This is to certify that the
thesis entitled

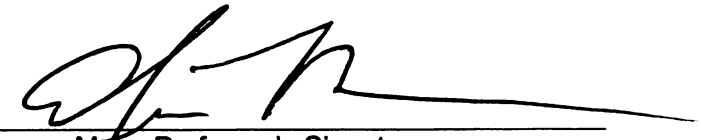
DESIGN, IMPLEMENTATION AND TESTING OF A HYBRID ALGORITHMIC ΣΔ A/D CONVERTER

presented by

Cheong Kun

has been accepted towards fulfillment
of the requirements for the

Master of Science degree in Electrical and Computer Engineering



Major Professor's Signature

8/19/05

Date

DESIGN, IMPLEMENTATION AND TESTING OF A HYBRID ALGORITHMIC ΣΔ A/D CONVERTER

By

Cheong Kun

A THESIS

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

MASTER OF SCIENCE

ELECTRICAL AND COMPUTER ENGINEERING

2005

ABSTRACT

DESIGN, IMPLEMENTATION AND TESTING OF A HYBRID ALGORITHMIC ΣΔ A/D CONVERTER

By

Cheong Kun

With the proliferation of miniaturized and autonomous sensors there has been an ever-increasing demand for reconfigurable analog-to-digital converter (ADC) architectures that can efficiently trade-off speed and resolution with its power consumption. This work describes design, implementation and testing of a dynamically reconfigurable ADC, which can be integrated as a smart front-end to low-power sensor. The ADC utilizes the principle of extended data conversion to achieve adaptation between a sigma-delta (ΣΔ) and an algorithmic converter. Based on the input signal statistics, the adaptation can be directly parameterized and used for dynamic reconfigurability to achieve an optimum trade-off. A pseudo-differential ADC architecture has been implemented using switched-capacitor techniques and its functionality has been validated through extensive simulations. A prototype of the ADC has been fabricated using a standard 0.5μm CMOS process and has been tested to be fully functional. The ADC can achieve 8-bit in first-order ΣΔ mode and 4-bit in algorithmic ΣΔ mode and consumes only 8.6μW during operation.

To my parents

Siu-Lai Chow and Yam-Chun Kun

and my fiancée

Jingjing JIANG

ACKNOWLEDGMENTS

I would like to start by giving my sincere appreciation to my advisor, Dr. Andrew Mason. I have been very honored and privileged to have worked under his supervision, and I have benefited a lot from his continuous guidance and from being exposed to his excellent professionalism during my entire graduate study. I am really moved by his kind understanding and generous efforts to help me rush out of school. My earnest appreciation will go to my co-advisor, Dr. Shantanu Chakrabartty, who gave me direction and encouragement throughout the work, without which it will be impossible for me to accomplish so many. I will remember forever those days and nights we worked together in the lab, and his discussions with me, which gave me a glance of the brilliance as an analog circuit designer. I would like to thank Dr. Peixin Zhong, for serving the thesis committee in his busy schedule, and also his FPGA board which played a major role in the testing of my chip.

I sincerely thank Mr. Wilhelm Gattinger, former President and CEO of Siemens Ltd. Hong Kong, Ms. Shih-ying Tan, Ms. Rebecca Tse, and all other colleagues in Siemens AG, who supports my graduate study and providing me the opportunity to start my career.

I would like to thank Mr. Jichun Zhang and Mr. Zhaohui Huang for valuable discussions on analog circuit design techniques. I would like to thank Mr. Junwei Zhou and Mr. Jinwen Xi for their help on digital circuits and Verilog programming. I would like to thank Mr. Amit Gore, for his assistance on setting up the test station. I

would like to thank Mr. Chao Yang and Mr. Yue Huang and for helping me to complete my thesis.

I would like to thank some good friends here in East Lansing, including Lucy Lee, Timothy and Annabelle Chang, Yu Luo, Chuan Lu, Zhiwei Zeng, Jun Yuan, Yuan Fan, Yin Zhan, Xin Liu and Na Yang. They helped me in so many ways, especially during my hard time, that till now I couldn't find a good way to express my appreciation.

My special thanks will go to my fiancée, Jingjing Jiang, whose patience, support, and impeccable understanding allowed me to write this thesis. Finally, warmest thanks must go to my parents, Siu-lai Chow and Yam-chun Kun. Without their love this work would never have come to existence.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1
Introduction

A/D converters are one of the most important elements in modern mixed-signal systems. In many sensor systems, there is tremendous value to dynamically controlling the resolution vs. speed tradeoff to optimize the quality of measurement data and assist in power management.

This chapter first describes the importance of ADC in modern mixed-signal systems. Then the motivation of this work is introduced: to dynamically controlling the resolution vs. speed tradeoff to optimize the quality of measurement data and assist in power management. Next, previous research on reconfigurable ADC architectures are reviewed and a possible solution is given: to design a reconfigurable ADC that can reconfigure between Algorithmic and Sigma-Delta ($\Sigma\Delta$) architectures.

1.1 Mixed-Signal Systems and Data Converters

1.1.1 Mixed-Signal Systems

The world is becoming more digital everyday: Cell phone (digital communication), MP3 (digital audio), digital camera (digital image), DVD (digital video), HDTV, to name just a few. Benefits from the fast development of digital signal processing (DSP) technology during the past decades and Moore's law driven very large scale integration (VLSI) technology, digital system often exhibit lower sensitivity to noise and higher accuracy, offer more robustness and better consistency, allow easier design and test automation, and so on.

However, our real world is analog. All naturally occurring signals in the physical world, such as sound waves, visual images, temperature, pressure, vibrations, and so on, are analog. Furthermore, human beings perceive and retain information in analog form [1]. For an electronic system to interact with the real world, as well as to utilize the powerful DSP technology and computing capability, data acquisition, conversion, and reconstruction circuits must be used as interface of digital processors with the analog world.

Modern electronic systems often convert analog signals to digital signals, perform the processing in digital domain, and convert the result back to analog signals, as illustrated in Figure 1.1. Such kinds of systems are called mixed-signal systems. Examples include wireless communication (GSM mobile phones, CDMA), measuring equipments/instruments (Medical & CCD imaging, digital oscilloscope), consumer electronics (HDTV, digital cameras), and so on.

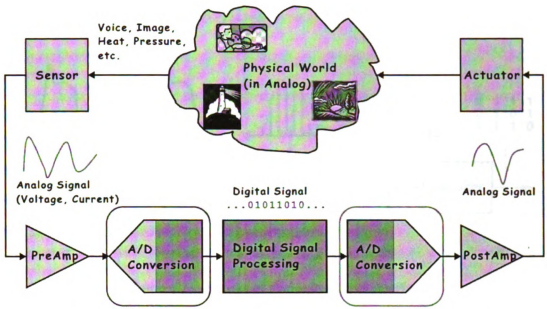


Figure 1.1. Modern mixed-signal system converts analog signals to digital domain in order to utilize the powerful DSP technology.

1.1.2 ADCs and DACs

Among the components in an analog interface, analog-to-digital (A/D) converters (ADCs) and digital-to-analog (D/A) converters (DACs), are key components since they generally define the resolution and bandwidth of the overall system[1][2][3].

The analog-to-digital interface converts a continuous-time, continuous-amplitude analog input to a discrete-time, discrete-amplitude digital signal. Shown in Figure 1.2 is this interface in more detail. First, an analog low-pass filter limits the input signal bandwidth so that subsequent sampling does not "alias" any unwanted noise or signal components into the actual signal band. Next, the filer output is "sampled" so as to produce a discrete-time signal. The amplitude of this waveform is then "quantized," i.e., approximated with a level from a set of fixed reference, thus generating a discrete-amplitude signal. Finally, a digital representation of that level is established at

the output [1].

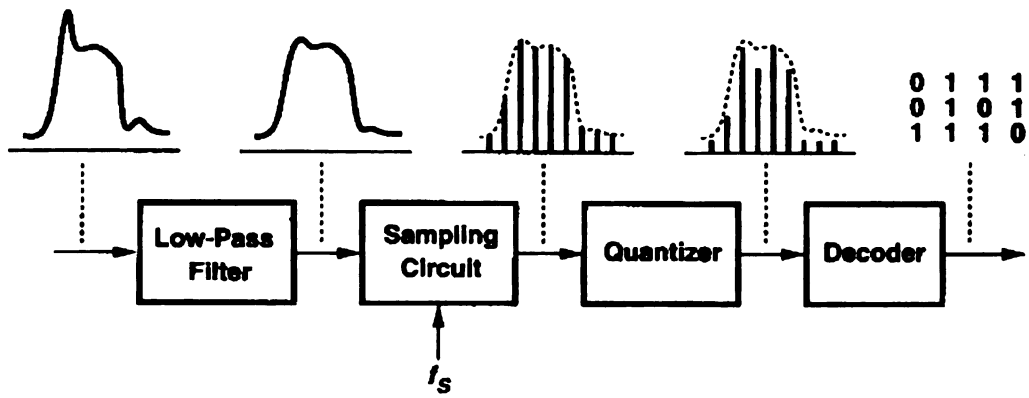


Figure 1.2. Blocks of analog-to-digital interface.

The digital-to-analog interface converts a discrete-time, discrete-amplitude digital input to a continuous-time, continuous-amplitude analog output. This interface is depicted in more detail in Figure 1.3.

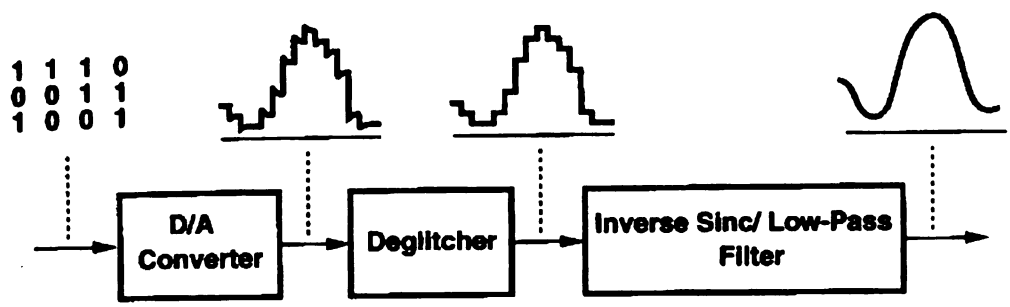


Figure 1.3. Blocks of digital-to-analog interface.

1.1.3 ADC Parameters and ADC Types

The most fundamental parameters for the ADC and DAC are: resolution, speed and power consumption.

- Resolution, normally expressed in number of bits, describes how accurately the ADC or DAC can represent their input analog or digital signals. For example, an

8-bit ADC can distinguish the difference equal to 1/256 of reference voltage.

- Speed, often equivalent as Sampling Rate and expressed in Sample-per- second (Sps), describes how fast is the ADCs or DAC conversion. For example, a 1M Sps ADC can finish one conversion in 1uS.
- Power Consumption, expressed in Watt, describes how much power the ADCs or DACs consume when running. Sometimes we care more about Energy Consumption for each conversion, which is the product of power consumption and conversion time and expressed in J, J-per-conversion, or J-per-bit.

Other ADC parameters include: static parameters: error, gain error, differential non-linearity error (DNL), and integral non-linearity error (INL); dynamic parameters: signal-to-noise ratio (SNR), signal-to-noise and distortion ratio (SINAD), effective number of bits (ENOB), total harmonic distortion (THD), and spurious-free dynamic range (SFDR). They are explained in detail in Appendix A.

A/D converters' performance is generally determined by their structure. A/D converters can be classified into two general groups: Nyquist-rate and oversampled. Nyquist-rate A/D converters can be further divided into several subclasses. Refer to Appendix B for ADC architectures.

1.2 Motivation: Dynamic Reconfigurable ADC

With the proliferation of miniaturized and autonomous sensors there has been an ever-increasing demand for reconfigurable analog-to-digital converter (ADC) architectures that can efficiently trade-off speed and resolution with its power consumption.

For example, in wireless sensor networks [6] environmental monitoring systems or 'listening' devices (acoustic, vibration, etc.), there are opportunity to minimize power consumption when the incoming signal is inactive. Figure 1.4 shows a typical output waveform from a micro gas chromatography (μGC) system[7]. The arriving time of the peak identifies the molecular species present, and the size of the response indicates the amount of gas involved. It can been seen that the signal is under a threshold during the most of time, which allows to monitor the signal with low resolution during the inactive period to save power, and adjust to high-resolution mode when the peaks arrive.

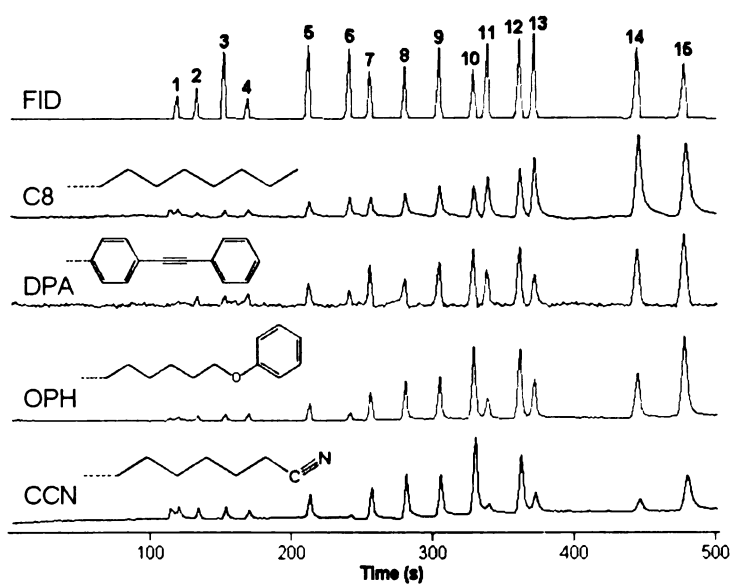


Figure 1.4. Typical signals from a micro gas chromatography (μGC) system.

In some other sensor systems, it is desirable to quickly scan sensor arrays before

measuring a specific element with high resolution For example, in multi-channel neural probes [8] it is desirable to quickly scan many channels at low resolution, as shown in Figure 1.5(a), and then sample signals at high resolution from the most active channels, as shown in Figure 1.5(b).

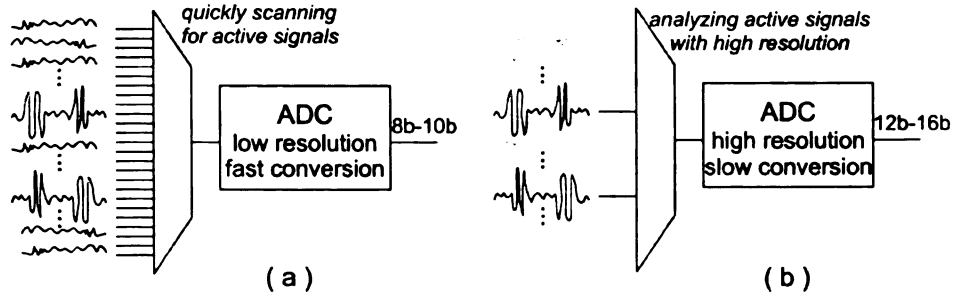


Figure 1.5. Dynamic reconfigurable ADC in multi-channel neural probes applications.

To meet the demands described above, the goal of this thesis is to design an A/D with the following specifications:

- Resolution: configurable, moderate to high, 10-bit ~16-bit;
- Speed: configurable, low to moderate; 10 kHz ~ 100 kHz;
- Power: As low as possible;
- Dynamic reconfigurability of the resolution/speed trade-off to assist sensor applications and power management.

1.3 Direction: Reconfigurable Architecture

1.3.1 Selection of ADC Architectures

For low-power ADCs, the most commonly used architectures are: Successive Approximation Register (SAR) architecture [9], Integrating architecture [10], Algorithmic architecture [11][12], and Sigma-Delta ($\Sigma\Delta$) architecture[13]-[21]. They achieve low power consumption because of their simplicity in circuits [Appendix B].

Algorithmic/SAR ADCs normally find application where the input signal ranges at 100 kHz – 1 MHz and resolution requirement is not critical [9][10][11][12]. Due to the limitations of component mismatch, it is very hard to achieve a resolution higher than 10 bits.

On the other hand, $\Sigma\Delta$ ADCs can easily achieve a resolution higher than 16-bit using over-sampling and noise shaping techniques [13]-[21]. However, because of the large number of clocking cycles required for each conversion, $\Sigma\Delta$ architecture are generally used for low speed applications (less than 100 kHz).

Although each of these A/D architectures is well suited to achieve one of the design goals of this thesis, neither can meet all the requirements. This thesis seeks to explore the feasibility of developing a circuit that combines the benefits of each single architecture into a hybrid structure that achieves all design specifications and can be dynamically reconfigured between the architectures.

1.3.2 Review of Adaptive / Reconfigurable Techniques

The variation of ADC resolution/speed can be accomplished by:

(1) Clocking reconfiguration: adjusting the clock frequency or oversampling ratio (OSR). This technique can be applied to any ADC architecture to increase the speed of ADC. Particularly in ΣΔ architecture, increasing the OSR could increase the resolution.

(2) Circuit parameter reconfiguration: varying circuit parameters, such as size of capacitors, biasing currents of the opamps, and so on.

(3) Architecture reconfiguration: reconfiguring the architecture between architectures. This reconfiguration could possible enjoy the advantage of both architectures and provide more range of control than adjusting clocking or circuit parameters within a single architecture.

Previous research in reconfigurable ADC architectures focus mainly on the pipelined architecture[Appendix B], where reconfigurable networks similar to those in Field Programmable Gate Arrays (FPGA) are used to scale pipeline stage depth or re-group the pipeline stages to optimize resolution and/or speed of a pipelined ADC [22][23][24]. However, pipelined architectures are tailored to high speed (1M-100MHz) applications, and their circuit complexity is not well suited to low-power sensor applications where input signal range rarely exceeds 100kHz.

