

# DIGITALLY ASSISTED ADCS

by

Ivan Timothy Bogue

A dissertation submitted in partial fulfillment  
of the requirements for the degree of  
Doctor of Philosophy  
(Electrical Engineering)  
in The University of Michigan  
2008

Doctoral Committee:

Associate Professor Michael Flynn, Chair  
Professor Fred L. Terry Jr.  
Assistant Professor Jerome P. Lynch  
Assistant Professor David Dale Wentzloff

*To moje zlatičko*

## Acknowledgements

I would like to gratefully acknowledge Professor Michael Flynn for his guidance and direction throughout this work. His keen intuition and deep technological insight were invaluable. Without his friendship and support this work could never have been completed. I would also like to thank my other doctoral committee members, Professor Fred L. Terry Jr., Professor Jerome P. Lynch and Professor David D. Wentzloff for their suggestions and help.

I would also like to thank my wife Nicole for her encouragement, patience and unstinting support throughout my years in Ann Arbor. I could not have done it without her. Special thanks also to my parents for their years of support and encouragement.

I would like to thank all the students with whom I have been fortunate to share laboratory and office space with. The technical discussions helped the research and the friendly discussions helped me keep me sane. In particular I would like to thank all students past and present of Prof. Flynn's group, especially Dr. Fatih Kocer, Dr. Jia-yi Chen, Dr. Sunghyun Park, Junyoung Park and Dan Shi.

I would also like to thank the great staff of both the WIMS ERC and the EECS department. A special thanks to Paulette Ream and Julia Hrycko.

I am extremely grateful to Brendan Casey for all the hard work that he did to help me with my testing and Joel van Laven for his help with CAD issues. I would also like to thank Stacy Ho of Analog Devices for his assistance.

This work was supported by the WIMS-ERC, Engineering Research Centers program of the NSF under Award EEC-9986866 and by NSF Award CCF0346874.

## Table of Contents

Dedication .....	ii
Acknowledgements.....	iii
List of Figures .....	vi
List of Tables .....	ix
Abstract.....	x
Chapter 1. Introduction .....	1
1.1 Background.....	1
1.2 ADC Offsets.....	2
1.3 Offset reduction/correction techniques .....	4
1.4 ADC primer .....	6
1.5 Recent ADC publications .....	7
1.6 Technology Scaling .....	12
1.7 Porting Designs.....	14
1.8 Contributions of this work .....	15
1.9 Thesis Outline .....	16
Chapter 2. Folding ADC review .....	17
2.1 Introduction.....	17
2.2 Folding ADC architecture.....	18
Chapter 3. A digitally calibrated 8-bit folding ADC .....	25
3.1 Introduction.....	25
3.2 Digital Calibration Scheme.....	26
3.3 Implementation .....	33
3.4 Distortion and Matching .....	40
3.5 Prototype Measurements.....	53
Chapter 4. 9-bit folding ADC with novel folders .....	59
4.1 Introduction.....	59

4.2 Thermal noise in parallel folders .....	59
4.3 Digital calibration .....	65
4.4 Implementation .....	70
4.5 Comparator matching and yield.....	74
4.6 Prototype .....	77
Chapter 5. Conclusion and suggestions for future work.....	79
5.1 Conclusion .....	79
5.2 Suggestions for future research.....	80
References.....	83

## List of Figures

### Figure

1.1 ADC usage.....	2
1.2 (a) Unbalanced differential pair. (b) Equivalent differential pair with offset voltage .....	3
1.3 3-bit flash ADC.....	6
1.4 Figure of Merit for published ADCs.....	11
1.5 Active area of published ADCs .....	12
1.6 Intel x86 number of transistors per processor.....	13
1.7 Published ADC performance .....	13
1.8 Technology node used for x86 processor .....	15
2.1 (a) 3-bit flash ADC with 7 comparators. (b) 3-bit folding ADC with 2 comparators.....	18
2.2 (a) Four cross coupled differential pairs. (b) Folder output currents cross four times, each crossover point comes from a single differential pair.....	19
2.3 (a) Cascaded folding-by-four generated by two stages of folding-by-two. (b) Equivalent CMOS block diagram.....	21
2.4 Interpolation is used to reduce the number of required folders. In this example every second folder is omitted .....	22
2.5 (a) 4-bit folding ADC has 4 folders and comparators. (b) Output waveforms .....	23
2.6 Offsets in coarse ADC cause large errors in output.....	24
2.7 Use of 2 comparators in coarse ADC to align coarse ADC with folders .....	24
3.1 Yield versus redundancy for an 8-bit ADC. ....	27

3.2 Small, low accuracy redundant folders and comparators are used to generate redundant zero-crossings .....	28
3.3 (a) Nominal comparator trip-points with a redundancy of 4 comparators for each code. (b) Offsets move trip-points. Example of reassignment, 2A is assigned to code 3 .....	29
3.4 Redundancy and reassignment of folder zero-crossings.....	30
3.5 Block diagram of digitally calibrated folding ADC .....	32
3.6 Example of Zero-crossing memory, 8 words required for 3-bit coarse ADC .....	33
3.7 Polarity Corrector memory consists of 4 words of 256 bits. Here 4 bits are shown .....	34
3.8 Coarse ADC comparator search. In this example, a comparator is found on the 4th search iteration.....	35
3.9 The parasitic capacitance of the routing and folder inputs forms the sample-and-hold capacitance.....	36
3.10 (a) Resistor ladder suffers from parasitic corner resistance. (b) Resistor ladder with 'L' shaped resistors. (c) DAC switches and substrate connections are embedded in ladder cavities.....	37
3.11 (a) Calibration of coarse ADC. (b) Zero-crossing search .....	38
3.12 Folder output is mirrored into two comparators .....	39
3.13 Increase in folder input voltage decreases tail current source current.....	41
3.14 Folder output currents for three different differential pair transconductance.....	44
3.15 Transconductance of each folder, versus input voltage. As differential pair transconductance reduces so does folder transconductance .....	45
3.16 DNL plots derived from Spectre simulation of an 8-bit ADC using these folders. The distorted output of a single folder as the input voltage is swept is in the inset.....	47
3.17 (a) Random comparator offsets generate 8 unique zero-crossings. (b) Comparator latch, matching is dominated by the size of MP1 and MP2 .....	48
3.18 ADC yield as comparator offset is varied for various folder offsets.....	50
3.19 Coarse ADC yield as comparator offset is increased.....	51
3.20 Monte Carlo results of mean ENOB versus DAC precision.....	52

3.21 Folding ADC die micrograph .....	53
3.22 PCB test board .....	54
3.23 ADC test setup .....	55
3.24 DNL before calibration is 6.8LSB and after calibration DNL is 0.8LSB.....	56
3.25 SFDR, SNDR and THD versus sampling frequency with $F_{IN}=1\text{MHz}$ .....	57
3.26 SFDR, SNDR and THD versus input frequency with $F_S=500\text{MS/s}$ .....	57
4.1 Folding by 4 folder.....	60
4.2 Equivalent circuit for the folder with $V_{in}$ close to $v_{r2}$ .....	60
4.3 (a) Stacked Folder reuses single tail current. (b) Output currents as input is swept .....	62
4.4 Stacked folder, folding by 9 requires 3 large current sources and 3 small .....	64
4.5 Yield versus redundancy for an 9-bit ADC. ....	67
4.6 A 4-bit Coarse ADC gives 16 sections. A selection, (folder 0,15,31,47 and 63) of the total 64 folding by 8 waveform outputs is also shown .....	68
4.7 9-bit folding ADC uses 64 folders and 256 comparators. ....	70
4.8 $V_{in}$ is sampled into $C_L$ and a source follower provides $V_{out}$ .....	71
4.9 (a) Simple sampling circuit. (b) Bootstrapped circuit generates a constant $V_{GS}$ .....	72
4.10 Bootstrapping Circuit.....	73
4.11 Output $v_{boot}$ is $V_{DD}$ above $v_{in}$ when $ch$ is high .....	74
4.12 Yield versus standard deviation comparator offset.....	75
4.13 Comparator mismatch dominated by $M1$ , $M2$ and $MP1$ , $MP2$ matching.....	76
4.14 Die micrograph of 9-bit folding ADC prototype .....	77



## List of Tables

### Table

1.1 Summary of reported ADCs .....	10
3.1 Summary of 8-bit ADC performance .....	58
4.1 Polarity of zero-crossings .....	69
4.2 Simulated 9-bit ADC performance .....	78

# ABSTRACT

## DIGITALLY ASSISTED ADCS

by

Ivan Timothy Bogue

Chair: Michael Flynn

This work involves the development of digital calibration techniques for Analog-to-Digital Converters. According to the 2001 International Technology Roadmap for Semiconductors, improved ADC technology is a key factor in the development of present and future applications.

The switched-capacitor (SC) pipeline technique is the most popular method of implementing moderate resolution ADCs. However the advantages of CMOS, which originally made SC circuits feasible, are being eroding by process scaling. Good switches and opamps are becoming increasingly difficult to design and the growing gate leakage of deep submicron MOSFETs is causing difficulty. Traditional ADC schemes do not work well with supply voltages of 1.8V and below. Furthermore, the performance

required by present and future wireless and IT applications will not be met by the present day ADC circuits techniques.

Bearing in mind the challenges associated with deep sub-micron analog circuitry a new calibration technique for folding ADCs has been developed. Since digital circuitry scales well, this calibration relies heavily on digital techniques. Hence it reduces the amount of analog design involved. As this folding ADC is dominated, in terms of both functionality and power, by digital circuitry, the performance of folding will improve when implemented in smaller geometry processes.

An 8-bit, 500MS/s, digitally calibrated folding ADC was designed in TSMC 0.18 $\mu$ m. A second prototype, 9-bit 400MS/s, was designed in ST 90nm. This ADC uses novel folders to reduce thermal noise.

The major accomplishments of this work are:

- The creation of a new folding ADC architecture that is digitally dominated allowing large transistor mismatch to be tolerated so that small devices can be utilized in the signal path.
- The development of modeling techniques, to investigate and analyze the effects of transistor mismatch, folder linearity and redundancy in ADCs.
- The design of a new folder circuit topology that decreases the required power consumption for a given noise budget.
- The design of a resistor ladder DAC that uses a unique resistor layout to allow any shape ladder to be designed.

## Chapter 1

### Introduction

#### **1.1 Background**

In the world of electronics there are 2 types of signals: analog signals and digital signals. Analog signals pertain to the real world values, heat, pressure, sound or temperature. Digital signals are discrete time signals and have distinct levels. Analog signals are converted to the digital domain for storage, digital signal processing, transmission and display. Analog to digital converters (ADCs) perform this function. An ADC compares the analog input voltage to known reference voltages and then produces a digital output. By its nature an ADC introduces a quantization error. This is simply the information lost rounding an infinite set of analog voltages to a finite set of digital codes. The more digital codes that the ADC can resolve the less information lost to quantization error.

ADCs are a key component in many systems. Today's ICs are mixed-signal systems consisting of a large digital signal processing (DSP) core surrounded by analog circuitry including RF front-ends and I/O. Fig. 1.1 shows how ADCs connect the analog to the digital domain.

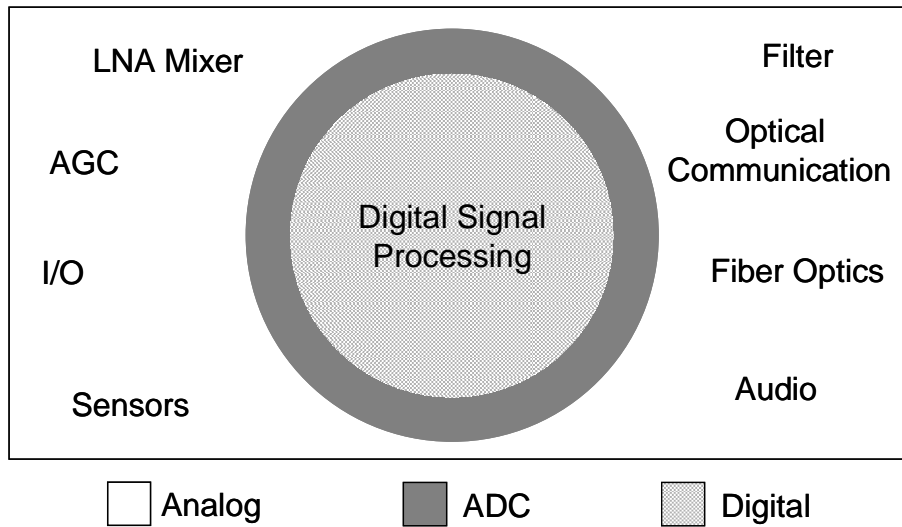


Fig. 1.1. ADC usage

This work concentrates on fast moderate resolution ADCs that are typically used in RF devices, wireline communications and consumer video applications.

## 1.2 ADC Offsets

Mismatch in key components of the analog section of the ADC can cause errors in the output binary code of the converter. These mismatches become larger as the device size shrinks [1]. The main causes of the mismatch and the resultant offsets experienced in ADCs are detailed in this section.

### 1.2.1 Analog Mismatch

For an ADC to accurately convert analog signals to the digital domain its analog components ideally should match. Differential pairs are the basic building blocks of the analog circuits in ADCs and are used extensively in pre-amps, folders and comparators. Across any die, transistors with identical geometries are likely to have slightly different

characteristics. The impact of MOS transistor mismatch becomes more serious as the sizes of the devices are reduced and the available signal swing decreases.

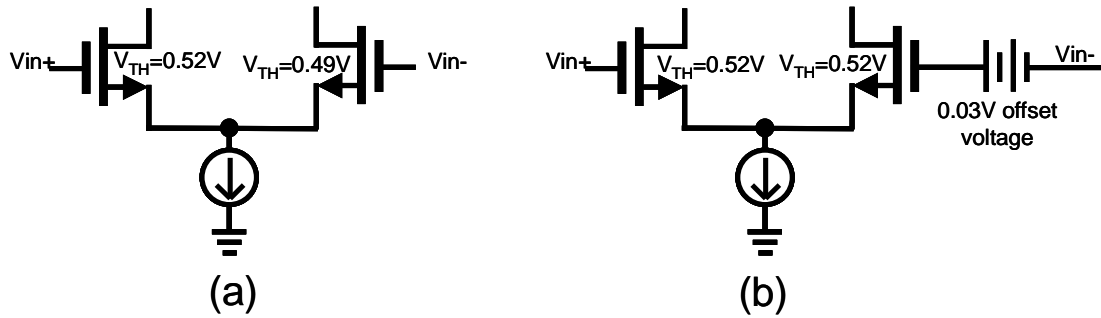


Fig. 1.2. (a) Unbalanced differential pair. (b) Equivalent differential pair with offset voltage.

Consider a single differential pair, Fig. 1.2(a). If the  $V_{TH}$  of the 2 transistors is the same, the pair is said to be balanced, however a  $V_{TH}$  mismatch causes a voltage offset. This can also be considered as a symmetrical differential pair with a voltage source in series with one input - the voltage source being equivalent to the offset voltage, Fig. 1.2(b).

Other key analog components in ADC design include:

- Resistors: resistors are used extensively in the reference voltage resistance ladder. If one resistor in the ladder is larger or smaller than desired then there will be an associated error in the reference ladder voltages.
- Capacitors: capacitors are used to perform voltage addition, subtraction and multiplication in ADCs. The fabricated value of the capacitance can vary from the desired value.

### 1.3 Offset reduction/correction techniques

To allow successful operation of ADCs various analog techniques have been utilized. The major techniques are presented in this section.

#### 1.3.1 Large transistors

The deviation in threshold voltage is seen to be inversely proportional to the transistor area [1].

$$\sigma(\Delta V_{TH}) \propto \frac{1}{\sqrt{WL}} \quad (1.1)$$

It is apparent that using large transistors, with a large W and L, would reduce  $V_{TH}$  mismatch. Therefore comparator and folder errors would be lessened. This is a simple method and requires no extra offset correction circuitry. However chip area and power consumption increase dramatically when this technique is used because of the larger parasitic capacitance. When designing moderate-to-fast speed ADCs, it is desirable to keep parasitic capacitance as low as possible.

#### 1.3.2 Offset Storage

Offset storage can also be used to reduce voltage offset in the comparators [2]. It involves using capacitors to store charge to cancel the offset seen at the inputs. This allows small transistors to be used. Complicated circuitry is required with this method and this doesn't scale well because switches are difficult to implement with a low voltage supply. Also, because the offset has to be refreshed periodically, continuous conversion is not possible.

### **1.3.3 Offset Trimming**

A DAC current can be used to reduce the differential pair offset [3]. This method facilitates continuous conversion but increases analog design complexity.

### **1.3.4 Digital Calibration**

Digital calibration is either implemented on chip power up [15], a ‘one-shot’ calibration or as continuous background calibration [29][35] where the converter is in its normal operation while being calibrated. Digital calibration is typically implemented on-chip using digitally synthesized logic to correct the analog circuitry. Digital calibration is commonly used in pipeline ADCs to correct errors due to DAC and inter-stage gain errors, and capacitor mismatch [34][44].

### **1.3.5 Current ADC design**

Traditional techniques used in ADCs to overcome transistor mismatch include switched capacitor techniques and DAC trimming. Switched capacitor techniques measure the mismatch in the analog circuits and correct these errors by using charge stored on capacitors[29] [33] [35]. DAC trimming is used to compensate for threshold voltage mismatch by digitally controlling the current flowing through the devices [28] [3]. Since modern, smaller transistors suffer badly from gate leakage these techniques are harder to implement. ADCs now primarily use digital circuitry to overcome analog mismatches. One of these methods is to use calibration routines at chip power up to select comparators [15]. The ADC described in [28] uses calibration just one time at chip power up to correct errors in the pre-amplifiers. Other ADCs use continuous background



calibration, for example to adjust reference voltages in a pipeline ADC [33]. Digital error correction to overcome the analog errors is especially common in pipeline ADCs [32-49].

## 1.4 ADC primer

The flash ADC is conceptually the simplest type of ADC and most other ADC architectures are derivatives of flash. An N-bit flash ADC consists of 3 main components:

- $2^N-1$  comparators
- A resistor ladder which provides  $2^N-1$  reference voltages
- An encoder

Each comparator has 2 inputs; one connected to the ADC input, the other connected to the reference ladder. If the input signal exceeds a reference voltage of a comparator, the output of the comparator will be high. The output of the comparator block forms a thermometer code. An encoder converts the  $2^N - 1$  bit code into N binary signals. A 3-bit flash ADC would require 7 comparators and 7 resistors as shown in Fig. 1.3.

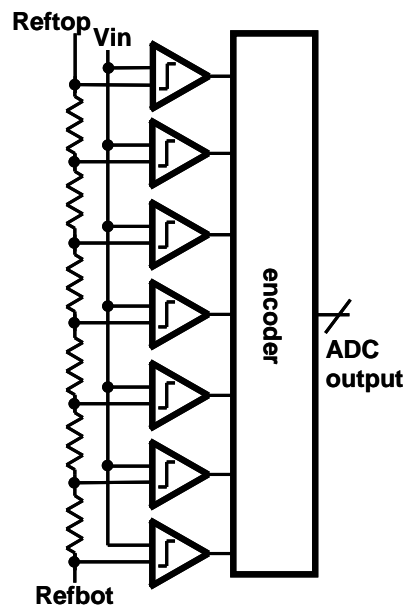


Fig. 1.3. 3-bit flash ADC.

### **1.4.1 ADC metrics**

The main ADC metrics are conversion speed and output resolution. This work concentrates on moderate-speed and moderate-resolution ADCs. These are typically defined as ADCs in the 30MHz-1GHz range with an output resolution of 6-10 bits.

Two main types of tests are applied to characterize ADCs- static and dynamic tests. Conceptually, in the static test a very slow linear ramp voltage is applied to the input of the ADC and output is recorded. For an ADC, the ideal output for this test is a staircase with each step having the same width. Differential and integral non-linearity (INL and DNL) are a measure of the deviation from the ideal step size. INL and DNL values of zero are ideal.

Dynamic testing is performed by applying a high-speed sinusoidal input voltage waveform to the ADC. The reconstructed ADC output should be a sine wave. The quality of the reconstructed output sine wave can be determined by running a Fast Fourier Transform (FFT). This provides some key information including Signal to Noise and Distortion Ratio (SNDR). From this, an effective number of bits (ENOB) of the ADC is calculated. The greater the ENOB of a given ADC the more resolution it can provide.

### **1.5 Recent ADC publications**

Fast, moderate resolution ADCs are typically implemented as flash, folding or pipeline ADCs. While successive approximation ADCs generally have a maximum conversion rate of about 100MHz, faster ADCs have been successively implemented by interleaving 2 or more of these slower building blocks [30][31].

Pipeline ADCs are typically small in area and have low power consumption. They have a large latency as the signal being converted has to pass through all the stages of the

pipeline, anywhere from 3-8 stages, before the final output can be determined. They are frequently used for higher bit resolution, 10-14 bit designs, where conversion rate is in the order of 20MHz-100MHz. But some faster ADCs have also been published, recording conversion rates of 220MHz while providing 10 bits output [38]. Background calibration and digital error correction are used extensively in pipeline ADC design [32-49].

Successive approximation ADCs (SAR ADCs) can be interleaved to form an ADC that can convert at a speed of 600MS/s [31]. SAR ADCs use a large capacitor array in the conversion process. The capacitors usually require some form of digital calibration [30]. Flash ADCs have the highest conversion rate of any ADC architecture [4]. However each additional bit of resolution approximately doubles the area of the flash ADC [5]. Therefore flash ADCs are popular for fast, low-medium resolution ADCs. Pre-amps and resistive averaging have been used to reduce matching requirements in the comparators [13]. DAC trimming has also been used to calibrate the comparators [14]. Flash ADCs with 6 bits resolution have been reported with conversion rates in the GHz range [13,14,17,17,19]. Both two-step and folding ADCs are derivatives of the flash ADC. They trade the speed and parallelism of the flash technique for increased complexity and reduced area. A lot of the techniques such as auto-zeroing and resistive averaging used in the design of flash ADCs also extend to these ADCs. Folding ADCs are usually used to provide 6-8 bits resolution with an operating speed of 200-800MHz. A faster folding ADC has been reported [28] but this uses 2 interleaved folding ADCs.

A figure of merit for ADCs that is commonly used is:

$$E_Q = \frac{P}{2^B(2ERBW)} \quad (1.2)$$

This figure of merit is known as the “quantization energy” and has units of Joules per conversion. P is the power dissipation, B is the high-frequency ENOB (calculated

from SNDR) and ERBW is either the effective resolution bandwidth or the Nyquist frequency, whichever is less.

Table 1.1 shows a summary of the performance of fast moderate resolution presented ADCs presented at the 3 main solid-state circuits conferences, ISSCC, CICC and VLSI symposium, for the years 1997 to 2007. These are all CMOS ADCs implemented in transistor technologies ranging from 0.6 $\mu$ m all the way down to 90nm.

Publication Year	Fs [MHz]	Resolution	Technology [ $\mu\text{m}$ ]	ERBW [MHz]	ENOB @ ERBW	Power [mW]	Figure of Merit [pJ/conv]	Reference
Flash								
1997	48	10	0.5	12	8.7	240	24.05	[6]
1998	400	6	0.35	100	5	190	29.69	[7]
1999	500	6	0.4	125	5.5	400	35.36	[8]
1999	500	6	0.6	30	5	330	171.88	[9]
2000	700	6	0.25	136	5.55	187	14.67	[10]
2000	800	6	0.25	50	5.3	400	101.53	[11]
2000	1000	6	0.35	450	4.1	900	58.24	[12]
2001	1300	6	0.35	600	5.02	55	1.41	[13]
2001	1600	6	0.35	300	5.05	350	17.61	[14]
2001	300	6	0.25	50	5.2	110	29.95	[15]
2001	900	6	0.25	150	4.95	450	52.01	[16]
2002	1600	6	0.18	660	5	328	7.77	[17]
2002	400	6	0.18	100	4.85	70	12.14	[18]
2003	2000	6	0.18	941	5.6	310	3.39	[19]
Folding and Two Step								
1998	400	6	0.5	30	5	200	104.17	[20]
1999	50	6	0.35	1	5.2	20	272.05	[21]
2000	125	8	0.35	62.5	6.4	110	10.42	[22]
2002	100	10	0.12	12.5	8.83	180	15.82	[23]
2004	600	8	0.35	200	7.3	200	3.17	[24]
2004	600	8	0.18	200	6.35	207	6.34	[25]
2004	125	8	0.13	8	7.6	21	6.77	[26]
2007	160	10	0.09	80	9.1	84	0.95	[27]
Interleaved								
2004	1600	8	0.18	797	7.25	1400	5.77	[28]
2004	150	8	0.18	80	7.18	71	3.06	[29]
2004	50	10	0.18	25	9.2	29	0.99	[30]
2004	600	6	0.09	300	4.89	13	0.73	[31]
Pipeline								
1999	100	8	0.5	50	6.68	165	16.09	[32]
2000	80	8	0.5	4.1	7.4	250	180.51	[33]
2001	100	10	0.18	50	9.4	180	2.66	[34]
2001	30	8	0.18	15	6.18	18	8.28	[35]
2003	150	10	0.18	10	8.37	123	18.59	[36]
2003	80	10	0.18	100	9.29	69	0.55	[37]
2004	220	10	0.13	10	9.01	135	13.09	[38]
2005	50	10	0.18	21	8.84	35	1.82	[39]
2005	125	10	0.18	80	8.17	40	0.87	[40]
2005	200	8	0.18	99	7.68	30	0.74	[41]
2006	50	10	0.18	20	8.8	18	1.01	[42]
2006	64	10	0.09	19.2	9	32	0.5	[43]
2006	800	6	0.18	100	5.3	105	13.32	[44]
2007	80	10	0.09	40	8.25	6.5	.26	[45]
2007	30	10	0.09	2	9.4	3.7	1.72	[46]
2007	205	10	0.09	30	8.87	18	1.01	[47]
2007	200	8	0.18	100	6.4	8.5	0.50	[48]
2007	205	10	0.13	102.5	9	92.5	0.88	[49]

Table 1.1 Summary of reported ADCs.

The ENOB recorded is the ENOB at the ERBW and using this number and reported power, the figure of merit for all the ADCs was calculated. Fig. 1.4 shows a plot of figure of merit change versus publication year. The solid line is the best-fit line. The data shows approximately a factor of 100 improvement for figure of merit over the last 11 years.

