{2kT\gamma}{C_i^2} T_{clk} \left(g_{min} + g_{mref} \frac{I_{IN}}{I_{REF}} + g_{mref} \frac{3}{2} \right) \tag{D.10}$$

Each voltage noise source translates to temporal noise by the "gain" of the integration rate, I_{REF}/C_i. Therefore, the total temporal noise due to all integration cycles is $V^2_{NZ,int}$ divided by the integration rate squared. Specifically,

$$T^2_{NZ,int} = \frac{2kT\gamma}{I_{REF}^2} T_{clk} \left(g_{min} + g_{mref} \frac{I_{IN}}{I_{REF}} + g_{mref} \frac{3}{2} \right) \tag{D.11}$$

For the 12 bit converter, the reference and input current source transistors are operating above-threshold. Therefore, if the input current is at its full-scale value of $4I_{REF}$, $g_{min} = \sqrt{4} g_{mref}$, and the upper-bound on (D.11) is

$$T^2_{NZ,int} = \frac{10kT g_{mref}}{I_{REF}^2} T_{clk}. \tag{D.12}$$

Bibliography

[1] K. Uyttenhove and M. Steyaert, "Speed-power-accuracy tradeoff in high-speed CMOS ADCs," *IEEE Trans. Circuits Syst. II*, vol. 49, pp. 280–287, Apr. 2002.

[2] S. K. Ammann, "Integrating analog-to-digital converter." U.S. Patent 3,316,547, Apr. 25 1967.

[3] J.-J. Sit and R. Sarpeshkar, "A micropower logarithmic A/D with offset and temperature compensation," *IEEE J. Solid-State Circuits*, vol. 39, pp. 308–319, Feb. 2004.

[4] R. Plassche, *Integrated Analog-to-Digital and Digital-to-Analog Converters*, pp. 277–279. In [58].

[5] P. Petrovic, S. Marjanovic, and M. Stevanovic, "New algorithm for measuring 50/60 Hz AC values based on the usage of slow A/D converters," *IEEE Trans. Instrum. Meas.*, vol. 49, pp. 166–171, Feb. 2000.

[6] B. Breten, T. Lehmann, and E. Braun, "Integrating data converters for picoampere currents from electrochemical transducers," in *IEEE Int. Symp. Circuits and Systems*, vol. 5, (Geneva, Switzerland), pp. 709–712, May 2000.

[7] B. Murmann and B. E. Boser, "A 12-bit 75-Ms/s pipelined ADC using open-loop residue amplification," *IEEE J. Solid-State Circuits*, vol. 38, pp. 2040–2050, Dec. 2003.

[8] P. Quinn and M. Pribytko, "Capacitor matching insensitive 12-bit 3.3 Ms/s algorithmic ADC in 0.25μm CMOS," in *IEEE Custom Integrated Circuits Conference*, pp. 425–428, Sept. 2003.

[9] B. Razavi, *Principles of Data Conversion System Design*, pp. 286–291. In [59].

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

[11] T. B. Cho and P. R. Gray, "A 10 b, 20 Msample/s, 35 mW pipeline A/D converter," *IEEE J. Solid-State Circuits*, vol. 30, pp. 166–172, Mar. 1995.

[12] A. M. Abo and P. R. Gray, "A 1.5-V, 10-bit, 14.3-Ms/s CMOS pipeline analog-to-digital converter," *IEEE J. Solid-State Circuits*, vol. 34, pp. 599–606, May 1999.

[13] K. Hwang, *Computer Arithmetic-Principles, Architecture and Design*. New York, New York: John Wiley & Sons, Inc., 1979.

[14] Y. Chiu, P. R. Gray, and B. Nikolic, "A 14-b 12-Ms/s CMOS pipeline ADC with over 100-dB SFDR," *IEEE J. Solid-State Circuits*, vol. 39, pp. 2139–2151, Dec. 2004.

[15] S.-T. Ryu *et al.*, "A 14-b linear capacitor self-trimming pipelined ADC," *IEEE J. Solid-State Circuits*, vol. 39, pp. 2046–2051, Nov. 2004.

[16] I. Galton, "Digital cancellation of D/A converter noise in pipelined A/D converters," *IEEE Trans. Circuits Syst. II*, vol. 47, pp. 185–196, Mar. 2000.

[17] X. Wang, P. J. Hurst, and S. H. Lewis, "A 12-bit 20-Msample/s pipelined analog-to-digital converter with nested digital background calibration," *IEEE J. Solid-State Circuits*, vol. 39, pp. 1799–1808, Nov. 2004.