In 2001, K. Gulati and H.-S. Lee proposed a low-power reconfigurable ADC that can reconfigure its architecture between pipelined and ΣΔ modes [25]. This ADC can achieve a wide range of bandwidth and resolution with adaptive power consumption. However, due to the intrinsic difference between the pipelined and ΣΔ architectures, the system

requires very complicated circuits in order to realize the reconfigurability.

To achieve the goals of this thesis, another approach is needed. Recall that the algorithmic ADC has approximately 8-bit to 12-bit resolution at ~100k Hz while the ΣΔ can easily achieve 12-bit to 16-bit resolution at ~10k Hz. A combination of these two would be well suited to sensor applications. An important observation is that the algorithmic ADC and ΣΔ shares very similar analog components: a multiply-by-two (for algorithmic) or an integrator (for ΣΔ) that can be easily be implemented using the same core in switch-capacitor circuits; a comparator; and a 1-bit DAC. This suggests there should be a smart way to reconfigure between these two architectures. We will show in Chapter 2 that, by using extended counting technique, we can realize the reconfigrability between algorithmic and ΣΔ architectures and enjoy the benefits of both architectures.

1.4 Conclusion

We will build an A/D converter for implantable or wireless sensor applications that will benefit from the dynamic reconfiguration of the resolution/speed tradeoff.

Summary of requirements:

- Resolution: configurable, moderate to high, 10-bit ~16-bit;
- Speed: configurable, low to moderate; 10 kHz ~ 100 kHz;
- Power: As low as possible;
- Dynamic reconfigurability of the resolution/speed trade-off to assist sensor applications and power management.

Possible Solution:

Reconfigure between Algorithmic architecture and $\Sigma\Delta$ architecture

REFERENCES

[1] B. Razavi, *Principles of Data Conversion System Design*, IEEE Press, New York, NY, 1995.

[2] J. Franca and Y. Tsivids, *Design of Analog-to-Digital VLSI Circuits for Telecommunications and Signal Processing*, Chapters 9 & 12, Prentice Hall Inc., Englewood Cliffs, NJ, 1994.

[3] R. Geiger, P. Allen, and N. Strader, *VLSI Design Techniques for Analog and Digital Circuits*, Chapter 8. McGraw-Hill Publishing Inc., New York, NY, 1990.

[4] J. Candy and G. Temes, *Oversampling Delta-Sigma Data Converters: Theory, Design, and Simulation*, IEEE Press, New York, NY, 1992.

[5] S. Norsworthy, R. Schreier, and G. Temes, *Delta-Sigma Data Converters: Theory, Design, and Simulation*, IEEE Press, New York, NY, 1997.

[6] C. Chong and S. Kumar, "Sensor networks: evolution, opportunities, and challenges," *Proc. IEEE*, vol. 91, pp. 1247–1256, Aug. 2003.

[7] K.D. Wise, K. Najafi, R.D. Sacks, and E.T. Zellers, "A wireless integrated microsystem for environmental monitoring," in Digest of Technical Papers, ISSCC, 2004, pp. 434-537.

[8] K. D. Wise, D. J. Anderson, J. F. Hetke, D. R. Kipke, and K. Najafi, "Wireless Implantable MicroSystems: High-density Electronic Interfaces to the Nervous System," *Proc. IEEE*, vol. 92, pp. 76–97, Jan. 2004.

[9] 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, Jul. 2003.

[10] M. Schienle, C. Paulus, A. Frey, F. Hofmann, B. Holzapfl, P. Schindler-Bauer, and R. Thewes, "A fully electronic DNA sensor with 128 positions and in-pixel A/D conversion," *IEEE J. Solid-State Circuits*, vol. 39, pp. 2438 - 2445, Dec. 2004.

[11] P.W. Li, M.J. Chin, P.R. Gray, and R. Castello, "A ratio-independent algorithmic analog-to-digital conversion technique," *IEEE J. Solid-State Circuits*, vol. 19, pp. 828 - 836, Dec. 1984.

[12] Shu-Yuan Chin and Chung-Yu Wu, "A CMOS ratio-independent and gain-insensitive algorithmic analog-to-digital converter," *IEEE J. Solid-State Circuits*, vol. 31, pp. 1201 - 1207, Aug. 1996.

[13] S. Rabii and B. Wooley, "A 1.8V digital-audio sigma-delta modulator in 0.8μm CMOS," *IEEE J. Solid-State Circuits*, vol. SC-32, no. 6, pp. 783-796, Jun. 1997.

[14] E. Fogleman *et al.*, "A 3.3-V single-poly CMOS audio ADC delta-sigma modulator with 98-dB peak SINAD and 105-dB Peak SFDR," *IEEE J. Solid-State Circuits*, vol. SC-35, no. 3, pp. 297-307, Mar. 2000.

[15] E. Zwan and E. Dijkmans, "A 0.2-mW CMOS ΣΔ modulator for speech coding with 80dB dynamic range," *IEEE J. Solid-State Circuits*, vol. SC-31, no. 12, pp. 1873-1880, Dec. 1996.

[16] P. Maulik, N. Bavel, K. Albright, and X. Gong, "An analog/digital interface for cellular telephony," *IEEE J. Solid-State Circuits*, vol. SC-30, no. 3, pp. 201-209, Mar. 1995.

[17] Y. Greets, M. Marques, M. Steyaert, and W. Sansen, "A 3.3-V, 15-bit delta-sigmal ADC with a signal bandwidth of 1.1 MHz for ADSL applications," *IEEE J. Solid-State Circuits*, vol. SC-34, no. 7, pp. 927-936, Jul. 1999.

[18] J. Morizio *et al.*, "14-b 2.2-MS/s sigma-delta ADC," *IEEE J. Solid-State Circuits*, vol. SC-35, no. 6, pp. 968-976, Jul. 2000.

[19] K. Falakshahi, C. Yang, and B. Wooley, "A 14-bit 10-Msamples/s D/A converter using multibit ΣΔ modulation," *IEEE J. Solid-State Circuits*, vol. SC-34, no. 5, pp. 607-615, May 1998.

[20] F. Mederio, B. Perez-Verdu, and A. Rodriguez-Vazquez, "A 13-b 2.2-MS/s, 55-mW, Multi-bit cascaded sigma-delta modulator in CMOS 0.7-μm signal-poly technology," *IEEE J. Solid-State Circuits*, vol. SC-34, no. 6, pp. 748-760, Jun. 1999.

[21] L. Breems, E. van der Zwan, and J. Huijsing, "A 1.8mW CMOS ΣΔ modulator with integrated mixer for A/D conversion of IF signals," *IEEE J. Solid-State Circuits*, vol. SC-35, no. 4, pp. 468-475, Apr. 2000.

[22] Mortezapour, and E.K.F. Lee, "A reconfigurable pipelined data converter," *Proc. IEEE ISCAS 2001*, vol. 4, pp. 314 - 317, May. 2001.

[23] R. Veljanovski, A. Stojcevski, J. Singh, A. Zayegh, and M. Faulkner, "Reconfigurable architecture for UTRA-TDD system," *Electronics Letters*, Vol. 38, pp. 1732 - 1733, Dec. 2002

[24] Hui Liu, and M. Hassoun, "A 9-b 40-MSample/s reconfigurable pipeline analog-to-digital converter," *IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing*, Vol. 49, pp. 449 - 456, July 2002

[25] K. Gulati and H.-S. Lee, "A low-power reconfigurable analog-to-digital converter," in *Proc. ISSCC*, 2001, pp. 54–55.

Chapter 2
Hybrid Algorithmic ΣΔ ADC

Chapter 1 demonstrated that, for wireless and implantable sensor application, there is tremendous value to an A/D converter with dynamic speed vs. resolution reconfigurability. The ΣΔ and algorithmic ADC architectures are candidates for sensor applications because of their high/moderate resolution and moderate/fast conversion speed, and their simplicity in circuit design compared to other ADC architectures.

This chapter first reviews the *extended counting technique*. Then the Hybrid Algorithmic ΣΔ ADC, developed through this thesis research, is introduced to meet reconfigurability and low-power requirements. Finally, behavioral analysis of system reconfigurability, energy-saving benefits and design tradeoffs are presented.

2.1 Review of Extended Counting Technique

In 2001 P. Rombouts, W. D. Wilde and L. Weyten introduced the extended counting technique, which is a compromise between the high accurate but relatively low speed of ΣΔ modulation and higher speed but lower accuracy of algorithmic A/D conversion [1]. The converter successively operates first as a first-order ΣΔ modulator to convert the most significant bits, and then the same hardware is used as an algorithmic converter to convert the remaining least significant bits.

For one A/D conversion, the converter first operates as a resettable 1st-order ΣΔ modulator, as shown in Figure 2.1. `V[i]`, `D[i]` and `Vfb[i]` are respectively the

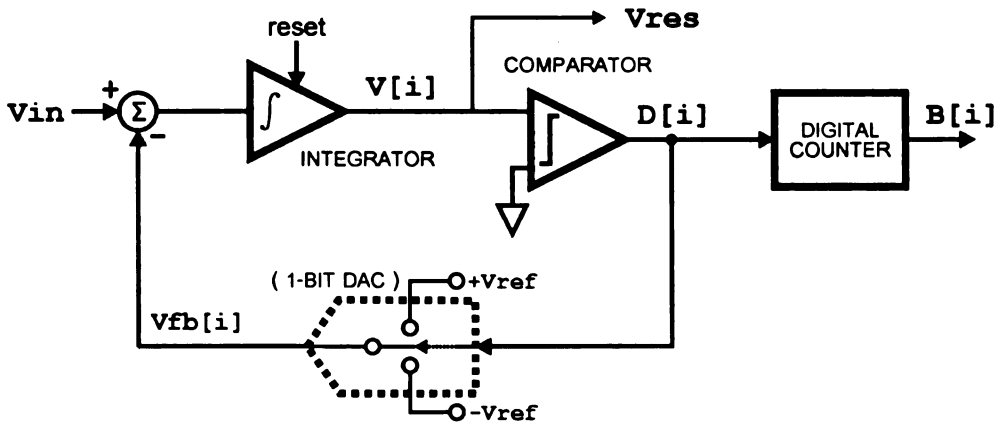


Figure 2.1. Extended Counting ADC in counting conversion: resettable 1st-order ΣΔ.

outputs of the integrator, comparator and the 1-bit DAC . The integrator integrates the difference of its two inputs `Vin` and `Vfb[i]`:

$$V[i+1] = V[i] + (Vin - Vfb[i]). \qquad (2.1\text{-}1)$$

The comparator compares `V[i]` with respect to analog ground and generates the digital code `D[i]`:

$$D[i] = \begin{cases} 1, & if\ V[i] > Agnd \\ -1, & if\ V[i] \le Agnd \end{cases}. \qquad (2.1\text{-}2)$$

The 1-bit DAC takes the output of comparator `D[i]` and convert it back to `+Vref` if `D[i]` is +1 or `-Vref` if `D[i]` is -1:

$$Vfb[i] = \left\{ \begin{matrix} +Vref, & if\ D[i] = 1 \\ -Vref, & if\ D[i] = -1 \end{matrix} \right\} = D[i] \cdot Vref \,. \qquad (2.1\text{-}3)$$

Combining Equation 2.1-1 to 2.1-3 provides the transfer function of the 1st-order ΣΔ modulator:

$$V[i] = V[i-1] + (Vin - D[i-1] \cdot Vref) \,. \qquad (2.1\text{-}4)$$

If the converter is reset before every conversion, thus V[0] = 0 and D[0] = 0, then

$$\begin{aligned} &V[1] = Vin, \\ &V[2] = V[1] + (Vin - D[1] \cdot Vref) = 2 \cdot Vin - D[1] \cdot Vref \\ &V[3] = V[2] + (Vin - D[2] \cdot Vref) = 3 \cdot Vin - (D[1] + D[2]) \cdot Vref \\ &\quad \vdots \end{aligned}$$

If the recursion continues for L steps, then:

$$V[L] = L \cdot Vin - \sum_{i=1}^{L-1} D[i] \cdot Vref \qquad (2.1\text{-}5)$$

It can be proven that V[i] remains bounded [1]:

$$-2 \cdot Vref \le V[i] \le 2 \cdot Vref \qquad (2.1\text{-}6)$$

One additional step is performed to obtain the residue voltage for extended conversion. In this step, the input is `Vagnd` instead of `Vin`. Thus,

$$Vres = V[L] - D[L] \cdot Vref = L \cdot Vin - \sum_{i=1}^{L} D[i] \cdot Vref \qquad (2.1\text{-}7)$$

This mode is called the "counting conversion." It can be proven that Vres is more restrictively bounded [1]:

$$-Vref \le V[i] \le Vref \qquad (2.1\text{-}8)$$

The input Vin can be reconstructed by:

$$Vin = \frac{\sum_{i=1}^{L} D[i] \cdot Vref + Vres}{L} \qquad (2.1\text{-}9)$$

If Vres is known, then Vin can be perfectly reconstructed (assuming there are no errors in D[i]). Furthermore, Vres will be divided by L, which means, the error when we measure Vres will be suppressed by the number of clocks in the counting conversion phase. Thus, a coarse but faster algorithmic A/D conversion can be used to measure the Vres and improve the resolution of the A/D conversion while does not cost too much additional clock steps. This mode is called the "extended conversion", which is shown in Figure 2.2.

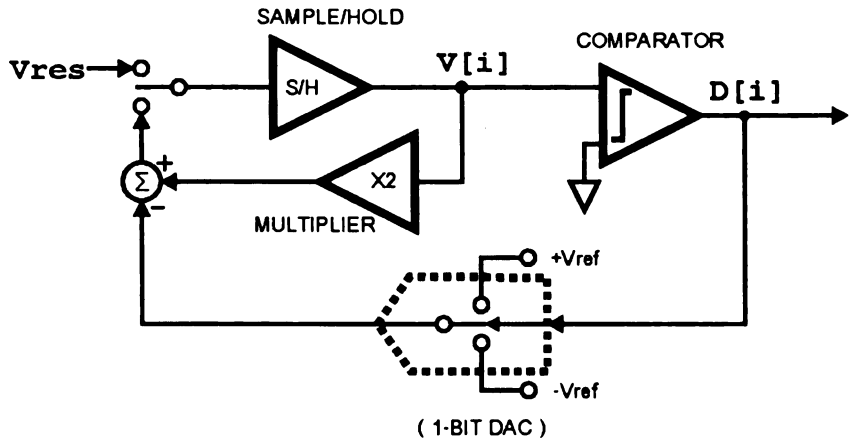


Figure 2.2. Extended Counting ADC in extended conversion: algorithmic mode.

During the extended conversion, the same hardware is used to convert the least significant bits. The integrator is configured as the X2 multiplier and also sample and hold (SaH). The SaH samples Vres in the first step of extended conversion,

$$V[L+1] = Vres \qquad (2.1\text{-}10)$$

and then converts the Vres in standard algorithmic conversion:

$$V[i+1] = 2 \cdot V[i] - D[i] \cdot Vref \qquad for\ i > L \qquad (2.1\text{-}11)$$

After M steps,

$$V[L+M+1] = 2^M \cdot Vres - \sum_{j=1}^{M} 2^{M-j} \cdot D[L+j] \cdot Vref \qquad (2.1\text{-}12)$$

From Equation 2.1-10 to 2.1-11, Vres can be measured as:

$$Vres \approx \sum_{j=1}^{M} 2^{-j} \cdot D[L+j] \cdot Vref \qquad (2.1\text{-}13)$$

Combine Equation 2.1-9 and 2.1-13,

$$Vin = \frac{\sum_{i=1}^{L} D[i] + \sum_{j=1}^{M} 2^{-j} \cdot D[L+j]}{L} \cdot Vref \qquad (2.1\text{-}14)$$

The resolution of extended conversion R and the required number of clock cycles for one conversion N is given by:

$$\begin{aligned} R &= \log_2(L) + M \\ N &= L + M + 1 \end{aligned} \qquad (2\text{-}15)$$

Where L is the number of clocks in counting conversion and M is the number of clocks in extended conversion.

The key ideas behind this extended counting technique are:

I. An algorithmic conversion follows a 1st-order ΣΔ conversion to extend the resolution.

II. The error during the second phase will be suppressed by the number of clock cycles L in the first phase. i.e., the oversampling in the first phase helps the resolution of the second phase too. This is why the second phase could be a coarse conversion. If the second phase is not an algorithmic conversion, but an SAR or other conversion type, it will also work. The algorithmic type is chosen due to the similarity of hardware components between 1st-order ΣΔ and

algorithmic converters; both have a comparator, a 1-bit DAC, a subtractor, and an integrator/x2 multiplier, allowing the same hardware to be reused.

An understanding of the key ideas of the extended counting technique leads to some interesting questions posed below. Exploring these questions and the subsequent expansion of the extended counting technique to a reconfigurable A/D platform: represent the initial innovation of this thesis:

- What if a third phase is added?
- What if the second phase is not algorithmic but remains $\Sigma\Delta$?
- Will the oversampling in the first and second conversion help to suppress the error made in the third conversion or not?

2.2 Introduction to Hybrid Algorithmic ΣΔ

2.2.1 Architecture and Operation

Consider the architecture shown in Figure 2.3. A Sample and Hold (SaH) component and an analog MUX is added to the conventional first-order ΣΔ architecture. By iteratively sample and hold the residue voltage Vres from the integrator, and switch the analog MUX to feed the Vres to the input of the first-order ΣΔ, multiple extended conversions can be achieved.

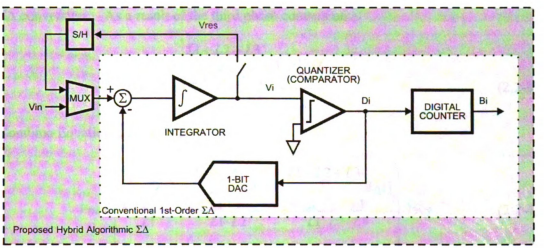


Figure 2.3. Proposed Hybrid Algorithmic ΣΔ ADC architecture that adds a sample and hold component and an analog MUX to a conventional 1st-order ΣΔ architecture.