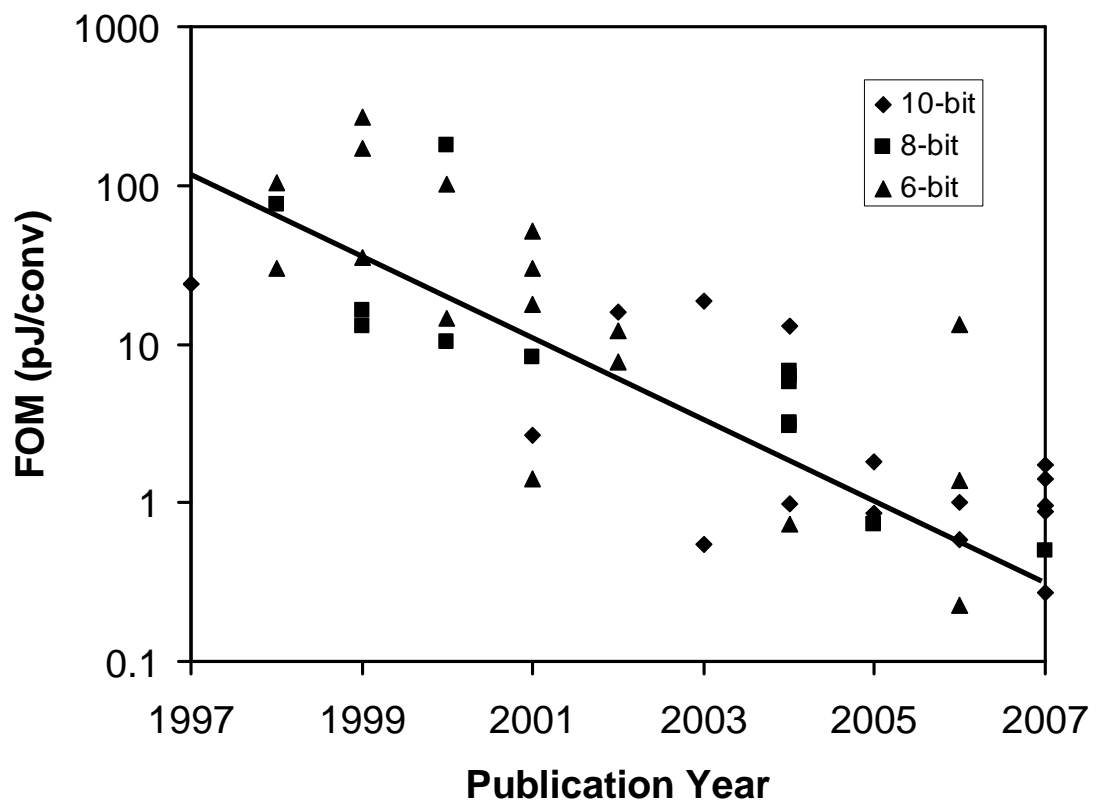


Fig. 1.4. Figure of Merit for published ADCs.

Fig. 1.5 shows the reported active area of the same ADCs. Again the solid line is the best-fit line. The ADCs display only a very slight decrease in area over the last 11 years.

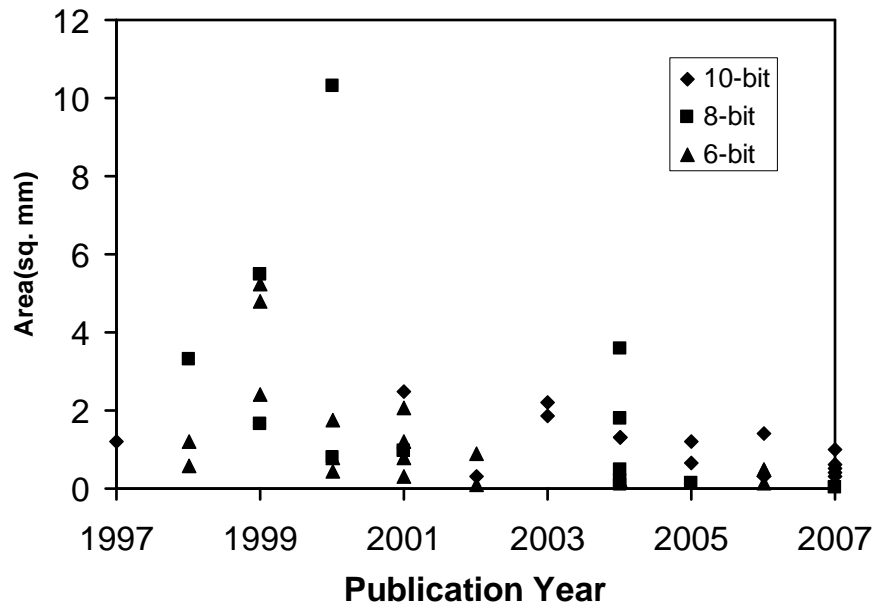


Fig. 1.5. Active area of published ADCs.

## 1.6 Technology Scaling

The past 15 years has seen continuous improvement in CMOS fabrication techniques leading to far smaller transistor sizes. With this reduction in transistor size it has become easier to include extra functionality in a single die. Digital circuits scale particularly well, hence the increasing functionality and speed of microprocessors. While most aspects of digital circuitry are enhanced as transistor sizes scale, analog design, because of the decreased supply voltage, becomes more problematic and performance deteriorates.

As the feature size decreases the speed ( $f_T$ ) and the power efficiency ( $g_m/I_D$ ) of the transistors increase. Digital circuits capitalize on these new improved devices. The x86 processor, from Intel, is an industry leading example of digital circuit innovation. A simple metric for microprocessor circuit performance is the number of transistors per processor. Fig. 1.6 shows the number of transistor in millions in each of the main Intel

x86 processors over the last 11 years [50]. The figure shows about a factor of 100 increase in transistor number.

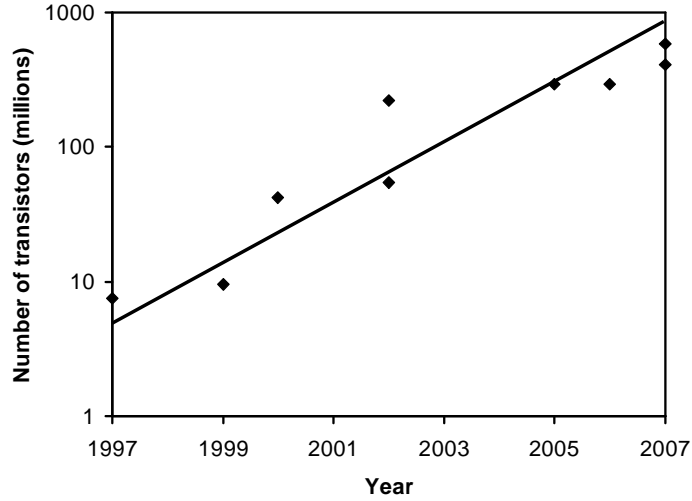


Fig. 1.6. Intel x86 number of transistors per processor.

A common ADC performance metric is the sampling rate of the converter times the number of bits that it outputs. This metric shows the processing power of the ADC and is plotted in Fig. 1.7 for the ADCs in Table 1.1. There is only a slight improvement over the last 11 years.

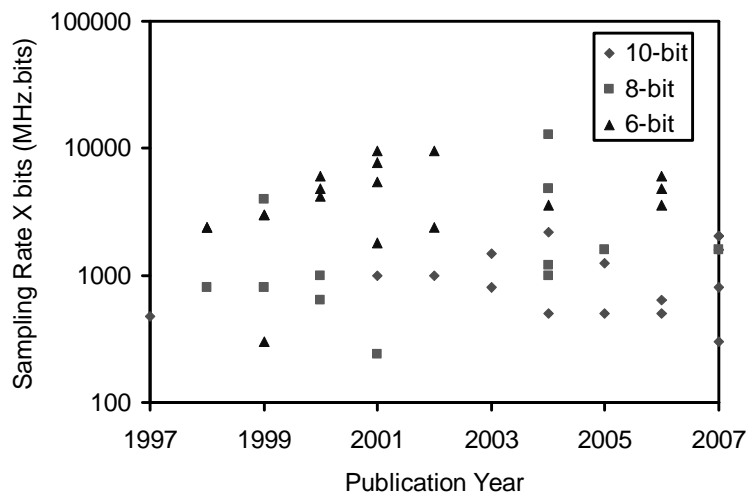


Fig. 1.7. Published ADC performance.



Over the same time frame the digital performance has increased by a factor of 100 while the analog has increased by only a factor of 10. Signal processing can be implemented in either the analog or digital domain. Due to the smaller size of digital transistors and the associated deterioration in analog performance, analog signal processing is rapidly being replaced by digital signal processing. In the past an electronic system consisted of a number of ICs, however using modern CMOS processes much of the functionality can be incorporated onto a single chip. These devices are commonly referred to as a “System on a Chip”, (SOC) ICs. A SOC has the benefits of:

- Reduced power, space, weight and cost
- Enhanced performance
- Increased reliability

Since digital performance is improving at a faster rate than the analog performance, analog circuits are the bottlenecks in SOCs. The relative area and power of analog circuitry in relation to digital circuitry is increasing as the feature size shrinks. This means that analog circuit design and in particular ADC design has to evolve rapidly to overcome these system issues.

## **1.7 Porting Designs**

ADCs are commonly integrated with other devices in a SOC. For faster performance and low power consumption in digitally dominated SOCs it is advantageous to implement these SOC in the latest process technologies. Fig. 1.8 shows the technology nodes used by the Intel x86 processors over the last 11 years. In total there were 6 different processes used, from gate lengths of 0.25 $\mu\text{m}$  in 1997 down to 45nm in 2007. Digital circuits are easier than analog circuits to port from one process to the next.

Therefore a predominantly digital ADC would be desirable for implementation on any SOC designed with a view to migration to finer geometry processes in the future.

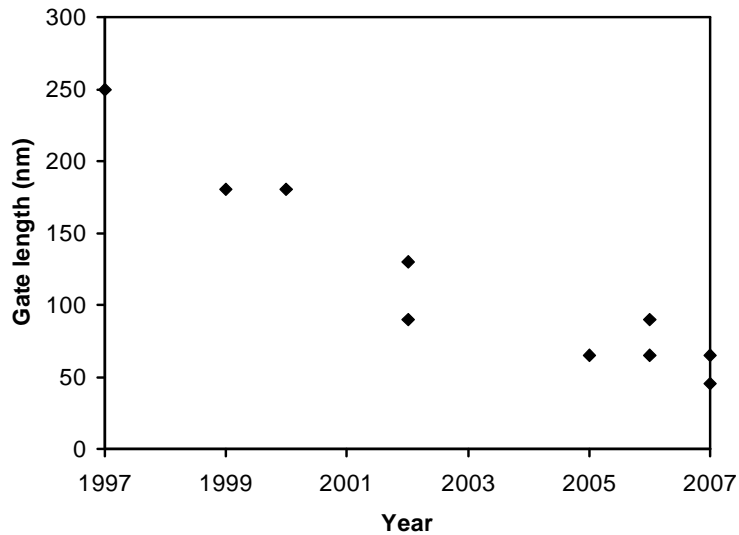


Fig. 1.8. Technology node used for x86 processor.

## 1.8 Contributions of this work

The major accomplishments of this work are:

- The creation of a new folding ADC architecture that is digitally dominated allowing large transistor mismatch to be tolerated so that small devices can be utilized in the signal path.
- The development of modeling techniques, to investigate and analyze the effects of transistor mismatch, folder linearity and redundancy in ADCs.
- The design of a new folder circuit topology that decreases the required power consumption for a given noise budget.
- The design of a resistor ladder DAC that uses a unique resistor layout to allow any shape ladder to be designed.

## **1.9 Thesis Outline**

The motivation for investigating design techniques for digitally calibrated, fast, moderate resolution ADCs is discussed in this chapter. The folding ADC architecture is reviewed in Chapter 2. A new digital calibration technique is presented in Chapter 3 and the prototype results of 8-bit folding ADC implemented in TSMC 0.18 $\mu$ m are presented. In Chapter 4 a second prototype, fabricated in ST 90nm, is described. This design uses digital calibration and a unique folder scheme to build a 400MS/s, 9-bit ADC. The conclusion and suggestions for future work are given Chapter 5.

## Chapter 2

### Folding ADC review

#### **2.1 Introduction**

In a flash ADC, fast conversion speed and a high input bandwidth can be achieved thanks to the parallel nature of the converter. The input signal is applied to all comparators at once and is converted in a single clock cycle. Flash ADCs are advantageous in nanometer processes, since, unlike pipeline ADCs, operational amplifiers are not required. Very fast ADCs have been reported using the flash architecture; however these are usually confined to 4-6 bits of resolution [13,14]. Where more than 6 bits is required the number of comparators in a flash ADC becomes prohibitive and folding [20] becomes an attractive option. Folding preprocesses the analog signal before it enters the comparators, reducing the total required number of comparators by the degree of folding performed. In this manner, fast, moderate-resolution folding converters have been presented [20-27]. These devices are typically required for various applications such as RF quadrature demodulation front-end (e.g. 8-bits, 40MS/s [22]), wireline communications such as 10G Ethernet (e.g. 8-bit, 700MS/s [25]) and consumer video applications (e.g. 10-bits at 54MS/s [23]).

## 2.2 Folding ADC architecture

The flash ADC is conceptually the simplest ADC architecture and most other ADC architectures can be considered as derivatives of the flash scheme. A flash ADC, Fig. 2.1(a), consists of 3 main components; comparators, a resistor ladder and an encoder.

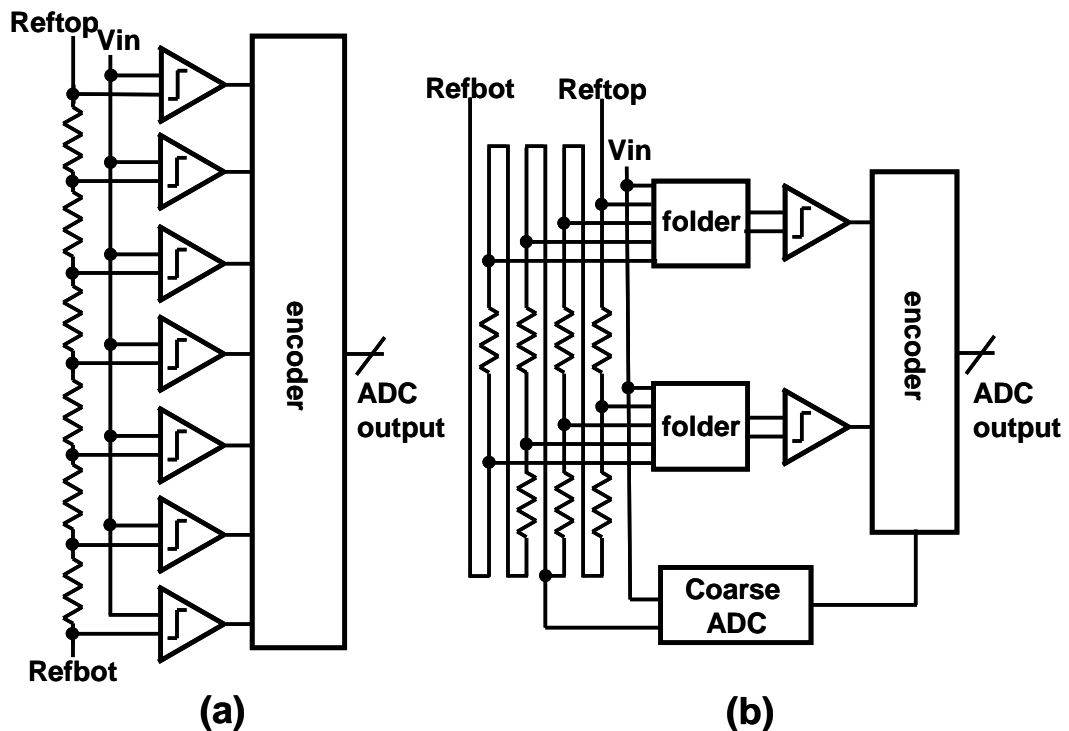


Fig. 2.1. (a) 3-bit flash ADC with 7 comparators. (b) 3-bit folding ADC with 2 comparators.

The output of the comparator block forms a thermometer code, where the location of the transition from high outputs to low, known as the meniscus, is related to the value of the input signal. Although there is a separate comparator for each ADC code, at any given time only a few comparators (those around the transition of the thermometer code) provide useful information. A folding ADC applies analog pre-processing to reduce the

number of comparators and a block called a folder does this pre-processing. A 3-bit folding ADC, folding by 4, requires just 2 comparators, as shown in Fig. 2.1(b). A separate coarse ADC generates the MSB. In general, a N-bit folding ADC, folding by  $f$ , requires  $2^N/f$  folders and comparators, while the coarse ADC requires  $\log_2(f)-1$  comparators.

### 2.2.1 Folder

A folder, folding by 4, Fig. 2.2(a), consists of an even number of differential pairs whose outputs are connected in alternating fashion. One input of the differential pair is connected to the analog input,  $V_{in}$ , the other input is connected to one of the four reference voltages,  $vr1-vr4$ . Each current source supplies a fixed current,  $I_t$ .

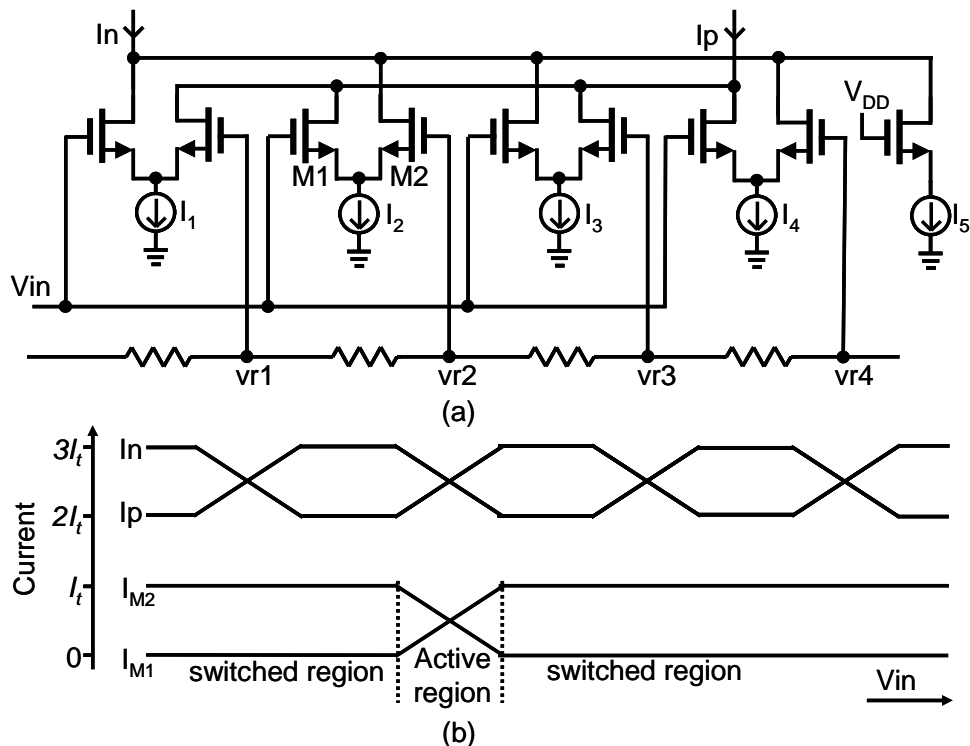


Fig. 2.2. (a) Four cross coupled differential pairs. (b) Folder output currents cross four times, each crossover point comes from a single differential pair.

The polarity of the differential output signal changes each time the input voltage,  $V_{in}$ , reaches a reference voltage level. In this way, the input signal is “folded” at each reference voltage. Fig. 2.2(b) shows the folder outputs,  $I_p$  and  $I_n$ , as the input voltage,  $V_{in}$ , is swept from low to high.  $I_p$  and  $I_n$  change polarity four times. The fifth current source,  $I_5$ , acts as a level shift so that  $I_n$  and  $I_p$  cross [20]. The differential output formed by  $I_p$  and  $I_n$ , is fed to a comparator.

The output currents of a single differential pair are also shown in Fig. 2.2(b). For  $V_{in}$  close to  $v_{r2}$ , current flows in both  $M1$  and  $M2$ , this is the *active region* of the differential pair. For  $V_{in}$  much less than  $v_{r2}$ ,  $I_t$  flows in  $M2$  and for  $V_{in}$  a lot greater than  $v_{r2}$ ,  $I_t$  flows in  $M1$ . These regions are referred to as the *switched region* of the differential pair.

### 2.2.2 Cascaded folding

The larger the folding factor,  $f$ , the greater the reduction in the number of comparators required in a folding ADC. A large value of  $f$  is difficult to achieve in a single stage because it is hard to avoid overlapping of adjacent folding differential pairs. This is especially prevalent in finer geometry CMOS process with shrinking supply voltages and reduced available signal swings.

The use of cascaded folding removes the need to generate all the folds in a single stage. Cascading folders together allows the folds to be generated in connected stages. Cascading folders simply means that the output of the first stage of folders is folded again in the second stage. The overall folding factor is the product of the folding performed by each of the two stages. Cascaded folding is analogous to a double reduction gear system, Fig. 2.3 [51]. Here the input is attached to the first gear which turns in proportion to its

relative motion with respect to the reference ladder. One rotation of the large gear causes two rotations of the small gear and four rotations of the second small gear. Gain is also illustrated in Fig. 2.3(a) as the output of the small gear in the first stage shares the same axle as the large gear in the second stage. The equivalent CMOS block diagram is shown in Fig. 2.3(b). The first stage consists of 2 folding-by-4 blocks and their outputs are folded by 2 to create the overall folded-by-8 output. Cascading folding is commonly used in folding ADCs [21][27][51]

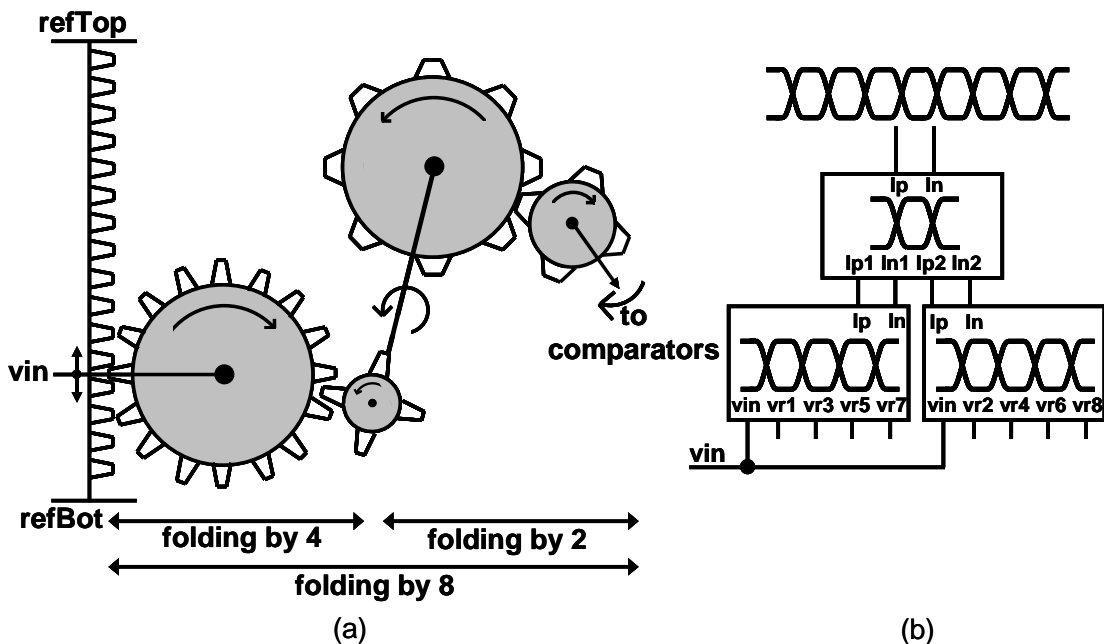


Fig. 2.3. (a) Cascaded folding-by-four generated by two stages of folding-by-two [51] (b)

Equivalent CMOS block diagram.

### 2.2.3 Interpolation

Interpolation can be used to reduce the number of folders required in a folding ADC [20-22]. Fig 2.4 shows a simple example of interpolation. Every second folder is omitted and the missing information is recovered by interpolation. Interpolation can be easily implemented by inserting resistors between the outputs of remaining folders



(folders 1,3,5) and the taps between the inserted resistors recreate the omitted folder outputs (folders 2,4). In this example the outputs of folders 1, 3 and 5 are approximately linear over a limited range near each zero-crossing. The interpolated signals show some distortion, the output is not linear near the zero-crossing in the recreated folder outputs, but the distortion suffered is not a problem as long as the location of the zero-crossing itself is not affected. As the overall number of folders is decreased the power consumption and input capacitance is reduced, however the output impedance of the folders is decreased.

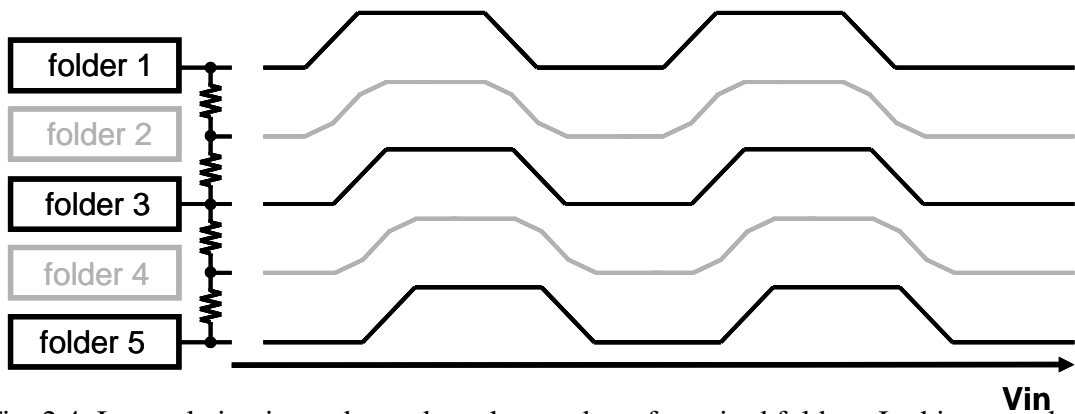


Fig. 2.4. Interpolation is used to reduce the number of required folders. In this example every second folder is omitted.

## 2.2.4 Encoder

The encoder combines the output of the folding comparators and the coarse ADC to produce a binary ADC output. A 4-bit folding ADC, folding by 4, consists of 4 folders, 4 comparators and a coarse ADC, Fig. 2.5(a).

The output of the coarse ADC is the MSB, B[3]. The coarse ADC in this instance is a single comparator with a trip-point at the middle at the reference ladder. The other bits, B[2:0], are evaluated from the comparator outputs. The output *CI* provides MSB-1,

B[2]. Each of the folders has 4 unique reference voltages providing the comparator outputs  $C1$  to  $C4$ , Fig. 2.5(b).