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

[19] P. Wambacq and W. M. Sansen, *Distortion Analysis of Analog Integrated Circuits*. Boston, Massachusetts: Kluwer, 1998.

[20] S. Mortezapour and E. K. F. Lee, "A 1-V, 8-bit successive approximation ADC in standard CMOS process," *IEEE J. Solid-State Circuits*, vol. 35, pp. 642–646, Apr. 2000.

[21] C. J. B. Fayomi, G. W. Roberts, and M. Sawan, "A 1-V, 10-bit rail-to-rail successive approximation analog-to-digital converter in standard 0.18μm CMOS technology," in *IEEE Int. Symp. Circuits and Systems*, vol. 1, (Sydney, Australia), pp. 460–463, Apr. 2001.

[22] P. H. Saul, "Successive approximation analog-to-digital conversion at video rates," *IEEE J. Solid-State Circuits*, pp. 147–151, June 1981.

[23] J. Sauerbrey, D. Schmitt-Landsiedel, and R. Thewes, "A 0.5-V 1-μW successive approximation ADC," *IEEE J. Solid-State Circuits*, vol. 38, pp. 1261–1265, July 2003.

[24] M. D. Scott, B. E. Boser, and K. S. J. Pister, "An ultralow-energy ADC for smart dust," *IEEE J. Solid-State Circuits*, vol. 38, pp. 1123–1129, July 2003.

[25] D. Johns and K. Martin, *Analog Integrated Circuit Design*, pp. 531–536. In [60].

[26] S. Norsworthy, R. Scjreier, and G. Temes, *Delta-Sigma Data Converters Theory, Design, and Simulation*, pp. 165–167. New York: IEEE Press, 1997.

[27] Y.-I. Park, S. Karthikeyan, W. M. Koe, Z. Jiang, and T.-C. Tan, "A 16-bit, 5Mhz multi-bit sigma-delta ADC using adaptively randomized DWA," in *IEEE Custom Integrated Circuits Conference*, pp. 115–118, Sept. 2003.

[28] P. Balmelli and F. Qiuting Huang, "A 25-Ms/s 14-b 200-mW $\Sigma\Delta$ modulator in 0.18-μm CMOS," *IEEE J. Solid-State Circuits*, vol. 39, pp. 2161–2169, Dec. 2004.

[29] P. Rombouts, J. D. Maeyer, and L. Weyten, "A 250-kHz 94-dB double-sampling $\Sigma\Delta$ modulation A/D converter with a modified noise transfer function," *IEEE J. Solid-State Circuits*, vol. 38, pp. 1657–1662, Nov. 2003.

[30] L. Yao, M. Steyaert, and W. Sansen, "A 1-V 140-μW 88-dB audio sigma-delta modulator in 90-nm CMOS," *IEEE J. Solid-State Circuits*, vol. 39, pp. 1809–1818, Nov. 2004.

[31] H. Thompson, M. Hufford, W. Evans, and E. Naviasky, "A low-voltage low-power sigma-delta modulator with improved performance in overload condition," in *IEEE Custom Integrated Circuits Conference*, pp. 1657–1662, Sept. 2004.

[32] B. Razavi, *Principles of Data Conversion System Design*, pp. 100–110. In [59].

[33] R. Plassche, *Integrated Analog-to-Digital and Digital-to-Analog Converters*, pp. 276–277. In [58].

[34] Y. Ding and R. Harjani, "A universal analytic charge-injection model," in *IEEE Int. Symp. Circuits and Systems*, vol. 1, (Geneva, Switzerland), pp. 144–147, May 2000.

[35] D. Johns and K. Martin, *Analog Integrated Circuit Design*, pp. 308–311. In [60].

[36] K. Karadamoglou *et al.*, "An 11-bit high-resolution and adjustable-range CMOS time-to-digital converter for space science instruments," *IEEE J. Solid-State Circuits*, vol. 39, pp. 214–222, Jan. 2004.

[37] R. Nutt, K. Milam, and C. W. Williams, "Digital intervalometer." U.S. Patent 3,983,481, Sept. 26 1976.

[38] C. Salthouse and R. Sarpeshkar, "A micropower band-pass filter for use in bionic ears," in *IEEE Int. Symp. Circuits and Systems*, vol. 5, (Scottsdale, Arizona), pp. 189–192, May 2002.

[39] C. Mead, *Analog VLSI and Neural Systems*, pp. 198–201. Reading, Massachusetts: Addison Wesley, 1989.