A three phase extended conversion case is analyzed below to explain the operation of the proposed hybrid algorithmic ΣΔ ADC architecture. During the first phase, the MUX chooses Vin, the system performs L_1 clock cycles of the first-order ΣΔ counting conversion, and the residue V_{res1} is generated and held by the SaH. As a result of the first phase conversion:

$$Vin = \frac{\sum_{i=1}^{L1} D[i] \cdot Vref + Vres1}{L1} \tag{2.2-1}$$

During the second phase, the MUX switches to the output of SAH, which should ideally equal to V_{res1}. The system performs L_2 clock cycles of first-order ΣΔ conversion and generates a second residue V_{res2}, which is also held by the SaH. As a result of the second phase conversion:

$$Vres1 = \frac{\sum_{i=L1+1}^{L1+L2} D[i] \cdot Vref + Vres2}{L2} \tag{2.2-2}$$

During the third phase, the same circuit is reused to perform L_3 clock cycles of first-order ΣΔ conversion. As a result of the third phase conversion:

$$Vres2 \approx \frac{\sum_{i=L1+L2+1}^{L1+L2+L3} D[i] \cdot Vref}{L3}. \tag{2.2-3}$$

Combine Equation 2.2-1 to 2.2-3,

$$Vin \approx \left(\frac{\sum_{i=1}^{L1} D[i]}{L1} + \frac{\sum_{i=L1+1}^{L1+L2} D[i]}{L1 \cdot L2} + \frac{\sum_{i=L1+L2+1}^{L1+L2+L3} D[i]}{L1 \cdot L2 \cdot L3} \right) \cdot Vref \tag{2.2-4}$$

Error Suppression:

To analyze the ability of this circuit on suppressing errors, assume V_{err1}, V_{err2} represent error signals when measuring V_{res1} in the second phase and V_{res2} in the third phase, respectively. Thus,

$$Vres1 = \frac{\sum_{i=L1+1}^{L1+L2} D[i]\cdot Vref + Vres2}{L2} + Verr1,$$
$$Vres2 = \frac{\sum_{i=L1+L2+1}^{L1+L2+L3} D[i]\cdot Vref}{L3} + Verr2 \qquad (2.2\text{-}5)$$

Take Equation 2.2-5 into Equation 2.2-1, the signal *Vin* can now be expressed as:

$$Vin = \left(\frac{\sum_{i=1}^{L1} D[i]}{L1} + \frac{\sum_{i=L1+1}^{L1+L2} D[i]}{L1\cdot L2} + \frac{\sum_{i=L1+L2+1}^{L1+L2+L3} D[i]}{L1\cdot L2\cdot L3} \right) \cdot Vref + \frac{Verr1}{L1} + \frac{Verr2}{L1\cdot L2} \qquad (2.2\text{-}6)$$

Equation 2.2-6 shows that V_{err1} is suppressed by a factor of L_1 (as in basic extended counting conversion) and that V_{err2} is suppressed by $(L_1 \cdot L_2)$, the product of counting numbers in the first and second phase, meaning in a three phase extended conversion case, the error during the third phase will be suppress by the number of clocks in the second phase AND the first phase.

Resolution

If there is no error or the error is limited to an acceptable level, the third phase will generate V_{res2} with a resolution (in bits) defined by $\log_2(L_3)$. Together with the second phase, V_{res1} has $(\log_2(L_2) + \log_2(L_3))$ bits of resolution, and together with the first phase, Vin has $(\log_2(L_1) + \log_2(L_2) + \log_2(L_3))$ bits of resolution.. Thus, the optimal resolution of the three phase example is:

$$R = \log_2(L_1) + \log_2(L_2) + \log_2(L_3) \qquad (2.2\text{-}7)$$

Speed:

In the first phase, 1 clock of integrator reset and L_1 clocks of counting is required. The final step generates V_{res1}, which combines with the integrator reset step for the second phase. Thus the second phase requires only L_2 clocks of counting. In the same way the third phase requires L_3 clocks. Thus, the total clock cycles required, N, is:

$$N = 1 + L_1 + L_2 + L_3 \tag{2.2-8}$$

Summary

This analysis shows that all the beneficial properties of the extended counting technique are retained. Additionally, this modified architecture has some significant benefits over traditional extended counting, including capability for dynamic reconfigurability, as discussed later in more detail.

2.2.2 Multiple Extended Conversion

The analysis above describes a three phase extended conversion technique. This can be expanded to an arbitrary number of phases, M. This multiple-phase extended conversion concept is described by:

$$Vin = \frac{\sum_{i=1}^{L1} D[i]\cdot Vref + Vres1}{L1}, \quad Vres1 = \frac{\sum_{i=L1+1}^{L1+L2} D[i]\cdot Vref + Vres2}{L2} \quad \cdots \tag{2.2-9}$$

$$Vres_{(M-1)} = \frac{\sum_{(i=L1+L2+\cdots+L_{M-1})+1}^{(i=L1+L2+\cdots+L_{M-1})+L_M} D[i]\cdot Vref + Vres_M}{L_M} \tag{2.2-10}$$

$$Vin \approx \left(\frac{\sum_{i=1}^{L1} D[i]}{L1} + \frac{\sum_{i=L1+1}^{L1+L2} D[i]}{L1 \cdot L2} + \cdots + \frac{\sum_{i=L1+\cdots+L_{M-1}+1}^{(i=L1+\cdots+L_M} D[i]}{L1 \cdot L2 \cdots \cdot L_M} \right) \cdot Vref \quad (2.2\text{-}11)$$

$$R = \log_2(L1) + \log_2(L2) + L + \log_2(L_M) = \sum_{i=1}^{M} \log_2(L_i) \quad (2.2\text{-}12)$$

$$N = 1 + L_1 + L_2 + \cdots + L_M = 1 + \sum_{i=1}^{M} L_i \quad (2.2\text{-}13)$$

where all variables are as previously defined.

If the number of clock cycles in each phase are equal, i.e., $L_1 = L_2 = \cdots = L_M = L$, then:

$$Vin = \frac{\sum_{i=1}^{L} D[i] \cdot Vref + Vres1}{L}, \quad Vres1 = \frac{\sum_{i=L+1}^{2L} D[i] \cdot Vref + Vres2}{L} \cdots \quad (2.2\text{-}14)$$

$$Vres_{(M-1)} = \frac{\sum_{i=(M-1)\cdot L+1}^{M \cdot L} D[i] \cdot Vref + Vres_M}{L} \quad (2.2\text{-}15)$$

$$Vin = \left(\sum_{j=1}^{M} L^{-j} \sum_{i=(j-1)L+1}^{jL} D[i] \right) \cdot Vref \quad (2.2\text{-}16)$$

$$R = M \cdot \log_2(L) \quad (2.2\text{-}17)$$

$$N = 1 + M \cdot L \quad (2.2\text{-}18)$$

Equation 2.2-17 and 2.2-18 implies the resolution/speed reconfigurability capability of the Hybrid Algorithmic-ΣΔ architecture, which will be addressed in detail in Section 2.3.2.

2.2.3 Hybrid Algorithmic ΣΔ

The multiple-phase extended conversion technique introduced by this thesis work has been termed Hybrid Algorithmic ΣΔ conversion. Although there are no algorithmic conversions in any phase, the rational for this terminology is discussed below:

1. The reconstructed equation of Hybrid Algorithmic ΣΔ, Equation 2.2-16,

$$Vin = \left(\sum_{j=1}^{M} L^{-j} \sum_{i=(j-1)L+1}^{jL} D[i] \right) \cdot Vref \qquad (2.2\text{-}16)$$

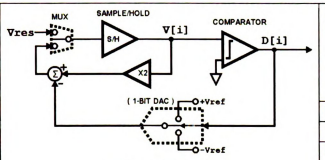
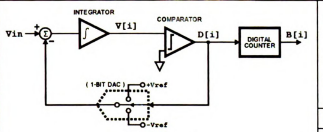
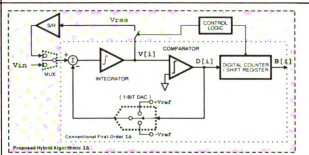
consists of two nested summations. The inside part $\sum_{i=(j-1)L+1}^{jL} D[i]$ is exactly the same as in first-order ΣΔ, while the outside part $\sum_{j=1}^{M} L^{-j}$ is very similar to the $\sum_{i=1}^{M} 2^{-i} \cdot D[i]$ function of an algorithmic ADC. The difference is, in algorithmic conversion, each cycle converts 1 bit resulting in the 2^{-i} term, while in Hybrid Algorithmic ΣΔ conversion the number of bits converted during each cycle/phase is not necessary 1, but ($\log_2$L). Thus the $\sum_{j=1}^{M} L^{-j}$ term appears. If we choose L=2, which means each ΣΔ conversion only converts 1 bit, then the behavior of Hybrid Algorithmic ΣΔ conversion will be the same as an algorithmic conversion.

2. From a coding point of view, the digital output codes of a Hybrid Algorithmic ΣΔ conversion is **bit-weight encoded,** as is an algorithmic or SAR conversion. Each output code is bit weight encoded: the first output D[1] is divided by 2, the second output D[2] is divided by 2^2, and so on, such that. $Vin = \sum_{i=1}^{M} 2^{-i} \cdot D[i] \cdot Vref$. In other words, the output code is binary bit-weight encoded. In comparison, all the digital outputs of a first-order ΣΔ conversion are equally weighted. When reconstructing the

output, all the digital outputs are indistinguishably summed up and divided by the total number of clock cycles in one conversion: $Vin = \frac{\sum_{i=1}^{L} D[i]}{L} \cdot Vref$. This is referred to as **pulse width density (PWD) encoded**. In the Hybrid Algorithmic ΣΔ conversion, the output from the 1st conversion phase is first summed up and then divided by L, the output from the 2nd phase is summed up and divided by L^2, and so on. Thus, the output within one conversion phase is PWD encoded, while the output between conversion phases are bit-weight encoded (the output from the 1st phase has the most significance, and the output from the last phase has the least significance). The output code shows weight encoded characteristic because bit weight information is encoded into the system by feeding back the residue from between phases.

3. From component architecture point of view, the core of Hybrid Algorithmic ΣΔ is the same as a first-order ΣΔ conversion, as shown in Table 2.1. However, in adding a SaH component and an analog MUX to the front end, the structure is very similar to the SaH and MUX in Algorithmic conversion.

TABLE 2.1. Comparison of Algorithmic, First-order ΣΔ, and Hybrid Algorithmic ΣΔ architectures.

Algorithmic	 (refer to Figure 2.2)	**Data-Reconstruct Equation:** $Vin = \sum_{i=1}^{M} 2^{-i} \cdot D[i] \cdot Vref$ **Encoding: Binary Bit-Weight** **N=M** **R=M**
Sigma-Delta (ΣΔ)	 (refer to Figure 2.1)	**Data-Reconstruct Equation:** $Vin = \frac{\sum_{i=1}^{L} D[i]}{L} \cdot Vref$ **Encoding: PWD** **N=L** **R=$\log_2$(L)**
Hybrid Algorithmic ΣΔ	 (refer to Figure 2.3)	**Data-Reconstruct Equation:** $V_{in} = \left(\sum_{j=1}^{M} L^{-j} \sum_{i=(j-1)L+1}^{jL} D[i] \right) \cdot V_{ref}$ **Encoding: Combined PWD and Big-Weight** **N=1+M·L** **R= M· $\log_2$(L)**

2.2.4 Summary

The new Hybrid Algorithmic ΣΔ conversion essentially has **two conversion loops**: an **inner first-order ΣΔ conversion loop**, which is characterized by the number of counting steps L in each phase, and an **outer algorithmic conversion loop**, which is characterized by the number of phases M, or the number of times of residue feed-back (M-1).

If L is large and M is kept small, which means longer counting conversion and fewer residue feed-backs, the system behaves more ΣΔ-like, with high resolution but low speed. The number of clock cycles increases ***exponentially*** with respect to resolution. Notice that when M=1, there is only one conversion phase, and the architecture degenerates into first-order ΣΔ conversion.

If L is small and M is large, which means less counting but frequently residue feed-back, the system behavior will is more Algorithmic-like, with relatively low resolution but high speed. The total number of clocks increases ***linearly*** with respect to resolution. Notice that when L=2, during each phase only 1 bit is converted and the system behavior becomes very similar to a conventional algorithmic conversion.

M and L is fully determined by the clocking scheme: how many steps of counting and when to feed-back the residue. By adjusting M and L through different clock schemes, the system can be dynamically **reconfigured** at the **architectural level** to be more ΣΔ-like or more algorithmic-like and is thus easily tailored in real time to application demands.

2.3 Analysis of Hybrid Algorithmic ΣΔ Architecture

2.3.1 Behavioral Verification

Behavior of the Hybrid Algorithmic ΣΔ conversion is simulated and verified in MATLAB. Figure 2.4 shows a case where 12-bit is achieved in 49 clocks by setting L=16 and M=3 (3 phases and 4-bit each phase). The input is a DC voltage equals to $(0.6 \cdot Vref)$; the triangle wave shows the output of the integrator and the square wave shows the comparator output. Within each phase the waveform is exactly similar as first-order ΣΔ conversion. At the end of each phase the residue is generated, sampled and fed as input to the next phase.

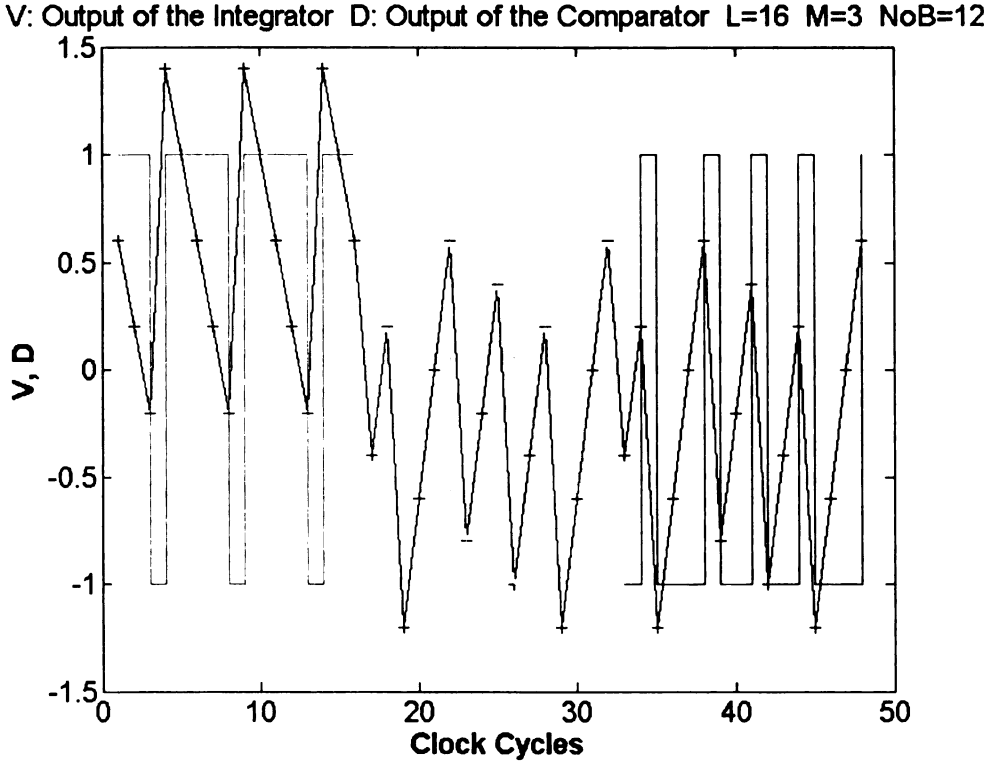


Figure 2.4. Behavioral simulation of the Hybrid Algorithmic ΣΔ conversion. A 12-bit conversion is achieved in 49 clocks by setting L=16 and M=3.

To check if the output digital code represents the input, a sine wave is applied at the

input and output digital code is reconstructed back to analog voltages. As shown in Figure 2.5, the reconstructed voltage tracks the input voltage, verifies the function of the Hybrid Algorithmic ΣΔ conversion.

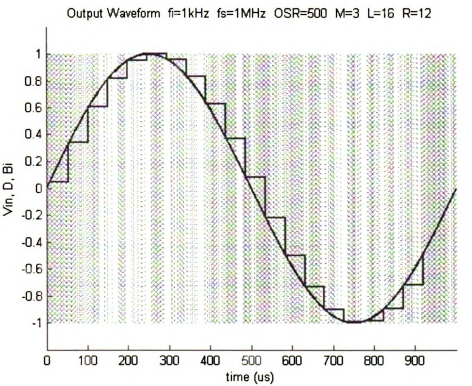


Figure 2.5. Behavioral simulation of the Hybrid Algorithmic ΣΔ conversion.

2.3.2 Resolution/Speed Configurability

The optimal resolution R for the Hybrid Algorithmic ΣΔ conversion are given by:

$$R = M \cdot \log_2(L) \tag{2.2-17}$$

At one specific resolution R, two parameters M and L are adjustable, providing a wider range of configurability than is available in other conventional ADC architectures. To illustrate the flexibility of configuration of M and L, consider a 12-bit Hybrid Algorithmic ΣΔ conversion as an example. As shown in Figure 2.6, the 12-bit conversion can be done

in one phase using first-order ΣΔ conversion by configuring M=1 and $\log_2 L=12$, as in (a); or be divide into two phases, 6-bit + 6-bit by making M=2 and $\log_2 L=6$, as in (b). It can also be divided into three phases, 4-bit + 4-bit + 4-bit, by making M=3 and $\log_2 L=4$, as in (c), or in other configurations shown in Figure 2.6. Furthermore, as derived in section 2.2.2, the Hybrid Algorithmic ΣΔ circuit does not necessarily have equal clock cycles in each phase, which means the 12-bit conversion can be implemented as for example 3-bit + 3-bit + 2-bit + 2-bit + 2-bit, as shown in (g), or other combinations of bit-per-phase and phases, using this very flexible architecture.