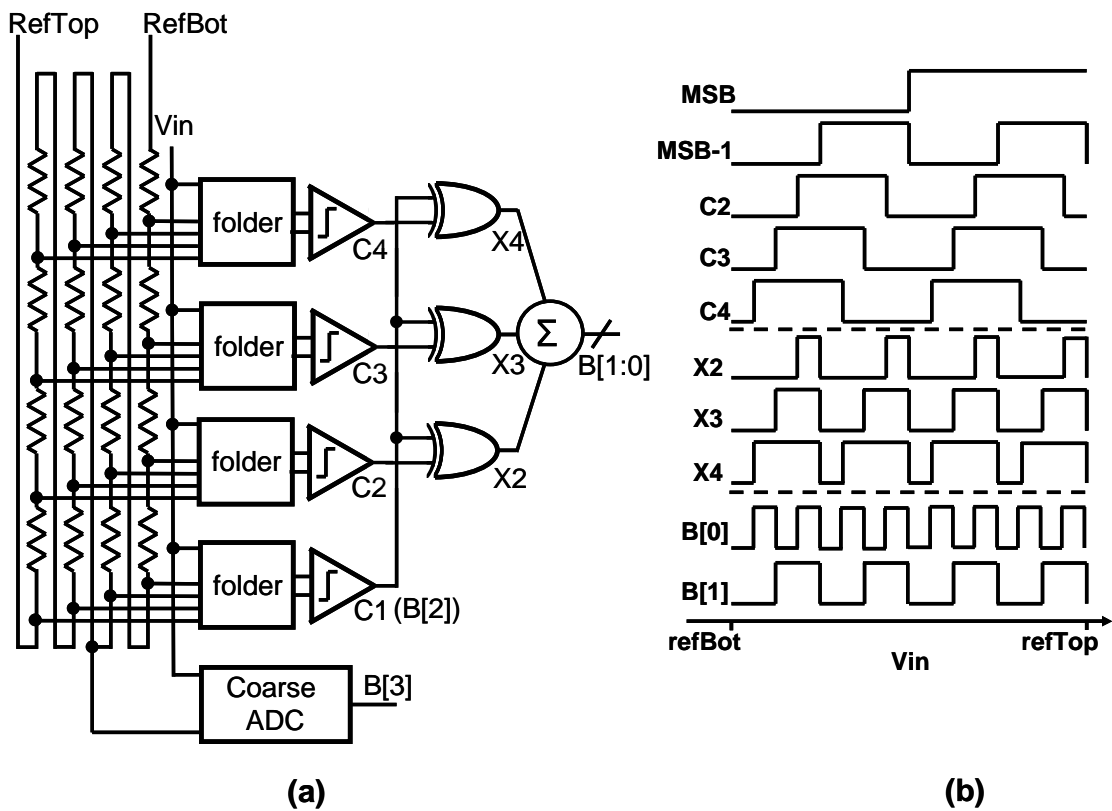


Fig. 2.5. (a) 4-bit folding ADC has 4 folders and comparators. (b) Output waveforms.

The simplest encoder architecture inverts the polarity of the folding comparator outputs,  $C2$ - $C4$ , when  $C1$  is high. XOR gates can readily accomplish this. An adder converts the XOR gate outputs to the two least significant bits (LSBs),  $B[1]$  and  $B[0]$ .

### 2.2.5 Coarse ADC Accuracy

In the simple folding encoder scheme described above, it is assumed that the zero-crossing provided by the coarse ADC aligns correctly with the mid-scale zero-crossing of

C1. Even with relatively large transistors, this can be difficult to achieve in practice. Fig.2.6 [20] shows the ADC output with a misalignment of  $\Delta V$  between the output of the coarse ADC and the MSB2 folder, B[3] and B[2].

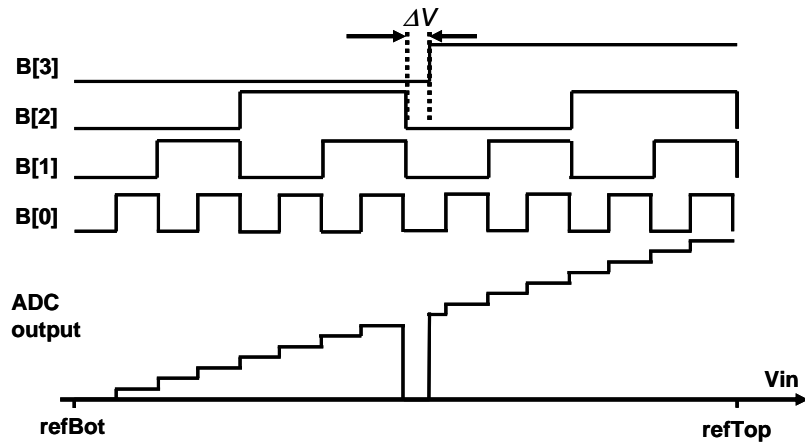


Fig. 2.6. Offsets in coarse ADC cause large errors in output [20].

In practice two coarse ADC comparators and a ‘bit sync’ are used to generate the MSB [20], Fig. 2.7. The zero-crossings of the two comparators B3a and B3b are nominally placed as shown. Combining the comparator outputs with the output of B[2] generates B[3]. This technique is resilient to large offsets in the coarse ADC comparators.

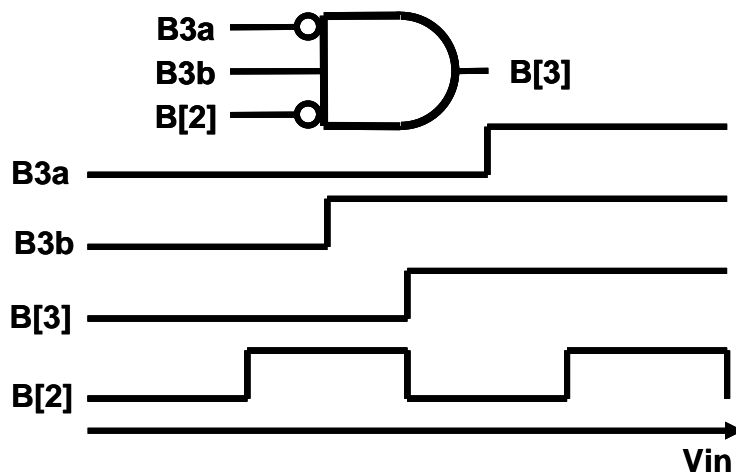


Fig. 2.7. Use of 2 comparators in coarse ADC to align coarse ADC with folders.

## Chapter 3

### A digitally calibrated 8-bit folding ADC

#### 3.1 Introduction

Threshold voltage,  $V_{TH}$ , mismatch causes mismatch in the differential pairs used in the folders leading to poor folding ADC performance. Traditional methods used to overcome mismatch in folding ADCs are implemented in the analog domain. The simplest approach is to use large devices, as mismatch is inversely proportional to gate area [1]; this however leads to large, slow devices with large input capacitance. DAC trimming has also been used in folding ADCs [28] but this increases the complexity of the folder design.

This chapter describes a digital technique, based on redundancy and re-assignment to calibrate and correct an 8-bit folding ADC. Unlike other folding ADC calibration techniques [52] this approach also compensates for unintended non-linearities and distortion in the folders. Much as the technique in [53] overcomes amplifier non-linearity through digital processing, this approach trades analog accuracy and distortion for simple digital calibration techniques. Digital calibration decouples analog accuracy from ADC accuracy allowing small, minimum length devices to be used in the signal path. A fully-integrated self-calibrated prototype 500MS/s 8-bit converter is fabricated in 0.18 $\mu$ m digital CMOS.

The digital calibration scheme is briefly described in section 2. The ADC implementation is described in section 3. Issues in design of the critical analog blocks, particularly distortion and matching requirements, are discussed in section 4. Measured performance results of the prototype are shown in section 5.

### 3.2 Digital Calibration Scheme

The new architecture works by generating redundant zero-crossings. At startup, a calibration engine controls a DAC that searches the set of available zero-crossings and selects and enables circuitry that generates the  $2^N-1$  most appropriate zero-crossings. Redundancy can ensure a high yield even with low accuracy analog components, decoupling speed and accuracy [54]. Calibration information is stored in SRAM. During normal conversion the calibration engine and unselected redundant circuitry is powered down.

A conventional N-bit folding ADC generates a folder zero-crossing for each of the  $2^N-1$  codes. In this scheme, a calibration routine selects the  $2^N-1$  most appropriate crossings from the set of available redundant crossings. Reassignment allows any zero-crossing to be assigned to any code.

A folding-by-4 folder is used in this design. Folding by 4 reduces the required number of comparators by a factor of 4. With an input applied to the folders of frequency  $f_{in}$ , the input applied to comparators is at least 4 times  $f_{in}$ . Increasing the folding factor further decreases the required number of comparators but requires a comparator with a larger bandwidth.

Matlab scripts, modeling transistor mismatch and folder nonlinearity, were run to determine the degree of redundancy required. 1000 ADCs were generated for each of

four different values of standard deviation of comparator offset (0.5, 2, 5 and 10LSBs) and the yield was determined as redundancy was increased. A ‘good’ 8-bit ADC is one with a ENOB of more than 7.5 bits. If a redundancy of 4 is used, a yield of over 96% can be attained even with standard deviation of offset of 10LSB (Fig. 3.1.). Increasing the redundancy above 4 slightly increases the overall yield but also increase power and area required.

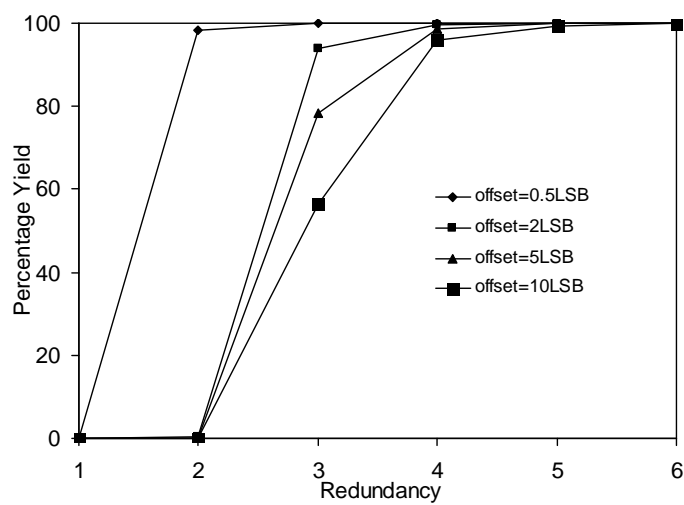


Fig. 3.1. Yield versus redundancy for an 8-bit ADC.

To generate four redundant zero-crossings for each code transition the following configurations could be used;

- a) four redundant folders each connected to single comparator
- b) a single folder connected to four redundant comparators
- c) two redundant folders each connected to two redundant comparators

In each case the number of comparators increases by a factor of 4 but the number of folders differs. Reducing the number of folders decreases power consumption however connecting a single folder to 4 comparators would mean that the zero-crossings generated

by the four comparators would show a large dependence on the mismatches of the single folder. For this design a redundancy of 2 is used for the folders and each of the folders is in turn connected to 2 redundant comparators redundantly, assigning four zero-crossings to each ADC code transition. Details on the folder and comparator matching requirements are given in section 3.4.3.

Fig. 3.2 shows a 3-bit folding ADC using redundancy. Instead of using large, high accuracy folders as in Fig. 2.1(b), small redundant folders are used. Even though the number of folders is doubled, (i.e. a redundancy of 2 is used), since these folders have low matching requirements the overall area is about the same. Redundancy is also used for the comparators. Each folder is connected to two low accuracy comparators. Again these comparators have relaxed matching requirements and can be made small.

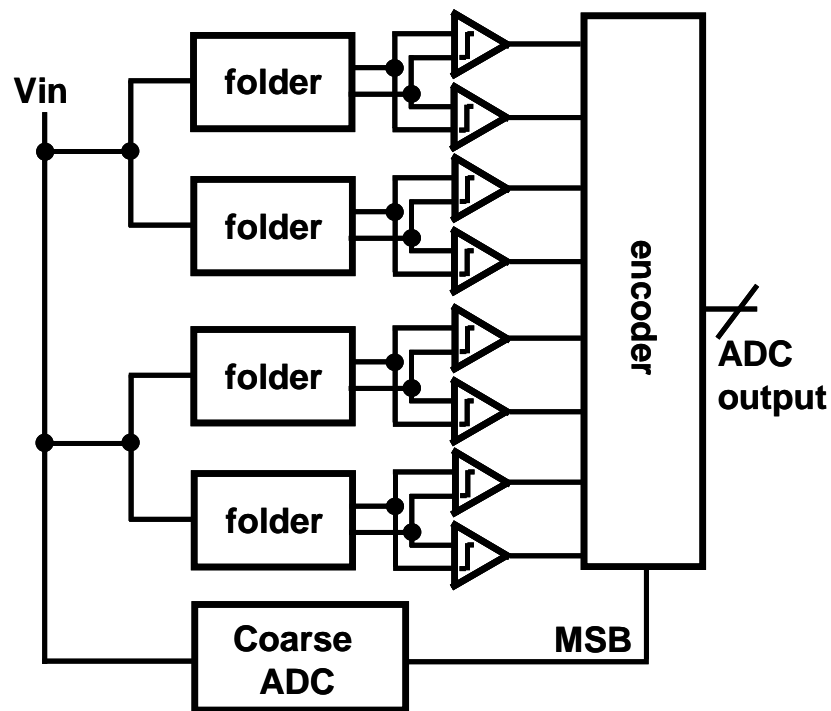


Fig. 3.2. Small, low accuracy redundant folders and comparators are used to generate redundant zero-crossings.

To illustrate the technique, Fig. 3.3 shows a simple example of redundancy and reassignment in a 2-bit flash ADC [54]. Four comparators are redundantly assigned to each code transition. Comparators, *1A*, *1B*, *1C* and *1D* are nominally assigned to code transition 1, Fig. 3.3(a), similarly *2A-2D* to code 2 and *3A-3D* to code 3. Offsets in the comparators cause the trip-points of the comparators to deviate from the nominal values, Fig. 3.2(b). In this example comparator *1A* has a trip-point close to the ideal value and is assigned to code 1. Comparator *2D* is selected for code 2. In an example of reassignment comparator *2A*, nominally assigned to code 2, is selected to represent code 3.

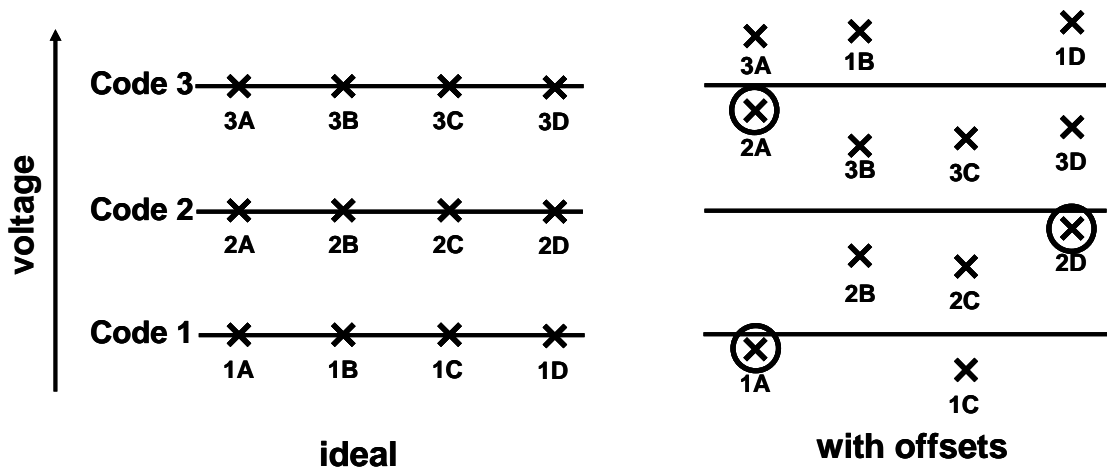


Fig. 3.3. (a) Nominal comparator trip-points with a redundancy of 4 comparators for each code. (b) Offsets move trip-points. Example of reassignment, 2A is assigned to code 3.

Fig. 3.4 describes how redundancy and reassignment are applied in a folding ADC. The figure shows two ideal folding waveforms *A*, *B*, as well as examples of low accuracy folding waveforms from redundant folder pairs (*A1* and *A2*, *B1* and *B2*) that approximate *A* and *B*. By digitally selecting the most appropriate individual zero-crossings, the correct folder characteristic is constructed. Similar to a conventional folding ADC, the calibrated ADC also incorporates a coarse ADC or cycle pointer, but in



this case the coarse ADC functions in a somewhat different manner. In this example, the coarse ADC divides the ADC input range into four *quadrants*; labeled 0-3. If a zero-crossing is selected, then a folder output is enabled for an entire *quadrant*. In an example of reassignment, a zero-crossing from *B2*, *X1*, is selected for folding waveform *A*. Since no zero-crossings are selected from folder *A2*, it is powered down.

Calibration must ensure that a folder with two zero-crossovers in one *quadrant* is not selected, and also must correct crossover polarity. In *quadrant 3*, folder *B1* has an ideal zero-crossing (*X2*) for *A*. However if the output of folder *B1* is enabled in *quadrant 3*, two zero-crossings (*X2* and *X3*), not one, would be selected, and therefore *A1* is selected instead.

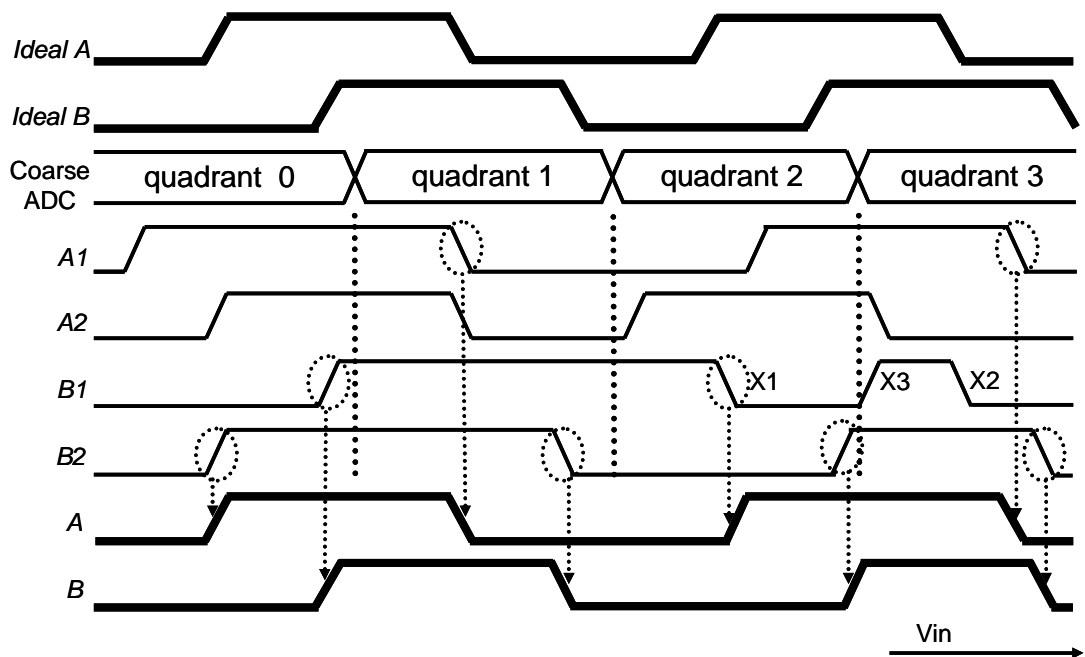


Fig. 3.4. Redundancy and reassignment of folder zero-crossings.

To count the number of zero-crossings a folder has in a given *quadrant* a *boundary check* is performed. During the boundary check, the folder output is tested at

the top and bottom of the *quadrant* of interest. For example, *B1* is 0 at the bottom of *quadrant 3*, and is 0 at the top of *quadrant 3*. Because *B1* has the same value at both ends of *quadrant 3*, it means that *B1* has 2 zero-crossings in this *quadrant*. On the other hand, a *boundary check* on *B2* yields a 1 and 0 respectively at the bottom and top of *quadrant 3*. This means that *B2* has only a single zero-crossing in *quadrant 3*. The calibration also corrects zero-crossings that occur close to an ideal zero-crossing voltage but with the wrong polarity - the zero-crossing of *B1* in *quadrant 2*, *X3*, is inverted by the calibration logic. Comparator redundancy is also used in the digital calibration scheme. Redundant comparators increase the number of zero-crossings. A conventional 8-bit folding ADC, with fold-by-4 folders, consists of 64 folders connected to 64 comparators. The prototype incorporates 128 folders, feeding 256 comparators, generating 1024 potential zero-crossings as shown in Fig. 3.5.

At power-up, the ADC undergoes calibration during which the resistor ladder is configured as a 10-bit search DAC. The calibration engine controls the DAC to search the available 1024 zero-crossings for the best 255 zero-crossings. The calibration results are stored in the *Zero-Crossing Selector* and *Polarity Corrector* blocks, both implemented as SRAM.

The prototype uses a 3-bit coarse flash ADC to divide the ADC conversion range into *octants*. Although a 2-bit coarse ADC could be used dividing the ADC range into *quadrants*, the use of a 3-bit rather than 2-bit coarse ADC decreases the risk of having more than one zero-crossing in a given segment. The coarse ADC itself is calibrated from a set of 21 available redundant comparators. The locations of the coarse ADC trip-points are stored in a *Coarse ADC Trip-point Register*. Synchronization between the coarse ADC and folders is achieved through overlap and digital correction.

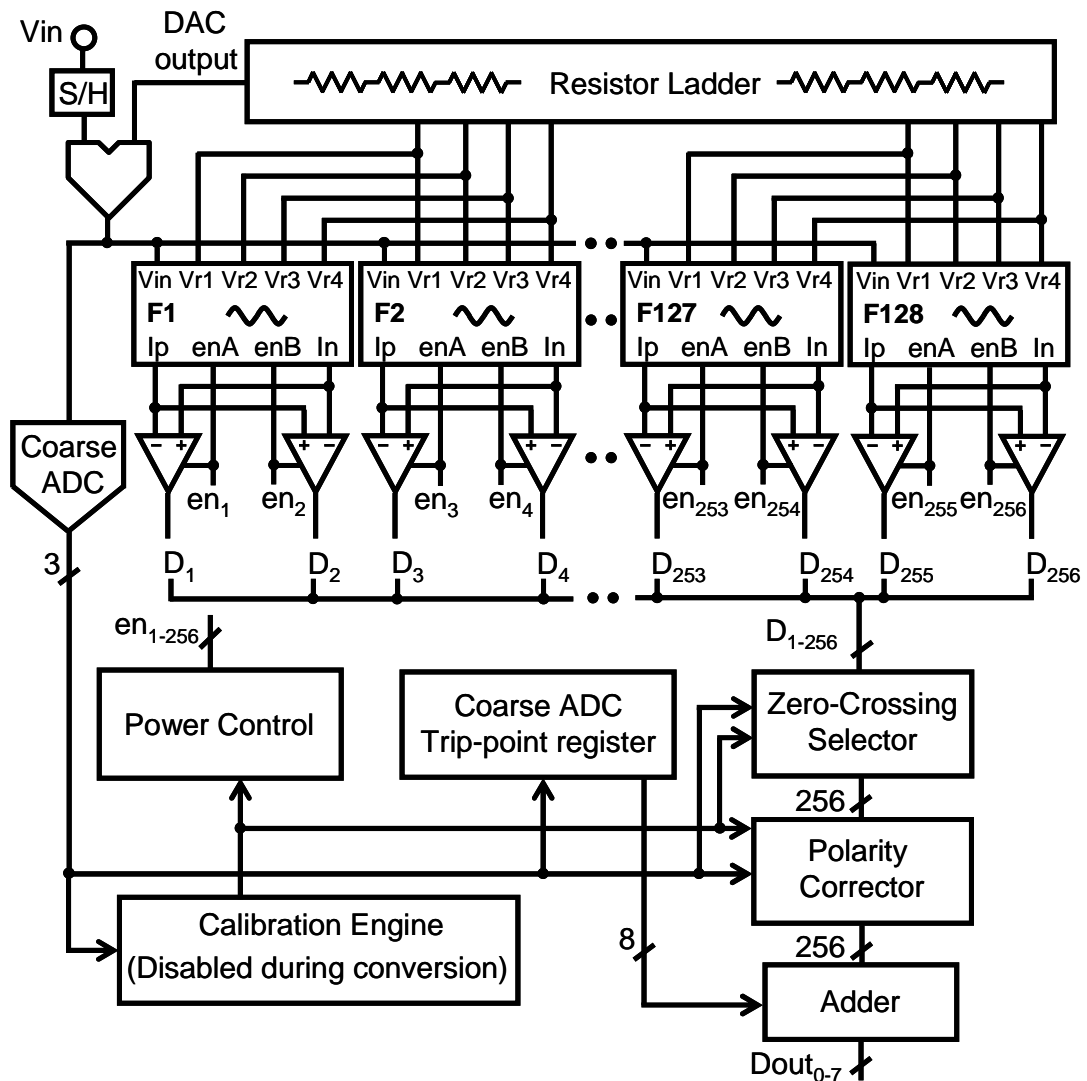


Fig. 3.5. Block diagram of digitally calibrated folding ADC.

In conversion mode, the DAC and the calibration engine are disabled. The *Power Control* block, shown in Fig. 3.5, drives a 256 bit bus,  $en[1-256]$ , that powers down unselected redundant circuitry. Comparator outputs are decoded on-chip using information from the coarse ADC and the information in the *Polarity Corrector* and *Zero-Crossing Selector* memories, (details of these memories are discussed in the next

section). An adder sums the decoded comparator outputs and combines this sum with the output of the *Coarse ADC Trip-point Register* to give the 8-bit ADC output.

### 3.3 Implementation

This section describes the implementation of the key building blocks of the converter.

#### 3.3.1 Zero-Crossing Selector and Polarity Corrector memories

The coarse ADC divides the conversion range into 8 sections or *octants*, and there are 256 comparators. To be able to select zero-crossings in each of the *octants* requires the *Zero-Crossing Selector* block to use an SRAM of 8 words of 256 bits. As an example, Fig. 3.6 shows 4 rows of the memory array loaded with the correct values to select the highlighted zero-crossings.