[40] R. Sarpeshkar, L. Watts, and C. A. Mead, "Refractory neuron circuits," Tech. Rep. CNS-TR-92-08, California Institute of Technology, Pasadena, California, 1992.

[41] Y. Tsividis, *Operation and Modeling of the MOS Transistor*. New York: McGraw-Hill, 2 ed., 1999.

[42] D. Johns and K. Martin, *Analog Integrated Circuit Design*, pp. 389–391. In [60].

[43] H. Tian, B. Fowler, and A. E. Gamal, "Analysis of temporal noise in CMOS photodiode active pixel sensor," *IEEE J. Solid-State Circuits*, vol. 36, pp. 92–101, Jan. 2001.

[44] R. Sarpeshkar, R. F. Lyon, and C. A. Mead, "A low-power wide-linear-range transconductance amplifier," *Analog Integrated Circuits and Signal Processing*, vol. 13, pp. 123–151, may/jun 1997.

[45] M. Kontiala, A. Heinonen, and J. Nurmi, "Low-power methodology issues in digital circuit design," in *IEEE Int. Symp. Circuits and Systems*, vol. 1, (Scottsdale, Arizona), pp. 493–496, May 2002.

[46] R. H. Walden, "Analog-to-digital converter survey and analysis," *IEEE J. Select. Areas Commun.*, vol. 17, pp. 539–550, Apr. 1999.

[47] D. G. Gata *et al.*, "A 1.1-V 270-μA mixed-signal hearing aid chip," *IEEE J. Solid-State Circuits*, vol. 37, pp. 1670–1678, Dec. 2002.

[48] P. Rombouts, W. D. Wilde, and L. Weyten, "A 13.5-b 1.2-V micropower extended counting A/D converter," *IEEE J. Solid-State Circuits*, vol. 36, pp. 176–183, Feb. 2001.

[49] E. Räisänen-Ruotsalainen, T. Rahkonen, and J. Kostamovaara, "An integrated time-to-digital converter with 30-ps single-shot resolution," *IEEE J. Solid-State Circuits*, vol. 35, pp. 1507–1510, Oct. 2000.

[50] P. Andreani *et al.*, "Multihit multichannel time-to-digital converter with ± 1% differential nonlinearity and near optimal time resolution," *IEEE J. Solid-State Circuits*, vol. 33, pp. 650–656, Apr. 1998.

[51] B. K. Swann *et al.*, "A 100-ps time-resolution CMOS time-to-digital converter for positron emission tomography imaging applications," *IEEE J. Solid-State Circuits*, vol. 39, pp. 1839–1851, Nov. 2004.

[52] P. Palojärvi, K. Määttä, and J. Kostamovaara, "Pulsed time-of-flight laser radar module with millimeter-level accuracy using full custom receiver and TDC ASICs," *IEEE Trans. Instrum. Meas.*, vol. 51, pp. 1102–1108, Oct. 2002.

[53] P. Dudek, S. Szczepanski, and J. V. Hatfield, "A high-resolution CMOS time-to-digital converter utilizing a vernier delay line," *IEEE J. Solid-State Circuits*, vol. 35, pp. 240–247, Feb. 2000.

[54] E. Aksenov *et al.*, "Biomedical data acquisition systems based on sigma-delta analogue-to-digital converters," in *23rd Annual Int. Conf. IEEE EMBS*, vol. 4, pp. 3336–3337, Oct. 2001.

[55] A. Gerosa, A. Maniero, and A. Neviani, "A fully integrated two-channel a/d interface for the acquisition of cardiac signals in implantable pacemakers," *IEEE J. Solid-State Circuits*, vol. 39, pp. 1083–1093, July 2004.

[56] M. Tavakoli, *An Analog VLSI Front-End for Pulse-Oximetry*. PhD thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, Oct. 2005.

[57] R. Sarpeshkar *et al.*, "An analog bionic ear processor with zero-crossing detection," in *IEEE Int. Solid-States Circuits Conf.*, vol. 1, pp. 78–79, Feb. 2005.

[58] R. Plassche, *Integrated Analog-to-Digital and Digital-to-Analog Converters*. Boston, Massachusetts: Kluwer Academic Publishers.

[59] B. Razavi, *Principles of Data Conversion System Design*. New York, New York: John Wiley & Sons, Inc.

[60] D. Johns and K. Martin, *Analog Integrated Circuit Design*. New York, New York: John Wiley & Sons, Inc.