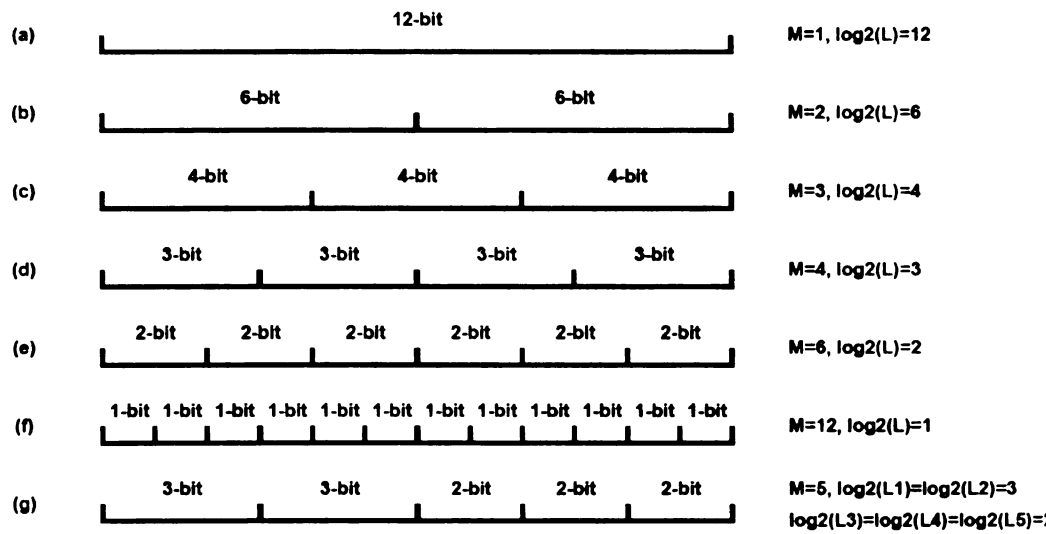


Figure 2.6. Different configurations for a 12-bit Hybrid Algorithmic ΣΔ conversion.

The total number of clocks needed N for the Hybrid Algorithmic ΣΔ conversion is given by:

$$N = 1 + M \cdot L \tag{2.2-18}$$

With the wide range of M and L configurations, we can get different kinds of resolution versus speed combination. Figure 2.7 shows the possible resolution and speed combinations when M and $\log_2(L)$ varies from 1 to 16 respectively.

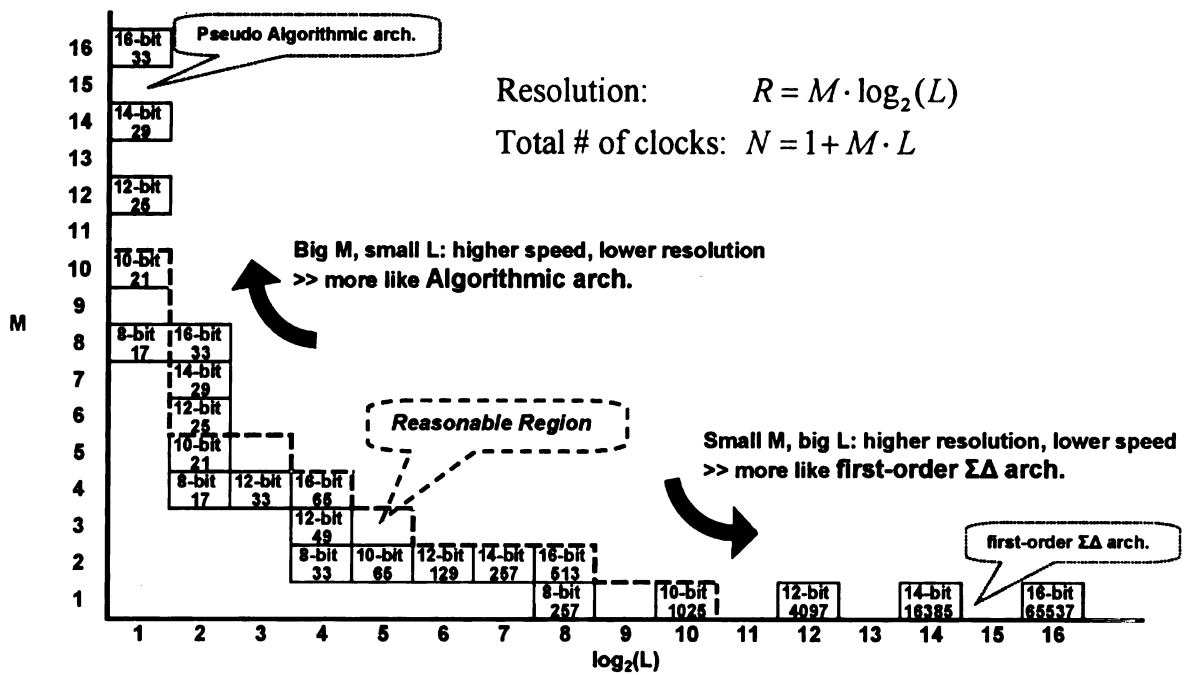


Figure 2.7. The ADC can be reconfigured to different speed vs. resolution combinations by varying M and L.

2.3.3 Benefits in Energy Saving

As described in Chapter 1, power or energy consumption is a key circuit design constraint for implantable or portable devices. Hybrid Algorithmic ΣΔ converter architecture provides the opportunity of trading off resolution and speed with power consumption according to signal activity or channel activities.

The energy consumed in one conversion is given by:

$$E = (P_{Ana\log} + P_{Digital}) \cdot T_{conv} \tag{2.3-1}$$

where $P_{Ana\log}$ and $P_{Digital}$ stands for analog and digital power consumption respectively. $P_{Ana\log}$ is the product of supply voltage and tail current:

$$P_{Ana\log} = V_{DD} \cdot I_D \tag{2.3-2}$$

Thus $P_{Ana\log}$ is fixed once V_{DD} is determined and the biasing is fixed (thus I_D fixed). $P_{Digital}$ consist of two parts:

$$P_{Digital} = P_{static} + P_{dynamic} \tag{2.3-3}$$

P_{static} is the static power consumption of digital circuits. If the digital circuits are all static CMOS circuit, P_{static} can be ignored. $P_{dynamic}$, the dynamic power of digital circuit, can be estimated by:

$$P_{dynamic} = \alpha \cdot C_L \cdot V_{DD}^2 \cdot f_{CLK} \tag{2.3-4}$$

in which α is switching activity and C_L is the average load capacitance. α is fixed once the algorithm is fixed. C_L is fixed once the circuit is determined.

T_{conv} is the conversion time, and is the product of clock cycle and number of clocks in one conversion:

$$T_{conv} = t_{CLK} \cdot N = \frac{1}{f_{CLK}} \cdot N \tag{2.3-5}$$

Taking Equation 2.3-2, 2.3-3, 2.3-4, 2.3-5 into Equation 2.3-1, we get:

$$\begin{aligned} E &= (V_{DD} \cdot I_D + \alpha \cdot C_L \cdot {V_{DD}}^2 \cdot f_{CLK}) \cdot \frac{1}{f_{CLK}} \cdot N \\ &= (\frac{V_{DD} \cdot I_D}{f_{CLK}} + \alpha \cdot C_L \cdot {V_{DD}}^2) \cdot N \end{aligned} \tag{2.3-6}$$

If only the case of fixed clock rate f_{CLK} is considered, then the energy consumption for each conversion is directly proportional to the number of clocks required for each conversion.

For Hybrid Algorithmic ΣΔ converter architecture, the number of needed clocks N for R-bit resolution is given by Equation 2.2-17 and 2.2-18, as listed below:

$$R = M \cdot \log_2(L) \tag{2.2-17}$$

$$N = 1 + M \cdot L \tag{2.2-18}$$

The flexibility of the combinations of L and M shown in Figure 2.7 provide the possibility of energy saving. Figure 2.8 shows the normalized energy consumption of the Hybrid Algorithmic ΣΔ architecture at different L and M configurations. It can been seen from Figure 2.8, and also be proven from Equation 2.2-17 and 2.2-18, that increasing M, the number of conversion phases, will reduce N, the total number of clocks required, and thus energy consumption for each conversion. It matches the fact that with larger M, the architecture is more like algorithmic architecture, and achieves lower energy consumption. In best case, L=2 and M=R, meaning an R-bit conversion is completed by 1-bit each phase and R phases, Hybrid Algorithmic ΣΔ architecture degrades to pseudo algorithmic architecture, the total number of clocks becomes

$$N = 1 + 2R \tag{2.3-7}$$

approximately twice of the clocks needed for real algorithmic conversion.

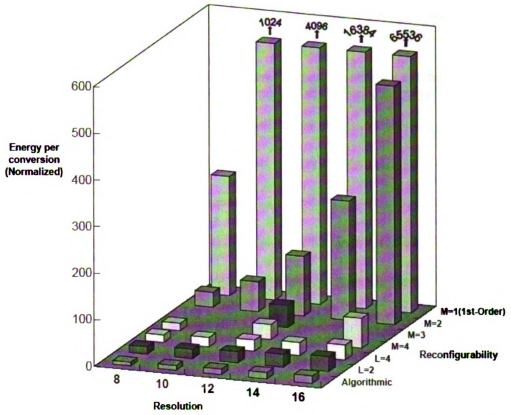


Figure 2.8. Normalized energy consumption for the Hybrid Algorithmic ΣΔ architecture at different configurations.

REFERENCES

[1] P. Rombouts,W. DeWilde, 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.

[2] J. D. Maeyer, P. Rombouts, and L. Weyten, "A double-sampling extended-counting ADC," *IEEE J. Solid-State Circuits*, vol. 39, pp. 411 - 418, Mar. 2004.

Chapter 3
Circuit Implementation

A Pseudo-differential circuit implementation of the Hybrid Algorithmic ΣΔ A/D architecture described in chapter 2 was designed and fabricated using standard 0.5μm CMOS process available through MOSIS. The circuits were implemented with switched-capacitor topology and correlated-double sampling techniques were used to overcome the effects systematic biases in the circuits.

3.0 Process parameters

The standard 0.5μm CMOS process available through MOSIS [1][2][3] has a Lambda of 0.3μm and allows operation at 3V~5V supply voltage. It provides 3 metal layers and 2 poly layers, and a high resistance layer. PiP (poly2 over poly) capacitors (950 aF/μm^2) and the HRP (High Resistance) option are available. Some basic process parameters for hand calculation are summarized in Table 3.1. The SPICE corner parameters and the electrical parameters extracted by MOSIS for the fabrication run is summarized in Appendix D.

Table 3.1 Approximate Process Data for 0.5μm CMOS process

$L_{min} = 0.6\ \mu m$	$t_{ox} = 140$ Å
$\mu_n C_{ox}' = 120\ \mu A/V^2$	$\mu_p C_{ox}' = 40\ \mu A/V^2$
$V_{T0n} = 0.7$ V	$V_{T0p} = -0.9$ V
$\lambda = 0.06\ V^{-1}$	
$\gamma_n = 0.47\ V^{1/2}$	$\gamma_p = 0.58\ V^{1/2}$
$\Phi_{Fn} = 0.38$ V	$\Phi_{Fp} = 0.39$ V
$\kappa = 0.7$	

3.1 Circuit Design and Simulation

The circuit components, such as the integrator, the comparator and the sample/hold were designed using switched-capacitor circuit techniques [6]. Correlated Double Sampling (CDS) technique was used for the sample/hold circuit. The system level simulations verify the functionality of the circuit. A critical component of the implementation is a cascoded inverter which is used instead of conventional operational amplifiers to achieve low-power operation.

3.1.1 Cascoded Inverter Amplifier

A cascoded inverter, as shown in Figure 3.1, was designed to act as a high gain amplifier. M1 and M2 are current source load. M4 is a common source amplifier and M3 is a common base amplifier.

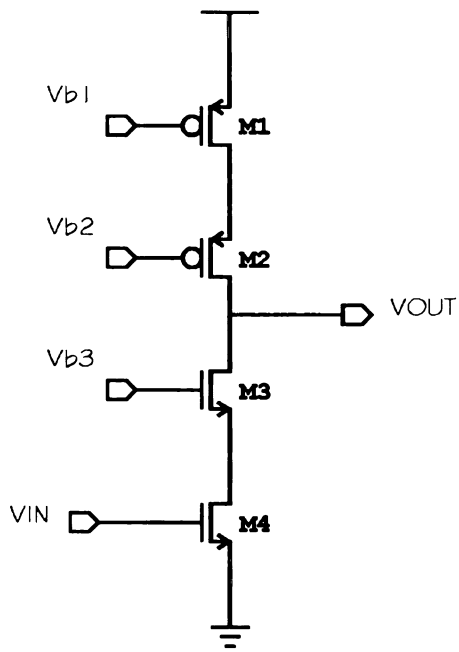


Figure 3.1. Schematic for the cascoded inverter amplifier.

The transistor sizes and biasing voltages of the cascoded inverter were carefully

chosen in order to achieve a high gain and moderate speed with a reasonably power consumption. With a size of W/L = 30μ/6μ for M1 and M2, W/L = 20μ/6μ for M3 and M4, and biasing of V_{DD} = 3V, V_{b1} = 1.9V, V_{b2} = 1.7V, and V_{b3} = 1.4V, simulation shows that the CIA has a gain of 57dB around the middle point where V_{IN} = V_{OUT} = V_{MID} = 828mV. Figure 3.2 shows the operation of the cascoded inverter amplifier. A drain current Id of 2.54 μA leads to a power consumption of less than 8μW.

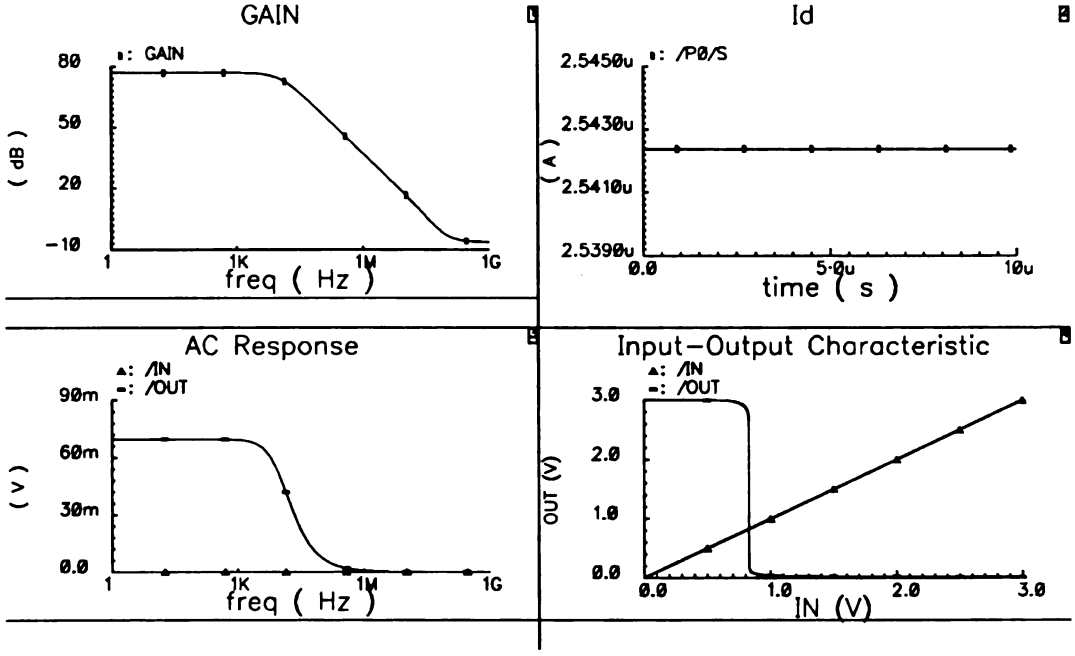


Figure 3.2. Simulation results for the cascoded inverter amplifier.

Compared to conventional operational amplifiers, this cascoded inverter amplifier has advantages of simple structure, low power and compact size. The disadvantages include relatively lower gain and lower dynamic range. The lower dynamic range results from asymmetrical position of V_{MID} relative to the rail voltages. This is unlike the case in traditional op-amp based designs where virtual ground can be dynamically adjusted. For a cascoded inverter V_{MID} is fixed by transistor size and biasing. V_{MID} eventually determines the analog virtual ground of the system. However, in switched-capacitor

circuits, biasing the input at V_{MID} is automatically achieved by shorting the input and output during circuit reset as discussed below.

3.1.2 Integrator

A cascoded-inverter based switched-capacitor integrator, as shown in Figure 3.3, was designed and simulated. CMOS switches are used in order to reduce the on-resistance and balance clock feed through. C1=C2=100fF. S_1 and S_2 are none-overlapping clocks.

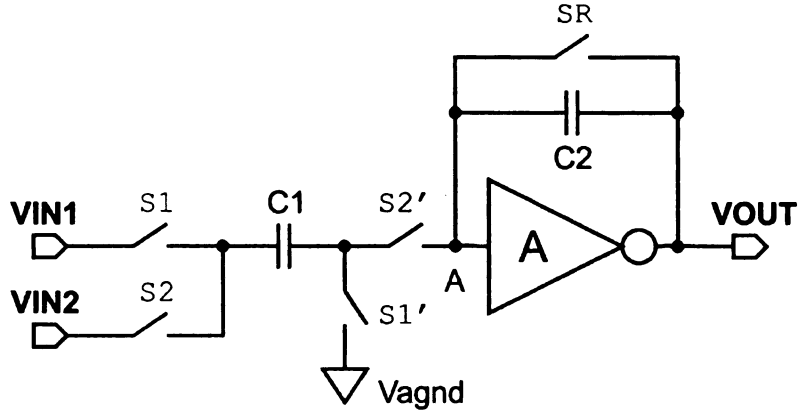


Figure 3.3. Schematic of the inverter based switched-capacitor integrator.