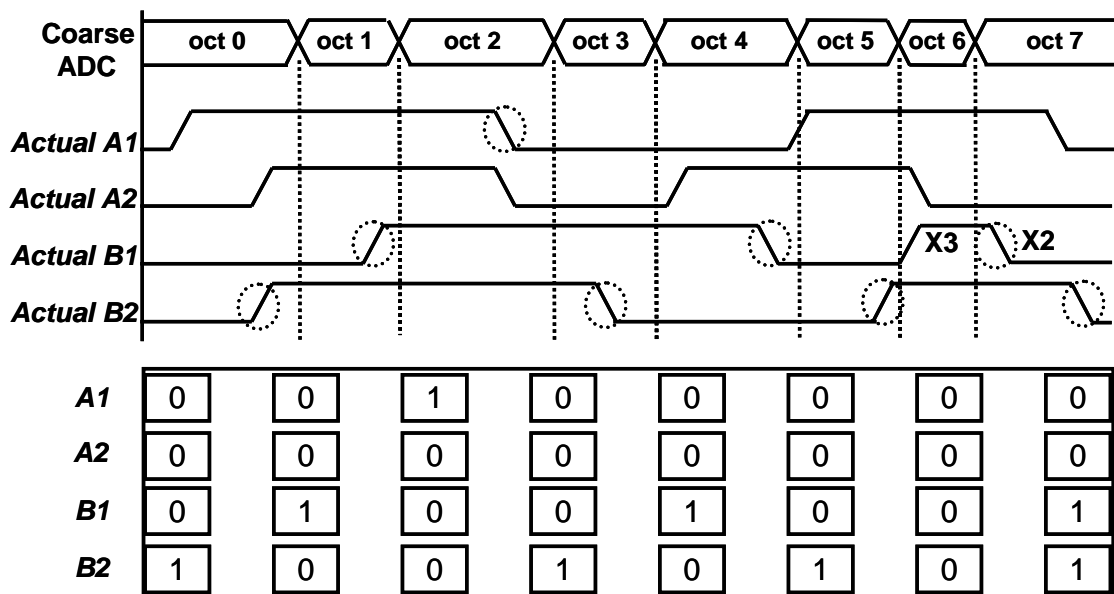


Fig. 3.6. Example of Zero-crossing memory, 8 words required for 3-bit coarse ADC.

As discussed earlier in the *Digital Calibration* section, the use of a 2-bit coarse ADC would mean that selecting *X2* would also select *X3*. Here, the use of a 3-bit flash ADC to identify *octants* rather than a 2-bit flash ADC that identifies *quadrants*, enables *X2* to be selected independently of *X3*. However the increase in coarse ADC resolution causes an increase in the SRAM size.

Information, on whether a *1-0* or a *0-1* zero-crossing is selected, is stored so that the encoder can properly encode the correct 8-bit result. During the *boundary check*, registers temporarily store the value of the comparator output at the top and the bottom of the *octant* in which it lies. By examining these registers it is straightforward to determine whether a zero-crossing is a *1-0* or *0-1* transition. Zero-crossings from the same folder selected in adjacent *octants* always have an opposite switching polarity, i.e. if one is *1-0* the other must be *0-1*. This reduces the SRAM requirements so that a 4 word, 256 bit configuration is sufficient. Fig. 3.7 shows the *Polarity Corrector* for the waveforms displayed in Fig. 3.6. A *0-1* transition in an odd numbered octant is a 1 as is a *1-0* transition in even numbered octant.

	<b>Octant 0/1</b>	<b>Octant 2/3</b>	<b>Octant 4/5</b>	<b>Octant 6/7</b>
<b>A1</b>	0	1	0	0
<b>A2</b>	0	0	0	0
<b>B1</b>	1	0	1	0
<b>B2</b>	0	0	1	0

Fig. 3.7. Polarity Corrector memory consists of 4 words of 256 bits. Here 4 bits are shown.

### 3.3.2 Coarse ADC calibration

The coarse ADC is calibrated first. An example comparator search is shown in Fig. 3.8. Generating an input to the bank of redundant comparators with the search DAC, the search range expands beyond the ideal value until a comparator is found. The location of the comparator trip-point is stored in the *Coarse ADC Trip-point Register*.

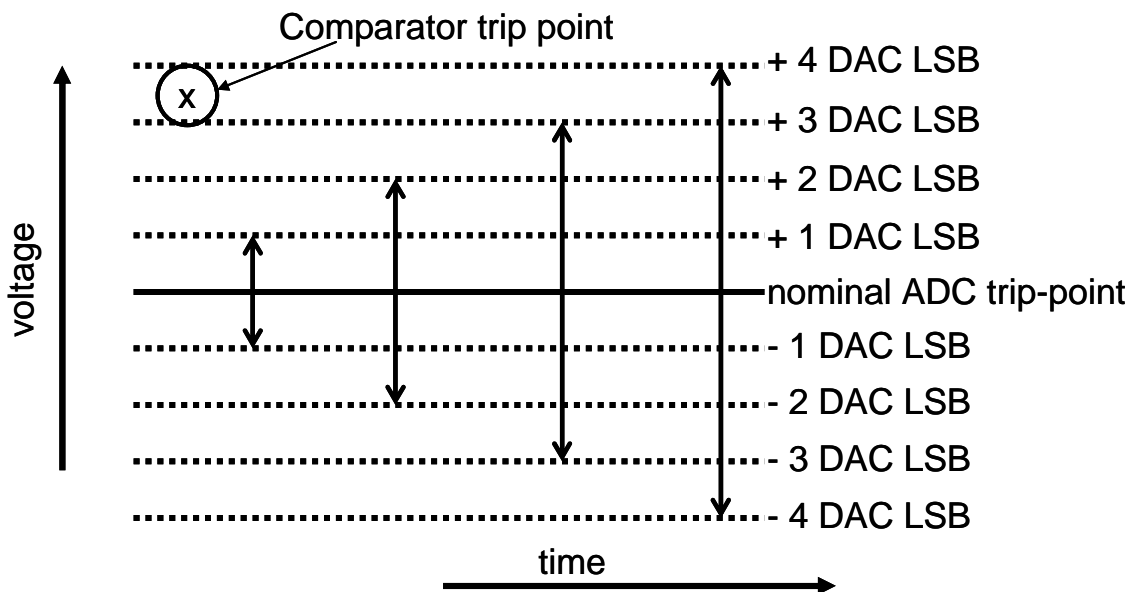


Fig. 3.8. Coarse ADC comparator search. In this example, a comparator is found on the 4<sup>th</sup> search iteration.

### 3.3.3 Sample-and-hold

A folding ADC, with an input frequency  $F_{IN}$  and folding by 4, applies an input to the comparators at a frequency of 4 times  $F_{IN}$ . A 3.3V sample-and-hold is used at the input to the chip, to relax the requirements on the comparator input bandwidth [55] and to reduce current slewing at the folder output, Fig. 3.9.

Instead of using a dedicated hold capacitor,  $C_L$  consists of the input capacitance of folders and input wiring. Although the input capacitance of the folders varies non-linearly with applied voltage, thanks to the small folder devices the folding input capacitance is only 1.3pF of the total 2.1pF. M2 is a dummy transistor and cancels charge injection when M1 turns off.

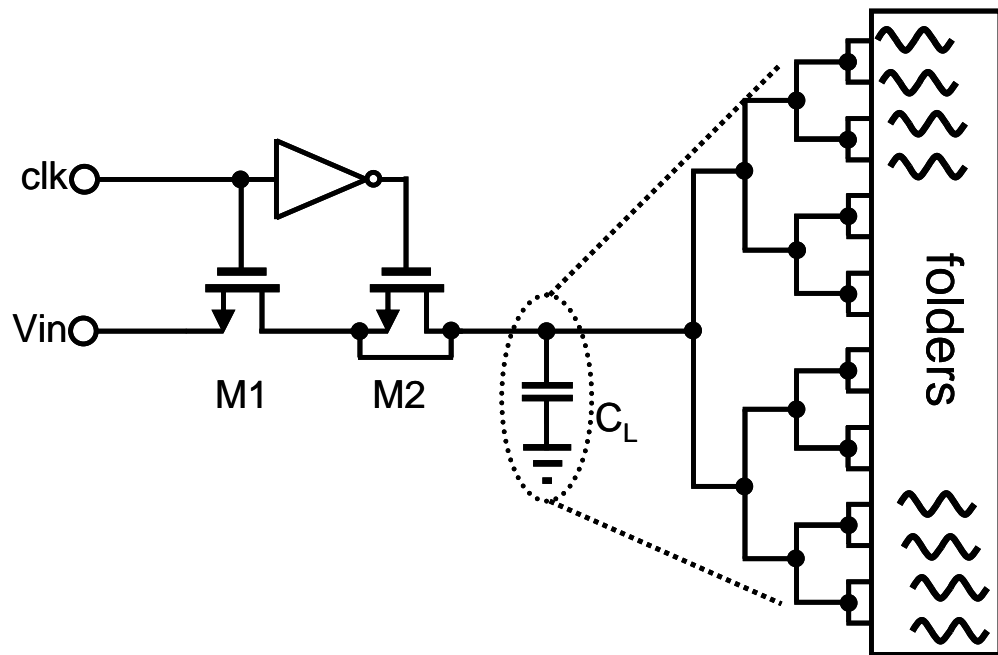


Fig. 3.9. The parasitic capacitance of the routing and folder inputs forms the sample-and-hold capacitance.

### 3.3.4 Resistor ladder DAC

A resistor string generates the ADC reference voltages and during calibration the resistor string is configured as a DAC. Each folder requires four reference voltages, equally spaced over the entire reference range. Bending the metal 1 resistor ladder simplifies the connection of the reference taps to the folders but this adds extra parasitic resistance at the corners, Fig. 3.10(a). These corner errors are eliminated by building a

ladder comprised of 'L' shaped metal resistors so that the ladder can take almost any shape, Fig. 3.10(b). Switches for the DAC and substrate connections fit into the 'cavities' of the resistor string, Fig. 3.10(c).

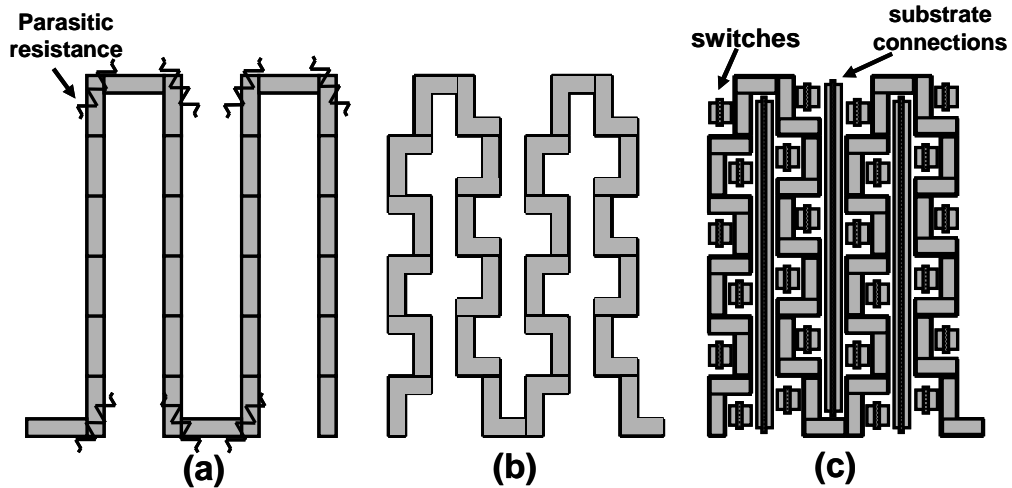


Fig. 3.10. (a) Resistor ladder suffers from parasitic corner resistance. (b) Resistor ladder with 'L' shaped resistors. (c) DAC switches and substrate connections are embedded in ladder cavities.

Averaging is used to increase the effectiveness of the calibration, since the analog circuitry may be subject to noise giving spurious zero-crossings. Averaging is performed by applying the same test voltage twice during the zero-crossing search. If the same zero-crossing is found on both searches, it is *boundary checked*. However if no zero-crossing is found twice, but one is found in either the first or second search, then it is *boundary checked*. Likewise, the *boundary check* routine is implemented twice. Since a failure in the boundary check search could lead to a glitch on the ADC output, a zero-crossing is only selected if it passes the *boundary check* on both iterations; otherwise the search range is extended. To relax the DAC settling time requirement the DAC is allowed nine



clock cycles before the comparators are clocked. Details of the DAC precision are given in the next section.

### 3.3.5 Zero-crossing search

The zero-crossing search algorithm is similar to the coarse ADC calibration. For each code, the search begins by looking for a zero-crossing between -1 DAC LSB and +1 DAC LSB of the ideal value. If no zero-crossing is found then the search is extended to -2 DAC LSB to +2 DAC LSB and so on. As discussed earlier the *boundary check* ensures that selecting a zero-crossing in a given octant only enables a single zero-crossing. If the *boundary check* fails then the search range is extended. The overall ADC calibration process is summarized in Fig. 3.11.

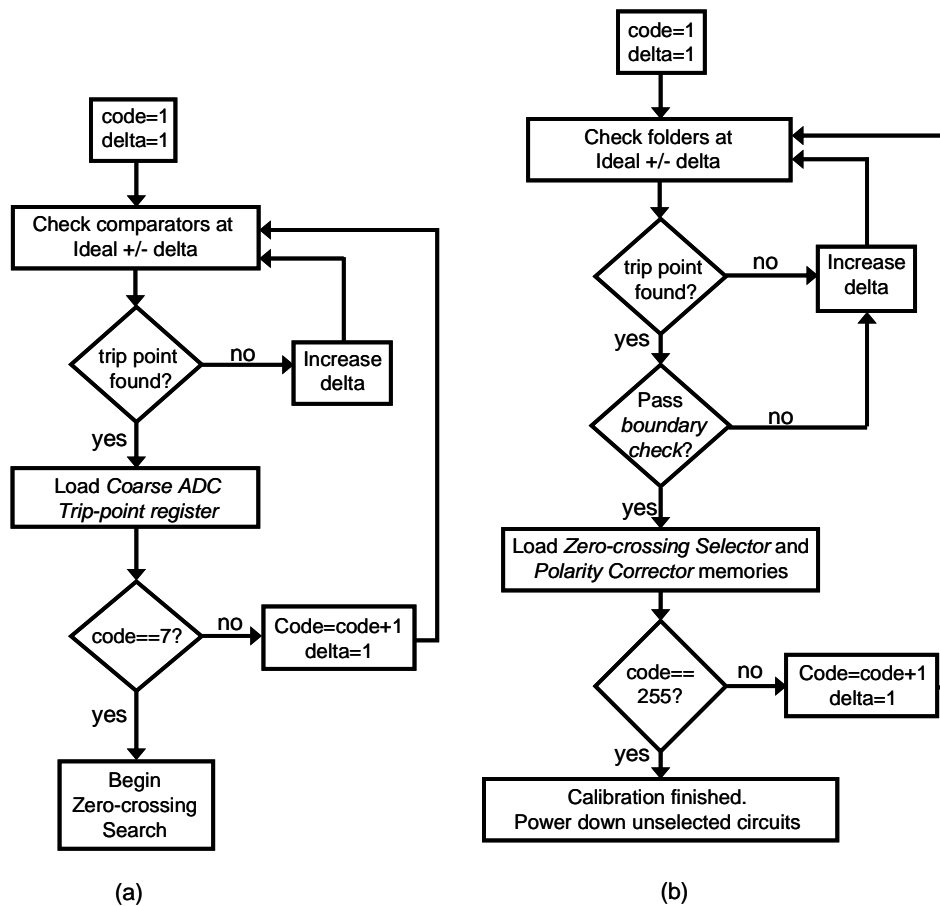


Fig. 3.11. (a) Calibration of coarse ADC. (b) Zero-crossing search

### 3.3.6 Folder and comparator implementation

To nominally place 4 zero-crossings at each code transition, a redundancy of 4 could be used for the folders with each folder connected to a single comparator. However as comparators consume less power and occupy less area than folders, a redundancy of 2 is used for the folders and each folder is in turn connected to 2 comparators, nominally places 4 zero-crossings at each code. Fig. 3.12 shows the overall folder and comparator configuration.

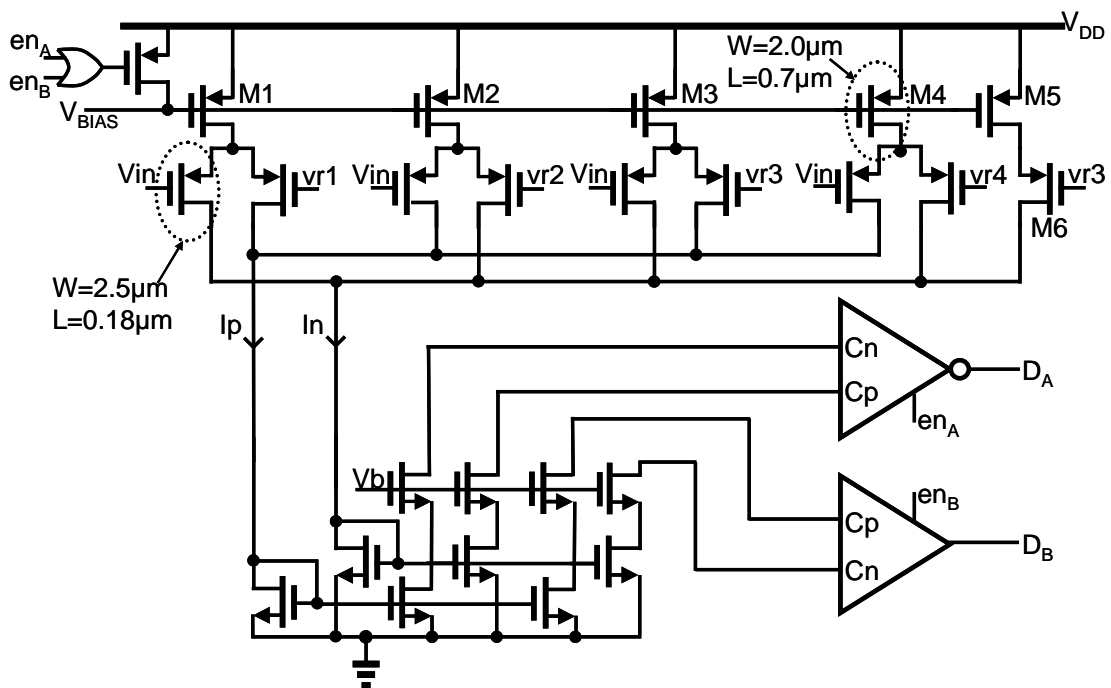


Fig. 3.12. Folder output is mirrored into two comparators.

The output currents of the folder are mirrored into the comparators through a cascode. With an input frequency  $F_{in}$  the nature of folding causes the output currents to be at a frequency of 4 times  $F_{in}$ . Higher mobility NMOS transistors are used in the current mirror and the folders are built using PMOS devices. The input transistors in the

differential pairs have a width of  $2.5\mu\text{m}$  and are minimum length. The tail current transistors have a width of  $2\mu\text{m}$  and a length of  $0.7\mu\text{m}$ . The area of a single comparator is less than  $300\mu\text{m}^2$ . The comparators are powered down if their enable signal,  $enA$  or  $enB$ , is low. If both are low then the folder is powered off.

$I_p$  from the folder is mirrored into the positive input,  $C_p$ , of one comparator and into the negative input,  $C_n$ , of the other comparator. This protects against the effects of a systematic offset in the comparators.

### **3.4 Distortion and Matching**

The low transconductance of the differential pairs and the low output impedance of the tail current sources cause distortion in the folder outputs. These issues are examined in this section, as are the matching requirements of the folders and comparators used in the folding section of the ADC, as well as the matching requirements of the comparators used in the coarse ADC. The DAC precision is also discussed.

#### **3.4.1 Tail current output impedance**

The output resistance of the tail current transistors (M1 to M5) used in the folders was thus far assumed to be infinite. However to reduce both area and the tail node capacitance, to allow faster folder operation, short, narrow, devices are used. These short devices have low output resistance and this distorts the folder zero-crossings. Fig. 3.13 shows the current flowing in the tail current sources of Fig. 3.12, and the resultant folder output as a function of input voltage.  $I_t$  nominally flows in each current source however as the tail node varies and the output resistance is low, different currents may flow in each current source.

For a small value of  $V_{in}$  all PMOS transistors connected to  $V_{in}$  are on. Current flows in the left side of each different pair. As  $V_{in}$  is increased the tail node voltages also increase, decreasing the voltage drop across the PMOS current sources and decreasing the output current. The decrease in output current with input voltage is shown as the diagonal line in Fig. 3.13.

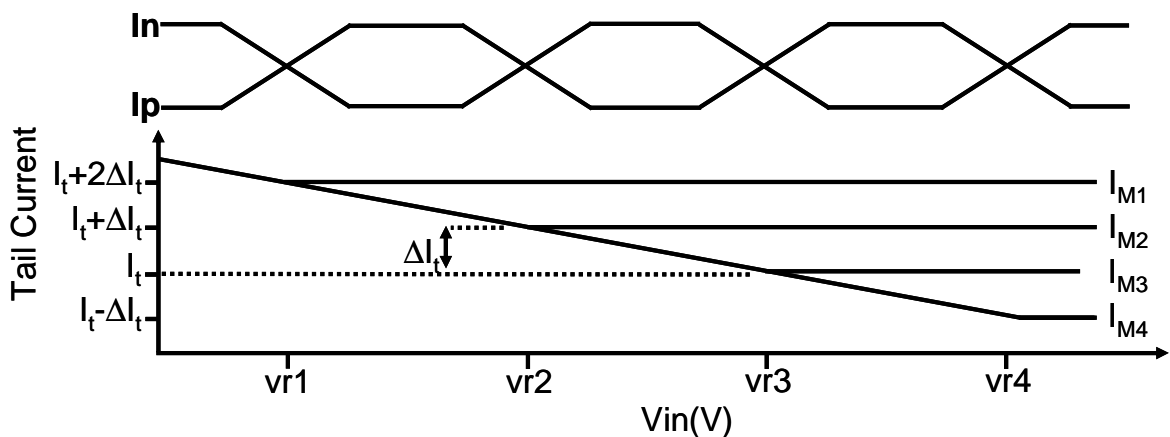


Fig. 3.13. Increase in folder input voltage decreases tail current source current.

For  $V_{in}$  greater than  $vr1$ , the tail node of the  $vr1$  differential pair, in Fig. 3.12, is dependent on  $vr1$  and the tail current no longer decreases with  $V_{in}$ . The  $vr2$  differential pair behavior is similar. For  $V_{in}$  less than  $vr2$  the tail node tracks  $V_{in}$  and for  $V_{in}$  above  $vr2$  the tail node voltage is dependent on  $vr2$  only. The tail nodes of the  $vr3$  and  $vr4$  differential pairs also track  $V_{in}$  up to their respective reference voltages. The 5th current source provides a constant current which we define this as  $I_t$ , and note that for large  $V_{in}$ , the drain of both  $M5$  and  $M3$  are at the same potential, dependent on  $vr3$ , and  $I_t$  also flows in  $M3$ . Assuming that the tail current transistor exhibits a constant output resistance,  $R_{out}$ , then at  $vr1$ ,  $I_{M1}$ , the current flowing in  $M1$ , is  $I_t + 2\Delta I_t$ , where  $\Delta I_t$  is  $(vr2 -$

$vr1)/R_{out}$ . The current sources  $IM_2$ ,  $IM_3$  and  $IM_4$ , for  $V_{in}$  equal to  $vr1$ , are supplying the same current. When  $V_{in}$  is equal to  $vr1$  the outputs  $I_p$  and  $I_n$  are:

$$I_{p(vr1)} = 0.5IM_1 + IM_2 + IM_4 = 2.5I_t + 5\Delta I_t \quad (3.1)$$

$$I_{n(vr1)} = 0.5IM_1 + IM_3 + IM_5 = 2.5I_t + 3\Delta I_t \quad (3.2)$$

Instead of  $I_p$  and  $I_n$  being equal at  $vr1$ ,  $I_p$  is  $2\Delta I_t$  greater than  $I_n$ . From the folder output plot in Fig. 3.13 we see that near  $vr1$  a slight decrease in  $V_{in}$ , decreases  $I_p$  and increases  $I_n$ . In effect, the zero-crossing moves from  $vr1$  to  $vr1 - 2\Delta I_t / G_m$  where  $G_m$  is the transconductance of the folder. We define the error in the zero-crossing voltage,  $2\Delta I_t / G_m$ , as  $V_e$ .

Performing a similar analysis for the zero-crossing nominally placed at  $vr2$  and noting that when  $V_{in}$  equals  $vr2$ ,  $IM_2$ ,  $IM_3$  and  $IM_4$  are  $I_t + \Delta I_t$  and  $IM_1$  is  $I_t + 2\Delta I_t$ ,  $I_p$  and  $I_n$  are:

$$I_{p(vr2)} = IM_1 + 0.5IM_2 + IM_4 = 2.5I_t + 3.5\Delta I_t \quad (3.3)$$

$$I_{n(vr2)} = 0.5IM_2 + IM_3 + IM_5 = 2.5I_t + 1.5\Delta I_t \quad (3.4)$$

Again  $I_p$  is  $2\Delta I_t$  greater than  $I_n$  at  $vr2$  however in this case a slight increase in  $V_{in}$  will decrease  $I_p$  and increase  $I_n$  causing the zero-crossing at  $vr2$  to shift to  $vr2 + V_e$ . Similar analysis for the zero-crossings at  $vr3$  and  $vr4$  show that they shift to  $vr3 - (V_e/2)$  and  $vr4 + (V_e/2)$  respectively.

In an ideal 8-bit, fold-by-4, folding ADC, consisting of 64 folders, the zero-crossings for the first 64 codes come from the  $vr1$  differential pair of the 64 folders. Codes 65 to 128 come from the  $vr2$  differential pairs of the folders. Codes 129 to 192 are from the  $vr3$  pair and 193 to 256 are from the  $vr4$  pair. The low output resistance of the tail current source moves codes 1 to 64 down by  $V_e$ , codes 65 to 128 up by  $V_e$ , codes 128

to 192 down by  $V_e/2$  and codes 193 to 256 up by  $V_e/2$ . This in turn causes a large DNL error at code 64 of  $2V_e$  and another smaller DNL error at code 192 of  $V_e$ .

In a traditional folding ADC,  $V_e$  is reduced by increasing the output impedance of the tail current transistor, however this involves using long devices which, for a given overdrive voltage, requires a corresponding increase in width, increasing both area and tail node capacitance. The transconductance of the differential pair input transistors could also be increased to reduce  $V_e$  but again this involves using large devices.

### 3.4.2 Folder transconductance

In a differential pair all the current is completely steered through one of the input transistors when the difference between the two applied voltages is a factor of  $\sqrt{2}$  greater than the overdrive voltage of the input transistors. In ideal folder operation, for any DC input, at most only one differential pair is in the *active region*. This requires the overdrive voltage on the input transistors to be at least  $\sqrt{2}$  less than the difference between two adjacent references  $\Delta V_r$ , (i.e.  $\Delta V_r = v_{r4} - v_{r3}$ ). Shrinking supply voltages associated with finer geometry processes leads to a reduced reference voltage range, forcing adjacent references closer together.

The transconductance is also reduced in the finer processes due to both mobility degradation with vertical field and velocity saturation. Fig. 3.14 shows folder output currents for three different values of differential pair transconductance,  $G_m$ . The larger  $G_m$  is, the greater the folder gain and the larger the difference between the two folder outputs.