The circuit operates as follows: In the initial step, S_R is closed to set the voltage on node A, the input to the cascoded inverter amplifier, to V_{MID} where the gain is maximum. Then S_R is opened, validating the operation of the integrator. During phase I (sample phase), S_1, S_1' are closed while S_2, S_2' are open. $V_{IN1}[n]$ is applied to C1 storing charge amount = $(V_{MID}-V_{IN1}[n])\cdot C1$ on the right plate of C1. Assuming at this time V_{OUT} = $V_{OUT}[n]$, the charge on the left plate of C2 is $(V_{MID}-V_{OUT}[n])\cdot C2$. During phase II (integrate phase), S_1, S_1' are open while S_2, S_2' are closed. $V_{IN2}[n+1/2]$ is applied at C1 forcing the charge on the right plate of C1 to be $(V_{MID}-V_{IN2}[n+1/2])\cdot C1$. Denoting V_{OUT} = $V_{OUT}[n+1]$ at this time, then charge on the left plate of C2 is $(V_{MID}-V_{OUT}[n+1])\cdot C2$.

Table 3.2 summarizes the operation of the inverter based integrator.

Table 3.2 Operation of the inverter based integrator

Phase	S_R	S_1	S_2	Vol. on C1	Vol. on C2
Initial	1	0	0	N/A	0
Sample	0	1	0	V_{MID} -V_{IN1} [n]	V_{MID} -V_{OUT} [n]
Integrate	0	0	1	V_{MID} -V_{IN2} [n+1/2]	V_{MID} –V_{OUT} [n+1/2]

Appling charge conservation at node A:

$$-V_{IN1}[n]\cdot C1 - V_{OUT}[n]\cdot C2 = -V_{IN2}[n+1/2]\cdot C1 - V_{OUT}[n+1]\cdot C2 \qquad (3.1\text{-}1)$$

Thus,

$$V_{OUT}[n+1] = V_{OUT}[n] + (V_{IN1}[n] - V_{IN2}[n+1/2])\cdot C1/C2 \qquad (3.1\text{-}2)$$

During operation, node A should be kept at V_{MID} = 828mV all the time, otherwise the gain of the CIA will greatly decrease. Figure 3.4 shows the simulation clocking and waveforms of the integrator. 200mV is integrated in each clock.

Figure 3.5 demonstrates the linearity and large dynamic range achieved by the integrator when its output integrated from 0V to 3V. It also compares the result with an ideal integrator, as shown by the dashed line. The simulations results indicate that the circuit can achieve linearity of > 10 bits with a dynamic range of 0.3V-2.5V.

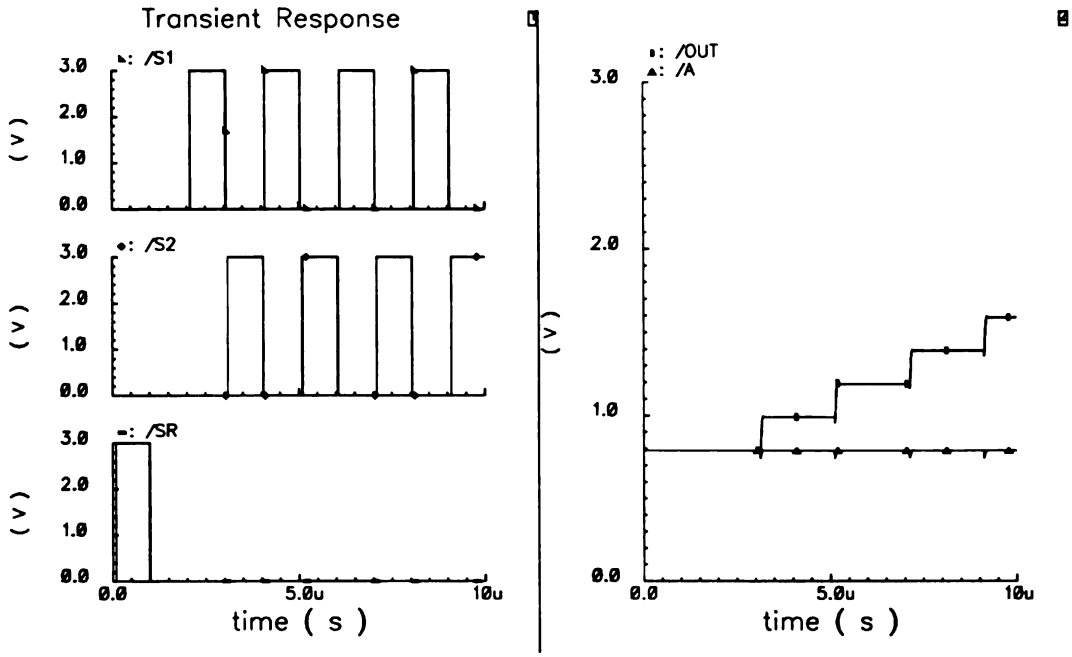


Figure 3.4. Operation of the inverter-based integrator: integrating 200mV each clock.

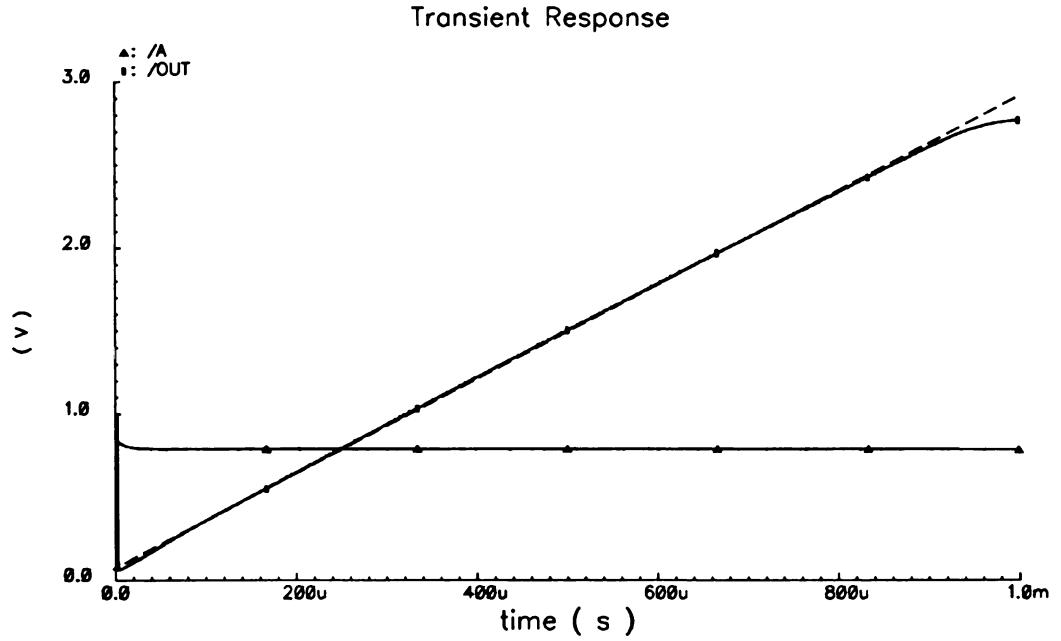


Figure 3.5. Simulation result shows the linearity of the inverter-based integrator.

3.1.3 Comparator

Figure 3.6 shows the schematic for a basic inverter based switched-capacitor comparator circuit. I0 and I1 are cascoded inverter amplifiers, whose inverting point is at $V_{MID} \approx 800mV$. I2 is a normal CMOS inverter, whose inverting point is at $V_{DD}/2 \approx 1.5V$. A D Flip-Flop is added at the output to latch the comparison result.

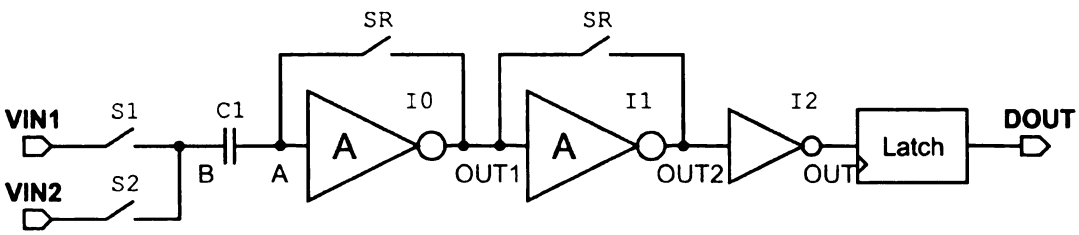


Figure 3.6. Schematic of a basic switched-capacitor comparator.

During phase I (sample phase), S_1 and S_R are closed, setting node A, OUT1 and OUT2 to V_{MID}, and storing charge $(V_{IN1}[n]-V_{MID})\cdot C1$ on capacitor C1. Then in phase II (compare phase), S_1 and S_R are open and S_2 are closed. The voltage on the node B becomes $V_{IN2}[n+1/2]$. Assume the voltage on node A becomes V_A. Since S_R is open, charge conservation on C1 gives:

$$(V_{IN1}[n]-V_{MID})\cdot C1 = (V_{IN2}[n+1/2]-Vx)\cdot C1 \qquad (3.1\text{-}3)$$

Then,

$$V_A = V_{MID} + (V_{IN2}[n+1/2]-V_{IN1}[n]). \qquad (3.1\text{-}4)$$

If $V_{IN1}[n] < V_{IN2}[n+1/2]$, V_A is higher than V_{MID}, which will pull down V_{OUT1}, thus pulling up V_{OUT2}, then pulling down V_{OUT} and $D_{OUT} = 0$.

If $V_{IN1}[n] > V_{IN2}[n+1/2]$, V_A is lower than V_{MID}, which will pull up V_{OUT1}, thus pulling down V_{OUT2}, then pulling up V_{OUT} and $D_{OUT} = 1$.

Table 3.3 summarizes the operation of the basic inverter based comparator.

Table 3.3 Operation of the inverter based comparator

Phase	S_R	S_1	S_2	Vol. on node B	Vol. on node A
Sample	1	1	0	$V_{IN1}[n]$	V_{MID}
Compare	0	0	1	$V_{IN2}[n+1/2]$	$V_{MID}+(V_{OUT}[n+1/2]-V_{IN1}[n])$

The problem with this basic inverter based switched-capacitor comparator is that it compares two voltage from two different times, $V_{IN1}[n]$ and $V_{IN2}[n+1/2]$. To compare two voltages at the same time instant, some modifications to the topology are necessary. Figure 3.7 shows the schematic of a differential comparator that eliminates the time discrepancy [7].

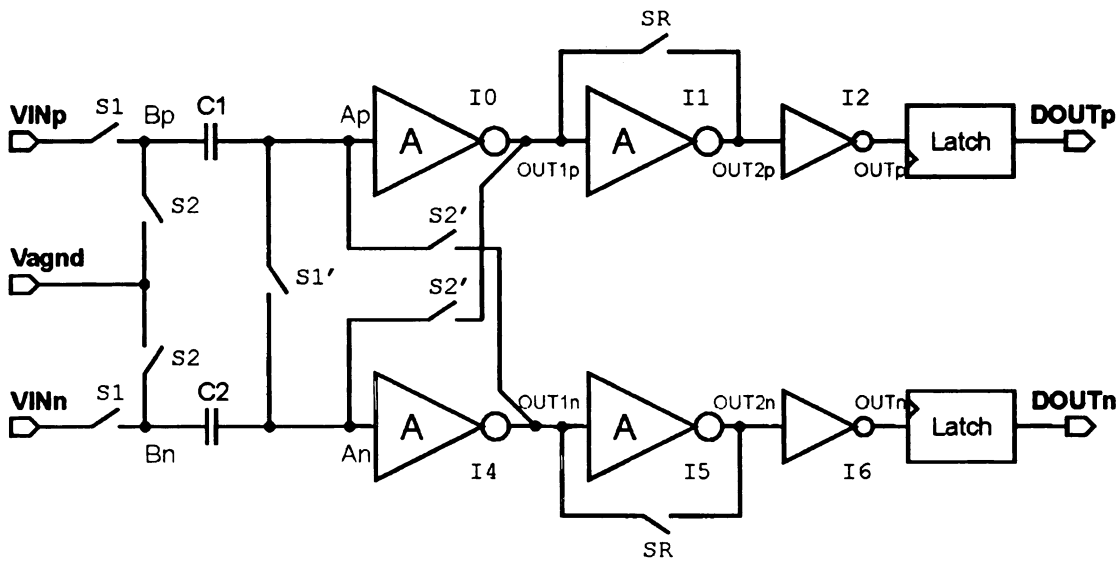


Figure 3.7. Schematic of a differential comparator.

During phase I (sample phase), the S_R switches are closed to settle node OUT1p, OUT1n and OUT2p, OUT2n to the middle point. S_1 and S_1' are closed, settling node Ap and An to the common mode voltage $(V_{IN+}+V_{IN-})/2$. Thus $(V_{IN+}-V_{IN-})/2$ and $-(V_{IN+}-V_{IN-})/2$ are sampled on C1 and C2, respectively. During phase II (compare phase), S_2 and S_2' are closed, pushing the voltage on Ap and An to $[V_{AGND}-(V_{IN+}-V_{IN-})/2]$ and

$[V_{AGND} + (V_{IN+} - V_{IN-})/2]$ respectively. Eventually $V_{AGND} = V_{MID}$.

If $V_{IN+} > V_{IN-}$, V_{Ap} is lower than V_{MID} and V_{An} is higher than V_{MID}. Thus $D_{OUT+} = 1$ and $D_{OUT-} = 0$.

If $V_{IN+} < V_{IN-}$, V_{Ap} is higher than V_{MID} and V_{An} is lower than V_{MID}. Thus $D_{OUT+} = 0$ and $D_{OUT-} = 1$.

Table 3.4 summarizes the operation of the differential comparator.

Table 3.4 Operation of the differential comparator

Phase	S_R	S_1	S_2	Vol. on node B		Vol. on node A	
Sample	1	1	0	V_{Bp}	V_{IN+}	V_{Ap}	$(V_{IN+}+V_{IN-})/2$
				V_{Bn}	V_{IN-}	V_{An}	
Compare	0	0	1	V_{Bp}	V_{MID}	V_{Ap}	$V_{MID} - (V_{IN+}-V_{IN-})/2$
				V_{Bn}		V_{An}	$V_{MID} + (V_{IN+}-V_{IN-})/2$

Switch S_1' helps to cancel out the common mode voltage, and only the differential mode voltage on C1 and C2 is sampled during the sample phase. In normal operation, the cross-coupled S_2' structure bypasses the inputs to the second stage inverters before the output of the first stage inverters change, thus helping the speed of the comparator. If somehow the inputs to both of inverters are higher or lower than V_{MID}, then because of the cross-coupled structure, a competition will decide the output of the inverters, ensuring that the inverter chain gives out complementary digital output. At the end of compare phase, the two latches latch the output of inverters and hold the value during the sample phase. Figure 3.8 shows the simulation waveforms of the differential comparator.

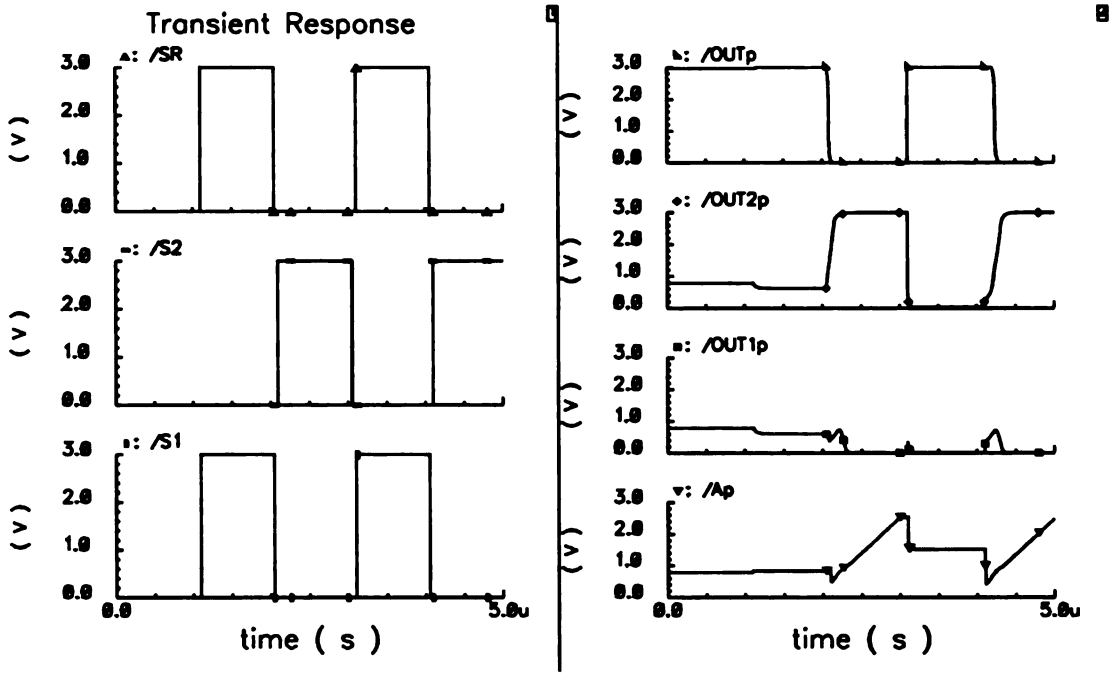


Figure 3.8. Simulation waveforms of the differential comparator.

In order to determine the resolution and offset of the comparator, a sinusoid wave is applied to one input of the comparator while the other input is kept at a DC value.

```
vIN1n IN1n 0 sin (2.00 50u 50k 2u)
vIN1p IN1p 0 dc 2.00
```

Ideally the latched comparator output should be a square wave with 50% duty cycle. By changing the amplitude of the sine signal, varying the DC offset, and watching if the output is still a square wave, we can know the resolution and offset of the comparator. The simulation shown in Figure 3.9 verifies that the differential comparator has a resolution less than 50μV and offset less than 50μV.

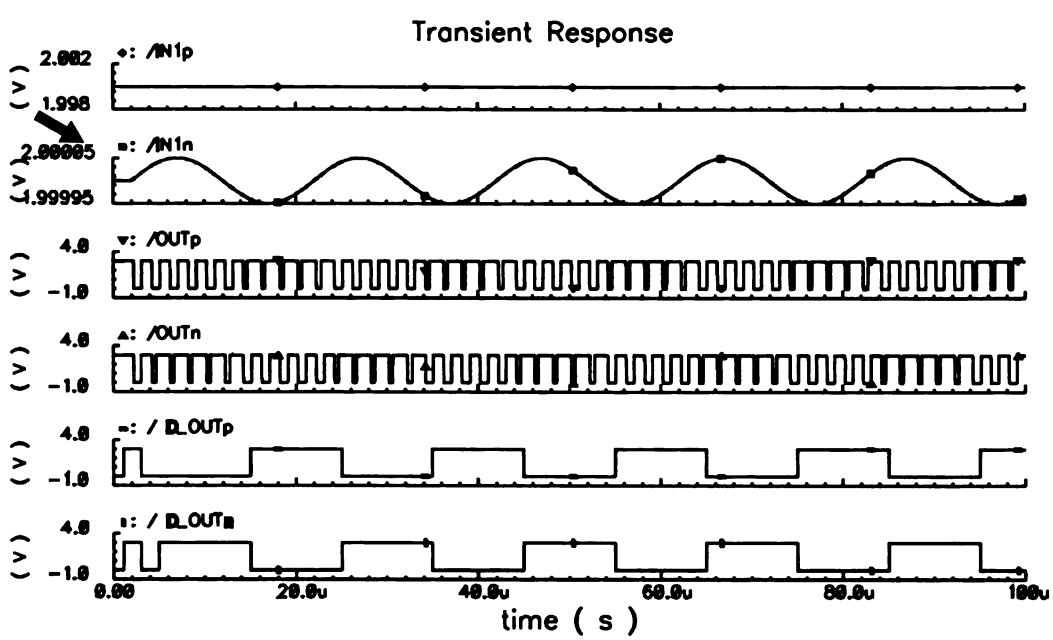


Figure 3.9. Simulation testing the resolution and offset of the differential comparator.

3.1.4 Sample and Hold

Figure 3.10 shows a simple inverter based sample and hold circuit. At initialization, S_R is closed to set the voltage on node A at V_{MID}. During phase I (sample phase), S_1 is closed to sample $(V_{IN}-V_{MID})$ in capacitor C. During phase II (hold phase), S_1 is open and S_2 is closed, storing V_{IN} at V_{OUT}. Table 3.5 summarizes the operation of the simple sample and hold. Simulation waveforms shown in Figure 3.11 verify the function of the circuit.

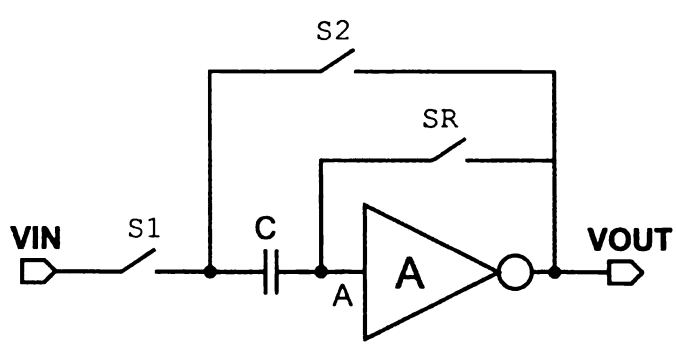


Figure 3.10. Schematic of the simple sample and hold circuit.

Table 3.5 Operation of the simple sample and hold

Phase	S_R	S_1	S_2	V_{IN}	Vol. on C	V_{OUT}
Sample	1	1	0	V_{IN} [n]	V_{IN} [n]-V_{MID}	V_{MID}
Hold	0	0	1	V_{IN} [n+1/2]	V_{IN} [n]-V_{MID}	V_{IN} [n]

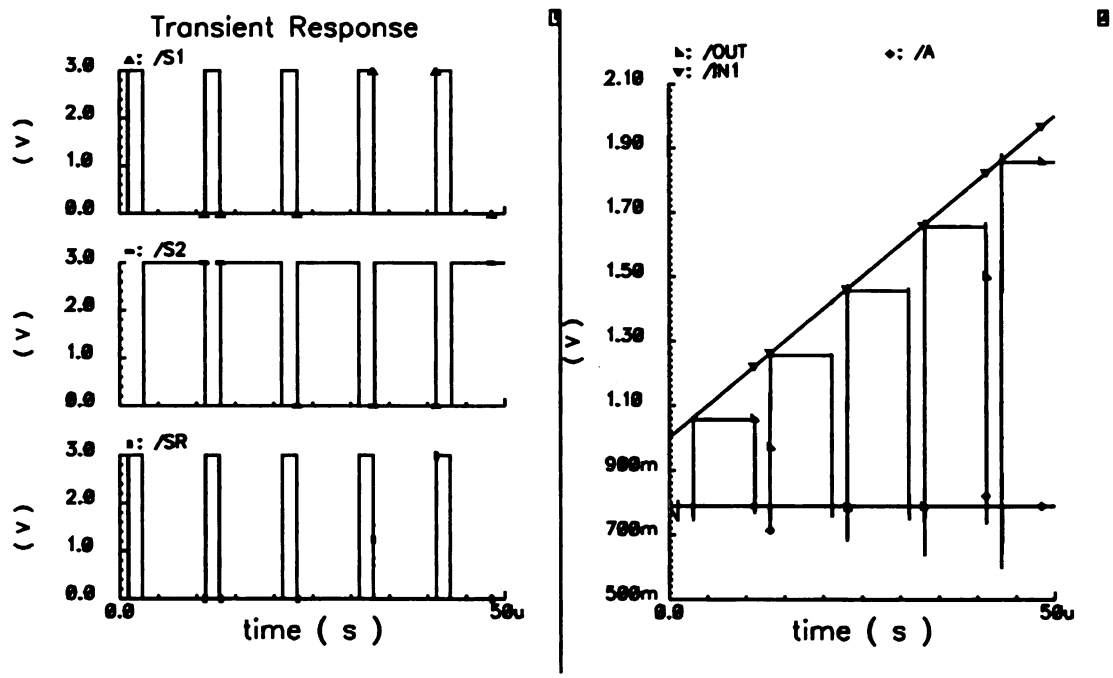


Figure 3.11. Simulation waveforms to verify the function of the simple sample and hold.

A disadvantage of circuit is that it is very sensitive to the finite gain of the inverter amplifier. If the amplifier gain is not infinite but –A, when the output is V_{OUT}, the input biasing to the inverter will become (V_{MID}-V_{OUT}/A). For the inverter amplifier, the gain is maximum when the input is biased at V_{MID}. Drift of the input bias will quickly degrade the gain of the inverter amplifier and the circuit will malfunction. To overcome this problem, the inverter amplifier has to be reset in each sample phase to set the bias at V_{MID}. However, this will pull the sample and hold output back to V_{MID} in every cycle, introducing unwanted noise into the circuit.

An improvement on the simple sample and hold circuit is to employ the Correlated Double Sampling (CDS) technique [9]. Figure 3.12 shows the schematic of a sample and hold circuit with CDS technique.

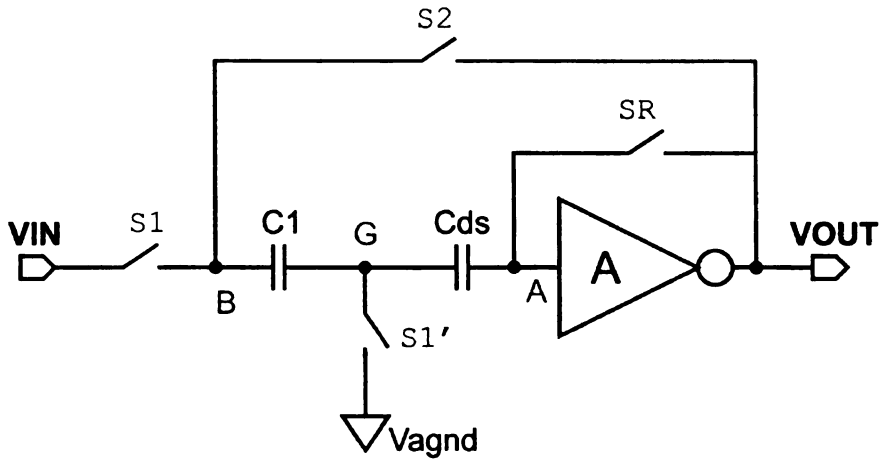


Figure 3.12. Schematic of the sample and hold with CDS technique.

Capacitor Cds is added to sample the offset between node A (V_{MID}) and node G (Vagnd) during initialization. In this case the offset should ideally be zero. Since the charge on capacitor Cds can not change after S_R is open, the voltage over Cds will not change. Furthermore, node G is reset to Vagnd every cycle, thus node A is indirectly reset to V_{MID}. Table 3.6 summarizes the operation of the sample and hold with CDS technique. Figure 3.13 shows the simulation waveforms to verify the function of the circuit.

Table 3.6 Operation of the sample and hold with CDS technique

Phase	S_R	S_1	S_2	V_{IN}	V_G	V_A	Vol. on C1	Vol. on Cds	V_{OUT}
Initial	1	1	0	N/A	Vagnd	V_{MID}	N/A	Vagnd-V_{MID}=0	N/A
Sample	0	1	0	V_{IN} [n]	Vagnd	V_{MID}	V_{IN} [n]-V_{MID}	Vagnd-V_{MID}=0	V_{IN} [n-1]
Hold	0	0	1	V_{IN} [n+1/2]	Vagnd	V_{MID}	V_{IN} [n]-V_{MID}	Vagnd-V_{MID}=0	V_{IN} [n]

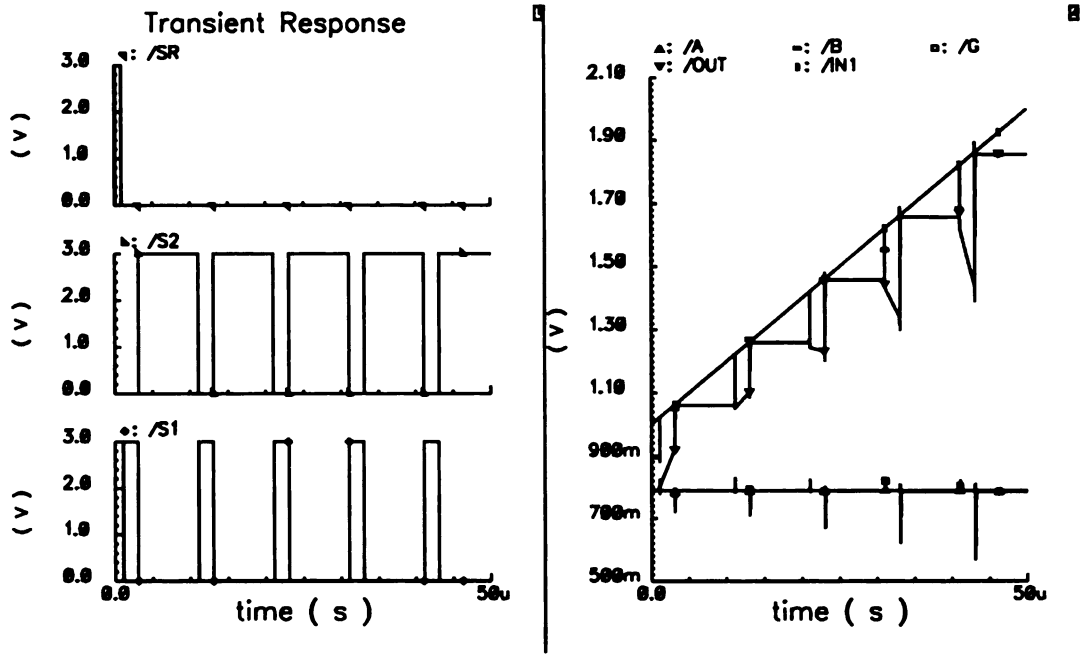


Figure 3.13. Operation of the sample and hold with CDS technique.

It can be seen that the inverter amplifier only has to be reset once during initialization. After that Cds will hold the V_{MID} and bias the amplifier. The voltage output will not be forced to V_{MID} during sample phase. However, during sample phase there is no feedback for the amplifier since both S_R and S_2 are open, making the output of the amplifier unstable, as shown in Figure 3.13. This problem can be overcome using Time-Shifted CDS (TS-CDS) technique [9] as shown in Figure 3.14.

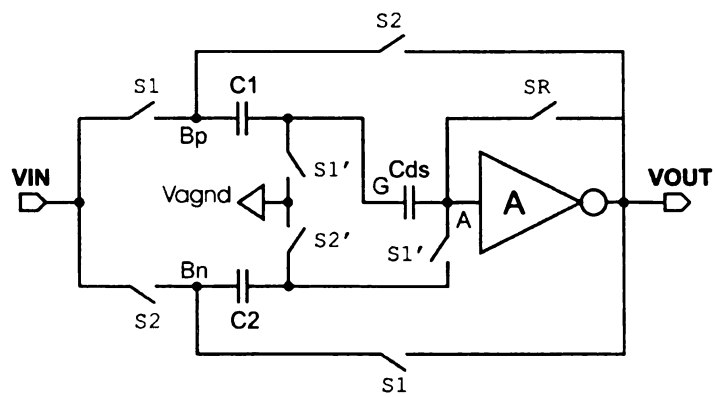


Figure 3.14. Schematic of the Sample and Hold with Time-Shifted CDS technique.

Another set of CDS sample and hold with opposite clocking phase is added into the

original CDS sample and hold to form the other half circuit. When the upper half circuit is in sample phase, meaning S_1 is closed and S_2 is open, the lower half is in its hold phase, presenting the voltage stored on C2 at the output. When the upper half circuit goes to hold phase, S_1 is open and S_2 is closed, the lower half goes to its sample phase. Thus at any time one half of the circuit will drive the output. Figure 3.15 shows the operation of the sample and hold circuit with Time-Shifted CDS technique [9]. Notice the OUT signal more closely follows the input than the results in Fig. 3.11 and 3.13, demonstrating the improvement in using this technique.

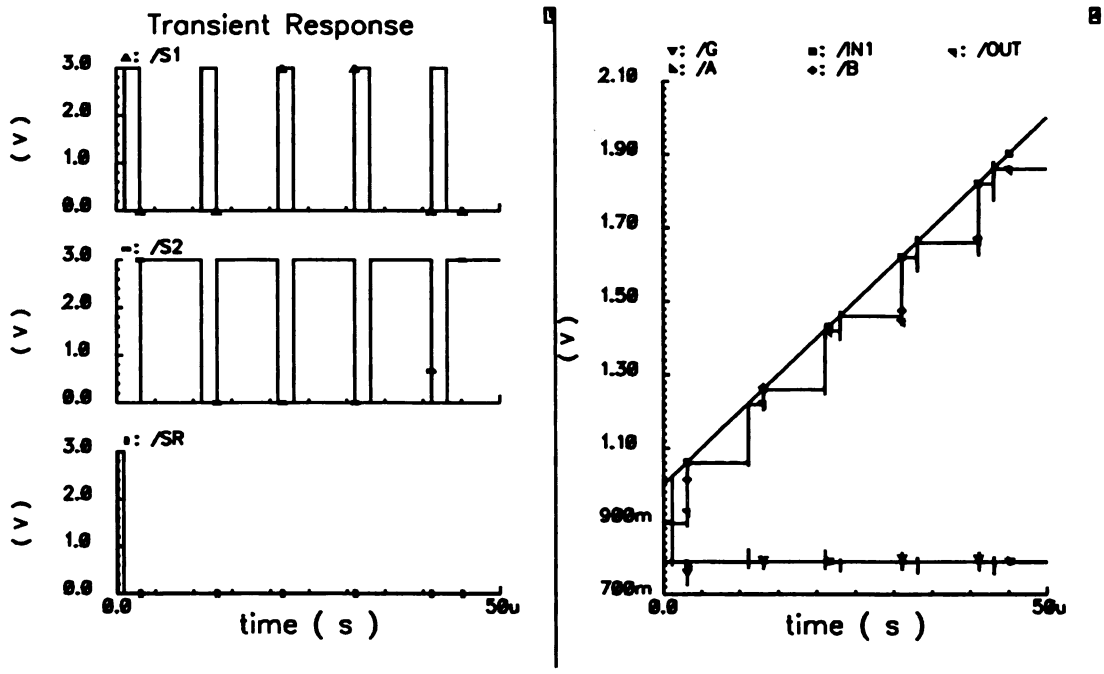


Figure 3.15. Operation of the Sample and Hold with Time-Shifted CDS technique.

3.1.5 1-bit DAC (Analog MUX)

The analog MUX is constructed using two CMOS switches. Figure 3.16 shows the schematic. S and Sn are complementary signals. When S is high, $V_{OUT}=V_{IN1}$; when S is low, $V_{OUT}=V_{IN0}$.

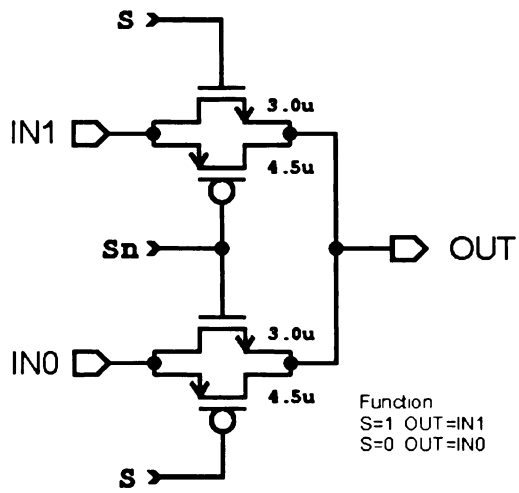


Figure 3.16. Schematic of the 1-bit DAC (analog MUX).