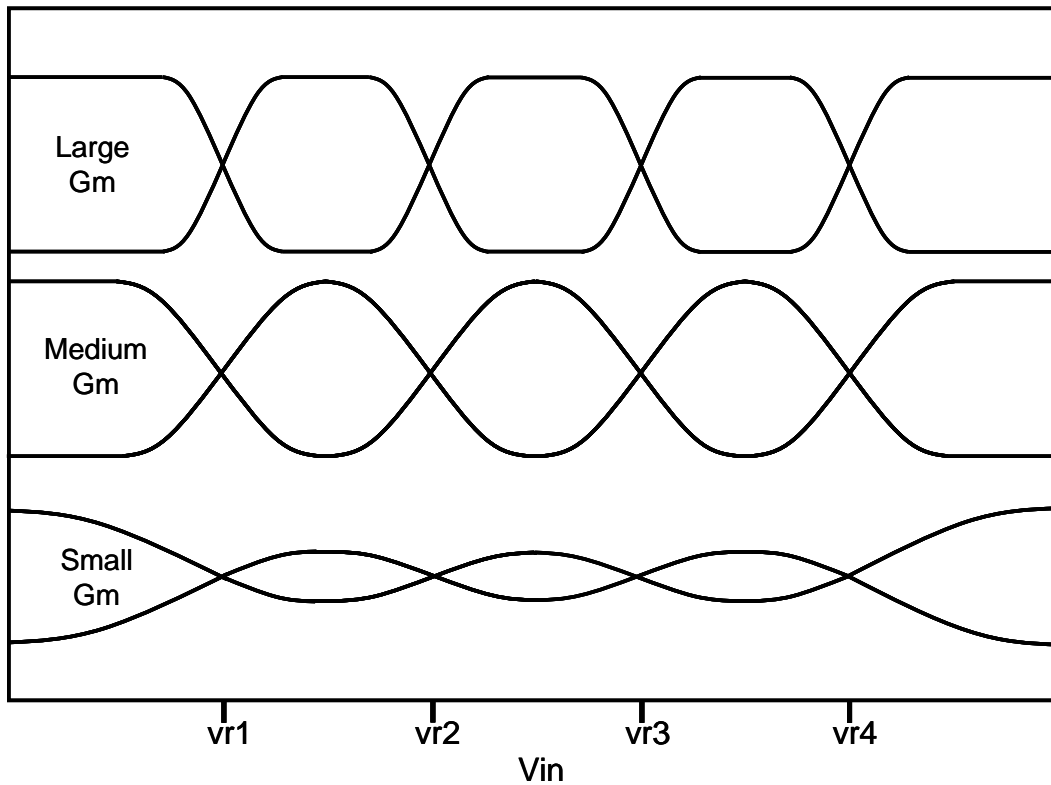


Fig. 3.14. Folder output currents for three different differential pair transconductance.

Fig. 3.15 shows the folder transconductance for the three different values of  $G_m$ . Differential pair transconductance is a function of input voltage. When  $V_{in}$  is close to the reference voltage, the transconductance is large. The greater the difference between the two inputs to the differential pair the less the transconductance. For a large  $G_m$  the folder transconductance has four distinct large peaks, one at each of the reference voltages. As the differential pair  $G_m$  is reduced the *active region* of a differential pair overlaps with its neighbor and the overall folder transconductance is reduced. For small  $G_m$  the folder transconductance is significantly reduced by this effect and at the reference voltages more than one differential pair are in the *active region*.

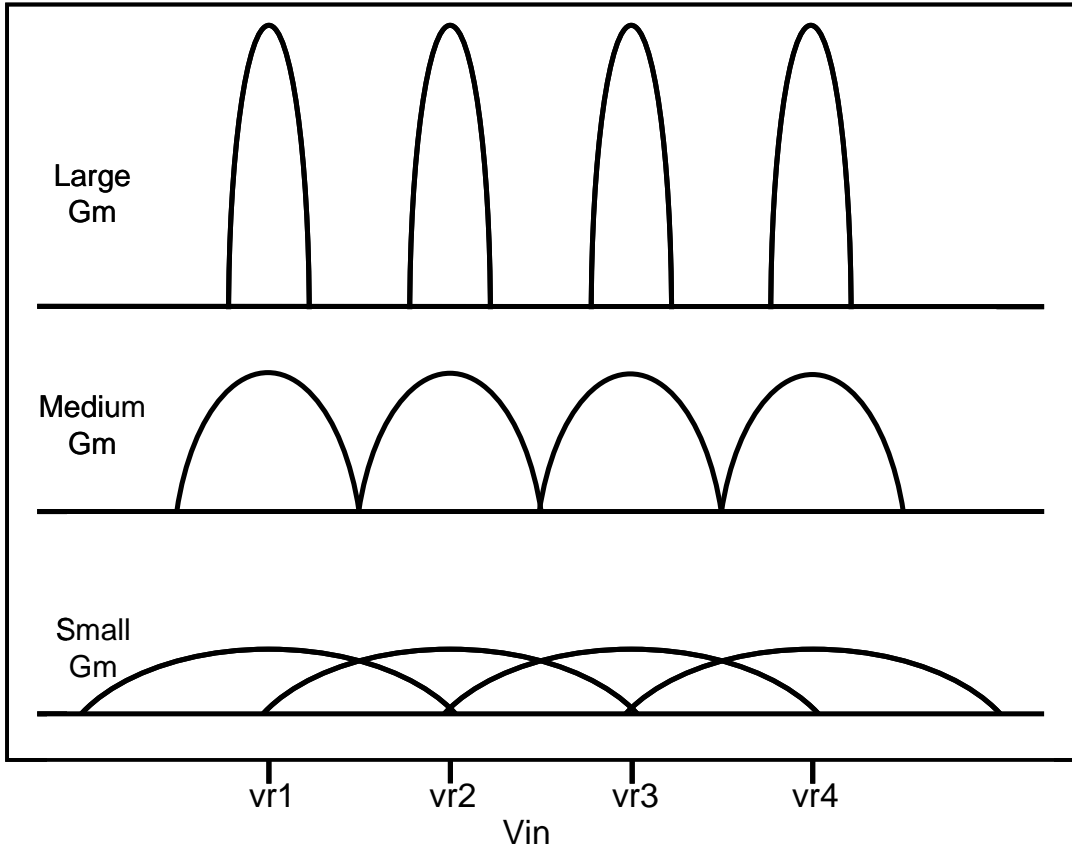


Fig. 3.15. Transconductance of each folder, versus input voltage. As differential pair transconductance reduces so does folder transconductance.

The overlapping of the *active regions* of adjacent differential pairs causes distortion in the folder outputs. When  $V_{in}$  is close to  $vr1$ , in the middle of the *active region* of the 1<sup>st</sup> differential pair, for the large  $G_m$  case the output currents are;

$$I_{p(vr1)} = 0.5I_{M1} + I_{M2} + I_{M4} = 2.5I_t \quad (3.5)$$

$$I_{n(vr1)} = 0.5I_{M1} + I_{M3} + I_{M5} = 2.5I_t \quad (3.6)$$

However for small  $G_m$  the 2<sup>nd</sup> differential pair is also in its *active region* at  $vr1$  and its tail current is not fully switched. If we define the error, the amount of current that has not been switched by the 2<sup>nd</sup> differential pair, as  $I_e$ , then at  $vr1$ ,  $I_p = 2.5I_t - I_e$  and  $I_n = 2.5I_t + I_e$ . The low  $G_m$  causes  $I_p$  to be less than  $I_n$  by  $2I_e$  at  $vr1$  and the zero-crossing



moves from  $vr1$  to  $vr1+(2Ie/Gm)$ . For  $Vin$  close to  $vr4$  and low  $Gm$ , the 3<sup>rd</sup> differential pair is in its *active region*,  $I_p=2.5I_t - Ie$  and  $I_n=2.5I_t + Ie$ , and the zero-crossing moves from  $vr4$  to  $vr4-(2Ie/Gm)$ .

When  $Vin$  is at  $vr2$  the positive output and negative outputs, for large  $Gm$ , are equal.

$$I_{p(vr2)}=IM_1+0.5IM_2+IM_4 \quad (3.7)$$

$$I_{n(vr2)}=0.5IM_2+IM_3+IM_5 \quad (3.8)$$

For low  $Gm$ , at  $vr2$ , the 1st and 3rd differential pairs are still in their *active regions* and Eqn. (3.7) and (3.8) become;

$$I_{p(vr2)}= IM_1-Ie_1+0.5IM_2+Ie_3+IM_4 \quad (3.9)$$

$$I_{n(vr2)}=Ie_1+0.5IM_2+IM_3-Ie_3+IM_5 \quad (3.10)$$

$Ie_1$  and  $Ie_3$  are the errors currents that have not switched from the 1<sup>st</sup> and 3<sup>rd</sup> differential pair respectively. Assuming that the transconductance of all 4 pairs is identical,  $Ie_1$  and  $Ie_3$  are the same and  $I_p$  and  $I_n$  are equal. With  $Vin$  at  $vr3$ ,  $Ie$  from the 2<sup>nd</sup> and 4<sup>th</sup> differential pairs cancel. Low folder transconductance causes errors in the zero-crossings nominally placed at  $vr1$  and  $vr4$  but the zero-crossings at  $vr2$  and  $vr3$  are unaffected.

A conventional 8-bit folding ADC using 64 of the distorted folders, the same folders as those used in the final ADC design, which have both low transconductance and low tail current impedance, was simulated with Spectre, Fig. 3.16. The simulation results show the DNL at codes 64 and 192 as 3.8LSB and 1LSB respectively. This simulation shows the effects of folder distortion due to both low transconductance and low tail current resistance and does not include the effects of transistor mismatch.

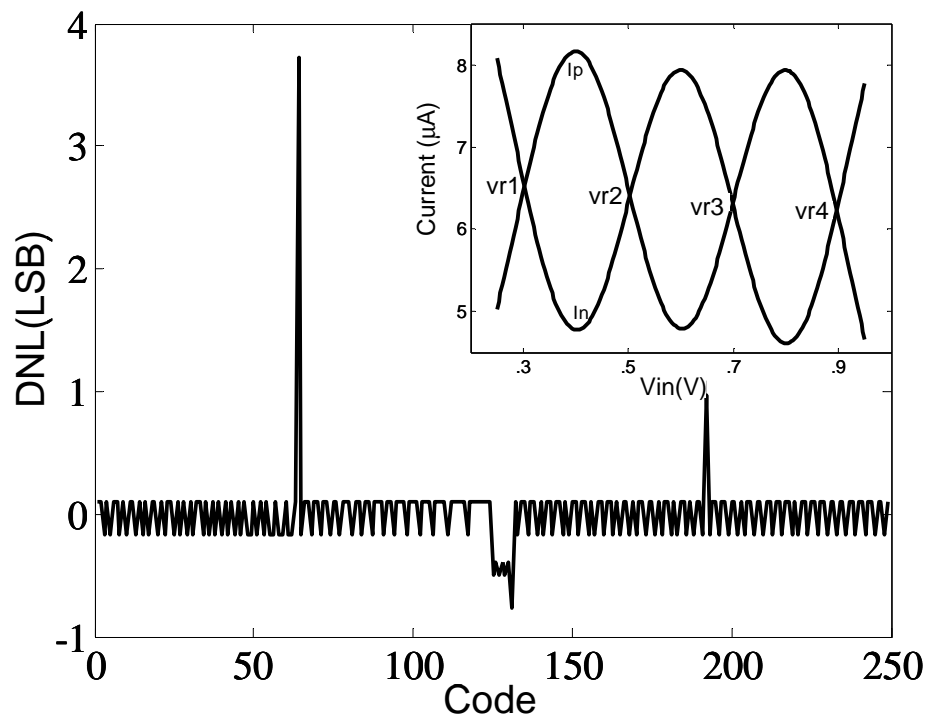


Fig. 3.16. DNL plots derived from Spectre simulation of an 8-bit ADC using these folders. The distorted output of a single folder as the input voltage is swept is in the inset.

In this design the use of small transistors in the folders and comparators with large offsets decouples the zero-crossing locations from the effect of low tail current impedance and low transconductance. The calibration engine can then pick zero-crossings and the calibrated ADC shows no DNL errors caused by either the low impedance or low distortion issues.

### 3.4.3 Folder and comparator matching

Folder offset is dominated by the size of the differential pair input transistors used in the folder. This ADC uses small devices, of width  $2.5\mu\text{m}$  and length  $0.18\mu\text{m}$ , and the input referred offset of the folder due to differential pair mismatch alone is over 3 LSB.

Redundant, low accuracy comparators together with low accuracy folders are used to ensure high yield. If two identical comparators,  $C1$  and  $C2$ , are connected to a folder, but have no offset, they provide the same outputs, Fig. 3.17(a). As the zero-crossings provided from  $C1$  and  $C2$  are identical, in effect only two zero-crossings are nominally placed at each code. Increasing the comparator mismatch causes the zero-crossings of  $C1$  and  $C2$  to become unique, assigning four zero-crossings to each code. With four zero-crossing per code an acceptable yield is achieved. The comparator used is shown in Fig. 3.17(b). The input referred offset of the comparator is dominated by the matching of the PMOS latch transistors, MP1 and MP2, and to a lesser degree the transistors in the NMOS latch, MN1 and MN2. By using small transistors in the latches, the overall mismatch is increased.

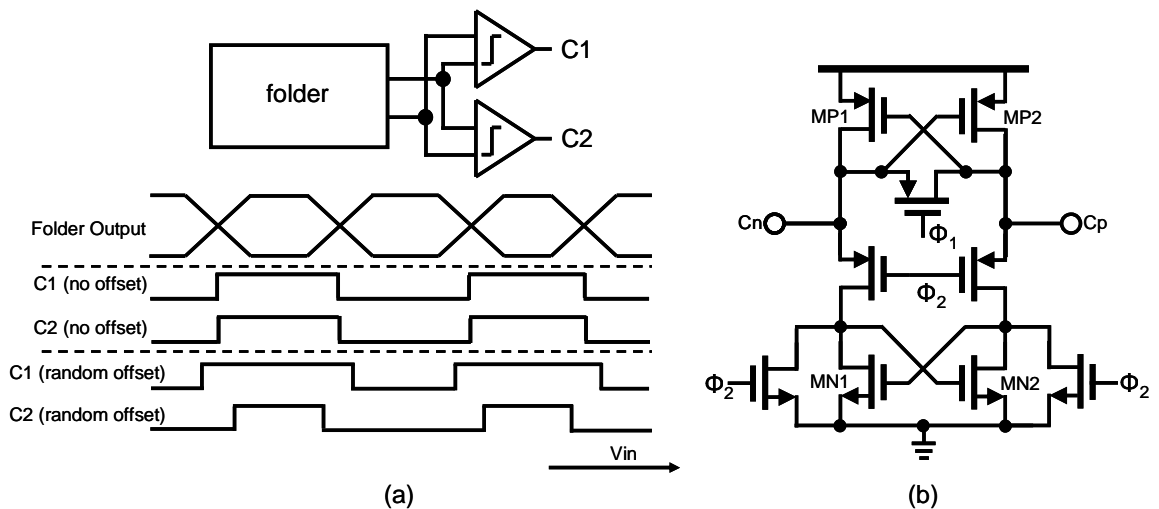


Fig. 3.17. (a) Random comparator offsets generate 8 unique zero-crossings. (b)

Comparator latch, matching is dominated by the size of MP1 and MP2.

The overall input referred standard deviation offset  $\delta(e_{IN})$  of a folder connected to a comparator is;

$$\delta(e_{IN}) = \sqrt{\delta(f_M)^2 + \frac{\delta(c_M)^2}{A_f}} \quad (3.11)$$

$\delta(f_M)$  is the standard deviation folder mismatch and  $\delta(c_M)$  is the standard deviation comparator mismatch. The voltage gain of the folder,  $A_f$  is the small signal voltage gain from the folder input to the comparator input when the input voltage is close to a reference voltage. Making the folder gain large would reduce the second term in Eqn. (3.11), having the same effect as making the comparator mismatch small. The combination of low folder gain together with small devices in the comparator is required to ensure randomly distributed zero-crossings.

Fig. 3.18 shows Monte Carlo simulation results of the digitally calibrated 8-bit ADC, as both  $\delta(f_M)$  and  $\delta(c_M)$  are varied. The gain of the folder is 4 and the results are shown for  $\delta(f_M)$  of 1,4,7 and 10 LSB. Here a ‘good’ ADC is defined as an ADC with more than 7.5 effective bits of accuracy. For small  $\delta(c_M)$  the yield is very low as the comparator pair connected to a folder does not provide unique zero-crossings. As  $\delta(c_M)$  is increased the yield also increases. For  $\delta(f_M)$  of 10 LSB a yield of over 93% can be attained with a  $\delta(c_M)$  of 20LSB.

From a mismatch perspective, minimum size transistors can be used in the comparator latch. In practice slightly larger devices are used for metastability and speed requirements. MP1 and MP2, the transistors in the PMOS latch in Fig. 3.17(b), and MN1 and MN2, the NMOS latch transistors, are minimum length with respective widths of 0.8 $\mu\text{m}$  and 0.5 $\mu\text{m}$ .

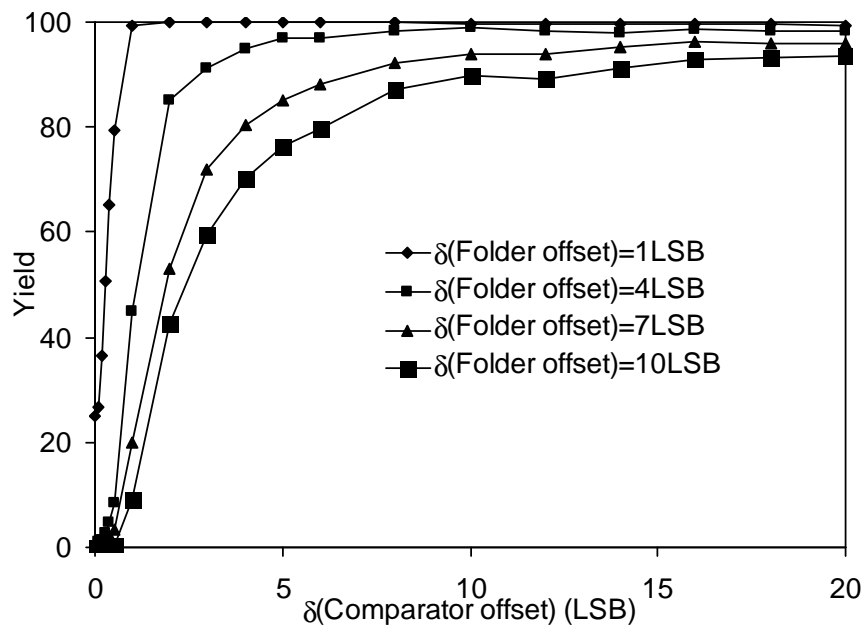


Fig. 3.18. ADC yield as comparator offset is varied for various folder offsets.

### 3.4.4 Coarse ADC matching

The coarse ADC is a 3-bit flash ADC with a redundancy of 3. From a bank of 21 comparators, 7 after calibration are enabled and the remaining 14 are powered down. The coarse ADC is used to divide the ADC conversion range into 8 *octants* so comparators are nominally at the MSB-2 transition voltages. Much as a traditional folding ADC uses synchronization [20] to correct offsets between the coarse ADC and fine ADC this ADC uses digital error correction. Overlap and digital error correction are used if a selected zero-crossing is within 6 LSB of a coarse ADC trip point. The error correction works by ensuring that the zero-crossing is selected in the two neighboring *octants*. Thus if the coarse ADC and folders drift apart during conversion, there are no errors in the output code provided the coarse ADC stays within 6 LSB of its original calibrated value. For the digital error correction algorithm to work two different digital error correction regions

cannot overlap. To achieve this, the only restriction on the coarse ADC is that adjacent coarse ADC comparators, after calibration, are at least 12LSB apart. Fig. 3.19 shows Monte Carlo simulation of the coarse ADC with redundancy as comparator offset is varied. Here a successful coarse ADC is one where the minimum code width is greater than 12 LSB.

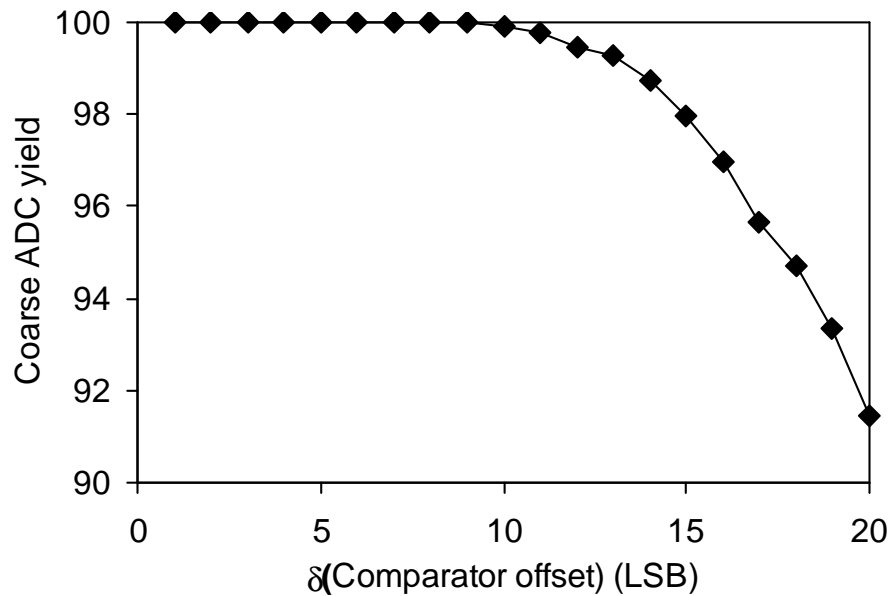


Fig. 3.19. Coarse ADC yield as comparator offset is increased.

Large comparator offsets can be tolerated in this manner. With a standard deviation comparator offset of 10 LSB a yield of over 99.8% can be achieved.

The comparator consists of a single differential pair, the outputs of which are mirrored through a cascode to a regenerative latch. The differential pair used is the same as that used in the folders and the comparator latch is the same as the latch used in the comparators connected to the folders. This helps ensure that the coarse ADC and folders remain aligned over all frequencies.

### 3.4.5 DAC precision

Monte Carlo simulations were used to determine the required DAC precision. Five random combinations of folder offset standard deviation,  $\delta(f_M)$ , and comparator offset standard deviation,  $\delta(c_M)$ , were used. The folder gain in each case is 4. The mean effective number of bits (ENOB) for 100 of each ADC configuration, after calibration, versus DAC precision is plotted in Fig. 3.20.

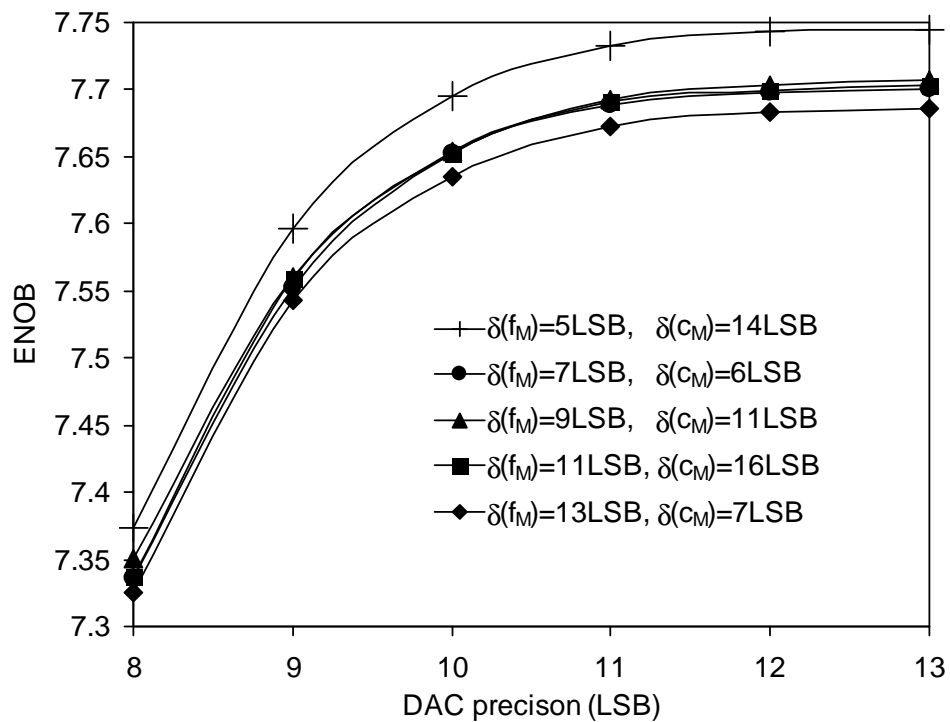


Fig. 3.20. Monte Carlo results of mean ENOB versus DAC precision.

Irrespective of the folder and comparator mismatch in the ADC, there is only a slight improvement in the mean ENOB, for each configuration, for increases in search DAC precision above 10 bits. However, for every bit extra bit of DAC resolution, the

number of switches doubles and the DAC decoder logic becomes more complex so a 10-bit DAC is used.

### 3.5 Prototype Measurements

The prototype 8-bit ADC, fabricated in  $0.18\mu\text{m}$  digital CMOS, Fig. 3.21, has an active area of  $1.22\text{mm}^2$ . The total analog active area (i.e. folders, comparators and sample-and-hold), is only  $0.2\text{mm}^2$ . The prototype is entirely self-contained, incorporating all calibration and decoding logic. All digital circuitry is synthesized from verilog, and automatically placed and routed.

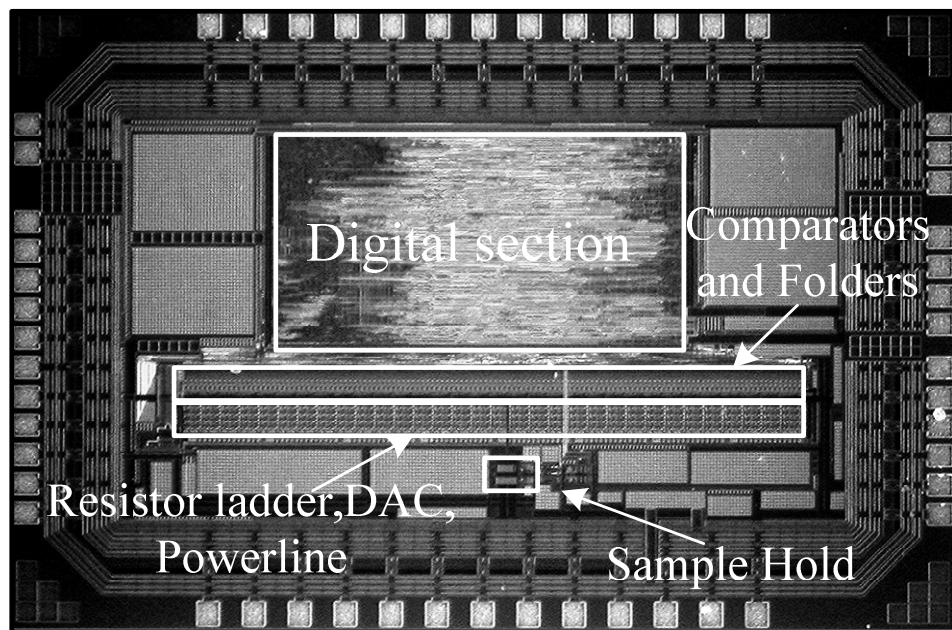


Fig. 3.21. Folding ADC die micrograph.

#### 3.5.1 Printed circuit board

The chip was packed in QFN48 package. A custom printed circuit board (PCB) was designed for testing the chip. It is a 4 layer board with copper layers, Fig. 3.22.



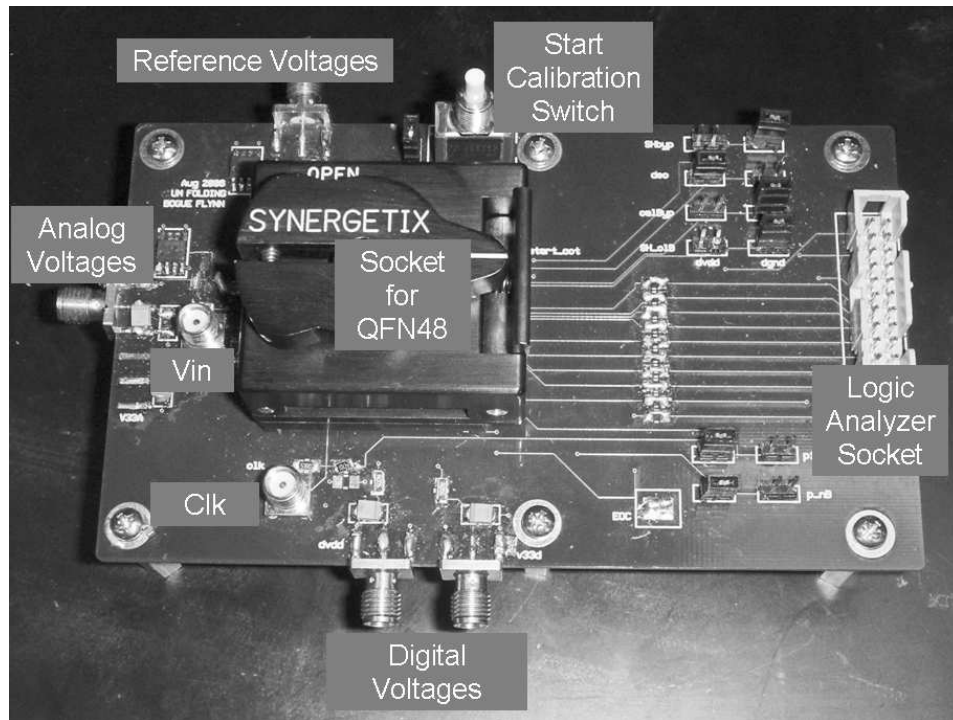


Fig. 3.22. PCB test board

The middle two layers are used as power and ground planes. The power and ground planes are both separated in two sections, analog and digital. Decoupling capacitors are used on the board to reduce noise caused by bonding wire inductance and clock kickback.

### 3.5.2 Test Setup

The test setup is shown in Fig. 3.23. Separate power supplies were used for the analog and digital power supply voltages. The clock source and input source were locked to a 10MHz reference signal to allow coherent testing. The 8-bit ADC output was fed to a Logic Analyzer. Matlab is used to control all the test equipment and is also used to read the ADC output from the Logic Analyzer. Matlab programs measure ADC performance,

while adjusting setup for both calibration and conversion, allowing the optimal ADC performance to be determined.

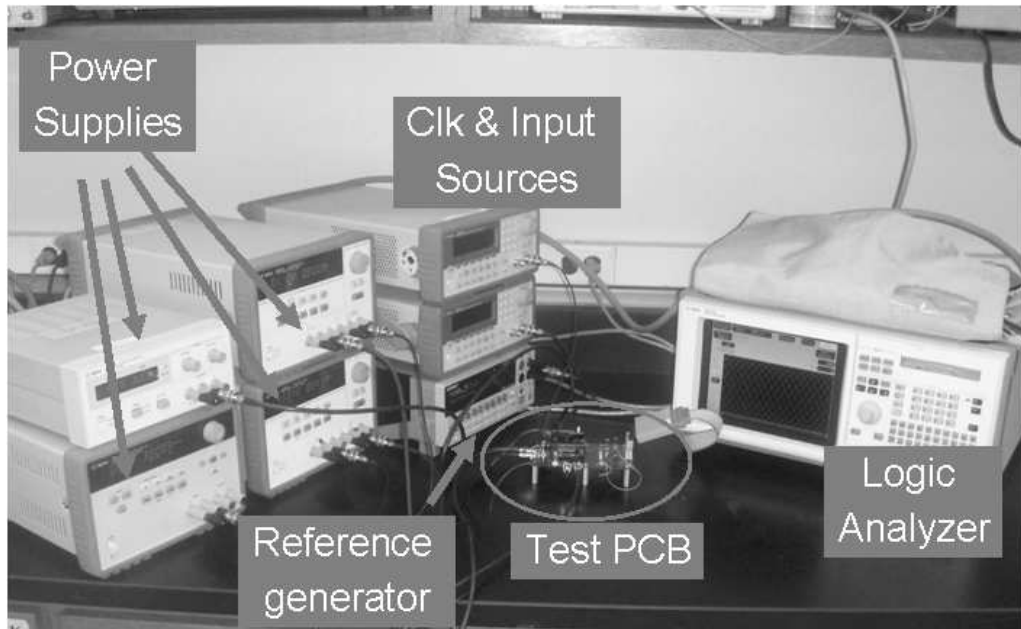


Fig. 3.23. ADC test setup.

### 3.5.3 Prototype results

Fig. 3.24 shows the DNL before and after calibration measured at a sampling rate,  $F_s$ , of 500MS/s. Before calibration the maximum DNL is 6.7LSB and after calibration maximum DNL is reduced to 0.8LSB. The folder distortion described earlier causes large DNL at codes 64 and 192 however the uncalibrated DNL shows missing codes around code 64, 192 as well as near code 164. This is due to distortion in the folders and also due to the lack of synchronization between the coarse ADC and the folders [20]. The DNL of 6.7LSB at code 17 is due to threshold voltage mismatch. The calibrated INL is between -0.57 and 0.73LSB.

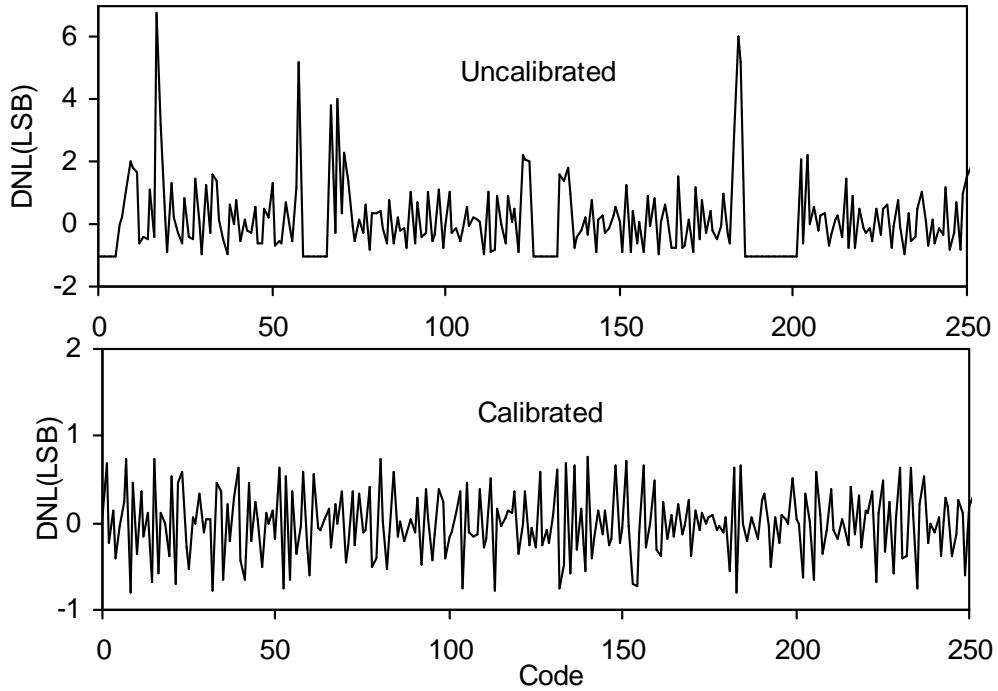


Fig. 3.24. DNL before calibration is 6.8LSB and after calibration DNL is 0.8LSB.

The SFDR versus sampling rate, measured with a 1MHz input ( $F_{IN}$ ), is shown in Fig. 3.25. At  $F_S=39\text{MS/s}$ , SFDR is 63dB and remains above 55dB up to  $F_S=550\text{MS/s}$ . Thermal noise in the folders limits the SNDR to 44dB. Simulations show that thermal noise can be reduced significantly by increasing the transconductance of the input transistors in the folder differential pairs. This requires only a small increase in total analog power. The SFDR, measured at  $F_S=500\text{MS/s}$ , is above 52dB for  $F_{IN}$  up to 120MHz, Fig. 3.26. Using the method described in [55], no metastability error was detected in  $10^{12}$  samples indicating a BER less than  $10^{-12}$ . At 500MS/s analog circuitry (clocking, S/H, folders, comparators) consumes 132mW while the digital backend consumes 216mW. The thermal noise of the folders in this design limits the ENOB, measured at an ERBW of 120MHz, to 6.2 bits.

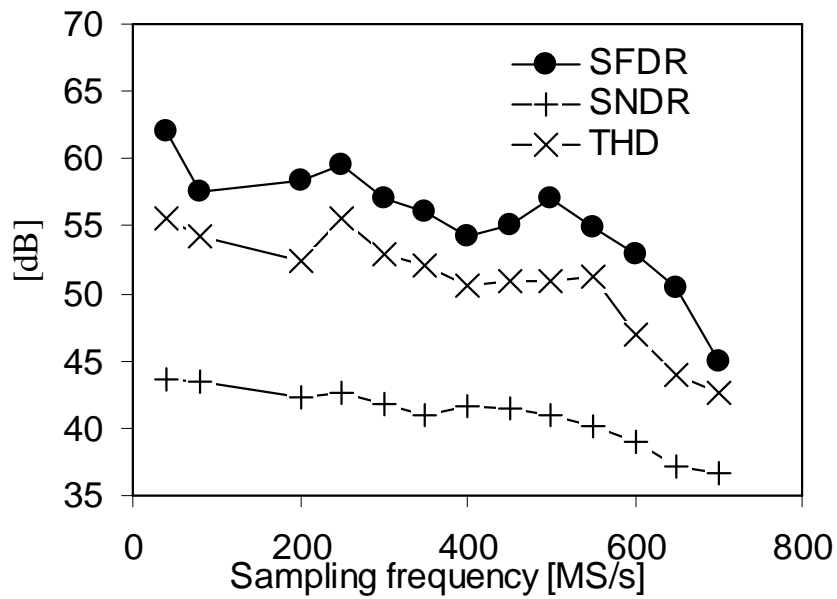


Fig. 3.25. SFDR, SNDR and THD versus sampling frequency with  $F_{IN}=1$  MHz.

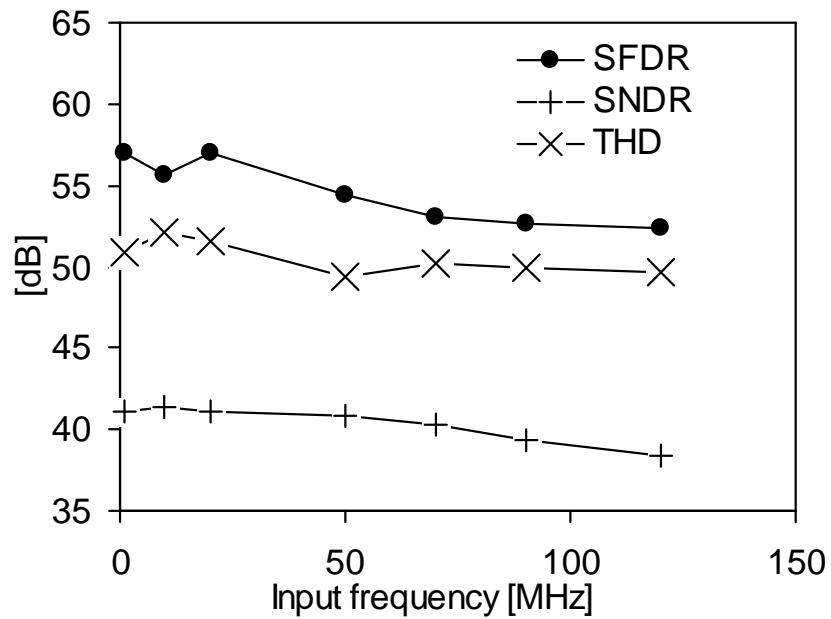


Fig. 3.26. SFDR, SNDR and THD versus input frequency with  $F_S=500$  MS/s.

The overall ADC performance is summarized in Table 3.1.

Analog Power (@500MS/s)	132mW
Digital Power (@500MS/s)	216mW
Area	1.16mm <sup>2</sup> (0.2mm <sup>2</sup> analog)
DNL(before and after calibration)	6.8LSB,0.81LSB
SFDR (F <sub>IN</sub> =120MHz,F <sub>S</sub> =500MS/s)	52dB
SFDR (F <sub>IN</sub> =1MHz,F <sub>S</sub> =39MS/s)	63dB

Table 3.1. Summary of 8-bit ADC performance.

## Chapter 4

### 9-bit folding ADC with novel folders

#### **4.1 Introduction**

This chapter describes the design of a 9-bit folding ADC implemented in ST Microelectronics 90nm CMOS. A unique folder architecture reduces thermal noise for a given power budget. Thermal noise in folders is discussed in section 2, digital calibration and implementation in sections 3 and 4. Matching and yield is examined in section 5 and the prototype is presented in section 6.

#### **4.2. Thermal noise in parallel folders**

For moderate to high resolution ADCs thermal noise in the analog circuitry is a significant source of error [56]. Thermal noise is caused by the random motion of electrons in a conductor. Transistors exhibit thermal noise, primarily due to the noise generated in the channel. In a folding ADC, the transistors in the folders and the comparators can suffer from thermal noise. The folders provide gain and the input referred thermal noise due to the transistors in the comparators is reduced by this folder gain. If the folder gain is large enough the comparator thermal noise can be ignored. In this section the thermal noise from a folder, folding by 4, is examined, Fig. 4.1.

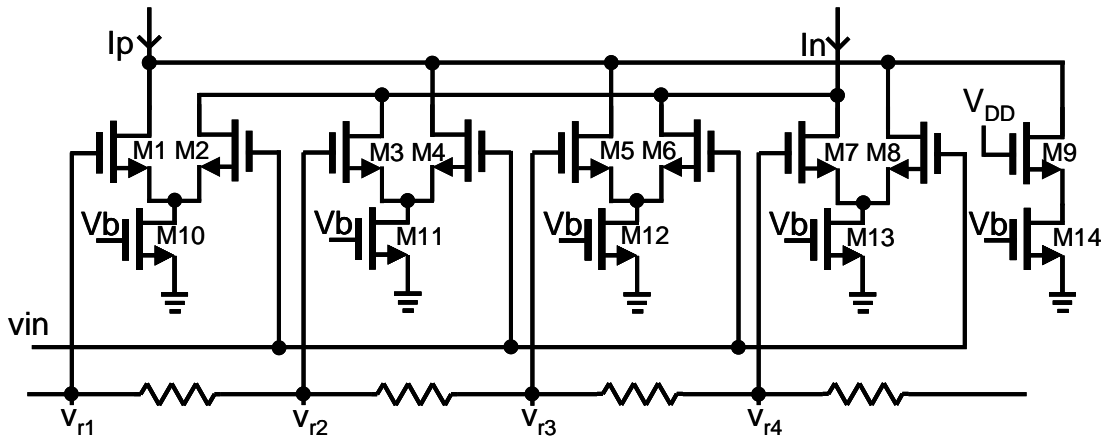


Fig. 4.1. Folding by 4 folder.

If the input voltage,  $vin$ , is close to reference voltage  $vr2$  and the transconductance of the differential pairs is large enough then the circuit can be simplified to that shown in Fig. 4.2.,  $vin$  is a lot greater than  $vr1$  so M1 is off and the gate voltage of M2 is connected to an AC ground.  $vr3$  is greater than  $vin$  so M6 is off and M5 is connected to an AC ground, and similarly M8 is also off and M7 is connected to an AC ground. The transistors M2, M5, M7 and M9, in this particular case, are cascodes and so do not contribute noise. M10-M14 are connected to  $Vb$  which is an AC ground.

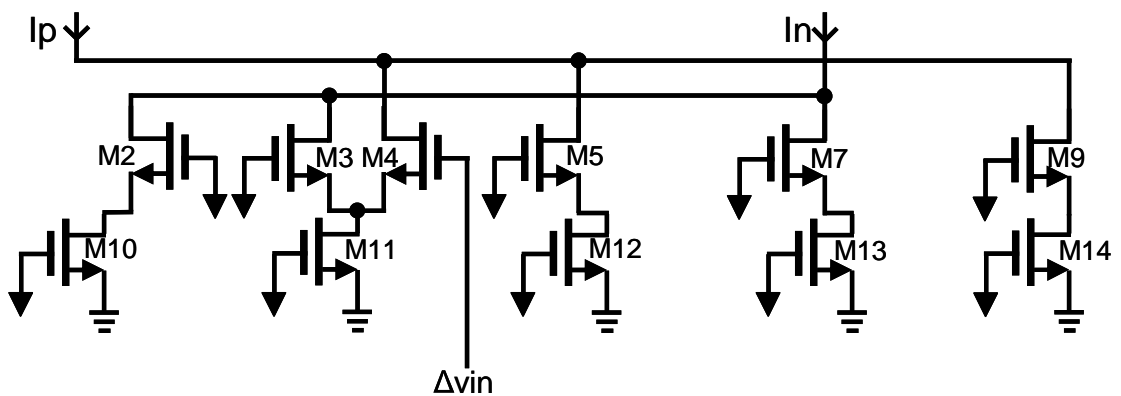


Fig. 4.2. Equivalent circuit for the folder with  $V_{in}$  close to  $vr2$ .

The input referred thermal noise power of this circuit, where  $\Delta f$  is the bandwidth of the folder, the transconductance of the current sources are  $gm_t$  and the

transconductance of M3 and M4 is  $gm_{3,4}$ , is given in Eqn. (4.1). The first term is the input referred noise from transistors M3 and M4. The second term is the noise from M10, M12, M13 and M14. Any noise from M11 is differential and does not affect the folder output.

$$v_{nin}^2 = \left[ 2(4kT \frac{2}{3gm_{3,4}}) + 4(4kT \frac{2gm_t}{3gm_{3,4}}) \right] \Delta f \quad (4.1)$$

In general for any value of  $V_{in}$ , the input referred thermal noise of the folder is inversely proportional to the transconductance of the input transistors of the differential pair operating in its *active region* and is proportional to the transconductance of the tail current sources in the differential pairs acting in their *switched regions*.

The transconductance of the tail current sources can be made small, reducing the second term in Eqn. (4.1). The primary source of noise is the low transconductance of the input transistors,  $gm_{3,4}$ . The transconductance of the M3 and M4 can be approximated as  $2I_t/V_{DSat}$  where  $I_t$  is the current flowing in the transistor and  $V_{DSat}$  is the overdrive voltage. In 90nm technology and below,  $V_{DSat}$  typically used is about 100mV. For high speed operation the folder bandwidth should be in the order of 1GHz. Using these folders to create a 9-bit ADC with an input range of 600mV, giving an LSB size of just over 1mV, would mean that to keep the input referred noise less than 0.3LSB, the current flowing through each of the differential pairs has to be greater than 50 $\mu$ A

#### 4.2.1 Reducing thermal noise

Traditional folding circuits perform their folding using parallel differential pairs. For each degree of folding a differential pair is required. An extra current source is also



required to provide a DC offset so that the output currents overlap. As an example, a folder, folding by 8, would require 9 current sources.

To reduce thermal noise in a folder there are two options:

- Increase  $gm_{3,4}$ : This would require either more current, increasing power consumption or bigger devices which would work slower and require more area.
- Limit the folder bandwidth,  $\Delta f$ , by adding extra capacitance, this would slow down the overall ADC performance.

For low power folding ADCs in nanometer technologies a new folder architecture is required.

#### 4.2.2 Stacked Folders

A more efficient folder is shown in Fig. 4.3(a). Here the differential pairs are stacked so that one current source is reused by 3 differential pairs.

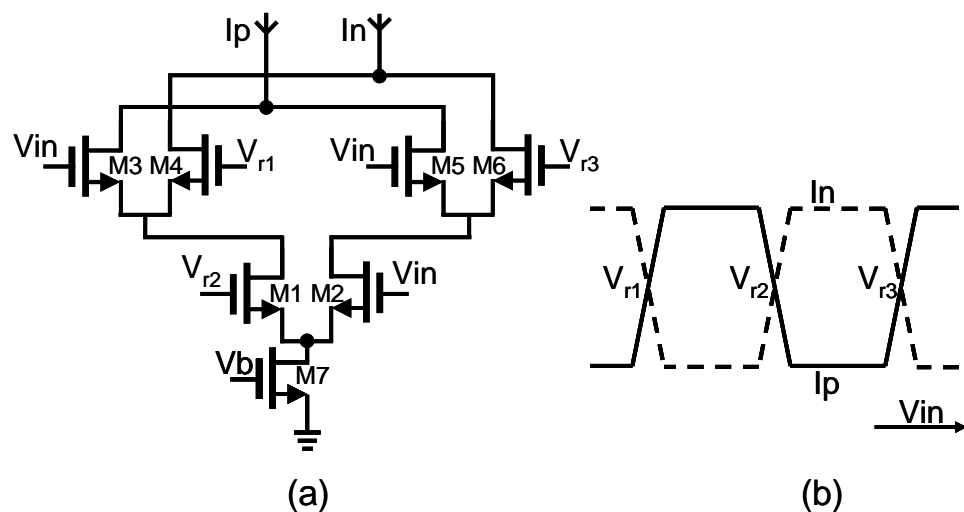


Fig. 4.3. (a) Stacked Folder reuses single tail current. (b) Output currents as input is swept.

If  $V_{in}$  is less than  $V_{r2}$  current is steered to the differential pair, M3 and M4, and if  $V_{in}$  is less than  $V_{r1}$  then all the current flows through  $I_n$ . The same current flows through

all active differential pairs. The current outputs as  $V_{in}$  is increased from low to high are shown in Fig 4.3(b).

If the differential pair transconductance is large then for any value of  $V_{in}$  only one of the differential pairs is in its *active region*, the other two are in their *switched regions*. The thermal noise of the folder is due to the noise performance of the differential pair in its *active region*. If  $V_{in}$  is close to  $V_{r1}$  then the input referred thermal noise is inversely proportional to  $g_{m_{3,4}}$ , close to  $V_{r2}$  the noise is inversely proportional to  $g_{m_{1,2}}$  and close to  $V_{r3}$  its inversely proportional to  $g_{m_{5,6}}$ . The input referred thermal noise of the folder can be reduced by increasing the transconductance of transistors M1-M6. The transconductance of each transistor is proportional to the current flowing through it. Therefore the input referred thermal noise of the folder can be reduced by simply increasing a single current source, the tail current flowing through M1. This new structure means that folding-by-3 requires only one large current source, a traditional, parallel folder folding-by-3 requires 3 large current sources.

In this design the differential pairs, M3, M4 and M5, M6 have a length of  $0.1\mu\text{m}$  and a width of  $6\mu\text{m}$ . To increase the output impedance of M1 and M2 they have a length of  $0.5\mu\text{m}$  and a width of  $12\mu\text{m}$ . This folder has significant distortion but the digital calibration technique overcomes the distortion.

### **4.2.3 Stacked folding by nine**

In an N-bit folding ADC, the number of comparators required is  $2^N/f$ , where  $f$  is the degree of folding performed in the folders. The larger  $f$  is, the smaller the number of comparators required. Similar to the cascaded folder described in section 2.2 of chapter 2, here the outputs of 3 of the stacked folders are folded together in a second stage to

increase the overall folding factor to 9; Fig. 4.4. The second stage is a simple parallel folding by 3 folder.

The addition of the cross-coupled latches, the  $M_g$  transistors, to the individual fold-by-3 stacked folders increases the stacked folder gain. The diode connected transistors forms a load resistance. The  $M_g$  transistors form negative resistance that is in parallel with the load resistance. By carefully selecting the value of the negative resistance the gain can be substantially increased.

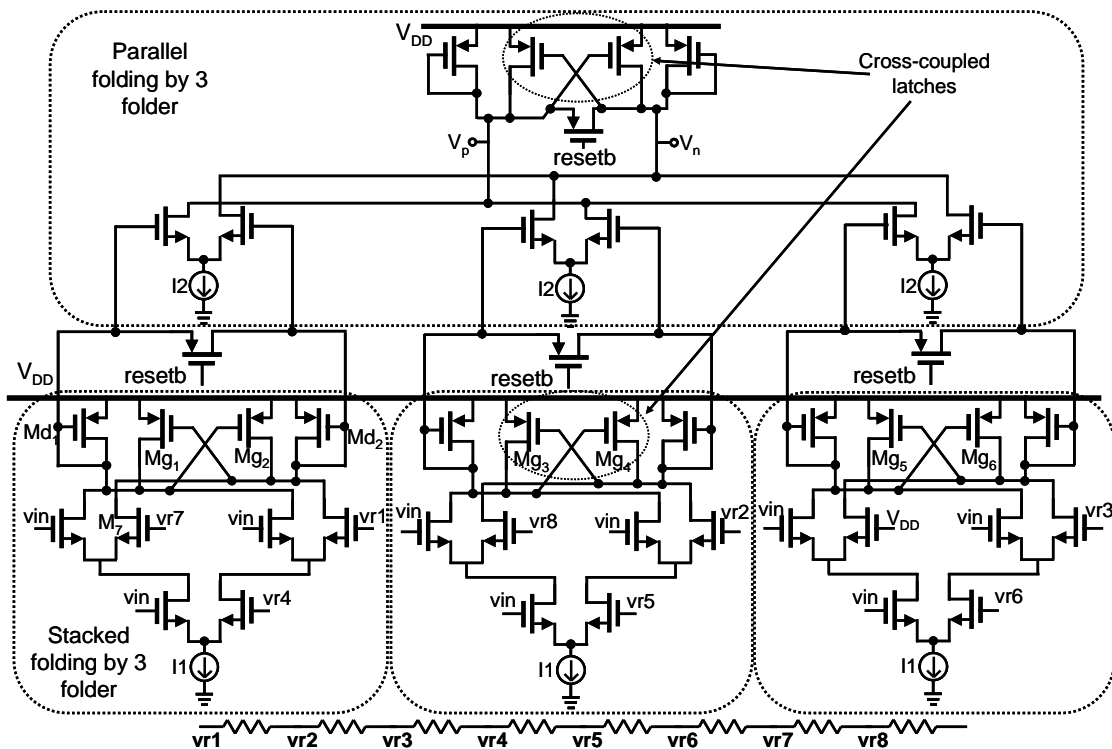


Fig. 4.4. Stacked folder, folding by 9 requires 3 large current sources and 3 small current sources.