3.1.6 Whole System Schematic and Simulation

A fully differential circuit implementation of the Hybrid Algorithmic ΣΔ ADC architecture proposed in Chapter 2 was designed using the sub-blocks described above: the inverter-based integrator, the differential comparator, the sample and hold with Time-Shifted CDS technique, and the 1-bit DAC. The system schematic is included in Appendix E.

The Hybrid Algorithmic ΣΔ circuit is simulated and the transient simulation waveform is shown in Figure 3.17. It shows a case of L=16 and M=2 (4-bit each phase and 2 phases). Clock rate is 500 kHz. 0uS – 2uS is the initial step (reset). 2 uS – 34uS is the first phase and 34uS-66uS is the second phase. By comparing Figure 3.20 and Figure 2.4, especially noticing the similarity of the waveform on the integrator output (INT_OUTp), the function of the Hybrid Algorithmic ΣΔ circuit is verified.

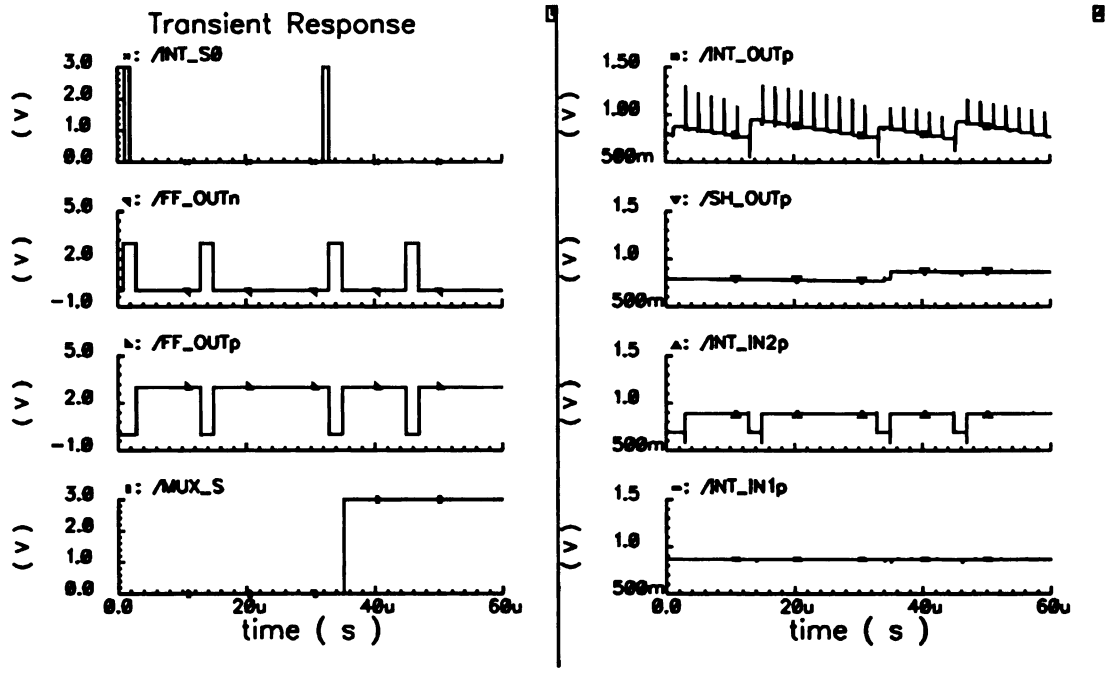


Figure 3.17. Simulation waveforms of the Hybrid Algorithmic ΣΔ ADC: L=16, M=2.

3.2 Circuit Layout

3.2.1 Floor Planning

A typical mixed-signal circuit contains active analog cells (opamps or comparators), passive components (resistors or capacitors), switches, and digital logic (for example, none-overlapping clock generator). Before designing the layout of the components, it is necessary to define the floor plan of the analog block. The general rules to follow are:

- put the analog critical components as far as possible from the digital elements
- make the connections to the critical nodes as short as possible
- avoid crossover between the analog biasing lines and digital busses

The layout of the chip follows the typical floor plan of a switched-capacitor circuit and a typical floor plan for a fully differential switched-capacitor circuit suggested by Franco Maloberti in [10].

3.2.2 Layout of the Basic Components

The basic components in the switched-capacitor circuits are the cascoded inverting amplifier, the capacitor pool, and the switches. Figure E.2 in Appendix E shows the layout of the cascoded inverting amplifier. It occupies a compact area of 35μm X 40μm.

The capacitors are made by PiP (poly2 over poly) capacitors. Figure E.3 in Appendix E shows the layout of a capacitor pool containing two matched 100fF capacitors. The layout follows the common centroid structure rule [10]. Each small capacitor block has around 50fF capacitance. The upper-left and lower-right blocks form C1 and the upper-right and lower-left blocks form C2. The whole capacitor pool is shielded by an

nwell, which is connected to a reference (V_{DD} in this design).

3.2.3 Layout of the Sub-Blocks

The circuit sub-blocks: the integrator, the comparator, the sample and hold, and the analog MUX (1-bit DAC) are carefully layout-ed following the floor plan described in section 3.2.1. All layout use only metal 1 and metal 2, providing the possibility to cover the entire analog core using metal 3 for shielding. All the sub-blocks passed DRC and LVS. The layout of the sub-blocks can be found in Appendix E.

3.2.4 Layout of the Whole Chip

The integrator, comparator, sample and hold, and analog MUX (1-bit DAC) were placed and connected to form the analog core of the Hybrid Algorithmic ΣΔ ADC. The layout follows the floor plan discussed in section 3.2.1. Analog bus was put on the right hand side and the digital bus was put on the left hand side. The analog core passed DRC and LVS.

Non-overlapping clock generator circuits for the integrator, comparator and sample and hold were added to the layout. Opamp voltage followers were added for observing the analog voltages during test. The whole circuit was put in a 1.5mm X 1.5mm 40-pin pad frame. The I/O pins, observation pins and power pins were connected to the pad frame. Layout of the whole chip is included in Appendix E.

REFERENCES

[1] MOSIS Integrated Circuits Fabrication Service: http://www.mosis.org/ .

[2] MOSIS Educational Program (MEP): http://www.mosis.org/products/mep/ .

[3] AMI C5N process webpage: http://www.mosis.org/Technical/Processes/proc-ami-c5n.html

[4] NCSU Cadence Design Kit: http://www.cadence.ncsu.edu/CDK.html .

[5] The IIT standard cell library: http://vlsi.ece.iit.edu/projects/scells/.

[6] P.E. Allen & D. R. Holberg, *CMOS Analog Circuit Design* Second Edition Chapter 9.

[7] P. R. Gray, P. J. Hurst, S. H. Lewis, R. G. Meyer, *Analysis and Design of Analog Integrated Circuits*, Wiley, 2001

[8] B. Razavi, *Design of Analog CMOS Integrated Circuits*, McGraw-Hill, 2000

[9] Jipeng Li and Un-Ku Moon, "A 1.8-V 67-mW 10-bit 100-MS/s pipelined ADC using time-shifted CDS technique," *IEEE J. Solid-State Circuits*, vol. 39, pp. 1468 - 1476, Sept. 2004.

[10] J.E. Franca, Yannis Tsividis, edited, "*Design of Analog-Digital VLSI Circuits for Telecommunications and Signal Processing*", Second Edition, Prentice Hall, 1994, Chapter 11 by Franco Maloberti, "*Layout of Analog and Mixed Analog-Digital Circuits*".

Chapter 4
Testing & Results

4.1 Fabricated Chip

The Hybrid Algorithmic ΣΔ ADC v1.0 chip was fabricated using a standard 0.5u CMOS process available through MOSIS. A micrograph of the fabricated prototype is shown in Figure 4.1. The chip is packaged in DIP40. The pin out and description is given in Appendix D.

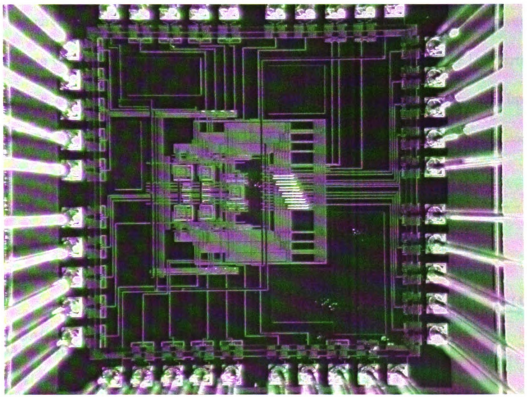


Figure 4.1 Fabricated Hybrid Algorithmic ΣΔ ADC v1.0chip photo.

4.2 Test Setup

A key component in mixed-signal design is a reliable test-station, which can be used for evaluating and benchmarking the performance of fabricated chips. A general-purpose analog/mixed-signal IC testing station has been developed in the Adaptive Integrated Microsystems Laboratory to verify the function and measure the performance of the A/D converter chip. Details about the specifications and design of the test station is summarized in Appendix E and can be also be found at http://www.egr.msu.edu/aimlab/teststation.htm.

Figure 4.2 shows the setup for testing the A/D converter chip. All the biasing, reference voltages and input signals are generated using NI-DAQ card and the DAC array. Clocking, control logic and the counter/shift register for the ADC are implemented from the FPGA. The MATLAB code for generating analog biasing and the FPGA Verilog code for generating clocking and control signals are attached as Appendix F.

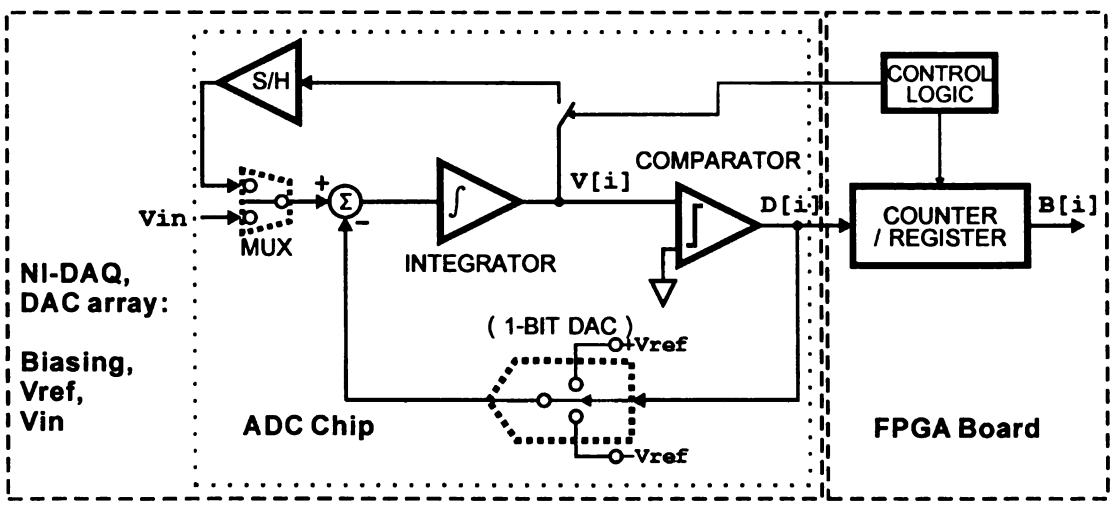


Figure 4.2. Setup of testing the A/D converter chip.

Figure 4.3 shows a photograph of the testing station ready to test the ADC chip. The NI-DAQ card and the FPGA board are connected to the testing motherboard. The

daughter board is mounted on the motherboard and the A/D converter chip is put in the socket.

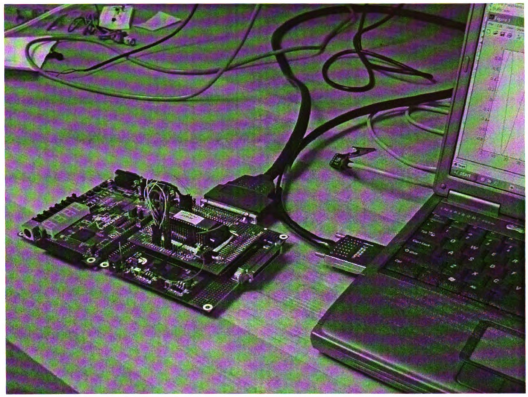


Figure 4.3. Photo of the testing station ready to test the chip.

4.3 Testing of Circuit Components

4.3.1 Testing of the Op Amp Voltage Followers

The first step for testing the ADC chip is to test the Op Amp voltage followers and find suitable biasing to the Op Amps so that we can have observation to the internal analog voltages available. All the voltage followers have the same circuit and layout, and share the same biasing, thus we can test any one of the voltage followers.

To test the voltage follower connected to the INT_IN1p node (refer to Appendix E for schematic and pin description), whose input can be directly accessed through the INp pin by switching the analog MUX to 0, apply a ramp signal in INp pin and observe the output at INT_IN1p pin. Figure 4.4 shows the waveform when testing the Op Amp voltage followers. Channel 1 is the input and Channel 3 is the output. It can be seen that when the input is 780mV ~ 2.5V, the output tracks the input, while for input voltage lower than 780mV, the output is always 780mV; and for input voltage higher than 2.5V, there is a noticeable voltage drop at the output. In latter testing, it should be remembered that an observation voltage of 780mV could mean that the input is actually less than 780mV.

In Cadence simulations, with a biasing of `VPB1=2.3V, VPB2=1.8V, VNB1= 0.9V, VNB2=0.7V,` the voltage follower can function correctly with input signal frequency as high as 100 kHz. However, in real circuit testing, with the above biasing, the voltage follower can only track its input when the input signal is less than 100 Hz. A possible explanation is that in Cadence simulation, the pad capacitance is not considered while in the actual chip the voltage follower does not have enough drive capability for the big pad

capacitance with above biasing. After changing the biasing to `VPB1=1.7V; VPB2=1.5V; VNB1=1.3V; VNB2=1.1V;` the voltage follower can correctly follow the input signal up to 100 kHz.

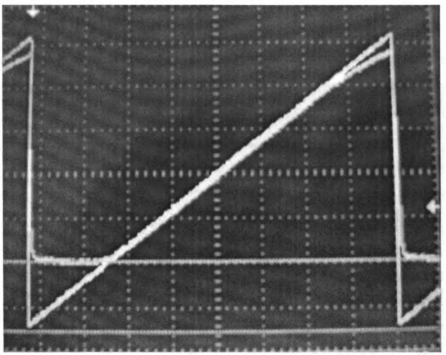


Figure 4.4. Waveform of testing the op amp voltage follower.

4.3.2 Testing for the Cascoded Inverter Amplifiers

The second step was to test the cadcoded inverter amplifiers and set the biasing. A trick is used there: INT_CLK1 and INT_CLK2 are closed at the same time in order to vary the voltage on node INT_Ap by varying REF (agnd). A summary of the configuration for testing the cascoded inverter amplifiers is shown below:

INT_S1 = 1; INT_S2=1;
Apply a ramp signal in REF pin (pin 24);
Observe the INT_Ap pin (pin 13) and INT_OUTp pin (pin 12).

Figure 4.5 shows the waveform when testing the cascoded inverting amplifer. Channel 1 is the input and Channel 4 is the output. Comparison to the input-output

characteristic shown in Figure 3.2, this waveform demonstrates that the cascoded inverter amplifier is functioning as expected. Notice that the output can only go as low as 780mV. This is due the problem of voltage follower as described in section 4.3.1. Testing shows that when the biasing is VB1=2.2V, VB2=2.0V, VB3=1.4V, the cascoded inverter amplifier has a gain around 2000 (66 dB). Testing results also show that Vmid ≅780mV.

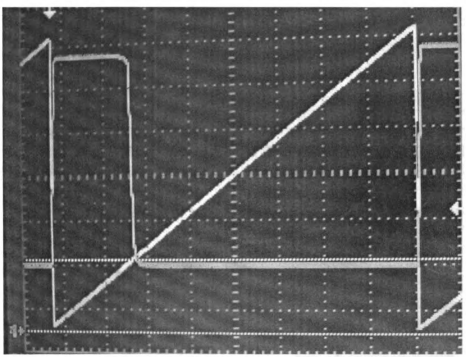


Figure 4.5. Waveform of testing the cascoded inverting amplifier.

4.3.3 Testing for the Integrator

To test the integrator, the clocks shown in Figure 4.6 are applied and the switches are set as follows:

MUX_S = 0; CMP_S1 = 0; SH_S1 = 0; SH_S2 = 0;

The waveform shown in Figure 4.7 is generated and the inputs are set as:

INp = Vmid+0.1V; REFp= Vmid; REFn= Vmid

In Figure 4.7 channel 1 (top) is the reset (INT_SR), channel 2 is the clock (INT_CLK1), channel 3 is the output of the integrator (INT_OUTp), and channel 4 (bottom) is the input to the cascoded inverter amplifier (INT_Ap). It can be seen clearly that the integrator is integrating the difference between its two inputs (INp and DAC_OUTp), and the integrator output is increasing while the cascoded inverter amplifier input remains at Vmid=780mV. This waveform matches the integrator simulation result as shown in Figure 3.4, demonstrating the integrator operates correctly.

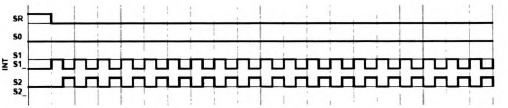


Figure 4.6. Clock for testing the integrator.

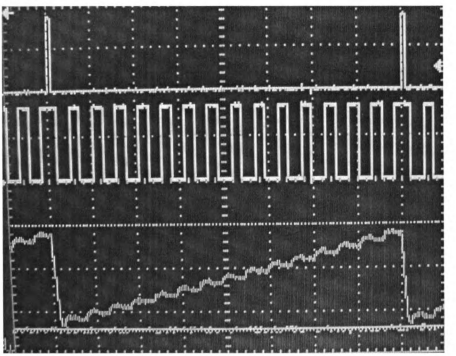


Figure 4.7. Waveform when testing the integrator.