For  $v_{in}$  approximately equal to any of the reference voltages the magnitude of the gain through the stacked folder increases from  $g_{m_{in}}/g_{m_d}$  to  $g_{m_{in}}/(g_{m_d}-g_{m_g})$ , where  $g_{m_{in}}$ ,

$gm_d$  and  $gm_g$  are the transconductance of the input transistors, diode connected transistors and cross-coupled transistors respectively in the stacked folder.

The folders are reset when *resetb* is low. Resetting the folders allows for faster folder operation and overcomes hysteresis. With *resetb* low the outputs are forced equal, opening the reset switches. When *resetb* goes high, the folder outputs quickly switch in the correct direction [24]. The transistors used in the parallel folder differential pair have a width of  $5\mu\text{m}$  and a length of  $0.1\mu\text{m}$ . A cross-coupled latch is also added to the parallel folder to increase the gain of that stage.

The input-referred thermal noise due to the parallel stage is reduced by the gain of the stacked folder, so the current,  $I_2$  in the parallel folder does not have to be as large as  $I_1$  used in the stacked folder. The current  $I_1$  is  $45\mu\text{A}$  whereas the current  $I_2$  is only  $15\mu\text{A}$ .  $V_{DD}$  is used in place of  $V_{r9}$ , providing more orthodox folding by 8. This new stacked folder requires only 3 large current sources and 3 small current sources meaning the total current is only  $180\mu\text{A}$ . A parallel folder, with the same thermal noise, folding-by-8, would require 9 large current sources, i.e. 9 current sources each supplying about  $45\mu\text{A}$ , giving a total of  $405\mu\text{A}$ .

### 4.3 Digital calibration

Digital calibration is also used in this ADC. The scheme is similar to the calibration described in Chapter 3. However, the ADC described in Chapter 3, is an 8-bit folding ADC, folding by 4, this design is a 9-bit folding ADC, folding by 8. To reduce both area and power consumption and to accommodate the higher degree of folding some changes were made to the digital calibration routine. This includes an increase in the

coarse ADC resolution, and changes to how the redundancy is implemented and how the polarity of the selected zero-crossings is handled.

### **4.3.1 Coarse ADC**

A coarse ADC divides the ADC conversion range into sections. This ADC uses folding by 8 folders. This high degree of folding increases the required bandwidth of the comparator above that required in the 8 bit, 0.18 $\mu$ m version, but this higher bandwidth is achievable in the finer geometry. To reduce the probability of a folder output having 2 zero crossings in a given section a 4-bit flash ADC is used to divide the ADC conversion range into 16 sections. The coarse ADC is calibrated from a bank of 45 available comparators (i.e. a redundancy of 3 used), and 15 comparators are selected, their trip point locations stored in SRAM and the unselected comparators are powered down.

### **4.3.2 Redundancy and reassignment**

A 9-bit folding ADC, folding by 8, would require 64 folders each connected to a single comparator. This design uses redundant comparators only, and there are no redundant folders.

To determine the degree of redundancy required, a Matlab simulation was run. Redundancy was varied for different offsets, and the yield recorded, Fig. 4.5. 1000 ADCs were generated in Matlab for each configuration of redundancy and offset. With a ‘good’ ADC was defined as one with an ENOB of over 8.5 bits, a redundancy of 4 gives a yield of over 98% even with a standard deviation of comparator offset of more than 10LSB. Increasing the redundancy beyond 4 shows only a marginal improvement in yield.

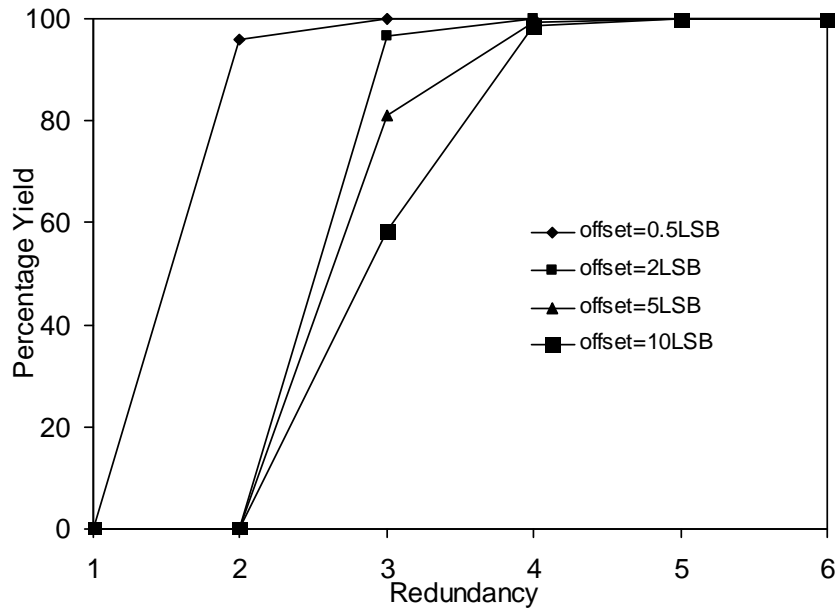


Fig.4.5. Yield versus redundancy for the 9-bit ADC.

To generate the 4 redundant zero-crossings the number of comparators used in the design was increased by a factor of 4. No redundant folders were used in the design to reduce power consumption. The use of 4 comparators connected to a single folder nominally places 4 zero-crossings at each ADC code. The overall mismatch of the block, the folder and the 4 comparators, is again given by Eqn. 3.11. To reduce the distortion of the folder, larger devices are used in the folder design of this ADC than were used in the 8-bit version, decreasing the folder mismatch,  $\delta(f_M)$ . The comparators used are very small and their large offset,  $\delta(c_M)$ , coupled with the low folder gain,  $A_f$ , means that the overall input referred standard deviation offset,  $\delta(e_{IN})$ , of the block is dominated by the comparator mismatch. From the available set of zero-crossings the calibration engine selects the most suitable zero-crossing for each code

### 4.3.3 Polarity vectors

In a folding-by-8 folder the output of the folder changes polarity 8 times. The polarity of the zero-crossing selected during calibration has to be determined and is used to encode the ADC output. To reduce the amount of SRAM required in the ADC, *polarity vectors* are used. Fig. 4.6 shows the outputs of the folding by 8 folders.

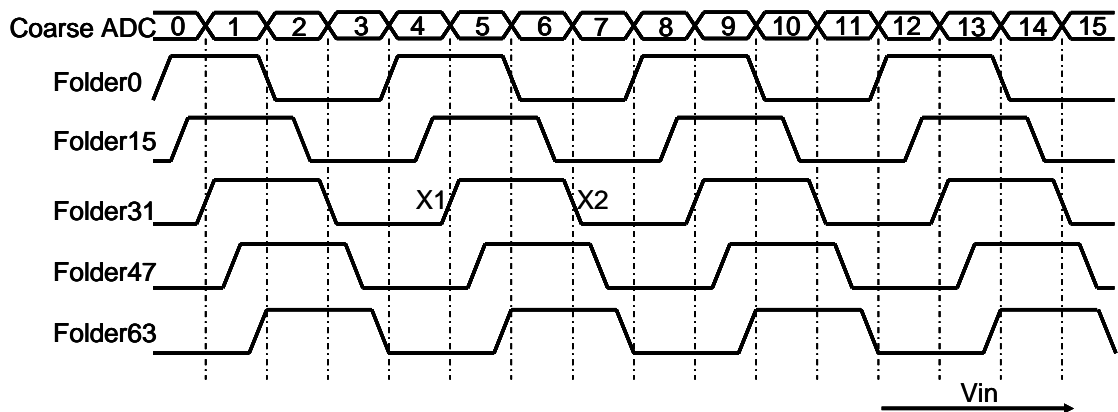


Fig. 4.6. A 4-bit Coarse ADC identifies 16 sections. A selection, (folder 0,15,31,47 and 63) of the total 64 folding by 8 waveform outputs is also shown.

For simplicity only a 5 of the 64 folder outputs are shown, folders 0,15,31,47 and 63. The folder outputs are identical but offset from each other. The ADC conversion range is divided into 16 sections by the coarse ADC. The polarity of a zero-crossing is positive if the folder output goes from low to high as the input voltage is increased, (e.g. *X1*), otherwise the polarity is negative (.e.g. *X2*). Mismatch in the folders causes the zero-crossings to move but providing the standard deviation of the folder zero-crossing offsets is not large the polarity of a zero-crossing for any folder in any coarse ADC section can be predicted.

During calibration a zero-crossing is selected only if it is close to an ideal cross-over value and it has the correct polarity. A zero-crossing is only selected in section 0 if it

has positive polarity. Similarly a zero-crossing is only selected in section 15 if it has negative polarity. In the 8-bit design, described in Chapter 3, a zero-crossing was selected irrespective of polarity. However that required a dedicated zero-crossing memory, the *Polarity Corrector* SRAM. In this design the need for a dedicated memory for polarity is removed.

As the folder folds by 8, and there are 16 coarse ADC sections, a folder nominally places a zero-crossing in every second section. Of these 8 zero-crossings, alternate ones are positive and negative. Therefore the sections repeat in every 4<sup>th</sup> section. For example in Fig. 4.6 section 1 has a positive zero-crossing from folders 31, 47 and 63. Sections 5, 9 and 13 also have positive zero-crossings from folders 31, 47 and 63. All sections repeat, sections 1, 5, 9 and 13 are identical to each other as are sections 2, 6, 10, 14 and 4, 8, 12.

Table 4.1 shows the polarity vectors used. For example, an offset could cause the zero-crossing *XI*, of folder 31, to shift up or down while still remaining in section 4 or 5. From the table we see that a positive edge of folder 31 can be selected in section 4 or 5. However if the causes the zero-crossing to shift into either section 3 or 6, then as a positive zero-crossing on folder 31 it cannot be selected in those sections and *XI* would be ignored. Using polarity vectors requires a slight increase in the matching requirements of the folders however this removes the need for a dedicated polarity SRAM. During the conversion phase the polarity vectors are used to correctly decode the folder outputs.

Coarse ADC section	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Folder 0 to 15	+	-	-	+	+	-	-	+	+	-	-	+	+	-	-	-
Folder 16 to 47	+	+	-	-	+	+	-	-	+	+	-	-	+	+	-	-
Folder 48 to 63	+	+	+	-	-	+	+	-	-	+	+	-	-	+	+	-

Table 4.1 Polarity of zero-crossings



## 4.4 Implementation

A prototype ADC was designed in ST Microelectronics 90nm CMOS. The full chip implementation is shown in Fig. 4.7. There are 64 folders each of which are connected to 4 comparators.

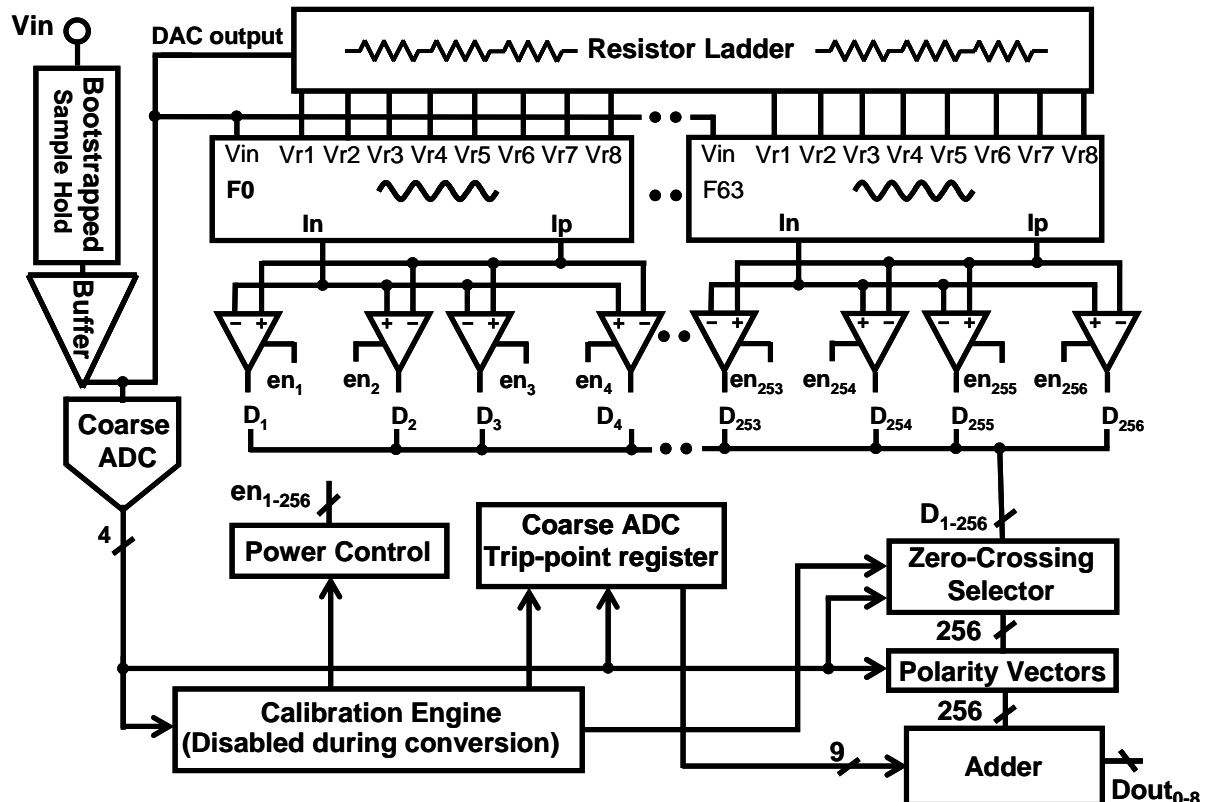


Fig. 4.7. 9-bit folding ADC uses 64 folders and 256 comparators.

During calibration the bootstrapped sample and hold is disabled and the DAC voltage is fed to the folders and to the coarse ADC. The calibration engine loads the Coarse ADC Trip-point Register and the Zero-Crossing Selector memories. In the conversion phase the DAC is turned off and the analog input to be converted is fed into the bootstrapped sample-and-hold. The Power Control block disables the unselected comparators. The comparator outputs are decoded by the Zero-Crossing Selector and

Polarity Vectors blocks and the resultant 256 bit output is summed and added to the output of the Coarse ADC Trip-point Register block to give the overall 9 bit ADC output.

#### 4.4.1 Sampling Circuits

A sampling network and source follower, Fig. 4.8, are used at the chip input to relax the requirements on folder and comparator bandwidths. When  $v_{boot}$ , the signal at the gate of M1, is high, M1 is on, and  $V_{in}$  is sampled onto capacitor  $C_L$ . M2 is a dummy transistor to cancel the charge injection introduced as M1 is turning off. M3 with M4 and M5 form a source follower. The bulk of M3 is connected to its source to remove the body effect. M5 is a current source and adding M4 forms a cascode load. M5 is a high threshold voltage transistor and M4 is a low threshold voltage transistor. The threshold voltage difference between M4 and M5 is large enough to keep both transistors saturated.

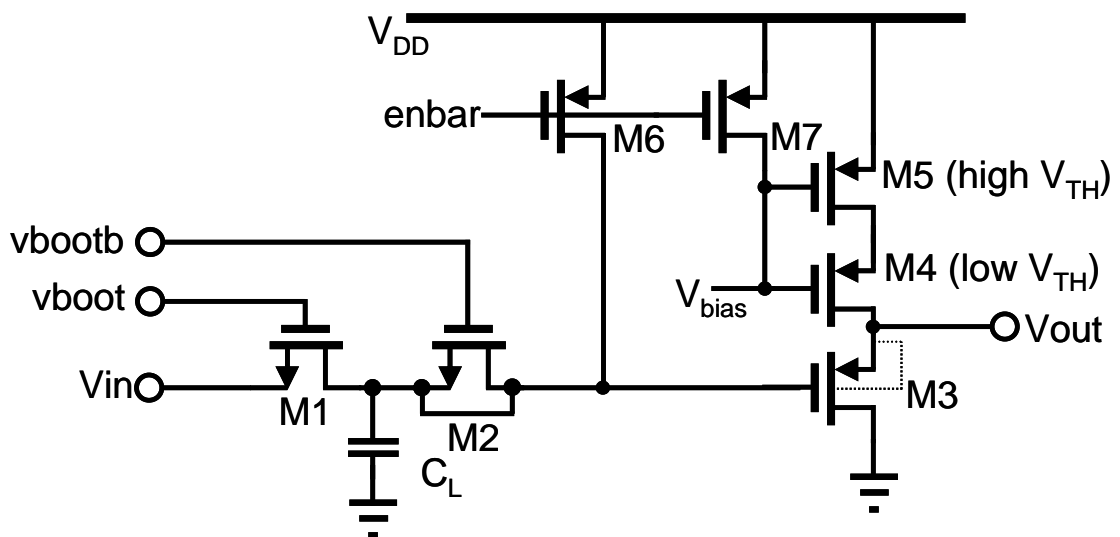


Fig 4.8.  $V_{in}$  is sampled into  $C_L$  and a source follower provides  $V_{out}$ .

In calibration mode the DAC output is connected directly to the output of the source follower,  $V_{out}$ . The relatively low output resistance of the source follower would

pull the resistor-string DAC output low. Adding M6 and M7 allows M3, M4 and M5 to be turned off when *enbar* goes low. This presents a high output resistance to the output of the DAC.

The signal *vboot* is a bootstrapped signal. Bootstrapping is necessary when the input *Vin* approaches the supply voltage,  $V_{DD}$ . To demonstrate the need for bootstrapping consider the simple sample-and-hold, consisting of a single transistor M1 and a load capacitor  $C_L$ , Fig. 4.9(a). Here *clk* is a digital signal with two levels ground and  $V_{DD}$ . When *clk* is high M1 is on and the capacitor  $C_L$  begins to charge up to *Vin*. As the drain of the transistor approaches *Vin* the voltage drop across the transistor decreases, and M1 enters the linear region.

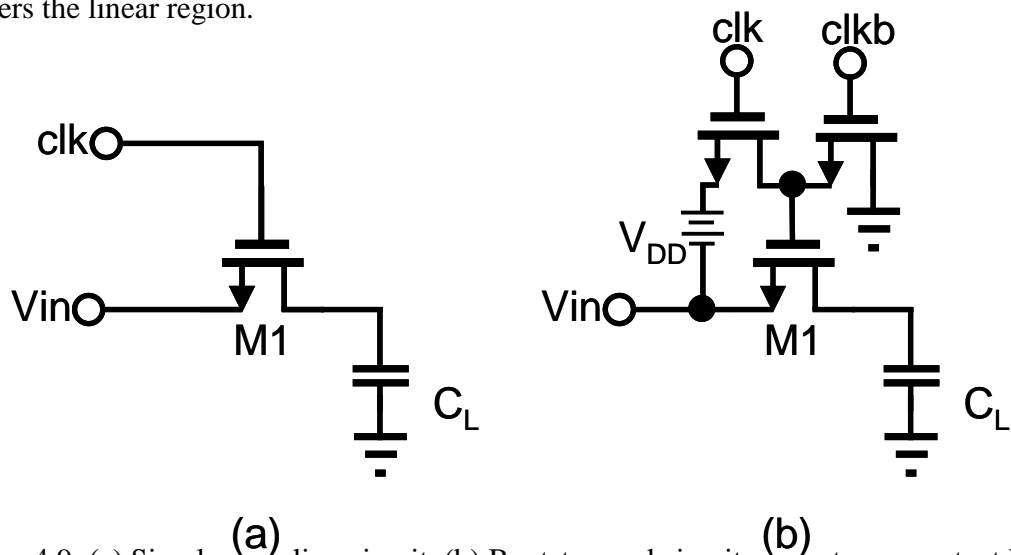


Fig. 4.9. (a) Simple sampling circuit. (b) Bootstrapped circuit generates a constant  $V_{GS}$ .

The on resistance of M1 in the linear region is given in Eqn. 4.2.

$$R_{ON} = \frac{1}{\mu_n C_{OX} \frac{W}{L} (V_{GS} - V_{TH})} \quad (4.2)$$

$V_{GS}$ , the difference between the gate and source potential is  $V_{DD} - V_{in}$ . When *Vin* is large,  $V_{GS}$  is small and the on resistance of M1 is large giving a large RC time constant

and the load capacitor does not charge completely to  $V_{in}$ . For constant low on resistance the  $V_{GS}$  has to be kept constant irrespective of the value of  $V_{in}$ . A bootstrapping circuit can perform this function. Fig. 4.9(b) shows how bootstrapping works. When  $clk$  is high the gate of M1 is at  $V_{in}+V_{DD}$  generating a constant  $V_{GS}$  of  $V_{DD}$ . When  $clk$  is low,  $clkb$  is high and the gate of M1 is pulled low.

Fig. 4.10 shows the complete bootstrapping circuit used in the design. This circuit charges  $C_{boot}$  to  $V_{DD}$ . To be able to turn M3 on and off, with its source at  $V_{DD}$ , the gate of M3 has to switch between  $V_{DD}+V_{GS}$  and  $V_{DD}$ . The signal  $ch$  is a clock signal. Initially when  $ch$  is low and without any charge on the capacitors, the gate of M2 is at ground, and M1 turns on charging  $C_{c1}$ . When  $ch$  goes high the gate of M1 is low, M2 turns on and  $C_{c2}$  charges.  $C_{c1}$  and  $C_{c2}$  eventually charge to  $V_{DD}$ . With  $C_{c1}$  charged to  $V_{DD}$ , and the signal  $ch$  switching between ground and  $V_{DD}$ , the gate of M3 switches between  $V_{DD}$  and  $2V_{DD}$  turning M3 on and off. When M3 is on M4 is also on. The bottom plate of  $C_{boot}$  is connected to ground and the top plate of  $C_{boot}$  is connected to  $V_{DD}$ .

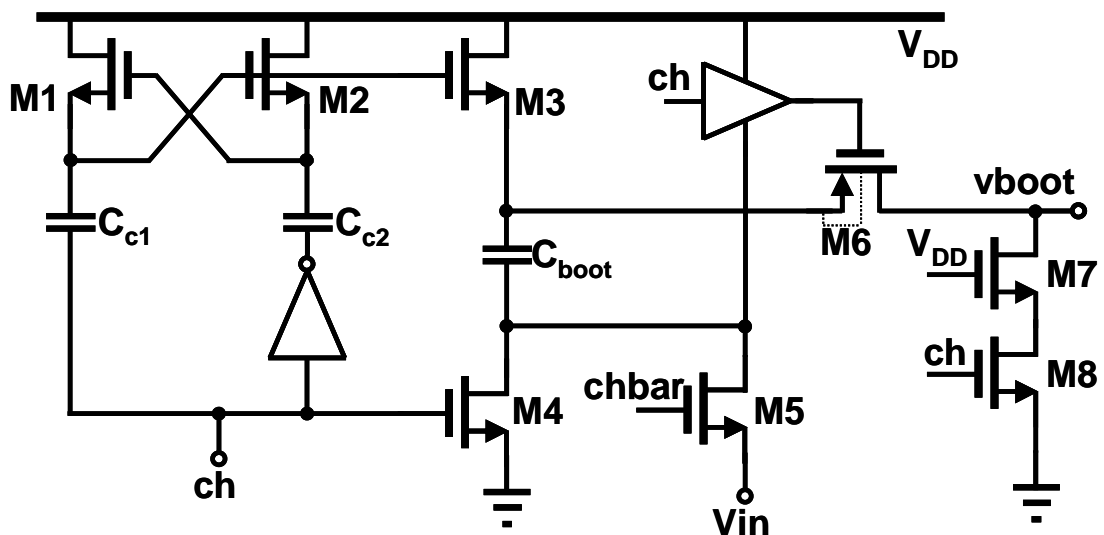


Fig. 4.10. Bootstrapping Circuit.

When  $ch$  is low,  $chbar$  is high M5 is on and  $Vin$  is connected to the bottom plate of  $C_{boot}$ . The output of U2 turns on M6 and the output of the bootstrapping circuit  $vboot$  is  $V_{DD}$  above  $Vin$ .

When  $ch$  is high  $vboot$  is pulled low by M8. M7 is always on and reduces the  $V_{DS}$  drop across M7. The bulk of M6 is connected to the top plate of  $C_{boot}$  to prevent the bulk-source diode of M6 from becoming reverse biased. To avoid overstressing the NMOS transistor in U2 its bulk is tied to the bottom plate of  $C_{boot}$ .

The output waveforms are shown in Fig. 4.11. When  $ch$  is low, the output,  $vboot$ , is low. For high  $ch$ ,  $vboot$  is  $V_{DD}$  above  $Vin$ .

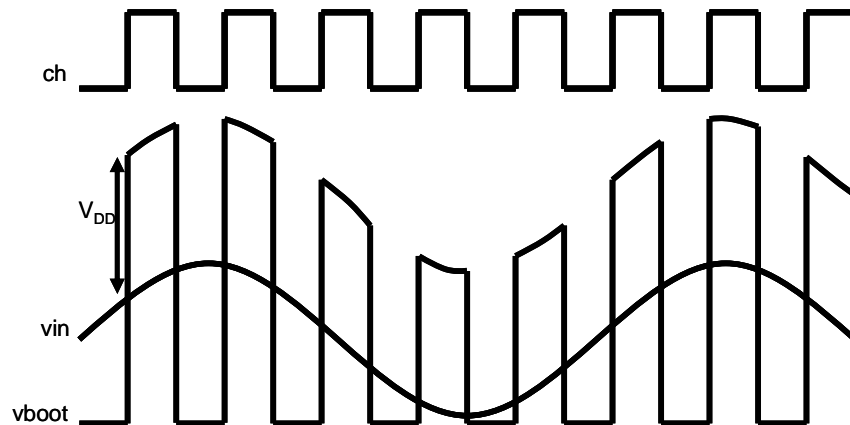


Fig. 4.11. Output  $vboot$  is  $V_{DD}$  above  $vin$  when  $ch$  is high.

## 4.5 Comparator matching and yield

A comparator redundancy of 4 is used to ensure high yield. Similar to section 4.3 in Chapter 3, large comparators with good matching do not provide unique zero-crossings and the yield would be low. To ensure unique zero-crossings small transistors, introducing random mismatch, are used in the comparator design.