4.3.4 Testing for the Comparator

Because the inputs to the comparator could not be directly accessed, the integrator was used to vary the inputs to the comparator, and then the output of the comparator was examine. Figure 4.8 shows the clocks used to test the comparator. Other switches value and input stimulus for testing the comparator are as follows:

MUX_S = 0; SH_S1 = 0; SH_S2 = 0;
INp = Vmid+0.08V; REFp= Vmid+0.2V; REFn= Vmid-0.2V;

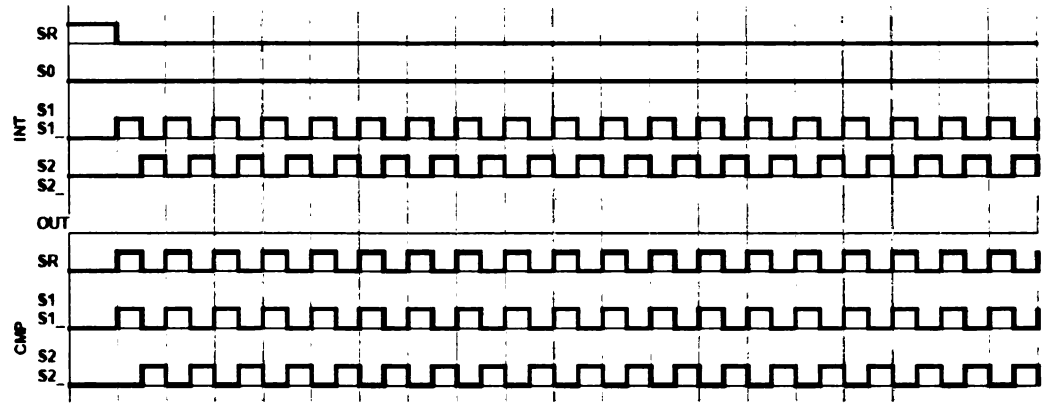


Figure 4.8. Clocks for testing the comparator.

Under this configuration, the output waveform is shown in Figure 4.9. Channel 1 is the reset signal (INT_SR), channel 3 (next to top) is one input to the comparator (INT_OUTp) and channel 4 is the other input to the comparator (INT_OUTn). Channel 2 (bottom) is the latched output of the comparator (FF_OUTp). It can be seen clearly that when VINp > VINn, FF_OUTp = 1, and when VINp < VINn, FF_OUTp = 0. This is exactly the same as expected in Section 3.1.3. Further observation of the FF_OUTp pin and the FF_OUTn pin shows that they always produce complementary outputs. The comparator operates correctly.

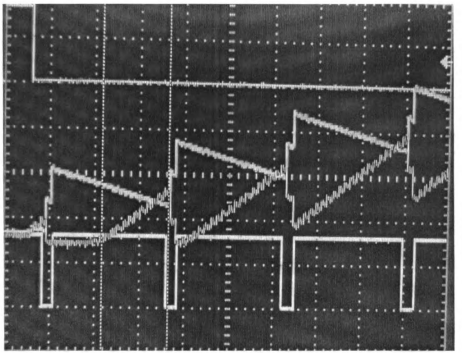


Figure 4.9. Waveform testing the comparator.

4.3.5 Testing for the Sample and Hold (SaH)

To test the sample and hold, the integrator is kept running to vary the input to the sample and hold. The input is sampled and held and the output is observed. Figure 4.10 shows the clock scheme for testing the sample and hold. Figure 4.11 shows the waveform when testing the sample and hold. Channel 1 (top) is the reset signal (SH_SR). Channel 2 (bottom) is the clock of the sample and hold (SH_CLK1). Channel 3 is the output of the comparator (INT_OUTp) and also the input to the sample and hold. Channel 4 is the output of the sample and hold (INT_IN1p). Test results show that the sample and hold functions correctly but has noticeable offset error between the input signal and the output signal. As observed in Matlab system behavior simulation, this offset error will greatly degrade the resolution for algorithmic mode operation.

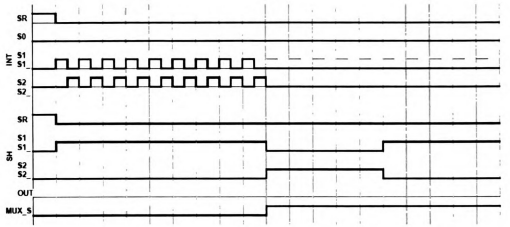


Figure 4.10. Clocks for testing the sample and hold.

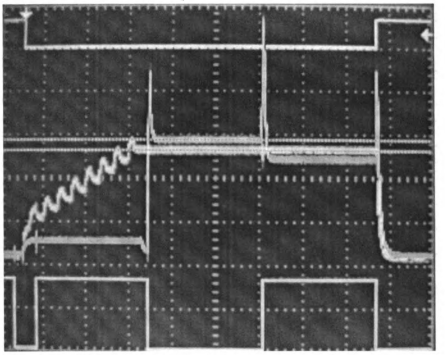


Figure 4.11. Waveform when testing the sample and hold.

4.4 System Testing

4.4.1 1st-order ΣΔ operation

For the chip to operate in 1st-order ΣΔ mode, the clocks were generated as shown in Figure 4.12. The integrator and comparator were kept run and the sample and hold was shut off. The biasing and stimulus are set as:

```
VDD = 3.3 V; VB1 = 2.2 V; VB2 = 2.0 V; VB3 = 1.4 V;
VPB1 = 1.7 V; VPB2 = 1.5 V; VNB1 = 1.3 V; VNB2 = 1.1 V;
Vmid = 780 mV; VREF = 200 mV; Vin = 160 mV;
INp = Vmid+Vin;  INn = Vmid-Vin;
REFp= Vmid+VREF; REFn= Vmid-VREF;
```

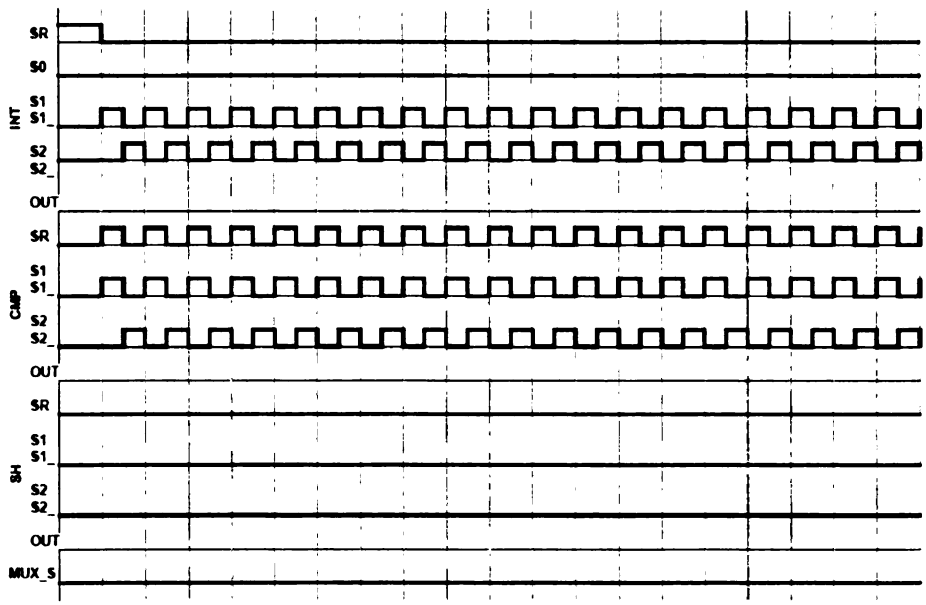


Figure 4.12. Clocks for 1st-order ΣΔ operation.

Figure 4.13 shows a 1st-order ΣΔ operation runs in 256 clock cycles (8-bit). Channel 1 is the reset signal (INT_SR). Channel 3 is the output of the integrator (INT_OUTp). Channel 2 is the latched output of the comparator (FF_OUTp), and channel 4 is the LSB of the counter. The counter is implemented in the FPGA, and increases its value by one at

the end of each clock if the comparator output is 1. The counter value is displayed in the LED display on the FPGA board, as shown in Figure 4.14.

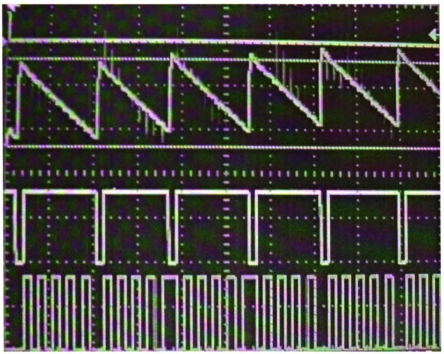


Figure 4.13. Waveform of 1st-order ΣΔ operation.

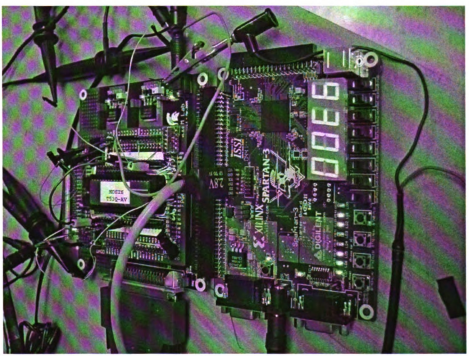


Figure 4.14. The counter value displayed in the LED at the end of each conversion.

It can be seen from Figure 4.13 and Figure 4.14 that:

- The integrator output has a similar shape as in the MATLAB system behavioral simulation shown in Figure 2.4 and in the Cadence system level simulation shown in Figure 3.17.
- The latched comparator output goes to 0 every 10 clocks, which means the ratio of 1s over to overall clocks is 9/10. This corresponds to the input voltage value over reference voltage value = (Vin+VREF)/(REFp-REFn) = 200mV+160mV / 2*200mV = 0.9.
- The LSB of the counter flips only when the latched comparator output is 1. The implementation of the counter is correct.
- The displayed counter output is E6 in hex value. (E6/FF)h = (230/256)d = 0.898, which is very close to the input voltage value.

All of the above suggests that the chip in 1st-order ΣΔ operation is functionally correctly. Further testing and performance measurement shows that the 1st-order ΣΔ can achieve 10 bits resolution.

4.4.2 Hybrid Algorithmic ΣΔ operation

Figure 4.15 shows the clock setup for the ADC chip operating in the Hybrid Algorithmic ΣΔ mode with 3 phases algorithmic conversion and 4 clocks of ΣΔ conversion in each phase. INT_S0 goes to 1 at the end of each phase to generate the residue. The residue is then sampled by the sample and hold and feed back to the input by switch the MUX.

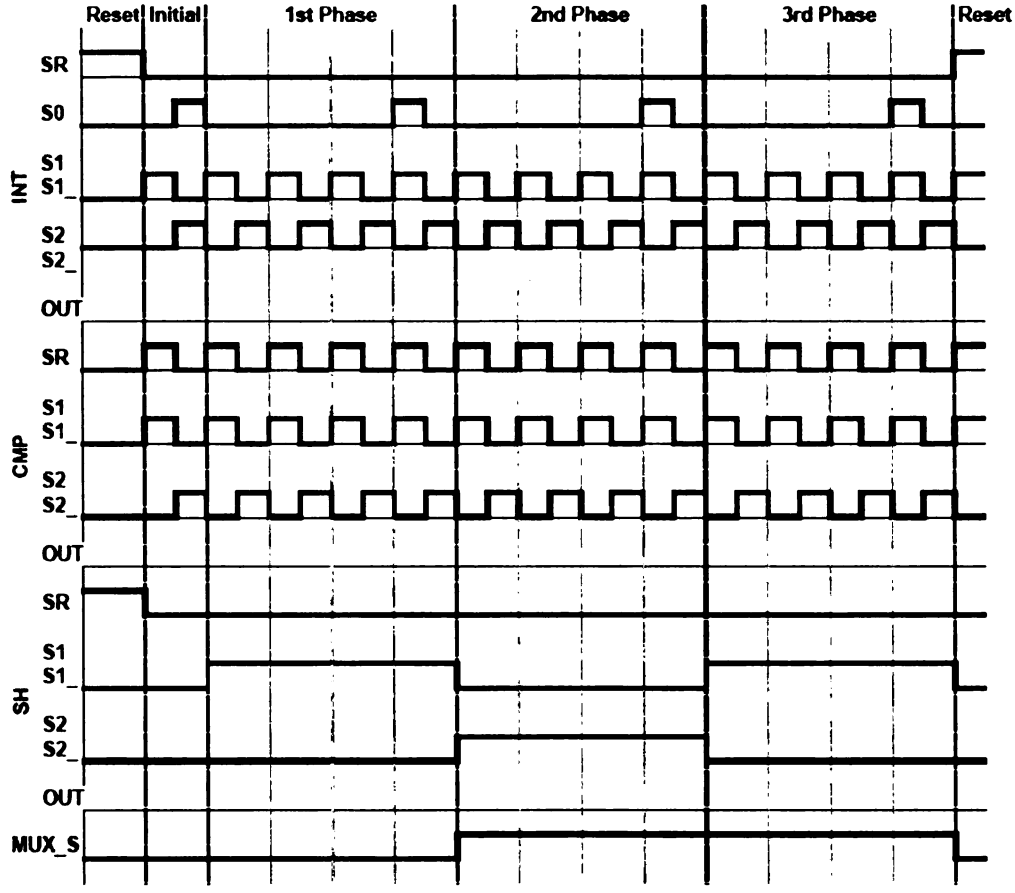


Figure 4.15 Clocks for Hybrid Algorithmic ΣΔ operation

The biasing and stimulus is the same as in the 1st-order ΣΔ operation. The waveforms for Hybrid Algorithmic ΣΔ operation are shown in Figure 4.16. Channel 1 is the reset signal (INT_SR). Channel 4 is the output of the integrator (INT_OUTp) and channel 3 is the

output of the opposite integrator (INT_OUTn). Channel 2 is the clock of the sample and hold (SH_CLK1). It can be seen from Figure 4.16 that:

- In the first phase, the ADC runs in normal 1st-order ΣΔ mode. The integrator outputs have the similar shape as in Figure 4.13.
- In the second phase, the integrator outputs go high dramatically. Obviously the ADC does not function correctly.

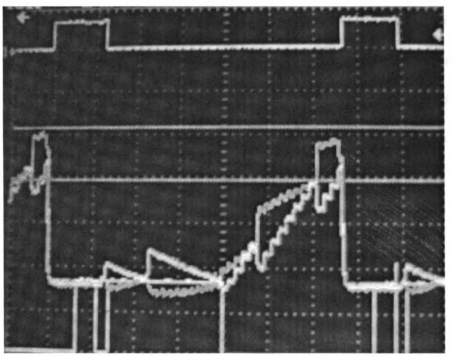


Figure 4.16 Testing waveform of Hybrid Algorithmic ΣΔ operation

Possible reasons for the Hybrid Algorithmic ΣΔ operation does not work well are:

- The offset of the sample and hold phase added a common mode voltage at both the positive half and negative half integrator outputs. This common mode error gets integrated on both of the integrators, forming the second phase waveform in Figure

4.16

- The small error between the input signal common mode and the analog ground (V_{MID}) gets integrated on the positive half and negative half integrators together, forming a large common mode offset error at the end of first phase. This common mode error is sampled and gets integrated again in the second phase. But since now the error is large, the integrator outputs go high at a much faster speed, as shown in Figure 4.16.

The integrators are not differential, but rather are formed by two separate integrators. Since there is no common mode cancellation scheme in the integrators, any offset in the input will get integrated on both integrators at the same time. An improvement to the current design would be to use differential integrator and add common mode cancellation in the circuit. This should allow the ADC to function as expected.

REFERENCES

[11] M. Burns, G. W. Roberts, *An Introduction to Mixed-Signal Ic Test and Measurement*, Oxford University Press, 2001.

[12] J. Franca and Y. Tsivids, *Design of Analog-to-Digital VLSI Circuits for Telecommunications and Signal Processing*, Chapters 9 & 12, Prentice Hall Inc., Englewood Cliffs, NJ, 1994.

Chapter 5
Conclusions

5.1 Achievements

This work has introduced a novel Hybrid Algorithmic ΣΔ A/D topology that can achieve real-time reconfiguration between ΣΔ and algorithmic architectures to optimize the trade off between its resolution and speed and to assist power management. Analysis has demonstrated that by iteratively feeding back and resample the residual of a ΣΔ conversion, the behavior of the A/D converter approaches to an algorithmic conversion. By varying the number and ratio of sampling and feedback cycles, the A/D converter is capable of self-adjustment between a more ΣΔ-like architecture (with higher resolution and slower speed) and a more algorithmic-like architecture (with faster speed and lower resolution).

A pseudo-differential circuit implementation of the Hybrid Algorithmic ΣΔ A/D converter was designed in switched-capacitor topology and fabricated in a in a standard 0.5μm CMOS process. The prototype chip was tested to be fully functional. The A/D converter can be configured in first-order ΣΔ mode to convert 8-bit resolution at 2 kHz, or in algorithmic ΣΔ mode to convert 4-bit resolution at 62.5 kHz. The analog circuits consumes only 8.6μW during operation. The concept of the Hybrid Algorithmic ΣΔ A/D converter has been verified.

5.2 Possible Improvements

Future generations of the Hybrid Algorithmic ΣΔ A/D converter could be improved by addressing the following issues:

- Employ a fully differential integrator rather than two separate integrators to overcome the common mode voltage problem described in section 4.3.2.
- Increase the drive capability of the comparator latch output to the chip pad frame. The current version uses minimum sized transistors that created a bottleneck of the chip operation speed.
- Employ rail-to-rail voltage follower buffers to analog test pads for better observation of the internal analog voltages.