Monte Carlo simulations were run in Matlab to determine the optimum comparator mismatch. The stacked folder gain is approximately 4. The gain through the entire folder, including the parallel folder is 8. Assuming that the standard deviation of

the folder mismatch is dominated by the differential pair sizing, then the folder mismatch of each stage can be calculated from the mismatch of the differential pairs using Eqn 4.3;

$$\partial(dp) = \frac{\sqrt{2}A_{VT}}{\sqrt{WL}} \quad (4.3)$$

$A_{VT}$  is a process parameter and for the process geometries of interest for this work is approximately  $5mV\mu m$ [57].  $W$  and  $L$  for the stacked folder was given in section 2.2 and the input transistors used in the parallel folder have a width of  $5\mu m$  and a length of  $0.1\mu m$ .

In Fig 4.12 the simulated yield, of calibrated ADCs, is plotted against standard deviation comparator offset. A good ADC is defined as one with an ENOB of over 8.4 bits. For low comparator mismatch the yield is very low. As mismatch increases yield approaches 100%.

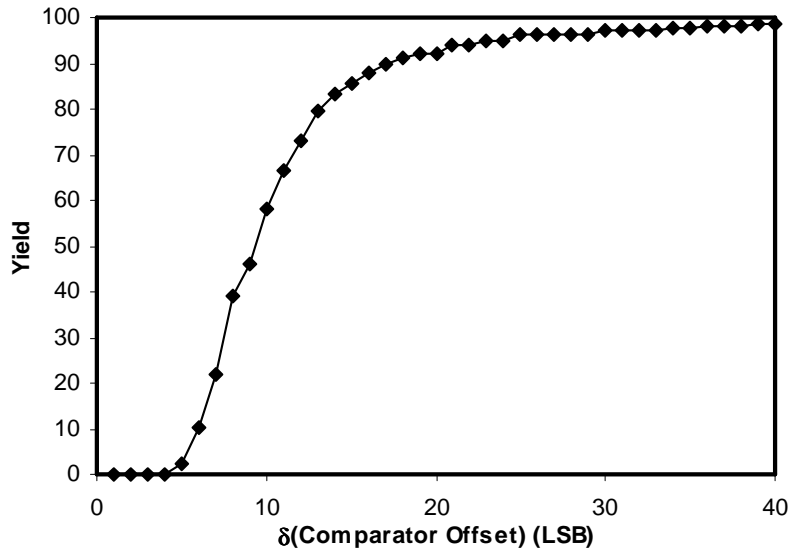


Fig. 4.12. Yield versus standard deviation comparator offset.

Fig. 4.13 shows the comparator [58] used in the 9 bit design. The inputs  $V_A$  and  $V_B$  are the outputs of the folder. The input differential pair is protected from the latch by

a cascode [59]. Transistors MP1 and MP2 form a PMOS latch. When the comparator is in tracking mode  $\Phi_1$  is low and the latch is reset. When  $\Phi_1$  goes high the latch enters a regenerative phase and any difference between the voltages at nodes C and D is amplified. When  $\Phi_2$  goes low the NMOS latch, MN1 and MN2, amplifies the difference further.

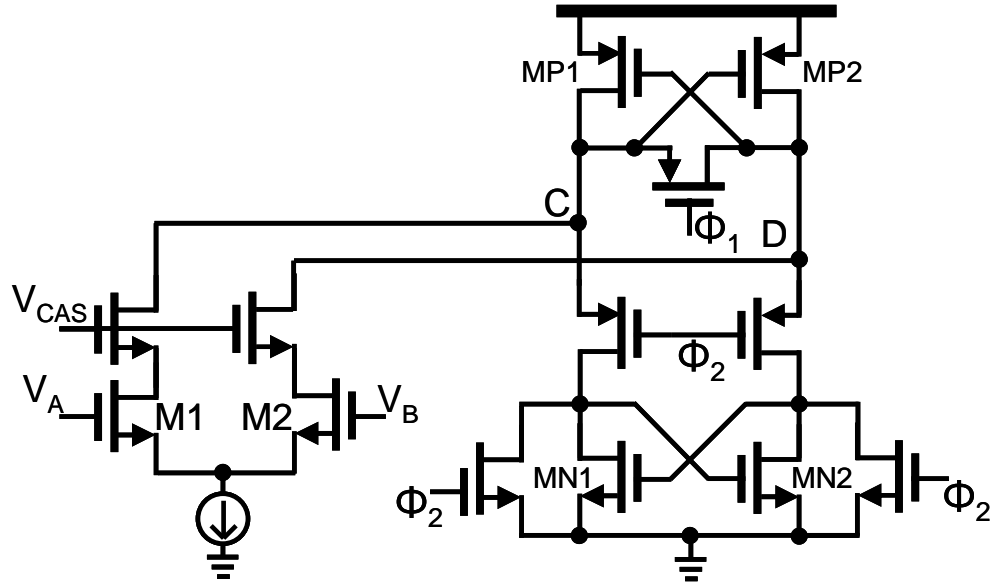


Fig. 4.13. Comparator mismatch dominated by M1, M2 and MP1, MP2 matching.

The input referred mismatch of the latch is primarily from the mismatch of MP1 and MP2. The overall mismatch of the comparator is given by Eqn. (4.4).

$$(e_{IN}) = \sqrt{\delta(dp)^2 + \frac{\delta(MP)^2}{dp_g}} \quad (4.4)$$

$\delta(dp)$  is the differential pair mismatch and  $\delta(MP)$  is the mismatch of the MP1, MP2 latch.  $dp_g$  is the differential pair gain; the gain from the comparator input to nodes C and D. M1 and M2, the input transistors, and MP1, MP2, the latch transistors, have a width of 0.6 $\mu$ m and a length of 0.1 $\mu$ m. The gain of the differential pair is approximately 2. Combining Eqn. 4.3 and Eqn. 4.4 gives a comparator input referred offset of about 32mV. For a 9-bit ADC with a 600mV input range, the LSB size is about 1.2mV

meaning that the comparator mismatch is about 26LSB. From Fig. 4.12 a standard deviation of comparator offset of 26LSB gives a yield of about 96%. The low matching requirements allows for small comparators so that the overall area of the comparator is only  $60\mu\text{m}^2$ .

## 4.6 Prototype

A prototype ADC was fabricated in ST Microelectronics 90nm CMOS, a 7 layer process. The ADC layout is shown in Fig. 4.14. The total area is only  $0.4\text{mm}^2$ , of which the analog area, comparators, folders and sample and hold is less than  $0.1\text{mm}^2$ .

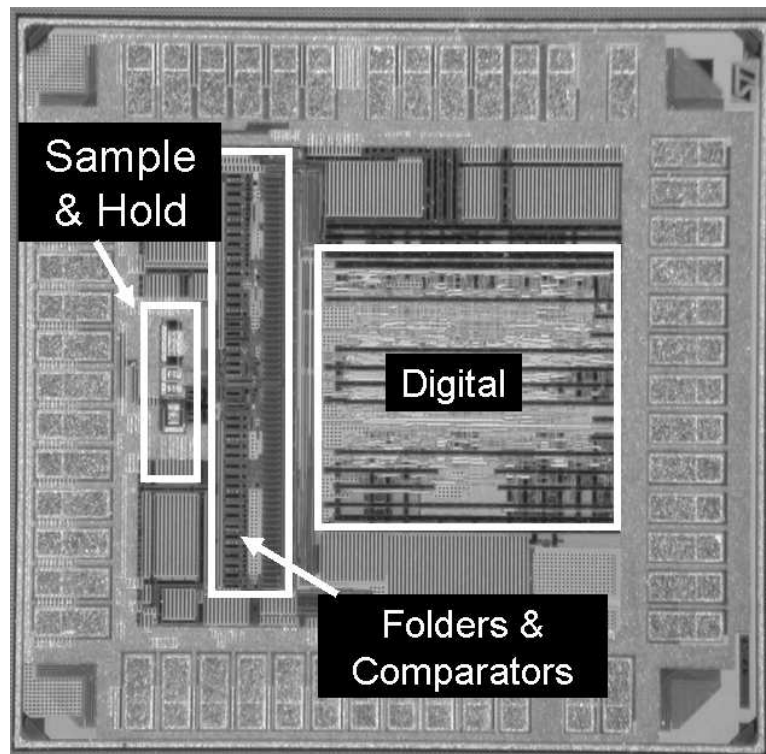


Fig. 4.14. Die micrograph of 9-bit folding ADC prototype.

The simulated ADC performance is summarized in Table 4.2. The total number of comparators enabled after calibration depends on the selected zero-crossings. A Matlab Monte Carlo model was run to determine the average number of comparators selected,



Over 100 calibrations the average was 181. All folders in this design are always enabled. The figure of merit calculated from these Matlab simulations and Spectre power simulations is approximately 0.8pJ/conv and compares favorably with the Figure of Merit for the surveyed ADCs plotted in Fig. 1.4.

Power Consumption @ 400MS/s	Sampling Circuits	6mW
	64 Folders	14mW
	181 Comparators	28mW
	Digital Circuits	65mW
ENOB		8.35
ERBW		181MHz

Table 4.2 Simulated 9-bit ADC performance

## Chapter 5

### Conclusion and suggestions for future work

#### 5.1 Conclusion

A new digital calibration technique for folding ADCs has been developed. The calibration algorithm removes the need for large transistors as critical analog components. This greatly simplifies analog design and is also robust to variations in die fabrication that may otherwise reduce ADC accuracy. The analog circuitry is simplified at the expense of digital complexity. By doing so, the overall performance improves with processing speed and digital density, thus, aligning precisely with the benefits of CMOS scaling. Therefore, as technology shrinks, these circuits will scale accordingly, port easily, and will not suffer performance degradation.

Two different ADCs were described, an 8-bit 500MS/s ADC in TSMC 0.18 $\mu$ m and another ADC, a 9-bit 500MS/s in ST Microelectronics 90nm. Both of these designs are predominantly digital, the analog area of the designs is relatively small. The designs have very low matching requirements and also compensate for non-linearities in the key analog blocks.

The matching requirements of the comparators and folders used in the designs were determined by Matlab modeling. Matlab was also used to calculate DAC resolution and the amount of redundancy to achieve an acceptable yield. The first design described, the 8-bit 500MS/s ADC, used a unique digital calibration technique and also had some

novel analog design features including a new method for layout of a resistor ladder DAC and a Sample-and-Hold circuit that did not use a dedicated load capacitor. A new folder technique was developed for the 9-bit folding ADC that significantly reduces power consumption over a traditional folder circuit for a given noise specification.

The new design techniques used in this design, particularly digital calibration and redundancy, will improve with finer processes. The idea of trading analog accuracy for digital techniques decreases matching and linearity requirements of the analog circuitry but increases the digital power consumption. For the processes used in this work, TSMC 0.18 $\mu$ m CMOS and ST Microelectronics 90nm CMOS, the digital power is a significant percentage of the overall power consumption, however for finer geometry processes the energy per transition decreases reducing the digital power consumption.

The area and performance of the digitally calibrated folding ADC architecture is limited by the SRAM area and the thermal noise of the analog circuits. The large amount of SRAM required in the designs consumes a lot of power and area and in future work techniques to reduce this overhead should be considered. Thermal noise in analog circuits reduces ADC accuracy. While a new ‘stacked’ folder has been designed in this work to reduce noise, the analog power consumption is still dictated by noise requirements. In finer geometry processes new analog techniques are required for moderate, 6-8 bit, resolution ADCs. This work reduces the requirement for transistor threshold matching however thermal noise caused by the reduced input conversion range of ADCs will continue to be a significant issue in deep submicron analog circuits.

## **5.2 Suggestions for future research**

Although significant research advances have been made during this research, there are several areas in which further investigation would be useful. A few of these areas are discussed in this section.

### **5.2.1 Reducing the size of on chip memory**

Both prototype designs are predominantly digital, over 70% of total area of both chips is digital. Of that digital area on-chip SRAM accounts for a significant portion. A more efficient design could be used for the *Zero-Crossing Selector* memories, since the SRAM needed is large. The number of words is set by the coarse ADC resolution, e.g. a 4-bit coarse ADC requires 16 words, and the word length is dependent on the number of comparators in the design, so that 256 comparators means a word length of 256 bits. The effect of decreasing the coarse ADC resolution, at the cost of increasing the risk of having more than one zero-crossing per section could be explored. Similarly ‘sharing’ words could be examined. If a certain comparator *CI* has 4 zero-crossings, nominally in section 1, 5, 9 and 13 and comparator *CI29* has 4 zero-crossings, nominally in section 3, 7, 11 and 15 then they could possibly share the same *Zero-Crossing Selector* word. Data compression could also be investigated to reduce SRAM size.

Reducing the SRAM size would not only reduce die area but would also reduce power consumption as the SRAM is read repeatedly during conversion mode.

### **5.2.2 Continuous background calibration**

Both designs described in this work use power-on calibration. The ADC is calibrated once and then enters conversion mode and the calibration engine is turned off. A background calibration could continuously check the selected zero-crossings and compensate for any drift caused by changes in either supply voltage or temperature.

Assuming that a random analog input is applied to the ADC then each code in the ADC should be ‘hit’ the same number of times. If one code is ‘hit’ more often than its neighboring code then the zero-crossing has moved from its calibrated code and a new zero-crossing would need to be selected. New zero-crossings could be repeatedly selected until a more uniform distribution of codes is found.

### **5.2.3 Sample-and-hold distortion**

The digital calibration technique compensates for non-linearity in the folders. The sample-and-hold, with the signal-dependent on-resistance of its switching transistor, is non-linear and adds distortion. In the ADCs designed in this work, the sample-and-hold is not in the calibration loop. The test voltages, generated by the DAC, are applied directly to the folder inputs. Instead, the DAC voltage can be sent through the sample-and-hold and the distortion in the sample-and-hold can be overcome through calibration.

### **5.2.4 Redundancy in other ADC architectures**

This work focuses on digital calibration techniques for folding ADCs but the techniques used could be considered for other architectures. Matching of capacitors is crucial in both pipeline and SAR ADCs. Redundant capacitors could be used and a calibration engine could pick comparators from the available bank.

## References

1. M. Pelgrom, A.C.J. Duinmaijer, and A.P.G. Welbers, "Matching Properties of MOS Transistors", *IEEE J. of Solid-State Circuits*, vol. 24, no. 5, October 1989.
2. R. Poujois, B. Baylac, D. Barbier, J. Ittel, "Low-level MOS transistor amplifier using storage techniques", *ISSCC Dig. Tech. Papers*, 1973
3. S. Park, Y. Palaskas, M.P. Flynn, "A 4-GS/s 4-bit Flash ADC in 0.18 $\mu$ m CMOS", *IEEE J. of Solid-State Circuits*, vol. 42, no. 9, September 2007.
4. Lee Jaesik, P. Roux, Koc Ut-Va, T. Link, Y. Baeyens, Chen Young-Kai, "A 5-b 10-GSamples A/D converter for 10-Gb/s optical receivers", *IEEE J. of Solid-State Circuits*, vol. 39, no. 10, October 2007
5. B. Ravazi, Principles of Data Conversion Systems Design, IEEE Press 1st ed., 1995
6. K. Bult, A. Buchwald, J. Laskowski, "A 170 mW 10 b 50 Msample/s CMOS ADC in 1mm<sup>2</sup>", *ISSCC Dig. Tech. Papers*, 1997
7. S. Tsukamoto, T. Endo, W.G. Schofield, "A CMOS 6b 400 M sample/s ADC with error correction", *ISSCC Dig. Tech. Papers*, 1998
8. Y. Tamba, K. Yamakido, "A CMOS 6 b 500 MSample/s ADC for a hard disk drive read channel", *ISSCC Dig. Tech. Papers*, 1999
9. Kwangho Yoon; Sungkyung Park; Wonchan Kim, "A 6b 500 MSample/s CMOS flash ADC with a background interpolated auto-zeroing technique", *ISSCC Dig. Tech. Papers*, 1999
10. K. Nagaraj, D.A. Martin, M. Wolfe, R. Chattopadhyay, S. Pavan, J. Cancio, T.R. Viswanathan, "A 700M Sample/s 6 b read channel A/D converter with 7 b servo mode", *ISSCC Dig. Tech. Papers*, 2000
11. K. Sushihara, H. Kimura, Y. Okamoto, K. Nishimura, A. Matsuzawa, "A 6 b 800 MSample/s CMOS A/D converter", *ISSCC Dig. Tech. Papers*, 2000
12. K. Uyttenhove, A. Marques, M. Steyaert, "A 6-bit 1 GHz acquisition speed CMOS flash ADC with digital error correction", *CICC Dig. Tech. Papers*, 2000
13. M. Choi, A.A. Abidi, "A 6 b 1.3 GSAMPLE/s A/D converter in 0.35  $\mu$ m CMOS", *ISSCC Dig. Tech. Papers*, 2001
14. G. A. Geelen, "6 b 1.1 GSAMPLE/s CMOS A/D converter", *ISSCC Dig. Tech. Papers*, 2001
15. C. Donovan, M. Flynn, "A "digital" 6-bit ADC in 0.25- $\mu$ m CMOS", *CICC Dig. Tech. Papers*, 2001
16. B. Yu, W.C. Black, "A 900MS/s 6b Interleaved CMOS Flash ADC", *CICC Dig. Tech. Papers*, 2001
17. P. Scholtens, M. Vertregt, "A 6b 1.6 Gsample/s flash ADC in 0.18  $\mu$ m CMOS using averaging termination", *ISSCC Dig. Tech. Papers*, 2002
18. K. Ono, H. Shimizu, J. Ogawa, M. Takeda, M. Yano, "A 6bit 400Msps 70mW ADC using interpolated parallel scheme", in *Symp. VLSI Circuits Dig*, 2002
19. X. Jiang, Z. Wang, M. F. Chang, "A 2GS/s 6b ADC in 0.18 $\mu$ m CMOS", *ISSCC Dig. Tech. Papers*, 2003
20. M. Flynn, B. Sheahan, "A 400 Msample/s 6b CMOS folding and interpolating ADC", *ISSCC Dig. Tech. Papers*, 1998
21. Song Bang-Sup, Choe Myung-Jun, P. Rakers, S. Gillig, "A 1V 6b 50 MHz current-interpolating CMOS ADC", in *Symp. VLSI Circuits Dig*, 1999

22. Yoon Kwangho, Lee Jeongho, Jeong Deog-Kyoon, Kim Wonchan, "An 8-bit 125 MS/s CMOS folding ADC for Gigabit Ethernet LSI", in *Symp. VLSI Circuits Dig*, 2000
23. A.S. Blum, B.H. Engl, H.P. Eichfeld, R. Hagelauer, A.A. Abidi, "A 1.2V 10-b 100-MSamples A/D converter in 0.12um CMOS", in *Symp. VLSI Circuits Dig*, 2002
24. G. Geelen, E. Paulus, "An 8b 600MS/s 200mW CMOS folding A/D converter using an amplifier preset technique", *ISSCC Dig. Tech. Papers*, 2004
25. Wang Zheng-Yu, Pan Hui Chang Chung-Ming, Yu Hai-Rong, M.F. Chang, "A 600 MSPS 8-bit folding ADC in 0.18 um CMOS", in *Symp. VLSI Circuits Dig*, 2004
26. J. Mulder, C.M. Ward, Lin Chi-Hung, D. Kruse, J.R. Westra, M.L. Lugthart, E. Arslan, R. van de Plassche, K. Bult, "A 21mW 8b 125MS/s ADC occupying 0.09mm<sup>2</sup> in 0.13um CMOS", *ISSCC Dig. Tech. Papers*, 2004
27. D. Huber, R. Chandler, A. Abidi, "A 10b 160MS/s 84mW 1V Subranging ADC in 90nm CMOS", *ISSCC Dig. Tech. Papers*, 2007
28. R. Taft, C. Menkus, M.R. Tursi, O. Hidri, V. Pons, "A 1.8V 1.6GS/s 8b self-calibrating folding ADC with 7.26 ENOB at Nyquist frequency", *ISSCC Dig. Tech. Papers*, 2004
29. S. Limotyakis, S.D. Kulchyski, D. Su, B..A. Wooley, "A 150MS/s 8b 71mW time-interleaved ADC in 0.18um CMOS", *ISSCC Dig. Tech. Papers*, 2004
30. B. Vaz, J. Goes, N. Paulino, "A 1.5-V 10-b 50 MS/s time-interleaved switched-opamp pipeline CMOS ADC with high energy efficiency", in *Symp. VLSI Circuits Dig*, 2004
31. D. Draxelmayr, "A 6b 600MHz 10mW ADC array in digital 90nm CMOS", *ISSCC Dig. Tech. Papers*, 2004
32. Choe Myung-Jun, Song Bang-Sup, K. Bacrania, "An 8 b 100 MSample/s CMOS pipelined folding ADC", in *Symp. VLSI Circuits Dig*, 1999
33. Ming Jun, S.H. Lewis, "An 8b 80MSample/s pipelined ADC with background calibration", *ISSCC Dig. Tech. Papers*, 2000
34. Park Yong-In, S. Karthikeyan, F. Tsay, E. Bartolome, "A 10 b 100 MSample/s CMOS pipelined ADC with 1.8 V power supply", *ISSCC Dig. Tech. Papers*, 2001
35. T. Sigenobu, M. Ito, T. Miki, "An 8-bit 30 MS/s 18 mW ADC with 1.8 V single power supply", in *Symp. VLSI Circuits Dig*, 2001
36. Yoo Sang-Min, Park Jong-Bum, Yang Hee-Suk, Bae Hyuen-Hee, Moon Kyoung-Ho, Park Ho-Jin, Lee Seung-Hoon, Kim Jae-Hwui, "A 10 b 150 MS/s 123 mW 0.18 um CMOS pipelined ADC", *ISSCC Dig. Tech. Papers*, 2003
37. Min Byung-Moo; P. Kim, D. Boisvert, A. Aude, "A 69 mW 10 b 80 MS/s pipelined CMOS ADC", *ISSCC Dig. Tech. Papers*, 2003, Page(s): 324- 325 vol.1
38. B. Hernes, A. Briskemyr, T.N. Andersen, F. Telste, T.E. Bonnerud, O. Moldsvor, "A 1.2V 220MS/s 10b pipeline ADC implemented in 0.13um digital CMOS", *ISSCC Dig. Tech. Papers*, 2004
39. I. Ahmed, "A 500MS/s (35mW) to 1 kS/s (15uW) Power Scalable 10b Pipelined ADC with Minimal Bias Current Variation", *ISSCC Dig. Tech. Papers*, 2005
40. Y. Masato, K. Masairo, G. Kuniho, W. Yuu, "A 10b 125Ms/s 40mW Pipelined ADC in 0.18um CMOS", *ISSCC Dig. Tech. Papers*, 2005

41. K. Hwi-Cheol, J. Deog-Koon, K. Wonchan Kim, "A 30mW 8b 200Ms/s Pipelined CMOS ADC Using a Switched-Opamp Technique", *ISSCC Dig. Tech. Papers*, 2005
42. R. Seung-Tak, S. Bang-Sup, B. Kanti, "A 10b 50MS/s Pipelined ADC with Opamp Current Reuse", *ISSCC Dig. Tech. Papers*, 2006
43. G. Geelen, E. Paulus, D. Simanjuntak, H. Pastoor, R. Verlinden, "A 90nm CMOS 1.2V 10b power and speed programmable pipelined ADC with 0.5pJ/conversion-step", *ISSCC Dig. Tech. Papers*, 2006
44. Ding-Lan Shen and Tai-Cheng Lee, "A 6-bit 800-MS/s Pipelined A/D Converter with Open-loop Amplifiers", in *Symp. VLSI Circuits Dig*, 2006
45. M. Yoshioka, M. Kudo, T. Mori, S. Tsukamoto, "A 0.8V 10b 80MS/s 6.5mW Pipelined ADC with Regulated Overdrive Voltage Biasing", *ISSCC Dig. Tech. Papers*, 2007
46. Y-D. Jeon, S-C. Lee, K-D. Kim, J-K. Kwon, J. Kim, "A 4.7mW 0.32mm<sup>2</sup> 10b 30MS/s Pipelined ADC without a Front-End S/H in 90nm CMOS", *ISSCC Dig. Tech. Papers*, 2007
47. S-C. Lee, Y-D. Jeon, K-D. Kim, J-K. Kwon, J. Kim, J-W. Moon, W. Lee, "A 10b 205MS/s 1mm<sup>2</sup> 90nm CMOS Pipeline ADC for Flat-Panel Display Applications", *ISSCC Dig. Tech. Papers*, 2007
48. L. Brooks, H-S. Lee, "A Zero-Crossing-Based 8b 200MS/s Pipelined ADC", *ISSCC Dig. Tech. Papers*, 2007
49. B. Hernes, J. Bjørnsen, T. Andersen, A. Vinje, H. Korsvoll, F. Telstø, A. Briskemyr, C. Holdø, Ø. Moldsvor, "A 92.5mW 205MS/s 10b Pipelined IF ADC Implemented in 1.2V/3.3V 0.13µm CMOS", *ISSCC Dig. Tech. Papers*, 2007
50. Intel Processor History,  
<http://download.intel.com/pressroom/kits/IntelProcessorHistory.pdf>
51. K. Bult, A Buchwald, "An embedded 240-mW 10-b 50-MS/s CMOS ADC in 1-mm", *IEEE J. Solid-State Circuits*, Vol. 32, Issue 12, Dec. 1997, pp.1887 – 1895.
52. R.C. Taft, C.A. Menkus, M.R. Tursi, O. Hidri and V. Pons, "A 1.8-V 1.6-GSample/s 8-b self-calibrating folding ADC with 7.26 ENOB at Nyquist frequency", *IEEE J. Solid-State Circuits*, Vol. 39, Issue 12, Dec. 2004, pp.2107 – 2115.
53. 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, Issue 12, pp. 2040 – 2050, Dec 2003.
54. C. Donovan and M.P. Flynn, "A 'digital' 6-bit ADC in 0.25µm CMOS", *IEEE J. Solid-State Circuits*, vol. 37, no. 3, pp. 432-437, March 2002.
55. A.G.W. Venes, R.J van-de-Plassche, "An 80-MHz, 80-mW, 8-b CMOS folding A/D converter with distributed track-and-hold preprocessing", *IEEE J. Solid-State Circuits*, Vol. 31, no. 12, pp. 1846-1853, December 1996.
56. R.H. Walden, "Analog-to-digital converter survey and analysis", *IEEE J. Selected Areas in Communications*, vol. 17, no. 4, pp. 539-550, April 1999.
57. R. Difrenza, P. Llinares, G. Morin, E. Granger, G. Ghibaudo, "A New Model for Threshold Voltage Mismatch Based on the Random Fluctuations of Dopant Number in the MOS Transistor Gate", *Proc. of Solid-State Device Research Conference*, September 2001.



58. G.M. Yin, F.O. Eynde, W. Sansen, "A high-speed CMOS comparator with 8-b resolution", *IEEE J. Solid-State Circuits*, Vol. 27, no. 2, pp. 208-211, Feb. 1992.
59. R. Price, "The cascode as a low noise audio amplifier", *Audio, Transactions of the IRE Professional Group on* Vol. 2, no. 2, Part 1, Mar 1954 pp.60 - 64